6.1 Initial Considerations

Chapter 5 showed how optimal phase spaces can be characterized and computed using techniques and methods from Statistical Learning Theory and Machine Learning. The good results obtained in this process motivated us to extend our investigation on Dynamical Systems, which led to our second research question:

*RQ2. “Is it better to use phase-space rather than time-series modeling?”*

After measuring the forecasting accuracy to tackle RQ1, we next decided to validate phase-space analysis on classification scenarios. More precisely, we focused on the problem of Positive and Unlabeled (PU) data in dealing with semi-supervised learning. In this case, few labeled examples \( P \) from a single class of interest are available to proceed with the classification of unseen instances \( U \), according to their similarities with the known class.

In the scope of time series, most of the current studies propose to address this topic using a self-training approach based on similarity measurements on the time domain, such as the Euclidean Distance (ED) or the Dynamic Time Warping-Delta (DTW-D), to provide features for the self-training classification stage, which is typically performed with the 1-Nearest Neighbor (1-NN) algorithm (Wei and Keogh, 2006; Ratanamahatana and Wanichsan, 2008; Chen et al., 2013). Self-training is employed to accumulate knowledge and, consequently, improve the classification of new instances. Despite the relevant contributions of time-domain measurements, we claim that such approaches do not consider temporal recurrences commonly found in natural phenomena (*e.g.*, population growth, climate studies) and are more sensitive to local noise and fluctuations, as already mentioned in Chapter 5.

To exemplify and reinforce such drawbacks of time-domain measurements, consider the analysis of a cyclical phenomenon, whose
behavior is described by a sinusoidal signal (Equation 3.1, repeated below for ease of reading)

\[ x(t) = A(t) \sin(2\pi t/n) + \theta + U(a, b), \]

where \( A(t) \) is the amplitude along time, sampled at moments \( t = 0, \ldots, n - 1 \); \( \theta \) changes the sinusoidal phase; and \( U(a, b) \) adds noise to the samples following a Uniform probability distribution in range \([a, b]\).

Now consider that we create three examples of time series with the same length \( n = 200 \). The first series is a free-noise sinusoidal function with \( A(t) = 1 \forall t \), \( \theta = 0 \) and \( U(0, 0) \). The second is a dissipative sine whose observations were produced using \( A(t) = \frac{n-t}{n} \), \( \theta = \pi/2 \) and \( U(-0.1, 0.1) \). The third series represents a random noise following a Uniform probability distribution \( U(-0.5, 0.5) \). All series are illustrated in Figure 6.1. Note that, although we made a couple of changes in the second signal, it remains sinusoidal-like. This simulates a real-world scenario in which we have two signals collected from the same phenomenon representing distinct behaviors at different time instants. The first series corresponds to the time interval in which the phenomenon is conservative. After some interaction (coupling) with another system, the signal begins to lose power eventually converging to zero, such as in the case of the damped harmonic oscillator (Alligood et al., 1996). This leads us to the second signal.

![Figure 6.1: Examples of two series produced by variations of the sinusoidal function (a, b) and another series generated using a uniform distribution (c). Assuming the first signal as the initial known positive example, time-domain measurements (see Section 6.3) may consider the conservative series (a) as more similar to the uniform distribution (c) than to the dissipative one (b), thereby misleading a classifier. Adapted from (Pagliosa and de Mello, 2018).](image)

Assume some specialist told us that the first series belongs to the positive set \( P \), which was already studied in the context of our application domain (i.e., cyclical phenomena), and that the other two series compose the unlabeled dataset \( U \). Next, assume,
for the sake of example, that we use a self-training strategy to label the most similar time series in \( U \) to \( P \), using a 1-NN algorithm (other learning algorithms can be used as well). Although we know that the dissipative sine should be classified as a positive instance, time-domain measurements provide us the undesired result that the random noise should be added to the positive set instead (more details in Table 3). As discussed further in this chapter, besides not comparing recurrences (a feature that should be considered when dealing with natural phenomena), time-domain measurements are more sensitive to local differences enhanced by noise and mean-valued observations, so they can mislead classification. In a self-training scenario, this can lead to inconsistent and undesired results.

The above issues have motivated us to investigate the use of phase-space representations as an alternative to time-series representations for building classifiers of temporal data. In detail, we propose the use of the Maximum Diagonal Line of the Cross-Recurrence Quantification Analysis (MDL-CRQA), applied on phase spaces (Takens, 1981), as similarity measurement for classification. By comparing phase spaces rather than the series themselves, we can assess how their trajectories change along time (Marwan and Webber, 2015), including their periodicities and temporal cycles, as well as decreasing noise influences.

The remaining of this chapter is organized as follows. Section 6.2 shows the related work of time-series semi-supervised learning. Different methods typically used to compare time series are described in Section 6.3. Our approach is given in Section 6.4. We perform experiments comparing time-domain and phase-space domain in Section 8.5, to later discuss our results in Section 6.5.4 and finally draw conclusions.

6.2 RELATED WORK FOR SEMI-SUPERVISED LEARNING IN TIME SERIES

Despite the proposal of semi-supervised techniques like self-training (Li and Zhou, 2005), generative models (Baluja, 1999), co-training (Blum and Mitchell, 1998), density-based (Bennett and Demiriz, 1998), graph-based (Blum and Chawla, 2001), outlier detection (Janssens et al., 2009), and their extensions/modifications to tackle specific scenarios (Nigam et al., 2000; Chapelle and Zien, 2005; Zhu et al., 2009; Chapelle et al., 2010; Daneshpazhouh and Sami, 2014; Wang et al., 2016; Sheikhpour et al., 2017; Pereira and da Silva Torres, 2018; Wu and Prasad, 2018, etc.), to the extent of our knowledge, few studies addressed semi-supervised classification for time-series analysis in the literature.
The first study related to semi-supervised time series was proposed by Wei and Keogh (2006). By starting with a single positive instance $s$ representing the positive set $P$, their self-learning method classifies a new positive instance belonging to the unlabeled dataset $U$ as the most similar series in $U$ to $P$, and the process continues until some stopping criterion is met. Despite their seminal contribution to the area, their approach has a couple of problems: i) they used the Euclidean Distance (ED) to compute the 1-NN algorithm, which is known to be less accurate than the Dynamic Time Warping (DTW) method in the presence of time-displacements (Ratanamahatana and Keogh, 2004); and ii) their stopping criterion (later referred to as Minsofar) was confirmed to be inadequate in several scenarios (Ratanamahatana and Wanichsan, 2008). Based on these observations, Ratanamahatana and Wanichsan (2008) proposed the Stop Criterion Confidence (SCC), which despite improving upon Minsofar, is not yet ideal, as it yields early termination for multiple datasets. Separately, Chen et al. (2013) used DTW-D (ratio of DTW by ED) to compare similarities between time series, improving the results reported by Wei and Keogh (2006).

Alternatively, Nguyen et al. (2011) relied on the method proposed by Wei and Keogh (2006) to classify a positive initial (training) set to later run $k$-means on the unlabeled dataset. Afterwards, the method applies PCA (Jolliffe, 1986) on both labeled and unlabeled sets to finally classify clusters based on their similarities provided by eigenpairs. Zhong (2004) uses self-training with Hidden Markov Models (HMM) to summarize time-series information. The algorithm first initializes the number of states using parameter $k$, from $k$-means, then maximizes the likelihood estimation for the HMM using positive labeled examples. Unlabeled instances are set as positive when their accumulated transition probability (similarity) is high for the trained positive model.

Out of the scope of self-training approaches, few other techniques exist for this problem. Marussy and Buza (2013) proposed a cluster-and-label multi-class algorithm which computes a minimum spanning forest (using DTW) among all instances, so that each tree has one labeled instance as root. Each tree starts with one labeled instance, proceeding with the addition of further nodes. At the end, series belonging to a tree are labeled according to the label of its root.

In short, we found that the majority of studies tackling semi-supervised time-series classification on PU problems have used the 1-NN algorithm with a self-training approach, including the seminal research proposed by Wei and Keogh (2006). Therefore, we also decided to follow this approach in order to support a fair comparison of results. A detailed analysis of the sensitivity of the 1-
NN classifier and a comparison thereof against other classification methods are out of the scope of our work. More importantly, we also noticed that all related methods use time-domain measurements, such as ED and DTW, to measure similarities among time series. As already outlined above, this strategy may not be the best approach in many situations, especially when time series present strong cyclical patterns and trends. Hence, in contrast to existing research, we propose a novel self-training approach to tackle semi-supervised PU time-series classification using MDL-CRQA as similarity measurement, applied on the series phase spaces rather than on the series themselves. This allows us to assess and compare time-series recurrences more fairly, as we describe in Section 6.4. Our approach can also be extended to use other classification algorithms rather than 1-NN, without loss of generality.

6.3 TIME-DOMAIN SIMILARITY MEASUREMENTS

Current self-training methods use time-domain measurements to find the most similar instance to be labeled as a positive example throughout iterations. For instance, given two unidimensional time series $T_i, T_j$ with $n_i, n_j$ observations each, the Euclidean Distance (ED) computes the similarity between them as

$$ED(T_i, T_j) = \sqrt{\sum_{t=0}^{n_i-1} (x_i(t) - x_j(t))^2}. \quad (6.2)$$

Despite simple, ED is not suitable to compare time-displaced series, additionally requiring both series to have the same length $n_i = n_j$ (although this constraint can be relaxed via interpolation approaches (Ratanamahatana and Keogh, 2004)).

Dynamic Time Warping (DTW) was proposed to address the comparison of time-displaced series, by finding the best match between shifted observations along time by computing

$$DTW(T_i, T_j) = \sqrt{\sum_{t=0}^{n_i-1} p_{i,j}(t)^2}, \quad (6.3)$$

where $p_{i,j}(t) = x_i(t+\alpha(t)) - x_j(t+\beta(t))$ corresponds to the shortest warping path, with $\alpha(t), \beta(t) \in \mathbb{Z}$.

However, there are scenarios in which ED and DTW lead to incorrect results (Chen et al., 2013). To mitigate this drawback, a combination of both approaches was proposed, referred to as Dynamic Time Warping-Delta (DTW-D), computed by

$$DTW-D(T_i, T_j) = \frac{DTW(T_i, T_j)}{ED(T_i, T_j)}, \quad (6.4)$$

97
when compared to DTW and ED, DTW-D measurements improve classification results in several contexts.

Finally, Mean Distance from the Diagonal Line (MDDL) (Rios and de Mello, 2013) is another method to measure time-series similarities, defined as

$$\text{MDDL}(T_i, T_j) = \sum_{t=0}^{n_i-1} (p_{i,j}(t) - d_{i,j}(t))^2,$$

(6.5)

where $d_{i,j}(t)$ indicates the diagonal line (perfect match) in the space found by DTW, as illustrated in Figure 5.3(b). MDDL is adequate to compare time-series trends and disregard mean-valued time series, similarly to DTW-D.

### 6.4 Semi-Supervised Time-Series Classification Using CRQA

Let two phase spaces $\Phi_i$ and $\Phi_j$ be properly reconstructed after applying Takens’ embedding theorem on time series $T_i$ and $T_j$, respectively, as discussed in Chapter 5. A Cross Recurrence Plot (CRP) between these two phase spaces yields the matrix $R$ having as entries the values

$$R_{a,b} = \begin{cases} 
1, & \text{if } \phi_i(a) \text{ is a neighbor of } \phi_j(b) \text{ according to an open ball centered at } \phi_i(a) \text{ with radius } \varepsilon, \\
0, & \text{otherwise},
\end{cases}$$

(6.6)

which indicates when (and where in the attractor) states $\phi_i(a) \in \Phi_i$ and $\phi_j(b) \in \Phi_j$ are close enough to each other. The CRP matrix $R$ can also be used to measure for how long two phase spaces remain similar to each other. For instance, horizontal or vertical traces suggest trajectories on the given state are bound by an attractor in one space, while changing “normally” in the other. Similarly, sparse and small areas in $R$ can indicate phase spaces do not share trajectories. However, such interpretations are delicate and require specialized domain knowledge. To avoid such complications, the Cross Recurrence Quantification Analysis (CRQA) was designed to extract a set of predefined measurements based on the CRP, which can be automatically used to reveal important features such as patterns and statistical distributions between phase spaces. Lastly, it is worth to say that, when computing the CRP, the number of states $N_i$ and $N_j$ may vary, but the dimension $m$ must be the same for both embeddings being compared.

Based on Serrà et al. (2009), we consider the Maximal Diagonal Line (MDL) to indicate for how long the trajectories of two differ-
ent phase spaces remain similar (close) to each other. In order to
compute MDL, we start by filling the first column and row of \( \mathbf{R} \)
with zeros (\( \mathbf{R}[*,0] = \mathbf{R}[0,*) = 0 \)) and define

\[
R_{a,b} = \Theta(\varepsilon_a^i - \| \phi_i(a) - \phi_j(b) \|_2) \Theta(\varepsilon_b^j - \| \phi_j(b) - \phi_i(a) \|_2),
\]

for \( a = 1, \ldots, N_i - 1 \), \( b = 1, \ldots, N_j = 1 \), where \( \varepsilon_a^i \) and \( \varepsilon_b^j \) are open
ball radii for \( \phi_i(a) \) and \( \phi_j(b) \), respectively, and \( \Theta(\cdot) \) is a Heaviside
step function given by

\[
\Theta(v) = \begin{cases} 
0, & \text{if } v < 0, \\
1, & \text{if } v \geq 0.
\end{cases}
\]

(6.8)

The radius \( \varepsilon_b^j \) is found by firstly computing the Euclidean distances
from state \( \phi_i(a) \in \phi_i \) to every other state \( \phi_j \in \phi_y \). Then, we sort
all those distances out in increasing order and set \( \varepsilon_b^j = \varepsilon \), having \( \varepsilon \)
as big enough to include the \( k \)-th nearest neighbor from \( \phi_i(a) \). In
practice, we set \( k \) to 1% of the number of states in phase space.
This way, we ensure that every state in \( \phi_i \) will always have the
same number of neighbors in the other phase space \( \phi_j \) during the
similarity analysis. The finding of the radius \( \varepsilon_b^j \) proceeds analogously.

The orbits of similar phase spaces may suffer from noise or small
fluctuations. Thus, a perfect diagonal line may not occur in most
scenarios. To model this, we can relax the concept of similarity by
also allowing some “bumps” while computing the maximum diagonal in Equation 6.8. Thus, our next step consists of running a
Dynamic Programming algorithm to fill the matrix \( \mathbf{Q} \) which
accumulates and penalizes recurrence similarities stored in \( \mathbf{R} \). The
matrix \( \mathbf{Q} \) is defined by its entries

\[
Q_{a,b} = \begin{cases} 
\max\{Q_{a-1,b-1}, Q_{a-2,b-1}, Q_{a-1,b-2}\} + 1, & \text{if } R_{a,b} = 1, \\
\max\{0, \\
Q_{a-1,b-1} - \gamma(R_{a-1,b-1}), \\
Q_{a-2,b-1} - \gamma(R_{a-2,b-1}), \\
Q_{a-1,b-2} - \gamma(R_{a-1,b-2})\}, & \text{otherwise}.
\end{cases}
\]

(6.9)

We then use the Maximal Diagonal Line, defined as \( \max(\mathbf{Q}) \) to
compare two phase spaces. To compute \( \mathbf{Q} \), we initially set its two
first columns and rows to zero, and use the auxiliary function

\[
\gamma(z) = \begin{cases} 
\gamma_o, & \text{if } z = 1, \\
\gamma_e, & \text{if } z = 0,
\end{cases}
\]

(6.10)
with $\gamma_o = 5$, $\gamma_e = 0.5$ as disruption penalties\textsuperscript{1}. As we are interested in providing a dissimilarity measure, we consider the inverse of MDL as $1/\max(Q)$. Note that $\max(Q)$ is never zero.

We claim that measuring similarity in phase space (rather than in the time domain) leads to better classification results in the context of semi-supervised PU learning. In order to support our claim, consider the three time series in Figure 6.1. In this situation, although the first two time series were produced by the same generating rule (sinusoidal function), local differences enhanced by different parametrizations lead time-domain similarities such as ED, DTW, DTW-D and MDDL to wrongly classify unlabeled instances.

Consider now performing comparisons in phase space rather than in the time domain. If we know or discover that the positive class contains sinusoidal-based time series, we could reconstruct the positive phase space using Takens’ embedding theorem and analyze the similarity of this space against the other unfolded phase spaces (from the unlabeled dataset) using the same embedding parameters. More precisely, we use MDL from CRQA of two spaces, here referred to as MDL-CRQA. Figure 6.2 shows the phase spaces for the series in Figure 6.1 after reconstructing them using $m = 2$ and $\tau = 1$.

![Figure 6.2: Phase spaces obtained for the time series illustrated in Figure 6.1. Dark circles, blue triangles and red crosses represent phase-space states of the first, second and third time series, respectively. Adapted from (Pagliosa and de Mello, 2018).](image)

Table 3 lists the dissimilarities of the above time series when using time-domain dissimilarity methods (ED, DTW, DTW-D, and MDDL) as well as the phase-space-based MDL-CRQA. The results confirm that MDL-CRQA supports a better classification than local time-based measurements for this example.

\textsuperscript{1} Disruption penalties are heuristic weights used by Serrà et al. (2009) to improve the measurement of the longest diagonal line such as in Edit distance (Ristad et al., 1998).
Table 3: Comparing time series by using different similarity measures. We show the minimum, mean and maximum dissimilarities for each series (after adding random noise to it) vs the target series over 30 experiments. We show, in bold, the most similar "unknown" series that would be classified as the next positive instance.

<table>
<thead>
<tr>
<th>Measure</th>
<th>Series</th>
<th>Minimum</th>
<th>Mean</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>ED</td>
<td>Sine 1 / Sine 2</td>
<td>0.037</td>
<td>0.037</td>
<td>0.037</td>
</tr>
<tr>
<td></td>
<td>Sine 1 / (U)</td>
<td>0.034</td>
<td>0.033</td>
<td>0.035</td>
</tr>
<tr>
<td>DTW</td>
<td>Sine 1 / Sine 2</td>
<td>0.266</td>
<td>0.262</td>
<td>0.269</td>
</tr>
<tr>
<td></td>
<td>Sine 1 / (U)</td>
<td>0.234</td>
<td>0.227</td>
<td>0.241</td>
</tr>
<tr>
<td></td>
<td>Sine 1 / (U)</td>
<td>13.712</td>
<td>13.199</td>
<td>14.331</td>
</tr>
<tr>
<td>MDDL</td>
<td>Sine 1 / Sine 2</td>
<td>64.301</td>
<td>35.508</td>
<td>83.686</td>
</tr>
<tr>
<td></td>
<td>Sine 1 / (U)</td>
<td>22.396</td>
<td>10.137</td>
<td>53.695</td>
</tr>
<tr>
<td>MDL-CRQA</td>
<td>Sine 1 / Sine 2</td>
<td>0.026</td>
<td>0.020</td>
<td>0.032</td>
</tr>
<tr>
<td></td>
<td>Sine 1 / (U)</td>
<td>0.158</td>
<td>0.111</td>
<td>0.200</td>
</tr>
</tbody>
</table>

The idea of above experiment is not to induce that time-domain dissimilarities perform incorrectly in all scenarios and should be discarded. Conversely, we just want to show that it might not be difficult to find examples for which all these dissimilarities would lead to wrong classification results. Additionally, we reinforce why phase-space measurements should be included in time-series analysis.

In summary, our semi-supervised classification method requires three initial settings: i) one initial positive example; ii) the embedding dimension \(m\); and iii) the time delay \(\tau\) (both \(m\) and \(\tau\) are used to represent the phase space for the positive class). Given those, we compute the embedding parameters for the single starting positive example to unfold the phase space of the positive class according to Takens’ embedding theorem. Next, we unfold any new instance phase space using the same embedding parameters, and compare it with the positive example using MDL-CRQA as similarity function. As proposed by Wei and Keogh (2006), we use the 1-NN algorithm for classification, i.e., to specify the unlabeled instance with the greatest probability to belong to the positive class. As already mentioned, other classification algorithms can also be easily used.
6.5 EXPERIMENTS

In order to get additional insights on the behavior and performance of our proposed phase-space dissimilarity, we performed three sets of experiments: i) the first with synthetic time series to validate our method; ii) the second with real-world time series; and iii) and a third, also involving real-world data, that simulates a more difficult scenario when non-positive time series have similar features both in time and phase domains with the positive set.

Each experiment considers three dynamical systems. When such systems are represented by generating rules $R(\cdot)$ in the form of unidimensional signals/functions, such as the sinusoidal function (Equation 6.1), we basically take $n$ observations from $R(\cdot)$; when dealing with multidimensional fluxes/maps, such as the Lorenz system, we take $n$ observations from the first dimension (any other dimension could have been used equally well) to represent trajectories in the phase space (Kantz and Schreiber, 2004). For all experiments, we use $n = 10^5$ samples.

One of the functions in Chapter 3 is chosen to represent the phenomenon under the positive class. A few other functions are defined to compose the unlabeled dataset $U$. To create the training and test datasets, we divide the positive and unlabeled series into 32 and 50 sub-series, respectively, with 200 observations each.

To test the classification performance in the presence of noisy data, we generated time series using $\mathcal{N}(0,1^2)$, i.e., a normal probability distribution with mean 0 and standard deviation 1, added to $2/3$ of all positive instances. In addition, we also included in the unlabeled set time series representing the mean-valued series from the positive series, as well series deriving from Normal distributions $\mathcal{N}(0,0.05^2)$ and $\mathcal{N}(0,0.1^2)$. Summarizing, our experiments use 32 positive instances and 232 unlabeled series (200 plus 32 mean-valued series from the positive set), as illustrated in Figure 6.3.

We used 50% of positive and 90% of unlabeled instances for training, leaving the remaining time series for testing. Only a single positive example was used to initiate the self-learning algorithm; the remaining positive instances were added to the unlabeled dataset. As the stopping criterion is an open problem in the PU literature, we decided to employ the method proposed in (Chen et al., 2013) to train our classifier until all positive instances (belonging to the unlabeled dataset) were labeled. We then used the labeled training series to classify the test observations using the 1-NN algorithm. As the classifier can be influenced by the choice of the selected positive instance, we ran it using different values for $s$ multiple times. As final results, the mean precision, recall and F1-score performances over all experiments are reported.
Figure 6.3: The set of unlabeled instances $U$ is composed of: i) positive instances $P$ but one randomly selected series $s$; ii) series from other systems; iii) two random series representing noise and iv) constant mean-valued series from $P \setminus \{s\}$. The self-learning algorithm continues until all positive instances are correctly classified, i.e., when $P \subseteq P'$. Adapted from (Pagliosa and de Mello, 2018).

6.5.1 Case Study 1: Synthetic Data

We start our experiments by analyzing synthetic time series in order to validate our method. For such time series, their generating rules $R(\cdot)$ are well known. Hence, the embedding parameters to reconstruct their phase spaces are also known. In this context, we chose the Logistic map (Equation 3.2), the Hénon map (Equation 3.3), and the Lorenz system (Equation 2.6) to compose the synthetic experiments.

Among all possibilities, we defined the Lorenz system to compose the positive class, randomly choosing one series from it and leaving all remaining series to form the unlabeled set. We used the well-known embedding dimension $m = 3$ and time delay $\tau = 8$ for reconstructing the phase space associated with the Lorenz system. Classification performances are shown in Table 4. As one can notice, time-domain measurements are more sensitive to local disturbances, such as, but not limited to, noisy observations and mean-valued series. Therefore, by comparing time-series trajectories and recurrences along a wider period of time, MDL-CRQA becomes a global measurement that suffers less from those fluctuations, achieving better classification results.

6.5.2 Case Study 2: Real-World Data

In this experiment, we consider the real-world Sunspot dataset (Andrews and Herzberg, 1985) to belong to the positive set. As this series follows sinusoidal-like trajectories (but with significant noise), we chose the embedding pair $(m = 2, \tau = 8)$ to define the positive class phase space. The unlabeled set was formed by series deriv-
Table 4: MDL-CRQA supports better classification results for the Case Study 1: our method correctly classified 100% of the positive instances.

<table>
<thead>
<tr>
<th>Dissimilarity</th>
<th>Precision</th>
<th>Recall</th>
<th>F1-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>ED</td>
<td>0.410</td>
<td>1.000</td>
<td>0.581</td>
</tr>
<tr>
<td>DTW</td>
<td>0.410</td>
<td>1.000</td>
<td>0.581</td>
</tr>
<tr>
<td>DTW-D</td>
<td>0.842</td>
<td>1.000</td>
<td>0.914</td>
</tr>
<tr>
<td>MDDL</td>
<td>0.444</td>
<td>1.000</td>
<td>0.615</td>
</tr>
<tr>
<td>MDL-CRQA</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
</tbody>
</table>

ing from the Rössler system and the Ikeda map (as well as the mean-valued and noise series for the positive class).

The performance results are listed in Table 5. Similarly to the first experiment, MDL-CRQA yielded the best classification performance, achieving almost 20% more precision than DTW-D. The best explanation for such results is again the presence of noise, which usually misleads classification when time-domain dissimilarities such as ED and DTW are used.

Table 5: MDL-CRQA supports better classification results for the Case Study 2.

<table>
<thead>
<tr>
<th></th>
<th>Precision</th>
<th>Recall</th>
<th>F1-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>ED</td>
<td>0.410</td>
<td>1.000</td>
<td>0.581</td>
</tr>
<tr>
<td>DTW</td>
<td>0.410</td>
<td>0.992</td>
<td>0.581</td>
</tr>
<tr>
<td>DTW-D</td>
<td>0.787</td>
<td>0.898</td>
<td>0.820</td>
</tr>
<tr>
<td>MDDL</td>
<td>0.432</td>
<td>1.000</td>
<td>0.603</td>
</tr>
<tr>
<td>MDL-CRQA</td>
<td>0.962</td>
<td>1.000</td>
<td>0.979</td>
</tr>
</tbody>
</table>

6.5.3 Case Study 3: Recurrent Time Series

In the last experiment, we analyze how our method behaves when non-positive time series have similar (up to a certain limit) recurrences of the positive instances. In other words, we simulate the case where some unlabeled time series have similar phase spaces to the positive instance, but should not be taken as positive due to small variations. This case is more challenging than the first two described so far, where the distinction between the classes is sharper.

In order to construct this scenario, we used the same series from previous experiment, i.e., the Sunspot dataset as the positive class, and series deriving from the Rössler system and the Ikeda map de-
fined the unlabeled set, respectively (as well as the mean-valued and noise series). We also included a sinusoidal function with parameters \( A(t) = 1 \), \( \theta = 0 \) and \( \mathcal{U}(0,0) \) and noise \( \mathcal{N}(0,0.05^2) \) added to it. This last addition creates a non-positive phase space (the sine phase space) whose dynamics partially mimics the Sunspot phenomenon, as illustrated in Figure 3.5. Nevertheless, as observed in this figure and in Figure 3.1, the Sunspot and the sine time series model different phenomena. Consequently, sine instances should not be classified as positive.

Although similarities between phase spaces may lead MDL-CRQA to wrongly classify some positive instances, this measurement still provides the best classification results, as shown in Table 6. Therefore, we empirically conclude that our method is robust enough to classify PU time series even when non-positive time series share some common patterns and recurrences with the positive examples.

Table 6: Case Study 3: Although positive and unlabeled series (especially the ones generated from the sine function) present similar trends and recurrences, MDL-CRQA still supports better classification when compared to time-domain measurements.

<table>
<thead>
<tr>
<th>Dissimilarity</th>
<th>Precision</th>
<th>Recall</th>
<th>F1-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>ED</td>
<td>0.410</td>
<td>1.000</td>
<td>0.581</td>
</tr>
<tr>
<td>DTW</td>
<td>0.381</td>
<td>0.875</td>
<td>0.530</td>
</tr>
<tr>
<td>DTW-D</td>
<td>0.628</td>
<td>1.000</td>
<td>0.760</td>
</tr>
<tr>
<td>MDDL</td>
<td>0.444</td>
<td>1.000</td>
<td>0.615</td>
</tr>
<tr>
<td>MDL-CRQA</td>
<td>0.849</td>
<td>1.000</td>
<td>0.917</td>
</tr>
</tbody>
</table>

Even for unlabeled series with sinusoidal behavior, MDL-CRQA was capable of separating those series from the Sunspot ones. This happened since the Sunspot series are not a perfect sinusoidal function and, in addition, it contains noise. If our classifier had not achieved good results, we could also consider to overembed the positive class to obtain a more representative phase space (Kantz and Schreiber, 2004), i.e., increase the embedding dimension \( m \) in order to unfold more complex data (such as data containing noise).

By adding extra dimensions to the phase space (up to a certain limit), one can analyze the details of more complex dynamical system trajectories (Alligood et al., 1996) and improve the separation of Sunspot versus the sine series. Although the Sunspot resembles sinusoidal, it is not as much sinusoidal as the sine function itself. Therefore, the trend is that those phase spaces become more dissimilar to each other as we increase their embedding dimensions.
6.5.4 Discussion

According to our experiments, we confirmed that MDL-CRQA supports better classification results for both synthetic and real-world time series whose data present recurrent observations. In order to mitigate the influences of the starting positive instance in the self-learning algorithm, we performed each experiment several times varying the starting example, and reported the mean performances achieved at each iteration as the final result. Nonetheless, we noticed the results barely vary when different positive examples were used.

As reported in Section 6.5.1, our method correctly classified 100% of the positive instances in the first experiment (synthetic data with added noise). When dealing with the real-world Sunspot dataset (Section 6.5.2), although making some errors, MDL-CRQA still achieved the best classification results when compared to time-domain measurements, even when non-positive time series share common recurrences with the positive set (Section 6.5.3).

In addition, we also have tested our method with datasets from the UCR collection (Chen et al., 2015), which contain several time series commonly used as benchmark. Training and testing files are already defined for each dataset in this collection. In this context, we think two relevant aspects are worth to be mentioned. First, the majority of those datasets were not designed to simulate PU problems, bringing relevant issues when defining the positive class. As consequence, the proper embedding parameters to unfold the positive phase space were also unknown. In order to overcome those issues, we naively defined all instances under the class label 1 as positive (we observed this class usually has fewer observations) and left all remaining classes as being part of the unlabeled dataset; and, to create the phase space for positive instances, we relied on current estimation methods (Kennel et al., 1992; Fraser and Swinney, 1986). As in the previous experiments, we assume all positive instances were unlabeled except one which was used to start the self-training process. For these datasets, our results of precision and recall did not surpass 0.5 on average. However, we noticed that none of the time-domain measurements achieved good results, to mention, they did not surpass 0.5 in terms of F1-score on average. While DTW achieved better classification performances for some datasets, DTW-D, MDDL, and even ED measurements provided better results for others.

Although the results for the UCR datasets are far less positive than the ones for the three types of datasets discussed in the previous sections, we report them here for two reasons: i) to confirm one of the main motivations of our study here, namely that time-domain measurements can lead to inconsistent results; ii) to high-
light the importance of having a proper phase-space embedding (we cannot get good dissimilarities if this embedding is not found).

Limitations of our method include a higher computational effort when compared to time-domain methods\textsuperscript{2}, since MDL-RQA needs to compare phase states ($O(N_i)^2$ steps) while time-domain measurements perform computations only using the time series itself ($O(n)$ and $O(n^2)$ steps for ED and DTW). Thus, despite the computation of the Maximal Diagonal Line (MDL) demands extra processing time specifically when studying high dimensional phase spaces, we believe this is not prohibitive in several practical scenarios due the increase in the use of cluster computing, optimization packages and parallel programming. As future work, one could use more than one simultaneous measurement to enhance classification results. As we stated, the unlabeled instances may contain any time series outside the positive class. Therefore, even MDL-CRQA may wrongly classify certain datasets. In this situation, one could use a co-training technique to learn from the time domain using ED, DTW, DTW-D or MDDL and from the phase domain using MDL-CRQA. Finally, the setting the stopping criterion is a fundamental, but open, question in PU problems, which still requires further research.

\textbf{6.6 Final Considerations}

The PU scenario is a well-known problem in semi-supervised classification (Wei and Keogh, 2006; Ratanamahatana and Wanichsan, 2008; Chen et al., 2013). Despite relevant contributions, current methods tackling PU problems using measurements such as ED, DTW and DTW-D do not compare temporal recurrences. This feature may be of great importance when comparing time series, especially when observations repeat themselves as is the case of many real-world scenarios, some of them studied along the manuscript such as population growth, meteorological data and sunspot activity.

In this chapter, we investigated the comparison of time series by using a dissimilarity measure, called MDL-CRQA, defined on their phase spaces, computed by using suitable embedding parameters. Our proposal, measured over two attractors in the same phase space (where all possible scenarios are unfolded, and therefore recurrences are easily modeled) is a feasible approach to measure the amount of recurrence one time series has with another. This approach attempts to mitigate local problems caused by mislead-

\textsuperscript{2} For example, the Case Studies took around 30 minutes while running on a 40-core Xeon processor at 2.8Ghz.
ing noise, chaotic events and eventually different observations produced along the collection of the phenomenon of interest.

In order to apply our method, we require two parameters from the user: the embedding dimension $m$ and time delay $\tau$ for reconstructing the series under the positive class according to Takens’ embedding theorem (Takens, 1981). These parameters can be either known (for a given problem domain) or else estimated using the methods discussed in Chapter 5.

Experimental results confirm MDL-CRQA improves classification results for PU time series when compared against the mostly used time-domain similarity measurements. This answers our research question 2 (RQ2) positively: yes, phase-space methods do lead to better models of time series, when properly unfolded. However, the answer needs to be nuanced: we have only shown that phase-space methods are superior to time-domain modeling for a subclass of problems, namely PU scenarios; and even for those, there exist datasets for which both phase-space models and time-domain models perform poorly. Lastly, refining our initial phase-space model, e.g., by using different distance measures or better classifiers, is an open and interesting direction for future work.