Gait Assessment of Patients with Parkinson's Disease using Inertial Sensors and Non-Linear Dynamics Features

Paula Andrea Pérez Toro
Bachelor student in Electronics Engineering
Research Group on Applied Telecommunications
GITA

Department of Electronics and Telecommunication Engineering
Universidad de Antioquia
Colombia
October, 2018

Gait Assessment of Patients with Parkinson's Disease using Inertial Sensors and Non-Linear Dynamics Features

Paula Andrea Pérez Toro
Bachelor student in Electronics Engineering

Document in partial fulfillment of the requirements for the degree of: electronics engineering

Director:
Juan Rafael Orozco Arroyave
Ph.D. in Electronics Engineering and Computer Science
Advisor:
Juan Camilo Vásquez Correa
MSc. in Telecommunications Engineering and Computer Science

Research line:
Signal digital processing and pattern analysis
Research Group on Applied Telecommunications
GITA

Department of Electronics and Telecommunication Engineering
Universidad de Antioquia
Colombia
October, 2018

Acknowledgements

First of all, I would like to express my gratitude to the advisor of this study Prof. Juan Rafael Orozco Arroyave and also the co-advisor and big friend of mine Juan Camilo Vásquez Correa, who supported me in this learning process, spending their time and dedication for making it possible.

To the pattern recognition team of GITA in the University of Antioquia: Tomás Arias, Surley, Daniel Escobar, Nicanor García, Luis Felipe Gómez, Felipe López, Felipe Parra, Cristian Rios among others, for supporting me in this process, giving me their friendship and their academic and personal support. Also, my friends Sebastián Arango, David G. and Juan David Robles who urged me not to give up.

To my mother María Lucía Pérez Toro, she made that all of this was possible with her invaluable support, unconditional love and understanding. She made her big effort for giving me the chances to study and she taught me to be a good person.

To “Grupo de Investigación en Telecomunicaciones Aplicadas” GITA, where i developed my research. To the CODI project from University of Antioquia grant #PR18-2-09 for supporting the under-graduated thesis projects and to the Friedrich Alexander Universität Erlangen-Nürnberg for the eGait system to the gait acquisition and for supporting our research group in several projects. Also, Fundalizanza, a Parkinson’s foundation in Medellín that helped me to obtain the necessary data to perform this research.

Contents

List of Figures

List of Tables

Abstract

Parkinson's disease is a neuro-degenerative disorder characterized by different motor symptoms, including several gait impairments. Gait analysis is a suitable tool to support the diagnosis and to monitor the state of the disease. The gait of the patients is mainly evaluated from signals captured with inertial sensors attached to the limbs of the patients, where kinematics features are commonly computed. However, there are non-linear effects of the gait process that cannot be properly characterized with the kinematic features. This study proposes the use of non-linear dynamics features extracted from gait signals obtained from inertial sensors for the automatic detection of the disease. It is considered classical non-linear features such as the correlation dimension, the largest Lyapunov exponent, and the Hurst exponent, among others. In addition we propose a novel non-linear analysis based on applying a Gaussian mixture model to find clusters in Poincaré sections. The non-linear dynamics features are used to discriminate between Parkinson's patients and healthy subjects, and to classify patients in several stages of the disease.

The obtained results point out that it is possible to perform the discrimination between PD patients and healthy subjects with accuracies up to 93%. Classifying patients in several stages of the disease with accuracies up to 65%. As far as we know, this is one of the first studies that considers a full non-linear dynamics analysis to assess the gait impairments of patients with Parkinson's disease.

Chapter 2

Introduction

2.1 Context

Parkinson's disease (PD) is a neuro-degenerative disorder characterized by the progressive loss of dopaminergic neurons in the mid brain [1], which are in charge of controlling movement and emotions. Motor symptoms include lack of coordination, tremor, rigidity and postural instability. Gait impairments appear in most of patients and include freezing, shuffling, and festinating gait. The standard scale to evaluate the neurological state of the patients is the Movement Disorder Society-Unified Parkinson's Disease Rating Scale (MDS-UPDRS) [2]. The third section (MDS-UPDRS-III) of the scale evaluates the deficits in the motor system, besides it contains 14 items dedicated to the lower extremities of the patients. The therapy mainly focuses on treating the symptoms of the patients with individual medication. This medication dose always has to be adjusted according to the current stage of disease.

Gait changes are a hallmark of PD, where the main symptoms include reductions in speed, decreased step length, altered cadence, and increased gait variability. While gait abnormalities are not pronounced in the early stages, their prevalence and severity increases with the disease progression [3]. More than 85% of PD patients develop gait impairments after three years of diagnosis [3]. The potential consequences of gait impairments in PD may include increased disability, increased risk of falls, and reduced quality of life. As the disease progresses, PD patients typically exhibit shuffling gait with a forward-stooped posture and festinating gait. These characteristics make the patient to spend a lot of energy, which causes that the walking routine leads the patients to their maximum metabolic capacity [4]. Gait impairments also include the presence of bradykinesia, rigidity, and postural instability, reducing the quality of life. In addition, PD patients consider mobility and walking limitations to be one of the most impaired aspects of the disease. Patients consistently identify improvement in walking as the most relevant outcome when rating the success of the treatment for the disease [5].

2.2 State of the art

In related studies, the scientific community has shown interest in the gait analysis to evaluate the neurological state of PD patients. Gait analysis has been performed commonly with inertial sensors e.g., accelerometers and gyroscopes attached to the shoes of the patients [6–8].

One of the first studies was performed in [9], where the authors classified gait signals captured from 42 PD patients and 39 healthy control (HC) subjects using the eGait system [10], which consisted of accelerometers and gyroscopes attached to the lateral heel of the shoe. The participants performed several exercises, including the straight walking of 10 meters 4 times ($4\times$ 10), heel-toe tapping, and circling foot movements. The authors computed several spectral and statistical features from the gait signals, including the energy content in different frequency bands, entropy, the root-mean square energy and others. The classification of HC subjects and PD patients was performed with several classifiers. The results indicated that it was possible to classify PD patients vs. HC subjects with accuracies up to 82%. The accuracy improved up to 91% when considering only PD patients in severe state of the disease.

In [7] the authors proposed an algorithm to segment the strides of PD patients based on the dynamic time warping (DTW) algorithm. The authors developed a stride template using gait information from 25 HC subjects. The segmentation algorithm was tested with information from 40 HC subjects, 15 PD patients and 25 geriatric patients, who performed a 40 m straight walking exercise. The proposed approach showed to be accurate to segment the steps of the patients (F-score up to 0.9).

In [11] the authors proposed to evaluate a set of features to perform an early detection and monitoring PD implementing a classification between PD patients and HC subjects by Linear Discriminant Analysis (LDA). The authors considered kinematic features such as step distance, stride time, stance and swing phases and others, and frequency-domain characteristics from the signals such as amplitude, power distribution, frequency dispersion, and others. The database was obtained from force-sensitive resistors [12] in each foot of 93 idiopathic PD patients and 73 HC subjects, this while they walked at their usual, self-selected pace for approximately 2 minutes on level ground. The above was digitized and recorded at 100 samples per second. Accuracies up to 86.9% when using the kinematic feature set.

In [8] the authors considered inertial sensors attached to the chest and to the kneels to evaluate the neurological state of 34 PD patients according to the UPDRS score. The participants performed several exercises, including 20 meters walking, rising from a chair, and foot tapping. The authors computed kinematic features such as the standing time, the stride length, the stride velocity, and others. The regression algorithm was based on K-nearest neighbors (KNN) algorithm to predict the MDS-UPDRS-III score of the patients, based on LA, S2S, and G tasks of it. A Spearman's correlation coefficient (ρ) of 0.60 was

reported for the prediction.

In [13] the authors proposed a statistical transformation called shifted one-dimensional local binary pattern domain to classify gait signals from PD patients and HC subjects collected in different conditions and with a different experimental protocol. The authors considered three different databases for their experiments, forming a dataset with signals collected from 93 PD patients and 73 HC subjects. The data were collected using 8 sensors to measure the pressure of the foot when the subjects walk. The authors computed several statistical features from the available data. The extracted features were transformed with the proposed method, and they were classified using eight different algorithms. The highest accuracy was obtained with a multi-layer perceptron classifier (88.9%).

In [14] the authors classified gait signals obtained from 15 PD patients and 16 HC subjects. The signals were captured with ultra-thin force-sensitive switches placed inside the shoes of the participants. The authors computed several kinematics features such as the stride time, the swing time, the stance time, among others. Afterwards, the features extracted from each foot were used to compute a phase synchronization coefficient and the conditional entropy, with the aim to analyze the gait rhythm fluctuations between both feet. The authors considered several classifiers. The most accurate results were reported with a multi-layer perceptron, where an area under the receiving operating characteristic curve (AUC) of 0.928 was reported.

In previous studies [15] our team computed kinematics features from gait signals captured with the eGait system [10] to evaluate the neurological state of the patients. A Spearman's correlation of up to 0.72 was reported between the MDS-UPDRS-III score of the patients and the predicted values obtained with a Support Vector Regressor (SVR).

Recently in [16], the authors proposed new features to assess gait impairments of PD patients. Those new features were the peak forward acceleration in the loading phase and peak vertical acceleration around heel-strike, which encode the engagement in stride initiation and the hardness of the impact at heel-strike, respectively. The results indicated that the proposed features correlate with the disease progression and the loss of postural agility/stability of the patients.

In [17] the authors aimed to detect freezing of gait (FoG) episodes in PD patients using a deep learning approach. The authors collected data from 21 PD patients with FoG using a waist-placed inertial sensor. The exercises performed by the participants included free walking inside an apartment, walking ten meters outdoors, and rising from a chair. The authors considered a six-layer one-dimensional convolutional neural network (CNN), whose inputs were formed by spectral representations of consecutive time intervals. In total, the input of the CNN consisted of 64 frequency bins obtained from 9 inertial sensors (3-axis accelerometer, gyroscope and magnetometer). The author reported accuracies of up to 90% when detecting FoG episodies.

Although most of the studies considered only kinematics, spectral, or statistical features to assess gait impairments of PD patients, there are some studies that evaluated

the non-linear effects of the walking process of PD patients. For instance, in [18], the authors performed a combined analysis of spectral, statistical, kinematics, and NLD features to characterize gait signals of 14 HC subjects, 10 PD patients and 11 patients with peripheral neuropathy. The exercises consisted on 3 minutes of continuous walking on a treadmill. The NLD features included the largest Lyapunov exponent (LLE), the Lempel Ziv complexity (LZC), several entropy measures, and others. The extracted features were compared the three groups using the Kruskal-Wallis and Mann-Whitney tests. Significant differences were reported in features such as the LZC and the cross-entropy, which indicates that it is possible to automatically disriminate between HC subjects and PD patients using the feature set introduced by the authors.

In [19] the authors classified 13 PD patients, 13 Amyotrophic lateral sclerosis patients and 13 Huntington patients and 13 HC subjects using data obtained from force-sensitive resistors [12]. The authors computed NLD features such as the shannon entropy, the recurrence rate, and recurrence quantification analysis (RQA). The classification was performed with a support Vector Machine (SVM) and a Probabilistic Neural Network to discriminate between patients with the different neuro-degenerative diseases and the HC subjects. The classification followed a leave-one-out cross-validation strategy, which could be slighly optimistic, and report accuracies close to 100%.

2.3 Contributions of this study

Non-linear Dynamics allows us to analyze problems at the physiological level, as in this case people with Parkinson's disease presents deficiencies in the motor system, it can be to analyze fluctuations and changes in gait, trying to get closer the MDS-UPDRS-III scale.

This study proposes the use of several NLD features to model the walking process of PD patients and HC subjects. The features include correlation dimension (CD), LLE, Hurst exponent (HE), LZC, and several entropy measures, which have proved to be accurate for the NLD analysis of PD [18, 20, 21]. In addition, we propose a new set of NLD features to model the dispersion of Poincaré sections using Gaussian mixture models (GMM), which allow to represent the Poincare sections as a probability distribution.

The extracted features are used to discriminate between PD patients and HC subjects; and to classify PD patients in several stages of the disease. The classification is performed with three different algorithms: KNN, SVM and Random Forest (RF). Additionally, we aim to predict the neurological state of the patient according to the MDS-UPDRS-III score using an SVR. The algorithms were implemented in Matlab and Python. The results of the proposed approach indicate that it is possible to classify PD patients and HC subjects with accuracies up to 86.7%, and to discriminate between PD patients in several stages of the disease with accuracies up to 65.2%. To the best of our knowledge, this

is one of the first studies that consider only NLD features and includes Poincaré maps and their probabilistic representation (based on GMM-UBM) to characterize the walking process of PD patients.

2.4 Hypothesis

Gait signals collected with inertial sensors help in the assessment of the neurological state of patients with PD in different stages of the disease (low, intermediate, and severe).

2.5 Objectives

2.5.1 General Objective

To develop a methodology based on gait analysis and pattern recognition techniques, to perform the automatic classification and evaluation of the neurological state of PD patients according to the MDS-UPDRS-III scale [2].

2.5.2 Specific Objectives

1. To model several gait tasks performed by PD and HC subjects using different non-linear dynamics features and probabilistic representations of Poincaré maps.
2. To analyze the suitability of different classification and regression methods to model the neurological state of Parkinson's disease patients.
3. To evaluate the developed methodology with several performance metrics.

2.6 Manuscript distribution

This work is divided into six chapters. Chapter one contains the context, state of the art and the contribution of this study. Chapter two describes the measures obtained in the feature extraction step. Chapter three contains the description of the pattern recognition methods to perform the classification and regression. In Chapter four there is a description of the database and the gait tasks. Chapter five includes experiments, results and the discussion of each experiment. Finally, chapter six includes the main conclusions derived from this study.

Chapter 3

Feature extraction

3.1 Nonlinear dynamics.

3.1.1 Phase Space

The embedding process is the first step to reconstruct the state space based on a time series. The most common method is based on the Taken's Theorem [22], which gives the conditions to reconstruct a chaotic dynamic system by means of a observation sequence of the dynamic system state.

In order to analyze attractor properties, measured signals have to be projected onto a suitable phase space. From a single time series $\boldsymbol{S_t}$, a phase space can be constructed as in Equation 3.1.

$$\boldsymbol{S_t} = \left\{ \left[s_t, s_{t-\tau}, ...s_{t-(m-1)\tau} \right] \right\} \tag{3.1}$$

Where $i = 1, 2...N_m$

$$\boldsymbol{S_t} = \left\{ \begin{array}{cccccc} s_1 & s_{1-\tau} & s_{1-2\tau} & \cdots & s_{1-(m-2)\tau} & s_{1-(m-1)\tau} \\ s_2 & s_{2-\tau} & s_{2-2\tau} & \cdots & s_{2-(m-2)\tau} & s_{2-(m-1)\tau} \\ s_3 & s_{3-\tau} & s_{3-2\tau} & \cdots & s_{3-(m-2)\tau} & s_{3-(m-1)\tau} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ s_{N_m-1} & s_{N_m-1-\tau} & s_{N_m-1-2\tau} & \cdots & s_{N_m-1-(m-2)\tau} & s_{N_m-1-(m-1)\tau} \\ s_{N_m} & s_{N_m-\tau} & s_{N_m-2\tau} & \cdots & s_{N_m-(m-2)\tau} & s_{N_m-(m-1)\tau} \end{array} \right\} \tag{3.2}$$

Thus, a point in the reconstructed phase space is given by m "delay-coordinates" and $N_m = N - (m-1)\tau$ is the reconstructed vector length. The selection of the delay-time τ and the embedding dimension m is somewhat arbitrary, but attractor dimensions should not depend on τ and m.

To compute these two embedding parameters (m and τ), we need to consider the following:

- **Embedding dimension (m)**

  The embedding dimension is defined as the smallest dimension that requires an object to be embedded. It means, it is the minimum dimension of the phase space in which it will describe its behaviour. If the embedding dimension is higher the time series representation will be influenced by noise, but in a smaller dimension the phase space reconstruction do not reflect the original system dynamics [23].

  To estimate the embedding dimension we used the method of the false neighbors, which was proposed by M. Kennel [24].The idea is to search points of the data set which are neighbors in an embedding space but not necessarily in the time series [25].

  The nearest neighbor to the vector $\boldsymbol{S_t}$ in the phase space is defined as in Equation 3.3.

$$\boldsymbol{S_t^N} = \left\{[s_t^N, s_{t-\tau}^N, ...s_{t-(m-1)\tau}^N]\right\} \tag{3.3}$$

  And the distance between the vectors $\boldsymbol{S_t}$ and $\boldsymbol{S_t^N}$ shown in Equation 3.4.

$$R_{m_n}^2 = \sum_{i=0}^{m_0}(s_{t-i\tau} - s_{t-i\tau}^N)^2 \tag{3.4}$$

  Then, the reconstructed phase space in $m_0 + 1$:

$$R_{m+1_t}^2 = \sum_{i=0}^{m_0+1}(s_{t-i\tau} - s_{t-i\tau}^N)^2 = R_{m_n}^2 + (s_{t-(m_0+1)\tau} - s_{t-(m_0+1)\tau}^N)^2 \tag{3.5}$$

  It is considered a false neighbor when R_{m_t} is smaller than R_{m+1_t} and the embedding dimension must be increased. To find a false neighbor, it is performed a comparison with a threshold level (R_{NN}) in Equation 3.6, where the embedding dimension is right when the results tends to zero.

$$\sqrt{\frac{R_{(m_0+1)_t}^2 - R_{(m_0)_t}^2}{R_{(m_0)_t}^2}} \geq R_{NN} \tag{3.6}$$

- **Time delay (τ)**

  The Taken's theorem does not provide enough information regarding the time delay to complete the phase space reconstruction. The known methods to search this value (τ)) are the first zero in the Auto Correlation Function (FZA) and mutual information.
  To estimate τ, and in general to reconstruct any attractor or any nonlinear dynamic representation, we need quasi-periodic signals [26]. Speech signals which are not quasi-periodic, are divided into small segments of duration between 40 to 100 milliseconds [26]; however, gait signals are quasi-periodic signals and can be used as they are.

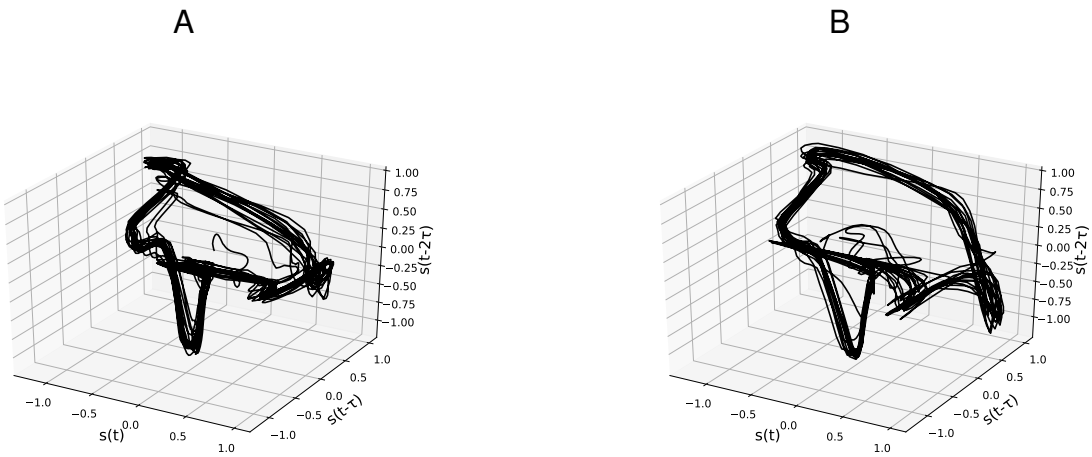

Figure 3.1: Phase space from gait signal produced by: A: YHC subject, B: EHC subject

Figures 3.1 and 3.2 show the phase space obtained from gait signals corresponding to 20 meters walking with a stop at 10 meters from different subjects. Note that the phase space for the HC subjects exhibit well defined trajectories and a clear recurrence. Conversely the trajectories of the phase space for the patients are more dispersed, especially when the disease state of the patients is severe. Several NLD features can be computed from the phase space to assess the complexity and stability of the walking process.

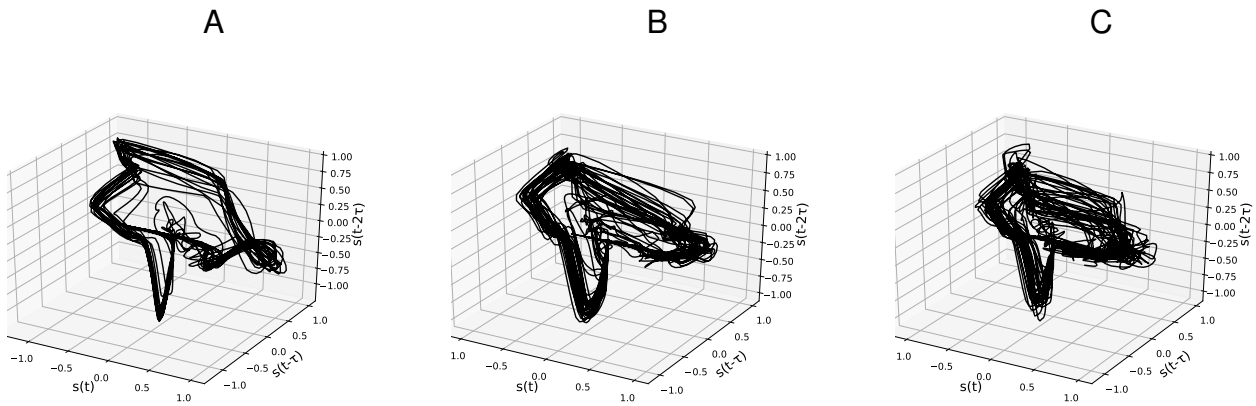

Figure 3.2: Phase space from gait signal produced by: A: PD patient in low state of the disease (MDS-UPDRS-III=6 for lower limbs), B: PD patient in intermediate state of the disease (MDS-UPDRS-III=13 for lower limbs). C: PD patient in severe state of the disease (MDS-UPDRS-III=34 for lower limbs)

3.1.2 Correlation Dimension (CD)

This feature establishes a measure over the exact space that is occupied by an attractor. The computation of CD starts with the estimation of the correlation sum is a probability function where the possible cases are the points that are contained in a hyper-sphere of radius ε. This sum is defined for a set of points x_n using Equation 3.7. This notion was introduced by Grassberger & Procaccia [27].

$$C(\epsilon) = \lim_{n\to\infty} \frac{1}{N(N-1)} \sum_{i=1}^{N} \sum_{j=i+1}^{N} \theta(\epsilon - |x_i - x_j|) \tag{3.7}$$

Where θ is the Heaviside step function, $C(\varepsilon)$ counts the distances between x_i and x_j lower than the threshold ε, and N is the number of embedded points. For a small ϵ value, it can be demonstrated as in Equation 3.8.

$$C(\epsilon) = \lim_{\epsilon\to 0} \epsilon^{CD} \tag{3.8}$$

Where CD is the correlation dimension. A good estimation of CD guarantees that the embedded dimension is almost $m = 2CD + 1$.

To compute the CD, a linear regression of $\ln(C(\varepsilon))$ vs. $\ln(\varepsilon)$ is performed. The slope of the resultant line for a small ε value corresponds to CD [25] as in Equation 3.9.

$$CD = \lim_{\epsilon \to 0} \frac{ln(C(\epsilon))}{ln(\epsilon)} \tag{3.9}$$

3.1.3 Largest Lyapunov Exponent (LLE)

The Largest Lyapunov Exponent (LLE) measures the sensitivity to initial conditions of the signal according to the rate at which the nearby trajectories of the phase space converge or diverge. This feature gives information about stability properties of the time-series, which implies that a small perturbation introduced to the system at any period, makes its behavior unpredictable. There are as many Lyapunov exponents as the dimension of the state space of the system. The exponent of interest for us in this work is the largest one.

LLE quantifies the exponential divergence of the neighbor paths in a phase space, i.e. it measures the degree of aperiodicity in a given signal. The estimation of this process begins with the reconstruction of the phase space using the Taken's theorem [22].

After the reconstruction of the attractor, the nearest neighbor of each point in the trajectory is located. The nearest neighbor $x_{\widehat{j}}$ minimizes the Euclidean distance from the reference point x_j according to Equation 3.10, where $d_j(0)$ is the initial divergence between x_j and $x_{\widehat{j}}$, and || || is the Euclidean distance between two points.

$$d_j(0) = \min||x_j - x_{\widehat{j}}|| \tag{3.10}$$

To guarantee different phase space trajectories between the neighbors paths, they have to be separated a distance larger than the average period of the signal, as in Equation 3.11.

$$|j - \widehat{j}| > \text{Average period} \tag{3.11}$$

The LLE is estimated as the mean separation rate between the nearest neighbors, according to the Oseledec theorem [28], which is expressed in Equation 3.12 assuming that the j-th pair of neighbors diverges, where λ corresponds to the LLE, $d(t)$ is the mean divergence in an instant t and A is a constant for normalization.

$$d_j(i) = A_j e^{\lambda(i\Delta t)} \tag{3.12}$$

When the logarithm function is evaluated in both sides of the previous equation we can obtain the equation that represent parallel lines with slopes λ [23].

If $\lambda < 0$ the trajectories converge in time and the dynamic system is not sensitive to the initial conditions. If λ is positive, the distances between neighbors trajectories will grow exponentially in time and the system will show sensitive dependence to initial conditions. $\lambda = 0$ indicates that the system exhibits Lyapunov stability [29].

3.1.4 Hurst Exponent (HE)

Hurst exponent is used to evaluate the presence of the long-term dependency and its degree in time series. The Hurst exponent is a smoothness measuring of a fractal time-series based on asymptotics behavior of the rescaled range of the process and the fractal dimension is defined as follows:

$$D_F = D_T - H \tag{3.13}$$

Where D_T is the topological dimension (two for temporal series) and H is the Hurst exponent. This exponent also quantifies time series dynamic and allows to weigh until the data points can be represented by a Brownian motion (The Brownian motion of a particle is due to random collisions and allows modeling phenomena of diffusion and aggregation in various disciplines. The derivative of Brownian noise is white noise).

- $H = 0.5$ the series behavior is like a Brownian motion.
- $0.5 < H < 1$ the time serie has a smooth dynamic, that indicates that past tends persists in the future.
- $0 < H < 0.5$ the series is characterized to have higher high frequency content and this reflects that past trends tend to be inverse in the future [30].

Hurst quantifies the chaotic dynamic of the system in analysis. The Hurst exponent is calculated by the follow empirical relation:

$$T^H = \frac{R}{S} \tag{3.14}$$

Where T is the duration of the sample of data and R/S is the corresponding value of rescaled range. H is estimated as the logarithmic representation slope as in Equation 3.15.

$$H = \frac{logR/S}{logT} \tag{3.15}$$

3.1.5 Lempel Ziv complexity (LZ)

This feature is related to the number of different patterns of a given binary sequence. It reflects the order that is retained in a one-dimensional temporal pattern or in a string of n symbols. The algorithm is based on the original string reconstruction through copy operations and symbols insertions in the new string. The time-series $x(t)$ have to be transformed into a symbolic sequence

$P = \{s(1), s(2), ..., s(t)\}$ to compute LZC. Related studies have shown that binary codification is suitable for physiologic time-series [31,32]. To perform the codification is necessary to define a threshold TH in Equation 3.16. LZC ranges from 0 (deterministic sequence) to 1 (random sequence) [32].

$$s(t) = \begin{cases} 0 & if\ x(t) < \mathrm{TH} \\ 1 & if\ x(t) \geqslant \mathrm{TH} \end{cases} \tag{3.16}$$

With each new value the complexity counter $c(n)$ is increased. This process has the following steps [33]:

1. Definition:
   - S,Q: sequences of P.
   - SQ: concatenation of S and Q.
   - SQπ: SQ sequence after remove the last character.
   - v(SQπ): vocabulary that contains all different sequences of SQπ.
2. Variable initialization:
   - $c(t) = 1$
   - $S = s(1)$
   - $Q = s(2)$
   - $SQ\pi = s(1)$
3. Usually, S and Q are defined as $S = \{s(1), s(2), ..., s(r)\}$ and $Q = s(r+1)$ respectively, where r is the analyzed symbol index. Whether Q belongs to the dictionary $v(SQ\pi)$, with $SQ\pi = \{s(1), s(2), ..., s(r)\}$, so Q is a sub-sequence.
4. The sequence Q is updated $Q = \{s(r+1), s(r+2)\}$ and is checked whether it belongs to $v(SQ\pi)$ or not.
5. The above steps are repeated until Q does not belong to $v(SQ\pi)$, then just we can know $Q = \{s(r+1), s(r+2), ..., s(r+i-1)\}$ is not a sub-sequence of $SQ\pi$ and is increased $c(t)$.
6. The sequences S and Q are updated $S = \{s(1), s(2), ..., s(r+i)\}$ and $Q = \{S(r+i+1)\}$.

7. All of the above steps are repeated until the last character $r = n$. Now, it is applied a normalization on the counter $c(t)$ per $t/(1-\varepsilon_t)\log_\alpha(t)$ as in Equation 3.17, where the number of different symbols in the symbol set is α and ε_t is a very small value.

$$c(t) < \frac{t}{(1-\varepsilon_t)\log_\alpha(t)} \tag{3.17}$$

Then

$$c(t) = \frac{t}{log_\alpha(t)} \equiv b(t) \tag{3.18}$$

And $c(t)$ can be normalized that its objective is to obtain an independent complexity value of time-series length as is shown in Equation 3.19.

$$C(t) < \frac{c(t)}{b(t)} \tag{3.19}$$

3.1.6 Entropy measurements

Entropy is an uncertainty measurement, it quantifies the amount of the disorder of a system. An important reason to extract a numeral value of the entropy from time series is that its inverse is an important time scale for the prediction of a system. Unfortunately, to extract entropies from time series is a difficult task because it is required more data points than dimensions [25]. Then, based on the Kolmogorov-Sinai entropy an straightforward implementation would require box counting. Kolgomorov-Sinai entropy is explained more deeply in Appendice A.

Approximate entropy (ApEn)

Approximate entropy (ApEn) is a measure based on searching similar patterns into a time-series. ApEn is computed as the logarithmic probability of data patterns to be close to each other with longer patterns in the next comparison. ApEn provides a general regularity measure, where in a random signal will have low regularity and thus producing a higher ApEn value [34].

ApEn is typically applied to relatively short and noisy data. Two parameters, m and r, need to be chosen before computing this entropy, where m is the pattern length and r is the effective filter. In addition, after computing the correlation sum C_i defined by Equation 3.7, where N is the number of points of the embedded time series. The ApEn is computed according to the following procedure.

- $C_i^m(r)$ in Equation 3.20 is defined as relative frequency of similar patterns in a window of length m.

$$C_i^m(r) = \frac{n_{im}}{N - m + 1} \tag{3.20}$$

- Then the logarithm of each $C_i^m(r)$ is calculated and averaged over i, defining as a result in Equation 3.21.

$$\phi^m(r) = \frac{1}{N - m + 1} \sum_{i=1}^{N-m+1} \log C_i^m(r) \tag{3.21}$$

- Let ApEn be the increment of $\phi^m(r)$ between two immediate steps of m (see Equation 3.22).

$$\phi^m(r) - \phi^{m+1}(r) \tag{3.22}$$

Replacing expression 3.21 in 3.22, it is obtained Equation 3.23.

$$\frac{1}{N - m + 1} \sum_{i=1}^{N-m+1} \log C_i^m(r) - \frac{1}{N - (m+1) + 1} \sum_{i=1}^{N-(m+1)+1} \log C_i^{m+1}(r) \tag{3.23}$$

Sample Entropy (SampEn)

The ApEn estimation may be affected by $\log(0)$ in the time-series. To avoid this, each template vector counts itself in the comparisons. In addition, the ApEn is highly dependent of the time-series length, making short time series have a lower estimation than the expected, and also affecting the measure consistency. It means that if a data set has a higher complexity than another one, it should continue to be for all the performed tests [35].

The sample entropy was proposed by Richman et al [36] as a measure capable of dealing with the problems of ApEn. SampEn is defined in Equation 3.24.

$$SampEn(m, r, N) = -log \frac{\phi^m(r)}{\phi^{m+1}(r)} \tag{3.24}$$

Where m, r and N are defined as for the ApEn, $\phi^m(r)$ measures without self-comparisons to avoid the occurrence of $\log(0)$ in the estimation.

ApEn and SampEn with Gaussian kernel

Although the improvement offered by SampEn over ApEn, the validity and accuracy of the estimated regularity is affected by the discontinuity of the Heaviside function. In [37], the authors proposed a solution, which consists of replacing the Heaviside step function with a Gaussian kernel in the estimation of the correlation sum C_i, which is estimated using Equation 3.25 when the kernel is considered and where $|| \, ||_1$ is the L1 norm.

$$C_i^m(r) = \frac{1}{N-m} \sum_{j=1, j \neq i}^{N-m+1} \exp\left(-\frac{(||x(i), x(j)||_1)^2}{10r^2}\right) \tag{3.25}$$

Recurrence Probability Density Entropy (RPDE)

Another entropy measure considered to analyze chaotic and deterministic dynamic of gait signals is the Recurrence Probability Density Entropy (RPDE), which is computed by the close returns method [38]. Let's assume there is a small sphere $B(S_{t0}, r)$ with radius $r > 0$, which is located close to the embedded point S_{t0}. The orbit continues in time going like this $S_{t0+1}, S_{t0+2}, S_{t0+3} \ldots$ until it has left the small sphere, i.e. until $|S_{t0} - S_{t0+j} > r|$ for a given $j > 0$. The difference between two time instants is the recurrence time $T = t_1 - t_0$. The recurrence time is computed for all embedded data points S_t, forming a histogram of recurrence times $R(T)$, which is normalized to give the recurrence probability density according to Equation 3.26.

$$\text{RPDE} = -\frac{R(t) \cdot \ln(R(t))}{\ln(T_{max})} \tag{3.26}$$

Where T_{max} is the maximum recurrence time, a fixed parameter which is chosen before so that all recurrence time that was obtained empirically for the given finite length signal is less or equal than this value. The chosen of r is important to get the properties of interest because if the orbit is quasi-periodic, r has to be large enough to capture all recurrences, but no so large to find recurrences that are due to noise or other external factors but not because of the periodicity.

3.1.7 Detrended Fluctuation Analysis (DFA)

The Detrended Fluctuation Analysis (DFA) is a simple method to identify different stages of the same system with different behaviour of the scale. The DFA algorithm is used to estimate the stochastic component of the gait process. DFA searchs trends over intervals of size L from the signal, which allow to obtain long-term dependencies of the time-series and measures the RMS average deviation $F(L)$ around the tend lines.

The first step in this algorithm is the integration the signal by means of the sum in Equation 3.27, where $t = 0, 1, 2...N - 1$ and N is the signal length S_t.

$$u_t = -\sum_{i=1}^{n} S_i \tag{3.27}$$

Next step is the division of the resulting signal u_t into non-overlaping intervals with size L. For each interval the best adjustment of the straight trend line for u_t is computed, producing a signal with a piecewise linear trend for this interval size. We can denote this as u_t^L. The fluctuation for this time scale is calculated by the Equation 3.28.

$$F(L) = \left[\frac{1}{N} \sum_{t=0}^{N-1} (u_t - u_t^L) \right]^{1/2} \tag{3.28}$$

The final step is to fit a straight line α which depends on the points $[logL, logF(L)$ over all interval size L. The signal S_t represents a combination of deterministic and stochastic dynamics, the deterministic part are dictated by the F function that will result in slow changes of the signal over this relative time length scale. The stochastic fluctuations in the signals indicate changes in the shorter time scale. Since the objective of DFA is to analyze the signal stochastic properties, just a limit range of the intervals size is investigated, where the signal stochastic component exhibit self-similarity indicated by a straight line in the log-log graph of the interval length vs the fluctuation.

The resulting scaling exponent can assume any value in a real line. However, it will be more convenient to represent it on a finite scale $[0, 1]$, thus we need to find a mapping function $g : \mathbb{R} \to [0, 1]$. One of these functions that find common use in patterns recognition is the logistic function $g(x) = (1 + \exp(-x))^{-1}$, so the normalized scaling exponent will be represented by the Equation 3.29.

$$\alpha_{norm} = \frac{1}{1 + \exp(-\alpha)} \tag{3.29}$$

We expected that the PD patients gait signals have α_{norm} value closer to 1 than healthy people gait signals [39]. This is similar to the HE, except that DFA may be applied to time-series whose underlying statistics are non-stationary. In addition, DFA allows to detect the embedded intrinsic self-similarity in a non-stationary time-series and avoids the spurious detection of the apparent self-similarity.

3.2 Poincare Sections.

A Poincaré section is the intersection of an orbit in the phase space of a continuous dynamical system with an hyperplane transverse to the flow of the trajectory of the phase

space. The Poincaré section is used to reduce the dimensionality of the phase space, and to transform the continuous flow time into a discrete time map. The map iterations are given by the point where the trajectory intersects the surface in a specific direction as in Figure 3.3.

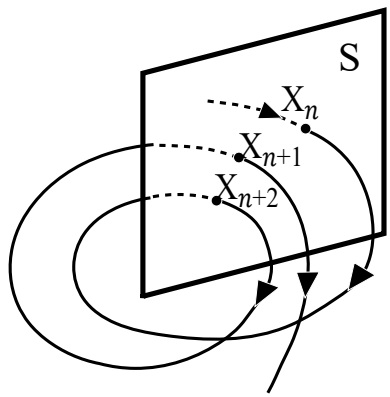

Figure 3.3: Poincaré Section of a surface

The discrete time of this map is the recount of intersections and commonly is not simply proportional to the original time of the flow. The time variation of a trajectory between two successive intersection points, depends on the current path in the reconstructed state space and on the chosen section surface. Each period leads at least to one point in the Poincaré Section. Although it does not exist any general method to build the Poincaré maps, they can capture all maximums or minimums in this phase space., so those maps are suitable to study the dynamical range of a system.

Then, in the expansion of the reconstructed space by $s(t), \dot{s}(t), \ddot{s}(t), ...$, the intersections of the trajectory with the given surface by $\dot{s}(t) = 0$ are given by the maxima.
The maxima will be a special measure function, which is projected over the first component of an applied vector to the state of vectors into this surface. The discrete period of a periodic orbit of a corresponding Poi. Due to movement of the surface the number of intersection points per trajectory can be reduced.

After the reconstruction of Poincaré sections, a clustering algorithm is performed to model the maps in a probabilistic way. We consider a GMM to model the sections of the map with the aim to extract features from the Gaussian clusters. Figures 3.4 and 3.5 show examples of the Poincaré maps and clusters extracted with the GMM. Note that there are differences in the maps obtained from the different subjects. The maps obtained for the HC subjects in Figure 3.4 reflect more data concentration than PD patients maps. In PD patients Poincaré maps in Figure 3.5 while the patient is more affected by the disease, there are more dispersion in the data.

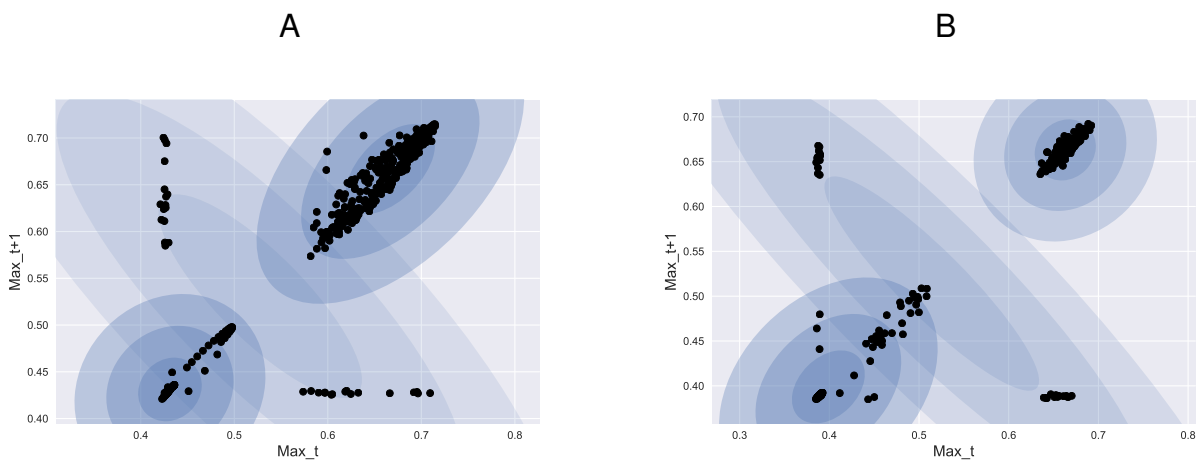

Figure 3.4: Poincaré Sections from gait signal of A: Young Healthy Control, B: Elderly Healthy Control

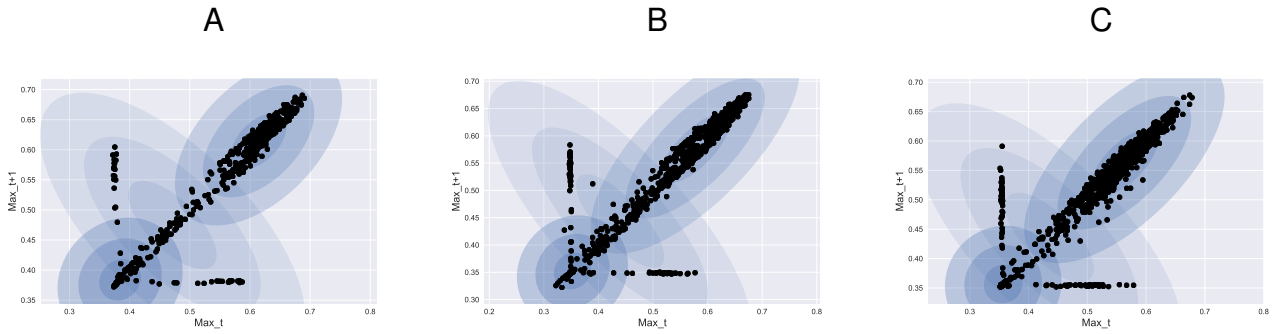

Figure 3.5: Poincaré sections from gait signals produced by A: PD patient in low state of the disease (MDS-UPDRS-III=6 for lower limbs), B: PD patient in intermediate state of the disease (MDS-UPDRS-III=13 for lower limbs). C: PD patient in severe state of the disease (MDS-UPDRS-III=34 for lower limbs)

3.2.1 Gaussian Mixture Model.

Gaussian Mixture Model can be seen as a clustering algorithm that implements the Expectation Maximization algorithm (EM) in Apendix B to find the mixture of Gaussian distributions that best model any dataset.

To find the initial values the GMM algorithm is initialized by means of the K-Means algorithm. Then, it is assumed that each sample is generated by a probability distribution.

The goal is to estimate the parameters μ_k (means), Σ_k (co-variances) and $\boldsymbol{\omega}$ (weight of the mixture) according to the maximum likelihood, computed according to Equation 3.30.

$$L(\boldsymbol{X}|\theta) = \prod_{t=1}^{n} \sum_{k=1}^{K} \omega_k P_k(\boldsymbol{x_t}|\theta_k) \tag{3.30}$$

Where K is the number of clusters in the Θ model, n is the number of observations (feature vectors), $\boldsymbol{x_t}$ is the t-th feature vector and Θ_k is the set of Gaussian parameters μ_k and Σ_k. P_k is the probability density distribution, which is estimated using Equation 3.31, where d is the number of features.

$$P_k(\boldsymbol{x_t}|\theta_k) = \frac{\exp(\frac{1}{2}(\boldsymbol{x_t} - \mu_k)^T \Sigma_k^{-1} (\boldsymbol{x_t} - \mu_k))}{\sqrt{(2\pi)^d \Sigma_k}} \tag{3.31}$$

The means vector μ_k and the co-variances matrix Σ_k determine the centers and geometric characteristics of each Gaussian component. The following are the steps of this algorithm:

1. The initialization of all Gaussian components is the first step for an EM algorithm, initialized θ_k and all Gaussian components are assumed as equiprobable.

2. The posterior probability of the model is computed for each vector $\boldsymbol{x_t}$, according to the Bayes theorem [40] as follows:

$$P_i(\theta_k|\boldsymbol{x_t}) = \frac{P_k(\boldsymbol{x_t}|\theta_k) P_k(\theta_k)}{\sum_{k=1}^{K} P_k(\boldsymbol{x_t}|\theta_k) P_k(\theta_k)} \tag{3.32}$$

3. Then, the weights of the Gaussian components are computed using Equation 3.33, where N is the number of feature vectors.

$$P_k(\theta_k) = \sum_{t=1}^{N} \frac{P_i(\theta_k|x_t)}{N} \tag{3.33}$$

4. When all weights are computed, μ_k and Σ_k are re-estimated:

$$\mu_k = \frac{\sum_{t=1}^{N} P_t(\theta_k|x_t) x_t}{\sum_{t=1}^{N} P_i(\theta_k|x_t)} \tag{3.34}$$

$$\Sigma_k = \sum_{t=1}^{N} \frac{(x_t - \mu_k)^T P_t(\theta_k|x_t)(x_t - \mu_k)}{\sum_{t=1}^{N} P_i(\theta_k|x_t)} \tag{3.35}$$

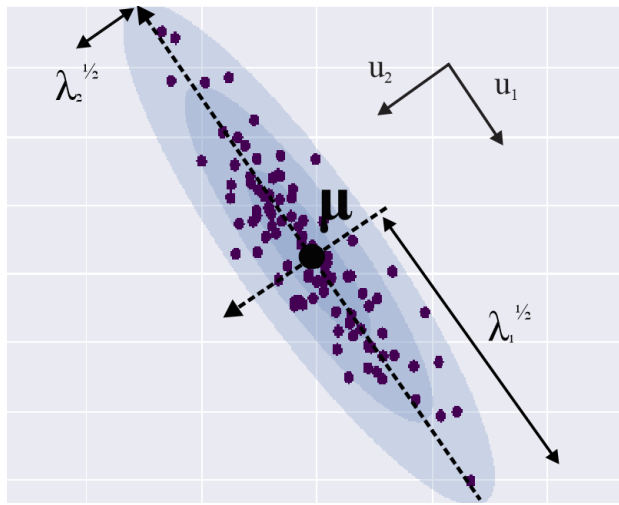

Figure 3.6: Representation of the characteristics of a GMM

5. The procedure is repeated until it converges. The number of iterations depends on the amount of the data.

In Figure 3.6 is shown the characteristics that compounds a Gaussian such as its mean $\boldsymbol{\mu}$ and the eigenvalues (λ) and eigenvectors ($\boldsymbol{u}$) of the covariance matrix $\boldsymbol{\mu}$. $\boldsymbol{\Sigma}$ can be assumed as symmetric, the eigenvalues (λ) and eigenvectors ($\boldsymbol{u}$) of the covariance matrix $\boldsymbol{\mu}$ in Equation 3.36.

$$\boldsymbol{\Sigma u_i} = \lambda_i \boldsymbol{u_i} \tag{3.36}$$

Then, expressing $\boldsymbol{\Sigma}$ in terms of its eigenvector, it takes the form defined in Equation 3.37, where D corresponds to the dimension.

$$\boldsymbol{\Sigma} = \sum_{i=1}^{D} \lambda_i \boldsymbol{u_i} \boldsymbol{u_i^T} \tag{3.37}$$

Chapter 4

Classification and regression methods

4.1 K-Nearest-Neighbors (KNN)

KNN algorithm [41] belongs to the family of lazy and competitive learning algorithms. Lazy means that the algorithm does not make a model up to the moment that is required the prediction. Competitive because it uses competition between data instances to perform a predictive decision. A new data x is classified using a majority vote among the $\mathbf{K}$ instances, defining competencies as a distance measure d in Equation 4.1, and the most likely class is assigned to the input between their K–neighbors as is shown in the Figure 4.1.

$$d(\boldsymbol{x}, \mathbf{x}') = \sqrt{(x_1 - x_1')^2 + (x_2 - x_2')^2 + ... + (x_n - x_n')^2} \tag{4.1}$$

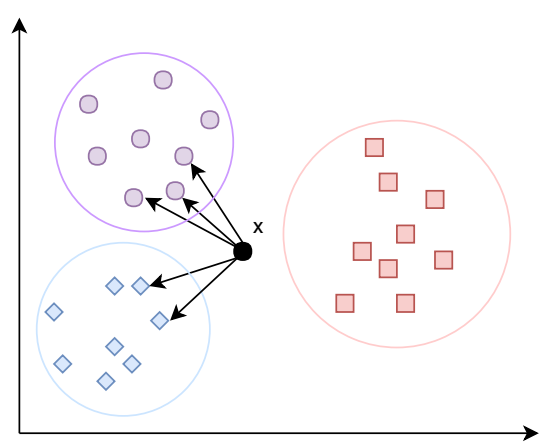

Figure 4.1: KNN selection representation with K=5

KNN does not assume anything about the data, only that a distance measure can be computed in a consistent way between two instances. Based on probability distribution, given a positive integer $\mathbf{k}$-nearest points, taking into account that $\mathbf{k}$ is usually odd to avoid tie situations, a new data x and a likelihood measure d (distance), a KNN classifier performs the following steps:

1. If we want to estimate a probability density, it is considered a small sphere centred in x and a fixed value of $\mathbf{k}$. An appropriate for the volume of the sphere (V) has to be defined. The distance between x and each training data point is computed.

2. The density is given by $p(x)$ in Equation 4.2, where it has a data set with N observations. This is known as KNN method.

$$p(x) = \frac{K}{NV} \tag{4.2}$$

3. To extend this into a classification problem, this density estimation method is applied to each class separately and make use to the Bayes Theorem [40].

4. The conditional probability in Equation 4.3 is estimated for each class.

$$P(Y = j|X = x) = \frac{1}{k} \sum I(Y^{(i)} = j) \tag{4.3}$$

   Where $I(x)$ is the indicator function (class prior) to evaluate 1 when the argument is true.

5. Finally, for the input x, the class with the highest probability is assigned.

In general, a large $\mathbf{k}$ value is more accurate, reducing general noise, but there is no guarantee. The cross–validation is another way to determine retrospectively a good value of $\mathbf{k}$ by means of using a independent data set to validate $\mathbf{k}$.

4.2 Support Vector Machine.

The algorithm for linear SVM was proposed by Vapnik in [42]. The aim of this algorithm is to find a hyperplane with maximum margin given a training set S of l training points as follows:

$$S = \left\{\mathbf{x}_i, y_i\right\} \ , i = 1, 2, ..., l \tag{4.4}$$

Where each point $\mathbf{x}_i \ \epsilon \ \mathbb{R}^N$ belongs to two different classes and a label is assigned for each one $y_i; \epsilon \ \{-1, 1\}$. The decision function of the SVM is defined according to Equation 4.5.

$$y(\boldsymbol{x}) = \boldsymbol{\omega}^T \boldsymbol{\phi}(\boldsymbol{x}) - b \tag{4.5}$$

Where $\boldsymbol{\omega}$ is the perpendicular vector to the hyperplane and $\boldsymbol{\phi}(\boldsymbol{x})$ denotes a fixed feature space transformation (kernel). If the SVM predicts that the positive class is given when $\boldsymbol{\omega} \cdot \boldsymbol{x} - b > 0$ and the negative class by $\boldsymbol{\omega} \cdot \boldsymbol{x} - b < 0$ as is showed in the Figure 4.2.

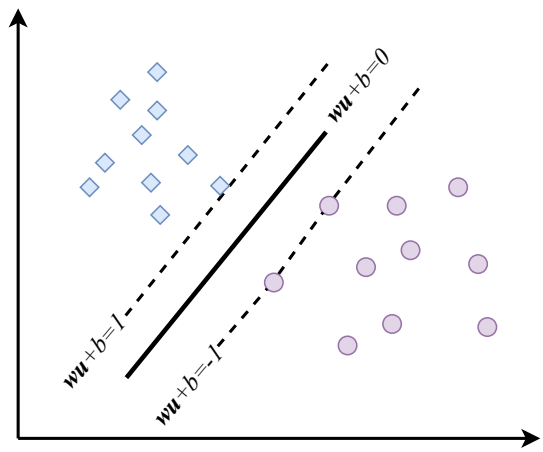

Figure 4.2: Best fitting hyperplane for the example training set S

The distance between the feature vectors nearest to the hyperplane showed in the Figure 4.2 is given by $2\,\|\boldsymbol{w}\|$ aiming to maximize the distance,$\|\boldsymbol{w}\|$.

The optimal hyperplane can be found as a quadratic programming problem and it is represented by the following equation:

$$\arg\min_{\boldsymbol{\omega},b} \frac{1}{2}\,\|\boldsymbol{\omega}\|^2 + C\sum_{i=1}^{l} \xi_i \tag{4.6}$$

Where C is an adjustment parameter that compute the error between the separation of the two classes and the training set.$\sum_{i=1}^{l} \xi_i$ on the other hand, it is a miss-classification errors measure referring to a set of slack variables. The slack variables denote by $\xi_i \geq 0$. When $\xi_i = 0$, it means that data points are inside of the correct margin boundary, otherwise $\xi_i = |\hat{y}_i - y_i|$. If a data point is on the decision boundary ($y_i = 0$), $\xi_i = 1$ and when $\xi_i > 1$ will be miss-classified. Then, the classification constrains will be replaced by $\hat{y}_i y_i \geq 1 - \xi_i$. The optimization problem solution it is shown in Appendix C.1.

4.3 Random Forest (RF)

RF is a learning supervised algorithm used because its precision and its robustness against noisy data. It can process a lot of inputs, without variables suppression and also estimating the importance of the variables in the classification. It consists of a set of classification trees from selected data samples of a randomly way, called *"forest"*, where each tree contributes with a single vote to the assignation of the most frequently class as is shown in Figure 4.3. It uses a combination of features at each node to grow a tree, instead using the best variables, which reduces the generalized error.

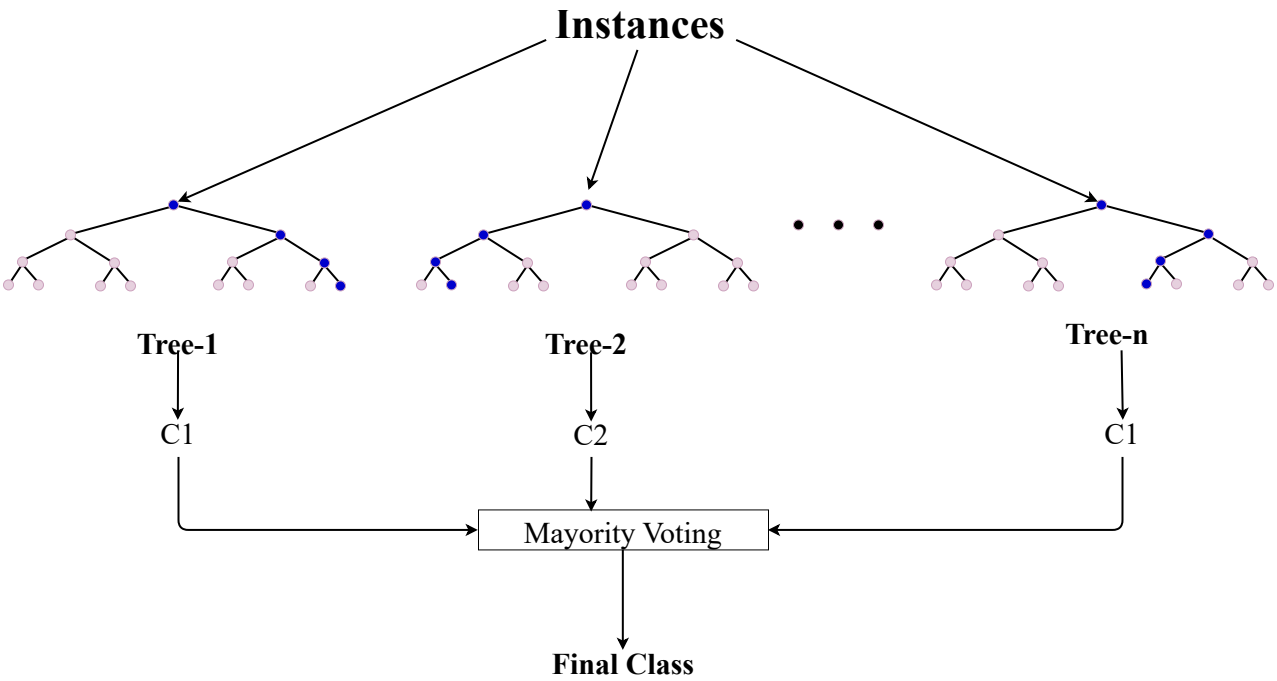

Figure 4.3: Architecture of the random forest model

To achieve the above is employed Bagging method to generate a training data–set by means of re–sample of original data–set randomly. Each selected subset using Bagging to perform each individual growing that contains a certain proportion of training data.

One of its advantages is that it does not suffer over-fitting problems, because RF take the all prediction average, which cancels the biases. Also missing values can be treated, either using the median of the values to replace the continuous variables or computing the proximity weighted average of the missing values. The individual trees are generated using an indicator of attribute selection being one of the most frequently used the Gini Index [43] for each attribute or also known as *"the Total Decrease in Node Impurity"*.

Gini in Equation 4.7, measures the impurity of a given element with regard to the other classes, obtaining the relative importance of the feature. The more it decreases, the more significant is the feature, being the mean decreasing an important parameter to the feature selection. Thus, using a given combination of features is made to grow up until their maximum depth.

$$\sum \sum_{j \neq i} (f(C_i, \boldsymbol{T})/|\boldsymbol{T}|)(f(C_j, \boldsymbol{T})/|\boldsymbol{T}|) \tag{4.7}$$

Where T is the given training set, C_i is the class and $f(C_i, \boldsymbol{T})/|\boldsymbol{T}|$ is the probability that the selected case belongs to C_i.

The RF algorithm is performed by the following steps:

1. It is selected randomly samples from the dataset, draw n-trees of these samples.
2. It is building a decision tree for each sample and it is obtained a prediction from every decision tree. For the sample grows a classification tree in which the features form each node.
3. Each tree will grow to its maximum extension without pruning.
4. It is selected the final result by majority vote of each one of performed prediction.

4.4 Support Vector Regression.

The Support Vector Regression (SVR) is an extension of SVM algorithm to regression. To this method instead a regularization error function for a loss function (ϵ-insensitive) defined as $L_\epsilon(y, \hat{y})$ of width 2ϵ. This loss function (see Figure 4.4) ensures the existence of a global optimum, having associated a linear cost with errors out of the ϵ-insensitive region given by:

$$L_\epsilon(y, \hat{y}) = \begin{cases} 0 & \text{if } |y - \hat{y}| < \epsilon \\ |y - \hat{y}| - \epsilon & \text{otherwise} \end{cases} \tag{4.8}$$

The ϵ-insensitive penalizes the model if there are differences between the training set and model predictions. The feature vectors $\boldsymbol{x}$ are mapped in a m-dimensional space using a lineal kernel, where y is shown in Equation 4.5. Now the minimization is given by:

$$C \sum_{n=1}^{N} L_\epsilon(y_n - \hat{y_n}) + \frac{1}{2}||\boldsymbol{w}||^2 \tag{4.9}$$

Where C is the regularization parameter and $\boldsymbol{w}$ establishes the each support vector weight.

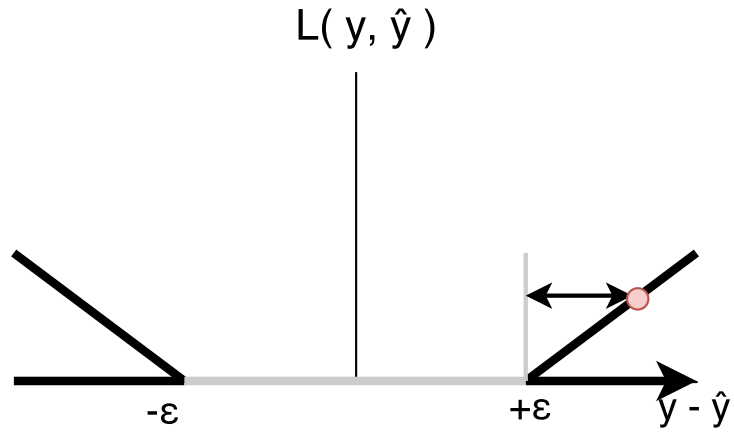

Figure 4.4: Loss Function

To the optimization problem is introduced the slack variables. For each point is necessary two slack variables, $\hat{\xi} \geq 0$ and $\xi \geq 0$. A point lies inside the ϵ-insensitive region (shows in Figure 4.5) when $y_n - \epsilon \leq \hat{y_n} \leq y_n + \epsilon$. Then, when if it is introducing the slack variables this allows that some points lie outside of the above mentioned region, corresponding in two following conditions:

$$\hat{y_n} \leq y_n + \epsilon + \xi_n \tag{4.10}$$

$$\hat{y_n} \geq y_n - \epsilon - \hat{\xi_n} \tag{4.11}$$

Then, according to the above, the error function is re-writing as in Equation 4.12, but this expression must be minimized subject to a certain constraints (slack variables) showed in the conditions 4.10 and 4.11. This is achieved by the Lagrange multipliers and optimizing the Lagrangian. The optimization problem solution it is shown in Appendix C.1.

$$C \sum_{n=1}^{N} (\xi_n + \hat{\xi_n}) + \frac{1}{2} ||\boldsymbol{w}||^2 \tag{4.12}$$

4.5 Cross Validation

Its procedure is based on in the generation of K-folds of given data set groups. It is often used in machine learning to estimate the capability of the model on unseen data. Besides, is less optimistic than other common methods. Before to define the algorithm, it is important specify the K-splits number, in according to:

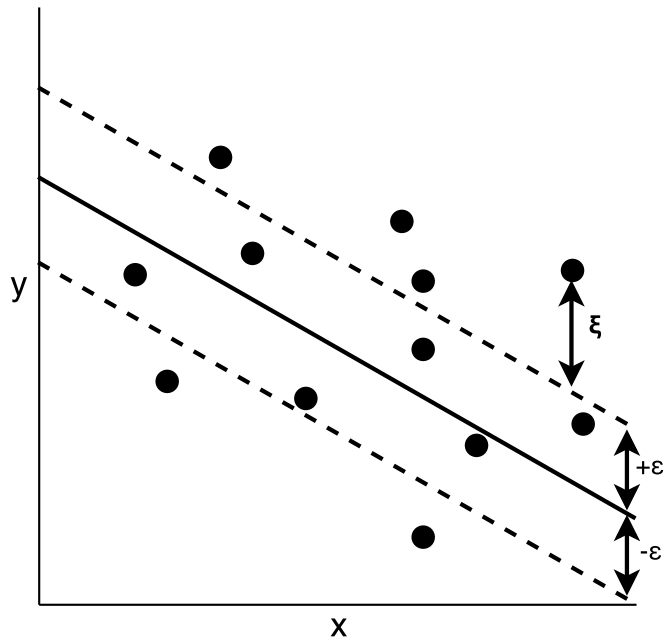

Figure 4.5: Ilustration of One-dimensional linear regression with the ϵ-insensitive "tube"

- The data is divided into k subsets.
- Leave-One-Out is when the K is equal to the data set length, where each test sample participates on the hold out data set.

To this application was chosen the division into k subsets. To perform it, is suggest the following steps:

1. Usually, the data set is shuffle randomly, but in this case the characteristics of the population are known allowing to balance each K–fold taking half of HC subjects (balance per gender) and half of PD patients (balanced across the stage of the disease).
2. The data set is split in K-folds, in this case K=5 taking into account that the data set length is 90.
3. It is chosen K-1 folds to train and to validate data set.
4. One fold is to test. This folds are not in the set of the last step.
5. Then, the model is trained and evaluated with the testing set.
6. The procedure is repeated for k times.

4.6 Performance metrics

A performance metric is a regular measurement of outcomes and results, is defined as a process that quantifies the effectiveness and efficiency of past actions [44].

To evaluate results of Machine Learning experiments several metrics will perform. This metrics are related to the measure of effectiveness, efficiency or correlation [45].

To understand the following metrics, it is defined some concepts:

- True positive (TP): the number of cases correctly identified as PD patient.
- True negative (TN): the number of cases correctly identified as healthy control.
- False positive (FP): the number of cases incorrectly identified as PD patient.
- False negative (FN): the number of cases incorrectly identified as healthy control.

Confusion Matrix

The confusion matrix allows the visualization of the algorithm performance of a supervised learning. The number of class predictions are represented by columns, while the instances of the real classes are representing by rows. As shown in table 4.1.

Table 4.1: Confusion matrix structure

	Actual Class	
Predicted Class	TP	FN
	FP	TN

Accuracy

The accuracy in Equation 4.13 is the ability to discriminate patients and healthy cases correctly. To compute it, is estimated the proportion of true positive and true negative in all evaluated cases.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4.13}$$

Sensitivity

The sensitivity in Equation 4.14, is the ability to discriminate patients correctly. To compute it, it is estimated the proportion of true positive in patient cases.

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4.14}$$

Specificity

The specificity in equation4.15, is the ability to discriminate healthy controls correctly. To compute it, it was estimated the proportion of true negative in healthy control cases.

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{4.15}$$

Receiver Operating Characteristic curve (ROC)

The ROC curve is a graphical representation which shows the binary classifier performance, while its discrimination threshold is varied.

Where y-label is the True positives (Sensitivity) and x-label is The False positives ($1 - \text{Specificity}$).

Cohen's Kappa

Cohen's Kappa (κ) in Equation 4.16 is frequently used to test the interrater reliability. It is similar to the correlation coefficients ranging between –1 to 1 but it measures the degree of agreement that can be attributed to chance. It has to take into account that this measures only is used between two raters.

$$\kappa = \frac{p_o - p_\epsilon}{1 - p_\epsilon} \tag{4.16}$$

Where p_o is the empirical probability of agreement on the label assigned to any sample and p_ϵ is the expected agreement when both raters assign labels randomly. κ can be interpreted as following [46]:

- $\text{Values} \leq 0$ no agreement.
- $0.01 - 0.20$ none to slight.
- $0.41 - 0.60$ moderate.
- $0.61 - 0.80$ substantial.

- $0.81 - 1.00$ almost perfect agreement.

Spearman's Correlation

Spearman's is a correlation metric that measures the strength of the relationship between two variables. It has the following characteristics:

- Its value ranges from -1 to 1.
- The Spearman's correlation advantage over Pearson correlation is that Spearman leaves to find out nonlinear correlations between variables.

As following, this correlation is define:

$$\rho = 1 - \frac{6 \sum D^2}{N(N^2 - 1)} \tag{4.17}$$

Where D is the difference between the statistics corresponding and N is the number of data couples.

Chapter 5

Database and tasks

Gait signals were captured with the eGaIT system[1], which consists of a 3D-accelerometer (range ±6g) and a 3D gyroscope (range ±500°/s) attached to the lateral heel of the shoes [9]. Figure 5.1 shows the eGait system and the inertial sensor attached to the lateral heel of the shoe. The signals are transmitted by bluetooth to a tablet where they are received by an android app.

Data from both foot were captured with a sampling rate of 102.4 Hz and 12-bit resolution. The tasks performed by the patients include 20 meters walking with a stop at 10 meters (Two times 10 m walk, 2x10m), and 40 meters walking with a stop every 10 meters (Four times 10 m walk, 4x10m).

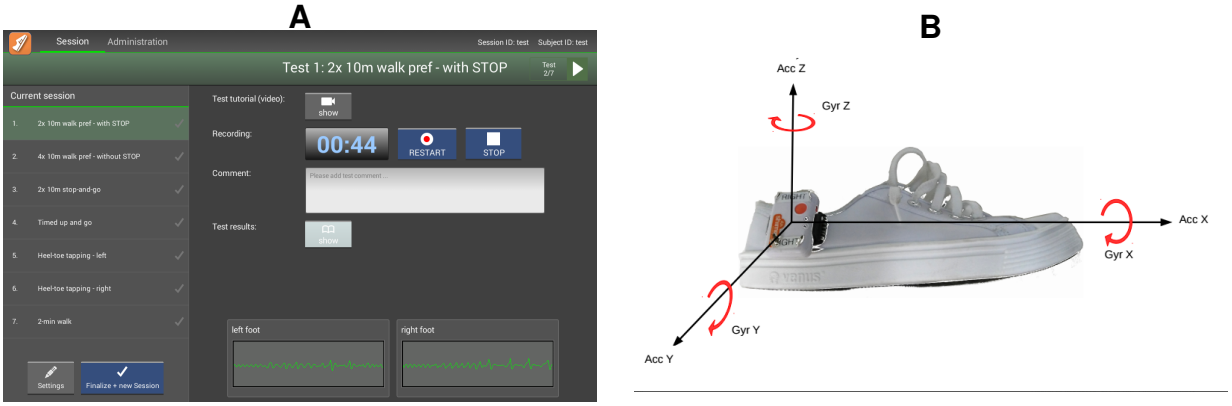

Figure 5.1: Interface eGaiT and shoe with its attached inertial sensor.

Data are obtained from 45 PD patients and 89 HC subjects. The HC subjects were divided into two groups: the first one formed with 44 YHC (Young Healthy Controls), and the second one with 45 EHC (Elderly Healthy Controls) subjects. The patients were

[1] Embedded Gait analysis using Intelligent Technology, `http://www.egait.de/`

evaluated by an expert neurologist and labeled according to the MDS-UPDRS-III score. Table 5.1 shows additional information of the participants of this study.

Table 5.1: General information of the subjects. **PD** patients: Parkinson's disease patients. **HC**: healthy controls. **LL**:Lower limbs sub-score. μ: average. σ: standard deviation. **T**: disease duration.

	PD patients		YHC subjects		EHC subjects	
	male	female	male	female	male	female
Number of subjects	17	28	26	18	23	22
Age ($\mu \pm \sigma$)	65 $\pm$ 10.3	58.9 $\pm$ 11.0	25.3 $\pm$ 4.8	22.8 $\pm$ 3.0	66.3 $\pm$ 11.5	59.0 $\pm$ 9.8
Range of age	41–82	29–75	21–42	19–32	49–84	50-74
T ($\mu \pm \sigma$)	9 $\pm$ 4.6	12.6 $\pm$ 12.2				
Range of duration of the disease	2–15	0–44				
MDS-UPDRS-III ($\mu \pm \sigma$)	37.6 $\pm$ 21.0	33 $\pm$ 20.3				
Range of MDS-UPDRS-III	8–82	9–106				
Range of MDS-UPDRS-III-LL	3–41	0–50				

Chapter 6

Experiments and results

6.1 Feature extraction

The above features were extracted from over the entire gait signal. Table 6.1 shows the number of computed features for each task performed by the patients. Ten NLD and ten Poincaré features are extracted, which are computed for each of the six signals from the inertial sensors, forming the feature matrix used to classify PD patients and HC subjects.

Table 6.1: Number of features per task

Foot	Task	Number of axes	Number of features		Total
			NLD	Poincare	
Left	2x10m	6	10	10	120
Left	4x10m	6	10	10	120
Left	Fusion	6	20	20	240
Right	2x10m	6	10	10	120
Right	4x10m	6	10	10	120
Right	Fusion	6	20	20	240
Both	2x10m	12	10	10	240
Both	4x10m	12	10	10	240
Both	Fusion	12	20	20	480

Three experiments were performed: (1) classification of PD patients and HC subjects, (2) prediction of the neurological state of the patients according to the MDS-UPDRS-III sub-score for lower limbs, and (3) the classification of PD patients in different stages of the disease. Three sets of features were considered in each experiment: (1) The NLD and entropy features, described in Section 3.1, (2) the features extracted from Poincaré sections using the GMM approach, described in Section 3.2, and (3) the combination of both feature sets.

For all experiments, we followed a 5-fold cross-validation strategy, where 3 folds were used for training, one to optimize the hyper-parameters of the classifiers, i.e., development set, and one for test. The parameters were optimized in a grid search over the train folds. For the case of KNN, we optimize the number of neighbors $K \in \{3, 5, \cdots, 11\}$. For the SVM, we optimize the complexity parameter $C \in \{10^{-4}, 10^{-3}, \cdots 10^{4}\}$ and the bandwidth of the kernel $\gamma \in \{10^{-4}, 10^{-3}, \cdots, 10^{4}\}$. For the RF, we optimize the number of trees $N \in \{5, 10, 20, 30, 50, 100\}$ and the maximum depth of the trees $D \in \{2, 5, 10, 20, 30, 50, 100\}$. A similar approach is performed with the SVR for the regression experiment. In this case we optimize the complexity parameter $C \in \{10^{-6}, 10^{-5}, \cdots, 10^{4}\}$ and the insensitive parameter $\varepsilon \in \{0.0001, 0.001, 0.01, 1, 10, 50, 100\}$. The performance of the regression method is obtained with the Spearman's correlation coefficient ρ.

6.2 Classification of PD patients and HC subjects

Two experiments are performed: (1) classification of PD vs. YHC , and (2) classification of PD vs. EHC. Individual experiments are performed per foot and per task. In addition, the features computed from the two tasks and the two feet are combined.Three different cases are considered depending on the feature set: (1) NLD and entropy features, (2) the features extracted from Poincaré sections and GMMs, and (3) the combination of both feature sets. The results obtained for these feature sets are shown in Tables 6.2 to 6.7. Also, individual experiments are performed per foot and per task, and the combination of features is performed via early fusion.

Table 6.2 shows the results for the PD vs. YHC subjects. In general the best results are obtained with the RF classifier. The fusion of features from both feet and the two tasks also provides the highest accuracy (93.3%$\pm$7.2).

Although the highest accuracies were obtained in experiments classifying PD vs. YHC subjects with NLD and entropy features, it does not consider the effect of age in the walking process. The results classifying PD patients vs. EHC subjects with similar age to the patients for NLD and entropy features are shown in Table 6.3. Note that the results are slightly lower than those obtained in the previous experiment. Although such an impact, relatively high accuracies are obtained, specially when we combine the features from both tasks and both feet. For the separate classification using features computed from each foot, the highest accuracies were obtained for the left foot, which may indicate that the left lower limbs are more affected due to the disease, having in mind that most of the patients are right dominant foot. This fact is known as cross laterality [47].

Table 6.3 shows relatively high accuracies, especially when we combined the features from both tasks and both feet (85.6% obtained with the RF classifier). The results also show that the specificity (91.1% for the RF classifier when the features are combined) is higher than sensitivity (80.0% in the same scenario), which indicates that our system is

Table 6.2: Results of NLD features to classify PD patients vs. YHC subjects. **ACC:** accuracy in the test set. μ: average. σ: standard deviation. **Sen:** Sensibility. **Spe:** Specficity. **AUC:** Area under ROC curve. **K:** number of neighbors in the KNN. **C and γ:** complexity parameter and bandwidth of the kernel in the SVM, **N and D:** Number of trees and depth of the decision trees in the RF.

		KNN				SVM					RF				
Foot	**Task**	**ACC(%) ($\mu \pm \sigma$)**	**Sen(%)/Spe(%)**	**AUC**	**K**	**ACC(%)**	**Sen(%)/Spe(%)**	**AUC**	**C**	γ	**ACC(%)**	**Sen(%)/Spe(%)**	**AUC**	**N**	**D**
Left	2x10	85.4±6.4	75.6/95.3	0.91	5	82.0±5.0	75.6/88.6	0.92	10^1	10^{-3}	86.5±7.6	80.0/93.3	0.92	20	20
Left	4x10	84.4±8.1	73.3/95.6	0.95	9	91.1±6.3	86.7/95.6	0.96	10^0	10^{-3}	93.3±7.2	93.3/93.3	0.94	30	20
Left	Fusion	88.8±3.8	77.8/100.0	0.94	5	88.9±8.8	82.2/95.6	0.94	10^1	10^{-3}	91.1±7.4	86.7/95.6	0.95	20	100
Right	2x10	85.5±4.8	71.1/100.0	0.88	9	79.9±8.3	77.8/81.9	0.91	10^1	10^{-3}	82.0±4.6	77.8/86.1	0.92	10	5
Right	4x10	78.8±9.0	60.0/97.8	0.90	7	92.2±4.9	84.4/100.0	0.92	10^1	10^{-3}	86.6±6.2	84.4/88.9	0.95	10	100
Right	Fusion	82.1±9.0	68.9/95.6	0.91	7	88.8±4.9	80.0/61.8	0.93	10^1	10^{-3}	89.9±6.2	84.4/95.6	0.95	50	5
Both	2x10	86.7±5.0	73.3/97.8	0.93	7	83.2±6.7	80.0/86.7	0.94	10^1	10^{-3}	85.5±11.5	80.0/91.1	0.92	5	2
Both	4x10	84.2±10.7	71.1/97.8	0.93	5	86.6±4.8	80.0/93.3	0.90	10^0	10^{-3}	92.2±6.3	88.9/95.6	0.94	20	2
Both	Fusion	86.5±2.9	73.3/100.0	0.93	5	91.0±4.9	84.4/97.8	0.96	10^0	10^{-3}	91.1±4.9	84.4/97.8	0.96	30	10
Average		**84.7**	**71.6/97.7**	**0.92**	–	**87.1**	**81.2/89.0**	**0.93**	–	–	**88.7**	**84.4/92.9**	**0.94**	–	–
STD		**2.7**	**5.1/2.0**	**0.0**	–	**4.2**	**3.5/11.6**	**0.0**	–	–	**3.5**	**4.8/3.7**	**0.0**	–	–

Table 6.3: Results of NLD features to classify PD patients vs. EHC subjects.**ACC:** accuracy in the test set. μ: average. σ: standard deviation. **Sen:** Sensibility. **Spe:** Specficity. **AUC:** Area under ROC curve. **K:** number of neighbors in the KNN. **C and γ:** complexity parameter and bandwidth of the kernel in the SVM, **N and D:** Number of trees and depth of the decision trees in the RF.

		KNN				SVM					RF				
Foot	**Task**	**ACC(%)**	**Sen(%)/Spe(%)**	**AUC**	**K**	**ACC(%)**	**Sen(%)/Spe(%)**	**AUC**	**C**	γ	**ACC(%)**	**Sen(%)/Spe(%)**	**AUC**	**N**	**D**
Left	2x10	81.1±9.3	80.0/82.2	0.84	5	77.78±13.0	66.7/88.9	0.74	10^{-4}	10^{-4}	83.3±14.2	73.3/93.3	0.89	30	2
Left	4x10	72.2±11.1	68.9/75.6	0.80	5	81.11±12.8	86.7/75.6	0.90	10^0	10^{-3}	84.4±7.2	82.2/86.7	0.89	10	5
Left	Fusion	80.0±8.4	73.3/86.7	0.86	5	83.33±6.8	82.2/84.4	0.84	10^{-4}	10^{-4}	83.3±8.8	77.8/88.9	0.89	30	30
Right	2x10	70.0±9.3	60.0/80.0	0.82	5	67.78±7.2	51.1/84.4	0.73	10^{-4}	10^{-4}	78.9±6.1	73.3/84.4	0.79	10	2
Right	4x10	77.8±6.8	73.3/82.2	0.82	3	76.67±7.2	73.3/80.0	0.83	10^1	10^{-3}	80.0±11.5	80.0/80.0	0.87	20	2
Right	Fusion	81.1±8.4	73.3/88.9	0.85	3	82.22±4.6	75.6/88.9	0.87	10^1	10^{-3}	85.6±6.3	82.2/88.9	0.91	20	5
Both	2x10	76.7±12.7	68.9/84.4	0.79	5	80.00±8.4	68.9/91.1	0.85	10^1	10^{-4}	78.9±11.4	71.1/86.7	0.86	30	50
Both	4x10	72.2±3.9	75.6/68.9	0.80	3	81.11±6.3	77.8/84.4	0.83	10^{-4}	10^{-4}	82.2±12.7	82.2/82.2	0.91	100	50
Both	Fusion	85.6±5.0	77.8/93.3	0.89	3	82.22±4.6	71.1/93.3	0.86	10^{-4}	10^{-4}	85.6±2.5	80.0/91.1	0.91	30	30
Average		**77.4**	**72.3/82.5**	**0.83**	–	**79.1**	**72.6/85.7**	**0.83**	–	–	**82.3**	**78.0/86.9**	**0.88**	–	–
STD		**4.8**	**5.8/7.2**	**0.0**	–	**4.5**	**10.3/5.6**	**0.1**	–	–	**2.4**	**4.4/4.2**	**0.0**	–	–

more accurate to detect correctly the EHC subjects.

Table 6.4 shows the results of Poincaré features for the PD vs YHC classification. This time applying Poincaré features the results are worsened in comparison with NLD features. While PD vs EHC (see Table 6.5) obtained similar results respect to NLD features, where the fusion of features from both feet and two tasks provided higher accuracies in the RF classifier (85.6% and 86.7%).

In comparison to the results from the set of NLD Features from Table 6.3, a higher accuracy was obtained with the Poincaré features in Table 6.5. The Poincaré features keep the premise that the fusion of both feet and tasks are more effective to discriminate between PD patients and HC subjects. Although the accuracy obtained with the Poincaré features seems to be reduced compared to the reported with the NLD and entropy features (especially for the KNN and SVM classifiers), note that the highest accuracy

Table 6.4: Results of Poincare Features to classify PD patients vs. YHC subjects. **ACC:** accuracy in the test set. μ: average. σ: standard deviation. **Sen:** Sensibility. **Spe:** Specficity. **AUC:** Area under ROC curve. **K:** number of neighbors in the KNN. **C and γ:** complexity parameter and bandwidth of the kernel in the SVM, **N and D:** Number of trees and depth of the decision trees in the RF.

		KNN				SVM					RF				
Foot	**Task**	**ACC(%) ($\mu \pm \sigma$)**	**Sen(%)/Spe(%)**	**AUC**	**K**	**ACC(%) ($\mu \pm \sigma$)**	**Sen(%)/Spe(%)**	**AUC**	**C**	γ	**ACC(%) ($\mu \pm \sigma$)**	**Sen(%)/Spe(%)**	**AUC**	**N**	**D**
Left	2x10	51.8±10.5	35.4/68.1	0.56	3	64.3±12.4	63.6/65.0	0.74	10^1	10^{-3}	64.1±17.7	52.1/76.1	0.70	10	5
Left	4x10	47.1±12.7	17.5/76.7	0.49	5	50.7±12.7	38.6/62.8	0.54	10^2	10^{-4}	61.6±17.9	51.1/72.0	0.66	30	2
Left	Fusion	47.7±13.6	25.7/69.7	0.53	7	52.7±19.2	33.6/71.7	0.58	10^2	10^{-4}	59.2±17.9	48.6/69.7	0.66	20	20
Right	2x10	54.8±8.3	38.9/70.6	0.57	5	64.1±16.9	49.3/78.9	0.73	10^1	10^{-3}	65.0±20.8	58.6/71.4	0.73	100	2
Right	4x10	62.3±8.9	56.8/67.8	0.67	5	56.6±8.8	38.9/74.2	0.61	10^1	10^{-3}	72.5±11.5	66.4/78.6	0.81	10	50
Right	Fusion	62.9±6.9	48.9/76.9	0.68	7	58.1±9.9	46.4/69.7	0.68	10^1	10^{-3}	66.7±16.4	66.1/67.2	0.74	30	5
Both	2x10	53.2±6.6	33.6/72.8	0.59	5	65.9±11.3	66.8/65.0	0.69	10^2	10^{-4}	70.1±17.3	61.8/78.3	0.74	100	2
Both	4x10	57.7±11.3	41.1/74.2	0.58	7	60.2±12.4	64.3/56.1	0.69	10^2	10^{-3}	65.4±11.9	61.4/69.4	0.72	5	20
Both	Fusion	58.1±10.8	26.1/90.1	0.54	7	61.8±16.8	51.4/72.2	0.69	10^1	10^{-4}	71.0±14.4	66.1/75.8	0.79	50	5
Average		**55.1**	**36.0/74.1**	**0.58**	–	**59.4**	**50.3/68.4**	**0.66**	–	–	**66.2**	**59.1/73.2**	**0.73**	–	–
STD		**5.7**	**12.1/6.8**	**0.1**	–	**5.3**	**12.2/6.8**	**0,1**	–	–	**4.4**	**7.0/4.1**	**0,0**	–	–

Table 6.5: Results of Poincaré features to classify PD patients vs. EHC subjects. **ACC:** accuracy in the test set. μ: average. σ: standard deviation. **Sen:** Sensibility. **Spe:** Specficity. **AUC:** Area under ROC curve. **K:** number of neighbors in the KNN. **C and γ:** complexity parameter and bandwidth of the kernel in the SVM, **N and D:** Number of trees and depth of the decision trees in the RF.

		KNN				SVM					RF				
Foot	**Task**	**ACC(%) ($\mu \pm \sigma$)**	**Sen(%)/Spe(%)**	**AUC**	**K**	**ACC(%) ($\mu \pm \sigma$)**	**Sen(%)/Spe(%)**	**AUC**	**C**	γ	**ACC(%) ($\mu \pm \sigma$)**	**Sen(%)/Spe(%)**	**AUC**	**N**	**D**
Left	2x10	67.8±4.1	53.3/82.2	0.70	3	68.9±8.3	64.4/73.3	0.74	10^1	10^{-4}	77.8±6.1	77.8/77.8	0.85	30	5
Left	4x10	62.2±10.7	46.7/77.8	0.65	5	72.2±3.5	73.3/71.1	0.81	10^2	10^{-4}	72.2±10.5	71.1/73.3	0.77	100	10
Left	Fusion	57.8±9.0	37.8/77.8	0.66	5	57.8±4.4	53.3/62.2	0.63	10^{-4}	10^{-4}	75.6±12.4	75.6/75.6	0.85	20	30
Right	2x10	61.1±11.6	44.4/77.8	0.68	9	81.1±7.5	77.8/84.4	0.86	10^2	10^{-3}	85.6±5.6	77.8/93.3	0.87	30	50
Right	4x10	54.4±9.5	42.2/66.7	0.61	5	72.2±9.9	71.1/73.3	0.80	10^1	10^{-3}	71.1±8.1	68.9/73.3	0.79	50	5
Right	Fusion	57.8±9.0	35.6/80.0	0.66	3	72.2±14.4	71.1/73.3	0.80	10^2	10^{-4}	82.2±8.8	82.2/82.2	0.86	100	10
Both	2x10	61.1±6.1	37.8/84.4	0.70	3	77.8±7.0	77.8/77.8	0.87	10^1	10^{-3}	82.2±2.2	82.2/82.2	0.90	100	20
Both	4x10	63.3±7.5	46.7/80.0	0.71	7	73.3±6.4	73.3/73.3	0.80	10^2	10^{-3}	76.7±6.5	75.6/77.8	0.84	20	30
Both	Fusion	65.6±8.2	35.6/95.6	0.71	3	81.1±5.7	75.6/86.7	0.87	10^1	10^{-3}	86.7±2.7	80.0/93.3	0.91	50	5
Average		**61.2**	**42.2/82.3**	**0.70**	–	**73.0**	**70.8/75.0**	**0.80**	–	–	**78.9**	**76.8/81.0**	**0.80**	–	–
STD		**4.2**	**6.1/7.6**	**0,0**	–	**7.1**	**7.7/7.2**	**0,1**	–	–	**5.6**	**4.6/7.7**	**0,0**	–	–

improved in up to 1.1% the highest one obtained with the NLD and entropy features.

The performance of PD vs YHC classification added Poincaré features (see Table 6.4 and 6.6) decreased in comparison with only NLD features (see Table 6.2). Results for the combination of NLD, entropy, and Poincaré features (see Table 6.6) show that the highest accuracy was obtained with the SVM classifier (86.7%). This result was similar to those obtained only with the Poincaré features; however, the feature combination provides more stable results in terms of the specificity and sensitivity. Note also the presence of the cross-laterality effect in the results, i.e., the highest accuracies are obtained with the features computed from the left foot.

Figures 6.1 to 6.9 show an additional comparison among the best results obtained in the classification of PD patients vs. the two groups of HC subjects. The ROC curves

Table 6.6: Results of NLD+Poincaré features to classify PD patients vs. YHC subjects. **ACC:** accuracy in the test set. μ: average. σ: standard deviation. **Sen:** Sensibility. **Spe:** Specficity. **AUC:** Area under ROC curve. **K:** number of neighbors in the KNN. **C and γ:** complexity parameter and bandwidth of the kernel in the SVM, **N and D:** Number of trees and depth of the decision trees in the RF.

		KNN				SVM					RF				
Foot	**Task**	**ACC(%) ($\mu \pm \sigma$)**	**Sen(%)/Spe(%)**	**AUC**	**K**	**ACC(%) ($\mu \pm \sigma$)**	**Sen(%)/Spe(%)**	**AUC**	**C**	γ	**ACC(%) ($\mu \pm \sigma$)**	**Sen(%)/Spe(%)**	**AUC**	**N**	**D**
Left	2x10	64.0±8.8	56.1/71.9	0.72	5	74.4±13.5	74.3/74.4	0.81	10^2	10^{-4}	76.1±12.3	71.4/80.8	0.83	100	30
Left	4x10	60.6±10.1	38.1/83.1	0.72	5	72.9±17.8	71.8/73.9	0.76	10^1	10^{-4}	78.7±11.3	76.8/80.6	0.82	100	5
Left	Fusion	70.7±17.8	56.1/85.3	0.84	5	77.9±16.1	81.8/73.9	0.81	10^1	10^{-4}	72.6±17.4	66.8/78.3	0.81	30	10
Right	2x10	53.8±10.7	33.2/74.4	0.67	7	71.8±14.4	71.8/71.7	0.74	10^1	10^{-4}	69.9±20.2	61.5/78.3	0.78	50	5
Right	4x10	65.0±13.4	44.3/85.6	0.78	7	72.6±10.4	66.8/78.3	0.77	10^1	10^{-4}	74.9±13.5	68.9/80.8	0.79	20	5
Right	Fusion	68.4±12.9	49.3/87.5	0.77	7	78.6±8.1	74.3/82.8	0.81	10^1	10^{-4}	78.6±9.2	74.3/82.8	0.81	100	5
Both	2x10	65.8±9.0	50.7/80.8	0.75	3	71.5±8.9	66.8/76.1	0.81	10^1	10^{-4}	73.4±18.5	63.9/82.8	0.82	10	5
Both	4x10	61.1±13.4	36.8/85.3	0.79	7	72.9±18.9	71.8/73.9	0.82	10^0	10^{-3}	75.8±18.1	79.3/78.3	0.82	100	2
Both	Fusion	68.2±12.9	48.9/87.5	0.84	7	79.8±9.2	76.8/82.8	0.82	10^0	10^{-3}	76.2±14.1	71.8/80.6	0.82	50	50
Average		**64.2**	**45.9/82.4**	**0.76**	–	**74.7**	**72.9/76.4**	**0.79**	–	–	**75.1**	**69.9/80.4**	**0.81**	–	–
STD		**5.1**	**5.6/5.6**	**0.0**	–	**3.2**	**4.7/4.0**	**0.0**	–	–	**2.8**	**5.0/1.8**	**0.0**	–	–

Table 6.7: Results of NLD+Poincaré features to classify PD patients vs. EHC subjects. **ACC:** accuracy in the test set. μ: average. σ: standard deviation. **Sen:** Sensibility. **Spe:** Specficity. **AUC:** Area under ROC curve. **K:** number of neighbors in the KNN. **C and γ:** complexity parameter and bandwidth of the kernel in the SVM, **N and D:** Number of trees and depth of the decision trees in the RF.

		KNN				SVM					RF				
Foot	**Task**	**ACC(%) ($\mu \pm \sigma$)**	**Sen(%)/Spe(%)**	**AUC**	**K**	**ACC(%) ($\mu \pm \sigma$)**	**Sen(%)/Spe(%)**	**AUC**	**C**	γ	**ACC(%) ($\mu \pm \sigma$)**	**Sen(%)/Spe(%)**	**AUC**	**N**	**D**
Left	2x10	77.8±11.6	66.7/88.9	0.85	9	80.0±6.7	73.3/86.7	0.82	10^1	10^{-3}	82.2±9.5	80.0/84.4	0.88	30	10
Left	4x10	78.9±7.3	66.7/91.1	0.86	3	82.2±4.1	82.2/82.2	0.89	10^{-4}	10^{-4}	84.4±8.1	77.8/91.1	0.90	100	2
Left	Fusion	83.3±6.0	73.3/93.3	0.91	5	86.7±8.3	88.9/84.4	0.91	10^{-4}	10^{-3}	85.6±9.0	73.3/97.8	0.93	5	5
Right	2x10	67.8±4.1	40.0/95.6	0.75	9	80.0±7.5	68.9/91.1	0.86	10^1	10^{-3}	77.8±3.5	68.9/86.7	0.85	30	10
Right	4x10	78.9±4.1	73.3/84.4	0.84	3	78.9±9.5	77.8/80.0	0.76	10^{-4}	10^{-4}	77.8±15.3	84.4/71.1	0.81	20	2
Right	Fusion	75.6±7.5	60.0/91.1	0.82	5	76.7±7.4	71.1/82.2	0.81	10^{-4}	10^{-4}	82.2±4.1	80.0/84.4	0.88	100	10
Both	2x10	74.4±7.5	53.3/95.6	0.84	7	78.9±4.1	71.1/86.7	0.83	10^0	10^{-4}	85.6±4.4	80.0/91.1	0.92	10	50
Both	4x10	76.7±7.3	64.4/88.9	0.87	5	81.1±7.5	84.4/77.8	0.82	10^{-4}	10^{-3}	80.0±5.7	75.6/84.4	0.88	10	2
Both	Fusion	77.8±6.1	57.8/97.8	0.89	5	83.3±7.9	82.2/84.4	0.91	10^{-4}	10^{-3}	84.4±5.4	82.2/86.7	0.90	20	30
Average		**76.8**	**61.7/90.0**	**0.8**	–	**80.9**	**77.8/83.9**	**0.8**	–	–	**82.2**	**78.0/86.4**	**0.9**	–	–
STD		**4.2**	**10.5/5.1**	**0,0**	–	**2.9**	**7.0/4.0**	**0,1**	–	–	**3.1**	**4.8/7.2**	**0,0**	–	–

represent the results in a more compact way and it is a standard measure of performance in medical applications. The three classifiers produce similar results for both experiments. The impact of age in the results is also observed.

In addition, Figures 6.2 to 6.4 (for NLD), 6.6 to 6.8 (for Poincaré) and 6.10 to 6.12 (for NLD+Poincaré) show the scores of each classifier. In KNN and RF, the score is the probability with which sample belongs to the selected class and in SVM is the distance of the hyperplane to the sample.

According to KNN scores (see Figure 6.2) in NLD and entropy features, most of the YHC subjects were accurately classified but it is affected by the age factor. Figure 6.3 illustrates the distances between PD patients and EHC subjects according to their representation in the classification hyperspace, i.e., according to their distance to the hyper-

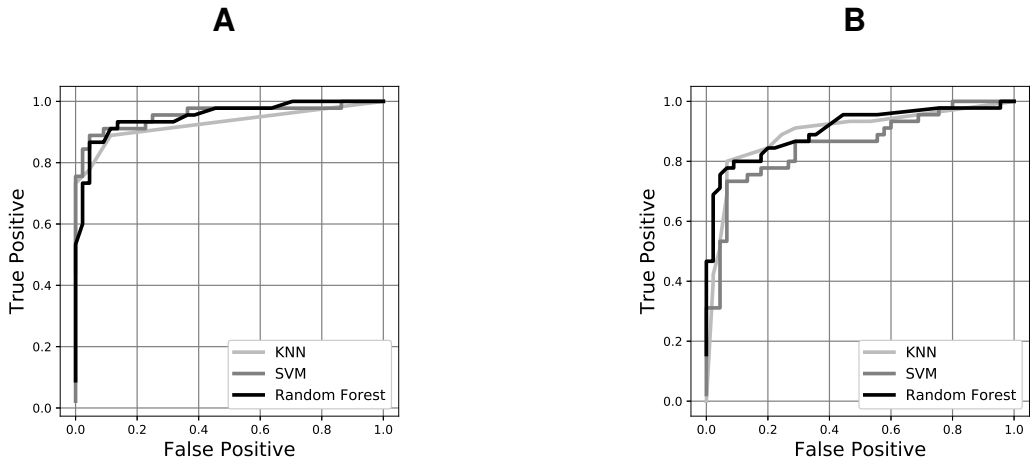

Figure 6.1: ROC curve graphics of the best NLD Features results. A) PD vs YHC. B) PD vs EHC. In both cases the fusion of features from both feet and both tasks are considered.

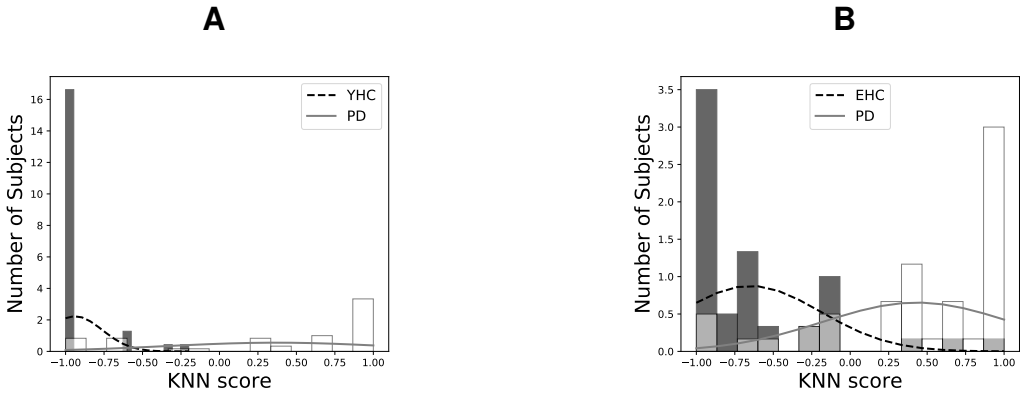

Figure 6.2: KNN Scores of NLD Features: A) PD vs YHC Fusion Both Feet task. B) PD vs EHC Fusion Both Feet task.

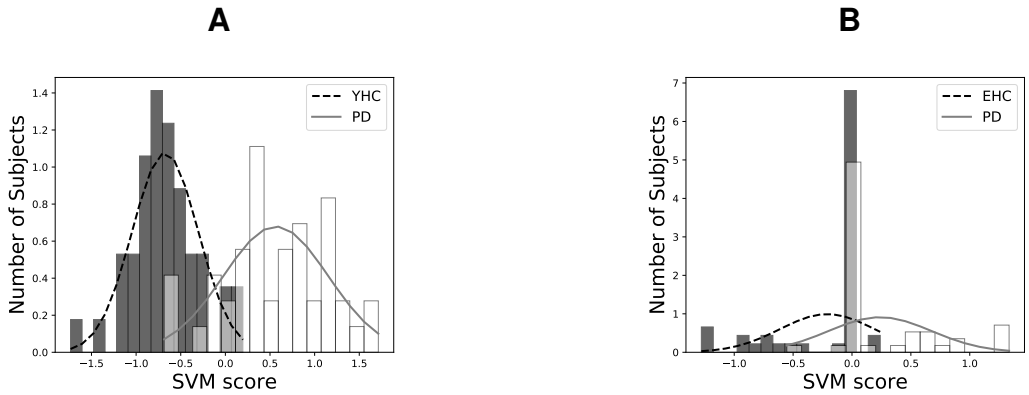

Figure 6.3: SVM Scores of NLD Features: A) PD vs YHC Fusion Both Feet task. B) PD vs EHC Fusion Both Feet task.

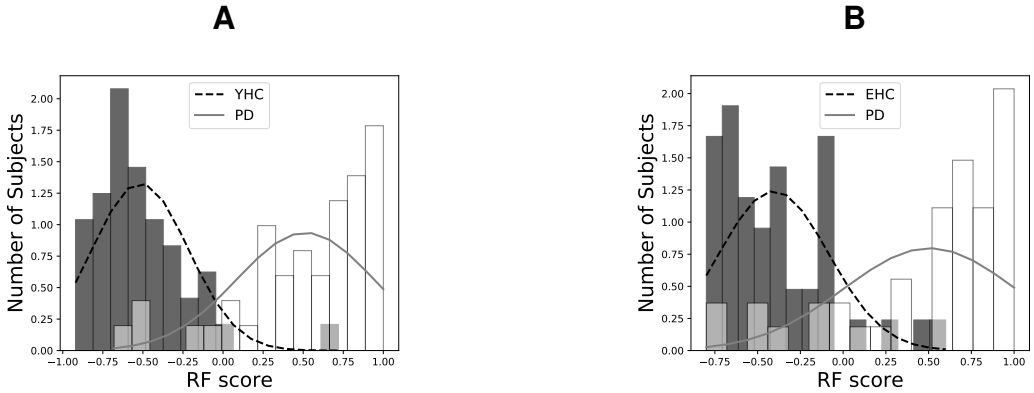

Figure 6.4: RF Scores of NLD Features: A) PD vs YHC Fusion Both Feet task. B) PD vs EHC Fusion Both Feet task.

plane of the SVM. Figure 6.4 shows that for the experiment PD vs YHC there are not large amount of miss–clasified respect to the YHC in comparison with PD vs EHC where some patients were miss-classified, being the closeness between the mean age of these populations an important factor for this result.

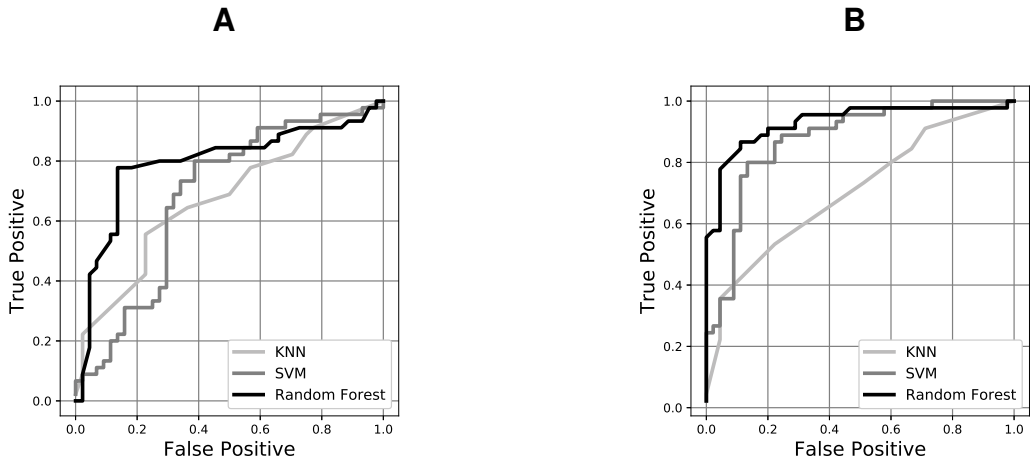

Figure 6.5: ROC curve graphics of the best Poincaré Features results. A) PD vs YHC Fusion Right Foot task. B) PD vs EHC Fusion Both Feet task.

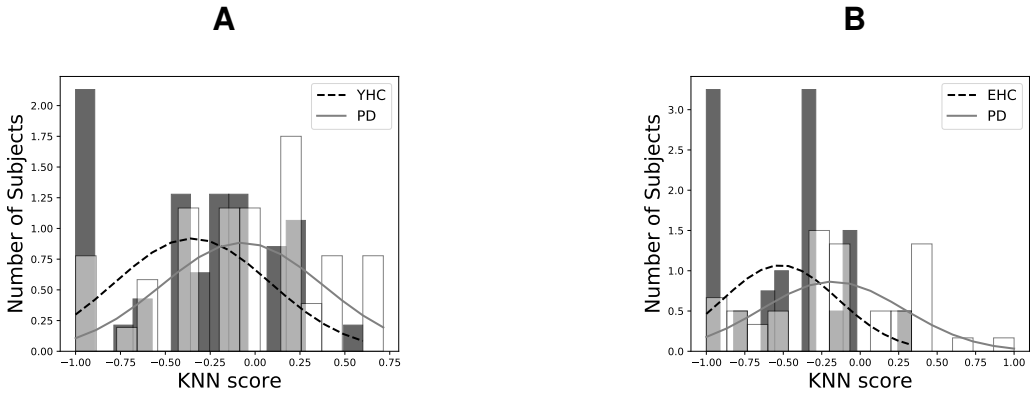

Figure 6.6: KNN Scores of Poincaré Features: A) PD vs YHC Fusion Right Foot task. B) PD vs EHC Fusion Both Feet task.

In Poincaré features the scores respect to KNN in Figure 6.6 that present a lot of miss-classified data. Figure 6.7 shows the SVM Scores, where for PD vs. YHC the distances of the SVM increased, but the data is close to each other. For PD vs. EHC the range of distances are lower, but the data is well defined in the decision boundary and the data is accurately classified. In Figure 6.8 for PD vs. EHC there is not any sample with probability 1, but despite this, the classification indicates good performance.

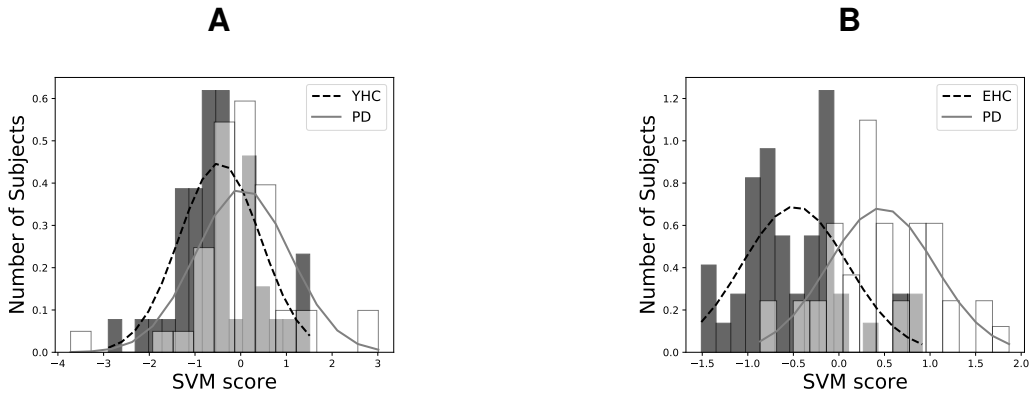

Figure 6.7: SVM Scores of Poincaré Features: A) PD vs YHC Fusion Right Foot task. B) PD vs EHC Fusion Both Feet task.

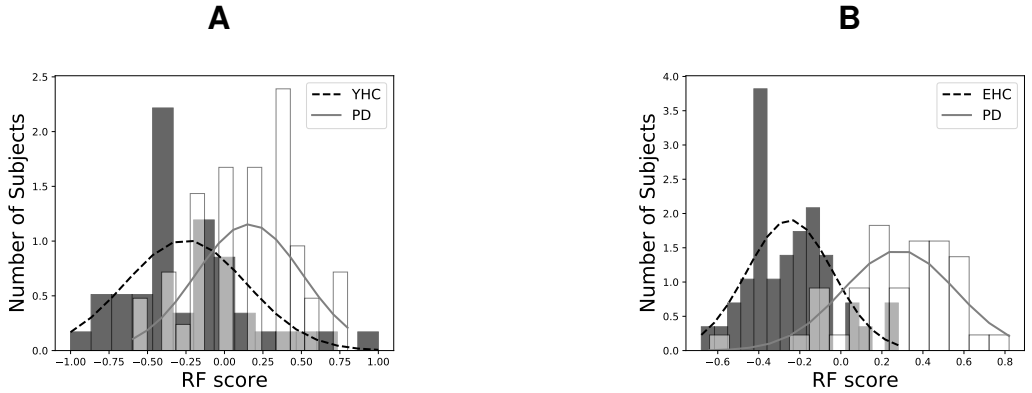

Figure 6.8: RF Scores of Poincaré Features: A) PD vs YHC Fusion Right Foot task. B) PD vs EHC Fusion Both Feet task.

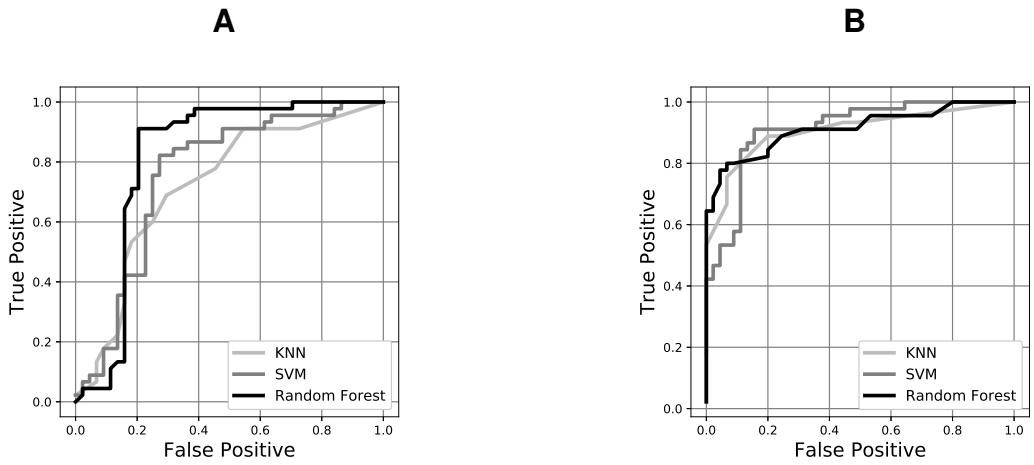

Figure 6.9: ROC curve graphics of the best NLD+Poincaré results. A) PD vs YHC 4x10 Left Foot task. B) PD vs EHC Fusion Left Foot task.

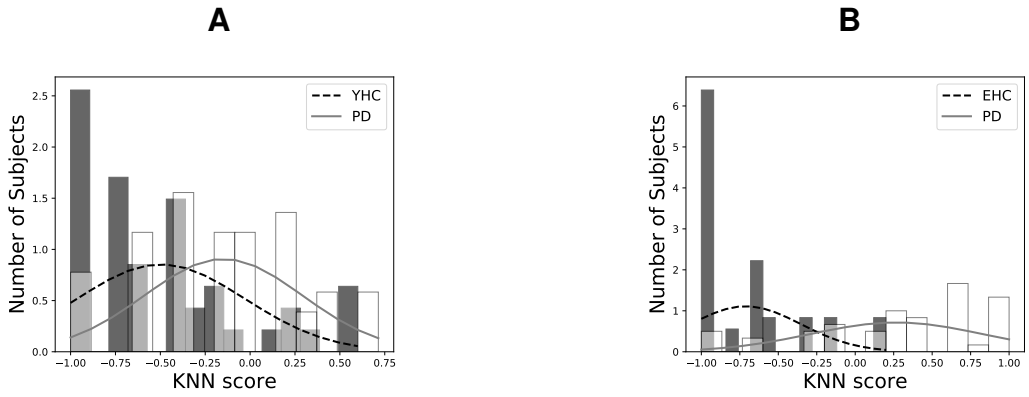

Figure 6.10: KNN Scores of NLD+Poincaré Features: A) PD vs YHC 4x10 Left Foot task. B) PD vs EHC Fusion Left Foot task.

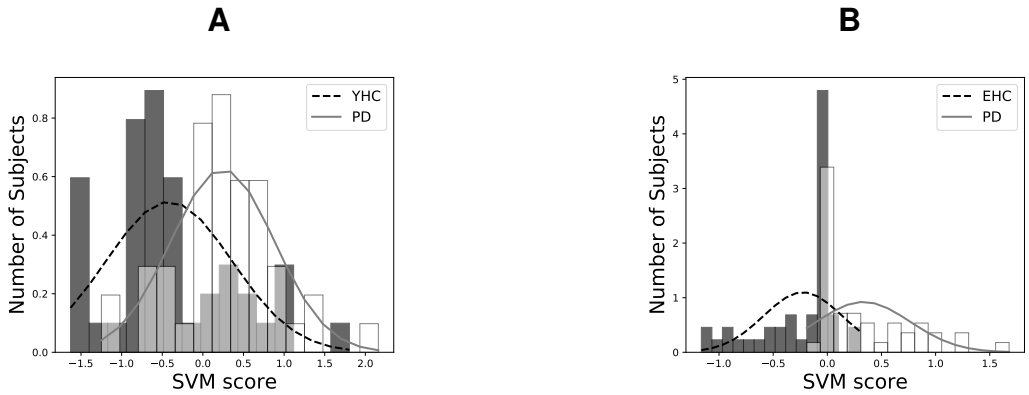

Figure 6.11: SVM Scores of NLD+Poincaré Features: A) PD vs YHC 4x10 Left Foot task. B) PD vs EHC Fusion Left Foot task.

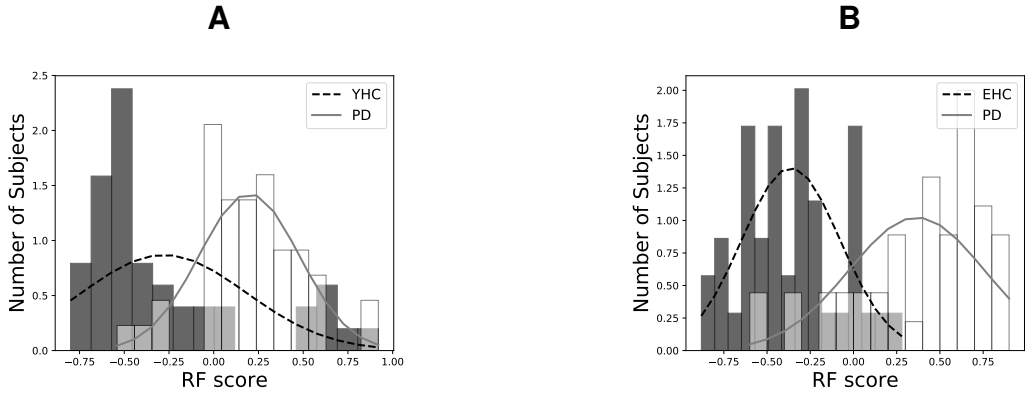

Figure 6.12: RF Scores of NLD+Poincaré Features: A) PD vs YHC 4x10 Left Foot task. B) PD vs EHC Fusion Left Foot task.

In NLD+Poincaré the score shows us that it accomplishes the same pattern that in the other set of features, the difference is PD vs. EHC obtained good performance and taking into account that the average age of the patient is 61.95 years old and the average age of EHC subjects is 62.65 years old, the effect of the age do not affect the results in PD vs. EHC.

6.3 Regression: Neurological State Prediction

Two experiments were performed according to the MDS-UPDRS-III and to the MDS-UPDRS-III subscore for lower limbs. Table 6.8 shows the results of the prediction of the neurological state of the patients according to the MDS-UPDRS-III. The highest Spearman's correlation (ρ) was obtained by NLD features, where they up to $\rho = 0.65,\ \mathrm{pval} < 0.005$ in Left foot fusion task.

Table 6.8: Results of the Neurological State Prediction with the total UPDRS. **MAE:** mean absolute error. $\boldsymbol{\rho}$: Spearman's Correlation.

		NLD Features					**Poincaré Features**					**NLD+Poincaré Features**				
Foot	**Task**	ρ	pval	**MAE**	**C**	ε	ρ	pval	**MAE**	**C**	ε	ρ	pval	**MAE**	**C**	ε
Left	2x10	0.46	<0.005	15.40	10^{0}	10^{1}	0.44	<0.005	15.46	10^{0}	10^{0}	0.45	<0.005	14.29	10^{-5}	10^{-5}
Left	4x10	0.39	0.01	13.80	10^{-1}	10^{0}	0.18	0.24	17.99	10^{-5}	10^{0}	0.33	0.03	13.70	10^{-5}	10^{1}
Left	Fusion	0.65	<0.005	13.30	10^{-1}	10^{0}	0.26	0.08	18.00	10^{-1}	10^{0}	0.05	0.73	15.85	10^{-5}	10^{-5}
Right	2x10	0.50	<0.005	12.95	10^{-5}	10^{1}	0.07	0.64	15.27	10^{-2}	10^{0}	0.34	0.02	15.46	10^{0}	10^{1}
Right	4x10	0.01	0.93	17.04	10^{-5}	10^{-5}	0.47	<0.005	15.07	10^{0}	10^{1}	0.40	0.01	15.28	10^{0}	10^{-5}
Right	Fusion	0.50	<0.005	14.76	10^{0}	10^{1}	0.24	0.12	18.21	10^{0}	10^{-5}	0.22	0.16	14.90	10^{-2}	10^{1}
Both	2x10	0.44	<0.005	13.80	10^{-1}	10^{1}	0.24	0.11	16.26	10^{0}	10^{-5}	0.45	<0.005	13.98	10^{-2}	10^{1}
Both	4x10	0.21	0.17	14.10	10^{-5}	10^{0}	0.15	0.32	17.27	10^{0}	10^{-5}	<0.005	0.80	15.20	10^{-5}	10^{-5}
Both	Fusion	0.42	<0.005	14.18	10^{-2}	10^{0}	-0.10	0.53	16.11	10^{-2}	10^{-5}	-0.06	0.68	14.96	10^{-5}	10^{1}
Average		**0.40**	**0.12**	**14.37**	**–**	**–**	**0.22**	**0.23**	**16.63**	**–**	**–**	**0.24**	**0.29**	**14.85**	**–**	**–**
STD		**0.19**	**0.31**	**1.24**	**–**	**–**	**0.17**	**0.23**	**1.26**	**–**	**–**	**0.20**	**0.40**	**0.71**	**–**	**–**

Table 6.9 shows the results of the prediction of the neurological state of the patients according to the MDS-UPDRS-III subscore for lower limbs. In general, the highest correlations are obtained for the NLD features too (ρ up to 0.31).

Table 6.9: Results of the Neurological State Prediction with the lower extremities UPDRS. **MAE:** mean absolute error. $\boldsymbol{\rho}$: Spearman's Correlation.

		NLD Features					**Poincaré Features**					**NLD+Poincaré Features**				
Foot	**Task**	ρ	pval	**MAE**	**C**	ε	ρ	pval	**MAE**	**C**	ε	ρ	pval	**MAE**	**C**	ε
Left	2x10	-0.13	0.38	10.19	10^{-2}	10^{1}	-0.31	0.04	9.10	10^{-1}	10^{-5}	-0.20	0.18	8.02	10^{-5}	10^{0}
Left	4x10	0.16	0.28	6.93	10^{-5}	10^{-5}	-0.15	0.33	11.74	10^{-4}	10^{0}	0.08	0.60	7.70	10^{-5}	10^{-5}
Left	Fusion	0.07	0.63	8.88	10^{-1}	10^{-5}	-0.09	0.56	8.44	10^{-4}	10^{0}	-0.15	0.32	8.37	10^{-5}	10^{0}
Right	2x10	0.19	0.21	8.43	10^{0}	10^{1}	0.16	0.30	10.82	10^{-5}	10^{1}	-0.02	0.91	9.44	10^{0}	10^{1}
Right	4x10	-0.00	0.98	10.77	10^{0}	10^{0}	-0.24	0.10	14.37	10^{-5}	10^{0}	0.17	0.26	9.64	10^{0}	10^{1}
Right	Fusion	0.31	0.03	8.00	10^{0}	10^{1}	-0.02	0.90	11.41	10^{1}	10^{0}	-0.05	0.77	8.71	10^{-2}	10^{1}
Both	2x10	0.26	0.07	8.25	10^{-1}	10^{0}	0.18	0.24	8.45	10^{0}	10^{-5}	0.16	0.31	8.98	10^{-5}	10^{1}
Both	4x10	0.30	0.04	7.57	10^{0}	10^{0}	0.09	0.54	9.41	10^{0}	10^{-5}	0.03	0.86	8.30	10^{-5}	10^{1}
Both	Fusion	0.31	0.03	8.26	10^{-1}	10^{1}	-0.09	0.58	8.46	10^{-2}	10^{1}	0.07	0.65	7.93	10^{-2}	10^{1}
Average		**0.17**	**0.30**	**8.59**	**–**	**–**	**-0.05**	**0.40**	**10.24**	**–**	**–**	**0.01**	**0.54**	**8.56**	**–**	**–**
STD		**0.16**	**0.33**	**1.21**	**–**	**–**	**0.17**	**0.27**	**2.02**	**–**	**–**	**0.13**	**0.27**	**0.68**	**–**	**–**

Although the MDS-UPDRS-III obtained highest results, but it has its counterpart, the signal was capture from all the body and the sensor were just attached in the feet. The reason because MDS-UPDRS-III has higher results than with the subscore of lower limbs is the range of the total UPDRS is larger and some parameters were a little bit affected by this.

6.4 Classification of patients in several sages of the disease

Although there is some correlation between the predicted and the real MDS-UPDRS-III scores, we believe that from the clinical point of view it is more suitable for the patients to know in which stage of the disease they are, rather than to have the prediction of a continuous scale. In addition, for medical applications is difficult to have a great amount of data to train suitable regression algorithms like an SVR. For those reasons we believe that it is better to divide the patients into three groups according to their neurological state: lower, intermediate, and severe. Figures 6.13 and 6.14 show the division of the neurological stage of the patients into the three groups. Additionally, EHC subjects are considered as a separate group to perform a four-class classification strategy.

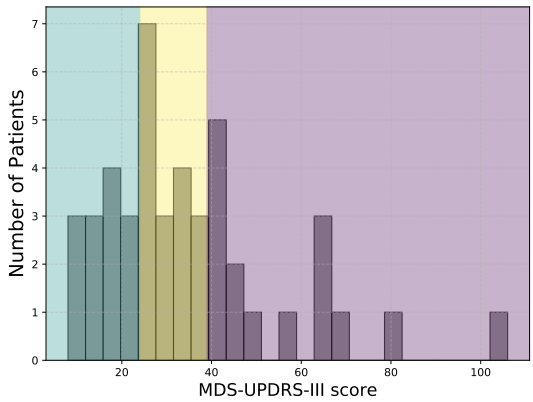

Figure 6.13: Histogram for the neurological state of the patients according to the MDS-UPDRS-III. Patients in initial stage (green), patients in intermediate stage (yellow), and patients in severe stage (purple).

Table 6.10 shows the results of classification applying NLD features and classes divided by MDS-UPDRS-III. The highest performance was obtained in Left foot 2x10 task for the SVM classifier with accuracies up to 64.4% and κ up to 0.41 (indicating a moderate result).

In Poincaré (see Table 6.11), the standard deviation was reduced indicating less data dispersion. The performance decreased obtaining as highest accuracy 58.9% that is given by Right foot fusion task.

In Table 6.12 is shown the results of the combination of features, obtaining accuracies up to 63.3% and κ up to 0.39, close to a moderate result.

Table 6.10: Results of Multi-Class Classification for NLD Measurements according to the MDS-UPDRS-III. **ACC:** accuracy in the test set. κ: Cohen's kappa index.

		KNN			SVM				RF			
Foot	**Task**	**ACC(%)** ($\mu \pm \sigma$)	κ	**K**	**ACC(%)** ($\mu \pm \sigma$)	κ	**C**	γ	**ACC(%)** ($\mu \pm \sigma$)	κ	**N**	**D**
Left	2x10	54.4±6.5	0.22	3	62.2±14.2	0.40	10^1	10^{-3}	64.4±9.6	0.41	50	20
Left	4x10	56.7±4.2	0.25	5	58.9±9.6	0.32	10^1	10^{-3}	60.0±5.2	0.33	20	5
Left	Fusion	61.1±10.5	0.35	5	56.7±11.3	0.24	10^1	10^{-4}	62.2±9.5	0.37	50	20
Right	2x10	52.2±4.4	0.17	5	61.1±9.3	0.35	10^1	10^{-3}	55.6±6.1	0.28	10	10
Right	4x10	50.0±7.0	0.20	5	61.1±7.8	0.38	10^1	10^{-4}	55.6±3.5	0.24	10	10
Right	Fusion	55.6±4.9	0.25	5	55.6±12.1	0.29	10^1	10^{-3}	53.3±14.3	0.20	20	20
Both	2x10	55.5±11.6	0.27	5	52.2±8.3	0.20	10^1	10^{-3}	62.2±10.7	0.38	50	10
Both	4x10	50.0±6.0	0.18	5	55.6±9.3	0.29	10^1	10^{-3}	62.2±8.8	0.35	10	20
Both	Fusion	56.7±5.4	0.29	3	51.1±5.4	0.18	10^1	10^{-3}	60.0±8.8	0.36	5	10
Average		**54.7**	**0.24**	–	**57.2**	**0.29**	–	–	**59.5**	**0.32**	–	–
STD		**3.6**	**0.1**	–	**4.0**	**0.1**	–	–	**3.8**	**0.1**	–	–

Table 6.11: Results of Multi-Class Classification for Poincaré Measurements according to the MDS-UPDRS-III. **ACC:** accuracy in the test set. κ: Cohen's kappa index.

		KNN			SVM				RF			
Foot	**Task**	**ACC(%)** ($\mu \pm \sigma$)	κ	**K**	**ACC(%)** ($\mu \pm \sigma$)	κ	**C**	γ	**ACC(%)** ($\mu \pm \sigma$)	κ	**N**	**D**
Left	2x10	51.1±8.2	0.14	5	48.9±2.2	0.13	10^1	10^{-3}	55.5±3.5	0.22	5	20
Left	4x10	54.4±9.6	0.18	5	55.6±9.3	0.29	10^1	10^{-4}	52.2±4.4	0.17	50	5
Left	Fusion	52.2±2.7	0.13	5	50.0±5.0	0.14	10^1	10^{-4}	52.2±2.7	0.13	20	5
Right	2x10	52.2±2.7	0.14	7	55.6±7.0	0.20	10^1	10^{-3}	55.6±6.1	0.21	50	20
Right	4x10	44.4±6.1	0.00	9	51.1±7.8	0.10	10^1	10^{-3}	54.4±4.2	0.21	10	5
Right	Fusion	50.0±3.5	0.11	3	50.0±3.5	0.17	10^{-4}	10^{-4}	58.9±6.7	0.26	50	50
Both	2x10	47.8±4.4	0.00	7	47.8±4.4	0.11	10^{-4}	10^{-4}	53.3±4.4	0.15	10	10
Both	4x10	46.7±4.4	0.00	5	51.1±5.4	0.20	10^1	10^{-4}	55.6±5.0	0.18	10	5
Both	Fusion	52.2±4.4	0.11	9	52.2±7.5	0.19	10^1	10^{-3}	54.4±4.4	0.18	10	50
Average		**50.1**	**0,09**	–	**51.4**	**0.17**	–	–	**49.3**	**0.19**	–	–
STD		**3.2**	**0.1**	–	**2.7**	**0.1**	–	–	**16.6**	**0.0**	–	–

Table 6.13 shows the confusion matrices for the best results of the multi-class experiment for MDS-UPDRS-III when we consider only the NLD features, only the Poincaré based measures, and the combination of both feature sets. The confusion matrices for the three feature sets indicate that is easier to the classifiers discriminate EHC subjects than discriminate the PD patients in different stages, i.e., the proposed approach has a high specificity. The third group of PD patients (severe stage) presented difficulties in the classification, it is because this class (PD_3) contains big variability in its data(see Figure 6.13).

Table 6.12: Results of Multi-Class Classification for NLD+Poincaré Measurements according to the MDS-UPDRS-III. **ACC:** accuracy in the test set. κ: Cohen's kappa index.

		KNN			**SVM**				**RF**			
Foot	**Task**	**ACC(%)** $(\mu \pm \sigma)$	κ	**K**	**ACC(%)** $(\mu \pm \sigma)$	κ	**C**	γ	**ACC(%)** $(\mu \pm \sigma)$	κ	**N**	**D**
Left	2x10	57.8±12.4	0.32	5	55.6±3.5	0.29	10^1	10^{-3}	56.7±5.4	0.28	5	5
Left	4x10	54.4±4.2	0.20	5	55.6±13.6	0.29	10^1	10^{-3}	57.8±7.5	0.27	20	5
Left	Fusion	60.0±4.2	0.34	3	58.9±6.7	0.36	10^1	10^{-3}	56.7±5.4	0.25	20	5
Right	2x10	47.8±2.7	0.01	7	57.6±8.3	0.31	10^1	10^{-4}	57.8±10.3	0.30	5	20
Right	4x10	51.1±9.6	0.16	7	46.7±2.7	0.10	10^1	10^{-4}	53.3±5.7	0.21	10	5
Right	Fusion	55.6±3.5	0.20	3	50.0±3.5	0.10	10^{-4}	10^{-3}	55.6±5.0	0.23	30	20
Both	2x10	52.2±4.4	0.14	3	57.8±5.7	0.33	10^1	10^{-3}	56.7±4.2	0.26	10	10
Both	4x10	55.6±8.6	0.22	5	53.3±9.0	0.21	10^1	10^{-3}	58.9±7.5	0.29	30	20
Both	Fusion	58.9±6.7	0.29	3	60.0±7.4	0.34	10^1	10^{-4}	63.3±9.0	0.39	50	5
Average		**54.8**	**0.21**	–	**55.1**	**0.26**	–	–	**57.4**	**0.28**	–	–
STD		**3.9**	**0.1**	–	**4.4**	**0.1**	–	–	**2.7**	**0.1**	–	–

Table 6.13: Confusion Matrices of Multi-Class Classification for best results in each experiment according to the MDS-UPDRS-III. **EHC:** EHC subjects. **PD_1:** MDS-UPDRS III (Lower Limbs) ranging between 0-10. **PD_2:** MDS-UPDRS III (Lower Limbs) ranging between 11-17. **PD_3:** MDS-UPDRS III (Lower Limbs) of 18+.

	NLD Feature Left 2x10 RF				**Poincaré Features Right Fusion RF**				**NLD+Poincaré Features Both Fusion RF**			
Class	**EHC**	**PD_1**	**PD_2**	**PD_3**	**EHC**	**PD_1**	**PD_2**	**PD_3**	**EHC**	**PD_1**	**PD_2**	**PD_3**
EHC	45	0	0	0	44	0	1	0	45	0	0	0
PD_1	3	2	5	4	9	3	2	0	4	1	6	3
PD_2	6	2	7	1	9	0	6	1	5	3	8	0
PD_3	4	2	5	4	10	3	2	0	4	1	7	3

But taking into account that the signals were capture only from the gait, it is a best comparison a class division based on the MDS-UPDRS-III subscore of lower limbs.

Table 6.14 shows the results of performing a Multi-Class Classification for the NLD Features using the MDS-UPDRS-III subscore for lower limbs. The highest accuracies are also obtained with the signals captured with the combination of features from both sensors and with the RF classifier (63.3%), as in the previous experiments when we discriminate only PD patients and EHC subjects. The κ-index (0.41) indicates moderate agreement in the classification.

The results when the Poincaré features are considered are shown in Table 6.15. In general, the accuracy is reduced in around 3.4% respect to the obtained with the classical NLD and entropy features. However, note that the standard deviation is lower when we consider the Poincaré features, which make the results more stable across different partitions of the test set.

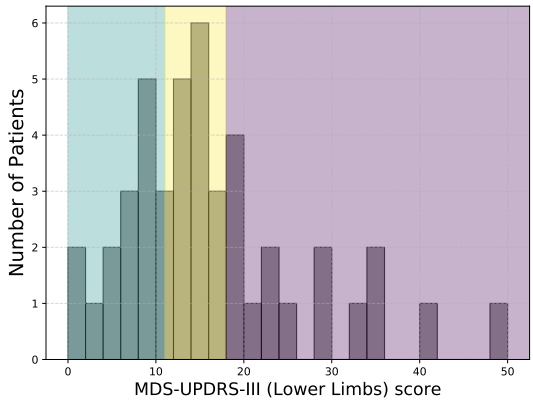

Figure 6.14: Histogram for the neurological state of the patients according to the MDS-UPDRS-III subscore for lower limbs. Patients in initial stage (green), patients in intermediate stage (yellow), and patients in severe stage (purple).

Table 6.14: Results of Multi-Class Classification for NLD Measurements according to the MDS-UPDRS-III subscore for lower limbs. **ACC:** accuracy in the test set. κ: Cohen's kappa index.

		KNN			**SVM**				**RF**			
Foot	**Task**	**ACC(%)** ($\mu \pm \sigma$)	κ	**K**	**ACC(%)** ($\mu \pm \sigma$)	κ	**C**	γ	**ACC(%)** ($\mu \pm \sigma$)	κ	**N**	**D**
Left	2x10	54.4±6.5	0.22	3	61.1±9.9	0.36	10^1	10^{-3}	61.1±7.0	0.37	50	10
Left	4x10	56.7±4.2	0.25	5	58.9±9.6	0.32	10^1	10^{-3}	60.0±4.2	0.33	30	30
Left	Fusion	62.2±9.5	0.37	5	62.2±13.3	0.40	10^1	10^{-3}	61.1±12.1	0.34	10	20
Right	2x10	52.2±4.4	0.17	5	57.8±5.6	0.29	10^1	10^{-4}	57.8±10.3	0.30	30	20
Right	4x10	52.2±9.0	0.19	5	58.9±5.6	0.34	10^1	10^{-4}	50.0±11.3	0.15	5	10
Right	Fusion	55.6±4.9	0.25	5	54.4±8.8	0.27	10^1	10^{-3}	56.7±8.8	0.25	20	10
Both	2x10	56.7±9.6	0.28	5	55.6±6.0	0.24	10^1	10^{-3}	63.3±6.7	0.40	50	10
Both	4x10	50.0±6.0	0.18	5	56.7±9.5	0.30	10^1	10^{-3}	55.6±9.2	0.23	5	5
Both	Fusion	60.0±8.1	0.36	5	51.1±5.4	0.18	10^1	10^{-3}	62.2±7.3	0.40	5	10
Average		**55.6**	**0.25**	–	**50.7**	**0.30**	–	–	**58.6**	**0.31**	–	–
STD		**3.9**	**0,1**	–	**18.4**	**0,1**	–	–	**4.1**	**0.1**	–	–

The results when we combine the classical NLD and the Poincaré features are shown in Table 6.16. This early-fusion strategy shows to be the most accurate to discriminate between PD patients in several stages of the disease and EHC subjects. We obtain accuracies up to 65.2%, which is slightly higher than the obtained only with the NLD features. This fact indicates that our proposed features based on Poincaré sections and GMM models provide complementary information to the classical NLD analysis about the non-linear effects of the gait process of PD patients.

Table 6.15: Results of Multi-Class Classification for Poincaré Measurements according to the MDS-UPDRS-III subscore for lower limbs. **ACC:** accuracy in the test set. κ: Cohen's kappa index.

		KNN			**SVM**				**RF**			
Foot	**Task**	**ACC(%)** $(\mu \pm \sigma)$	κ	**K**	**ACC(%)** $(\mu \pm \sigma)$	κ	**C**	γ	**ACC(%)** $(\mu \pm \sigma)$	κ	**N**	**D**
Left	2x10	51.1±8.2	0.14	5	51.1±4.2	0.18	10^1	10^{-3}	50.0±3.5	0.10	20	5
Left	4x10	54.4±9.5	0.18	5	56.7±13.3	0.29	10^1	10^{-4}	50.0±3.5	0.10	5	50
Left	Fusion	52.2±2.7	0.13	5	56.7±9.5	0.28	10^1	10^{-4}	52.2±6.6	0.17	5	20
Right	2x10	52.2±2.7	0.14	7	55.6±5.0	0.22	10^1	10^{-3}	57.8±9.6	0.25	30	20
Right	4x10	42.2±4.4	-0.10	9	51.1±2.2	0.10	10^1	10^{-4}	53.3±5.6	0.16	20	2
Right	Fusion	50.0±3.5	0.10	3	51.1±2.2	0.11	10^{-4}	10^{-4}	53.3±8.3	0.21	20	30
Both	2x10	47.8±5.6	0.04	7	48.9±2.2	0.10	10^{-4}	10^{-4}	54.4±4.2	0.22	10	30
Both	4x10	46.7±4.4	-0.01	7	51.1±8.8	0.20	10^1	10^{-3}	50.0±3.5	0.12	10	5
Both	Fusion	48.9±2.2	0.05	9	51.1±7.3	0.18	10^1	10^{-3}	53.3±5.7	0.15	10	5
Average		**49.5**	**0.07**	–	**52.6**	**0.18**	–	–	**52.7**	**0.16**	–	–
STD		**3.6**	**0.1**	–	**2.9**	**0.1**	–	–	**2.5**	**0.1**	–	–

Table 6.16: Results of Multi-Class Classification for NLD+Poincaré Measurements according to the MDS-UPDRS-III subscore for lower limbs. **ACC:** accuracy in the test set. κ: Cohen's kappa index.

		KNN			**SVM**				**RF**			
Foot	**Task**	**ACC(%)** $(\mu \pm \sigma)$	κ	**K**	**ACC(%)** $(\mu \pm \sigma)$	κ	**C**	γ	**ACC(%)** $(\mu \pm \sigma)$	κ	**N**	**D**
Left	2x10	49.5±6.6	0.08	9	58.5±5.1	0.32	10^1	10^{-3}	57.1±9.3	0.26	5	10
Left	4x10	50.1±3.7	0.12	5	60.1±6.1	0.34	10^1	10^{-3}	58.4±7.8	0.26	100	20
Left	Fusion	57.3±4.1	0.21	5	65.2±8.1	0.43	10^1	10^{-3}	61.8±5.2	0.33	30	10
Right	2x10	53.9±4.1	0.11	9	55.1±5.7	0.25	10^1	10^{-3}	52.8±2.5	0.14	10	2
Right	4x10	53.9±7.0	0.16	5	60.7±7.1	0.34	10^1	10^{-4}	58.9±4.9	0.24	10	50
Right	Fusion	56.1±4.4	0.18	5	54.0±9.2	0.20	10^1	10^{-3}	60.6±5.6	0.31	30	5
Both	2x10	52.9±5.0	0.11	5	60.7±9.0	0.29	10^1	10^{-3}	61.8±5.6	0.34	10	30
Both	4x10	56.2±6.2	0.20	5	61.8±6.2	0.37	10^1	10^{-3}	51.6±3.3	0.14	5	2
Both	Fusion	59.6±6.1	0.25	5	64.1±8.0	0.38	10^1	10^{-3}	59.5±4.1	0.28	5	50
Average		**55.0**	**0.16**	–	**60.0**	**0.32**	–	–	**58.0**	**0.26**	–	–
STD		**3.7**	**0.1**	–	**3.7**	**0.1**	–	–	**3.6**	**0.1**	–	–

Table 6.17 shows the confusion matrices for the best results of the multi-class experiment when we consider only the NLD features, only the Poincaré based measures, and the combination of both feature sets. The confusion matrices for the three feature sets indicate that EHC subjects are accurately classified compared to patients in different stages, i.e., the proposed approach has a high specificity. The group of PD patients in severe stage of the disease (PD_3) seems to be the most difficult group to classify correctly, which could be explained due to the high variability of the original MDS-UPDRS-III score

for those patients (see Figure 6.14).

Table 6.17: Confusion Matrices of Multi-Class Classification for best results in each experiment according to the MDS-UPDRS-III subscore for lower limbs. **EHC:** EHC subjects. **PD_1:** MDS-UPDRS III (Lower Limbs) ranging between 0-10. **PD_2:** MDS-UPDRS III (Lower Limbs) ranging between 11-17. **PD_3:** MDS-UPDRS III (Lower Limbs) of 18+.

	NLD Features Both 2x10m RF				**Poincaré Features Right 2x10m RF**				**NLD+Poincaré Features Left Fusion SVM**			
Class	**EHC**	**PD_1**	**PD_2**	**PD_3**	**EHC**	**PD_1**	**PD_2**	**PD_3**	**EHC**	**PD_1**	**PD_2**	**PD_3**
EHC	45	0	0	0	43	0	2	0	41	1	3	0
PD_1	3	2	5	5	8	4	2	1	5	4	3	2
PD_2	7	1	5	2	9	1	4	1	5	0	11	0
PD_3	3	2	5	5	8	4	2	1	6	4	3	2

Chapter 7

Conclusions.

An automatic assessment of the gait in PD patients is proposed in this study. A NLD approach is considered to evaluate stability, long-term dependency, and complexity of the walking process of the patients. The study includes classical NLD features such as CD, LLE, HE, LZC and several entropy measures. In addition, we proposed a new characterization scheme based on a clustering analysis of Poincaré sections using a GMM algorithm. An automatic discrimination between PD patients and two groups of HC subjects is performed to assess the impact of age in the walking process and also the extracted features were used to predict the neurological state of the patients, and to classify patients in several stages of the disease. The set of NLD features included features computed from the phase space and several entropy measures.

The results show that it is possible to discriminate between PD patients and HC subjects with accuracies up to 93.3% for YHC subject and 85.6% for EHC subjects, using the proposed NLD analysis. The proposed NLD features based on Poincaré sections also provide complementary information to the classical NLD features to discriminate between PD patients and EHC subjects. While in the PD patients vs. YHC subjects experiment, NLD presents higher performance than other set of features.

The proposed approach seems to be promising to classify PD patients in different stages of the disease. The combination of the classical NLD features with the proposed features from Poincaré sections is also the most accurate approach to classify PD patients in several stages of the disease (up to 65.2%) and was achieved a moderate result with Cohen Kappa up to 0.43. Patients in lower stages of disease are mainly miss-classified with EHC subjects, which is explained due to in the early stages of the disease, the symptoms appear mainly only in the upper limbs. PD patients in severe stages of the disease are miss-classified mainly due to the those patients are more spread out in terms of the original MDS-UPDRS scale than the other groups of patients. Additional labeled data is need to improve the accuracy to classify patients in several stages of the disease.

The combination of features extracted from different tasks and from both feet is more

effective in the classification process, i.e., both tasks and feet provide complementary information to discriminate between PD patients and HC subjects. The most correct way to model the gait signal obtained by inertial sensor attached to the foot, is using the MDS-UPDRS-III subscore of the lower limbs. The results also indicate the presence of the cross laterality effect [47], since higher accuracies are obtained classifying the features computed from the left foot rather than those computed from the right foot, although most of the subjects from this study are right-handed.

The proposed approach can be extended to other applications. For instance the discrimination between PD and other neurological disorders with similar symptoms, such as Huntington's disease, amyotrophic lateral sclerosis, or essential tremor. There is evidence for this application in the literature [19, 48]. The proposed features can also be considered to analyze movement signals. Besides, following the same line of NLD, it can be considered other features such as Recurrency Quantification Analysis (RQA), that has been applied in gait analysis in different injuries and diseases as in related work [19, 49].

Appendices

Appendix A

Kolgomorov-Sinai entropy

The entropy is a thermodynamic quantity that describes the amount of disorder in a system. This concept can be generalized to characterize the amount of information stored in a more general probability distribution. The information theory provides an important approach to the time series analysis, considering a sample of a system as a information source. An entropy for an static distribution quantifies the amount of information which is necessary to specify a sample value $\boldsymbol{x}$ with certain precision. When it is known the probability density $d\mu/d\boldsymbol{x}$. Then, we cover the space where the sample is with disjoint boxes P_j of side length $\leq \epsilon$. Defining $P_j = \int p_j \, d\mu(x)$ is the fraction into the jth box that in Equation A.1 is defined it, where P_ϵ is set boxes and order$-q$ refers to Renyi entropy.

$$\widetilde{H_q}(P_\epsilon) = \frac{1}{1-q} ln \sum_j p_j^q \tag{A.1}$$

When $q = 1$ can be evaluated the l'Hospital rule, being $\widetilde{H_1}$ the Shannon entropy as is shown in Equation A.2

$$\widetilde{H_1}(P_\epsilon) = ln \sum_j p_j^q \tag{A.2}$$

Introducing P_ϵ in a dynamic range of a sample, being scalar samples which is common in a time series situation. Introducing the joint probability $p_{i_1,i_2,\ldots,i_m}$, being the Kolmogorov-Sinai entropy that defines block entropies of block with size m, show in Equation A.3.

$$H_q(m, P_\epsilon) = \frac{1}{1-q} ln \sum_{i_1,i_2,\ldots,i_m} p_{i_1,i_2,\ldots,i_m}^q \tag{A.3}$$

The order-q entropies are then given in Equation A.4, where Sup indicates it must be maximize over all possible partitions P and usually implies $\epsilon \to 0$. Strictly h_1 is defined as the Kolgomorov-Sinai entropy.

$$h_q = \underset{P}{Sup} \lim_{m\to\infty} \frac{1}{m} H_q(m, P_\epsilon) \equiv \underset{P}{Sup} \lim_{m\to\infty} h_q(m, P_\epsilon) \tag{A.4}$$

Appendix B

Expectation maximization algorithm (EM)

This algorithm is an iterative optimization method to estimate the maximum likelihood of the parameters that best represent the statistics of a random variable. It is modeled using a parametric model, the random variable behaviour x, in a sense of maximum likelihood shown in the equation B.1.

$$\varphi = max_{\varphi}[L_x(\varphi|x^n)] \tag{B.1}$$

Where x^n is a set of n complete data (features). The likelihood in equation B.2, given the data is the same as probability density assumed to this observed data.

$$L(\varphi|\boldsymbol{x}) = P(\boldsymbol{x}|\varphi) \tag{B.2}$$

Usually the log-likelihood (equation B.3) is more used than the above.

$$L(\varphi|\boldsymbol{x}) = ln(P(\boldsymbol{x}|\varphi)) \tag{B.3}$$

After of the k–th iteration $L(\varphi|\boldsymbol{x})$ is increase, searching maximize the following difference:

$$\varphi = \arg\max_{\varphi}[L_x(\varphi^{(k)}|\boldsymbol{x}) - L_x(\varphi^{(k-1)}|\boldsymbol{x})] \tag{B.4}$$

Appendix C

Optimization process

C.1 Optimization process SVM

To the optimization problem solution it is performed by the following steps:

1. Lagrangian Function: now taking into account the constrains to the minimization of the Equation 4.6, to perform the optimization the Lagrangian is used and given by Equation C.1. $a_i \geq 0$ and $\mu_i \geq 0$ are the Lagrange multipliers.

$$L = \frac{1}{2}||\omega||^2 + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N} a_i \left\{\hat{y}_i y_i - 1 + \xi_i\right\} - \sum_{i=1}^{N}\mu_i\xi_i \tag{C.1}$$

2. Karush Kuhn Tucker (KKT) conditions: the constrained optimization is performed with KKT conditions, which are given by:

$$a_n \geq 0 \tag{C.2}$$

$$\hat{y}_i y_i - 1 + \xi_i \geq 0 \tag{C.3}$$

$$a_i(\hat{y}_i y_i - 1 + \xi_i) = 0 \tag{C.4}$$

$$\mu_i \geq 0 \tag{C.5}$$

$$\xi_i \geq 0 \tag{C.6}$$

$$\mu_i \xi_i = 0 \tag{C.7}$$

3. Now, the set of derivatives of the Lagragian respect to $\boldsymbol{\omega}, b$ and ξ_i to zero.

$$\frac{\partial L}{\partial \boldsymbol{\omega}} = \omega - \sum_{i=1}^{N} a_i \hat{y}_i \phi(x_i) = 0 \tag{C.8}$$

$$\frac{\partial L}{\partial b} = -\sum_{i=1}^{N} a_i \hat{y}_i = 0 \tag{C.9}$$

$$\frac{\partial L}{\partial \xi_i} = C - \mu_i - a_i = 0 \tag{C.10}$$

4. Relation between variables:

$$\omega = \sum_{i=1}^{N} a_i \hat{y}_i \phi(x_i) \tag{C.11}$$

$$a_i = C - \mu_i \tag{C.12}$$

5. Replace in the Lagrangian function:

$$L = \sum_{i=1}^{N} a_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{n=1}^{N} a_i a_n \hat{y}_i \hat{y}_n k(\boldsymbol{x_i}, \boldsymbol{x_n}) \tag{C.13}$$

   Where kernel function is defined by $k(x_i, x_n) = \phi(x_i)^T \phi(x_n)^T$.

6. Additional constraints: there are additional constraints respect to a_i, as $\mu_i \geq 0$, $a_i \geq 0$ and referring to the Equation C.12, therefore have to maximize with respect to a_i subject to:

$$0 \leq a_i \leq C \Rightarrow \text{Box Constraints} \tag{C.14}$$

   Now, replacing Equation C.11 into Equation 4.5, we can observe the predictions for new data using Equation C.15.

$$y(\boldsymbol{x}) = \sum_{i=1}^{N} a_i \hat{y}_i k(\boldsymbol{x}, \boldsymbol{x_i}) + b \tag{C.15}$$

We can observe that a subset of data points could be $a_i = 0$ presenting no contribution to the model.Thus, when $a_i > 0$ the data points can be support vectors. Then, if $a_i > 0$ it must satisfy that $\hat{y}_i y_i = 1 - \xi_i$. If $a_i < C$ and $\mu_i > 0$, it requires that $\xi_i = 0$ lying these points on the margin. and finally, when $a_i = C$ may stay inside the margin and maybe miss-classified if $\xi_i > 1$ or correctly classified if $\xi \leq 1$.

7. To find the independent term b: when $0 < a_i < C$ have $\xi_i = 0$ and the data points are support vectors, thus, in this case, $\hat{y}_i y_i = 1$ and thus, replacing that with the Equation C.15 results the Equation C.16, where $\boldsymbol{S}$ is the set of indices of the support vectors.

$$\hat{y}_i \left(\sum_{n \in \boldsymbol{S}} a_n \hat{y}_n k(\boldsymbol{x_n}, \boldsymbol{x_i}) \right) = 1 \tag{C.16}$$

Finally, solving the last equation for b, it is obtained the Equation C.17, where $\boldsymbol{M}$ is the set of indices of data points with $0 < a_i < C$.

$$b = \frac{1}{NM} \sum_{i \in \boldsymbol{M}} \left(\hat{y}_i - \sum_{n \in \boldsymbol{S}} a_n \hat{y}_n k(\boldsymbol{x_n}, \boldsymbol{x_i}) \right) \tag{C.17}$$

The previous description is only to discriminate between two classes; however, the SVM can be adpated to solve multi-class classification problems. To solve this, there are several methods to combine multiples two classes to build a multi-class classifiers. To perform the multi-class classification, we used a method called ***"one-vs-the-rest"*** (OVR), that consists of fitting a classifier per class. The intial approach of OVR requires certain unanimity between all SVMs, it means a data point could be classified if and only if this SVM's class is accepted it and the others rejected it. An advantage of this model is its interpretability, because it is possible to obtain some knowledge about the class inspecting its classifier.

C.1.1 Kernel Function

Kernel function converts what would be a nonlinear classification problem in an original space into a simple lineal classification problem into a grater dimensionality space. To implement it, it is chosen a feature space mapping $\phi(\boldsymbol{x})$ (basis function) , using this to find the corresponding kernel, defined in the Equation C.18 to one-dimensional [50].

$$k(\boldsymbol{x}, \boldsymbol{x'}) = \phi(\boldsymbol{x})^T \phi(\boldsymbol{x'}) \tag{C.18}$$

To chose the kernel function, it must correspond to a scalar product in some feature space. To see if the kernel is valid without to build a $\phi(\boldsymbol{x})$ function, it is necessary that satisfies the condition of that the matrix $k(\boldsymbol{x}, \boldsymbol{x'})$ should be positive semi-definite.

To build new kernel functions, it can be performed a simple building as if they were blocks. For our case, we chose a kernel function called *"Radial Basis Function"* (RBF) defined by the Equation C.19.

$$k(\boldsymbol{x}, \boldsymbol{x'}) = \exp[-\gamma||\boldsymbol{x} - \boldsymbol{x'}||^2] \tag{C.19}$$

The γ parameter establishes the width of the bell-shaped curve. RBF or better known as Gaussian kernel, has the property that each basis function depends only on the radial distance, most commonly used the Euclidean distance.

C.2 Optimization process SVR

To the optimization problem solution it is performed by the following steps:

1. Lagrangian Function: the Equation 4.12 must be minimized subject to the constraints. Thus, Lagrange multipliers $a_i \geq 0$, $\hat{a}_i \geq 0$, $\mu_i \geq 0$ and $\hat{\mu}_i \geq 0$ are introduced now taking into account the constrains to the minimization of the Equation 4.12 and then performing the optimization the Lagrangian using by Equation C.20.

$$\begin{aligned} L = \frac{1}{2}||\omega||^2 + C\sum_{i=1}^{N}(\xi_i + \hat{\xi}_i) - \sum_{i=1}^{N}(\mu_i\xi_i + \hat{\mu}_i\hat{\xi}_i) \\ -\sum_{i=1}^{N} a_i\left\{y_i + \epsilon + \xi_i - \hat{y}_i\right\} - \sum_{i=1}^{N}\hat{a}_i\left\{y_i + \epsilon + \hat{\xi}_i - \hat{y}_i\right\} \end{aligned} \tag{C.20}$$

2. Karush Kuhn Tucker (KKT) conditions: the constrained optimization is perform with KKT conditions, which are given by:

$$a_i(y_i - \hat{y}_i + \xi_i + \epsilon) = 0 \tag{C.21}$$

$$\hat{a}_i(y_i - \hat{y}_i + \hat{\xi}_i + \epsilon) = 0 \tag{C.22}$$

$$(C - \hat{a}_i)\hat{\xi}_i = 0 \tag{C.23}$$

$$(C - a_i)\xi_i = 0 \tag{C.24}$$

3. Now, the set of derivates of the Lagragian respect to $\boldsymbol{\omega}, b$ and ξ_i to zero giving.

$$\frac{\partial L}{\partial \boldsymbol{\omega}} = \omega - \sum_{i=1}^{N}(a_i - \hat{a}_i)\hat{y}_i\phi(x_i) = 0 \tag{C.25}$$

$$\frac{\partial L}{\partial b} = -\sum_{i=1}^{N}(a_i - \hat{a}_i) = 0 \tag{C.26}$$

$$\frac{\partial L}{\partial \xi_i} = C - \mu_i - a_i = 0 \tag{C.27}$$

$$\frac{\partial L}{\partial \hat{\xi}_i} = C - \hat{\mu}_i - \hat{a}_i = 0 \tag{C.28}$$

4. Relation between variables:

$$\omega = \sum_{i=1}^{N}(a_i - \hat{a}_i)\hat{y}_i\phi(x_i) \tag{C.29}$$

5. Replace in the Lagrangian function:

$$L(a, \hat{a}) = -\frac{1}{2}\sum_{i=1}^{N}\sum_{n=1}^{N}(a_i - \hat{a}_i)(a_n - \hat{a}_n)k(\boldsymbol{x_i}, \boldsymbol{x_n}) - \epsilon\sum_{i=1}^{N}(a_i - \hat{a}_i) + \sum_{i=1}^{N}(a_i - \hat{a}_i)\hat{y}_i \tag{C.30}$$

Where kernel function is defined by $k(x_i, x_n) = \phi(x_i)^T\phi(x_n)^T$.

6. Additional constraints: there are additional constraints respect to a_i and $\hat{a}_i$, as $\mu_i \geq 0$, $\hat{\mu}_i \geq 0$, $a_i \geq 0$ and $\hat{a}_i \geq 0$, therefore have to maximize with respect to a_i and $\hat{a}_i$ subject to:

$$0 \leq a_i \leq C \Rightarrow \text{Box Constraints} \tag{C.31}$$

$$0 \leq \hat{a}_i \leq C \Rightarrow \text{Box Constraints} \tag{C.32}$$

Now, replacing Equation C.29 into Equation 4.5, we can observe the predictions for new data using Equation C.33.

$$y(\boldsymbol{x}) = \sum_{i=1}^{N}(a_i - \hat{a}_i)\hat{y}_i\phi(x_i) + b \tag{C.33}$$

Then, when $a_i = \hat{a}_i = 0$, all points are inside the $\epsilon - \mathrm{tube}$, while the data points that contribute to predictions using Equation are called support vectors, where $a_i \not\equiv 0$ or $\hat{a}_i \not\equiv 0$ lying on the boundary of the tube or outside of it.

7. To find the independent term b: when $0 < a_i < C$ have $\xi_i = 0$ and the data points are support vectors, also satisfying $y_i + \epsilon - \hat{y}_i = 0$. Solving the Equation 4.5 is obtained b in Equation C.34, where $\boldsymbol{M}$ is the set of indices of data points with $0 < a_i < C$.

$$b = \hat{y}_i - \epsilon - \sum_{n=1}^{N}(a_i - \hat{a}_i)k(\boldsymbol{x_i}, \boldsymbol{x_n}) \tag{C.34}$$

Then, it can obtain and equivalent result with $0 < \hat{a}_i < C$, being better way, an average of the all b estimations. As was previously mentioned kernel function transform an original space into a grater dimensionality space, to model the problem in an easier way. For SVR, we used a Linear Kernel that it was defined in Equation C.18.

Bibliography

[1] O. Hornykiewicz. Biochemical aspects of Parkinson's disease. *Neurology*, 51(2 Suppl 2):S2–S9, 1998.

[2] C. G. Goetz, et al. Movement Disorder Society-sponsored revision of the Unified Parkinson's Disease Rating Scale (MDS-UPDRS): Scale presentation and clinimetric testing results. *Movement Disorders*, 23(15):2129–2170, 2008.

[3] V. E. Kelly, A. J. Eusterbrock, and A. Shumway-Cook. A review of dual-task walking deficits in people with parkinson's disease: motor and cognitive contributions, mechanisms, and clinical implications. *Parkinson's Disease*, 2012:1–14, 2012.

[4] P. M. Coen, S. A. Jubrias, G. Distefano, et al. Skeletal muscle mitochondrial energetics are associated with maximal aerobic capacity and walking speed in older adults. *Journals of Gerontology Series A: Biomedical Sciences and Medical Sciences*, 68(4):447–455, 2012.

[5] C. J. Hass et al. Quantitative normative gait data in a large cohort of ambulatory persons with parkinson's disease. *PloS one*, 7(8):e42337, 2012.

[6] Q. W. Oung, H. Muthusamy, et al. Technologies for assessment of motor disorders in Parkinson's disease: a review. *Sensors*, 15(9):21710–21745, 2015.

[7] J. Barth, J. Klucken, et al. Biometric and mobile gait analysis for early diagnosis and therapy monitoring in parkinson's disease. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 868–871, 2011.

[8] F. Parisi, G. Ferrari, et al. Body-sensor-network-based kinematic characterization and comparative outlook of UPDRS scoring in leg agility, sit-to-stand, and gait tasks in Parkinson's disease. *IEEE Journal of Biomedical and Health Informatics*, 19(6):1777–1793, 2015.

[9] J. Klucken, J. Barth, et al. Unbiased and mobile gait analysis detects motor impairment in Parkinson's disease. *PloS one*, 8(2):e56956, 2013.

[10] eGaIT - embedded Gait analysis using Intelligent Technology, http://www.egait.de/, 2018.

[11] S.V. Perumal and R. Sankar. Gait and tremor assessment for patients with parkinson's disease using wearable sensors. *ICT Express*, 2(4):168–174, 2016.

[12] Physionet-Gait Dynamics in Neuro-Degenerative Disease Data Base, https://physionet.org/physiobank/database/gaitndd/, 2018.

[13] Ö .F. Ertuğrul, Y. Kaya, R. Tekin, et al. Detection of parkinson's disease by shifted one dimensional local binary patterns from gait. *Expert Systems with Applications*, 56:156–163, 2016.

[14] P. Ren, W. Zhao, Z. Zhao, et al. Analysis of gait rhythm fluctuations for neurodegenerative diseases by phase synchronization and conditional entropy. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(2):291–299, 2016.

[15] J. C. Vasquez-Correa et al. Multi-view representation learning via gcca for multimodal analysis of parkinson's disease. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2966–2970, 2017.

[16] J. Hannink, H. Gaßner, et al. Inertial sensor-based estimation of peak accelerations during heel-strike and loading as markers of impaired gait patterns in PD patients. *Basal Ganglia*, 8:1, 2017.

[17] J. Camps, A. Sama, M. Martin, et al. Deep learning for freezing of gait detection in parkinson's disease patients in their homes using a waist-worn inertial measurement unit. *Knowledge-Based Systems*, 139:119–131, 2018.

[18] E. Sejdic et al. A comprehensive assessment of gait accelerometry signals in time, frequency and time-frequency domains. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(3):603–612, 2014.

[19] P. Prabhu, A. Karunakar, H. Anitha, et al. Classification of gait signals into different neurodegenerative diseases using statistical analysis and recurrence quantification analysis. *Pattern Recognition Letters (IN PRESS)*, 2018.

[20] J. R. Orozco-Arroyave et al. Analysis of speech from people with parkinson's disease through nonlinear dynamics. In *International Conference on Nonlinear Speech Processing*, pages 112–119. Springer, 2013.

[21] R. K. Begg, M. Palaniswami, and B. Owen. Support vector machines for automated gait classification. *IEEE Transactions on Biomedical Engineering*, 52(5):828–838, 2005.

[22] F. Takens. On the numerical determination of the dimension of an attractor. *Lecture notes in mathematics*, 1125(11):99–106, 1985.

[23] J. C. Vásquez-Correa, J. R. Orozco-Arroyave, J. D. Arias-Londoño, J. F. Vargas-Bonilla, and E. Noth. New computer aided device for real time analysis of speech of people with parkinson's disease. *Revista Facultad de Ingeniería Universidad de Antioquia*, (72):87–103, 2014.

[24] H.D. Abarbanel and M.B. Kennel. Local false nearest neighbors and dynamical dimensions from observed chaotic data. *Physical Review E*, 47(5):3057, 1993.

[25] H. Kantz and T. Schreiber. *Nonlinear time series analysis*, volume 7. Cambridge university press, 2004.

[26] H. Herzel. Bifurcations and chaos in voice signals. *Appl. Mech. Rev*, 46(7):399–413, 1993.

[27] P. Grassberger and I. Procaccia. Characterization of strange attractors. *Physical review letters*, 50(5):346, 1983.

[28] V. I. Oseledets. A multiplicative ergodic theorem. characteristic lyapunov, exponents of dynamical systems. *Trudy Moskovskogo Matematicheskogo Obshchestva*, 19:179–210, 1968.

[29] Glenn E. The chaos hypertextbook. `https://hypertextbook.com/chaos/lyapunov-1/`, 2016.

[30] R. Acharya, O. Faust, N. Kannathal, et al. Non-linear analysis of eeg signals at various sleep stages. *Computer methods and programs in biomedicine*, 80(1):37–45, 2005.

[31] N. Radhakrishnan and B.N. Gangadhar. Estimating regularity in epileptic seizure time-series data. *IEEE engineering in medicine and biology magazine*, 17(3):89–94, 1998.

[32] A. Lempel and J. Ziv. On the complexity of finite sequences. *IEEE Transactions on information theory*, 22(1):75–81, 1976.

[33] D. Abásolo, R. Hornero, C. Gómez, et al. Analysis of eeg background activity in alzheimer's disease patients with lempel–ziv complexity and central tendency measure. *Medical engineering & physics*, 28(4):315–322, 2006.

[34] J.A. Gómez-García. Biosignal characterization methodology by means of complexity analysis on non-uniform embedding space. Master's thesis, Universidad Nacional de Colombia-Sede Manizales, 2010.

[35] A. Widodo, M.C. Shim, W. Caesarendra, and B.S. Yang. Intelligent prognostics for battery health monitoring based on sample entropy. *Expert Systems with Applications*, 38(9):11763–11769, 2011.

[36] J.S. Richman and J.R. Moorman. Physiological time-series analysis using approximate entropy and sample entropy. *American Journal of Physiology-Heart and Circulatory Physiology*, 278(6):H2039–H2049, 2000.

[37] L.S. Xu, K.Q. Wang, and L. Wang. Gaussian kernel approximate entropy algorithm for analyzing irregularity of time-series. In *Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on*, volume 9, pages 5605–5608. IEEE, 2005.

[38] G. M. Mindlin and R. Gilmore. Topological analysis and synthesis of chaotic time series. *Physica D: Nonlinear Phenomena*, 58(1-4):229–242, 1992.

[39] I.M. Moroz. *Biomechanically informed nonlinear speech signal processing*. PhD thesis, University of Oxford, 2007.

[40] T Bayes. An essay towards solving a problem in the doctrine of chances. 1763. *MD computing: computers in medical practice*, 8(3):157, 1991.

[41] C.M. Bishop. *Pattern recognition and machine learning*. springer, 2006.

[42] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[43] L. Breiman. *Classification and regression trees*. Routledge, 2017.

[44] M. Moullin. Delivering excellence in health and social care: Quality, excellence and performance measurement. In *Delivering excellence in health and social care: quality, excellence and performance measurement*. Palgrave Macmillan, 2002.

[45] D.M. Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. 2011.

[46] M. L. McHugh. Interrater reliability: the kappa statistic. *Biochemia medica: Biochemia medica*, 22(3):276–282, 2012.

[47] H. Sadeghi, P. Allard, F. Prince, et al. Symmetry and limb dominance in able-bodied gait: a review. *Gait & posture*, 12(1):34–45, 2000.

[48] G. Ebersbach et al. Clinical syndromes: Parkinsonian gait. *Movement Disorders*, 28(11):1552–1559, 2013.

[49] H. Negahban, M. Salavati, M. Mazaheri, et al. Non-linear dynamical features of center of pressure extracted by recurrence quantification analysis in people with unilateral anterior cruciate ligament injury. *Gait & posture*, 31(4):450–455, 2010.

[50] A.J. Smola and B. Schölkopf. *Learning with kernels*, volume 4. Citeseer, 1998.