
Random Warping Series: A Random Features Method for Time-Series Embedding

Lingfei Wu
IBM Research

Ian En-Hsu Yen
Carnegie Mellon University

Jinfeng Yi
IBM Research

Fangli Xu
College of William and Mary

Qi Lei
University of Texas at Austin

Michael J. Witbrock
IBM Research

Abstract

Time series data analytics has been a problem of substantial interests for decades, and Dynamic Time Warping (DTW) has been the most widely adopted technique to measure dissimilarity between time series. A number of global-alignment kernels have since been proposed in the spirit of DTW to extend its use to kernel-based estimation method such as support vector machine. However, those kernels suffer from diagonal dominance of the Gram matrix and a quadratic complexity w.r.t. the sample size. In this work, we study a family of alignment-aware positive definite (p.d.) kernels, with its feature embedding given by a distribution of *Random Warping Series (RWS)*. The proposed kernel does not suffer from the issue of diagonal dominance while naturally enjoys a *Random Features* (RF) approximation, which reduces the computational complexity of existing DTW-based techniques from quadratic to linear in terms of both the number and the length of time-series. We also study the convergence of the RF approximation for the domain of time series of unbounded length. Our extensive experiments on 16 benchmark datasets demonstrate that RWS outperforms or matches state-of-the-art classification and clustering methods in both accuracy and computational time.

1 Introduction

Over the last two decades, time series classification and clustering have received considerable interests in many applications such as genomic research [Leslie et al., 2002], image alignment [Peng et al., 2015b,a], speech recognition [Cuturi et al., 2007, Shimodaira et al., 2001], and motion detection [Li and Prakash, 2011]. One of the main challenges in time series data stems from the fact that there are no explicit features in sequences [Xing et al., 2010]. Therefore, a number of feature representation methods have been proposed recently, among which the approaches deriving features from phase dependent intervals [Deng et al., 2013, Baydogan et al., 2013], phase independent shapelets [Ye and Keogh, 2009, Rakthanmanon and Keogh, 2013], and dictionary based bags of patterns [Senin and Malinchik, 2013, Schäfer, 2015] have gained much popularity due to their highly competitive performance [Bagnall et al., 2016]. However, since the aforementioned approaches only consider the local patterns rather than global properties, the effectiveness of these features largely depends on the underlying characteristics of sequences that may vary significantly across applications. More importantly, these approaches may typically not be a good first choice for large scale time series due to their quadratic complexity in terms of both the number N and (or) length L of time series.

Another family of research defines a distance function to measure the similarity between a pair of time series. Although Euclidean distance is a widely used option and has been shown to be competitive with other more complex similarity measures [Wang et al., 2013], various elastic distance measures designed to address the temporal dynamics and time shifts are more appropriate [Xing et al., 2010, Kate, 2016]. Among them, dynamic time warping (DTW) [Berndt and Clifford, 1994] is the standard elastic distance measure for time series. Interestingly, an 1NN classifier with DTW has been

demonstrated as the gold standard benchmark, and has been proved difficult to beat consistently [Wang et al., 2013, Bagnall et al., 2016]. Recently, a thread of research has attempted to directly use the pair-wise DTW distance as features [Hayashi et al., 2005, Gudmundsson et al., 2008, Kate, 2016, Lei et al., 2017]. However, the majority of these approaches have quadratic complexity in both number and length of time series in terms of the computation and memory requirements.

Despite the successes of various explicit feature design, kernel methods have great promise for learning non-linear models by implicitly transforming a simple representations into a high-dimension feature space [Rahimi and Recht, 2007, Chen et al., 2016, Wu et al., 2016, Yen et al., 2014]. The main obstacle for applying kernel method to time series is largely due to two distinct characteristics of time series, (a) variable length; and (b) dynamic time scaling and shifts. Since elastic distance measures, such as DTW, take into account these two issues, there have been several attempts to apply DTW directly as a similarity measure in a kernel-based classification model [Shimodaira et al., 2001, Gudmundsson et al., 2008]. Unfortunately, the DTW distance does not correspond to a *valid positive-definite* (p.d.) kernel and thus direct use of DTW leads to an indefinite kernel matrix that neither corresponds to a loss minimization problem nor giving a convex optimization problem [Bahlmann et al., 2002, Cuturi et al., 2007]. To overcome these difficulties, a family of global alignment kernels have been proposed by taking softmax over all possible alignments in DTW to give a p.d. kernel [Cuturi et al., 2007, Cuturi, 2011, Marteau and Gibet, 2015]. However, the effectiveness of the global alignment kernels is impaired by the diagonal dominance of the resulting kernel matrix. Also, the quadratic complexity in both the number and length of time series make it hard to scale.

In this paper, inspired by the latest advancement of kernel learning methodology from distance [Wu et al., 2018], we study *Random Warping Series* (RWS), a generic framework to generate vector representation of time-series, where we construct a family of p.d. kernels from an explicit feature map given by the DTW between original time series and a distribution of random series. To admit an efficient computation of the kernel, we give a random features approximation that uniformly converges to the proposed kernel using a finite number of random series drawn from the distribution. The RWS technique is fully parallelizable, and highly extensible in the sense that the building block DTW can be replaced by recently proposed elastic distance measures such as CID [Batista et al., 2014] and DTDC [Górecki and Łuczak, 2014]. With a number R of random series, RWS can substantially reduce the compu-

tational complexity of existing DTW-based techniques from $O(N^2L^2)$ to $O(NRL)$ and memory consumption from $O(NL + N^2)$ to $O(NR)$. We also extend existing analysis of random features to handle time series of unbounded length, showing that $R = \Omega(1/\epsilon^2)$ suffices for the uniform convergence to ϵ precision of the exact kernel. We evaluate RWS on 16 real-world datasets on which it consistently outperforms or matches state-of-the-art baselines in terms of both testing accuracy and runtime. In particular, RWS often achieves orders-of-magnitude speedup over other methods to achieve the same accuracy.

2 DTW and Global Alignment Kernels

We first introduce the widely-used technique DTW and nearest-neighbor DTW (1NN-DTW), and then illustrate the existing global alignment kernels for time series and their disadvantages.

Time Series Alignment and 1NN-DTW. Let \mathcal{X} be the domain of input time series, and $\{x_i\}_{i=1}^N$ be the set of time series, where the length of each time series $|x_i| \leq L$, taking numeric values in \mathbb{R} . A special challenge in time series lies in the fact that the series could have different lengths, and a signal could be generated with time shifts and different scales, but with a similar pattern. To take these factors into account, an alignment (also called a warping function) is often introduced to provide a better distance/similarity measure between two time series $x_i = (x_i^1, \dots, x_i^n)$ and $x_j = (x_j^1, \dots, x_j^m)$ of lengths n and m respectively. Specifically, an alignment $a = (a_1, a_2)$ of length $|a| = p$ between two time series x_i and x_j is a pair of increasing vectors (a_1, a_2) such that $1 = a_1(1) \leq \dots \leq a_1(p) = n$ and $1 = a_2(1) \leq \dots \leq a_2(p) = m$ with unitary increments and no simultaneous repetitions. The set of all alignments between x_i and x_j is defined as $\mathcal{A}(x_i, x_j)$. In the literature of DTW [Berndt and Clifford, 1994], the DTW distance between x_i and x_j is defined as follows in its simplest form:

$$S(x_i, x_j) = \min_{a \in \mathcal{A}(x_i, x_j)} \tau(x_i, x_j; a), \quad (1)$$

where $\tau(x_i, x_j; a) = \sum_{t=1}^{|a|} \tau(x_i(a_1(t)), x_j(a_2(t)))$.

Here $\tau(x_i, x_j; a)$ is a dissimilarity measure between x_i and x_j under alignment a . Typically, *Dynamic Programming* (DP) is employed to find the optimal alignment a^* and then compute DTW distance. The dissimilarity function τ could be defined as any commonly used distance such as the squared Euclidean distance. To accelerate the computation and improve the performance, a Sakoe and Chiba band is often used

to constrain the search window size for DTW [Sakoe and Chiba, 1978, Rakthanmanon et al., 2012].

DTW has been widely used for time series classification in combination with the 1NN algorithm, and this combination has been shown to be exceptionally difficult to beat [Wang et al., 2013, Bagnall et al., 2016]. However, there are two disadvantages of 1NN-DTW. First, this method incurs the high computational cost of $O(N^2)$ complexity for computing DTW similarity between all pairs of time series, where each evaluation of DTW without constraints takes $O(L^2)$ computation. Second, Nearest-Neighbor methods often suffers from the problems of high variance. For example, if a class label is determined by a small portion of time series, a Nearest-Neighbor identification on the basis of similarity with the whole time series will be ineffective due to noise and irrelevant information.

Existing Global Alignment Kernels. To take the advantage of DTW in other prediction methods based on *Empirical Risk Minimization* (ERM) such as SVM and Logistic Regression, a thread of research has been trying to derive a *valid p.d. kernel* that resembles $S(x_i, x_j)$. A framework for designing such kernel is the *time series global-alignment kernel* proposed in [Cuturi et al., 2007] and further explored in [Cuturi, 2011]. The kernel replaces the minimum in (1) with a soft minimum that sums over all possible DTW alignments between two series x_i, x_j :

$$\begin{aligned} k(x_i, x_j) &:= \sum_{a \in \mathcal{A}(x_i, x_j)} \exp(-\tau(x_i, x_j; a)) \\ &:= \sum_{a \in \mathcal{A}(x_i, x_j)} \prod_{t=1}^{|a|} \kappa(x_i[a_1(t)], x_j[a_2(t)]) \end{aligned} \quad (2)$$

where $\kappa(.,.)$ is some local similarity function induced from the divergence τ as $\kappa = \exp(-\tau)$. The function (2) is a p.d. kernel when $\kappa(.,.)$ satisfies certain conditions [Cuturi et al., 2007]. However, it is known that a soft minimum can be orders of magnitude larger than the minimum when summing over exponentially many terms, which results in a serious *diagonally dominant problem* for the kernel (2). In other words, the kernel value between a series to itself $k(x_i, x_i)$ is orders of magnitude larger than other values $k(x_i, x_j)$. Thus in practice, one must take the log of the kernel (2) even though such operation is known to break the p.d. property [Cuturi, 2011]. In addition, the evaluation of kernel (2) requires running DP over all pairs of samples and thus gives a high complexity of $O(N^2 L^2)$.

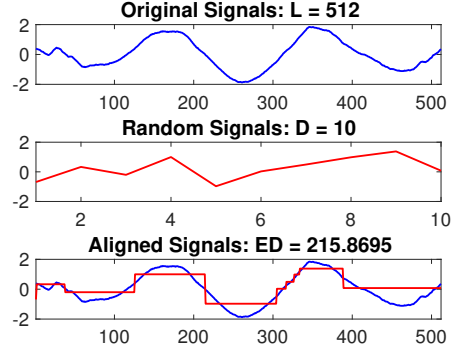


Figure 1: Example of the DTW alignment between original time series of length $L = 512$ and random time series of length $D = 10$.

3 Novel Time-Series Kernels via Alignments to Random Series

In this section, we study a new approach to build a family of p.d. kernels for time series based on DTW, inspired by the latest advancement of kernel learning methodology from distance [Wu et al., 2018].

Formally, the kernel is defined by integrating a feature map over a distribution of random time series $p(\omega)$, with each feature produced by alignments between original time series x and random series ω :

$$\begin{aligned} k(x, y) &= \int_{\omega} p(\omega) \phi_{\omega}(x) \phi_{\omega}(y) d\omega, \\ \text{where } \phi_{\omega}(x) &:= \sum_{a \in \mathcal{A}(\omega, x)} p(a|\omega) \tau(\omega, x; a). \end{aligned} \quad (3)$$

The kernel (3) enjoys several advantages. First, (3) is a p.d. kernel by its construction.

Proposition 1. *The kernel (3) is positive definite, that is, $\sum_{i=1}^N \sum_{j=1}^N c_i c_j k(x_i, x_j) \geq 0$ for any $\{c_i \mid c_i \in \mathbb{R}\}_{i=1}^N$ and any $\{x_i \mid x_i \in \mathcal{X}\}_{i=1}^N$.*

Proof. By definition (3), we have

$$\begin{aligned} \sum_{i=1}^N \sum_{j=1}^N c_i c_j k(x_i, x_j) &= \sum_{i=1}^N \sum_{j=1}^N c_i c_j \int_{\omega} p(\omega) \phi_{\omega}(x_i) \phi_{\omega}(x_j) d\omega \\ &= \int_{\omega} p(\omega) \sum_{i=1}^N \sum_{j=1}^N c_i c_j \phi_{\omega}(x_i) \phi_{\omega}(x_j) d\omega \\ &= \int_{\omega} p(\omega) \left(\sum_{i=1}^N c_i \phi_{\omega}(x_i) \right)^2 d\omega \geq 0. \end{aligned}$$

□

Secondly, by choosing

$$p(a|\omega) = \begin{cases} 1, & a^* = \arg \min_a \tau(\omega, x; a) \\ 0, & o.w. \end{cases} \quad (4)$$

one can avoid the *diagonal dominance problem* of the kernel matrix, since the kernel value between two time series $k(x, y)$ depends only on the correlation of $\tau(\omega, x; a_x)$ and $\tau(\omega, y; a_y)$ under their *optimal alignments*. It thus avoids the dominance of the diagonal terms $k(x, x)$ caused by the summation over exponentially many alignments. We can interpret the random series ω of length D as the possible *shapes* of a time series, defined by D segments, each associated with a random number. Figure 1 gives an example of a random series ω of length $D = 10$, which divides a time series x into D segments and outputs a dissimilarity score as the feature $\phi_\omega(x)$. The third advantage of (3) is its computational efficiency due to a simple *random features approximation*. Although the kernel function (3) seems hard to compute, we show that there is a low-dimensional representation of each series $T(x)$, by which one can efficiently find an approximate solution to that of the exact kernel (3) within ϵ precision. This is in contrast to the global-alignment kernel (2), where although one can evaluate the kernel matrix exactly in $O(N^2L^2)$ time, it is unclear how to efficiently find a low-rank approximation.

3.1 Computation of Random Warping Series

Although the kernel (3) does not yield a simple analytic form, it naturally yields a random approximation of the form using a simple MC method,

$$k(x, y) \approx \langle T(x), T(y) \rangle = \frac{1}{R} \sum_{i=1}^R \langle \phi_{\omega_i}(x), \phi_{\omega_i}(y) \rangle.$$

The feature vector $T(x)$ is computed using dissimilarity measure $\tau(\{\omega_i\}_{i=1}^R, x)$, where $\{\omega_i\}_{i=1}^R$ is a set of random series of variable length D with each value drawn from a distribution $p(\omega)$. In particular, the function τ could be any elastic distance measure but without loss of generality we consider DTW as our similarity measure since it has proved to be the most successful metric for time series [Wang et al., 2013, Xi et al., 2006].

Algorithm 1 summarizes the procedure to generate feature vectors for raw time series. There are several comments worth making here. First of all, the distribution of $p(\omega)$ plays an important role in capturing the global properties of original time-series. Since we explicitly define a kernel from this distribution, it is flexible to search for the best distribution that fits data well for underlying applications. In our experiments, we find the Gaussian distribution is generally applicable for time series from various applications. Specifically, the parameter σ stems from a distribution $p(\omega)$ that should well capture the characteristics of time series $\{x_i\}_{i=1}^N$. Second, as shown in Figure 1, a short random warping series could typically identify the local patterns as well as global patterns in raw time series. It

Algorithm 1 RWS Approximation: An Unsupervised Feature Representation for Time Series

Input: Time series $\{x_i\}_{i=1}^N, 1 \leq |x_i| \leq L, D_{min}, D_{max}, R, \sigma$ associated to $p(\omega)$.

Output: Feature matrix $T_{N \times R}$ for time series

- 1: **for** $j = 1, \dots, R$ **do**
 - 2: Draw D uniformly from $[D_{min}, D_{max}]$. Generate random time series ω_j of length D_j with each value drawn from distribution $p(\omega)$ normalized by σ .
 - 3: Compute a feature vector $T(:, j) = \phi_{\omega_j}(\{x_i\}_{i=1}^N)$ using DTW with or without a window size.
 - 4: **end for**
 - 5: Return feature matrix $T_{N \times R} = \frac{1}{\sqrt{R}}[T(:, 1 : R)]$
-

suggests that there are some optimal alignments that allow short random series to segment raw time series to obtain discriminatory features. In practice, there is no prior information for this optimal alignment and thus we choose to uniformly sample the length of random series between $[D_{min}, D_{max}]$ to give an unbiased estimate of D , where $D_{min} = 1$ is used in our experiments. Additional benefits lie in the fact that random series with variable lengths may simultaneously identify multi-scale patterns hidden in the raw time series.

In addition to giving a practical way to approximate the proposed kernel, applying these random series also enjoys the double benefits of reduced computation and memory consumption. Compared to the family of global alignment kernels [Cuturi et al., 2007, Cuturi, 2011], computing the dense kernel matrix $K \in \mathbb{R}^{N \times N}$ requires $O(N^2)$ times evaluation of DTW which usually takes $O(L^2)$ complexity based on DP. It also needs $O(NL + N^2)$ to store the original time series and resulting kernel matrix. In contrast, our RWS approximation only requires linear complexity of $O(NRL)$ computation and $O(NR)$ storage size, given D is a small constant. This dramatic reduction in both computation and memory storage empowers much more efficient training and testing when combining with ERM classifiers such as SVM.

3.2 Convergence of Random Warping Series

In the following, we extend standard convergence analysis of Random Features [Rahimi and Recht, 2007] from a kernel between two fixed-dimensional vectors to a kernel function measuring similarity between two time series of variable lengths. Note [Wu et al., 2018] has proposed a general analysis for any distance-based kernel through covering number w.r.t. the distance, which however, does not apply directly here since DTW is not a distance metric.

Let (A, B) be $l \times D$ and $l \times L$ matrices that map each

element of ω and x to an element of a DTW alignment path. The feature map of RWS can be expressed as

$$\phi_\omega(x) := \min_{(A,B) \in \mathcal{A}(\omega,x)} \tau(A\omega, Bx) = \sum_{i=1}^l \tau([A\omega]_i, [Bx]_i). \quad (5)$$

Note that in practice one can often convert a similarity function into a dissimilarity function to fit into the above setting. The goal is to approximate the kernel $k(x, y) := \int_\omega p(\omega) \phi_\omega(x) \phi_\omega(y) d\omega$ via a sampling approximation $s_R(x, y) = \frac{1}{R} \sum_{i=1}^R \phi_{\omega_i}(x) \phi_{\omega_i}(y)$ with $\omega_i \sim p(\omega)$. Note we have $E[s_R(x, y)] = E_{\omega_i}[\phi_{\omega_i}(x) \phi_{\omega_i}(y)] = k(x, y)$. The question is how many samples R are needed to guarantee

$$|s_R(x, y) - k(x, y)| \leq \epsilon \quad \forall x, y \in \mathcal{X} \quad (6)$$

In the standard analysis of RF, the required sample size is $\Omega(\frac{d}{\epsilon^2} \log \frac{\sigma_p \text{diam}(\mathcal{X})}{\epsilon})$ where \mathcal{X} comprises all d -dimensional vectors of diameter $\text{diam}(\mathcal{X})$. The standard analysis does not apply to our case for two reasons: (a) our domain \mathcal{X} contains time series of different lengths, and (b) our kernel involves a minimization (5) over all possible DTW alignments, and thus is not shift-invariant as required in [Rahimi and Recht, 2007]. To obtain a uniform convergence bound that could potentially handle time series of unbounded length, we introduce the notion of *minimum shape-preserving length*.

Definition 1. The Minimum Shape-Preserving Length (MSPL) d_ϵ of tolerance ϵ is the smallest L such that $\forall x \in \mathcal{X}, \exists \tilde{x} \in \mathbb{R}^L$,

$$\min_{(A,B) \in \mathcal{A}(\tilde{x},x): B=I} \|A\tilde{x} - Bx\| \leq \epsilon \quad (7)$$

where $\mathcal{A}(\tilde{x}, x)$ is the set of possible alignments between \tilde{x} and x considered by DTW, and I is an identity matrix.

In other words, d_ϵ defines the smallest length one can compress a time series to with approximation error no more than ϵ , measured by DTW in the ℓ_2 distance. Then the following gives the number of RWS required to guarantee an ϵ uniform convergence over all possible inputs $x, y \in \mathcal{X}$.

Theorem 1. Assume the ground metric $\tau(A\omega, Bx)$ satisfies $|\tau(\cdot, \cdot)| \leq \gamma$ and is Lipschitz-continuous w.r.t. x with parameter $\beta(\omega)$ where $\text{Var}[\beta(\omega)] \leq \sigma_\tau^2$. The RWS approximation with R features satisfies

$$P \left[\max_{x,y \in \mathcal{X}} |s_R(x, y) - k(x, y)| \geq 3\epsilon \right] \leq 8r^2 \left(\frac{4\gamma\sigma_\tau}{\epsilon} \right)^2 e^{-\frac{R\epsilon^2}{32\gamma^4(1+d_\epsilon)}}. \quad (8)$$

where r is the radius of time series domain \mathcal{X} in the ℓ_∞ norm and d_ϵ is the MSPL with precision ϵ .

Proof Sketch. Let $f(x, y) := s_R(x, y) - k(x, y)$. We have $E[f(x, y)] = 0$ and $|f(x, y)| \leq 2\gamma^2$ by the boundedness of function $\tau(\cdot, \cdot)$. Then by Hoeffding inequality, we have

$$P[|f(x, y)| \geq t] \leq 2 \exp(-Rt^2/8\gamma^4) \quad (9)$$

for a given pair $(x, y) \in \mathcal{X} \times \mathcal{X}$. To get a uniform bound that holds for all pairs of series $(x, y) \in \mathcal{X} \times \mathcal{X}$, consider the pair of series (\tilde{x}, \tilde{y}) of minimum shape-preserving length $d(\epsilon)$ under precision ϵ . We have an ϵ -net \mathcal{E} with $|\mathcal{E}| = (\frac{2r}{\epsilon})^d$ that covers the d -dimensional ℓ_∞ -ball of radius r . Then through union bound and (9), we have

$$P \left[\max_{\tilde{x}, \tilde{y} \in \mathcal{E}} |f(\tilde{x}, \tilde{y})| \geq t \right] \leq 2|\mathcal{E}|^2 \exp(-Rt^2/8\gamma^4). \quad (10)$$

Let $\mathcal{B}_\infty(d)$ be the d -dimensional ℓ_∞ -ball. Given any time series $x, y \in \mathcal{X}$ of arbitrary length, we can first find $\tilde{x}, \tilde{y} \in \mathcal{B}_\infty(d)$ with $\|\tilde{A}x - x\| \leq \epsilon$, $\|\tilde{A}y - y\| \leq \epsilon$ and then find $\tilde{x}_i, \tilde{y}_j \in \mathcal{E}$ such that $\|\tilde{x} - \tilde{x}_i\| \leq \epsilon$, $\|\tilde{y} - \tilde{y}_j\| \leq \epsilon$. By the result of Lemma 1 (see appendix 6.1), the closeness of (x, y) to $(\tilde{x}_i, \tilde{y}_j)$ implies the closeness of $f(x, y)$ to $f(\tilde{x}_i, \tilde{y}_j)$, which leads to

$$P[|f(x, y) - f(\tilde{x}_i, \tilde{y}_j)| \geq 2t] \leq \frac{8\gamma^2\sigma_\tau^2\epsilon^2}{t^2}. \quad (11)$$

Combining (10) and (11), we have

$$P \left[\max_{x,y \in \mathcal{X}} |f(\tilde{x}, \tilde{y})| \geq 3t \right] \leq 2 \left(\frac{2r}{\epsilon} \right)^{2d} e^{-\frac{Rt^2}{32\gamma^4}} + \frac{16\gamma^2\sigma_\tau^2\epsilon^2}{t^2}. \quad (12)$$

This is of the form $\kappa_1\epsilon^{-2d} + \kappa_2\epsilon^2$. Choosing $\epsilon = (\kappa_1/\kappa_2)^{1/(2+2d)}$ to balance the two terms in (12), the RHS becomes $2\kappa_1^{1/(1+d)}\kappa_2^{d/(1+d)}$. This yields the result

$$P \left[\max_{x,y \in \mathcal{X}} |f(x, y)| \geq 3t \right] \leq 8r^2 \left(\frac{4\gamma\sigma_\tau}{t} \right)^2 e^{-\frac{Rt^2}{32\gamma^4(1+d)}}.$$

□

The above theorem 1 shows that, to guarantee $\sup_{x,y \in \mathcal{X}} |s_R(x, y) - k(x, y)| \leq \epsilon$ with probability $1 - \delta$, it suffices to have $R = \Omega(\frac{d_\epsilon\gamma^4}{\epsilon^2} \log \frac{\gamma r \sigma_\tau}{\delta \epsilon})$. In practice, the constants r, γ are not particularly large due to the normalization on series $x, y \in \mathcal{X}$ and dissimilarity function $\tau(\cdot, \cdot)$. The main factor determining the rate of convergence is the shape-preserving length d_ϵ . Note that for problems with time series length bounded by L , we always have $d_\epsilon \leq L$, which means the number of features required would be only of order $R = \Omega(L/\epsilon^2)$.

4 Experiments

We conduct experiments to demonstrate the efficiency and effectiveness of the RWS, and compare against 9

baselines on 16 real-world datasets from the widely-used UCR time-series classification archive [Chen et al., 2015] as shown in Table 1. We evaluate RWS on the datasets with variable number and length to achieve these goals: 1) competitive or better accuracy for small problems; 2) matches or outperforms other methods in terms of both performance and runtime for middle or large scale tasks. We implement our method in Matlab and use C Mex function¹ for computationally expensive component of DTW. For other methods we use the same routine to promote a fair runtime comparison, where the window size of DTW is set as $\min(L/10, 40)$ similar to [Lei et al., 2017, Paparrizos and Gravano, 2015]. More details about datasets and parameter settings are in Appendix 6.2.

Table 1: Properties of the datasets. The number and the length of time series are sorted increasingly.

Name	C:Classes	N:Train	M:Test	L:length
Beef	5	30	30	470
DPTW	6	400	139	80
IPD	2	67	1,029	24
PPOAG	3	400	205	80
MPOC	2	600	291	80
POC	2	1,800	858	80
LKA	3	375	375	720
IWBS	11	220	1,980	256
TWOP	4	1,000	4,000	128
ECG5T	5	500	4,500	140
CHCO	3	467	3,840	166
Wafer	2	1,000	6,174	152
MALLAT	8	55	2,345	1,024
FordB	2	3636	810	500
NIFECG	42	1,800	1,965	750
HO	2	370	1,000	2,709

4.1 Effects of σ , R and D on RWS

Setup. We first perform experiments to investigate the characteristics of the RWS method by varying the kernel parameter σ , the rank R and the length D of random series. Due to limited space, we only show typical results and see Appendix 6.3 for complete ones.

Effects of σ . It is well known that the choice of the kernel parameter σ determines the quality of various kernels. Figure 2 shows that in most cases the training and testing performance curves agree well in the sense that they consistently increase at the beginning, stabilize around $\sigma = 1$ (which corresponds to the standard distribution), and finally decrease in the end. In a few cases like NIFECG, the optimal performance is slightly shifted from $\sigma = 1$. This observation is favorable since it suggests that one may easily tune our approach over a smaller interval around $\sigma = 1$ for good performance.

Effects of R . We evaluate the training and testing performance when varying the rank R from 4 to 512 with fixed σ and D . Figure 3 shows that the training

Table 2: Classification performance comparison among RWS, TSEigen, and TSMC with $R = 32$.

Classifier	RWS		TSEigen		TSMC	
	Accu	Time	Accu	Time	Accu	Time
Beef	0.733	0.3	0.633	2.1	0.433	0.6
DPTW	0.79	0.5	0.738	7.1	0.738	1.5
IPD	0.969	0.3	0.911	8.6	0.80	1.7
PPOAG	0.868	0.4	0.82	8.9	0.82	1.8
MPOC	0.711	0.8	0.653	19.3	0.653	2.4
POC	0.711	2.4	0.686	172.3	0.66	8.2
LKA	0.792	7.3	0.528	401.5	0.525	39.5
IWBS	0.619	8.9	0.633	784.6	0.57	31.9
TWOP	99.9	4.4	0.976	1395	0.946	32.8
ECG5T	0.933	10.6	0.932	1554	0.918	36.0
CHCO	0.572	6.3	0.529	1668	0.402	45.7
Wafer	0.993	9.6	0.89	3475	0.89	59.3
MALLAT	0.937	33.9	0.898	7982	0.888	282.6
FordB	0.727	43.5	0.704	10069	0.686	216.3
NIFECG	0.907	19.8	0.867	10890	0.582	265
HO	0.843	43.3	0.845	46509	0.82	979.1

and testing accuracy generally converge almost exponentially when increasing R from very small number ($R = 4$) to a relative large number ($R = 64$), and then slowly saturate to the optimal performance. Empirically, this feature is the most favorable because the performance of RWS is relatively stable even for small R . More importantly, this confirms our analysis in Theorem 1 that our RWS approximation can guarantee (rapid) convergence to the exact kernel.

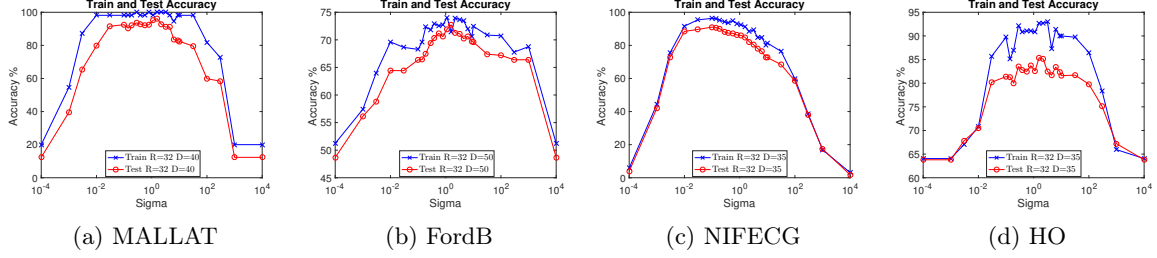
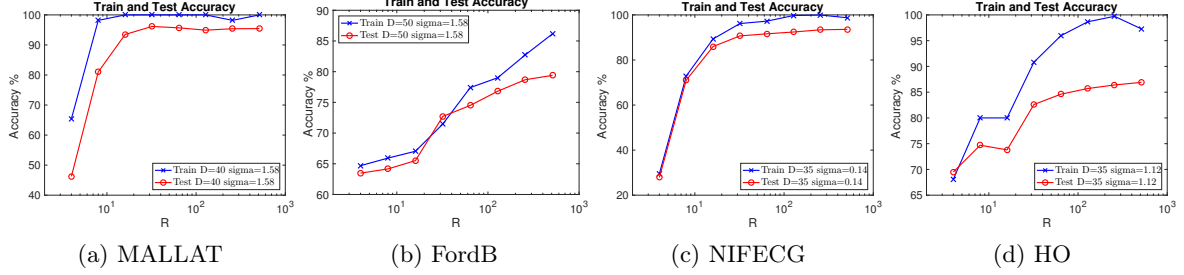
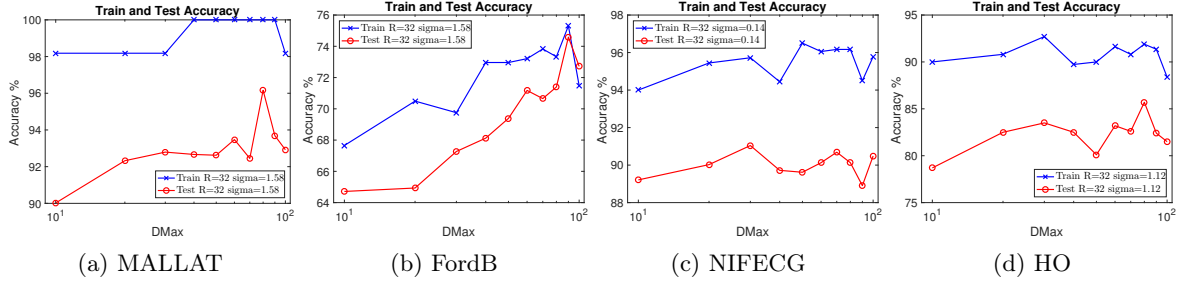
Effects of D . We investigate the effect of the length D of the random series on training and testing performance. As hinted at earlier, a key insight behind the proposed time-series kernel depends on the assumption that a random series of short length can effectively segment raw time series in a way that captures its patterns. Figure 4 shows that although testing accuracy seems to fluctuate when varying D_{max} from 10 to 100, it is clear that the near-peak performance can be achieved when D_{max} is small in the most of cases.

4.2 Comparing Feature Representations

Baselines and Setup. We compare our approach with two recently developed methods: 1) TSEigen [Hayashi et al., 2005]: learn a low-rank feature representation for a similarity matrix computed using DTW distance through Singular Value Decomposition [Wu and Stathopoulos, 2015, Wu et al., 2017]; 2) TSMC [Lei et al., 2017]: a recently proposed similarity preserving representation for DTW-based similarity matrix using matrix completion approach. We set $R = 32$ for all methods. We employ a linear SVM implemented in LIBLINEAR [Fan et al., 2008] since it can separate the effectiveness of the feature representation from the power of the nonlinear learning solvers.

Results. Table 2 clearly demonstrates the significant advantages of our approach compared to other representations in terms of both classification accuracy and computational time. Indeed, TSMC improves the

¹<https://www.mathworks.com/matlabcentral/fileexchange/43156-dynamic-time-warping-dtw>


 Figure 2: Train (Blue) and test (Red) accuracy when varying σ with fixed D and R .

 Figure 3: Train (Blue) and test (Red) accuracy when varying R with fixed σ and D .

 Figure 4: Train (Blue) and test (Red) accuracy when varying D with fixed σ and R .

computational efficiency compared to TSEigen without compromising large loss of the accuracy as claimed in [Lei et al., 2017]. However, RWS is corroborated to achieve both higher accuracy and faster train and testing time compared to TSMC and TSEigen. The improved accuracy of RWS suggests that a truly p.d. time series kernel admits better feature representations than those obtained from a similarity or kernel (not p.d.) matrix. In addition, improved computational time illustrates the effectiveness of using random series to approximate the exact kernel.

4.3 Comparing Time-Series Classification

Baselines. We now compare our method with other state-of-the-art time series classification methods that also take advantage of DTW distance or employ DTW-like kernels: 1) 1NN-DTW: use window size $\min(L/10, 40)$; 2) 1NN-DTW^{opt}: use optimal window size using leave-one-out cross validation from test data in [Chen et al., 2015] 3) DTWF [Kate, 2016]: a re-

cently proposed method that combines DTW without and with constraints and SAX [Lin et al., 2007] as features; 4) TGAK [Cuturi, 2011]: a fast triangular global alignment kernel for time-series; 5) RWS(LR): RWS with large rank that achieves the best accuracy with more computational time; 6) RWS(SR): small rank that obtains comparable accuracy in less time. We conduct grid search for important parameters in each method suggested in [Kate, 2016, Cuturi, 2011].

Results. Table 3 corroborates that RWS consistently outperforms or matches other state-of-the-art methods in terms of testing accuracy while requiring significantly less computational time. First, RWS(SR) can achieve better or similar performance compared to 1NN-DTW and 1NN-DTW^{opt} for all datasets. This is a strong sign that our learned feature representation is very effective, since, using it, even a linear SVM can beat the well-recognized benchmark. Meanwhile, the clear computational advantages of RWS over 1NN-DTW can be observed when the number or the length of time series samples become large. This is not surprising since

Table 3: Classification performance comparison among methods using DTW or DTW-like kernels.

Classifier	RWS(LR)		RWS(SR)		1NN-DTW		1NN-DTW ^{opt}		TGAK		DTWF	
Dataset	Accu	Time	Accu	Time	Accu	Time	Accu	Time	Accu	Time	Accu	Time
Beef	0.767	0.8	0.733	0.3	0.567	1.1	0.633	0.3	0.633	24.7	0.60	3.7
DPTW	0.865	4.2	0.80	0.2	0.73	1.4	0.718	0.8	0.738	27.9	0.77	3.0
IPD	0.965	1.0	0.962	0.4	0.947	55.3	0.962	56.0	0.739	3.7	0.953	0.5
PPOAG	0.868	0.3	0.859	0.2	0.776	2.0	0.785	1.2	0.854	118.2	0.829	9.7
MPOC	0.773	6.8	0.708	0.8	0.635	4.4	0.663	2.7	0.627	117.3	0.653	10.2
POC	0.815	38.2	0.746	4.7	0.721	36.9	0.751	20.1	0.613	2373	0.79	202.7
LKA	0.84	54.9	0.816	13.6	0.712	97.7	0.837	573.6	0.645	13484	0.80	1220
IWBS	0.641	132.4	0.619	8.8	0.504	70.9	0.589	36.1	0.126	2413	0.609	260.3
TWOP	1	16.1	0.999	4.4	1	222.2	1	157.5	0.269	5690	1	481.7
ECG5T	0.94	9.2	0.934	4.9	0.928	137.8	0.928	70.1	0.927	2822	0.933	278.3
CHCO	0.777	189.1	0.683	48.1	0.627	160.8	0.627	57.0	0.545	3122	0.666	333.6
Wafer	0.995	143.6	0.993	9.6	0.986	412.3	0.996	210.1	0.896	11172	0.994	980.5
MALLAT	0.952	72.8	0.937	33.8	0.937	150.3	0.925	65.5	0.257	11882	0.915	988.4
FordB	0.793	543.8	0.62	5.6	0.589	1476	0.581	577.6	N/A	N/A	0.83	8402
NIFECG	0.936	140.2	0.903	20.0	0.845	2699	0.857	1432	N/A	N/A	0.906	32493
HO	0.871	336.9	0.834	41.9	0.816	4883	0.807	5837	N/A	N/A	0.898	40407

Table 4: Clustering performance comparison among different methods.

Clustering	RWS(LR)		RWS(SR)		KMeans-DTW		CLDS		K-Shape	
Dataset	NMI	Time	NMI	Time	NMI	Time	NMI	Time	NMI	Time
Beef	0.29	1.1	0.27	1.0	0.25	377	0.24	61.3	0.22	1.8
DPTW	0.52	0.6	0.56	0.5	0.55	182	0.55	176.8	0.45	14.9
PPOAG	0.56	0.5	0.54	0.2	0.44	105.4	0.55	191.1	0.27	40.2
IWBS	0.43	43.9	0.36	6.3	0.37	5676	0.38	1109	0.43	377.6
TWOP	0.23	11.2	0.3	4.7	0.12	1960	0.02	1312	0.4	292.1
ECG5T	0.46	25.7	0.4	7.0	0.48	2539	0.37	1308	0.35	360.7
MALLAT	0.92	48.2	0.91	25.4	0.72	95218	0.92	2448	0.75	900.4
NIFECG	0.71	346.1	0.68	43.7	0.63	101473	0.67	3442	0.73	5387

RWS reduces both number and length of time series from quadratic complexity to linear complexity. Second, RWS is much better than another family of time series kernels represented by TGAK, which probably indicates that considering the soft-minimum of all alignment distances does not capture well hidden patterns of time series. Third, DTWF shows significant performance difference compared to 1NN-DTW, which is consistent with the reported results in [Kate, 2016]. However, compared to DTWF, RWS(LR) can still show clear advantages in accuracy among 11 cases out of the total 16 datasets while achieving one or two orders of magnitude speedup. More importantly, RWS can support a trade-off between the accuracy and run-time. This feature is highly desirable in real applications that may have a variety of priorities and constraints.

4.4 Comparing Time-Series Clustering

Baselines. We compare our method against several time-series clustering baselines: 1) KMeans-DTW [Petitjean et al., 2011, Paparrizos and Gravano, 2015]: accelerate computation with lower bounding approach LB_{Keogh} [Keogh, 2002]; 2) CLDS [Li and Prakash, 2011]: learns a feature representation with R hidden variables through complex-valued linear dynamical systems; 3) K-Shape [Paparrizos and Gravano, 2015]: recently proposed clustering method demonstrated to outperform state-of-the-art clustering approaches in accuracy and computational time; 4) RWS(LR); 5) RWS(SR). We combine our learned feature represen-

tation with the classic KMeans algorithm [Hartigan and Wong, 1979]. We employ a commonly used clustering metric, the normalized mutual information (NMI scaling between 0 and 1) to measure the performance, where higher value indicates better accuracy.

Results. Table 4 shows that RWS provides similar or better performance and typically is substantially faster than KMeans-DTW when the number or the length of time-series become large. In addition, RWS can consistently outperform CLDS in terms of both accuracy and runtime. Interestingly, even compared to the state-of-the-art method K-Shape, RWS can still yield a clear advantage in terms of accuracy; RWS yields 5 wins, 1 even, and 2 loses over K-Shape for 8 datasets. Besides its accuracy, the better computational efficiency of RWS over K-Shape is also corroborated.

5 Conclusions and Future Work

In this work, we have studied an effective and scalable time-series (p.d.) kernel for large-scale time series problems based on RWS approximation, and the feature embedding generated by the technique is generally applicable to most of learning problems. There are several interesting directions of future work, including: i) studying the effects of different random time-series distribution $p(\omega)$ and ii) exploring more elastic dissimilarity measure between time series such as CID and DTDC.

References

- Anthony Bagnall, Jason Lines, Aaron Bostrom, James Large, and Eamonn Keogh. The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery*, pages 1–55, 2016.
- Claus Bahlmann, Bernard Haasdonk, and Hans Burkhardt. Online handwriting recognition with support vector machines—a kernel approach. In *Frontiers in Handwriting Recognition*, pages 49–54. IEEE, 2002.
- Gustavo EAPA Batista, Eamonn J Keogh, Oben Moses Tataw, and Vinicius MA De Souza. Cid: an efficient complexity-invariant distance for time series. *Data Mining and Knowledge Discovery*, 28(3):634–669, 2014.
- Mustafa Gokce Baydogan, George Runger, and Eugene Tuv. A bag-of-features framework to classify time series. *IEEE transactions on pattern analysis and machine intelligence*, 35(11):2796–2802, 2013.
- Donald J Berndt and James Clifford. Using dynamic time warping to find patterns in time series. In *KDD workshop*, volume 10, pages 359–370. Seattle, WA, 1994.
- Jie Chen, Lingfei Wu, Kartik Audhkhasi, Brian Kingsbury, and Bhuvana Ramabhadhari. Efficient one-vs-one kernel ridge regression for speech recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 2454–2458. IEEE, 2016.
- Yanping Chen, Eamonn Keogh, Bing Hu, Nurjahan Begum, Anthony Bagnall, Abdullah Mueen, and Gustavo Batista. The ucr time series classification archive, July 2015. www.cs.ucr.edu/~eamonn/time_series_data/.
- Marco Cuturi. Fast global alignment kernels. In *Proceedings of the 28th international conference on machine learning*, pages 929–936, 2011.
- Marco Cuturi, Jean-Philippe Vert, Oystein Birkenes, and Tomoko Matsui. A kernel for time series based on global alignments. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages II–413. IEEE, 2007.
- Houtao Deng, George Runger, Eugene Tuv, and Martynov Vladimir. A time series forest for classification and feature extraction. *Information Sciences*, 239:142–153, 2013.
- Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. Liblinear: A library for large linear classification. *Journal of machine learning research*, 9(Aug):1871–1874, 2008.
- Tomasz Górecki and Maciej Łuczak. Non-isometric transforms in time series classification using dtw. *Knowledge-Based Systems*, 61:98–108, 2014.
- Steinn Gudmundsson, Thomas Philip Runarsson, and Sven Sigurdsson. Support vector machines and dynamic time warping for time series. In *2008 IEEE International Joint Conference on Neural Networks*, pages 2772–2776. IEEE, 2008.
- John A Hartigan and Manchek A Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.
- Akira Hayashi, Yuko Mizuhara, and Nobuo Suematsu. Embedding time series data for classification. In *International Workshop on Machine Learning and Data Mining in Pattern Recognition*, pages 356–365. Springer, 2005.
- Rohit J Kate. Using dynamic time warping distances as features for improved time series classification. *Data Mining and Knowledge Discovery*, 30(2):283–312, 2016.
- Eamonn Keogh. Exact indexing of dynamic time warping. In *Proceedings of the 28th international conference on Very Large Data Bases*, pages 406–417. VLDB Endowment, 2002.
- Qi Lei, Jinfeng Yi, Roman Vaculin, Lingfei Wu, and Inderjit S. Dhillon. Similarity preserving representation learning for time series analysis. <https://arxiv.org/abs/1702.03584>, 2017.
- Christina S Leslie, Eleazar Eskin, and William Stafford Noble. The spectrum kernel: A string kernel for svm protein classification. In *Pacific symposium on biocomputing*, volume 7, pages 566–575, 2002.
- Lei Li and B Aditya Prakash. Time series clustering: Complex is simpler! In *Proceedings of the 28th International Conference on Machine Learning*, pages 185–192, 2011.
- Jessica Lin, Eamonn Keogh, Li Wei, and Stefano Lonardi. Experiencing sax: a novel symbolic representation of time series. *Data Mining and knowledge discovery*, 15(2):107–144, 2007.
- Pierre-François Marteau and Sylvie Gibet. On recursive edit distance kernels with application to time series classification. *IEEE transactions on neural networks and learning systems*, 26(6):1121–1133, 2015.
- John Paparrizos and Luis Gravano. k-shape: Efficient and accurate clustering of time series. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pages 1855–1870. ACM, 2015.
- Xi Peng, Junzhou Huang, Qiong Hu, Shaoting Zhang, Ahmed Elgammal, and Dimitris Metaxas. From cir-

- cle to 3-sphere: Head pose estimation by instance parameterization. *Computer Vision and Image Understanding*, 136:92–102, 2015a.
- Xi Peng, Shaoting Zhang, Yu Yang, and Dimitris N Metaxas. Piefa: Personalized incremental and ensemble face alignment. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3880–3888, 2015b.
- François Petitjean, Alain Ketterlin, and Pierre Gançarski. A global averaging method for dynamic time warping, with applications to clustering. *Pattern Recognition*, 44(3):678–693, 2011.
- Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, volume 3, page 5, 2007.
- Thanawin Rakthanmanon and Eamonn Keogh. Fast shapelets: A scalable algorithm for discovering time series shapelets. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pages 668–676. SIAM, 2013.
- Thanawin Rakthanmanon, Bilson Campana, Abdullah Mueen, Gustavo Batista, Brandon Westover, Qiang Zhu, Jesin Zakaria, and Eamonn Keogh. Searching and mining trillions of time series subsequences under dynamic time warping. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 262–270. ACM, 2012.
- Hiroaki Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal processing*, 26(1):43–49, 1978.
- Patrick Schäfer. The boss is concerned with time series classification in the presence of noise. *Data Mining and Knowledge Discovery*, 29(6):1505–1530, 2015.
- Pavel Senin and Sergey Malinchik. Sax-vsm: Interpretable time series classification using sax and vector space model. In *Proceedings of the 13th IEEE International Conference on Data Mining*, pages 1175–1180. IEEE, 2013.
- Hiroshi Shimodaira, Ken-ichi Noma, Mitsuru Nakai, Shigeki Sagayama, et al. Dynamic time-alignment kernel in support vector machine. In *Advances in Neural Information Processing Systems*, volume 2, pages 921–928, 2001.
- Xiaoyue Wang, Abdullah Mueen, Hui Ding, Goce Trajcevski, Peter Scheuermann, and Eamonn Keogh. Experimental comparison of representation methods and distance measures for time series data. *Data Mining and Knowledge Discovery*, pages 1–35, 2013.
- Lingfei Wu and Andreas Stathopoulos. A preconditioned hybrid svd method for accurately computing singular triplets of large matrices. *SIAM Journal on Scientific Computing*, 37(5):S365–S388, 2015.
- Lingfei Wu, Ian EH Yen, Jie Chen, and Rui Yan. Revisiting random binning features: Fast convergence and strong parallelizability. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1265–1274. ACM, 2016.
- Lingfei Wu, Eloy Romero, and Andreas Stathopoulos. Primme_svds: A high-performance preconditioned svd solver for accurate large-scale computations. *SIAM Journal on Scientific Computing*, 39(5): S248–S271, 2017.
- Lingfei Wu, Ian En-Hsu Yen, Fnagli Xu, Pradeep Ravikumar, and Witbrock Michael. D2ke: From distance to kernel and embedding. <https://arxiv.org/abs/1802.04956>, 2018.
- Xiaopeng Xi, Eamonn Keogh, Christian Shelton, Li Wei, and Chotirat Ann Ratanamahatana. Fast time series classification using numerosity reduction. In *Proceedings of the 23rd international conference on Machine learning*, pages 1033–1040. ACM, 2006.
- Zhengzheng Xing, Jian Pei, and Eamonn Keogh. A brief survey on sequence classification. *ACM Sigkdd Explorations Newsletter*, 12(1):40–48, 2010.
- Lexiang Ye and Eamonn Keogh. Time series shapelets: a new primitive for data mining. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 947–956. ACM, 2009.
- Ian E.H. Yen, Ting-Wei Lin, Shou-De Lin, Pradeep Ravikumar, and Inderjit S. Dhillon. Sparse random features algorithm as coordinate descent in hilbert space. In *Advances in Neural Information Processing Systems*, 2014.