NORWEGIAN UNIVERSITY OF SCIENCE AND TECHNOLOGY

TTK4550 - SPECIALIZATION PROJECT

Classification and Interpretation of Cerebral Palsy-related movements by means of Multivariate Analysis

---

Author
Marie Pauline KRISTIANSEN

Supervisor
Frank WESTAD

December 2018

Preface

This report is written as a part of the course 'TTK4550 - Engineering Cybernetics, Specialization Project' at the Norwegian University of Science and Technology (NTNU). In this project I work with a research group at St. Olav's Hospital in Trondheim, Norway, which researches early prediction of Cerebral Palsy in preterm children. Parallel with my work, students from the Department of Computer Science at NTNU are developing a model based on Neural Networks to solve the same task; early prediction of CP using computer based analysis.

I want to thank my supervisor, Frank Westad, for all the help during this fall and all his good ideas regarding which methods to look into. I also want to thank Lars Adde for introducing me to this exciting project and accepting me into the research group, and Espen Alexander F. Ihlen and Daniel Groos at St. Olav's Hospital for answering all my questions regarding the project and helping me when I feel lost in my work.

Summary

In this report Multivariate Analysis methods are explored and evaluated for use on motion data sets from preterm infants. The goal is to use these methods to classify which infants have Cerebral Palsy through locating fidgety movements in the time series. A subset of the motion data is analyzed and evaluated through both time and frequency domain. Using Fourier Transform on the motion time series gives a $93,1\%$ accuracy when applying Quadratic LDA on PCA scores. Other features are extracted from the time series using the HCTSA-framework [14] and Wavelet Transform, step detection and autocorrelation are found to be interesting features that can be used in further research.

Table of Contents

List of Tables

List of Figures

Abbreviations

CP = Cerebral Palsy
GMA = General Movement Assessment
PCA = Principal Component Analysis
PCs = Principal Components
FFT = Fast Fourier Transform
PSE = Power Spectral Entropy
LDA = Linear Discriminant Analysis
PLS-DA = Partial Least Squares Discriminant Analysis
SVM = Support Vector Machine
EDA = Exploratory Data Analysis
SVD = Singular Value Decomposition
HCTSA = Highly Comparative Time-Series Analysis

Chapter 1

Introduction

1.1 Background and motivation

1.1.1 Cerebral Palsy

Cerebral Palsy (CP) is a syndrome of motor impairment that results from a lesion occurring in the developing brain. The syndrome affects everyone differently and several classification systems exists to assess the type and form of CP that an individual has. These classifications are defined according to the anatomical site of the brain lesion, clinical symptoms and signs, topographical involvement of extremities and classification of degree of muscle tone [2].

With advances in neonatal intensive care, the survival of very preterm (born 32 weeks of gestation) and very low-birth-weight (VLBW) (weighing 1500 g) children has improved considerably [23]. Cerebral Palsy has a prevalence of 2.0-3.5 per 1000 live-births [37], but multiple studies demonstrate an increasing prevalence of CP with decreasing birthweight and gestational age. In a report from Sweden, the prevalence of cerebral palsy was 6.7 per 1000 live births for children born at 32 to 36 weeks of gestation, 40.4 per 1000 live births for children born at 28 to 31 weeks of gestation, and as high as 76.6 per 1000 live births for children born before 28 weeks of gestation [20]. There was a similar increase when looking at birth weight.

Because of the different classifications of Cerebral Palsy, prediction at an early age is a challenge. About 85% of children with CP show an abnormal MRI scan, which can provide an estimate of the timing of the lesion and whether it causes a motoric impairment. However, MRI scans are not optimal for children. It requires them to lie still for 30 to 60 minutes, and the hollow tube is easily considered scary.

In a clinical report from 2013 [16], the American Academy of Pediatrics writes about the importance of early diagnosis of Cerebral Palsy. In the report the Academy stresses the importance of early diagnosis as a way to receive interventions that will help the child master everyday tasks, increase mobility and improve their quality of life. Early diagnosis can also address the ongoing anxiety parents have about their childs health condition [7].

1.1.2 General Movement Assessment and Fidgety Movements

After birth, infants have a spontaneous movement pattern with a writhing character. At the age of 6 to 9 weeks post term the form and character of the general movements change from the writhing type into a fidgety pattern. These fidgety movements are defined as an ongoing stream of small, circular and elegant movements of neck, trunk and limbs. Fidgety movements in a healthy infant is a transient phenomenon; they emerge gradually at 6 weeks, come to full expression between 9 and 13 weeks post term and taper off again between the age of 14 to 20 weeks post term [17].

In 1997 Prechtl et al. presented a tool to predict motoric dysfunction in infants based on their movement pattern [18]. The tool, known as General Movement Assessment (GMA), uses fidgety movements as a marker for a normal neurological outcome. In [18], the 60 infants with abnormal and absent fidgety movements included 57 infants with an abnormal outcome. 49 of the ones with abnormal FM had cerebral palsy and eight had developmental retardation or minor neurological signs. Only three were diagnosed as normal at age 2 years [18].

GMA provides a method to predict CP at an earlier age. GMA is also non-invasive, cost-efficient and easy to learn for physicians [9]. A disadvantage is the subjectivity of the physicians, but either way Prechtl et al. found this method to have a higher specificity and sensitivity than ultrasound, where their method had a specificity of 96% and sensitivity 95% and ultrasound only 83% and 80%.

1.1.3 Related work

The study of fidgety movements in relation with CP is done is several studies [9] [34] [18] [17] [1]. In some, physicians analyze the movements visually [9] [34] and in others they use sensors like pressure sensitive mats, Kinect cameras and motion sensors [27]. In 2010, Adde et. al. [1] did a feasibility study on computer-based video analysis of general movements and found it to be an objective and feasible tool for early prediction of CP in high-risk infants.

The study of movements through data analysis is an active field of research. In [3], Principal Components Analysis (PCA) and Probabilistic PCA are used for segmenting motion capture data, only unordered sets of poses are analyzed and no information about temporal dynamics is taken into account. In [22] Hidden Markov Model's are applied to discover groupings of similar objects motions observed in a video collection. [6] analyzes movements and classifies them through Support Vector Machines. [24] estimates person-independent head poses using Random Forest regression.

1.2 Goal and hypothesis

The main goal for this project is to develop a model that can evaluate and separate movements from babies with CP from those without. This model will be trained on motion data from a video database recorded as part of a large research project driven by Lars Adde and his research group at St. Olav's Hospital in Trondheim, Norway.

The hypothesis is that fidgety movements is recognizable as repeating movements in the healthy babies. In this project, frequency of the motion data will be investigated with the belief that the babies may do the fidgety movements at a certain frequency or frequency range. Other features that finds repeating patterns in time series will also be investigated closely. The field of computer learning algorithms has gained large success over the previous years, and the hypothesis is that a computer based model can obtain an accuracy that is just as good, or better, as gestalt perception used by the physicians.

1.3 Outline of this work

This project and in the master thesis to come will analyze motion data from babies and use multivariate methods to classify whether the movements indicate CP or not. The input will be raw coordinate time series and suitable transforms thereof, and by means of feature extraction, dimension reduction and classification I will investigate ways to get the classification accuracy as high as possible and at the same time provide interpretable results.

The report is structured as follows: Section 1 presents the background for the project and motivation for the task, Section 2 looks into theory and methods for pre-processing of the data and data analysis methods, Section 4 summarizes the results that are gained so far and Section 5 briefly discusses the work that is going to be done in the master thesis during spring 2019.

Chapter 2

Theory and methods

Data mining is the process of discovering patterns in large data sets [11]. The full process from raw data to classification can be divided into four parts, which are shown in Figure 2.1. This analysis will go through the full process and find the best methods and combinations of them.

2.1 Pre-processing

In computer science there is a saying: "garbage in, garbage out" (GIGO) [15], which describes the concept that bad input data can't give good results, and instead produces nonsense or "garbage". In order to avoid this it is needed to do some pre-processing of the data before running the learning algorithms to produce good models.

One aspect of pre-processing is data cleansing, which is about detecting and correcting corrupt or inaccurate parts of the data. This report will look into outlier detection and removal through use of a Hampel filter.

Other aspects are centering and scaling. Centering of the data is done to remove the constant levels[4], while scaling may be needed to account for differences in the measurements that are known prior to the analysis.

2.1.1 Hampel filter

Hampel Filters is in the decision-based filter class, and is closely related to a standard median filter as it uses the local median and median absolute deviation (MAD) to detect outliers.

Given a sequence of data points, $x_1, x_2, \ldots, x_n$, and a sliding-window of length k, the Hampel filter calculates the local median m_i and standard deviation σ_i for each window.

$$m_i = median(x_{i-k}, x_{i-k+1}, \ldots, x_{i+k-1}, x_{i+k}) \tag{2.1}$$

$$\sigma_i = \kappa \text{median}(|x_{i-k} - m_i|, |x_{i-k+1} - m_i|, \ldots, |x_{i+k-1} - m_i|, |x_{i+k} - m_i|) \tag{2.2}$$

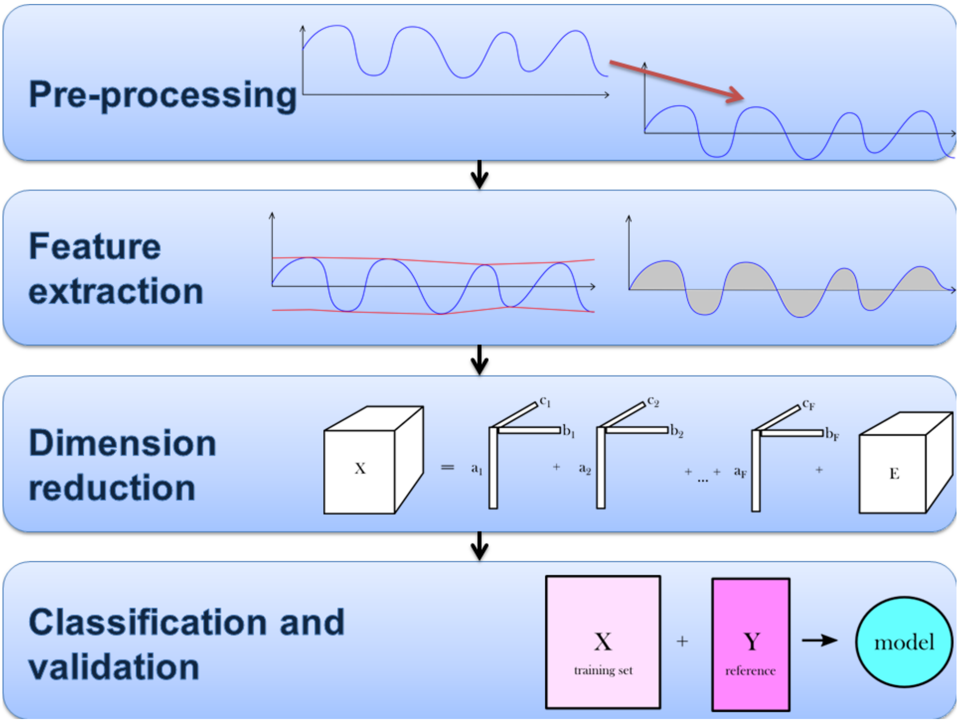

Figure 2.1: Overview of full classification system.

where $\kappa = \frac{1}{\sqrt{2}\text{erfc}^{1}\frac{1}{2}} \approx 1.4826$

A sample x_i is declared an outlier if it is such that

$$|x_i - m_i| > n_\sigma \sigma_i \tag{2.3}$$

for a given threshold n_σ. If the point is marked as an outlier it is replaced with m_i.

The Hampel filter has the advantage that it doesn't introduce distortion into the signal. A standard median filter replaces every point with the median of the window which can lead to information getting lost, which makes the Hampel Filter better.

2.1.2 Centering

Given a vector, X, the centered vector X_c is obtained by calculating the mean of the series and substracting this from every entry in the original vector.

$$\boldsymbol{X_c} = \boldsymbol{X} - \text{mean}(\boldsymbol{X}) \tag{2.4}$$

2.1.3 Scaling and normalizing

Scaling of data sets can be done in various ways. A common way is to standardize the data set by making them unit-variant:

$$\boldsymbol{X}_{sc} = \frac{\boldsymbol{X} - \text{mean}(\boldsymbol{X})}{\text{standard deviation}(\boldsymbol{X})} = \frac{\boldsymbol{X}_c}{\sigma} \tag{2.5}$$

Another method is the min-max-normalization (or min-max scaling) which scales all features to be within a given range:

$$\boldsymbol{X}_{sc} = \frac{\boldsymbol{X} - \min(\boldsymbol{X})}{\max(\boldsymbol{X}) - \min(\boldsymbol{X})} \tag{2.6}$$

2.2 Feature Extraction and Feature Selection

In order to build a good predictor, good features are needed. Features are measurable properties or characteristics of the phenomenon being observed, and usually the term ”features” for variables constructed from the input variables is used while the raw input variables are called ”variables” [21]. There are thousands and thousands of features that can be calculated from a time series data set [14], and in order to build a computationally effective model one should select the best subset of them to use in developing a model. There are many benefits of feature selection: simplification of the model to make it easier to understand and interpret, reducing the measurement and storage requirements, reducing training and utilization times, defying the curse of dimensionality to improve prediction performance and enhanced generalization as it reduces overfitting [12] [21].

2.2.1 Feature extraction

There are several ways to extract features from the variables [21], including: clustering, basic linear transformation of the input variables (e.g. PCA/SVD, LDA), more sophisticated linear transforms like spectral transforms (e.g. Fourier), wavelet transforms or convolutions of kernels.

Fourier Transform

To gain insight in the frequencies present in the signal one can perform Fourier Analysis to do a transformation from time domain to the frequency domain. The definition of the Discrete Fourier Transform for a given signal X is given as

$$\hat{X}_{FT}(k) = \sum_{j=0}^{N-1} X(j) W_n^{jk} \tag{2.7}$$

where $W_n = \exp\left(-2\pi i\right)/N$.

Wavelet transform

The Wavelet Transform is in principle the same as a Fourier Transform, but instead of decomposing the signal into infinite waves (sinusoids) the signal is decomposed using wavelets using a wavelet analyzing function $\psi_{jk}^{*}(t)$.

The signal is divided into shorter segments and a Wavelet transform is calculated on each segment. Each segment varies in size, which gives a flexible resolution in both time- and frequency-domains. In contrast with sinusoids, wavelets are localized in both the time and frequency domains, so wavelet signal processing is more suitable for nonstationary signals than the Fourier Transform.

When using Wavelet transform for decomposing of a signal an iterative procedure is followed. Starting with the signal s one computes two sets of coefficients at each iteration: the approximation coefficients and detail coefficients.

First, s is sent in parallel through a low-pass filter F and a high-pass filter G, both followed by downsampling. This produces the approximation coefficient cA_1 and detail coefficient cD_1. See Figure 2.2. The approximation coefficient cA_1 is then sent through the filters again to produce the next set of coefficients. This proceeds until the given number of levels is reached.

EMD

The Empirical Mode Decomposition (EMD) is a method to decompose a signal into a finite set of components which can be described as intrinsic mode functions (IMFs) [19]. Finding the IMFs is done by a process called sifting, which iteratively improves the estimate of the IMF. The iterative sifting process is repeated until the IF fulfills the following criteria[8]:

1. The number of extrema and number of zero crossings must either be equal or siffer by at most one.

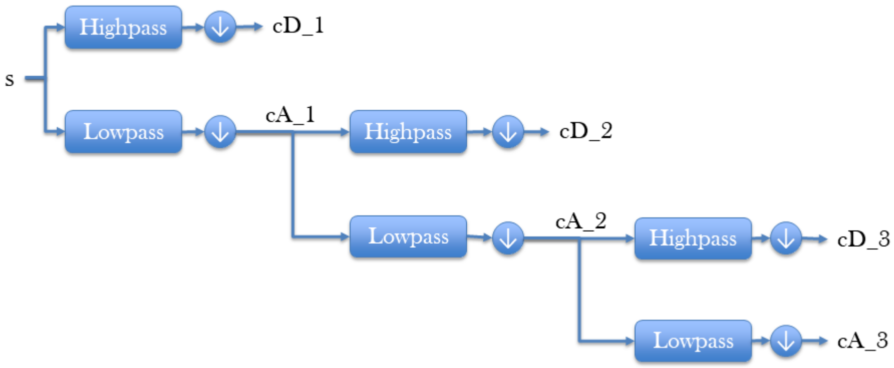

Figure 2.2: Discrete Wavelet Analysis through filter banks.

2. The mean value of the upper and lower envelope is zero at any point.

The method is comparable to a Fourier Transform as it breaks the signal down into its components, but since the decomposition is based on the local characteristic time scale of the data, it can be applied to nonlinear and nonstationary processes[19].

Cross-correlation

Cross-correlation is a measurement of similarity between two signals.

Cross-correlation for discrete signals is defined as

$$(f \star g)[n] \overset{\text{def}}{=} \sum_{m=-\infty}^{\infty} f^*[m]\, g[m+n] \tag{2.8}$$

where f^* denotes the complex conjugate of f. The cross-correlation of the two signals, $f(t)$ and $g(t)$, is equivalent to the convolution between $f(-t)$ and $g(t)$.

One property of cross-correlation that is often used in implementations is that

$$\mathcal{F}\{f \star g\} = \mathcal{F}\{f\}^* \cdot \mathcal{F}\{g\} \tag{2.9}$$

where $\mathcal{F}$ denotes the Fourier transform, and an asterisk indicates the complex conjugate. Using fast Fourier Transform algorithms for computing cross-correlation can give more efficient computations [10].

Autocorrelation

Autocorrelation is cross-correlation between a signal y_t and a lag of the same signal, y_{t+k}, where $k = 0, \ldots, K$.

Algorithms for computing autocorrelation is usually implemented using Fourier Transform. Fourier Transform is used to compute the autocorrelation function in the frequency domain, then converts back to time domain using an inverse Fourier Transform.

2.2.2 Feature selection

[21] stated that there are mainly three different ways of doing feature selection: wrappers, filters and embedded methods. Wrappers utilize the learning algorithm of choice as a black box and scores subsets of features according to their predictive power. Filters select subsets as a pre-processing step, independently of the learning algorithm. Embedded methods perform feature selection embedded in the training process and are usually specific to given learning algorithms.

2.2.3 HCTSA

In 2017, Ben D. Fulcher and Nick S. Jones presented *HCTSA* (Highly Comparative Time-Series Analysis): a computational framework for feature extraction from time-series. The comparative feature-based approach to time-series classification was first introduced in [13] and then the computational framework was presented in [14]. *HCTSA* includes an architecture for computing over 7700 time-series features and a suite of analysis and visualization algorithms for use in selecting useful and interpretable features for a given application. It includes classification algorithms and computes which features are most important in this classification.

HCTSA is MATLAB-based, and in this analysis it's used as a way of extracting multiple features in a tidy manner and then analyze which features are worth looking into. The features that shows most promise will then be analyzed further with PCA and PARAFAC.

2.3 Exploratory Data Analysis

From an early age people are told that the easiest way to investigate a problem is to do it piece by piece. In mathematics this is done by changing one variable at a time to see how the system reacts, only this is too simple for most complex real life systems. With a large number of variables with unknown, complex relationships Multivariate Analysis is needed. Multivariate Analysis is a way of investigation a large number of variables simultaneously to understand the relationship that may exist between them [33]. For small data sets with few variables it may be enough to present the data as disjointed graphs, but for big data sets this will be too complex and it will be very hard to find the dependencies manually.

Exploratory Data Analysis (EDA), or *data mining*, attempts to find the hidden structure in large, complex data sets. EDA finds the structure that results from the influence of all variables acting simultaneously, not just the influence of one variable. The two main methods used in EDA are cluster analysis and Principal Component Analysis.

2.3.1 Principal Component Analysis

Principal Component Analysis (PCA) is a method that analyzes the variability in a data set. It's a mathematical procedure that transforms a number of variables into a smaller number of orthogonal variables called *principal components* (PCs). PCA transforms the data using an orthogonal linear transformation onto a new coordinate system, where the coordinates

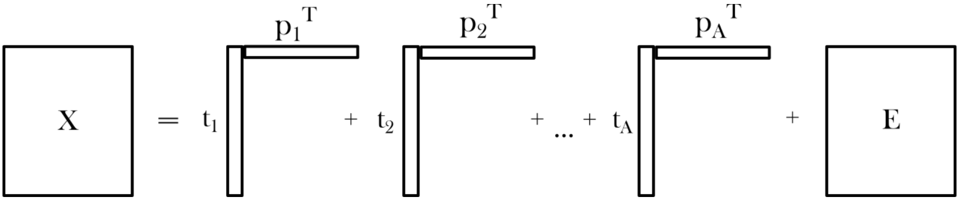

Figure 2.3: PCA decomposition

are the principal components. The PCs are extracted so that the first PC explains a larger part of the total variance than the next PC, and so on. This can be visualized in the terms of the eigenvalues, often in a cumulative way.

Given a zero-mean data matrix $\boldsymbol{X}$, with n rows containing data from a new repetition of the experiment and p columns each gives a particular feature, the PCA splits $\boldsymbol{X}$ into a structure part $\boldsymbol{M}$ and an error part $\boldsymbol{E}$.

$$\boldsymbol{X} = Structure + Noise = \boldsymbol{M} + \boldsymbol{E} \tag{2.10}$$

The structure matrix $\boldsymbol{M}$ may be regarded as a sum of contributions from different functions of the rows and columns

$$\boldsymbol{M} = f(columns) \cdot g(rows) \tag{2.11}$$

where each function can be approximated by a linear model, which together forms

$$\boldsymbol{M} = \boldsymbol{T}\boldsymbol{P}^T \tag{2.12}$$

The matrix $\boldsymbol{T}$ contains the *scores* and the matrix $\boldsymbol{P}^T$ contains the *loadings*. The decomposition is shown in Figure 2.3. The scores and loadings can be estimated in different ways, e.g. through Singular Value Decomposition (SVD) or the NIPALS algorithm [28].

A non-trivial task when using PCA is choosing the number of dimensionality, aka the number of principal components A. A model with a high percentage of explained variance is wanted, but one does not want to include the noise in the scores and loadings. The validation methods of Section 2.5 will be used to find A.

Singular Value Decomposition

Given the matrix $\boldsymbol{X}$ of size $n \times p$, the singular value decomposition of $\boldsymbol{X}$ is

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T \tag{2.13}$$

where $\boldsymbol{U}$ is a $n \times n$ unitary matrix, $\boldsymbol{S}$ is a diagonal $n \times p$ matrix with non-negative real numbers on the diagonal, and $\boldsymbol{V}$ is a $p \times p$ unitary matrix. In PCA, the matrix $\boldsymbol{V}$ defines the loadings,

$$\boldsymbol{V} = \boldsymbol{P} \tag{2.14}$$

and $\boldsymbol{U}$ and $\boldsymbol{S}$ together form the scores

$$\boldsymbol{T} = \boldsymbol{U}\boldsymbol{S} \tag{2.15}$$

.

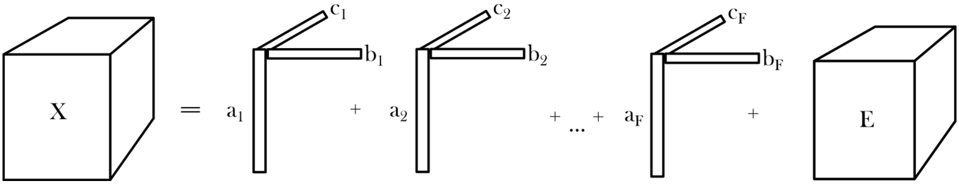

Figure 2.4: PARAFAC decomposition

2.3.2 PARAFAC

Parallel Factor Analysis (PARAFAC) is a generalization of PCA to higher order arrays [4]. An example of another method of higher order PCA is the Tucker3 decomposition. A decomposition of the data is made into trilinear components, as shown for three-way data in Figure 2.4. Instead of one score vector and one loading vector, as in PCA, each component consists of one score vector and two loading vectors. A PARAFAC model of a three-way array is given by the matrices $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ with elements a_{if}, b_{jf} and c_{kf} such that:

$$\boldsymbol{M} = \sum_{f=1}^{F} \boldsymbol{a}_f \otimes \boldsymbol{b}_f \otimes \boldsymbol{c}_f \tag{2.16}$$

where $\otimes$ is the Kronecker product. The Kronecker product, also called the matrix direct product, of a $k \times \ell$ matrix $\boldsymbol{A}$ and a $m \times n$ matrix $\boldsymbol{B}$ is the $km \times \ell n$ matrix:

$$\boldsymbol{A} \otimes \boldsymbol{B} := \begin{pmatrix} a_{1,1}\boldsymbol{B} & a_{1,2}\boldsymbol{B} & \cdots & a_{1,\ell}\boldsymbol{B} \\ a_{2,1}\boldsymbol{B} & a_{2,2}\boldsymbol{B} & \cdots & a_{2,\ell}\boldsymbol{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k,1}\boldsymbol{B} & a_{k,2}\boldsymbol{B} & \cdots & a_{k,\ell}\boldsymbol{B} \end{pmatrix} \tag{2.17}$$

where $a_{h,j}$ denotes the element of $\boldsymbol{A}$ in row h and column j.

2.4 Classification

The problem of classification is to identify which category or *class* a new observation belongs to [31].

There are numerous different classification algorithms. Some examples are: Linear Discriminant Analysis, PLS-DA, Support Vector Machines, Random Forests, Neural Networks and Cluster analysis (e.g. K-means).

PLS-DA Partial Least Squares Discriminant Analysis (PLS-DA) is a variant of PLS regression to use when Y is categorical, i.e. represents a class. PLS-DA is performed in order to sharpen the separation between groups of observations, by hopefully rotating the PCA components such that a maximum separation among classes is obtained, and to understand which variables carry the class separating information [5].

SVM Support Vector Machines (SVM) separate a set of binary-labeled training data by means of a hyperplane that is maximally distant from the two classes. The function that maps the variables onto the new space is called a kernel function, and there is no theoretical tool to find the best one. There are several options, such as a linear function, polynomial function or the Radial Basis function [36]. In addition to being dependent on the choice of kernel function, SVMs are in risk of overfitting.

SVMs are widely used in classification. [6] uses SVMs to classify motion from a set of filtered images, [29] uses SVM for nonlinear prediction of chaotic time series, [30] uses multi-class SVM for recognition of abnormal human activity and [26] uses a SVM-based computer-aided diagnosis system for early detection of Alzheimer's disease.

Random Forests Random Forests (RF) are built from individual decision trees, where a random subset of variables is selected for each tree. Random Forest is unexcelled in accuracy among current algorithms, and runs efficiently on large data bases [36]. An advantage for the classification problem for this method is that it gives an estimate and visuliazation of which variables that are important.

Random Forests has become a popular technique for both classification, prediction, studying variable importance, variable selection and outlier detection. Examples of studies where RF have been applied and compared are explored as a survey in [35].

2.4.1 Score vectors from PCA as input to classification methods

In traditional classification problems one extract and selects features from a data set and applies the classification algorithm directly on these features. The feature selection step is done in various ways (see Section 2.2), but this report will focus on using EDA-methods to reduce the feature space. Both PCA and PARAFAC can reduce numerous features or variables down to the most important ones and can be used to make the classification step more efficient.

The way to do this is to use the score-vector from PCA or PARAFAC as input to the SVM, as opposed to the features or variables directly. This leads to a smaller feature set, and the SVM will be less in risk of overfitting and also the noise part of the input matrix $\boldsymbol{X}$ is removed. In this context it is not so critical if one or two components ”too many” are used as input to SVM. Note that validation of such a procedure must be done by dividing the samples into calibration and test set before PCA is computed. The test set samples are then projected onto the PCA based on the calibration set prior to classification with SVM or another method of choice.

In [26] they have used kernel PCA and LDA as dimension reduction and feature extraction before training a SVM. This lead to a simpler classification line and a lower number of support vectors to be evaluated. They also reduced the number of features necessary to describe the data distributions sufficiently for a SVM classifier down to 4, solving the small sample size problem of multivariate analysis.

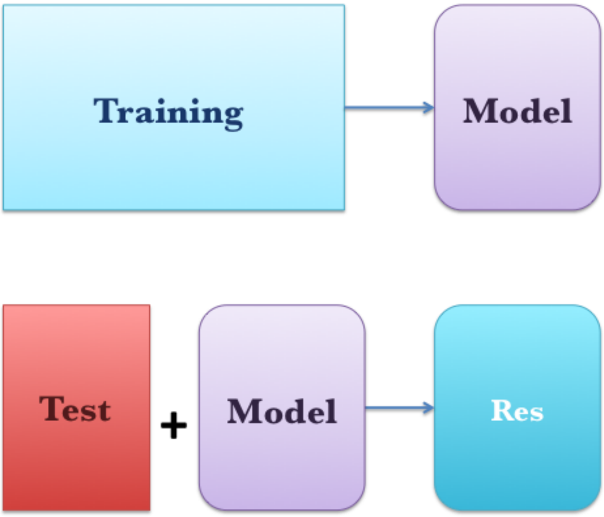

Figure 2.5: Test set validation.

2.5 Validation

In order to validate that a classification model is useful for new data and evaluate how well it performs one needs validation methods. Conceptually it is distinguished between *external* and *internal* validation; external validation concerns whether it is used correct information in making of the model and validating that different models give the same results, while internal validation is based on numerical validation. Some internal validation methods are cross validation, test set validation and cross model validation.

2.5.1 Test set validation

For test set validation the full data set is split into two groups: test set and training set. The model is built with the training samples and then the prediction error is computed by predicting the outcome for the test samples.

A challenge with test set validation is how to split the data set. Splitting at random is not sufficient, as it will be subgroups in the data due to underlying sources of variation. In this context this could e.g. be ethnicity of the infants or age group. An algorithm developed to split the data is the Kennard-Stone sampling algorithm (KS). It selects a subset of samples that has an uniform distribution over the predictor space, and hence tries to avoid the problem of subgroups in the two sample sets.

Kennard-Stone algorithm

KS iteratively selects the pair of points in the set that are farthest apart until the required number of samples have been chosen. For each step it computes the distance d between

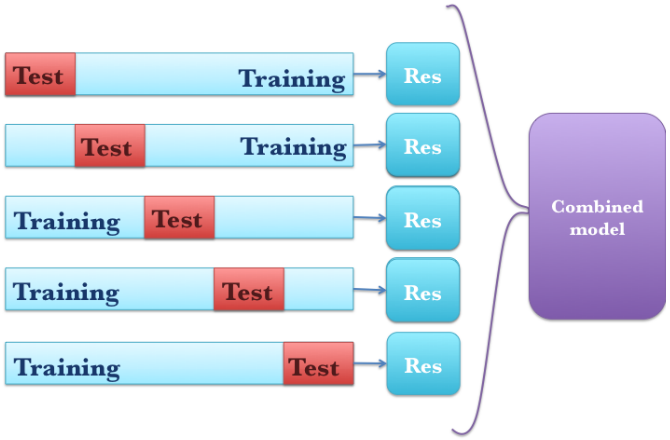

Figure 2.6: Cross-validation.

each unassigned points i_0 and selected points i and finds the point i_0 for which:

$$d_{selected} = \max_{i_0}(\min_{i}(d(i_0, i)) \tag{2.18}$$

This selects point i_0 which is the farthest apart from its closest neighbors i in the calibration set. Measuring the distance can be done in several ways, but the distance measurement used in the implementation in [32] is Euclidean distance:

$$d(i_0, i) = \sqrt{(i_0 - i)^2} \tag{2.19}$$

2.5.2 Cross validation

In cross validation it is iteratively chosen a new subset of samples to be training and test set.

The procedure is as follow: pick out k samples from the calibration set and build a model with the remaining samples. Predict the outcome on the k left-out samples and calculate the residual. Put these samples back into the calibration set, and repeat the procedure until all samples have been left out at least once. Combine the prediction residuals for all iterations.

In cross validation, all samples are used in both training and testing. This avoids the problem of subgroups in the sample set. It is also applicable to smaller data sets that doesnt have sufficient number of samples to take out an independent test set.

Chapter 3

Problem definition and dataset

3.1 Problem definition

The goal for this project is to use multivariate methods to analyze motion data and separate healthy subjects from those with Cerebral Palsy. The goal is to find the fidgety movements using computer based analysis and use those movements to separate.

The movements are not easy to recognize, and several steps will be taken towards finding them. Feature extraction will be done by computing several numerical features for each of the timeseries. Then a dimension reduction will be done on the features to find the most significant ones. The classification algorithm will then take the reduced feature space as input and separate the two classes.

To be able to use this model for CP-prediction, the results must be interpretable. The model will be evaluated based on its classification accuracy, but an important step will also be to look into what numerical features separates the groups the most. Feature extraction will also focus on finding features that are physically interpretable. An example of a physical interpretation of a numerical feature is that correlation between to timeseries will tell us how two body parts move together (or independently).

The model must also be computationally efficient. The training process will evaluate a big set of features and therefore needs a less stricter time limit, but the finished model must be computationally inexpensive to use for classification of new subjects.

The model should also be as easy as possible. A good rule of thumb is that if two algorithms gives the same result, choose the one that is less complex.

3.2 The data set

The data in this project comes from a database of 900 standardized video recordings at St. Olav's University Hospital of infants at risk of neurological dysfunctions from Norway, USA and India. The video's are analyzed using a tracker algorithm developed in a master thesis in 2018 done for the InMotion project. The output of this tracker algorithm is

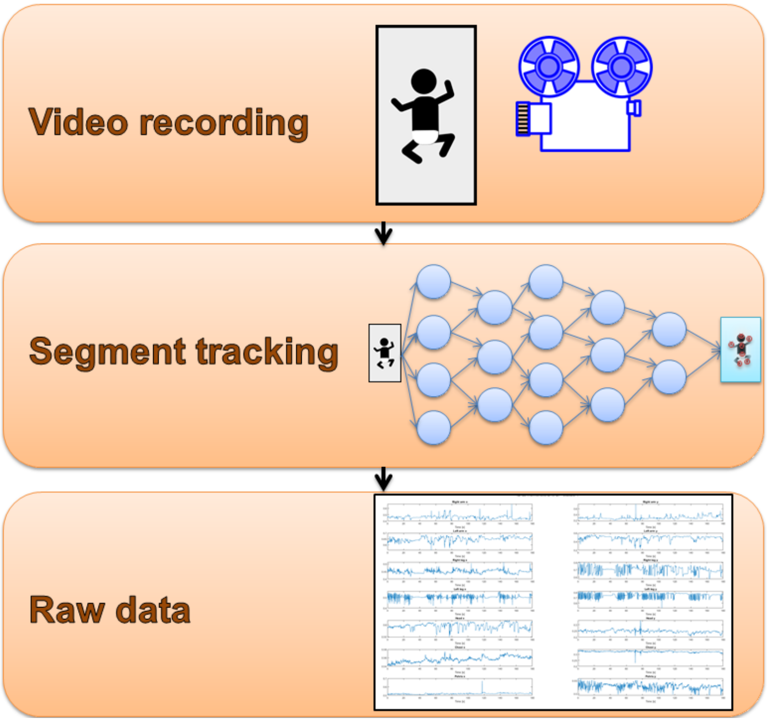

Figure 3.1: Overview of collection of raw data.

timeseries for the x- and y-coordinates for 7 segments on the infants in the videos. This tracking is done on 378 of the videos which gives 378 × 14 timeseries. These coordinate timeseries are the input for the model developed through this project. In the future it is hoped to expand this set by using the tracking algorithm on more of the videos, but this is still a work in progress. An overview of this data collecting system is shown in Figure 3.1.

The coordinates are normalized so that the coordinate system is the same for every frame. This is shown in Figure 3.2. Here it is shown that origo is set it the top left corner, and max value for both x and y is 1. Normalizing the coordinates makes movement in the x-direction seem bigger when looking at plots of the time-series. The amplitude change is bigger, but this is simply due to the fact that the video's are mainly filmed with a $9:16$ ratio. A graphic representation of the data structure is shown in Figure 3.3, where, for the time being, $I = 378$, $K = 14$ and J varies.

Plot of the data from two subjects is shown in Figure 3.4 and 3.5. Every subject has 14 time-series of the same length, but the length of the time-series vary with the subjects. This is because the video's are of varying length.

3.2.1 Errors from tracking

The time-series for the coordinates are made through analyzing video's frame by frame. There is no comparison of the tracking between frames, so if a segment is labeled wrong

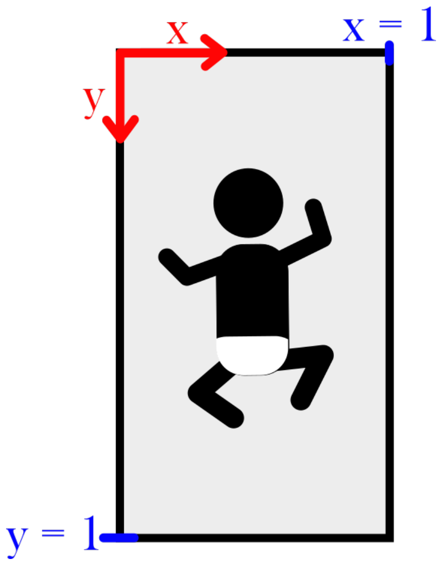

Figure 3.2: Coordinate system on video frames.

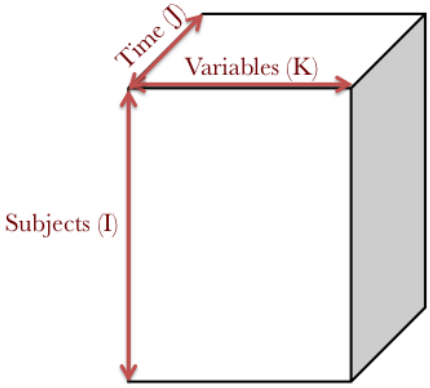

Figure 3.3: Data structure.

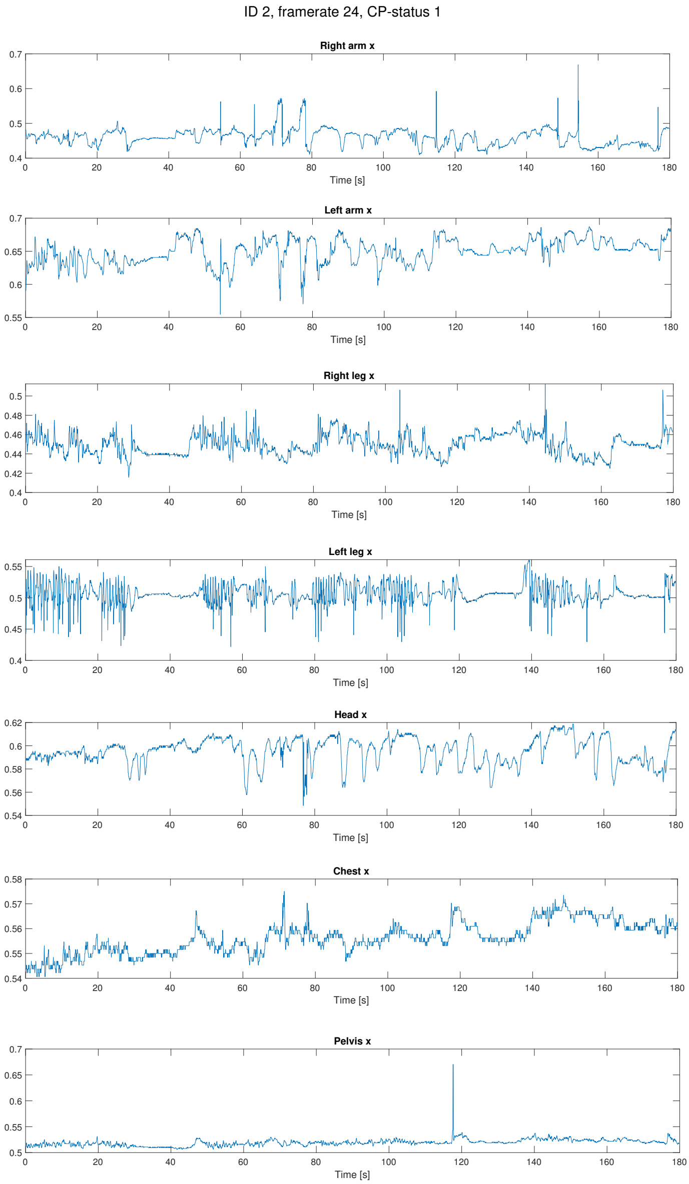

Figure 3.4: Raw data from a subject with CP.

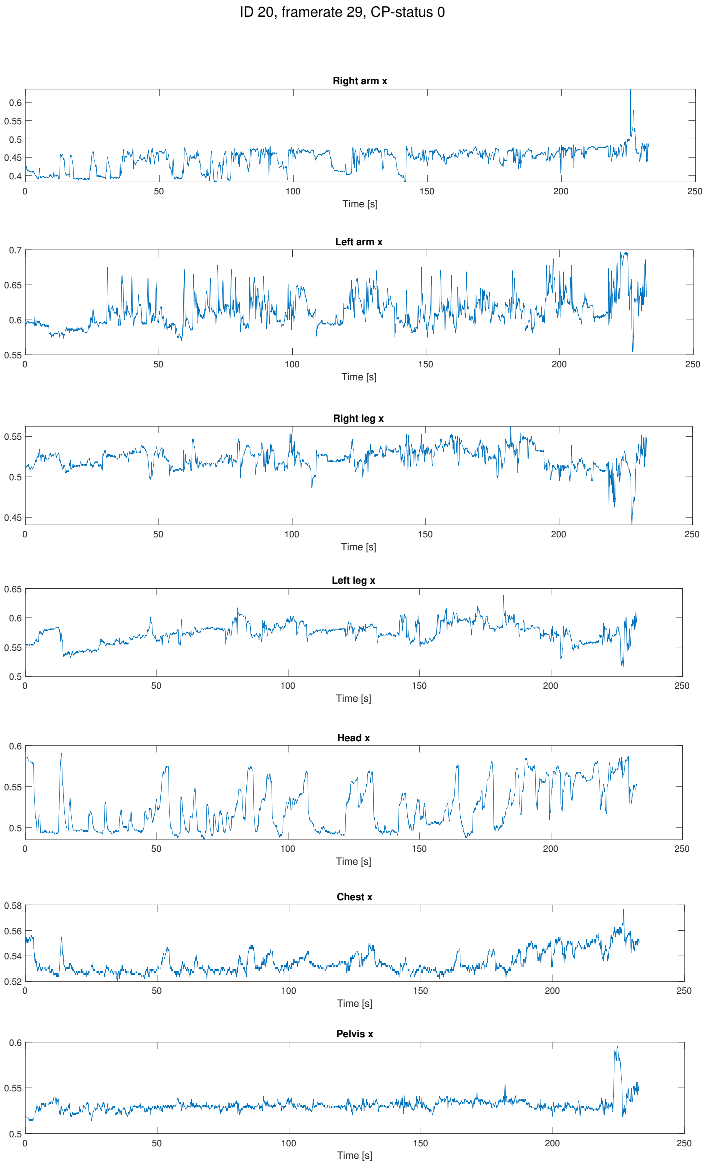

Figure 3.5: Raw data from a healthy subject.

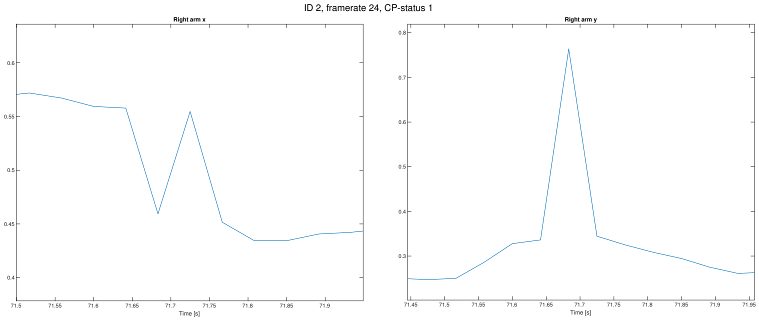

Figure 3.6: Typical tracking error in original data.

the model does not correct it. The score for each labeling is not part of the dataset, but some of the errors are still quite visible. One example is shown in Figure 3.6. Here it is shown that the y-coordinate jumps from ~ 0.3 to ~ 0.7 for the right arm from one frame to the next. One frame corresponds to $\sim$ 90cm in real life, which means that a change in amplitude of 0.4 means that the baby would have to move $\sim$ 36cm back and forth in $1/24$ second.

The pinpointing and removal of these points can be done in several ways. One way is to use a Hampel filter (details in section 2.1.1) which detects and removes outliers. This is a good method to use for this dataset because it keeps the temporal dynamics without altering all points; only the ones that are marked as outliers. The result from using a Hampel filter on our raw data is shown in Figure 3.7.

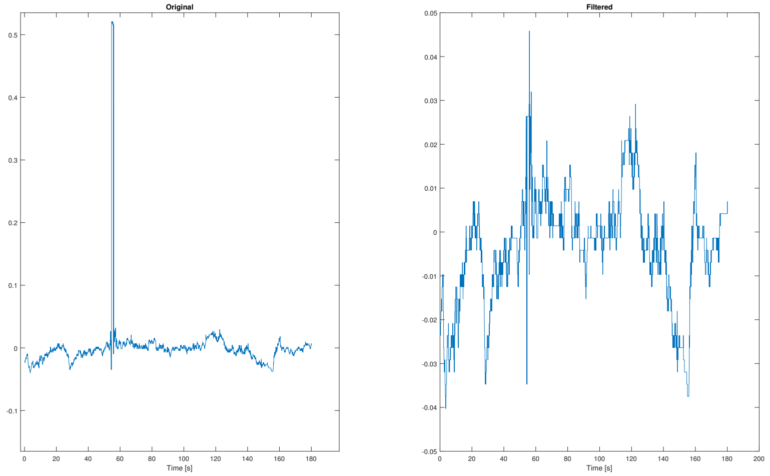

Figure 3.7: Raw data pre-processed with a Hampel Filter.

Chapter 4

Results

4.1 Initial testing of methods

As part of the subject ’TTK19 Structures and contexts in complex systems’ a subset of the data was analyzed to start investigate the data and look for structures.

4.1.1 Method

A subset of 29 infants of which 10 have CP was used. Two of the original 14 variables was chosen, hence only looking at the x- and y-coordinates from the right hand.

First a PCA was done on the coordinate data to see if there was some structure in the original data. Then some features were extracted and analyzed. The features that were extracted were Power Spectral Density and frequency (through a Fast Fourier Transform).

4.1.2 Results

Using PCA on the coordinates unfolded over time gave no results. This is expected as PCA does not care about the ordering of columns and rows and only compare the coordinates point by point [25]. As our hypothesis is that it’s the movements and movement qualities that holds the information, only looking at the spatial information gives us nothing.

Power Spectral Entropy finds a little more structure than the coordinates themselves. From Figure 4.1 it is shown that using a Quadratic Discriminant Analysis on the PSE gives $86,21\%$ accuracy. The confusion matrix for this model is shown in Table 4.1. The model has a sensitivity of $\frac{\text{true positives}}{\text{true positives}+\text{false negatives}} = \frac{9}{9+1} = 90\%$ and a specificity of $\frac{\text{true negatives}}{\text{true negatives}+\text{false positives}} = \frac{16}{16+3} = 84,2\%$.

The frequency distribution given from FFT gives an even more satisfying results. Quadratic LDA has an accuracy of $93,1\%$. This shows that it may be possible to find some structure in the frequencies of the motion that can separate the healthy babies from the sick. The confusion matrix for this model in shown in Table 4.2. The model has a sensi-

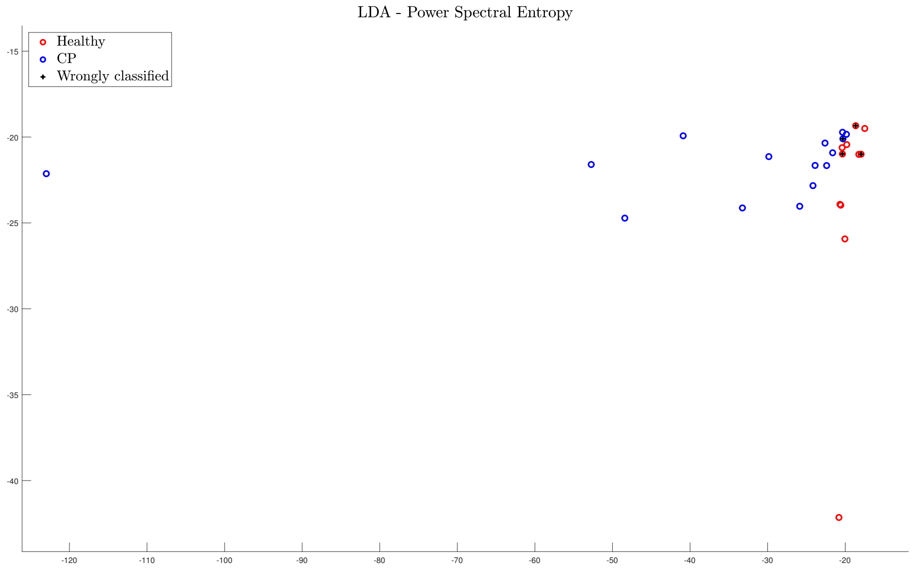

Figure 4.1: Quadratic LDA on PSE data. Accuracy: $86,21\%$

Actual→ **Predicted ↓**	**CP**	**Healthy**
CP	9	3
Healthy	1	16

Table 4.1: Confusion matrix for LDA on PSE

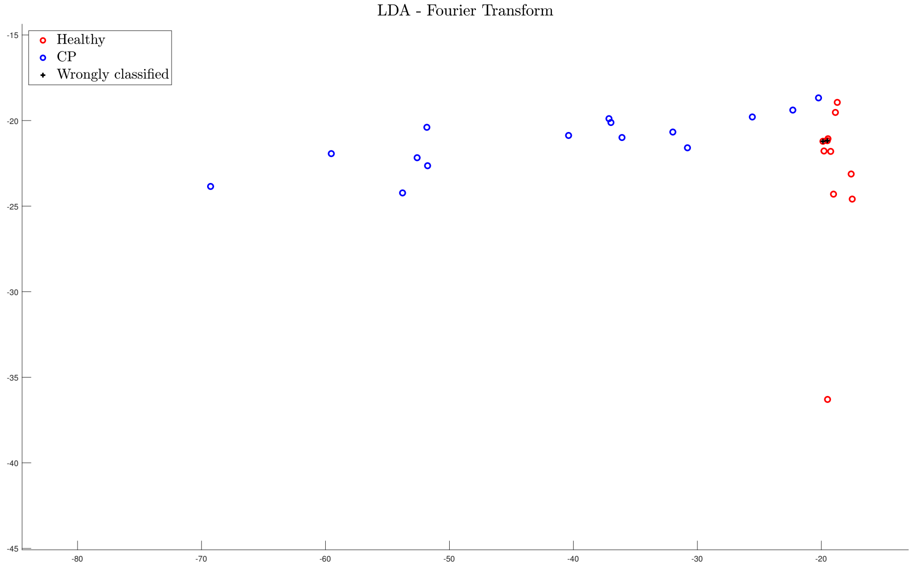

Figure 4.2: Quadratic LDA on FFT data. Accuracy: $93,1\%$

Actual→ Predicted ↓	**CP**	**Healthy**
CP	10	2
Healthy	0	17

Table 4.2: Confusion matrix for LDA on FFT

tivity of $\frac{\text{true positives}}{\text{true positives+false negatives}} = \frac{10}{10} = 100\%$ and a specificity of $\frac{\text{true negatives}}{\text{true negatives+false positives}} = \frac{17}{17+2} = 89,5\%$.

4.1.3 Conclusion

Both Power Spectral Entropy and frequency distribution gives us satisfying results. Both models have a higher sensitivity than specificity. This is better than the other way around, as it is better to treat a patient as if it has CP a little while longer than it is to give false hope.

Because of the small size of the subset chosen the resulting models obtained above cannot be used without further research. The results show that even a quick test of feature extraction where almost all parameters are chosen at random may lead to some interesting findings. Further testing is needed in both frequency calculations and choice of classification algorithm, but both features shows great promise.

4.2 Feature extraction with HCTSA

A subset of 20 and 100 subjects from the data set is run through HCTSA. The data is divided into groups using the labels 'CP_status0' for healthy and 'CP_status1' for those with CP.

HCTSA uses $\sim$ 2 full days to compute 7873 features for 100 subjects. Computation for one time series only takes $\sim$ 2 minutes, but because the data set consists of 14 time series per subject this gives a total of 1400 time series. This is ok to do before feature selection, but it is clear that one cannot do this full computation for all 900 subjects. This computation is done to gain an insight into which features may be usable in classification.

HCTSA includes framework for comparing the performance of all the operations (features) using different classification algorithms. By default it uses a linear classifier, but it also supports evaluation using SVM. The top operations when evaluating using a linear SVM are shown in 4.3. These operations are based on different numerical computations, and they are based on a set of the same methods.

The top features (or operations) are given in a certain system. An overview of the system is shown in Figure 4.3. The first parts tells which operation is done (in this case the time-series is divided into segments of length l at random and for each some statistic is calculated), then the parameters that are given to the operation is listed (here $l = 50$ is the length of the segments and 100 is the number of segments to extract) and last the structure of the output (here the feature is the standard deviation for the autocorrelation for each segment with $\tau = 1$).

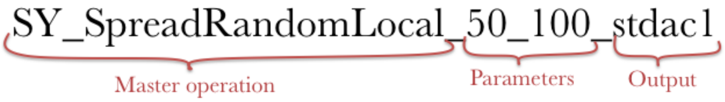

Figure 4.3: Structure of operations given by HCTSA.

In Table 4.3 it is shown that the top features includes Wavelet Decompositions, different measures of autocorrelation, computations of stepdetections and much more. One specially interesting feature is 'NL_TSTL_ReturnTime', which is a nonlinear method. This operations analyzes the histogram of return times. Return time is the time taken for the time-series to return to a similar location in phase space for a given reference point. Strong peaks in the histogram indicates periodicities in the time-series. Periodicities in the data may be an indication of repeating movements, which is typical for fidgety movements. This feature may help us locate these movements, and is worth looking further into.

Another tool in HCTSA is classification. HCTSA has built in support for investigating how accurately a classifier can performing using all of the computed features. Given a classification method and number of PCs it computes the classification results using a principal components reduced version of the data matrix. The classification rate result of a classification with SVM is shown in Figure 4.4. The total accuracy for this classification was 73.45% using all features. The default validation method for the classification with HCTSA is 10-fold cross validation.

Name	Accuracy
WL_coeffs_db3_2_wb99m (wavelet,lengthdep)	67.97%
WL_coeffs_db3_4_wb99m (wavelet,lengthdep)	67.94%
WL_coeffs_db3_5_wb99m (wavelet,lengthdep)	67.94%
CP_ML_StepDetect_l1pwc_005_E (stepdetection)	65.57%
CP_ML_StepDetect_l1pwc_10_minstepint (stepdetection)	65.10%
CP_ML_StepDetect_l1pwc_005_minstepint (stepdetection,lengthdep)	65.10%
CP_ML_StepDetect_l1pwc_02_minstepint (stepdetection)	65.05%
NL_TSTL_ReturnTime_5_1_40_n1_1_8_iqr (nonlinear,tstool)	64.70%
CP_ML_StepDetect_l1pwc_10_medianstepint (stepdetection)	64.62%
EN_rpde_3_ac_H (entropy)	63.83%
SY_SpreadRandomLocal_50_100_stdac2 (stationarity)	62.90%
SY_SpreadRandomLocal_50_100_stdac1 (stationarity)	62.79%
SC_FluctAnal_2_nothing_50_logi_r1_alpha (scaling)	62.31%
EN_rpde_3_ac_H_norm (entropy)	62.28%
CO_Embed2_Shapes_tau_circle_1_hist_ent (correlation,embedding)	62.22%
ST_LocalExtrema_n100_meanabsext (distribution,stationarity)	61.83%
ST_LocalExtrema_n100_medianabsext (distribution,stationarity)	61.68%
PH_ForcePotential_sine_1_1_1_tau (dynamicalSystem)	61.64%
SC_FluctAnal_2_nothing_50_logi_r1_se2 (scaling)	61.61%
SY_SpreadRandomLocal_50_100_meanac1 (stationarity)	61.48%

Table 4.3: Top 20 operations for subset of 100 subjects classified using a linear SVM.

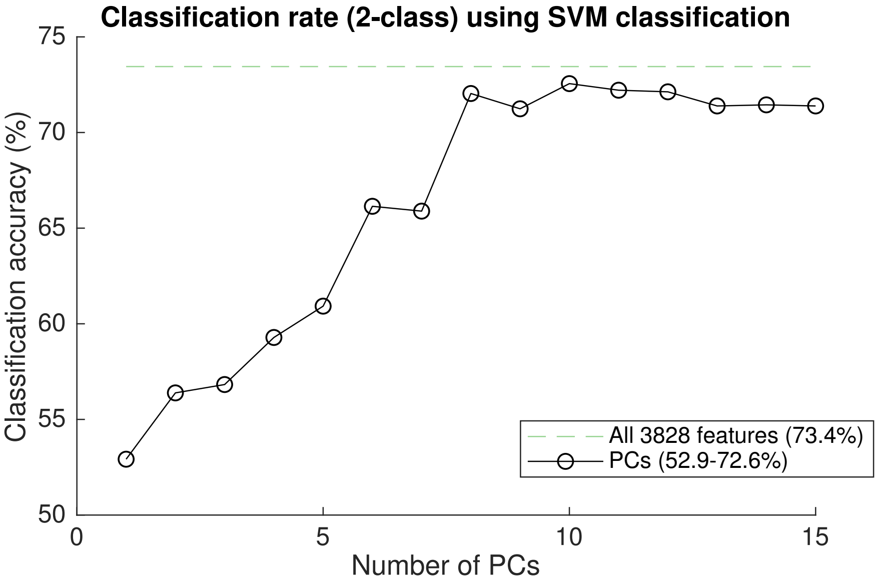

Figure 4.4: Classification rate for PCs using SVM classification

Chapter 5

Future Work

The work done in this report is mainly theoretical and focuses on exploring possible analysis methods. The implementation of the methods, analyzing the data and finding a good model is work that needs to be done in the full master thesis.

The master thesis in the spring of 2019 will focus on using the features found by HCTSA and look even further into which of these gives useful information about the data. It will also check whether adding cross correlation and other features that compares the time-series for a given subject can increase the classification accuracy.

It will look into the use of PARAFAC and PCA on the feature set to reduce the dimension. These methods will also be used when choosing the feature subset and comparing which features provides the most information.

A classifier must also be trained using a well-chosen training set and validated against a test set. This classifier will be chosen based on computational efficiency, physically interpretable results and a high classification accuracy.

Bibliography

[1] Jensenius A. R. Taraldsen G. Grunewaldt K. H. Sten R. Adde L., Helbostad J. Early prediction of cerebral palsy by computer-based video analysis of general movements: a feasibility study. *Developmental Medicine Child Neurology*, 2010.

[2] Peter O D Pharoah Allan Colver, Charles Fairhurst. Cerebral palsy. *Lancet*, 2014.

[3] Jernej Barbič, Alla Safonova, Jia-Yu Pan, Christos Faloutsos, Jessica K. Hodgins, and Nancy S. Pollard. Segmenting motion capture data into distinct behaviors. pages 185–194, 2004.

[4] Ramus Bro. Parafac. tutorial and applications. *Chemomemcs and Intelligent Laboratory Systems 38 (1997) 149-171*, 1997.

[5] Camo. Pls-da. `https://www.camo.com/resources/pls-da.html`. Accessed: 2018-12-05.

[6] Boley D. Papanikolopoulos N. Cao D., Masoud O. T. Online motion classification using support vector machines. *International Conference on Robotics Automation*, April 2004.

[7] CerebralPalsy.org. Aap urges for early diagnosis. `https://www.cerebralpalsy.org/about-cerebral-palsy/diagnosis/aap`, 2013. Accessed: 2018-10-23.

[8] Chowdhury, Reaz, Ali, Bakar, Chellappan, and Chang. Surface electromyography signal processing and classification techniques. *Sensors*, sep 2013.

[9] Peter B. Marschik Christa Einspieler, Robert Peharz. Fidgety movements - tiny in appearance, but huge in impact. *Journal de Pediatria*, dec 2015.

[10] Cross-correlation. Cross-correlation — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/wiki/Cross-correlation`, 2018. Accessed: 2018-12-06.

[11] Data mining. Data mining — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/wiki/Data_mining`, 2018. Accessed: 2018-12-04.

[12] Feature selection. Feature selection — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/wiki/Feature_selection`, 2018. Accessed: 2018-12-04.

[13] Ben D. Fulcher and Nick S. Jones. Highly comparative feature-based time-series classification. *IEEE Transactions On Knowledge and Data Engineering*, 26, December 2014.

[14] Ben D. Fulcher and Nick S. Jones. hctsa: A computational framework for automated time-series phenotyping using massive feature extraction. *Cell Systems*, 5, November 2017.

[15] Garbage in, garbage out. Garbage in, garbage out — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/wiki/Garbage_in,_garbage_out`, 2018. Accessed: 2018-12-05.

[16] Nancy A. Murphy MD Garey H. Noritz, MD. Motor delays: Early identification and evaluation. *The American Academy of Pediatrics*, 2013.

[17] Prechtl HFR Hadders-Algra M. Developmental course of general movements in early infancy. i: descriptive analysis of change in form. *Ealry Hum Dev*, 1992.

[18] Giovanni Cioni Arend F Bos Fabrizio Ferrari Dieter Sontheimer Heinz F R Prechtl, Christa Einspieler. An early marker for neurological deficits after perinatal brainlesions. *Lancet*, 1997.

[19] Hilbert-Huang transform. Hilbert-huang transform — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/wiki/Hilbert%E2%80%93Huang_transform`, 2018. Accessed: 2018-12-06.

[20] Beckung E Hagberg B Uvebrant P. Himmelmann K, Hagberg G. The changing panorama of cerebral palsy in sweden. ix. prevalence and origin in the birth-year period 1995-1998. *Acta Paediatr.*, 2005.

[21] Andr Elisseeff Isabelle Guyon. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3, March 2003.

[22] George Kollios Jonathan Alon, Stan Sclaroff. Discovering clusters in motion time-series data. *Proc. IEEE CVPR*, June 2003.

[23] Cornelieke S. Aarnoudse-Moens Jorrit F. de Kieviet, Jan P. Piek. Motor development in very preterm and very low-birth-weight children from birth to adolescence. *JAMA*, 2009.

[24] Y. Li, S. Wang, and X. Ding. Person-independent head pose estimation based on random forest regression. In *2010 IEEE International Conference on Image Processing*, pages 1521–1524, Sept 2010.

[25] Prakash B. A. Li L. Time series clustering: Complex is simpler! *Proceedings of the 28 th International Conference on Machine Learning*, 2011.

[26] M.M. Lpez, J. Ramrez, J.M. Grriz, I. lvarez, D. Salas-Gonzalez, F. Segovia, and R. Chaves. Svm-based cad system for early detection of the alzheimer's disease using kernel pca and lda. *Neuroscience Letters*, 464(3):233 – 238, 2009.

[27] B. Marschik P. A novel way to measure and predict development: A heuristic approach to facilitate the early detection of neurodevelopmental disorders. *Pediatric Neurology*, 2017.

[28] Harald Martens. Simple algorithms for pca and plsr, 2016. Accessed: 2018-12-0.

[29] Sayan Mukherjee, Edgar Osuna, and Federico Girosi. Nonlinear prediction of chaotic time series using support vector machines. *Neural Networks for Signal Processing - Proceedings of the IEEE Workshop*, 07 1999.

[30] A. Palaniappan, R. Bhargavi, and V. Vaidehi. Abnormal human activity recognition using svm based approach. In *2012 International Conference on Recent Trends in Information Technology*, pages 97–102, April 2012.

[31] Statistical classification. Statistical classification — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/wiki/Statistical_classification`, 2018. Accessed: 2018-12-05.

[32] Antoine Stevens. kenstone - kennard-stone algorithm for calibration sampling. `https://www.rdocumentation.org/packages/prospectr/versions/0.1.3/topics/kenStone`. Accessed: 2018-12-05.

[33] Brad Swarbrick. *Multivariate Analysis for Dummies*. 07 2012.

[34] Gunn Kristin berg Nils Thomas Songstad Cathrine Labori Inger Elisabeth Silberg Marianne Loennecken Unn Inger Minichen Randi Vgen Ragnhild Sten Lars Adde Toril Fjrtoft, Kari Anne I. Evensen. High prevalence of abnormal motor repertoire at 3months corrected age in extremely preterm infants. *Official Journal of the European Paediatric Neurology Society*, 2015.

[35] A. Verikas, A. Gelzinis, and M. Bacauskiene. Mining data with random forests: A survey and results of new tests. *Pattern Recognition*, 44(2):330 – 349, 2011.

[36] Frank Westad. Cluster analysis, classification and discrimination. 2018.

[37] Doernberg NS Benedict RE Kirby RS Durkin MS Yeargin-Allsopp M, Van Naarden Braun K. Prevalence of cerebral palsy in 8-year-old children in three areas of the united states in 2002: a multisite collaboration. *Pediatrics*, 2008.