ETH zürich

A Wasserstein Subsequence Kernel for Time Series

Christian Bock, Matteo Togninalli, Elisabetta Ghisu, Thomas Gumbsch, Bastian Rieck, Karsten Borgwardt

Department of Biosystems Science and Engineering
Machine Learning and Computational Biology Group

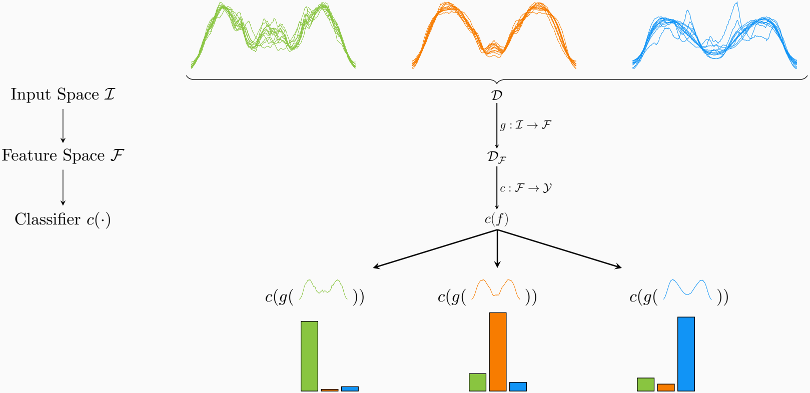
ETH Colors

ETHa ETHb ETHc ETHd ETHe ETHf ETHg ETHh ETHi

Table of contents (internal)

Motivation

Time Series Classification in a Nutshell



Time Series Classification in a Nutshell

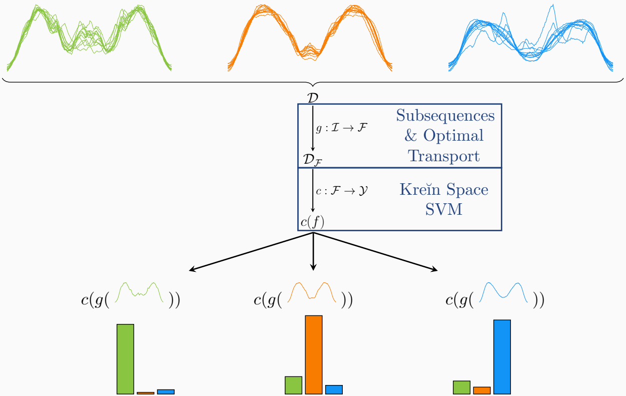
Input Space $\mathcal{I}$

$\downarrow$

Feature Space $\mathcal{F}$

$\downarrow$

Classifier $c(\cdot)$



Time Series Decomposition

Optimal Transport and the Wasserstein Distance

A Wasserstein Kernel for Time Series

From Distance to Kernels

Intuitive explanation

Definition (Wasserstein time series kernel)
Let T_i and T_j be two time series, and $s_{i1}, \ldots, s_{iU}$ as well as $s_{j1}, \ldots, s_{jV}$ be their respective subsequences. Moreover, let D be a $U \times V$ matrix that contains the pairwise distances of all subsequences, such that $D_{uv} := \operatorname{dist}(s_{iu}, s_{jv})$, where $\operatorname{dist}(\cdot, \cdot)$ denotes the usual Euclidean distance. The optimisation problem

$$\mathrm{W}_1(T_i, T_j) := \min_{P \in \Gamma(T_i, T_j)} \langle D, P \rangle_{\mathrm{F}}, \tag{1}$$

yields the optimal transport cost to transform T_i into T_j by means of their subsequences. Then, given $\lambda \in_{>0}$, we can define

$$\mathrm{WTK}(T_i, T_j) := \exp\left(-\lambda \, \mathrm{W}_1(T_i, T_j)\right), \tag{2}$$

which we refer to as our *Wasserstein-based subsequence kernel*;

Wasserstein Time Series Kernel

Algorithm 1 Wasserstein Time Series Kernel

Input: Time series for training and testing $\mathcal{T}_{\text{train}}$, $\mathcal{T}_{\text{test}}$; subsequence length w; kernel weight factor λ

Output: $\mathcal{K}^{\text{train}}, \mathcal{K}^{\text{test}}$

$\mathcal{S}^{\text{train}} \leftarrow \textsc{Subsequences}(\mathcal{T}_{\text{train}}, w)$ // Extract subsequences
$\mathcal{S}^{\text{test}} \leftarrow \textsc{Subsequences}(\mathcal{T}_{\text{test}}, w)$ // Extract subsequences
for $T_i \in \mathcal{T}_{\text{train}}$ **do**
    for $T_j \in \mathcal{T}_{\text{train}}$ **do**
        $\mathcal{D}^{\text{train}}_{ij} \leftarrow \mathrm{W}_1\left(\mathcal{S}^{\text{train}}_i, \mathcal{S}^{\text{train}}_j\right)$ // Wasserstein distance calculation (train)
    end for
    for $T_k \in \mathcal{T}_{\text{test}}$ **do**
        $\mathcal{D}^{\text{test}}_{ik} \leftarrow \mathrm{W}_1\left(\mathcal{S}^{\text{train}}_i, \mathcal{S}^{\text{test}}_k\right)$ // Wasserstein distance calculation (test)
    end for
end for
$\mathcal{K}^{\text{train}} \leftarrow \exp\left(-\lambda \mathcal{D}^{\text{train}}\right)$ // Kernel matrix calculation
$\mathcal{K}^{\text{test}} \leftarrow \exp\left(-\lambda \mathcal{D}^{\text{test}}\right)$ // Kernel matrix calculation
return $\mathcal{K}^{\text{train}}, \mathcal{K}^{\text{test}}$

Results

Experiments

Comparison with Other Kernels

Critical Difference Plot

Take aways

Questions?

References