

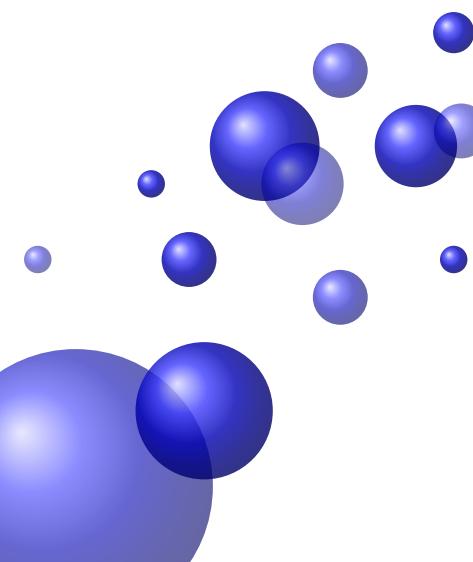


Rediger un gros document avec L^AT_EX

Nom de l'auteur

Sous la direction de Le Tuteur.

Le May 6, 2018



Rediger un gros document avec L^AT_EX

Nom de l'auteur

Sous la direction de Le Tuteur.

Abstract

My abstract: vous pouvez notez ici que l'espacement entre le mot abstract et les : n'est pas le même qu'en français, comme le veut la typographie anglaise.

Acknowledgment

Je tiens à remercier toutes les personnes qui m'ont aidé à rédiger cet article, Namrod pour la partie bibliographie, Francis Walter, pour ses conseils ainsi que les personnes ayant participé à la correction de ce document.

A mon père, ma mère, mes frères et sœurs

Summary

Summary	viii
Introduction	3
I Background and related works	5
1 Analysis of the wheelchair locomotion	7
2 Knowledge discovery on time series	17
II Contributions	33
3 Preprocessing of time series	35
4 Uncertain time series u-shapelet discovery	57
5 SAX-P	75
6 Application to manual wheelchair locomotion	85
Conclusion	87
A Hellinger Based Distance for Uncertain time series	89
B An optimal approach to time series segmentation: Application to the supervised classification	93
Index	109
Glossaire	109

Liste des abreviations, des sigles et des symboles	109
References	128
Table des figures	131
Liste des tableaux	134
Table des matieres	138

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], 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 catalogs of time series.
- In paleoecology, ...
- In medicine, the analysis of electrocardiogram is used to prevent heart attacks[)]. 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 problems arise from the uncertainty contained in time series data, the difficulty of defining a form of similarity measure based on human perception, and the high

SUMMARY

SUMMARY

dimensionality of time-series data. These constraints show us that three major issues are involved.

- 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?
- 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.
- 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.

The aim of our work is to propose algorithm to deal with thoses 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 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 Ergometer Wheelchair 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, and a field ergometer wheelchair (FRET-1) has been designed and manufactured for this purpose [1, 2]. This ergometer is equipped with a moment sensor that measures the forces applied to the handrails as well as the acceleration and movement of the FRET-1 [1, 2]. 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 period of pushing and freewheeling,

- 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 them.

Plan

The thesis is organised as follow

- **Chapter 1** present the state of art
- **Chapter 2** 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 paper, we demonstrate that to compress time series using Piecewise Aggregate Approximation (PAA) is a simple strategy that greatly increases the quality of the alignment with DTW this is particularly true for synthetic data sets.
- **Chapter 3** The abstract.
- **Chapter 4** The analysis of cyclic time series from biomechanics 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 (S1, S2, S3) 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.

SOMMARY

SOMMARY

Part I

Background and related works

Chapter 1

Analysis of the wheelchair locomotion

1.1 Introduction

Wheelchair locomotion concerns many people, for different reasons: genetic(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 The2016]. However, the use of the manual wheelchair is not without risk.

1.2 The problem of locomotion manual wheelchair locomotion

Although the use of FRM improves the mobility of its users, doctors quickly realized that its use often led to sedentarization, leading to 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 FRM can lead to specific injuries and pains [Johnson *et al.*2004], especially in the shoulder, and at the elbow, wrist and hand. In [Pentland and Twomey1991] authors claim that 73% of paraplegic individuals suffered from shoulder pain. In addition, sitting and prolonged sitting of FRM users causes dermatological problems such as bedsores or pressure ulcers, due to immobility, loss of sensitivity and incontinence. In addition, these symptoms are recognized as a major cause of discontinuation of the use of FRM [Van der Woude *et al.*2006, Ville and Winance2006], thus the sedentarization of users. Lundqvist et al. [Lundqvist *et al.*1991] showed that upper limb pain was

1.3 Tools to evaluate manual wheelchair locomotion

the only factor correlated with poor quality of life in FRM subjects. The difficulty for the therapist is to practice a daily physical activity, adapted to the individual, and limit orthopedic problems and thus promote the use of the FRM 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 Groot2005, 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 to say, the ergonomics and the adequacy of the system {subject + manual wheelchair} with the external physical environment (ramps, lifts, corridor widths, etc.).
- The improvement of the manual wheelchair regarding the design and the mechanical principles of propulsions;
- **Improving the subject's physical abilities**, that is, improving propulsion techniques, as well as rehabilitation techniques and training programs.

Bio-mechanics work has been conducted in LIMOS to identify and quantify traumatic factors such as. This work led to the construction of a measuring tool: an Ergo-meter Field Chair.

1.3 Tools to evaluate manual wheelchair locomotion

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

1.3.1 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. The crank ergometers established that the mechanical work of the upper limbs was less efficient than that of the lower limbs and also that the physical capacities evaluated by the maximum oxygen consumption of manual wheelchair users depended on their level of spinal injury (cervical, thoracic or lumbar injury)¹. One of the main limitations of crank Ergometers is that the motion measured from a crank Ergometer is not representative of the FRM propulsion motion, most of which is propelled by handrails [Åstrand and Saltin1961, Bergh *et al.*1976, ?].

¹This assertion will be discussed later in chapter 5

1.3.2 Roller Ergometers

To reproduce more precisely the specificities of hand-held FRM locomotion, Brouha and Krobath [Brouha and Krobath 1967], as early as 1967, used a roller ergometer to measure cardiac and respiratory responses during an FRM exercise continuously. 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 FRM. The FRM frame was attached to the ergometer, and the subjects simulated locomotion by applying forces to the handrails, causing the rear wheels of the FRM 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) from the relationship between oxygen consumption and mechanical power calculated during an incremental exercise on a crank ergometer.

The problem with the 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 in the field. To take this phenomenon into account, by connecting the rollers to a small flywheel. However, the two rear wheels were on the same rollers, which did not allow 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 FRM installation (Figure 1.1). 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 wheels².

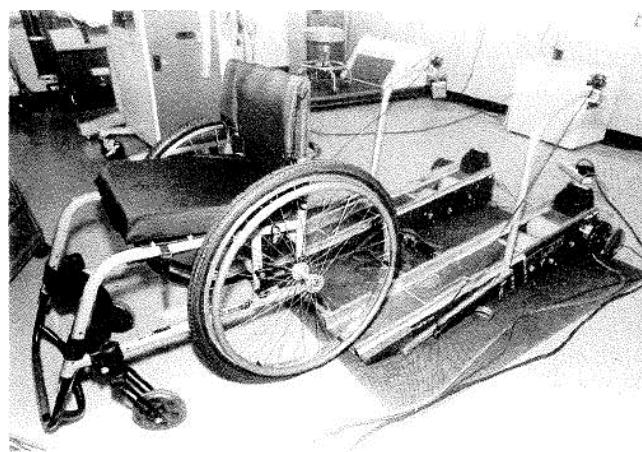


Figure 1.1: WAFT photograph [Langbein and Fehr 1993].

²This separation is essential to establish the dissymmetry of wheelchair locomotion and will be discussed in more detail in chapter 5

1.3 Tools to evaluate manual wheelchair locomotion

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 4), an electric motor creating a frictional moment around the roller rotation axes [Coutts and Stogryn 1987, Kerk *et al.* 1995, Patterson and Draper 1997, 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.

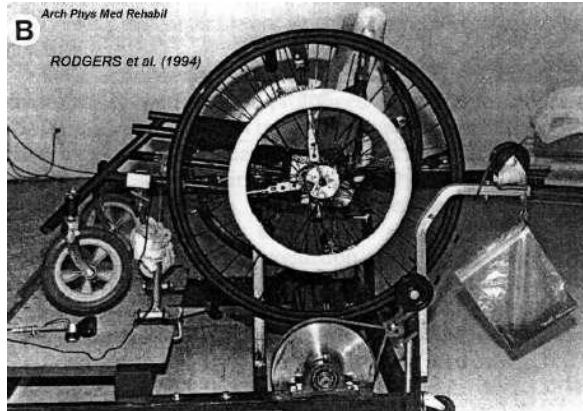


Figure 1.2: Photograph of a wheelchair on a roller ergometer with mechanical braking by friction belt on a flywheel. (Rodgers *et al.*, 1994).

The main advantage of roller ergometers is that they allow subjects to be studied with their FRM. Moreover, they occupy little space in the laboratory and allow the FRM to be completely immobilized, thus ensuring the stability of the subject on the FRM and facilitating the measurement of the various physiological parameters. However, the various methods of determining the 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.

1.3.3 Treadmill

Like roller ergometers, the main advantage of treadmills is that they allow subjects to be studied with their FRM. Since the four wheels of the FRM roll on the belt,

the rolling friction forces are most certainly equivalent to those that exist in the field. However, treadmills, unlike roller ergometers, allow to define a rolling speed of the treadmill and also a slope, i.e., an inclination of the treadmill concerning the horizontal. The main disadvantage of the 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 the sides using a surface strip that limits lateral movements [Claremont and Maksud1985]. However, it has still not been demonstrated that the propulsion technics used was identical on a treadmill and in the field.



Figure 1.3: Photograph of an experiment on a conveyor belt (van der Woude et al., 2006)

1.3.4 Wheelchair simulators

To overcome the problems related to rolling resistance, researchers chose to fix the rear wheels of the FRM without contact with the ground, on a rigid and fixed chassis on which the Subject could sit. The advantage of FRM 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 quantified. This aspect is the major disadvantage of the simulators because neither the forces of resistance to the advance nor the kinematics of the FRM is modified according to the movements of the Subject on the seat.

1.3.5 Wheelchair Field-Ergometer

To improve the efficiency of wheelchair propulsion, a Wheelchair Ergometer (FRET-2) equipped with sensors has been manufactured. The sensors installed on the

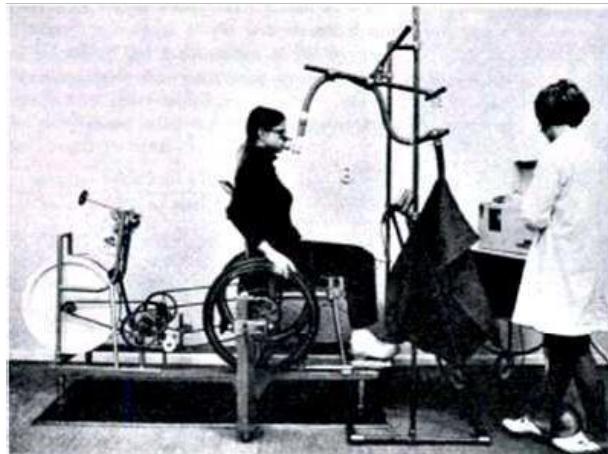


Figure 1.4: Photograph of an experiment on a simulator connected to a flywheel (Brattgard et al., 1970)

wheelchair measure the physical stresses applied to the Manual Wheelchair (FRM) during actual use and record them.

The sensors are located on the right and left wheels of the manual wheelchair, on the footrest, on the seat and the backrest (see Figure 6.1). These sensors measure the forces and moments of these forces applied to each of the systems mentioned above. The moment of a force concerning a given point is a vectorial physical quantity which translates the ability of a force to turn a mechanical system around that point, often called a pivot [20]. The sensors installed on the FRM were used to measure the kinematic parameters (speed, acceleration) of the movement of the Manual Wheelchair (FRM), as well as its position relative to the Earth's magnetic north.

The measurements recorded by the sensors and subjected to our analysis consist of 44 attributes; 30 of the 44 attributes relate to the measurement of the torque constituted by force applied to the systems mentioned above and the moment of this force to an axis of rotation. For each of the five systems, we have three components of the force (F_x , F_y , F_z) and the momentum (M_x , M_y , M_z) that apply to it. The 14 other attributes tell us about the kinematics of the manual wheelchair and its position relative to the Earth's magnetic north.

1.4 Knowledge discovery on wheelchair time series

After the construction of measuring instruments, these are used to measure the efforts made by Manual Wheelchair users. Thus, several experiments were conducted with subjects, and the forces involved in the user locomotion of the Manual Wheelchair were measured. The abundance of measurements made poses the problem of the exploitation of these measures for knowledge extraction. Two main

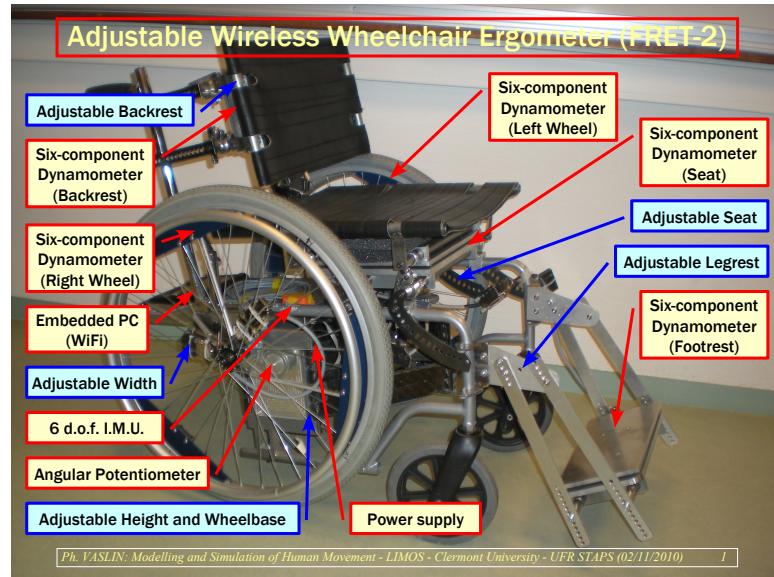


Figure 1.5: Presentation of the different sensors of wheelchair field-ergometer

approaches can be used to analyze measurements from Manual Wheelchair 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. Those two approaches are complementary. In this section, we present the work of these two families, which will allow us to position our work to the existing.

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

- The doctoral thesis of Nicolas de Saint REMY (2005) [1] who proposed a mechanical model relating the forces applied to a Manual Wheelchair and its displacement as illustrated in Figure 1.1 and Equation 1.1. It made it possible to highlight the fact that the acceleration of the FRM is a function of the movements of the subject:
- The doctoral thesis of Christophe SAURET (2010) [2] who proposed a method of calculating the mechanical power developed by manual wheelchair users to move. This model analyzes the kinetics (trajectory and speed) of the subject segments and the Manual Wheelchair. A segment is the body part of a user or a manual wheelchair between two markers. Figure 1.2 shows the layout of the markers used for this analysis.

When it comes to conducting classes, the use of data mining models is more appropriate. For example, ranking wheelchair users is an essential task for Paralympic sports, it is usually based on a set of tests and a (subjective) assessment by an

1.4 Knowledge discovery on wheelchair locomotion

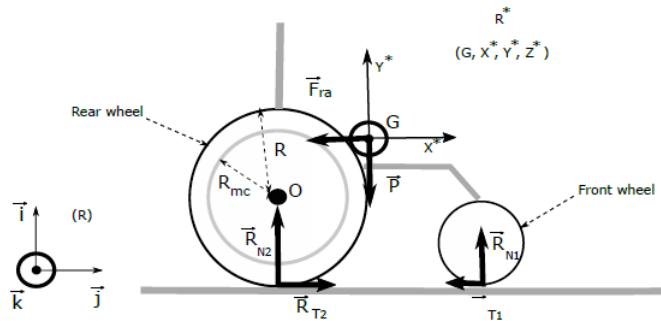


Figure 1.6: Balance of forces applied to a manual wheelchair during its use; the analysis of the movement of the subject-chair system has been reduced to that of its center of gravity

expert. On the other hand, technical advances have made it possible to measure the efforts made by wheelchair users during their movement using sensors. That's how 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. This question is very interesting, in fact, more and more works in the literature suggest using the tools developed in data mining for a better understanding of human locomotion.

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

In [Athanasios and Clark 2009] the others present a model based on Bayesian networks to improve the treatment of patients in wheelchairs with a spinal injury. Treatment of patients in wheelchairs is based on the level of spinal injury and symptoms. A lesion in the spine has three consequences: an inconsistency of the bowel, an inconsistency of the bladder, a loss of sensitivity of the skin. The higher the lesion, the more widespread its effects on patients are. Thus a patient with a low lesion will see his legs affected 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.

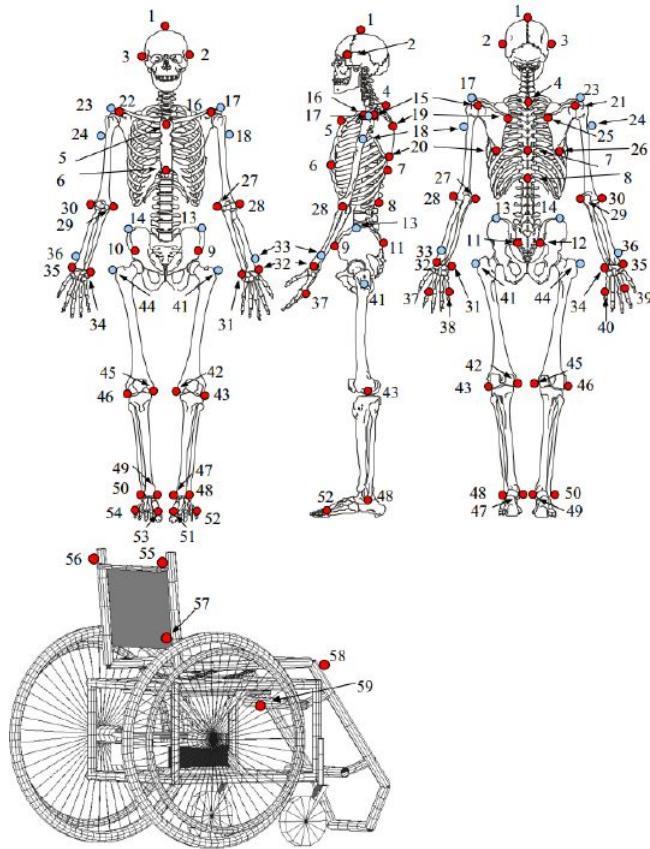


Figure 1.7: Balance of forces applied to a manual wheelchair during its use; the analysis of the movement of the subject-chair system has been reduced to that of its center of gravity

1.5 Conclusion

Throughout this chapter, we show that there is a significant number of Manual Wheelchair users and that it is crucial to analyze this exercise to improve the living conditions of people moving in a Manual Wheelchair. We have presented tools designed and manufactured for locomotion analysis in Manual Wheelchair and some work using data mining mechanics models to improve the study of locomotion in Manual Wheelchair or to help poster better diagnosis for the adverse effects treatment of spinal cord injury causing paralysis and requiring the use of Wheelchair.

The literature **uses** mechanical or data mining models for manual wheelchair locomotion analysis. In this work, however, we want to **design** data mining models that take into account both the specificities of manual wheelchair locomotion data and their use to analyze manual wheelchair locomotion from a new angle. In the rest of this work, we present the existing works in the literature of knowledge extraction on time-series which will allow us to identify an chose useful approaches for the

1.5 Conclusion

Chapter 1. Analysis of the wheelchair locomotion

analysis of wheelchair locomotion.

Chapter 2

Knowledge discovery on time series

2.1 Introduction

Datasets can be grouped into four main categories regarding their temporality [Roddick and Spiliopoulou2002] :

- 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, preprocessing technics have been proposed in the literature.

2.2 Preprocessing of time series

2.2.1 Denoising time series

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

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 Jones1994].

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 : $g(x) = \beta_0 + \beta_1 x + \dots + \beta_m x^m$ of a certain degree m for which $E(Y - g(x))^2$ is as small as possible [Kendall1961].

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 [Pollock2007].

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. That's how you distinguish low-pass, high-pass, bandpass, or notch filter. These filters can also be combined in many ways: in cascade, in parallel, etc [Buttkus2012].

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].

2.2.2 Reducing uncertainty

Another important step of preprocessing time series consists of reducing the uncertainty that they contained. Some transformations have been introduced in literature for this purpose.

Uncertain moving average : For uncertain time series, each value is associated with a standard deviation representing uncertainty. The uncertain moving average 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: $x^{UMA} = \langle x_1^{UMA}, \dots, x_m^{UMA} \rangle$ 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 Shiric]

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: $X = \langle X_1, \dots, X_m \rangle$, its normal form: $\hat{X} = \langle \hat{X}_1, \dots, \hat{X}_m \rangle$ 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 Shiric]. That is,

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

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

2.2.3 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] where he uses a Discrete Fourier Transform to compress the time series. In the same family, we can also cite the following time series representations : Discrete Wavelet Transform (DWT) [Chan and Fu1999a], 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.*] 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 Agon2012], 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 ready to extract information from them. Several datamining tasks can be performed with time series.

2.3 Similarity Measures

It is essential To be able to compare time series for time series data mining tasks. Most often, time series similarity functions compare time series as humans would do, based on their shape. Human recognition can without much of a stretch understand and look at the likenesses between two time series based on amplitude, scale, temporal warping, noise, and outliers. As indicated by [Fu2011, Ralanamahatana *et al.* 2005, Esling and Agon2012], 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 be universal meaning that not be restricted to particular time series and 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 still known as invariance.

2.3.1 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.*].

Scaling and translation invariances : We should be able to perceive the similarity of sequences in spite of contrasts in amplitude (scaling) and offset (translation). 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 a phrase 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 with measurement periods of different duration.

Occlusion invariance : We should be able to compare two time series even if sub-sequences of the time series are missing; we can still compare the sequences by ignoring the sub-sequences that do not match well. For example, suppose an archaeologist 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 skull found to those stored in the database. This comparison should be possible even if the skull found 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, audio signals that were recorded indoors and outdoors might be considered similar, despite the fact that outdoor signals will 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 are then considered by more sophisticated distances or dissimilarities functions. In the next section, we review the most common such distance measures.

2.3.2 Categories of time series similarity function

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

Shape Based similarity function compares two time-series based on the sum of the distances in a Euclidian space between data point of approximately the same timestamp. By doing so, the distance between two time-series with similar shape will be low. On the contrary, the distance between time-series that has a different shape will be high. In this family, we have L_p norm [Yi and Faloutsos2000, Keogh and Kasetty2003a], Dynamic time warping distance [Myers *et al.*]. However, those distances are sensitive to noise.

Edit Based distance allows evaluating the dissimilarity between two character strings. Their principle is the following: Edit based distance 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. It 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 time series data point can be compared to several other time series data points (insertion). We can cite three edit based distances in time series domain : 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 are among well-known distance measures under this category. 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. These dissimilarity functions are able to handle noisy regions and outliers.

Feature Based distance : this distance has been designed to ensure some invariances e.g. rotation invariance. Time series can be compared based on their properties rather than comparing them directly based on their shape. So, Feature Based similarity measures compare two time series by computing a feature set for each time series that reflects their properties¹. For example, DFT and DWT coefficients can be used to compare the similarity between time series [Shatkay and Zdonik1996].

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 Li2009].

2.4 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 techniques have been used: for instance, in [Agrawal *et al.* 1993], time series has 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 Fu1999a]. Other representation approaches used for query by content are PLA, PAA, APCA [Keogh *et al.*], and SAX [Lin *et al.* 2007]. In [Lin *et al.* 2007] it has shown that SAX outperforms other representations for query by content applications.

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.*]. Associating each class

¹We will use this type of distance later when analyzing the locomotion in the Manual Wheelchair.

2.4 Datamining task on time series Chapter 2. Knowledge discovery on time series

with a motif can speed-up the classification of time series; this idea is used by the shapelet transform algorithm² [Lines *et al.* 2012].

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 for 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:

- **Time searcher:** it is a query by content summarization tool. Here, a user specifies a set of constraints (intervals) graphically within time series data points should belong. Those constraints are called time series boxes [Hochheiser and Shneiderman 2011].
- **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

²We will use this type of distance later when analyzing the locomotion in the Manual Wheelchair.

daily data patterns and visualized the average patterns as graphs and the corresponding days on a calendar [Van Wijk and Van Selow1999].

- 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.*].

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. 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 guest 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 string 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 frequency of occurrence of each word in this one. 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 the latter. 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 Malinchik2013], Bag of SFA Symbols(BOSS) [Schäfer2015], DTW Features[Kate2016].
- **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 Pazzani2001b], Weighted Dynamic Time Warping [Jeong *et al.*], Complexity-Invariant Distance [Batista *et al.*2011]), on their properties, on their structures or on their symbolic representation (Time Warp Edit Distance[Marteau2008], 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 Keogh2013]
- **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 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 Runger2016].

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 Sikka2012]. "Homogeneous" means that the intra-group variance is small and "Well separated" means that the inter-groups variance is 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 have gathered 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, Density-based, Grid-based and Model-based clustering.

- **Distance-based** clustering which comes in two sub-categories:
 - **Partitioning clustering algorithms** typical 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 as follows: given a time series that will be considered as the center of the cluster, we gathered in the same 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. Doing so the algorithm splits the space into more or less dense regions: then dense small regions can be merged into more significant regions: This algorithm allows to identify a cluster of arbitrary shapes [Kriegel *et al.* 2011].
- **Grid-based clustering** divides the data space into 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 we estimate the parameters of the model that maximize the likelihood of the model to the data using the expectation-maximization algorithm (EM). On this basis we may estimate the generative probabilities that will be used to construct the generative model [Merugu and Ghosh2003].

- **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 consist of clustering simultaneously 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 technique 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) simultaneously [Hofmann2017].
- **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 that has a lower dimension than X , which will reduce the number of dimensions [Wang and Zhang2013].
- **Spectral clustering:** the idea here is to cluster time series or data object based on the spectrum of their similarity matrix. The spectrum being used here for dimensionality reduction[Filippone *et al.* 2008].

We are mainly interested in time series clustering, tables 2.1, 2.2 and 2.3 present some works from the literature on time series clustering and highlight the interaction between clustering on one hand and the representations of time series and similarity measures used on the other hand. Detailed informations is presented in [Rani and Sikka2012].

Table 2.1: 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 Shumway	Short time series (STS) distance Kullback–Leibler discrimination information measure	Modified Fuzzy C-Means Agglomerative Hierarchical	DNA microarray Earthquakes and mining explosions
Vit Niennattrakul Pooya Sobhe Bidari	Dynamic Time Warping Pearson Correlation	K-Means, K-Medoids K-Means, Fuzzy C-Means	Multimedia time series 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+ MFP(Most Frequent Pattern)	Stock and inventory data
Mengfan Zhang	CVT(Computational Verb Theory)	K-Means	Stock market data
S.R.Nanda Jianfei Wu	Euclidean N/A	K-Means K-Means	Portfolio management Stock data

2.4 Datamining task on time series Chapter 2. Knowledge discovery on time series

Table 2.2: 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 Verdoollaeghe	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 2.3: 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

2.5 CONCLUSIONS

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 ask 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.

Part II

Contributions

Chapter 3

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 paper, we demonstrate that to compress time series using Piecewise Aggregate Approximation (PAA) is a simple strategy that greatly increases the quality of the alignment with DTW this is particularly true for synthetic data sets.*

3.1 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 [Lin *et al.* 2003, Sun *et al.* 2014, Lkhagva *et al.*], in order

- To construct symbolic representations of time series; [Camerra *et al.*, Ulanova *et al.*].
- To construct an index for time series; [Zhao and Itti, Keogh and Pazzanib, Kate]. 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.

- To classify time series.

3.1.1 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 the ED [Chen *et al.* 2015]. An exhaustive comparison of time series algorithms [Bagnall *et al.*] shows that DTW is among the efficient techniques to be used. However, DTW has two major drawbacks: the comparison of two time series under DTW is time-consuming [Rakthanmanon *et al.* 2012a] and DTW sometimes produces pathological alignments [Keogh and Pazzani 2001a]. 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.*, Sakoe and Chiba 1978c, Jeong *et al.*, Salvador and Chan 2007]. The main idea here 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 datapoints that produce pathological alignment during the comparison of two time series [Longin *et al.*, Itakura, 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 2001a].

Another simple but yet interesting way to capture local properties of time series is to consider 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 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.

3.1.2 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. 3.1). Our idea is that a number of segments for PAA will be considered as good if it allows obtaining a compact representation of the time series, and also if it preserves the quality of the alignment of time series. So when considering classification task, one of the best classification 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.*b].

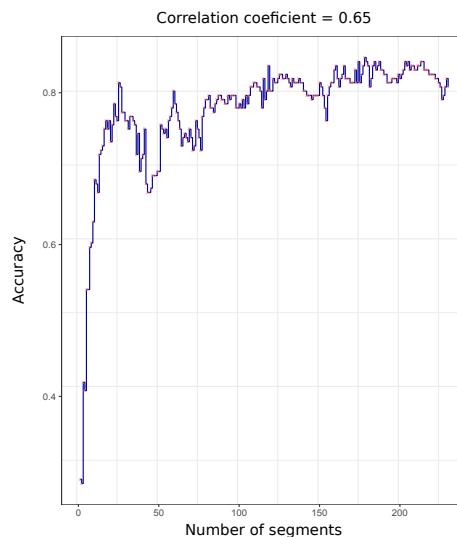


Figure 3.1: Relation between Accuracy and the number of segment on FISH dataset. The accuracy is computed from the algorithm one nearest neighbor (1NN) associated with PDTW. When the number of segments considered is very small, there is a loss of information and the accuracy is reduced. However, considering all the points in the time series, we also do not obtain maximum accuracy due to the presence of noise or singularities [Keogh and Pazzani2001a] in the data.

3.1.3 Summary of Contributions

In this paper,

- 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.
- We make our source code and all our results available to allow the reproducibility of our experiments.

The rest of the paper is organized as follow: in Section 3.2 we recall the definitions and background; Section 3.3 explains our approach; Section 3.4 presents experimental results and comparisons to others methods; Section 3.5 offers conclusions and venues for future work.

3.2 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:

$$\bar{X}_i = \frac{1}{l} \sum_{j=0}^{l-1} x_{i+j} \quad (3.1)$$

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

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

$$(X, N) \mapsto PAA(X, N) = \begin{cases} \bar{X}_1, \dots, \bar{X}_N & \text{if } N < |X| \\ X & \text{otherwise} \end{cases} \quad (3.2)$$

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

3.2.1 Dynamic Time Warping algorithm.

DTW [Sakoe and Chiba1978c] 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:

$$\begin{aligned}\gamma(i, j) = d(x_i, y_j) + \min\{\gamma(i - 1, j - 1), \\ \gamma(i - 1, j), \gamma(i, j - 1)\},\end{aligned}\quad (3.3)$$

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] 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:

$$PDTW(X, Y, N) = DTW(PAA(X, N), PAA(Y, N)). \quad (3.4)$$

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.*] proposes the Iterative Deepening Dynamic Time Warping Algorithm (IDDTW).

3.2.2 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

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 3.2): 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.

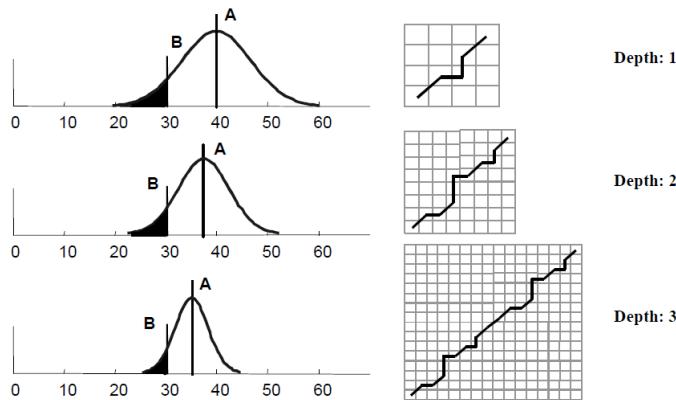


Figure 3.2: IDDTW operating principle. Depth represents approximation levels, A represents approximate distance and B is best_so_far [Chu *et al.*].

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 Pazzani2001a].

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

3.3 GRASP based heuristic

3.3.1 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 the shape of the latter. For instance, in the ECG dataset, each time series traces the electrical activity recorded during one heartbeat. The two classes are a normal heartbeat and a Myocardial Infarction.

Time series classification is a classic problem with time series 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 (ϵ), or its accuracy ($a = 1 - \epsilon$). When considering classification task, one of the best classification 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.*b].

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.

3.3.2 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. :

$$1NNDTW : D \rightarrow [0, 1] \quad (3.5)$$

$$d_i \mapsto 1NNDTW(d_i) \quad (3.6)$$

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

Definition 7. :

$$1NNPDTW : D \times \{1 \dots n\} \rightarrow [0, 1] \quad (3.7)$$

$$\begin{aligned} (d_i, N) &\mapsto 1NNPDTW(d_i, N) \\ &= 1NNDTW \circ PAAsset(d_i, N) \end{aligned}$$

$1NNPDTW(d_i, N)$ 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 , whose time series have a length n , we look for the number of segments $N \in \{1 \dots n\}$ such that

$$1NNPDTW(d_i, N) = \min_{1 \leq \alpha \leq n} \{1NNPDTW(d_i, \alpha)\}. \quad (3.8)$$

3.3.3 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 :

$$O(|d|^2 \times \sum_{N \in C} N^2), |C| = n, \quad (3.9)$$

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

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

3.3.4 Greedy Randomized Adaptive Search Procedures

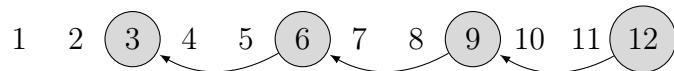
The Greedy Randomized Adaptive Search Procedures (GRASP) is a multi-start, or iterative metaheuristic proposed by Feo and Resende (1995) [Feo and Resende1995], in which each iteration consists of two phases: firstly a new solution is constructed by a greedy randomized procedure and then is improved using a local search procedure.

The greediness criterion establishes that elements with the best quality are added to a restricted candidate list and chosen at random when building up the solution. The candidates obtained by greedy algorithms are not necessarily optimal. So, those candidates 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.

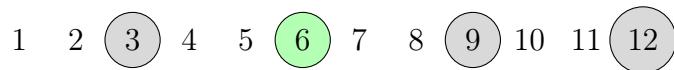
3.3.5 Parameter free heuristic.

The idea of our heuristic is the following:

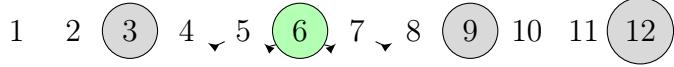
1. We choose N_c candidates distributed in the space of possible values to ensure that we are going to have small, medium and large values as candidates. The candidates values are: $n, n - \left\lfloor \frac{n}{N_c} \right\rfloor, n - 2 \times \left\lfloor \frac{n}{N_c} \right\rfloor, \dots, n - N_c \times \left\lfloor \frac{n}{N_c} \right\rfloor$. 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.



2. 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, we may suppose that we get the minimal value with the candidate 6 : it is thus the best candidate at this step.



3. 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 N_c candidates to find the best one. Then, we test $\frac{2n}{N_c}$ other candidates to find the local minimum. We finally perform $nb(N_c) = N_c + \frac{2n}{N_c}$ 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 $N_c = \sqrt{2n}$ and its second derivative is positive; so the minimal number of tests is obtained when the number of candidates is : $N_c = \sqrt{2n}$. At each iteration, the heuristic tests $nb(\sqrt{2n}) = 8\sqrt{n}$ number of segments. As we have k iterations the number of candidates tested is: $|C| = \sum_{i=0}^{k-1} 8\sqrt{n-i}$. 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

$$O(|d|^2 \times \sum_{N \in C} N^2), |C| = \sum_{i=0}^{k-1} 8\sqrt{n-i}. \quad (3.10)$$

where $|d|^2$ comes from 1NN algorithm and $\sum_{N \in C} N^2$ comes from PDTW with N being one value of the number of segments, and C being the set of values for the number of segments.

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 $FDTW$.

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_candidats  $\leftarrow |\text{tab\_N\_possible\_candidates}|$ 
7     for i  $\in \{1, 2, \dots, \text{nb\_candidats}\} do
8       if tab_N_possible_candidates[i]  $\neq 0$  then
9         tab_N_candidates[j]  $\leftarrow \text{tab\_N\_possible\_candidates}[i]$ 
10        j  $\leftarrow j + 1$ 
11    mat_r  $\leftarrow \text{1NNPDTW}(\text{training\_set}, \text{tab\_N\_candidates})$ 
12    /* 1NNPDTW return a matrix of couple (N, error) */
13    for i  $\in \text{tab\_N\_candidates} do
14      tab_N[i]  $\leftarrow 0$ 
15      min  $\leftarrow \text{minimun}(\text{mat}_r)$ 
16      /* minimum return the couple (N, error) with the minimum
17         error */
18      result $[(i + 1)] \leftarrow \text{localMinimun}(\text{min.N}, \text{min.error}, \text{training\_set}, \text{tab\_N})$ 
19    m  $\leftarrow \text{minimun}(\text{result})$ 
20  return m$$$ 
```

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 in an interval with respect to the optimal value [Ibarra and Kim].

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:

$acc_{max(d_i)} = 1 - \min_{1 \leq \alpha \leq n} \{1NNPDTW(d_i, \alpha)\}$ 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.*]. 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, $c_i \in \mathbb{N}^*$,

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

Proposition 1 shows that when 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

$$\frac{1}{\beta} \times acc_{max} \leq acc_{FDTW} \Leftrightarrow \frac{acc_{max}}{acc_{FDTW}} \leq \beta \quad (3.11)$$

$$\text{However, } \frac{acc_{max}}{acc_{FDTW}} \leq \frac{1}{acc_{FDTW}} \text{ because } acc_{max} \leq 1 \quad (3.12)$$

$$\text{And, } \frac{1}{acc_{FDTW}} \leq \frac{1}{acc_{DTW}} \text{ because } acc_{DTW} \leq acc_{FDTW} \quad (3.13)$$

$$\text{So, } \frac{1}{acc_{DTW}} \leq c_i \text{ because } \frac{1}{c_i} \leq acc_{DTW} \text{ by hypothesis} \quad (3.14)$$

3.4 Experiment and results

Throughout the experiments described in this paper, 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 have been compared to the IDDTW algorithm in terms of :

- heuristic execution speed;
- time series compression ratio;
- classification error associated with the number of segments found by the heuristic.

3.4.1 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 the compression ratios. A segment number N_1 will be better than a segment number N_2 if it makes it possible 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 3.1. 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 us 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 3.3).

Classification

PAA is used by ShapeDTW [Zhao and Itti] and DTW_F [Kate] 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. Here, 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 based on the 3 fold cross validation

3.4 Experiment and results

Chapter 3. Preprocessing of time series

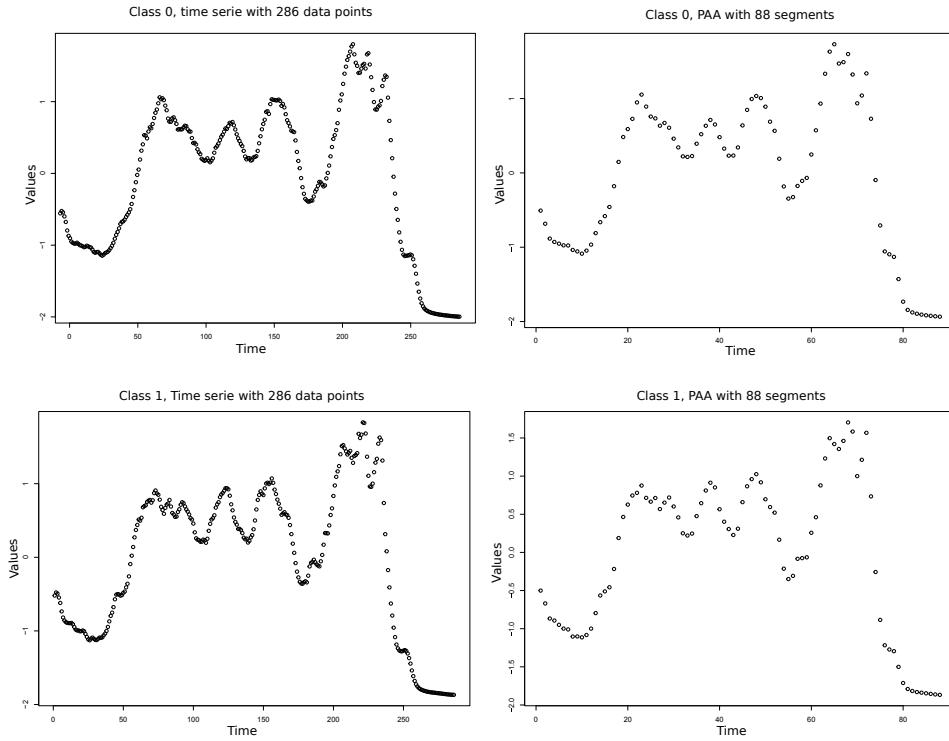


Figure 3.3: Visual comparison of two time series from the two classes of the coffee dataset. Left: the original time series, right: representation using PAA with 88 segments

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 3.1).

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 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.

N°	Datasets (training set)	DTW	IDDTW	N	FDTW	N
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
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

continued on next page

N°	Datasets (training set)	DTW	IDDTW	N	FDTW	N
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 3.1: 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.

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 3.2).

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°	[Bagnall et al.a]			Our experiments					
	1NN	1NN	1-NN	Brute	$N(\ell)$	IDDTW	$N(\ell)$	FDTW	$N(\ell)$
	Eucli	DTW	DTW (r)	force					
1	0.369	0.310	0.242 (6)	0.262	251(1)	0.268	256(1)	0.268	258(1)
2	0.389	0.396	0.391 (3)	0.379	162(1)	0.432	128(1)	0.414	143(1)
3	0.333	0.367	0.333 (0)	0.233	286(2)	0.3	8(59)	0.367	94(5)
4	0.267	0.267	0.233 (1)	0.183	52(11)	0.367	8(72)	0.367	377(1)

continued on next page

continued from previous page

N°	[Bagnall et al.a]			Our experiments					
	1NN	1NN	1-NN	Brute	$N(\ell)$	IDDTW	$N(\ell)$	FDTW	$N(\ell)$
	Eucli	DTW	DTW (r)	force	search				
	dean distance								
5	0.148	0.003	0.004 (11)	0	118(1)	0.003	128(1)	0.001	128(1)
6	0.000	0.000	0.000 (0)	0	13(22)	0	64(4)	0.000	286(1)
7	0.423	0.246	0.228 (10)	0.228	142(2)	0.256	256(1)	0.269	84(4)
8	0.433	0.256	0.238 (17)	0.231	271(1)	0.241	256(1)	0.244	294(1)
9	0.413	0.246	0.254 (5)	0.221	249(1)	0.223	256(1)	0.233	276(1)
10	0.218	0.208	0.228 (1)	0.2	78(1)	0.225	16(5)	0.223	80(1)
11	0.273	0.29	0.272 (0)	0.263	35(2)	0.288	16(5)	0.278	80(1)
12	0.326	0.258	0.258 (22)	0.198	176(2)	0.258	512(1)	0.276	101(5)
13	0.120	0.230	0.120 (0)	0.13	38(3)	0.19	8(12)	0.180	11(9)
14	0.203	0.232	0.203 (0)	0.117	11(12)	0.289	32(4)	0.117	11(12)
15	0.286	0.192	0.192 (3)	0.091	79(2)	0.194	128(1)	0.148	99(1)
16	0.216	0.170	0.114 (2)	0.08	107(3)	0.352	32(11)	0.102	140(3)
17	0.217	0.177	0.154(4)	0.154	149(3)	0.257	16(29)	0.177	27(17)
18	0.087	0.093	0.087 (0)	0.02	38(4)	0.073	32(5)	0.020	38(4)
19	0.4	0.533	0.400 (0)	0.343	21(20)	0.026	32(13)	0.432	32(13)
20	0.630	0.623	0.588 (2)	0.549	328(3)	0.588	64(17)	0.594	948(1)
21	0.658	0.616	0.613 (14)	0.578	1770(1)	0.627	64(29)	0.622	171(11)
22	0.045	0.050	0.045 (0)	0.033	20(1)	0.043	8(3)	0.033	24(1)
23	0.246	0.131	0.131 (6)	0.082	70(9)	0.246	16(40)	0.180	524(1)
24	0.425	0.274	0.288 (5)	0.192	150(2)	0.301	64(5)	0.301	51(6)
25	0.316	0.263	0.253 (20)	0.255	95(1)	0.271	64(2)	0.280	34(3)
26	0.26	0.25	0.253 (5)	0.233	27(2)	0.283	2(40)	0.283	2(40)
27	0.439	0.416	0.419 (2)	0.398	27(2)	0.414	32(2)	0.416	80(1)
28	0.121	0.165	0.134 (1)	0.135	14(6)	0.197	16(5)	0.165	84(31)
29	0.133	0.167	0.133 (0)	0.1	191(3)	0.167	32(18)	0.100	234(2)
30	0.479	0.409	0.388 (7)	0.364	31(14)	0.372	32(13)	0.409	35(12)
31	0.038	0	0.000 (6)	0	35(4)	0	128(1)	0	135(1)
32	0.292	0.263	0.263 (6)	0.24	75(1)	0.288	4(20)	0.288	4(20)
33	0.461	0.35	0.300 (3)	0.083	54(9)	0.239	64(7)	0.122	48(10)
34	0.305	0.275	0.305 (0)	0.206	37(2)	0.208	16(4)	0.304	26(3)
35	0.141	0.169	0.141 (0)	0.14	5(13)	0.197	16(4)	0.178	45(1)
36	0.211	0.208	0.154 (2)	0.165	59(2)	0.195	64(2)	0.208	55(2)
37	0.100	0.050	0.062 (8)	0.044	376(1)	0.059	32(12)	0.060	34(12)

continued on next page

continued from previous page

N°	[Bagnall <i>et al.</i> a]			Our experiments					
	1NN	1NN	1-NN	Brute	$N(\ell)$	IDDTW	$N(\ell)$	FDTW	$N(\ell)$
	Eucli	DTW	DTW (r)	force					
38	0.120	0.007	0.017 (6)	0.007	60(1)	0.437	2(30)	0.007	60(1)
39	0.240	0.000	0.010 (3)	<i>0</i>	47(6)	0	64(4)	<i>0</i>	275(1)
40	0.090	0.000	0.002 (4)	<i>0</i>	21(6)	0	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)	0.008	95(2)
43	0.389	0.426	0.389 (0)	0.204	3(78)	0.463	20(11)	0.37	128(1)
44	0.382	0.351	0.252 (8)	0.337	133(2)	0.365	64(4)	0.343	135(2)
45	0.170	0.164	0.155 (2)	0.149	117(4)	0.158	128(3)	0.154	384(1)
\bar{X}	0.268	0.227	0.242	0.175		0.232		0.214	

Table 3.2: 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 ℓ is the number of data points in a segment ($l = \lfloor \frac{n}{N} \rfloor$).

The 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 show that FDTW managed 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 :

As already suggested by the time complexity of 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 3.4). Actually, the number of candidates to be tested ranges 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

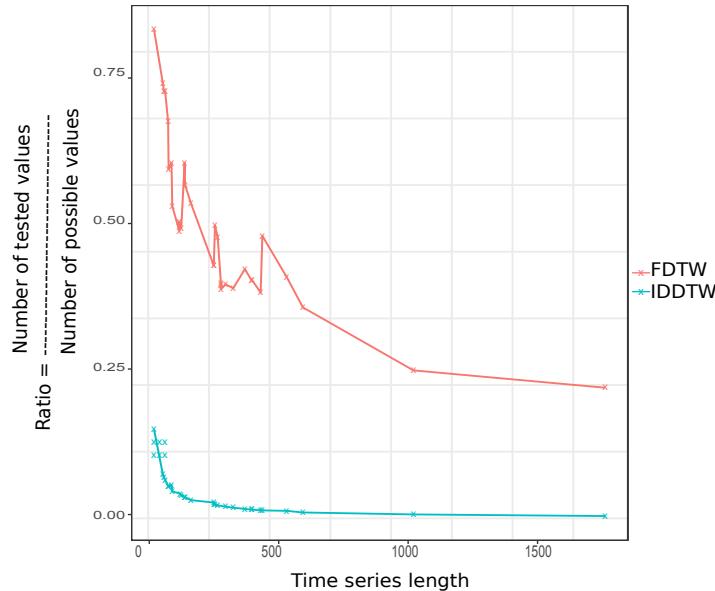


Figure 3.4: Comparison of the number of tested values of the parameter number of segments with the FDTW and IDDTW. x-axis datasets are sorted according to the length of the time series.

(Figure 3.5). The increase of Brute-force search execution time is faster than that of FDTW and IDDTW. This is observable from the datasets Lightning-2 whose time series have a length equal to 637 data points. 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.

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.b*] 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. 3.6). 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, Fig. 3.6) because the data of the training set and the testing set are generated by the same models.

However, to evaluate the significance of the difference between the classification algorithms on 84 datasets, we used the Wilcoxon signed rank test with continuity correction, which has more statistical power. The results of these experiments show that despite data compression,

- FDTW have better performance than Naive Bayes (NB), C45, logistic regres-

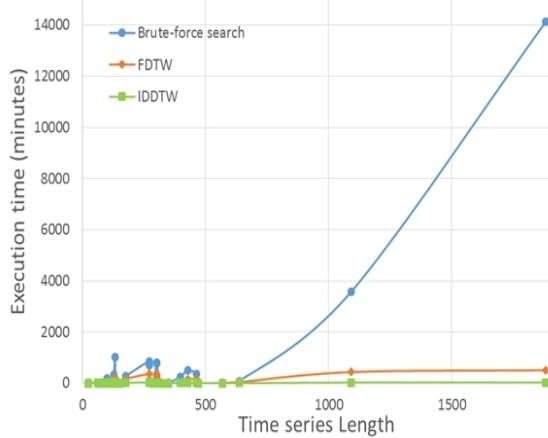


Figure 3.5: Comparison of the execution time of the Brute-force search algorithm, FDTW and IDDTW.

sion (Logistic), BN;

- FDTW has similar performance to that of 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 perform 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. 3.7). 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.*] 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.*].

3.5 Conclusion and perspective

This paper 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 and was rank 3/37 behind two ensemble classification algorithms COTE

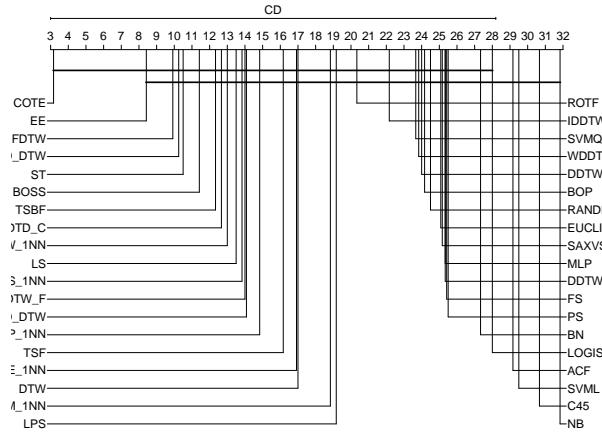


Figure 3.6: Critical difference diagram for FDTW and 36 other classification algorithms on 6 simulated datasets. FDTW is ranked 3rd / 37 algorithms

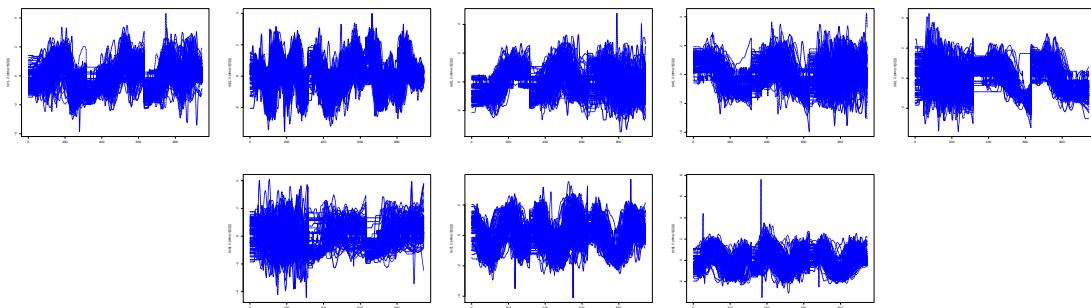


Figure 3.7: Eight types of time series corresponding to the vocabulary of 8 gestures.

and EE. This algorithm 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 paper could be modeled as a multi-objective optimization problem where one objective function would be compression and the other the classification of time series.

Chapter 4

Frobenius correlation based u-shapelets discovery for time series clustering

Abstract : *An u-shapelet is a sub-sequence of a time series used for clustering a time series dataset. The purpose of this paper is to discover u-shapelets on uncertain time series. To achieve this goal, we propose a dissimilarity score 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 is 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 the clustering with respect to the Rand Index. This work defines a novel framework for the clustering of uncertain time series.*

4.1 Introduction

Uncertainty in time series comes from several sources. For instance, to protect privacy, privacy-preserving transformation [Papadimitriou *et al.*, Aggarwal] deliberately introduce uncertainty to the confidential data before further processing. In a sensor network, sensor readings are imprecise because of the presence of noise generated either by the equipment itself or other external influences [Cheng *et al.*]. Ignoring the uncertainty of the data 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.*, Hwang *et al.*, Rehfeld and Kurths, Orang and Shiria,

Wang *et al.*, Orang and Shirib], or its impact is reduced by transformations performed on the data [Orang and Shiric]. This latter strategy is used natively by the u-shapelet algorithm.

4.1.1 U-shapelets algorithm for clustering Uncertain Time Series

U-shapelets clustering is a framework introduced by[Zakaria *et al.*] who suggested clustering time series from the local properties of their sub-sequences rather than using their global features of the time series [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 for 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.*]. 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. Despite this advantage, it is highly desirable to take into account the adverse impact of uncertainty during u-shapelet discovery.

4.1.2 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 to cluster uncertain time series with u-shapelets algorithm. Our work leverages the observation that the use of a dissimilarity function robust to uncertainty could improve the quality of the u-shapelets discovered and thus improve the clustering quality of uncertain time series.

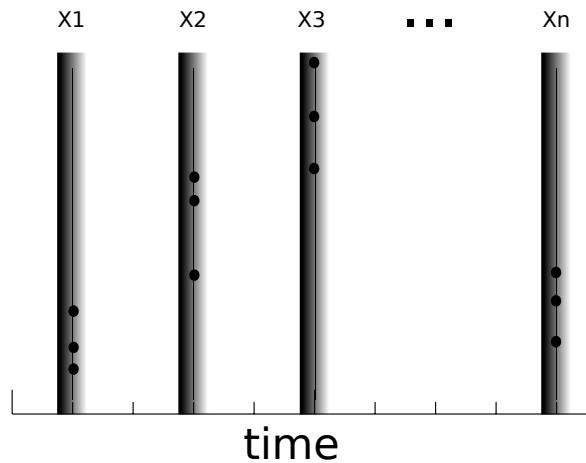


Figure 4.1: Multiset-based model of uncertain time series

4.1.3 Summary of contributions

- We review 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 uShapelet discovery (FOTS), a new dissimilarity score based on local correlation, which has interesting properties useful 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[?].

4.2 Definitions and Background

4.2.1 Related work

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 (Fig. 4.1) and N_i denotes the number of observed values at timestamp i .

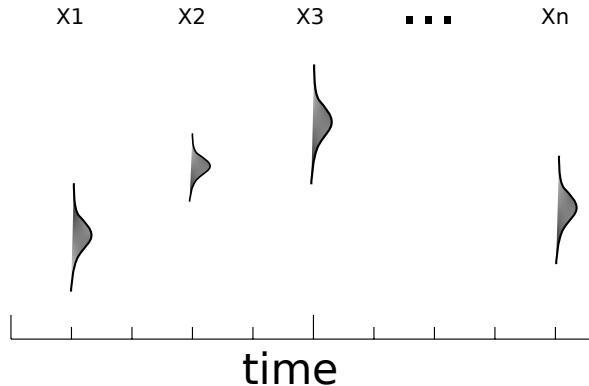


Figure 4.2: PDF-based model of uncertain time series

In **PDF-based model**, each element X_i , ($1 \leq i \leq n$) of UTS $X = < X_1, \dots, X_n >$ is represented as a random variable $X_i = x_i + X_{e_i}$, where x_i is the exact value that is unknown and X_{e_i} is a random variable representing the error (Fig. 4.2). It is this model that we consider this work.

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

- 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.*]. [Orang and Shiric] demonstrates that the performances of Uncertain similarity measures associated with pre-processing of data are higher than those of traditional similarity measurements.

4.2.2 Review of u-shapelets

Definition 1. 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 2. 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 3. *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 opens 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 Shiria]. 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 4. *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 sub-sequences. The strategy used by [Ulanova *et al.*] 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 made efficiently thanks to a hash algorithm.

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

$$gap = \mu_B - \sigma_B - (\mu_A - \sigma_A), \quad (4.1)$$

where μ_A (resp. μ_B) denotes $\text{mean}(sdist(S, D_A))$ (resp. $\text{mean}(sdist(S, D_B))$), and σ_A (resp. σ_B) represents standard deviation of $sdist(S, D_A)$ (resp. standard deviation of $sdist(S, D_B)$). If D_A or D_B consists of only one element (or of 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.

4.2.3 Review 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] 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 $r(X_i), r(Y_i)$: $dist(X_i, Y_i) = |r(X_i) - r(Y_i)|$. 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 :

$$\varphi(|X_i - Y_i|) = Pr(dist(0, |X_i - Y_i|) = 0). \quad (4.2)$$

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

$$dust(X_i, Y_i) = \sqrt{-\log(\varphi(|X_i - Y_i|)) + \log(\varphi(0))}. \quad (4.3)$$

The distance between uncertain time series $X = \langle X_1, \dots, X_n \rangle$ and $Y = \langle Y_1, \dots, Y_n \rangle$ in *DUST* is then defined as follows:

$$DUST(X, Y) = \sqrt{\sum_{i=1}^n dust(X_i, Y_i)^2}. \quad (4.4)$$

The problem with the deterministic uncertain distances like *DUST* 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.*] 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 ε is equal to the number of distances between X and Y, which are less than ε , over the possible number of distances:

$$Pr(distance(X, Y) \leq \varepsilon) = \frac{|\{d \in dists(X, Y) | d \leq \varepsilon\}|}{|dists(X, Y)|} \quad (4.5)$$

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

PROUD [Yeh *et al.*] Let $X = \langle X_1, \dots, X_n \rangle$ and $Y = \langle Y_1, \dots, Y_n \rangle$ be two UTS each 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 [?], the cumulative distribution of the distances approaches asymptotically a normal distribution:

$$d(X, Y) \propto N\left(\sum_i E[(X_i - Y_i)^2], \sum_i Var[(X_i - Y_i)^2]\right) \quad (4.6)$$

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:

$$Pr(d(X, Y)_{norm} \leq \epsilon). \quad (4.7)$$

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 it follows **asymptotically** a normal distribution. Thus, this probability will be all the more accurate as the compared time series are long (more than 30 data points).

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

Definition 5. *The normal form of a standard time series $X = \langle X_1, \dots, X_n \rangle$ is defined as $\hat{X} = \langle \hat{X}_1, \dots, \hat{X}_n \rangle$ in which for each timestamp i ($1 \leq i \leq n$), we have:*

$$\hat{X}_i = \frac{X_i - \bar{X}}{S_X}, \bar{X} = \sum_{i=1}^n \frac{X_i}{n}, S_X = \sqrt{\sum_{i=1}^n \frac{(X_i - \bar{X})^2}{(n-1)}}. \quad (4.8)$$

PROUDS defines the distance between two normalized time series $\hat{X} = \langle \hat{X}_1, \dots, \hat{X}_n \rangle$ and $\hat{Y} = \langle \hat{Y}_1, \dots, \hat{Y}_n \rangle$ (Definition 5) as follows:

$$Eucl(\hat{X}, \hat{Y}) = 2(n-1) + 2 \sum_{i=1}^n \hat{X}_i \hat{Y}_i \quad (4.9)$$

For the same reasons as PROUD, PROUDS is not suitable for short time series comparison. 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 Shirib] : 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 6. (*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:

$$\text{Corr}(X, Y) = \sum_{i=1}^n \hat{X}_i \hat{Y}_i / (n - 1), \quad (4.10)$$

where \hat{X}_i and \hat{Y}_i are normal forms of X_i and Y_i (Definition 5), 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 thoses 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.*] 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 7. (*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.4.11) :*

$$\hat{\Gamma}_t(X, w, m) = \sum_{\tau=t-m+1}^t x_{\tau,w} \otimes x_{\tau,w}. \quad (4.11)$$

where $x_{\tau,w}$ is a sub-sequence of the time series of length w and started at τ , $x \otimes y = xy^T$ is the outer product of x and y . The sample set of m windows is centered around time t . We typically fix the number of windows 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 autocorrelations 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 8 formalizes this notion:

Definition 8. (*LoCo score*). *Given two series X and Y , their LoCo score is defined by*

$$\ell_t(X, Y) = \frac{1}{2}(\|U_X^T u_Y\| + \|U_Y^T u_X\|) \quad (4.12)$$

Where U_X and U_Y are the k first eigenvector matrices of the local autocorrelation $\hat{\Gamma}_t(X)$ and $\hat{\Gamma}_t(Y)$ respectively, and u_X and 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 α 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 the $\cos(\alpha)$ will be 1. 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 achieve with z-normalization. LoCo similarity function has many interesting properties and does not require to:

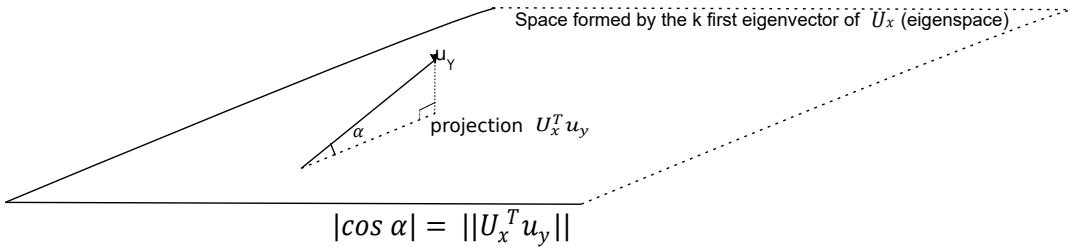


Figure 4.3: Geometric representation of loco similarity.

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

It 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.

4.3 Our Approach

4.3.1 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 α -value as a dissimilarity function but this approach does not work so well; the sine and the angle are not discriminant enough for eigenvector comparison for clustering purpose. We thus propose the following dissimilarity measure (Definition 9).

Definition 9. (*FOTS : Frobenius cOrrelation for uncertain Time series uShapelet discovery*) Given two series X and Y , their FOTS score is defined by

$$FOTS(X, Y) = \|U_X - U_Y\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^k (U_{Xj} - U_{Yj})^2} \quad (4.13)$$

where $\|\cdot\|_F$ is the Frobenius norm.

Because the 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 relationship** 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 timeseries**.

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 the uncertainty is confirmed by the following theorem:

Theorem 1. (*Hoffman Wielandt*) [Bhatia and Bhattacharyya] If X and $X + E$ are $n \times n$ symmetric matrices, then :

$$\sum_{i=1}^n (\lambda_i(X + E) - \lambda_i(X))^2 \leq \|E\|_F^2. \quad (4.14)$$

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.

4.3.2 Scalable u-shapelets Algorithm with FOTS score

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

Two main criteria make possible to evaluate the quality of a u-shapelet:

- It has to produce two r-balanced groups.
- It must build two well separated groups, i.e., groups whose gap is maximal.

The gap is, therefore, 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. Algorithms 4 and 5 present a more formal definition:

Definition 10. The sub-sequence FOTS dissimilarity $sd_f(S, T)$ between a time series T and a sub-sequence S is the minimum of the 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    $dis \leftarrow \{\}$ 
3    $s \leftarrow zNorm(s)$ 
4   forall  $i \in \{1, 2, \dots, |D|\}$  do
5      $ts \leftarrow D(i, :)$ 
6      $dis(i) \leftarrow sd_f(s, ts)$ 
7   return  $dis|s|$ 

```

4.4 Experimental Evaluation

4.4.1 Clustering with u-shapelets

The algorithm iteratively splits the data with each discovered u-shapelet: each u-shapelet splits the dataset into two groups D_A and D_B . The time series that belong to D_A are considered as members of the cluster form 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. This approach is a direct implementation of the u-shapelet definition

Choosing the length N of a uShapelet : 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 consider as length of u-shapelet the one allowing 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 consider 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 use $k = 4$ throughout all experiments. Indeed, key aperiodic trends

Algorithm 3: ComputeGap

Input: u-shapeletCandidate : s,
timeseries 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   gap  $\leftarrow$  0
4   for i  $\leftarrow$  lb to ub do
5     DA  $\leftarrow$  dis  $\leq$  dis(i), DB  $\leftarrow$  dis  $>$  dis(i)
6     if lb  $\leq$  |DA|  $\leq$  ub then
7       mA  $\leftarrow$  mean(DA), mB  $\leftarrow$  mean(DB)
8       sA  $\leftarrow$  std(DA), sB  $\leftarrow$  std(DB)
9       currGap  $\leftarrow$  mB - sB - (mA + sA)
10      if currGap  $>$  gap then
11        | gap  $\leftarrow$  currGap
12   return gap

```

are captured by one eigenvector, whereas key oscillatory trends manifest themselves in a pair of eigenvectors.

4.4.2 Evaluation Metric

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

Let Lc be the cluster labels returned by a clustering algorithm and Lt be the set of ground truth class labels. Let A be the number of time series that are placed in the same cluster in Lc and Lt, B be the number of time series in different clusters in Lc and Lt, C be the number of time series in the same cluster in Lc but not in Lt and D be the number of time series in different clusters in Lc but in same cluster in Lt. The Rand Index is equals to :

$$\text{Rand Index} = (A + B) / (A + B + C + D) \quad (4.15)$$

4.4.3 Comparison with u-shapelet

All measurements performed by a mechanical system have 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 Sitaram1997]. So, similarly to [Dallachiesa *et al.*], we tested our method on 17 real world datasets coming from UCR archive [Chen *et al.*2015] representing a wide range of application domains. The training and testing sets have been joined to obtained bigger datasets. Table 4.1 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 4.1: Datasets

Table 4.2 presents the comparison of the two algorithms.

4.4.4 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 compare the Rand Index obtained by FOTS-UShapelet and the one obtained by k-Shape and USLM on 5 datasets(Table 4.3). The results of k-Shape and USLM was previously reported in [Zhang *et al.*2016]. This comparison shows that in general, FOTS-UShapelet perform better than k-Shape and USLM.

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

Table 4.2: Comparison of the Rand Index of SUSH (RI_SUSh) and FOTS-SUSh (RI_FOTS). The best Rand Index is in bold

4.4.5 Discussion

The use of the FOTS score associated with the SUShapelet algorithm makes it possible 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 containing several small oscillations. Indeed, these oscillations are not captured by the Euclidean distance but are by the FOTS score whose calculation is based on the autocorrelation matrix. This observation is illustrated by the result obtained on 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 Chen1999]. The computation of FOTS score is then more expensive than that of ED (Fig. 4.4). However, its use remains relevant for u-shapelet research as they are often small.

Kappter 4. Uncertain time series u-shapelet discovery

Table 4.3: Comparison between k-Shape, USLM and FOTS-UShapelet

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

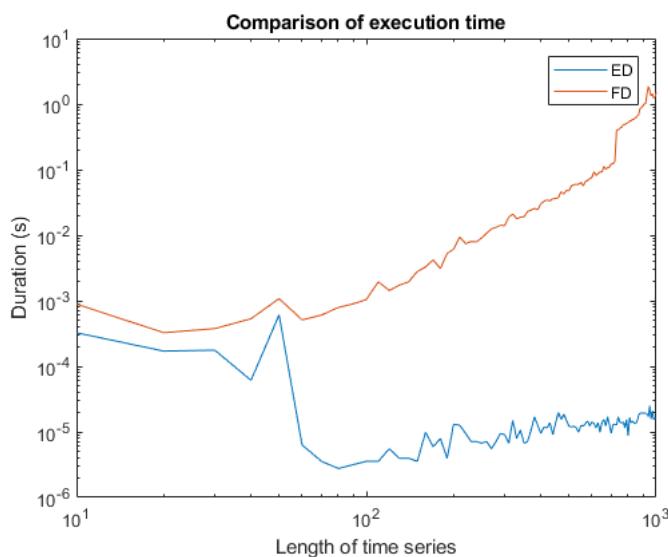


Figure 4.4: Comparison between the execution time of the Euclidean distance and that of the FOTS score as a function of the length of the time series

4.5 Conclusion and Future Work

The purpose of this work was to discover u-shapelets on uncertain time series. To answer this question, we have proposed a dissimilarity score (FOTS) adapted to the comparison of short time series, whose computation is based on the comparison of the eigenvector 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, and it 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 showed an improvement in the quality of clustering evaluated using the Rand Index. By combining 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, this

Chapter 4. Uncertain time series u-shapelet discovery Conclusion and Future Work

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

Chapter 5

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.*

5.1 Introduction

Generally, during his locomotion, the human being performs cyclic movements (eg 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. 5.1).

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. 5.2).

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

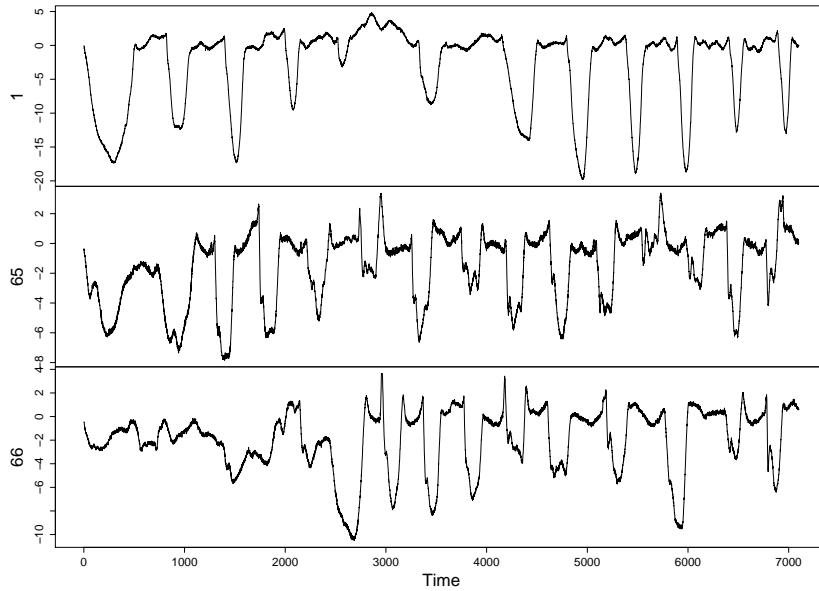


Figure 5.1: Cyclic time series form manual wheelchair locomotion

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. 5.3). 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 :

$$DR(\bar{C}, \bar{Q}) = \sqrt{\frac{n}{N} \sum_{i=1}^N (\bar{c}_i - \bar{q}_i)^2} \quad (5.1)$$

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 5.1).

[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. 5.4).

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 5.2).

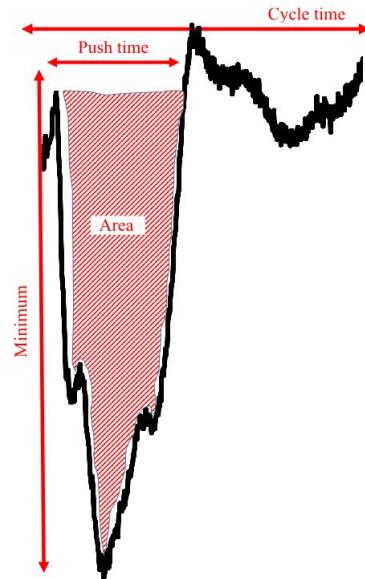


Figure 5.2: Properties of a cycle

$$MINDIST(\hat{Q}, \hat{C}) = \sqrt{\frac{n}{N} \sum_{i=1}^N (dist(\hat{q}_i, \hat{c}_i))^2} \quad (5.2)$$

\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 Kawagoe2006] proposed the ESAX model that considers three properties for each segment: its mean, its minimum and maximum (Fig. 5.5).

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. 5.6).

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

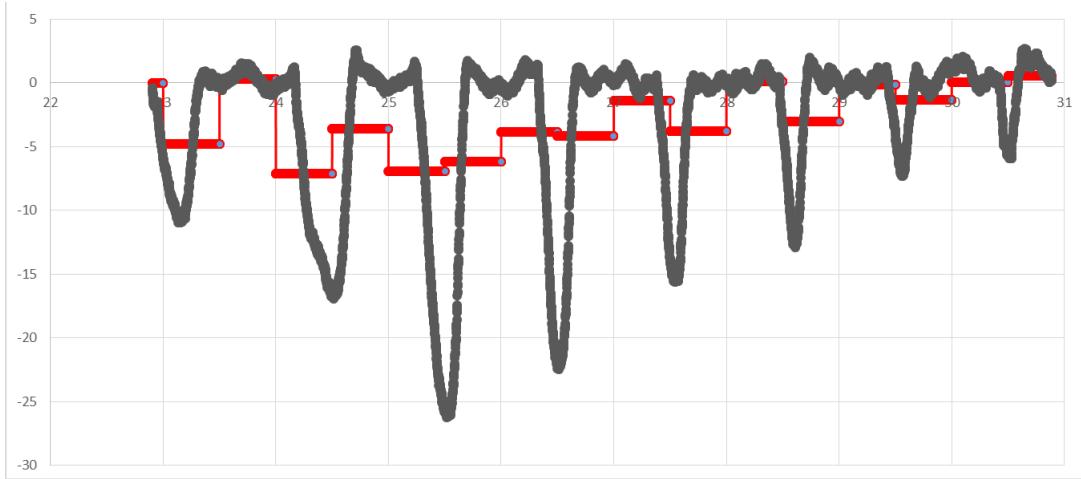


Figure 5.3: Piecewise aggregate approximation of a cyclic time series

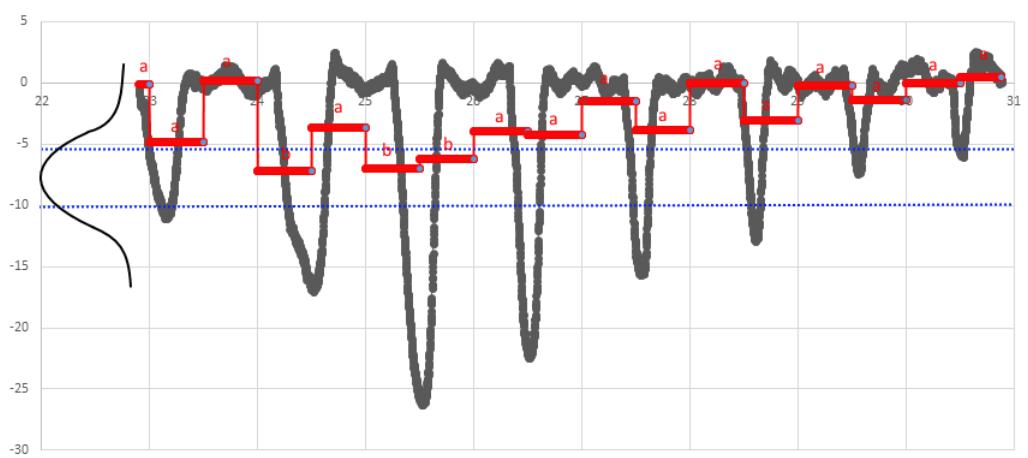


Figure 5.4: Symbolic Aggregate approXimation of a cyclic time series

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 Keogh2014], clustering, classification, indexing and also some interesting algorithms for text processing or the bio-informatics community [Aach and Church2001, Papapetrou *et al.*2011, Dietterich2002].

5.2 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.

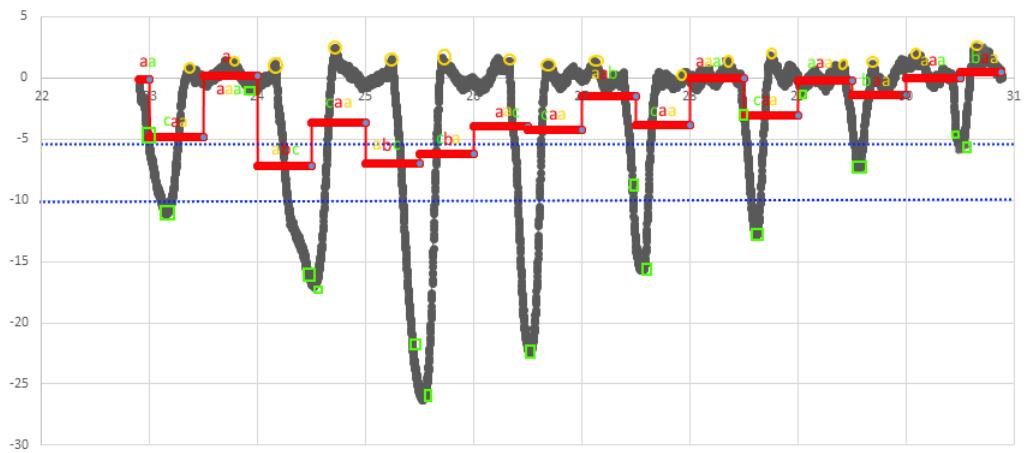


Figure 5.5: Extended Symbolic Aggregate approXimation of a cyclic time series

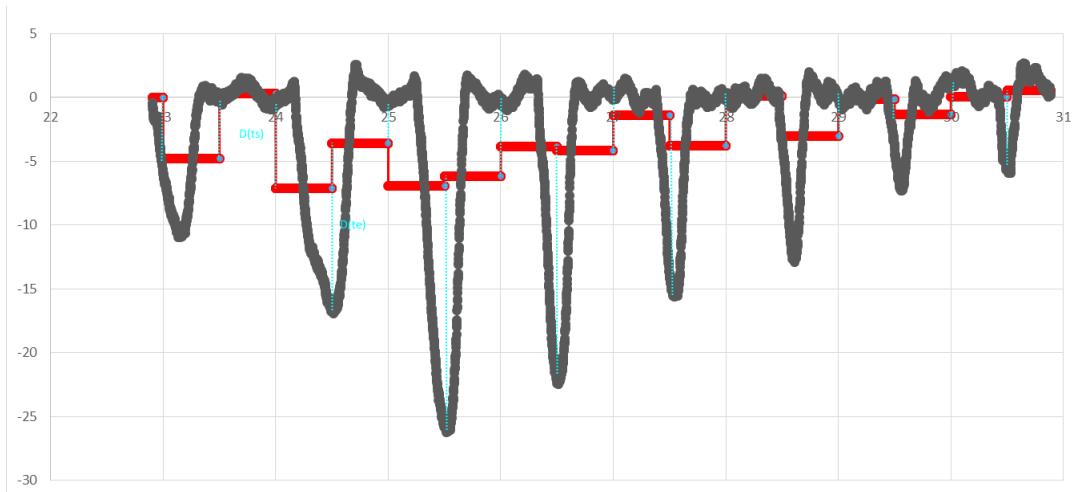


Figure 5.6: Trend Symbolic Aggregate approXimation of a cyclic time series

5.2.1 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. 5.8). 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. 5.9) presents the results obtained after segmentation of a cyclic time series.

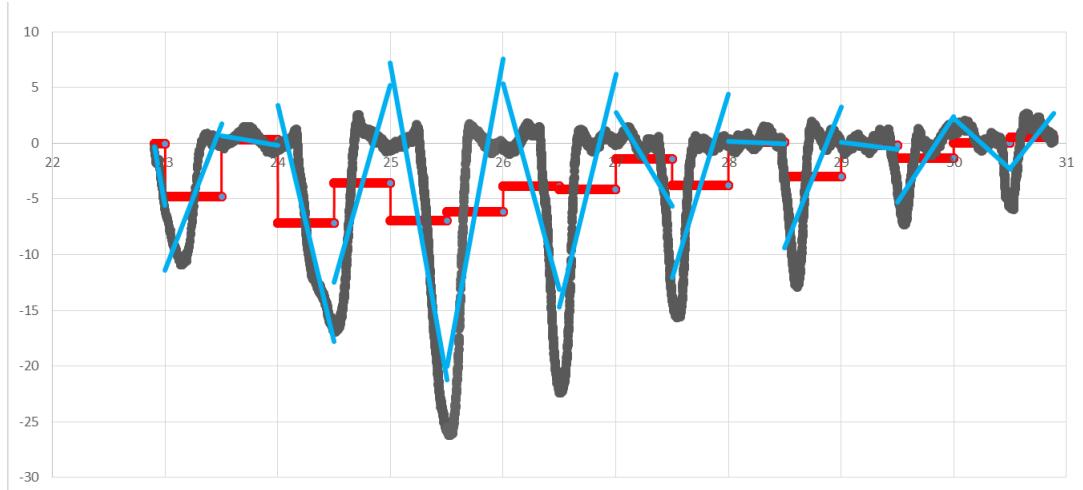


Figure 5.7: Properties of a cycle

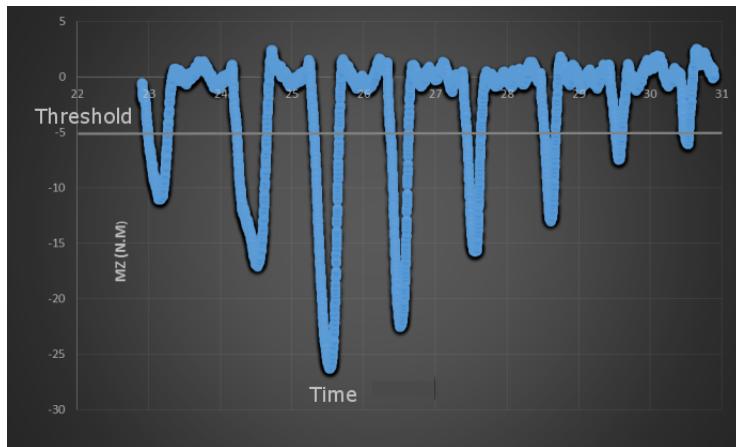


Figure 5.8: Threshold for the segmentation of cyclic time series

5.2.2 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. 5.10);
3. Segments are then gathered in clusters using a classification algorithm [Esling and Agon2012] and each cluster is named by a capital letter (Fig. 5.11);

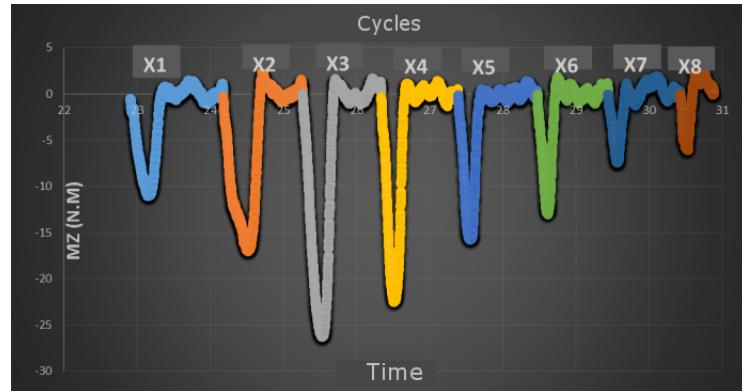


Figure 5.9: Segmentation

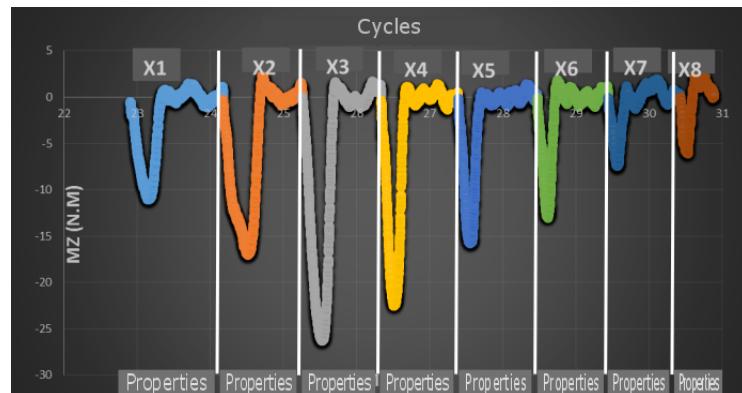


Figure 5.10: Some properties are computed on each cycle

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. 5.12);

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 5.12 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.

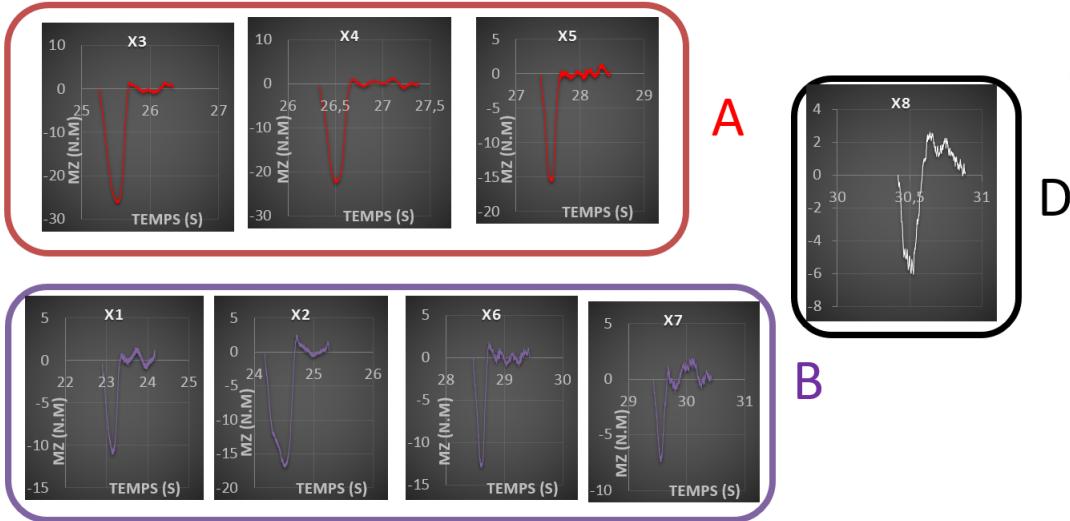


Figure 5.11: Classification of cycles based on properties

5.3 Application to manual wheelchair locomotion

This method has been applied to the axial moment (M_z) measured by both right and left rear wheels of an instrumented Manual WheelChair (MWC) during five there and back 10-m linear displacements between two cones performed by three handicapped subjects. We group propulsion cycles into 5 clusters (Table 5.1) and we obtained a symbolic representation for M_z (Table 5.2).

An important task of analyzing manual wheelchair locomotion is the comparison of its rolling movement. Experts from the fields seek to compare two movements simultaneously taking into account several criteria or properties. Applying SAX-P method to the time series of axial moments exerted by a wheelchair user on the right and left wheels greatly facilitates the comparison, the analysis and the interpretation of these time series:

- At first sight (Table 2), it immediately appears that during their second 10-m run the three subjects analyzed here did not exert the same number of pushes for moving a MWC on the same distance (S1: 7-8; S2: 12-13; S3: 5-7);
- It is also obvious that each subject did not exert the same number of pushes on both rear wheels. Moreover, although right and left pushes exerted by one subject globally belong to the same clusters, the total distance between all these pushes can be more (S2: 354) or less (S3: 44) high. Both these observations clearly demonstrate that the three subjects did not propel their MWC symmetrically during this particular exercise. The first results of the evaluation of SAX-P on a classification task are presented on the web page [?]

Cluster	A	B	C	D	E
Nb of cycles	18	36	59	18	104
Cycle time (s)	1.2	1.0	1.0	1.7	0.8
Push time (s)	0.6	0.3	0.4	1.0	0.3
Mz Min (Nm)	-22.3	-17.4	-11.4	-8.7	-6.4
Mz Max (Nm)	0.1	0.1	0.1	0.7	0.1
Mean (Nm)	13.6	-8.1	-6.2	-3.0	-3.3
Median (Nm)	-16.1	-10.8	-7.4	-4.2	-3.9
IRQ (Nm)	12.4	10.7	6.0	4.3	3.1
SD (Nm)	7.1	5.6	3.4	2.6	1.8
Area (Nm.s)	-7.1	-2.3	-2.2	-1.8	-1.0

Table 5.1: Average vectors of the properties of classes (A, B, C, D, E) used for the symbolic representation of the axial moment (Mz) SAX-P takes into account the surface under the push, the time-push and the time-cycle.

Subject	S1		S2		S3		
	Push	Right	Left	Right	Left	Right	Left
1	C	A	C	D	E	D	
2	B	B	E	E	D	E	
3	B	B	C	E	C	E	
4	B	B	C	E	E	E	
5	B	B	C	D	E	E	
6	C	B	E	C		D	
7	B	C	E	E		E	
8	E		E	E			
9			C	C			
10			E	E			
11			C	E			
12			C	E			
13				E			
DTW	268		354		44		

Table 5.2: Strings of characters obtained with SAX-P method on times series of axial moments applied by the three subjects on right and left rear wheels of an instrumented MWC during their second 10-m run.

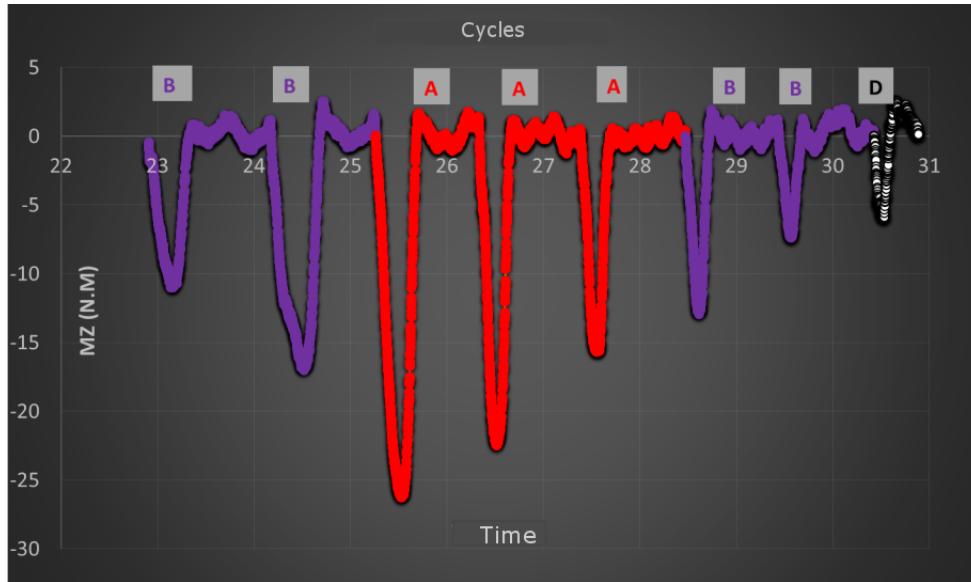


Figure 5.12: Symbolic representation of cyclic time series

5.4 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 the Dynamic Time Warping distance. The SAX-P model has been applied to propulsive moments measured during the movements in a straight line by three subjects in MWC. The preliminary results obtained have particularly showed that these subjects had different modes of propulsion and propulsion cycles of the same subject were not symmetrical. Ongoing research is devoted on applying this new symbolic representation to a supervised classification of cyclic time-series in bio-mechanics.

Chapter 6

Application to manual wheelchair locomotion

6.1 Introduction

To improve the efficiency of wheelchair propulsion, a Wheelchair Ergometer (FRET-2) equipped with sensors has been manufactured. The sensors installed on the wheelchair measure the physical stresses applied to the Manual Wheelchair (FRM) during actual use and record them. The following paragraphs present.

- A description of the measurements recorded by the sensors during actual use;
- An analysis of the data obtained by applying the algorithms as mentioned above (in our contribution).

6.2 Description of the dataset

The sensors are located on the right and left wheels of the manual wheelchair, on the footrest, on the seat and the backrest (see Figure 6.1). These sensors measure the forces and moments of these forces applied to each of the systems mentioned above. The moment of a force concerning a given point is a vectorial physical quantity which translates the ability of a force to turn a mechanical system around that point, often called a pivot [20]. The sensors installed on the FRM were used to measure the kinematic parameters (speed, acceleration) of the movement of the Manual Wheelchair (FRM), as well as its position relative to the Earth's magnetic north.

The measurements recorded by the sensors and subjected to our analysis consist of 44 attributes; 30 of the 44 attributes relate to the measurement of the torque

6.2 Description of the data

Chapter 6. Application to manual wheelchair locomotion

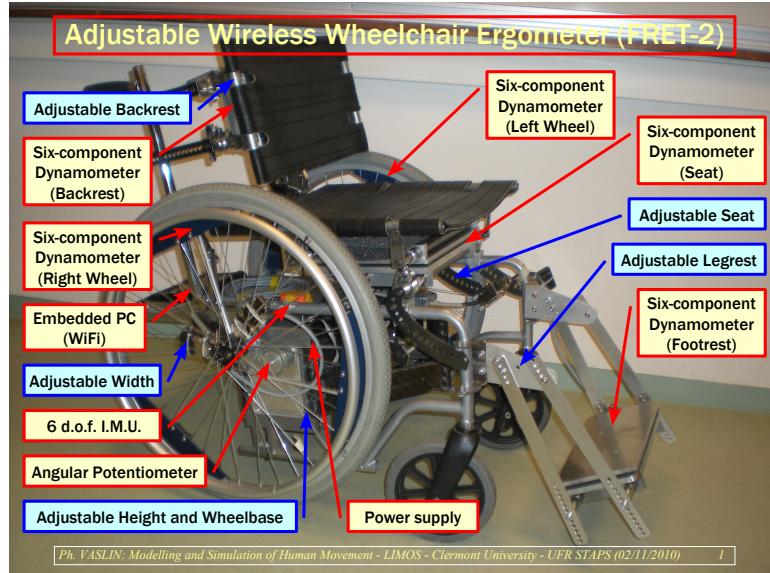


Figure 6.1: title

constituted by force applied to the systems mentioned above and the moment of this force to an axis of rotation. For each of the five systems, we have three components of the force (F_x , F_y , F_z) and the momentum (M_x , M_y , M_z) that apply to it. The 14 other attributes tell us about the kinematics of the manual wheelchair and its position relative to the Earth's magnetic north. The detailed description of these data is presented in Table 4.1.

To analyze the propulsion in FRM, we are interested in the moment along the axis of Z (M_z) applied to the right and left wheels of the FRM. This will allow us to identify the propulsion cycles during the move. A propulsion cycle consists of a push time interval and a consecutive freewheeling time interval that materializes on the Z-moment by a peak and drag as shown in Figure 4.2. The description of the data recorded by the sensors is presented in Appendix B.

6.2.1 Torque sensor

6.2.2 Characteristic properties of the data

The length

The cycles

The uncertainty

6.3 Analysis based on propulsion technique

6.4 Analysis based on propulsion capabilities

6.5 Propulsion technique versus propulsion capabilities

6.6 Conclusion

Appendix **A**

Hellinger Based Distance for Uncertain time series

Deterministic Measures like traditional similarity measures, return a real number as the distance between two uncertain time series.

A.1 DUST

[Murthy and Sarangi] is the only deterministic similarity measure defined for uncertain time series. 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 $r(X_i), r(Y_i)$: $dist(X_i, Y_i) = |r(X_i) - r(Y_i)|$. This distance is used to measures the similarity of two uncertain values. $\varphi(|X_i - Y_i|)$ is the probability that the reals values at timestamp i are equal, given the observed values at that timestamp i.e.

$$\varphi(|X_i - Y_i|) = Pr(dist(0|r(X_i) - r(Y_i)|) = 0). \quad (\text{A.1})$$

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

$$dust(X_i, Y_i) = \sqrt{-\log(\varphi(|X_i - Y_i|)) + \log(\varphi(0))}. \quad (\text{A.2})$$

The distance between uncertain time series $X = \langle X_1, \dots, X_n \rangle$ and $Y = \langle Y_1, \dots, Y_n \rangle$ in DUST is then defined as follows:

$$DUST(X, Y) = \sqrt{\sum_{i=1}^n dust(X_i, Y_i)^2}. \quad (\text{A.3})$$

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,

Figure A.1: Bhattacharyya

for density based clustering two time series A and B are considered similar if the distance between them is less than ϵ . Thus, if the sum of the distances $d(A, B)$ and $d(B, C)$ is less than ϵ , 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.*].

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.

A.2 Hellinger Based Distance

To evaluate the similarity between two probability distributions, we can measure the area of intersection between these two probability distributions (Figure A.1). 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.*].

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 11. *The Hellinger distance between two probability measures P and Q that are absolutely continuous relative to some σ -finite measure μ on a measurable space (x, β) is defined by the formula:*

$$H(P, Q) = \{2[1 - B(P, Q)]\}^{\frac{1}{2}}, \quad (\text{A.4})$$

where

$$B(P, Q) = \int \sqrt{\frac{dP}{d\mu}} \sqrt{\frac{dQ}{d\mu}} d\mu. \quad (\text{A.5})$$

Theorem 2. *The Hellinger distance satisfy the triangle inequality [Ibragimov and Has' minskii].*

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

Definition 12. *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:*

$$HBD(X, Y) = \sqrt{\sum_{i=1}^n H(X_i, Y_i)^2}. \quad (\text{A.6})$$

Theorem 3. *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

$$\sqrt{\sum_{i=1}^n H(X_i, Y_i)^2} + \sqrt{\sum_{i=1}^n H(Y_i, Z_i)^2} \geq \sqrt{\sum_{i=1}^n H(X_i, Z_i)^2}. \quad (\text{A.7})$$

First, let us show that:

$$\left(\sqrt{\sum_{i=1}^n H(X_i, Y_i)^2} \right) \times \left(\sqrt{\sum_{i=1}^n H(Y_i, Z_i)^2} \right) \geq \sum_{i=1}^n H(X_i, Y_i)H(Y_i, Z_i) \quad (\text{A.8})$$

By squaring the two members of the inequality (A.8) we obtain

$$\left(\sum_{i=1}^n H(X_i, Y_i)^2 \right) \times \left(\sum_{i=1}^n H(Y_i, Z_i)^2 \right) \geq \left(\sum_{i=1}^n H(X_i, Y_i)H(Y_i, Z_i) \right)^2 \quad (\text{A.9})$$

$$\text{i.e. } \left(\sum_{i=1}^n H(X_i, Y_i)^2 \right) \times \left(\sum_{i=1}^n H(Y_i, Z_i)^2 \right) - \left(\sum_{i=1}^n H(X_i, Y_i)H(Y_i, Z_i) \right)^2 \geq 0 \quad (\text{A.10})$$

By developing and reducing the expression(A.10), we obtain

$$\text{i.e. } \sum_{i,j \in \{1, \dots, n\} \text{ and } i \neq j} (H(X_i, Y_i) - H(Y_j, Z_j))^2 \geq 0 \quad (\text{A.11})$$

This shows that the inequality (A.8) is true. Let us now show that HBD satisfies the triangular inequality : according to Theorem 2,

$$H(X_i, Y_i) + H(Y_i, Z_i) \geq H(X_i, Z_i) \quad (\text{A.12})$$

By squaring the two members of the inequality, we obtain

$$H(X_i, Y_i)^2 + H(Y_i, Z_i)^2 + 2H(X_i, Y_i)H(Y_i, Z_i) \geq H(X_i, Z_i)^2. \quad (\text{A.13})$$

$$\text{i.e. } \sum_{i=1}^n H(X_i, Y_i)^2 + \sum_{i=1}^n H(Y_i, Z_i)^2 + 2 \sum_{i=1}^n H(X_i, Y_i)H(Y_i, Z_i) \geq \sum_{i=1}^n H(X_i, Z_i)^2 \quad (\text{A.14})$$

according to inequality A.8, we obtain

$$\sum_{i=1}^n H(X_i, Y_i)^2 + \sum_{i=1}^n H(Y_i, Z_i)^2 + 2 \left(\sqrt{\sum_{i=1}^n H(X_i, Y_i)^2} \right) \times \left(\sqrt{\sum_{i=1}^n H(Y_i, Z_i)^2} \right) \geq \sum_{i=1}^n H(X_i, Z_i)^2 \quad (\text{A.15})$$

$$\text{i.e. } \left(\sqrt{\sum_{i=1}^n H(X_i, Y_i)^2} + \sqrt{\sum_{i=1}^n H(Y_i, Z_i)^2} \right)^2 \geq \sum_{i=1}^n H(X_i, Z_i)^2. \quad (\text{A.16})$$

$$\text{i.e. } \sqrt{\sum_{i=1}^n H(X_i, Y_i)^2} + \sqrt{\sum_{i=1}^n H(Y_i, Z_i)^2} \geq \sqrt{\sum_{i=1}^n H(X_i, Z_i)^2}. \quad (\text{A.17})$$

This is what had to be demonstrated \square

The problem with the deterministic uncertain distance distances DUST and HBD is that their expression varies as a function of the probability law that uncertainty follows. Their use therefore requires knowledge of the law of probability of the uncertainty contained in the data, which is not always possible in practice.

Appendix **B**

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

B.1 Introduction

Time series databases are often extremely large. This is particularly the case of the Large Synoptic Survey Telescope (LSST) database which records data from of telescopes [lss]. She 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 1000 Hz [SAC]. Ten minutes recording time series of 600 000 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 Fu1999b] and approximate aggregation by segments (PAA) [Keogh *et al.*]. 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 size, then replace each segment by the average of its points. This method generates a representation compact, able to have a few segments as possible to reduce space storage and time comparison time series. However, too compact a 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 average arithmetic is

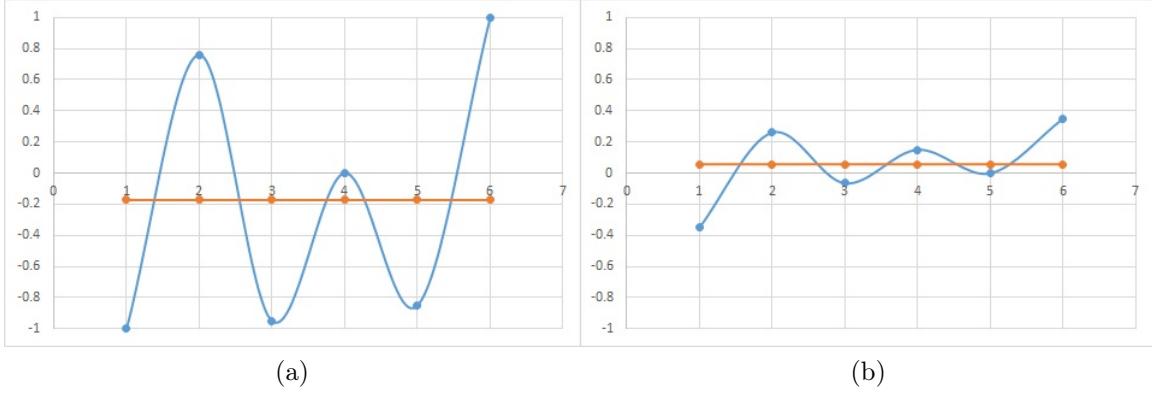


Figure B.1: These figures show the average of two segments. In the first case (a) the data points of the segment are far from the average, in the second case (b) they are close to the average. Replace data points of a segment by their average introduces an error that can be measured from the gap between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

relevant when the variance of the population is small as illustrated by the figure B.1.

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, this to reduce the length of the time series without altering the information they contain.

The rest of this chapter is organized as follows: the B.2 section presents a formal definition of our problem and an algorithm used to solve it; the section B.3 presents and comments on 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: $X_i = x_i, x_{i+1}, \dots, x_{i+l-1}$

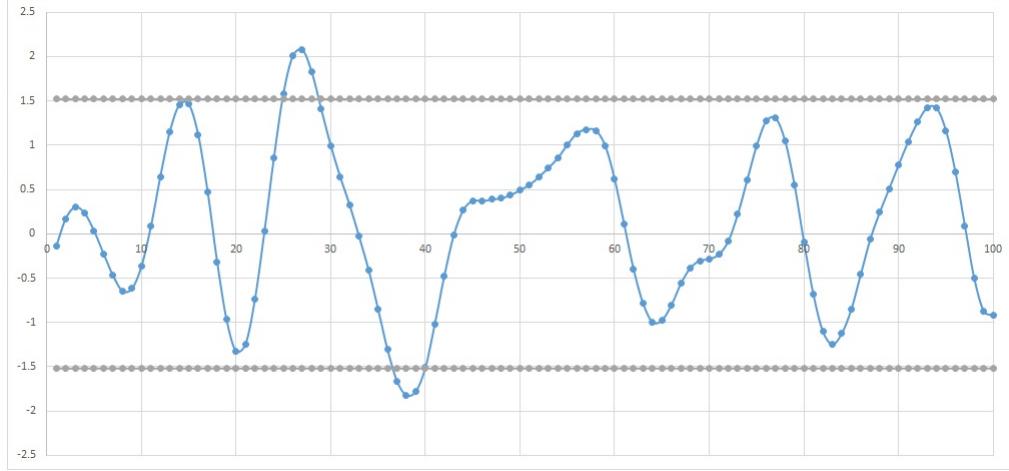


Figure B.2: This figure shows the first 100 points of the first time series of the fordA dataset available in the UCR [Chen *et al.* 2015] database. Time series are normalized. The two horizontal lines delimit the interval corresponding to twice the standard deviation and minus two times the standard 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 (σ) 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 [Shannon2001]. 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 of (α) segments less than 5 % of the length of the time series, we run the risk of aggregating the points within the interval $[-2\sigma, 2\sigma]$ and those outside this range. This will have two consequences: on the one hand, to alter the information carried by these points. On the other hand, increase the variance of the segments obtained. Because we will aggregate the points at the ends and those near the average. 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 $\mathbb{N} \rightarrow \mathbb{N}$:

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

$$\beta : n \mapsto \beta(n) = \left\lfloor \frac{n}{2} \right\rfloor$$

β 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 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 an average squared error minimum;

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  $\in$  {length_min, ..., 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.
nbPoints : the length of a segment
Output: The sum of squares error associated with a segment length

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

B.2.5 Dynamic Time Warping Algorithm and Comparison of Representations compact

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: 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.

Approximate aggregation by segment provides distance-based distance measurement Euclidean 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 size n , we can apply our algorithm to the first time series, 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 performance of the proposed method from the 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		467	3840
10	CinC_ECG_torso4		40	1380
Continue to the next page				

N	Name	Nb. of classes	Size of training set	Size of testing set
Following ...				
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		16	306
17	DistalPhalanxOutLineAgeGroup		139	400
18	DistalPhalanxOutLineCorrect		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		220	1980
38	ItalyPowerDemand2		67	1029
39	LargeKitchenAppliances		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		154	400
...

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 tables ?? and B.2 present 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 tables ?? and 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 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	0,279

Continue to the next page

(1)	(2)	(3)	(4)	(5)	(6)	(7)
Following ...						
2	35	0,20	0,389	0,391 (3)	0,396	0,425
3	50	0,20	0,2	0,200 (0)	0,297	0,246
4	78	0,17	0,333	0,333 (0)	0,367	0,433
5	85	0,17	0,25	0,300 (7)	0,3	0,300
6	85	0,17	0,45	0,300(6)	0,25	0,250
7	96	0,17	0,267	0,233 (1)	0,267	0,217
8	32	0,25	0,148	0,004 (11)	0,003	0,002
9	33	0,20	0,35	0,35 (0)	0,352	0,414
10	234	0,14	0,103	0,07 (1)	0,349	0,285
11	57	0,20	0	0,000 (0)	0	0,036
12	120	0,17	0,424	0,380 (13)	0,3	0,416
13	60	0,20	0,423	0,228 (10)	0,246	0,241
14	60	0,20	0,433	0,238 (17)	0,256	0,277
15	60	0,20	0,413	0,254 (5)	0,246	0,244
16	69	0,20	0,065	0,065 (0)	0,033	0,072
17	20	0,25	0,218	0,228 (1)	0,208	0,198
18	20	0,25	0,248	0,232 (2)	0,232	0,255
19	20	0,25	0,273	0,272 (0)	0,29	0,310
20	85	0,17	0,326	0,258 (22)	0,258	0,276
21	24	0,25	0,12	0,120 (0)	0,23	0,180
Continue to the next page						

(1)	(2)	(3)	(4)	(5)	(6)	(7)
Following ...						
22	35	0,25	0,075	0,075 (1)	0,076	0,072
23	34	0,25	<i>0,203</i>	<i>0,203 (0)</i>	0,232	0,259
24	24	0,25	0,45	<i>0,376 (14)</i>	0,399	0,438
25	32	0,24	0,286	<i>0,192 (3)</i>	0,192	0,253
26	70	0,20	0,216	<i>0,114 (2)</i>	0,17	0,170
27	32	0,24	0,231	<i>0,088 (12)</i>	0,095	0,177
28	77	0,17	0,217	<i>0,154(4)</i>	0,177	0,263
29	83	0,17	<i>0,341</i>	<i>0,341 (0)</i>	0,438	0,359
30	83	0,17	0,442	0,414 (1)	0,406	0,360
31	30	0,20	0,087	0,087 (0)	0,093	0,047
32	71	0,16	<i>0,4</i>	<i>0,400 (0)</i>	0,533	0,419
33	387	0,14	0,199	<i>0,197 (1)</i>	0,202	0,206
34	182	0,17	0,63	<i>0,588 (2)</i>	0,623	0,623
35	85	0,17	0,484	<i>0,469 (5)</i>	0,469	0,500
36	268	0,14	0,658	<i>0,613 (14)</i>	0,616	0,615
37	51	0,20	0,438	<i>0,422 (2)</i>	0,645	0,611
38	8	0,33	<i>0,045</i>	<i>0,045 (0)</i>	0,05	0,048
39	120	0,17	0,507	0,205 (94)	0,205	0,203
40	106	0,17	0,246	<i>0,131 (6)</i>	0,131	0,164

Continue to the next page

(1)	(2)	(3)	(4)	(5)	(6)	(7)
Following ...						
41	63	0,20	0,425	0,288 (5)	0,274	0,219
42	170	0,17	0,086	0,086 (0)	0,066	0,097
43	74	0,17	<i>0,067</i>	<i>0,067</i> <i>(0)</i> <i>0,067</i>		0,067
44	24	0,24	0,316	<i>0,253</i> <i>(20)</i>	0,263	0,288
45	20	0,25	0,26	0,253 (5)	0,25	0,268
46	20	0,25	<i>0,247</i>	0,318 (1)	0,352	0,268
47	20	0,25	0,439	0,419 (2)	0,416	0,419
48	21	0,25	<i>0,121</i>	0,134 (1)	0,165	0,133
49	125	0,17	<i>0,171</i>	0,185 (1)	0,209	0,222
50	125	0,17	<i>0,12</i>	0,129 (1)	0,135	0,146
51	95	0,17	<i>0,133</i>	<i>0,133</i> (0)	0,167	0,167
52	71	0,17	0,479	0,388 (7)	0,409	0,355
53	20	0,25	<i>0,239</i>	<i>0,239</i> (0)	0,272	0,273
54	170	0,17	0,891	0,773 (14)	0,772	0,809
55	36	0,25	0,038	<i>0,000</i> (6)	0	0,000
56	20	0,25	0,215	0,215 (0)	0,195	0,249
57	20	0,25	<i>0,192</i>	0,210 (1)	0,216	0,251
58	20	0,25	0,292	<i>0,263</i> (6)	0,263	0,280
59	120	0,17	0,605	0,560 (8)	0,536	0,501

Continue to the next page

(1)	(2)	(3)	(4)	(5)	(6)	(7)
Following ...						
60	120	0,17	0,64	0,589 (17) 0,300 (3)	0,603	0,645
61	83	0,17	0,461	0,198 (4)	0,35	0,339
62	85	0,17	0,248	0,328 (15)	0,232	0,210
63	120	0,17	0,659	0,305 (0)	0,357	0,349
64	17	0,24	0,305	0,141 (0)	0,275	0,250
65	16	0,25	0,151	0,141 (0)	0,169	0,189
66	170	0,17	0,095	0,093 (16)	0,093	0,124
67	47	0,20	0,062	0,062 (0)	0,06	0,055
68	32	0,25	0,211	0,154 (2)	0,208	0,184
69	79	0,20	0,1	0,062 (8)	0,05	0,048
70	15	0,25	0,12	0,017 (6)	0,007	0,017
71	55	0,20	0,32	0,250 (8)	0,228	0,193
72	68	0,20	0,192	0,092 (5)	0,162	0,154
73	55	0,20	0,24	0,010 (3)	<i>0</i>	0,070
74	32	0,25	0,09	0,002 (4)	<i>0</i>	0,000
75	20	0,24	0,253	0,132 (5)	0,096	0,283
76	63	0,20	0,261	0,227 (4)	0,273	0,252
77	63	0,20	0,338	0,301 (4)	0,366	0,346
78	63	0,20	0,35	0,322 (6)	0,342	0,334

Continue to the next page

(1)	(2)	(3)	(4)	(5)	(6)	(7)
Following ...						
79	157	0,17	0,052	<i>0,034</i> (4)	0,108	0,067
80	30	0,20	0,005	0,005 (1)	0,02	0,021
81	46	0,20	0,389	0,389 (0)	0,426	0,315
82	54	0,20	0,382	<i>0,252</i> (8)	0,351	0,320
83	150	0,17	0,635	0,586 (3)	0,536	0,508
84	150	0,17	0,414	0,414 (9)	0,337	0,320
85	71	0,17	0,17	<i>0,155</i> (2)	0,164	0,174
\bar{X}			0,288	0,237	0,256	0,258
σ			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

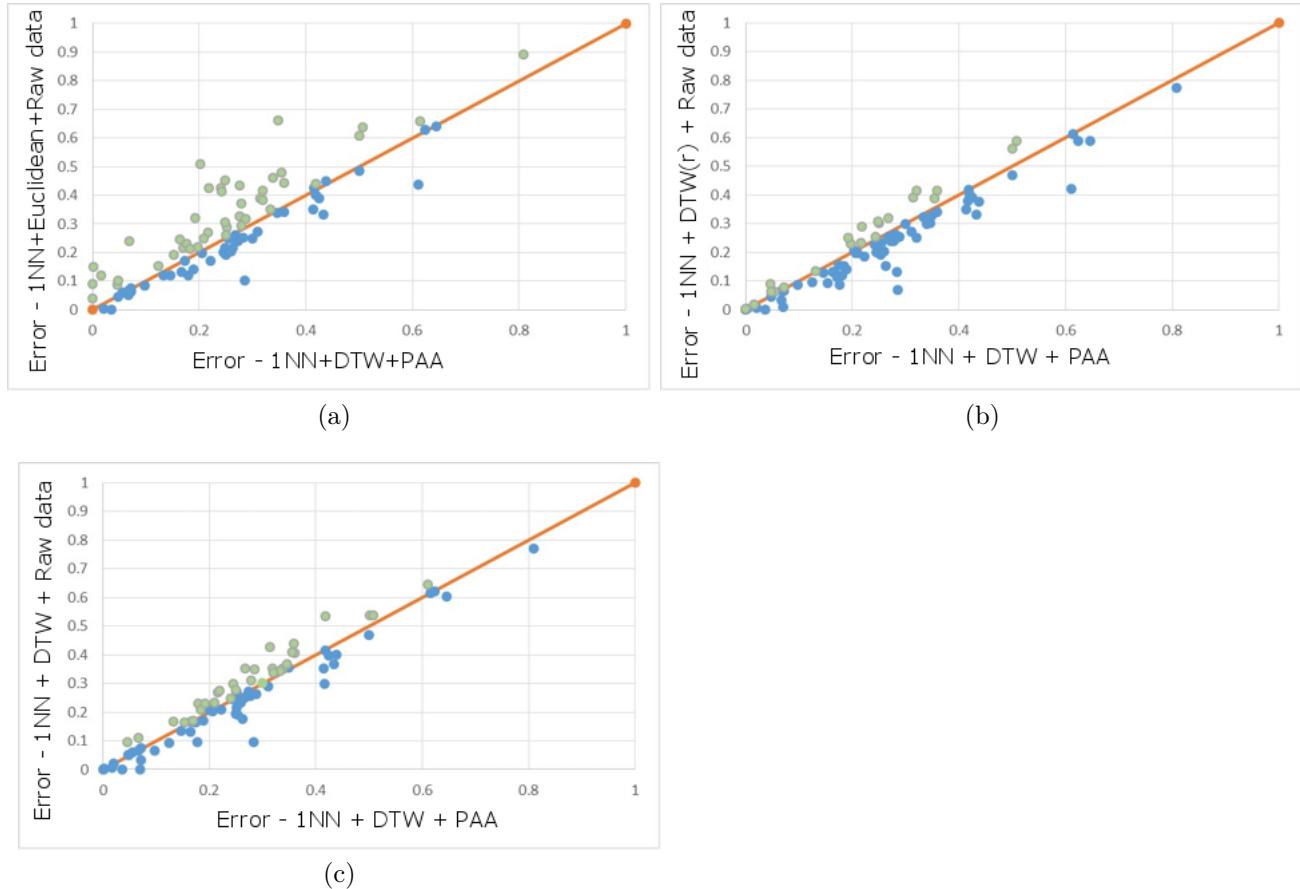


Figure B.3: Two-to-two comparison of the classification errors of the algorithm 1-Nearest Neighbor (1-NN) using Euclidean distance with 1-NN using two variations of the temporal warping algorithm on data raw and compact data

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.*], 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.

Bibliography

- [0Åstrand and Saltin1961] Per-Olof 0Åstrand and Bengt Saltin. Maximal oxygen uptake and heart rate in various types of muscular activity. *Journal of Applied Physiology*, 16(6):977–981, 1961.
- [Aach and Church2001] J. Aach and G. M. Church. Aligning gene expression time series with time warping algorithms. *Bioinformatics*, 17(6):495–508, June 2001.
- [Abonyi *et al.*2003] Janos Abonyi, Balazs Feil, Sandor Nemeth, and Peter Arva. Fuzzy clustering based segmentation of time-series. *Advances in Intelligent Data Analysis V*, pages 275–285, 2003.
- [Aggarwal] Charu C Aggarwal. On unifying privacy and uncertain data models. In *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*, pages 386–395. IEEE.
- [Agrawal *et al.*1993] Rakesh Agrawal, Christos Faloutsos, and Arun N. Swami. 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, oct 1993.
- [Amini *et al.*2011] Amineh Amini, Teh Ying Wah, Mahmoud Reza Saybani, and Saeed Reza Aghabozorgi Sahaf Yazdi. 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, 2011.
- [Aßfalg *et al.*] Johannes Aßfalg, Hans-Peter Kriegel, Peer Kröger, and Matthias Renz. Probabilistic similarity search for uncertain time series. In *SSDBM*, pages 435–443. Springer.
- [Athanasios and Clark2009] Maria Athanasios and Jonathan Y Clark. 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, 2009.

- [Bagnall *et al.*a] Anthony Bagnall, Eamonn Keogh, Jason Lines, Aaron Bostrom, and James Large. Time Series Classification Website. <http://timeseriesclassification.com>.
- [Bagnall *et al.*b] Anthony Bagnall, Jason Lines, Aaron Bostrom, James Large, and Eamonn Keogh. The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. pages 1–55.
- [Batista *et al.*] Gustavo EAPA Batista, Eamonn J Keogh, Oben Moses Tataw, and Vinicius MA De Souza. Cid: an efficient complexity-invariant distance for time series. 28(3):634–669.
- [Batista *et al.*2011] Gustavo EAPA Batista, Xiaoyue Wang, and Eamonn J Keogh. A complexity-invariant distance measure for time series. In *Proceedings of the 2011 SIAM international conference on data mining*, pages 699–710. SIAM, 2011.
- [Baydogan and Runger2016] Mustafa Gokce Baydogan and George Runger. Time series representation and similarity based on local autopatterns. *Data Mining and Knowledge Discovery*, 30(2):476–509, 2016.
- [Baydogan *et al.*2013] Mustafa Gokce Baydogan, George Runger, and Eugene Tuv. A bag-of-features framework to classify time series. *IEEE transactions on pattern analysis and machine intelligence*, 35(11):2796–2802, 2013.
- [Begum and Keogh2014] Nurjahan Begum and Eamonn Keogh. Rare time series motif discovery from unbounded streams. *Proceedings of the VLDB Endowment*, 8(2):149–160, October 2014.
- [Bergh *et al.*1976] U Bergh, IL Kanstrup, and B Ekblom. Maximal oxygen uptake during exercise with various combinations of arm and leg work. *Journal of Applied Physiology*, 41(2):191–196, 1976.
- [Bhatia and Bhattacharyya] Rajendra Bhatia and Tirthankar Bhattacharyya. A generalization of the Hoffman-Wielandt theorem. 179:11–17.
- [Brouha and Krobath1967] Lucien Brouha and Hans Krobath. Continuous recording of cardiac and respiratory functions in normal and handicapped people. *Human factors*, 9(6):567–571, 1967.
- [Buttkus2012] Burkhard Buttkus. *Spectral analysis and filter theory in applied geophysics*. Springer Science & Business Media, 2012.
- [Camerra *et al.*] Alessandro Camerra, Themis Palpanas, Jin Shieh, and Eamonn Keogh. 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.*] K Selçuk Candan, Rosaria Rossini, Xiaolan Wang, and Maria Luisa Sapino. sdtw: computing dtw distances using locally relevant constraints based on salient feature alignments. 5(11):1519–1530.
- [Chan and Fu1999a] Kin-Pong Chan and Ada Wai-Chee Fu. Efficient time series matching by wavelets. In *Data Engineering, 1999. Proceedings., 15th International Conference on*, pages 126–133. IEEE, 1999.
- [Chan and Fu1999b] Kin-Pong Chan and Ada Wai-Chee Fu. Efficient time series matching by wavelets. In *Proceedings 15th International Conference on Data Engineering (Cat. No.99CB36337)*, pages 126–133. IEEE, 1999.
- [Chandola *et al.* 2009] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41(3):15, 2009.
- [Chen *et al.* 2005] Lei Chen, M Tamer Özsu, and Vincent Oria. 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, 2005.
- [Chen *et al.* 2015] Yanping Chen, Eamonn Keogh, Bing Hu, Nurjahan Begum, Anthony Bagnall, Abdullah Mueen, and Gustavo Batista. The ucr time series classification archive. http://www.cs.ucr.edu/~eamonn/time_series_data/, July 2015. www.cs.ucr.edu/~eamonn/time_series_data/.
- [Cheng *et al.*] Reynold Cheng, Dmitri V Kalashnikov, and Sunil Prabhakar. Evaluating probabilistic queries over imprecise data. In *Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, pages 551–562. ACM.
- [Chiu *et al.*] Bill Chiu, Eamonn Keogh, and Stefano Lonardi. Probabilistic discovery of time series motifs. In *ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 493–498. ACM.
- [Chu *et al.*] Selina Chu, Eamonn J Keogh, David M Hart, Michael J Pazzani, et al. Iterative deepening dynamic time warping for time series. In *SDM*, pages 195–212. SIAM.
- [Claremont and Maksud1985] AD Claremont and MG Maksud. A model treadmill adaptation for wheelchair ergometry. *Canadian journal of applied sport sciences. Journal canadien des sciences appliquées au sport*, 10(4):178–181, 1985.
- [Cooper *et al.* 2003] Rory A Cooper, Michael L Boninger, Rosemarie Cooper, RN Robertson, and FD Baldini. Wheelchair racing efficiency. *Disability and rehabilitation*, 25(4-5):207–212, 2003.

- [Cooper1990] Rory A Cooper. An exploratory study of racing wheelchair propulsion dynamics. *Adapted Physical Activity Quarterly*, 7(1):74–85, 1990.
- [Coutts and Stogryn1987] KENNETH D Coutts and JANET L Stogryn. Aerobic and anaerobic power of canadian wheelchair track athletes. *Medicine and science in sports and exercise*, 19(1):62–65, 1987.
- [Coutts1990] Kenneth D Coutts. Kinematics of sport wheelchair propulsion. *Journal of rehabilitation research and development*, 27(1):21, 1990.
- [Cuřín *et al.*] Jan Cuřín, Pascal Fleury, Jan Kleindienst, and Robert Kessl. Meeting state recognition from visual and aural labels. In *International Workshop on Machine Learning for Multimodal Interaction*, pages 24–25. Springer.
- [Dallachiesa *et al.*] Michele Dallachiesa, Besmira Nushi, Katsiaryna Mirylenka, and Themis Palpanas. Uncertain time-series similarity: return to the basics. *VLDB Endowment*, 5(11):1662–1673.
- [Dallachiesa *et al.*2012] Michele Dallachiesa, Besmira Nushi, Katsiaryna Mirylenka, and Themis Palpanas. Uncertain time-series similarity: return to the basics. *VLDB Endowment*, 5(11):1662–1673, 2012.
- [Das *et al.*1997] Gautam Das, Dimitrios Gunopoulos, and Heikki Mannila. Finding similar time series. In *European Symposium on Principles of Data Mining and Knowledge Discovery*, pages 88–100. Springer, 1997.
- [De Lathauwer *et al.*1994] L De Lathauwer, B De Moor, J Vandewalle, and Blind Source Separation by Higher-Order. Singular value decomposition. In *Proc. EUSIPCO-94, Edinburgh, Scotland, UK*, volume 1, pages 175–178, 1994.
- [Dean and Ghemawat2010] Jeffrey Dean and Sanjay Ghemawat. MapReduce. *Communications of the ACM*, 53(1):72, jan 2010.
- [Deng *et al.*2013] Houtao Deng, George Runger, Eugene Tuv, and Martyanov Vladimir. A time series forest for classification and feature extraction. *Information Sciences*, 239:142–153, 2013.
- [Devillard *et al.*2001] Xavier Devillard, Paul Calmels, Bernard Sauvignet, Alain Belli, Christian Denis, Clermont Simard, and Vincent Gautheron. Validation of a new ergometer adapted to all types of manual wheelchair. *European journal of applied physiology*, 85(5):479–485, 2001.
- [Devillard1999] X Devillard. A wheelchair ergometer for physiological and biomechanical measurement. In *Proceedings of the 11th*, 1999.

- [Dhillon *et al.*2003] Inderjit S Dhillon, Subramanyam Mallela, and Dharmendra S Modha. Information-theoretic co-clustering. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 89–98. ACM, 2003.
- [Dietterich2002] Thomas G. Dietterich. Machine Learning for Sequential Data: A Review. *Proceedings of the Joint IAPR International Workshop on Structural, Syntactic, and Statistical Pattern Recognition*, pages 16–30, August 2002.
- [DiGiovine *et al.*2001] Carmen P DiGiovine, Rory A Cooper, and Michael L Boninger. Dynamic calibration of a wheelchair dynamometer. *Journal of rehabilitation research and development*, 38(1):41, 2001.
- [Ding *et al.*2008] Hui Ding, Goce Trajcevski, Peter Scheuermann, Xiaoyue Wang, and Eamonn Keogh. Querying and mining of time series data. *Proceedings of the VLDB Endowment*, 1(2):1542–1552, August 2008.
- [Eriksson *et al.*2004] Kenneth Eriksson, Donald Estep, and Claes Johnson. Piecewise linear approximation. In *Applied Mathematics: Body and Soul*, pages 741–753. Springer, 2004.
- [Esling and Agon2012] Philippe Esling and Carlos Agon. Time-series data mining. *ACM Computing Surveys (CSUR)*, 45(1):1–34, 2012.
- [Faria *et al.*2012] Brígida Mónica Faria, Luís Paulo Reis, Nuno Lau, João Couto Soares, and Sérgio Vasconcelos. Patient classification and automatic configuration of an intelligent wheelchair. In *International Conference on Agents and Artificial Intelligence*, pages 268–282. Springer, 2012.
- [Feo and Resende1995] Thomas A Feo and Mauricio GC Resende. Greedy randomized adaptive search procedures. *Journal of global optimization*, 6(2):109–133, 1995.
- [Filippone *et al.*2008] Maurizio Filippone, Francesco Camastra, Francesco Masulli, and Stefano Rovetta. A survey of kernel and spectral methods for clustering. *Pattern recognition*, 41(1):176–190, 2008.
- [Folland and Sitaram1997] Gerald B Folland and Alladi Sitaram. The uncertainty principle: a mathematical survey. *Journal of Fourier analysis and applications*, 3(3):207–238, 1997.
- [Fu2011] Tak-chung Fu. A review on time series data mining. *Engineering Applications of Artificial Intelligence*, 24(1):164–181, 2011.
- [Gao *et al.*] Yifeng Gao, Jessica Lin, and Huzeifa Rangwala. Iterative grammar-based framework for discovering variable-length time series motifs. In *Machine*

Learning and Applications (ICMLA), 2016 15th IEEE International Conference on, pages 7–12. IEEE.

[Ghalwash *et al.*] Mohamed F Ghalwash, Vladan Radosavljevic, and Zoran Obradovic. Utilizing temporal patterns for estimating uncertainty in interpretable early decision making. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 402–411. ACM.

[Godsill] Simon J Godsill. On the relationship between Markov chain Monte Carlo methods for model uncertainty. 10(2):230–248.

[Goosey *et al.* 1998] Victoria L Goosey, Ian G Campbell, and Neil E Fowler. The relationship between three-dimensional wheelchair propulsion techniques and pushing economy. *Journal of applied biomechanics*, 14(4):412–427, 1998.

[Goosey-Tolfrey *et al.* 2001] Victoria L Goosey-Tolfrey, Neil E Fowler, Ian G Campbell, and Simon D Iwnicki. A kinetic analysis of trained wheelchair racers during two speeds of propulsion. *Medical engineering and physics*, 23(4):259–266, 2001.

[Grabocka *et al.* 2014] Josif Grabocka, Nicolas Schilling, Martin Wistuba, and Lars Schmidt-Thieme. Learning time-series shapelets. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 392–401. ACM, 2014.

[Gusfield 1997] Dan Gusfield. *Algorithms on strings, trees and sequences: computer science and computational biology*. Cambridge university press, 1997.

[Halkidi *et al.* 2001] Maria Halkidi, Yannis Batistakis, and Michalis Vazirgiannis. On clustering validation techniques. *Journal of intelligent information systems*, 17(2-3):107–145, 2001.

[Hamilton] Jd Hamilton. A new approach to the economic analysis of nonstationary time series and the business cycle. 57(2):357–384.

[Hamilton 1989] Jd Hamilton. A new approach to the economic analysis of nonstationary time series and the business cycle. *Econometrica: Journal of the Econometric Society*, 57(2):357–384, 1989.

[Hochheiser and Shneiderman 2003] Harry Hochheiser and Ben Shneiderman. Interactive exploration of time series data. In *The Craft of Information Visualization*, pages 313–315. Elsevier, 2003.

[Hofmann 2017] Thomas Hofmann. Probabilistic latent semantic indexing. In *ACM SIGIR Forum*, volume 51, pages 211–218. ACM, 2017.

[Huang and Kinsner] B Huang and W Kinsner. ECG frame classification using dynamic time warping. 2:1105–1110.

- [Huang and Kinsner2002] B Huang and W Kinsner. Ecg frame classification using dynamic time warping. *Electrical and Computer Engineering*, 2:1105–1110, 2002.
- [Hwang *et al.*] Jun Hwang, Yusuke Kozawa, Toshiyuki Amagasa, and Hiroyuki Kitagawa. 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] Oscar H Ibarra and Chul E Kim. Fast approximation algorithms for the knapsack and sum of subset problems. 22(4):463–468.
- [Ibragimov and Has' minskii] Ildar Abdulovič Ibragimov and Rafail Z Has' minskii. *Statistical estimation: asymptotic theory*, volume 16. Springer Science & Business Media.
- [Itakura] Fumitada Itakura. Minimum prediction residual principle applied to speech recognition. 23(1):67–72.
- [Jeong *et al.*] Young-Seon Jeong, Myong K Jeong, and Olufemi A Omitaomu. Weighted dynamic time warping for time series classification. 44(9):2231–2240.
- [Johnson *et al.*2004] B Johnson, C Mushett, Ken Richter, and G Peacock. Sport for athletes with physical disabilities: injuries and medical issues. *BlazeSports America*, 2004.
- [Kalpakis *et al.*2001] Konstantinos Kalpakis, Dhiral Gada, and Vasundhara Puttagunta. Distance measures for effective clustering of arima time-series. In *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, pages 273–280. IEEE, 2001.
- [Kate] Rohit J. Kate. Using dynamic time warping distances as features for improved time series classification. 30(2):283–312.
- [Kate2016] Rohit J Kate. Using dynamic time warping distances as features for improved time series classification. *Data Mining and Knowledge Discovery*, 30(2):283–312, 2016.
- [Kendall1961] Maurice George Kendall. *The advanced theory of statistics: Inference and relationship*, volume 2. Charles Griffin, 1961.
- [Keogh and Kasetty2003a] Eamonn Keogh and Shruti Kasetty. On the need for time series data mining benchmarks: a survey and empirical demonstration. *Data Mining and knowledge discovery*, 7(4):349–371, 2003.
- [Keogh and Kasetty2003b] Eamonn Keogh and Shruti Kasetty. On the Need for Time Series Data Mining Benchmarks: A Survey and Empirical Demonstration. *Data Mining and Knowledge Discovery*, 7(4):349–371, 2003.

- [Keogh and Pazzania] E J Keogh and M J Pazzani. Derivative dynamic time warping. pages 1–11.
- [Keogh and Pazzanib] Eamonn J Keogh and Michael J Pazzani. Scaling up dynamic time warping for datamining applications. In *sixth ACM SIGKDD*, pages 285–289. ACM.
- [Keogh and Pazzani2001a] E J Keogh and M J Pazzani. Derivative dynamic time warping. *1st SIAM International Conference on Data Mining*, pages 1–11, 2001.
- [Keogh and Pazzani2001b] Eamonn J Keogh and Michael J Pazzani. Derivative dynamic time warping. In *Proceedings of the 2001 SIAM International Conference on Data Mining*, pages 1–11. SIAM, 2001.
- [Keogh and Ratanamahatana] Eamonn Keogh and Chotirat Ann Ratanamahatana. Exact indexing of dynamic time warping. *7(3):358–386*.
- [Keogh and Ratanamahatana2004] E Keogh and A Ratanamahatana. Everything you know about dynamic time warping is wrong. In *3rd Workshop on Mining Temporal and Sequential Data, in conjunction with 10th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD-2004), Seattle, WA*, 2004.
- [Keogh *et al.*] Eamonn Keogh, Kaushik Chakrabarti, Michael Pazzani, and Sharad Mehrotra. Locally adaptive dimensionality reduction for indexing large time series databases. *30(2):151–162*.
- [Keogh *et al.*2001a] Eamonn Keogh, Kaushik Chakrabarti, Michael Pazzani, and Sharad Mehrotra. Dimensionality Reduction for Fast Similarity Search in Large Time Series Databases. *Knowledge and Information Systems*, 3(January):263–286, 2001.
- [Keogh *et al.*2001b] Eamonn Keogh, Kaushik Chakrabarti, Michael Pazzani, and Sharad Mehrotra. Dimensionality reduction for fast similarity search in large time series databases. *Knowledge and information Systems*, 3(3):263–286, 2001.
- [Keogh *et al.*2001c] Eamonn Keogh, Kaushik Chakrabarti, Michael Pazzani, and Sharad Mehrotra. Dimensionality Reduction for Fast Similarity Search in Large Time Series Databases. *Knowledge and Information Systems*, 3(January):263–286, 2001.
- [Kerk *et al.*1995] Julie K Kerk, Philip S Clifford, Ann C Snyder, Thomas E Prieto, Kathleen P O’hagan, Philip K Schot, Joel B Myklebust, and Barbara M Myklebust. Effect of an abdominal binder during wheelchair exercise. *Medicine and science in sports and exercise*, 27(6):913–919, 1995.

- [Kriegel *et al.*2011] Hans-Peter Kriegel, Peer Kröger, Jörg Sander, and Arthur Zimek. Density-based clustering. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(3):231–240, 2011.
- [Kuczera and Parent] George Kuczera and Eric Parent. Monte Carlo assessment of parameter uncertainty in conceptual catchment models: the metropolis algorithm. 211(1):69–85.
- [Kulig *et al.*2001] Kornelia Kulig, Craig J Newsam, Sara J Mulroy, Sreesha Rao, JoAnne K Gronley, Ernest L Bontrager, and Jacqueline Perry. The effect of level of spinal cord injury on shoulder joint kinetics during manual wheelchair propulsion. *Clinical biomechanics*, 16(9):744–751, 2001.
- [Langbein and Fehr1993] W Edwin Langbein and Linda Fehr. Research device to preproduction prototype: a chronology. *Journal of rehabilitation research and development*, 30(4):436, 1993.
- [Langbein *et al.*1993] WE Langbein, CJ Robinson, L Kynast, and L Fehr. Calibration of a new wheelchair ergometer: the wheelchair aerobic fitness trainer. *IEEE Transactions on Rehabilitation Engineering*, 1(1):49–58, 1993.
- [Langbein *et al.*1994] W Edwin Langbein, Kevin C Maki, Lonnie C Edwards, Ming H Hwang, et al. Initial clinical evaluation of a wheelchair ergometer for diagnostic exercise testing: a technical note. *Journal of rehabilitation research and development*, 31(4):317, 1994.
- [Lin and Li2009] Jessica Lin and Yuan Li. 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, 2009.
- [Lin *et al.*2003] Jessica Lin, Eamonn Keogh, Stefano Lonardi, and Bill Chiu. 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, 2003. ACM, ACM Press.
- [Lin *et al.*2007] Jessica Lin, Eamonn Keogh, Li Wei, and Stefano Lonardi. Experiencing SAX: a novel symbolic representation of time series. *Data Mining and Knowledge Discovery*, 15(2):107–144, apr 2007.
- [Lin *et al.*2012] Jessica Lin, Rohan Khade, and Yuan Li. Rotation-invariant similarity in time series using bag-of-patterns representation. *Journal of Intelligent Information Systems*, 39(2):287–315, 2012.
- [Lines *et al.*2012] Jason Lines, Luke M Davis, Jon Hills, and Anthony Bagnall. 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, 2012.

[Lkhagva and Kawagoe2006] B. Lkhagva and K. Kawagoe. New Time Series Data Representation ESAX for Financial Applications. In *22nd International Conference on Data Engineering Workshops (ICDEW'06)*, pages x115–x115. IEEE, April 2006.

[Lkhagva *et al.*] Battuguldur Lkhagva, Yu Suzuki, and Kyoji Kawagoe. Extended sax: Extension of symbolic aggregate approximation for financial time series data representation. 7.

[Longin *et al.*] JL Longin, M Vasilis, W Qiang, L Rolf, AR Chotirat, and EJ Keogh. Elastic partial matching of time series. In *9th European Conference On Principles And Practice Of Knowledge Discovery In Databases, Porto, Portugal*.

[lss] LSST time series catalog - XLDB Use Cases.

[Lundqvist *et al.*1991] C Lundqvist, A Siösteen, C Blomstrand, B Lind, and M Sullivan. Spinal cord injuries. clinical, functional, and emotional status. *Spine*, 16(1):78–83, 1991.

[Machida *et al.*2013] Moe Machida, Brandon Irwin, and Deborah Feltz. Resilience in competitive athletes with spinal cord injury: the role of sport participation. *Qualitative health research*, 23(8):1054–1065, 2013.

[Mahalakshmi *et al.*2016] G Mahalakshmi, S Sridevi, and S Rajaram. A survey on forecasting of time series data. In *Computing Technologies and Intelligent Data Engineering (ICCTIDE), International Conference on*, pages 1–8. IEEE, 2016.

[Malinowski *et al.*2013] Simon Malinowski, Thomas Guyet, René Quiniou, and Romain Tavenard. *1d-SAX: A Novel Symbolic Representation for Time Series*, volume 8207 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[Marszalek and Burczynski] A. Marszalek and T. Burczynski. Modeling and forecasting financial time series with ordered fuzzy candlesticks. 273:144–155.

[Marszalek and Burczynski2014] A. Marszalek and T. Burczynski. Modeling and forecasting financial time series with ordered fuzzy candlesticks. *Information Sciences*, 273:144–155, 2014.

[Marteau2008] Pierre-François Marteau. Time warp edit distance. *arXiv preprint arXiv:0802.3522*, 2008.

- [Marteau2009] Pierre-François Marteau. Time warp edit distance with stiffness adjustment for time series matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):306–318, 2009.
- [Massee *et al.*1992] LC Massee, M Lamontagne, and MD O’riain. Biomechanical analysis of wheelchair propulsion for various seating positions. *Journal of rehabilitation research and development*, 29(3):12–28, 1992.
- [Matthies *et al.*1989] Larry Matthies, Takeo Kanade, and Richard Szeliski. Kalman filter-based algorithms for estimating depth from image sequences. *International Journal of Computer Vision*, 3(3):209–238, 1989.
- [Merugu and Ghosh2003] Srujana Merugu and Joydeep Ghosh. Privacy-preserving distributed clustering using generative models. In *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, pages 211–218. IEEE, 2003.
- [Murthy and Sarangi] Karin Murthy and Smruti Ranjan Sarangi. Generalized notion of similarities between uncertain time series. US Patent 8,407,221.
- [Myers *et al.*] Cory Myers, Lawrence R. Rabiner, and Aaron E. Rosenberg. Performance tradeoffs in dynamic time warping algorithms for isolated word recognition. 28(6):623–635.
- [Myers *et al.*1980a] Cory Myers, Lawrence Rabiner, and Aaron Rosenberg. Performance tradeoffs in dynamic time warping algorithms for isolated word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(6):623–635, 1980.
- [Myers *et al.*1980b] Cory Myers, Lawrence R. Rabiner, and Aaron E. Rosenberg. Performance tradeoffs in dynamic time warping algorithms for isolated word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(6):623–635, 1980.
- [Needs and The2016] Wheelchair Needs and In The. Analysis of Wheelchair Need - Wheelchair Foundation, 2016.
- [Newsam *et al.*1996] Craig J Newsam, Sara J Mulroy, JoAnne K Gronley, Ernest L Bontrager, and Jacquelín Perry. 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, 1996.
- [Orang and Shiria] Mahsa Orang and Nematollaah Shiri. 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 Shirib] Mahsa Orang and Nematollaah Shiri. Correlation analysis techniques for uncertain time series. 50(1):79–116.
- [Orang and Shiric] Mahsa Orang and Nematollaah Shiri. 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.
- [Pan and Chen1999] Victor Y Pan and Zhao Q Chen. The complexity of the matrix eigenproblem. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 507–516. ACM, 1999.
- [Panuccio *et al.*2002] Antonello Panuccio, Manuele Bicego, and Vittorio Murino. 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, 2002.
- [Papadimitriou *et al.*] Spiros Papadimitriou, Feifei Li, George Kolios, and Philip S Yu. Time series compressibility and privacy. In *Proceedings of the 33rd international conference on Very large data bases*, pages 459–470. VLDB Endowment.
- [Papapetrou *et al.*2011] Panagiotis Papapetrou, Vassilis Athitsos, Michalis Potamias, George Kolios, and Dimitrios Gunopulos. Embedding-based subsequence matching in time-series databases. *ACM Transactions on Database Systems*, 36(3):1–39, August 2011.
- [Parsons *et al.*2004] Lance Parsons, Ehtesham Haque, and Huan Liu. Subspace clustering for high dimensional data: a review. *Acm Sigkdd Explorations Newsletter*, 6(1):90–105, 2004.
- [Patra *et al.*] Bidyut Kr Patra, Raimo Launonen, Ville Ollikainen, and Sukumar Nandi. A new similarity measure using bhattacharyya coefficient for collaborative filtering in sparse data. 82:163–177.
- [Patterson and Draper1997] Patrick Patterson and Scott Draper. Selected comparisons between experienced and non-experienced individuals during manual wheelchair propulsion. *Biomedical sciences instrumentation*, 33:477–481, 1997.
- [Pentland and Twomey1991] WE Pentland and LT Twomey. The weight-bearing upper extremity in women with long term paraplegia. *Spinal Cord*, 29(8):521, 1991.
- [Petitjean *et al.*2011] François Petitjean, Alain Ketterlin, and Pierre Gançarski. A global averaging method for dynamic time warping, with applications to clustering. *Pattern Recognition*, 44(3):678–693, March 2011.

- [Petitjean *et al.*2014] François Petitjean, Germain Forestier, Geoffrey I. Webb, Ann E. Nicholson, Yanping Chen, and Eamonn Keogh. Dynamic Time Warping Averaging of Time Series Allows Faster and More Accurate Classification. In *2014 IEEE International Conference on Data Mining*, pages 470–479. IEEE, December 2014.
- [Pollock2007] David Stephen Pollock. Wiener–kolmogorov filtering, frequency-selective filtering and polynomial regression. *Econometric Theory*, 23(1):71–88, 2007.
- [Price and Campbell1999] MJ Price and IG Campbell. Thermoregulatory and physiological responses of wheelchair athletes to prolonged arm crank and wheelchair exercise. *International journal of sports medicine*, 20(07):457–463, 1999.
- [Rakthanmanon and Keogh2013] Thanawin Rakthanmanon and Eamonn Keogh. 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, 2013.
- [Rakthanmanon *et al.*2012a] Thanawin Rakthanmanon, Bilson Campana, Abdullah Mueen, Gustavo Batista, Brandon Westover, Qiang Zhu, Jesin Zakaria, and Eamonn Keogh. Searching and mining trillions of time series subsequences under dynamic time warping. *18th ACM SIGKDD*, pages 262–270, 2012.
- [Rakthanmanon *et al.*2012b] Thanawin Rakthanmanon, Bilson Campana, Abdullah Mueen, Gustavo Batista, Brandon Westover, Qiang Zhu, Jesin Zakaria, and Eamonn Keogh. 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, 2012.
- [Ralanamahatana *et al.*2005] Chotirat Ann Ralanamahatana, Jessica Lin, Dimitrios Gunopoulos, Eamonn Keogh, Michail Vlachos, and Gautam Das. Mining time series data. In *Data mining and knowledge discovery handbook*, pages 1069–1103. Springer, 2005.
- [Rand] William M Rand. Objective criteria for the evaluation of clustering methods. 66(336):846–850.
- [Rani and Sikka2012] Sangeeta Rani and Geeta Sikka. Recent techniques of clustering of time series data: a survey. *International Journal of Computer Applications*, 52(15), 2012.
- [Ratanamahatana and Keogh] Chotirat Ann Ratanamahatana and Eamonn Keogh. *Making Time-series Classification More Accurate Using Learned Constraints*, pages 11–22.

- [Ratanamahatana and Keogh2004] Chotirat Ann Ratanamahatana and Eamonn Keogh. *Making Time-series Classification More Accurate Using Learned Constraints*, pages 11–22. 2004.
- [Rehfeld and Kurths] K. Rehfeld and J. Kurths. Similarity estimators for irregular and age-uncertain time series. 10(1):107–122.
- [Rizvandi *et al.*] Nikzad Babaii Rizvandi, Javid Taheri, Reza Moraveji, and Albert Y. Zomaya. A study on using uncertain time series matching algorithms for MapReduce applications. 25(12):1699–1718.
- [Roddick and Spiliopoulou2002] John F Roddick and Myra Spiliopoulou. A survey of temporal knowledge discovery paradigms and methods. *IEEE Transactions on Knowledge and data engineering*, 14(4):750–767, 2002.
- [Rodgers *et al.*1994] Mary M Rodgers, G William Gayle, Stephen F Figoni, Masuo Kobayashi, Junghsen Lieh, and Roger M Glaser. Biomechanics of wheelchair propulsion during fatigue. *Archives of physical medicine and rehabilitation*, 75(1):85–93, 1994.
- [Ruggles *et al.*1994] Deborah L Ruggles, Thomas Cahalan, and Kai-Nan An. Biomechanics of wheelchair propulsion by able-bodied subjects. *Archives of physical medicine and rehabilitation*, 75(5):540–544, 1994.
- [SAC] Projet SACR-FRM (Approches de la Sociologie, de la Biomécanique et de l’Intelligence Artificielle...) | ANR - Agence Nationale de la Recherche.
- [Sakoe and Chiba1978a] H Sakoe and S Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(1):43–49, February 1978.
- [Sakoe and Chiba1978b] H Sakoe and S Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 26(1):43–49, 1978.
- [Sakoe and Chiba1978c] Hiroaki Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal processing*, 26(1):43–49, 1978.
- [Salvador and Chan2007] Stan Salvador and Philip Chan. Toward accurate dynamic time warping in linear time and space. *intelligent data analysis*, 11(5):561–580, 2007.
- [Schäfer2015] Patrick Schäfer. The boss is concerned with time series classification in the presence of noise. *Data Mining and Knowledge Discovery*, 29(6):1505–1530, 2015.

- [Senin and Malinchik2013] Pavel Senin and Sergey Malinchik. 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, 2013.
- [Senin *et al.*] Pavel Senin, Jessica Lin, Xing Wang, Tim Oates, Sunil Gandhi, Arnold P Boedihardjo, Crystal Chen, Susan Frankenstein, and Manfred Lerner. 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.
- [Shannon2001] C. E. Shannon. A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3, jan 2001.
- [Shatkay and Zdonik1996] Hagit Shatkay and Stanley B Zdonik. Approximate queries and representations for large data sequences. In *Data Engineering, 1996. Proceedings of the Twelfth International Conference on*, pages 536–545. IEEE, 1996.
- [Siyou Fotso *et al.*] Vanel Steve Siyou Fotso, Engelbert Mephu Nguifo, and Philippe Vaslin. Comparison of classification algorithms to fdtw. <http://fc.isima.fr/~siyou/fdtw>.
- [Siyou Fotso *et al.*2015] Vanel Steve Siyou Fotso, Engelbert Mephu Nguifo, and Philippe Vaslin. Symbolic representation of cyclic time series: application to biomechanics. In *ICML'15 Workshop on Constructive Machine Learning*, July 2015.
- [Stefan *et al.*2013] Alexandra Stefan, Vassilis Athitsos, and Gautam Das. The move-split-merge metric for time series. *IEEE transactions on Knowledge and Data Engineering*, 25(6):1425–1438, 2013.
- [Stenberg *et al.*1967] Jesper Stenberg, PO Astrand, B Ekblom, J Royce, and B Saltin. Hemodynamic response to work with different muscle groups, sitting and supine. *Journal of Applied Physiology*, 22(1):61–70, 1967.
- [Stoboy *et al.*1971] Hans Stoboy, Bryan Wilson Rich, and Mathew Lee. Workload and energy expenditure during wheelchair propelling. *Spinal Cord*, 8(4):223, 1971.
- [Sun *et al.*2014] Youqiang Sun, Jiuyong Li, Jixue Liu, Bingyu Sun, and Christopher Chow. An improvement of symbolic aggregate approximation distance measure for time series. *Neurocomputing*, 138:189–198, August 2014.
- [Theisen *et al.*1996] Daniel Theisen, Marc Francaux, A Fay, and Xavier Sturbois. 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, 1996.

[Ulanova *et al.*] Liudmila Ulanova, Nurjahan Begum, and Eamonn Keogh. 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] Rienk MA van der Slikke, Daan JJ Bregman, Monique AM Berger, Annemarie MH de Witte, and Dirk-Jan (H) EJ Veeger. 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, 2017.

[van der Woude and de Groot 2005] Lucas HV van der Woude and Sonja de Groot. Wheelchair propulsion: a straining form of ambulation. *Indian Journal of Medical Research*, 121(6):719, 2005.

[Van der Woude *et al.* 2006] Lucas HV Van der Woude, Sonja de Groot, and Thomas WJ Janssen. Manual wheelchairs: research and innovation in rehabilitation, sports, daily life and health. *Medical Engineering and Physics*, 28(9):905–915, 2006.

[Van Wijk and Van Selow 1999] Jarke J Van Wijk and Edward R Van Selow. 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, 1999.

[Vanlandewijck *et al.* 1999] YC Vanlandewijck, DJ Daly, and DM Theisen. Field test evaluation of aerobic, anaerobic, and wheelchair basketball skill performances. *International journal of sports medicine*, 20(08):548–554, 1999.

[Vasimalla 2017] Kumar Vasimalla. A Survey on Time Series Data Mining. *International Journal of Innovative Research in Computer and Communication Engineering*, 2(5):170–179, 2017.

[Vegter *et al.* 2014] Riemer J K Vegter, Claudine J Lamoth, Sonja de Groot, Dirkjan H E J Veeger, and Lucas H V van der Woude. Inter-individual differences in the initial 80 minutes of motor learning of handrim wheelchair propulsion. *PloS one*, 9(2):e89729, January 2014.

[Ville and Winance 2006] Isabelle Ville and Myriam Winance. To work or not to work? the occupational trajectories of wheelchair users. *Disability and rehabilitation*, 28(7):423–436, 2006.

[Wand and Jones 1994] Matt P Wand and M Chris Jones. *Kernel smoothing*. Crc Press, 1994.

- [Wang and Zhang2013] Yu-Xiong Wang and Yu-Jin Zhang. Nonnegative matrix factorization: A comprehensive review. *IEEE Transactions on Knowledge and Data Engineering*, 25(6):1336–1353, 2013.
- [Wang *et al.*a] Wei Wang, Guohua Liu, and Dingjia Liu. Chebyshev Similarity Match between Uncertain Time Series. 2015:1–13.
- [Wang *et al.*b] Xiaoyue Wang, Abdullah Mueen, Hui Ding, Goce Trajcevski, Peter Scheuermann, and Eamonn Keogh. Experimental comparison of representation methods and distance measures for time series data. 26(2):275–309.
- [Weber *et al.*2001] Marc Weber, Marc Alexa, and Wolfgang Müller. Visualizing time-series on spirals. In *Infovis*, volume 1, pages 7–14, 2001.
- [Woude *et al.*1986] Lhv Van Der Woude, G DE GROOT, AP Hollander, GJ Van Ingen Schenau, and RH Rozendal. Wheelchair ergonomics and physiological testing of prototypes. *Ergonomics*, 29(12):1561–1573, 1986.
- [Wu *et al.*1996] Daniel Wu, Ambuj Singh, Divyakant Agrawal, Amr El Abbadi, and Terence R. Smith. 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, nov 1996. ACM Press.
- [Yeh *et al.*] Mi-Yen Yeh, Kun-Lung Wu, Philip S Yu, and Ming-Syan Chen. 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 Faloutsos2000] Byoung-Kee Yi and Christos Faloutsos. Fast time sequence indexing for arbitrary l_p norms. VLDB, 2000.
- [Yu *et al.*] Daren Yu, Xiao Yu, Qinghua Hu, Jinfu Liu, and Anqi Wu. Dynamic time warping constraint learning for large margin nearest neighbor classification. 181(13):2787–2796.
- [Yu *et al.*2011] Daren Yu, Xiao Yu, Qinghua Hu, Jinfu Liu, and Anqi Wu. Dynamic time warping constraint learning for large margin nearest neighbor classification. *Information Sciences*, 181(13):2787–2796, 2011.
- [Zakaria *et al.*] Jesin Zakaria, Abdullah Mueen, and Eamonn Keogh. Clustering time series using unsupervised-shapelets. In *Data Mining (ICDM), 2012 IEEE 12th International Conference on*, pages 785–794. IEEE.

- [Zhang *et al.*a] XX Zhang, Q Liu, DM Liu, and W Xie. A survey on anonymity for privacy preserving data mining. In *Information Science and Electronic Engineering: Proceedings of the 3rd International Conference of Electronic Engineering and Information Science (ICEEIS 2016), 4-5 January, 2016, Harbin, China*, pages 343–346. CRC Press.
- [Zhang *et al.*b] Zheng Zhang, Ping Tang, and Rubing Duan. Dynamic time warping under pointwise shape context. 315:88–101.
- [Zhang *et al.*2015] Zheng Zhang, Ping Tang, and Rubing Duan. Dynamic time warping under pointwise shape context. *Information Sciences*, 315:88–101, 2015.
- [Zhang *et al.*2016] Qin Zhang, Jia Wu, Hong Yang, Yingjie Tian, and Chengqi Zhang. Unsupervised feature learning from time series. In *IJCAI*, pages 2322–2328, 2016.
- [Zhao and Itti] Jiaping Zhao and Laurent Itti. shapedtw: shape dynamic time warping.

List of Figures

1.1	WAFT photograph [Langbein and Fehr1993].	9
1.2	Photograph of a wheelchair on a roller ergometer with mechanical braking by friction belt on a flywheel. (Rodgers et al., 1994).	10
1.3	Photograph of an experiment on a conveyor belt (van der Woude et al., 2006)	11
1.4	Photograph of an experiment on a simulator connected to a flywheel (Brattgard et al., 1970)	12
1.5	Presentation of the different sensors of wheelchair field-ergometer	13
1.6	Balance of forces applied to a manual wheelchair during its use; the analysis of the movement of the subject-chair system has been reduced to that of its center of gravity	14
1.7	Balance of forces applied to a manual wheelchair during its use; the analysis of the movement of the subject-chair system has been reduced to that of its center of gravity	15
3.1	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, there is a loss of information and the accuracy is reduced. However, considering all the points in the time series, we also do not obtain maximum accuracy due to the presence of noise or singularities [Keogh and Pazzani2001a] in the data.	37
3.2	IDDTW operating principle. Depth represents approximation levels, A represents approximate distance and B is best_so_far [Chu et al.].	40
3.3	Visual comparison of two time series from the two classes of the coffee dataset. Left: the original time series, right: representation using PAA with 88 segments	48
3.4	Comparison of the number of tested values of the parameter number of segments with the FDTW and IDDTW. x-axis datasets are sorted according to the length of the time series.	53

3.5	Comparison of the execution time of the Brute-force search algorithm, FDTW and IDDTW.	54
3.6	Critical difference diagram for FDTW and 36 other classification algorithms on 6 simulated datasets. FDTW is ranked 3rd / 37 algorithms	55
3.7	Eight types of time series corresponding to the vocabulary of 8 gestures.	55
4.1	Multiset-based model of uncertain time series	59
4.2	PDF-based model of uncertain time series	60
4.3	Geometric representation of loco similarity.	66
4.4	Comparison between the execution time of the Euclidean distance and that of the FOTS score as a function of the length of the time series	72
5.1	Cyclic time series form manual wheelchair locomotion	76
5.2	Properties of a cycle	77
5.3	Piecewise aggregate approximation of a cyclic time series	78
5.4	Symbolic Aggregate approXimation of a cyclic time series	78
5.5	Extended Symbolic Aggregate approXimation of a cyclic time series .	79
5.6	Trend Symbolic Aggregate approXimation of a cyclic time series . . .	79
5.7	Properties of a cycle	80
5.8	Threshold for the segmentation of cyclic time series	80
5.9	Segmentation	81
5.10	Some properties are computed on each cycle	81
5.11	Classification of cycles based on properties	82
5.12	Symbolic representation of cyclic time series	84
6.1	title	86
A.1	Bhattacharyya	90
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.	94
B.2	This figure shows the first 100 points of the first time series of the fordA dataset available in the UCR [Chen <i>et al.</i> 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. . .	95

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	107
--	-----

List of Tables

2.1	Temporal-Proximity-Based Clustering Approach	29
2.2	Representation-Based Clustering Approach Paper	30
2.3	Model-Based Clustering Approach	31
3.1	Classification errors associated with the number of segments \mathbf{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.	50
3.2	Comparison of generalization errors. In italics , the smallest generalization error. In bold , the smallest generalization error between IDDTW and FDTW. \mathbf{N} is the number of segments selected and ℓ is the number of data points in a segment ($\ell = \lfloor \frac{n}{N} \rfloor$).	52
4.1	Datasets	70
4.2	Comparison of the Rand Index of SUSH (RI_SUSH) and FOTS-SUSH (RI_FOTS). The best Rand Index is in bold	71
4.3	Comparison between k-Shape, USLM and FOTS-UShapelet	72
5.1	Average vectors of the properties of classes (A, B, C, D, E) used for the symbolic representation of the axial moment (Mz) SAX-P takes into account the surface under the push, the time-push and the time-cycle.	83
5.2	Strings of characters obtained with SAX-P method on times series of axial moments applied by the three subjects on right and left rear wheels of an instrumented MWC during their second 10-m run.	83
B.1	85 UCR datasets used for experimental validation. The full list is available here [Chen <i>et al.</i> 2015]	100

Contents

Summary	viii
Introduction	3
I Background and related works	5
1 Analysis of the wheelchair locomotion	7
1.1 Introduction	7
1.2 The problem of locomotion manual wheelchair locomotion	7
1.3 Tools to evaluate manual wheelchair locomotion	8
1.3.1 Crank Ergometers	8
1.3.2 Roller Ergometers	9
1.3.3 Treadmill	10
1.3.4 Wheelchair simulators	11
1.3.5 Wheelchair Field-Ergometer	11
1.4 Knowledge discovery on wheelchair time series	12
1.5 Conclusion	15
2 Knowledge discovery on time series	17
2.1 Introduction	17
2.2 Preprocessing of time series	17
2.2.1 Denoising time series	17
2.2.2 Reducing uncertainty	19
2.2.3 Dimensionality reduction	19
2.3 Similarity Measures	20
2.3.1 Time-Series Invariances	21
2.3.2 Categories of time series similarity function	22
2.4 Datamining task on time series	23
2.5 CONCLUSIONS	32

II Contributions	33
3 Preprocessing of time series	35
3.1 Introduction	35
3.1.1 Why the use of PAA can improve alignment with Dynamic Time Warping	36
3.1.2 The problem of choosing a suitable segment number for PAA	37
3.1.3 Summary of Contributions	37
3.2 Background and related works	38
3.2.1 Dynamic Time Warping algorithm.	39
3.2.2 Iterative Deepening Dynamic Time Warping	39
3.3 GRASP based heuristic	41
3.3.1 Evaluation procedures for the compression quality	41
3.3.2 Problem definition.	42
3.3.3 Brute-force search.	42
3.3.4 Greedy Randomized Adaptive Search Procedures	43
3.3.5 Parameter free heuristic.	43
3.4 Experiment and results	46
3.4.1 Case studies	46
3.5 Conclusion and perspective	54
4 Uncertain time series u-shapelet discovery	57
4.1 Introduction	57
4.1.1 U-shapelets algorithm for clustering Uncertain Time Series	58
4.1.2 Uncertainty and u-shapelets discovery issue	58
4.1.3 Summary of contributions	59
4.2 Background	59
4.2.1 Related work	59
4.2.2 Review of u-shapelets	60
4.2.3 Review on uncertain similarity functions	61
4.3 Our Approach	66
4.3.1 Dissimilarity function	66
4.3.2 Scalable u-shapelets Algorithm with FOTS score	67
4.4 Experimental Evaluation	68
4.4.1 Clustering with u-shapelets	68
4.4.2 Evaluation Metric	69
4.4.3 Comparison with u-shapelet	70
4.4.4 Comparison with k-Shape and USLM	70
4.4.5 Discussion	71
4.5 Conclusion and Future Work	72

5 SAX-P	75
5.1 Introduction	75
5.2 SAX-P	78
5.2.1 Segmentation of cyclic time series	79
5.2.2 From cycles to letters	80
5.3 Application to manual wheelchair locomotion	82
5.4 Conclusion	84
6 Application to manual wheelchair locomotion	85
6.1 Introduction	85
6.2 Description of the dataset	85
6.2.1 Torque sensor	87
6.2.2 Characteristic properties of the data	87
6.3 Analysis based on propulsion technique	87
6.4 Analysis based on propulsion capabilities	87
6.5 Propulsion technique versus propulsion capabilities	87
6.6 Conclusion	87
Conclusion	87
A Hellinger Based Distance for Uncertain time series	89
A.1 DUST	89
A.2 Hellinger Based Distance	90
B An optimal approach to time series segmentation: Application to the supervised classification	93
B.1 Introduction	93
B.2 Granularity of time series segments	94
B.2.1 Notations and definitions	94
B.2.2 information theory and minimum number of segments	95
B.2.3 Notations and definitions	95
B.2.4 Minimize the squared error to choose the number of segments	96
B.2.5 Dynamic Time Warping Algorithm and Comparison of Representations compact	98
B.3 Results and Discussion	99
B.3.1 Datasets	99
B.3.2 Comparison of algorithm performance	101
B.4 Conclusion	106
Index	109
Glossaire	109
Liste des abbreviations, des sigles et des symboles	109

References	128
Table des figures	131
Liste des tableaux	134
Table des matieres	138