Differentiable Divergences Between Time Series

Mathieu Blondel
Google Research, Brain team

Arthur Mensch
École Normale Supérieure

Jean-Philippe Vert
Google Research, Brain team

Abstract

Computing the discrepancy between time series of variable sizes is notoriously challenging. While dynamic time warping (DTW) is popularly used for this purpose, it is not differentiable everywhere and is known to lead to bad local optima when used as a "loss". Soft-DTW addresses these issues, but it is not a positive definite divergence: due to the bias introduced by entropic regularization, it can be negative and it is not minimized when the time series are equal. We propose in this paper a new divergence, dubbed soft-DTW divergence, which aims to correct these issues. We study its properties; in particular, under conditions on the ground cost, we show that it is non-negative and minimized when the time series are equal. We also propose a new "sharp" variant by further removing entropic bias. We showcase our divergences on time series averaging and demonstrate significant accuracy improvements compared to both DTW and soft-DTW on 84 time series classification datasets.

1 Introduction

Designing a meaningful discrepancy or "loss" between two sequences of variable lengths and integrating it in an end-to-end differentiable pipeline is challenging. For sequences on finite alphabets, differentiable local alignment kernels (Saigo et al., 2006) and edit distances (McCallum et al., 2012) have been proposed. For sequences on continuous domains, connectionist temporal classification (CTC) is popularly used in speech recognition (Graves et al., 2006). A related approach for time series motivated by geometry is dynamic time warping (DTW), which seeks a minimum-cost alignment between time series and can be computed by dynamic programming in quadratic time (Sakoe and Chiba, 1978). However, DTW is not differentiable everywhere, is sensitive to noise and is known to lead to bad local optima when used as a loss. Soft-DTW (Cuturi and Blondel, 2017) addresses these issues by replacing the minimum over alignments with a soft minimum, which has the effect of inducing a probability distribution over all alignments. Despite considering all alignments, it is shown that soft-DTW can still be computed by dynamic programming in the same complexity. Since then, soft-DTW has been successfully applied for attention models in neural machine translation (Mensch and Blondel, 2018), video segmentation (Chang et al., 2019), spatial-temporal sequences (Janati et al., 2020), and end-to-end differentiable text-to-speech synthesis (Donahue et al., 2020), to name but a few examples. Soft-DTW is included in popular R and Python packages for time series analysis (Sardá-Espinosa, 2017; Tavenard et al., 2020).

In this paper, we show that, despite recent successes, soft-DTW has some limitations which have been overlooked in the literature. First, it can be negative, which is a nuisance when used as a loss. Second, and more problematically, when used with a squared Euclidean cost, we show that it is never minimized when the two time series are equal. Put differently, given an input time series, the closest time series in the soft-DTW sense is never the input time series. This is due to the entropic bias introduced by replacing the minimum with a soft one. We propose in this paper a new divergence, dubbed soft-DTW divergence, which is based on soft-DTW but corrects for these issues. We study its properties; in particular, under condition on the ground cost, we show that it is non-negative and minimized when the two time series are equal. Our approach is related to Sinkhorn divergences (Ramdas et al., 2017; Genevay et al., 2018; Feydy et al., 2019), which use similar correction terms as we do for optimal transport distances, but our proof techniques are completely different. We also propose a new "sharp" variant by further removing entropic bias. We showcase our divergences on time series averaging and demonstrate significant accuracy improvements compared to both DTW and soft-DTW on 84 time series classification datasets.

The rest of the paper is organized as follows. After reviewing some background in §2, we introduce the soft-DTW divergence and its "sharp" variant in §3. We study their properties and limit behavior. We study their empirical performance in §4 with experiments on time series averaging, interpolation and classification.

2 Background

2.1 Dynamic time warping

Let $\boldsymbol{X} \in \mathbb{R}^{m \times d}$ and $\boldsymbol{Y} \in \mathbb{R}^{n \times d}$ be two d-dimensional time series of lengths m and n. We denote their elements by $\boldsymbol{x}_i \in \mathbb{R}^d$ and $\boldsymbol{y}_j \in \mathbb{R}^d$, for $i \in [m]$ and $j \in [n]$. We say that $\boldsymbol{A} \in \{0,1\}^{m \times n}$ is an alignment matrix between $\boldsymbol{X}$ and $\boldsymbol{Y}$ when $[\boldsymbol{A}]_{i,j} = 1$ if $\boldsymbol{x}_i$ is aligned with $\boldsymbol{y}_j$ and 0 otherwise. We say that $\boldsymbol{A}$ is a monotonic alignment matrix if the 1's in $\boldsymbol{A}$ form a path starting from the upper-left corner $(1,1)$ that connects the lower-right corner (m,n) using only $\downarrow$, $\rightarrow$, $\searrow$ moves. We denote the set of all such monotonic alignment matrices by $\mathcal{A}(m,n) \subset \{0,1\}^{m \times n}$. The cardinality $|\mathcal{A}(m,n)|$ grows exponentially in $\min(m,n)$ and is equal to the Delannoy number, Delannoy$(m-1, n-1)$, named after French amateur mathematician Henri Delannoy (Sulanke, 2003; Banderier and Schwer, 2005).

Let $C \colon \mathbb{R}^{m \times d} \times \mathbb{R}^{n \times d} \to \mathbb{R}^{m \times n}$ be a function which maps $\boldsymbol{X} \in \mathbb{R}^{m \times d}$ and $\boldsymbol{Y} \in \mathbb{R}^{n \times d}$ to a distance or cost matrix $\boldsymbol{C} = C(\boldsymbol{X}, \boldsymbol{Y}) \in \mathbb{R}^{m \times n}$. A popular choice is the squared Euclidean cost

$$[C(\boldsymbol{X},\boldsymbol{Y})]_{i,j} = \frac{1}{2}\|\boldsymbol{x}_i - \boldsymbol{y}_j\|^2 \quad i \in [m], j \in [n]. \quad (1)$$

The Frobenius inner product $\langle \boldsymbol{A}, \boldsymbol{C} \rangle \coloneqq \mathrm{Trace}(\boldsymbol{C}^\top \boldsymbol{A})$ between $\boldsymbol{C}$ and $\boldsymbol{A}$ is the sum of the costs along the alignment (Figure 1). Dynamic time warping (Sakoe and Chiba, 1978) can then be naturally formulated as the minimum cost among all possible alignments,

$$\textsc{dtw}(\boldsymbol{C}) \coloneqq \min_{\boldsymbol{A} \in \mathcal{A}(m,n)} \langle \boldsymbol{A}, \boldsymbol{C} \rangle. \quad (2)$$

The corresponding optimal alignment is

$$\boldsymbol{A}^\star(\boldsymbol{C}) \in \operatorname*{argmin}_{\boldsymbol{A} \in \mathcal{A}(m,n)} \langle \boldsymbol{A}, \boldsymbol{C} \rangle. \quad (3)$$

Despite the exponential number of alignments, (2) and (3) can be computed in $O(mn)$ time using dynamic programming and backtracking, respectively. The quantity $\textsc{dtw}(C(\boldsymbol{X},\boldsymbol{Y}))$ is popularly used as a discrepancy measure between time series in numerous applications. In the rest of the paper, we will make the following assumptions about C:

- A.1. $C(\boldsymbol{X},\boldsymbol{Y}) \geq \mathbf{0}_{m \times n}$ (non-negativity),
- A.2. $[C(\boldsymbol{X},\boldsymbol{X})]_{i,i} = 0$ for all $i \in [m]$,

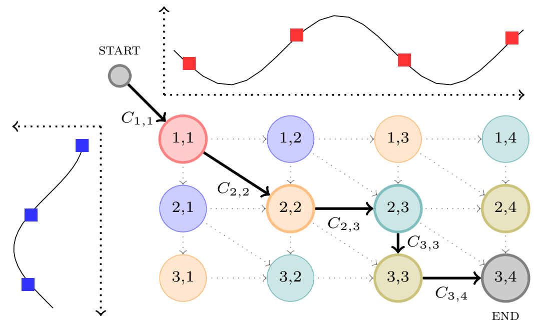

Figure 1: An alignment between two time series $\boldsymbol{X} \in \mathbb{R}^{m \times d}$ and $\boldsymbol{Y} \in \mathbb{R}^{n \times d}$ corresponds to a path in a directed acyclic graph (DAG) and can be encoded as a binary matrix $\boldsymbol{A} \in \{0,1\}^{m \times n}$. The sum of the costs along the path is then $\langle \boldsymbol{A}, \boldsymbol{C} \rangle$. DTW seeks the minimum-cost alignment, while soft-DTW induces a Gibbs distribution over alignments.

- A.3. $C(\boldsymbol{X},\boldsymbol{Y}) = C(\boldsymbol{Y},\boldsymbol{X})^\top$ (symmetry).

The properties of $\textsc{dtw}$ under these assumptions are summarized in Table 1.

2.2 Soft dynamic time warping

Definitions. In order to obtain a fully differentiable discrepancy measure between time series, Cuturi and Blondel (2017) proposed to replace the min operator in (2) by a smooth one,

$$\min_{x \in \mathcal{S}}{}_{\gamma}\; f(x) \coloneqq -\gamma \log \sum_{x \in \mathcal{S}} \exp(-f(x)/\gamma),$$

where $\gamma > 0$ is a parameter which controls the trade-off between approximation and smoothness. For convenience, we define the extension $\min_0 \coloneqq \min$. The resulting "soft" dynamic time warping formulation is

$$\begin{aligned} \textsc{sdtw}_\gamma(\boldsymbol{C}) &\coloneqq \min_{\boldsymbol{A} \in \mathcal{A}(m,n)}{}_{\gamma} \; \langle \boldsymbol{A}, \boldsymbol{C} \rangle \\ &= -\gamma \log \sum_{\boldsymbol{A} \in \mathcal{A}(m,n)} \exp(-\langle \boldsymbol{A}, \boldsymbol{C} \rangle/\gamma). \quad (4) \end{aligned}$$

Instead of only considering the minimum-cost alignment as in (2), (4) induces a Gibbs distribution over alignments. The probability of $\boldsymbol{A}$ given $\boldsymbol{C} \in \mathbb{R}^{m \times n}$ is

$$\mathbb{P}_\gamma(\boldsymbol{A}; \boldsymbol{C}) \coloneqq \frac{\exp(-\langle \boldsymbol{A}, \boldsymbol{C} \rangle/\gamma)}{\sum_{\boldsymbol{A}' \in \mathcal{A}(m,n)} \langle -\langle \boldsymbol{A}', \boldsymbol{C} \rangle/\gamma)} \in (0,1]. \quad (5)$$

We can see (4) as the negative log-partition of (5). For convenience, we also gather the probabilities of all possible alignments in a vector

$$\boldsymbol{p}_\gamma(\boldsymbol{C}) \coloneqq (\mathbb{P}_\gamma(\boldsymbol{A}; \boldsymbol{C}))_{\boldsymbol{A} \in \mathcal{A}(m,n)} \in \triangle^{|\mathcal{A}(m,n)|},$$

Table 1: Properties of time-series losses under assumptions A.1-A.3 and differentiability of C. For the soft-DTW divergence, we prove the first two properties using the cost (10) and 1d absolute value (11). For the soft-DTW and sharp divergences with the squared Euclidean cost (1), we prove that $\boldsymbol{X} = \boldsymbol{Y}$ is a stationary point.

	Non-negativity	Minimized at $\boldsymbol{X} = \boldsymbol{Y}$	Symmetry	Differentiable everywhere
DTW	✓	✓	✓	×
Soft-DTW	×	×	✓	✓
Sharp soft-DTW	✓	×	✓	✓
Soft-DTW divergence	✓	✓	✓	✓
Sharp divergence	✓	✓	✓	✓
Mean-cost divergence	✓	✓	✓	✓

where $\triangle^k := \{\boldsymbol{p} \in \mathbb{R}^k : \boldsymbol{p} \geq \boldsymbol{0}_k, \boldsymbol{p}^\top \boldsymbol{1}_k = 1\}$ is the probability simplex. Let A be a random variable distributed according to (5). The expected alignment matrix under the Gibbs distribution induced by $\boldsymbol{C}$ is

$$\boldsymbol{E}_\gamma(\boldsymbol{C}) := \mathbb{E}_\gamma[A; \boldsymbol{C}] = \sum_{\boldsymbol{A} \in \mathcal{A}(m,n)} \mathbb{P}_\gamma(\boldsymbol{A}; \boldsymbol{C}) \boldsymbol{A} \in (0,1]^{m \times n}.$$

Note that because the matrices in $\mathcal{A}(m,n)$ are binary ones, $[\boldsymbol{E}_\gamma(\boldsymbol{C})]_{i,j}$ is also equal to the marginal probability $\mathbb{P}_\gamma(A_{i,j} = 1; \boldsymbol{C})$, i.e., the probability that any of the paths goes through the cell (i,j).

Properties. The following proposition summarizes known properties of SDTW_γ (Cuturi and Blondel, 2017; Mensch and Blondel, 2018).

Proposition 1. *Properties of* SDTW_γ

The following properties hold for all $\boldsymbol{C} \in \mathbb{R}^{m \times n}$.

1. **Gradient:** $\text{SDTW}_\gamma(\boldsymbol{C})$ *is differentiable everywhere and its gradient is the expected alignment,*

$$\nabla_{\boldsymbol{C}} \text{SDTW}_\gamma(\boldsymbol{C}) = \boldsymbol{E}_\gamma(\boldsymbol{C}) \in (0,1]^{m \times n}.$$

2. **Concavity:** $\text{SDTW}_\gamma(\boldsymbol{C})$ *is concave in* $\boldsymbol{C}$.
3. **Variational form:** *letting* $H(\boldsymbol{p}) = -\langle \boldsymbol{p}, \log \boldsymbol{p} \rangle$,

$$\text{SDTW}_\gamma(\boldsymbol{C}) = \min_{\boldsymbol{p} \in \triangle^{|\mathcal{A}(m,n)|}} \langle \boldsymbol{p}, \boldsymbol{s}(\boldsymbol{C}) \rangle - \gamma H(\boldsymbol{p}) \tag{6}$$

where $\boldsymbol{s}(\boldsymbol{C}) := (\langle \boldsymbol{A}, \boldsymbol{C} \rangle)_{\boldsymbol{A} \in \mathcal{A}(mmn)} \in \mathbb{R}^{|\mathcal{A}(m,n)|}$.
4. **Scaling:** $\text{SDTW}_\gamma(\boldsymbol{C}) = \gamma \text{SDTW}_1(\boldsymbol{C}/\gamma)$, $\boldsymbol{E}_\gamma(\boldsymbol{C}) = \boldsymbol{E}_1(\boldsymbol{C}/\gamma)$ *and* $\boldsymbol{p}_\gamma(\boldsymbol{C}) = \boldsymbol{p}_1(\boldsymbol{C}/\gamma)$.
5. **Asymptotics:** $\text{DTW}(\boldsymbol{C}) \xleftarrow[0 \leftarrow \gamma]{} \text{SDTW}_\gamma(\boldsymbol{C})$ *and* $\boldsymbol{A}^\star(\boldsymbol{C}) \xleftarrow[0 \leftarrow \gamma]{} \boldsymbol{E}_\gamma(\boldsymbol{C})$.
6. **Lower and upper bounds:**

$$\text{DTW}(\boldsymbol{C}) - \gamma \log |\mathcal{A}(m,n)| \leq \text{SDTW}_\gamma(\boldsymbol{C}) \leq \text{DTW}(\boldsymbol{C}).$$

Note that $\text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))$ is generally neither convex nor concave in $\boldsymbol{X}$ and $\boldsymbol{Y}$, as is the case when C is the squared Euclidean cost (1). A notable exception is $C(\boldsymbol{X}, \boldsymbol{Y}) = -\boldsymbol{X}\boldsymbol{Y}^\top$, for which $\text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))$ is concave in $\boldsymbol{X}$ and $\boldsymbol{Y}$ (separately).

Computation. Surprisingly, even though (4) contains a sum over all $\boldsymbol{A}$ in $\mathcal{A}(m,n)$, it can be computed in $O(mn)$ time by simply replacing the min operator with $\min_\gamma$ in the original dynamic programming recursion (Cuturi and Blondel, 2017). The equivalence between (4) and this "locally smoothed" recursion was later formally proved using the associativity of the $\min_\gamma$ operator (Mensch and Blondel, 2018). The expected alignment can also be computed in $O(mn)$ time by backpropagation through the dynamic programming recursion (Cuturi and Blondel, 2017).

Use as a loss function. The differentiability of SDTW_γ makes it particularly suitable to use as a loss function between time series, of potentially variable lengths. An example of application is the computation of Fréchet means (1948) with respect to SDTW_γ. Specifically, given a set of k time series $\boldsymbol{Y}_1 \in \mathbb{R}^{n_1 \times d}$, $\ldots$, $\boldsymbol{Y}_k \in \mathbb{R}^{n_k \times d}$, we compute its average (barycenter) according to SDTW_γ by solving

$$\underset{\boldsymbol{X} \in \mathbb{R}^{m \times d}}{\operatorname{argmin}} \sum_{i=1}^{k} w_i \, \text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}_i)), \tag{7}$$

where $\boldsymbol{w} \in \mathbb{R}^k$ is a vector of pre-defined weights. When the time series $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_k$ have different lengths, a typical choice would be $w_i = 1/n_i$, to compensate for the fact that SDTW_γ increases (roughly linearly) with the length of the time series. Although it is non-convex, objective (7) can be solved approximately by gradient-based methods. Compared to DTW barycenter averaging (DBA) (Petitjean et al., 2011), it was shown that smoothing helps to avoid bad local optima. Using the chain rule and item 1 of Proposition 1, the gradient of

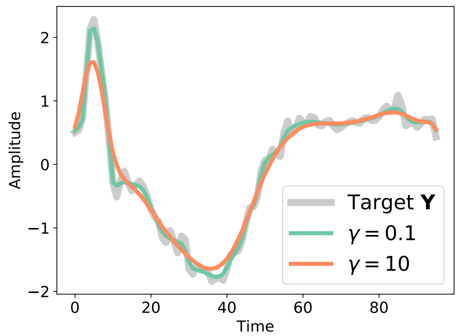

Figure 2: **Denoising effect of soft-DTW.** We show the result of $\operatorname{argmin}_{\boldsymbol{X}} \mathrm{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))$, solved by L-BFGS with $\boldsymbol{X} = \boldsymbol{Y}$ as initialization, for two values of γ. As stated in Proposition 2, SDTW_γ with $\gamma > 0$ and squared Euclidean cost never achieves its minimum at $\boldsymbol{X} = \boldsymbol{Y}$. While this denoising can be useful, this means that SDTW_γ is not a positive divergence.

$\mathrm{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))$ w.r.t. $\boldsymbol{X}$ is

$$\nabla_{\boldsymbol{X}} \mathrm{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y})) = (J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{Y}))^\top \boldsymbol{E}_\gamma(\boldsymbol{C}(\boldsymbol{X}, \boldsymbol{Y})). \tag{8}$$

Here, we assume that C is differentiable and $J_{\boldsymbol{X}}$ denotes the Jacobian matrix of $\boldsymbol{C}(\boldsymbol{X}, \boldsymbol{Y})$ w.r.t. $\boldsymbol{X}$, a linear map from $\mathbb{R}^{m\times d}$ to $\mathbb{R}^{m\times n}$ (its transpose is a linear map from $\mathbb{R}^{m\times n}$ to $\mathbb{R}^{m\times d}$).

2.3 Global alignment kernel

Although it was introduced before soft dynamic time warping, the global alignment kernel (Cuturi et al., 2007) can be naturally expressed using SDTW_γ as

$$K_\gamma^C(\boldsymbol{X}, \boldsymbol{Y}) := \exp(-\mathrm{SDTW}_1(C(\boldsymbol{X}, \boldsymbol{Y})/\gamma)). \tag{9}$$

Using a constructive proof, it was shown that (9) is a positive definite (p.d.) kernel under certain cost functions and in particular with

$$[C(\boldsymbol{X}, \boldsymbol{Y})]_{i,j} = \delta(\boldsymbol{x}_i, \boldsymbol{y}_i) + \log(2 - \exp(\delta(\boldsymbol{x}_i, \boldsymbol{y}_i)), \tag{10}$$

where $\delta(\boldsymbol{x}, \boldsymbol{y}) := \|\boldsymbol{x} - \boldsymbol{y}\|^2/2$. In the one-dimensional case ($d = 1$), we show in Appendix B.4 that

$$[C(\boldsymbol{X}, \boldsymbol{Y})]_{i,j} = \|\boldsymbol{x}_i - \boldsymbol{y}_j\|_1, \tag{11}$$

also has the property that the kernel (9) is p.d. Using these costs, (9) can be used in any kernel method, such as support vector machines. The positive definiteness of (9) using the squared Euclidean cost (1) has to our knowledge not been proved or disproved yet.

3 New differentiable divergences

In this section, we begin by pointing out potential limitations of soft-DTW. We then introduce two new divergences, the soft-DTW divergence and its sharp variant, which aim to correct for these limitations. We study their properties and limit behavior.

Limitations of soft-DTW. Despite recent empirical successes, soft-DTW has some inherent limitations that were not discussed in previous works. The following proposition clarifies these limitations.

Proposition 2. *Limitations of* SDTW_γ

The following holds.

1. *For all* $\boldsymbol{C} \in \mathbb{R}^{m\times n}$, $\gamma \mapsto \mathrm{SDTW}_\gamma(\boldsymbol{C})$ *is non-increasing, concave, and diverges to* $-\infty$ *when* $\gamma \to +\infty$. *In particular, there exists* $\gamma_0 \in [0, \infty)$ *such that* $\mathrm{SDTW}_\gamma(\boldsymbol{C}) \le 0$ *for all* $\gamma \ge \gamma_0$.
2. *For all cost functions* C *satisfying A.2,* $\boldsymbol{X} \in \mathbb{R}^{m\times d}$ *and* $\gamma \in [0, \infty)$, $\mathrm{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{X})) \le 0$.
3. *For the squared Euclidean cost* (1) *and any* $\gamma \in (0, \infty)$, *the minimum of* $\mathrm{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))$ *is not achieved at* $\boldsymbol{X} = \boldsymbol{Y}$.

A proof is given in Appendix B.3. Proposition 2 shows that that there exists values of γ or $\boldsymbol{C}$ for which $\mathrm{SDTW}_\gamma(\boldsymbol{C})$ is negative. Non-negativity is a useful property of divergences and the fact that SDTW_γ does not satisfy it can be a nuisance. More problematic is the fact that $\mathrm{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))$ is not minimized at $\boldsymbol{X} = \boldsymbol{Y}$. This is illustrated in Figure 2. While the denoising effect of soft-DTW can be useful, we would expect a proper differentiable divergence to be zero when the two time series are equal.

Soft-DTW divergences. To address these issues, we propose to use for all $\boldsymbol{X} \in \mathbb{R}^{m\times d}$ and $\boldsymbol{Y} \in \mathbb{R}^{n\times d}$

$$\begin{aligned} D_\gamma^C(\boldsymbol{X}, \boldsymbol{Y}) := {} & \mathrm{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y})) \\ & - \frac{1}{2}\mathrm{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{X})) \\ & - \frac{1}{2}\mathrm{SDTW}_\gamma(C(\boldsymbol{Y}, \boldsymbol{Y})). \end{aligned}$$

Since it is based on soft-DTW, we call it the soft-DTW divergence. Sinkhorn divergences (Ramdas et al., 2017; Genevay et al., 2018; Feydy et al., 2019), which are divergences between probability measures, use similar correction terms.

Sharp divergences. The variational form of SDTW_γ (Proposition 1) implies that it can be decomposed as the sum of a cost term and an entropy term,

$$\mathrm{SDTW}_\gamma(\boldsymbol{C}) = \langle \boldsymbol{E}_\gamma(\boldsymbol{C}), \boldsymbol{C}\rangle - \gamma H(\boldsymbol{p}_\gamma(\boldsymbol{C})). \tag{12}$$

On the other hand, we have

$$\mathrm{DTW}(\boldsymbol{C}) = \langle \boldsymbol{A}^\star(\boldsymbol{C}), \boldsymbol{C}\rangle.$$

Since $\boldsymbol{E}_\gamma(\boldsymbol{C}) \to \boldsymbol{A}^\star(\boldsymbol{C})$ when $\gamma \to 0$, this suggests a new discrepancy measure,

$$\text{SHARP}_\gamma(\boldsymbol{C}) \coloneqq \langle \boldsymbol{E}_\gamma(\boldsymbol{C}), \boldsymbol{C} \rangle. \tag{13}$$

It is the directional derivative of $\text{SDTW}_\gamma(\boldsymbol{C})$ in the direction of $\boldsymbol{C}$, since $\boldsymbol{E}_\gamma(\boldsymbol{C}) = \nabla_{\boldsymbol{C}} \text{SDTW}_\gamma(\boldsymbol{C})$. Inspired by Luise et al. (2018), who studied a similar idea in an optimal transport context,we call it sharp soft-DTW, since it removes the entropic regularization term $-\gamma H(\boldsymbol{p}_\gamma(\boldsymbol{C}))$ from (12). Its gradient is equal to

$$\nabla_{\boldsymbol{C}} \text{SHARP}_\gamma(\boldsymbol{C}) = \boldsymbol{E}_\gamma(\boldsymbol{C}) + \frac{1}{\gamma} \nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(\boldsymbol{C}) \boldsymbol{C} \in \mathbb{R}^{m \times n}, \tag{14}$$

where $\nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(\boldsymbol{C}) \boldsymbol{C}$ is a Hessian-vector product (that can be computed efficiently, as we detail below). The gradient w.r.t. $\boldsymbol{X}$ is obtained by the chain rule, similarly to (8). Although SHARP_γ is trivially non-negative, it suffers from the same issue as SDTW_γ, namely, $\text{SHARP}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))$ is not minimized at $\boldsymbol{X} = \boldsymbol{Y}$. We therefore propose to use instead

$$\begin{aligned} S^C_\gamma(\boldsymbol{X}, \boldsymbol{Y}) \coloneqq\ & \text{SHARP}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y})) \\ & - \frac{1}{2} \text{SHARP}_\gamma(C(\boldsymbol{X}, \boldsymbol{X})) \\ & - \frac{1}{2} \text{SHARP}_\gamma(C(\boldsymbol{Y}, \boldsymbol{Y})). \end{aligned}$$

We call it the sharp soft-DTW divergence.

Non-negativity. By construction, we have $D^C_\gamma(\boldsymbol{X}, \boldsymbol{X}) = 0$ and $S^C_\gamma(\boldsymbol{X}, \boldsymbol{X}) = 0$ for all $\boldsymbol{X} \in \mathbb{R}^{m \times d}$. Moreover, we have the following.

Proposition 3. *Non-negativity*

If C is the cost defined in (10) *with $d \in \mathbb{N}$, or, if C is the absolute value* (11) *with $d = 1$, then $D^C_\gamma(\boldsymbol{X}, \boldsymbol{Y}) \geq 0$ for all $\boldsymbol{X} \in \mathbb{R}^{m \times d}$ and $\boldsymbol{Y} \in \mathbb{R}^{n \times d}$.*

A proof is given in Appendix B.4. This implies that, for the costs (10) and (11), $D^C_\gamma(\boldsymbol{X}, \boldsymbol{Y})$ is minimized at $\boldsymbol{X} = \boldsymbol{Y}$. The proof relies on the fact that the global alignment kernel (9) is positive definite under these costs. Unfortunately, since the positive definiteness of (9) under the squared Euclidean cost (1) has not been proved or disproved, the same proof technique does not apply. Nevertheless, we can prove the following.

Proposition 4. *Stationary point under cost* (1)

If C is the squared Euclidean cost (1)*, then $\boldsymbol{X} = \boldsymbol{Y}$ is a stationary point of $D^C_\gamma(\boldsymbol{X}, \boldsymbol{Y})$ and $S^C_\gamma(\boldsymbol{X}, \boldsymbol{Y})$ w.r.t. $\boldsymbol{X} \in \mathbb{R}^{n \times d}$ for all $\boldsymbol{Y} \in \mathbb{R}^{n \times d}$.*

A proof is given in Appendix B.6. Based on Proposition 4 and ample numerical evidence (cf. Appendix B.5), we conjecture that $D^C_\gamma(\boldsymbol{X}, \boldsymbol{Y})$ and $S^C_\gamma(\boldsymbol{X}, \boldsymbol{Y})$ are also non-negative under the squared Euclidean cost.

Asymptotic behavior. We now study the behavior of our divergences when $\gamma \to 0$ and $\gamma \to \infty$. As we saw, $\boldsymbol{E}_\gamma(\boldsymbol{C})$ is the expected alignment matrix under the Gibbs distribution $\mathbb{P}_\gamma(\boldsymbol{A}; \boldsymbol{C})$. Let A be a random alignment matrix *uniformly* distributed over $\mathcal{A}(m, n)$, i.e., independent of the cost matrix $\boldsymbol{C}$. Replacing $\boldsymbol{E}_\gamma(\boldsymbol{C})$ with $\mathbb{E}[A]$ in (13), we obtain the mean cost, the average of the cost along all possible paths,

$$\begin{aligned} \text{MEAN\_COST}(\boldsymbol{C}) &\coloneqq \langle \mathbb{E}[A], \boldsymbol{C} \rangle \\ &= \frac{1}{|\mathcal{A}(m, n)|} \sum_{\boldsymbol{A} \in \mathcal{A}(m,n)} \langle \boldsymbol{A}, \boldsymbol{C} \rangle. \end{aligned} \tag{15}$$

We also define the mean-cost divergence,

$$\begin{aligned} M^C(\boldsymbol{X}, \boldsymbol{Y}) \coloneqq\ & \text{MEAN\_COST}(C(\boldsymbol{X}, \boldsymbol{Y})) \\ & - \frac{1}{2} \text{MEAN\_COST}(C(\boldsymbol{X}, \boldsymbol{X})) \\ & - \frac{1}{2} \text{MEAN\_COST}(C(\boldsymbol{Y}, \boldsymbol{Y})). \end{aligned}$$

It bears some similarity with energy distances (Baringhaus and Franz, 2004; Székely et al., 2004), with the key difference that the probability distribution is over the alignments, not over the time series.

We now show that our proposed divergences are all intimately related through their asymptotic behavior, and that D^C_γ and S^C_γ share the same limits to the right when $m = n$ but not when $m \neq n$.

Proposition 5. *Limits w.r.t. γ*

For all $\boldsymbol{C} = C(\boldsymbol{X}, \boldsymbol{Y}) \in \mathbb{R}^{m \times n}$, $m = n$:

$$\text{DTW}(\boldsymbol{C}) \xleftarrow[0 \leftarrow \gamma]{} D^C_\gamma(\boldsymbol{X}, \boldsymbol{Y}) \xrightarrow[\gamma \to \infty]{} M^C(\boldsymbol{X}, \boldsymbol{Y}).$$

For all $\boldsymbol{C} = C(\boldsymbol{X}, \boldsymbol{Y}) \in \mathbb{R}^{m \times n}$, $m \neq n$:

$$\text{DTW}(\boldsymbol{C}) \xleftarrow[0 \leftarrow \gamma]{} D^C_\gamma(\boldsymbol{X}, \boldsymbol{Y}) \xrightarrow[\gamma \to \infty]{} \infty.$$

For all $\boldsymbol{C} = C(\boldsymbol{X}, \boldsymbol{Y}) \in \mathbb{R}^{m \times n}$:

$$\text{DTW}(\boldsymbol{C}) \xleftarrow[0 \leftarrow \gamma]{} S^C_\gamma(\boldsymbol{X}, \boldsymbol{Y}) \xrightarrow[\gamma \to \infty]{} M^C(\boldsymbol{X}, \boldsymbol{Y}).$$

Computation. The value, gradient, directional derivative and Hessian product of $\text{SDTW}_\gamma(\boldsymbol{C})$ for $\boldsymbol{C} \in \mathbb{R}^{m \times n}$ can all be computed in $O(mn)$ time (Cuturi and Blondel, 2017; Mensch and Blondel, 2018). Therefore, both $D^C_\gamma(\boldsymbol{X}, \boldsymbol{Y})$ and $S^C_\gamma(\boldsymbol{X}, \boldsymbol{Y})$ take $O(\max\{m, n\}^2)$ time to compute. Sharp divergences take roughly twice more time to compute, as computing a Hessian-vector product requires one more pass through the dynamic programming recursion. The mean alignment and mean cost can also both be computed in $O(mn)$ time. We detail all algorithms in Appendix A.

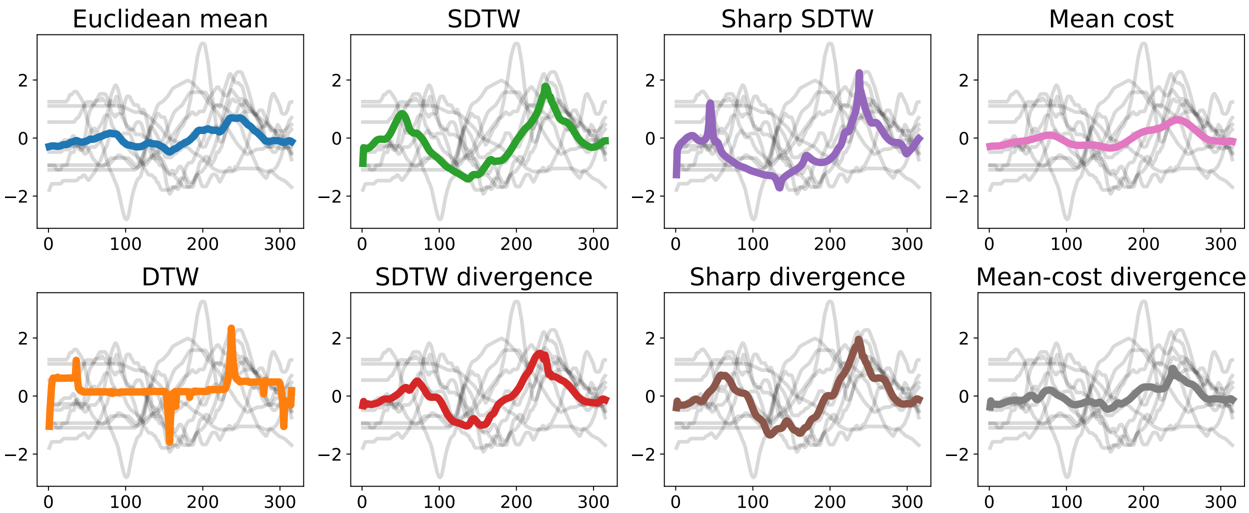

Figure 3: Average of 10 time series $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_{10}$, on the **uWaveGestureLibrary_Y** dataset.

Comparison with Sinkhorn divergences. Since our proposed divergences use similar correction terms as Sinkhorn divergences, we briefly review them and discuss their differences. Given two input probability measures $\boldsymbol{\alpha} \in \triangle^m$ and $\boldsymbol{\beta} \in \triangle^n$, entropy-regularized optimal transport is now commonly defined as

$$\mathrm{OT}_\gamma(\boldsymbol{\alpha}, \boldsymbol{\beta}) \coloneqq \min_{\boldsymbol{T} \in \mathcal{U}(\boldsymbol{\alpha}, \boldsymbol{\beta})} \langle \boldsymbol{T}, \boldsymbol{C} \rangle + \gamma \mathrm{KL}(\boldsymbol{T} || \boldsymbol{\alpha} \otimes \boldsymbol{\beta}), \quad (16)$$

where KL is the Kullback-Leibler divergence and $\mathcal{U}(\boldsymbol{\alpha}, \boldsymbol{\beta})$ is the so-called transportation polytope (Peyré et al., 2019). To address the entropic bias of OT_γ, Sinkhorn divergences include correction terms, i.e., they are defined as $(\boldsymbol{\alpha}, \boldsymbol{\beta}) \mapsto \mathrm{OT}_\gamma(\boldsymbol{\alpha}, \boldsymbol{\beta}) - \frac{1}{2}\mathrm{OT}_\gamma(\boldsymbol{\alpha}, \boldsymbol{\alpha}) - \frac{1}{2}\mathrm{OT}_\gamma(\boldsymbol{\beta}, \boldsymbol{\beta})$. There are however two important differences between OT_γ and $\mathrm{SDTW}_\gamma(C(\cdot, \cdot))$. First, the former is convex in its inputs (separately) while the latter is not. This means that the proof technique for non-negativity of Sinkhorn divergences (Feydy et al., 2019) does not apply to the soft-DTW divergence. Second, the entropic regularization in SDTW_γ is on the probability distribution (Proposition 1), not on the soft alignment, as is the case for the transportation map $\boldsymbol{T}$ in (16). Contrary to Sinkhorn divergences, the soft-DTW and sharp divergences are non-convex in their inputs. For time-series averaging, an initialization scheme that works well in practice is to use the SDTW_γ solution as initialization, itself initialized from the Euclidean mean.

4 Experimental results

Throughout this section, we use the UCR (University of California, Riverside) time series classification archive (Chen et al., 2015). We use a subset containing 84 datasets encompassing a wide variety of fields (astronomy, geology, medical imaging) and lengths. Datasets include class information (up to 60 classes) for each time series and are split into train and test sets. Due to the large number of datasets in the UCR archive, we choose to report only a summary of our results in the main manuscript. Detailed results are included in the appendix for interested readers. In all experiments, we use the squared Euclidean cost (1). Our Python source code is available on github.

4.1 Time series averaging

Experimental setup. To investigate the effect of our divergences on time series averaging, we replace SDTW_γ in objective (7) with our divergences. For this task, we focus on a visual comparison and refrain from reporting quantitative results, since the choice of evaluation metric necessarily favors one divergence over others. For each dataset, we pick 10 time series $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_{10}$ randomly. Since the time series all have the same length, we use uniform weights $w_1 = \cdots = w_k = 1$. To approximately minimize the objective function, we use 200 iterations of L-BFGS (Liu and Nocedal, 1989). Because the objective is non-convex in $\boldsymbol{X}$, initialization is important. For DTW, SDTW_γ, SHARP_γ and MEAN_COST, we use the Euclidean mean as initialization and set $\gamma = 1$. For D_γ^C, S_γ^C and M^C, we use as initialization the solution of their "biased couterpart", i.e., SDTW_γ, SHARP_γ, MEAN_COST, respectively, and we set $\gamma = 10$.

Results. We show the time series averages obtained on the *uWaveGestureLibrary_Y* dataset in Figure 3. With DTW, the obtained average does not match well the time series, confirming the conclusion of Cuturi

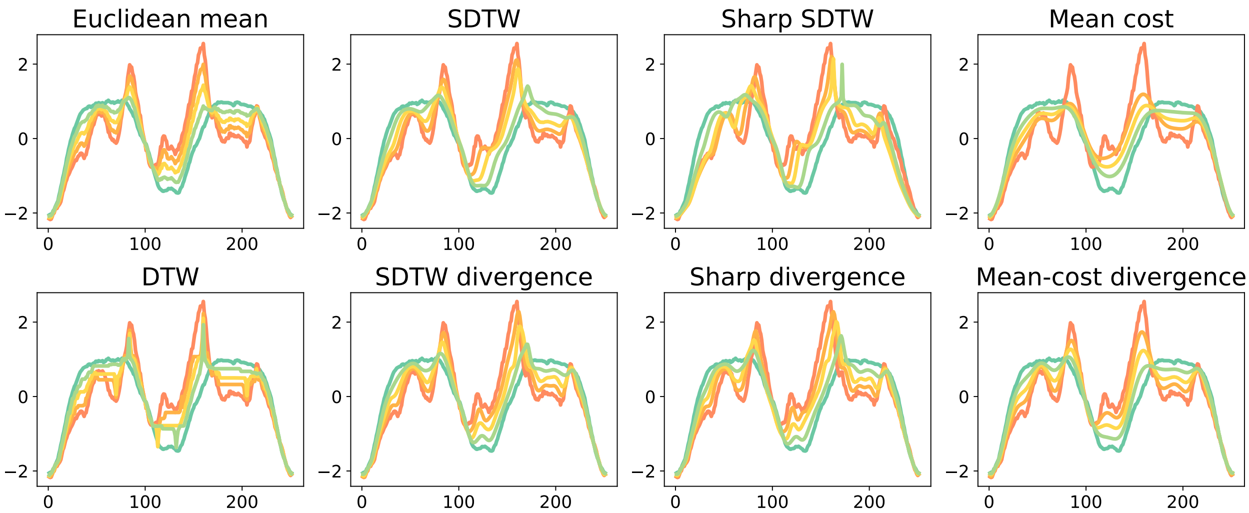

Figure 4: Interpolation between two time series $\boldsymbol{Y}_1$ (red) and $\boldsymbol{Y}_2$ (dark green), from the **ArrowHead** dataset.

and Blondel (2017). This is because the objective is both highly non-convex and non-smooth, rendering optimization difficult, despite the use of Euclidean mean as initialization. On the other hand, the averages obtained by other divergences appear to match the time series much better, thanks to the smoothness of their objective function. We observe that D_γ^C (soft-DTW divergence), S_γ^C (sharp divergence) and M^C (mean-cost divergence) produce different results from their biased counterpart, SDTW_γ (soft-DTW), SHARP_γ (sharp soft-DTW) and MEAN_COST (mean cost), respectively. This is to be expected, since the variable $\boldsymbol{X}$ with respect to which we minimize is involved in the correcting term using $C(\boldsymbol{X}, \boldsymbol{X})$. The averages obtained with SHARP_γ and S_γ^C tend to include sharper peaks, a trend confirmed on other datasets as well. More average examples are included in the appendix.

4.2 Time series interpolation

Experimental setup. As a simple variation of time series averaging, we now consider time series interpolation. We pick two times series $\boldsymbol{Y}_1$ and $\boldsymbol{Y}_2$ and set the weights in objective (7) to $w_1 = \pi$ and $w_2 = 1 - \pi$, for $\pi \in \{0.25, 0.5, 0.75\}$, i.e., we seek an interpolation of the two time series. We again minimize the objective approximately using L-BFGS, with the same initialization scheme and the same γ as before.

Results. Results on the *ArrowHead* dataset are shown in Figure 4. We observe similar trends as for time series averaging. The interpolations obtained by DTW include artifacts that do not represent well the data. Our divergences obtain slightly more visually pleasing results than their biased counterparts. More examples are included in the appendix. The interpolation obtained by the sharp soft-DTW includes a peak (light green) which is slightly off, but this is not the case of the sharp divergence.

4.3 Time series classification

Experimental setup. To quantitatively compare our proposed divergences, we now consider time series classification tasks. To better isolate the effect of the divergence itself, we choose two simple classifiers: nearest neighbor and nearest centroid. To predict the class of a time series, the well-known nearest neighbor classifier assigns the class of the nearest time series in the training set, according to the chosen divergence. Note that this does not require differentiability of the divergence. The lesser known nearest centroid classifier (Hastie et al., 2001) first computes the centroid (average) of each class in the training set. We compute the centroid by minimizing (7) for each class, according to the chosen divergence. To predict the class of a time series, we then assign the class of the nearest centroid, according to the same divergence. Although very simple, this method is known to be competitive with the nearest neighbor classifier, while requiring much lower computational cost at prediction time (Petitjean et al., 2014).

For all datasets in the UCR archive, we use the predefined test set. For divergences including a γ parameter, we select γ by cross-validation. More precisely, we train on $2/3$ of the training set and evaluate the goodness of a γ value on the held-out $1/3$. We repeat this procedure 5 times, each with a different random split, in order to get a better estimate of the goodness of γ. We do so for $\gamma \in \{10^{-4}, 10^{-3}, \dots, 10^4\}$ and select

Table 2: **Nearest neighbor results.** Each number indicates the percentage of datasets in the UCR archive for which using A in the nearest neighbor classifier is within 99% or better than using B .

A ($\downarrow$) vs. B ($\rightarrow$)	Euc.	DTW	SDTW	SDTW div	Sharp	Sharp div	Mean cost	Mean-cost div
Euclidean	-	41.67	34.62	22.37	29.49	27.63	95.29	71.43
DTW	71.43	-	42.31	39.47	50.00	39.47	89.29	79.76
SDTW	75.64	82.05	-	52.63	73.08	55.26	97.44	80.77
SDTW div	93.42	93.42	86.84	-	84.21	82.67	97.37	96.05
Sharp	83.33	84.62	76.92	53.95	-	52.63	98.72	87.18
Sharp div	94.74	86.84	77.63	66.67	81.58	-	98.68	96.05
Mean cost	9.41	13.10	8.97	5.26	5.13	6.58	-	44.05
Mean-cost div	42.86	32.14	25.64	19.74	21.79	18.42	98.81	-

Table 3: **Nearest centroid results.** Each number indicates the percentage of datasets in the UCR archive for which using A in the nearest neighbor classifier is within 99% or better than using B .

A ($\downarrow$) vs. B ($\rightarrow$)	Euc.	DTW	SDTW	SDTW div	Sharp	Sharp div	Mean cost	Mean-cost div
Euclidean	-	44.71	27.06	28.57	30.95	32.50	77.65	78.82
DTW	63.53	-	36.47	36.90	41.67	37.50	83.53	80.00
SDTW	82.35	85.88	-	55.95	77.38	62.50	94.12	94.12
SDTW div	82.14	83.33	82.14	-	78.57	70.00	91.67	94.05
Sharp	79.76	78.57	54.76	48.81	-	55.00	91.67	91.67
Sharp div	82.50	82.50	70.00	63.75	78.75	-	92.50	93.75
Mean cost	37.65	22.35	11.76	11.90	15.48	11.25	-	77.65
Mean-cost div	41.18	23.53	14.12	14.29	17.86	15.00	90.59	-

the best one. Finally, we retrain on the entire training set using that γ value.

Results. Due to the large number of datasets in the UCR archive, we only show a summary of the results in Table 2 and Table 3. Detailed results are in Appendix C. We observe consistent trends for both the nearest neighbor and the nearest centroid classifiers. The mean-cost divergence appears to perform poorly, even worse than the squared Euclidean distance and DTW. This shows that considering all possible alignments uniformly does not lead to a good divergence measure. On the other hand, our proposed divergences, the soft-DTW divergence and the sharp divergence, outperform on the majority of the datasets the Euclidean distance, DTW, soft-DTW, and sharp soft-DTW. Furthermore, each proposed divergence (i.e., with correction term) clearly outperforms its biased counterpart (i.e., without correction term). This shows that proper divergences, which are minimized when the two time series are equal, indeed translate to higher classification accuracy in practice. Overall, the soft-DTW divergence works better than the sharp divergence.

5 Conclusion

Due to entropic bias, soft-DTW can be negative and is not minimized when the two time series are equal. To address these issues, we proposed the soft-DTW divergence and its sharp variant. By studying their limit behavior when the regularization parameter γ goes to infinity, we also obtained a new mean-cost divergence, which is of independent interest. Experiments on 84 time series classification datasets established that the soft-DTW divergence performs the best among all discrepancies and divergences considered.

References

Cyril Banderier and Sylviane Schwer. Why Delannoy numbers? *Journal of statistical planning and inference*, 135(1):40–54, 2005.

Ludwig Baringhaus and Carsten Franz. On a new multivariate two-sample test. *Journal of multivariate analysis*, 88(1):190–206, 2004.

Chien-Yi Chang, De-An Huang, Yanan Sui, Li Fei-Fei, and Juan Carlos Niebles. D3tw: Discriminative differentiable dynamic time warping for weakly supervised action alignment and segmentation. In *Proc. of CVPR*, pages 3546–3555, 2019.

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 and Mathieu Blondel. Soft-DTW: A differentiable loss function for time-series. In *Proc. of ICML*, 2017.

Marco Cuturi, Jean-Philippe Vert, Oystein Birkenes, and Tomoko Matsui. A kernel for time series based on global alignments. In *Proc. of ICASSP*, volume 2, pages II–413. IEEE, 2007.

Jeff Donahue, Sander Dieleman, Mikołaj Bińkowski, Erich Elsen, and Karen Simonyan. End-to-end adversarial text-to-speech. *arXiv preprint arXiv:2006.03575*, 2020.

Jean Feydy, Thibault Séjourné, François-Xavier Vialard, Shun-ichi Amari, Alain Trouvé, and Gabriel Peyré. Interpolating between optimal transport and mmd using sinkhorn divergences. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2681–2690, 2019.

Maurice Fréchet. Les éléments aléatoires de nature quelconque dans un espace distancié. In *Annales de l'institut Henri Poincaré*, volume 10, pages 215–310. Presses universitaires de France, 1948.

Aude Genevay, Gabriel Peyré, and Marco Cuturi. Learning generative models with sinkhorn divergences. In *International Conference on Artificial Intelligence and Statistics*, pages 1608–1617, 2018.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proc. of ICML*, pages 369–376, 2006.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer New York Inc., 2001.

Hicham Janati, Marco Cuturi, and Alexandre Gramfort. Spatio-temporal alignments: Optimal transport through space and time. In *Proc. of AISTATS*, pages 1695–1704. PMLR, 2020.

Dong C Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1):503–528, 1989.

Giulia Luise, Alessandro Rudi, Massimiliano Pontil, and Carlo Ciliberto. Differential properties of sinkhorn approximation for learning with wasserstein distance. In *Proc. of NeurIPS*, pages 5859–5870, 2018.

Andrew McCallum, Kedar Bellare, and Fernando Pereira. A conditional random field for discriminatively-trained finite-state string edit distance. *arXiv preprint arXiv:1207.1406*, 2012.

Arthur Mensch and Mathieu Blondel. Differentiable dynamic programming for structured prediction and attention. In *Proc. of ICML*, 2018.

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.

François Petitjean, Germain Forestier, Geoffrey I Webb, Ann E Nicholson, Yanping Chen, and Eamonn Keogh. Dynamic time warping averaging of time series allows faster and more accurate classification. In *ICDM*, pages 470–479. IEEE, 2014.

Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11 (5-6):355–607, 2019.

Aaditya Ramdas, Nicolás García Trillos, and Marco Cuturi. On wasserstein two-sample testing and related families of nonparametric tests. *Entropy*, 19 (2):47, 2017.

Hiroto Saigo, Jean-Philippe Vert, and Tatsuya Akutsu. Optimizing amino acid substitution matrices with a local alignment kernel. *BMC bioinformatics*, 7(1):246, 2006.

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.

Alexis Sardá-Espinosa. Comparing time-series clustering algorithms in r using the dtwclust package. *R package vignette*, 12:41, 2017.

Robert A Sulanke. Objects counted by the central delannoy numbers. *J. Integer Seq*, 6(1), 2003.

Gábor J Székely, Maria L Rizzo, et al. Testing for equal distributions in high dimension. *InterStat*, 5 (16.10):1249–1272, 2004.

Romain Tavenard, Johann Faouzi, Gilles Vandewiele, Felix Divo, Guillaume Androz, Chester Holtz, Marie Payne, Roman Yurchak, Marc Rußwurm, Kushal Kolar, et al. Tslearn, a machine learning toolkit for time series data. *JMLR*, 21(118):1–6, 2020.

Martin J Wainwright and Michael I Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.

Appendix

A Algorithms

We begin by recalling the algorithms derived by Mensch and Blondel (2018) for computing the value, gradient, directional derivative and Hessian product of $\text{SDTW}_\gamma(\boldsymbol{C})$ in $O(mn)$ time and space. The lines in light gray indicate values that must be set in order to handle edge cases. The Gibbs distribution (5) is equivalent to a random walk (finite Markov chain) on the directed acyclic graph pictured in Figure 1. The matrix $\boldsymbol{P} \in (0, 1]^{m \times n \times 3}$ computed in Algorithm 1 contains the transition probabilities for this random walk. Although modern automatic differentiation frameworks can in principle derive Algorithms 2–4 automatically from the first output of Algorithm 1, these frameworks are typically not well suited for tight loops operating over triplets of values, such as the ones in Algorithm 1. We argue that a manual implementation of the algorithms below is more efficient on CPU. The algorithms also play an important role to compute $\text{SHARP}_\gamma(\boldsymbol{C})$ and $\text{MEAN\_COST}(\boldsymbol{C})$, as we describe later.

Algorithm 1 Soft-DTW value and transition probabilities

Input: Cost matrix $\boldsymbol{C} \in \mathbb{R}^{m \times n}$, $\gamma \geq 0$
$\boldsymbol{V}_{:,0} \leftarrow \infty$, $\boldsymbol{V}_{0,:} \leftarrow \infty$, $V_{0,0} \leftarrow 0$
for $i \in [1, \ldots, m]$, $j \in [1, \ldots, n]$ **do**
  $V_{i,j} \leftarrow C_{i,j} + \min_\gamma(V_{i,j-1}, V_{i-1,j-1}, V_{i-1,j}) \in \mathbb{R}$
  $\boldsymbol{P}_{i,j} \leftarrow \nabla \min_\gamma(V_{i,j-1}, V_{i-1,j-1}, V_{i-1,j}) \in \triangle^3$
Return: $\text{SDTW}_\gamma(\boldsymbol{C}) = V_{m,n} \in \mathbb{R}$, $\boldsymbol{P} \in (0, 1]^{m \times n \times 3}$

Algorithm 2 Soft-DTW gradient (expected alignment)

Input: $\boldsymbol{P} \in (0, 1]^{m \times n \times 3}$ (Algorithm 1 or Algorithm 5)
$E_{m+1,:} \leftarrow 0$, $E_{:,n+1} \leftarrow 0$, $E_{m+1,n+1} \leftarrow 1$, $\boldsymbol{P}_{m+1,:} \leftarrow (0, 0, 0)$, $\boldsymbol{P}_{:,n+1} \leftarrow (0, 0, 0)$, $\boldsymbol{P}_{m+1,n+1} \leftarrow (0, 1, 0)$
for $j \in [n, \ldots, 1]$, $i \in [m, \ldots, 1]$ **do**
  $E_{i,j} \leftarrow P_{i,j+1,1} \cdot E_{i,j+1} + P_{i+1,j+1,2} \cdot E_{i+1,j+1} + P_{i+1,j,3} \cdot E_{i+1,j}$
Return: $\nabla_{\boldsymbol{C}} \text{SDTW}_\gamma(\boldsymbol{C}) = \boldsymbol{E} \in (0, 1]^{m \times n}$

Algorithm 3 Soft-DTW directional derivative in the direction of $\boldsymbol{Z}$ and intermediate computations

Input: $\boldsymbol{P} \in (0, 1]^{m \times n \times 3}$ (Algorithm 1 or Algorithm 5), $\boldsymbol{Z} \in \mathbb{R}^{m \times n}$
$\dot{\boldsymbol{V}}_{:,0} \leftarrow 0$, $\dot{\boldsymbol{V}}_{0,:} \leftarrow 0$
for $i \in [1, \ldots, m]$, $j \in [1, \ldots, n]$ **do**
  $\dot{V}_{i,j} \leftarrow Z_{i,j} + P_{i,j,1} \cdot \dot{V}_{i,j-1} + P_{i,j,2} \cdot \dot{V}_{i-1,j-1} + P_{i,j,3} \cdot \dot{V}_{i-1,j}$
Return: $\langle \nabla_{\boldsymbol{C}} \text{SDTW}_\gamma(\boldsymbol{C}), \boldsymbol{Z} \rangle = \dot{V}_{m,n} \in \mathbb{R}$, $\dot{\boldsymbol{V}} \in \mathbb{R}^{m \times n}$

Algorithm 4 Soft-DTW Hessian product

Input: $\boldsymbol{P} \in (0, 1]^{m \times n \times 3}$ (Algorithm 1), $\dot{\boldsymbol{V}} \in \mathbb{R}^{m \times n}$ (Algorithm 3), $\boldsymbol{Z} \in \mathbb{R}^{m \times n}$
$\dot{\boldsymbol{E}}_{m+1,:} \leftarrow 0$, $\dot{\boldsymbol{E}}_{:,n+1} \leftarrow 0$ $\dot{\boldsymbol{P}}_{m+1,:} \leftarrow (0, 0, 0)$ $\dot{\boldsymbol{P}}_{:,n+1} \leftarrow (0, 0, 0)$
for $j \in [n, \ldots, 1]$, $i \in [m, \ldots, 1]$ **do**
  $s \leftarrow P_{i,j,1} \cdot \dot{V}_{i,j-1} + P_{i,j,2} \cdot \dot{V}_{i-1,j-1} + P_{i,j,3} \cdot \dot{V}_{i-1,j}$
  $\dot{P}_{i,j,1} \leftarrow P_{i,j,1} \cdot (s - \dot{V}_{i,j-1})$, $\dot{P}_{i,j,2} \leftarrow P_{i,j,2} \cdot (s - \dot{V}_{i-1,j-1})$, $\dot{P}_{i,j,3} \leftarrow P_{i,j,3} \cdot (s - \dot{V}_{i-1,j})$
  $\dot{E}_{i,j} \leftarrow \dot{P}_{i,j+1,1} \cdot E_{i,j+1} + P_{i,j+1,1} \cdot \dot{E}_{i,j+1} + \dot{P}_{i+1,j+1,2} \cdot E_{i+1,j+1} + P_{i+1,j+1,2} \cdot \dot{E}_{i+1,j+1} +$
    $\dot{P}_{i+1,j,3} \cdot E_{i+1,j} + P_{i+1,j,3} \cdot \dot{E}_{i+1,j}$
Return: $\nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(\boldsymbol{C}) \boldsymbol{Z} = \dot{\boldsymbol{E}} \in \mathbb{R}^{m \times n}$

Since $\text{SHARP}_\gamma(\boldsymbol{C})$ is the directional derivative of $\text{SDTW}_\gamma(\boldsymbol{C})$ in the direction of $\boldsymbol{C}$, we can compute it using Algorithm 3 with $\boldsymbol{P}$ coming from Algorithm 1 and $\boldsymbol{Z} = \boldsymbol{C}$. The gradient of $\text{SHARP}_\gamma(\boldsymbol{C})$ w.r.t. $\boldsymbol{C}$, see (14), involves the product with the Hessian of $\text{SDTW}_\gamma(\boldsymbol{C})$ and can be computed using Algorithm 4, again with $\boldsymbol{Z} = \boldsymbol{C}$.

We continue with an algorithm to compute $\text{MEAN\_COST}(\boldsymbol{C})$. This algorithm is new to our knowledge. We start by a known recursion for computing the cardinality $|\mathcal{A}(m,n)|$ (Sulanke, 2003). The key modification we make is to build a transition probability matrix $\boldsymbol{P}$ along the way, mirroring Algorithm 1.

Algorithm 5 Cardinality $|\mathcal{A}(m,n)|$ and transition probabilities

Input: Cost matrix $\boldsymbol{C} \in \mathbb{R}^{m \times n}$

$\boldsymbol{V}_{:,0} \leftarrow 0$, $\boldsymbol{V}_{0,:} \leftarrow 0$, $V_{0,0} \leftarrow 1$

for $i \in [1, \dots, m]$, $j \in [1, \dots, n]$ **do**

$\quad V_{i,j} \leftarrow V_{i,j-1} + V_{i-1,j-1} + V_{i-1,j}$

$\quad P_{i,j,1} \leftarrow V_{i,j-1}/V_{i,j}$, $P_{i,j,2} \leftarrow V_{i-1,j-1}/V_{i,j}$, $P_{i,j,3} \leftarrow V_{i-1,j}/V_{i,j}$.

Return: $|\mathcal{A}(m,n)| = V_{m,n} \in \mathbb{N}$, $\boldsymbol{P} \in (0,1]^{m \times n \times 3}$

This modification allows us to reuse previous algorithms. Indeed, we can now compute $\text{MEAN\_COST}(\boldsymbol{C})$ by using Algorithm 3 with the above $\boldsymbol{P}$ and $\boldsymbol{Z} = \boldsymbol{C}$ as inputs. Alternatively, we can use Algorithm 2 to compute $\boldsymbol{E} = \mathbb{E}[A]$, where A is uniformly distributed over $\mathcal{A}(m,n)$, to then obtain $\text{MEAN\_COST}(\boldsymbol{C}) = \langle \boldsymbol{E}, \boldsymbol{C} \rangle$. Note that $\boldsymbol{E}$ is also the gradient of $\text{MEAN\_COST}(\boldsymbol{C})$ w.r.t. $\boldsymbol{C}$.

To summarize, we have described algorithms for computing $\text{SDTW}_\gamma(\boldsymbol{C})$, $\text{SHARP}_\gamma(\boldsymbol{C})$ and $\text{MEAN\_COST}(\boldsymbol{C})$ in $O(mn)$ time and space. These, in turn, can be used to compute $D_\gamma^C(\boldsymbol{X}, \boldsymbol{Y})$ (soft-DTW divergence), $S_\gamma^C(\boldsymbol{X}, \boldsymbol{Y})$ (sharp divergence) and $M^C(\boldsymbol{X}, \boldsymbol{Y})$ (mean-cost divergence).

B Proofs

B.1 Sensitivity analysis w.r.t. γ

Proposition 6. *Derivatives w.r.t.* γ

We have for all $\boldsymbol{C} \in \mathbb{R}^{m \times n}$

$$\frac{\partial \text{SDTW}_\gamma(\boldsymbol{C})}{\partial \gamma} = -H(\boldsymbol{p}_\gamma(\boldsymbol{C})) \leq 0 \quad \textit{and} \quad \frac{\partial^2 \text{SDTW}_\gamma(\boldsymbol{C})}{\partial \gamma^2} = \frac{1}{\gamma^3} \langle \boldsymbol{C}, \nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(\boldsymbol{C}) \boldsymbol{C} \rangle \leq 0.$$

Proof. Recalling that $\text{SDTW}_\gamma(\boldsymbol{C}) = \gamma \text{SDTW}_1(\boldsymbol{C}/\gamma)$, we have

$$\begin{aligned} \frac{\partial \text{SDTW}_\gamma(\boldsymbol{C})}{\partial \gamma} &= \text{SDTW}_1(\boldsymbol{C}/\gamma) - \frac{1}{\gamma} \langle \boldsymbol{E}_1(\boldsymbol{C}/\gamma), \boldsymbol{C} \rangle \\ &= \frac{1}{\gamma} \text{SDTW}_\gamma(\boldsymbol{C}) - \frac{1}{\gamma} \langle \boldsymbol{E}_\gamma(\boldsymbol{C}), \boldsymbol{C} \rangle \\ &= -H(\boldsymbol{p}_\gamma(\boldsymbol{C})) \leq 0, \end{aligned}$$

where we used (12) and the fact that H is non-negative over the simplex. Similarly, we have

$$\begin{aligned} \frac{\partial^2 \text{SDTW}_\gamma(\boldsymbol{C})}{\partial \gamma^2} &= -\frac{1}{\gamma^2} \langle \boldsymbol{E}_1(\boldsymbol{C}/\gamma), \boldsymbol{C} \rangle + \frac{1}{\gamma^2} \langle \boldsymbol{E}_1(\boldsymbol{C}/\gamma), \boldsymbol{C} \rangle + \frac{1}{\gamma^3} \langle \boldsymbol{C}, \nabla^2_{\boldsymbol{C}} \text{SDTW}_1(\boldsymbol{C}/\gamma) \boldsymbol{C} \rangle \\ &= \frac{1}{\gamma^3} \langle \boldsymbol{C}, \nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(\boldsymbol{C}) \boldsymbol{C} \rangle \leq 0, \end{aligned}$$

where we used the concavity of SDTW_γ w.r.t. $\boldsymbol{C}$. $\square$

B.2 Product with the Jacobian of the squared Euclidean cost

For the squared Euclidean cost (1), we have

$$C(\boldsymbol{X},\boldsymbol{Y}) = \frac{1}{2}\operatorname{diag}(\boldsymbol{X}\boldsymbol{X}^\top)\mathbf{1}_n^\top + \frac{1}{2}\mathbf{1}_m \operatorname{diag}(\boldsymbol{Y}\boldsymbol{Y}^\top)^\top - \boldsymbol{X}\boldsymbol{Y}^\top \in \mathbb{R}^{m\times n}$$

where $\operatorname{diag}(\boldsymbol{M})$ is a vector containing the diagonal elements of $\boldsymbol{M}$. With some abuse of notation, we denote

$$C(\boldsymbol{X}) \coloneqq C(\boldsymbol{X},\boldsymbol{X}) \in \mathbb{R}^{m\times m}.$$

Product with the Jacobian transpose ("VJP"). For fixed $\boldsymbol{Y} \in \mathbb{R}^{n\times d}$, we have for all $\boldsymbol{E} \in \mathbb{R}^{m\times n}$

$$[(J_{\boldsymbol{X}} C(\boldsymbol{X},\boldsymbol{Y}))^\top \boldsymbol{E}]_{i,k} = \sum_{j=1}^n e_{i,j}(x_{i,k} - y_{j,k}) \quad i \in [m], k \in [d] \tag{17}$$

or equivalently

$$(J_{\boldsymbol{X}} C(\boldsymbol{X},\boldsymbol{Y}))^\top \boldsymbol{E} = \boldsymbol{X} \circ (\boldsymbol{E}\mathbf{1}_{n\times d}) - \boldsymbol{E}\boldsymbol{Y} \in \mathbb{R}^{m\times d},$$

where $\circ$ denotes the Hadamard product. Similarly, we have for all $\boldsymbol{E} \in \mathbb{R}^{m\times m}$

$$[(J_{\boldsymbol{X}} C(\boldsymbol{X}))^\top \boldsymbol{E}]_{i,k} = \sum_{j=1}^n (e_{i,j} + e_{j,i})(x_{i,k} - x_{j,k}) \quad i \in [m], k \in [d] \tag{18}$$

or equivalently

$$(J_{\boldsymbol{X}} C(\boldsymbol{X}))^\top \boldsymbol{E} = \boldsymbol{X} \circ ((\boldsymbol{E} + \boldsymbol{E}^\top)\mathbf{1}_{m\times d}) - (\boldsymbol{E} + \boldsymbol{E}^\top)\boldsymbol{X} \in \mathbb{R}^{m\times d}.$$

If $\boldsymbol{E}$ is symmetric, we therefore have at $\boldsymbol{X} = \boldsymbol{Y}$

$$(J_{\boldsymbol{X}} C(\boldsymbol{X}))^\top \boldsymbol{E} = 2(J_{\boldsymbol{X}} C(\boldsymbol{X},\boldsymbol{Y}))^\top \boldsymbol{E}. \tag{19}$$

Product with the Jacobian ("JVP"). For fixed $\boldsymbol{Y}$, we have for all $\boldsymbol{Z} \in \mathbb{R}^{m\times d}$

$$[J_{\boldsymbol{X}} C(\boldsymbol{X},\boldsymbol{Y})\boldsymbol{Z}]_{i,j} = \sum_{k=1}^d z_{i,k}(x_{i,k} - y_{j,k}) \quad i \in [m], j \in [n]$$

or equivalently

$$J_{\boldsymbol{X}} C(\boldsymbol{X},\boldsymbol{Y})\boldsymbol{Z} = \operatorname{diag}(\boldsymbol{X}\boldsymbol{Z}^\top)\mathbf{1}_n^\top - \boldsymbol{Z}\boldsymbol{Y}^\top \in \mathbb{R}^{m\times n}.$$

Similarly, we have for all $\boldsymbol{Z} \in \mathbb{R}^{m\times d}$

$$[J_{\boldsymbol{X}} C(\boldsymbol{X})\boldsymbol{Z}]_{i,j} = \sum_{k=1}^d (z_{i,k} - z_{j,k})(x_{i,k} - x_{j,k}) \quad i \in [m], j \in [m]$$

or equivalently

$$J_{\boldsymbol{X}} C(\boldsymbol{X})\boldsymbol{Z} = \operatorname{diag}(\boldsymbol{X}\boldsymbol{Z}^\top)\mathbf{1}_m^\top + \mathbf{1}_m \operatorname{diag}(\boldsymbol{Z}\boldsymbol{X}^\top)^\top - \boldsymbol{Z}\boldsymbol{X}^\top - \boldsymbol{X}\boldsymbol{Z}^\top \in \mathbb{R}^{m\times m}.$$

We therefore have at $\boldsymbol{X} = \boldsymbol{Y}$

$$J_{\boldsymbol{X}} C(\boldsymbol{X})\boldsymbol{Z} = J_{\boldsymbol{X}} C(\boldsymbol{X},\boldsymbol{Y})\boldsymbol{Z} + (J_{\boldsymbol{X}} C(\boldsymbol{X},\boldsymbol{Y})\boldsymbol{Z})^\top, \tag{20}$$

i.e., $J_{\boldsymbol{X}} C(\boldsymbol{X})\boldsymbol{Z}$ is the symmetrization of $J_{\boldsymbol{X}} C(\boldsymbol{X},\boldsymbol{Y})\boldsymbol{Z}$.

B.3 Proof of Proposition 2 (limitations of sdtw_γ)

We assume assumptions A.1-A.3 hold.

1. The fact that $\text{SDTW}_\gamma(\boldsymbol{C}) \xrightarrow[\gamma\to\infty]{} -\infty$ follows from (12). From Proposition 6, for all $\boldsymbol{C} \in \mathbb{R}^{m\times n}$, $\text{SDTW}_\gamma(\boldsymbol{C})$ is concave w.r.t. γ and non-increasing on $[0,\infty)$. Since $\text{DTW}(\boldsymbol{C}) \geq 0$ and $\text{SDTW}_\gamma(\boldsymbol{C}) \xrightarrow[\gamma\to\infty]{} -\infty$, from the intermediate value theorem, there exists $\gamma_0 \in [0,\infty)$ such that $\text{SDTW}_\gamma(\boldsymbol{C}) \leq 0$ for all $\gamma \geq \gamma_0$.

2. If the cost C satisfies assumption A.2, then for any $\boldsymbol{X} \in \mathbb{R}^{m\times d}$ the diagonal alignment $I_m \in \mathcal{A}(m,m)$ satisfies $\langle I_m, C(\boldsymbol{X},\boldsymbol{X})\rangle = \sum_{i=1}^m [C(\boldsymbol{X},\boldsymbol{X})]_{i,i} = 0$. Therefore, $\text{DTW}(C(\boldsymbol{X},\boldsymbol{X})) = 0$. Using the fact that $\gamma \mapsto \text{SDTW}_\gamma(\boldsymbol{C})$ is non-increasing on $\gamma \in [0,\infty)$, we obtain $\text{SDTW}_\gamma(C(\boldsymbol{X},\boldsymbol{X}) \leq 0$ for all $\gamma \in [0,\infty)$.

3. If the minimum of $\text{SDTW}_\gamma(C(\boldsymbol{X},\boldsymbol{Y}))$ is achieved at $\boldsymbol{X} = \boldsymbol{Y}$, then the gradient (8) should be equal to $\mathbf{0}_{m\times d}$ or put differently, $\boldsymbol{E}_\gamma(C(\boldsymbol{X},\boldsymbol{Y}))$ should be in the nullspace of $(J_{\boldsymbol{X}} C(\boldsymbol{X},\boldsymbol{Y}))^\top$. For the squared Euclidean cost, from (17), a matrix $\boldsymbol{E} \in \mathbb{R}^{m\times n}$ is in the nullspace of $(J_{\boldsymbol{X}} C(\boldsymbol{X},\boldsymbol{Y}))^\top$ if for all $i \in [m], k \in [d]$

$$\sum_{j=1}^n e_{i,j}(x_{i,k} - y_{j,k}) = 0.$$

Since $e_{i,j} > 0$, this is equivalent to

$$x_{i,k} = \frac{\sum_{j=1}^n e_{i,j} y_{j,k}}{\sum_{j=1}^n e_{i,j}} \neq y_{i,k}.$$

B.4 Proof of Proposition 3 (non-negativity)

Non-negativity with the log-augmented squared Euclidean cost. The fact that (9) is positive definite (p.d.) under the cost (10) was proved by Cuturi et al. (2007). More precisely, in their Theorem 1, the authors show that the kernel $K_\gamma^C(\boldsymbol{X},\boldsymbol{Y}) = \exp(-\text{SDTW}_1(\boldsymbol{X},\boldsymbol{Y}))$ is positive definite if the kernel $k(\boldsymbol{x},\boldsymbol{y}) \coloneqq \exp(-c(\boldsymbol{x},\boldsymbol{y}))$ is such that $\tilde{k} \coloneqq \frac{k}{1+k}$ is positive definite. In particular, setting

$$k(\boldsymbol{x},\boldsymbol{y}) = \frac{\frac{1}{2}\exp(-||\boldsymbol{x}-\boldsymbol{y}||^2/2)}{1 - \frac{1}{2}\exp(-||\boldsymbol{x}-\boldsymbol{y}||^2/2)} = \frac{\exp(-||\boldsymbol{x}-\boldsymbol{y}||^2/2)}{2 - \exp(-||\boldsymbol{x}-\boldsymbol{y}||^2/2)}$$

ensures that $\tilde{k}$ is positive definite, and therefore so is K_γ^C. The associated cost is then, for all $\boldsymbol{x},\boldsymbol{y} \in \mathbb{R}^d$,

$$c(\boldsymbol{x},\boldsymbol{y}) = -\log(k(\boldsymbol{x},\boldsymbol{y})) = \frac{||\boldsymbol{x}-\boldsymbol{y}||^2}{2} + \log\left(2 - \exp\left(\frac{||\boldsymbol{x}-\boldsymbol{y}||^2}{2}\right)\right),$$

which is exactly the cost (10). Using this cost, the fact that the kernel K_γ^C is positive definite implies that the Gram matrix

$$\boldsymbol{K} = \begin{bmatrix} K_\gamma^C(\boldsymbol{X},\boldsymbol{X}) & K_\gamma^C(\boldsymbol{X},\boldsymbol{Y}) \\ K_\gamma^C(\boldsymbol{Y},\boldsymbol{X}) & K_\gamma^C(\boldsymbol{Y},\boldsymbol{Y}) \end{bmatrix}$$

is positive semi-definite (p.s.d.), i.e., its determinant is non-negative. Using (9), we obtain using the cost (10)

$$\det(\boldsymbol{K}) = K_\gamma^C(\boldsymbol{X},\boldsymbol{X})K_\gamma^C(\boldsymbol{Y},\boldsymbol{Y}) - K_\gamma^C(\boldsymbol{X},\boldsymbol{Y})^2 \geq 0 \Leftrightarrow S_\gamma^C(\boldsymbol{X},\boldsymbol{Y}) \geq 0.$$

Non-negativity with absolute value cost. We now consider the absolute value on $\mathbb{R}\times\mathbb{R}$

$$c(x,y) = |x-y|,$$

and show that K_γ^C is positive definite for this cost. The corresponding kernel is

$$k(x,y) = \exp(-c(x,y)) = \exp(-|x-y|),$$

namely the Laplacian kernel. Following the paragraph above, we show that $\tilde{k} = \frac{k}{1+k}$ is p.d. We first note that $\tilde{k}$ is translation invariant and rewrites $\tilde{k}(x,y) = f(x-y)$, where

$$f(w) \coloneqq \frac{1}{1+\exp(|w|)}.$$

From Bochner's theorem, the function $f : \mathbb{R} \to \mathbb{R}$ is p.d. (i.e. $\tilde{k}$ is p.d.) if and only if it is the Fourier transform of a positive measure. Since f is integrable and square integrable, it suffices to study the sign of its Fourier transform. For all $\omega \in \mathbb{R}$,

$$\begin{aligned}
\mathcal{F}[f](\omega) \coloneqq \int_{-\infty}^{\infty} \frac{e^{-i\omega x}}{1+e^{|x|}} dx &= \int_{-\infty}^{0} \frac{e^{-i\omega x}}{1+e^{-x}} dx + \int_{0}^{\infty} \frac{e^{-i\omega x}}{1+e^{x}} dx \\
&= \int_{0}^{\infty} \frac{e^{-i\omega x}}{1+e^{x}} dx + \int_{0}^{\infty} \frac{e^{i\omega x}}{1+e^{x}} dx \\
&= 2 \int_{0}^{\infty} \frac{\cos(\omega x)}{1+e^{x}} dx \\
&= \frac{2}{\omega} \int_{0}^{\infty} \frac{\cos(x)}{1+e^{x/\omega}} dx \\
&= \frac{2}{\omega} \sum_{k=0}^{\infty} \int_{0}^{2\pi} \frac{\cos(x)}{1+e^{x/\omega + 2k\pi/\omega}} dx \\
&\coloneqq \frac{2}{\omega} \sum_{k=0}^{\infty} \int_{0}^{2\pi} a_k.
\end{aligned}$$

Let us further decompose the sequence $(a_k)_{k=0}^{\infty}$ by splitting the integral into four parts and using the periodicity of the cosine function. For all $k \geq 0$,

$$a_k = \int_0^{\frac{\pi}{2}} \cos(x)\Big(\sigma_k(x) + \sigma_k(2\pi - x) - \sigma_k(\pi + x) - \sigma_k(\pi - x)\Big) dx \coloneqq \int_0^{\frac{\pi}{2}} \cos(x) f_k(x) dx$$

where $\sigma_k(x) \coloneqq \frac{1}{1+e^{\frac{2k\pi + x}{\omega}}}$. Note that σ_k is convex, so that its derivative σ_k' is increasing on $\mathbb{R}$. Therefore, for all $x \in [0, \frac{\pi}{2}]$, we have $\sigma_k'(x) \leq \sigma_k'(\pi - x)$ and $\sigma_k'(\pi + x) \leq \sigma_k'(2\pi - x)$. Hence, for all $x \in [0, \frac{\pi}{2}]$, $f_k'(x) \leq 0$, which implies $f_k(x) \geq f_k(\frac{\pi}{2}) = 0$. We conclude that $\mathcal{F}[f] \geq 0$ on $\mathbb{R}$, and therefore $\tilde{k} = \frac{k}{1+k}$ is p.d. Theorem 1 of Cuturi et al. (2007) ensures that K_γ^C is positive definite, so that D_γ^C is non-negative.

B.5 Numerical verifications for the squared Euclidean cost case

Numerical evidence of the positive definiteness of K_γ^C. We conjecture that K_γ^C is positive definite when C is the squared Euclidean cost (1). This is evidenced by the following numerical experiment. Given M time series $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_M$, we can form the $M \times M$ Gram matrix defined by

$$[\boldsymbol{K}]_{i,j} = K_\gamma^C(\boldsymbol{X}_i, \boldsymbol{X}_j) \quad i, j \in [M].$$

If K_γ^C were not positive definite, the following minimization problem

$$\min_{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_M, \boldsymbol{v}} \frac{1}{||\boldsymbol{v}||^2} \boldsymbol{v}^\top \boldsymbol{K} \boldsymbol{v}$$

would give negative values. We solved this non-convex optimization problem for different values of M using L-BFGS, and could never find negative values. The positive definiteness of K_γ^C would imply the non-negativity of D_γ^C using the squared Euclidean cost.

Disproving a conjecture. Cuturi et al. (2007) notice that the Gaussian kernel $k(\boldsymbol{x}, \boldsymbol{y}) \coloneqq \exp(-||\boldsymbol{x} - \boldsymbol{y}||^2/2)$ is such that $\frac{k}{1+k}$ empirically yields positive semidefinite Gram matrices, and leave open the question of whether $\frac{k}{1+k}$ is indeed a p.d. kernel, which would prove that K_γ^C is p.d. as well (cf. Appendix B.4). We rigorously derive a counter-example showing that this is not the case. The kernel $\tilde{k} = \frac{k}{1+k}$ is translation invariant and rewrites

$$\tilde{k}(\boldsymbol{x}, \boldsymbol{y}) = f(\boldsymbol{x} - \boldsymbol{y}) \quad \text{where} \quad f(\boldsymbol{t}) \coloneqq \frac{\exp(-\|\boldsymbol{t}\|^2/2)}{1 + \exp(-\|\boldsymbol{t}\|^2)}.$$

From Bochner's theorem, the function $f : \mathbb{R}^d \to \mathbb{R}$ is p.d. if and only if it is the Fourier transform of a positive measure. Since f is integrable and square integrable, it suffices to study the sign of its Fourier transform. For that purpose, let us rewrite f as a power series:

$$\forall \boldsymbol{t} \in \mathbb{R}^d : \quad f(\boldsymbol{t}) = \frac{e^{-\frac{||\boldsymbol{t}||^2}{2}}}{1 + e^{-\frac{||\boldsymbol{t}||^2}{2}}} = \sum_{n=1}^{\infty} (-1)^{n+1} e^{-\frac{n||\boldsymbol{t}||^2}{2}} .$$

The convergence is absolute since

$$\sum_{n=1}^{\infty} e^{-\frac{n||\boldsymbol{t}||^2}{2}} = \frac{1}{e^{\frac{||\boldsymbol{t}||^2}{2}} - 1} < \infty.$$

Moreover, this function is integrable. By the theorem of dominated convergence, the Fourier transform of f,

$$\mathcal{F}[f](\boldsymbol{\omega}) \coloneqq \int_{\mathbb{R}^d} f(\boldsymbol{x}) e^{-i\boldsymbol{\omega}^\top \boldsymbol{x}} d\boldsymbol{x} ,$$

is equal to a converging series of Fourier transforms:

$$\mathcal{F}[f](\boldsymbol{\omega}) = \sum_{n=1}^{\infty} (-1)^{n+1} \mathcal{F}\left[e^{-\frac{n||\cdot||^2}{2}} \right] (\boldsymbol{\omega}) .$$

It is well-known that, for any $a \in \mathbb{R}_+$,

$$\mathcal{F}\left[e^{-a||\cdot||^2} \right] (\boldsymbol{\omega}) = \left(\frac{\pi}{a} \right)^{\frac{d}{2}} e^{-\frac{||\boldsymbol{\omega}||^2}{4a}} ,$$

which gives with $a = \frac{n}{2}$

$$\mathcal{F}[f](\boldsymbol{\omega}) = (\pi)^{\frac{d}{2}} \sum_{n=1}^{\infty} \frac{(-1)^{n+1}}{n^{\frac{d}{2}}} e^{-\frac{||\boldsymbol{\omega}||^2}{2n}} .$$

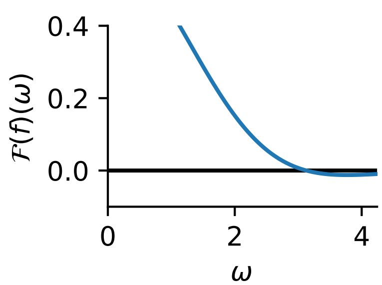

Figure 5: Fourier transform of $\tilde{k} = \frac{k}{1+k}$ when k is the Gaussian kernel. The Fourier transform can be negative.

We may thus compute approximately the coefficients $\mathcal{F}[f](\boldsymbol{\omega})$ for all $\boldsymbol{\omega} \in \mathbb{R}^d$. In dimension $d = 1$, truncating the series at $N = 10^6$, we obtain the curve presented in Figure 5, and observe negative coefficients. To ensure that the infinite sum is negative, we now bound the residual when we truncate the sum at 2N (for $d = 1$):

$$\begin{aligned}
R_N(\boldsymbol{\omega}) &= \sqrt{\pi} \sum_{n=2N+1}^{\infty} \frac{(-1)^{n+1}}{\sqrt{n}} e^{-\frac{||\boldsymbol{\omega}||^2}{2n}} \\
&= \sqrt{\pi} \sum_{n=N}^{\infty} \left[\frac{e^{-\frac{||\boldsymbol{\omega}||^2}{2(2n+1)}}}{\sqrt{2n+1}} - \frac{e^{-\frac{||\boldsymbol{\omega}||^2}{2(2n+2)}}}{\sqrt{2n+2}} \right] \\
&\leq \sqrt{\pi} \sum_{n=N}^{\infty} \left[\frac{e^{-\frac{||\boldsymbol{\omega}||^2}{2(2n+2)}}}{\sqrt{2n+1}} - \frac{e^{-\frac{||\boldsymbol{\omega}||^2}{2(2n+2)}}}{\sqrt{2n+2}} \right] \\
&\leq \sqrt{\pi} \sum_{n=N}^{\infty} \left[\frac{1}{\sqrt{2n+1}} - \frac{1}{\sqrt{2n+2}} \right] \\
&= \sqrt{\pi} \sum_{n=N}^{\infty} \frac{1}{\sqrt{2n+1}} \left[1 - \sqrt{1 - \frac{1}{2n+2}} \right] \\
&\leq \sqrt{\pi} \sum_{n=N}^{\infty} \frac{1}{\sqrt{2n+1}(2n+2)} \\
&\leq \sqrt{\frac{\pi}{8}} \sum_{n=N}^{\infty} \frac{1}{n\sqrt{n}} \\
&\leq \sqrt{\frac{\pi}{8}} \int_{N-1}^{\infty} \frac{dx}{x\sqrt{x}} \\
&= \sqrt{\frac{\pi}{2(N-1)}}.
\end{aligned}$$

For $N = 10^6$, this gives $R_N(\boldsymbol{\omega}) < 2 \times 10^{-3}$. We observed numerically some values strictly smaller than -2×10^{-3} for the truncation at $N = 10^6$ of the series: in particular, $\mathcal{F}[f](2.65) = -.012$, which implies that the infinite sum is negative. We therefore conclude that $\frac{k}{k+1}$ is not positive definite when k is the Gaussian kernel. Note that this does not disprove the positive definiteness of K_γ^C using the squared Euclidean cost.

B.6 Proof of Proposition 4 (stationary point using the squared Euclidean cost)

Soft-DTW divergence. We recall that we denote $C(\boldsymbol{X}) \coloneqq C(\boldsymbol{X}, \boldsymbol{X}) \in \mathbb{R}^{m \times m}$. Using (8), we have

$$\nabla_{\boldsymbol{X}} D_\gamma^C(\boldsymbol{X}, \boldsymbol{Y}) = (J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{Y}))^\top \boldsymbol{E}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y})) - \frac{1}{2}(J_{\boldsymbol{X}} C(\boldsymbol{X}))^\top \boldsymbol{E}_\gamma(C(\boldsymbol{X})).$$

Note that at $\boldsymbol{X} = \boldsymbol{Y}$, $\boldsymbol{E}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y})) = \boldsymbol{E}_\gamma(C(\boldsymbol{X})) \in \mathbb{R}^{m \times m}$ is a symmetric matrix. Therefore, in order to have $\nabla_{\boldsymbol{X}} D_\gamma^C(\boldsymbol{X}, \boldsymbol{Y}) = \boldsymbol{0}_{m \times d}$ at $\boldsymbol{X} = \boldsymbol{Y}$, it suffices that $(J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{Y}))^\top$ and $\frac{1}{2}(J_{\boldsymbol{X}} C(\boldsymbol{X}))^\top$ map symmetric matrices to the same matrix. From (19), this is indeed the case for the squared Euclidean cost.

Sharp divergence. Using (14), we get

$$\begin{aligned}
\nabla_{\boldsymbol{X}} \text{SHARP}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y})) &= (J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{Y}))^\top \nabla_{\boldsymbol{C}} \text{SHARP}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y})) \\
&= (J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{Y}))^\top [\boldsymbol{E}_\gamma(\boldsymbol{C}) + \frac{1}{\gamma} \nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y})) C(\boldsymbol{X}, \boldsymbol{Y})] \\
&= \nabla_{\boldsymbol{X}} \text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y})) + \frac{1}{\gamma}(J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{Y}))^\top \nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y})) C(\boldsymbol{X}, \boldsymbol{Y}).
\end{aligned}$$

We therefore have

$$\begin{aligned}
\nabla_{\boldsymbol{X}} S_\gamma^C(\boldsymbol{X}, \boldsymbol{Y}) = \nabla_{\boldsymbol{X}} D_\gamma^C(\boldsymbol{X}, \boldsymbol{Y}) &+ \frac{1}{\gamma}(J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{Y}))^\top \nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y})) C(\boldsymbol{X}, \boldsymbol{Y}) \\
&- \frac{1}{2\gamma}(J_{\boldsymbol{X}} C(\boldsymbol{X}))^\top \nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(C(\boldsymbol{X})) C(\boldsymbol{X}). \qquad (21)
\end{aligned}$$

From the previous paragraph, we know that $\nabla_{\boldsymbol{X}} D_\gamma^C(\boldsymbol{X}, \boldsymbol{Y}) = \mathbf{0}_{m \times d}$ at $\boldsymbol{X} = \boldsymbol{Y}$ using the squared Euclidean cost. It remains to show that the sum of the other two terms in (21) is also equal to $\mathbf{0}_{m \times d}$. Since $(J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{Y}))^\top$ and $\frac{1}{2}(J_{\boldsymbol{X}} C(\boldsymbol{X}))^\top$ map symmetric matrices to the same matrix using the squared Euclidean cost, it suffices to show that $\nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(C(\boldsymbol{X}))C(\boldsymbol{X})$ is a symmetric matrix.

It is well-known that the Hessian of the log-partition under a Gibbs distribution is equal to the covariance matrix (Wainwright and Jordan, 2008). The Hessian can be seen as a $mn \times mn$ matrix. Accounting for the negative sign in (4), we have

$$\begin{aligned}
\nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(\boldsymbol{C}) &= -\mathbb{E}_\gamma[\text{vec}(A - \boldsymbol{E}_\gamma(\boldsymbol{C}))\text{vec}(A - \boldsymbol{E}_\gamma(\boldsymbol{C}))^\top] \\
&= -\sum_{\boldsymbol{A} \in \mathcal{A}(m,n)} \mathbb{P}_\gamma(\boldsymbol{A}; \boldsymbol{C})\text{vec}(\boldsymbol{A} - \boldsymbol{E}(\boldsymbol{C}))\text{vec}(\boldsymbol{A} - \boldsymbol{E}(\boldsymbol{C}))^\top \\
&= \mathbb{E}_\gamma[\text{vec}(A)]\mathbb{E}_\gamma[\text{vec}(A)]^\top - \mathbb{E}_\gamma[\text{vec}(A)\text{vec}(A)^\top],
\end{aligned}$$

where A is a random alignment matrix distributed according to $\mathbb{P}_\gamma(\boldsymbol{A}; \boldsymbol{C})$. Equivalently, we can see the Hessian as linear map from $\mathbb{R}^{m \times n}$ to $\mathbb{R}^{m \times n}$. Applying that map to a matrix $\boldsymbol{M} \in \mathbb{R}^{m \times n}$, we obtain

$$\begin{aligned}
\nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(\boldsymbol{C})\boldsymbol{M} &= -\sum_{\boldsymbol{A} \in \mathcal{A}(m,n)} \mathbb{P}_\gamma(\boldsymbol{A}; \boldsymbol{C})(\boldsymbol{A} - \boldsymbol{E}_\gamma(\boldsymbol{C}))\langle \boldsymbol{A} - \boldsymbol{E}_\gamma(\boldsymbol{C}), \boldsymbol{M} \rangle \\
&= \langle \boldsymbol{E}_\gamma(\boldsymbol{C}), \boldsymbol{M} \rangle \boldsymbol{E}_\gamma(\boldsymbol{C}) - \sum_{\boldsymbol{A} \in \mathcal{A}(m,n)} \mathbb{P}_\gamma(\boldsymbol{A}; \boldsymbol{C})\langle \boldsymbol{A}, \boldsymbol{M} \rangle \boldsymbol{A} \\
&= \langle \boldsymbol{E}_\gamma(\boldsymbol{C}), \boldsymbol{M} \rangle \boldsymbol{E}_\gamma(\boldsymbol{C}) - \mathbb{E}_\gamma[\langle A, \boldsymbol{M} \rangle A],
\end{aligned}$$

which is the difference of two symmetric matrices when $\boldsymbol{C} = \boldsymbol{M} = C(\boldsymbol{X})$.

B.7 Multiplication with the Hessian

For completeness, we also include a discussion on the multiplication with the Hessian w.r.t. $\boldsymbol{X}$. The product between the Hessian $\nabla^2_{\boldsymbol{X}} \text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))$ and any $\boldsymbol{Z} \in \mathbb{R}^{m \times d}$ is equal to the product between the Jacobian of $\nabla_{\boldsymbol{X}} \text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))$ and $\boldsymbol{Z}$:

$$\nabla^2_{\boldsymbol{X}} \text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))\boldsymbol{Z} = J_{\boldsymbol{X}}[\nabla_{\boldsymbol{X}} \text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))]\boldsymbol{Z} = J_{\boldsymbol{X}}[J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{Y})^\top \boldsymbol{E}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))]\boldsymbol{Z}.$$

Using the product rule and the chain rule, we obtain

$$\nabla^2_{\boldsymbol{X}} \text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))\boldsymbol{Z} = \underbrace{[J_{\boldsymbol{X}}(J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{Y}))^\top \boldsymbol{E}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))]}_{\boldsymbol{B}_\gamma(\boldsymbol{X}, \boldsymbol{Y})} \boldsymbol{Z} + (J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{Y}))^\top \nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{Y})\boldsymbol{Z}.$$

Similarly,

$$\nabla^2_{\boldsymbol{X}} \text{SDTW}_\gamma(C(\boldsymbol{X}))\boldsymbol{Z} = \underbrace{[J_{\boldsymbol{X}}(J_{\boldsymbol{X}} C(\boldsymbol{X}))^\top \boldsymbol{E}_\gamma(C(\boldsymbol{X}))]}_{\boldsymbol{B}_\gamma(\boldsymbol{X})} \boldsymbol{Z} + (J_{\boldsymbol{X}} C(\boldsymbol{X}))^\top \nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(C(\boldsymbol{X}))J_{\boldsymbol{X}} C(\boldsymbol{X})\boldsymbol{Z}.$$

From now on, we assume the squared Euclidean cost. Using (17), we obtain

$$[\boldsymbol{B}_\gamma(\boldsymbol{X}, \boldsymbol{Y})\boldsymbol{Z}]_{i,k} = \sum_{j=1}^n [\boldsymbol{E}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))]_{i,j} z_{i,k} \quad i \in [m], k \in [d]$$

or equivalently

$$\boldsymbol{B}_\gamma(\boldsymbol{X}, \boldsymbol{Y})\boldsymbol{Z} = \boldsymbol{Z} \circ (\boldsymbol{E}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))\mathbf{1}_{n \times d}) \in \mathbb{R}^{m \times d}.$$

Similarly, using (18) and the fact that $\boldsymbol{E}_\gamma(C(\boldsymbol{X}))$ is a symmetric matrix, we obtain

$$[\boldsymbol{B}_\gamma(\boldsymbol{X})\boldsymbol{Z}]_{i,k} = 2\sum_{j=1}^n [\boldsymbol{E}_\gamma(C(\boldsymbol{X}))]_{i,j}(z_{i,k} - z_{j,k})$$

or equivalently

$$\boldsymbol{B}_\gamma(\boldsymbol{X})\boldsymbol{Z} = 2\boldsymbol{Z} \circ (\boldsymbol{E}_\gamma(C(\boldsymbol{X})\mathbf{1}_{m \times d}) - 2\boldsymbol{E}_\gamma(C(\boldsymbol{X}))\boldsymbol{Z} \in \mathbb{R}^{m \times d}.$$

At $\boldsymbol{X} = \boldsymbol{Y}$, we therefore get

$$\boldsymbol{B}_\gamma(\boldsymbol{X}, \boldsymbol{Y})\boldsymbol{Z} - \frac{1}{2}\boldsymbol{B}_\gamma(\boldsymbol{X})\boldsymbol{Z} = \boldsymbol{E}_\gamma(C(\boldsymbol{X}))^\top \boldsymbol{Z} = \boldsymbol{E}_\gamma(C(\boldsymbol{X}))\boldsymbol{Z}.$$

At $\boldsymbol{X} = \boldsymbol{Y}$, from (19) and (20), we also have

$$(J_{\boldsymbol{X}} C(\boldsymbol{X}))^\top \nabla^2_{\boldsymbol{C}} \boldsymbol{E}_\gamma(C(\boldsymbol{X})) J_{\boldsymbol{X}} C(\boldsymbol{X})\boldsymbol{Z} = 2 J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{X})^\top \nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(C(\boldsymbol{X}))(J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{X})\boldsymbol{Z} + (J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{X})\boldsymbol{Z})^\top).$$

Putting everything together, at $\boldsymbol{X} = \boldsymbol{Y}$, we have

$$\begin{aligned}
\nabla^2_{\boldsymbol{X}} D^C_\gamma(\boldsymbol{X}, \boldsymbol{Y})\boldsymbol{Z} &= \nabla^2_{\boldsymbol{X}} \text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y}))\boldsymbol{Z} - \frac{1}{2}\nabla^2_{\boldsymbol{X}} \text{SDTW}_\gamma(C(\boldsymbol{X}))\boldsymbol{Z} \\
&= \boldsymbol{E}_\gamma(C(\boldsymbol{X}))\boldsymbol{Z} - J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{X})^\top \nabla^2_{\boldsymbol{C}} \text{SDTW}_\gamma(C(\boldsymbol{X}))(J_{\boldsymbol{X}} C(\boldsymbol{X}, \boldsymbol{X})\boldsymbol{Z})^\top.
\end{aligned}$$

An open question is to prove that $\boldsymbol{X} = \boldsymbol{Y}$ is a local minimum, i.e., $\langle Z, \nabla^2_{\boldsymbol{X}} D^C_\gamma(\boldsymbol{X}, \boldsymbol{Y})\boldsymbol{Z}\rangle > 0$ for all $\boldsymbol{Z} \in \mathbb{R}^{m \times d}$.

B.8 Proof of Proposition 5 (limits w.r.t. γ)

Limit to zero. Since both $\text{SDTW}_\gamma(\boldsymbol{C})$ and $\text{SHARP}_\gamma(\boldsymbol{C})$ converge to $\text{DTW}(\boldsymbol{C})$ when $\gamma \to 0$, both $D^C_\gamma(\boldsymbol{X}, \boldsymbol{Y})$ and $S^C_\gamma(\boldsymbol{X}, \boldsymbol{Y})$ converge to

$$\text{DTW}(C(\boldsymbol{X}, \boldsymbol{Y})) - \frac{1}{2}\text{DTW}(C(\boldsymbol{X}, \boldsymbol{X})) - \frac{1}{2}\text{DTW}(C(\boldsymbol{Y}, \boldsymbol{Y})).$$

Since the optimal alignment of $\boldsymbol{A}^\star(\boldsymbol{C}(\boldsymbol{X}, \boldsymbol{X}))$ is the identity matrix under assumption A.2, we have $\text{DTW}(C(\boldsymbol{X}, \boldsymbol{X})) = 0$ and similarly $\text{DTW}(C(\boldsymbol{Y}, \boldsymbol{Y})) = 0$. Therefore, both $D^C_\gamma(\boldsymbol{X}, \boldsymbol{Y})$ and $S^C_\gamma(\boldsymbol{X}, \boldsymbol{Y})$ converge to $\text{DTW}(C(\boldsymbol{X}, \boldsymbol{Y}))$.

Limit to infinity. From (6), when $\gamma \to \infty$, the solution becomes the maximum entropy one, $\boldsymbol{p}^\star = \mathbf{1}/|\mathcal{A}(m, n)|$. Hence, $\langle \boldsymbol{p}^\star, s(\boldsymbol{C})\rangle$ converge to the mean cost (15). This gives the limit for the S^C_γ case. For the D^C_γ case, we also need to take into account the entropy terms

$$-\gamma H(\boldsymbol{p}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y})) + \frac{\gamma}{2} H(\boldsymbol{p}_\gamma(C(\boldsymbol{X}, \boldsymbol{X}))) + \frac{\gamma}{2} H(\boldsymbol{p}_\gamma(C(\boldsymbol{Y}, \boldsymbol{Y}))).$$

When $\gamma \to \infty$, each term attains the maximum entropy value and we get

$$-\gamma \log |\mathcal{A}(m, n)| + \frac{\gamma}{2} \log |\mathcal{A}(m, m)| + \frac{\gamma}{2} \log |\mathcal{A}(n, n)| = \frac{\gamma}{2} \log \frac{|\mathcal{A}(m, m)||\mathcal{A}(n, n)|}{|\mathcal{A}(m, n)|^2}.$$

When $m = n$, the terms cancel out. Hence, $D^C_\gamma(\boldsymbol{X}, \boldsymbol{Y})$ converge. When, $m \neq n$, the positive terms are stronger, and the limit goes to ∞. By definition, we have

$$\begin{aligned}
D^C_\gamma(\boldsymbol{X}, \boldsymbol{Y}) &= \text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{Y})) - \frac{1}{2}\text{SDTW}_\gamma(C(\boldsymbol{X}, \boldsymbol{X})) - \frac{1}{2}\text{SDTW}_\gamma(C(\boldsymbol{Y}, \boldsymbol{Y})) \\
&= -\gamma \log \sum_{\boldsymbol{A} \in \mathcal{A}(m,n)} \exp(-\langle \boldsymbol{A}, C(\boldsymbol{X}, \boldsymbol{Y})\rangle/\gamma) \\
&\quad + \frac{\gamma}{2} \log \sum_{\boldsymbol{A} \in \mathcal{A}(m,m)} \exp(-\langle \boldsymbol{A}, C(\boldsymbol{X}, \boldsymbol{X})\rangle/\gamma) + \frac{\gamma}{2} \log \sum_{\boldsymbol{A} \in \mathcal{A}(n,n)} \exp(-\langle \boldsymbol{A}, C(\boldsymbol{Y}, \boldsymbol{Y})\rangle/\gamma) \\
&= -\frac{\gamma}{2} \log \frac{|\mathcal{A}(m, n)|^2}{|\mathcal{A}(m, m)||\mathcal{A}(n, n)|} - \gamma \log \left[\frac{1}{|\mathcal{A}(m, n)|} \sum_{\boldsymbol{A} \in \mathcal{A}(m,n)} \exp(-\langle \boldsymbol{A}, C(\boldsymbol{X}, \boldsymbol{Y})\rangle/\gamma)\right] \qquad (22) \\
&\quad + \frac{\gamma}{2} \log \left[\frac{1}{|\mathcal{A}(m, m)|} \sum_{\boldsymbol{A} \in \mathcal{A}(m,m)} \exp(-\langle \boldsymbol{A}, C(\boldsymbol{X}, \boldsymbol{X})\rangle/\gamma)\right] \\
&\quad + \frac{\gamma}{2} \log \left[\frac{1}{|\mathcal{A}(n, n)|} \sum_{\boldsymbol{A} \in \mathcal{A}(n,n)} \exp(-\langle \boldsymbol{A}, C(\boldsymbol{Y}, \boldsymbol{Y})\rangle/\gamma)\right]
\end{aligned}$$

Let us first consider the limit of the second term in this sum when $\gamma \to +\infty$:

$$\begin{aligned}\gamma \log \left[\frac{1}{|\mathcal{A}(m,n)|} \sum_{\boldsymbol{A} \in \mathcal{A}(m,n)} \exp(-\langle \boldsymbol{A}, C(\boldsymbol{X},\boldsymbol{Y})\rangle/\gamma)\right] &= \gamma \log \left[\frac{1}{|\mathcal{A}(m,n)|} \sum_{\boldsymbol{A} \in \mathcal{A}(m,n)} \left(1 - \frac{\langle \boldsymbol{A}, C(\boldsymbol{X},\boldsymbol{Y})\rangle}{\gamma} + o(1/\gamma)\right)\right] \\ &= \gamma \log \left[1 - \frac{\text{MEAN\_COST}(C(\boldsymbol{X},\boldsymbol{Y}))}{\gamma} + o(1/\gamma)\right] \\ &= -\text{MEAN\_COST}(C(\boldsymbol{X},\boldsymbol{Y})) + o(1)\,.\end{aligned}$$

A similar computation for the third and fourth term in (22) leads to

$$\begin{aligned}D_\gamma^C(\boldsymbol{X},\boldsymbol{Y}) &= -\frac{\gamma}{2} \log \frac{|\mathcal{A}(m,n)|^2}{|\mathcal{A}(m,m)||\mathcal{A}(n,n)|} + \text{MEAN\_COST}(C(\boldsymbol{X},\boldsymbol{Y})) - \frac{1}{2}\text{MEAN\_COST}(C(\boldsymbol{X},\boldsymbol{X})) \\ &\quad - \frac{1}{2}\text{MEAN\_COST}(C(\boldsymbol{Y},\boldsymbol{Y})) + o(1) \\ &= -\frac{\gamma}{2} \log \frac{|\mathcal{A}(m,n)|^2}{|\mathcal{A}(m,m)||\mathcal{A}(n,n)|} + M^C(\boldsymbol{X},\boldsymbol{Y}) + o(1)\,.\end{aligned}$$

When $m = n$, the first term is equal to 0, so we get $\lim_{\gamma \to +\infty} D_\gamma^C(\boldsymbol{X},\boldsymbol{Y}) = M^C(\boldsymbol{X},\boldsymbol{Y})$. When $m \neq n$, on the other hand, we can use the fact that for any integers m, n:

$$|\mathcal{A}(m,n)| = \text{Delannoy}(m-1, n-1)\,,$$

where $\text{Delannoy}(m,n)$ is the Delannoy number, i.e., the number of paths on a rectangular grid from the origin $(0,0)$ to the northeast corner (m,n), using only single steps north, east or northeast (the $(m-1, n-1)$ term stems from the fact that alignment matrices represent paths starting from $(1,1)$ and not $(0,0)$). We can now use Lemma 1 below to get, when $m \neq n$:

$$\log \frac{|\mathcal{A}(m,n)|^2}{|\mathcal{A}(m,m)||\mathcal{A}(n,n)|} = \log \frac{\text{Delannoy}(m-1,n-1)^2}{\text{Delannoy}(m-1,m-1) \times \text{Delannoy}(n-1,n-1)} < 0\,,$$

and therefore that $\lim_{\gamma \to +\infty} D_\gamma^C(\boldsymbol{X},\boldsymbol{Y}) = +\infty$.

Lemma 1. *For any $m, n \in \mathbb{N}$, if $m \neq n$ then*

$$\log \frac{\text{Delannoy}(m,n)^2}{\text{Delannoy}(m,m) \times \text{Delannoy}(n,n)} < 0\,.$$

Proof. We use the following characterization of Delannoy numbers (e.g., Banderier and Schwer, 2005):

$$\text{Delannoy}(m,n) = \sum_{k=0}^{\min(m,n)} \binom{m}{k} \binom{n}{k} 2^k\,,$$

to obtain, assuming without loss of generality that $m < n$:

$$\begin{aligned}\text{Delannoy}(m,n)^2 &= \left[\sum_{k=0}^{m} \binom{m}{k} \binom{n}{k} 2^k\right]^2 \\ &\leq \left[\sum_{k=0}^{m} \binom{m}{k}^2 2^k\right] \times \left[\sum_{k=0}^{m} \binom{n}{k}^2 2^k\right] \\ &< \left[\sum_{k=0}^{m} \binom{m}{k}^2 2^k\right] \times \left[\sum_{k=0}^{n} \binom{n}{k}^2 2^k\right] \\ &= \text{Delannoy}(m,m) \times \text{Delannoy}(n,n)\,,\end{aligned}$$

where we used Cauchy-Schwartz inequality for the first inequality, and the fact that $m < n$ for the second (strict) inequality. ∎

C Additional empirical results

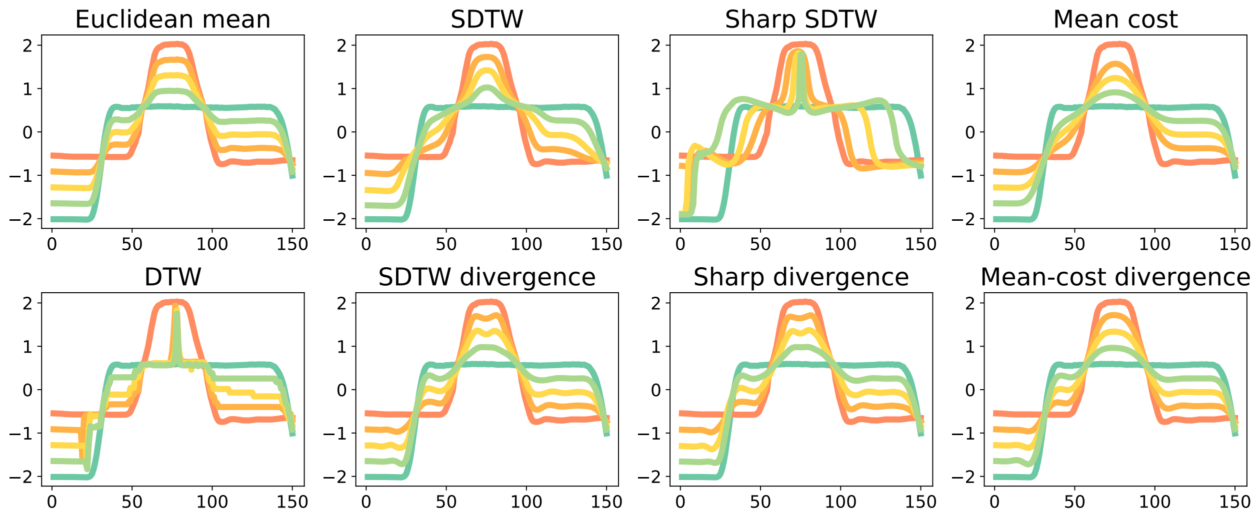

Figure 6: Interpolation between two time series, from the GunPoint dataset.

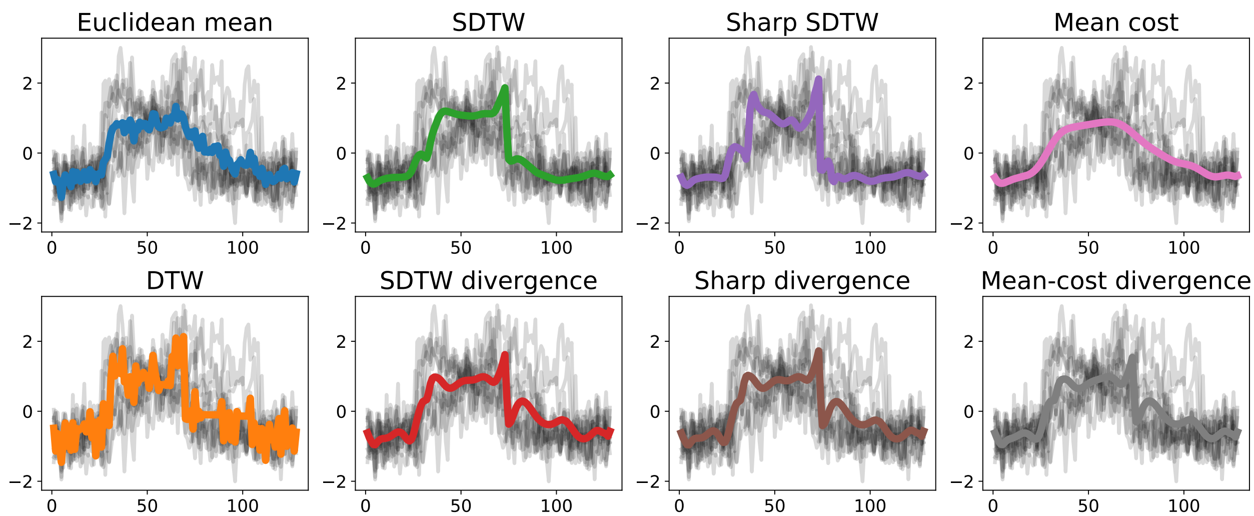

Figure 7: Barycenters on the **CBF** dataset.

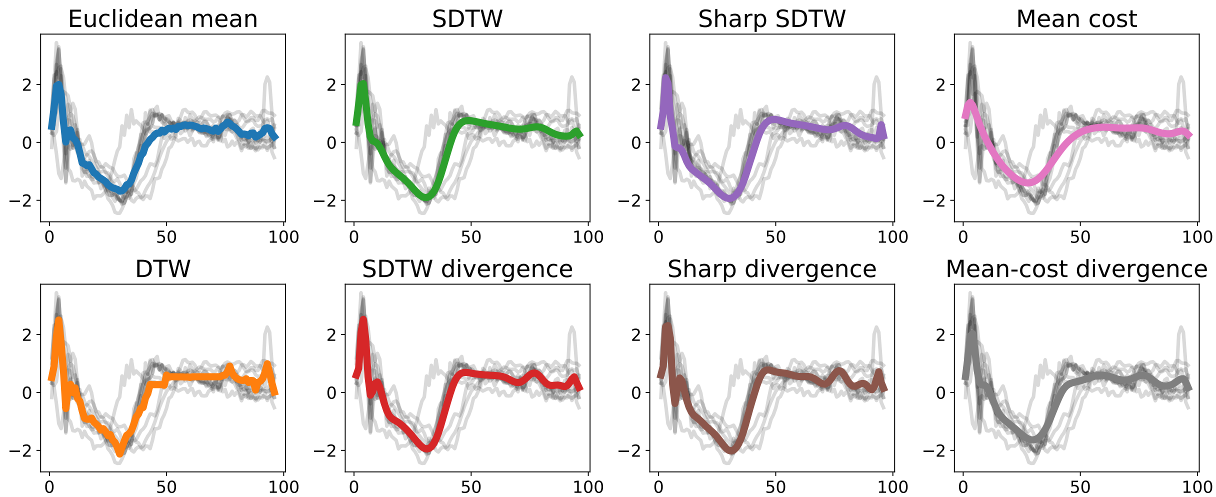

Figure 8: Barycenters on the **ECG200** dataset.

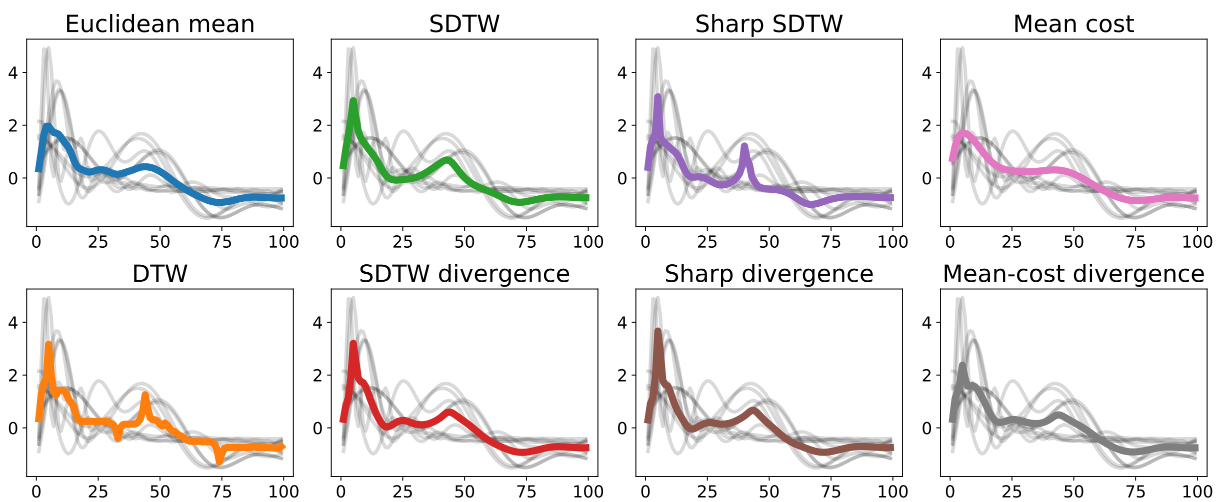

Figure 9: Barycenters on the **Medical Images** dataset.

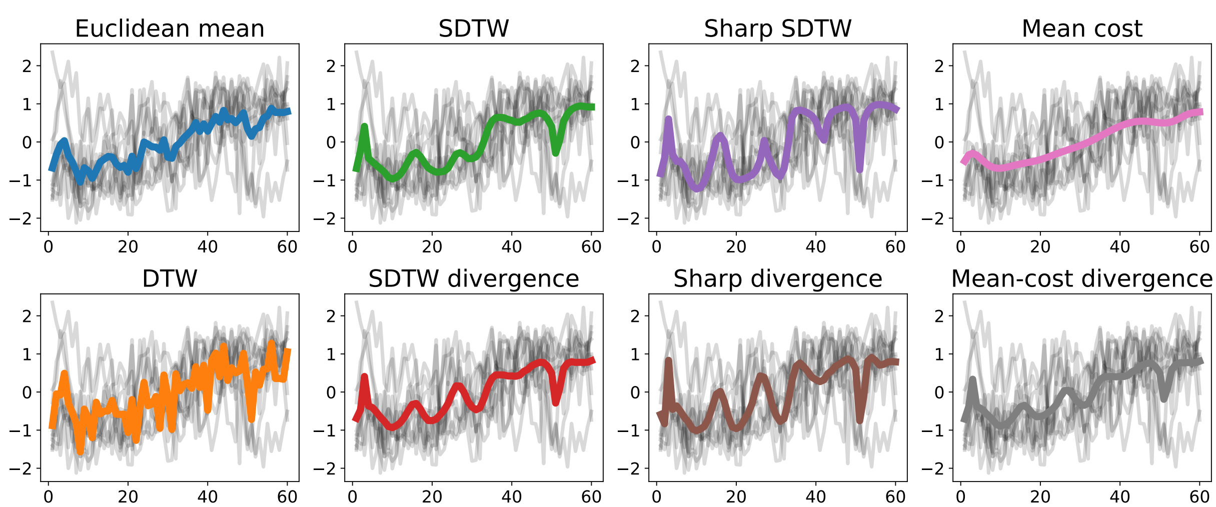

Figure 10: Barycenters on the **synthetic control** dataset.

Table 4: **Three nearest neighbors results**. Each number indicates the percentage of datasets in the UCR archive for which using A in the nearest neighbor classifier is within 99% or better than using B .

A (↓) vs. B (→)	Euc.	DTW	SDTW	SDTW div	Sharp	Sharp div	Mean cost	Mean-cost div
Euc.	-	39.29	29.49	31.17	37.18	28.00	95.24	65.48
DTW	70.24	-	53.85	45.45	57.69	42.67	90.48	83.33
SDTW	82.05	88.46	-	66.23	83.33	58.67	98.72	89.74
SDTW div	90.91	84.42	85.71	-	83.12	70.67	98.70	94.81
Sharp	78.21	82.05	64.10	58.44	-	53.33	98.72	87.18
Sharp div	86.67	90.67	81.33	77.33	89.33	-	98.67	96.00
Mean cost	8.33	13.10	6.41	3.90	5.13	4.00	-	44.05
Mean-cost div	46.43	34.52	24.36	20.78	24.36	21.33	98.81	-

Table 5: **Five nearest neighbor results**. Each number indicates the percentage of datasets in the UCR archive for which using A in the nearest neighbor classifier is within 99% or better than using B .

A (↓) vs. B (→)	Euc.	DTW	SDTW	SDTW div	Sharp	Sharp div	Mean cost	Mean-cost div
Euc.	-	40.48	30.77	28.57	33.33	24.68	95.29	70.24
DTW	73.81	-	48.72	44.16	55.13	45.45	88.10	83.33
SDTW	85.90	84.62	-	61.04	74.36	63.64	94.87	82.05
SDTW div	84.42	88.31	81.82	-	81.82	74.03	96.10	85.71
Sharp	85.90	87.18	70.51	58.44	-	59.74	97.44	82.05
Sharp div	90.91	84.42	80.52	76.62	84.42	-	96.10	87.01
Mean cost	10.59	13.10	10.26	7.79	7.69	7.79	-	45.24
Mean-cost div	45.24	32.14	26.92	20.78	26.92	19.48	98.81	-

Table 6: **Nearest neighbor classification accuracy with $k = 1$.**

Dataset name	Euc.	DTW	SDTW	SDTW div	Sharp	Sharp div	Mean cost	Mean-cost div
50words	63.08	69.01	80.66	**81.54**	79.12	79.78	58.90	67.91
Adiac	61.13	60.36	61.38	71.36	60.10	**72.12**	28.39	54.48
ArrowHead	80.00	70.29	77.14	**81.71**	80.57	79.43	72.57	78.86
Beef	**66.67**	63.33	63.33	63.33	63.33	63.33	20.00	20.00
BeetleFly	**75.00**	70.00	70.00	70.00	70.00	**75.00**	50.00	50.00
BirdChicken	55.00	**75.00**	**75.00**	**75.00**	**75.00**	**75.00**	50.00	50.00
CBF	85.22	**99.67**	**99.67**	**99.67**	**99.67**	**99.67**	78.78	95.00
Car	73.33	73.33	73.33	75.00	75.00	**78.33**	23.33	23.33
ChlorineConcentration	65.00	64.84	62.29	64.84	65.05	**65.65**	38.20	55.44
CinC_ECG_torso	89.71	65.07	93.41	93.55	92.54	**93.84**	25.36	25.36
Coffee	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	53.57	96.43
Computers	57.60	**70.00**	69.60	**70.00**	69.20	67.20	50.00	50.00
Cricket_X	57.69	75.38	77.69	**80.00**	77.95	79.23	42.56	61.54
Cricket_Y	56.67	74.36	76.67	**78.72**	74.36	77.18	47.95	61.28
Cricket_Z	58.72	75.38	77.69	**80.26**	77.69	79.74	43.08	63.33
DiatomSizeReduction	93.46	**96.73**	92.16	94.44	92.81	93.46	92.16	93.46
DistalPhalanxOutlineAgeGroup	78.25	79.25	79.25	79.75	79.50	**80.50**	59.50	76.75
DistalPhalanxOutlineCorrect	75.17	76.83	**79.00**	76.83	76.83	75.17	36.83	71.33
DistalPhalanxTW	72.75	70.75	73.25	72.25	**74.50**	72.50	51.00	71.00
ECG200	**88.00**	77.00	86.00	**88.00**	82.00	87.00	87.00	**88.00**
ECG5000	92.49	92.44	**93.07**	92.36	92.78	92.47	91.80	92.38
ECGFiveDays	79.67	76.77	61.67	**93.50**	62.49	91.17	61.44	83.86
Earthquakes	67.39	74.22	**82.61**	74.53	**82.61**	74.22	81.99	81.99
ElectricDevices	54.93	**60.02**	NA	NA	NA	NA	26.17	59.12
FISH	78.29	82.29	92.00	**92.57**	90.29	91.43	12.57	12.57
FaceAll	71.36	80.77	74.38	82.31	76.27	**82.78**	25.33	81.89
FaceFour	78.41	82.95	82.95	**89.77**	87.50	**89.77**	62.50	84.09
FacesUCR	76.93	90.49	92.34	**94.78**	92.34	94.54	45.90	80.44
FordA	**65.90**	56.21	NA	NA	NA	NA	51.26	51.26
FordB	55.78	**59.41**	58.55	NA	58.83	NA	48.84	48.84
Gun_Point	91.33	90.67	97.33	**98.00**	**98.00**	**98.00**	82.00	90.00
Ham	60.00	46.67	49.52	58.10	58.10	**61.90**	48.57	48.57
HandOutlines	**80.10**	79.80	NA	NA	NA	NA	63.80	63.80
Haptics	37.01	37.66	39.94	39.94	40.26	**41.56**	21.75	21.75
Herring	51.56	53.12	57.81	57.81	60.94	**62.50**	59.38	59.38
InlineSkate	34.18	38.36	42.55	**43.09**	42.00	42.36	15.64	15.64
InsectWingbeatSound	56.16	35.51	55.05	56.87	56.26	**57.07**	54.55	56.97
ItalyPowerDemand	**95.53**	95.04	93.68	95.04	94.07	95.43	90.38	94.95
LargeKitchenAppliances	49.33	79.47	**79.73**	**79.73**	**79.73**	**79.73**	33.33	33.33
Lighting2	75.41	86.89	**90.16**	88.52	**90.16**	86.89	54.10	54.10
Lighting7	57.53	72.60	73.97	78.08	75.34	**82.19**	57.53	68.49
MALLAT	91.43	**93.39**	89.72	91.39	90.62	92.24	12.54	12.54
Meat	93.33	93.33	**95.00**	93.33	**95.00**	93.33	33.33	33.33
MedicalImages	68.42	73.68	74.61	75.92	76.18	**77.76**	57.89	69.61
MiddlePhalanxOutlineAgeGroup	74.00	75.00	71.00	73.25	**75.25**	73.75	66.25	73.25
MiddlePhalanxOutlineCorrect	75.33	64.83	72.67	**76.33**	66.83	71.83	35.33	70.67
MiddlePhalanxTW	56.14	58.40	58.40	58.40	58.40	58.40	52.63	**59.15**
MoteStrain	87.86	83.47	90.18	89.86	**91.53**	87.62	88.18	80.35
NonInvasiveFatalECG_Thorax1	**82.90**	78.98	NA	NA	NA	NA	2.44	2.44
NonInvasiveFatalECG_Thorax2	**87.99**	86.46	NA	NA	NA	NA	2.44	2.44
OSULeaf	52.07	59.09	**70.25**	69.83	**70.25**	69.83	9.50	9.50
OliveOil	**86.67**	83.33	**86.67**	**86.67**	**86.67**	**86.67**	16.67	16.67
PhalangesOutlinesCorrect	76.11	72.61	74.59	77.04	71.91	**77.39**	42.31	73.08
Phoneme	10.92	22.84	**24.00**	22.73	21.89	23.26	2.00	2.00
Plane	96.19	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	84.76	96.19
ProximalPhalanxOutlineAgeGroup	78.54	80.49	75.12	**80.98**	**80.98**	**80.98**	46.34	76.59
ProximalPhalanxOutlineCorrect	80.76	77.66	79.04	**83.51**	74.23	**83.51**	31.96	73.20
ProximalPhalanxTW	70.75	74.00	74.75	70.25	**75.00**	73.25	45.25	70.25
RefrigerationDevices	39.47	**46.40**	45.87	44.80	45.60	NA	33.33	33.33
ScreenType	36.00	40.00	**41.33**	40.27	39.47	39.47	33.33	33.33
ShapeletSim	53.89	65.00	58.33	**87.22**	64.44	82.78	50.00	50.00
ShapesAll	75.17	76.83	83.67	**84.33**	80.83	82.17	1.67	1.67
SmallKitchenAppliances	34.40	64.27	66.67	66.67	**67.47**	65.87	33.33	33.33
SonyAIBORobotSurface	69.55	72.55	72.55	**76.71**	72.55	76.54	45.42	76.04
SonyAIBORobotSurfaceII	**85.94**	83.11	84.26	84.89	83.11	83.95	76.39	84.05
StarLightCurves	**84.88**	NA	NA	NA	NA	NA	57.72	NA
Strawberry	93.80	**93.96**	**93.96**	93.80	93.80	93.64	79.45	93.80
SwedishLeaf	78.88	79.20	82.40	88.16	82.24	**89.12**	46.72	79.84
Symbols	89.95	94.97	**96.18**	95.38	95.18	95.28	86.93	90.15
ToeSegmentation1	67.98	77.19	**83.33**	82.89	80.26	81.58	63.16	63.16
ToeSegmentation2	80.77	83.85	90.77	86.15	**92.31**	**92.31**	79.23	83.85
Trace	76.00	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	47.00	72.00
TwoLeadECG	74.71	**90.52**	**90.52**	90.43	89.73	88.59	57.77	70.15
Two_Patterns	90.68	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	94.78	96.72
UWaveGestureLibraryAll	**94.81**	89.17	NA	NA	NA	NA	12.53	12.53
Wine	61.11	57.41	55.56	**62.96**	55.56	**62.96**	50.00	61.11
WordsSynonyms	61.76	64.89	76.80	**78.06**	74.92	76.49	55.33	65.20
Worms	36.46	46.41	47.51	48.07	**49.17**	42.54	41.99	41.99
WormsTwoClass	58.56	66.30	55.80	**67.40**	57.46	64.09	41.99	41.99
synthetic_control	88.00	**99.33**	97.67	**99.33**	**99.33**	**99.33**	76.67	98.67
uWaveGestureLibrary_X	73.93	72.75	78.48	**78.73**	77.58	78.00	72.84	74.37
uWaveGestureLibrary_Y	66.16	63.40	70.30	NA	69.82	**71.13**	64.43	67.42
uWaveGestureLibrary_Z	64.96	65.83	68.51	**69.65**	68.06	68.90	62.90	64.91
wafer	99.55	97.99	99.30	99.56	99.43	**99.59**	99.25	99.51
yoga	83.03	83.67	83.97	**85.30**	84.70	83.57	46.43	46.43

Table 7: **Nearest neighbor classification accuracy with $k = 3$.**

Dataset name	Euc.	DTW	SDTW	SDTW div	Sharp	Sharp div	Mean cost	Mean-cost div
50words	61.98	66.37	80.22	**80.66**	77.80	78.90	59.34	66.81
Adiac	55.24	57.29	56.78	**69.05**	54.99	66.50	26.34	49.10
ArrowHead	79.43	70.86	80.57	79.43	78.86	82.86	72.57	**84.57**
Beef	**60.00**	56.67	53.33	56.67	56.67	56.67	20.00	20.00
BeetleFly	65.00	70.00	50.00	65.00	**75.00**	**75.00**	50.00	50.00
BirdChicken	45.00	**60.00**	**60.00**	**60.00**	**60.00**	**60.00**	50.00	50.00
CBF	83.78	**99.67**	**99.67**	**99.67**	**99.67**	**99.67**	82.56	89.78
Car	**66.67**	55.00	61.67	**66.67**	56.67	56.67	23.33	23.33
ChlorineConcentration	56.59	**56.69**	56.12	56.54	**56.69**	**56.69**	38.44	51.54
CinC_ECG_torso	85.22	49.78	**86.67**	**86.67**	85.87	85.58	24.78	24.78
Coffee	**100.00**	92.86	92.86	92.86	92.86	92.86	53.57	92.86
Computers	62.00	**71.20**	**71.20**	**71.20**	**71.20**	**71.20**	50.00	50.00
Cricket_X	51.79	74.36	75.38	**77.44**	72.56	75.13	42.05	55.38
Cricket_Y	50.51	70.51	71.03	**76.41**	71.03	73.33	44.62	56.92
Cricket_Z	54.62	75.38	77.95	78.72	76.92	**78.97**	42.31	59.23
DiatomSizeReduction	89.22	**92.81**	89.22	89.87	89.87	89.87	87.58	89.54
DistalPhalanxOutlineAgeGroup	78.50	83.50	**83.75**	79.75	83.25	79.25	59.25	79.25
DistalPhalanxOutlineCorrect	75.83	79.83	79.33	79.83	79.83	**80.67**	36.67	74.33
DistalPhalanxTW	75.75	73.00	72.75	75.00	75.00	**76.75**	53.75	72.75
ECG200	**90.00**	80.00	88.00	89.00	88.00	89.00	86.00	88.00
ECG5000	93.49	93.98	94.00	94.16	93.98	**94.20**	93.44	93.47
ECGFiveDays	73.98	62.02	67.25	82.00	66.32	**82.81**	52.50	80.02
Earthquakes	74.22	78.88	78.88	78.88	78.88	78.88	**81.99**	**81.99**
ElectricDevices	56.40	**61.08**	NA	NA	NA	NA	25.77	60.42
FISH	75.43	79.43	90.29	90.29	90.29	**91.43**	12.57	12.57
FaceAll	67.22	80.77	79.94	83.37	75.09	**84.97**	28.46	80.53
FaceFour	65.91	68.18	68.18	72.73	59.09	**77.27**	46.59	69.32
FacesUCR	67.76	88.63	90.44	**93.90**	91.32	93.41	47.17	71.32
FordA	**67.15**	57.46	NA	NA	NA	NA	51.26	51.26
FordB	58.33	61.83	**61.94**	NA	61.83	NA	51.16	51.16
Gun_Point	87.33	88.67	97.33	**98.00**	**98.00**	**98.00**	84.67	84.67
Ham	59.05	51.43	52.38	**62.86**	57.14	61.90	51.43	51.43
HandOutlines	**84.90**	81.00	NA	NA	NA	NA	63.80	63.80
Haptics	38.64	42.86	41.23	41.56	37.01	**43.51**	21.75	21.75
Herring	56.25	48.44	64.06	60.94	62.50	**65.62**	59.38	59.38
InlineSkate	23.82	35.64	37.45	**37.64**	35.82	35.45	15.64	15.64
InsectWingbeatSound	**59.24**	36.21	56.67	58.18	57.22	58.33	57.07	58.28
ItalyPowerDemand	**95.63**	94.56	94.95	95.14	94.56	95.04	89.60	94.95
LargeKitchenAppliances	45.60	**80.00**	**80.00**	77.60	**80.00**	77.07	33.33	33.33
Lighting2	77.05	86.89	**91.80**	90.16	83.61	85.25	45.90	45.90
Lighting7	60.27	71.23	79.45	**82.19**	78.08	**82.19**	57.53	71.23
MALLAT	91.98	92.84	92.54	**92.88**	92.15	92.75	12.45	12.45
Meat	**93.33**	**93.33**	**93.33**	**93.33**	**93.33**	91.67	33.33	33.33
MedicalImages	67.76	70.92	72.11	73.42	72.76	**74.61**	57.24	69.21
MiddlePhalanxOutlineAgeGroup	73.50	**76.00**	**76.00**	74.50	**76.00**	**76.00**	67.75	74.50
MiddlePhalanxOutlineCorrect	77.17	72.17	74.50	**77.67**	73.67	76.00	35.50	75.33
MiddlePhalanxTW	58.40	61.15	60.65	61.15	61.65	**62.16**	51.88	58.65
MoteStrain	86.18	81.39	88.18	87.46	**89.54**	87.86	85.14	83.87
NonInvasiveFatalECG_Thorax1	**82.54**	78.63	NA	NA	NA	NA	2.54	2.54
NonInvasiveFatalECG_Thorax2	**88.40**	86.31	NA	NA	NA	NA	2.54	2.54
OSULeaf	50.41	57.44	59.50	61.98	64.88	**65.29**	19.01	19.01
OliveOil	**90.00**	86.67	86.67	86.67	86.67	86.67	40.00	40.00
PhalangesOutlinesCorrect	77.97	75.41	76.57	**79.37**	76.57	79.14	42.07	73.66
Phoneme	10.34	23.95	21.99	23.58	23.10	**25.05**	7.07	7.07
Plane	96.19	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	84.76	96.19
ProximalPhalanxOutlineAgeGroup	**81.95**	80.98	81.46	80.98	**81.95**	**81.95**	48.78	80.49
ProximalPhalanxOutlineCorrect	84.88	83.16	81.79	**85.57**	78.01	84.19	31.62	74.91
ProximalPhalanxTW	77.00	**79.00**	78.50	77.50	77.25	78.75	45.50	78.00
RefrigerationDevices	39.20	46.40	46.13	45.87	**46.67**	46.13	33.33	33.33
ScreenType	38.40	39.20	**42.13**	36.53	39.20	37.07	33.33	33.33
ShapeletSim	52.78	62.78	62.78	80.00	68.33	**81.67**	50.00	50.00
ShapesAll	69.00	71.00	77.33	**77.67**	75.67	NA	1.67	1.67
SmallKitchenAppliances	36.53	67.47	**70.67**	**70.67**	67.73	67.20	33.33	33.33
SonyAIBORobotSurface	57.40	61.73	61.73	61.73	61.73	61.73	43.59	**67.22**
SonyAIBORobotSurfaceII	79.85	80.27	77.65	79.12	79.01	**80.90**	76.50	80.06
StarLightCurves	**84.82**	NA	NA	NA	NA	NA	NA	NA
Strawberry	**92.33**	91.84	91.68	92.01	90.05	91.03	78.96	90.38
SwedishLeaf	71.84	77.92	80.48	86.56	78.88	**87.36**	47.84	77.44
Symbols	85.03	92.86	**96.18**	**96.18**	95.98	96.08	81.91	86.13
ToeSegmentation1	60.53	75.44	**82.02**	77.63	75.88	78.51	57.46	63.60
ToeSegmentation2	82.31	81.54	89.23	89.23	91.54	**93.08**	82.31	86.15
Trace	65.00	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	47.00	64.00
TwoLeadECG	63.48	85.16	**85.34**	63.48	82.44	63.74	55.66	63.21
Two_Patterns	85.95	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	90.72	94.20
UWaveGestureLibraryAll	**94.39**	89.53	NA	NA	NA	NA	12.62	12.62
Wine	55.56	57.41	**62.96**	**62.96**	51.85	61.11	50.00	61.11
WordsSynonyms	56.74	59.56	**72.41**	69.59	70.85	72.10	54.23	59.56
Worms	36.46	**42.54**	**42.54**	**42.54**	**42.54**	**42.54**	13.81	13.81
WormsTwoClass	59.12	64.09	**70.17**	**70.17**	65.19	65.19	58.01	58.01
synthetic_control	91.00	98.33	98.33	98.33	98.33	98.33	74.67	**98.67**
uWaveGestureLibrary_X	73.03	73.73	78.00	**78.31**	76.97	77.41	71.94	73.84
uWaveGestureLibrary_Y	66.67	63.18	70.63	**71.36**	70.18	NA	65.47	67.17
uWaveGestureLibrary_Z	65.75	66.78	68.37	**69.43**	67.87	68.87	64.38	66.50
wafer	99.38	97.52	99.06	99.42	99.06	**99.45**	99.06	**99.45**
yoga	79.23	82.17	**82.53**	82.33	82.23	82.33	46.43	46.43

Table 8: **Nearest neighbor classification accuracy with $k = 5$.**

Dataset name	Euc.	DTW	SDTW	SDTW div	Sharp	Sharp div	Mean cost	Mean-cost div
50words	61.98	66.15	77.80	**79.12**	75.60	77.80	57.80	65.93
Adiac	52.17	53.20	59.34	**63.68**	55.75	61.64	25.06	46.55
ArrowHead	66.86	**68.57**	62.86	64.57	63.43	66.86	62.29	**68.57**
Beef	**50.00**	43.33	46.67	43.33	43.33	43.33	20.00	20.00
BeetleFly	60.00	70.00	60.00	65.00	70.00	**80.00**	50.00	50.00
BirdChicken	55.00	65.00	70.00	**75.00**	65.00	60.00	50.00	50.00
CBF	76.67	**98.22**	**98.22**	**98.22**	**98.22**	**98.22**	75.56	88.78
Car	63.33	50.00	**66.67**	**66.67**	63.33	**66.67**	31.67	31.67
ChlorineConcentration	**54.87**	54.82	**54.87**	**54.87**	54.82	54.66	44.32	51.46
CinC_ECG_torso	77.39	42.61	80.14	80.22	82.46	**83.26**	24.78	24.78
Coffee	**96.43**	**96.43**	**96.43**	**96.43**	**96.43**	**96.43**	60.71	**96.43**
Computers	60.40	68.80	**69.60**	68.40	**69.60**	68.00	50.00	50.00
Cricket_X	48.21	72.56	71.79	71.79	71.54	**72.82**	40.77	57.18
Cricket_Y	50.26	68.46	68.46	71.79	68.72	**73.85**	42.82	55.64
Cricket_Z	49.49	76.67	77.18	79.49	76.15	**80.26**	39.23	58.46
DiatomSizeReduction	86.93	70.92	85.62	85.62	80.07	78.43	**87.25**	86.93
DistalPhalanxOutlineAgeGroup	79.75	**83.50**	**83.50**	**83.50**	**83.50**	82.75	60.50	80.00
DistalPhalanxOutlineCorrect	76.33	78.17	79.17	78.17	78.17	**79.67**	35.83	74.83
DistalPhalanxTW	76.75	76.25	78.25	78.00	76.50	**79.00**	53.25	73.50
ECG200	**90.00**	79.00	86.00	87.00	87.00	88.00	85.00	89.00
ECG5000	93.91	93.84	**94.33**	93.84	94.24	93.84	93.89	93.87
ECGFiveDays	61.21	60.16	75.38	77.82	68.99	**77.93**	51.34	77.00
Earthquakes	78.57	79.19	79.19	79.19	79.19	79.19	**81.99**	**81.99**
ElectricDevices	58.38	**61.03**	NA	NA	NA	NA	27.19	60.80
FISH	72.00	73.14	89.14	90.86	90.86	**91.43**	16.57	16.57
FaceAll	64.62	81.01	71.66	**85.03**	74.44	80.89	30.59	79.59
FaceFour	52.27	**68.18**	**68.18**	**68.18**	44.32	67.05	42.05	50.00
FacesUCR	62.20	86.20	88.20	**92.78**	89.61	91.76	45.07	67.22
FordA	**68.62**	58.71	NA	NA	NA	NA	51.26	51.26
FordB	58.33	63.97	**64.11**	NA	63.28	NA	48.84	48.84
Gun_Point	80.00	82.67	92.67	**94.67**	92.00	92.67	81.33	80.67
Ham	62.86	53.33	60.95	63.81	62.86	**64.76**	51.43	51.43
HandOutlines	**85.10**	81.40	NA	NA	NA	NA	63.80	63.80
Haptics	41.56	41.23	**51.30**	50.97	47.73	49.03	19.16	19.16
Herring	51.56	54.69	54.69	56.25	**59.38**	56.25	**59.38**	**59.38**
InlineSkate	22.55	33.27	**37.64**	33.82	33.45	33.45	15.45	15.45
InsectWingbeatSound	**59.90**	35.45	57.27	59.55	56.67	59.80	56.01	59.65
ItalyPowerDemand	**95.24**	94.36	95.04	94.46	95.04	94.46	88.34	94.46
LargeKitchenAppliances	45.60	78.67	**78.93**	78.67	78.67	75.47	33.33	33.33
Lighting2	72.13	81.97	**85.25**	83.61	**85.25**	**85.25**	54.10	54.10
Lighting7	57.53	75.34	76.71	75.34	**79.45**	75.34	49.32	63.01
MALLAT	78.89	**82.77**	81.32	81.75	80.68	81.49	12.54	12.54
Meat	91.67	**93.33**	91.67	90.00	90.00	**93.33**	33.33	33.33
MedicalImages	66.05	69.74	**71.45**	**71.45**	71.18	71.32	54.74	69.47
MiddlePhalanxOutlineAgeGroup	76.50	76.75	76.75	75.50	76.25	**77.25**	68.00	74.50
MiddlePhalanxOutlineCorrect	76.00	74.50	74.33	77.17	74.50	**77.50**	35.67	74.67
MiddlePhalanxTW	62.16	62.91	60.15	61.15	**63.66**	60.65	51.38	59.90
MoteStrain	85.14	82.43	87.54	85.62	**88.82**	88.18	83.95	82.91
NonInvasiveFatalECG_Thorax1	**82.60**	78.78	NA	NA	NA	NA	2.90	2.90
NonInvasiveFatalECG_Thorax2	**88.65**	85.24	NA	NA	NA	NA	2.90	2.90
OSULeaf	47.11	54.55	57.44	58.26	**64.46**	62.40	18.18	18.18
OliveOil	**83.33**	73.33	80.00	80.00	80.00	76.67	40.00	40.00
PhalangesOutlinesCorrect	77.86	75.64	78.55	**79.60**	77.16	79.37	42.89	75.87
Phoneme	12.03	24.95	25.95	25.58	24.74	**26.85**	7.07	7.07
Plane	96.19	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	83.81	96.19
ProximalPhalanxOutlineAgeGroup	82.44	82.44	83.41	83.41	82.93	**85.85**	48.78	81.46
ProximalPhalanxOutlineCorrect	84.19	80.76	84.54	**86.94**	80.07	86.25	31.62	79.38
ProximalPhalanxTW	79.75	79.50	79.00	78.75	79.25	79.25	45.00	**80.25**
RefrigerationDevices	38.93	**48.27**	46.40	**48.27**	47.47	47.47	33.33	33.33
ScreenType	41.60	**42.67**	42.13	40.53	**42.67**	39.20	33.33	33.33
ShapeletSim	54.44	63.89	63.89	72.22	63.89	**76.67**	50.00	50.00
ShapesAll	65.83	68.17	72.00	72.83	72.83	**73.33**	1.67	1.67
SmallKitchenAppliances	36.53	68.00	68.00	67.73	**68.80**	68.27	33.33	33.33
SonyAIBORobotSurface	46.92	52.25	52.25	52.25	52.25	52.25	42.93	**56.57**
SonyAIBORobotSurfaceII	77.12	77.65	74.29	76.92	77.33	77.75	75.13	**79.33**
StarLightCurves	**84.51**	NA	NA	NA	NA	NA	57.72	NA
Strawberry	**92.33**	91.68	87.77	90.86	91.19	90.54	79.45	89.40
SwedishLeaf	71.84	78.72	78.24	85.12	77.76	**85.44**	48.48	78.88
Symbols	73.37	90.45	93.47	77.39	**94.37**	77.89	71.36	76.58
ToeSegmentation1	61.40	71.49	72.81	**76.32**	73.25	72.81	58.33	61.40
ToeSegmentation2	84.62	83.08	83.85	84.62	85.38	84.62	84.62	**86.92**
Trace	54.00	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	49.00	53.00
TwoLeadECG	59.70	81.39	74.54	**81.56**	72.61	72.87	55.14	60.76
Two_Patterns	82.50	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**	87.62	91.52
UWaveGestureLibraryAll	**93.89**	89.06	NA	NA	NA	NA	12.67	12.67
Wine	53.70	48.15	59.26	51.85	**66.67**	59.26	50.00	53.70
WordsSynonyms	54.70	55.33	67.40	64.89	66.93	**68.03**	51.88	58.62
Worms	38.12	44.20	49.17	**50.28**	46.96	48.62	13.81	13.81
WormsTwoClass	60.22	66.85	**70.72**	**70.72**	67.40	67.96	58.01	58.01
synthetic_control	87.00	97.33	97.33	97.33	97.33	97.33	76.00	**98.67**
uWaveGestureLibrary_X	72.89	73.73	77.22	**77.69**	76.52	77.05	71.50	73.73
uWaveGestureLibrary_Y	66.36	64.10	70.46	**71.08**	69.74	70.71	65.75	67.59
uWaveGestureLibrary_Z	65.97	67.11	68.79	68.90	68.57	**69.29**	64.82	66.22
wafer	**99.17**	97.13	98.91	99.01	99.01	99.08	98.78	99.08
yoga	75.63	78.53	78.40	**78.70**	78.27	78.57	46.43	46.43

Table 9: **Nearest centroid classification accuracy.**

Dataset name	Euc.	DTW	SDTW	SDTW div	Sharp	Sharp div	Mean cost	Mean-cost div
50words	51.65	59.78	76.26	**78.02**	69.45	76.70	50.33	51.21
Adiac	54.99	47.06	67.52	**68.54**	66.75	67.26	44.25	46.55
ArrowHead	**61.14**	50.86	51.43	57.71	49.71	**61.14**	58.86	59.43
Beef	**53.33**	43.33	46.67	36.67	43.33	46.67	20.00	20.00
BeetleFly	**85.00**	80.00	70.00	70.00	80.00	70.00	50.00	50.00
BirdChicken	55.00	60.00	**65.00**	60.00	60.00	60.00	50.00	50.00
CBF	76.33	96.89	**97.11**	**97.11**	97.00	97.00	73.00	74.44
Car	61.67	61.67	70.00	73.33	73.33	**75.00**	23.33	23.33
ChlorineConcentration	33.31	32.45	**35.23**	32.19	31.98	33.41	34.82	34.95
CinC_ECG_torso	38.55	40.29	**71.88**	70.36	59.49	64.42	25.36	25.36
Coffee	**96.43**	**96.43**	**96.43**	**96.43**	**96.43**	**96.43**	89.29	89.29
Computers	41.60	**63.20**	51.60	56.80	62.80	**63.20**	50.00	50.00
Cricket_X	23.85	57.69	56.92	56.67	58.46	**58.97**	25.64	26.15
Cricket_Y	34.87	52.56	**55.64**	54.87	53.59	55.13	33.59	33.59
Cricket_Z	30.51	60.00	61.03	60.00	58.21	**62.31**	30.26	30.26
DiatomSizeReduction	95.75	95.10	**96.73**	96.41	96.08	95.42	94.44	95.42
DistalPhalanxOutlineAgeGroup	81.75	84.00	84.50	84.75	84.50	**85.00**	80.25	81.25
DistalPhalanxOutlineCorrect	47.17	**48.17**	48.00	47.33	47.00	47.17	**48.17**	47.17
DistalPhalanxTW	74.75	**75.75**	74.50	74.50	74.50	73.00	73.00	72.75
ECG200	**75.00**	**75.00**	72.00	73.00	69.00	73.00	74.00	74.00
ECG5000	86.04	84.53	**86.73**	85.98	86.02	86.09	81.44	83.64
ECGFiveDays	68.99	65.27	80.60	83.39	80.95	**85.60**	79.56	80.26
Earthquakes	75.47	58.07	**82.30**	65.22	71.12	72.98	81.99	81.99
ElectricDevices	48.27	53.60	57.07	**61.57**	53.61	51.28	50.55	50.37
FISH	56.00	65.71	81.14	**84.00**	81.14	82.86	13.71	13.71
FaceAll	49.17	80.71	81.60	88.58	85.98	**89.17**	58.88	64.56
FaceFour	84.09	82.95	86.36	89.77	88.64	**90.91**	78.41	77.27
FacesUCR	53.95	79.22	88.98	91.07	90.78	**91.85**	57.37	59.46
FordA	49.60	55.57	55.62	52.43	54.96	**56.32**	51.26	51.26
FordB	49.97	**60.70**	47.58	55.94	58.33	54.81	51.16	51.16
Gun_Point	75.33	68.00	82.00	81.33	**92.00**	86.00	68.67	71.33
Ham	76.19	73.33	71.43	75.24	**79.05**	72.38	48.57	48.57
HandOutlines	81.80	79.20	**82.40**	NA	NA	NA	36.20	36.20
Haptics	39.29	35.71	46.10	46.10	**48.38**	47.73	19.48	19.48
Herring	54.69	60.94	**64.06**	**64.06**	59.38	62.50	59.38	59.38
InlineSkate	19.27	22.73	23.45	**26.36**	22.73	21.45	9.64	9.64
InsectWingbeatSound	**60.10**	29.80	58.18	58.64	58.43	58.79	58.43	58.38
ItalyPowerDemand	**91.84**	74.15	88.14	90.48	85.62	87.37	71.62	84.35
LargeKitchenAppliances	44.00	71.47	72.00	73.60	**74.67**	72.53	33.33	33.33
Lighting2	68.85	62.30	67.21	**72.13**	65.57	62.30	45.90	45.90
Lighting7	58.90	72.60	78.08	**83.56**	56.16	58.90	61.64	63.01
MALLAT	**96.67**	94.93	95.74	94.84	94.80	94.88	12.54	12.54
Meat	**93.33**	**93.33**	85.00	85.00	90.00	85.00	33.33	33.33
MedicalImages	38.55	44.21	40.39	40.92	**45.53**	45.00	32.11	33.55
MiddlePhalanxOutlineAgeGroup	73.25	72.50	72.75	72.75	72.75	**75.25**	73.75	73.25
MiddlePhalanxOutlineCorrect	**55.17**	48.50	52.17	52.83	51.83	52.83	51.83	52.83
MiddlePhalanxTW	59.15	56.64	58.15	58.15	58.90	58.65	**59.40**	**59.40**
MoteStrain	86.10	82.43	**90.42**	90.18	82.27	88.82	82.99	83.87
NonInvasiveFatalECG_Thorax1	76.95	70.13	81.63	**82.29**	81.12	NA	2.44	2.44
NonInvasiveFatalECG_Thorax2	80.20	76.28	87.23	87.68	**87.74**	NA	2.44	2.44
OSULeaf	35.95	45.87	**52.07**	51.24	50.00	50.41	13.22	13.22
OliveOil	**86.67**	76.67	83.33	**86.67**	83.33	83.33	16.67	16.67
PhalangesOutlinesCorrect	62.59	63.64	63.75	**64.45**	**64.45**	63.99	61.42	62.47
Phoneme	7.86	17.67	20.15	20.57	19.83	**20.99**	2.00	2.00
Plane	96.19	99.05	99.05	99.05	**100.00**	**100.00**	95.24	96.19
ProximalPhalanxOutlineAgeGroup	81.95	82.93	**84.39**	**84.39**	**84.39**	83.90	81.46	80.49
ProximalPhalanxOutlineCorrect	64.60	**64.95**	**64.95**	**64.95**	**64.95**	**64.95**	64.26	64.60
ProximalPhalanxTW	70.75	73.50	81.25	**81.50**	80.00	80.75	69.75	68.50
RefrigerationDevices	35.47	57.87	58.13	55.20	**61.60**	58.13	33.33	33.33
ScreenType	**44.27**	38.13	37.33	40.00	37.60	40.80	33.33	33.33
ShapeletSim	50.00	61.67	**73.33**	72.78	57.22	68.89	50.00	50.00
ShapesAll	51.33	62.17	65.50	**68.67**	64.50	66.83	1.67	1.67
SmallKitchenAppliances	41.87	64.53	68.00	**68.80**	65.87	64.53	33.33	33.33
SonyAIBORobotSurface	81.20	**82.86**	82.70	**82.86**	80.37	81.53	80.70	78.70
SonyAIBORobotSurfaceII	79.33	76.60	79.85	76.50	**80.27**	78.91	77.12	76.92
StarLightCurves	76.17	82.93	**83.57**	83.35	81.64	NA	14.29	14.29
Strawberry	66.88	61.17	65.58	68.84	67.54	**72.43**	65.74	65.58
SwedishLeaf	70.24	70.40	79.36	**81.12**	77.12	80.00	71.36	71.52
Symbols	86.43	95.78	95.08	95.58	95.58	**96.08**	88.74	87.84
ToeSegmentation1	57.46	62.72	73.25	71.05	69.30	**74.56**	52.63	54.39
ToeSegmentation2	54.62	**86.92**	86.15	85.38	80.77	84.62	55.38	54.62
Trace	58.00	98.00	98.00	97.00	**99.00**	**99.00**	56.00	57.00
TwoLeadECG	55.49	76.21	78.05	83.06	78.49	**89.38**	57.33	57.16
Two_Patterns	46.48	98.40	**98.65**	98.18	98.42	98.55	56.30	50.75
UWaveGestureLibraryAll	84.95	83.45	89.31	**90.90**	90.09	NA	12.20	12.20
Wine	55.56	53.70	**57.41**	55.56	**57.41**	55.56	55.56	55.56
WordsSynonyms	27.12	34.33	**52.19**	51.72	49.84	50.78	26.33	26.49
Worms	21.55	40.33	43.65	**44.75**	42.54	42.54	41.99	41.99
WormsTwoClass	54.14	62.98	67.96	**70.72**	65.19	56.91	41.99	41.99
synthetic_control	91.67	98.33	98.00	**98.67**	98.33	98.00	90.33	93.00
uWaveGestureLibrary_X	63.12	**69.96**	67.98	69.71	68.40	69.40	63.34	63.18
uWaveGestureLibrary_Y	54.83	53.24	61.25	**62.09**	60.61	60.72	54.30	54.69
uWaveGestureLibrary_Z	53.74	60.58	63.34	**64.52**	62.53	63.04	53.38	53.69
wafer	65.44	31.86	68.82	68.93	67.86	**85.92**	64.93	65.07
yoga	49.70	59.97	57.10	**61.70**	54.50	56.23	46.43	46.43