A Structured Study of Multivariate Time-Series Distance Measures

JENS E. D'HONDT*, Eindhoven University of Technology, the Netherlands
HAOJUN LI*, The Ohio State University, USA
FAN YANG*, The Ohio State University, USA
ODYSSEAS PAPAPETROU, Eindhoven University of Technology, the Netherlands
JOHN PAPARRIZOS, The Ohio State University, USA

Distance measures are fundamental to time series analysis and have been extensively studied for decades. Until now, research efforts mainly focused on univariate time series, leaving multivariate cases largely under-explored. Furthermore, the existing experimental studies on multivariate distances have critical limitations: (a) focusing only on lock-step and elastic measures while ignoring categories such as sliding and kernel measures; (b) considering only one normalization technique; and (c) placing limited focus on statistical analysis of findings. Motivated by these shortcomings, we present the most complete evaluation of multivariate distance measures to date. Our study examines 30 standalone measures across 8 categories, 2 channel-dependency models, and considers 13 normalizations. We perform a comprehensive evaluation across 30 datasets and 3 downstream tasks, accompanied by rigorous statistical analysis. To ensure fairness, we conduct a thorough investigation of parameters for methods in both a supervised and an unsupervised manner. Our work verifies and extends earlier findings, showing that insights from univariate distance measures also apply to the multivariate case: (a) alternative normalization methods outperform Z-score, and for the first time, we demonstrate statistical differences in certain categories for the multivariate case; (b) multiple lock-step measures are better suited than Euclidean distance, when it comes to multivariate time series; and (c) newer elastic measures outperform the widely adopted Dynamic Time Warping distance, especially with proper parameter tuning in the supervised setting. Moreover, our results reveal that (a) sliding measures offer the best trade-off between accuracy and runtime; (b) current normalization techniques fail to significantly enhance accuracy on multivariate time series and, surprisingly, do not outperform the no normalization case, indicating a lack of appropriate solutions for normalizing multivariate time series; and (c) independent consideration of time series channels is beneficial only for elastic measures. In summary, we offer guidelines to aid in designing and selecting preprocessing strategies and multivariate distance measures for our community.

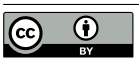
CCS Concepts: • **Information systems → Similarity measures**; **Top-k retrieval in databases**; *Clustering and classification*; **Retrieval effectiveness**; *Retrieval efficiency*; **Nearest-neighbor search**; *Clustering*.

Additional Key Words and Phrases: Multivariate Time Series, Distance Measures, Similarity Search

ACM Reference Format:
Jens E. d'Hondt, Haojun Li, Fan Yang, Odysseas Papapetrou, and John Paparrizos. 2025. A Structured Study of Multivariate Time-Series Distance Measures. *Proc. ACM Manag. Data* 3, 3 (SIGMOD), Article 121 (June 2025), 29 pages. https://doi.org/10.1145/3725258

*Equal contribution, ordered alphabetically

Authors' Contact Information: Jens E. d'Hondt, j.e.d.hondt@tue.nl, Eindhoven University of Technology, Eindhoven, the Netherlands; Haojun Li, li.14118@osu.edu, The Ohio State University, Columbus, Ohio, USA; Fan Yang, yang.7007@osu.edu, The Ohio State University, Columbus, Ohio, USA; Odysseas Papapetrou, o.papapetrou@tue.nl, Eindhoven University of Technology, Eindhoven, the Netherlands; John Paparrizos, john@paparrizos.org, The Ohio State University, Columbus, Ohio, USA.



ACM 2836-6573/2025/6-ART121
https://doi.org/10.1145/3725258

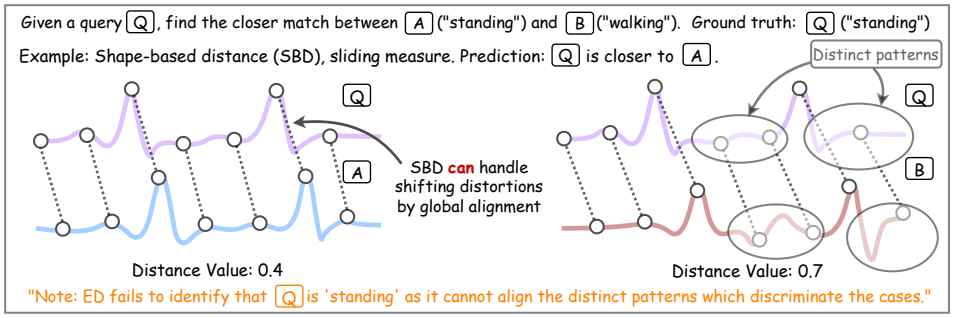


Fig. 1. An illustrative example highlighting the importance of distance measure selection in time series analysis, using synthetic data inspired by UEA's StandWalkJump.

1 Introduction

Time series, ordered sequences of real-valued observations, have become ubiquitous in diverse domains, such as meteorology, astrophysics, neuroscience, behavioral science, and finance [3, 44, 50, 58, 68, 73, 78, 91, 92, 95, 103]. In recent decades, the rapid development of sensing technologies has facilitated the collection of vast amounts of time series data [52, 53, 60, 64, 65, 87, 99], commonly categorized into *univariate time series* (UTS) and *multivariate time series* (MTS). UTS refers to time-varying measurements where each observation is a scalar, while MTS consists of a collection of co-evolving UTS, making each observation multi-dimensional [36, 86, 115]. For example, an MTS in meteorology is a collection of measurements from sensors, where each sensor represents a distinct channel, such as temperature or wind speed [51, 79, 111]. The identification of similarities between time series, through a *distance or similarity measure*, constitutes the foundation for time series analytics tasks such as querying [2, 28, 34, 55, 59, 62, 77, 89, 93], indexing [20–22, 24, 37, 38, 40, 57, 81, 96], clustering [6, 7, 10, 35, 54, 83, 84, 90, 94, 102], classification [4, 49, 100, 104, 105, 114, 115], motif discovery [9, 25, 63, 75, 116, 117], and anomaly detection [11–19, 32, 66, 67, 80, 85, 107, 108].

However, the design of such measures is non-trivial. In contrast to other data types (e.g., text), time series data has a temporal aspect that plays a crucial role in conducting an effective comparison, and is often very high-dimensional [37]. Moreover, the difficulty in defining a suitable distance measure for time series stems from the lack of a clear formulation of what constitutes a meaningful similarity [88]. Specifically, humans can recognize perceptually similar time series, disregarding various *distortions* in the series, such as temporal misalignments, scaling differences, and noise–dissimilarities that are irrelevant to the comparison. Nevertheless, the implicit rules guiding this process are intricate and context-dependent, making them difficult to mathematically formalize [39]. These intricacies are illustrated in Figure 1 in the context of motion capture data, where a phase shift in a query signal leads to a misclassification when using a measure that does not correct for distortions, but a correct prediction in the case of a measure like Shape-based distance (SBD), which considers all possible shifts between time series.

The difficulty of handling various distortions in time series has resulted in the development of dozens of distance measures over decades of research. Paparrizos et al. [88] introduced a taxonomy categorizing distance measures into five groups: (a) *lock-step* measures, which compare values at corresponding time points and aggregate the differences across time; (b) *sliding* measures, which find the optimal shift between two time series; (c) *elastic* measures, which allow one-to-many mappings of time steps to handle local temporal distortions; (d) *kernel* measures, which implicitly project time series into a higher-dimensional space using a kernel function; (e) *embedding* measures, which explicitly transform time series to new representations, which then serve as the new basis for

Table 1. Summary of our experimental evaluation of 30 standalone MTS measures (46 variants in total when considering channel-dependency models and ensembles) across two channel-dependency models (I: independent, D: dependent) and three downstream tasks (CLS: classification, CLU: clustering, AD: anomaly detection). The last two columns show the summary of prior evaluation studies [5] and [104].

Temp. Model	Measures	Dep. Models, #Norms	Downstream Tasks	[5]	[104]
Lock-step	11	(I, 13)	CLS, CLU, AD	1 | (I,1) | CLS	1 | (I,1) | CLS
Sliding	1	(I & D,13)	CLS, CLU, AD	✗	✗
Elastic	5	(I & D, 13)	CLS, CLU, AD	1 | (I&D,1) | CLS	5 | (I&D,1) | CLS
Kernel	4	(I & D, 13)	CLS	✗	✗
Feature	2	(I, 13)	CLS	✗	✗
Model	2	(I & D, 13)	CLS	✗	✗
Embedding	5	(I & D, 13)	CLS	✗	✗
Ensemble	2	(I & D, 13)	CLS	✗	✗

comparison. The authors note that no single distance measure can effectively address all types of distortions simultaneously. This becomes even more challenging the context of MTS, as distortions may or may not be propagated across channels [105].

Despite extensive efforts in developing distance measures, misconceptions about their properties and the trade-offs between accuracy and efficiency persist, due to the limited focus on performing comprehensive evaluation studies [88]. Recent works managed to debunk several of these misconceptions by performing large-scale evaluations across a variety of datasets and distance measures [34, 88]. Unfortunately, these evaluations were restricted to UTS, leaving the challenges associated with MTS under-explored. The handful of existing studies on multivariate distance measures have key limitations [5, 104], including (a) mainly focusing on lock-step and elastic measures while disregarding competing families such as sliding and kernel measures; (b) considering only a single normalization; and (c) performing limited statistical analysis. Notably, with these limitations, the guidance these studies offer remains constrained.

Motivated by these issues, this paper aims to provide a rigorous comparison of multivariate distance measures through a holistic view. Namely, to accurately map the current landscape of measures, we take a systematic approach that considers the three key axes: (a) ***Normalization***, which involves the rescaling of time series as a preprocessing step; (b) ***Temporal Model***, which determines how measures address temporal distortions between time series; (c) and the ***Channel-Dependency Model***, which determines the criteria to address correlations between channels, either by assuming independence across all channels (*channel-independent*) or incorporating interdependencies (*channel-dependent*). These three axes serve as the basis for categorizing and comparing existing MTS distance measures, facilitating the selection and discovery of effective preprocessing techniques and new distance measure designs.

Guided by these three key axes, we present the most comprehensive study of MTS distance measures to date: while prior works have faced notable limitations [5, 104] – from the lack of extensions for MTS distance measures to the absence of meta-analyses – we implement, properly parameterize, and evaluate a total of 30 standalone MTS distance measures. Specifically, we (a) conduct a meta-analysis on 13 normalizations, and critically examine whether existing methods outperform the case of not normalizing (Nonorm); (b) propose a taxonomy of eight categories (seven temporal models + measure ensembles), assessing each measure based on both accuracy and runtime performance; (c) explore two channel-dependency models, providing guidelines on how to implement each model for each measure category and normalization method. We pay careful attention to the parameterization of measures by considering broad ranges of values, validated to

ensure convergence to optimal values. Furthermore, as a byproduct of the study, we provide an easy-to-use library with implementations of all evaluated measures [1].

To quantify the discriminative power of each measure, we evaluate their accuracy in the one-nearest-neighbor (1NN) classification task across all 30 datasets from the UEA archive [5], under both supervised and unsupervised settings. Moreover, we perform two additional experiments on clustering and anomaly detection to extend our findings. Alongside accuracy comparisons, we place a strong emphasis on runtime performance, measuring how the runtime of measures scales across varying time series lengths and numbers of channels in MTS. We validate our results and findings using two statistical tests to assess the significance of differences, one for pairwise comparisons and the other for evaluating global performance across multiple measures. In summary, we reaffirm previous findings on distance measures in the context of multivariate time series, confirming that: (a) Z-score normalization is not the best normalization technique; (b) various lock-step measures outperform Euclidean distance; and (c) newer elastic measures outperform Dynamic Time Warping (DTW) [101], the current state of the art, in the supervised setting. Furthermore, our results show that: (a) sliding measures offer the best trade-off between accuracy and runtime efficiency; (b) embedding measures, including deep-learning-based methods, are not superior to traditional measures; (c) current normalization techniques do not significantly improve accuracy on multivariate data; and (d) channel-dependent variants of measures generally outperform their channel-independent counterparts, with the exception of elastic measures. Table 1 compares statistics of our experimental evaluation against related studies [5, 104], highlighting the key observations that motivated this research and demonstrating the comprehensiveness of our work.

We start with the related work in the field of distance measure evaluations (Section 2). Then, we summarize our contributions in this work, which are detailed as follows:

- We present three axes that ensure the comprehensiveness of our evaluations on MTS measures: normalization, temporal model, and channel-dependency model (Section 3).
- We provide a complete overview of existing distance measures for time-series analysis, structured by the introduced measure properties (Section 4).
- We conduct an evaluation of 30 standalone measures across 30 datasets with 13 normalization techniques, measuring their performance in terms of accuracy and runtime efficiency (Section 5).
- We provide guidelines for MTS measures selection (Section 6).

Finally, we conclude with the implications of our work and a discussion of challenges and new directions for possible future research (Section 7).

2 Related work

In this section, we review recent experimental studies on time series distance measures, emphasizing the relevance of our work and illustrating how it aligns with, or challenges prior research. Additional details on individual distance measures within each category are presented in Section 4.

Recently, Paparrizos et al. [88] introduced a taxonomy that classifies univariate distance measures into five distinct categories and conducted a comprehensive comparison to challenge long-standing misconceptions in the field. The experimental results yielded four key conclusions: (a) alternative normalization methods can outperform Z-score normalization; (b) other lock-step distance measures surpass the widely used Euclidean distance; (c) sliding measures demonstrate competitive performance compared to elastic measures in supervised and unsupervised settings; and (d) DTW is not always the best-performing elastic measure, with newer measures like MSM outperforming it. Inspired by this work, we extend this taxonomy to MTS, introduce two new categories, and propose a principled approach to extending UTS distance measures to MTS. Our study partly confirms the four main conclusions of the study of [88] in the multivariate case, but also uncovers several

discrepancies and novel findings: (a) Not normalizing (Nonorm) is currently the best choice for MTS, (b) elastic measures for MTS *only* outperform sliding measures under supervised parameter tuning, and (c) state-of-the-art deep-learning-based measures are not superior to traditional measures. Furthermore, our study additionally considers ensemble methods and includes two extra downstream tasks on clustering and outlier detection to extend our findings beyond classification.

Shokoohi-Yekta et al. [105] proposed two channel-dependency extension strategies for extending DTW to the multivariate case: *dependent* (DTW-D) and *independent* (DTW-I). In DTW-D, all channels are treated collectively and share a single warping path, whereas in DTW-I, each channel is treated independently with its own warping path. The authors demonstrate the relevance of each variant through real-world examples of RGB images (i.e., matrices with three channels). Namely, the authors address the case of uneven color fading in manual illustrations in sixteenth-century manuscripts; a case where DTW-I is more appropriate as it treats each color separately. Conversely, in the case of photos that are uniformly faded to sun exposures or scanning artifacts, the authors show that DTW-D is more appropriate as it corrects for hue shifts across all channels.

Bagnall et al. [5] introduced the UEA archive, one of the largest collections of datasets for MTS classification. The work also includes an evaluation on Euclidean distance, DTW-D, and DTW-I, comparing the effects of a single normalization method as well as Nonorm. Unfortunately, due to the limited range of measures considered and the lack of comprehensive statistical testing, it is challenging to draw general conclusions from this study about MTS distance measures as a whole.

Shifaz et al. [104] argued that the channel-dependency extension strategies are also applicable to elastic measures besides DTW. They extended four additional measures to MTS and conducted evaluations on the UEA archive. Similar to [105], the authors of [104] concluded that there exists no superior measure or extension strategy that consistently outperforms others.

We challenge these findings by exploring a broader range of measures, normalizations, and downstream tasks, and by considering parameter tuning to ensure comparison of measures in their best possible form. Particularly, our study (a) considers measures from seven categories rather than only lock-step and elastic measures, (b) considers 13 normalization methods (including 5 novel ones) rather than one, (c) evaluates measures on clustering, anomaly detection, and classification, and (d) analyzes results in a principled manner, with both pairwise and grouped comparisons, supported by statistical tests. As such, the study provides a robust evaluation with a more complete view of the full landscape of MTS distance measures (cf. Table 1). Further details of our findings are presented in Section 5, with the resulting guidelines detailed in Section 6.

3 Primer on multivariate time series

We will first establish the necessary background and describe the three axes that will form a structured framework for our comparison of multivariate distance measures; (a) normalization; (b) temporal models; and (c) channel-dependency models. Normalization is a critical preprocessing step, which corrects for distortions in the form of scale imbalances and offsets. As we will show in Section 5.1.1, the choice of normalization methods is non-trivial, as it can have notable impact on the accuracy of distance measures. The temporal model and channel-dependency model, as fundamental properties of MTS distance measures, serve as the basis for our measure categorization. By incorporating these two models, our study enables evaluation at both the level of individual measures and the level of models. As such, the results can ultimately aid the design of future measures and the choice between existing ones. We will now establish some key terminology, definitions, and data assumptions that will be used throughout the paper. We then introduce the three axes and their role in the comparison of MTS.

3.1 Background

We consider a MTS as a collection of C UTS, or *channels*, i.e., a real-values matrix with C rows; $\boldsymbol{X} = [X^{(1)}, X^{(2)}, \ldots, X^{(C)}] \in \mathbb{R}^{C \times T}$, where each row $X^{(i)} = [X_1^{(i)}, X_2^{(i)}, \ldots, X_T^{(i)}]$ is an ordered sequence with length T. Accordingly, we consider an MTS dataset as a set of n matrices $\mathcal{D} = [\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_n] \in \mathbb{R}^{n \times C \times T}$. That definition implies that we consider each MTS to have the same number of channels, and each channel to have the same length. Following standard practices [4, 5, 34, 88, 104, 112], we consider the sampling rates of all time series the same, and therefore, time stamps are omitted from the notation. To ensure equal lengths and no missing values for all MTS, certain preprocessing steps were applied to the data (i.e., resampling and imputation), as detailed in Section 5, in order to create a standardized test bed for our evaluation, consistent with [5].

3.2 Axis #1: Normalization

While resampling and imputation are common preprocessing steps to ensure equal lengths and no missing values, normalization is a critical preprocessing step to mitigate distortions caused by differences in scale, variance, or offsets. As for all distortions, these differences can hinder the identification of similar patterns between time series and are therefore crucial to address before further analysis. To illustrate the relevance of normalization, consider two examples: (a) two stocks following similar price patterns but trading at different volumes; and (b) two recordings of the same song but played at different octaves. Directly comparing the raw values in these cases could lead to key issues: the first case suffers from a difference in scale, while the second case has a difference in offset. Although time series samples share similar patterns that are likely relevant for several tasks (e.g., portfolio management or song recognition), they can still produce large dissimilarity scores due to the illusion of distortion. This highlights the need for normalization.

Compared with the UTS case, the extra channels of MTS introduce new challenges. On the one hand, when considering channels generated by different devices or entities, the nature of the distortions can vary between channels, requiring different corrections to be applied to each channel independently. On the other hand, this independent normalization strategy overlooks the interdependencies between channels and may lead to suboptimal performance (further discussed in Section 3.4). Despite the extensive studies on the normalization of UTS [88], multivariate-specific normalization techniques remain unexplored. Particularly, existing works on multivariate distances have (at best) only considered applying Z-score normalization on each channel independently [4, 105], without regard for other methods such as Min-Max scaling or normalization based on the entire MTS (e.g., normalize with the global mean and standard deviation). Here, we consider a wide of normalization techniques, and extend them to the multivariate case with two strategies. We introduce the considered techniques in the context of UTS here, and explain their extension to the multivariate case in Section 3.4. In the context of UTS, several normalizations have been proposed throughout the years, each focusing on different types of distortions in the data. We briefly name 8 normalization techniques here, but refer to [88] for a more detailed description: (a) Z-score (b) Min-Max (Minmax); (c) Mean (d) Median (e) Unit-length (Unit) (f) Adaptive scaling (Adaptive) [27]; (g) Sigmoid and (h) Hyperbolic tangent normalization (Tanh). Additionally, we consider not normalizing the data, which we refer to as *Nonorm*.

3.3 Axis #2: Temporal Models

The temporal model of a time series distance measure defines how the time dimension of data is handled in the comparison. This involves the strategy of handling temporal distortions (e.g., by aligning the time steps in the original space or constructing a new representation to address the distortion). In contrast to resampling, imputation, and normalization, temporal misalignments

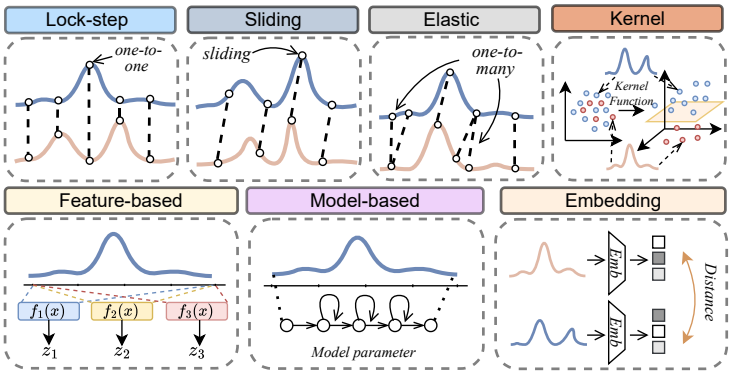


Fig. 2. Visualization of the seven temporal models.

are "pairwise distortions;" they only arise in the context of comparing two time series, and are therefore handled on the fly by the distance measure during the comparison. As such, the temporal model is a key property of a distance measure, and can be used as the basis for categorization. Furthermore, as these concepts only regard the time dimension, they are not specific to univariate or multivariate distance measures. For consistency, we therefore introduce the temporal models in the context of UTS. The authors of [88] categorize UTS distances into five families of measures; *lock-step*, *sliding*, *elastic*, *kernel*, and *embedding* measures. We adopt this categorization and extend the taxonomy with two additional families, *feature-based* and *model-based* measures, which reflect recent innovations in the field. Additionally, we introduce the concept of *ensemble* measures, which combine the distance scores from multiple base measures to improve the accuracy and robustness of the comparison. At a high level, lock-step, sliding, and elastic measures focus on handling temporal distortions in the original space, differing in their freedom to align time points; Kernel, feature-based, model-based, and embedding measures focus on implicitly or explicitly transforming the time series to handle distortions. We will review the detailed information in Section 4. The taxonomy of temporal models is visualized in Figure 2.

3.4 Axis #3: Channel-dependency models

In the multivariate case, questions may arise whether the transformations of normalizations and temporal models should be derived and performed on the whole MTS, or individually per channel. For example, in the case of sliding measures, should the optimal shift be derived across channels or can each channel be shifted independently? This question holds for all normalizations and temporal models, and determines how measure-specific concepts like scale, optimal shift and warping paths should be derived and applied. The answer to this question is defined as a *channel-dependency model*, and serves as a property of distance measures and normalizations for MTS. We identify two channel-dependency models; *channel-independent* and *channel-dependent*, and provide their realizations for each normalization and temporal model below.

Channel-independent model. The channel-independent model involves treating the channels as independent UTS, normalizing and computing distances channel-by-channel. For normalization, this implies that statistics used in the normalization are computed for each channel independently. For the temporal model, this implies that temporal distortions are handled independently over channels, and that representations are derived for each channel. Then, the distances between channels are aggregated to form a final distance.

Channel-dependent model. The channel-dependent model involves treating the channels as a single entity, applying normalization and computing distances over all channels simultaneously.

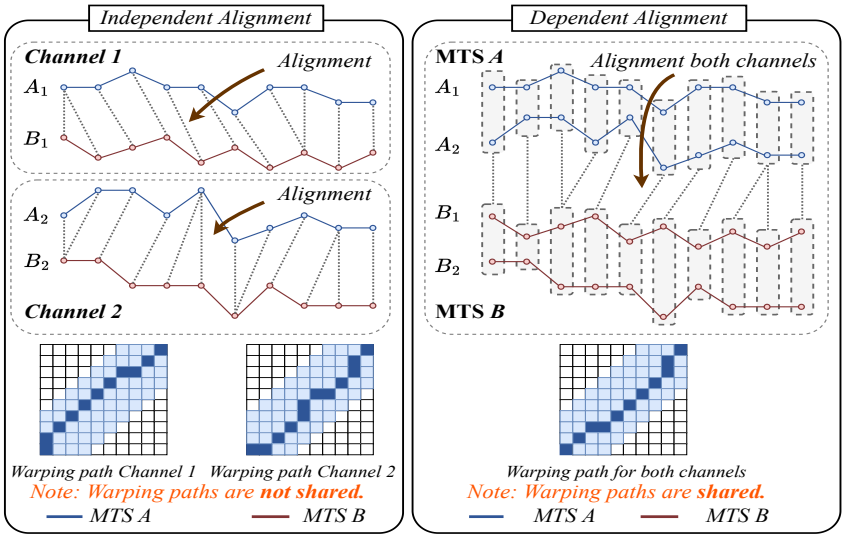


Fig. 3. Visualization of the alignment and resulting warping paths when utilizing either a channel-independent (left) or channel-dependent (right) extension of DTW.

This essentially implies that distortions – either temporal or in scale – are assumed to be shared across all channels. For example, the dependent version of elastic measures aims to find one global warping path for an entire MTS, rather than individual warping paths for each channel (Figure 2). In contrast to channel-independent extensions, the channel-dependent extension of measures is specific to its temporal model. In the next section, we will demonstrate how to extend a representative measure for each temporal model to be channel-dependent. For normalization methods, the channel-dependent extension involves rescaling time series based on global statistics rather than channel-specific ones. For example, channel-dependent Z-score involves computing the mean and standard deviation over all channels, and subtracting and dividing each channel by these values. Note that this model is not applicable to all normalizations; for example, Sigmoid normalization includes no such statistics so it is inherently channel-independent. Only the following methods have a channel-dependent extension: Z-score, Min-Max, Mean, Median, and Unit-length.

Channel weighting. An additional challenge with multivariate data is controlling the influence of each channel. This can be relevant in two scenarios; (a) when channels differ in scale, the distance score might be biased towards specific channels, and (b) when some channels are more important than others to the task at hand (e.g., only channel is predictive of the class label). Case (a) is a matter of normalization; both channel-dependent and channel-independent normalizations can rescale channels to control their impact on the distance. Case (b) is a matter of the distance measure itself; all temporal models and channel-dependency models provide the conceptual freedom to weigh channels differently in the distance computation. This, however, always requires some form of supervision, either by learning weights through ground-truth training data as done by some embedding measures [41, 115, 118], or through the input of a domain expert, which falls beyond the scope of this study and is therefore not considered.

4 Multivariate Distance Measures

In this section, we provide more detailed definitions of each temporal model and present an overview of the measures to be evaluated in Section 5. In view of space, we focus on introducing representative measures to illustrate the extension of univariate distance measures to a multivariate

setting. We only discuss temporal models in the context of the *channel-dependent* model. The channel-independent versions of measures can be derived by applying the univariate distance measure to each channel individually, and summing the resulting distances.

Lock-step measures, compare values at corresponding time points between two UTS and aggregate these differences across time. The lack of temporal alignment makes lock-step measures applicable to cases where temporal distortions are unlikely to exist in the time series. Due to the inherent properties of lock-step measures, the channel-independent and channel-dependent extensions are identical. To illustrate the concept of multivariate lock-step measures, we provide an example formula for the L_p distance, defined as:

$$L_p(\mathbf{X}, \mathbf{Y}) = \sqrt[p]{\sum_{i=1}^{C}\sum_{j=1}^{T} |X_j^{(i)} - Y_j^{(i)}|^p}. \tag{1}$$

In our evaluation, we selected representative measures that demonstrated strong performance in the univariate case [88], including Euclidean, L_1, Lorentzian, $L_{1,avg,\infty}$, Canberra, Chord, Clark, Emannon4, Jaccard, Soergel, and Topsoe, as detailed in [88].

Sliding measures, shift one time series relative to another to find the alignment that minimizes distance. To extend sliding measures to the multivariate case in a channel-dependent manner, a single optimal shift is derived and applied uniformly across all channels. To illustrate this concept, we consider the Shape-based Distance (SBD) [83]. The channel-dependent SBD (SBD-D) involves finding the optimal shift w only along the time dimension that maximizes the 2D Normalized Cross-Correlation (NCC2) between $\mathbf{X}$ and $\mathbf{Y}$, and substracts that from 1. The computation of NCC2 can be accelerated using 2D Fast Fourier Transform (FFT2) and Inverse FFT (IFFT2) and, thus, the formula can be expressed in the following form:

$$NCC2(\mathbf{X}, \mathbf{Y}) = \frac{IFFT2(FFT2(\mathbf{X}) * FFT2(\mathbf{Y}))}{||\mathbf{X}|| \cdot ||\mathbf{Y}||} \tag{2}$$

$$SBD\text{-}D(\mathbf{X}, \mathbf{Y}) = 1 - max_w(NCC2_w(\mathbf{X}, \mathbf{Y})) \tag{3}$$

where $*$ represents taking the complex conjugate in the frequency domain. Sliding measures are particularly useful in cases where time series can suffer from phase shifts. In our evaluation, we focus exclusively on SBD [83] as the representative sliding measure due to its highly competitive performance in the evaluation of univariate measures [88].

Elastic measures handle temporal distortions in time series by deriving a non-linear mapping (i.e., alignment) between time points, minimizing the distance between the aligned series. The mapping of each time point makes elastic measures very applicable to cases where the temporal distortions in time series can be complex, i.e., when individual readings can be delayed, repeated, or missing. With the channel-dependent extension, these mappings are constructed simultaneously for all channels at each time point, using three atomic operations: diagonal, vertical, and horizontal movements. Each atomic operation incurs a specific cost, denoted as c^D, c^V, and c^H for diagonal, vertical, and horizontal movements, respectively. The optimal alignment is determined using dynamic programming, with the total alignment cost (i.e., distance) being the sum of the selected movements. Formally, the alignment cost up to time points i and j for two MTS is defined as:

$$Cost(i, j) = \min \begin{cases} Cost(i-1, j-1) + c^D & diagonal \\ Cost(i-1, j) + c^V & vertical \\ Cost(i, j-1) + c^H & horizontal \end{cases} \tag{4}$$

where the border conditions are specific to the individual measures. The total alignment cost is then defined as $Cost(T_1, T_2)$, with T_1 and T_2 the lengths of the two time series. Elastic measures

vary in the cost functions assigned to each movement type and the constraints they impose on the alignment process. For instance, in DTW, the cost is calculated as the squared Euclidean distance between corresponding data points, and the difference between DTW-I and DTW-D is displayed in Figure 3. In our evaluation, based on elastic measures' performance in [88], we consider the five representative measures: DTW [101], Longest Common Subsequence (LCSS) [110], Edit Distance with Real Penalty (ERP) [23], Time Warp Edit (TWE) [72], and Move-Split-Merge (MSM) [106].

Kernel measures implicitly map the raw time series data to a higher-dimensional space through using a kernel function. As such, the extension of kernel measures differs according to the base measures on which the kernel functions are applied. For example, the dependent extension of Shift Invariant Kernel (SINK) relies on the channel-dependent version of NCC (NCC2). In this work, we consider four kernel measures: one using a lock-step base (Radial Basis Function (RBF) [30]), two using an elastic base (global alignment kernels (GAK) [31], Kernel Dynamic Time Warping (KDTW) [71]), and one using a sliding base (SINK [82]).

Feature-based measures transform a raw time series to a feature vector based on pre-defined features, e.g., mean, variance, slope, entropy, etc. Feature-based measures differ in the features used to describe the MTS and the distance measure used to compare the feature vectors. In this work, we consider two widely adopted feature sets: (a) the CAnonical time series CHaracteristics set (catch22) [70], and (b) the TSFresh feature set [26]. We also propose two feature-based methods accordingly, referred to as Catch22 and TSFresh, which both use the Euclidean distance to compare the feature vectors of two MTS. While conceptually possible, both methods were initially proposed for UTS, lacking a straightforward extension to incorporate multivariate features (e.g., capturing inter-channel correlations). Therefore, they are inherently channel-independent. Developing new features to create a channel-dependent variant is beyond our scope as it requires in-depth understanding of feature contributions to downstream tasks [46, 47].

Model-based measures construct probabilistic models to represent time series – either a single model for the entire MTS in channel-dependent cases or separate models for each channel in channel-independent cases – which are then used as proxies for computing distances. Model-based measures differ in the model used to represent the MTS, and the measure for comparing the models. In this work, we consider two types of models: Multivariate Gaussian Models (Gauss) and Hidden Markov Models (HMM) [8], and compare them using the Kullback-Leibler (KL) divergence as a measure, which has a closed form for Gaussian models and can be approximated for HMMs [43].

Embedding measures project the time series into a latent space for capturing the most important characteristics, allowing for the calculation of dissimilarity. Embedding measures differ in their choice of the embedding method. For example, the extension process of Generic Representation Learning (GRAIL) [82] directly depends on the multivariate SINK, its core kernel function for representation learning. In this work, we consider and extend five representative embedding measures: TS2Vec [118] and TLoss [41], two deep learning-based embedding methods, GRAIL [82], the best-performing embedding method in [88], and PCA similarity factor (D_{PCA}) [115] and Eros (D_{Eros}) [115], two embedding methods based on PCA projections that are inherently channel-dependent. TS2Vec and TLoss were transformed to distance functions by taking the Euclidean distance over the embeddings obtained by the encoder-based models.

Ensemble measures combine the distance scores of multiple measures to improve robustness. They cannot be classified as a temporal model, but rather as a *"meta-model"* that combines the strengths of individual measures. This is achieved by aggregating (e.g., averaging) the distance scores of the individual measures, optionally with weights assigned to each measure. As the distances scores of different measures can have varying ranges and distributions, distances are

Table 2. Parameter spaces for MTS distance measures. d denotes the number of input channels for MTS comparison. Specifically, $d = C$ for channel-dependent measures (using all channels), and $d = 1$ for channel-independent measures.

Distance Measure	Parameter Range
DTW	$\delta \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 100\}$
LCSS	$\delta \in \{5, 10, 20, 50, 100\}$ $\epsilon \in \{0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1.0, 2.0\} * \sqrt{d}$
MSM	$\mathbf{c} \in \{0.01, 0.1, 1, 10, 100, 0.5, 5, 50, 500\} * \sqrt{d}$
TWE	$\lambda \in \{0, 0.25, 0.5, 0.75, 1\} * \sqrt{d}$ $\nu \in \{0.00001, 0.0001, 0.001, 0.01, 0.1, 1.0\}$
SINK	$\gamma \in \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 20, 100, 1000\}$
GAK	$\gamma \in \{0.01, 0.05, 0.1, 0.25, 0.5, 0.75, 1, 5, 10, 15, 20\}$
KDTW	$\gamma \in \{2^{-15}, 2^{-14}, 2^{-13}, 2^{-12}, 2^{-11}, 2^{-10}, 2^{-9}, 2^{-8}, 2^{-7}, 2^{-6}, 2^{-5}$ $2^{-4}, 2^{-3}, 2^{-2}, 2^{-1}, 2^{0}\}$
RBF	$\gamma \in \{-1, 1\}$
GRAIL	$\gamma \in \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 100, 1000\}$
TS2Vec	Embedding size $\in \{40, 80, 160, 320, 640\}$
TLoss	Embedding size $\in \{40, 80, 160, 320, 640\}$
D_{PCA}	$\sigma^2_{\text{covered}} \in \{50\%, 60\%, 70\%, 80\%, 95\%, 100\%\}$

normalized before aggregation. In this work, we consider two ensembles: SBD-D + DTW-I and SBD-D + MSM-I, with Minmax normalization applied to the distance scores before averaging.[1]

Time complexities: The differences in the temporal models' approaches to handling temporality and distortions lead to varying computational complexities. Namely, lock-step measures, which process each value in the MTS only once, have a linear complexity with respect to the number of channels and time steps, $O(CT)$, making them the most efficient. Sliding measures utilize FFTs to compute distances efficiently, resulting in a complexity of $O(CT \log T)$ for SBD-I and $O(CT \log CT)$ for SBD-D. For elastic measures, computation of the optimal time-alignment worst-case requires to compare all pairs of time points, resulting in a quadratic complexity of $O(CT^2)$ for both channel-dependent and channel-independent versions. Kernel measures and ensembles inherit the complexities of their base measures. The complexities of feature-based, model-based, and embedding measures are highly dependent on the underlying methods used to extract features, fit models, or learn representations, respectively; and thus do not have a general complexity. Empirical complexities are investigated and presented in Section 5.4.

5 Evaluation

The purpose of our evaluation is threefold: (a) to evaluate the discriminative power of individual measures; (b) to analyze what normalization methods, temporal models, and channel dependency models are best suited for MTS; and (c) to determine whether previous findings on UTS also hold for the MTS case [88]. With this evaluation, we aim to lay the foundation for future research, paving the way for a practical handbook to guide the selection of MTS measures for downstream tasks.

Datasets: We utilize the *UEA archive* for experiments on classification and clustering. The UEA archive is the largest set of labeled datasets for MTS classification, comprising 30 labeled real-world datasets from diverse domains, with varying number of channels and lengths [5]. We adopt the predefined train-test splits, which have been widely used in MTS tasks [4, 74, 100, 104, 105]. Following the author's recommendation [5], we resample shorter time series to match the length of the longest for each dataset, and impute missing values using linear interpolation for MTS

[1]The min and max distance scores are derived over a set of training MTS.

comparison. For the task of anomaly detection (AD), we use the TSB-AD-M archive [67], which is one of the largest collections of datasets for AD on MTS, consisting of 200 real-world time series with labeled anomalous time points and varying lengths and channels.

Distance Measures: We consider the MTS measures listed in Section 4, and indicate the channel-dependency model of each measure by adding a suffix to the measure name. For example, *SBD-D* and *SBD-I* denote the channel-dependent and independent variants of multivariate SBD, respectively.

Evaluation framework: In line with prior works [5, 88, 104], we evaluate the performance of distance measures on classification, clustering, and anomaly detection using kNN-based algorithms (i.e., 1NN classifier for classification, k-means for clustering, and a 1NN detector for AD), which are well-suited to this evaluation for three key reasons [34, 88]: (a) they reflect a similar workflow to time series similarity search tasks, the primary application of distance measures [37]; (b) they are parameter-free and straightforward to implement; (c) they solely depend on the discriminative power of distance measures. Our study includes a *parameterization phase*, where we determine the best parameters for each distance measure and dataset, and a *performance phase*. The performance phase consists of six analyses, three of which focus on the axes of MTS comparison in the context of *classification* (Section 5.1), two focus on the validation of findings in *clustering* and *anomaly detection* (Section 5.2-5.3), and one is a runtime analysis, where we focus on the accuracy-to-runtime trade-off of the distance measures (Section 5.4).

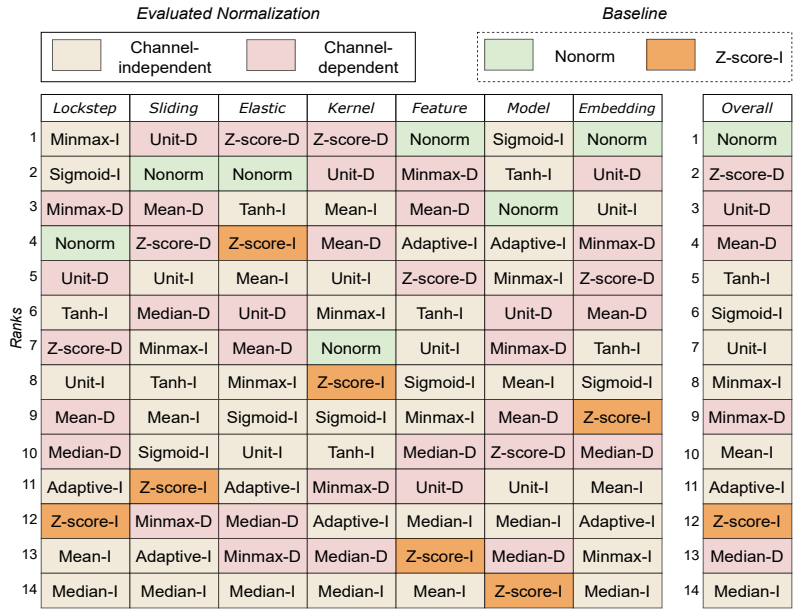
Parameterization: To ensure comparison of distance measures in their best possible form, we start by fine-tuning their parameters (where applicable) under two experimental settings: (i) supervised setting, where the optimal model parameters are selected on the training set of each dataset using Leave-One-Out Cross-Validation (LOOCV); (ii) unsupervised setting, we derive a default parameter setting for each distance measure by analyzing which single set of values performed *generally* well across all datasets. Our parameterization phase not only incorporates a broad parameter range, but it also accounts for both channel-dependency models for each measure when applicable. Furthermore, parameterization was performed across different normalizations, though optimal parameter values showed to be fairly similar across different normalizations. Note that for clustering and anomaly detection only the unsupervised setting is considered as both methods are inherently unsupervised tasks. The parameter ranges for all measures are shown in Table 2. The default parameters are presented along with the results (e.g., Table 5).

Statistical analysis: We assess the significance of the differences in classification accuracy between measures, normalization methods, and channel-dependency models using two widely adopted statistical tests [4, 34, 88, 104]. Specifically, for pairwise comparisons, we employ one-sided Wilcoxon tests [113] with $\alpha = 0.05$, following the guidelines of [33]. For global comparisons, we apply the Friedman test [42] followed by a post-hoc Nemenyi test [76] with $\alpha = 0.1$ to determine the significance of differences (referred to as the Friedman-Nemenyi test). Such global comparisons are then visualized by critical diagrams (e.g., Figure 4b), which show the average rankings of the compared methods, with horizontal lines connecting methods that exhibit insignificant differences.

Platform and implementations: We conduct our experiments on a server equipped with 2xAMD EPYC 7713 64-Core processors and 1 TB RAM, running Ubuntu 22.04.3 LTS. All measures are implemented in Python 3.8.5. We use the HMMLEARN library [61] for HMMs, the SCIPY library [109] for PCA, and the SKTIME library [69] for feature-based measures. Experiments for runtime analysis are evaluated in a single-threaded fashion without use of acceleration libraries such as numba or Cython, excluding the data preprocessing time. All code used in this work is publicly available [1].

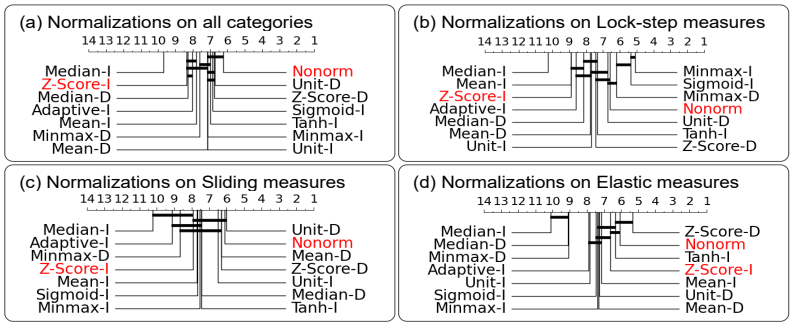
5.1 Task 1: Classification

The following present the evaluation of distance measures on classification. The analysis is organized into three subsections, each focusing on an axes of our taxonomy in Section 3.



	Lockstep	Sliding	Elastic	Kernel	Feature	Model	Embedding	Overall
1	Minmax-I	Unit-D	Z-score-D	Z-score-D	Nonorm	Sigmoid-I	Nonorm	Nonorm
2	Sigmoid-I	Nonorm	Nonorm	Unit-D	Minmax-D	Tanh-I	Unit-D	Z-score-D
3	Minmax-D	Mean-D	Tanh-I	Mean-I	Mean-D	Nonorm	Unit-I	Unit-D
4	Nonorm	Z-score-D	Z-score-I	Mean-D	Adaptive-I	Adaptive-I	Minmax-D	Mean-D
5	Unit-D	Unit-I	Mean-I	Unit-I	Z-score-D	Minmax-I	Z-score-D	Tanh-I
6	Tanh-I	Median-D	Unit-D	Minmax-I	Tanh-I	Unit-D	Mean-D	Sigmoid-I
7	Z-score-D	Minmax-I	Mean-D	Nonorm	Unit-I	Minmax-D	Tanh-I	Unit-I
8	Unit-I	Tanh-I	Minmax-I	Z-score-I	Sigmoid-I	Mean-I	Sigmoid-I	Minmax-I
9	Mean-D	Mean-I	Sigmoid-I	Sigmoid-I	Minmax-I	Mean-D	Z-score-I	Minmax-D
10	Median-D	Sigmoid-I	Unit-I	Tanh-I	Median-D	Z-score-D	Median-D	Mean-I
11	Adaptive-I	Z-score-I	Adaptive-I	Minmax-D	Unit-D	Unit-I	Mean-I	Adaptive-I
12	Z-score-I	Minmax-D	Median-D	Adaptive-I	Median-I	Median-I	Adaptive-I	Z-score-I
13	Mean-I	Adaptive-I	Minmax-D	Median-D	Z-score-I	Median-D	Minmax-I	Median-D
14	Median-I	Median-I	Median-I	Median-I	Mean-I	Z-score-I	Median-I	Median-I

(a) Meta ranking of normalization methods.



(b) Ranking of normalization methods based on the average of their ranks across measures and datasets.

Fig. 4. Meta-analysis of normalization methods

5.1.1 Normalization methods

In this section, we consider 13 unique normalizations, along with Nonorm, and assess their effect on the classification accuracy of multivariate distance measures. Similar to distance measures, the channel-dependency model of each normalization is indicated by a suffix (e.g., Z-score-I and Z-score-D). In UTS literature, Z-score normalization has long been regarded as the optimal choice [2, 40, 45]. Paparrizos et al. [88] debunked this misconception for the UTS case. However, previous evaluations of multivariate distance measures have limited exploration of the impact of normalization methods. To address this gap, we extend the analysis of [88] to the multivariate case, conducting a meta-analysis to evaluate whether their findings on normalization apply to MTS as well.

Figure 4a presents the average ranks of normalization methods based on the classification accuracies across measure-dataset combinations. The results indicate that Z-score-I – the natural extension of the popular UTS normalization – consistently underperforms compared to other

normalization methods across the temporal models; it ranks 12th in the overall meta-ranking and ranks 8th or lower for all temporal models except elastic measures. Our findings confirm and extend the conclusion of [88] that Z-score is not the best normalization method, showing that this also holds true for the multivariate case. Additionally, we observe that Nonorm consistently outperforms other normalization methods; ranking first overall and placing in the top four across all temporal models except kernel measures. This indicates that normalization methods do not always offer improvements over the natural baseline of no normalization, as also observed in previous studies on the UEA archive [100, 104]. In those studies, this finding was interpreted as a sign that the scales and variances of channels serve as discriminatory factors in classification tasks, and that normalization removes this information. While we recognize the validity of this hypothesis, we now also show that channel-dependent normalizations – that should better preserve this information – also do not significantly improve upon Nonorm. This suggests that the loss of channel-specific information is not the only factor at play, and that the existing normalization methods themselves (i.e., designed for UTS) may not be well-suited for MTS data.

To gain deeper insight into the performance of normalization methods, we evaluate the statistical significance of the differences between the methods using the Friedman-Nemenyi test. The results (Fig. 4b) reveal that no single method or group of methods performs statistically better than the others, neither on a global level, nor when considering specific temporal models. This, combined with the fact that Nonorm ranks first overall, suggests that current normalization methods, which are direct extensions of methods for UTS, do not significantly impact classification accuracy for most temporal models. This demonstrates that existing normalization approaches do not generalize well to the multivariate case, and *highlights the need for new MTS-specific normalizations* that better accommodate the multivariate nature of the data. In the current absence of those methods, we will use Nonorm as the default for the remainder of our analyses.

5.1.2 Temporal models

The objective of the next round of experiments is (a) to identify the best-performing measure within each temporal model, but also (b) to compare the performance of these temporal models as a whole. To achieve this, we employ an approach that starts with lock-step measures, comparing each to a baseline, and based on the results, decide whether to update the baseline before proceeding to the next temporal model. This method enables a dynamic evaluation, continuously refining comparisons with a reasonable baseline to achieve accurate and reliable results. We begin with Euclidean distance as the baseline, since it is the most widely used measure in the literature [88].

5.1.2.1 Lock-step measures. Table 3 presents the pairwise comparison of lock-step measures under Nonorm based on the Wilcoxon test, using Euclidean distance as the baseline, and their order is determined by the average ranks across datasets. The results show that three lock-step measures, $L_{1,avg,\infty}$, L_1, and Lorentzian, significantly outperform the baseline. This confirms and extends a key finding from [88], that Euclidean distance is not necessarily the best lock-step measure in the multivariate case, and that $L_{1,avg,\infty}$, L_1, and Lorentzian offer superior alternatives. To identify the best-performing lock-step measure, we recompute the average ranks for the top-performing measures, and assess their significance using the Friedman-Nemenyi test. The results (Fig. 5a) show (a) additional evidence that $L_{1,avg,\infty}$ and Lorentzian significantly outperform Euclidean, and (b) that Lorentzian ranks the highest, making it the new baseline for subsequent comparisons.

5.1.2.2 Sliding measures. Table 4 presents the pairwise comparison of sliding measures with Lorentzian, all under Nonorm. We observe that only SBD-D significantly outperforms Lorentzian; SBD-I shows better performance on most datasets but without reaching statistical significance.

Table 3. Pairwise comparison of lock-step measures with Euclidean under Nonorm, with measures ordered by their average rank across datasets. ✓, ✗, and ≈ indicate significantly better, worse, or equal performance compared to the baseline based on the Wilcoxon test ($\alpha = 0.05$). ">", "=", and "<" indicate how many datasets the given combination performs better, equal, or worse than the baseline, respectively.

Measure	Diff	Average Accuracy	>	=	<
$L_{1,avg,\infty}$	✓	0.6321	18	4	8
Lorentzian	✓	0.6334	20	2	8
Jaccard	≈	0.6507	15	7	8
L_1	✓	0.6321	21	2	7
Chord	≈	0.6261	15	1	14
Topsoe	✗	0.5589	11	0	19
Soergel	✗	0.4153	9	1	20
Clark	✗	0.4462	8	0	22
Canberra	✗	0.3104	8	1	21
Emanon4	✗	0.3246	6	0	24
Euclidean	≈	**0.6264**	**0**	**30**	**0**

Table 4. Pairwise comparison of sliding measures with Lorentzian (Nonorm). See Table 3 for descriptions.

Measure	Diff	Average Accuracy	>	=	<
SBD-D	✓	0.6817	19	0	11
SBD-I	≈	0.6520	17	0	13
Lorentzian	≈	**0.6334**	**0**	**30**	**0**

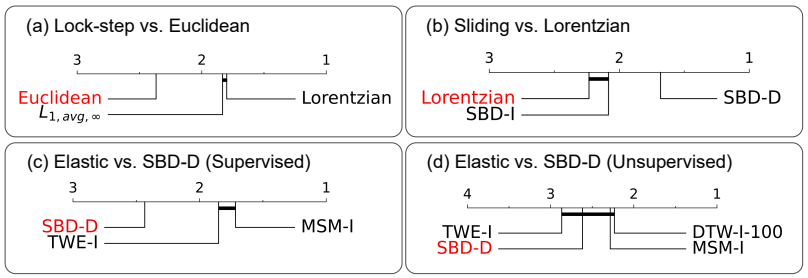


Fig. 5. Ranking of lock-step, sliding, and elastic measures under Nonorm based on the average of their ranks across datasets.

These findings are validated through a Friedman-Nemenyi test (Fig. 5b), where both sliding measures rank higher than Lorentzian, with SBD-D showing statistical significance. These findings demonstrate that SBD-D's ability to address global misalignments can increase classification accuracy. Moreover, the superior performance of SBD-D over SBD-I suggests that the additional flexibility in temporal alignment provided by SBD-I does not always lead to better accuracy and may result in overly liberal alignments. We explore this difference between channel-dependency models further in Section 5.1.3. As SBD-D outperformed the previous baseline, Lorentzian, SBD-D will be used as the new baseline for the following experiment on elastic measures.

5.1.2.3 Elastic measures. We now present a comparison of elastic measures, both in supervised and unsupervised settings, with SBD-D as the baseline on a downsampled version of the UEA archive. The reason for downsampling the original archive is that variants of MSM and TWE were very time-consuming; the classification process exceeded server's time limit of 7 days on 7

Table 5. Pairwise comparison of elastic measures against SBD-D (Nonorm). See Table 3 for descriptions.

Measure	Parameter Tuning	Diff.	Average Accuracy	>	=	<
MSM-I	LOOCV	✓	0.6573	20	2	8
	$c = 0.5$	≈	0.6525	18	0	12
DTW-I	LOOCV	≈	0.6570	16	3	11
	$\delta = 100$	≈	0.6529	16	2	12
	$\delta = 10$	≈	0.6413	15	2	13
TWE-I	LOOCV	✓	0.6561	21	2	7
	$\lambda = 0.5, \nu = 0.01$	≈	0.6325	13	1	16
MSM-D	LOOCV	≈	0.6384	13	3	14
	$c = 0.5 * \sqrt{d}$	≈	0.6200	15	2	13
DTW-D	LOOCV	≈	0.6296	13	1	16
	$\delta = 100$	≈	0.6276	12	2	16
	$\delta = 10$	≈	0.6237	12	2	16
ERP-D	$\delta = 100$	≈	0.6301	14	2	14
LCSS-I	LOOCV	≈	0.6385	16	0	14
	$\delta = 5, \epsilon = 1.0$	✗	0.5456	6	0	24
TWE-D	LOOCV	≈	0.6298	13	2	15
	$\lambda = \sqrt{d}, \nu = 0.0001$	✗	0.6184	13	3	14
ERP-I	$\delta = 100$	≈	0.6400	12	3	15
LCSS-D	LOOCV	✗	0.5969	9	3	18
	$\delta = 10, \epsilon = 0.5 * \sqrt{d}$	✗	0.4981	5	3	22
SBD-D	-	-	**0.6496**	**0**	**30**	**0**

different datasets. All datasets that had over 500 MTS in their training/testing size, 500 channels, or 500 time points per MTS, were downsampled in their respective dimensions, using stratified sampling, random sampling, and polyphase resampling [48] respectively. The results in Table 5 demonstrate that only MSM-I and TWE-I significantly outperform SBD-D *under supervised tuning*. This shows that SBD-D, being parameter-free, is a highly competitive baseline, with most elastic measures unable to outperform it in either the supervised or unsupervised setting. Furthermore, we conclude that (a) elastic measures are highly sensitive to parameter settings, with fixed values often not being optimal across all datasets, and that (b) the ability to handle local temporal distortions does not necessarily lead to improved accuracy. We also assess the significance of the differences when considering multiple elastic measures alongside the baseline. Figures 5c-d reinforce previous findings, showing that MSM-I and TWE-I outperform SBD-D significantly under supervised tuning, but not in the unsupervised setting. These findings align with those from the univariate case reported in [88], allowing us to both confirm and extend two key conclusions to the multivariate setting: elastic measures are not necessarily superior to sliding measures and alternative elastic methods can outperform DTW in the supervised setting. While tuning parameters is crucial for the performance of elastic measures, we stress that it substantially increases the computational load as it requires a grid search over the full parameter range. This cost comes on top of the already high theoretical complexity of elastic measures, as discussed in Section 4. This computational load may hinder the scalability and application of elastic measures on large-scale datasets. Further details are provided in the runtime analysis in Section 5.4. In view of this observation, we retain SBD-D as the running baseline for the next comparison.

5.1.2.4 Kernel measures. Table 6 shows the performance of kernel measures compared to SBD-D, evaluated on the downsampled UEA archive discussed in Section 5.1.2.3. We observe here that no measure is able to significantly outperform SBD-D; only GAK-D and SINK-D are able to match the performance of SBD-D with and without tuning. KDTW-I and RBF perform significantly worse, even with supervision. We further observe no statistical significance in a grouped comparison

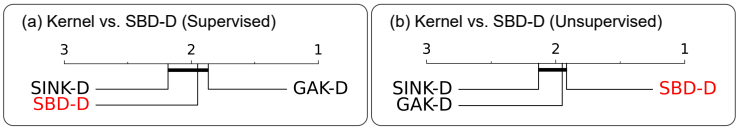


Fig. 6. Ranking of kernel measures under Nonorm across UEA datasets, using (a) supervised and (b) unsupervised tuning for their parameters

Table 6. Pairwise comparison of kernel measures under Nonorm normalization with SBD-D as a baseline. See Table 3 for column descriptions.

Measure	Parameter Tuning	Diff.	Average Accuracy	>	=	<
GAK-D	LOOCV	≈	0.6302	16	2	12
	$\sigma = 0.5$	≈	0.6140	14	2	14
SINK-D	LOOCV	≈	0.6482	10	3	17
	$\gamma = 5$	≈	0.6404	11	3	16
GAK-I	LOOCV	≈	0.6440	12	0	18
	$\sigma = 0.05$	✗	0.5950	6	2	22
KDTW-D	LOOCV	≈	0.6256	12	2	16
	$\sigma = 64$	✗	0.5769	8	2	20
SINK-I	LOOCV	≈	0.6270	9	3	18
	$\gamma = 5$	✗	0.6166	7	1	22
KDTW-I	LOOCV	✗	0.6142	9	0	21
	$\sigma = 32$	✗	0.5372	2	2	26
RBF	LOOCV	✗	0.5845	4	0	26
	$\gamma = -1$	✗	0.5757	3	0	27
SBD-D	-	-	**0.6496**	**0**	**30**	**0**

(Fig. 6), where GAK-D ranks slightly higher than SBD-D in the supervised setting but not in the unsupervised setting. These findings can be attributed to the sensitivity of kernel measures to their scaling parameter, which influences the similarity between data points in the kernel space and, consequently, impacts the classifier's decision boundaries. Naturally, the optimal scaling parameter is dataset-dependent, requiring tuning for good performance. This sensitivity also explains why kernel measures are generally paired with classifiers that adaptively learn these decision boundaries, such as Support Vector Machines (SVM) [29], rather than being used as a standalone measure with 1NN classifiers. Still, what is particularly surprising is that none of the measures significantly outperform SBD-D; not even those with an elastic foundation like GAK-D and GAK-I. This comes in contrast to the UTS cases, which showed that kernel measures – when properly tuned – were able to improve upon the performance of their base measures [88], e.g., KDTW versus DTW. This discrepancy may hint that the temporal alignment happening in these measures does not generalize well to the multivariate case, and either better temporal or channel-dependency models may be required for optimal performance. We leave this for future investigation.

5.1.2.5 Feature-based, model-based, and embedding measures. Moving on to the final three temporal models, we again observe that these measures perform significantly worse than SBD-D. To prevent only comparing against an overly strong baseline, we report comparison of these measures to Euclidean distance on the downsampled archive in Table 7. From the results we observe that only GRAIL-D, TS2Vec-D, and TS2Vec-I significantly outperform Euclidean distance in both the supervised and unsupervised settings; all other measures except TLoss, GRAIL-I, and KL_{Gauss}-D perform significantly worse in the comparison. These observations are reinforced by the global comparisons in Figure 7, where most methods rank lower than Euclidean distance (though not significantly), with GRAIL, TLoss, and TS2Vec being the only exceptions. Still, the results also show

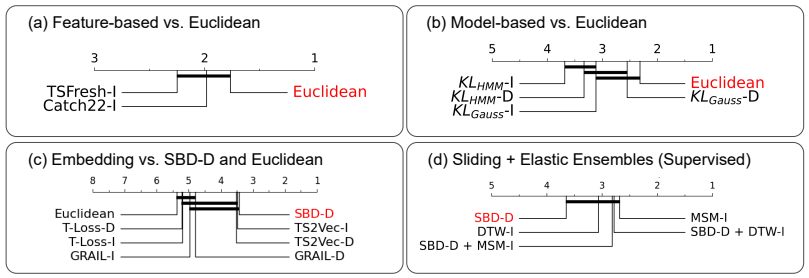


Fig. 7. Ranking of different measure families (Nonorm) based on the average of their ranks across datasets.

Table 7. Pairwise comparison of representation-based measures under Nonorm with Euclidean as a baseline. See Table 3 for column descriptions.

	Measure	Parameter Tuning	Diff.	Average Accuracy	>	=	<
Embedding	TS2Vec-I	LOOCV	✓	0.6538	21	2	7
		Embedding length = 320	✓	0.6409	19	3	8
	TS2Vec-D	LOOCV	✓	0.6559	23	0	7
		Embedding length = 320	✓	0.6454	19	1	10
	GRAIL-D	LOOCV	✓	0.6371	16	2	12
		$\gamma = 2$	✓	0.6337	18	0	12
	GRAIL-I	LOOCV	≈	0.6144	15	6	9
		$\gamma = 2$	✓	0.6241	18	2	10
	TLoss-I	LOOCV	≈	0.6159	13	1	16
		Embedding length = 320	≈	0.6088	12	0	18
	TLoss-D	LOOCV	≈	0.6129	13	0	17
		Embedding length = 320	≈	0.6038	12	1	17
	D_{Eros}	-	✗	0.4842	10	2	18
	D_{PCA}	LOOCV	✗	0.2793	7	0	23
		Covered Variance = 95%	✗	0.3031	10	1	19
Feat.	Catch22-I	-	✗	0.4977	10	4	16
	TSFresh-I	-	✗	0.4102	10	2	18
Model	KL_{Gauss}-D	-	≈	0.5191	10	1	19
	KL_{Gauss}-I	-	✗	0.4322	11	1	18
	KL_{HMM}-D	$h = 2$	✗	0.4164	9	1	20
	KL_{HMM}-I	$h = 2$	✗	0.4406	8	0	22
	Euclidean	-	-	**0.5853**	**0**	**30**	**0**

that these measures still no not outrank SBD-D.

For embedding measures, the unexpectedly poor performance from the PCA-based measures (D_{PCA} and D_{Eros}) may be due to the fact that current PCA-based approaches in the MTS context primarily capture correlations between channels. When these correlations are not strongly indicative of class labels, they may lead to suboptimal performance. Looking at the deep-learning-based measures, we see that TS2Vec performs a lot better than TLoss. This ranking is in line with the results in the original paper [118], where TS2Vec outperformed TLoss on the UEA archive when paired with a supervised SVM classifier. However, it also shows their dependency on such classifiers to obtain the state-of-the-art performance reported in the original paper, as the obtained accuracies here are notably lower than those reported in [118] for both methods. Most importantly, TS2Vec, being one of the state-of-the-art encoder-based deep learning methods, is still worse than SBD-D in terms of performance (Figure 7), demonstrating that deep-learning-based measures do not necessarily outperform traditional measures in the MTS domain.

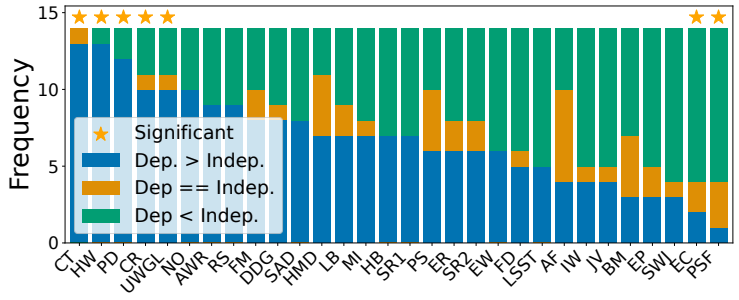


Fig. 8. Histogram of the number of measures whose channel-dependent variant outperforms the channel-independent variant, and vice versa, for each dataset. Stars indicate significant differences between the models according to a two-sided Wilcoxon test ($p < 0.1$).

As shown in Table 7 and Figure 7a, we observe the low performance of feature-based measures compared to Euclidean distance. This can be attributed to two key reasons: (a) Catch22 and TSFresh primarily focus on *global* properties of the time series, overlooking critical local information that the considered embedding measures do capture in their features, and (b) the feature-based measures included in this study consider only channel-independent features, neglecting channel-dependent characteristics. Further exploration of these two directions represents a potential future work.

For model-based measures, we note that the considered measures employ models that assume the time series to be i.i.d samples from a distribution, overlooking the temporal structure critical to time series classification. Namely, while HMMs can conceptually capture temporal dependencies, the limited number of hidden states (fewer than 5) likely restricts their focus to long-term trends, missing short-term patterns. We therefore conclude that while model-based approaches are promising, simple probabilistic models like Gaussian distributions and HMMs are not yet effective as demonstrated from this study. Future work should therefore investigate utilizing models that can better capture temporal patterns in time series.

5.1.2.6 Ensemble measures. Finally, we evaluate the ability of ensemble measures to improve upon the performance of their base measures. Motivated by the findings up to now, we combine SBD-D with MSM-I and DTW-I to investigate if we can surpass the current state-of-the-art performance of supervised MSM-I and unsupervised DTW-I. Our results (Fig. 7d) show that this is not the case; while combining SBD-D with DTW-I seems to improve upon the performance of DTW-I alone, the ensemble of SBD-D and MSM-I performs worse than MSM-I alone. This shows that ensembles are not guaranteed to improve upon the performance of their components, and thus that careful selection of the ensemble members is crucial. Furthermore, we stress that there currently exist no principled approach to aggregating distance scores; the current method of averaging and Minmax scaling is a simple example introduced for this study, though a lot of work remains to be done in this area to determine the optimal ensembling approach.

5.1.3 Channel-dependency models

We now investigate whether the channel-independent model delivers better performance than the channel-dependent model. As the impact of normalizations was shown to be insignificant, we limit ourselves here to the comparison of channel-dependent vs. channel-independent *measures* in this section. Comparison of dependency models for normalizations is left for future work, when normalizations are found that do significantly impact performance. Previous studies have consistently suggested that the optimal channel-dependency model is data-dependent, as it reflects the nature of distortions inherent to the dataset [104, 105]. To validate this claim beyond elastic

Table 8. Pairwise comparison of channel-dependent vs. independent model for different temporal models.

Measures	Diff.	>	=	<
All dependent	≈	180	42	175
Sliding-D	✓	19	3	8
Elastic-D	✗	60	14	76
Kernel-D	✓	53	10	27
Model-based-D	≈	33	6	21
Embedding-D	≈	38	8	44
Independent measures	≈	0	360	0

measures, we conduct a comprehensive analysis across all temporal models. Specifically, our methodology involves evaluating measures that have channel-dependent and channel-independent variants, and tallying the number of times each variant outperforms its counterpart for each dataset.

The results in Fig. 8 reveal that there are indeed datasets where the channel-dependent variant outperforms its counterpart for most measures (i.e., the left-most bars), and vice versa (i.e., the right-most bars). Furthermore, the star-highlighted bars indicate that the channel-dependent model significantly outperforms the independent model on six datasets, while the opposite is true for two datasets. The remaining datasets show no significant difference between the models. Based on our findings, we validate the claim that the choice of the channel-dependency model is data-dependent. Therefore, the optimal choice of channel-dependency model should be determined through training and validation, as suggested by the introduction of adaptive DTW (DTW-A) in [105].

To gain deeper insights into the two channel-dependency models, we perform a pairwise comparison between them for each temporal model. Table 8 shows that there is no statistically significant difference between the two models in general. However, we do observe significant differences at the level of temporal models. Specifically, for sliding and kernel measures, the dependent model significantly outperforms the independent model, whereas for elastic measures, the channel-independent model is significantly better. Furthermore, we observe that although no statistical difference is observed for model-based and embedding measures, the channel-dependent model outperforms its counterpart in at least half of the comparisons. These findings indicate that, in general, independent consideration of time series channels is advantageous only for elastic measures. This conclusion also aligns with our previous findings in Section 5.1.2. This discrepancy between elastic measures and the other temporal models can be attributed to the types of distortions that they aim to correct for. Sliding measures, for example, correct for global shifts between time series, which are typically caused by uncalibrated sensor arrays or different starting points of measurements. Consequently, those shifts are likely to be the same for all channels, favouring channel-dependent alignment in case such distortions are present. Elastic measures, on the other hand, correct for local distortions through time warping, which are generally caused by short-term events or measurement errors like sensor latency or jitter. The nature of such distortions makes that they are more likely to happen only in one or a few channels, and not across all channels, which makes that channel-independent alignment is more suitable and effective in these cases.

5.2 Task 2: Clustering

To extend our findings on classification to other tasks, we perform a study on clustering. For this experiment, we use Partitioning Around Medoids (PAM) [56], a clustering algorithm that iteratively updates k medoids[2], which are actual time series from the dataset, and assigns each time series to its closest medoid until convergence. The method is parameter-free besides the choice of k,

[2]The value of k is set equal to the number of unique classes in the respective dataset.

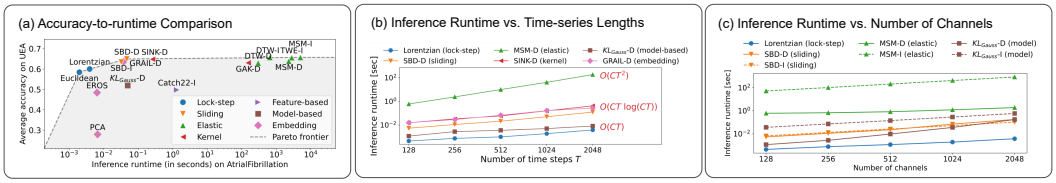


Fig. 9. Time efficiency analysis. (a) Accuracy-to-runtime comparison across all categories. Effect of (b) time series length and (c) the number of channels on the runtime of the top measures per category.

Table 9. Pairwise comparison of lock-step and sliding measures with Euclidean and elastic measures with SBD-D on the task of clustering on the UEA archive (Nonorm). See Table 3 for column descriptions.

Clustering Method	Parameters	Diff	Avg. RI	>	=	<
SBD-D	-	✓	0.7522	20	0	10
SBD-I	-	✓	0.7408	18	0	12
Lorentzian	-	≈	0.6979	15	3	12
Euclidean	-	≈	**0.6647**	**0**	**30**	**0**
DTW-D	$\delta = 100$	≈	0.7048	17	2	11
DTW-I	$\delta = 100$	≈	0.7384	17	0	13
ERP-I	$\delta = 100$	≈	0.7296	17	0	13
MSM-D	$c = 0.5 * \sqrt{d}$	≈	0.6785	14	0	16
MSM-I	$c = 0.5$	≈	0.7117	16	0	14
ERP-D	$\delta = 100$	≈	0.7001	16	0	14
TWE-I	$\lambda = 0.5, \nu = 0.01$	≈	0.7119	17	1	12
TWE-D	$\lambda = \sqrt{d}, \nu = 0.0001$	✗	0.6582	11	0	19
LCSS-I	$\delta = 5, \epsilon = 1.0$	✗	0.6638	8	0	22
LCSS-D	$\delta = 10, \epsilon = 0.5 * \sqrt{d}$	✗	0.6205	6	0	24
SBD-D	-	≈	**0.7522**	**0**	**30**	**0**

and performance directly depends on the distance matrix of the respective measure, making it an appropriate proxy to evaluate distance measures. In view of computational feasibility, we focus on the most prominent lock-step, sliding, and elastic measures under unsupervised parameter settings, as their comparisons constitute the main findings of our study so far and as clustering is inherently an unsupervised task. Furthermore, we conduct the evaluation on the UEA archive with Nonorm and report the average performance over 10 random initializations to mitigate variance.

Table 9 first presents the pairwise comparison of Lorentzian and SBD variants with Euclidean distance, based on the Rand Index (RI) [98] which measures the similarity between the cluster assignment of PAM and the ground truth of the dataset, with a value of 1 indicating perfect agreement and 0 indicating no agreement. The results reconfirm our finding that sliding measures with SBD-D significantly outperform lock-step measures. Lorentzian again outperforms Euclidean among the lock-step measures, though lacking statistical significance in this case. Furthermore, the comparison of elastic measures with SBD-D in Table 9 shows that SBD-D ranks first in terms of average performance, and that again no elastic measure significantly outperforms SBD-D with unsupervised parameters, which is consistent with the findings from Section 5.1.2.3. In conclusion, the study on clustering demonstrates the generalizability of our previous findings from the classification task.

5.3 Task 3: Anomaly detection

To further extend our findings to more tasks, we also perform a study on anomaly detection (AD), which involves computation of distances between MTS subsequences to identify anomalous periods in a time series. We consider the top-performing lock-step, sliding, and elastic measures under

Table 10. Pairwise comparison of lock-step, sliding, and elastic measures on the TSB-AD-M archive, using a 1NN anomaly detector. See Table 3 for column descriptions.

AD Methods	Diff	Avg. VUS-PR	>	=	<
Lorentzian	✗	0.4238	62	8	130
SBD-D	✗	0.3720	69	6	125
DTW-D	✗	0.3909	55	5	140
DTW-I	✗	0.3694	46	4	150
SBD-I	✗	0.2469	58	2	140
Euclidean	≈	**0.4413**	**0**	**200**	**0**

unsupervised parameter settings. MSM and TWE measures are excluded in this study for their extremely high computation cost (estimated 3-4 months of computation time). We employ 1NN detection as our AD algorithm [97], which involves computing for each w-length subsequence in an MTS, the closest w-length subsequence from that same MTS. The distance between them indicates the anomaly score for each subsequence, where high values indicate greater dissimilarity from the rest of the time series, suggesting a higher likelihood of an anomaly. In our experiment we use $w = 100$ and a stride of 50 to keep the experiment tractable for expensive distance measures.

Table 10 presents the average Volume Under Surface Precision-Recall (VUS-PR) [80] of the distance measures, and their pairwise comparison with Euclidean distance through the Wilcoxon test. We observe a clear discrepancy with the results on classification and clustering: here Euclidean distance significantly outperforms all other measures. The ranking of measures is almost perfectly inverse to the rankings observed for classification and clustering. While these results might seem counterintuitive, they are actually in line with expectations for AD. Namely, where it is imperative for classification and clustering to *ignore or correct for distortions*, AD actually focuses on *identifying distortions*, as these frequently correspond to anomalies. As lock-step measures do not address distortions, their distance score will be more sensitive to them, making it a key indicator to identify an anomaly. Sliding and elastic measures, on the other hand, correct for distortions in their distance computation, thereby losing critical information. These observations show the importance of considering the downstream task in measure selection for similarity search, and particularly, to what case of similarity search this task belongs: the case where distortions should be *corrected* for, or the case where distortions are the *target*. Lastly, we note that these results shed some light on measures' behavior on subsequences, but they do not signify subsequence search in general. Subsequence search is expected to follow the same rules as similarity search on whole time series, where the optimal choice of measure depends on the role of distortions in the downstream task. Unfortunately, as there currently exist no ground-truth datasets for MTS subsequence classification or clustering, this hypothesis cannot be tested as part of this study.

5.4 Runtime analysis

Up to this point, our analysis has focused exclusively on the discriminative power of the measures. However, in practical applications, the computational efficiency of these measures is equally critical. Figure 9a presents a comparative analysis of the accuracy-to-runtime performance of the top measures per category on the task of classification. The reported accuracies are averaged across all 30 datasets from the downsampled UEA archive [5] under the supervised setting, while the runtime represents the median inference time over five runs on the AtrialFibrillation dataset (to accommodate the slower methods). In line with the theoretical analysis in Section 4, we observe that lock-step measures are generally the fastest, with the top-performing lock-step measure, Lorentzian, exhibiting a runtime of only 0.004 seconds on AtrialFibrillation dataset, albeit with a relatively modest accuracy of 0.60. SBD-D and SINK-D, both of which leverage FFT2, also lie on

Table 11. Guidelines for measure selection.

Available time	Best temporal model	Best measure
Limited time (< ms)	Lock-step	Lorentzian
General case (ms - sec)	**Sliding**	**SBD-D**
Unlimited time (days)	Elastic	MSM-I + tuning

the Pareto frontier, i.e., the sets of methods that achieve the highest accuracy at a given runtime. These measures outperform lockstep by a significant margin in downstream accuracy, with SBD-D reaching up to 0.65, while maintaining a relatively low runtime of 0.05 seconds.

Elastic measures do achieve marginally higher accuracies (≈ 0.66), yet this improvement requires a high computational cost, with MSM-I taking 4982 seconds and DTW-D taking 297 seconds, making them 4-5 orders of magnitude slower than SBD-D. Again note though that elastic measure and their kernel variants require parameter tuning in the absence of domain experts, which further multiplies the runtime proportional to the parameter searching space. For instance, given our parameter space for MSM, the total runtime would approximate 3 hours (including both parameterization and inference) on this relatively small dataset, assuming parallel execution of validation runs. Considering the quadratic complexity of elastic measures, this would result in weeks of runtime on the largest datasets in the UEA archive, as observed during the execution of our experiments. Lastly, feature-based, model-based, and embedding measures generally fall below the Pareto frontier with exception of GRAIL, which remains near the Pareto frontier by benefiting from high inference speed through dimensionality reduction. Note that TS2Vec and TLoss were excluded from this analysis due to their dependency on GPU acceleration, making them incomparable to all other CPU-based measures. Still, we note that their training phase never exceeded 1 hour on the downsampled datasets, using a single NVIDIA A100 GPU.

Regarding measure scalability, the results in Figure 9b reveal that lock-step, sliding, elastic, and kernel measures scale accordingly to their theoretical complexity w.r.t. the length of the time series T (cf. Section 4). Furthermore, we see that current model-based measures scale linearly with the length of the time series, and that GRAIL scales identically to SBD due to their reliance on FFT2. Regarding the number of channels C, Figure 9c reveals that empirical scaling w.r.t. C can deviate substantially from the theoretical worst-case complexity. Namely, we find for elastic measures that while both channel-dependency variants share the same theoretical complexity, the channel-dependent variants of measures often exhibit substantially lower runtimes. This is possibly due to the construction of a single warping path, reducing the overhead from resursive method calls and enabling vectorization to compute the cost of alignments.

In summary, these findings emphasize the importance of selecting temporal and channel-dependency models that balance accuracy and runtime in practical applications.

6 Guidelines

Based on this evaluation study, we provide the following guidelines for selecting a distance measure for MTS similarity search when the downstream task requires *correction of distortions* (e.g., classification, clustering, pattern matching):

- In the **general case**, we recommend SBD-D, a parameter-free sliding measure that handles global misalignment, providing highly competitive performance with low computational complexity.
- In the case of **runtime performance** being the primary concern, lock-step measures are the optimal choice. Among them, *Lorentzian* outperforms the commonly used Euclidean distance, making it the recommended choice for performance.
- In the case **maximizing accuracy** being the primary concern and runtime is not a constraint, a channel-independent elastic measure like *MSM-I* with supervised parameter tuning is the

best option. However, it is worth noting that this measure requires significant runtime on large datasets for both parameterization and inference.
- In the case of selecting a **channel-dependency model** for a given distance measure, we recommend the channel-independent variant for elastic measures, while the channel-dependent variant is generally preferred for all other measures.

We summarize these guidelines in Table 11. When the downstream task requires *preservation of distortions* (e.g., anomaly detection), we provide the following guideline:

- Choose a measure family that **does not correct for distortions**. Lock-step measures are the most efficient and effective choice here, with the recommended choice being Euclidean distance.

These guidelines were obtained by studying distance measures in the context of MTS of *equal length* and *equal number of channels*. Appropriate preprocessing steps are expected to be performed (prior to analysis) to ensure the applicability of these guidelines.

7 Conclusion

In this paper, we conducted a structured evaluation of MTS distance measures, benchmarking 30 standalone measures across 8 categories and 2 channel-dependency models, utilizing 13 normalization methods on 30 datasets, and evaluating over 3 downstream tasks with proper parameter tuning. This evaluation was structured through considering the three key axes of MTS comparison: normalization, temporal model, and channel-dependency model. The results extend findings of prior works to the multivariate case but also provide insights specific to multivariate distances: (a) SBD-D offers the best accuracy-to-runtime trade-off across all temporal models, delivering performance comparable to elastic measures while being significantly faster and parameter-free; (b) no existing normalization method provides significant benefit over not normalizing, highlighting the need for MTS-specific normalizations to be developed; and (c) channel-independent variants of measures prove beneficial only for elastic measures. Through this study, we pave the way for a practical handbook that guides the selection and design of MTS distance measures.

Acknowledgments

The authors thank anonymous reviewers whose comments greatly improved this manuscript. We also thank Ryan DeMilt for his contributions in the early phase of this project. This research was supported in part by Cisco Systems, Meta, and received funding from the European Union's Horizon Europe research and innovation programme STELAR under grant agreement No. 101070122.

References

[1] 2025. A Structured Study of Multivariate Time-Series Distance Measures. https://github.com/TheDatumOrg/MTSDistEval. Accessed: 2025-03-24.

[2] Rakesh Agrawal, Christos Faloutsos, and Arun Swami. 1993. Efficient similarity search in sequence databases. In *Foundations of Data Organization and Algorithms*, David B. Lomet (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 69–84.

[3] Martin Bach-Andersen, Bo Rømer-Odgaard, and Ole Winther. 2017. Flexible non-linear predictive models for large-scale wind turbine diagnostics. *Wind Energy* 20 (2017), 753–764.

[4] Anthony Bagnall, Jason Lines, Aaron Bostrom, James Large, and Eamonn Keogh. 2017. The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery* 31, 3 (01 May 2017), 606–660.

[5] Anthony J. Bagnall, Hoang Anh Dau, Jason Lines, Michael Flynn, James Large, Aaron Bostrom, Paul Southam, and Eamonn J. Keogh. 2018. The UEA multivariate time series classification archive, 2018. *CoRR* abs/1811.00075 (2018).

[6] A. J. Bagnall and G. J. Janacek. 2004. Clustering time series from ARMA models with clipped data. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Seattle, WA, USA) *(KDD '04)*. Association for Computing Machinery, New York, NY, USA, 49–58.

[7] Mohini Bariya, Alexandra von Meier, John Paparrizos, and Michael J Franklin. 2021. k-shapestream: Probabilistic streaming clustering for electric grid events. In 2021 IEEE Madrid PowerTech. *IEEE, 1ś6* (2021).

[8] Leonard E Baum and Ted Petrie. 1966. Statistical inference for probabilistic functions of finite state Markov chains. *The annals of mathematical statistics* 37, 6 (1966), 1554–1563.

[9] Nurjahan Begum and Eamonn Keogh. 2014. Rare time series motif discovery from unbounded streams. *Proc. VLDB Endow.* 8, 2 (oct 2014), 149–160. doi:10.14778/2735471.2735476

[10] Angela Bonifati, Francesco Del Buono, Francesco Guerra, Miki Lombardi, and Donato Tiano. 2023. Interpretable Clustering of Multivariate Time Series with Time2Feat. *Proc. VLDB Endow.* 16, 12 (2023), 3994–3997.

[11] Paul Boniol, Ashwin K Krishna, Marine Bruel, Qinghua Liu, Mingyi Huang, Themis Palpanas, Ruey S Tsay, Aaron Elmore, Michael J Franklin, and John Paparrizos. 2025. VUS: effective and efficient accuracy measures for time-series anomaly detection. *The VLDB Journal* 34, 3 (2025), 32.

[12] Paul Boniol, Qinghua Liu, Mingyi Huang, Themis Palpanas, and John Paparrizos. 2024. Dive into Time-Series Anomaly Detection: A Decade Review. *arXiv preprint arXiv:2412.20512* (2024).

[13] Paul Boniol, John Paparrizos, Yuhao Kang, Themis Palpanas, Ruey S Tsay, Aaron J Elmore, and Michael J Franklin. 2022. Theseus: navigating the labyrinth of time-series anomaly detection. *Proceedings of the VLDB Endowment* 15, 12 (2022), 3702–3705.

[14] Paul Boniol, John Paparrizos, and Themis Palpanas. 2023. New Trends in Time Series Anomaly Detection.. In *EDBT*. 847–850.

[15] Paul Boniol, John Paparrizos, and Themis Palpanas. 2024. An interactive dive into time-series anomaly detection. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE, 5382–5386.

[16] Paul Boniol, John Paparrizos, Themis Palpanas, and Michael J Franklin. 2021. SAND in action: subsequence anomaly detection for streams. *Proceedings of the VLDB Endowment* 14, 12 (2021), 2867–2870.

[17] Paul Boniol, John Paparrizos, Themis Palpanas, and Michael J Franklin. 2021. SAND: streaming subsequence anomaly detection. *Proceedings of the VLDB Endowment* 14, 10 (2021), 1717–1729.

[18] Paul Boniol, Emmanouil Sylligardos, John Paparrizos, Panos Trahanias, and Themis Palpanas. 2024. Adecimo: Model selection for time series anomaly detection. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE, 5441–5444.

[19] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. 2000. LOF: identifying density-based local outliers. *SIGMOD Rec.* 29, 2 (may 2000), 93–104. doi:10.1145/335191.335388

[20] Yuhan Cai and Raymond Ng. 2004. Indexing spatio-temporal trajectories with Chebyshev polynomials. In *Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data* (Paris, France) *(SIGMOD '04)*. Association for Computing Machinery, New York, NY, USA, 599–610.

[21] Alessandro Camerra, Themis Palpanas, Jin Shieh, and Eamonn Keogh. 2010. iSAX 2.0: Indexing and Mining One Billion Time Series. In *2010 IEEE International Conference on Data Mining*. 58–67.

[22] Kaushik Chakrabarti, Eamonn Keogh, Sharad Mehrotra, and Michael Pazzani. 2002. Locally adaptive dimensionality reduction for indexing large time series databases. *ACM Trans. Database Syst.* 27, 2 (jun 2002), 188–228.

[23] Lei Chen and Raymond Ng. 2004. On the marriage of Lp-norms and edit distance. In *Proceedings of the Thirtieth International Conference on Very Large Data Bases - Volume 30* (Toronto, Canada) *(VLDB '04)*. VLDB Endowment, 792–803.

[24] Qiuxia Chen, Lei Chen, Xiang Lian, Yunhao Liu, and Jeffrey Xu Yu. 2007. Indexable PLA for efficient similarity search. In *Proceedings of the 33rd International Conference on Very Large Data Bases* (Vienna, Austria) *(VLDB '07)*. VLDB Endowment, 435–446.

[25] Bill Chiu, Eamonn Keogh, and Stefano Lonardi. 2003. Probabilistic discovery of time series motifs. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Washington, D.C.) *(KDD '03)*. Association for Computing Machinery, New York, NY, USA, 493–498. doi:10.1145/956750.956808

[26] Maximilian Christ, Nils Braun, Julius Neuffer, and Andreas W. Kempa-Liehr. 2018. Time Series FeatuRe Extraction on basis of Scalable Hypothesis tests (tsfresh – A Python package). *Neurocomputing* 307 (2018), 72–77.

[27] Kelvin Kam Wing Chu and Man Hon Wong. 1999. Fast time-series searching with scaling and shifting. In *Proceedings of the Eighteenth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems* (Philadelphia, Pennsylvania, USA) *(PODS '99)*. Association for Computing Machinery, New York, NY, USA, 237–248. doi:10.1145/303976.304000

[28] Richard Cole, Dennis Shasha, and Xiaojian Zhao. 2005. Fast window correlations over uncooperative time series. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining* (Chicago, Illinois, USA) *(KDD '05)*. Association for Computing Machinery, New York, NY, USA, 743–749.

[29] Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine Learning* 20, 3 (01 Sep 1995), 273–297.

[30] Nello Cristianini and John Shawe-Taylor. 2000. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press.

[31] Marco Cuturi. 2011. Fast global alignment kernels. In *Proceedings of the 28th international conference on machine learning (ICML-11)*. 929–936.
[32] Michele Dallachiesa, Themis Palpanas, and Ihab F. Ilyas. 2014. Top-k nearest neighbor search in uncertain data series. *Proc. VLDB Endow.* 8, 1 (sep 2014), 13–24. doi:10.14778/2735461.2735463
[33] Janez Demšar. 2006. Statistical Comparisons of Classifiers over Multiple Data Sets. *J. Mach. Learn. Res.* 7 (dec 2006), 1–30.
[34] Hui Ding, Goce Trajcevski, Peter Scheuermann, Xiaoyue Wang, and Eamonn Keogh. 2008. Querying and mining of time series data: experimental comparison of representations and distance measures. *Proc. VLDB Endow.* 1, 2 (aug 2008), 1542–1552.
[35] Rui Ding, Qiang Wang, Yingnong Dang, Qiang Fu, Haidong Zhang, and Dongmei Zhang. 2015. YADING: fast clustering of large-scale time series data. *Proc. VLDB Endow.* 8, 5 (jan 2015), 473–484.
[36] Jens E d'Hondt, Odysseas Papapetrou, and John Paparrizos. 2024. Beyond the Dimensions: A Structured Evaluation of Multivariate Time Series Distance Measures. In *2024 IEEE 40th International Conference on Data Engineering Workshops (ICDEW)*. IEEE, 107–112.
[37] Karima Echihabi, Kostas Zoumpatianos, Themis Palpanas, and Houda Benbrahim. 2018. The Lernaean Hydra of Data Series Similarity Search: An Experimental Evaluation of the State of the Art. *Proc. VLDB Endow.* 12, 2 (oct 2018), 112–127.
[38] Karima Echihabi, Kostas Zoumpatianos, Themis Palpanas, and Houda Benbrahim. 2019. Return of the Lernaean Hydra: experimental evaluation of data series approximate similarity search. *Proc. VLDB Endow.* 13, 3 (nov 2019), 403–420.
[39] Philippe Esling and Carlos Agon. 2012. Time-series data mining. *ACM Comput. Surv.* 45, 1, Article 12 (dec 2012), 34 pages.
[40] Christos Faloutsos, M. Ranganathan, and Yannis Manolopoulos. 1994. Fast subsequence matching in time-series databases. *SIGMOD Rec.* 23, 2 (may 1994), 419–429.
[41] Jean-Yves Franceschi, Aymeric Dieuleveut, and Martin Jaggi. 2019. Unsupervised scalable representation learning for multivariate time series. *Advances in neural information processing systems* 32 (2019).
[42] Milton Friedman. 1937. The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance. *J. Amer. Statist. Assoc.* 32, 200 (1937), 675–701.
[43] Shima Ghassempour, Federico Girosi, and Anthony Maeder. 2014. Clustering multivariate time series using Hidden Markov Models. *Int J Environ Res Public Health* 11, 3 (March 2014), 2741–2763.
[44] Rahul Goel, Sandeep Soni, Naman Goyal, John Paparrizos, Hanna Wallach, Fernando Diaz, and Jacob Eisenstein. 2016. The social dynamics of language change in online networks. In *International conference on social informatics*. Springer, 41–57.
[45] Dina Q. Goldin and Paris C. Kanellakis. 1995. On similarity queries for time-series data: Constraint specification and implementation. In *Principles and Practice of Constraint Programming — CP '95*, Ugo Montanari and Francesca Rossi (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 137–153.
[46] Suchit Gupte and John Paparrizos. 2025. ShapX Engine: A Demonstration of Shapley Value Approximations. In *Companion of the 2025 International Conference on Management of Data (SIGMOD-Companion '25)*. ACM, Berlin, Germany, 4. doi:10.1145/3722212.3725135
[47] Suchit Gupte and John Paparrizos. 2025. Understanding the Black Box: A Deep Empirical Dive into Shapley Value Approximations for Tabular Data. In *Proceedings of the ACM on Management of Data*, Vol. 3. Article 232. doi:10.1145/3725420
[48] Fred Harris. 2021. Polyphase Interpolators with Reversed Order of Up-Sampling and Down-Sampling. In *2021 55th Asilomar Conference on Signals, Systems, and Computers*. 918–924. doi:10.1109/IEEECONF53345.2021.9723220
[49] Jon Hills, Jason Lines, Edgaras Baranauskas, James Mapp, and Anthony Bagnall. 2014. Classification of time series by shapelet transformation. *Data Min. Knowl. Discov.* 28, 4 (jul 2014), 851–881.
[50] Pablo Huijse, Pablo A. Estevez, Pavlos Protopapas, Jose C. Principe, and Pablo Zegers. 2014. Computational Intelligence Challenges and Applications on Large-Scale Astronomical Time Series Databases. *IEEE Computational Intelligence Magazine* 9, 3 (2014), 27–39. doi:10.1109/MCI.2014.2326100
[51] Hoyoung Jeung, Sofiane Sarni, Ioannis Paparrizos, Saket Sathe, Karl Aberer, Nicholas Dawes, Thanasis G Papaioannou, and Michael Lehning. 2010. Effective metadata management in federated sensor networks. In *2010 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*. IEEE, 107–114.
[52] Hao Jiang, Chunwei Liu, Qi Jin, John Paparrizos, and Aaron J Elmore. 2020. PIDS: attribute decomposition for improved compression and query performance in columnar storage. *Proceedings of the VLDB Endowment* 13, 6 (2020), 925–938.
[53] Hao Jiang, Chunwei Liu, John Paparrizos, Andrew A Chien, Jihong Ma, and Aaron J Elmore. 2021. Good to the Last Bit: Data-Driven Encoding with CodecDB. In *Proceedings of the 2021 International Conference on Management of Data*.

843–856.

[54] K. Kalpakis, D. Gada, and V. Puttagunta. 2001. Distance measures for effective clustering of ARIMA time-series. In *Proceedings 2001 IEEE International Conference on Data Mining*. 273–280. doi:10.1109/ICDM.2001.989529

[55] Shrikant Kashyap and Panagiotis Karras. 2011. Scalable kNN search on vertically stored time series. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (San Diego, California, USA) *(KDD '11)*. Association for Computing Machinery, New York, NY, USA, 1334–1342.

[56] Leonard Kaufman and Peter J Rousseeuw. 2009. *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons.

[57] Eamonn Keogh. 2006. A decade of progress in indexing and mining large time series databases. In *Proceedings of the 32nd International Conference on Very Large Data Bases* (Seoul, Korea) *(VLDB '06)*. VLDB Endowment, 1268.

[58] S. Knieling, J. Niediek, E. Kutter, J. Bostroem, C.E. Elger, and F. Mormann. 2017. An online adaptive screening procedure for selective neuronal responses. *Journal of Neuroscience Methods* 291 (2017), 36–42.

[59] Flip Korn, H. V. Jagadish, and Christos Faloutsos. 1997. Efficiently supporting ad hoc queries in large datasets of time sequences. In *Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data* (Tucson, Arizona, USA) *(SIGMOD '97)*. Association for Computing Machinery, New York, NY, USA, 289–300.

[60] Sanjay Krishnan, Aaron J Elmore, Michael Franklin, John Paparrizos, Zechao Shang, Adam Dziedzic, and Rui Liu. 2019. Artificial intelligence in resource-constrained and shared environments. *ACM SIGOPS Operating Systems Review* 53, 1 (2019), 1–6.

[61] Sergei Lebedev. 2010. hmmlearn. https://github.com/hmmlearn/hmmlearn.

[62] Xiang Lian, Lei Chen, Jeffrey Xu Yu, Guoren Wang, and Ge Yu. 2007. Similarity Match Over High Speed Time-Series Streams. In *2007 IEEE 23rd International Conference on Data Engineering*. 1086–1095.

[63] Michele Linardi and Themis Palpanas. 2018. Scalable, variable-length similarity search in data series: the ULISSE approach. *Proc. VLDB Endow.* 11, 13 (sep 2018), 2236–2248. doi:10.14778/3275366.3284968

[64] Chunwei Liu, Hao Jiang, John Paparrizos, and Aaron J Elmore. 2021. Decomposed bounded floats for fast compression and queries. *Proceedings of the VLDB Endowment* 14, 11 (2021), 2586–2598.

[65] Chunwei Liu, John Paparrizos, and Aaron J Elmore. 2024. AdaEdge: A Dynamic Compression Selection Framework for Resource Constrained Devices. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE, 1506–1519.

[66] Qinghua Liu, Paul Boniol, Themis Palpanas, and John Paparrizos. 2024. Time-Series Anomaly Detection: Overview and New Trends. *Proceedings of the VLDB Endowment (PVLDB)* 17, 12 (2024), 4229–4232.

[67] Qinghua Liu and John Paparrizos. 2024. The Elephant in the Room: Towards A Reliable Time-Series Anomaly Detection Benchmark. In *NeurIPS 2024*.

[68] Shinan Liu, Tarun Mangla, Ted Shaowang, Jinjin Zhao, John Paparrizos, Sanjay Krishnan, and Nick Feamster. 2023. AMIR: Active Multimodal Interaction Recognition from Video and Network Traffic in Connected Environments. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 7, 1 (2023), 1–26.

[69] Markus Löning, Anthony J. Bagnall, Sajaysurya Ganesh, Viktor Kazakov, Jason Lines, and Franz J. Király. 2019. sktime: A Unified Interface for Machine Learning with Time Series. *CoRR* abs/1909.07872 (2019). arXiv:1909.07872 http://arxiv.org/abs/1909.07872

[70] Carl H Lubba, Sarab S Sethi, Philip Knaute, Simon R Schultz, Ben D Fulcher, and Nick S Jones. 2019. catch22: CAnonical Time-series CHaracteristics: Selected through highly comparative time-series analysis. *Data Mining and Knowledge Discovery* 33, 6 (2019), 1821–1852.

[71] Pierre-François Marteau and Sylvie Gibet. 2010. Constructing Positive Elastic Kernels with Application to Time Series Classification. *CoRR* abs/1005.5141 (2010). arXiv:1005.5141 http://arxiv.org/abs/1005.5141

[72] Pierre-François Marteau. 2009. Time Warp Edit Distance with Stiffness Adjustment for Time Series Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31, 2 (2009), 306–318.

[73] Kathy McKeown, Hal Daume III, Snigdha Chaturvedi, John Paparrizos, Kapil Thadani, Pablo Barrio, Or Biran, Suvarna Bothe, Michael Collins, Kenneth R Fleischmann, et al. 2016. Predicting the impact of scientific concepts using full-text features. *Journal of the Association for Information Science and Technology* 67, 11 (2016), 2684–2696.

[74] Qianwen Meng, Hangwei Qian, Yong Liu, Lizhen Cui, Yonghui Xu, and Zhiqi Shen. 2023. MHCCL: Masked Hierarchical Cluster-Wise Contrastive Learning for Multivariate Time Series. In *AAAI*. AAAI Press, 9153–9161.

[75] Abdullah Mueen, Eamonn Keogh, Qiang Zhu, Sydney Cash, and Brandon Westover. 2009. Exact discovery of time series motifs. In *Proceedings of the 2009 SIAM International Conference on Data Mining*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 473–484.

[76] Peter Björn Nemenyi. 1963. *Distribution-free Multiple Comparisons*. Ph. D. Dissertation. Princeton University.

[77] Panagiotis Papapetrou, Vassilis Athitsos, Michalis Potamias, George Kollios, and Dimitrios Gunopulos. 2011. Embedding-based subsequence matching in time-series databases. *ACM Trans. Database Syst.* 36, 3, Article 17 (aug 2011), 39 pages.

[78] Ioannis Paparrizos. 2018. *Fast, scalable, and accurate algorithms for time-series analysis.* Ph. D. Dissertation. Columbia University, USA.

[79] Ioannis Paparrizos, Hoyoung Jeung, and Karl Aberer. 2011. Advanced search, visualization and tagging of sensor metadata. In *2011 IEEE 27th International Conference on Data Engineering*. IEEE, 1356–1359.

[80] John Paparrizos, Paul Boniol, Themis Palpanas, Ruey S Tsay, Aaron Elmore, and Michael J Franklin. 2022. Volume under the surface: a new accuracy evaluation measure for time-series anomaly detection. *Proceedings of the VLDB Endowment* 15, 11 (2022), 2774–2787.

[81] John Paparrizos, Ikraduya Edian, Chunwei Liu, Aaron J Elmore, and Michael J Franklin. 2022. Fast Adaptive Similarity Search through Variance-Aware Quantization. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 2969–2983.

[82] John Paparrizos and Michael J. Franklin. 2019. GRAIL: efficient time-series representation learning. *Proc. VLDB Endow.* 12, 11 (jul 2019), 1762–1777. doi:10.14778/3342263.3342648

[83] John Paparrizos and Luis Gravano. 2015. k-Shape: Efficient and Accurate Clustering of Time Series. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data* (Melbourne, Victoria, Australia) *(SIGMOD '15)*. Association for Computing Machinery, New York, NY, USA, 1855–1870. doi:10.1145/2723372.2737793

[84] John Paparrizos and Luis Gravano. 2017. Fast and Accurate Time-Series Clustering. *ACM Transactions on Database Systems (TODS)* 42, 2 (2017), 1–49.

[85] John Paparrizos, Yuhao Kang, Paul Boniol, Ruey S Tsay, Themis Palpanas, and Michael J Franklin. 2022. TSB-UAD: an end-to-end benchmark suite for univariate time-series anomaly detection. *Proceedings of the VLDB Endowment* 15, 8 (2022), 1697–1711.

[86] John Paparrizos, Haojun Li, Fan Yang, Kaize Wu, Jens E d'Hondt, and Odysseas Papapetrou. 2024. A survey on time-series distance measures. *arXiv preprint arXiv:2412.20574* (2024).

[87] John Paparrizos, Chunwei Liu, Bruno Barbarioli, Johnny Hwang, Ikraduya Edian, Aaron J Elmore, Michael J Franklin, and Sanjay Krishnan. 2021. VergeDB: A Database for IoT Analytics on Edge Devices.. In *CIDR*.

[88] John Paparrizos, Chunwei Liu, Aaron J. Elmore, and Michael J. Franklin. 2020. Debunking Four Long-Standing Misconceptions of Time-Series Distance Measures. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data (SIGMOD '20)*. Association for Computing Machinery, New York, NY, USA, 1887–1905.

[89] John Paparrizos, Chunwei Liu, Aaron J Elmore, and Michael J Franklin. 2023. Querying Time-Series Data: A Comprehensive Comparison of Distance Measures. *Data Engineering* (2023), 69.

[90] John Paparrizos and Sai Prasanna Teja Reddy. 2023. Odyssey: An Engine Enabling the Time-Series Clustering Journey. *Proceedings of the VLDB Endowment* 16, 12 (2023), 4066–4069.

[91] John Paparrizos, Ryen W White, and Eric Horvitz. 2016. Detecting devastating diseases in search logs. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 559–568.

[92] John Paparrizos, Ryen W White, and Eric Horvitz. 2016. Screening for pancreatic adenocarcinoma using signals from web search logs: Feasibility study and results. *Journal of Oncology Practice* 12, 8 (2016), 737–744.

[93] John Paparrizos, Kaize Wu, Aaron Elmore, Christos Faloutsos, and Michael J Franklin. 2023. Accelerating Similarity Search for Elastic Measures: A Study and New Generalization of Lower Bounding Distances. *Proceedings of the VLDB Endowment* 16, 8 (2023), 2019–2032.

[94] John Paparrizos, Fan Yang, and Haojun Li. 2024. Bridging the gap: A decade review of time-series clustering methods. *arXiv preprint arXiv:2412.20582* (2024).

[95] Pavlos Paraskevopoulos, Thanh-Cong Dinh, Zolzaya Dashdorj, Themis Palpanas, Luciano Serafini, et al. 2013. Identification and characterization of human behavior patterns from mobile phone data. *D4D Challenge session, NetMob* (2013).

[96] Davood Rafiei and Alberto Mendelzon. 1997. Similarity-based queries for time series data. In *Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data* (Tucson, Arizona, USA) *(SIGMOD '97)*. Association for Computing Machinery, New York, NY, USA, 13–25.

[97] Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim. 2000. Efficient algorithms for mining outliers from large data sets. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data* (Dallas, Texas, USA) *(SIGMOD '00)*. Association for Computing Machinery, New York, NY, USA, 427–438. doi:10.1145/342009.335437

[98] William M Rand. 1971. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association* 66, 336 (1971), 846–850.

[99] John F. Roddick and Kathleen Stewart Hornsby. 2001. Temporal, Spatial, and Spatio-Temporal Data Mining. In *Lecture Notes in Computer Science*.

[100] Alejandro Pasos Ruiz, Michael Flynn, James Large, Matthew Middlehurst, and Anthony Bagnall. 2021. The great multivariate time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery* 35, 2 (01 Mar 2021), 401–449.

[101] H. Sakoe and S. Chiba. 1978. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 26, 1 (1978), 43–49.
[102] A Salarpour and H Khotanlou. 2018. An Empirical Comparison of Distance Measures for Multivariate Time Series Clustering. *International Journal of Engineering* 31, 2 (2018), 250–262.
[103] Dennis Shasha and Surajit Chaudhuri. 1999. Tuning time series queries in finance: Case studies and recommendations. *IEEE Data Engineering Bulletin* Special Issue on Performance Tuning for Database Systems (July 1999).
[104] Ahmed Shifaz, Charlotte Pelletier, François Petitjean, and Geoffrey I. Webb. 2023. Elastic similarity and distance measures for multivariate time series. *Knowledge and Information Systems* 65, 6 (01 Jun 2023), 2665–2698.
[105] Mohammad Shokoohi-Yekta, Bing Hu, Hongxia Jin, Jun Wang, and Eamonn J. Keogh. 2017. Generalizing DTW to the multi-dimensional case requires an adaptive approach. *Data Min. Knowl. Discov.* 31, 1 (2017), 1–31.
[106] Alexandra Stefan, Vassilis Athitsos, and Gautam Das. 2013. The Move-Split-Merge Metric for Time Series. *IEEE Transactions on Knowledge and Data Engineering* 25, 6 (2013), 1425–1438.
[107] Emmanouil Sylligardos, Paul Boniol, John Paparrizos, Panos Trahanias, and Themis Palpanas. 2023. Choose Wisely: An Extensive Evaluation of Model Selection for Anomaly Detection in Time Series. *Proceedings of the VLDB Endowment* 16, 11 (2023), 3418–3432.
[108] Shreshth Tuli, Giuliano Casale, and Nicholas R. Jennings. 2022. TranAD: Deep Transformer Networks for Anomaly Detection in Multivariate Time Series Data. *Proc. VLDB Endow.* 15, 6 (2022), 1201–1214.
[109] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. 2020. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* 17 (2020), 261–272.
[110] Michail Vlachos, Marios Hadjieleftheriou, Dimitrios Gunopulos, and Eamonn Keogh. 2003. Indexing Multi-Dimensional Time-Series with Support for Multiple Distance Measures. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Washington, D.C.) *(KDD '03)*. Association for Computing Machinery, New York, NY, USA, 216–225.
[111] Renzhuo Wan, Shuping Mei, Jun Wang, Min Liu, and Fan Yang. 2019. Multivariate temporal convolutional network: A deep neural networks approach for multivariate time series forecasting. *Electronics* 8, 8 (2019), 876.
[112] Xiaoyue Wang, Abdullah Mueen, Hui Ding, Goce Trajcevski, Peter Scheuermann, and Eamonn Keogh. 2013. Experimental comparison of representation methods and distance measures for time series data. *Data Mining and Knowledge Discovery* 26, 2 (01 Mar 2013), 275–309. doi:10.1007/s10618-012-0250-5
[113] Frank Wilcoxon. 1945. Individual Comparisons by Ranking Methods. *Biometrics Bulletin* 1, 6 (1945), 80–83.
[114] Fan Yang and John Paparrizos. 2025. SPARTAN: Data-Adaptive Symbolic Time-Series Approximation. In *Proceedings of the ACM on Management of Data*, Vol. 3. Article 220. doi:10.1145/3725357
[115] Kiyoung Yang and Cyrus Shahabi. 2004. A PCA-Based Similarity Measure for Multivariate Time Series. In *Proceedings of the 2nd ACM International Workshop on Multimedia Databases* (Washington, DC, USA) *(MMDB '04)*. Association for Computing Machinery, New York, NY, USA, 65–74.
[116] Chin-Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Diego Furtado Silva, Abdullah Mueen, and Eamonn Keogh. 2016. Matrix Profile I: All Pairs Similarity Joins for Time Series: A Unifying View That Includes Motifs, Discords and Shapelets. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*. 1317–1322. doi:10.1109/ICDM.2016.0179
[117] Chin-Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Zachary Zimmerman, Diego Furtado Silva, Abdullah Mueen, and Eamonn Keogh. 2018. Time series joins, motifs, discords and shapelets: a unifying view that exploits the matrix profile. *Data Min. Knowl. Discov.* 32, 1 (jan 2018), 83–123. doi:10.1007/s10618-017-0519-9
[118] Zhihan Yue, Yujing Wang, Juanyong Duan, Tianmeng Yang, Congrui Huang, Yunhai Tong, and Bixiong Xu. 2022. Ts2vec: Towards universal representation of time series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 8980–8987.

Received October 2024; revised January 2025; accepted February 2025