

UNIVERSITY CLERMONT AUVERGNE  
DOCTORAL SCHOOL  
SCIENCES FOR ENGINEER OF CLERMONT-FERRAND  
THESIS  
Defended by  
**Vanel Steve SIYOU FOTSO**  
To obtain the title of  
**DOCTOR OF UNIVERSITY**  
**SPECIALITY : COMPUTER SCIENCE**

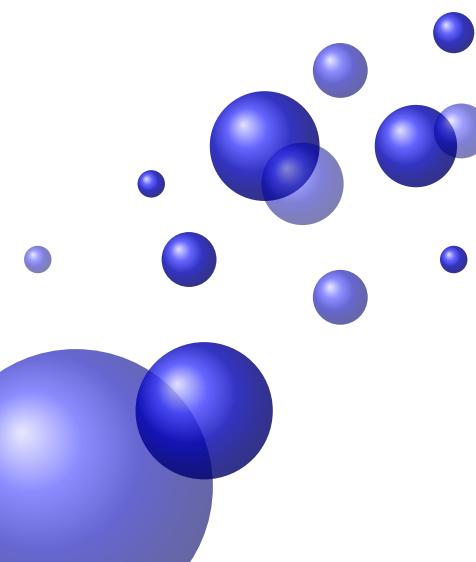
---

**EXTRACTION OF KNOWLEDGE FROM  
CYCLIC AND UNCERTAIN TIME SERIES:  
APPLICATION TO MANUAL  
WHEELCHAIR LOCOMOTION ANALYSIS**

---

Jury :

September 15, 2018





UNIVERSITY CLERMONT AUVERGNE  
DOCTORAL SCHOOL  
SCIENCES FOR ENGINEER OF CLERMONT-FERRAND  
THESIS  
Defended by  
**Vanel Steve SIYOU FOTSO**  
To obtain the title of  
**DOCTOR OF UNIVERSITY**  
**SPECIALITY : COMPUTER SCIENCE**

---

**EXTRACTION OF KNOWLEDGE FROM  
CYCLIC AND UNCERTAIN TIME SERIES:  
APPLICATION TO MANUAL  
WHEELCHAIR LOCOMOTION ANALYSIS**

---

Jury :

September 15, 2018

To my parents Mr. and Mrs. Liyou

# Acknowledgment

First of all, I thank my supervisors, Pr. Engelbert Mephu Nguifo and Dr. Philippe Vaslin for following my work since Master 2. For their high availability, their support and for all the educational exchanges we had. Beyond the thesis work, they are admirable people whose qualities positively influence my person. Thank you very much for all.

I thank LIMOS, which has allowed me to carry out these years of work in a stimulating environment and for having placed at my disposal everything necessary for the realization of this work. I thank the Ministry of Higher Education, Research and Innovation for the support it gives to the research and the researcher in general, and for having financed my thesis work.

I thank the members of my thesis jury for having accepted to take part in this jury. I thank them for the time they devoted to reading the manuscript, for their questions and their remarks which will allow me to improve the quality of this work.

A special thanks to my teachers in general, to those in particular with whom I exchanged during this work; I think of Libo Ren and Violaine Antoine. I also thank Séverine Miginiac and Béatrice Bourdieu for their kindness and their availability.

I spent pleasant moments with my doctoral colleagues from LIMOS whom I would like to thank. The list is too long to mention them all, I will mention Damien Lamy with whom I did the master, Maxime Chassaing and Benjamin Vincent who welcomed me the first year of my thesis, Benjamin Dalmas for our informative discussions and for his good humour, Angeline Plaud, Nestor Koueya, Romain, Jerry Lonlac, my colleagues and friends from MINERS, Nina Bekono and Raissa Saleu my little sisters. A special thank you also to my colleagues in the office with whom we made the world over coffee or a meal, I think of Jinhua Zhao, Jocelyn de Goer, Suan Tay, Danh Nguyen, Ismaela Seck, Rachid Jijiane, Jean Conier, Alexis Cornet, Carlos Cepeda.

Finally, I thank my family for their support, encouragement and wise advice. I think of the Siyou's Family, Njomgang's Family, Ngainsi's Family and also my whole big family. I especially thank my beloved Lucia and our son Luka for all the love and affection they bring to me.



# Contents

<b>Dedications</b>	ii
<b>Acknowledgment</b>	iii
<b>I General introduction</b>	1
<b>II Background and related work</b>	7
<b>1 Knowledge discovery on time series</b>	9
1.1 Introduction . . . . .	9
1.2 Preprocessing of time series . . . . .	9
1.2.1 Denoising time series . . . . .	9
1.2.2 Reducing uncertainty . . . . .	11
1.2.3 Dimensionality reduction . . . . .	11
1.3 Similarity Measures . . . . .	12
1.3.1 Time-Series Invariances . . . . .	13
1.3.2 Categories of time series similarity function . . . . .	14
1.4 Datamining task on time series . . . . .	15
1.5 Conclusion . . . . .	24
<b>III Contributions</b>	25
<b>2 Preprocessing of time series</b>	27
2.1 Introduction . . . . .	27
2.1.1 Why the use of PAA can improve alignment with Dynamic Time Warping . . . . .	28
2.1.2 The problem of choosing a suitable segment number for PAA .	29
2.1.3 Summary of Contributions . . . . .	29

2.2	Background and related works . . . . .	30
2.2.1	Dynamic Time Warping algorithm. . . . .	31
2.2.2	Iterative Deepening Dynamic Time Warping . . . . .	31
2.3	FDTW: a GRASP based heuristic . . . . .	33
2.3.1	Evaluation procedures for the compression quality . . . . .	33
2.3.2	Problem definition. . . . .	34
2.3.3	Brute-force search. . . . .	34
2.3.4	Greedy Randomized Adaptive Search Procedures . . . . .	35
2.3.5	Parameter free heuristic . . . . .	35
2.4	Experiment and results . . . . .	38
2.4.1	Case studies . . . . .	38
2.5	Conclusion and perspective . . . . .	47
<b>3</b>	<b>Uncertain time series u-shapelet discovery</b>	<b>49</b>
3.1	Introduction . . . . .	49
3.1.1	Review of u-shapelets . . . . .	50
3.1.2	U-shapelets algorithm for clustering Uncertain Time Series . . . . .	52
3.1.3	Uncertainty and u-shapelets discovery issue . . . . .	52
3.1.4	Summary of contributions . . . . .	52
3.2	Background . . . . .	53
3.2.1	Background . . . . .	53
3.2.2	State of the art on uncertain similarity functions . . . . .	54
3.3	FOTS . . . . .	62
3.3.1	Dissimilarity function . . . . .	62
3.3.2	Scalable u-shapelets Algorithm with FOTS score . . . . .	63
3.4	Experiments . . . . .	64
3.4.1	Clustering with u-shapelets . . . . .	64
3.4.2	Evaluation Metrics . . . . .	65
3.4.3	Comparison with u-shapelet . . . . .	65
3.4.4	Comparison with k-Shape and USLM . . . . .	66
3.4.5	Discussion . . . . .	66
3.5	Conclusion . . . . .	67
<b>4</b>	<b>SAX-P</b>	<b>71</b>
4.1	Introduction . . . . .	71
4.2	SAX-P . . . . .	74
4.2.1	Segmentation of cyclic time series . . . . .	75
4.2.2	From cycles to letters . . . . .	76
4.3	Application to manual wheelchair locomotion . . . . .	78
4.3.1	Dataset description . . . . .	78
4.3.2	The symmetry of Manual Wheelchair Locomotion . . . . .	80
4.3.3	Group Manual Wheelchair users according to their motor skills	86
4.4	Conclusion . . . . .	91

<b>General conclusion</b>	<b>94</b>
<b>A Wheelchair locomotion</b>	<b>95</b>
A.1 Introduction . . . . .	95
A.2 Problem . . . . .	95
A.3 Evaluation tools . . . . .	96
A.3.1 Crank Ergometers . . . . .	96
A.3.2 Roller Ergometers . . . . .	97
A.3.3 Treadmill . . . . .	99
A.3.4 Wheelchair simulators . . . . .	99
A.3.5 Wheelchair Field-Ergometer . . . . .	100
A.4 Wheelchair time series . . . . .	103
A.5 Conclusion . . . . .	106
<b>B An optimal approach for time series segmentation: Application to the supervised classification</b>	<b>107</b>
B.1 Introduction . . . . .	107
B.2 Granularity of time series segments . . . . .	108
B.2.1 Notations and definitions . . . . .	108
B.2.2 Information theory and minimum number of segments . . . . .	109
B.2.3 Notations and definitions . . . . .	109
B.2.4 Minimize the squared error to choose the number of segments	110
B.2.5 Dynamic Time Warping Algorithm and comparison of compact representations . . . . .	112
B.3 Results and Discussion . . . . .	113
B.3.1 Datasets . . . . .	113
B.3.2 Comparison of algorithm performance . . . . .	115
B.4 Conclusion . . . . .	119
<b>C Determination of the probability density function of Mz uncertainty</b>	<b>121</b>
C.1 Ampiric probability distribution . . . . .	121
C.2 Theoric probability distribution . . . . .	122
<b>D Training and changes on propulsion technics</b>	<b>125</b>
<b>References</b>	<b>146</b>



# Part I

## General introduction



## Context

In almost every scientific field, measurements are performed over time. These observations lead to a collection of organized data called time series. Today time series data are being generated at an unprecedented speed from almost every application domain, e.g.:

- In astronomy, telescopes scan the sky and capture light rays that are used in the study of the universe. In Large Synoptic Survey Telescope (LSST) project [lss, 2016], telescopes will capture the electromagnetic radiation of the sky during ten years to calculate the acceleration of the expansion of the universe. This will result in an astronomical catalog of time series.
- In paleoecology, scientists study the evolution of living animal and plant species in the past. To do this, they extract cores from the soil and look for the presence of fossils at each depth, thus creating time series representing the growth or decline in the size of fauna and flora populations over time [Lonlac et al., 2018].
- In medicine, the analysis of electrocardiogram is used to prevent heart attacks [Ding, 2011]. Those electrocardiograms are long time series obtained by recording the electrical activity of the heart over a period.
- In biomechanics, the study of human locomotion is performed using sensors that record the efforts performed and the movements of the body during the locomotion.

As a consequence, in the last decade there has been a dramatically increasing amount of interest in querying and mining such data.

## Issues

Time series data mining unveils numerous facets of complexity. The most prominent problem arises from the uncertainty contained in time series data, the difficulty of defining a form of similarity measure based on human perception, and the high dimensionality of time series data. These constraints show us that three major issues are involved :

- Data representation. How can the fundamental shape characteristics of a time series be represented? What invariance properties should the representation satisfy? A representation technique should derive the notion of shape by reducing the dimensionality of data while retaining its essential characteristics.

- Similarity measurement. How can any pair of time series be distinguished or matched? How can an intuitive distance between two series be formalized? This measure should establish a notion of similarity based on perceptual criteria, thus allowing the recognition of perceptually similar objects even though they are not mathematically identical.
- Uncertainty. How to compare the shape of time series without knowing their exact value? How to measure the impact of uncertainty contained in time series or to reduce the adverse effects of uncertainty?

The aim of our work is to propose algorithms to deal with those characteristics of time series.

## Context of the thesis

This thesis apprehends these scientific questions from a data mining point of view, within the framework of the analysis of time series coming from Manual Wheelchair (MWC) locomotion. Also, even if the issues addressed are not limited to the field of biomechanics time series and concern other areas of applications, this thesis will deal with the analysis of time series coming from the wheelchair ergometer FRET-2.

For improving the mobility of persons confined to manual wheelchairs, it is necessary to be able to assess people in their daily environment. For this purpose, a field ergometer wheelchair (FRET-1) has been designed and manufactured [[Dabonneville et al., 2005](#)]. This ergometer is equipped with moment sensors that measures the forces applied to the handrims as well as the movement of the FRET-1 [[Couétard, 2000](#)]. It, therefore, makes it possible to measure and calculate a large number of the mechanical parameters of manual wheelchair locomotion.

However, the time series produced by this moment sensor have specific characteristics:

- they are long because of the acquisition frequency of the sensor (between 80 and 100 Hz),
- they are cyclic; these cycles come from the cyclical character of the locomotion in Manual Wheelchair, which consists of a succession of push phases and recovery (or freewheeling) phases,
- they are uncertain, this uncertainty is observed during the calibration of the sensor.

Our work consists of proposing algorithms to extract relevant information from these time series while taking into account their characteristics. The methods developed in this work have the aim to assist practitioners for the analysis of Manual Wheelchair locomotion; then, special attention will be given to the readability and ease of interpretation of the results provided by the algorithms.

# Plan

The thesis is organised as follows:

- **Chapter 1** explains existing models in the field of time series processing that could be useful for the analysis of Wheelchair locomotion. Thus, chapter 1 presents strategies for the preprocessing of time series (e.g. noise reduction, length reduction), their comparisons, their exploitation through visualization, classification, clustering or prediction.
- **Chapter 2** introduces an algorithm called FDTW, which aims to reduce the length of time series while preserving the information it contains. Its operating principle is based on that of GRASP, but it is original in that it defines its global search strategy. Experiments conducted on a classification task have shown that compression does not alter classification performance.
- **Chapter 3** proposes a novel framework for uncertain time series clustering, which is based on the use of a clustering algorithm (UShapelet), and on the use of a dissimilarity function (FOTS) the both are robust to the presence of uncertainty in time series. We tested this clustering strategy on 17 data sets from the literature, which allowed us to observe an improvement in the quality of the obtained results.
- **Chapter 4** presents a novel symbolic representation of cyclic time series based on cycle properties, which we use for the analysis of cyclic time series issued by human locomotion. This symbolic representation facilitates the visualization and evaluation of cyclic time series. This chapter also gives some applications of the proposed algorithms to data from manual wheelchair locomotion. The results allowed us to measure the asymmetry of wheelchair locomotion and to establish that this asymmetry decreases with years of practice. We have also observed that the propulsion capabilities of wheelchair users with similar levels of spinal cord injury may differ. These two results highlight the importance of monitoring manual wheelchair locomotion using measuring instruments.



## Part II

### Background and related work



# Chapter 1

## Knowledge discovery on time series

### 1.1 Introduction

Datasets can be grouped into four main categories regarding their temporality [Roddick and Spiliopoulou, 2002]:

- Static datasets: these are datasets with no temporal context. We have for example the radius of a wheel, the circumference of a circle, the gravity in a place.
- Sequences datasets: they consist of ordered sequences of events. This category includes an order but not time. As an example, we can cite a DNA sequence (GTTTTCCCAGTCACGAC).
- Time-indexed datasets: they consist of a set of temporal data sequences ; for example a set of measures taken at a more or less regular time interval.
- Full-time data: Each tuple has one or more time components; time series belongs to this latter category.

Time series have several characteristic properties: usually, they are noisy, uncertain and they often have high dimensionality and high auto-correlation. Each of those features can interfere with the mining of time series. To remedy this, preprocessing technics have been proposed in the literature.

### 1.2 Preprocessing of time series

#### 1.2.1 Denoising time series

Several filters have been proposed in the literature to remove noise contained in time series. In this section, the most frequently used filters are presented.

**Kernel smoothing:** this filter refers to a statistical technique for recovery of underlying structure in data sets. Its basic principle is to estimate a real-valued function as the weighted average of neighboring observed data. The weight is defined by a function named kernel, such that closer points to real values are given higher weights [Wand and Jones, 1994].

**Polynomial Regression:** this filter consists in fitting a nonlinear relationship between the values of an independent variable  $x$  (predictor variable) and the corresponding conditional mean of  $y$  (variable to explain), denoted  $E(y|x)$ . This filter has been used to describe nonlinear phenomena. More formally, polynomial regression is defined as the problem of finding a polynomial:  $g(x) = \beta_0 + \beta_1x + \dots + \beta_mx^m$  of a certain degree  $m$  for which  $E(Y - g(x))^2$  is as small as possible [Kendall, 1961].

**Wiener-Kolmogorov Filtering of Short Stationary Sequences:** The idea of this filter is to produce a statistical estimate of the actual signal from the noisy signal. Using the Wiener-Kolmogorov filter assumes the knowledge of stationary signal, noise spectra, and additive noise [Pollock, 2007].

**Filtering in the Frequency Domain:** The purpose of frequency-based filters is to remove the noise contained in a signal. To achieve this goal, the signal is initially broken down into a set of frequencies using a Fourier transform. This set of frequencies is called the signal spectrum. Depending on the application, it may be appropriate to suppress high or low frequencies, or both, to suppress signal noise. These filters are generally named low-pass, high-pass, bandpass, or notch filter. These filters can also be combined in many ways: in cascade, in parallel, etc [Buttkus, 2012].

**Kalman Filter and the Smoothing Algorithm,** also known as linear quadratic estimation (LQE), is a Bayesian estimation technique used to track stochastic dynamic systems being observed with noisy sensors. The filter produces estimates of unknown variables that tend to be more accurate than those based on a single measurement alone, by estimating a joint probability distribution over the variables for each timeframe. The algorithm works in two phases: extrapolation (prediction) and update (correction). In the extrapolation step, the Kalman filter produces estimates of the current state variables, along with their uncertainties, based on the previous state variables and their uncertainties. Once the outcome of the next measurement is observed, these estimates are updated using a weighted average, with a higher weight being given to estimates with higher certainty. The algorithm is recursive. It can run in real time, using only the current input measurements and the previously calculated state and its uncertainty matrix [Matthies et al., 1989].

### 1.2.2 Reducing uncertainty

Another important step of preprocessing time series is to reduce the uncertainty that they contained. For this purpose some transformations have been introduced in literature.

**Uncertain moving average:** For uncertain time series, each value is associated with a standard deviation representing uncertainty. The uncertain moving average (UMA) filter is then defined as the weighted average of the consecutive data points of a time series over a given time interval. The weights at each timestamp  $i$  are calculated from the inverse of the uncertainty. Thus, in the calculation of the mean, a weight ( $w$ ) inversely proportional to the uncertainty will be given to each data point in the time series. Uncertain moving average returns times series:  $x^{UMA} = < x_1^{UMA}, \dots, x_m^{UMA} >$  for which  $x_i^{UMA} = \frac{1}{2w+1} \sum_{k=i-w}^{i+w} \frac{x_k}{\sigma_k}$ ,  $1 \leq i \leq m$  [Orang and Shiri, 2015].

**Z-normalization** is used with uncertain moving average to reduce the advert effect of uncertainty in time series. In general, z-normalization improves similarity search quality, because it makes similarity measures invariant to scaling and shifting. Given an uncertain time series:  $X = < X_1, \dots, X_m >$ , its normal form:  $\hat{X} = < \hat{X}_1, \dots, \hat{X}_m >$  is defined as follows:

$$\hat{X}_i = \frac{X_i - \bar{X}}{S_X},$$

where  $\bar{X}$  and  $S_X$  denote the sample mean and standard deviation of expected values of  $X$ , respectively [Orang and Shiri, 2015]. That is,

$$\bar{X} = \frac{1}{n} \sum_{i=1}^n E(X_i),$$

$$S_X = \sqrt{\frac{1}{(n-1)} \sum_{i=1}^n (E(X_i) - \bar{X})^2}.$$

### 1.2.3 Dimensionality reduction

Time complexity of a mining time series algorithm depends on the length of the time series. Reducing dimensionality of time series allows reducing their processing time. To achieve this goal, many representations have been proposed and can be grouped into three main categories:

**Non-data-adaptive:** Dimension reduction methods are called non-data-adaptive because they take parameters of which value does not vary according to the considered data set. One of the first work in this family was done by Agrawal [Agrawal et al., 1993] who used a Discrete Fourier Transform to compress time series. In the same family, we can also cite the following time series representations: Discrete Wavelet Transform (DWT) [Chan and Fu, 1999a], Piecewise Linear Approximation (PLA) [Eriksson et al., 2004], Piecewise Aggregate Approximation (PAA) [Keogh et al., 2001a].

**Data adaptive:** This family of time series representation consists of methods that take the properties of the dataset into account when choosing the method parameters. All non-data-adaptive representations can be transformed into data-adaptive representations by adding a parameter selection method to them. As examples of data-adaptive representations, there is Adaptive Piecewise Constant Approximation (APCA) [Keogh et al., 2001c] and Singular Value Decomposition (SVD) [De Lathauwer et al., 1994] and Symbolic Aggregate Approximation (SAX) [Lin et al., 2003].

**Model based:** The assumption here is that time series are described by an underlying model. Dimensionality reduction is achieved by identifying the model parameters that generate the time series. Several approaches use temporal parametric models such as statistical modeling by feature extraction [Esling and Agon, 2012], Auto Regressive Moving Average (ARMA) models [Kalpakis et al., 2001], Markov Chains (MCs) and Hidden Markov Models (HMM) [Panuccio et al., 2002].

After cleaning the time series, we are now ready to extract relevant information from them. Several data mining tasks can be performed with time series.

## 1.3 Similarity Measures

Before performing data mining tasks, it is essential to be able to compare time series. Most often, similarity functions compare time series as humans would do. Indeed, without much of stretch, human recognition understand and look at the likenesses between two time series based on their amplitude, scale, temporal warping, noise, and outliers. As indicated by [Fu, 2011] [Ralanamahatana et al., 2005] [Esling and Agon, 2012], any similarity measure for time series comparison ought to be reliable with human recognition and perception and have the following properties:

- It should perceive perceptually comparative datasets even if they are not mathematically identical;

- It should resemble human intuition;
- It should be able to capture global and local similarities;
- It should have a universal meaning that is not restricted to particular type of time series datasets and do not assume some constraints on time series data;
- It should be robust to distortions and set of transformations. More specifically it should be robust to amplitude shifting, uniform amplification, uniform time scaling, dynamic amplification, dynamic time scaling, adding noise and outliers transformations or any combination of these transformations.

The latter property is also known as invariance.

### 1.3.1 Time-Series Invariances

In this section, we review common time-series distortions and their invariances. More detailed information can be found in [Batista et al., 2014].

**Scaling and translation invariances:** We should be able to perceive the similarity of sequences in spite of contrasts in amplitude (scaling) and offset (translation). For instance, these invariances may be helpful to analyze seasonal variations in currency values on foreign trade markets without being biased by inflation.

**Shift invariance:** We should be able to recognize two similar sequences even if they vary in phase (global alignment) or when there are regions of the sequences that are aligned and others are not (local alignment). For instance, heartbeats can be out of phase depending on when we start recording, and handwritings of the same sentence from various people will require alignment depending on the size of the letters and on the spaces between words (local alignment).

**Uniform scaling invariance:** We should be able to compare two sequences even if they have different length. To do so, sequences that differ in length require either extending of the shorter sequence or, contracting of the longer sequence. For instance, this invariance is required for heartbeats recorded at different sample frequencies (e.g.: 10, 50 ou 100 Hz).

**Occlusion invariance:** We should be able to compare two time series even if some of their sub-sequences are missing; we can also compare the sequences by ignoring the sub-sequences that do not match well. For example, suppose an archaeologist who has just found a skull in a research site, and would like to determine to which species this skull belongs. Let us also suppose that we have a database of time series corresponding to the skulls of living species. We could then compare the time series

from the found skull to those stored in the database. This comparison should be possible even if the found skull is damaged. In other words, we should be able to make a comparison even if the time series extracted from the found skull has missing sub-sequences.

**Complexity invariance:** We should be able to recognize time series with similar shape even if they have different complexity. For example, the same audio signals that were recorded indoors and outdoors might be considered similar, although outdoor signals will surely be noisier than indoor ones.

Depending on the application domain, some or all the invariances can be required for the comparison of time series. The preprocessing step can handle some of those invariances; for instance, z-normalization of time series allows their comparison to be scaling invariant. However, all invariances cannot be handled by preprocessing step and should then be considered by more sophisticated distances or dissimilarities functions. In the next section, we review the most common of such distance measures.

### 1.3.2 Categories of time series similarity function

Time series similarity measures can be generally divided into following four main categories:

**Based similarity function** compares two time-series based on the sum of the distances in a Euclidian space between data points of both times series located at approximately the same timestamp. By doing so, the distance between two time series with similar shapes will be low. On the opposite, the distance between time-series that have a different shapes will be high. In this family, there are  $L_p$  norm [Yi and Faloutsos, 2000] [Keogh and Kasetty, 2003a], and Dynamic time warping distance [Myers et al., 1980b] for instance. However, those distances are sensitive to noise.

**Edit Based distance** allows evaluating the dissimilarity between two character strings. These dissimilarity functions are able to handle noisy regions and outliers. The principle of these similarity functions is the following: Edit based distances count the minimum number of operation necessary to transform a character string to another. Different edit based dissimilarity functions use different operations to transform one string to another. A well known edit based distance is Levenshtein distance. That uses three operations: suppression, insertion and substitution of letters. Edit based distances in time series domain are based on the same principle. Time series data points can be skipped during the comparison (deletion) or one data point can be compared to several data points of the other time series (insertion). Among

the well-known edit based distances in time series domain we can cited: Longest Common SubSequence (LCSS) [Das et al., 1997], Edit Distance on Real sequence (EDR) [Chen et al., 2005] and Time Warp Edit Distance (TWED) [Marteau, 2009] algorithms. LCSS distance uses a threshold parameter for point matching as well as a warping threshold for allowing gaps for matching two time series. EDR is a variant of the edit distance for real-valued series. Opposite to LCSS, EDR assigns penalties based on the length of existing gaps between two series. TWED is a dynamic programming algorithm that introduces a parameter to control the elasticity measure along the time axis.

**Feature Based distance:** this distance has been designed to ensure some invariances such as rotation invariance. Time series can be compared based on their properties rather than on their shape. So, Feature Based similarity measures compare two time series by computing a feature set for each time series that reflects their properties<sup>1</sup>. For example, DFT and DWT coefficients can be used to compare the similarity between time series [Shatkay and Zdonik, 1996].

**Structure Based similarity measures:** These measures are designed to compare time series on a global scale based on their structure. The general principle of those similarity functions is to compare time series based on a high-level representation that captures global properties of the time series, such as histogram, for instance [Lin and Li, 2009].

## 1.4 Datamining task on time series

**Indexing time series:** The problem of indexing or query by content can be defined as follows: given a query time series  $Q$ , and some similarity/dissimilarity measure  $D(Q,C)$ , find the most similar time series in database DB. When querying time series by content, a challenge consists in finding as fast as possible a time series in the database that is similar to the query. To achieve this goal, some dimensionality reduction technics have been used: for instance, in [Agrawal et al., 1993], time series have been transformed into a more compact representation using DFT before their comparison. Many other dimensionality reduction techniques have been used for the same purpose, such as Discrete Wavelet Transform (DWT) and Discrete Cosine Transform (DCT) [Chan and Fu, 1999a]. Other representation approaches used for query by content are PLA, PAA, APCA [Keogh et al., 2001c], and SAX [Lin et al., 2007]. These latter authors [Lin et al., 2007] have shown that SAX outperforms other representations for query by content applications. Another strategy used to ensure the effectiveness of research is parallelism [Yagoubi et al., 2017].

---

<sup>1</sup>We will use this type of distance later when analyzing manual wheelchair locomotion (Chap. 4).

**Motif Discovery:** Time series motifs are pairs of individual time series, or subsequences of a longer time series, which are very similar to each other and carry precise information about the underlying source of the time series. The idea for motif discovery in time series is inspired from DNA analysis. When they exist, motifs can be used to construct meaningful clusters when clustering time series, which is the case of unsupervised shapelet algorithm [Ulanova et al., 2015]. Associating each class with a motif can speed-up the classification of time series; this idea is used by the shapelet transform algorithm<sup>2</sup> [Lines et al., 2012], [Yagoubi et al., 2018].

**Anomaly Detection:** Anomaly detection refers to the problem of finding patterns in data that do not conform to the expected behavior. These nonconforming patterns are often referred to as anomalies, outliers, discordant observations, exceptions, aberrations, surprises, peculiarities, or contaminants in different application fields. Among these, anomalies and outliers are two terms most commonly used in the context of anomaly detection; sometimes interchangeably. Anomaly detection finds extensive use in a wide variety of applications such as fraud detection for credit cards, insurance, or healthcare, intrusion detection for cyber-security, fault detection in safety-critical systems, and military surveillance of enemy activities [Chandola et al., 2009].

**Temporal Association Rule Discovery:** In a transactional database, association rules allow searching for items that often appear together in the same transaction. For instance, in the database of a Shop, the discovered rules will indicate, which products are often bought together. The association rules do not give any information on the precedence of the occurrence of one event concerning the other. Hence the need to define temporal association rules, which are particularly appropriate as candidates for causal rules' analysis in temporally adorned medical data, such as in the histories of patients' medical visits. Patients are associated with both static properties, such as gender, and temporal properties, such as age or current medical treatments, any or all of which may be taken into account during mining [Vasimalla, 2017].

**Summarization (Visualization):** The problem of time series visualization or summarization can be defined as follows: given a time series  $Q$  containing  $n$  data points where,  $n$  is an extremely large number, create a (possibly graphics) approximation of  $Q$  which retains its essential features but fits on a single page, computer screen, executive summary. Summarization can be viewed as a higher level clustering of time series where clusters are associated with text or graphical descriptions. Some famous approaches of time series summarization are:

---

<sup>2</sup>We will use this type of distance later when analyzing manual wheelchair locomotion (Chap. D ).

- **Time searcher:** it is a query by content summarization tool. Here, a user specifies a set of constraints (intervals) graphically to which time series data points should belong. Those constraints are called time series boxes [Hochheiser and Shneiderman, 2003].
- **Calendar based visualization** of univariate time series data: its goal is to simultaneously identify patterns and trends on multiple time scales (days, weeks, seasons). To do so, Calendar based visualization first clustered similar daily data patterns and visualized the average patterns as graphs and the corresponding days on a calendar [Van Wijk and Van Selow, 1999].
- The **spiral visualization** is appropriated with large data sets and supports much better than line graphs the identification of periodic structures in the data. Spiral visualization supports both the visualization of nominal and quantitative data based. The extension of the spiral visualization to 3D gives access to concepts for zooming and focusing and linking in the data set [Weber et al., 2001].
- **GrammarViz** is a visualization tool that allows efficient discovery of frequent and rare patterns of variable length in time series. It is based on the symbolic representation of time series sax and context-free grammar [Senin et al., 2014].

**Prediction** or time series forecasting is one of the most useful data mining tasks on time series: for example, time series forecasting is used to predict the weather, the cost of an action in the stock exchange market, or early identified epidemiological risks and raised up alarms. A time series forecasting method is based on a mathematical model that capture the main characteristics of the time series like seasonality, periodicity, trend, and that can be used to guess unknown (or future) values of the time series. Many other algorithms used for time series forecasting are based on Auto-Regressive (AR) models. More sophisticated approaches are also used such as neural networks and cluster function approximation [Mahalakshmi et al., 2016].

**Classification:** Classified time series consists of assigning an unlabelled time series to one, two or more classes. Many classification algorithms for time series have been proposed in the literature and can be gathered into four main groups

- **Dictionary classifiers:** generally, these classifiers first transform time series into characters strings that can be decomposed into a set of word or bag of words, a word being simply a subsequence of the characters string. Each time series is then described by the occurrence frequency of each word in it. The set of time series represented in the space of words is called a dictionary; The classification of time series is then based on the presence or absence of words in this dictionary. Several algorithms of the literature are

based on this principle, such as Bag of Patterns [Lin et al., 2012], SAX and Vector Space Model [Senin and Malinchik, 2013], Bag of SFA Symbols(BOSS) [Schäfer, 2015], DTW Features[Kate, 2016a], temporal-robust text classification [Salles et al., 2017] .

- **Classifier-based on the alignment of whole time series:** those classifiers are based on distance functions that operate over the entire length of the time series. The difference between the classifiers of this family is based in part on the characteristics of the distance functions used. These distance functions can be based on the shape of the time series (Derivative Dynamic Time Warping [Keogh and Pazzani, 2001a], Weighted Dynamic Time Warping [Jeong et al., 2011], Complexity-Invariant Distance [Batista et al., 2011]), on their properties, on their structures or on their symbolic representation (Time Warp Edit Distance [Marteau, 2008], Move Split Merge [Stefan et al., 2013]).
- **Shapelets Classifiers:** Unlike classifiers based on the comparison of the time series over their entire length, shapelets classifiers look for characteristic subsequences in time series called shapelet whose presence or absence indicates whether or not a time series belongs to a class. We have for example: Shapelet Transform [Lines et al., 2012], Learned Shapelets [Grabocka et al., 2014], Fast Shapelet Tree [Rakthanmanon and Keogh, 2013]
- **Intervals Classifiers:** The idea here is to find localized discriminatory features on time series based on some statistical properties calculated over intervals of variable length. A time series of length  $m$  will have  $m(m-1)/2$  possible contiguous intervals. An interval associated with some statistical properties and a condition is a literal, which gives some information about what happened in an interval: for instance, is the mean of a sequence of data points greater or less than a define threshold? The classifier tries to find a relationship between what happened in an interval and time series classes. Many classifiers are based on this principle, such as: Time Series Bag of Features [Baydogan et al., 2013], Time Series Forest [Deng et al., 2013], Learned Pattern Similarity [Baydogan and Runger, 2016].

**Clustering:** The clustering of time series consists of grouping them to build very homogeneous and well-separated groups under some similarity/dissimilarity measure  $D(Q, C)$  [Rani and Sikka, 2012]. "Homogeneous" means that the intra-group variance is small and "Well separated" means that the inter-groups variance is high. There are many ways to categorize time series clustering algorithms depending on the **distance function** used, the **data transformation** or the **clustering strategy**.

When considering **distance function**, we have two categories of time series clustering algorithms: those that operate on the whole time series and those that operates on a sub-sequences of time series.

When considering **data transformation**, we can gather time series clustering algorithms into three groups: raw data, feature-based and model-based clustering.

When considering **clustering strategy**, we have five categories of clustering algorithms:

- **Distance-based** clustering, which is divided in two sub-categories:
  - **Partitioning clustering algorithms** partition the data in high dimensional space into multiple clusters. We have for example kMeans like algorithms (kMeans, kMedians, kMedoids, XMeans, KMeans++)
  - **Hierarchical clustering algorithms** are grouped into two subcategories: **Agglomerative clustering algorithms** first consider each object of the dataset as a cluster and then try to merge clusters until obtaining one cluster: it is a bottom-up merging strategy. **Divisive clustering algorithms** first considers that all the data points are in the same cluster and then try to split this cluster to obtain more homogenous ones: it is the top-down merging strategy.
- **Density-Based clustering and grid-based clustering algorithm:**
  - The principle of **density-based clustering** is the following: given a time series that will be considered as the center of the cluster, all the time series of the database that have a distance less or equal to a defined threshold to the center of the cluster are gathered. Thus, the algorithm splits the space into more or less dense regions; then small dense regions can be merged into more significant regions. This algorithm allows to identify clusters of arbitrary shapes [Kriegel et al., 2011].
  - **Grid-based clustering** divides the data space into a grid-like structure, which allows determining the characteristics of the data [Amini et al., 2011].
- **Probabilistic and generative models** can be modelled with a generative process assuming the data follow a particular distribution like a mixture of Gaussian. Then the model parameters are estimated using the expectation-maximization algorithm (EM), that consider the parameters that maximize the likelihood of the model to the data. On this basis we may estimate the generative probabilities that will be used to construct the generative model [Merugu and Ghosh, 2003].
- **High-dimensional clustering algorithms:** time series may be set in a high dimensional feature space. To cluster them, many methods have been proposed:
  - **Subspace clustering:** Subspace clustering looks for a cluster in different subspaces of a dataset. A subspace is a subset of the  $d$  dimensions

of a given dataset; all the dimensions of high dimensional data are not useful. Subspace clustering algorithm identifies relevant dimensions allowing them to find clusters. There are two main subspace clustering branches based on their search strategy. Top-down algorithms find an initial clustering in the full set of dimensions and evaluate the subspaces of each cluster, iteratively improving the results. Bottom-up approaches find dense regions in low dimensional spaces and combine them to form clusters [Parsons et al., 2004].

- **Dimensionality reduction:** many dimensionality reduction techniques have been proposed for clustering purpose. A well-known one is co-clustering, which consists of simultaneously clustering columns (or dimensions) and rows (data points) of a matrix [Dhillon et al., 2003].
- **Probabilistic latent semantic indexing (PLSI) and latent dirichlet allocation (LDA)** are typical clustering techniques for text data. Indeed, text can be clustered in multiple topics and each topic can be associated with a set of words (or dimension) or a set of rows (documents) can be simultaneously associated [Hofmann, 2017].
- **Nonnegative matrix factorization** is a kind of co-clustering algorithm. It proceeds as follows: a nonnegative matrix  $X \in \mathbb{R}^{M \times N}$  can be factorised into two lower rank matrices  $U \in \mathbb{R}^{M \times L}$  and  $V \in \mathbb{R}^{L \times N}$  with  $L < M$  and  $L < N$ . The idea here is to identify clusters using the matrix  $U$  which has a lower dimension than  $X$  [Wang and Zhang, 2013].
- **Spectral clustering:** the principle is to cluster time series or data objects based on the spectrum of their similarity matrix. The spectrum being used here for dimensionality reduction [Filippone et al., 2008].

In the context of this thesis, the analysis of manual wheelchair locomotion implies to grouping manual wheelchair users with similar motor abilities, based on measurements made during their locomotion in actual conditions: this approach is equivalent to time series clustering. Several clustering algorithms proposed in the literature are harmonious compositions, consisting of a representation of time series, an adequate distance function and an appropriate clustering strategy as illustrated in the Tables 1.1, 1.2 and 1.3. Detailed informations is presented in [Rani and Sikka, 2012].

Table 1.1: Temporal-Proximity-Based Clustering Approach

Paper	Distance Measure	Algorithm	Application
M. Kumar	Based on the assumed independent Gaussian models of data errors	Agglomerative Hierarchical	Seasonality pattern in retail
T.-W. Liao	Euclidean and symmetric version of Kullback–Liebler distance	K-Means and Fuzzy C-Means	Battle simulations
T.-W. Liao	Dynamic Time Warping	K-Medoids Based Genetic Clustering	Battle simulations
C.S. Möller-Levet	Short time series (STS) distance	Modified Fuzzy C-Means	DNA microarray
Shumway	Kullback–Leibler discrimination information measure	Agglomerative Hierarchical	Earthquakes and mining explosions
Vit Niennattrakul	Dynamic Time Warping	K-Means, K-Medoids	Multimedia time series
Pooya Sobhe Bidari	Pearson Correlation	K-Means, Fuzzy C-Means	Pattern extraction in genes
Hardy Kremer	Dynamic Time Warping	Density Based Subsequence Clustering	Detecting climate change
Jian Yin	Grey Relation	Hierarchical Clustering	Change trend of traffic flow data
S. Chandrakala	Euclidean	Kernal DBScan	Multivariate time series clustering
Aurangzeb Khan	Euclidean	K-Mean+ MFPP(Most Frequent Pattern)	Stock and inventory data
Mengfan Zhang	CVT(Computational Verb Theory)	K-Means	Stock market data
S.R.Nanda	Euclidean	K-Means	Portfolio management
Jianfei Wu	N/A	K-Means	Stock data

Table 1.2: Representation-Based Clustering Approach Paper

Paper	Features	Distance Measure	Clustering Algorithm	Application
T.-C. Fu	Perceptually important points	Sum of the mean squared distance along the vertical and horizontal scales	Modified SOM	Hong Kong stock market
M. Vlachos	Haar wavelet transform	Euclidean	Modified k-means	Non-specific
Huiting Liu	Empirical mode decomposition	Euclidean	Forward propagation learning algorithm	Non-specific
Chonghui GUO	Independent component analysis	Euclidean	Modified k-means	Real world stock time-series
Jian Xin Wu	Independent component analysis	N/A	support vector regression	Financial time-series
Geert Verdoolaege	Wavelet transform	Kullback- Liebler divergence	k-means	Detection of activated voxels in fMRI data
Liu Suyi	Hough transform	N/A	Mean shift algorithm	Feature recognition of underwater images
Dong Jixue	Wavelet transform	N/A	Grid-based partitioning method	Financial time-series

Table 1.3: Model-Based Clustering Approach

Paper	Model	Distance measure	Clustering algorithm	Application
Baragona	ARMA	Cross-correlation based	Tabu search, GA and	Non-specific
K. Kalpakis	AR	Euclidean	Partitioning around medoids	Public data
Xiong and Yeung	ARMA mixture	Log-likelihood	EM learning	Public data
L. Wang	Discrete HMM	Log-likelihood	EM learning	Tool condition monitoring
Xin Huang	Fuzzy set and R/S analysis model	N/A	Fuzzy clustering iteration method	Predicting agriculture drought
Shan Gao	ARMA-ARCH	N/A	N/A	To analyze the effects of wind data series

## 1.5 Conclusion

Time series are ubiquitous in science and are more and more used in the analysis of human locomotion. This chapter presents a general framework for extracting knowledge from time series, starting by time series pre-processing, which allows reducing the adverse effects of noise, uncertainty, and dimensionality. Then, we present strategies that are used to compare time series, and we finally present data mining tasks on time series. This chapter presents what has been already done in the literature and raises the question of extracting relevant information from time series coming from wheelchair locomotion. The following chapters present some new strategies that we introduce and that are adapted to characteristics of time series issued from manual wheelchair locomotion.

### Key points

We highlight the different models of data mining present in the literature that could be useful to our work, particularly for

- Pre-processing of time series (noise reduction, dimensionality reduction)
- Comparison of time series.
- The different data mining tasks performed on time series.

# **Part III**

## **Contributions**



# Chapter 2

## Compression for a better classification with Dynamic Time Warping

**Abstract :** *Dynamic Time Warping (DTW) is a time series alignment algorithm that is often used because it considers that it exits small distortions between time series during their alignment. However, DTW sometimes produces pathological alignments that occur when, during the comparison of two time series  $X$  and  $Y$ , one data point of the time series  $X$  is compared to a large subsequence of data points of  $Y$ . In this chapter, we demonstrate that compressing time series using Piecewise Aggregate Approximation (PAA) is a simple strategy that greatly increases the quality of the alignment with DTW. This result is particularly true for synthetic data sets.*

### 2.1 Introduction

Time series databases are often large and several transformations have been introduced in order to represent them in a more compact way. One of these transformations is Piecewise Aggregate Approximation (PAA) [Keogh et al., 2001b], which consists in dividing a time series into several segments of fixed length and replacing the data points of each segment with their averages. Due to its simplicity and low computational time, PAA has been widely used as a basic primitive by other temporal data mining algorithms such as SAX [Lin et al., 2003], SAX-TD [Sun et al., 2014], ESAX [Lkhagva et al., 2006], in order to:

- Construct symbolic representations of time series; [Camerra et al., 2010] [Ulanova et al., 2015].
- Construct an index for time series; [Zhao and Itti, 2016] [Keogh and Pazzani, 2000] [Kate, 2016b]. Indeed, PAA allows queries, which are shorter than length for which the index was built. This very desirable

feature is impossible with Discrete Fourier Transform, Singular Value Decomposition and Discrete Wavelet Transform.

- Classify time series.

### 2.1.1 Why the use of PAA can improve alignment with Dynamic Time Warping

Time series comparison is an important task that can be done in two main ways. Either the comparison method considers that there is no time distortion as in Euclidian distance (ED), or it considers that some small time distortions exist between time axis of time series as in Dynamic Time Warping alignment algorithm (DTW) [Zhang et al., 2015]. Since time distortion often exists between time series, DTW has often better results than ED [Chen et al., 2015]. An exhaustive comparison of time series algorithms [Bagnall et al., 2016a] showed that DTW is among the efficient techniques to be used. However, DTW has two major drawbacks: the comparison of two time series with this algorithm is time-consuming [Rakthanmanon et al., 2012a] and sometimes DTW produces pathological alignments [Keogh and Pazzani, 2001b]. A pathological alignment occurs when, during the comparison of two time series  $X$  and  $Y$ , one datapoint of the time series  $X$  is compared to a large subsequence of datapoints of  $Y$ ; A pathological alignment causes a wrong comparison.

Three categories of methods are used to avoid pathological alignments with DTW:

- The first one adds constraints to DTW [Ratanamahatana and Keogh, 2004] [Yu et al., 2011] [Candan et al., 2012] [Sakoe and Chiba, 1978] [Jeong et al., 2011] [Salvador and Chan, 2007]. The main idea of these methods is to limit the length of the subsequence of a time series that can be compared to a single datapoint of another time series.
- The second one suggests skipping data points that produce pathological alignment during the comparison of two time series [Longin et al., 2005] [Itakura, 1975] [Myers et al., 1980a].
- The third one proposes to replace the datapoints of time series with a high-level abstraction that captures the local behavior of those time series. A high-level abstraction can be a histogram of values that captures the distribution of time series datapoints in space [Zhang et al., 2015] or a feature that captures the local properties of time series, such as the trend with Derivative DTW (DDTW) [Keogh and Pazzani, 2001b].

Another simple but yet interesting way to capture local properties of time series is to consider the mean of segments of the time series as PAA does. Indeed, the use of the mean reduces the harmful effects of singularities contained in the data and thus

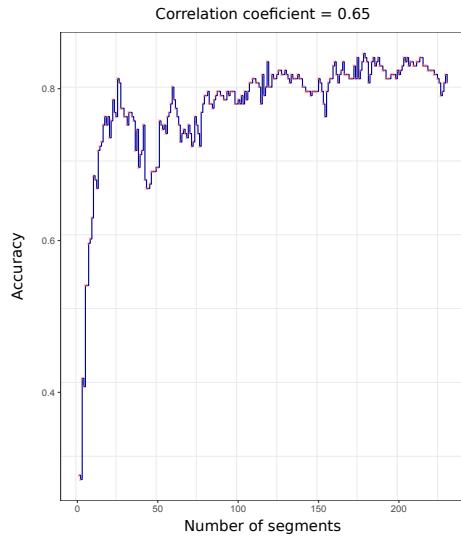


Figure 2.1: Relation between Accuracy and the number of segment on FISH dataset. The accuracy is computed from the algorithm one nearest neighbor (1NN) associated with PDTW. When the number of segments considered is very small (below 20), there is a loss of information and the accuracy is reduced. However, considering all the points in the time series, do not produce a maximum accuracy due to the presence of noise or singularities [Keogh and Pazzani, 2001b] in the data.

allows to avoid pathological alignments. However, one major challenge with PAA is the choice of the number of segments to consider especially with long time series.

### 2.1.2 The problem of choosing a suitable segment number for PAA

If the number of segments considered with PAA is too small, the resulting representation is compact, but it contains less information. On the other hand, if the number of segments is too large, the obtained representation is less compact and more prone to the noise contained in the original time series (Fig. 2.1). Our idea is that a number of segments for PAA will be considered as good (1) if it allows obtaining a compact representation of the time series, and (2) if it preserves the quality of the alignment of time series. So when considering classification task, one of the best algorithm to use for evaluating the quality of time series alignment is one Nearest Neighbor (1NN). Indeed, its classification error directly depends on time series alignment, since 1NN has no other parameters [Wang et al., 2013].

### 2.1.3 Summary of Contributions

In this chapter,

- We define the problem of preprocessing time series with PAA for a better classification with DTW;
- We propose a parameter free heuristic for aligning piecewise aggregate time series with DTW, which approximates the optimal value of the number of segments to be considered with PAA.

The rest of the chapter is organized as follows: in Section 2.2 we recall the definitions and background; Section 2.3 explains our approach; Section 2.4 presents experimental results and comparisons to other methods; Section 2.5 draws conclusions and venues for future work.

## 2.2 Background and related works

Let's recall some definitions.

**Definition 1.** A *time series*  $X = x_1, \dots, x_n$  is a sequence of numerical values representing the evolution of a specific quantity over time.  $x_n$  is the most recent value.

**Definition 2.** A segment  $X_i$  of length  $l$  of the time series  $X$  of length  $n$  ( $l < n$ ) is a sequence constituted by  $l$  variables of  $X$  starting at the position  $i$  and ending at the position  $i + l - 1$ . We have:  $X_i = x_i, x_{i+1}, \dots, x_{i+l-1}$ .

**Definition 3.** The arithmetic average of the data points of a segment  $X_i$  of length  $l$  is noted  $\bar{X}_i$  and is defined by:

$$\bar{X}_i = \frac{1}{l} \sum_{j=0}^{l-1} x_{i+j}. \quad (2.1)$$

**Definition 4.** Let  $T$  be the set of time series. The Piecewise Aggregate Approximation (PAA) is defined as follows:

$$PAA : T \times \mathbb{N}^* \rightarrow T$$

$$(X, N) \mapsto PAA(X, N) = \begin{cases} \bar{X}_k, k \in \{i \times \frac{n}{N} + 1, i = 0, \dots, N - 1\} \text{ if } N < |X| \\ X \text{ otherwise} \end{cases} \quad (2.2)$$

**Definition 5.** Let  $d \subseteq T$  be a subset of time series,  $N \in \mathbb{N}^*$ ,  $PAAset(d, N) = \{PAA(X, N), \forall X \in d\}$ .

### 2.2.1 Dynamic Time Warping algorithm.

DTW [Sakoe and Chiba, 1978] is an algorithm of time series alignment algorithm that performs a non-linear alignment while minimizing the distance between two time series. To align two time series:  $X = x_1, x_2, \dots, x_n$ ;  $Y = y_1, y_2, \dots, y_m$ , the algorithm constructs an  $n \times m$  matrix where the cell  $(i, j)$  of the matrix corresponds to the squared distance  $(x_i - y_j)^2$  between  $x_i$  and  $y_j$ . Then to find the best alignment between  $X$  and  $Y$ , DTW constructs the path that minimizes the sum of squared distances. This path, noted  $W = w_1, w_2, \dots, w_k, \dots, w_K$ , must respect the following constraints:

- Boundary constraint:  $w_1 = (1, 1)$  and  $w_K = (n, m)$ ;
- Monotonicity constraint: given  $w_k = (i, j)$  and :  $w_{k+1} = (i', j')$  then:  $i \leq i'$  and  $j \leq j'$ ;
- Continuity constraint: given  $w_k = (i, j)$  and:  $w_{k+1} = (i', j')$  then:  $i' \leq i + 1$  and:  $j' \leq j + 1$ .

The warping path is computed by an algorithm based on the dynamic programming paradigm that solves the following recurrence:

$$\begin{aligned}\gamma(i, j) = d(x_i, y_j) + \min\{\gamma(i - 1, j - 1), \\ \gamma(i - 1, j), \gamma(i, j - 1)\},\end{aligned}\quad (2.3)$$

where  $d(x_i, y_j)$  is the squared distance contained in the cell  $(i, j)$  and  $\gamma(i, j)$  is the cumulative distance at the position  $(i, j)$  that is computed by the sum of the squared distance at the position  $(i, j)$  and the minimal cumulative distance of its three adjacent cells.

Piecewise Dynamic Time Warping Algorithm (PDTW) [Keogh and Pazzani, 2000] is the DTW algorithm applied on Piecewise Aggregate time series [Keogh et al., 2001b]. Let  $N \in \mathbb{N}^*$ ,  $X$  and  $Y$  be two time series:

$$PDTW(X, Y, N) = DTW(PAA(X, N), PAA(Y, N)). \quad (2.4)$$

The number of segments  $N$  that one considers greatly influences the quality of the alignment of the time series. However, PDTW does not give any information on the way to choose it. For making this choice, [Chu et al., 2002] proposes the Iterative Deepening Dynamic Time Warping Algorithm (IDDTW).

### 2.2.2 Iterative Deepening Dynamic Time Warping

For determining the number of segments, IDDTW only considers values that are power of 2 and for each value, computes an error distribution by comparing PDTW with the standard DTW at each level of compression. It takes as inputs: the query

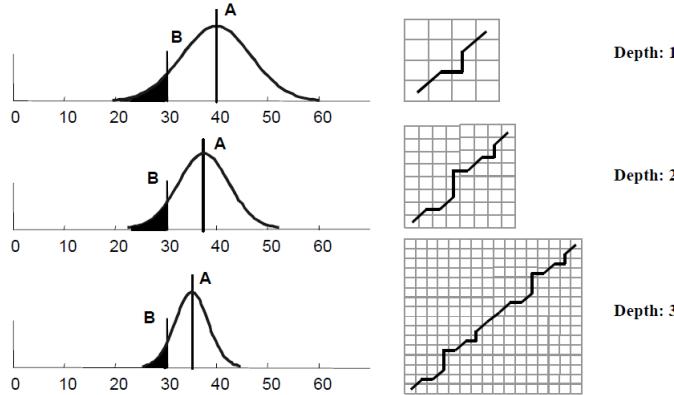


Figure 2.2: IDDTW operating principle. Depth represents approximation levels, A represents approximate distance and B is best\_so\_far [Chu et al., 2002].

$Q$ , the dataset  $D$ , the user's confidence (or tolerance for false dismissals)  $user\_conf$ , and the set of standard deviations  $StdDev$  obtained from the error distribution. Example: Let  $C$  and  $Q$  be two time series of the dataset  $D$ , let  $best\_so\_far$  be the  $DTW$  distance between two time series of the dataset. Suppose the distance  $D_{pdtw}(Q, D)$  is 40 and the  $best\_so\_far$  is 30. The difference between the estimated distance and the  $best\_so\_far$  is 10. Using the error distribution centred around the approximation (40), we can determine the probability that the candidate could be better by examining the area beyond the location of the  $best\_so\_far$  (shown in solid black in Figure 2.2): We disqualify a candidate if this probability is less than the user's specified error acceptance, the candidate is disqualified; otherwise, a finer approximation is used and the test is re-applied to the next depth. This process continues until the full  $DTW$  is performed.

More precisely, IDDTW proceeds as follows:

- the algorithm starts by applying the classic  $DTW$  to the first  $K$  candidates from the dataset. The results of the best matches to the query are contained in  $R$ , with  $|R| = K$ . The  $best\_so\_far$  is determined from  $argmaxR$ ;
- both the query  $Q$  and each subsequent candidate  $C$  are approximated using PAA representations with  $N$  segments to determine the corresponding  $PDTW$ ;
- a test is performed to determine whether the candidate  $C$  can be pruned off or not. If the result of the test is found to have a probability that it could be a better match than the current  $best\_so\_far$ , a higher resolution of the approximation is required. Then each segment of the candidate is split into two segments to obtain a new candidate;
- the process of approximating  $Q$  and  $C$  to determine the  $PDTW$  should be reapplied and the test is repeated for all approximations levels until they fail the test or their true distance  $DTW$  is determined.

In this way, IDDTW finds the number of segments that best approximates DTW and speeds up its computation. However, IDDTW has three main limitations:

- it only considers the numbers of segments for PDTW that are power of 2;
- it requires a user-specified tolerance for false dismissals that influences the quality of the approximation, but the algorithm does not give any indication on how to choose the tolerance;
- it considers DTW as a reference while looking for the number of segments that best aligns the time series. However, because of pathological alignments, DTW sometimes fails to align time series properly [Keogh and Pazzani, 2001b].

Our goal is to find the number of segments that best aligns the time series and also speeds up the computation of DTW.

## 2.3 FDTW: a GRASP based heuristic

We propose a heuristic named parameter Free piecewise DTW (FDTW) based on Greedy Randomized Adapted Search Procedure that deals with all the limitations of IDDTW: it considers all the possible values for the number of segments, it is parameter-free and it finds a number of segments for PDTW based on the quality of the time series alignment, namely the error rate for classification task. The next section introduces FDTW.

### 2.3.1 Evaluation procedures for the compression quality

Before explaining how to evaluate the quality of time series compression, we first describe the time series datasets that we considered. They are made up of time series associated with labels that identify their shapes. For instance, in the ECG dataset, each time series traces the electrical activity recorded during one heartbeat. These time series are split in two classes: normal heartbeat and myocardial infarction.

Time series classification is a classic problem, which consists in guessing the label of an unlabeled time series based on its shape. The quality of a time series classification model is evaluated from its classification error ( $\epsilon$ ), or its accuracy ( $a = 1 - \epsilon$ ). When considering a classification task, one of the best classification algorithm for evaluating the quality of time series alignment is **one Nearest Neighbor (1NN)**. Indeed, its classification error directly depends on time series alignment, since 1NN has no other parameters [Wang et al., 2013].

During this work, a compact representation of time series is considered to be good if it reduces the length of the original time series, but also if the classification error obtained by classifying the compact time series is small. The classification error is small when the time series keep their characteristic shape despite compression.

### 2.3.2 Problem definition.

Let  $D = \{d_i\}$  be a set of datasets composed of time series. We note  $|d_i|$  the number of time series of the dataset  $d_i$ .

Let  $X \in d_i$  be a time series of the dataset  $d_i$ ; we note  $|X| = n$  the length of the time series  $X$ . For simplicity of notation we suppose that all the time series of  $d_i$  have the same length.

**Definition 6.**

$$1NNDTW : D \rightarrow [0, 1] \quad (2.5)$$

$$d_i \mapsto 1NNDTW(d_i) \quad (2.6)$$

$1NNDTW(d_i)$  is the classification error of one Nearest Neighbour with Dynamic Time Warping on the dataset  $d_i$ .

**Definition 7.**

$$\begin{aligned} 1NNPDTW : D \times \{1 \dots n\} &\rightarrow [0, 1] \\ (d_i, N) &\mapsto 1NNPDTW(d_i, N) \\ &= 1NNDTW \circ PAset(d_i, N) \end{aligned} \quad (2.7)$$

$1NNPDTW(d_i, N)$  is the classification error of 1-NN with PDTW using  $N$  segments on  $d_i$ .

Our goal is to find the number of segments that allows PDTW to best align time series. PDTW gives a good alignment when its classification error with 1NN is low [Rakthanmanon et al., 2012a]. Our problem is then to find the number of segments  $N$  that minimizes  $1NNPDTW(d_i, N)$ .

Formally, given a dataset  $d_i$ , of time series that have a length  $n$ , we look for the number of segments  $N \in \{1 \dots n\}$  such that

$$\min_{1 \leq N \leq n} \{1NNPDTW(d_i, N)\}. \quad (2.8)$$

### 2.3.3 Brute-force search.

The simplest way to find the value for the number of segments that minimized the classification error is to test all the possible values. Obviously, this method is time consuming as it requires to test  $n$  values to find the best one. The time complexity is:

$$O(|d|^2 \times n^3), \quad (2.9)$$

where  $C$  is the set of values for the number of segments.

To reduce the time of the search, FDTW proposes to look for the number of segments with the minimal classification error without testing all the possible values.

### 2.3.4 Greedy Randomized Adaptive Search Procedures

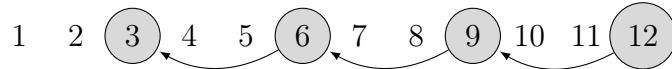
The Greedy Randomized Adaptive Search Procedures (GRASP) is a multi-start, or iterative metaheuristic proposed by Feo and Resende (1995) [Feo and Resende, 1995], in which each iteration consists of two phases: firstly a new solution is constructed by a greedy randomized procedure and this solution is then improved using a local search procedure.

The greediness criterion establishes that candidates with the best quality are added to a restricted list of candidates. Then, one of the candidate of the restricted list is chosen at random when building up the solution. The candidates obtained by greedy algorithms are not necessarily optimal, but they are used as initial solutions to be explored by local search. The heuristic we proposed is build upon GRASP and strengthened with an inclusion of specific global search component.

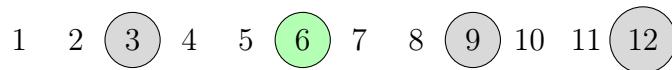
### 2.3.5 Parameter free heuristic

The idea of our heuristic is the following:

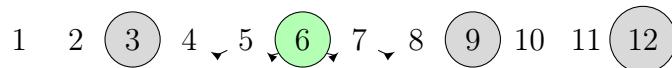
- We choose  $N_c$  candidates number of segments, distributed in the space of possible values to ensure that we are going to have small, medium and large values as candidates. The candidates values are:  $n, n - \left\lfloor \frac{n}{N_c} \right\rfloor, n - 2 \times \left\lfloor \frac{n}{N_c} \right\rfloor, \dots, n - N_c \times \left\lfloor \frac{n}{N_c} \right\rfloor$ . For instance, if the length of time series is  $n = 12$  and the number of candidates is  $N_c = 4$ , we are going to select the candidates 12, 9, 6, 3.



- We evaluate the classification error with  $1NNPDTW$  for each chosen candidate, and we select the candidate that has the minimal classification error: it is the best candidate. In our example, let us suppose that we get the minimal value with the candidate 6: it is thus the best candidate at this step.



- We respectively look between the predecessor (i.e., 3 here) and successor (i.e., 9 here) of the best candidate for a number of segments with a lower classification error: this number of segments corresponds to a local minimum. In our example, we are going to test values 4, 5, 7 and 8 to see if there is a local minimum.



4. We restart at step one while choosing different candidates during each iteration to ensure that we return a good local minimum. We fix the number of iterations to  $k \leq \lfloor \log(n) \rfloor$ . At each iteration, the first candidate is  $n - (\text{number\_of\_iteration} - 1)$ .

In short, in the worst case, we test the first  $M$  candidates to find the best one. Then, we test  $\frac{2n}{M}$  other candidates to find the local minimum. We finally perform  $nb(M) = M + \frac{2n}{M}$  tests. The number of tests to be performed is a function of the number of candidates. Hence, how many candidates should we consider to reduce the number of tests? The first derivative of  $nb$  function vanishes when  $M = \sqrt{2n}$  and its second derivative is positive; so the minimal number of tests is obtained when the number of candidates is :  $M = \sqrt{2n}$ . At each iteration, the heuristic tests  $nb(\sqrt{2n}) = \sqrt{8n}$  number of segments. As we have  $k$  iterations the number of candidates tested is:  $|C| = k\sqrt{8n}$ . The details of the heuristic are presented in Algorithm 1.

**Time complexity:** We use the training set to find the number of segments that should be considered with PDTW. For that purpose, we applied 1NN on the training set that costs

$$O(|d|^2 \times n^2 \sqrt{n}). \quad (2.10)$$

where  $|d|^2$  comes from 1NN algorithm and  $n^2 \sqrt{n}$  comes from PDTW.

### Lemma 1:

For a given dataset  $d_i$ , the quality of the alignment of our heuristic is better than that of DTW:  $FDTW(d_i) \leq 1NNDTW(d_i)$ .

### Proof:

$1NNDTW(d_i) = 1NNPDTW(d_i, n)$ . Then,  $1NNDTW(d_i)$  is one of the candidates considered by the heuristic  $FDTW$ . Since  $FDTW$  returns the minimal classification error from all candidates, the classification error of  $1NNDTW$  is always greater than or equal to  $FDTW$ .

Nevertheless, a heuristic does not always give the optimal value. To ensure that it gives a result not far from the optimal value, one approach is to guarantee that the result of the heuristic always lies within an interval with respect to the optimal value [Ibarra and Kim, 1975].

In our case, we are looking for the number of segments that allows a good alignment of time series. The alignment is good when the classification error with 1NN is minimal or when the accuracy is maximal.

Let  $d_i$  be a dataset:

---

**Algorithm 1:** Parameter Free Dynamic Time Warping

---

**Input:** training\_set, length of a time serie : n,  
number of iterations : nb\_rep

**Output:** The number of segments to be used N  
The accuracy associated to N

```

1 function FDTW(training_set, n, nb_rep)
2   l  $\leftarrow \text{floor}(n/\sqrt{2 * n})$ 
3   tab_N  $\leftarrow \text{ones}(n)$ 
4   forall i  $\in \{0, 1, \dots, (\text{nb\_rep} - 1)\} do
5     tab_N_possible_candidates  $\leftarrow \text{seq}(\text{from} = (n - i), \text{to} = 1, \text{by} = -l)$ 
6     nb_candidates  $\leftarrow |\text{tab\_N\_possible\_candidates}|$ 
7     for i  $\in \{1, 2, \dots, \text{nb\_candidates}\} do
8       if tab_N[tab_N_possible_candidates[i]]  $\neq 0$  then
9         tab_N_candidates[j]  $\leftarrow \text{tab\_N\_possible\_candidates}[i]$ 
10        tab_N[tab_N_candidates[j]]  $\leftarrow 0$ 
11        j  $\leftarrow j + 1$ 
12      mat_r  $\leftarrow \text{1NNPDTW}(\text{training\_set}, \text{tab\_N\_candidates})$ 
13      /* 1NNPDTW return a matrix of couple (N, error) */
14      min  $\leftarrow \text{minimun}(\text{mat}_r)$ 
15      /* minimum return the couple (N, error) with the minimum
16         error */
17      result[(i + 1)]  $\leftarrow$ 
18        localMinimun(min.N, min.error, training_set, tab_N)
19      m  $\leftarrow \text{minimun}(\text{result})$ 
20    return m$$ 
```

---

$acc_{max(d_i)} = 1 - \min_{1 \leq \alpha \leq n} \{1NNPDTW(di, \alpha)\}$  is the maximal accuracy for the dataset  $d_i$ ,

$acc_{DTW} = 1 - 1NNDTW(d_i)$  is the accuracy obtained with  $d_i$  and 1NNDTW, and

$acc_{FDTW} = 1 - FDTW(d_i)$  is the accuracy of our heuristic.

To ensure the quality of our heuristic FDTW, we hypothesized that 1NNDTW is better than Zero Rule classifier. Zero Rule classifier is a simple classifier that predicts the majority class of test data (if nominal) or average value (if numeric). Zero Rule is often used as baseline classifier [Cuřín et al., 2007]. The minimal value of the accuracy of Zero Rule is  $\frac{1}{c}$  where  $c$  is the number of classes of the dataset.

### Proposition 1.:

For a given dataset  $d_i$  that has  $c_i$  classes,  $c_i \in \mathbb{N}^*$ ,

*if*  $acc_{DTW} \geq \frac{1}{c_i}$  *then*  $\frac{1}{c_i} \times acc_{max} \leq acc_{FDTW} \leq acc_{max}$

Proposition 1 shows that 1NN associated with DTW has a better accuracy than the baseline classifier Zero Rule; the FDTW heuristic is a parametric approximation.

### Proof:

By definition,  $acc_{FDTW} \leq acc_{max}$  We look for  $\beta \in \mathbb{N}$  such that

$$\frac{1}{\beta} \times acc_{max} \leq acc_{FDTW} \Leftrightarrow \frac{acc_{max}}{acc_{FDTW}} \leq \beta \quad (2.11)$$

$$\text{However, } \frac{acc_{max}}{acc_{FDTW}} \leq \frac{1}{acc_{FDTW}} \text{ because } acc_{max} \leq 1 \quad (2.12)$$

$$\text{And, } \frac{1}{acc_{FDTW}} \leq \frac{1}{acc_{DTW}} \text{ because } acc_{DTW} \leq acc_{FDTW} \quad (2.13)$$

$$\text{So, } \frac{1}{acc_{DTW}} \leq c_i \text{ because } \frac{1}{c_i} \leq acc_{DTW} \text{ by hypothesis} \quad (2.14)$$

## 2.4 Experiment and results

Throughout the experiments described in this chapter, FDTW performs three iterations ( $k=3$ ) when searching for the appropriate number of segments for a dataset. To evaluate the ability of FDTW heuristic to propose a good number of segments for PAA, it has been compared to IDDTW algorithm in terms of:

- Execution speed;
- Time series compression ratio;
- Classification error associated with the number of segments found by the heuristic.

### 2.4.1 Case studies

PAA is widely used in temporal data mining and often as a primitive by other algorithms such as those allowing to construct a symbolic representation of time series, those allowing to index a time series or even those allowing to classify time series. In this section, we present some algorithms for which the pre-processing performed by FDTW allows to improve the final results.

#### Compression

**Compression ratio:** An immediate way to evaluate the quality of the segmentation is to compare compression ratios. A segment number  $N_1$  will be better than a

segment number  $N_2$  if it allows to obtain a more compact representation with PAA. The compression ratio is given by:

$$r = \frac{n - N}{n},$$

where  $n$  is the length of the time series and  $N$  is the number of segments considered with PAA. The closer  $r$  is to 1 the better is the compression.

The numbers of segments used here are shown in Table 2.1. For the considered datasets, the mean compression ratio of **IDDTW** (**r = 0.654**) is slightly higher than that of **FDTW** (**r = 0.605**). However, this difference is not significant. Indeed, the wilcoxon test gives a p-value greater than 0.1 (**p > 0.1**). Therefore, we cannot reject the hypothesis that the compression ratios of IDDTW and FDTW are equal.

**Application:** PAA used with a suitable segment number allows compression of the time series of the **Coffee** dataset without loss of information. Although they are more compact, the obtained time series capture the main variations of the original time series (figure 2.3).

## Classification

PAA is used by ShapeDTW [Zhao and Itti, 2016] and DTW\_F [Kate, 2016b] to classify time series. However, to evaluate the actual impact of the segment number considered on the classification, we tested FDTW to choose the number of segments to use with 1NN and PDTW.

PDTW was designed to speed up the calculation of DTW without degrading the accuracy. In our experiments, we observe that when the number of segments is chosen, this may even lead to an improvement of the results of the classification.

### Quality of the number of segments found:

A segment number  $N_1$  is better than a segment number  $N_2$  if the classification error associated with  $N_1$  is smaller than that associated with  $N_2$ . So, to evaluate the quality of our heuristic FDTW, we compared its classification errors with that of IDDTW. The classification error was calculated from the results of the 3 fold cross validation applied on the training set. IDDTW tested all the values of  $N$  that were equal to a power of two and kept the one that had a minimum classification error (Table 2.1).

### Application:

According to the announcement in **Lemma 1**, the classification error of FDTW during the learning phase (training error) is less than or equal to that of DTW for

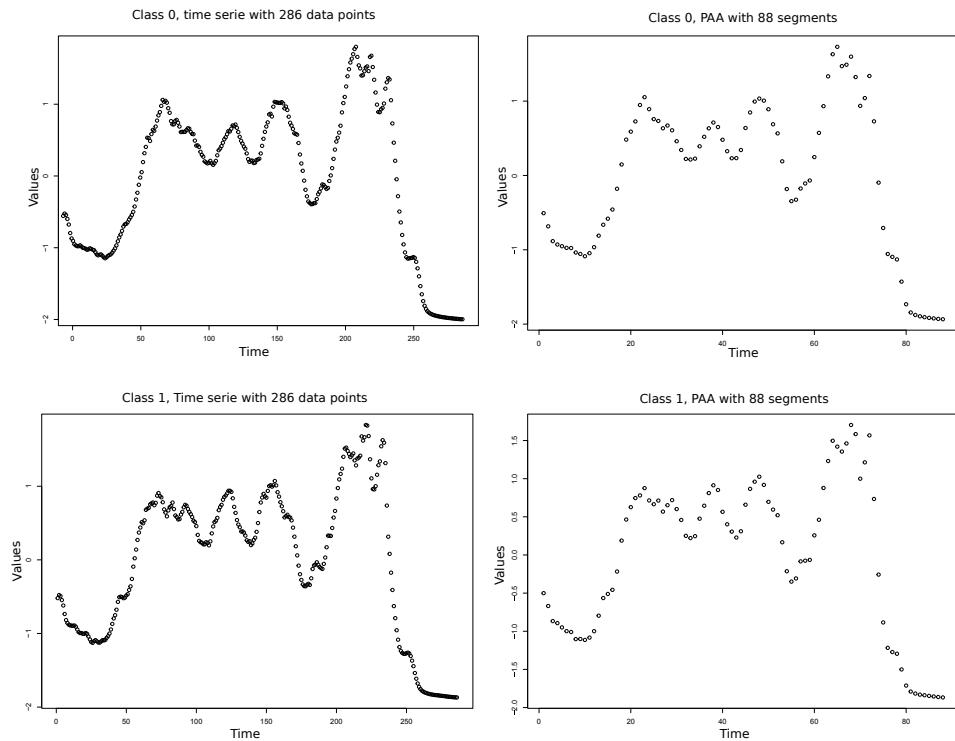


Figure 2.3: Visual comparison of two time series from the two classes of the coffee dataset. Left: the original time series(286 data points), right: representation using PAA with 88 segments

all the considered datasets. We used **Wilcoxon signed rank test** with continuity correction to test the significance of FDTW against IDDTW. The Wilcoxon signed rank test gives a p-values,  $p < 0.01$ , which demonstrates that FDTW achieves a significant reduction of the classification error of IDDTW. This also demonstrates that FDTW allows to find segment numbers for PAA that are of better quality than those found by IDDTW during the learning phase.

<b>N°</b>	<b>Datasets (training set)</b>	<b>DTW</b>	<b>IDDTW</b>	<b>N</b>	<b>FDTW</b>	<b>N</b>
1	50Words	0.349	0.340	256	0.318	80
2	Adiac	0.462	0.426	128	0.426	140
3	ArrowHead	0.250	0.167	16	0.111	14
4	Beef	0.567	0.900	8	0.567	169
5	Car	0.400	0.233	8	0.217	385
6	CBF	0.000	0.000	128	0.000	22
7	Coffee	0.033	0.133	64	0.000	88
8	Cricket_X	0.210	0.244	256	0.190	84
9	Cricket_Y	0.279	0.285	256	0.272	214
10	Cricket_Z	0.267	0.272	256	0.249	250
11	DistalPhalanxOutlineAgeGroup	0.570	0.541	16	0.534	14
12	DistalPhalanxTW	0.375	0.339	16	0.317	40
13	Earthquakes	0.266	0.266	512	0.223	101
14	ECG	0.240	0.170	8	0.170	11
15	ECGFive Days	0.387	0.220	32	0.220	7
16	Face (all)	0.875	0.873	128	0.870	50
17	Face (four)	0.208	0.125	32	0.083	140
18	Fish	0.343	0.314	16	0.303	27
19	Gun-point	0.201	0.039	32	0.020	38
20	Ham	0.650	0.512	32	0.512	32
21	Haptics	0.587	0.536	64	0.516	239
22	InlineSkate	0.519	0.519	64	0.499	48
23	ItalyPower Demand	0.045	0.060	8	0.045	20
24	Lightning-2	0.183	0.150	16	0.100	179
25	Lightning-7	0.315	0.344	64	0.200	155
26	Medical Images	0.286	0.307	64	0.278	94
27	MiddlePhalanxTW	0.429	0.442	32	0.429	80
28	MoteStrain	0.246	0.246	16	0.190	46
29	OliveOil	0.367	0.367	32	0.333	423
30	OSU leaf	0.310	0.335	32	0.270	33
31	Plane	0.000	0.000	32	0.000	32

*continued on next page*

<b>N°</b>	<b>Datasets (training set)</b>	<b>DTW</b>	<b>IDDTW</b>	<b>N</b>	<b>FDTW</b>	<b>N</b>
32	ProximalPhalanxTW	0.317	0.283	4	0.283	4
33	ShapeletSim	0.786	0.246	8	0.143	45
34	SonyAIBORobot Surface	0.198	0.095	16	0.048	22
35	SonyAIBORobot Surface II	0.148	0.111	64	0.037	42
36	Swedish	0.250	0.238	64	0.218	59
37	Symbols	0.037	0.037	32	0.000	34
38	Synthetic Control	0.350	0.410	32	0.350	60
39	Trace	0.000	0.000	64	0.000	108
40	Two patterns	0.000	0.000	32	0.000	32
41	TwoLead ECG	0.125	0.083	64	0.083	52
42	Wafer	0.014	0.012	8	0.008	111
43	Wine	0.684	0.632	128	0.632	20
44	Words Synonyms	0.419	0.423	64	0.382	57
45	Yoga	0.233	0.187	128	0.187	356

Table 2.1: Classification errors associated with the number of segments **N** chosen by the heuristics IDDTW and FDTW. When two numbers of segments  $N_1$  and  $N_2$  are associated with the same classification error, the smallest is considered. The classification error is calculated based on the 3 fold cross validation applied on the training set.

### Comparison with IDDTW:

To evaluate the quality of FDTW, we compared its classification errors with that of IDDTW and the minimal one. The minimal classification error was find by applying Brute-force search (BF) on both training set and testing set. FDTW and IDDTW used the training set to find the segment number  $N$  with minimal training error using 3 fold cross validation, and then used this number of segments on the testing set to compute the classification error. The value of the segment number  $N$  found on the training set may in some cases not be appropriate for the testing set. We speak of a generalization error, which is due to the representativeness of the training set (Table 2.2).

If two numbers of segments  $N_1$  and  $N_2$  are associated with the same training error, we retain the largest. IDDTW tested all the values of  $N$  that were equal to a power of two during the learning phase and kept the one that had a minimum classification error.

N°	[Chen et al., 2015]			Our experiments					
	1NN Eucli dean	1NN DTW	1-NN DTW (r)	Brute force search	N( $\ell$ )	IDDTW	N( $\ell$ )	FDTW	N( $\ell$ )
	distance								
1	0.369	0.310	0.242 (6)	0.262	251(1)	<b>0.268</b>	256(1)	<b>0.268</b>	258(1)
2	0.389	0.396	0.391 (3)	0.379	162(1)	0.432	128(1)	<b>0.414</b>	143(1)
3	0.333	0.367	0.333 (0)	0.233	286(2)	<b>0.3</b>	8(59)	0.367	94(5)
4	0.425	0.274	0.288 (5)	0.192	150(2)	<b>0.301</b>	64(5)	<b>0.301</b>	170(2)
5	0.26	0.25	0.253 (5)	0.233	27(2)	<b>0.283</b>	2(40)	<b>0.283</b>	385(2)
6	0.148	0.003	0.004 (11)	0	118(1)	0.003	128(1)	<b>0.001</b>	128(1)
7	0.000	0.000	0.000 (0)	0	13(22)	<b>0</b>	64(4)	<b>0.000</b>	286(1)
8	0.423	0.246	0.228 (10)	0.228	142(2)	<b>0.256</b>	256(1)	0.269	84(4)
9	0.433	0.256	0.238 (17)	0.231	271(1)	<b>0.241</b>	256(1)	0.244	294(1)
10	0.413	0.246	0.254 (5)	0.221	249(1)	<b>0.223</b>	256(1)	0.233	276(1)
11	0.218	0.208	0.228 (1)	0.2	78(1)	0.225	16(5)	<b>0.223</b>	80(1)
12	0.273	0.29	0.272 (0)	0.263	35(2)	0.288	16(5)	<b>0.278</b>	80(1)
13	0.326	0.258	0.258 (22)	0.198	176(2)	<b>0.258</b>	512(1)	0.276	101(5)
14	0.120	0.230	0.120 (0)	0.13	38(3)	0.19	8(12)	<b>0.180</b>	11(9)
15	0.203	0.232	0.203 (0)	0.117	11(12)	0.289	32(4)	<b>0.117</b>	11(12)
16	0.286	0.192	0.192 (3)	0.091	79(2)	0.194	128(1)	<b>0.148</b>	99(1)
17	0.216	0.170	0.114 (2)	0.08	107(3)	0.352	32(11)	<b>0.102</b>	140(3)
18	0.217	0.177	0.154(4)	0.154	149(3)	0.257	16(29)	<b>0.177</b>	27(17)
19	0.087	0.093	0.087 (0)	0.02	38(4)	0.073	32(5)	<b>0.020</b>	38(4)
20	0.4	0.533	0.400 (0)	0.343	21(20)	<b>0.026</b>	32(13)	0.432	32(13)
21	0.630	0.623	0.588 (2)	0.549	328(3)	<b>0.588</b>	64(17)	0.594	948(1)
22	0.658	0.616	0.613 (14)	0.578	1770(1)	0.627	64(29)	<b>0.622</b>	171(11)
23	0.045	0.050	0.045 (0)	0.033	20(1)	0.043	8(3)	<b>0.033</b>	24(1)
24	0.133	0.167	0.133 (0)	0.1	191(3)	0.167	32(18)	<b>0.100</b>	234(2)
25	0.267	0.267	0.233 (1)	0.183	52(11)	<b>0.367</b>	8(72)	<b>0.367</b>	377(1)
26	0.038	0	0.000 (6)	0	35(4)	<b>0</b>	128(1)	<b>0</b>	135(1)
27	0.439	0.416	0.419 (2)	0.398	27(2)	<b>0.414</b>	32(2)	0.416	80(1)
28	0.121	0.165	0.134 (1)	0.135	14(6)	0.197	16(5)	<b>0.165</b>	84(31)
29	0.246	0.131	0.131 (6)	0.082	70(9)	0.246	16(40)	<b>0.180</b>	524(1)
30	0.479	0.409	0.388 (7)	0.364	31(14)	<b>0.372</b>	32(13)	0.409	35(12)
31	0.316	0.263	0.253 (20)	0.255	95(1)	<b>0.271</b>	64(2)	0.280	34(3)
32	0.292	0.263	0.263 (6)	0.24	75(1)	<b>0.288</b>	4(20)	<b>0.288</b>	4(20)
33	0.461	0.35	0.300 (3)	0.083	54(9)	0.239	64(7)	<b>0.122</b>	48(10)
34	0.305	0.275	0.305 (0)	0.206	37(2)	<b>0.208</b>	16(4)	0.304	26(3)
35	0.141	0.169	0.141 (0)	0.14	5(13)	0.197	16(4)	<b>0.178</b>	45(1)

continued on next page

continued from previous page									
$N^{\circ}$	[Chen et al., 2015]			Our experiments					
	1NN Eucli	1NN DTW	1-NN DTW (r)  dean distance	Brute force search	N( $\ell$ )	IDDTW	N( $\ell$ )	FDTW	N( $\ell$ )
	36	0.211	0.208	0.154 (2)	0.165	59(2)	<b>0.195</b>	64(2)	0.208
37	0.100	0.050	0.062 (8)	0.044	376(1)	<b>0.059</b>	32(12)	0.060	34(12)
38	0.120	0.007	0.017 (6)	0.007	60(1)	0.437	2(30)	<b>0.007</b>	60(1)
39	0.240	0.000	0.010 (3)	<i>0</i>	47(6)	<i>0</i>	64(4)	<i>0</i>	275(1)
40	0.090	0.000	0.002 (4)	<i>0</i>	21(6)	<i>0</i>	64(2)	<i>0</i>	128(1)
41	0.253	0.096	0.132 (5)	0.045	55(1)	0.073	32(3)	0.112	70(1)
42	0.005	0.020	0.005 (1)	0.007	109(1)	0.013	8(19)	<b>0.008</b>	95(2)
43	0.389	0.426	0.389 (0)	0.204	3(78)	0.463	20(11)	<b>0.37</b>	128(1)
44	0.382	0.351	0.252 (8)	0.337	133(2)	0.365	64(4)	<b>0.343</b>	135(2)
45	0.170	0.164	0.155 (2)	0.149	117(4)	0.158	128(3)	<b>0.154</b>	384(1)
X	0.268	0.227	0.242	0.175		0.232		0.214	

Table 2.2: Comparison of generalization errors. In **italics**, the smallest generalization error. In **bold**, the smallest generalization error between IDDTW and FDTW.  $N$  is the number of segments selected and  $\ell$  is the number of data points in a segment ( $\ell = \lfloor \frac{n}{N} \rfloor$ ). The generalization error is computed on the testing set. Note : DTW (r) is a constraint version of DTW where the number of consecutive data points that can be compared to a single point during the warping is bounded. r represents the size of the warping windows

The results of our experiments showed that FDTW is more performant than IDDTW. Actually, FDTW resulted in a lower generalization error than IDDTW on 22 datasets and the same generalization error than IDDTW on 8 datasets. The Wilcoxon signed rank test gives a p-values,  $0.01 < p \leq 0.05$ , which demonstrates that FDTW achieved a significant reduction of the generalization error of IDDTW. Results also showed that FDTW was able to find the minimum error for 9 datasets (Coffee, ECGFiveDays, Gun-point, ItalyPowerDemand, OliveOil, Plane, Synthetic control, Trace, Two patterns) and outperforms the smallest classification error reported in the literature on dataset CBF ( $N^{\circ}5$ ).

### Heuristic execution speed<sup>1</sup>:

As already suggested by the time complexity of both FDTW and IDDTW heuristics, IDDTW tests fewer candidates than FDTW and is therefore faster. However, the number of candidates tested by FDTW reduces exponentially with the length of the time series (Figure 2.4). Actually, the number of candidates to be tested ranges

<sup>1</sup>Note: The experiments were conducted on a PC with an Intel Core i7 processor, 16GB of RAM and a Windows 7 64-bit operating system.

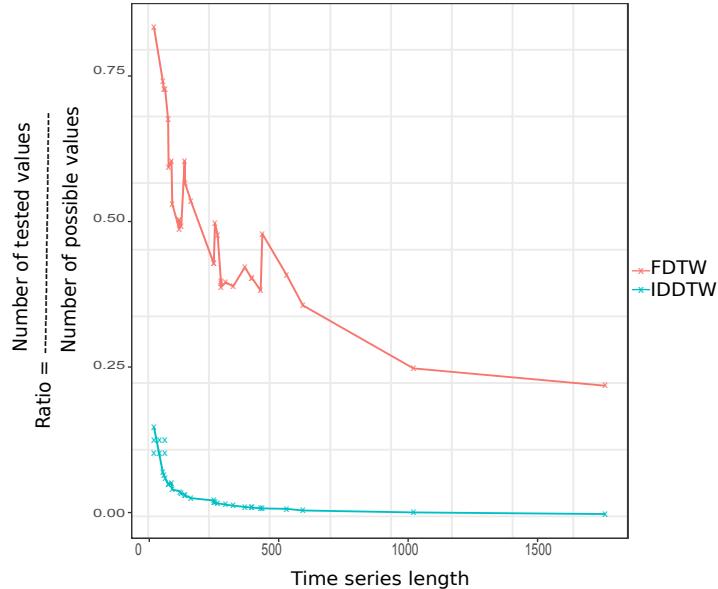


Figure 2.4: Comparison of the number of tested values of the parameter number of segments with the FDTW and IDDTW. Datasets are sorted according to the length of the time series (x-axis).

from 1 to  $n$ ,  $n$  being the length of time series, and FDTW considers  $\sqrt{n}$  candidates for each iteration.

In average, FDTW is 8 times faster than Brute-force search with an average execution time of 176 minutes against 1386 minutes for Brute-force search. IDDTW is 7 times faster than FDTW and remains the fastest with an average execution time of 24 minutes. The execution time increases with the length of the time series (Figure 2.5). The increase of Brute-force search execution time is much more important than that of FDTW and IDDTW, particularly on datasets that contain more than 600 points (e.g. Lightning-2).

### Comparison with other classification algorithms:

To evaluate the quality of FDTW, we compared its classification errors (generalization error) with that of 35 other classification algorithms [Bagnall et al., 2016b] of the literature on 84 datasets of UCR archive. The performances of the algorithms are compared using the Nemenyi test that compares all the algorithms pairwise and provides an intuitive way to visualize the results (Fig. 2.6). The Nemenyi test allows ranking classification algorithms according to their average accuracy on 84 datasets. FDTW obtained good results on the simulated datasets in terms of average accuracy (3rd / 37 algorithms) because data of the training set and of the testing set are generated by the same models.

However, to evaluate the significance of the difference between the 35 classi-

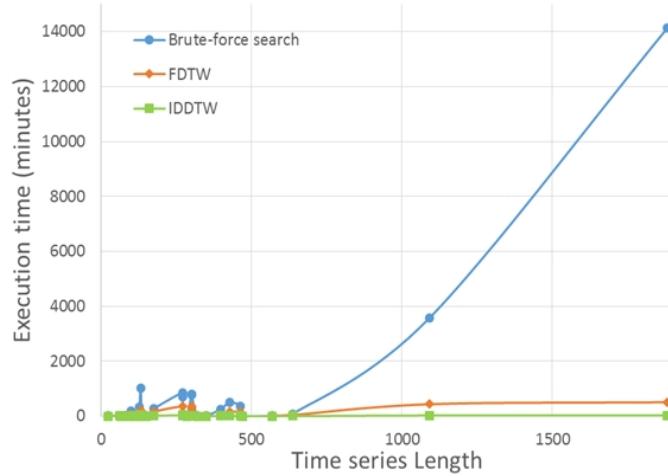


Figure 2.5: Comparison of the execution time of the Brute-force search algorithm, FDTW and IDDTW.

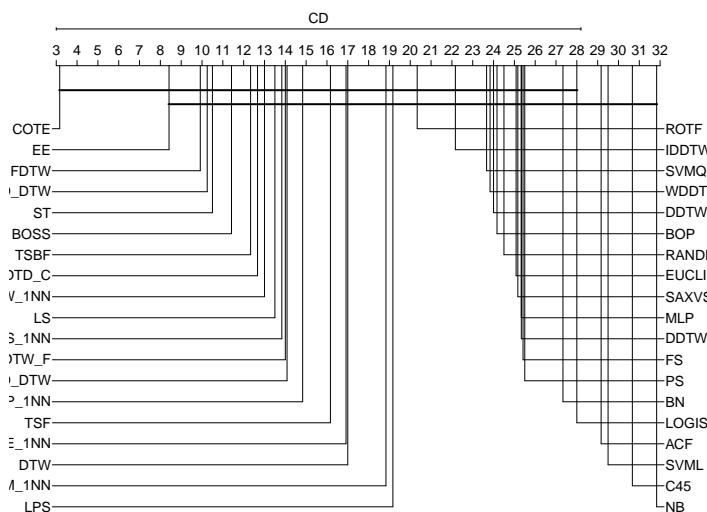


Figure 2.6: Critical difference diagram for FDTW and 36 other classification algorithms on 6 simulated datasets. FDTW is ranked 3rd / 37 algorithms

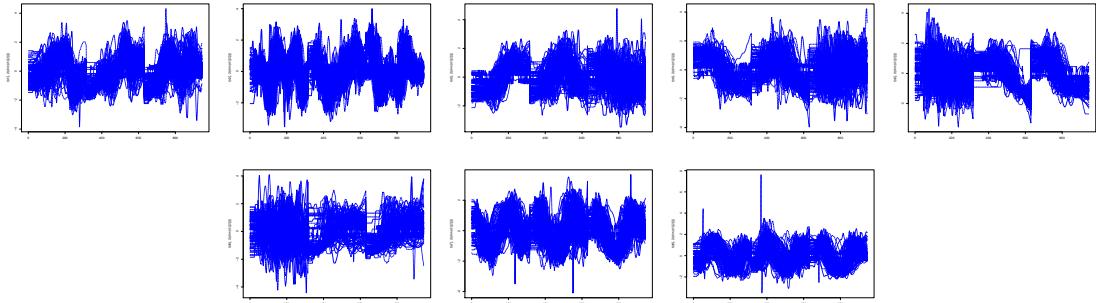


Figure 2.7: Eight types of time series corresponding to the vocabulary of 8 gestures.

fication algorithms on 84 datasets, we used the Wilcoxon signed rank test with continuity correction, which has a higher statistical power than Nemenyi test. The results of these tests show that despite data compression,

- FDTW had a better performance than Naive Bayes (NB), C45, logistic regression (Logistic), BN;
- FDTW had a similar performance as 26 other algorithms in the literature, namely: SVMQ, RANDF, ROTF, MLP, EUCLIDEAN\_1\_NN, DDTW\_R1\_1NN, DDTW\_RN\_1NN, ERP\_1NN, LCSS\_1NN, MSM\_1NN, TWE\_1NN, WD-DTW\_1NN, WDTW\_1NN, DD\_DTW, DTD\_C, LS, BOP, SAXVSM, TSF, TSBF, LPS, PS, CID\_DTW, SVML, FS, ACF;
- Only five algorithms DTW\_F, Shapelet Transform (ST), BOSS, Elastic Ensemble (EE) and COTE performed better overall than FDTW.

These results demonstrate the competitiveness of FDTW. Moreover, this algorithm outperforms the best result reported in the literature on UWAVEGestureLibraryAll dataset (Fig. 2.7). The challenge with this dataset is to recognize the gesture made by a user from measurements made by accelerometers. As reported in [Bagnall et al., 2016a] the best accuracy obtained on this dataset is 83.44% with TSBF algorithm; FDTW outperforms this result and allows to obtain **91.87%** of accuracy.

Additional experiments are available here [[Siyou Fotso et al., 2016](#)]. It is possible to consider the internal properties of time series to choose the compression ratio to use with them. However, this approach gives less good results (Appendix B).

## 2.5 Conclusion and perspective

This chapter deals with the problem of choosing an appropriate number of segments to compress time series with PAA in order to improve the alignment with DTW. In this aim, we proposed a parameter Free heuristic named FDTW, which approximates

the optimal number of segments to use. The experiments showed that, FDTW increased the quality of alignment of time series especially on synthetic datasets where DTW associated with PAA performed better than any other variant of DTW on a classification task. FDTW was rank 3/37 behind two ensemble classification algorithms COTE and EE. In general, FDTW is faster than Brute force search but run lower than IDDTW. However, FDTW gives a better result than IDDTW on classification task. FDTW also allows reducing the storage space and the processing time of time series while increasing the quality of the alignment of DTW.

As a perspective, the problem we have dealt with in this chapter could be modeled as a multi-objective optimization problem where one objective function would be compression and the other the classification of time series.

Another crucial aspect of time series knowledge discovery is the comparison of time series especially when those time series are uncertain. This aspect will be discussed more in-depth in the next chapter.

#### Key points

- We proposed a heuristic for time series compression with Piecewise Aggregate Approximation for classification purpose.
  - We experimentally showed that in addition to reducing the length and the processing time of time series, compression can improve the classification of time series.
- Communications:
- Siyou Fotso VS, Mephu-Nguifo E, Vaslin Ph. Parameter Free Piecewise Dynamic Time Warping. ROADEF, France, February 2017
  - Siyou Fotso VS, Mephu-Nguifo E, Vaslin Ph. Parameter free piecewise dynamic time warping for time series classification. *Time Series workshop at International Conference on Machine Learning*, Sydney, Australia, August 2017

# Chapter 3

## Frobenius correlation based u-shapelets discovery for time series clustering

**Abstract :** An unsupervised shapelet (*u*-shapelet) is a sub-sequence of a time series used for clustering a time series dataset. The purpose of this chapter is to discover *u*-shapelets on uncertain time series. To achieve this goal, we propose a dissimilarity score robust to uncertainty called FOTS whose computation is based on the eigenvector decomposition and the comparison of the autocorrelation matrices of the time series. This score is robust to the presence of uncertainty; it is not very sensitive to transient changes; it allows capturing complex relationships between time series such as oscillations and trends, and it is also well adapted to the comparison of short time series. The FOTS score has been used with the Scalable Unsupervised Shapelet Discovery algorithm for the clustering of 17 datasets, and it has shown a substantial improvement in the quality of clustering with respect to the Rand Index. This work defines a novel framework for clustering of uncertain time series.

### 3.1 Introduction

All measurements performed by a mechanical system contain uncertainty. Indeed, the uncertainty principle is partly a statement about the limitations of mechanical systems ability to perform measurements on a system without disturbing it [Folland and Sitaram, 1997]. Thus, time series obtained from sensors are unavoidably uncertain. These time series constitute a vast proportion of the data used in science, as in medicine with ECGs, in physics with measurements recorded by telescopes, or in computing with the Internet of Things and so on. Ignoring the uncertainty of the data during their analysis can lead to rough or inaccurate conclusions, hence the need to implement uncertain data management techniques.

Several recent studies have focused on the processing of uncertainty in data mining. Two main approaches allow to take uncertainty into account in data mining

tasks: either it is taken into account during the comparison by using appropriate distance functions [Rizvandi et al., 2013, Hwang et al., 2014, Rehfeld and Kurths, 2014, Orang and Shiri, 2014, Wang et al., 2015, Orang and Shiri, 2017], or its impact is reduced by transformations performed on the data [Orang and Shiri, 2015]. This latter strategy is used natively by the u-shapelet algorithm.

### 3.1.1 Review of u-shapelets

Let us consider a data set consisting of 6 time series corresponding to two birds' calls: 3 time series corresponding to **Olive-sided Flycatcher** (green) and three time series corresponding to calls of **White-crowned Sparrow** (blue). When these time series are classified using Euclidean distance as a measure of dissimilarity the groups obtained are not homogeneous (Fig. 3.1 left); in other words, it's not possible to recognize the bird from its calls. However, if characteristic sub-sequences (u-shapelets) are considered to classify time series, we obtain more homogeneous groups than with Euclidean distance (Fig. 3.1 right).

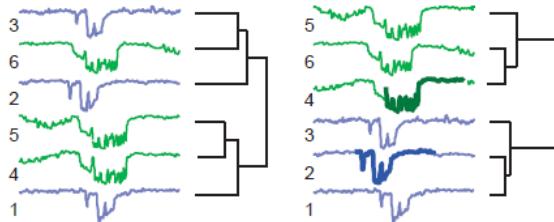


Figure 3.1: Example of classification of time series of two birds' calls (green: Olive-sided Flycatcher; blue: White-crowned Sparrow) using on the one hand the Euclidean distance(left), and on the other hand the u-shapelet (right) [Ulanova et al., 2015]

Once this feature has been observed, the natural question is how to find the sub-sequences that characterize a group of times series, that is, the sub-sequences that are observed only in a time series subgroup. The u-shapelet discovery algorithm answers this question and proceeds as follows: (a) The algorithm takes the length of the pattern as a parameter. (b) On each time series of the base, a window of the same length as the pattern is dragged, each new sub-sequence obtained by this process is a candidate pattern. (c) The candidate consider as a pattern is the subsequence capable of dividing the time series data set into two subsets  $D_a$  and  $D_b$  such that  $D_a$  contains all the time series which possess the pattern and  $D_b$  all those which do not contain the pattern.

Two other constraints are taken into account in the discovery of patterns:

- The first is the ability of the pattern to build subsets that are well separated.
- The second is the ability of employers to build subsets that are not unbalanced.

That is, the size of  $D_a$  must be at most  $k$  times larger than that of  $D_b$  and vice versa.

**Definition 8.** Two datasets  $D_A$  and  $D_B$  are said to be **r-balanced** if only if

$$\frac{1}{r} < \frac{|D_A|}{|D_B|} < (1 - \frac{1}{r}), r > 1$$

**Definition 9.** An **Unsupervised-Shapelet** is any sub-sequence that has a length shorter than or equal to the length of the shortest time series in the dataset, and that allows dividing the dataset into two **r-balanced** groups  $D_A$  and  $D_B$ ; where  $D_A$  is the group of time series that contains a pattern **similar** to the shapelet and  $D_B$  is the group of time series that does not contain the shapelet.

The similarity between a time series and a shapelet is evaluated using a distance function.

**Definition 10.** The sub-sequence distance  $sdist(S, T)$  between a time series  $T$  and a sub-sequence  $S$  is the minimum of the distances between the sub-sequence  $S$  and all possible sub-sequences of  $T$  of length equal to the length of  $S$ .

This definition raises the question of which distance measure to use for  $sdist$ . In general, the ubiquitous Euclidean distance (ED) is used, but it is not appropriate for uncertain time series [Orang and Shiri, 2014]. In the following section, we introduce a dissimilarity function that is more adapted to uncertainty.

Computing the  $sdist$  between a u-shapelet candidate and all time series in a dataset creates an orderline:

**Definition 11.** An orderline is a vector of sub-sequence distances  $sdist(S, Ti)$  between a u-shapelet and all time series  $Ti$  in the dataset.

The computation of the orderline is time-consuming. An orderline for a single u-shapelet candidate is  $O(NM\log(M))$  where  $N$  is the number of time series in the dataset and  $M$  is the average length of the time series. The brute force algorithm for U-shapelets discovery requires  $K$  such computations, where  $K$  is the number of subsequences. The strategy used by [Ulanova et al., 2015] in **Scalable Unsupervised Shapelet algorithm** consists in filtering the  $K$  candidate segments by considering only those allowing to build r-balanced groups. This selection is efficiently made using a hash algorithm.

The assessment of a u-shapelet quality is based on its separation power which is calculated as follows :

$$gap = \mu_B - \sigma_B - (\mu_A - \sigma_A), \quad (3.1)$$

where  $\mu_A$  (resp.  $\mu_B$ ) denotes  $\text{mean}(sdist(S, D_A))$  (resp.  $\text{mean}(sdist(S, D_B))$ ), and  $\sigma_A$  (resp.  $\sigma_B$ ) represents the standard deviation of  $sdist(S, D_A)$  (resp. standard deviation of  $sdist(S, D_B)$ ). If  $D_A$  or  $D_B$  contains only one element (or an insignificant number of elements that cannot represent a separate cluster), the gap score is assigned to zero. This ensures that a high gap scored for a u-shapelet candidate corresponds to a true separation power.

### 3.1.2 U-shapelets algorithm for clustering Uncertain Time Series

U-shapelets clustering is a framework introduced by [Zakaria et al., 2012] who suggested clustering time series from the local properties of their sub-sequences rather than using their global features [Zhang et al., 2016]. In that aim, u-shapelets clustering first seeks a set of sub-sequences characteristic of the different categories of time series and classifies a time series according to the presence or absence of these typical sub-sequences in it.

Clustering time series with u-shapelets has several advantages. Firstly, u-shapelets clustering is defined for datasets in which time series have different lengths, which is not the case of most techniques described in the literature. Indeed, in many cases, the equal length assumption is implied, and the trimming to equal length is done by exploiting expensive human skill [Ulanova et al., 2015]. Secondly, u-shapelets clustering is much more expressive regarding representational power. Indeed, it allows clustering only time series that can be clustered and do not cluster those that do not belong to any cluster.

Furthermore, it is very appropriate to use u-shapelets clustering with uncertain time series because it can ignore irrelevant data and thus, reduce the adverse effects of the presence of uncertainties in the time series. Nevertheless, it is highly desirable to take into account the adverse impact of uncertainty during u-shapelet discovery.

### 3.1.3 Uncertainty and u-shapelets discovery issue

Traditional measurement of similarity like Euclidean distance (ED) or Dynamic Time Warping (DTW) do not always work well for uncertain time series data. Indeed, they aggregate the uncertainty of each data point of the time series being compared and thus amplify the negative impact of uncertainty. However, ED plays a fundamental role in u-shapelet discovery because it is used to compute the gap, i.e. the distance between the two groups formed by a u-shapelet candidate. The discovery of u-shapelet on uncertain time series could thus lead to the selection of a wrong u-shapelet candidate or to assign a time series to the wrong cluster.

In this study, our goal is not to define an uncertain u-shapelet algorithm, but rather to use a dissimilarity function robust to uncertainty to improve the quality of the u-shapelets discovered and thus the clustering quality of uncertain time series.

### 3.1.4 Summary of contributions

- We review the state of the art on similarity functions for uncertain time series and evaluate them for the comparison of small, uncertain time series.
- We introduce the Frobenius cOrrelation for uncertain Time series u-shapelet discovery (FOTS), a new dissimilarity score based on local correlation, which

has interesting and useful properties for comparison of small, uncertain time series and that makes no assumption on the probability distribution of uncertainty in data.

- We put the source code at the disposal of the scientific community to allow extension of our work [[FOTS-SUSh, 2018](#)].

## 3.2 Background and Related works

### 3.2.1 Background

An Uncertain Time Series (UTS)  $X = \langle X_1, \dots, X_n \rangle$  is a sequence of random variables where  $X_i$  is the random variable modeling the unknown real value number at timestamp  $i$ . There are two main ways to model uncertain time series: multiset-based model and PDF-based model.

In **Multiset-based model**, each element  $X_i (1 \leq i \leq n)$  of an UTS  $X = \langle X_1, \dots, X_n \rangle$  is represented as a set  $\{X_{i,1}, \dots, X_{i,N_i}\}$  of observed values and  $N_i$  denotes the number of observed values at timestamp  $i$  (Fig. 3.2).

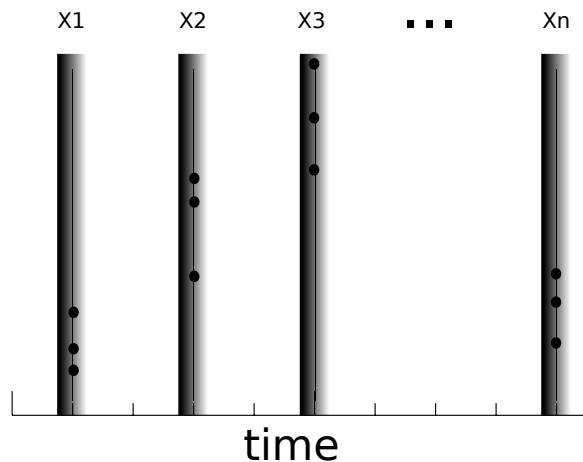


Figure 3.2: Multiset-based model of uncertain time series.

In **PDF-based model**, each element  $X_i, (1 \leq i \leq n)$  of UTS  $X = \langle X_1, \dots, X_n \rangle$  is represented as a random variable  $X_i = x_i + X_{e_i}$  (Fig. 3.3), where  $x_i$  is the exact but unknown value and  $X_{e_i}$  is a random variable representing the error. It is this model that we consider in this work.

Several similarity measures have been proposed for uncertain time series. They are grouped into two main categories:

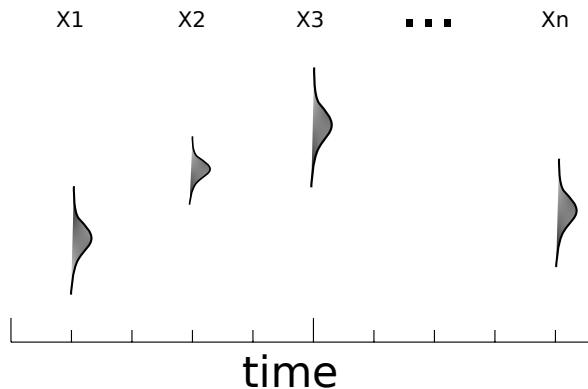


Figure 3.3: PDF-based model of uncertain time series

- **Traditional similarity measures** such as Euclidean distance are those conventionally used with time series. They use a single uncertain value at each timestamp as an approximation of the unknown real value.
- **Uncertain similarity measures** use additional statistical information that quantifies the uncertainty associated with each approximation of the real value : this is the case of DUST, PROUD, MUNICH [Dallachiesa et al., 2012]. [Orang and Shiri, 2015] demonstrated that the performances of uncertain similarity measures associated with pre-processing of data are higher than those of traditional similarity measurements.

### 3.2.2 State of the art on uncertain similarity functions

Uncertain similarity measures can be grouped into two broad categories : deterministic similarity measurements and probabilistic similarity measurements.

#### Deterministic Similarity Measures

Like traditional similarity measures, deterministic similarity measures return a real number as the distance between two uncertain time series. **DUST** is an example of deterministic similarity measure.

**DUST** [Murthy and Sarangi, 2013] Given two uncertain time series

$X = \langle X_1, \dots, X_n \rangle$  and  $Y = \langle Y_1, \dots, Y_n \rangle$ , the distance between two uncertain values  $X_i, Y_i$  is defined as the distance between their true (unknown) values  $r(X_i), r(Y_i)$ :

$$dist(X_i, Y_i) = |r(X_i) - r(Y_i)|. \quad (3.2)$$

This distance is used to measures the similarity of two uncertain values.

$\varphi(|X_i - Y_i|)$  is the probability that the real values at timestamp  $i$  are equal, given the observed values at that instant :

$$\varphi(|X_i - Y_i|) = \Pr(\text{dist}(0, |X_i - Y_i|) = 0). \quad (3.3)$$

This similarity function is then used inside the *dust* dissimilarity function:

$$\text{dust}(X_i, Y_i) = \sqrt{-\log(\varphi(|X_i - Y_i|)) + \log(\varphi(0))}. \quad (3.4)$$

The distance between uncertain time series  $X$  and  $Y$  in *DUST* is then defined as follows:

$$\text{DUST}(X, Y) = \sqrt{\sum_{i=1}^n \text{dust}(X_i, Y_i)^2}. \quad (3.5)$$

The disadvantage of DUST is that it breaks the triangle inequality for small distances. Triangular inequality is a desirable property of dissimilarity functions because it makes it possible to speed-up the comparison of time series. For example, for density based clustering two time series A and B are considered similar if the distance between them is less than  $\epsilon$ . Thus, if the sum of the distances  $d(A, B)$  and  $d(B, C)$  is less than  $\epsilon$ , we deduce that the distance  $d(A, C)$  is also without calculating it. The triangular inequality is also used for the exact indexing of time series [Keogh et al., 2001c].

To remedy this, we introduce a new deterministic distance function based on the Hellinger distance that evaluate the dissimilarity between uncertain time series and respects triangular inequality.

**Hellinger Based Distance** To evaluate the similarity between two probability distributions, we can measure the area of intersection between these two probability distributions (Figure 3.4) . If the area of this intersection is zero, then the probability distributions are disjoint, if it is 1 then the probability distributions are identical. The area of this intersection can be calculated using the Bhattacharyya coefficient (B) [Patra et al., 2015].

The **Hellinger** distance, based on the use of the Bhattacharyya coefficient, allows to measure the dissimilarity between two probability distributions. It is defined as follows:

**Definition 12.** *The Hellinger distance between two probability measures  $P$  and  $Q$  that are absolutely continuous relative to some  $\sigma$ -finite measure  $\mu$  on a measurable space  $(x, \beta)$  is defined by the formula:*

$$H(P, Q) = \{2[1 - B(P, Q)]\}^{\frac{1}{2}}, \quad (3.6)$$

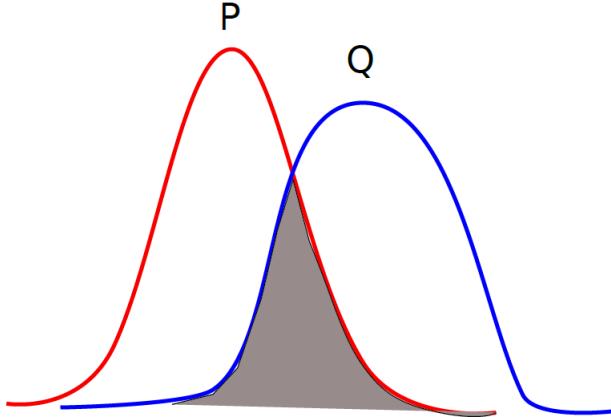


Figure 3.4: Bhattacharyya

where

$$B(P, Q) = \int \sqrt{\frac{dP}{d\mu}} \sqrt{\frac{dQ}{d\mu}} d\mu. \quad (3.7)$$

**Theorem 1.** *The Hellinger distance satisfy the triangle inequality*

[Ibragimov and Has'minskii, 2013].

Based on Hellinger distance we define the HBD distance (Hellinger Based Distance) which measures the dissimilarity between two uncertain time series:

**Definition 13.** *The distance between uncertain time series  $X = \langle X_1, \dots, X_n \rangle$  and  $Y = \langle Y_1, \dots, Y_n \rangle$  under Hellinger Based Distance is then defined as follows:*

$$HBD(X, Y) = \sqrt{\sum_{i=1}^n H(X_i, Y_i)^2}. \quad (3.8)$$

**Theorem 2.** *HBD distance satisfy the triangle inequality.*

*Proof.* Let  $X = \langle X_1, \dots, X_n \rangle$ ,  $Y = \langle Y_1, \dots, Y_n \rangle$ ,  $Z = \langle Z_1, \dots, Z_n \rangle$  be three uncertain time series, we want to proof that

$$\sqrt{\sum_{i=1}^n H(X_i, Y_i)^2} + \sqrt{\sum_{i=1}^n H(Y_i, Z_i)^2} \geq \sqrt{\sum_{i=1}^n H(X_i, Z_i)^2}. \quad (3.9)$$

First, let us show that:

$$\left( \sqrt{\sum_{i=1}^n H(X_i, Y_i)^2} \right) \times \left( \sqrt{\sum_{i=1}^n H(Y_i, Z_i)^2} \right) \geq \sum_{i=1}^n H(X_i, Y_i) H(Y_i, Z_i) \quad (3.10)$$

by squaring the two members of the inequality (3.10) we obtain

$$\left( \sum_{i=1}^n H(X_i, Y_i)^2 \right) \times \left( \sum_{i=1}^n H(Y_i, Z_i)^2 \right) \geq \left( \sum_{i=1}^n H(X_i, Y_i)H(Y_i, Z_i) \right)^2 \quad (3.11)$$

$$i.e. \left( \sum_{i=1}^n H(X_i, Y_i)^2 \right) \times \left( \sum_{i=1}^n H(Y_i, Z_i)^2 \right) \quad (3.12)$$

$$- \left( \sum_{i=1}^n H(X_i, Y_i)H(Y_i, Z_i) \right)^2 \geq 0 \quad (3.13)$$

by developing and reducing the expression (3.13), we obtain

$$i.e. \sum_{i,j \in \{1, \dots, k\} \text{ and } i \neq j} (H(X_i, Y_i) - H(Y_j, Z_j))^2 \geq 0 \quad (3.14)$$

This shows that the inequality (3.10) is true. Let us now show that HBD satisfies the triangular inequality : according to Theorem 1,

$$H(X_i, Y_i) + H(Y_i, Z_i) \geq H(X_i, Z_i) \quad (3.15)$$

By squaring the two members of the inequality, we obtain

$$H(X_i, Y_i)^2 + H(Y_i, Z_i)^2 + 2H(X_i, Y_i)H(Y_i, Z_i) \geq H(X_i, Z_i)^2. \quad (3.16)$$

$$i.e. \sum_{i=1}^n H(X_i, Y_i)^2 + \sum_{i=1}^n H(Y_i, Z_i)^2 + 2 \sum_{i=1}^n H(X_i, Y_i)H(Y_i, Z_i) \quad (3.17)$$

$$\geq \sum_{i=1}^n H(X_i, Z_i)^2. \quad (3.18)$$

according to inequality 3.10, we obtain

$$\sum_{i=1}^n H(X_i, Y_i)^2 + \sum_{i=1}^n H(Y_i, Z_i)^2 \quad (3.19)$$

$$+ 2 \left( \sqrt{\sum_{i=1}^n H(X_i, Y_i)^2} \right) \times \left( \sqrt{\sum_{i=1}^n H(Y_i, Z_i)^2} \right) \geq \sum_{i=1}^n H(X_i, Z_i)^2. \quad (3.20)$$

$$i.e. \left( \sqrt{\sum_{i=1}^n H(X_i, Y_i)^2} + \sqrt{\sum_{i=1}^n H(Y_i, Z_i)^2} \right)^2 \geq \sum_{i=1}^n H(X_i, Z_i)^2. \quad (3.21)$$

$$i.e. \sqrt{\sum_{i=1}^n H(X_i, Y_i)^2} + \sqrt{\sum_{i=1}^n H(Y_i, Z_i)^2} \geq \sqrt{\sum_{i=1}^n H(X_i, Z_i)^2}. \quad (3.22)$$

This is what had to be demonstrated  $\square$

In general, the problem with the deterministic uncertain distances like *DUST* or *HBD* is that their expression varies as a function of the probability distribution of uncertainty, and the probability distribution of the uncertainty is not always available in time series datasets.

### Probabilistic Similarity Measures

Probabilistic similarities measures do not require knowledge of the uncertainty probability distribution. Furthermore, they provide the users with more information about the reliability of the result. There are several probabilistic similarity functions, as MUNICH, PROUD, PROUDS or Local Correlation.

**MUNICH** [Aßfalg et al., 2009]: This distance function is suitable for uncertain time series represented by the multiset based model. The probability that the distance between two uncertain time series  $X$  and  $Y$  is less than a threshold  $\varepsilon$  is equal to the number of distances between  $X$  and  $Y$ , which are less than  $\varepsilon$ , over the possible number of distances:

$$Pr(distance(X, Y)) \leq \varepsilon = \frac{|\{d \in dists(X, Y) | d \leq \varepsilon\}|}{|dists(X, Y)|}. \quad (3.23)$$

The computation of this distance function is very time-consuming.

**PROUD** [Yeh et al., 2009] Let  $X$  and  $Y$  be two UTS each one modeled by a sequence of random variables, the PROUD distance between  $X$  and  $Y$  is  $d(X, Y) = \sum_{i=1}^n (X_i - Y_i)^2$ . According to the central limit theorem [Hoffmann-Jørgensen and Pisier, 1976], the cumulative distribution of the distances approaches asymptotically a normal distribution:

$$d(X, Y) \propto N(\sum_i E[(X_i - Y_i)^2], \sum_i Var[(X_i - Y_i)^2]). \quad (3.24)$$

As a consequence of that feature of PROUD distance, the table of the normal centered reduced law can be used to compute the probability that the normalized distance is lower than a threshold:

$$Pr(d(X, Y)_{norm} \leq \epsilon). \quad (3.25)$$

A major disadvantage of PROUD is its inadequacy for comparing time series of small lengths like u-shapelets. Indeed, the calculation of the probability that the PROUD distance is less than a value is based on the assumption that PROUD distance follows **asymptotically** a normal distribution. Thus, this probability will be all the more accurate as the compared time series are long (i.e. contain more than 30 data points).

**PROUDS**[Orang and Shiri, 2015] is an enhanced version of PROUD, which suppose that random variables coming from time series are independent and identically distributed.

**Definition 14.** *The normal form of a standard time series  $X = < X_1, \dots, X_n >$  is defined as  $\hat{X} = < \hat{X}_1, \dots, \hat{X}_n >$  in which for each timestamp  $i$  ( $1 \leq i \leq n$ ), we have:*

$$\hat{X}_i = \frac{X_i - \bar{X}}{S_X}, \quad \bar{X} = \sum_{i=1}^n \frac{X_i}{n}, \quad S_X = \sqrt{\sum_{i=1}^n \frac{(X_i - \bar{X})^2}{(n-1)}}. \quad (3.26)$$

PROUDS defines the distance between two normalized time series  $\hat{X} = < \hat{X}_1, \dots, \hat{X}_n >$  and  $\hat{Y} = < \hat{Y}_1, \dots, \hat{Y}_n >$  (Definition 14) as follows:

$$Eucl(\hat{X}, \hat{Y}) = 2(n-1) + 2 \sum_{i=1}^n \hat{X}_i \hat{Y}_i. \quad (3.27)$$

For the same reasons as PROUD, PROUDS is not suitable for the comparison of short time series. Another disadvantage of PROUDS is that it assumes that the random variables are independent: this hypothesis is strong and particularly inappropriate for short time series like u-shapelets. A more realistic hypothesis with time series would be to consider that the random variables constituting the time series are  $M$ -dependent. Random variables of a time series are called  $M$ -dependent if  $X_i, X_{i+1}, \dots, X_{i+M}$  are dependent (correlated) and the variables  $X_i$  and  $X_{i+M+1}$  are independent. However, the  $M$ -dependent assumption could make PROUDS writing more complex and its use more difficult because of the choice of the parameter  $M$ .

**Uncertain Correlation** [Orang and Shiri, 2017]: Correlation analysis techniques are useful for feature selection in uncertain time series data. Indeed, correlation indicates the degree of dependency of a feature on other features. Using this information, redundant features can be identified. The same strategy can be useful for u-shapelet discovery. Uncertain correlation is defined as follows :

**Definition 15.** (*Uncertain time series correlation*) Given UTS  $X = \langle X_1, \dots, X_n \rangle$  and  $Y = \langle Y_1, \dots, Y_n \rangle$ , their correlation is defined as:

$$\text{Corr}(X, Y) = \sum_{i=1}^n \hat{X}_i \hat{Y}_i / (n - 1), \quad (3.28)$$

where  $\hat{X}_i$  and  $\hat{Y}_i$  are normal forms of  $X_i$  and  $Y_i$  (Definition 14), respectively.  $X_i$  and  $Y_i$  are supposed to be independant continuous random variables.

If we know the probability distribution of random variables, it is possible to determine the probability density function associated with the correlation, which will subsequently be used to calculate the probability that the correlation between two time series is greater than a given threshold.

Uncertain correlation has however some drawbacks :

- It is too sensitive to transient changes, often leading to widely fluctuating scores;
- It cannot capture complex relationship in timeseries;
- It requires to know the probability distribution function of the uncertainty or to make some assumption on the independence of the random variables contained in time series.

Because of all theses drawbacks, uncertain correlation cannot be used as it is for u-shapelet discovery. The next paragraph presents a generalisation of correlation coefficient that is not an uncertain similarity function but is still interesting for u-shapelet discovery. **Local Correlation** [Papadimitriou et al., 2006] is a generalization of the correlation. It computes a time-evolving correlation scores that tracks a local similarity on multivariate time series based on local autocorrelation matrix. The autocorrelation matrix **allows capturing complex relationship** in time series like the key oscillatory (e.g., sinusoidal) as well as aperiodic trends (e.g., increasing or decreasing) that are present. The use of autocorrelation matrices which are computed based on overlapping windows allows **reducing the sensibility to transient changes** in time series.

**Definition 16.** (*Local autocovariance, sliding window*). Given a time series  $X$ , a sample set of windows with length  $w$ , the local autocorrelation matrix estimator  $\hat{\Gamma}_t$  using a sliding window is defined at time  $t \in \mathbb{N}$  as (Eq.3.29) :

$$\hat{\Gamma}_t(X, w, m) = \sum_{\tau=t-m+1}^t x_{\tau,w} \otimes x_{\tau,w}. \quad (3.29)$$

where  $\mathbf{x}_{\tau,w}$  is a sub-sequence of the time series of length  $w$  and started at  $\tau$ ,  $x \otimes y = xy^T$  is the outer product of  $x$  and  $y$ . The sample set of  $m$  windows is centered around time  $t$ . The number of windows is typically fixed to  $m = w$ .

Given the estimates  $\hat{\Gamma}_t(X)$  and  $\hat{\Gamma}_t(Y)$  for the two time series, the next step is how to compare them and extract a correlation score. This goal is reached using the spectral decomposition. The eigenvectors of the autocorrelation matrices capture the key aperiodic and oscillatory trends, even in short time series. Thus, the subspaces spanned by the first few ( $k$ ) eigenvectors are used to locally characterize the behavior of each series. Definition 17 formalizes this notion:

**Definition 17.** (LoCo score). Given two series  $X$  and  $Y$ , their LoCo score is defined by

$$\ell_t(X, Y) = \frac{1}{2}(\|\mathbf{U}_X^T \mathbf{u}_Y\| + \|\mathbf{U}_Y^T \mathbf{u}_X\|) \quad (3.30)$$

where  $\mathbf{U}_X$  and  $\mathbf{U}_Y$  are the  $k$  first eigenvector matrices of the local autocorrelation  $\hat{\Gamma}_t(X)$  and  $\hat{\Gamma}_t(Y)$  respectively, and  $\mathbf{u}_X$  and  $\mathbf{u}_Y$  are the corresponding eigenvectors with the largest eigenvalue.

Intuitively, two time series  $X$  and  $Y$  will be considered as close when the angle  $\alpha$  formed by the space carrying the information of the time series  $X$  and the vector carrying the information the time series  $Y$  is zero. In other words  $X$  and  $Y$  will be close when the value of  $\cos(\alpha)$  will be 1 (Fig. 3.5).

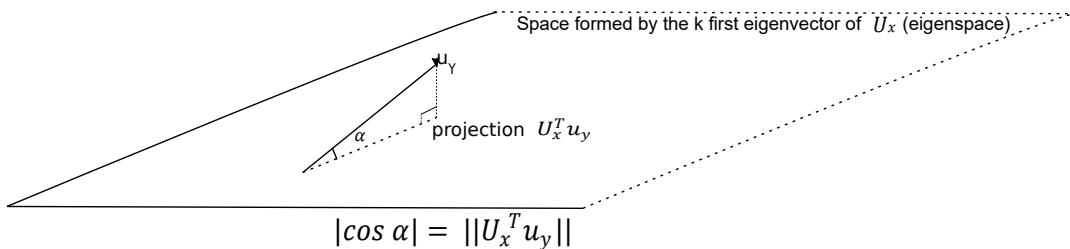


Figure 3.5: Geometric representation of LoCo similarity.

The only assumption made for the computation of LoCo similarity is that the mean of time series data point is zero. This could be easily achieved with z-normalization. LoCo similarity function has many interesting properties and does not require to:

- Know the probability distribution of the uncertainty,
- Assume the independence of the random variables or the length of u-shapelets.

LoCo is therefore interesting for feature selection, but we still need a dissimilarity function to be able to discover u-shapelet. In the next paragraph, we define a dissimilarity function that has the same properties as LoCo and that is robust to the presence of uncertainty.

### 3.3 Frobenius cOrrelation for uncertain Time series u-Shapelet discovery (FOTS)

#### 3.3.1 Dissimilarity function

The LoCo similarity function defined on two multivariate time series  $X$  and  $Y$  approximately corresponds to the absolute value of the cosine of the angle formed by the eigenspaces of  $X$  and  $Y$  ( $|\cos(\alpha)|$ ). A straightforward idea would be to use the  $\sin(\alpha)$  or  $\alpha$ -value as a dissimilarity function but this approach does not work so well, because the sine and the angle are not discriminant enough for eigenvector comparison for clustering purpose. We thus proposed a new dissimilarity measure named Frobenius cOrrelation for uncertain Time series u-Shapeletdiscovery (FOTS) and based on the following definition (Definition. 18):

**Definition 18.** *Given two series  $X$  and  $Y$ , their FOTS score is defined by*

$$FOTS(X, Y) = \|U_X - U_Y\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^k (U_X - U_Y)_{ij}^2} \quad (3.31)$$

where  $\|\cdot\|_F$  is the Frobenius norm,  $m$  is the length of time series and  $k$  is the number of eigenvectors.

Because FOTS computation is based on the comparison of the  $k$ -first eigenvectors of the autocorrelation matrices of the time series, it has the same desirable properties of the LoCo similarity function, that is:

- It **allows to capture complex relationships** in time series like the key oscillatory (e.g., sinusoidal) as well as aperiodic (e.g., increasing or decreasing) trends that are present;
- It allows to **reduce the sensibility to transient changes** in time series;
- It is appropriate for the **comparison of short time series**.

Moreover, the FOTS dissimilarity function is **robust to the presence of uncertainty** due to the spectral decomposition of the autocorrelation matrices of the time series. The robustness of FOTS to uncertainty is confirmed by the theorem of HoffmanWielandt:

**Theorem 3. (Hoffman Wielandt)** [Bhatia and Bhattacharyya, 1993] If  $X$  and  $X + E$  are  $n \times n$  symmetric matrices, then :

$$\sum_{i=1}^n (\lambda_i(X + E) - \lambda_i(X))^2 \leq \|E\|_F^2, \quad (3.32)$$

where  $\lambda_i(X)$  is the  $i$ th largest eigenvalue of  $X$ , and  $\|E\|_F^2$  is the squared of the Frobenius norm of  $E$ .

The next section explains how FOTS is integrated in the Scalable Unsupervised Shapelet discovery algorithm [Ulanova et al., 2015].

### 3.3.2 Scalable u-shapelets Algorithm with FOTS score

In this section we do not define a new SUShapelet algorithm [Ulanova et al., 2015], but we explain how we use SUShapelet algorithm with FOTS score (FOTS-SUSh) to deal with uncertainty.

The *gap* is an essential criterion for the selection of u-shapelets candidate. It is subject to uncertainty because its calculation is based on the Euclidean distance. To remedy this, we propose to use the FOTS score instead of a simple Euclidean distance when calculating the *gap* in the Scalable u-shapelet algorithm.

- Algorithm 2 presents how we compute the orderline using FOTS score,
- Algorithm 3 calculates the orderline and sorts the time series according to their proximity to the u-shapelet candidate (line 2 and 3). A u-shapelet is considered present in a time series if its distance to it is less than or equal to a given threshold. Thus, algorithm select thresholds to build a cluster  $Da$  whose size varies between  $lb$  and  $ub$  (line 5). The algorithm then searches among the selected thresholds the one that has a maximum *gap* (line 6 to 11).

**Definition 19.** The sub-sequence FOTS dissimilarity  $sd_f(S, T)$  between a time series  $T$  and a sub-sequence  $S$  is the minimum FOTS score between the sub-sequence  $S$  and all possible sub-sequences of  $T$  of length equal to the length of  $S$ .

---

**Algorithm 2:** ComputeOrderline

---

**Input:** u-shapeletCandidate :  $s$ ,  
time series dataset :  $D$

**Output:** Distance between the u-shapelet Candidate and all the time series  
of the dataset

```

1 function ComputeOrderline( $s, D$ )
2    $dis \leftarrow \{\}$   $s \leftarrow zNorm(s)$ 
3   forall  $i \in \{1, 2, \dots, |D|\}$  do
4      $ts \leftarrow D(i, :)$ 
5      $dis(i) \leftarrow sd_f(s, ts)$ 
6   return  $dis/|s|$ 
```

---

## 3.4 Experimental Evaluation

### 3.4.1 Clustering with u-shapelets

There are many ways to cluster time series data described by u-shapelets. This approach is a direct implementation of the u-shapelet definition. The algorithm iteratively splits the dataset with each discovered u-shapelet into two groups  $D_A$  and  $D_B$ . The time series that belong to  $D_A$  are considered as members of the cluster formed by the u-shapelet and are then removed from the dataset. A new u-shapelet search continues with the rest of the data until there is no more time series in the dataset or until the algorithm is no more able to find u-shapelet. As a stopping criterion for the number of u-shapelets extracted, the decline of the u-shapelet gap score is examined: the algorithm stops when the gap score of the newly-found u-shapelet becomes less than half of the gap score of the first discovered u-shapelet.

**Choosing the length  $N$  of a u-shapelet:** The choice of the length of u-shapelet is directed by the knowledge of the domain to which the time series belongs. As part of these experiments, we tested all numbers between 4 and half the length of the time series. We considered as length of u-shapelet the one that allows to better cluster the time series.

**Choosing the length  $w$  of the windows :** The use of overlapping windows for calculating the autocorrelation matrix makes it possible to capture the oscillations present in the time series. During these experiments, we considered that the size of the window is equal to half the length of the u-shapelet.

**Choosing the number  $k$  of eigenvectors:** A practical choice is to fix  $k$  to a small value; we used  $k = 4$  throughout all experiments. Indeed, key aperiodic trends are captured by one eigenvector, whereas key oscillatory trends manifest themselves in a pair of eigenvectors.

---

**Algorithm 3:** ComputeGap

---

**Input:** u-shapeletCandidate : s,  
time series dataset : D,  
lb, ub : lower/upper bound of reasonable number of time series in cluster  
**Output:** gap : gap score

```

1 function ComputeGap(s, D, lb, ub)
2   dis  $\leftarrow$  ComputeOrderline(s, D)
3   dis  $\leftarrow$  sort(dis) gap  $\leftarrow$  0
4   for i  $\leftarrow$  lb to ub do
5     DA  $\leftarrow$  dis  $\leq$  dis(i), DB  $\leftarrow$  dis  $>$  dis(i)
6     mA  $\leftarrow$  mean(DA), mB  $\leftarrow$  mean(DB)
7     sA  $\leftarrow$  std(DA), sB  $\leftarrow$  std(DB)
8     currGap  $\leftarrow$  mB - sB - (mA + sA)
9     if currGap  $>$  gap then
10      | gap  $\leftarrow$  currGap
11 return gap

```

---

### 3.4.2 Evaluation Metrics

To evaluate the quality of the u-shapelets found, we use them for a clustering task. Different measures for time series clustering quality have been proposed, including Jaccard Score, Rand Index, Folkes and Mallow index, etc. In our case we had ground truth class labels for the datasets, we could use this external information to evaluate the true clustering quality by using Rand Index. Moreover, Rand Index appears to be the quality measure most commonly used [Zakaria et al., 2012] [Ulanova et al., 2015] [Zhang et al., 2016], and many of the other measures can be seen as minor variants of it [Halkidi et al., 2001]. The Rand Index [Rand, 1971] is calculated as follows:

Let  $L_c$  be the cluster labels returned by a clustering algorithm and  $L_t$  be the set of ground truth class labels. Let  $A$  be the number of time series that are placed in the same cluster in  $L_c$  and  $L_t$ ,  $B$  be the number of time series in different clusters in  $L_c$  and  $L_t$ ,  $C$  be the number of time series in the same cluster in  $L_c$  but not in  $L_t$  and  $D$  be the number of time series in different clusters in  $L_c$  but in same cluster in  $L_t$ . The Rand Index is computed as follows:

$$\text{Rand Index} = (A + B) / (A + B + C + D). \quad (3.33)$$

### 3.4.3 Comparison with u-shapelet

Similarly to [Dallachiesa et al., 2012], we tested FOTS-SUSh on 17 real world datasets coming from UCR archive [Chen et al., 2015] and representing a wide range of ap-

plication domains. The training and testing sets have been joined to obtained bigger datasets. Table 3.1 present detailed information about tested datasets.

Data-set	Size of dataset	Length	No. of Classes	Type
50words	905	270	50	IMAGE
Adiac	781	176	37	IMAGE
Beef	60	470	5	SPECTRO
Car	120	577	4	SENSOR
CBF	930	128	3	SIMULATED
Coffee	56	286	2	SPECTRO
ECG200	200	96	2	ECG
FaceFour	112	350	4	IMAGE
FISH	350	463	7	IMAGE
Gun_Point	200	150	2	MOTION
Lighting2	121	637	2	SENSOR
Lighting7	143	319	7	SENSOR
OliveOil	60	570	4	SPECTRO
OSULeaf	442	427	6	IMAGE
SwedishLeaf	1125	128	15	IMAGE
synthetic_control	600	60	6	SIMULATED
FaceAll	2250	131	14	IMAGE

Table 3.1: Datasets.

Table 3.2 presents the comparison of the two algorithms.

### 3.4.4 Comparison with k-Shape and USLM

k-Shape and USLM are two u-shapelets based clustering algorithms for time series presented in [Zhang et al., 2016]. In this section, we have compared the Rand Index obtained by FOTS-SUShapelet and the one obtained by k-Shape and USLM on 5 datasets (Table 3.3). The results of k-Shape and USLM was previously reported in [Zhang et al., 2016]. This comparison shows that in general, FOTS-SUShapelet perform better than k-Shape and USLM.

### 3.4.5 Discussion

The use of the FOTS score associated with the SUShapelet algorithm allows to discover different u-shapelets than those found by the Euclidean distance. The FOTS-SUSh improves the results of time series clustering because the FOTS score takes into account the intrinsic properties of the time series when searching for u-shapelets and is robust to the presence of uncertainty. This improvement is particularly significant when the FOTS score is used for the clustering of time series

Datasets	RI_SUSH	RI_FOTS
50words	0.811	<b>0.877</b>
Adiac	0.796	<b>0.905</b>
Beef	0.897	<b>0.910</b>
Car	0.708	<b>0.723</b>
CBF	0.578	<b>0.909</b>
Coffee	0.782	<b>0.896</b>
ECG200	0.717	<b>0.866</b>
FaceFour	0.859	<b>0.910</b>
FISH	0.775	<b>0.899</b>
Gun_Point	0.710	<b>0.894</b>
Lighting2	0.794	<b>0.911</b>
Lighting7	0.757	<b>0.910</b>
OliveOil	0.714	<b>0.910</b>
OSULeaf	0.847	<b>0.905</b>
SwedishLeaf	0.305	<b>0.909</b>
synthetic_control	0.723	<b>0.899</b>
FaceAll	0.907	<b>0.908</b>

Table 3.2: Comparison of the Rand Index of SUSH (RI\_SUSH) and FOTS-SUSH (RI\_FOTS). The best Rand Index is in bold

containing several small oscillations. Indeed, these oscillations are not captured by the Euclidean distance whereas they are by the FOTS score whose calculation is based on the autocorrelation matrix. This observation is illustrated by the result obtained on the SwedishLeaf dataset.

### Time complexity analysis

ED can be computed in  $\mathcal{O}(n)$  and FOTS score is computed in  $\mathcal{O}(n^\omega)$ ,  $2 \leq \omega \leq 3$  due to the time complexity of the eigenvector decompositions [Pan and Chen, 1999]. The computation of FOTS score is then more expensive in time than that of ED (Fig. 3.6). However, its use remains relevant for u-shapelets research as they are often small.

## 3.5 Conclusion and Future Work

The purpose of this chapter was to discover u-shapelets on uncertain time series. In that aim, we have proposed a dissimilarity score (FOTS) adapted to the comparison of short time series, of which computation is based on the comparison of the eigenvector of the autocorrelation matrices of the time series. This score is robust in the presence of uncertainty, it is not very sensitive to transient changes, and it

Rand Index	k-Shape	USLM	FOTS-SUShapelet
CBF	0.74	<b>1</b>	0.909
ECG200	0.70	0.76	<b>0.866</b>
Fac.F.	0.64	0.79	<b>0.910</b>
Lig2	0.65	0.80	<b>0.911</b>
Lig.7	0.74	0.79	<b>0.910</b>
OSU L.	0.66	0.82	<b>0.905</b>

Table 3.3: Comparison between k-Shape, USLM and FOTS-SUShapelet.

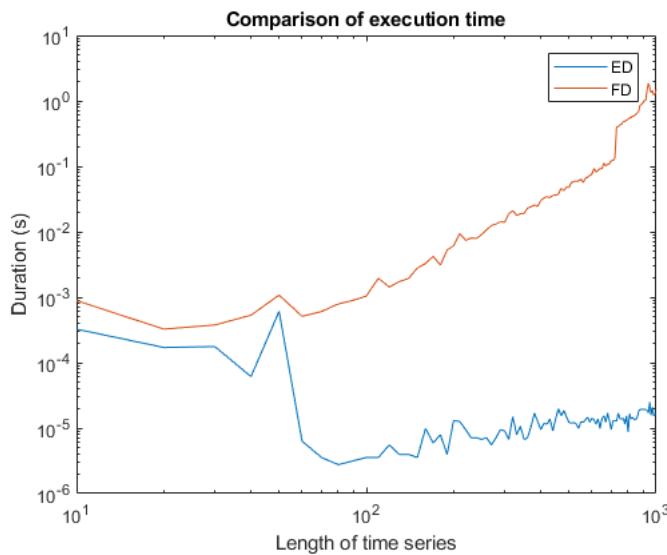


Figure 3.6: The execution time of ED and FOTS score is a function of the length of time series. The computation time of ED is smaller than that of FOTS.

allows capturing complex relationships between time series such as oscillations and trends.

The FOTS score was used with the Scalable Unsupervised Shapelet Discovery algorithm for clustering 17 literature datasets and show that FOTS-SUShapelet consumes more time than UShapelet, but has better performances regarding clustering Rand Index. FOTS-SUSh defines a new framework for clustering uncertain time series because it combines the benefits of the u-shapelets algorithm, which reduces the adverse effects of uncertainty, and the benefits of the FOTS score, which is robust to the presence of uncertainty.

As a perspective to this work, we plan to use the FOTS score for fuzzy clustering of uncertain time series.

Another important aspect of time series coming from wheelchair locomotion is that they are cyclic, and the analysis of this movement is based on those cycles. The

next chapter explains how we have captured this important feature.

#### Key points

- We proposed a correlation-based measure of dissimilarity that captures time series properties and is robust to the presence of uncertainties.
- We experimentally showed that this measure of dissimilarity improves the quality of clustering with UShapelets.

#### Communications :

- Siyou Fotso VS, Mephu-Nguifo E, Vaslin Ph. Découverte d'u-shpalet sur des séries temporelles incertaines à partir de corrélation Frobenius. Rencontres des Jeunes Chercheurs en Intelligence Artificielle (RJCIA 2018), Nancy, France, July 2018



# Chapter 4

## Symbolic representation of cyclic time series based on properties of cycles

**Abstract :** *The analysis of cyclic time series from bio-mechanics is based on the comparison of the properties of their cycles. As usual algorithms of time series classification ignore this particularity, we propose a symbolic representation of cyclic time series based on the properties of cycles, named SAX-P. The resulting character strings can be compared using the Dynamic Time Warping distance. The application of SAX-P to propulsive moments of three subjects ( $S_1$ ,  $S_2$ ,  $S_3$ ) moving in Manual Wheelchair highlight the asymmetry of their propulsion. The symbolic representation SAX-P facilitates the reading of the cyclic time series and the clinical interpretation of the classification results.*

### 4.1 Introduction

Generally, during his locomotion, the human being performs cyclic movements (e.g. : walking, running, swimming, cycling). The bio-mechanical analysis of these movements is performed with various measuring instruments (eg force and acceleration sensors, kinematic analysis systems) that enable continuous recording over long periods of many kinematic and dynamic parameters. These recordings produce long time series composed of many cycles or patterns, representative of the movements made and effort produced by the subject during his displacement (Fig. 4.1).

These cycles are the time series analysis units and have several characteristic properties such as the minimum value, the area under the cycle [Vegter et al., 2014] (Fig. 4.2).

For comparing time series, several previous studies suggested to break them into small segments and then to compare the properties of their segments. A segment of a time series is a sequence of consecutive values belonging to it [Abonyi et al., 2003].

[Keogh et al., 2001b] proposed replacing each segment of a time series  $X =$

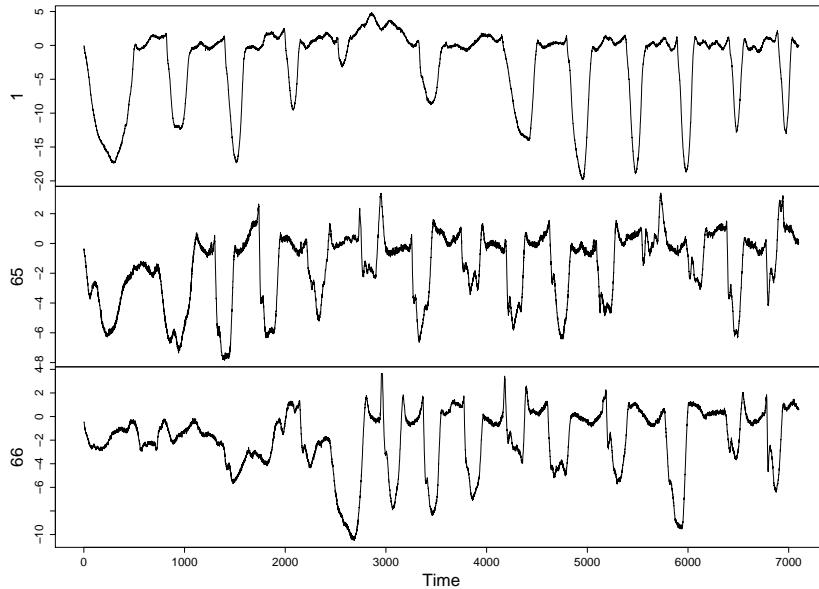


Figure 4.1: Cyclic time series form manual wheelchair locomotion

$x_1, x_2, \dots, x_n$  by its mean values;  $\bar{x}_i = \frac{N}{n} \sum_{j=\frac{n}{N}(i-1)+1}^{(\frac{n}{N})i} x_j$ , transforming the time series, which is a sequence of values, in the suite of the means of its  $N$  segments  $\bar{X} = \bar{x}_1, \bar{x}_2, \dots, \bar{x}_N$ . This method is known as Piecewise Aggregate Approximation (PAA) (Fig. 4.3). The time series  $C$  and  $Q$  are then compared by calculating the distance  $DR$  between the suite  $\bar{C}$  and  $\bar{Q}$  of the means of their segments :

$$DR(\bar{C}, \bar{Q}) = \sqrt{\frac{n}{N} \sum_{i=1}^N (\bar{c}_i - \bar{q}_i)^2}. \quad (4.1)$$

The main objective of PAA was to reduce the length of the time series. However, as it computes the segments means, it also allows us to compare two time series  $C$  and  $Q$  from the properties of their segments (Equation 4.1 ).

[Lin et al., 2003] were based on the PAA method to provide a symbolic representation of time series called Symbolic Aggregate Approximation (SAX). The objective of SAX is to assign a letter to each segment. To do this, the domain of the values of the time series is divided into intervals so that every point of the temporal series has approximately the same probability to belong to an interval and a letter is associated with each of these intervals. Then each segment of the time series is associated with the letter of the interval to which belongs its average (Fig. 4.4).

With SAX, the distance  $MINDIST$  between two strings  $\hat{Q}$  and  $\hat{C}$  of length  $N$  is calculated from the distance between the borders of the intervals represented by each character in the string (Equation 4.2).

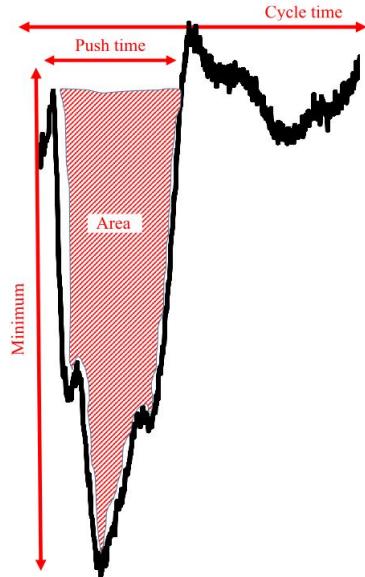


Figure 4.2: Properties of a cycle

$$MINDIST(\hat{Q}, \hat{C}) = \sqrt{\frac{n}{N} \sum_{i=1}^N (dist(\hat{q}_i, \hat{c}_i))^2}. \quad (4.2)$$

$\hat{q}_i$  et  $\hat{c}_i$  are characters and  $dist()$  is the distance between the borders of the intervals which represent these characters [Lin et al., 2003]. However, two segments with very different shapes can have the same average and be represented by the same letter: the mean is not enough to define a segment. In order to solve this problem, [Lkhagva and Kawagoe, 2006] proposed the ESAX model that considers three properties for each segment: its mean, its minimum and maximum (Fig. 4.5).

Thereafter, [Sun et al., 2014] proposed the SAX-TD model that takes into account two properties for each segment: its mean and trend. They then adjust the distance used by the SAX method for it to take into account the trend (Fig. 4.6).

Both methods provide better results than the SAX method [Sun et al., 2014]. However, they have the disadvantage of increasing the number of symbols required to represent the time series. Indeed, the method ESAX triple the size of the representation of a time series provided by the SAX method, while the SAX-TD method the double. In addition, the previous four methods have two major drawbacks: they consider fixed-size segments, while the cycles are variable-sized segments, and they do not take into account the characteristic properties of cycles such as the duration and the surface under a cycle. Our goal is to provide a symbolic representation that takes into account several properties for each cycle, but without increasing the number of symbols used for the representation.

The symbolic representations obtained have another advantage; they allow to

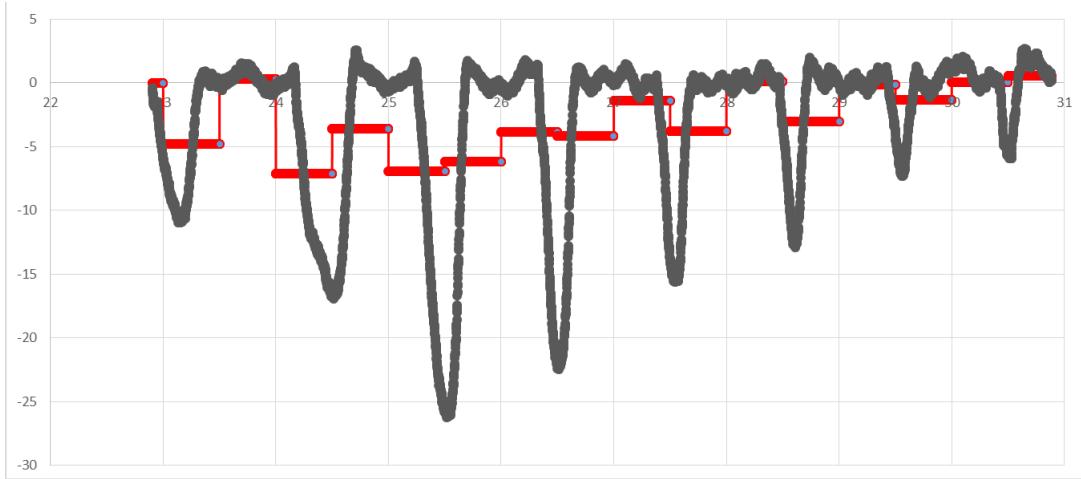


Figure 4.3: Piecewise aggregate approximation of a cyclic time series

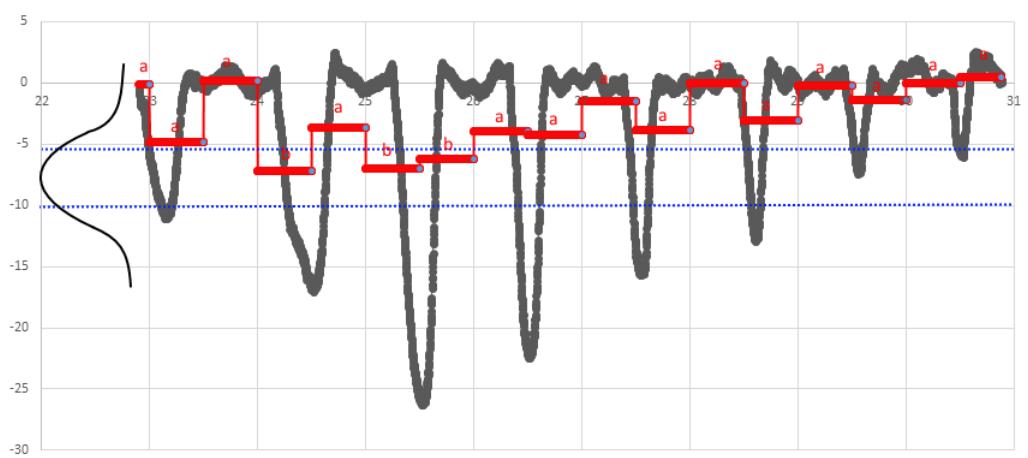


Figure 4.4: Symbolic Aggregate approXimation of a cyclic time series

use a large number algorithms available for sequence analysis like novelty detection (finding unusual shapes or sub-sequences), motif discovery (finding repeated shapes or sub-sequences) [Begum and Keogh, 2014], clustering, classification, indexing and also some interesting algorithms for text processing or the bio-informatics community [Aach and Church, 2001, Papapetrou et al., 2011, Dietterich, 2002].

## 4.2 SAX-P

A prerequisite to be able to build a symbolic representation based on the cycles of the cyclic time series is to be able to segment the cyclic time series into consecutive cycles.

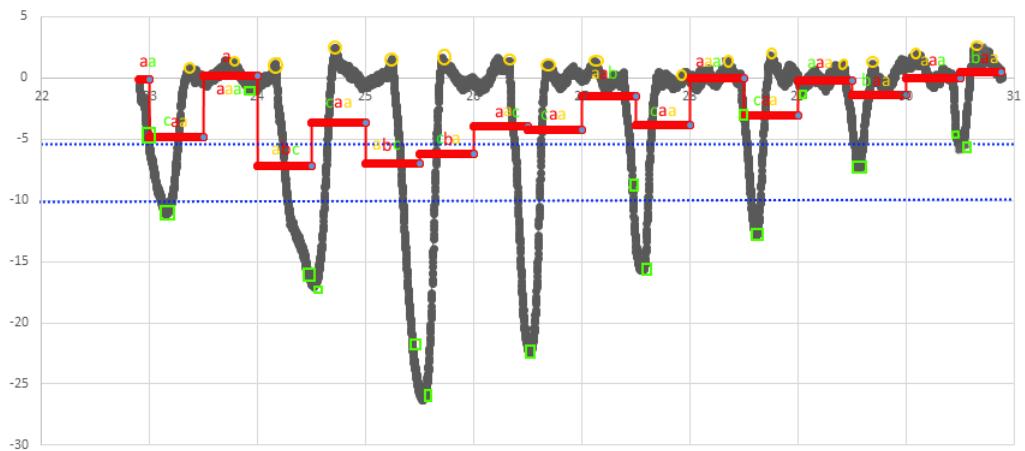


Figure 4.5: Extended Symbolic Aggregate approXimation of a cyclic time series

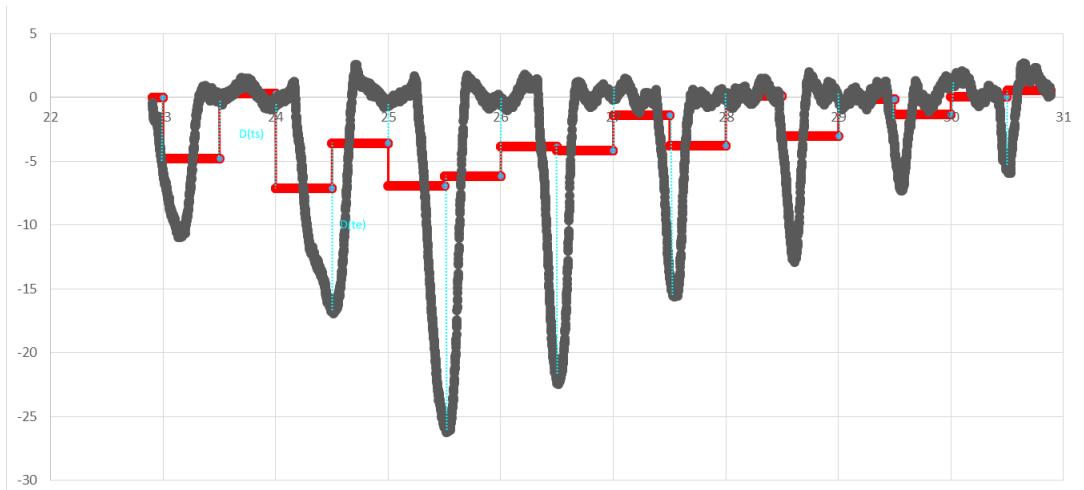


Figure 4.6: Trend Symbolic Aggregate approXimation of a cyclic time series

### 4.2.1 Segmentation of cyclic time series

The principle used to segment cyclic time series is as follows: A cycle contains all the data points between the beginning of two consecutive peaks. To locate the peaks, we set a threshold (Fig. 4.8). The threshold considered can be the first or the second quartile of the time series data point.

If the current value of the time series is below this threshold, then it is a peak. It is then necessary to turn back to find the moment of the beginning of the peak. The figure (Fig. 4.9) presents the results obtained after segmentation of a cyclic time series.

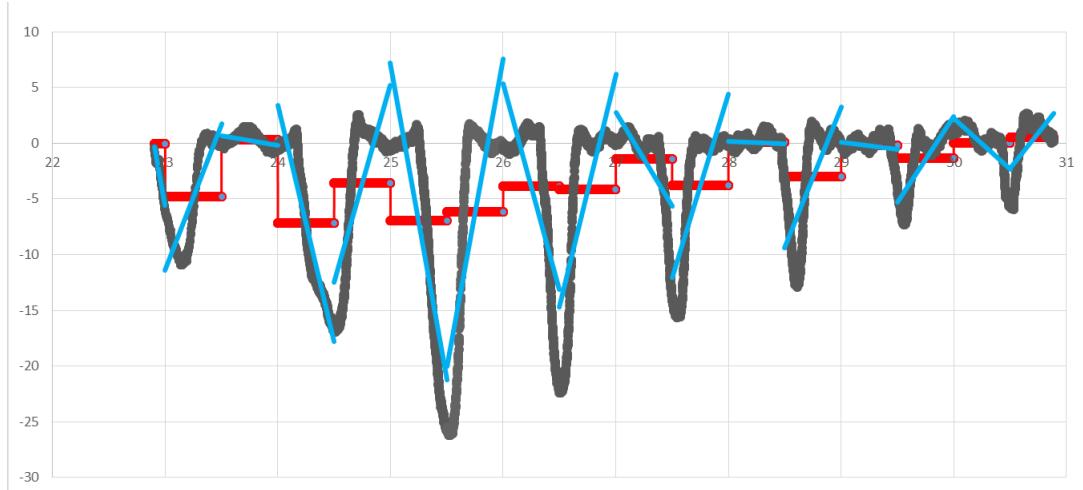


Figure 4.7: Properties of a cycle

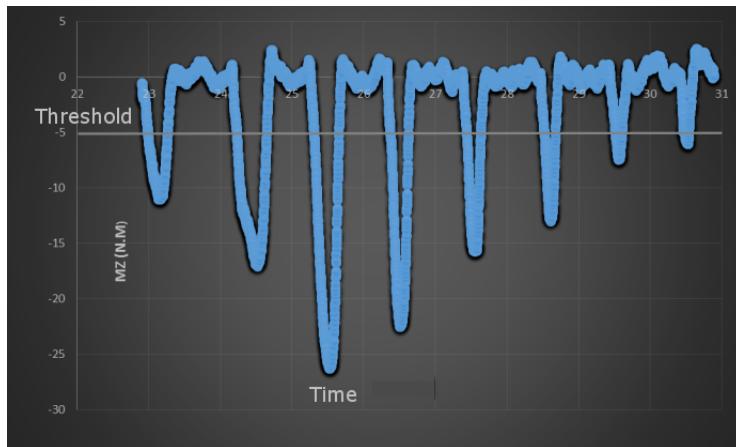


Figure 4.8: Threshold for the segmentation of cyclic time series

### 4.2.2 From cycles to letters

The method SAX-P is based on SAX and works as follows:

1. A cyclic time series is split in successive segments using a threshold for identifying the beginning and the end of cycles, which have variable durations;
2. Several parameters (properties) are computed on each segment: cycle time, push time, mean, median, standard deviation, minimum and maximum values, and the area under the time series curve. As all these parameters have different units, they must be normalized (i.e. centered and reduced) (Fig. 4.10 );
3. Segments are then gathered in clusters using a classification algorithm [Esling and Agon, 2012] and each cluster is named by a capital letter (Fig. 4.11);

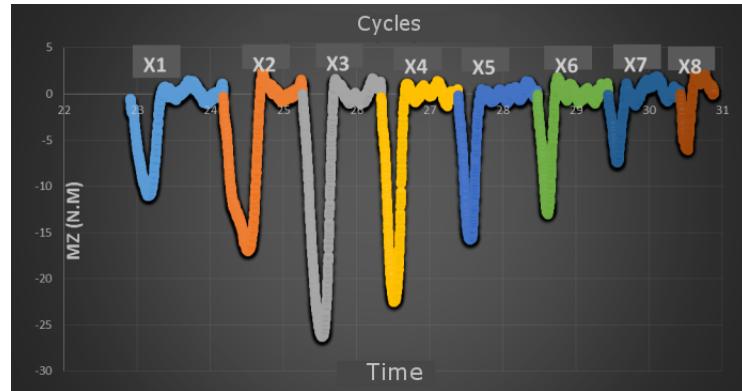


Figure 4.9: Segmentation

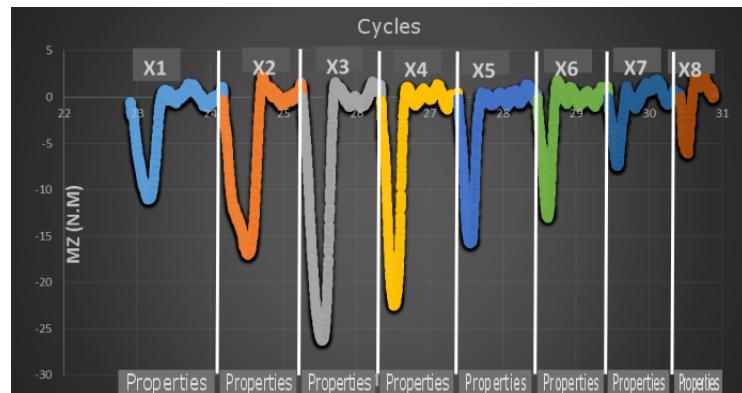


Figure 4.10: Some properties are computed on each cycle

4. Each segment is replaced by the letter of the cluster to which it belongs, so that the initial cyclic time series is then represented by a string of characters (Fig. 4.12);

The distance between two strings, which may have different numbers of characters, is computed using Dynamic Time Warping [Petitjean et al., 2014] which is known as the best distance measure for several domains [Ding et al., 2008]. The distance between two characters is the euclidean distance between the centers of the classes represented by those characters.

Unlike SAX, ESAX and SAX-TD methods that require to fix the length of segments to consider when building the symbolic representation of a time series, SAX-P considers the cycles which constitute basic unit of analysis of time series recorded during cyclic movements and also allows taking into account several characteristic features for each cycle. Figure 4.12 presents the symbolic representations obtained with the SAX method (in small letters) and SAX-P (in capital letters). It illustrates that SAX-P unlike SAX considers cycles of the time series during the construction of the symbolic representation.

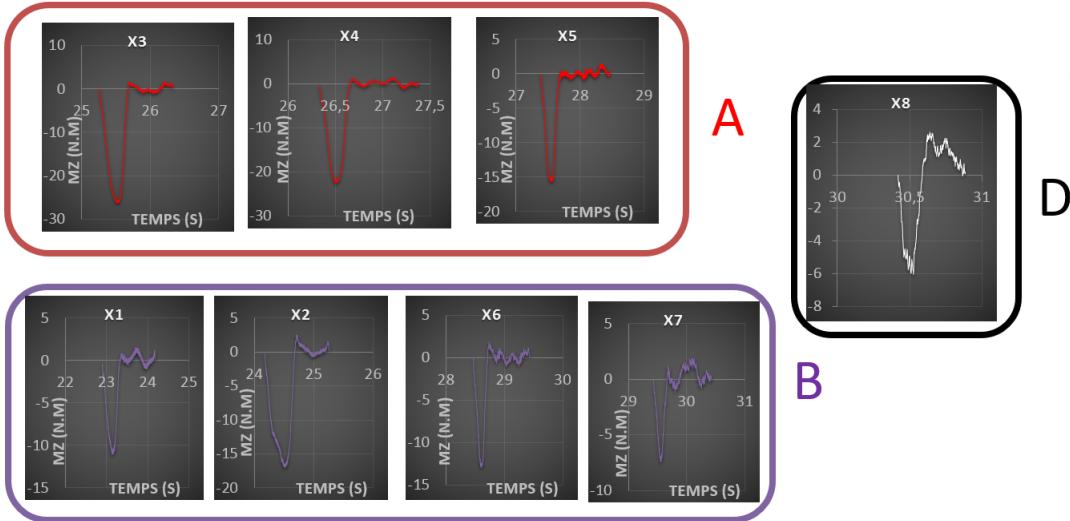


Figure 4.11: Classification of cycles based on properties

## 4.3 Application to manual wheelchair locomotion

### 4.3.1 Dataset description

The data sets used throughout these tests have been obtained from experiments conducted with 12 subjects<sup>1</sup> with disabilities in the aim to understand MWC locomotion. The measurements produced cyclic, uncertain and noisy time series (Fig. 4.13). All subsequent processing and analyses are performed on z-moment time series measured on both rear wheels of the FRET-2.

The data recorded have four main characteristics:

**Length of time series:** the length of the time series is due to the high acquisition frequency (100 Hz) of the force and torque sensor. For instance, a 10-minute recording generates a time series of:

$$100 \text{ Hz} \times 60 \text{ s} \times 10\text{mn} = 60,000 \text{ data points.}$$

The maximum length (107,227 data points) of the time series analysed here was reached by the z-moment of the left wheel of subject S02 (Fig. 4.13). The length of time series is a crucial issue because their processing time is highly dependent on their length. As an illustration, the time complexity of comparing two time series using the DTW alignment algorithm is  $O(n^2)$  where  $n$  is the length of the time series.

**Cycles in time series:** The cyclic aspect of the time series comes from the cyclic nature of wheelchair propulsion. Indeed, this movement consists in a succession of

---

<sup>1</sup>The measurements made with subject S01 being of very low intensity, they were not considered during our analysis.

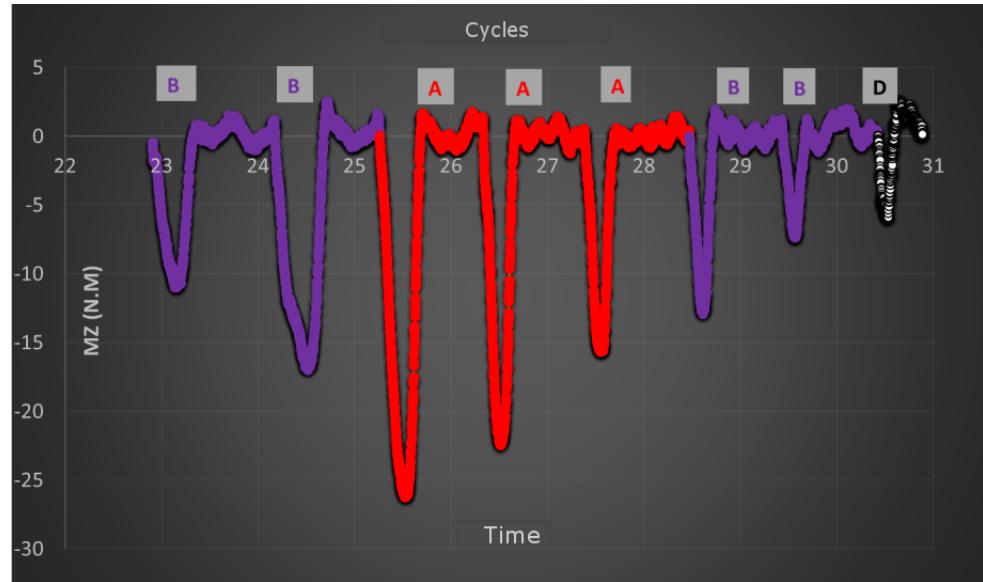


Figure 4.12: Symbolic representation of cyclic time series

push periods during which the user applies a force on the handrim of the wheelchair to propel it, and free-wheel periods during which the user moves his trunk and arms backwards for preparing the next push. The push phase is recorded by the sensor and materialized by a peak in Mz measurements, whereas during freewheel periods Mz values are close to zero (Fig. 4.14).

**Uncertainty in time series:** The presence of uncertainty in sensor measurements is intrinsic to the calibration process of sensors in general, and of the six-component dynamometer used in this work. When a force is applied to a piezo-resistive sensor, it causes a deformation of the sensing element, which consists in strain gauges <sup>2</sup> stuck on a small metallic beam and connected as a Wheatstone bridge <sup>3</sup>. This deformation induces a change in the resistance of the strain gauges and thus a change in the output voltage of the Wheatstone bridge. According to Hooke's law, this is proportional to the force intensity. Thus, when using a calibrated sensor, the user is to measure a variation in sensor signal and uses the proportionality relationship (Hooke's law) to infer the intensity of the force that has been applied. Calibrating a sensor consists in constructing this proportionality relationship by applying a wide range of forces on the sensor, within the limits of the mechanical characteristics of the sensing element. Doing so, we record a sequence of couples of force intensity and electrical voltage, which is used to compute a regression line which will then be used to deduce the applied force intensity knowing the voltage variation. However, the regression line generally does not define a

<sup>2</sup>[https://fr.wikipedia.org/wiki/Jauge\\_de\\_d%C3%A9formation](https://fr.wikipedia.org/wiki/Jauge_de_d%C3%A9formation)

<sup>3</sup>[https://fr.wikipedia.org/wiki/Pont\\_de\\_Wheatstone](https://fr.wikipedia.org/wiki/Pont_de_Wheatstone)

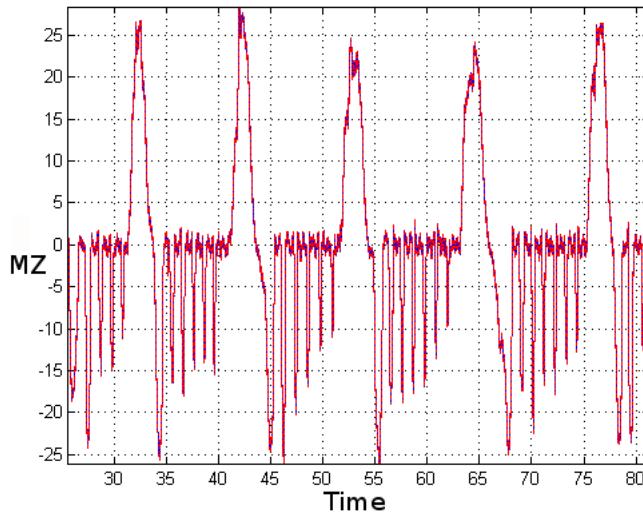


Figure 4.13: Example of a z-moment ( $M_z$ ) time series measured on the [left/right] rear wheel of the FRET-2.

perfect proportionality relationship; in fact, it minimizes the error made but does not cancel it. This error (Fig. 4.15) introduces uncertainty into the estimation of the applied force intensity. It is thus essential to take this error into account when evaluating time series to extract relevant information from them. Characterization of uncertainty is a time-consuming task (Appendix C) and is not always possible because sensor calibration data are generally not available.

**Noise in time series:** In the time series analysed here, the noise comes from the sensitivity of the dynamometer, which has been designed to measure low-intensity forces applied to the handrim during wheelchair locomotion. These forces can come from the texture of the ground (e.g. granular road), unexpected contacts of the user's hands and arms with the handrim during the recovery phase or other. In cyclic time series, noise is problematic because it influences the division into cycles, the computation of properties that characterize a propulsion cycle and the calculation of the distance between two time-series. In the following sections, we explain how we used these data for analysing wheelchair locomotion.

### 4.3.2 The symmetry of Manual Wheelchair Locomotion

For a long time, experts have assumed that wheelchair locomotion was symmetrical, which made it possible to construct measuring instruments consisting of a single wheel [Brouha and Krobath, 1967]. Subsequently, the conclusions that were drawn from the measurements made with one wheel were generalized to both upper limbs of the subject. Then, [Langbein and Fehr, 1993] built a roller ergometer able of

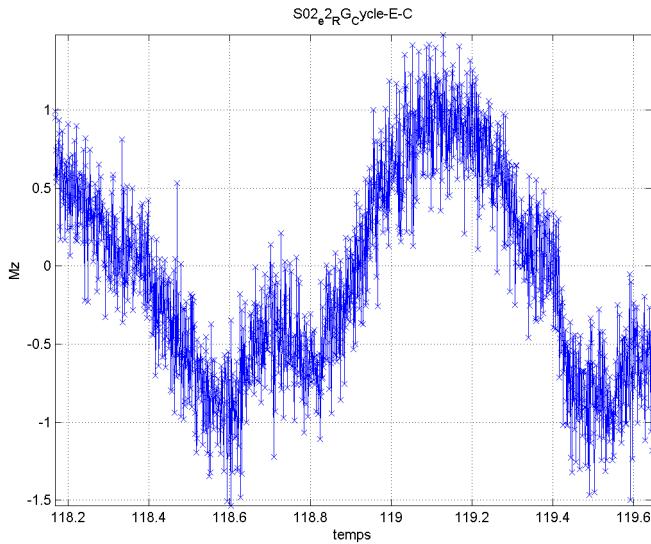


Figure 4.14: Time series recorded by torsor sensor are noisy

separately measuring the speed and resistance of the left and right wheels of the MWC during its use. The measurements made with this roller ergometer revealed a difference between the properties measured by the left and right wheels and exhibited the asymmetric character of wheelchair locomotion. In this paragraph, we used the SAX-P symbolic representation and the additional information we had on the subjects to carry out a new analysis of the symmetry of wheelchair locomotion. Even if a MWC user performs the same number of pushes on both rear wheels during a straight displacement, these cycles may have different properties, which in our case is expressed by different letters in the character strings representing the propulsion cycles applied by the user to the right and left rear wheels. This asymmetry of wheelchair locomotion can be evaluated by calculating a relative Edit distance that counts the number of different letters between the characters strings of the right and left wheels during a same straight displacement (lap).

$$D(X, Y) = \frac{1}{n} \sum_{i=1}^n [X_i \neq Y_i].$$

Where  $X$ , and  $Y$  are character strings.

Subject	Straight displacement	Relative Edit Distance
S02_e1_RD_H4-Cycle-A	AAAAAA	0,17
S02_e1_RG_H4-Cycle-A	AAAAAA	
S02_e1_RD_H4-Cycle-B	AAAAAAAAAAAAAA	0,06
S02_e1_RG_H4-Cycle-B	AAAAAAAAAAAAAA	
Continue to the next page		

Subject	Straight displacement	Relative Edit Distance
Following ...		
S02_e1_RD_H4-Cycle-C	AAAAAAAAAAAAA	0
S02_e1_RG_H4-Cycle-C	AAAAAAAAAAAAA	0
S02_e1_RD_H4-Cycle-D	AAAAAAA	0
S02_e1_RG_H4-Cycle-D	AAAAAAA	0,05
Moyenne		
S03_E3_T_RD-Cycle-A	EEBCB	0,40
S03_E3_T_RG-Cycle-A	EEECC	
S03_E3_T_RD-Cycle-B	BBBBCB	0,33
S03_E3_T_RG-Cycle-B	EBBBCC	
S03_E3_T_RD-Cycle-C	EBBBBBC	0,17
S03_E3_T_RG-Cycle-C	EBBBBCC	
S03_E3_T_RD-Cycle-D	EBCBC	0,86
S03_E3_T_RG-Cycle-D	EEECBCC	
S03_E3_T_RG-Cycle-E	EBBBBBC	0,29
S03_E3_T_RD-Cycle-E	EBEBBCC	
S03_E3_T_RD-Cycle-F	EEEBBCC	0,43
S03_E3_T_RG-Cycle-F	EEECCCD	
S03_E3_T_RG-Cycle-G	EEBCCC	0,57
S03_E3_T_RD-Cycle-G	BEECBCC	
S03_E3_T_RD-Cycle-H	EEBCBCC	0,29
S03_E3_T_RG-Cycle-H	EBBBBBC	
S03_E3_T_RG-Cycle-I	CC	1,00
S03_E3_T_RD-Cycle-I	BBD	0,48
Moyenne		
S04_E1_T_RD-Cycle-A	EEDBBCCC	0,38
S04_E1_T_RG-Cycle-A	DEEBBCCA	
S04_E1_T_RG-Cycle-B	EEBDACC	0,63
S04_E1_T_RD-Cycle-B	BBBBCCCC	
S04_E1_T_RD-Cycle-C	ECBCCCC	0,57
S04_E1_T_RG-Cycle-C	BBECCC	
S04_E1_T_RG-Cycle-D	EBBCC	0,57
S04_E1_T_RD-Cycle-D	EBBBBBC	
S04_E1_T_RD-Cycle-E	EBBC	0,50
S04_E1_T_RG-Cycle-E	EBC	
S04_E1_T_RD-Cycle-F	EBC	0,33
S04_E1_T_RG-Cycle-F	BBC	
Moyenne		0,5
Continue to the next page		

Subject	Straight displacement	Relative Edit Distance
Following ...		
S05_E3_T_RD-Cycle-A	BBBBCDD	0,86
S05_E3_T_RG-Cycle-A	EEBDBCA	
S05_E3_T_RD-Cycle-B	BBCDCDBC	0,63
S05_E3_T_RG-Cycle-B	BBBCCA	
S05_E3_T_RD-Cycle-C	EDDDDDCAD	0,78
S05_E3_T_RG-Cycle-C	BBBCBCA	
S05_E3_T_RD-Cycle-D	BBCDCA	0,29
S05_E3_T_RG-Cycle-D	BBCBCCA	
S05_E3_T_RD-Cycle-E	BBBCDCCD	0,75
S05_E3_T_RG-Cycle-E	BCDDCDA	
S05_E3_T_RD-Cycle-F	BCBDCDCD	0,63
S05_E3_T_RG-Cycle-F	DBBCDDAD	
S05_E3_T_RD-Cycle-G	BC	1,00
S05_E3_T_RG-Cycle-G	DA	
Moyenne		0,70
S07_e1_T_RD-Cycle-A	AEDBDA	0,33
S07_e1_T_RG-Cycle-A	EEBBD	
S07_e1_T_RG-Cycle-B	BBCCC	0,00
S07_e1_T_RD-Cycle-B	BBCCC	
S07_e1_T_RD-Cycle-C	ECDADAA	0,71
S07_e1_T_RG-Cycle-C	EBCCD	
S07_e1_T_RG-Cycle-D	BBBCCD	0,57
S07_e1_T_RD-Cycle-D	EBCCADA	
S07_e1_T_RD-Cycle-E	EBCCCAD	0,57
S07_e1_T_RG-Cycle-E	BBBCCC	
S07_e1_T_RD-Cycle-F	BC	1,00
S07_e1_T_RG-Cycle-F	EC	
Moyenne		0,53
S08_e3_T_RD-Cycle-A	DDDDDAD	0,29
S08_e3_T_RG-Cycle-A	DDDDDDA	
S08_e3_T_RD-Cycle-B	DDDADDDDA	0,40
S08_e3_T_RG-Cycle-B	BDDDDAADAA	
S08_e3_T_RD-Cycle-C	DDDDDDDDA	0,67
S08_e3_T_RG-Cycle-C	AAADAADAA	
S08_e3_T_RD-Cycle-D	DDDDDDDDC	0,89
S08_e3_T_RG-Cycle-D	AAADAAAAAA	
Continue to the next page		

Subject	Straight displacement	Relative Edit Distance
Following ...		
S08_e3_T_RD-Cycle-E	BDDD	0,50
S08_e3_T_RG-Cycle-E	DDDA	0,55
Moyenne		
S09_e1_T_RD-Cycle-A	DADDDA	0,33
S09_e1_T_RG-Cycle-A	DDDADA	0,92
S09_e1_T_RD-Cycle-B	DDAADAAAAA	0,40
S09_e1_T_RG-Cycle-B	AADDxDDDDAA	0,70
S09_e1_T_RD-Cycle-C	DDAAADADD	0,59
S09_e1_T_RG-Cycle-C	DDADDDCDDA	
S09_e1_T_RD-Cycle-D	DDDDAAA	
S09_e1_T_RG-Cycle-D	AADDxDDAA	
Moyenne		
S10_e3_RD_H-4-Cycle-A	EEBC	0,20
S10_e3_RG_H-4-Cycle-A	EEBCD	0,83
S10_e3_RD_H-4-Cycle-B	EBDCCA	0,29
S10_e3_RG_H-4-Cycle-B	BBBBCC	0,67
S10_e3_RD_H-4-Cycle-C	BBBCCC	0,50
S10_e3_RG_H-4-Cycle-C	EBCCDC	0,86
S10_e3_RD_H-4-Cycle-D	EECCCA	0,50
S10_e3_RG_H-4-Cycle-D	BCDCABC	0,50
S10_e3_RD_H-4-Cycle-E	C	0,55
S10_e3_RG_H-4-Cycle-E	CE	
Moyenne		
S11_e1_T_RD-Cycle-A	BDDA	0,50
S11_e1_T_RG-Cycle-A	BBD	0,38
S11_e1_T_RD-Cycle-B	DDDDADDA	0,50
S11_e1_T_RG-Cycle-B	BBDBADDA	
S11_e1_T_RD-Cycle-C	DDDADADA	
S11_e1_T_RG-Cycle-C	BDDDDDDD	
S11_e1_T_RD-Cycle-D	BDDDAAAA	
S11_e1_T_RG-Cycle-D	BDDDDDDD	
S11_e1_T_RD-Cycle-E	DA	0,67
Continue to the next page		

Subject	Straight displacement	Relative Edit Distance
Following ...		
S11_e1_T_RG-Cycle-E Moyenne	DDD	0.51
S12_e2_RD_H4-Cycle-A	DDDAAAA	0,71
S12_e2_RG_H4-Cycle-A	BDDDDDD	
S12_e2_RD_H4-Cycle-B	DADDDAAAA	0,44
S12_e2_RG_H4-Cycle-B	AADDADDADD	
S12_e2_RD_H4-Cycle-C	DDAADADDAAAA	0,67
S12_e2_RG_H4-Cycle-C	DBDDDDDDDDDD	
S12_e2_RD_H4-Cycle-D	DDAADD	0,50
S12_e2_RG_H4-Cycle-D Moyenne	DDADD	0.58

Table 4.1: Straight displacement in manual wheelchair

End

In all the laps analysed in this study (Table 4.1), almost all subjects showed a propulsion asymmetry during their displacements, and this asymmetry was different for all subjects. Indeed, it is almost null for subject S02 who has only performed A-type pushes throughout his displacements, whereas it is quite high (0.7) for subject S05.

Subject	Edit distance	Duration of practice
S02	0.05	33 years
S03	0.43	7 years
S04	0.5	6 months
S11	0.51	12 years
S07	0.53	7 months 6 days
S08	0.55	3 years 6 months
S10	0.55	2 years
S12	0.58	11 months
S09	0.59	9 months
S05	0.7	2 months

Table 4.2: Asymmetry (Edit distance) of subjects' propulsion with regard to their number of years of practice.

In order to identify which factors might influence propulsion asymmetry, we cross-referenced the previous results with the number of years of wheelchair locomotion practice of the subjects who participated in the experiment (Table 4.2). We observed that subject S05 who has the least experience in wheelchair locomotion

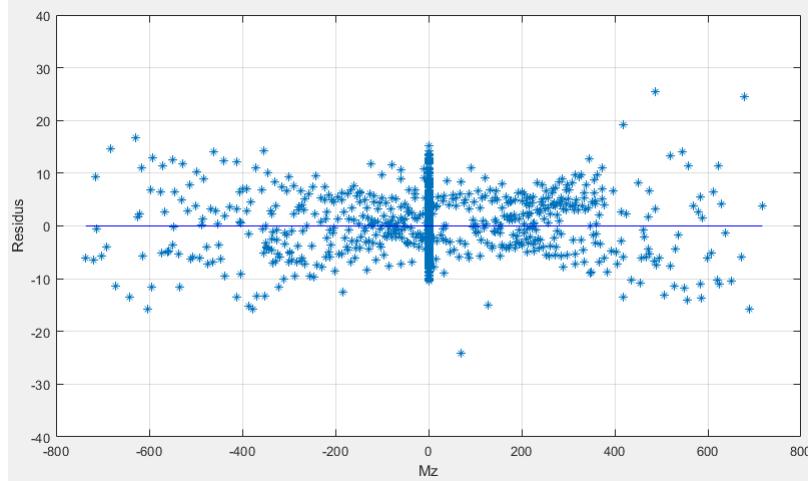


Figure 4.15: Plot of Mz residues, which represent the differences between Mz values measured by the sensor and real values applied during the calibration process.

**(2 months)** has a very asymmetric propulsion (**Edit distance = 0.7**) whereas subject S02 who has a long experience of wheelchair practice (**33 years**) has a very symmetric propulsion (**Edit distance = 0.05**). The Pearson's correlation coefficient ( $r = -0.93$ ) between asymmetry and number of years of wheelchair practice showed that the longer is the subjects' experience in wheelchair locomotion, the more symmetrical is their propulsion during a straight displacement.

### 4.3.3 Group Manual Wheelchair users according to their motor skills

In the context of the Paralympic Games, for instance, it is essential to be able to group wheelchair users according to their motor abilities. Indeed, this classification allows to form teams based on functional and not physiological criteria and thus to guarantee fairness of competitions. However, the assessment of motor skills can be subjective, as it is sometimes based on observation of matches, or on a test set offered to the subject in a MWC. In both cases, subjects' mobility is appreciated by an expert who scores this ability on a scale.

In this section, we present a different, more objective approach for comparing subjects' mobility based on measurements made during their actual use of the instrumented MWC (FRET-2). Our aim was that the method used for the evaluation of motor abilities remains understandable by clinical and practical experts in the field. The experiments presented in this section were performed by 11 subjects with different anthropometric and physiological characteristics who propelled the FRET-2 (Table 4.3)

	F/M	Age	H. (cm)	W. (kg)	Dominant Membre	injury	Affected Vertebrae	Severity	Experience (years)	hours / day	days / week
S2	F	33	162	50	Right	Cervical	C5-C6	Complet	33		1
S3	M	34	178	78	Right	Thoracic	D4-D5	Complet	7	14	6
S4	M	47	180	80	Right	Thoracic	D8	Complet	0,5	5	7
S5	M	48	170	66	Left	Thoracic	D6	Complet	0,167	6	7
S7	M	27	180	78	Right	Lumbar	L3	Incomplet	0,583	12	7
S8	M	60	177	75	Right	Cervical	C5	Incomplet	3,5	12	7
S9	M	72	167	70	Right	Thoracic	D5	Complet	0,75	7	7
S10	F	26	163	68	Left	Lumbar	L1-L2	Incomplet	2	7	7
S11	M	38	177	85	Right	Thoracic	D4-D6	Incomplet	12	12	7
S12	F	22	165	54	Right	Thoracic	D5	X	0,917	10	7
S13	M	47	178	78	Right	Thoracic	D4	Complet	18,5	17	6

Table 4.3: Anthropometric and physiological characteristics of the subjects who participated in this study.

First, we applied the SAX-P symbolic representation to the z-moments (Mz) measured on the right and left wheels of FRET-2 during subjects' displacements. This method allowed to compare the propulsion cycles performed by the subjects during their displacements. Next, we compared subjects based on the relative occurrence frequency of each cycle type in each subject's displacements (Table 4.4)

	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>
<b>S02</b>	64	0	0	0	0
<b>S03</b>	7	43	35	2	28
<b>S04</b>	2	25	26	3	13
<b>S05</b>	9	30	25	25	3
<b>S07</b>	7	17	21	9	7
<b>S08</b>	26	2	1	49	0
<b>S09</b>	29	0	1	39	0
<b>S10</b>	6	21	31	5	10
<b>S11</b>	13	9	0	38	0
<b>S12</b>	22	2	0	42	0
<b>S13</b>	35	11	0	80	0

Table 4.4: Occurrence frequency of each cycle type in each subject's displacements.

This comparison is based on cosine similarity, a method that is used in the literature to compare text documents using the frequency with which words appear in these documents. Cosine similarity is defined as follows:

Let  $V_1$  and  $V_2$  be two integer vectors,

$$\cos(V_1, V_2) = \frac{V_1 \cdot V_2}{\| V_1 \| \times \| V_2 \|}.$$

Where  $\cdot$  is the dot product and  $\| d \|$  is the norm of the vector  $d$ .

For example, we will evaluate the similarity between subjects S02 and S03.

$$\begin{cases} V_{S02} = (64, 0, 0, 0, 0), \\ V_{S03} = (7, 43, 35, 2, 28). \end{cases}$$

First we calculate the dot product between  $V_{S02}$  and  $V_{S03}$ :

$$V_{S02} \cdot V_{S03} = 64 \times 7 + 0 \times 43 + 0 \times 35 + 0 \times 2 + 0 \times 28 = 448.$$

Then we calculate the vector norm  $\| V_{S02} \|$ , and  $\| V_{S03} \|$ .

$$\| V_{S02} \| = \sqrt{64 \times 64 + 0 \times 0 + 0 \times 0 + 0 \times 0 + 0 \times 0} = 64,$$

$$\| V_{S03} \| = \sqrt{7 \times 7 + 43 \times 43 + 35 \times 35 + 2 \times 2 + 28 \times 28} = 62.54.$$

The cosine similarity is then equal to

$$\cos(V_{S02}, V_{S03}) = \frac{448}{64 \times 62.54} = 0.112.$$

The similarity matrix between all wheelchair users is defined using the same method (Table 4.5). The cosine similarity of any subject with himself is logically equal to 1.

	<b>S02</b>	<b>S03</b>	<b>S04</b>	<b>S05</b>	<b>S07</b>	<b>S08</b>	<b>S09</b>	<b>S10</b>	<b>S11</b>	<b>S12</b>	<b>S13</b>
<b>S02</b>	1.000	0.112	0.052	0.190	0.232	0.468	0.597	0.152	0.316	0.464	0.398
<b>S03</b>		1.000	0.984	0.798	0.917	0.116	0.104	0.938	0.215	0.109	0.160
<b>S04</b>			1.000	0.841	0.950	0.129	0.107	0.977	0.230	0.120	0.173
<b>S05</b>				1.000	0.942	0.588	0.548	0.863	0.686	0.582	0.635
<b>S07</b>					1.000	0.405	0.392	0.977	0.472	0.396	0.434
<b>S08</b>						1.000	0.988	0.216	0.971	1.000	0.993
<b>S09</b>							1.000	0.208	0.929	0.987	0.967
<b>S10</b>								1.000	0.281	0.205	0.242
<b>S11</b>									1.000	0.973	0.992
<b>S12</b>										1.000	0.994
<b>S13</b>											1.000

Table 4.5: Similarity matrix between all wheelchair users.

The definition of a good representation and a good distance function are the most critical step in any distance based clustering method. Having doing so, we use a hierarchical clustering algorithm for the ease of interpretation of the results as suggested in [Kumar et al., 2002]. The clustering algorithm starts with each subject being in a singleton cluster and at each stage combine the closest pair of clusters into a single cluster until a threshold value is reached or a predefined number of clusters is obtained. In this experiment, we would like to distinguish subject with high, middle and low locomotion capabilities, we then consider three clusters (Fig. 4.16).

The results of this classification showed that only subject S02 is classified in cluster C1 and he is also the only subject who mainly performed A-type cycles. Subjects S03, S04, S07 and S10 were classified in cluster C2 and all of them performed a majority of B- and C-type cycles. Subjects S05, S08, S09, S11, S12, and S13 were classified in cluster C3 and these subjects mainly performed A- and D-type cycles. Subject S05 is a particular case because he also performed B- and C-type cycles, and he could thus have been classified in cluster C2. He was finally classified in cluster C3 because of the significant occurrence of D-type cycles during his displacements. These results allow us to deduce that the subjects who are in the same cluster have similar motor abilities, and that subjects belonging to cluster C2 have higher motor abilities than those belonging to cluster C3, who have higher motor abilities than

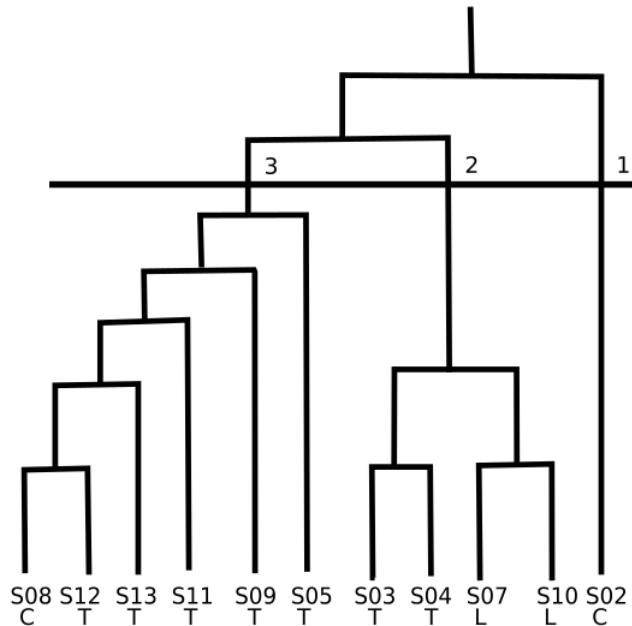


Figure 4.16: Classification tree of wheelchair users based on their propulsion abilities.

those belonging to cluster C1 ( $C_2 > C_3 > C_1$ ). We also infer that the majority presence of B- and C-type cycles characterize the propulsion of subjects with a significant motor capacity, that D-type cycles characterize the propulsion of subjects with an average motor capacity, and that A-type cycles characterize subjects with a weak motor capacity.

The additional information on wheelchair users (Table 4.3) indicates that the lonely subject in cluster C2 (S02) has indeed reduced physical abilities, but has been using his MWC once a week for 33 years. This long experience has allowed him to acquire a symmetrical propulsion technique, but not to increase his propulsion capacity. Subjects gathered in cluster C2 had either a thoracic lesion (S03 and S04) or a lumbar lesion (S07 and S10), whereas subjects gathered in cluster C3 had either a cervical lesion (S08) or a thoracic lesion (S11, S12 and S13). In this analysis, we wanted to establish that the data measured during locomotion with a wheelchair field-ergometer (FRET-2) provides a different but relevant view of wheelchair locomotion.

[Athanasios and Clark, 2009] established that motor abilities of wheelchair users primarily depend on their level of spinal cord injury. However, through these experiments, we show that assessing motor skills based on measurements made makes it possible to carry out a more detailed analysis of the motor skills of Manual Wheelchair users.

## 4.4 Conclusion

In this ongoing work, we proposed a method of symbolic representation of cyclic time series called SAX-P. This method is used to represent a cyclic time series as a string, each character representing a class of the cycles of the considered time series. The character strings obtained were then compared using Edith Distance and cosine similarity. Those comparisons allowed us to highlight the asymmetrical character of manual wheelchair locomotion, but also to show that manual wheelchair locomotion became more and more symmetrical with years of practice. The experiments also showed that knowledge of the physiological characteristics of subjects in manual wheelchairs was not sufficient for an evaluation of the motor abilities of manual wheelchair users and that an evaluation of locomotion requires measurements to be made in the actual situation of use of the manual wheelchair (Appendix A).

Ongoing research is devoted to the evaluation of the robustness of the symbolic representation (SAX-P) to the presence of uncertainty in the data and on applying SAX-P to a supervised classification of cyclic time-series in bio-mechanics.

### Key points

- We propose a symbolic representation of cyclic time series based on the properties of the cycles.
- We show that this symbolic representation improves the visualization and processing of cyclic time series from manual wheelchair locomotion.

### Communications :

- Siyou Fotso VS, Mephu-Ngouifo E, Vaslin Ph. Symbolic representation of cyclic time series: application to biomechanics. Constructive Machine Learning workshop at International Conference on Machine Learning , France, July 2015
- Siyou Fotso VS, Mephu-Ngouifo E, Vaslin Ph. Représentation symbolique de séries temporelles cycliques basée sur les propriétés des cycles : application à la biomécanique . Treizièmes Rencontres des Jeunes Chercheurs en Intelligence Artificielle (RJCIA 2015), Rennes, France, Jun 2015
- Siyou Fotso VS, E. M. Ngouifo, and P. Vaslin, “Symbolic representation of propulsion cycles in manual wheelchair locomotion,” Comput. Methods Biomed. Engin., vol. 18, no. sup1, pp. 2060–2061, 2015.
- C. Sauret, Siyou Fotso VS, J. Bascou, H. Pillet, E. Mephu-Ngouifo, P. Fodé, and P. Vaslin, “Cluster analysis to investigate biomechanical changes during learning of manual wheelchair locomotion: a preliminary study,” Comput. Methods Biomed. Engin., vol. 18, no. sup1, pp. 2058–2059, 2015.



# General conclusion and Future works

## Aims

Our primary objective throughout this work was to analyze manual wheelchair locomotion using measurements made by the sensors when using the wheelchair. This primary objective is divided into three specific goals: how to pre-treat time series to reduce their length and therefore their processing time, how to take into account the existence of uncertainty in the time series during their analysis and finally how to base the exploitation of time series on the cycles that constitute them. Each of these specific objectives has resulted in proposed models.

## Summary of contributions

### Reduce the length of time series with FDTW

We proposed a heuristic named FDTW that find a suitable parameter to use with the piecewise aggregate approximation algorithm with the aim to reduce the length of time series for classification purpose. This heuristic is based on Greedy Randomized Adaptive Search Procedure but defines a specific global search strategy. Extensive experimentation has been run out and shows that the compression with FDTW allows reducing the length of time series while keeping their main shape. Moreover, the compression with FDTW can enhance the accuracy of classification because it will enable avoiding pathological alignment with Dynamic Time Warping algorithm this amelioration is particularly perceptible with synthetic datasets.

### Dealing with uncertainty using FOTS score

We introduce a novel dissimilarity score for the comparison of time series named FOTS for Frobenius Correlation for Time series uShapelet discovery. This dissimilarity score is based on local correlation and allows to capture internal properties of the time series while being robust to uncertainty because its computation is based on

the comparison of eigenvectors of the autocorrelation matrices of time series. This score has been used for clustering purpose with UShapelet clustering algorithm and shows a significant improvement of the quality of clustering according to the rand Index.

### **Taking into accounts cycles with a symbolic representation**

Time series coming from wheelchair locomotion are cyclic due to the cyclic aspect of the wheelchair locomotion. The analysis of the wheelchair locomotion is based on those cycles, but none of the data mining models of the literature consider this aspect. We then proposed a symbolic representation of cyclic time series based on the properties of cycles that allow better visualization of the data and a better comprehension of the results obtained after the data mining process. We use this symbolic representation for the analysis of the wheelchair locomotion of eleven users, and this symbolic representation allows establishing that the wheelchair locomotion is asymmetric but this asymmetry get lower and lower with the years' of practice. This symbolic representation also allows a more precise evaluation of motor capabilities of manual wheelchair mainly based on the effort measure during their use of the manual wheelchair.

## **Future works and prospects**

### **Multidimensionality for a better characterization of the propulsion technic**

The analyses in this thesis are based on the Z moment of the wheels of the manual wheelchair. However, several measurements, including seat, back and footrest forces, were taken during the locomotion of the manual wheelchair users. Considering these signals could allow a better analysis of the subjects' movement and therefore suggests that we propose multidimensional data mining models that would simultaneously take into account all the measurements and their characteristics, namely their length, the presence of uncertainty, and the cyclical nature of specific measures.

### **Fuzziness for a more realistic categorization**

As we saw in Chapter 6 with subject S05, a subject can be very likely to belong to two or more groups. It would, therefore, be wise to associate each assignment with a degree of trust about the subject's group member. This data mining amounts to considering fuzzy approaches in the analysis of time series from manual wheelchair locomotion.

# Appendix A

## Analysis of manual wheelchair locomotion

### A.1 Introduction

Wheelchair locomotion concerns many people, for different reasons: genetic example: (myopathy), accidental (spinal cord injury, lower extremity amputee), degenerative (multiple sclerosis, poliomyelitis) or just related to the natural aging of locomotor functions (muscle degeneration, arthritis of the lower limbs, etc.). Then, in the 34 developed countries, it is estimated that 1% or 10,000,000 people require a wheelchair. In the 156 developing countries, it is estimated that at least 2% or 121,800,000 people require a wheelchair. Overall, of the 7,091,500,000 people in the world, approximately 131,800,000 or 1.85% need a wheelchair [Needs and The, 2016]. However, the use of manual wheelchair is not without risk.

### A.2 The problem of locomotion manual wheelchair locomotion

Although the wheelchair use improves the mobility of its users, doctors quickly realized that it often leads to sedentarization, and to related problems of obesity, diabetes, etc. Also, to promote daily physical activity, sport has been strongly encouraged [Machida et al., 2013]. However, intensive and prolonged sports practice in MWC can lead to specific injuries and pains [Johnson et al., 2004], especially in the shoulder, and at the elbow, wrist and hand. For instance in [Pentland and Twomey, 1991], the authors claimed that 73% of paraplegic individuals suffered from shoulder pain. In addition, prolonged sitting of users causes dermatological problems such as bed-sores or pressure ulcers, due to immobility, loss of sensitivity and incontinence. These symptoms are recognized as a major cause of discontinuation of wheelchair use [Van der Woude et al., 2006] [Ville and Winance, 2006], thus the sedentarization of users. [Lundqvist et al., 1991] showed that upper limb pain was the only

factor correlated with poor quality of life in MWC users. The challenge for the therapist then encourages a daily practice of physical activity, adapted to wheelchair users, while limiting orthopedic problems and thus promote the use of the MWC over time.

Given the problems faced by manual wheelchair users at the level of their autonomy and health, van der Woude et al. [[van der Woude and de Groot, 2005](#)] [[Woude et al., 1986](#)] summarized the issues of manual wheelchair locomotion research into three main areas:

- Improving the interface between the subject and his manual wheelchair, that is, the ergonomics and the adequacy of the system {subject + MWC} with the external physical environment (ramps, lifts, corridor widths, etc.).
- The improvement of the MWC regarding the design and the mechanical principles of propulsion;
- **Improving the subject's physical abilities**, that is, improving propulsion techniques, as well as rehabilitation techniques and training programs.

After the construction of a measuring tool: a wheelchair field ergometer, biomechanical works have been conducted in LIMOS to identify and quantify traumatic factors such as [[De Saint Remy, 2005](#)] [[Sauret, 2010](#)].

## A.3 Tools to evaluate manual wheelchair locomotion

This section summarises different tools designed over the last 60 years to measure the efforts made by subjects moving in a MWC. We will place particular emphasis on the wheelchair Ergometer designed and manufactured at LIMOS, which is at the origin of the time series that are the subject of our analysis throughout this thesis.

### A.3.1 Crank Ergometers

Crank ergometers allow a subject to manually operate a crankset connected to the flywheel of an ergo-cycle. The speed is determined by measuring the rotation speed of the flywheel, whose diameter is known, or by imposing a cadence, in which case the rotation speed is considered constant. Crank ergometers established that the mechanical work of the upper limbs is less efficient than that of the lower limbs and also that the physical capacities evaluated by the maximum oxygen consumption of MWC users depended on their level of spinal injury (cervical, thoracic or lumbar injury)<sup>1</sup>. One of the main limitations of crank ergometers is that the motion measured

---

<sup>1</sup>This assertion will be commented later in chapter 4

from a crank ergometer is not representative of the MWC propulsion motion, most of which is propelled by handrims [0Åstrand and Saltin, 1961] [Bergh et al., 1976] [Stenberg et al., 1967].

### A.3.2 Roller Ergometers

To reproduce more precisely the specificities of MWC locomotion, Brouha and Krobath[Brouha and Krobath, 1967], as early as 1967, used a roller ergometer to measure cardiac and respiratory responses during continuous MWC exercice. This tool consisted of a platform on which were fixed two rollers, each rotating around an axis and on which rested the rear wheels of a real MWC. The MWC frame was attached to the ergometer, and the subjects simulated locomotion by applying forces to the handrims, causing the rear wheels of the MWC and the rollers to rotate.

In 1971, Stoboy et al. [Stoboy et al., 1971], using an ergometer inspired by that of Brouha and Krobath, quantified the mechanical power (in watts) produced by the user from the relationship between oxygen consumption and mechanical power calculated during an incremental exercise on a crank ergometer.

The problem with roller ergometers of [Brouha and Krobath, 1967]

[Stoboy et al., 1971] was that they did not take into account the influence of the inertia of translation encountered by the Subject when he moves. To take this phenomenon into account, the rollers have been connected to a small flywheel. However, the both rear wheels were on the same rollers, which did not allow to measure the differences in propulsion between the right and left wheels to be explored [Brouha and Krobath, 1967] [Stoboy et al., 1971].

Then [Langbein and Fehr, 1993] [Langbein et al., 1993] [Langbein et al., 1994] designed a new roller ergometer called the Wheelchair Aerobic Fitness Trainer (WAFT), which had an access ramp to facilitate subject and MWC installation (Figure A.1). When the latter was attached to the ergometer, its rear wheels rested on three rollers each, which made it possible to differentiate the forces applied to the right and left wheels<sup>2</sup>.

Other roller ergometers have also been developed over the last four decades and particularly in the last fifteen years: Eagle Wheelchair Roller [Kerk et al., 1995], Bromking Turbo Trainer [Goosey-Tolfrey et al., 2001] [Goosey-Tolfrey et al., 2001] [Price and Campbell, 1999] or very recently the "Computer Monitored Wheelchair Dynamometer" [Cooper et al., 2003] [DiGiovine et al., 2001]. Other braking systems have been used, such as mechanical braking using a friction belt on a flywheel [Goosey et al., 1998] [Kulig et al., 2001] [Rodgers et al., 1994](Figure A.2), an electric motor creating a frictional moment around the roller rotation axes [Coutts and Stogryn, 1987] [Kerk et al., 1995] [Patterson and Draper, 1997]

---

<sup>2</sup>This separation is essential to establish the dissymmetry of wheelchair locomotion and is discussed in more details in chapter 4

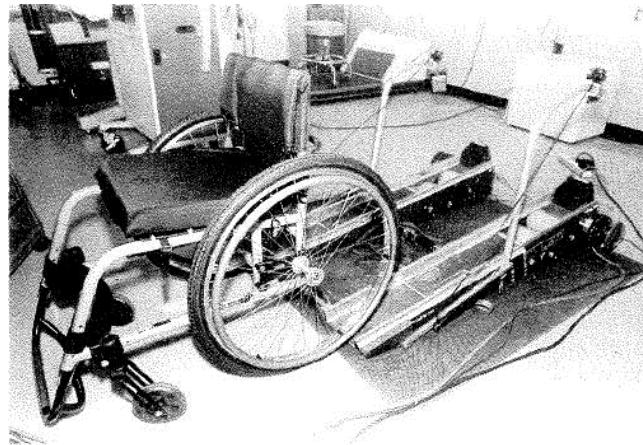


Figure A.1: Wheelchair Aerobic Fitness Trainer (WAFT) photograph [Langbein and Fehr, 1993].

[Vanlandewijck et al., 1999] or an isokinetic apparatus [Ruggles et al., 1994]. To determine the speed, angular position sensors [Brouha and Krobath, 1967] [Coutts and Stogryn, 1987] [Coutts, 1990] [Patterson and Draper, 1997] [Rodgers et al., 1994], optical encoders [Devillard, 1999] [Devillard et al., 2001] [Langbein et al., 1993] [Langbein et al., 1994] [Newsam et al., 1996] [Theisen et al., 1996], tachometers [Cooper, 1990] [Kerk et al., 1995] [Masse et al., 1992] [Vanlandewijck et al., 1999] or speedometers [Goosey et al., 1998] [Rodgers et al., 1994] were used.

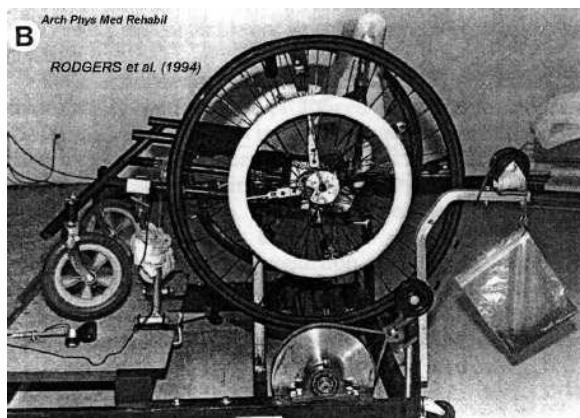


Figure A.2: Picture of a wheelchair on a roller ergometer with mechanical braking by friction belt on a flywheel. [Rodgers et al., 1994].

The main advantage of roller ergometers is that they allow subjects to be studied with their own MWC. Moreover, they occupy a little space in the laboratory and allow the MWC to be completely immobilized, thus ensuring the stability of the

subject on the MWC and facilitating the measurement of various physiological parameters. However, the various methods for estimating external mechanical power used up to now still need to be refined to better evaluate this parameter. Furthermore, the comparison between the results of studies carried out with different roller ergometers and different mechanical models must be done with caution since the parameters neglected or taken into account are not all the same.

### A.3.3 Treadmill

Like roller ergometers, the main advantage of treadmills is that they allow subjects to be studied with their own MWC. Since the four wheels of the MWC roll on the belt, the rolling friction forces are most certainly equivalent to those existing on the ground. Unlike roller ergometers, treadmills allow to define a rolling speed of the belt and also a slope, that is, an inclination of the treadmill with respect to the horizontal. The main disadvantage of a treadmill comes from the steering problem related to the control of the trajectory: Indeed, a subject could drift and be ejected from the treadmill; to remedy this, railings have been installed on both sides using a surface strip that limits lateral movements [[Claremont and Maksud, 1985](#)]. However, it has still not been demonstrated that the propulsion technique used was identical on a treadmill and on the ground.



Figure A.3: Exercise testing on a motor driven treadmill  
[[Van der Woude et al., 2006](#)]

### A.3.4 Wheelchair simulators

To overcome the problems related to rolling resistance, some researchers chose to fix the rear wheels of the MWC without contact with the ground, on a rigid and fixed chassis on which the Subject could sit. The advantage of MWC simulators is that they can test different settings such as seat position or rear wheel camber

angle, for example. The mechanical propulsion model is also simplified compared to roller ergometers and conveyor belts, which allows better quantification of work and external mechanical power. However, the influence of the Subject's movements on the seat is not taken into account. This aspect is the major disadvantage of the simulators because neither the forces of resistance to the advance nor the kinematics of the MWC is modified according to the movements of the Subject on the seat.

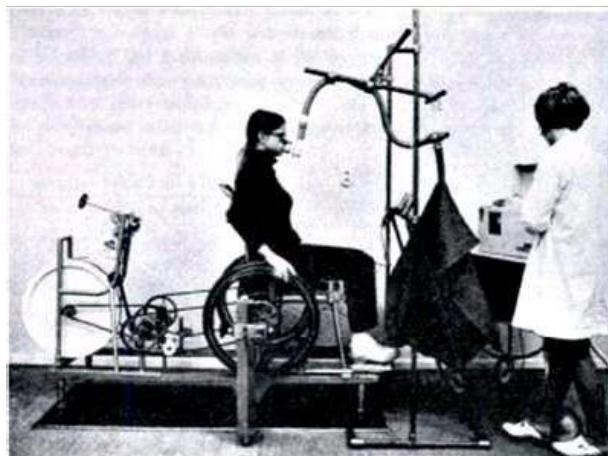


Figure A.4: Photograph of an experiment on a simulator connected to a flywheel ([Brattgård et al., 1970])

### A.3.5 Wheelchair Field-Ergometer

To analyse the efficiency of wheelchair propulsion, a Wireless Wheelchair Ergometer (WWE or FRET-1) equipped with several sensors has been manufactured [Dabonneville et al., 2005]. The sensors installed on the wheelchair measure the physical stresses applied to the MWC during actual use and record them.

The sensors are located on the right and left wheels of the MWC, on the footrest, on the seat and the backrest (Figure A.5). These sensors measure the torques applied to each of the systems mentioned above. Other sensors installed on the FRET-2 are used to measure the kinematic parameters (speed, acceleration) of the movement of the MWC.

The measurements recorded by the sensors, which are the purpose of our analysis, consist of 44 attributes; of which 30 represent the dynamic parameters ( $F_x$ ,  $F_y$ ,  $F_z$ ,  $M_x$ ,  $M_y$ ,  $M_z = 6$  components  $\times$  5 dynamometers). The 14 other attributes represent the kinematic parameters of the MWC and its position relative to the Earth's magnetic North.

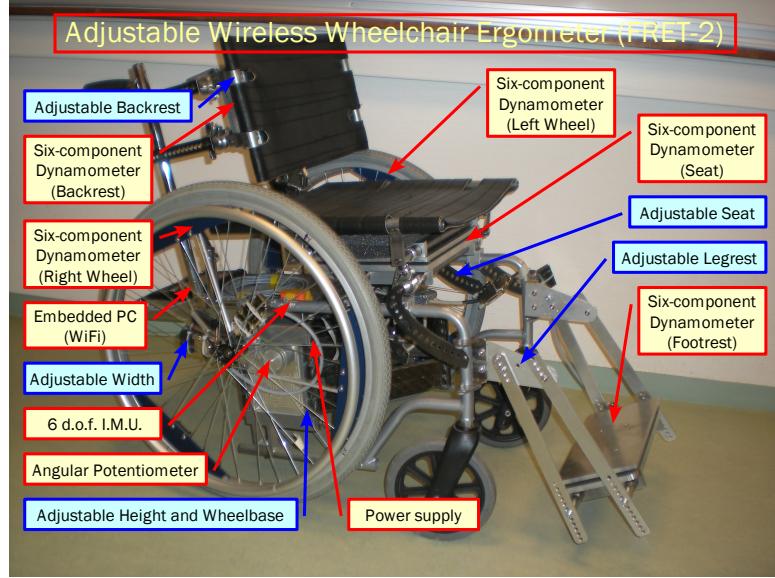


Figure A.5: Captioned picture of the adjustable wireless wheelchair ergometer (FRET-2).

### The force and torque dynamometer

A "torsor" is a mathematical object that characterizes the efforts applied on – or by – a solid. It is composed of two vectors, which have three components each: the three components ( $F_x$ ,  $F_y$ ,  $F_z$ ) of the resulting force ( $F$ ) and the three components ( $M_x$ ,  $M_y$ ,  $M_z$ ) of the resulting moment ( $M$ ) that are applied to this solid along three orthogonal axes ( $x$ ,  $y$ ,  $z$ ). To measure the six components of the "torsor" applied by wheelchair users on both handrims during their actual displacements on the ground, an original force and torque dynamometer has been designed, built and installed on both rear wheels of a MWC field ergometer – named FRET-2 (Fig. A.6). This dynamometer is composed of three bidirectional force sensors that measure all the forces applied to the handrim and are then used to compute the three components of the resulting force and the three components of the resulting moment.

#### Legend of Fig. A.6:

- $R^*(O, x, y, z)$ : moving reference frame linked to the wheel;
- $O$  : centre of the wheelchair rear wheel;
- $R$  : radius of the wheelchair rear wheel;
- $A, B, C$  : locations of the three two-component force transducers;
- $u_1, \dots, u_6$  : unit vectors of the six force transducers;

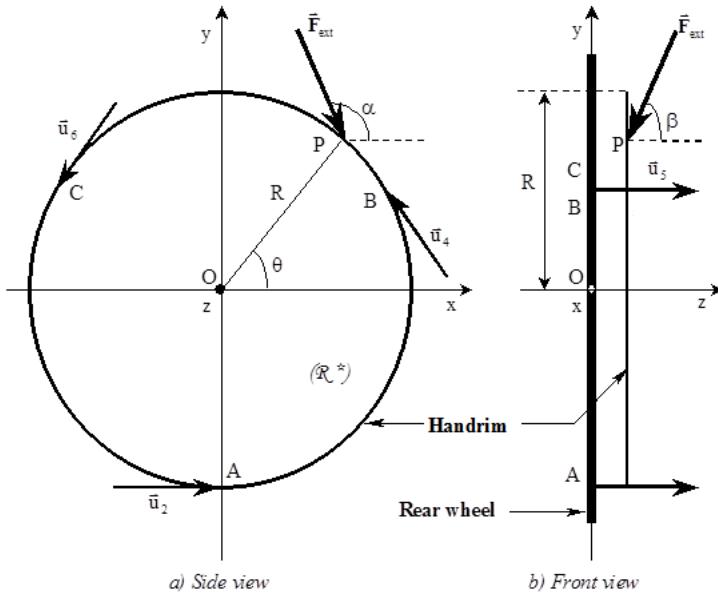


Figure A.6: Schematic principle of the six-component force and torque sensor (patent WO 1995001556 A1) mounted of both rear wheels of the MWC field ergometer used in this study (FRET-2).

- $\mathbf{F}_{ext}$ : external force applied by the user on the handrim;
- $P$  : point of application of  $\mathbf{F}_{ext}$  on the handrim;
- Angles:  $a = (ux, \mathbf{F}_{ext})$ ;  $b = (uz, \mathbf{F}_{ext})$ ;  $q = (ux, OP)$

The wheel dynamometer used in that study is based on a mechanical principle already applied for the design of a six-component dynamometer used for the measurement of the forces and torques applied by the pole-and-vaulter system in the vaulting box during the pole vault.

According to that principle, the handrim is assumed to be rigid and firmly fixed on three two-component force transducers designed to measure the handrim displacement with respect to the wheel. In that approach, the handrim is considered as hanging on the wheel through the force transducers. Each transducer measures one component of the resulting propulsive force in the tangential direction of the wheel and one in the direction perpendicular to the plane of the wheel (Fig. A.6). The vectorial sum of all these components is equal to the resulting propulsive force in the moving reference frame  $R^*$  linked to the wheel (Fig. A.6). This derives from the fact that the handrim is static in  $R^*$  with respect to the wheel.

When an external force  $\mathbf{F}_{ext}$  is applied on the handrim, it is instantaneously transmitted to the six force transducers so that each of them simultaneously measures a local force  $\mathbf{F}_i$  ( $i = 1$  to 6). As the transducers behave as springs, which stiffness  $k_i$  are determined through the calibration procedure, the measurement of

the displacements  $m_i$  by the strain gauges allows to compute the values of  $F_i$  using Hooke's law

$$F_i = k_i m_i. \quad (\text{A.1})$$

Finally, the force and torque components created by  $\vec{F}_{ext}$  are calculated by equation A.2

$$\vec{F}_{ext} = \sum_{i=1}^6 \vec{F}_i \iff [F_{ext}] = [A] [k_i] [m_i] \iff [F_{ext}] = \begin{bmatrix} F_x \\ F_y \\ F_z \\ M_x \\ M_y \\ M_z \end{bmatrix} \quad (\text{A.2})$$

Several mechanical and kinetic parameters can be computed from the forces and torques measured by the six-component dynamometers mounted on the MWC field ergometer [Sauret, 2010]. All of them have a specific and useful meaning as their relationships are defined by a complete mechanical (i.e. dynamic and kinematic) model of wheelchair propulsion. However, because of their number and their complexity, they can only be analysed and interpreted by specialists in biomechanics. To overcome this drawback, in the present study, relevant mechanical information has been extracted from only some dynamic data (z-moments) recorded by both rear wheel dynamometers of an instrumented MWC (FRET-2) [and then used to group subjects in homogeneous clusters, which have been compared to clinical injury levels.

## A.4 Knowledge discovery on wheelchair time series

After the construction of these measuring instruments (FRET-1 and FRET-2), they have been used to measure the efforts made by MWC in actual conditions of wheelchair locomotion. Thus, several experiments have been conducted with several subjects, where the efforts produced during actual wheelchair locomotion were measured with the FRET-2. The abundance of the recorded measurements highlighted the problem of the exploitation of these measures for knowledge extraction. Two main and complementary approaches can be used to analyze measurements from MWC locomotion. The first is to use mechanical models to calculate the physical

parameters of motion and the second is to use data mining models to exploit measurements. In this section, we present the contributions of these two approaches, which will allow us to position our work.

Manual wheelchair locomotion causes significant mechanical stresses in the upper limbs. To remedy this problem, biomechanical studies have been conducted to identify and quantify traumatic factors such as:

- The doctoral thesis of Nicolas DE SAINT REMY (2005) [De Saint Remy, 2005] who proposed a mechanical model relating the forces applied to a MWC and its displacement ( Figure A.7 ). This work particularly highlighted the fact that wheelchair acceleration is a function of subject's movements.

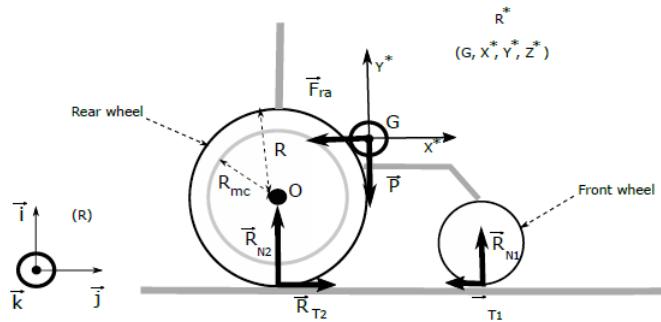


Figure A.7: Balance of forces applied to a manual wheelchair during propulsion. For the clarity of the figure, the analysis of the movement of the subject + MWC system is reduced to that of the system's centre of gravity, G [De Saint Remy, 2005]

- The doctoral thesis of Christophe SAURET (2010) [Sauret, 2010] who proposed a method of calculating the mechanical power developed by manual wheelchair users to move. This model analyzes the kinetics of the subject + MWC system. For that purpose, the author developed a detailed kinematic model of the subject + MWC system (Figure A.8) allowing to record their movements with a 3D motion analysis system during actual locomotion on the ground.

More and more works in the literature suggest using the tools developed in data mining for a better understanding of human locomotion. For example, in [van der Slikke et al., 2017], the authors ask whether advances in data science and technology could provide a different and perhaps more objective view of the analysis of wheelchair users' motor abilities. On one hand, technical advances have made it possible to measure the efforts made by wheelchair users during their movement using sensors. On the other hand, datamining models have been proposed and allow to perform several task on the data (clustering, classification, ...).

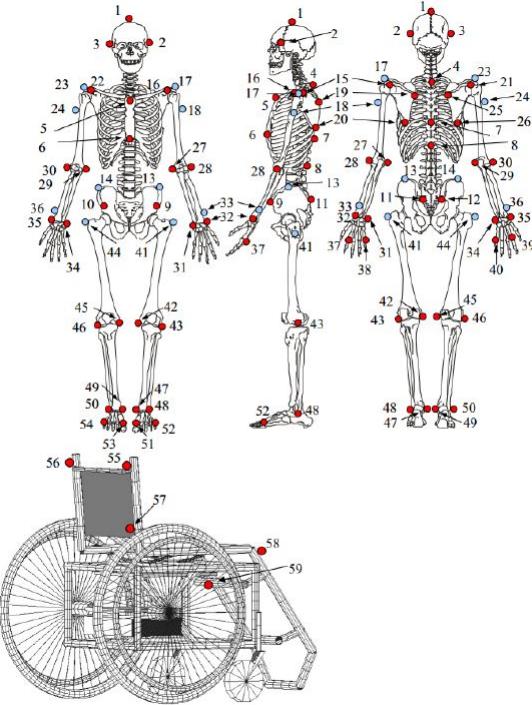


Figure A.8: Markers for kinematic analysis of manual wheelchair locomotion [Sauret, 2010]

In [Faria et al., 2012] the authors explained how they used robotics and data mining knowledge to build an Intelligent Manual Wheelchair, which can be controlled from multiple interfaces: joysticks, facial expressions, voice commands, head movements. Since Intelligent Wheelchair users have various characteristics, a series of tests have been carried out to classify them and to define profiles that allow the MWC to be adjusted appropriately for each user.

In [Athanasios and Clark, 2009] the authors presented a model based on Bayesian networks to improve the medical treatment of patients in wheelchairs with a spinal cord injury. Indeed, the treatment of these patients is based on the level of spinal cord injury and symptoms. A lesion in the spine has three consequences: an inconsistency of the bowel, an inconsistency of the bladder, a loss of the skin sensitivity. The higher the lesion, the more widespread its effects on patients are. Thus a patient with a low lesion will affect the patient's legs and a patient with a high lesion will see his four limbs affected. Because of this loss of sensitivity, symptoms observed in the patient are often incomplete, which introduces uncertainty into the diagnosis that is captured by Bayesian networks and conditional probabilities.

## A.5 Conclusion

Throughout this chapter, we showed that there is a large number of MWC users in the world and that it is crucial to analyze this particular means of locomotion to improve the living conditions of people moving in a MWC. We have presented the main tools designed and manufactured for wheelchair locomotion analysis and some previous works that used data mining mechanics models to improve the study of wheelchair locomotion or to help physicians to diagnose the adverse for the adverse effects treatment of spinal cord injury causing paralysis and requiring the use of MWC.

In the scientific literature, mechanical or data mining models are used for manual wheelchair locomotion analysis. In this thesis, however, we want to **design** data mining models that is able to take into account both the specificities of MWC locomotion data and their use to analyze wheelchair locomotion from a new point of view. In the next chapter, are presented the existing works in the literature of knowledge extraction on time-series, which will allow us to identify and then choose useful approaches for the analysis of wheelchair locomotion.

### Key points

- The first section of the chapter A presents the three central research questions related to the analysis of Manual Wheelchair locomotion

We explain why this thesis is focused on the study and improvement of motor skills of Manual Wheelchair users.

- The second section of the chapter A presents the measurement tools built to measure the efforts made by manual wheelchair users during their locomotion. Here we insist on the Wheelchair Field ergometer, which produces the time series that we use throughout our work for the analysis of manual wheelchair locomotion.
- The third section of the chapter A presents the mechanical models and data mining models used in the literature for manual wheelchair locomotion analysis.

# An optimal approach for time series segmentation: Application to the supervised classification

## B.1 Introduction

Time series databases are often huge. This is particularly the case of the Large Synoptic Survey Telescope (LSST) database which records data from of telescopes [Izs, 2016]. It has billions of time series (10 Petabytes). The time series recorded in these databases are sometimes very long. Another example is the SACR-FRM project that uses sensors to measure the efforts of a manual wheelchair user at a frequency of 100 Hz [SAC, 2016]. Ten minutes recording time series of 60 000 measurements. Faced with this, several scientific works were carried out with the aim of reducing the storage space of time series and accelerating their treatment. A widely used approach is to change the representation of time series to reduce their length. This technique was introduced by Agrawal et al. [Agrawal et al., 1993]; he uses the discrete Fourier transform to obtain a compact representation of the time series. Other methods have also been used: the decomposition in eigenvalue [Wu et al., 1996], the discrete wavelet transform [Chan and Fu, 1999b] and the piecewise approximate aggregation (PAA) [Keogh et al., 2001c]. This last method has shown its effectiveness compared to previous three because it is easier to understand, to implement, but also faster and allows to build indexes in linear time. PAA suggests splitting the time series into segments of the same length, then replace each segment by the average of its points. This method generates a compact representation, able to have a few segments as possible to reduce space storage and processing time of time series. However, a too compact representation distorts the time series and causes a loss of information. How then to choose the right number of segments to consider? Our work is based on a simple observation: the use of the arithmetic

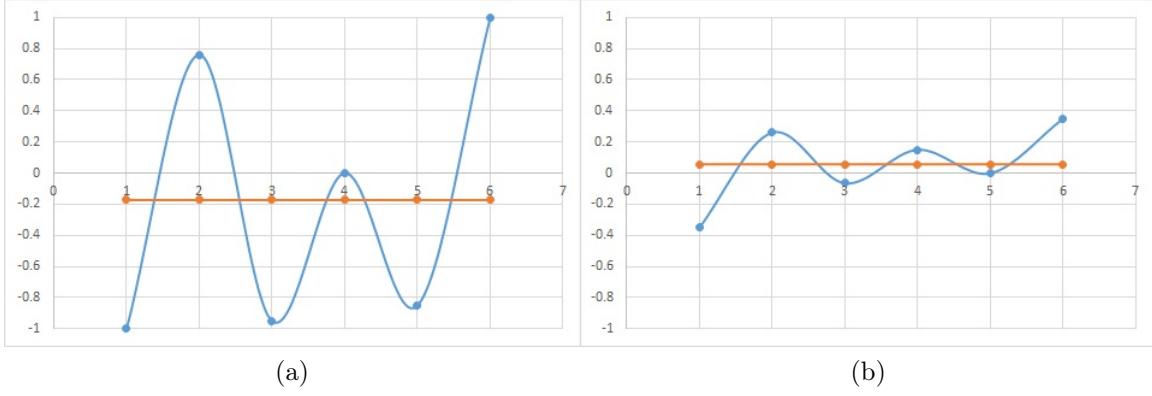


Figure B.1: These figures show the average of two segments. In the first case (a) the data points of the segment are far from the average, in the second case (b) they are close to the average. Replace data points of a segment by their average introduces an error that can be measured from the gap between the points and the average.

average is relevant when the variance of the population is small as illustrated by the figure C.2.

We define here a minimal bound for the number of segments to be considered, and we propose an algorithm which allows choosing the number of segments which minimizes their mean squared error, with the aim to reduce the length of the time series without altering the information they contain.

The rest of this chapter is organized as follows: the ?? section presents a formal definition of our problem and an algorithm used to solve it; the section B.3 presents and comments the results of the experiments and the section B.4 concludes the paper and presents perspectives for this work.

## B.2 Granularity of time series segments

### B.2.1 Notations and definitions

**Definition 1:** A **time series** or **time series**  $X = x_0, x_1, \dots, x_m$  is a sequence of numerical values representing the evolution of a specific quantity over time.  $x_m$  is the most recent value.

**Definition 2:** A segment  $X_i$  of length  $l$  of the time series  $X$  of size  $m$  ( $l < m$ ) is a sequence consisting of  $l$  consecutive variables  $X$  beginning at the position  $i$  and ending at the position  $i + l - 1$ . We have:  $X_i = x_i, x_{i+1}, \dots, x_{i+l-1}$

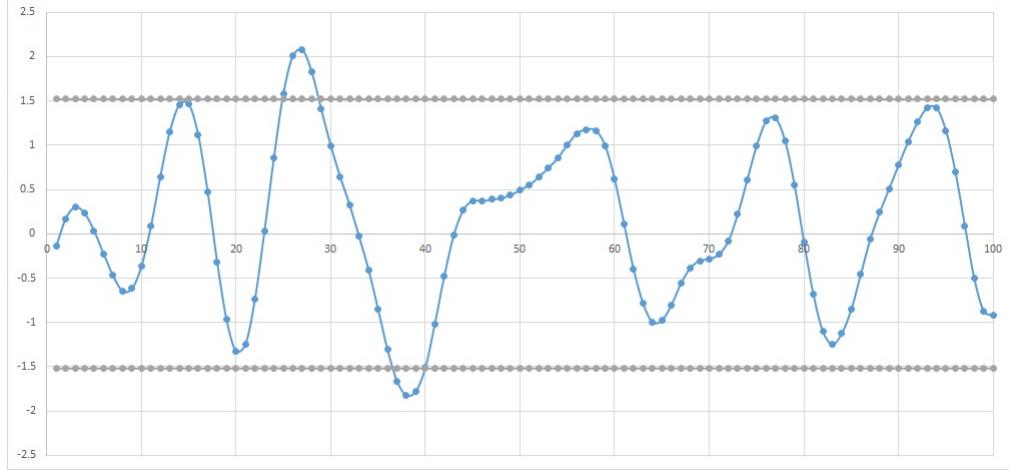


Figure B.2: This figure shows the first 100 points of the first time series of the fordA dataset available in the UCR [Chen et al., 2015] database. Time series are normalized. The two horizontal lines delimit the interval corresponding to twice the standard deviation and minus two times the standard deviation of the points of the time series. We can observe that the points outside this range are at the ends of the time series.

**Definition 3:** The arithmetic mean of the data points of a segment  $X_i$  of size  $l$  is denoted  $\bar{X}_i$  and is defined by

$$\bar{X}_i = \frac{1}{l} \sum_{j=0}^{l-1} x_{i+j}.$$

### B.2.2 Information theory and minimum number of segments

To mitigate the effects of noise during time series processing, Keogh and Kasetty [Keogh and Kasetty, 2003b] recommend that they are normalized. Normalizing the time series makes them compatible with a normal distribution [Lin et al., 2007]. In this case, 95 % of the points in the time series are between minus two times the standard deviation ( $\sigma$ ) and twice the standard deviation of the points, and thus 5 % of the points of the series are outside this range. These points correspond to the ends of the series as shown in the figure B.2.

### B.2.3 Notations and definitions

**Definition 1:** A **time series** or **time series**  $X = x_0, x_1, \dots, x_m$  is a sequence of numerical values representing the evolution of a specific quantity over time.  $x_m$  is the most recent value.

**Definition 2:** A segment  $X_i$  of length  $l$  of the time series  $X$  of size  $m$  ( $l < m$ ) is a sequence consisting of  $l$  consecutive variables  $X$  beginning at the position  $i$  and ending at the position  $i + l - 1$ . We have:  $X_i = x_i, x_{i+1}, \dots, x_{i+l-1}$

**Definition 3:** The arithmetic mean of the data points of a segment  $X_i$  of size  $l$  is denoted  $\bar{X}_i$  and is defined by

$$\bar{X}_i = \frac{1}{l} \sum_{j=0}^{l-1} x_{i+j}.$$

Also, information theory tells us that the amount of information relating to an event is equal to  $-\log_2(p)$  where  $p$  is the probability of the event [Shannon, 2001]. In other words, a very likely event ( $p \rightarrow 1$ ) brings less information than an unlikely event ( $p \rightarrow 0$ ). Therefore, a point outside the interval  $[-2\sigma, 2\sigma]$  provides more information than a point in that range. Indeed, the probability that one point is in the range is 0.95 while the probability that one point is out of range is  $0.05 = \frac{1}{20}$ .

If we choose a minimum number ( $\alpha$ ) of segments less than 5 % of the length of the time series, we risk to aggregate the data points within the interval  $[-2\sigma, 2\sigma]$  and those outside this range. This will have two consequences: firstly, this aggregation will alter the information carried by data points. Secondly, this aggregation will increase the variance of the segments obtained. So we chose to consider 5% of the number of points in the time series as the minimum number of segments. This allows us to define the following functions of  $\mathbb{N} \rightarrow \mathbb{N}$ :

$$\begin{aligned} \alpha : n \mapsto \alpha(n) &= \begin{cases} \lfloor \frac{n}{20} \rfloor & \text{if } \lfloor \frac{n}{20} \rfloor \geq 2 \\ 2 & \text{otherwise} \end{cases} \\ \beta : n \mapsto \beta(n) &= \left\lfloor \frac{n}{2} \right\rfloor \end{aligned}$$

$\beta$  gives the maximum number of segments. Indeed, a segment is made up of at least 2 points, so there is at most  $\left\lfloor \frac{n}{2} \right\rfloor$  segments. The number of segments  $W$  that we will consider is greater than or equal to  $\alpha(|X|)$  and less than or equal to  $\beta(|X|)$ . The next subsection explains how we choose the value of  $W$ .

#### B.2.4 Minimize the squared error to choose the number of segments

After dividing into segments, we replace each segment by the average of the data points that constitute it. The variance between the points of each segment can be measured from the mean squared error. Our problem is therefore the following:

Let  $X = x_0, x_1, \dots, x_n$  a time series of size  $n$ , look for  $W \in \mathbb{N}$  such that  $\alpha(n) \leq W \leq \beta(n)$  and  $\frac{1}{n} \sum_{i=1}^W \sum_{j=(i-1)k}^{ik} (\bar{X}_i - X_j)^2$  is minimal. Where  $\bar{X}_i$  is the arithmetic mean of a segment of length  $k$ .

To solve this problem, we propose an algorithm that proceeds as follows:

1. For each value of W, with  $\alpha(n) \leq W \leq \beta(n)$ 
  - (a) Calculate the squared error of each segment  $X_i = x_i, x_{i+1}, \dots, x_{i+l-1}$ ;
  - (b) Calculate the mean of the quadratic errors;
2. The value of W returned is the one that produces a minimum average squared error;

Algorithm 4 details the previous principle.

---

**Algorithm 4:** optimalNumberOfSegment

---

**Input:** length\_min, length\_max : respectively the minimal and the maximal length of a segment.

v : a time series

**Output:** The optimal number of segment to be use with Piecewise Aggregate approXimation

```

1 function optimalNumberOfSegment(length_min, length_max, v)
2   len_v  $\leftarrow$  length(v)
3   n  $\leftarrow$  length_max - length_min + 1
4   forall i  $\in \{\text{length\_min}, \dots, \text{length\_max}\} do
5     x[j, 1]  $\leftarrow$  i
6     z[j, 1]  $\leftarrow$  (1/len_v) * sum_SSE(v, i)
7     computation of the error j  $\leftarrow$  j + 1
8   ind_min  $\leftarrow$  indice_minimun(z)
9   return floor(len_v/x[ind_min, 1])$ 
```

---

**Complexity of the algorithm** The calculation of the squared error of a segment is done in  $O(\lfloor \frac{n}{W} \rfloor)$ .

The time complexity of calculating the mean squared error for segment splitting is  $O(n)$ .

The number of segments varies from  $\lfloor \frac{n}{20} \rfloor, \lfloor \frac{n}{19} \rfloor \dots \lfloor \frac{n}{2} \rfloor$ . There are 19 possible divisions in segments. The time complexity of calculating the value of W which minimizes the error mean squared is  $19 \times O(n) = O(n)$ .

To exploit the compact representations of the time series, we must be able to compare them. The next subsection presents how to compare compact time series that we get.

---

**Algorithm 5:** sum\_SSE

---

**Input:**  $v$  : a time series.  
 $\text{nbPoints}$  : the length of a segment  
**Output:** The sum of squares error associated with a segment length

```

1 function sum_SSE( $v, \text{nbPoints}$ )
2    $n \leftarrow \text{length}(v)$ 
3    $\text{ind\_debut} \leftarrow 1$ 
4    $\text{aux\_se} \leftarrow c()$ 
5    $\text{tab\_indices\_debut} \leftarrow c()$ 
6    $i \leftarrow 0$ 
7   while ( $\text{ind\_debut} + \text{nbPoints}) \leq n$  do
8      $\text{tab\_indices\_debut}[i] \leftarrow \text{ind\_debut}$ 
9      $\text{ind\_debut} \leftarrow \text{ind\_debut} + \text{nbPoints}$ 
10     $i \leftarrow i + 1$ 
11    $m \leftarrow \text{length}(\text{tab\_indices\_debut})$ 
12   forall  $i \in \{1, \dots, m\}$  do
13      $\text{aux\_se}[i] \leftarrow \text{SSE\_segment}(v, \text{nbPoints}, \text{tab\_indices\_debut}[i])$ 
14   return  $\text{sum}(\text{aux\_se})$ 
```

---

### B.2.5 Dynamic Time Warping Algorithm and comparison of compact representations

The dynamic time warping algorithm [Keogh and Ratanamahatana, ] allows to carry out a non-linear correspondence between two time series by minimizing the distance between the two. It proceeds as follows: Let  $X$  and  $Y$  be two time series;

$$X = x_1, x_2, \dots, x_n;$$

$$Y = y_1, y_2, \dots, y_m.$$

To align them, the algorithm constructs a matrix  $n \times m$  where the element  $(i, j)$  of the matrix corresponds to the square distance  $(x_i - y_j)^2$  which is the alignment between  $x_i$  and  $y_j$ . To find the best alignment between the two time series, we build the path in the matrix that minimizes the sum of the square distances. This path is calculated by dynamic programming from the following recurrence:

$$\gamma(i, j) = d(x_i, y_j) + \min\{\gamma(i - 1, j - 1), \gamma(i - 1, j), \gamma(i, j - 1)\},$$

where  $d(x_i, y_j)$  is the square of the distance contained in the cell  $(i, j)$  and  $\gamma(i, j)$  is the cumulative distance at the position  $(i, j)$  which is calculated by the sum of the square of the distance to the position  $(i, j)$  and the minimum cumulative distance of its three adjacent cells.

Piecewise aggregate approximation provides Euclidean-based distance measurement to compare compact representations. However, we chose to use the dynamic time warping algorithm. For the following reasons:

1. The dynamic time warping algorithm is known to have the best performance for sequence alignment in several areas: in robotics, biometrics, music, climatology, aviation, in gesture recognition, cryptanalysis, astronomy, exploration space [Rakthanmanon et al., 2012b].
2. Piecewise aggregate approximation of the time series leads to temporal deformation. Indeed, with two time series of length  $n$ , we can apply our algorithm to the first time series, then reduce it to  $N_1$  segments and reduce the second to  $N_2$  segments with  $N_1 < N_2$ .

## B.3 Results and Discussion

First, we present the datasets used during the experiment. Then we evaluate the performances of the proposed methods in terms of reduction of the length of time series and classification errors.

### B.3.1 Datasets

We performed tests on 85 datasets that come from the UCR database [Chen et al., 2015]. Detailed information on the datasets is presented in the table B.1. In the UCR database, each data set is divided into a learning set and a test set. Datasets contain between 2 and 60 classes and the time series of these datasets have lengths that range from 24 to 2709 points. The table B.1 presents a detailed description of the datasets. The following paragraph presents the assessment of the performance of our algorithm on these datasets.

N	Name	Nb. of classes	Size of training set	Size of testing set
1	50Words	50	450	455
2	Adiac	37	390	391
3	ArrowHead	3	36	175
4	Beef	5	30	30
5	BeetleFly	2	20	20
6	BirdChicken	2	20	20
7	Car	4	60	60
8	CBF	3	30	900
9	ChlorineConcentration	3	467	3840
Continue to the next page				

N	Name	Nb. of classes	Size of training set	Size of testing set
Following ...				
10	CinC_ECG_torso	4	40	1380
11	Coffee	2	28	28
12	Computers	2	250	250
13	Cricket_X	12	390	390
14	Cricket_Y	12	390	390
15	Cricket_Z	12	390	390
16	DiatomSizeReduction	4	16	306
17	DistalPhalanxOutlineAgeGroup	3	139	400
18	DistalPhalanxOutlineCorrect	2	276	600
19	DistalPhalanxTW	6	139	400
20	Earthquakes	2	139	322
21	ECG	2	100	100
22	ECG5000	5	500	4500
23	ECGFiveDays	2	23	861
24	ElectricDevices	7	8926	7711
25	Face (all)	14	560 1	690
26	Face (four)	4	24	88
27	FacesUCR	14	200	2050
28	Fish	7	175	175
29	FordA	2	1320	3601
30	FordB	2	810	3636
31	Gun-Point	2	50	150
32	Ham	2	109	105
33	HandOutlines	2	370	1000
34	Haptics	5	155	308
35	Herring	2	64	64
36	InlineSkate	7	100	550
37	InsectWingbeatSound	11	220	1980
38	ItalyPowerDemand	2	67	1029
39	LargeKitchenAppliances	3	375	375
40	Lightning-2	2	60	61
41	Lightning-7	7	70	73
42	MALLAT	8	55	2345
43	Meat	3	60	60
44	MedicalImages	10	381	760
45	MiddlePhalanxOutlineAgeGroup	3	154	400
...	...	...	...	...
Continue to the next page				

N	Name	Nb. of classes	Size of training set	Size of testing set
Following ...				

Table B.1: 85 UCR datasets used for experimental validation. The full list is available here [[Chen et al., 2015](#)]

End

### B.3.2 Comparison of algorithm performance

The table [B.2](#) presents the comparison of the classification error of 1-Nearest Neighbor (1-NN) algorithms associated with Euclidean distance (4), 1-NN, associated with the dynamic time warping algorithm using a constraint (a deformation window) (5), 1-NN associated with the algorithm of unconstrained dynamic time warping (DTW) applied to the raw data (6) and the 1-NN algorithm associated with DTW applied to the data pre-processed by our algorithm (7). The (4) algorithm gives the best results that are to say ((4)  $\leq$  (5) and (4)  $\leq$  (6) and (4)  $\leq$  (7)) on 20 datasets, the algorithm (5) is the best on 47 datasets, the (6) algorithm is the best on 21 datasets, the (7) algorithm is the best on 21 datasets. Although no of these algorithms have the best performance on all datasets, the algorithm (5) averaged the smallest misclassification **0,237** and the most expensive (4) algorithm large average error **0,288**. The (6) and (7) algorithms have relatively close average error rates equal to **0,256** and **0,258** respectively.

To evaluate the effects of the **change of representation** of the time series on their **classification**, we compare the length of the time series and the errors of classification presented by the columns (6) and (7) of the table [B.2](#). Indeed, these two columns use the same 1-NN classification algorithm and the same function distance DTW. The only difference between these columns is the nature of the data; the (6) column uses the raw data and the column (7) the compacted data with the method described above.

- Regarding the length of the time series; the (6) algorithm uses all the points of the time series. On the other hand, the (7) algorithm uses compact representations whose length varies between **15 %** and **34 %** of the initial length of the time series. On average, the compact representations have a length equal to **20 %** of the initial time series
- For classification errors, the error (7)  $>$  (6) on 40 datasets, the error of (7) = (6) on 3 datasets and the error of (7)  $<$  (6) on 42 datasets.

These results are encouraging because despite the reduction in the length of the time series errors, the classification error with the compact representation is less than or equal to that of the raw data classification for 45 datasets out of the 85

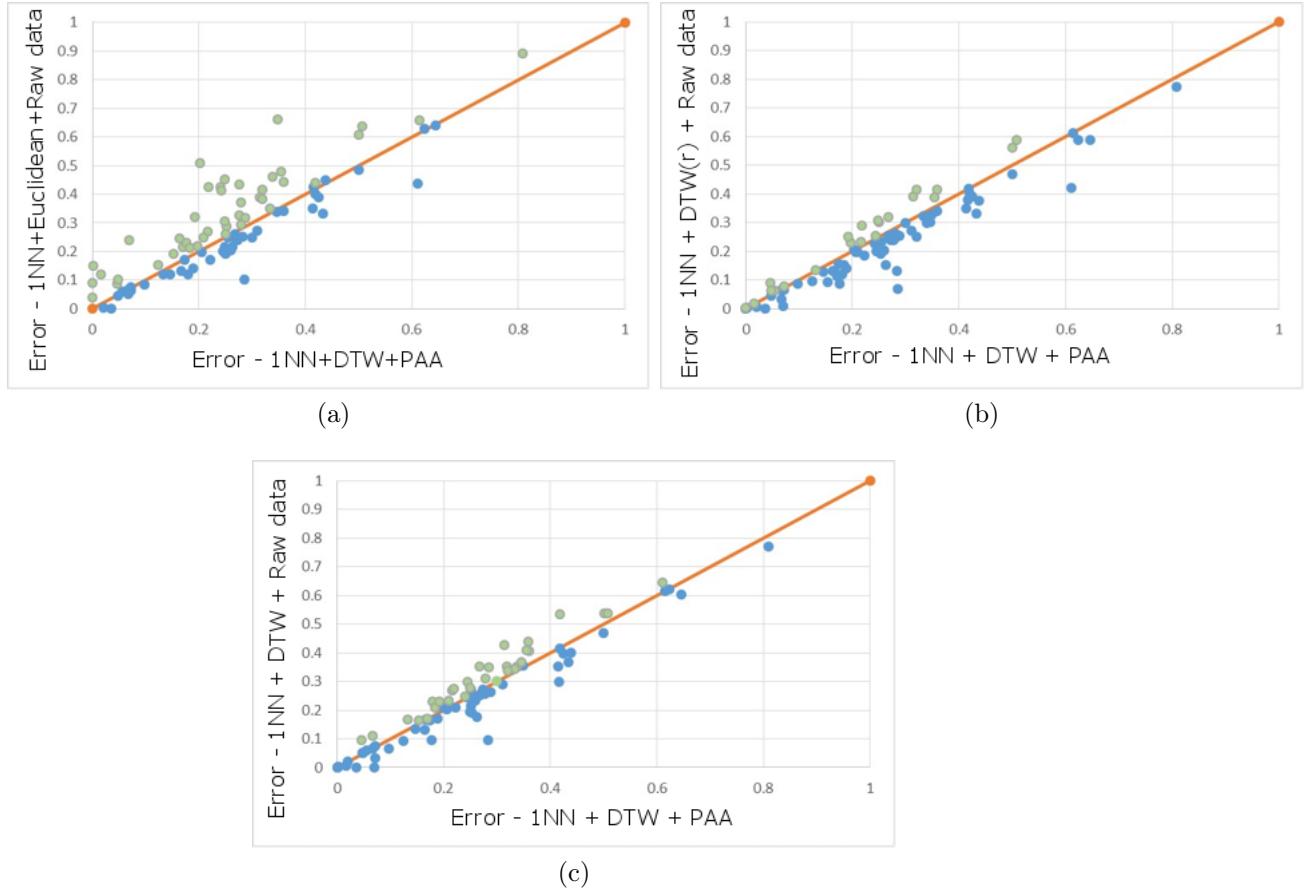


Figure B.3: Two-to-two comparison of the classification errors of the algorithm 1-Nearest Neighbor (1-NN) using Euclidean distance with 1-NN using two variations of the temporal warping algorithm on data raw and compact data

available in the UCR base. These results are summarized in Figure B.3. One of the reasons for this observed improvement over 42 datasets is as follows: the dynamic time warping algorithm is sensitive to noise, therefore by aggregating the points of the segments, we reduce the effects of noise.

(1)	(2)	(3)	(4)	(5)	(6)	(7)
1	54	0,20	0,369	0,242 (6)	0,31	<b>0,279</b>
2	35	0,20	<i>0,389</i>	0,391 (3)	<b>0,396</b>	0,425
3	50	0,20	<i>0,2</i>	<i>0,200 (0)</i>	0,297	<b>0,246</b>
4	78	0,17	<i>0,333</i>	<i>0,333 (0)</i>	<b>0,367</b>	0,433
5	85	0,17	<i>0,25</i>	0,300 (7)	0,3	<b>0,300</b>
6	85	0,17	0,45	0,300(6)	<i>0,25</i>	<b>0,250</b>
7	96	0,17	0,267	0,233 (1)	0,267	<b>0,217</b>

Continue to the next page

(1)	(2)	(3)	(4)	(5)	(6)	(7)
Following ...						
8	32	0,25	0,148	0,004 (11)	0,003	<b>0,002</b>
9	33	0,20	0,35	0,35 (0)	<b>0,352</b>	0,414
10	234	0,14	0,103	0,07 (1)	0,349	<b>0,285</b>
11	57	0,20	0	0,000 (0)	<b>0</b>	0,036
12	120	0,17	0,424	0,380 (13)	<b>0,3</b>	0,416
13	60	0,20	0,423	0,228 (10)	0,246	<b>0,241</b>
14	60	0,20	0,433	0,238 (17)	<b>0,256</b>	0,277
15	60	0,20	0,413	0,254 (5)	0,246	<b>0,244</b>
16	69	0,20	0,065	0,065 (0)	<b>0,033</b>	0,072
17	20	0,25	0,218	0,228 (1)	0,208	<b>0,198</b>
18	20	0,25	0,248	0,232 (2)	<b>0,232</b>	0,255
19	20	0,25	0,273	0,272 (0)	<b>0,29</b>	0,310
20	85	0,17	0,326	0,258 (22)	<b>0,258</b>	0,276
21	24	0,25	0,12	0,120 (0)	0,23	<b>0,180</b>
22	35	0,25	0,075	0,075 (1)	0,076	<b>0,072</b>
23	34	0,25	0,203	0,203 (0)	<b>0,232</b>	0,259
24	24	0,25	0,45	0,376 (14)	<b>0,399</b>	0,438
25	32	0,24	0,286	0,192 (3)	<b>0,192</b>	0,253
26	70	0,20	0,216	0,114 (2)	0,17	<b>0,170</b>
27	32	0,24	0,231	0,088 (12)	<b>0,095</b>	0,177
28	77	0,17	0,217	0,154(4)	<b>0,177</b>	0,263
29	83	0,17	0,341	0,341 (0)	0,438	<b>0,359</b>
30	83	0,17	0,442	0,414 (1)	0,406	<b>0,360</b>
31	30	0,20	0,087	0,087 (0)	0,093	<b>0,047</b>
32	71	0,16	0,4	0,400 (0)	0,533	<b>0,419</b>
33	387	0,14	0,199	0,197 (1)	<b>0,202</b>	0,206
34	182	0,17	0,63	0,588 (2)	0,623	<b>0,623</b>
35	85	0,17	0,484	0,469 (5)	<b>0,469</b>	0,500
36	268	0,14	0,658	0,613 (14)	0,616	<b>0,615</b>
37	51	0,20	0,438	0,422 (2)	0,645	<b>0,611</b>
38	8	0,33	0,045	0,045 (0)	0,05	<b>0,048</b>
39	120	0,17	0,507	0,205 (94)	0,205	<b>0,203</b>
40	106	0,17	0,246	0,131 (6)	<b>0,131</b>	0,164
41	63	0,20	0,425	0,288 (5)	0,274	<b>0,219</b>
42	170	0,17	0,086	0,086 (0)	<b>0,066</b>	0,097
43	74	0,17	0,067	0,067 (0)	<b>0,067</b>	0,268
44	24	0,24	0,316	0,253 (20)	<b>0,263</b>	0,288
45	20	0,25	0,26	0,253 (5)	<b>0,25</b>	0,268

Continue to the next page

(1)	(2)	(3)	(4)	(5)	(6)	(7)
Following ...						
46	20	0,25	<i>0,247</i>	0,318 (1)	0,352	<b>0,268</b>
47	20	0,25	0,439	0,419 (2)	<b>0,416</b>	0,419
48	21	0,25	<i>0,121</i>	0,134 (1)	0,165	<b>0,133</b>
49	125	0,17	<i>0,171</i>	0,185 (1)	<b>0,209</b>	0,222
50	125	0,17	<i>0,12</i>	0,129 (1)	<b>0,135</b>	0,146
51	95	0,17	<i>0,133</i>	<i>0,133 (0)</i>	0,167	<b>0,167</b>
52	71	0,17	0,479	0,388 (7)	0,409	<b>0,355</b>
53	20	0,25	<i>0,239</i>	<i>0,239 (0)</i>	<b>0,272</b>	0,273
54	170	0,17	0,891	0,773 (14)	<b>0,772</b>	0,809
55	36	0,25	0,038	<i>0,000 (6)</i>	0	<b>0,000</b>
56	20	0,25	0,215	0,215 (0)	<b>0,195</b>	0,249
57	20	0,25	<i>0,192</i>	0,210 (1)	<b>0,216</b>	0,251
58	20	0,25	0,292	<i>0,263 (6)</i>	<b>0,263</b>	0,280
59	120	0,17	0,605	0,560 (8)	0,536	<b>0,501</b>
60	120	0,17	0,64	<i>0,589 (17)</i>	<b>0,603</b>	0,645
61	83	0,17	0,461	<i>0,300 (3)</i>	0,35	<b>0,339</b>
62	85	0,17	0,248	0,198 (4)	0,232	<b>0,210</b>
63	120	0,17	0,659	<i>0,328 (15)</i>	0,357	<b>0,349</b>
64	17	0,24	0,305	0,305 (0)	0,275	<b>0,250</b>
65	16	0,25	<i>0,141</i>	<i>0,141 (0)</i>	<b>0,169</b>	0,189
66	170	0,17	0,151	0,095 (16)	<b>0,093</b>	0,124
67	47	0,20	0,062	0,062 (0)	0,06	<b>0,055</b>
68	32	0,25	0,211	<i>0,154 (2)</i>	0,208	<b>0,184</b>
69	79	0,20	0,1	0,062 (8)	0,05	<b>0,048</b>
70	15	0,25	0,12	0,017 (6)	<b>0,007</b>	0,017
71	55	0,20	0,32	0,250 (8)	0,228	<b>0,193</b>
72	68	0,20	0,192	<i>0,092 (5)</i>	0,162	<b>0,154</b>
73	55	0,20	0,24	0,010 (3)	0	0,070
74	32	0,25	0,09	0,002 (4)	0	<b>0,000</b>
75	20	0,24	0,253	0,132 (5)	<b>0,096</b>	0,283
76	63	0,20	0,261	0,227 (4)	0,273	<b>0,252</b>
77	63	0,20	0,338	<i>0,301 (4)</i>	0,366	<b>0,346</b>
78	63	0,20	0,35	<i>0,322 (6)</i>	0,342	<b>0,334</b>
79	157	0,17	0,052	<i>0,034 (4)</i>	0,108	<b>0,067</b>
80	30	0,20	0,005	0,005 (1)	<b>0,02</b>	0,021
81	46	0,20	0,389	0,389 (0)	0,426	<b>0,315</b>
82	54	0,20	0,382	<i>0,252 (8)</i>	0,351	<b>0,320</b>
83	150	0,17	0,635	0,586 (3)	0,536	<b>0,508</b>
84	150	0,17	0,414	0,414 (9)	0,337	<b>0,320</b>

Continue to the next page

(1)	(2)	(3)	(4)	(5)	(6)	(7)
Following ...						
85	71	0,17	0,17	<i>0,155 (2)</i>	<b>0,164</b>	0,174
$X$				0,288	0,237	0,256
$\sigma$				0,175	0,161	0,166
						0,160

Table B.2: The (1) column presents **numbers** of the datasets. The column (2) the **reduced length** of the time series. The column (3) is the **ratio** of the length of the reduced time series over the length of the initial time series. The (4) column designates the **1-Nearest Neighbor** algorithm, associated to the **Euclidean distance**. The (5) column designates the algorithm of **1- Nearer Neighbor**, associated with the algorithm of **dynamic dynamic temporal deformation** using a **constraint** called deformation window that allows to stop the comparison of time series when one perceives that they are very different. The (6) column represents **1-Nearest Neighbor** algorithm associated to the **unconstrained dynamic time warping** applied to the **raw data**. The (7) column represents the **algorithm**. **1-Nearest Neighbor** associated with the **dynamic time warping algorithm without constraints**, applied on the **compact representations** produced by our algorithm. We compare firstly, the classification error of the algorithms (6) and (7) the smallest error is in **bold**. We then compare the classification errors of algorithms (4), (5), (6) and (7) the smallest error is put **italicized**.

End

## B.4 Conclusion

The purpose of this article was to propose an algorithm for choosing the number of segments to consider for the compact representation of a time series. For this, we have defined a minimum bound for the number of segments to be considered which is equal to 5 % of the length of the time series. We have proposed an algorithm that chooses the number  $W$  of segments minimizing the mean squared error. Results of experiments conducted on 42 datasets have shown that the number of segments chosen allows two improvements

- significantly reduce the length of the series temporal; time series of reduced size has a length which varies between 15% and 34% of the initial time series length
- improves supervised classification results on a set of 85 datasets used in the literature.

As a perspective for this work, we plan to vary the number of segments  $W$  from 2 to  $\frac{n}{2}$  to see if our value of  $W$  is optimum for a task classification. We also plan to compare the results of this compact representation to those of other representations of literature. We also plan to parallelize our algorithm to calculate the right

number of segments in linear time (almost trivial). This work allows reducing the storage space and the processing time of the time series. It also allows choosing the number of segments to consider when designing representations symbolic of time series. Indeed, several symbolic representations of series of the literature (SAX [Lin et al., 2003], ESAX [Lkhagva et al., 2006], 1d-SAX [Malinowski et al., 2013], SAX-TD [Sun et al., 2014], SAX-P [Siyou Fotso et al., 2015]) use the division into segments recommended by PAA.

# Appendix C

## Determination of the probability density function of Mz uncertainty

### C.1 Ampiric probability distribution

Table C.1: descriptive

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-24.15	-0.4735	0	0.2416	0.78	25.4

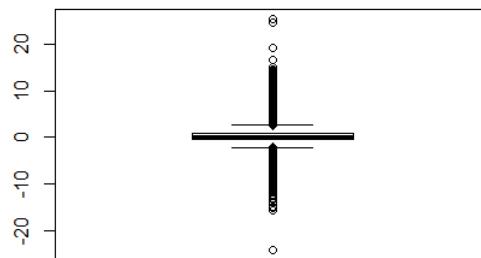


Figure C.1: Distribution of data around the mean

**descriptive statistics** 35.65% of values of uncertainty are equal to zero

**Distribution** The empirical probability density function does not allow us a priori to know which probability law follows the residuals (uncertainty). To determine this,

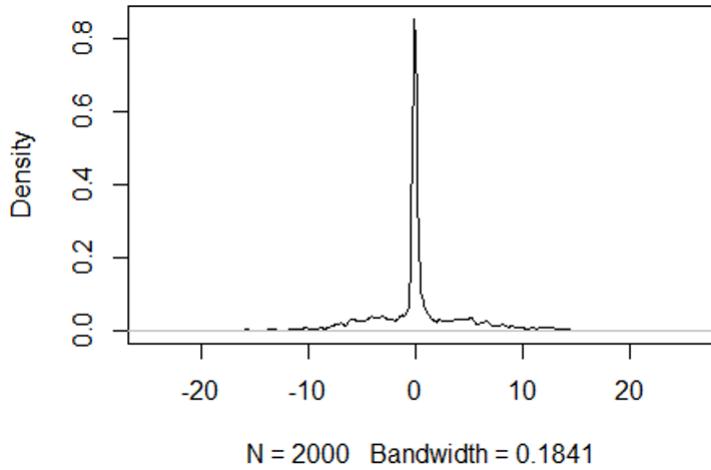


Figure C.2: Estimated probability density of data

we will use statistical tests. The principle of statistical tests is as follow: we assume that the residues follow a defined probability law. And we calculate a p-value. If it is less than 0.05, the above hypothesis is rejected. Another method to determine the probability law follows by the residuals is to hypothesize that data followed a particular law and estimated the parameters that best fit the law. If the theoretical probability density function estimated is closed to the empirical one, we assume that we find the probability law follows by residuals.

## C.2 Theoric probability distribution

Residuals are real values between minus infinity and plus infinity. The probability laws that residuals can follow are therefore

- A normal law
- A law of cauchy
- A Student's Law t
- A general exponential law
- A double exponential law
- Generalized error distribution (GED)
- A pearson law type 4
- A generalization of the student's law.

**Normality test** : For each tests in (table C.2), the p-value is significant, so the sample does not follow a normal distribution.

Table C.2: normality test show that residues(uncertainty) do not follow normal law

Test	Kolmogorov-Smirnov	Anderson-Darling	Shapiro-Wilk
p-value	<2.2e-16	<2.2e-16	<2.2e-16

**Cauchy's Law Test** : The qqplot line does not match correctly with the data particularly the two extremities of the diagram (Fig. C.3); we can say that the data probably do not follow a Cauchy law.

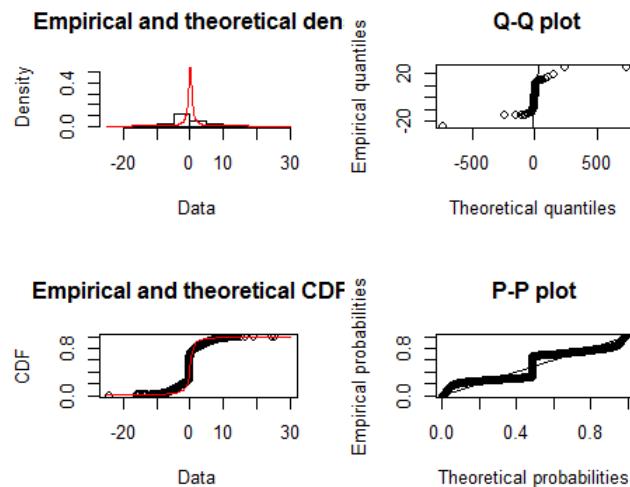


Figure C.3: Estimated probability density of data for cauchy's law

**Student's Law test** : The Quantile-Quantile diagram, from the distribution function, doesn't match correctly the data, particularly the two extremities of the diagram (Fig. C.4). We can then say that the data do not follow a Student law.

**Logistics law** The probability density function matches the data reasonably well (Fig. C.5), so we can say that the **residuals (uncertainty) following a zero-inflated logistic law**.

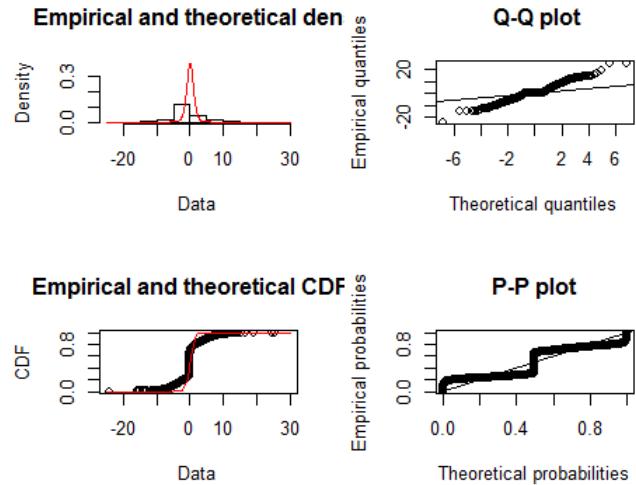


Figure C.4: Estimated probability density of data for Student's law

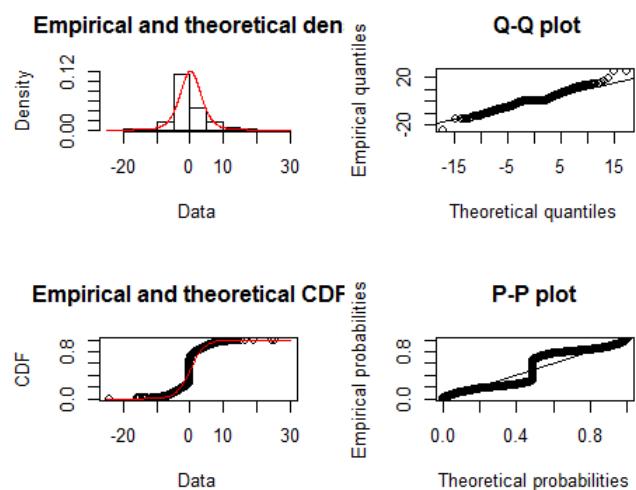


Figure C.5: Estimated probability density of data for Logistic law

# Appendix D

## Training and changes on propulsion techniques

In this section, we want to evaluate the effect of the experiment on the propulsion technique used by manual wheelchair users. We assume that the shape of the Z moment of the left and right wheels reflects the propulsion technique used. Thus, in this paragraph, rather than using an approach that uses cycle properties, we will directly compare time series from a distance function while taking into account the presence of uncertainty in measurements. To do this, we applied FOTS-UShapelet to the measurements made and we deliver here the analysis we make of the results obtained.

FOTS-UShapelet was applied to the time series from the measurement of efforts made by 08 valid subjects during the first week of use and after three weeks of use of the Manual Wheelchair. The table [D.1](#) details the results we get. From the shape of the time series, we can form three large groups that correspond to three techniques that we have named T1, T2 and T3. For each subject of the experiment, we observe a difference between the propulsion techniques used at the beginning of the trial period and after three weeks. The subject SA01 for example uses the T1 technique 83% and the T3 technique 17% of the time during the first week. However, after three weeks of using the Manual Wheelchair, he uses the T1 technique 33% of the time and the T3 technique 67% of the time. So there is a change in the way he moves. However, the changes observed vary according to the subjects ([Table D.1](#)). The variations in the evolution of propulsion techniques require personalized monitoring of locomotion and highlights the importance of using a Wheelchair Ergometer when learning locomotion in a Manual Wheelchair.

Subject	Week	Trial	wheel	technic	T1	T2	T3
SA01NE	T0S	ES1	RD	1			
SA01NE	T0S	ES1	RG	1			
Continue to the next page							

<b>Subject</b>	<b>Week</b>	<b>Trial</b>	<b>wheel</b>	<b>technic</b>	<b>T1</b>	<b>T2</b>	<b>T3</b>
Following ...							
SA01NE	T0S	ES2	RD	1			
SA01NE	T0S	ES2	RG	3			
SA01NE	T0S	ES3	RD	1			
SA01NE	T0S	ES3	RG	1	0,83	0,00	0,17
SA01NE	T3S	ES1	RD	1			
SA01NE	T3S	ES1	RG	3			
SA01NE	T3S	ES2	RD	3			
SA01NE	T3S	ES2	RG	1			
SA01NE	T3S	ES4	RD	3			
SA01NE	T3S	ES4	RG	3	0,33	0,00	0,67
SA02GG	T0S	ES1	RD	3			
SA02GG	T0S	ES1	RG	1			
SA02GG	T0S	ES2	RD	3			
SA02GG	T0S	ES2	RG	3	0,25	0,00	0,75
SA02GG	T3S	ES1	RD	1			
SA02GG	T3S	ES1	RG	1			
SA02GG	T3S	ES2	RD	3			
SA02GG	T3S	ES2	RG	1			
SA02GG	T3S	ES3	RD	3			
SA02GG	T3S	ES3	RG	3	0,50	0,00	0,50
SA03JM	T0S	ES1	RD	1			
SA03JM	T0S	ES1	RG	1			
SA03JM	T0S	ES2	RD	3			
SA03JM	T0S	ES2	RG	1			
SA03JM	T0S	ES3	RD	2			
SA03JM	T0S	ES3	RG	1	0,67	0,17	0,17
SA03JM	T3S	ES1	RD	1			
SA03JM	T3S	ES1	RG	3			
SA03JM	T3S	ES2	RD	1			
SA03JM	T3S	ES2	RG	1			
SA03JM	T3S	ES3	RD	1			

Continue to the next page

<b>Subject</b>	<b>Week</b>	<b>Trial</b>	<b>wheel</b>	<b>technic</b>	<b>T1</b>	<b>T2</b>	<b>T3</b>
Following ...							
SA03JM	T3S	ES3	RG	1	0,83	0,00	0,17
SA04BD	T0S	ES1	RD	3			
SA04BD	T0S	ES1	RG	1			
SA04BD	T0S	ES2	RD	3			
SA04BD	T0S	ES2	RG	1			
SA04BD	T0S	ES4	RD	3			
SA04BD	T0S	ES4	RG	3	0,33	0,00	0,67
SA04BD	T3S	ES1	RD	1			
SA04BD	T3S	ES1	RG	3			
SA04BD	T3S	ES2	RD	1			
SA04BD	T3S	ES2	RG	1			
SA04BD	T3S	ES3	RD	1			
SA04BD	T3S	ES3	RG	1	0,83	0,00	0,17
SA05AT	T0S	ES2	RD	3			
SA05AT	T0S	ES2	RG	1			
SA05AT	T0S	ES3	RD	3			
SA05AT	T0S	ES3	RG	3			
SA05AT	T0S	ES4	RD	2			
SA05AT	T0S	ES4	RG	1	0,33	0,17	0,50
SA05AT	T3S	ES1	RD	2			
SA05AT	T3S	ES1	RG	2			
SA05AT	T3S	ES2	RD	1			
SA05AT	T3S	ES2	RG	3			
SA05AT	T3S	ES3	RD	2			
SA05AT	T3S	ES3	RG	1	0,33	0,50	0,17
SA06BM	T0S	ES1	RD	3			
SA06BM	T0S	ES1	RG	3			
SA06BM	T0S	ES2	RD	1			
SA06BM	T0S	ES2	RG	2			
SA06BM	T0S	ES4	RD	1			
Continue to the next page							

Subject	Week	Trial	wheel	technic	T1	T2	T3
Following ...							
SA06BM	T0S	ES4	RG	3			
SA06BM	T0S	ES4	RD	3			
SA06BM	T0S	ES4	RG	2	0,25	0,25	0,50
SA06BM	T3S	ES1	RD	3			
SA06BM	T3S	ES1	RG	1			
SA06BM	T3S	ES2	RD	1			
SA06BM	T3S	ES2	RG	3			
SA06BM	T3S	ES3	RD	3			
SA06BM	T3S	ES3	RG	3	0,33	0,00	0,67
SA07HP	T0S	ES1	RD	1			
SA07HP	T0S	ES1	RG	1			
SA07HP	T0S	ES2	RD	1			
SA07HP	T0S	ES2	RG	1			
SA07HP	T0S	ES3	RD	1			
SA07HP	T0S	ES3	RG	1	1,00	0,00	0,00
SA07HP	T3S	ES1	RD	1			
SA07HP	T3S	ES1	RG	3			
SA07HP	T3S	ES2	RD	1			
SA07HP	T3S	ES2	RG	1			
SA07HP	T3S	ES3	RD	1			
SA07HP	T3S	ES3	RG	1	0,83	0,00	0,17
SA08CA	T0S	ES1	RD	1			
SA08CA	T0S	ES1	RG	1			
SA08CA	T0S	ES2	RD	1			
SA08CA	T0S	ES2	RG	1			
SA08CA	T0S	ES3	RD	1			
SA08CA	T0S	ES3	RG	1	1,00	0,00	0,00
SA08CA	T3S	ES1	RD	1			
SA08CA	T3S	ES1	RG	1			
SA08CA	T3S	ES2	RD	3			
SA08CA	T3S	ES2	RG	1			

Continue to the next page

<b>Subject</b>	<b>Week</b>	<b>Trial</b>	<b>wheel</b>	<b>technic</b>	<b>T1</b>	<b>T2</b>	<b>T3</b>
Following ...							
SA08CA	T3S	ES3	RD	1			
SA08CA	T3S	ES3	RG	3	0,67	0,00	0,33

Table D.1: Evolution of manual wheelchair propulsion technology with training

End



# Bibliography

- [Ils, 2016] (2016). LSST time series catalog - XLDB Use Cases.
- [SAC, 2016] (2016). Projet SACR-FRM (Approches de la Sociologie, de la Biomécanique et de l'Intelligence Artificielle...) | ANR - Agence Nationale de la Recherche.
- [0Åstrand and Saltin, 1961] 0Åstrand, P.-O. and Saltin, B. (1961). Maximal oxygen uptake and heart rate in various types of muscular activity. *Journal of Applied Physiology*, 16(6):977–981.
- [Aach and Church, 2001] Aach, J. and Church, G. M. (2001). Aligning gene expression time series with time warping algorithms. *Bioinformatics*, 17(6):495–508.
- [Abonyi et al., 2003] Abonyi, J., Feil, B., Nemeth, S., and Arva, P. (2003). Fuzzy clustering based segmentation of time-series. *Advances in Intelligent Data Analysis V*, pages 275–285.
- [Agrawal et al., 1993] Agrawal, R., Faloutsos, C., and Swami, A. N. (1993). Efficient Similarity Search In Sequence Databases. In *FODO '93 Proceedings of the 4th International Conference on Foundations of Data Organization and Algorithms*, pages 69–84. Springer-Verlag.
- [Amini et al., 2011] Amini, A., Wah, T. Y., Saybani, M. R., and Yazdi, S. R. A. S. (2011). A study of density-grid based clustering algorithms on data streams. In *Fuzzy Systems and Knowledge Discovery (FSKD), 2011 Eighth International Conference on*, volume 3, pages 1652–1656. IEEE.
- [Aßfalg et al., 2009] Aßfalg, J., Kriegel, H.-P., Kröger, P., and Renz, M. (2009). Probabilistic similarity search for uncertain time series. In *SSDBM*, pages 435–443. Springer.
- [Athanasios and Clark, 2009] Athanasios, M. and Clark, J. Y. (2009). A bayesian network model for the diagnosis of the caring procedure for wheelchair users with spinal injury. *Computer methods and programs in biomedicine*, 95(2):S44–S54.

- [Bagnall et al., 2016a] Bagnall, A., Keogh, E., Lines, J., Bostrom, A., and Large, J. (2016a). Time Series Classification Website. <http://timeseriesclassification.com>.
- [Bagnall et al., 2016b] Bagnall, A., Lines, J., Bostrom, A., Large, J., and Keogh, E. (2016b). The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. pages 1–55.
- [Batista et al., 2014] Batista, G. E., Keogh, E. J., Tataw, O. M., and De Souza, V. M. (2014). Cid: an efficient complexity-invariant distance for time series. 28(3):634–669.
- [Batista et al., 2011] Batista, G. E., Wang, X., and Keogh, E. J. (2011). A complexity-invariant distance measure for time series. In *Proceedings of the 2011 SIAM international conference on data mining*, pages 699–710. SIAM.
- [Baydogan and Runger, 2016] Baydogan, M. G. and Runger, G. (2016). Time series representation and similarity based on local autopatterns. *Data Mining and Knowledge Discovery*, 30(2):476–509.
- [Baydogan et al., 2013] Baydogan, M. G., Runger, G., and Tuv, E. (2013). A bag-of-features framework to classify time series. *IEEE transactions on pattern analysis and machine intelligence*, 35(11):2796–2802.
- [Begum and Keogh, 2014] Begum, N. and Keogh, E. (2014). Rare time series motif discovery from unbounded streams. *Proceedings of the VLDB Endowment*, 8(2):149–160.
- [Bergh et al., 1976] Bergh, U., Kanstrup, I., and Ekblom, B. (1976). Maximal oxygen uptake during exercise with various combinations of arm and leg work. *Journal of Applied Physiology*, 41(2):191–196.
- [Bhatia and Bhattacharyya, 1993] Bhatia, R. and Bhattacharyya, T. (1993). A generalization of the Hoffman-Wielandt theorem. 179:11–17.
- [Brattgård et al., 1970] Brattgård, S., Grimby, G., and Höök, O. (1970). Energy expenditure and heart rate in driving a wheel-chair ergometer. *Scandinavian journal of rehabilitation medicine*, 2(4):143.
- [Brouha and Krobath, 1967] Brouha, L. and Krobath, H. (1967). Continuous recording of cardiac and respiratory functions in normal and handicapped people. *Human factors*, 9(6):567–571.
- [Buttkus, 2012] Buttkus, B. (2012). *Spectral analysis and filter theory in applied geophysics*. Springer Science & Business Media.

- [Camerra et al., 2010] Camerra, A., Palpanas, T., Shieh, J., and Keogh, E. (2010). isax 2.0: Indexing and mining one billion time series. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on*, pages 58–67. IEEE.
- [Candan et al., 2012] Candan, K. S., Rossini, R., Wang, X., and Sapino, M. L. (2012). sdtw: computing dtw distances using locally relevant constraints based on salient feature alignments. 5(11):1519–1530.
- [Chan and Fu, 1999a] Chan, K.-P. and Fu, A. W.-C. (1999a). Efficient time series matching by wavelets. In *Data Engineering, 1999. Proceedings., 15th International Conference on*, pages 126–133. IEEE.
- [Chan and Fu, 1999b] Chan, K.-P. and Fu, A. W.-C. (1999b). Efficient time series matching by wavelets. In *Proceedings 15th International Conference on Data Engineering (Cat. No.99CB36337)*, pages 126–133. IEEE.
- [Chandola et al., 2009] Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys*, 41(3):15.
- [Chen et al., 2005] Chen, L., Özsü, M. T., and Oria, V. (2005). Robust and fast similarity search for moving object trajectories. In *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 491–502. ACM.
- [Chen et al., 2015] Chen, Y., Keogh, E., Hu, B., Begum, N., Bagnall, A., Mueen, A., and Batista, G. (2015). The ucr time series classification archive. [http://www.cs.ucr.edu/~eamonn/time\\_series\\_data/](http://www.cs.ucr.edu/~eamonn/time_series_data/). [www.cs.ucr.edu/~eamonn/time\\_series\\_data/](http://www.cs.ucr.edu/~eamonn/time_series_data/).
- [Chu et al., 2002] Chu, S., Keogh, E. J., Hart, D. M., Pazzani, M. J., et al. (2002). Iterative deepening dynamic time warping for time series. In *SDM*, pages 195–212. SIAM.
- [Claremont and Maksud, 1985] Claremont, A. and Maksud, M. (1985). A model treadmill adaptation for wheelchair ergometry. *Canadian journal of applied sport sciences. Journal canadien des sciences appliquées au sport*, 10(4):178–181.
- [Cooper, 1990] Cooper, R. A. (1990). An exploratory study of racing wheelchair propulsion dynamics. *Adapted Physical Activity Quarterly*, 7(1):74–85.
- [Cooper et al., 2003] Cooper, R. A., Boninger, M. L., Cooper, R., Robertson, R., and Baldini, F. (2003). Wheelchair racing efficiency. *Disability and rehabilitation*, 25(4-5):207–212.
- [Couétard, 2000] Couétard, Y. (2000). *Caractérisation et étalonnage des dynamomètres à six composantes pour torseur associé à un système de forces*. PhD thesis, Université Sciences et Technologies-Bordeaux I.

- [Coutts, 1990] Coutts, K. D. (1990). Kinematics of sport wheelchair propulsion. *Journal of rehabilitation research and development*, 27(1):21.
- [Coutts and Stogryn, 1987] Coutts, K. D. and Stogryn, J. L. (1987). Aerobic and anaerobic power of canadian wheelchair track athletes. *Medicine and science in sports and exercise*, 19(1):62–65.
- [Cuřín et al., 2007] Cuřín, J., Fleury, P., Kleindienst, J., and Kessl, R. (2007). Meeting state recognition from visual and aural labels. In *International Workshop on Machine Learning for Multimodal Interaction*, pages 24–25. Springer.
- [Dabonneville et al., 2005] Dabonneville, M., Kauffmann, P., Vaslin, P., De Saint Remy, N., Couétard, Y., and Cid, M. (2005). A self-contained wireless wheelchair ergometer designed for biomechanical measures in real life conditions. *Technology and Disability*, 17(2):63–76.
- [Dallachiesa et al., 2012] Dallachiesa, M., Nushi, B., Mirylenka, K., and Palpanas, T. (2012). Uncertain time-series similarity: return to the basics. 5(11):1662–1673.
- [Das et al., 1997] Das, G., Gunopulos, D., and Mannila, H. (1997). Finding similar time series. In *European Symposium on Principles of Data Mining and Knowledge Discovery*, pages 88–100. Springer.
- [De Lathauwer et al., 1994] De Lathauwer, L., De Moor, B., Vandewalle, J., and by Higher-Order, B. S. S. (1994). Singular value decomposition. In *Proc. EUSIPCO-94, Edinburgh, Scotland, UK*, volume 1, pages 175–178.
- [De Saint Remy, 2005] De Saint Remy, N. (2005). Modélisation et détermination des paramètres biomécaniques de la locomotion en fauteuil roulant manuel.
- [Deng et al., 2013] Deng, H., Runger, G., Tuv, E., and Vladimir, M. (2013). A time series forest for classification and feature extraction. *Information Sciences*, 239:142–153.
- [Devillard, 1999] Devillard, X. (1999). A wheelchair ergometer for physiological and biomechanical measurement. In *Proceedings of the 11th*.
- [Devillard et al., 2001] Devillard, X., Calmels, P., Sauvignet, B., Belli, A., Denis, C., Simard, C., and Gautheron, V. (2001). Validation of a new ergometer adapted to all types of manual wheelchair. *European journal of applied physiology*, 85(5):479–485.
- [Dhillon et al., 2003] Dhillon, I. S., Mallela, S., and Modha, D. S. (2003). Information-theoretic co-clustering. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 89–98. ACM.

- [Dietterich, 2002] Dietterich, T. G. (2002). Machine Learning for Sequential Data: A Review. *Proceedings of the Joint IAPR International Workshop on Structural, Syntactic, and Statistical Pattern Recognition*, pages 16–30.
- [DiGiovine et al., 2001] DiGiovine, C. P., Cooper, R. A., and Boninger, M. L. (2001). Dynamic calibration of a wheelchair dynamometer. *Journal of rehabilitation research and development*, 38(1):41.
- [Ding, 2011] Ding, H. (2011). *Key concepts for implementing SoC-Holter*. PhD thesis, Université Blaise Pascal-Clermont-Ferrand II.
- [Ding et al., 2008] Ding, H., Trajcevski, G., Scheuermann, P., Wang, X., and Keogh, E. (2008). Querying and mining of time series data. *Proceedings of the VLDB Endowment*, 1(2):1542–1552.
- [Eriksson et al., 2004] Eriksson, K., Estep, D., and Johnson, C. (2004). Piecewise linear approximation. In *Applied Mathematics: Body and Soul*, pages 741–753. Springer.
- [Esling and Agon, 2012] Esling, P. and Agon, C. (2012). Time-series data mining. *ACM Computing Surveys (CSUR)*, 45(1):1–34.
- [Faria et al., 2012] Faria, B. M., Reis, L. P., Lau, N., Soares, J. C., and Vasconcelos, S. (2012). Patient classification and automatic configuration of an intelligent wheelchair. In *International Conference on Agents and Artificial Intelligence*, pages 268–282. Springer.
- [Feo and Resende, 1995] Feo, T. A. and Resende, M. G. (1995). Greedy randomized adaptive search procedures. *Journal of global optimization*, 6(2):109–133.
- [Filippone et al., 2008] Filippone, M., Camastra, F., Masulli, F., and Rovetta, S. (2008). A survey of kernel and spectral methods for clustering. *Pattern recognition*, 41(1):176–190.
- [Folland and Sitaram, 1997] Folland, G. B. and Sitaram, A. (1997). The uncertainty principle: a mathematical survey. *Journal of Fourier analysis and applications*, 3(3):207–238.
- [FOTS-SUSh, 2018] FOTS-SUSh (2018). <https://sites.google.com/view/fots-sush/accueil>.
- [Fu, 2011] Fu, T.-c. (2011). A review on time series data mining. *Engineering Applications of Artificial Intelligence*, 24(1):164–181.
- [Goosey et al., 1998] Goosey, V. L., Campbell, I. G., and Fowler, N. E. (1998). The relationship between three-dimensional wheelchair propulsion techniques and pushing economy. *Journal of applied biomechanics*, 14(4):412–427.

- [Goosey-Tolfrey et al., 2001] Goosey-Tolfrey, V. L., Fowler, N. E., Campbell, I. G., and Iwnicki, S. D. (2001). A kinetic analysis of trained wheelchair racers during two speeds of propulsion. *Medical engineering and physics*, 23(4):259–266.
- [Grabocka et al., 2014] Grabocka, J., Schilling, N., Wistuba, M., and Schmidt-Thieme, L. (2014). Learning time-series shapelets. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 392–401. ACM.
- [Halkidi et al., 2001] Halkidi, M., Batistakis, Y., and Vazirgiannis, M. (2001). On clustering validation techniques. *Journal of intelligent information systems*, 17(2-3):107–145.
- [Hochheiser and Shneiderman, 2003] Hochheiser, H. and Shneiderman, B. (2003). Interactive exploration of time series data. In *The Craft of Information Visualization*, pages 313–315. Elsevier.
- [Hoffmann-Jørgensen and Pisier, 1976] Hoffmann-Jørgensen, J. and Pisier, G. (1976). The law of large numbers and the central limit theorem in banach spaces. *The Annals of Probability*, pages 587–599.
- [Hofmann, 2017] Hofmann, T. (2017). Probabilistic latent semantic indexing. In *ACM SIGIR Forum*, volume 51, pages 211–218. ACM.
- [Hwang et al., 2014] Hwang, J., Kozawa, Y., Amagasa, T., and Kitagawa, H. (2014). GPU Acceleration of Similarity Search for Uncertain Time Series. In *2014 17th International Conference on Network-Based Information Systems*, pages 627–632. IEEE.
- [Ibarra and Kim, 1975] Ibarra, O. H. and Kim, C. E. (1975). Fast approximation algorithms for the knapsack and sum of subset problems. 22(4):463–468.
- [Ibragimov and Has' minskii, 2013] Ibragimov, I. A. and Has' minskii, R. Z. (2013). *Statistical estimation: asymptotic theory*, volume 16. Springer Science & Business Media.
- [Itakura, 1975] Itakura, F. (1975). Minimum prediction residual principle applied to speech recognition. 23(1):67–72.
- [Jeong et al., 2011] Jeong, Y.-S., Jeong, M. K., and Omittaomu, O. A. (2011). Weighted dynamic time warping for time series classification. 44(9):2231–2240.
- [Johnson et al., 2004] Johnson, B., Mushett, C., Richter, K., and Peacock, G. (2004). Sport for athletes with physical disabilities: injuries and medical issues. *BlazeSports America*.

- [Kalpakis et al., 2001] Kalpakis, K., Gada, D., and Puttagunta, V. (2001). Distance measures for effective clustering of arima time-series. In *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, pages 273–280. IEEE.
- [Kate, 2016a] Kate, R. J. (2016a). Using dynamic time warping distances as features for improved time series classification. *Data Mining and Knowledge Discovery*, 30(2):283–312.
- [Kate, 2016b] Kate, R. J. (2016b). Using dynamic time warping distances as features for improved time series classification. 30(2):283–312.
- [Kendall, 1961] Kendall, M. G. (1961). *The advanced theory of statistics: Inference and relationship*, volume 2. Charles Griffin.
- [Keogh et al., 2001a] Keogh, E., Chakrabarti, K., Pazzani, M., and Mehrotra, S. (2001a). Dimensionality Reduction for Fast Similarity Search in Large Time Series Databases. *Knowledge and Information Systems*, 3(January):263–286.
- [Keogh et al., 2001b] Keogh, E., Chakrabarti, K., Pazzani, M., and Mehrotra, S. (2001b). Dimensionality reduction for fast similarity search in large time series databases. *Knowledge and information Systems*, 3(3):263–286.
- [Keogh et al., 2001c] Keogh, E., Chakrabarti, K., Pazzani, M., and Mehrotra, S. (2001c). Locally adaptive dimensionality reduction for indexing large time series databases. 30(2):151–162.
- [Keogh and Kasetty, 2003a] Keogh, E. and Kasetty, S. (2003a). On the need for time series data mining benchmarks: a survey and empirical demonstration. *Data Mining and knowledge discovery*, 7(4):349–371.
- [Keogh and Kasetty, 2003b] Keogh, E. and Kasetty, S. (2003b). On the Need for Time Series Data Mining Benchmarks: A Survey and Empirical Demonstration. *Data Mining and Knowledge Discovery*, 7(4):349–371.
- [Keogh and Ratanamahatana, ] Keogh, E. and Ratanamahatana, C. A. Exact indexing of dynamic time warping. 7(3):358–386.
- [Keogh and Pazzani, 2000] Keogh, E. J. and Pazzani, M. J. (2000). Scaling up dynamic time warping for datamining applications. In *sixth ACM SIGKDD*, pages 285–289. ACM.
- [Keogh and Pazzani, 2001a] Keogh, E. J. and Pazzani, M. J. (2001a). Derivative dynamic time warping. In *Proceedings of the 2001 SIAM International Conference on Data Mining*, pages 1–11. SIAM.

- [Keogh and Pazzani, 2001b] Keogh, E. J. and Pazzani, M. J. (2001b). Derivative dynamic time warping. *1st SIAM International Conference on Data Mining*, pages 1–11.
- [Kerk et al., 1995] Kerk, J. K., Clifford, P. S., Snyder, A. C., Prieto, T. E., O'hagan, K. P., Schot, P. K., Myklebust, J. B., and Myklebust, B. M. (1995). Effect of an abdominal binder during wheelchair exercise. *Medicine and science in sports and exercise*, 27(6):913–919.
- [Kriegel et al., 2011] Kriegel, H.-P., Kröger, P., Sander, J., and Zimek, A. (2011). Density-based clustering. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(3):231–240.
- [Kulig et al., 2001] Kulig, K., Newsam, C. J., Mulroy, S. J., Rao, S., Gronley, J. K., Bontrager, E. L., and Perry, J. (2001). The effect of level of spinal cord injury on shoulder joint kinetics during manual wheelchair propulsion. *Clinical biomechanics*, 16(9):744–751.
- [Kumar et al., 2002] Kumar, M., Patel, N. R., and Woo, J. (2002). Clustering seasonality patterns in the presence of errors. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 557–563. ACM.
- [Langbein et al., 1993] Langbein, W., Robinson, C., Kynast, L., and Fehr, L. (1993). Calibration of a new wheelchair ergometer: the wheelchair aerobic fitness trainer. *IEEE Transactions on Rehabilitation Engineering*, 1(1):49–58.
- [Langbein and Fehr, 1993] Langbein, W. E. and Fehr, L. (1993). Research device to preproduction prototype: a chronology. *Journal of rehabilitation research and development*, 30(4):436.
- [Langbein et al., 1994] Langbein, W. E., Maki, K. C., Edwards, L. C., Hwang, M. H., et al. (1994). Initial clinical evaluation of a wheelchair ergometer for diagnostic exercise testing: a technical note. *Journal of rehabilitation research and development*, 31(4):317.
- [Lin et al., 2003] Lin, J., Keogh, E., Lonardi, S., and Chiu, B. (2003). A symbolic representation of time series, with implications for streaming algorithms. In *8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, pages 2–11, New York, New York, USA. ACM, ACM Press.
- [Lin et al., 2007] Lin, J., Keogh, E., Wei, L., and Lonardi, S. (2007). Experiencing SAX: a novel symbolic representation of time series. *Data Mining and Knowledge Discovery*, 15(2):107–144.

- [Lin et al., 2012] Lin, J., Khade, R., and Li, Y. (2012). Rotation-invariant similarity in time series using bag-of-patterns representation. *Journal of Intelligent Information Systems*, 39(2):287–315.
- [Lin and Li, 2009] Lin, J. and Li, Y. (2009). Finding structural similarity in time series data using bag-of-patterns representation. In *International Conference on Scientific and Statistical Database Management*, pages 461–477. Springer.
- [Lines et al., 2012] Lines, J., Davis, L. M., Hills, J., and Bagnall, A. (2012). A shapelet transform for time series classification. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 289–297. ACM.
- [Lkhagva and Kawagoe, 2006] Lkhagva, B. and Kawagoe, K. (2006). New Time Series Data Representation ESAX for Financial Applications. In *22nd International Conference on Data Engineering Workshops (ICDEW'06)*, pages x115–x115. IEEE.
- [Lkhagva et al., 2006] Lkhagva, B., Suzuki, Y., and Kawagoe, K. (2006). Extended sax: Extension of symbolic aggregate approximation for financial time series data representation. 7.
- [Longin et al., 2005] Longin, J., Vasilis, M., Qiang, W., Rolf, L., Chotirat, A., and Keogh, E. (2005). Elastic partial matching of time series. In *9th European Conference On Principles And Practice Of Knowledge Discovery In Databases, Porto, Portugal*.
- [Lonlac et al., 2018] Lonlac, J., Negrevergne, B., Miras, Y., Beauger, A., and Nguifo, E. M. (2018). Fouille de motifs graduels fermés fréquents sous contrainte de la temporalité. *GAST–Gestion et Analyse de données Spatiales et Temporelles*.
- [Lundqvist et al., 1991] Lundqvist, C., Siösteen, A., Blomstrand, C., Lind, B., and Sullivan, M. (1991). Spinal cord injuries. clinical, functional, and emotional status. *Spine*, 16(1):78–83.
- [Machida et al., 2013] Machida, M., Irwin, B., and Feltz, D. (2013). Resilience in competitive athletes with spinal cord injury: the role of sport participation. *Qualitative health research*, 23(8):1054–1065.
- [Mahalakshmi et al., 2016] Mahalakshmi, G., Sridevi, S., and Rajaram, S. (2016). A survey on forecasting of time series data. In *Computing Technologies and Intelligent Data Engineering (ICCTIDE), International Conference on*, pages 1–8. IEEE.
- [Malinowski et al., 2013] Malinowski, S., Guyet, T., Quiniou, R., and Tavenard, R. (2013). *1d-SAX: A Novel Symbolic Representation for Time Series*, volume

8207 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Marteau, 2008] Marteau, P.-F. (2008). Time warp edit distance. *arXiv preprint arXiv:0802.3522*.

[Marteau, 2009] Marteau, P.-F. (2009). Time warp edit distance with stiffness adjustment for time series matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):306–318.

[Massee et al., 1992] Massee, L., Lamontagne, M., and O’riain, M. (1992). Biomechanical analysis of wheelchair propulsion for various seating positions. *Journal of rehabilitation research and development*, 29(3):12–28.

[Matthies et al., 1989] Matthies, L., Kanade, T., and Szeliski, R. (1989). Kalman filter-based algorithms for estimating depth from image sequences. *International Journal of Computer Vision*, 3(3):209–238.

[Merugu and Ghosh, 2003] Merugu, S. and Ghosh, J. (2003). Privacy-preserving distributed clustering using generative models. In *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, pages 211–218. IEEE.

[Murthy and Sarangi, 2013] Murthy, K. and Sarangi, S. R. (2013). Generalized notion of similarities between uncertain time series. US Patent 8,407,221.

[Myers et al., 1980a] Myers, C., Rabiner, L., and Rosenberg, A. (1980a). Performance tradeoffs in dynamic time warping algorithms for isolated word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(6):623–635.

[Myers et al., 1980b] Myers, C., Rabiner, L. R., and Rosenberg, A. E. (1980b). Performance tradeoffs in dynamic time warping algorithms for isolated word recognition. 28(6):623–635.

[Needs and The, 2016] Needs, W. and The, I. (2016). Analysis of Wheelchair Need - Wheelchair Foundation.

[Newsam et al., 1996] Newsam, C. J., Mulroy, S. J., Gronley, J. K., Bontrager, E. L., and Perry, J. (1996). Temporal-spatial characteristics of wheelchair propulsion: Effects of level of spinal cord injury, terrain, and propulsion rate1. *American journal of physical medicine & rehabilitation*, 75(4):292–299.

[Orang and Shiri, 2014] Orang, M. and Shiri, N. (2014). An experimental evaluation of similarity measures for uncertain time series. In *Proceedings of the 18th International Database Engineering & Applications Symposium on - IDEAS ’14*, pages 261–264. ACM Press.

- [Orang and Shiri, 2015] Orang, M. and Shiri, N. (2015). Improving performance of similarity measures for uncertain time series using preprocessing techniques. In *Proceedings of the 27th International Conference on Scientific and Statistical Database Management - SSDBM '15*, pages 1–12. ACM Press.
- [Orang and Shiri, 2017] Orang, M. and Shiri, N. (2017). Correlation analysis techniques for uncertain time series. 50(1):79–116.
- [Pan and Chen, 1999] Pan, V. Y. and Chen, Z. Q. (1999). The complexity of the matrix eigenproblem. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 507–516. ACM.
- [Panuccio et al., 2002] Panuccio, A., Bicego, M., and Murino, V. (2002). A hidden markov model-based approach to sequential data clustering. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pages 734–743. Springer.
- [Papadimitriou et al., 2006] Papadimitriou, S., Sun, J., and Philip, S. Y. (2006). Local correlation tracking in time series. In *Data Mining, 2006. ICDM'06. Sixth International Conference on*, pages 456–465. IEEE.
- [Papapetrou et al., 2011] Papapetrou, P., Athitsos, V., Potamias, M., Kollios, G., and Gunopulos, D. (2011). Embedding-based subsequence matching in time-series databases. *ACM Transactions on Database Systems*, 36(3):1–39.
- [Parsons et al., 2004] Parsons, L., Haque, E., and Liu, H. (2004). Subspace clustering for high dimensional data: a review. *Acm Sigkdd Explorations Newsletter*, 6(1):90–105.
- [Patra et al., 2015] Patra, B. K., Launonen, R., Ollikainen, V., and Nandi, S. (2015). A new similarity measure using bhattacharyya coefficient for collaborative filtering in sparse data. 82:163–177.
- [Patterson and Draper, 1997] Patterson, P. and Draper, S. (1997). Selected comparisons between experienced and non-experienced individuals during manual wheelchair propulsion. *Biomedical sciences instrumentation*, 33:477–481.
- [Pentland and Twomey, 1991] Pentland, W. and Twomey, L. (1991). The weight-bearing upper extremity in women with long term paraplegia. *Spinal Cord*, 29(8):521.
- [Petitjean et al., 2014] Petitjean, F. F., Forestier, G., Webb, G. I., Nicholson, A. E., Chen, Y., and Keogh, E. (2014). Dynamic Time Warping Averaging of Time Series Allows Faster and More Accurate Classification. In *2014 IEEE International Conference on Data Mining*, pages 470–479. IEEE.

- [Pollock, 2007] Pollock, D. S. (2007). Wiener–kolmogorov filtering, frequency-selective filtering and polynomial regression. *Econometric Theory*, 23(1):71–88.
- [Price and Campbell, 1999] Price, M. and Campbell, I. (1999). Thermoregulatory and physiological responses of wheelchair athletes to prolonged arm crank and wheelchair exercise. *International journal of sports medicine*, 20(07):457–463.
- [Rakthanmanon et al., 2012a] Rakthanmanon, T., Campana, B., Mueen, A., Batista, G., Westover, B., Zhu, Q., Zakaria, J., and Keogh, E. (2012a). Searching and mining trillions of time series subsequences under dynamic time warping. *18th ACM SIGKDD*, pages 262–270.
- [Rakthanmanon et al., 2012b] Rakthanmanon, T., Campana, B., Mueen, A., Batista, G., Westover, B., Zhu, Q., Zakaria, J., and Keogh, E. (2012b). Searching and mining trillions of time series subsequences under dynamic time warping. *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 262–270.
- [Rakthanmanon and Keogh, 2013] Rakthanmanon, T. and Keogh, E. (2013). Fast shapelets: A scalable algorithm for discovering time series shapelets. In *Proceedings of the 2013 SIAM International Conference on Data Mining*, pages 668–676. SIAM.
- [Ralanamahatana et al., 2005] Ralanamahatana, C. A., Lin, J., Gunopulos, D., Keogh, E., Vlachos, M., and Das, G. (2005). Mining time series data. In *Data mining and knowledge discovery handbook*, pages 1069–1103. Springer.
- [Rand, 1971] Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. 66(336):846–850.
- [Rani and Sikka, 2012] Rani, S. and Sikka, G. (2012). Recent techniques of clustering of time series data: a survey. *International Journal of Computer Applications*, 52(15).
- [Ratanamahatana and Keogh, 2004] Ratanamahatana, C. A. and Keogh, E. (2004). *Making Time-series Classification More Accurate Using Learned Constraints*, pages 11–22.
- [Rehfeld and Kurths, 2014] Rehfeld, K. and Kurths, J. (2014). Similarity estimators for irregular and age-uncertain time series. 10(1):107–122.
- [Rizvandi et al., 2013] Rizvandi, N. B., Taheri, J., Moraveji, R., and Zomaya, A. Y. (2013). A study on using uncertain time series matching algorithms for MapReduce applications. 25(12):1699–1718.

- [Roddick and Spiliopoulou, 2002] Roddick, J. F. and Spiliopoulou, M. (2002). A survey of temporal knowledge discovery paradigms and methods. *IEEE Transactions on Knowledge and data engineering*, 14(4):750–767.
- [Rodgers et al., 1994] Rodgers, M. M., Gayle, G. W., Figoni, S. F., Kobayashi, M., Lieh, J., and Glaser, R. M. (1994). Biomechanics of wheelchair propulsion during fatigue. *Archives of physical medicine and rehabilitation*, 75(1):85–93.
- [Ruggles et al., 1994] Ruggles, D. L., Cahalan, T., and An, K.-N. (1994). Biomechanics of wheelchair propulsion by able-bodied subjects. *Archives of physical medicine and rehabilitation*, 75(5):540–544.
- [Sakoe and Chiba, 1978] Sakoe, H. and Chiba, S. (1978). Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal processing*, 26(1):43–49.
- [Salles et al., 2017] Salles, T., Rocha, L., Mourão, F., Gonçalves, M., Viegas, F., and Meira Jr, W. (2017). A two-stage machine learning approach for temporally-robust text classification. *Information Systems*, 69:40–58.
- [Salvador and Chan, 2007] Salvador, S. and Chan, P. (2007). Toward accurate dynamic time warping in linear time and space. *intelligent data analysis*, 11(5):561–580.
- [Sauret, 2010] Sauret, C. (2010). Cinétique et énergétique de la propulsion en fauteuil roulant manuel.
- [Schäfer, 2015] Schäfer, P. (2015). The boss is concerned with time series classification in the presence of noise. *Data Mining and Knowledge Discovery*, 29(6):1505–1530.
- [Senin et al., 2014] Senin, P., Lin, J., Wang, X., Oates, T., Gandhi, S., Boedihardjo, A. P., Chen, C., Frankenstein, S., and Lerner, M. (2014). Grammarviz 2.0: a tool for grammar-based pattern discovery in time series. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 468–472. Springer.
- [Senin and Malinchik, 2013] Senin, P. and Malinchik, S. (2013). Sax-vsm: Interpretable time series classification using sax and vector space model. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pages 1175–1180. IEEE.
- [Shannon, 2001] Shannon, C. E. (2001). A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3.

- [Shatkay and Zdonik, 1996] Shatkay, H. and Zdonik, S. B. (1996). Approximate queries and representations for large data sequences. In *Data Engineering, 1996. Proceedings of the Twelfth International Conference on*, pages 536–545. IEEE.
- [Siyou Fotso et al., 2015] Siyou Fotso, V. S., Mephu Nguifo, E., and Vaslin, P. (2015). Symbolic representation of cyclic time series: application to biomechanics. In *ICML'15 Workshop on Constructive Machine Learning*.
- [Siyou Fotso et al., 2016] Siyou Fotso, V. S., Mephu Nguifo, E., and Vaslin, P. (2016). Comparison of classification algorithms to fdtw. <http://fc.isima.fr/~siyou/fdtw>.
- [Stefan et al., 2013] Stefan, A., Athitsos, V., and Das, G. (2013). The move-split-merge metric for time series. *IEEE transactions on Knowledge and Data Engineering*, 25(6):1425–1438.
- [Stenberg et al., 1967] Stenberg, J., Astrand, P., Ekblom, B., Royce, J., and Saltin, B. (1967). Hemodynamic response to work with different muscle groups, sitting and supine. *Journal of Applied Physiology*, 22(1):61–70.
- [Stoboy et al., 1971] Stoboy, H., Rich, B. W., and Lee, M. (1971). Workload and energy expenditure during wheelchair propelling. *Spinal Cord*, 8(4):223.
- [Sun et al., 2014] Sun, Y., Li, J., Liu, J., Sun, B., and Chow, C. (2014). An improvement of symbolic aggregate approximation distance measure for time series. *Neurocomputing*, 138:189–198.
- [Theisen et al., 1996] Theisen, D., Francaux, M., Fay, A., and Sturbois, X. (1996). A new procedure to determine external power output during handrim wheelchair propulsion on a roller ergometer: a reliability study. *International journal of sports medicine*, 17(08):564–571.
- [Ulanova et al., 2015] Ulanova, L., Begum, N., and Keogh, E. (2015). Scalable clustering of time series with u-shapelets. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, pages 900–908. SIAM.
- [van der Slikke et al., 2017] van der Slikke, R. M., Bregman, D. J., Berger, M. A., de Witte, A. M., and Veeger, D.-J. H. E. (2017). The future of classification in wheelchair sports; can data science and technological advancement offer an alternative point of view? *International journal of sports physiology and performance*, pages 1–25.
- [van der Woude and de Groot, 2005] van der Woude, L. H. and de Groot, S. (2005). Wheelchair propulsion: a straining form of ambulation. *Indian Journal of Medical Research*, 121(6):719.

- [Van der Woude et al., 2006] Van der Woude, L. H., de Groot, S., and Janssen, T. W. (2006). Manual wheelchairs: research and innovation in rehabilitation, sports, daily life and health. *Medical Engineering and Physics*, 28(9):905–915.
- [Van Wijk and Van Selow, 1999] Van Wijk, J. J. and Van Selow, E. R. (1999). Cluster and calendar based visualization of time series data. In *Information Visualization, 1999.(Info Vis' 99) Proceedings. 1999 IEEE Symposium on*, pages 4–9. IEEE.
- [Vanlandewijck et al., 1999] Vanlandewijck, Y., Daly, D., and Theisen, D. (1999). Field test evaluation of aerobic, anaerobic, and wheelchair basketball skill performances. *International journal of sports medicine*, 20(08):548–554.
- [Vasimalla, 2017] Vasimalla, K. (2017). A Survey on Time Series Data Mining. *International Journal of Innovative Research in Computer and Communication Engineering*, 2(5):170–179.
- [Vegter et al., 2014] Vegter, R. J. K., Lamoth, C. J., de Groot, S., Veeger, D. H. E. J., and van der Woude, L. H. V. (2014). Inter-individual differences in the initial 80 minutes of motor learning of handrim wheelchair propulsion. *PloS one*, 9(2):e89729.
- [Ville and Winance, 2006] Ville, I. and Winance, M. (2006). To work or not to work? the occupational trajectories of wheelchair users. *Disability and rehabilitation*, 28(7):423–436.
- [Wand and Jones, 1994] Wand, M. P. and Jones, M. C. (1994). *Kernel smoothing*. Crc Press.
- [Wang et al., 2015] Wang, W., Liu, G., and Liu, D. (2015). Chebyshev Similarity Match between Uncertain Time Series. 2015:1–13.
- [Wang et al., 2013] Wang, X., Mueen, A., Ding, H., Trajcevski, G., Scheuermann, P., and Keogh, E. (2013). Experimental comparison of representation methods and distance measures for time series data. 26(2):275–309.
- [Wang and Zhang, 2013] Wang, Y.-X. and Zhang, Y.-J. (2013). Nonnegative matrix factorization: A comprehensive review. *IEEE Transactions on Knowledge and Data Engineering*, 25(6):1336–1353.
- [Weber et al., 2001] Weber, M., Alexa, M., and Müller, W. (2001). Visualizing time-series on spirals. In *Infovis*, volume 1, pages 7–14.
- [Woude et al., 1986] Woude, L. V. D., GROOT, G. D., Hollander, A., Schenau, G. V. I., and Rozendal, R. (1986). Wheelchair ergonomics and physiological testing of prototypes. *Ergonomics*, 29(12):1561–1573.

- [Wu et al., 1996] Wu, D., Singh, A., Agrawal, D., El Abbadi, A., and Smith, T. R. (1996). Efficient retrieval for browsing large image databases. In *Proceedings of the fifth international conference on Information and knowledge management - CIKM '96*, pages 11–18, New York, New York, USA. ACM Press.
- [Yagoubi et al., 2018] Yagoubi, D. E., Akbarinia, R., Kolev, B., Levchenko, O., Masseglia, F., Valduriez, P., and Shasha, D. (2018). Parcorr: efficient parallel methods to identify similar time series pairs across sliding windows. *Data Mining and Knowledge Discovery*, 32(5):1481–1507.
- [Yagoubi et al., 2017] Yagoubi, D. E., Akbarinia, R., Masseglia, F., and Shasha, D. (2017). Radiussketch: Massively distributed indexing of time series. In *Data Science and Advanced Analytics (DSAA), 2017 IEEE International Conference on*, pages 262–271. IEEE.
- [Yeh et al., 2009] Yeh, M.-Y., Wu, K.-L., Yu, P. S., and Chen, M.-S. (2009). Proud: a probabilistic approach to processing similarity queries over uncertain data streams. In *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*, pages 684–695. ACM.
- [Yi and Faloutsos, 2000] Yi, B.-K. and Faloutsos, C. (2000). Fast time sequence indexing for arbitrary lp norms. VLDB.
- [Yu et al., 2011] Yu, D., Yu, X., Hu, Q., Liu, J., and Wu, A. (2011). Dynamic time warping constraint learning for large margin nearest neighbor classification. *Information Sciences*, 181(13):2787–2796.
- [Zakaria et al., 2012] Zakaria, J., Mueen, A., and Keogh, E. (2012). Clustering time series using unsupervised-shapelets. In *Data Mining (ICDM), 2012 IEEE 12th International Conference on*, pages 785–794. IEEE.
- [Zhang et al., 2016] Zhang, Q., Wu, J., Yang, H., Tian, Y., and Zhang, C. (2016). Unsupervised feature learning from time series. In *IJCAI*, pages 2322–2328.
- [Zhang et al., 2015] Zhang, Z., Tang, P., and Duan, R. (2015). Dynamic time warping under pointwise shape context. *Information Sciences*, 315:88–101.
- [Zhao and Itti, 2016] Zhao, J. and Itti, L. (2016). shapedtw: shape dynamic time warping.

# List of Figures

2.1	Relation between Accuracy and the number of segment on FISH dataset. The accuracy is computed from the algorithm one nearest neighbor (1NN) associated with PDTW. When the number of segments considered is very small (bellow 20), there is a loss of information and the accuracy is reduced. However, considering all the points in the time series, do not produce a maximum accuracy due to the presence of noise or singularities [Keogh and Pazzani, 2001b] in the data. . . . .	29
2.2	IDDTW operating principle. Depth represents approximation levels, A represents approximate distance and B is best_so_far [Chu et al., 2002]. . . . .	32
2.3	Visual comparison of two time series from the two classes of the coffee dataset. Left: the original time series(286 data points), right: representation using PAA with 88 segments . . . . .	40
2.4	Comparison of the number of tested values of the parameter number of segments with the FDTW and IDDTW. Datasets are sorted according to the length of the time series (x-axis). . . . .	45
2.5	Comparison of the execution time of the Brute-force search algorithm, FDTW and IDDTW. . . . .	46
2.6	Critical difference diagram for FDTW and 36 other classification algorithms on 6 simulated datasets. FDTW is ranked 3rd / 37 algorithms	46
2.7	Eight types of time series corresponding to the vocabulary of 8 gestures. . . . .	47
3.1	Example of classification of time series of two birds' calls (green: Olive-sided Flycatcher; blue: White-crowned Sparrow) using on the one hand the Euclidean distance(left), and on the other hand the u-shapelet (right) [Ulanova et al., 2015] . . . . .	50
3.2	Multiset-based model of uncertain time series. . . . .	53
3.3	PDF-based model of uncertain time series . . . . .	54
3.4	Bhattacharyya . . . . .	56

3.5	Geometric representation of LoCo similarity . . . . .	61
3.6	The execution time of ED and FOTS score is a function of the length of time series. The computation time of ED is smaller than that of FOTS. . . . .	68
4.1	Cyclic time series form manual wheelchair locomotion . . . . .	72
4.2	Properties of a cycle . . . . .	73
4.3	Piecewise aggregate approximation of a cyclic time series . . . . .	74
4.4	Symbolic Aggregate approXimation of a cyclic time series . . . . .	74
4.5	Extended Symbolic Aggregate approXimation of a cyclic time series .	75
4.6	Trend Symbolic Aggregate approXimation of a cyclic time series . .	75
4.7	Properties of a cycle . . . . .	76
4.8	Threshold for the segmentation of cyclic time series . . . . .	76
4.9	Segmentation . . . . .	77
4.10	Some properties are computed on each cycle . . . . .	77
4.11	Classification of cycles based on properties . . . . .	78
4.12	Symbolic representation of cyclic time series . . . . .	79
4.13	Example of a z-moment (Mz) time series measured on the [left/right] rear wheel of the FRET-2. . . . .	80
4.14	Time series recorded by torsor sensor are noisy . . . . .	81
4.15	Plot of Mz residues, which represent the differences between Mz values measured by the sensor and real values applied during the calibration process. . . . .	86
4.16	Classification tree of wheelchair users based on their propulsion abilities.	90
A.1	Wheelchair Aerobic Fitness Trainer (WAFT) photograph [Langbein and Fehr, 1993].	98
A.2	Picture of a wheelchair on a roller ergometer with mechanical braking by friction belt on a flywheel. [Rodgers et al., 1994]. . . . .	98
A.3	Exercise testing on a motor driven treadmill [Van der Woude et al., 2006]	99
A.4	Photograph of an experiment on a simulator connected to a flywheel ([Brattgård et al., 1970]) . . . . .	100
A.5	Captioned picture of the adjustable wireless wheelchair ergometer (FRET-2). . . . .	101
A.6	Schematic principle of the six-component force and torque sensor (patent WO 1995001556 A1) mounted of both rear wheels of the MWC field ergometer used in this study (FRET-2). . . . .	102
A.7	Balance of forces applied to a manual wheelchair during propulsion. For the clarity of the figure, the analysis of the movement of the subject + MWC system is reduced to that of the system's centre of gravity, G [De Saint Remy, 2005] . . . . .	104
A.8	Markers for kinematic analysis of manual wheelchair locomotion[Sauret, 2010]	105

B.1	These figures show the average of two segments. In the first case (a) the data points of the segment are far from the average, in the second case (b) they are close to the average. Replace data points of a segment by their average introduces an error that can be measured from the gap between the points and the average. . . . .	108
B.2	This figure shows the first 100 points of the first time series of the fordA dataset available in the UCR [Chen et al., 2015] database. Time series are normalized. The two horizontal lines delimit the interval corresponding to twice the standard deviation and minus two times the standard deviation of the points of the time series. We can observe that the points outside this range are at the ends of the time series. . . . .	109
B.3	Two-to-two comparison of the classification errors of the algorithm 1-Nearest Neighbor (1-NN) using Euclidean distance with 1-NN using two variations of the temporal warping algorithm on data raw and compact data . . . . .	116
C.1	Distribution of data around the mean . . . . .	121
C.2	Estimated probability density of data . . . . .	122
C.3	Estimated probability density of data for cauchy's law . . . . .	123
C.4	Estimated probability density of data for Student's law . . . . .	124
C.5	Estimated probability density of data for Logistic law . . . . .	124



# List of Tables

1.1	Temporal-Proximity-Based Clustering Approach . . . . .	21
1.2	Representation-Based Clustering Approach Paper . . . . .	22
1.3	Model-Based Clustering Approach . . . . .	23
2.1	Classification errors associated with the number of segments $N$ chosen by the heuristics IDDTW and FDTW. When two numbers of segments $N_1$ and $N_2$ are associated with the same classification error, the smallest is considered. The classification error is calculated based on the 3 fold cross validation applied on the training set. . . . .	42
2.2	Comparison of generalization errors. In <b>italics</b> , the smallest generalization error. In <b>bold</b> , the smallest generalization error between IDDTW and FDTW. $N$ is the number of segments selected and $\ell$ is the number of data points in a segment ( $\ell = \lfloor \frac{n}{N} \rfloor$ ). The generalization error is computed on the testing set. Note : DTW (r) is a constraint version of DTW where the number of consecutive data points that can be compared to a single point during the warping is bounded. r represents the size of the warping windows . . . . .	44
3.1	Datasets. . . . .	66
3.2	Comparison of the Rand Index of SUSH (RI_SUSH) and FOTS-SUSH (RI_FOTS). The best Rand Index is in bold . . . . .	67
3.3	Comparison between k-Shape, USLM and FOTS-SUShapelet. . . . .	68
4.1	Straight displacement in manual wheelchair . . . . .	85
4.2	Asymmetry (Edit distance) of subjects' propulsion with regard to their number of years of practice. . . . .	85
4.3	Anthropometric and physiological characteristics of the subjects who participated in this study. . . . .	87
4.4	Occurrence frequency of each cycle type in each subject's displacements.	88
4.5	Similarity matrix between all wheelchair users. . . . .	89

B.1	85 UCR datasets used for experimental validation. The full list is available here [Chen et al., 2015] . . . . .	115
B.2	The (1) column presents <b>numbers</b> of the datasets. The column (2) the <b>reduced length</b> of the time series. The column (3) is the <b>ratio</b> of the length of the reduced time series over the length of the initial time series. The (4) column designates the <b>1-Nearest Neighbor</b> algorithm, associated to the <b>Euclidean distance</b> . The (5) column designates the algorithm of <b>1- Nearer Neighbor</b> , associated with the algorithm of <b>dynamic dynamic temporal deformation</b> using a <b>constraint</b> called deformation window that allows to stop the comparison of time series when one perceives that they are very different. The (6) column represents <b>1-Nearest Neighbor</b> algorithm associated to the <b>unconstrained dynamic time warping</b> applied to the <b>raw data</b> . The (7) column represents the <b>algorithm</b> . <b>1-Nearest Neighbor</b> associated with the <b>dynamic time warping algorithm without constraints</b> , applied on the <b>compact representations</b> produced by our algorithm. We compare firstly, the classification error of the algorithms (6) and (7) the smallest error is in <b>bold</b> . We then compare the classification errors of algorithms (4), (5), (6) and (7) the smallest error is put <b>italicized</b> . . . . .	119
C.1	descriptive . . . . .	121
C.2	normality test show that residues(uncertainty) do not follow normal law . . . . .	123
D.1	Evolution of manual wheelchair propulsion technology with training .	129



# **Extraction de connaissances de séries temporelles cycliques et incertaines : application à l'analyse de la locomotion en fauteuil roulant manuel**

---

## **Résumé**

L'évaluation des capacités motrices des utilisateurs de Fauteuil roulant manuel est souvent subjective, car elle se base sur l'avis d'un expert. C'est pourquoi, un fauteuil roulant manuel ergomètre de terrain a été fabriqué. Il permet d'enregistrer les efforts effectués par les utilisateurs de fauteuil roulant manuel pendant leur déplacement. Les mesures ainsi effectuées sont des séries temporelles ayant les caractéristiques spécifiques suivantes : elles sont longues, incertaines et cycliques. En nous appuyant sur ces mesures ainsi que sur leurs propriétés, l'objectif de cette thèse est d'effectuer une analyse objective de la locomotion en fauteuil roulant manuel. À cet effet, nous proposons trois modèles. Le premier modèle est une heuristique permettant de trouver le nombre judicieux de segment à considérer pour la compression des séries temporelles avec l'algorithme d'approximation par morceau, tout en concevant l'information contenue dans les séries temporelles. Si le principe de fonction de cette heuristique est similaire à celui de la recherche gloutonne randomisée, elle a la particularité de proposer une stratégie spécifique de recherche globale. Le deuxième modèle est une mesure de similarité qui permet de capturer la structure fondamentale des séries temporelles et qui est robuste à la présence d'incertitude. Cette mesure de similarité est basée sur la comparaison à l'aide de la norme de Frobenius des vecteurs propres des matrices d'autocorrélation des séries temporelles. Le troisième modèle est une représentation symbolique de séries temporelle cycliques basée sur les propriétés de cycles qui utilise un algorithme de segmentation des séries temporelles cycliques en cycle et un algorithme de classification non supervisée pour comparer les cycles en fonction de leurs propriétés. Cette représentation symbolique permet une meilleure visualisation et une meilleure analyse basée sur les propriétés des cycles des séries temporelles cycliques. Nos modèles permettent de mettre en évidence le caractère asymétrique de la locomotion en fauteuil roulant manuel et d'établir que l'asymétrie de la locomotion diminue avec les années de pratique. Ils permettent également d'évaluer de manière objective et intelligible les capacités motrices des utilisateurs de fauteuil roulant manuel.

---

**Séries temporelles, compression, comparaison, représentation**

---

# Extraction of knowledge from cyclical and uncertain time series: application to Manual Wheelchair locomotion analysis

---

## Abstract

The assessment of the motor skills of manual wheelchair users is often subjective because it is based on expert opinion. Therefore, a manual field ergometer wheelchair was conceived and constructed. It records the efforts made by manual wheelchair users during their locomotion. The measurements made are time series with the following specific characteristics: they are long, uncertain and cyclical. Based on these measurements and their properties, the objective of this thesis is to perform an objective analysis of manual wheelchair locomotion. To this, we proposed three models. The first model is a heuristic to find the appropriate number of segments to consider for time series compression with the piece aggregate approximation algorithm while keeping the information they contained. If the principle of this heuristic is similar to that of greedy randomized adaptive search, it has the particularity of proposing a specific global search strategy. The second model is a similarity measure that captures the fundamental structure of time series and is robust to the presence of uncertainty. This similarity measure is based on the comparison using the Frobenius norm of eigenvectors of time series autocorrelation matrices. The third model is a symbolic representation of cyclic time series based on cycle properties that utilises a segmentation algorithm of cyclic time series in cycles and an unsupervised classification algorithm to compare cycles according to their properties. This symbolic representation allows better visualization and analysis based on the cycle properties of cyclic time series. Our models highlight the asymmetrical nature of manual wheelchair locomotion and establish that the asymmetry of locomotion decreases with years of practice. They also provide an objective and intelligible assessment of the motor abilities of manual wheelchair users.

---

Time series, compression, comparison,  
representation University Clermont Auvergne

