Mini-Project (Machine Learning for Time Series) - Study of Differentiable Divergences Between Time Series

Benmansour Mohammed benmansour.mohammed2002@gmail.com
Abdessamad El Kabid abdessamad.elkabid@ip-paris.fr

January 4, 2026

1 Introduction and contributions

Comparing time series of variable length is a central problem in time series analysis, with direct applications to classification, clustering, and averaging. Dynamic Time Warping (DTW) [4, 5] addresses temporal misalignment by searching for a minimum-cost monotone alignment between samples, but the resulting objective is non-differentiable and therefore not ideal as a loss in gradient-based pipelines.

Soft-DTW [2] replaces the hard minimum over alignments by a differentiable soft-min controlled by a regularization parameter $\gamma > 0$, while preserving the $\mathcal{O}(mn)$ dynamic-programming structure. The paper *Differentiable Divergences Between Time Series* [1] highlights a key limitation of soft-DTW with the standard squared Euclidean cost: it is not a divergence (it takes negative values and it is not minimized at $X = Y$). In practice, minimizing soft-DTW with respect to X for a fixed target Y tends to move the solution away from Y (a denoising effect). To address this issue, the paper proposes debiased divergences inspired of Sinkhorn divergences.

In this project, we studied the properties of the Soft-DTW divergence and its variants both theoretically and in practical applications for time series classification and prediction tasks. First, we analyze the paper experiments with variations in hyperparameters and datasets, then we apply the methods in classification and prediction tasks.

Contributions. All project orientation choices were made jointly by the two authors. Both authors contributed equally to the writing of the report. Abdessamad El Kabid analyzed the paper's theory, produced an explanation of the main results, set up the code and produced the experiments on barycenters, interpolation, and alignment. Benmansour Mohammed implemented the large-scale comparison across 81 datasets and consolidated the experimental pipeline. We reused the original implementation by [1][1] and wrote additional experiment and evaluation code in `tests/final_deliverable.ipynb`[2].

2 Method

Let $X = (x_1, \dots, x_m) \in \mathbb{R}^{m \times d}$ and $Y = (y_1, \dots, y_n) \in \mathbb{R}^{n \times d}$. We use the standard squared Euclidean ground cost $C(X, Y) \in \mathbb{R}^{m \times n}$, $\quad C_{ij} = \frac{1}{2}\|x_i - y_j\|_2^2$.

[1] https://github.com/google-research/soft-dtw-divergences
[2] https://github.com/AI-ELka/Soft_DTW_div_project

Dynamic Time Warping (DTW) minimizes the path cost $\langle A, C \rangle$ over all monotone alignment matrices $A \in \mathcal{A}(m, n)$. Soft-DTW replaces the hard minimum by a soft minimum, controlled by a regularization parameter $\gamma > 0$:

$$\text{sdtw}_\gamma(C) = -\gamma \log \sum_{A \in \mathcal{A}(m,n)} \exp\Big(-\tfrac{1}{\gamma} \langle A, C \rangle \Big).$$

It is differentiable and computable by dynamic programming in $\mathcal{O}(mn)$; its gradient with respect to the cost matrix is an expected alignment matrix $E_\gamma(C)$ under the induced Gibbs distribution over paths, and $\text{sdtw}_\gamma(C) \xrightarrow{\gamma \to 0} \text{dtw(C)}$.

However, the paper shows that, with the squared Euclidean cost, soft-DTW 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 (figure 5). The proposed correction (inspired from optimal transport theory) is **Soft-DTW divergence**:

$$D_\gamma(X, Y) = \text{sdtw}_\gamma(C(X, Y)) - \tfrac{1}{2}\text{sdtw}_\gamma(C(X, X)) - \tfrac{1}{2}\text{sdtw}_\gamma(C(Y, Y)).$$

By construction $D_\gamma(X, X) = 0$. Under specific costs (and empirically for squared Euclidean), this restores the expected behavior of a divergence.

Sharp divergence. The sharp soft-DTW value is defined as $\text{sharp}_\gamma(C) = \langle E_\gamma(C), C \rangle$, which removes the entropy term from the variational form of soft-DTW (Proposition 1 in [1]). The sharp divergence S_γ uses the same Sinkhorn divergence correction as above. In practice it is sharper (removes the entropic regularization term) but its gradient computation (in Appendix A) is more expensive, since it requires a Hessian-vector product in addition to the forward/backward dynamic programs.

Table 1: Properties of time-series losses (from [1])

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

3 Data

To perform our experiments, we use multiple datasets from UCR Archive [3], accessed through the `aeon` package.

Dataset selection. For the pair comparison benchmark in the classification task, we focus on univariate datasets. Since our evaluation recomputes one barycenter per class and then classifies the test set, a few datasets can dominate runtime. To keep the experiment feasible, we exclude datasets that repeatedly exceed our compute budget and also apply a runtime safeguard: if the soft-DTW pipeline exceeds a fixed time threshold on a dataset, we skip it. After these filters, the benchmark covers 81 datasets.

Datasets used for figures. For qualitative plots, we use small subsets from `ItalyPowerDemand` 4 (alignment matrices, bias demonstration, barycenter comparisons) and `BirdChicken` 2 (interpolation).

4 Results

We conducted three main experiments to validate the theoretical properties and practical performance of the soft-DTW divergence and its variants. Most figures can be found in the appendix.

4.1 Bias demonstration.

Figures 5 and 6 illustrate the bias highlighted in the paper. We fix a target series Y and optimize over X, initialized at $X = Y$. When minimizing $\text{sdtw}_\gamma(X, Y)$, the optimizer moves away from the target and produces a slightly smoother time series; this denoising effect becomes more pronounced as γ increases. In contrast, when minimizing the soft-DTW divergence $D_\gamma(X, Y)$, the optimum remains at the target series, since $D_\gamma(Y, Y) = 0$ by construction.

4.2 Alignment entropy visualization.

Figure 1 visualizes the expected alignment matrix $E_\gamma(C) = \nabla_C \text{sdtw}_\gamma(C)$ between two time series for multiple values of γ. For small γ, the soft-min is close to the hard minimum, hence $E_\gamma(C)$ concentrates most of its mass on a narrow band that resembles a single DTW path. As γ increases, the probability mass spreads over many nearby paths, yielding a thicker and more diffuse band.

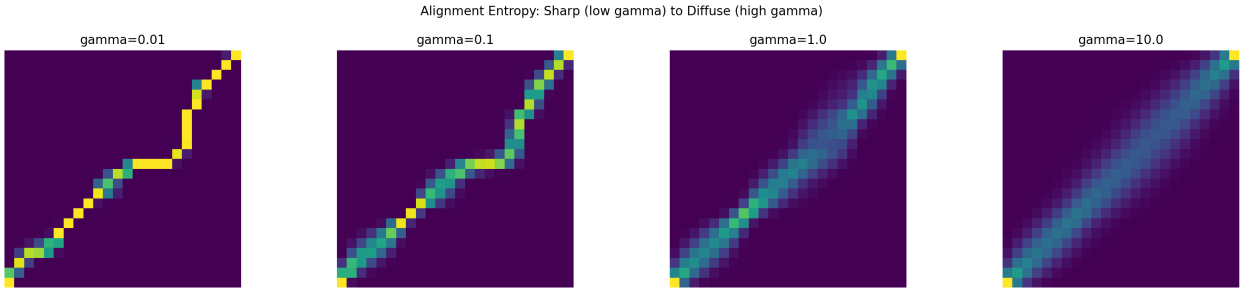

Figure 1: Expected alignment matrix for increasing γ for samples of `ItalyPowerDemand`. The matrix goes from sharp (γ small) to diffuse (larger γ).

4.3 Time Series Barycenter and Averaging

We replicated the paper's barycenter averaging experiments on multiple UCR datasets, computing the Fréchet mean of 10 randomly selected time series for different measures and divergences. Given series $Y_1, \ldots, Y_k$ and weights w_i, we compute barycenters by minimizing $\sum_i w_i \mathcal{L}(X, Y_i)$ with L-BFGS, where $\mathcal{L}$ is one of the above losses (soft-DTW or a divergence). In our experiments we use uniform weights ($w_i = 1$). Figure 7 shows results on the *ItalyPowerDemand* dataset with $\gamma = 1.0$, while Figure 8 presents results on the same dataset with $\gamma = 5.0$.

DTW produces barycenters with visible artifacts, likely due to convergence to local minima in the non-smooth optimization landscape. Soft-DTW and its variants produce significantly smoother barycenters that better represent the average temporal pattern. Notably, the sharp variants (sharp

soft-DTW and sharp divergence) produce barycenters with more pronounced peaks compared to their soft counterparts, as predicted by the theory.

The mean-cost divergence, while theoretically interesting as the $\gamma \to \infty$ limit, produces less satisfactory results in practice, confirming that considering all alignments uniformly is suboptimal.

We then focused on the impact of γ on the barycenter. Figures 9 and 10 compare barycenters computed on the same subset for several values of γ, using respectively soft-DTW and the soft-DTW divergence. We notice again, the sharp variant gives more defined peaks, but higher γ values, gives smoother time series.

4.4 Time Series Interpolation

Figure 2 reproduces the paper's interpolation experiment. We pick two training time series Y_1 and Y_2 (here from `BirdChicken`) and compute a family of weighted barycenters for weights $(\pi, 1-\pi)$ with $\pi \in \{0.25, 0.5, 0.75\}$. We repeat this procedure for all losses considered in the paper (including Euclidean averaging, soft-DTW, debiased divergences, sharp variants, and mean-cost losses) and overlay the resulting curves.

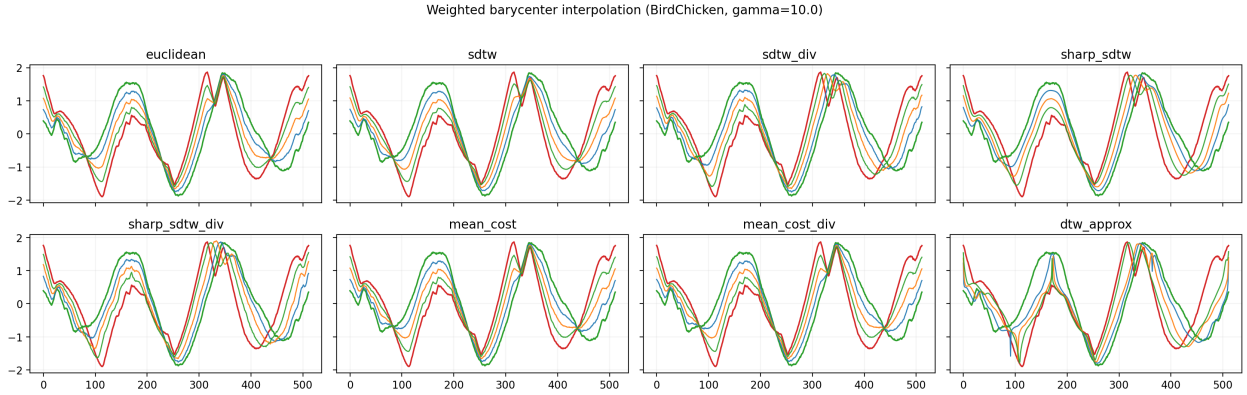

Figure 2: Interpolation via weighted barycenters on `BirdChicken` for all methods (weights $(\pi, 1-\pi)$ for $\pi \in \{0.25, 0.5, 0.75\}$).

4.5 Barycenters and classification.

For classification, we use a nearest-centroid classifier: we compute one barycenter per class on the training set and assign each test time series to the nearest centroid using the same discrepancy. In addition to D_γ and S_γ, we include two practical baselines in our comparisons: the mean-cost divergence from the paper and a DTW-like approximation obtained by running soft-DTW with a very small γ.

Classification task We evaluate a nearest-centroid classifier across a large subset of the UCR archive. For each dataset and each loss, we learn one prototype per class by computing a class barycenter on the training split, then classify each test series by the nearest prototype under the same discrepancy. In those experiments, alignment-based methods are run at a fixed regularization $\gamma = 1$. Table 2 reports mean test accuracy across these datasets. Table 3 follows the paper's robustness criterion: method A is counted as competitive with method B on a dataset if $\mathrm{acc}(A) \geq 0.99\,\mathrm{acc}(B)$, and entry (A,B) reports the corresponding percentage of datasets.

Runtime comparison. In addition to accuracy, we report computational cost in Table 4. The recorded runtime is the end-to-end wall-clock time of the classification pipeline for a given dataset and method. Since Euclidean computations are very fast in this setting, the normalized table (runtime divided by the Euclidean runtime) should be read as an order-of-magnitude indicator.

Accuracy vs. γ for divergences. To study sensitivity to the soft-min temperature, we repeat the large-scale evaluation while sweeping γ for the two debiased divergences (soft-DTW divergence and sharp divergence). Figure 3 reports the mean test accuracy across datasets as a function of γ, computed on the common subset of datasets for which all (method, γ) runs are available. Overall, smaller values of γ tend to yield higher accuracy, which is consistent with the fact that soft-DTW approaches DTW as $\gamma \to 0$ and therefore behaves as a sharper alignment-based discrepancy. Across the tested range, the sharp divergence is slightly above the soft-DTW divergence for all γ values. This gap should however be interpreted cautiously, since it is measured only on 81 datasets.

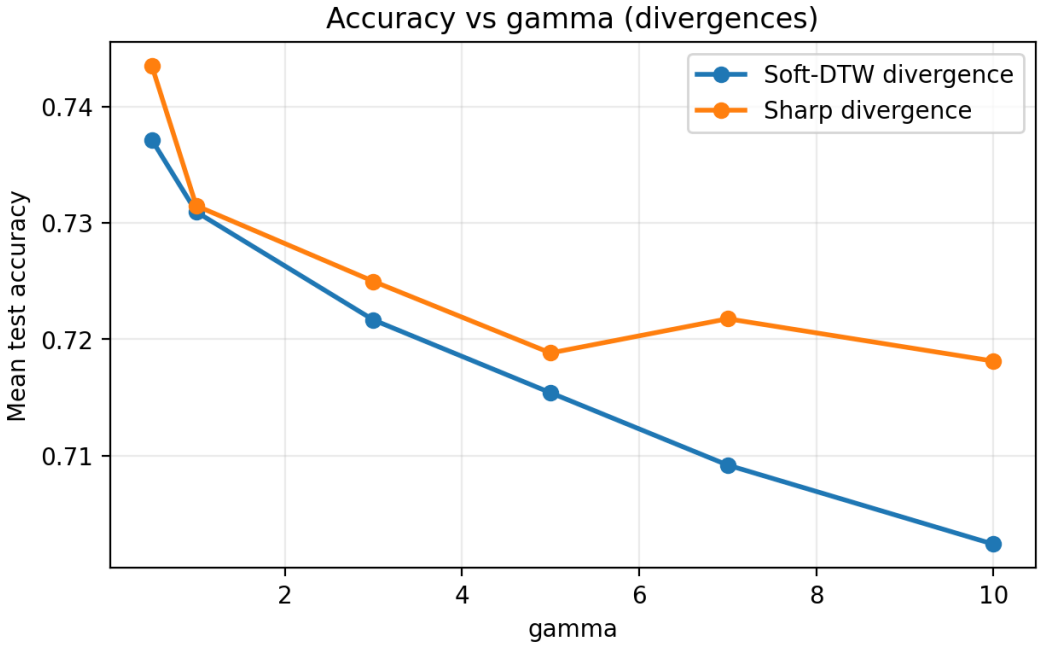

Figure 3: Mean test accuracy as a function of γ for the two debiased divergences (soft-DTW divergence and sharp divergence), computed on the large-scale benchmark datasets.

References

[1] M. Blondel, A. Mensch, and J.-P. Vert. Differentiable divergences between time series. In *International Conference on Artificial Intelligence and Statistics*, 2020.

[2] M. Cuturi and M. Blondel. Soft-dtw: a differentiable loss function for time-series. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 894–903. JMLR.org, 2017.

[3] H. A. Dau, E. Keogh, K. Kamgar, C.-C. M. Yeh, Y. Zhu, S. Gharghabi, C. A. Ratanamahatana, Yanping, B. Hu, N. Begum, A. Bagnall, A. Mueen, G. Batista, and Hexagon-ML. The ucr time

series classification archive, October 2018. https://www.cs.ucr.edu/~eamonn/time_series_data_2018/.

[4] H. Sakoe. Dynamic-programming approach to continuous speech recognition. In *Proceedings of the International Congress on Acoustics*, Budapest, Hungary, 1971.

[5] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(1):43–49, 1978.

A Notation and gradients

A.1 Notation

- $\mathbf{C} \in^{m \times n}$: Cost matrix between two time series
- $\mathbf{A} \in \mathcal{A}(m, n)$: Monotonic alignment matrix (set of all valid alignments)
- $\gamma > 0$: Regularization parameter controlling smoothness
- $\mathbf{E}_\gamma(\mathbf{C}) \in (0, 1]^{m \times n}$: Expected alignment matrix under Gibbs distribution
- $\mathbf{X} \in^{m \times d}$, $\mathbf{Y} \in^{n \times d}$: Time series of lengths m and n
- $\mathbf{J}_\mathbf{X}$: Jacobian operator with respect to $\mathbf{X}$
- $H(\cdot)$: Entropy function

A.2 Soft Dynamic Time Warping (Soft-DTW)

A.2.1 Formula

$$\boxed{\text{sdtw}_\gamma(\mathbf{C}) = -\gamma \log \sum_{\mathbf{A} \in \mathcal{A}(m,n)} \exp\left(-\frac{\langle \mathbf{A}, \mathbf{C} \rangle}{\gamma}\right)} \tag{1}$$

Alternative formulation using soft minimum:

$$\text{sdtw}_\gamma(\mathbf{C}) = \min_{\mathbf{A} \in \mathcal{A}(m,n)} \langle \mathbf{A}, \mathbf{C} \rangle \tag{2}$$

A.2.2 Gradient

$$\boxed{\nabla_\mathbf{C} \, \text{sdtw}_\gamma(\mathbf{C}) = \mathbf{E}_\gamma(\mathbf{C}) \in (0, 1]^{m \times n}} \tag{3}$$

where $\mathbf{E}_\gamma(\mathbf{C})$ is the expected alignment matrix containing marginal probabilities.

A.3 Soft-DTW Divergence

A.3.1 Formula

$$\boxed{D_\gamma^\mathbf{C}(\mathbf{X}, \mathbf{Y}) = \text{sdtw}_\gamma(\mathbf{C}(\mathbf{X}, \mathbf{Y})) - \frac{1}{2}\text{sdtw}_\gamma(\mathbf{C}(\mathbf{X}, \mathbf{X})) - \frac{1}{2}\text{sdtw}_\gamma(\mathbf{C}(\mathbf{Y}, \mathbf{Y}))} \tag{4}$$

A.3.2 Gradient (w.r.t. X)

$$\boxed{\nabla_\mathbf{X} D_\gamma^\mathbf{C}(\mathbf{X}, \mathbf{Y}) = (\mathbf{J}_\mathbf{X}\mathbf{C}(\mathbf{X}, \mathbf{Y}))^T \mathbf{E}_\gamma(\mathbf{C}(\mathbf{X}, \mathbf{Y})) - (\mathbf{J}_\mathbf{X}\mathbf{C}(\mathbf{X}, \mathbf{X}))^T \mathbf{E}_\gamma(\mathbf{C}(\mathbf{X}, \mathbf{X}))} \tag{5}$$

A.4 Sharp Soft-DTW

A.4.1 Formula

$$\boxed{\text{sharp}_\gamma(\mathbf{C}) = \langle \mathbf{E}_\gamma(\mathbf{C}), \mathbf{C} \rangle} \tag{6}$$

This removes the entropic regularization from soft-DTW:

$$\text{sharp}_\gamma(\mathbf{C}) = \text{sdtw}_\gamma(\mathbf{C}) + \gamma H(\mathbf{p}_\gamma(\mathbf{C})) \tag{7}$$

A.4.2 Gradient

$$\boxed{\nabla_\mathbf{C} \, \text{sharp}_\gamma(\mathbf{C}) = \mathbf{E}_\gamma(\mathbf{C}) + \frac{1}{\gamma} \nabla^2_\mathbf{C} \text{sdtw}_\gamma(\mathbf{C}) \cdot \mathbf{C}} \tag{8}$$

The second term is a Hessian-vector product.

A.5 Sharp Soft-DTW Divergence

A.5.1 Formula

$$\boxed{S^\mathbf{C}_\gamma(\mathbf{X}, \mathbf{Y}) = \text{sharp}_\gamma(\mathbf{C}(\mathbf{X}, \mathbf{Y})) - \frac{1}{2}\text{sharp}_\gamma(\mathbf{C}(\mathbf{X}, \mathbf{X})) - \frac{1}{2}\text{sharp}_\gamma(\mathbf{C}(\mathbf{Y}, \mathbf{Y}))} \tag{9}$$

A.5.2 Gradient (w.r.t. X)

$$\boxed{\begin{aligned}\nabla_\mathbf{X} S^\mathbf{C}_\gamma(\mathbf{X}, \mathbf{Y}) = (\mathbf{J}_\mathbf{X}\mathbf{C}(\mathbf{X}, \mathbf{Y}))^T \nabla_\mathbf{C} \text{sharp}_\gamma(\mathbf{C}(\mathbf{X}, \mathbf{Y})) \\ - (\mathbf{J}_\mathbf{X}\mathbf{C}(\mathbf{X}))^T \nabla_\mathbf{C} \text{sharp}_\gamma(\mathbf{C}(\mathbf{X}))\end{aligned}} \tag{10}$$

B Figures

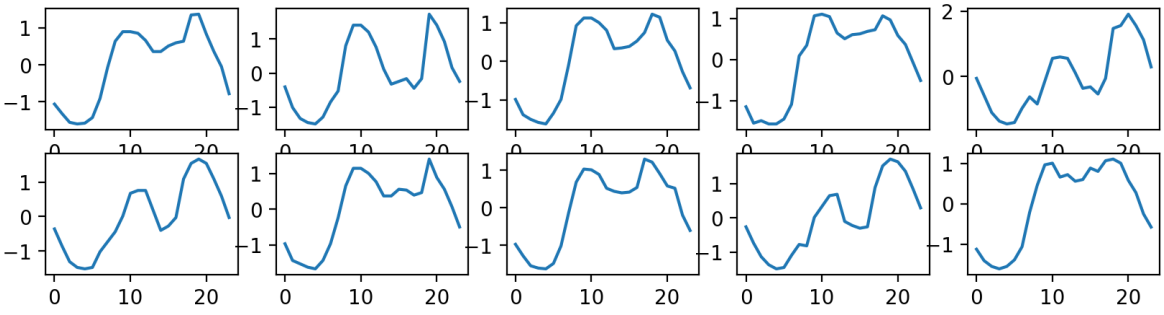

Figure 4: Ten time series sampled from one class of `ItalyPowerDemand`.

We validate one of the limitations of the soft-DTW: is not minimized when the time series are equal

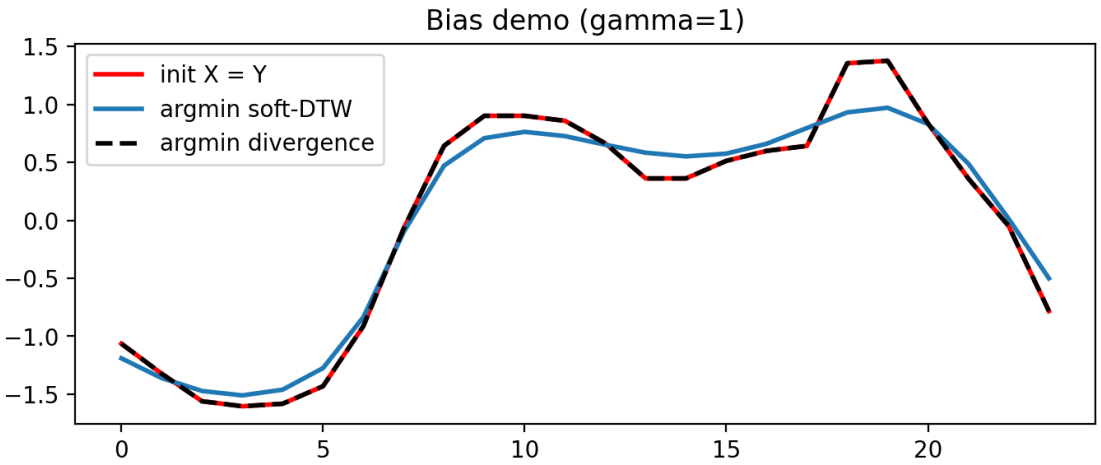

Figure 5: Bias demo on `ItalyPowerDemand` for $\gamma = 1$: minimizing $\text{sdtw}_\gamma(X, Y)$ yields a solution different from Y, while the divergence stays at Y.

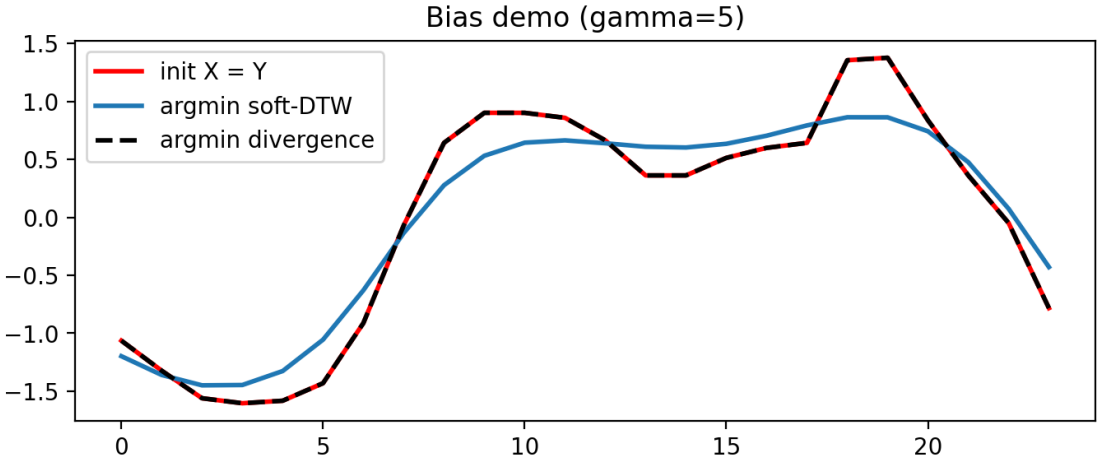

Figure 6: Bias demo on `ItalyPowerDemand` for $\gamma = 5$: the denoising effect of soft-DTW becomes stronger as γ increases, while the divergence remains anchored at Y.

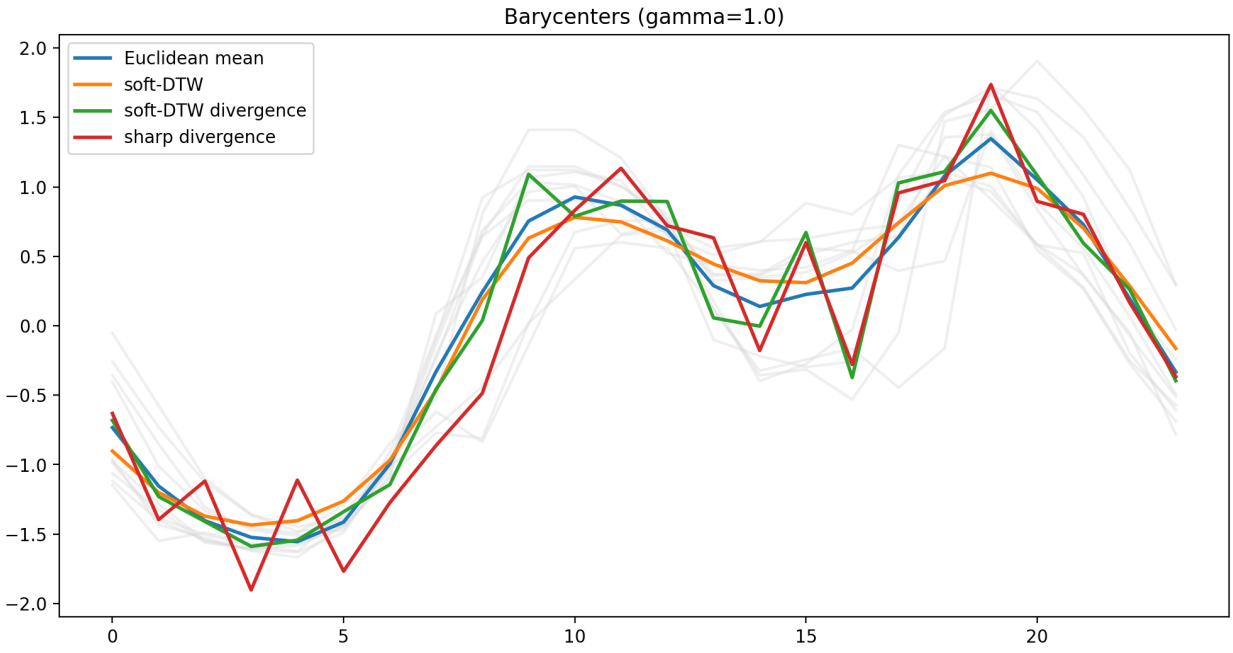

Figure 7: Barycenter comparison for $\gamma = 1$.

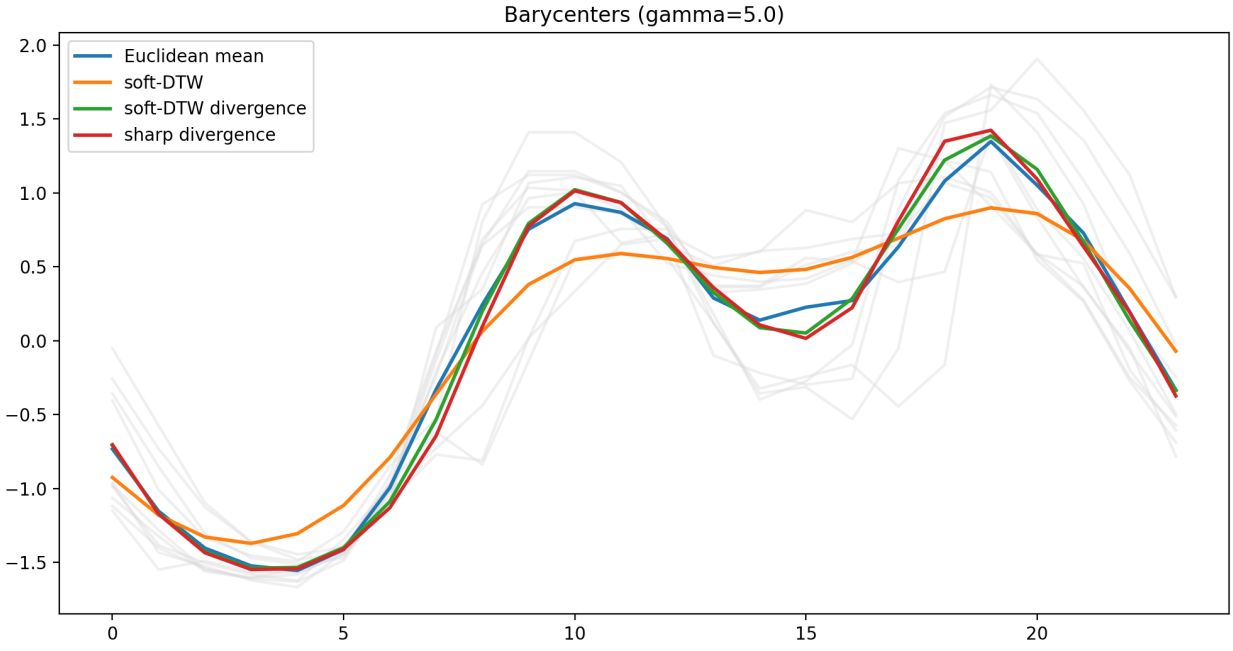

Figure 8: Barycenter comparison for $\gamma = 5$.

In the following figure we visualize the smoothing effect of γ

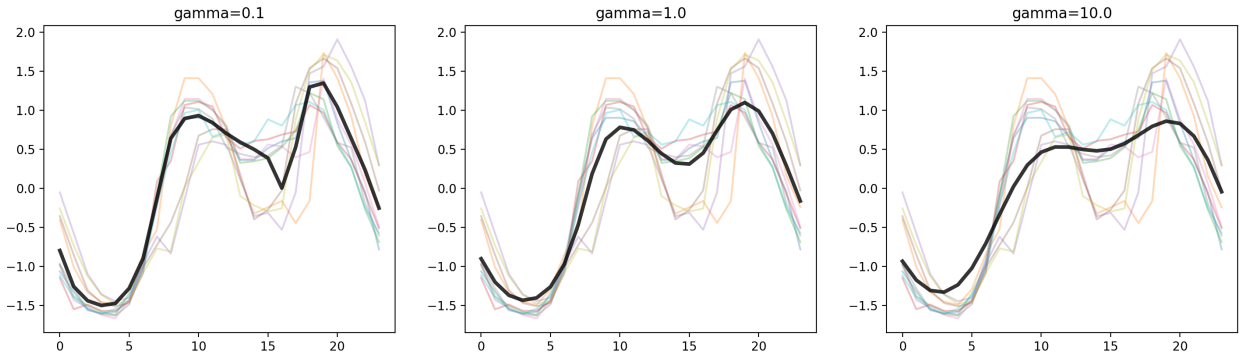

Figure 9: Soft-DTW barycenters for different values of γ on the same subset.

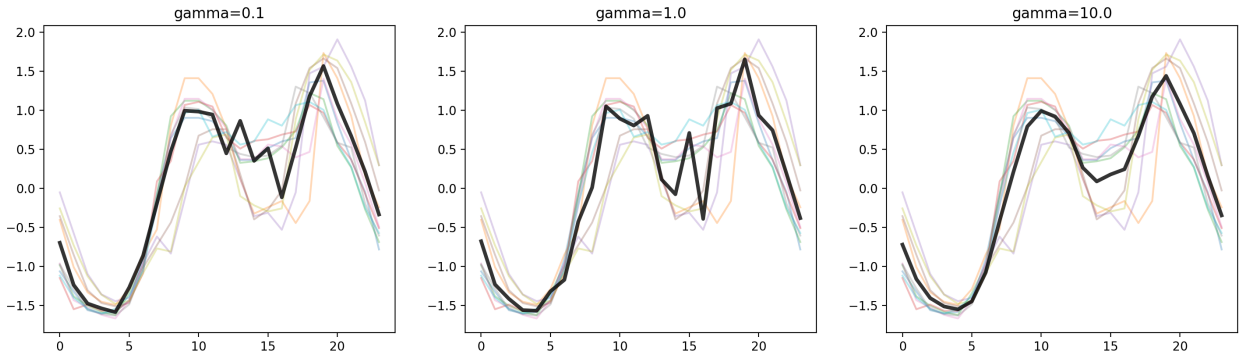

Figure 10: Soft-DTW divergence barycenters for different values of γ on the same subset.

B.1 Summary tables

Tables 2 and 3 are generated from `results/prototype_results.csv` by the final cells of `tests/final_deliverable.ipynb`. The win matrix follows the paper's convention: entry (A,B) is the percentage of datasets for which method A achieves at least 99% of method B's accuracy.

Method	Mean Accuracy
Dtw Approx	0.734
Sharp Sdtw Div	0.731
Sdtw Div	0.731
Sharp Sdtw	0.716
Sdtw	0.709
Euclidean	0.664
Mean Cost Div	0.532
Mean Cost	0.526

Table 2: Mean test accuracy across the benchmark datasets (nearest-centroid classifier).

	Euclidean	Sdtw	Sdtw Div	Sharp Sdtw	Sharp Sdtw Div	Mean Cost	Mean Cost Div	Dtw Approx
Euclidean	0.0	46.9	43.2	44.4	42.0	76.5	77.8	38.3
Sdtw	69.1	0.0	51.9	60.5	49.4	81.5	81.5	46.9
Sdtw Div	75.3	90.1	0.0	77.8	76.5	87.7	85.2	66.7
Sharp Sdtw	71.6	69.1	53.1	0.0	60.5	84.0	77.8	59.3
Sharp Sdtw Div	72.8	76.5	67.9	86.4	0.0	86.4	81.5	72.8
Mean Cost	43.2	25.9	22.2	25.9	24.7	0.0	66.7	22.2
Mean Cost Div	46.9	32.1	28.4	29.6	27.2	85.2	0.0	28.4
Dtw Approx	71.6	79.0	64.2	80.2	70.4	87.7	81.5	0.0

Table 3: Win matrix (percentages): entry (A,B) is the percentage of datasets for which A achieves at least 99% of B's accuracy.

Method	MeanRuntimeSec
Euclidean	0.03
Mean Cost	5.14
Sdtw	7.87
Dtw Approx	11.84
Mean Cost Div	14.26
Sharp Sdtw	15.68
Sdtw Div	38.03
Sharp Sdtw Div	73.39

Method	MeanRuntimeOverEuclidean
Euclidean	1.00
Sdtw	1122.45
Mean Cost	1694.53
Dtw Approx	2091.99
Sharp Sdtw	3396.64
Mean Cost Div	4906.28
Sdtw Div	7689.07
Sharp Sdtw Div	17835.96

Table 4: Runtime summary across benchmark datasets: mean runtime in seconds (left) and mean runtime normalized by Euclidean runtime (right).