

An Empirical Evaluation of Similarity Measures for Time Series Classification

Joan Serrà, Josep Ll. Arcos

*Artificial Intelligence Research Institute (IIA-CSIC),
Spanish National Research Council,
08193 Bellaterra, Barcelona, Spain.*

Abstract

Time series are ubiquitous, and a measure to assess their similarity is a core part of many computational systems. In particular, the similarity measure is the most essential ingredient of time series clustering and classification systems. Because of this importance, countless approaches to estimate time series similarity have been proposed. However, there is a lack of comparative studies using empirical, rigorous, quantitative, and large-scale assessment strategies. In this article, we provide an extensive evaluation of similarity measures for time series classification following the aforementioned principles. We consider 7 different measures coming from alternative measure ‘families’, and 45 publicly-available time series data sets coming from a wide variety of scientific domains. We focus on out-of-sample classification accuracy, but in-sample accuracies and parameter choices are also discussed. Our work is based on rigorous evaluation methodologies and includes the use of powerful statistical significance tests to derive meaningful conclusions. The obtained results show the equivalence, in terms of accuracy, of a number of measures, but with one single candidate outperforming the rest. Such findings, together with the followed methodology, invite researchers on the field to adopt a more consistent evaluation criteria and a more informed decision regarding the baseline measures to which new developments should be compared.

Keywords: Time Series, Similarity, Classification, Evaluation

1. Introduction

2 Data in the form of time series pervades a large number of scientific do-
3 mains (Keogh, 2011; Keogh et al., 2011). Observations that unfold over time

4 usually represent valuable information subject to analysis, classification, in-
5 dexing, prediction, or interpretation (Kantz and Schreiber, 2004; Han and
6 Kamber, 2005; Liao, 2005; Fu, 2011). Real-world examples include finan-
7 cial data (e.g., stock market fluctuations), medical data (e.g., electrocardio-
8 grams), computer data (e.g., log sequences), or motion data (e.g., location
9 of moving objects). Even object shapes or handwriting can be effectively
10 transformed into time series, facilitating their analysis and retrieval (Keogh
11 et al., 2009, 2011).

12 A core issue when dealing with time series is determining their pair-
13 wise similarity, i.e., the degree to which a given time series resembles an-
14 other. In fact, a time series similarity (or dissimilarity) measure is central to
15 many mining, retrieval, clustering, and classification tasks (Han and Kam-
16 ber, 2005; Liao, 2005; Fu, 2011; Keogh and Kasetty, 2003). Furthermore,
17 there is evidence that simple approaches to such tasks exploiting generic
18 time series similarity measures usually outperform more elaborate, some-
19 times specifically-targeted strategies. This is the case, for instance, with
20 time series classification, where a one-nearest neighbor approach using a
21 well-known time series similarity measure was found to outperform an ex-
22 haustive list of alternatives (Xi et al., 2006), including decision trees, multi-
23 scale histograms, multi-layer perceptron neural networks, order logic rules
24 with boosting, or multiple classifier systems.

25 Deriving a measure that correctly reflects time series similarities is not
26 straightforward. Apart from dealing with high dimensionality (time series
27 can be roughly considered as multi-dimensional data), the calculation of
28 such measures needs to be fast and efficient (Keogh and Kasetty, 2003).
29 Indeed, with better information gathering tools, the size of time series data
30 sets may continue to increase in the future. Moreover, there is the need
31 for generic/multi-purpose similarity measures, so that they can be readily
32 applied to any data set, whether this application is the final goal or just an
33 initial approach to a given task. This last aspect highlights another desirable
34 quality for time series similarity measures: their robustness to different types
35 of data (cf. Keogh and Kasetty, 2003; Wang et al., 2012).

36 Over the years, several time series similarity measures have been pro-
37 posed (for pointers to such measures see, e.g., Liao, 2005; Fu, 2011; Wang
38 et al., 2012). Nevertheless, few quantitative comparisons have been made in
39 order to evaluate their efficacy in a multiple-data framework. Apart from be-
40 ing an interesting and important task by itself, and as opposed to clustering,
41 time series classification offers the possibility to straightforwardly assess the
42 merit of time series similarity measures under a controlled, objective, and
43 quantitative framework (Keogh and Kasetty, 2003).

44 In a recent study, Wang et al. (2012) perform an extensive comparison of
45 classification accuracies for 9 measures (plus 4 variants) across 38 data sets
46 coming from various scientific domains. One of the main conclusions of the
47 study is that, even though the newly proposed measures can be theoretically
48 attractive, the efficacy of some common and well-established measures is,
49 in the vast majority of cases, very difficult to beat. Specifically, dynamic
50 time warping (DTW; Berndt and Clifford, 1994) is found to be consistently
51 superior to the other studied measures (or, at worst, for a few data sets,
52 equivalent). In addition, the authors emphasize that the Euclidean distance
53 remains a quite accurate, robust, simple, and efficient way of measuring the
54 similarity between two time series. Finally, by looking in detail at the results
55 presented by Wang et al. (2012), we can spot a group of time series similarity
56 measures that seems to have an efficacy comparable to DTW: those based
57 on edit distances. In particular, the edit distance for real sequences (EDR;
58 Chen et al., 2005) seems to be very competitive, if not slightly better than
59 DTW. Interestingly, none of the three measures above was initially targeted
60 to generic time series data, but were introduced with hindsight (Agrawal
61 et al., 1993; Berndt and Clifford, 1994; Chen et al., 2005). The intuition
62 behind Euclidean distance relates to spatial proximity, DTW was initially
63 devised for the specific task of spoken word recognition (Sakoe and Chiba,
64 1978), and edit distances were introduced for measuring the dissimilarity
65 between two strings (Levenshtein, 1966).

66 The study by Wang et al. (2012) is, to the best of our knowledge, the
67 only comparative study dealing with time series classification using multiple
68 similarity measures and a large collection of data. In general, the studies
69 introducing a new measure only compare against a few other measures¹,
70 and usually using a reduced data set corpus (cf. Keogh and Kasetty, 2003).
71 Furthermore, there is a lack of agreement in the literature regarding evalu-
72 ation methodologies. Besides, statistical significance is usually not studied
73 or, at best, improperly evaluated. This is very inconvenient, as robust eval-
74 uation methodologies and statistical significance are the principal tools by
75 which we can establish, in a formal and rigorous way, differences across the
76 considered measures (Salzberg, 1997; Hollander and Wolfe, 1999; Demšar,
77 2006). In addition, the chosen parameter values for every measure are rarely
78 discussed. All these issues impact the scientific development of the field as
79 one is never sure, e.g., of which measure should be used as a baseline for
80 future developments, or of which parameters are the most sensible choice.

¹In the majority of cases, as our results will show, not the most appropriate ones.

81 In this work, we perform an empirical evaluation of similarity measures
82 for time series classification. We follow the initiative by Wang et al. (2012),
83 and consider a big pool of publicly-available time series data sets (45 in our
84 case). However, instead of additionally focusing on representation meth-
85 ods, computational/storage demands, or more theoretical issues, we here
86 take a pragmatic approach and restrict ourselves to classification accuracy.
87 We believe that this is the most important aspect to be considered in a
88 first stage and that, in contrast to the other aforementioned issues, it is
89 not sufficiently well-covered in the existing literature. As for the consid-
90 ered measures, we decide to include DTW and EDR, as these were found
91 to generally achieve the highest accuracies among all measures compared
92 in Wang et al. (2012). Apart from these two, we choose the Euclidean dis-
93 tance plus 4 different measures not considered in such study, making up to
94 a total of 7. Further important contributions that differentiate the current
95 work from previous studies include (a) an extensive summary and back-
96 ground of the considered measures, with basic formulations, applications,
97 and references, (b) the formalization of a robust evaluation methodology,
98 exploiting standard out-of-sample cross-validation strategies, (c) the use of
99 rigorous statistical significance tests in order to assess the superiority of a
100 given measure, (d) the evaluation of both train and test accuracies, and (e)
101 the assessment of the chosen parameters for each measure and data set.

102 The rest of the paper is organized as follows. Firstly, we provide the
103 background on time series similarity measures, outline some of their appli-
104 cations, and detail their calculation (Sec. 2). Next, we explain the proposed
105 evaluation methodology (Sec. 3). Subsequently, we report the obtained re-
106 sults (Sec. 4). A conclusion section ends the paper (Sec. 5).

107 2. Time series similarity measures

108 The list of approaches for dealing with time series similarity is vast, and
109 a comprehensive enumeration of them all is beyond the scope of the present
110 work (for that, the interested reader is referred to Gusfield, 1997; Wang
111 et al., 2012; Han and Kamber, 2005; Liao, 2005; Marteau, 2009; Fu, 2011).
112 In this section, we present several representative examples of different ‘fam-
113 ilies’ of time series similarity measures: lock-step measures (Euclidean dis-
114 tance), feature-based measures (Fourier coefficients), model-based measures
115 (auto-regressive), and elastic measures (DTW, EDR, TWED, and MJC).
116 An effort has been made in selecting the most standard measures of each
117 group, emphasizing the approaches that are reported to have good perfor-
118 mance. We also try to avoid measures with too many parameters, since

119 such parameters may be difficult to learn in small training data sets and,
 120 furthermore, could lead to over-fitting. Alternative measures found to be
 121 consistently less accurate than DTW or EDR are not considered (see Wang
 122 et al., 2012). Apart from all the aforementioned measures, we also include a
 123 random dissimilarity measure, consisting of a uniformly distributed random
 124 number between 0 and 1. This will act as our random baseline, informing
 125 us of the error rates we can expect by chance. By comparing its accuracy to
 126 the one achieved by other measures, it also gives us qualitative information
 127 regarding their ‘usefulness’ or improved capacity for classification.

128 2.1. Euclidean distance

129 The simplest way to estimate the dissimilarity between two time series
 130 is to use any L_n norm such that

$$d_{L_n}(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^M (x_i - y_i)^n \right)^{\frac{1}{n}}, \quad (1)$$

131 where n is a positive integer, M is the length of the time series, and x_i and
 132 y_i are the i -th element of time series \mathbf{x} and \mathbf{y} , respectively. Measures based
 133 on L_n norms correspond to the group of so-called lock-step measures (Wang
 134 et al., 2012), which compare samples that are at exactly the same temporal
 135 location (Fig. 1, top). Notice that in case the time series \mathbf{x} and \mathbf{y} not being of
 136 the same length, one can always re-sample one to the length of the other, an
 137 approach that works well for a number of data sources (Keogh and Kasetty,
 138 2003).

139 Using Eq. 1 with $n = 2$ we obtain the Euclidean distance, one of the
 140 most used time series dissimilarity measures, favored by its computational
 141 simplicity and indexing capabilities. Applications range from early clas-
 142 sification of time series (Xing et al., 2011) to rule discovery in economic,
 143 communications, and ecological time series (Das et al., 1998). Some au-
 144 thors state that the accuracy of the Euclidean distance can be very diffi-
 145 cult to beat, specially for large data sets containing many time series (cf.
 146 Wang et al., 2012). To the best of our knowledge, these claims are only
 147 quantitatively supported by one-nearest neighbor classification experiments
 148 using two artificially-generated/synthetic data sets (Geurts, 2002). We be-
 149 lieve that such claims need to be carefully assessed with extensive experi-
 150 ments and under broader conditions, considering multiple measures, differ-
 151 ent distance-exploiting algorithms, and real-world data sets.

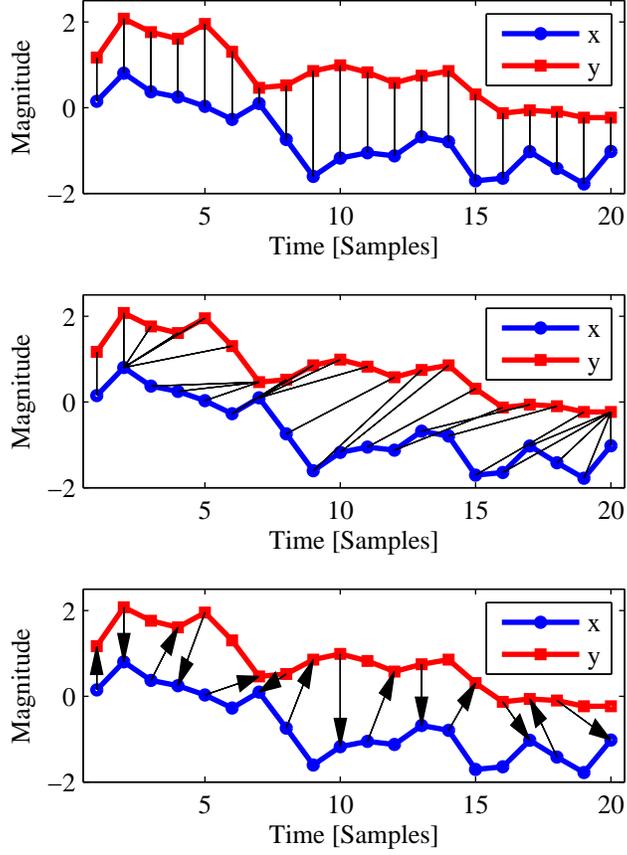


Figure 1: Examples of dissimilarity calculations between time series \mathbf{x} and \mathbf{y} : Euclidean distance (top), DTW alignment (center), and MJC (bottom). See text for details.

152 *2.2. Fourier coefficients*

153 A simple extension of the Euclidean distance is not to compute it directly
 154 using the raw time series, but using features extracted from it. For instance,
 155 by first representing the time series by their Fourier coefficients (FC), one
 156 uses

$$d_{\text{FC}}(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^{\theta} (\hat{x}_i - \hat{y}_i)^2 \right)^{\frac{1}{2}}, \quad (2)$$

157 where \hat{x}_i and \hat{y}_i are complex value pairs denoting the i -th Fourier coefficient
 158 of $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$, the discrete Fourier transforms (DFT) of the raw time series (Op-

penheim et al., 1999). Notice that in Eq. 2 we introduce the parameter θ , the actual number of considered coefficients. Because of the symmetry of the DFT, the sum only needs to be performed, at most, over half of the coefficients, so that $\theta = M/2$. Notice that, by the Parseval theorem (Oppenheim et al., 1999), the Euclidean distance between FCs is equivalent to the standard Euclidean distance between the raw time series (see, e.g., Agrawal et al., 1993). However, having parameter θ , one usually takes the opportunity to filter out high-frequency coefficients, i.e., coefficients \hat{x}_i and \hat{y}_i whose i is close to $M/2$. This has the (sometimes desired) effect of removing rapidly-fluctuating components of the signal. Hence, if high frequencies are not relevant for the intended analysis or we have some high-frequency noise, this operation will usually carry some increase in accuracy. Furthermore, if θ is relatively small, similarity computations can be substantially accelerated.

Computing the Euclidean distance on a reduced set of features is an extremely common approach in literature. FCs are the standard choice for efficient time series retrieval, exploiting the aforementioned acceleration capabilities. Pioneering work includes Agrawal et al. (1993) and Faloutsos et al. (1994) dealing with synthetic and financial data. More recent works use FCs with data from other domains. For instance, the case-based reasoning system of Montani et al. (2006) uses FCs to compare medical time series. Apart from FCs, wavelet coefficients have been extensively used (Chan and Fu, 1999). For instance, Olsson et al. (2004) use a wavelet analysis to remove noise and extract features in their system of fault diagnosis in industrial equipment. Research suggests that, although they provide some advantages, wavelet coefficients do not generally outperform FCs for the considered task (Wu et al., 2000). Comparatively less used time series features are based on singular value decomposition (Wu et al., 1996), piece-wise aggregate approximations (Keogh et al., 2001), or the coefficients of fitted polynomials (Cai and Ng, 2004) among others.

2.3. Auto-regressive models

A further option for computing similarities between time series using features extracted from them is to employ time series models (Liao, 2005; Fu, 2011). The main idea behind model-based measures is to learn a model of the two time series and then use its parameters for computing a similarity value. In the literature, several approaches follow this idea. For instance, Maharaj (2000) uses the p -value of a chi-square statistic to cluster auto-regressive coefficients representing stationary time series. Ramoni et al. (2002) present a Bayesian algorithm for clustering time series. They

198 transform each series into a Markov chain and then cluster similar chains
 199 to discover the most probable set of generating processes. Pavinelli et al.
 200 (2004) use Gaussian mixture models of reconstructed phase spaces to clas-
 201 sify time series of different sources. Serrà et al. (2012a) study the use of the
 202 error of several learned models to identify similar time series corresponding
 203 to musical information.

204 In the present study we consider the use of auto-regressive (AR) models
 205 for time series feature extraction. Given an AR model of the form

$$x_i = a_0 + \sum_{j=1}^{\eta} a_j x_{i-j}, \quad (3)$$

206 where a_j denotes the j -th regression coefficient and η is the order of the
 207 model, we can estimate its coefficients, e.g., by the Yule-Walker function (Marple,
 208 1987). Then, the dissimilarity between two time series can be calculated, for
 209 instance, using the Euclidean distance between their estimated coefficients,
 210 analogously as in Eq. 2 (Piccolo, 1990). The number of AR coefficients is
 211 controlled by the parameter η which, similarly to θ with FCs, directly affects
 212 the final speed of similarity calculations (AR and FCs are usually estimated
 213 offline, prior to similarity calculations).

214 2.4. Dynamic time warping

215 Dynamic time warping (DTW; Sakoe and Chiba, 1978; Berndt and Clif-
 216 ford, 1994) is a classic approach for computing the dissimilarity between two
 217 time series. It has been exploited in countless works: to construct decision
 218 trees (Rodríguez and Alonso, 2004), to retrieve similar shapes from large
 219 image databases (Bartolini et al., 2005), to match incomplete time series
 220 in medical applications (Tormene et al., 2009), to align signatures in an
 221 identity authentication task (Kholmatov and Yanikoglu, 2005), etc. In ad-
 222 dition, several extensions for speeding up its calculations exist (Keogh and
 223 Ratanamahatana, 2005; Salvador and Chan, 2007; Lemire, 2009).

224 DTW belongs to the group of so-called elastic dissimilarity measures (Wang
 225 et al., 2012), and works by optimally aligning (or ‘warping’) the time series
 226 in the temporal domain so that the accumulated cost of this alignment is
 227 minimal (Fig. 1, center). In its canonical form, this accumulated cost can
 228 be obtained by dynamic programming, recursively applying

$$D_{i,j} = f(x_i, y_j) + \min \{D_{i,j-1}, D_{i-1,j}, D_{i-1,j-1}\} \quad (4)$$

229 for $i = 1, \dots, M$ and $j = 1, \dots, N$, being M and N the lengths of time
 230 series \mathbf{x} and \mathbf{y} , respectively. Except for the first cell, which is initialized to

231 $D_{0,0} = 0$, the matrix D is initialized to $D_{i,j} = \infty$ for $i = 0, 1, \dots, M$ and $j =$
 232 $0, 1, \dots, N$. In the case of dealing with uni-dimensional time series, the local
 233 cost function $f()$, also called sample dissimilarity function, is usually taken
 234 to be the square of the difference between x_i and y_j (Berndt and Clifford,
 235 1994), i.e., $f(x_i, y_j) = (x_i - y_j)^2$. In the case of dealing with multidimensional
 236 time series or having some domain-specific knowledge, the local cost function
 237 $f()$ must be chosen appropriately, although the Euclidean distance is often
 238 used. The final DTW dissimilarity measure typically corresponds to the
 239 total accumulated cost, i.e., $d_{\text{DTW}}(\mathbf{x}, \mathbf{y}) = D_{M,N}$. A normalization of d_{DTW}
 240 can be performed on the basis of the alignment of the two time series, which
 241 is found by backtracking from $D_{M,N}$ to $D_{0,0}$ (Rabiner and Juang, 1993).
 242 However, in preliminary analysis we found the normalized variant to be
 243 equivalent, or sensibly less accurate, than the unnormalized one.

244 The canonical form of DTW presented in Eq. 4 can incorporate many
 245 variants. In particular, several constraints can be applied to the computation
 246 of D . A common constraint (Sakoe and Chiba, 1978) is to introduce a
 247 window parameter $\omega \in [0, N]$, such that the recursive formula of Eq. 4 is
 248 only applied for $i = 1, \dots, M$ and

$$j = \max\{1, i' - \omega\}, \dots, \min\{N, i' + \omega\}, \quad (5)$$

249 where i' is progressively adjusted for dealing with different time series lengths,
 250 i.e., $i' = \lfloor iN/M \rfloor$, using $\lfloor \cdot \rfloor$ as the round-to-the-nearest-integer operator.
 251 Notice that if $\omega = 0$ and $N = M$, d_{DTW} will correspond to the squared
 252 Euclidean distance (the value in $D_{M,N}$ will be the sum of the squared differ-
 253 ences, see Eqs. 1 and 4). Notice furthermore that, when $\omega = N$, we are using
 254 the unconstrained version of DTW (the constraints in Eq. 5 have no effect).
 255 Thus, we include two DTW variants in a single formulation. In general, the
 256 introduction of constraints, and specially of the window parameter ω , car-
 257 ries some advantages (Keogh and Kasetty, 2003; Rabiner and Juang, 1993;
 258 Wang et al., 2012). For instance, constraints prevent ‘pathological align-
 259 ments’ and, therefore, usually provide better similarity estimates (patho-
 260 logical alignments typically go beyond the main diagonal of D). Moreover,
 261 constraints allow for reduced computational costs, since only a percentage
 262 of the cells in D needs to be examined (Sakoe and Chiba, 1978; Rabiner and
 263 Juang, 1993).

264 DTW currently stands as the main benchmark against which new sim-
 265 ilarity measures need to be compared (Xi et al., 2006; Wang et al., 2012).
 266 Very few measures have been proposed that systematically outperform DTW
 267 for a number of different data sources. These measures are usually more

268 complex than DTW, sometimes requiring extensive tuning of one or more
 269 parameters. Additionally, it is often the case that no careful, rigorous, and
 270 extensive evaluation of the accuracy of such measures is done, and further
 271 studies fail to assess the statistical significance of their improvement. Thus
 272 we could say that the superiority of such measures is, at best, unclear. In
 273 this paper, we pay special attention to all these aspects in order to for-
 274 mally assess the considered measures under a common framework. As it
 275 will be shown, there exists a similarity measure outperforming DTW for a
 276 statistically significant margin (Sec. 4).

277 2.5. Edit distance on real sequences

278 Turning to previous evidence (Wang et al., 2012), we observe that per-
 279 haps the only measure able to seriously challenge DTW is the edit distance
 280 on real sequences (EDR; Chen et al., 2005). The EDR corresponds to the
 281 extension of the original edit or Levenshtein distance (Levenshtein, 1966)
 282 to real-valued time series. Such extensions are not commonplace, but re-
 283 cent research is starting to focus on them (Morse and Patel, 2007; Marteau,
 284 2009). As noted by Chen et al. (2005), EDR outperformed previous edit
 285 distance variants for time series similarity.

286 The computation of the EDR can be formalized by a dynamic program-
 287 ming approach. Specifically, we compute

$$D_{i,j} = \begin{cases} D_{i-1,j-1} & \text{if } m(x_i, y_j) = 1 \\ 1 + \min \{D_{i,j-1}, D_{i-1,j}, D_{i-1,j-1}\} & \text{if } m(x_i, y_j) = 0, \end{cases} \quad (6)$$

288 for $i = 1, \dots, M$ and $j = 1, \dots, N$. The match function used is

$$m(x_i, y_j) = \Theta(\varepsilon - f(x_i, y_j)), \quad (7)$$

289 where $\Theta()$ is the Heaviside step function such that $\Theta(z) = 1$ if $z \geq 0$ and
 290 0 otherwise, and $\varepsilon \in [0, \infty)$ is a suitably chosen threshold parameter that
 291 controls the degree of resemblance between two time series samples being
 292 considered as a match. The first row of D is initialized to $D_{i,0} = i$ for
 293 $i = 0, 1, \dots, M$ and the first column of D to $D_{0,j} = j$ for $j = 0, 1, \dots, N$.
 294 Following Chen et al. (2005), who initially reported some accuracy improve-
 295 ments of EDR over DTW, we set the local cost function $f()$ to the absolute
 296 difference between the sample values, i.e., $f(x_i, y_j) = |x_i - y_j|$. This has the
 297 additional advantage that we can easily relate ε to the standard deviation
 298 of the time series (Sec. 3.5).

299 *2.6. Time-warped edit distance*

300 The time-warped edit distance (TWED; Marteau, 2009) is perhaps the
 301 most interesting extension of dynamic programming algorithms like DTW
 302 and EDR. In a sense, it is a combination of these two. Like edit dis-
 303 tances, TWED comprises a mismatch penalty λ and, like dynamic time
 304 warping, it introduces a so-called stiffness parameter ν , controlling its ‘elas-
 305 ticity’ (Marteau, 2009). For uniformly-sampled time series, the formulation
 306 of TWED corresponds to

$$D_{i,j} = \min \{D_{i,j} + \Gamma_{\mathbf{xy}}, D_{i-1,j} + \Gamma_{\mathbf{x}}, D_{i,j-1} + \Gamma_{\mathbf{y}}\}, \quad (8)$$

307 for $i = 1, \dots, M$ and $j = 1, \dots, N$, with

$$\begin{aligned} \Gamma_{\mathbf{xy}} &= f(x_i, y_j) + f(x_{i-1}, y_{j-1}) + 2\nu|i - j|, \\ \Gamma_{\mathbf{x}} &= f(x_i, x_{i-1}) + \nu + \lambda, \\ \Gamma_{\mathbf{y}} &= f(y_j, y_{j-1}) + \nu + \lambda, \end{aligned} \quad (9)$$

308 where $f()$ can be any L_n metric (Eq. 1). Following Marteau (2009), and as
 309 done for EDR as well, we choose $f(x_i, y_j) = |x_i - y_j|$. Together with DTW
 310 and EDR, the final dissimilarity value is taken to be $d_{\text{TWED}}(\mathbf{x}, \mathbf{y}) = D_{M,N}$.

311 An interesting aspect of TWED is that, in its original formulation (Marteau,
 312 2009), it takes time stamp differences into account. Therefore, it is able to
 313 cope with time series of different sampling rates, including down-sampled
 314 time series. A further interesting aspect, and contrasting to DTW and other
 315 measures, is that TWED is a metric (Marteau, 2009). Thus, one can exploit
 316 the triangular inequality to speed up the search in the metric space. Finally,
 317 it is worth mentioning that the combination of the two previous characteris-
 318 tics results in a lower bound of the TWED dissimilarity, which can be used
 319 to speed up nearest neighbor retrieval.

320 *2.7. Minimum jump costs dissimilarity*

321 The main idea behind the minimum jump costs dissimilarity measure (MJC;
 322 Serrà and Arcos, 2012) is that, if a given time series \mathbf{x} resembles \mathbf{y} , the cu-
 323 mulative cost of iteratively ‘jumping’ between their samples should be small²
 324 (Fig. 1, bottom). Supposing that for the i -th jump we are at time step t_x

²An implementation of MJC is made available online by the authors: http://www.iiia.csic.es/~jserra/downloads/2012_SerraArcos_MJC-Dissim.tar.gz (last accessed on September 15, 2013).

325 of time series \mathbf{x} , and that we previously visited time step $t_y - 1$ of \mathbf{y} , the
 326 minimum jump cost is expressed as

$$c_{\min}^{(i)} = \min \left\{ c_{t_x}^{t_y}, c_{t_x}^{t_y+1}, c_{t_x}^{t_y+2}, \dots \right\}, \quad (10)$$

327 where $c_{t_x}^{t_y+\Delta}$ is the cost of jumping from x_{t_x} to $y_{t_y+\Delta}$ and $\Delta = 0, 1, 2, \dots$
 328 is an integer time step increment such that $t_y + \Delta \leq N$. After a jump is
 329 made, t_x and t_y are updated accordingly: t_x becomes $t_x + 1$ and t_y becomes
 330 $t_y + \Delta + 1$. In case we want to jump from \mathbf{y} to \mathbf{x} , only t_x and t_y need to be
 331 swapped (Serrà and Arcos, 2012).

332 To define a jump cost $c_{t_x}^{t_y+\Delta}$, the temporal and the magnitude dimensions
 333 of the time series are considered:

$$c_{t_x}^{t_y+\Delta} = (\phi\Delta)^2 + f(x_{t_x}, y_{t_y+\Delta}), \quad (11)$$

334 where ϕ represents the cost of advancing in time and $f()$ is the local cost
 335 function, which we take to be $f(x_{t_x}, y_{t_y+\Delta}) = (x_{t_x} - y_{t_y+\Delta})^2$, similarly to
 336 what is done with DTW (Eq. 4). Notice that, akin to the general formulation
 337 of TWED, the term $(\phi\Delta)^2$ introduces a nonlinear penalty that depends on
 338 the temporal gap. Here, the value of ϕ is set proportional to the standard
 339 deviation σ expected for the time series and, at the same time, proportional
 340 to the real-valued parameter $\beta \in [0, \infty)$, which controls how difficult is to
 341 advance in time (for more details see Serrà and Arcos, 2012). To obtain a
 342 symmetric dissimilarity measure, $d_{\text{MJC}}(\mathbf{x}, \mathbf{y}) = \min \{d_{XY}, d_{YX}\}$ can be used,
 343 where d_{XY} and d_{YX} are the cumulative MJCs obtained by starting at x_1 and
 344 y_1 , respectively.

345 **3. Evaluation methodology**

346 *3.1. Classification scheme*

347 The efficacy of a time series similarity measure is commonly evaluated by
 348 the classification accuracy it achieves (Keogh and Kasetty, 2003; Wang et al.,
 349 2012). For that, the error ratio of a distance-based classifier is calculated
 350 for a given labeled data set, understanding the error ratio as the number of
 351 wrongly classified items divided by the total number of tested items. The
 352 standard choice for the classifier is the one-nearest neighbor (1NN) classifier.
 353 Following Wang et al. (2012), we can enumerate several advantages of using
 354 this approach. First, the error of the 1NN classifier critically depends on the
 355 similarity measure used. Second, the 1NN classifier is parameter-free and
 356 easy to implement. Third, there are theoretical results relating the error

357 of an 1NN classifier to errors obtained with other classification schemes.
358 Fourth, some works suggest that the best results for time series classification
359 come from simple nearest neighbor methods. For more details on these
360 aspects we refer to Mitchell (1997); Hastie et al. (2009), and the references
361 provided by Wang et al. (2012).

362 *3.2. Data sets*

363 We perform experiments with 45 publicly-available time series data sets
364 from the UCR time series repository (Keogh et al., 2011). This is the world’s
365 biggest time series repository, and some authors estimate that it makes up to
366 more than 90% of all publicly-available, labeled data sets (Wang et al., 2012).
367 The repository comprises synthetic, as well as real-world data sets, and
368 also includes one-dimensional time series extracted from two-dimensional
369 shapes (Keogh et al., 2011). The 45 data sets considered here correspond
370 to the totality of the UCR repository, as by March 2013. Within such data
371 sets, the number of classes ranges from 2 to 50, the number of time series
372 per data set ranges from 56 to 9,236, and time series lengths go from 24
373 to 1,882 samples. For further details on these data sets we refer to (Keogh
374 et al., 2011).

375 *3.3. Cross-validation*

376 To properly assess a classifier’s error, out-of-sample validation needs to
377 be done (Salzberg, 1997). In our experiments, we follow a standard 3-
378 fold cross-validation scheme using balanced data sets (Mitchell, 1997; Hastie
379 et al., 2009), i.e., using the same number of items per class. We repeat the
380 validation 20 times and report average error ratios. Balancing the data sets
381 allows for balanced error estimations regarding the class distribution, and
382 repeating cross-fold validation several times allows for more precise estima-
383 tions (Mitchell, 1997; Hastie et al., 2009). The use of a cross-fold validation
384 scheme is essential for avoiding the bias that a particular split of the data
385 could introduce (Salzberg, 1997; Hastie et al., 2009).

386 We also computed error ratios for the original splits provided in the
387 UCR time series repository (Keogh et al., 2011). This allowed us to confirm
388 that the 1NN error ratios from our implementations of DTW and Euclidean
389 distance agree with the values reported there. In addition, we observed that
390 the error ratios obtained by such splits were substantially different from the
391 ones obtained by cross-validation, up to the point of even modifying the
392 ranking of some algorithms with respect to those error ratios in some data
393 sets. This indicates a potential bias in such individual splits, an aspect that
394 is well-known in the machine learning community (Salzberg, 1997; Mitchell,

395 1997; Hastie et al., 2009). We refer the interested reader to any machine
396 learning textbook for a more in-depth discussion of cross-fold validation
397 schemes and their appropriateness over individual splits. Besides, individual
398 splits difficult statistical significance assessment (see below). A full account
399 of the raw error ratios for all measures and data sets is available online³,
400 including the error ratios for the aforementioned original splits.

401 3.4. *Statistical significance*

402 To assess the statistical significance of the difference between two error
403 ratios we employ the well-known Wilcoxon signed-rank test (Hollander and
404 Wolfe, 1999). The Wilcoxon signed-rank test is a non-parametric statistical
405 hypothesis test used when comparing two repeated measurements (or related
406 samples, or matched samples) in order to assess whether their population
407 mean ranks differ. It is the natural alternative to the Student’s t -test for
408 dependent samples when the population distribution cannot be assumed to
409 be normal (Hollander and Wolfe, 1999). For a given data set, we use as
410 input the 20×3 accuracy values obtained for each classifier (i.e., the test
411 fold accuracies). Besides, for comparing similarity measures on a more global
412 basis using all data sets, we employ as input the 45 average accuracy values
413 obtained for each data set. Following common practice (Salzberg, 1997;
414 Hollander and Wolfe, 1999), the threshold significance level is set to 5%.
415 Additionally, to compensate for multiple pairwise comparisons, we apply
416 the Holm-Bonferroni method (Holm, 1979), a post-hoc statistical analysis
417 method controlling the so-called family-wise error rate that is more powerful
418 than the usual Bonferroni correction (Demšar, 2006).

419 3.5. *Parameter choices*

420 Before performing the experiments, all time series from all data sets were
421 z-normalized so that each individual time series had zero mean and unit vari-
422 ance. Furthermore, we optimized the measures’ parameters in the training
423 phase of our cross-validation. This optimization step consisted of a grid
424 search within a suitable range of parameter values, forcing the same number
425 of parameter combinations per algorithm (Table 1). The values of the grid
426 are chosen according to common practice and the specifications given in the
427 papers introducing each measure (Sec. 2). Specifically, for FC we used 25
428 linearly-spaced integer values of $\theta \in [2, N/2]$. For AR we used 25 linearly-
429 spaced integer values of $\eta \in [1, 0.25N]$ (because of the z-normalization, we

³http://www.iiia.csic.es/~jserra/downloads/2013_SerraArcos_AnEmpiricalEvaluation.tar.gz (last accessed on September 15, 2013).

Measure	Parameter	Minimum value	Maximum value	Number of steps	Extra value
FC	θ	2	$0.5N$	25	-
AR	η	1	$0.25N$	25	-
DTW	ω	0	$0.25N$	24	N
EDR	ε	0.02σ	σ	25	-
TWED	ν	10^{-5}	1	5	-
TWED	λ	0	1	5	-
MJC	β	0	25	24	10^{10}

Table 1: Parameter grid for the considered similarity measures (recall that N corresponds to the length of the time series and, since we z-normalize all time series, $\sigma = 1$). For DTW and MJC we consider an extra value corresponding to unconstrained DTW and to the Euclidean configuration of MJC, respectively. All parameter values were linearly spaced except ν , which was logarithmically spaced.

430 remove a_0 in Eq. 3). For DTW we used 24 linearly-spaced integer values
431 of $\omega \in [0, 0.25N]$ plus $w = N$, the unconstrained DTW variant (we also
432 considered $\omega \in [0, 0.1N]$ and $\omega \in [0, 0.15N]$, but obtained no statistically
433 significant differences from $\omega \in [0, 0.25N]$ and none of the overall results
434 changed; considering $\omega \in [0, 0.05N]$ made DTW closely approach the results
435 of the Euclidean distance). For EDR we used 25 linearly-spaced real values
436 of $\varepsilon \in [0.02\sigma, \sigma]$, σ being the standard deviation of the time series (because of
437 the z-normalization $\sigma = 1$). For TWED we used all possible 25 combinations
438 for $\nu = [10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1]$ and $\lambda = [0, 0.25, 0.5, 0.75, 1]$. For MJC we
439 used 24 linearly-spaced real values of $\beta \in [0, 25]$ plus $\beta = 10^{10}$ (in practice
440 corresponding to the squared Euclidean distance variant, Eq. 11). After the
441 grid search, the parameter value yielding to the lowest leave-one-out error
442 ratio for the training set was kept for out-of-sample testing.

443 4. Results

444 4.1. Classification performance: test

445 If we look at the overall results, we see that all considered measures
446 clearly outperform the random baseline for practically all the 45 data sets
447 (Table 2). Furthermore, we see that some of them achieve near-perfect accu-
448 racies for a number of data sets (e.g., *CBF*, *CinC_ECG_torso*, *ECGFiveDays*,
449 *Two_Patterns*, or *TwoLeadECG*). However, no single measure achieves the
450 best performance for all the data sets. The Euclidean distance is found to
451 be the best-performing measure in 2 data sets, FC is the best-performing in
452 4 data sets, AR in 1, DTW in 6, EDR in 7, TWED in 20, and MJC in 5.
453 If we count only the data sets where one measure statistically significantly

454 outperforms the rest, the numbers reduce to 0 for Euclidean, 2 for FC, 1 for
455 AR, 2 for DTW, 2 for EDR, 6 for TWED, and 0 for MJC. Thus, interest-
456 ingly, there are some data sets where choosing a specific similarity measure
457 can make a difference.

458 Beyond accuracies, this latter aspect can potentially highlight inherent
459 data set qualities. For instance, the fact that a feature/model-based measure
460 clearly outperforms the others for a particular data set indicates that such
461 time series may be very well characterized by the extracted features/fitted
462 model (e.g., FC with *Adiac* for features and AR with *ChlorineConcentra-*
463 *tion* for models). In addition, the good or bad performance of Euclidean
464 and elastic measures gives us an intuition of the importance of alignments,
465 warping, or sample correspondences (e.g., these may be very important for
466 *Trace* and the three *Face* data sets, where there is an order of magnitude
467 difference between Euclidean and warping-based measures, but not much
468 for *DiatomSizeReduction* or *NonInvasiveFetalECG2*, where Euclidean gets
469 numbers that are very close, or even better than the ones obtained by the
470 warping-based measures).

471 In general, we see that TWED outperforms the other measures in several
472 data sets, with an average rank of 2.29 (Table 2). In fact, if we compare
473 the considered measures on a more global scale, taking the matched error
474 ratios across data sets (Sec. 3.4), we obtain that TWED is statistically
475 significantly superior to the rest (Fig. 2). Next, we see that DTW, MJC, and
476 EDR form a group of equivalent measures, with no statistically significant
477 difference between them. The performed statistical analysis also separates
478 the remaining measures from these and also between themselves. Apart
479 from this more global analysis, further pairwise comparisons can be made,
480 confirming the aforementioned global tendencies (Fig. 3).

481 4.2. Classification performance: test vs. train

482 For choosing the parameters for a given measure and data set we solely
483 dispose of the training data. Hence, it is important to know whether the
484 error ratios for training and testing sets are similar, otherwise one could
485 be incurring into the so-called “Texas sharpshooter fallacy” (Batista et al.,
486 2011), i.e., one could not predict a measure’s utility ahead of time by just
487 looking at training data. For comparing train and test error ratios, we can
488 compute an error gain value for a couple of measures on each data set and
489 check whether such values for train and test agree. To do so, a kind of real-
490 valued contingency table can be plotted, called the “Texas sharpshooter
491 plot” by Batista et al. (2011). Due to space reasons, we here only show such
492 contingency tables for TWED against DTW and Euclidean distance (Fig. 4).

#	Data set	Random	Euc	FC	AR	DTW	EDR	TWED	MJC
1	<i>50words</i>	0.969	0.503	0.685	0.867	0.332	0.289	0.237*	0.319
2	<i>Adiac</i>	0.970	0.345	0.266*	0.725	0.355	0.423	0.335	0.346
3	<i>Beef</i>	0.763	0.417	0.390	0.504	0.472	0.439	0.506	0.448
4	<i>CBF</i>	0.655	0.013	0.358	0.432	0.000	0.002	0.000	0.001
5	<i>ChlorineConcentration</i>	0.673	0.071	0.063	0.038*	0.072	0.094	0.093	0.070
6	<i>CinC_ECG_torso</i>	0.749	0.002	0.008	0.102	0.001	0.000*	0.001	0.002
7	<i>Coffee</i>	0.394	0.019	0.024	0.139	0.014	0.031	0.021	0.023
8	<i>Cricket_X</i>	0.913	0.378	0.348	0.713	0.209	0.237	0.190*	0.253
9	<i>Cricket_Y</i>	0.928	0.423	0.411	0.814	0.222	0.224	0.209	0.267
10	<i>Cricket_Z</i>	0.920	0.380	0.353	0.731	0.212	0.235	0.194*	0.254
11	<i>DiatomSizeReduction</i>	0.744	0.008	0.011	0.222	0.010	0.016	0.012	0.007
12	<i>ECG200</i>	0.515	0.130	0.145	0.227	0.139	0.148	0.109	0.130
13	<i>ECGFiveDays</i>	0.505	0.007	0.000	0.072	0.003	0.003	0.005	0.001
14	<i>FaceAll</i>	0.931	0.139	0.152	0.649	0.053	0.019	0.019	0.034
15	<i>FaceFour</i>	0.679	0.111	0.149	0.545	0.069	0.028	0.025	0.024
16	<i>FacesUCR</i>	0.929	0.138	0.148	0.648	0.052	0.019	0.018	0.041
17	<i>Fish</i>	0.871	0.183	0.234	0.617	0.184	0.084	0.094	0.114
18	<i>Gun_Point</i>	0.506	0.058	0.031	0.149	0.023	0.010	0.017	0.014
19	<i>Haptics</i>	0.793	0.604	0.610	0.678	0.554	0.611	0.544	0.563
20	<i>InlineSkate</i>	0.862	0.524	0.601	0.497	0.462	0.456	0.416	0.411
21	<i>ItalyPowerDemand</i>	0.489	0.035	0.083	0.261	0.033	0.042	0.036	0.034
22	<i>Lighting2</i>	0.488	0.297	0.281	0.450	0.162	0.220	0.161	0.254
23	<i>Lighting7</i>	0.817	0.371	0.463	0.707	0.252	0.362	0.256	0.336
24	<i>Mallat</i>	0.870	0.018	0.020	0.058	0.015	0.006	0.006	0.014
25	<i>MedicalImages</i>	0.912	0.313	0.455	0.458	0.247	0.330	0.228	0.305
26	<i>MoteStrain</i>	0.513	0.087	0.162	0.336	0.058	0.024	0.021	0.034
27	<i>NonInvasiveFetalECG1</i>	0.978	0.171	0.213	0.401	0.175	0.186	0.182	0.169
28	<i>NonInvasiveFetalECG2</i>	0.975	0.106	0.146	0.296	0.107	0.118	0.108	0.110
29	<i>OliveOil</i>	0.644	0.104	0.185	0.663	0.154	0.194	0.146	0.127
30	<i>OSULeaf</i>	0.832	0.409	0.306	0.617	0.359	0.191*	0.232	0.256
31	<i>SonyAIBORobotSurface</i>	0.510	0.017	0.040	0.079	0.018	0.026	0.017	0.015
32	<i>SonyAIBORobotSurfaceII</i>	0.489	0.018	0.032	0.113	0.021	0.023	0.016	0.019
33	<i>StarLightCurves</i>	0.671	0.124	0.070*	0.274	0.083	0.107	0.097	0.109
34	<i>SwedishLeaf</i>	0.932	0.196	0.142	0.376	0.129	0.101	0.094	0.100
35	<i>Symbols</i>	0.838	0.038	0.074	0.260	0.019	0.015	0.016	0.018
36	<i>Synthetic_control</i>	0.834	0.087	0.393	0.511	0.009*	0.047	0.014	0.034
37	<i>Trace</i>	0.757	0.169	0.117	0.117	0.000*	0.034	0.011	0.038
38	<i>Two_Patterns</i>	0.743	0.020	0.491	0.724	0.000	0.000	0.000	0.001
39	<i>TwoLeadECG</i>	0.507	0.006	0.012	0.202	0.001	0.002	0.001	0.003
40	<i>UWaveGestureLibrary_X</i>	0.872	0.234	0.566	0.694	0.199	0.214	0.192*	0.203
41	<i>UWaveGestureLibrary_Y</i>	0.876	0.288	0.631	0.645	0.263	0.280	0.265	0.267
42	<i>UWaveGestureLibrary_Z</i>	0.879	0.298	0.546	0.678	0.265	0.271	0.250*	0.261
43	<i>Wafer</i>	0.497	0.004	0.003	0.013	0.005	0.002	0.003	0.005
44	<i>WordsSynonyms</i>	0.960	0.496	0.675	0.855	0.327	0.304	0.251*	0.310
45	<i>Yoga</i>	0.500	0.070	0.108	0.333	0.061	0.034	0.037	0.047
Average rank		7.99	4.40	5.07	6.80	3.00	3.42	2.29	3.04

Table 2: Error ratios for all considered measures and data sets. The symbol * denotes a statistically significant difference with respect to the other measures for a given data set ($p < 0.05$, Sec. 3.4). The last row contains the average rank of each measure across all data sets (i.e., the average position after sorting the errors for a given data set in ascending order).

493 The results show that error gains between TWED and DTW/Euclidean
494 mostly agree between training and testing. As mentioned in Sec. 3.3, a full,
495 raw account of train and test errors is available online. Having a close look

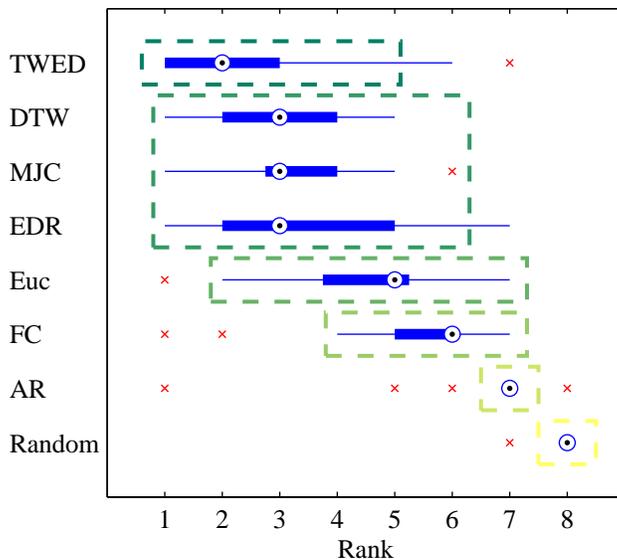


Figure 2: Box plot for the distribution of performance ranks of each measure across data sets. The dashed lines denote statistically significantly equivalent groups of measures ($p < 0.05$, Sec. 3.4).

496 at those full results, we can see that, in general, the best-performing measure
 497 at the training stage is also the best-performing measure at the testing stage.
 498 The few exceptions can be easily listed (Table 3). The relative rankings for
 499 the measures that do not perform best also mostly agree between train and
 500 test.

501 4.3. Parameter assessment

502 We finally report on the parameters chosen for each measure after train-
 503 ing with 66% of balanced data (Fig. 5). Firstly, we observe that, in the
 504 vast majority of cases, a specific value for a given parameter is consistently
 505 chosen across the 20×3 performed iterations (we see clear peaks in the dis-
 506 tributions of Fig. 5). Among these consistent choices, perhaps TWED and
 507 MJC present the most spread distributions. Such aspect, together with the
 508 fairly good accuracies obtained for these two specific measures (Sec. 4.1),
 509 indicates a certain degree of robustness against specific parameter choices.
 510 This is a very desirable quality of a time series similarity measure, even more
 511 if we have to train a classifier with a potentially incomplete set of training
 512 instances.

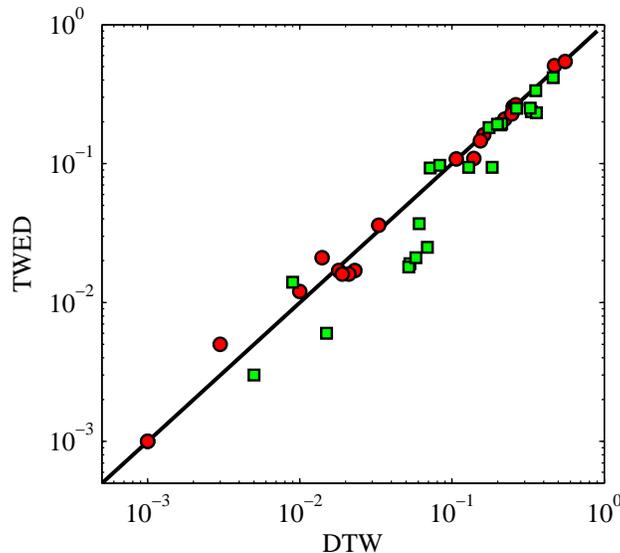


Figure 3: Error ratios comparison between DTW and TWED (notice the logarithmic axes). The lower-right triangular part corresponds to TWED outperforming DTW, whereas the upper-left part corresponds to the opposite case. The green squares indicate statistically significant performance differences ($p < 0.05$, Sec. 3.4).

513 Next, we see that the selected parameters are generally not in the borders
514 of the specified ranges, thus indicating that a reasonable choice for these has
515 been made (Fig 5). This is particularly true for DTW and EDR. Moreover,
516 in the case of DTW, we see that ω values generally coincide with the ones
517 suggested in the original data source (Keogh et al., 2011). In a total of 45
518 data sets we see 20 coincidences within ± 0.02 and 32 coincidences within
519 ± 0.04 . The only measure that could potentially benefit from reconsidering
520 the parameters' range is TWED. As it can be seen, ν and λ seem to be
521 consistently chosen in the lower and upper parts of the specified ranges,
522 respectively. This suggests that the best combination for some data sets
523 could lie outside the parameter space outlined by Marteau (2009), i.e., in
524 $0 < \nu < 10^{-4}$ and/or $\lambda > 1$. If that was the case, TWED could potentially
525 achieve even much higher accuracies. Interestingly, TWED is not the best-
526 performing measure for some of the data sets where 'border' parameter
527 values are chosen (e.g., *CBF*, *Fish*, *StarLightCurves*, *TwoPatterns*).

528 Finally, we can comment on the particularities of some data sets with
529 relation to classification. For instance, we see that a relatively large window

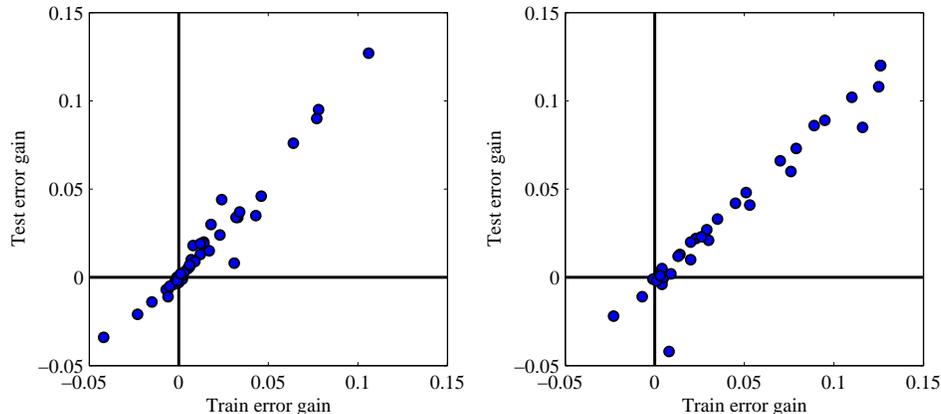


Figure 4: Texas sharpshooter plots for TWED against DTW (left) and Euclidean distance (right). Here, error gain is measured by subtracting the TWED error ratio from the one of DTW/Euclidean. Dots around the diagonal indicate agreement of error gain for train and test. False positives, i.e., dots in the lower-right quadrant, indicate that TWED, being the best measure after training, does not reach the lowest error at testing. For instance, in the case of TWED vs. Euclidean (right), the *OliveOil* data set false positive stands out at coordinates $(0.008, -0.042)$ (see also Table 3). For further details on the construction of Texas sharpshooter plots we refer to Batista et al. (2011).

530 parameter ω (DTW) is chosen for data sets 36 to 39 (i.e., *Synthetic_control*,
531 *Trace*, *Two_Patterns*, and *TwoLeadECG*). This denotes that tracking align-
532 ments or warping paths beyond the main diagonal of D (Eq. 4) might be
533 advantageous for classification in these data sets. In fact, when we re-ran the
534 same experiment restricting ω to be between 0 and 0.1 we obtained the same
535 or worse error rates, an effect that can also be observed by comparing the
536 results obtained for the UCR splits (Keogh et al., 2011) which, as mentioned
537 in Sec. 3.3, are available online. The stiffness parameter ν (TWED), which
538 accounts for a similar but opposite concept (Sec. 2.6), takes relatively small
539 values. Such agreement across different measures reinforces the hypothesis
540 that tracking intricate alignments or strongly warped paths may be advan-
541 tageous for these data sets. Analogous and complementary conclusions can
542 be derived for other data sets. For instance, in data sets 11 (*DiatomSizeRe-*
543 *duction*) and 13 (*ECGFiveDays*), a small number of both FCs θ and AR
544 coefficients η is chosen. As FC and AR achieve competitive accuracies in
545 those specific data sets, we could suspect that low-frequency components are
546 important for correctly classifying the instances in those data sets (Secs. 2.2
547 and 2.3).

#	Data set	Measure	Outperf. by	Gain
3	<i>Beef</i>	FC	EDR	0.049
4	<i>CBF</i>	TWED	DTW	<0.001
7	<i>Coffee</i>	DTW	FC	0.004
12	<i>ECG200</i>	TWED	MJC	0.002
15	<i>FaceFour</i>	MJC	EDR	0.007
18	<i>Gun_Point</i>	EDR	MJC	0.004
19	<i>Haptics</i>	TWED	MJC	0.009
21	<i>ItalyPowerDemand</i>	DTW	EDR	0.001
28	<i>NonInvasiveFetalECG2</i>	Euclidean	TWED	0.001
29	<i>OliveOil</i>	Euclidean	TWED	0.008
39	<i>TwoLeadECG</i>	TWED	DTW	<0.001

Table 3: List of best-performing measures in testing (the column “Measure”) but actually outperformed by others in training (the column “Outperf. by”). The column “Gain” corresponds to the absolute value of the train error gain, i.e., the absolute difference between error ratios at training stage (see also Fig. 4).

548 5. Conclusion

549 From a general perspective, the obtained results show that there is a
550 group of equivalent similarity measures, with no statistically significant dif-
551 ferences among them (DTW, EDR, and MJC). The existing literature sug-
552 gests that some longest common sub-sequence approaches, together with al-
553 ternative variants of DTW and EDR, could potentially join this group (Marteau,
554 2009; Wang et al., 2012). However, according to the results reported here,
555 the TWED measure originally proposed by Marteau (2009) seems to consis-
556 tently outperform all the considered distances, including DTW, EDR, and
557 MJC. Thus, we believe this often unconsidered measure should take a base-
558 line role in future evaluations of time series similarity measures (beyond
559 accuracy, additional properties enumerated in Sec. 2.6 make it also very
560 attractive). The Euclidean distance, although somehow competitive, gener-
561 ally performs statistically significantly worse than TWED, DTW, MJC, and
562 EDR. Its accuracy on large data sets was also not very impressive. Below
563 Euclidean distance, but statistically significantly above the random baseline,
564 we find FC and AR measures. Of course, the general statements above do
565 not exclude the possibility that a particular measure or variant could be very
566 well-suited for a specific data set and statistically significantly outperform
567 the rest (cf. Keogh and Kasetty, 2003). In Sec. 4.1 have enumerated several

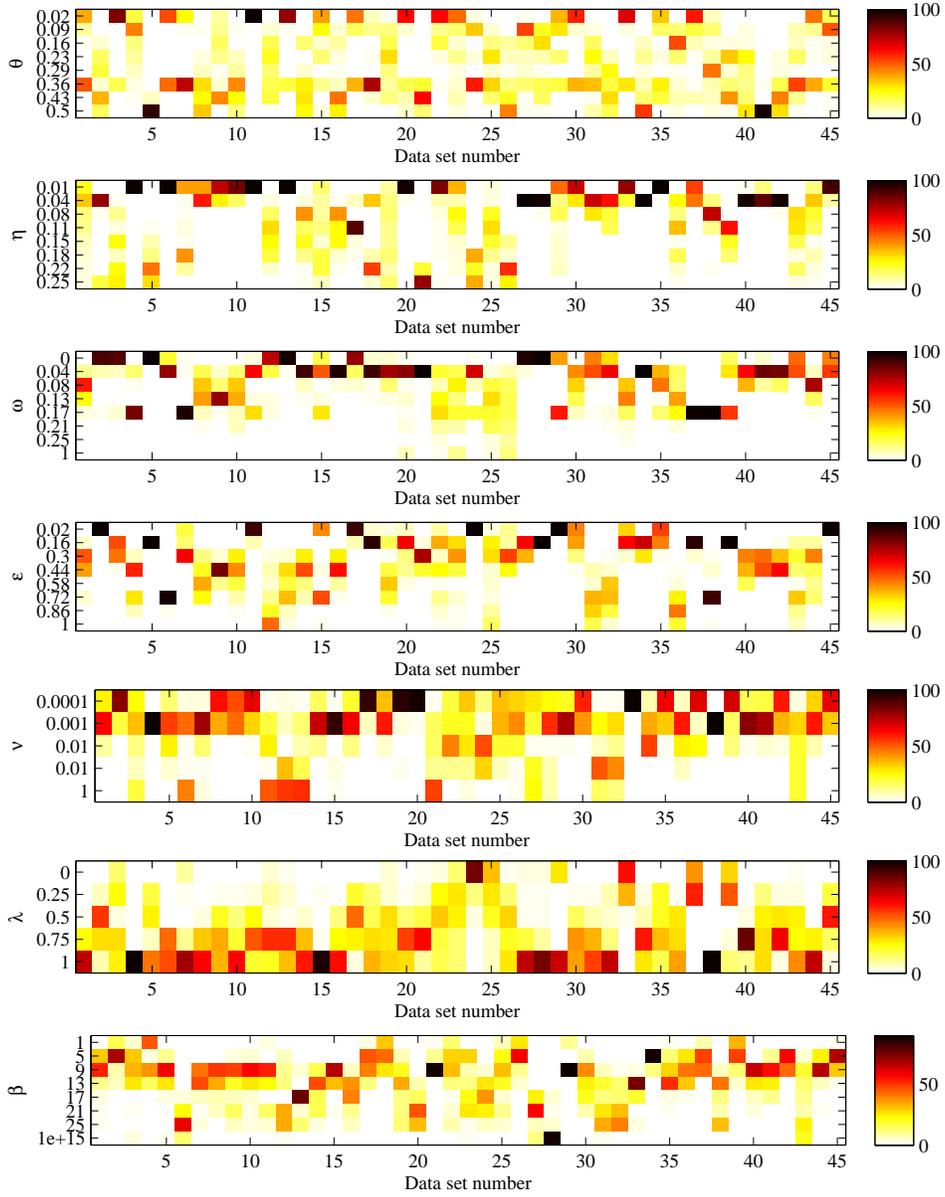


Figure 5: Percentage of times (color code) that a given parameter value (vertical axis) is chosen for each data set (horizontal axis; for the names behind each number see Table 2). From top to bottom, the plots correspond to FC (θ), AR (η), DTW (ω), EDR (ϵ), TWED (ν), TWED (λ), and MJC (β).

568 examples of that.

569 When comparing train and test errors, we have seen that these mostly
570 agree, with train errors generally providing a good guess of the test errors on
571 unseen data. We have listed some notable exceptions to this rule and used
572 Texas sharpshooter plots to further assess this aspect for TWED vs. DTW
573 and Euclidean. When assessing the best parameter choices for each measure,
574 we have seen that the considered ranges are typically suitable for the task
575 at hand. We have also discussed some particularities regarding parameter
576 choices and the nature of a few data sets.

577 The similarity measure is a crucial step in computational approaches
578 dealing with time series. However, there are some additional issues worth
579 mentioning, in particular with regard to post-processing steps focused on
580 improving similarity assessments (pre-processing steps are sufficiently well-
581 discussed in the existing literature, see, e.g., Keogh and Kasetty (2003);
582 Han and Kamber (2005); Wang et al. (2012) and references therein). A
583 very interesting post-processing step is the complexity-invariant correction
584 factor introduced by Batista et al. (2011). Such correction factor prevents
585 from assigning low dissimilarity values to time series of different complexity,
586 thus preventing the inclusion of time series of different nature in the same
587 cluster. The way to assess complexity depends on the situation, but Batista
588 et al. (2011) introduce a quite straightforward way: the L_2 norm of the
589 sample-based derivative of a time series. Overall, considering different types
590 of ‘invariance’ is a sensible approach (Batista et al., 2011, provide a good
591 overview). Here, we have already implicitly considered a number of them,
592 although more as a pre-processing or method-specific strategy: global ampli-
593 tude and scale invariance (z-normalization), warping invariance (any elastic
594 measure, in our case DTW, EDR, TWED, and MJC), phase invariance
595 (AR^4), and occlusion invariance (EDR and TWED).

596 Another interesting post-processing step is the hubness correction for
597 time series classification introduced by Radovanovič et al. (2010). Based on
598 the finding that some instances in high-dimensional spaces tend to become
599 hubs by being unexpectedly (and usually wrongly) considered nearest neigh-
600 bors of several other instances, a correction factor can be introduced. This
601 usually does not harm classification accuracy and can definitely improve per-
602 formance for some data sets (Radovanovič et al., 2010). A further strategy
603 for enhancing time series similarity and potentially reducing hubness is the
604 use of unsupervised clustering algorithms to prune nearest neighbor candi-

⁴For FC we use both phase and magnitude (Sec. 2.2).

605 dates (Serrà et al., 2012b). Future work should focus on the real quantitative
606 impact of strategies for enhancing time series similarity like the ones above,
607 with a special emphasis on its impact to different measures and classification
608 schemes.

609 The empirical comparison of multiple approaches across a large-scale case
610 basis is an important and necessary step towards any mature research field.
611 Besides getting a more global picture and highlighting relevant approaches,
612 it pushes towards unified validation procedures and analysis tools. It is
613 hoped that this article will serve as a steppingstone for those interested in
614 advancing in time series similarity, clustering, and classification.

615 **Acknowledgements**

616 We thank the people who made available or contributed to the UCR time
617 series repository. This research has been funded by 2009-SGR-1434 from
618 Generalitat de Catalunya, JAEDOC069/2010 from Consejo Superior de In-
619 vestigaciones Científicas, and TIN2009-13692-C03-01 and TIN2012-38450-
620 C03-03 from the Spanish Government, and EU Feder funds.

621 **References**

- 622 Agrawal, R., Faloutsos, C., Swami, A., 1993. Efficient similarity search in
623 sequence databases, in: Proc. of the Int. Conf. on Foundations of Data
624 Organization and Algorithms, pp. 69–84.
- 625 Bartolini, I., Ciaccia, P., Patella, M., 2005. WARP: accurate retrieval of
626 shapes using phase of Fourier descriptors and time warping distance. IEEE
627 Trans. on Pattern Analysis and Machine Intelligence 27, 142–147.
- 628 Batista, G.E.A.P.A., Wang, X., Keogh, E.J., 2011. A complexity-invariant
629 distance measure for time series, in: Proc. of the SIAM Int. Conf. on Data
630 Mining, pp. 699–710.
- 631 Berndt, D.J., Clifford, J., 1994. Using dynamic time warping to find patterns
632 in time series, in: Proc. of the AAAI Workshop on Knowledge Discovery
633 in Databases, pp. 359–370.
- 634 Cai, Y., Ng, R., 2004. Indexing spatio-temporal trajectories with Chebyshev
635 polynomials, in: Proc. of the ACM SIGMOD Int. Conf. on Management
636 of Data, pp. 599–610.

- 637 Chan, K.P., Fu, A.C., 1999. Efficient time series matching by wavelets, in:
638 Proc. of the IEEE Int. Conf. on Data Engineering, pp. 126–133.
- 639 Chen, L., Öszu, M.T., Oria, V., 2005. Robust and fast similarity search for
640 moving object trajectories, in: Proc. of the ACM SIGMOD Int. Conf. on
641 Management of Data, pp. 491–502.
- 642 Das, G., Lin, K.I., Mannila, H., Renganathan, G., Smyth, P., 1998. Rule
643 discovery from time series, in: Proc. of the AAAI Int. Conf. on Knowledge
644 Discovery and Data Mining, pp. 16–22.
- 645 Demšar, J., 2006. Statistical comparison of classifiers over multiple data
646 sets. *Journal of Machine Learning Research* 7, 1–30.
- 647 Faloutsos, C., Ranganathan, M., Manolopoulos, Y., 1994. Fast subsequence
648 matching in time-series databases, in: Proc. of the ACM SIGMOD Int.
649 Conf. on Management of Data, pp. 419–429.
- 650 Fu, T.C., 2011. A review on time series data mining. *Engineering Applica-*
651 *tions of Artificial Intelligence* 24, 164–181.
- 652 Geurts, P., 2002. Contributions to decision tree induction: bias/variance
653 tradeoff and time series classification. Ph.D. thesis. University of Liège,
654 Liège, Belgium.
- 655 Gusfield, D., 1997. Algorithms on strings, trees, and sequences: computer
656 science and computational biology. Cambridge University Press, Cam-
657 bridge, UK.
- 658 Han, J., Kamber, M., 2005. Data mining: concepts and techniques. Morgan
659 Kaufmann, Waltham, USA.
- 660 Hastie, T., Tibshirani, R., Friedman, J., 2009. The elements of statistical
661 learning. 2nd ed., Springer, Berlin, Germany.
- 662 Hollander, M., Wolfe, D.A., 1999. Nonparametric statistical methods. 2nd
663 ed., Wiley, New York, USA.
- 664 Holm, S., 1979. A simple sequentially rejective multiple test procedure.
665 *Scandinavian Journal of Statistics* 6, 65–70.
- 666 Kantz, H., Schreiber, T., 2004. Nonlinear time series analysis. Cambridge
667 University Press, Cambridge, UK.

- 668 Keogh, E.J., 2011. Machine learning in time series databases (and everything
669 is a time series!). Tutorial at the AAAI Int. Conf. on Artificial Intelligence.
- 670 Keogh, E.J., Chakrabarti, K., Pazzani, M., Mehrotra, S., 2001. Dimension-
671 ality reduction for fast similarity search in large time series databases.
672 Knowledge and Information Systems 3, 263–286.
- 673 Keogh, E.J., Kasetty, S., 2003. On the need for time series data mining
674 benchmarks: a survey and empirical demonstration. Data Mining and
675 Knowledge Discovery 7, 349–371.
- 676 Keogh, E.J., Ratanamahatana, C.A., 2005. Exact indexing of dynamic time
677 warping. Knowledge and Information Systems 7, 358–386.
- 678 Keogh, E.J., Xi, X., Wei, L., Ratanamahatana, C.A., 2009. Supporting
679 exact indexing of arbitrarily rotated shapes and periodic time series under
680 Euclidean and warping distance measures. VLDB Journal 11, 611–630.
- 681 Keogh, E.J., Zhu, Q., Hu, B., Hao, Y., Xi, X., Wei, L., Ratanamahatana,
682 C.A., 2011. The UCR time series classification/clustering homepage.
683 URL: http://www.cs.ucr.edu/~%7eeamonn/time_series_data.
- 684 Kholmatov, A., Yanikoglu, B., 2005. Identity authentication using improved
685 online signature verification method. Pattern Recognition Letters 26,
686 2400–2408.
- 687 Lemire, D., 2009. Faster retrieval with a two-pass dynamic time warping
688 lower bound. Pattern Recognition 42, 2169–2180.
- 689 Levenshtein, V.I., 1966. Binary codes capable of correcting deletions, inser-
690 tions, and reversals. Soviet Physics Doklady 10, 707–710.
- 691 Liao, T.W., 2005. Clustering of time series data: a survey. Pattern Recog-
692 nition 38, 1857–1874.
- 693 Maharaj, E.A., 2000. Clusters of time series. Journal of Classification 17,
694 297–314.
- 695 Marple, S.L., 1987. Digital spectral estimation. Prentice-Hall, Englewood
696 Cliffs, USA.
- 697 Marteau, R.F., 2009. Time warp edit distance with stiffness adjustment
698 for time series matching. IEEE Trans. on Pattern Analysis and Machine
699 Intelligence 31, 306–318.

- 700 Mitchell, T.M., 1997. *Machine Learning*. McGraw-Hill, New York, USA.
- 701 Montani, S., Portinale, L., Leonardi, G., Bellazzi, R., 2006. Case-based
702 retrieval to support the treatment of end stage renal failure patients. *Artificial Intelligence in Medicine* 37, 31–42.
703
- 704 Morse, M.D., Patel, J.M., 2007. An efficient and accurate method for eval-
705 uating time series similarity, in: *Proc. of the ACM SIGMOD Int. Conf. on Management of Data*, pp. 569–580.
706
- 707 Olsson, E., Funk, P., Xiong, N., 2004. Fault diagnosis in industry using
708 sensor readings and case-based reasoning. *Journal of Intelligent Fuzzy Systems* 15, 41–46.
709
- 710 Oppenheim, A.V., Schafer, R.W., Buck, J.R., 1999. *Discrete-time signal processing*. 2nd ed., Prentice-Hall, Upper Saddle River, USA.
711
- 712 Piccolo, D., 1990. A distance measure for classifying ARMA models. *Journal of Time Series Analysis* 11, 153–163.
713
- 714 Povinelli, R.J., Johnson, M.T., Lindgren, A.C., Ye, J., 2004. Time se-
715 ries classification using Gaussian mixture models of reconstructed phase
716 spaces. *IEEE Trans. on Knowledge Discovery and Data Engineering* 16,
717 779–783.
- 718 Rabiner, L.R., Juang, B., 1993. *Fundamentals of speech recognition*.
719 Prentice-Hall, Upper Saddle River, USA.
- 720 Radovanović, M., Nanopoulos, A., Ivanovic, M., 2010. Time-series classifi-
721 cation in many intrinsic dimensions, in: *Proc. of the SIAM Int. Conf. on Data Mining*, pp. 677–688.
722
- 723 Ramoni, M., Sebastiani, P., Cohen, P., 2002. Bayesian clustering by dynam-
724 ics. *Machine Learning* 47, 91–121.
- 725 Rodríguez, J.J., Alonso, C., 2004. Interval and dynamic time warping-based
726 decision trees, in: *Proc. of the ACM Symp. on Applied Computing (SAC)*,
727 pp. 548–552.
- 728 Sakoe, H., Chiba, S., 1978. Dynamic programming algorithm optimization
729 for spoken word recognition. *IEEE Trans. on Acoustics, Speech, and Language Processing* 26, 43–50.
730
- 731 Salvador, S., Chan, P., 2007. Toward accurate dynamic time warping in
732 linear time and space. *Intelligent Data Analysis* 11, 561–580.

- 733 Salzberg, S.L., 1997. On comparing classifiers: pitfalls to avoid and a rec-
734 ommended approach. *Data Mining and Knowledge Discovery* 1, 317–328.
- 735 Serrà, J., Arcos, J.L., 2012. A competitive measure to assess the similar-
736 ity between two time series, in: *Proc. of the Int. Conf. on Case-Based*
737 *Reasoning (ICCBR)*, pp. 414–427.
- 738 Serrà, J., Kantz, H., Serra, X., Andrzejak, R.G., 2012a. Predictability of
739 music descriptor time series and its application to cover song detection.
740 *IEEE Trans. on Audio, Speech and Language Processing* 20, 514–525.
- 741 Serrà, J., Zanin, M., Herrera, P., Serra, X., 2012b. Characterization and ex-
742 ploitation of community structure in cover song networks. *Pattern Recog-*
743 *nition Letters* 33, 1032–1041.
- 744 Tormene, P., Giorgino, T., Quaglini, S., Stefanelli, M., 2009. Matching
745 incomplete time series with dynamic time warping: an algorithm and an
746 application to post-stroke rehabilitation. *Artificial Intelligence in Medicine*
747 45, 11–34.
- 748 Wang, X., Mueen, A., Ding, H., Trajcevski, G., Scheuermann, P., Keogh,
749 E.J., 2012. Experimental comparison of representation methods and dis-
750 tance measures for time series data. *Data Mining and Knowledge Discov-*
751 *ery* In press. URL: <http://dx.doi.org/10.1007/s10618-012-0250-5>.
- 752 Wu, D., Agrawal, D., El Abbadi, A., Singh, A., Smith, T.R., 1996. Efficient
753 retrieval for browsing large image databases, in: *Proc. of the Int. Conf.*
754 *on Knowledge Information*, pp. 11–18.
- 755 Wu, Y.L., Agrawal, D., El Abbadi, A., 2000. A comparison of DFT and
756 DWT based similarity search in time-series databases, in: *Proc. of the*
757 *Int. Conf. on Information and Knowledge Management (CIKM)*, pp. 488–
758 495.
- 759 Xi, X., Keogh, E.J., Shelton, C.R., Wei, L., Ratanamahatana, C.A., 2006.
760 Fast time series classification using numerosity reduction, in: *Proc. of the*
761 *Int. Conf. on Machine Learning*, pp. 1033–1040.
- 762 Xing, Z., Pei, J., Yu, P.S., 2011. Early classification on time series. *Knowl-*
763 *edge and Information Systems* 31, 105–127.