Automatic Assessment of Medication States of Patients with Parkinson’s Disease using Wearable Sensors

Murtadha Dawood Hssayeni
mdh4169@rit.edu

Follow this and additional works at: https://scholarworks.rit.edu/theses

Recommended Citation
Automatic Assessment of Medication States of Patients with Parkinson’s Disease using Wearable Sensors

Murtadha Dawood Hssayeni
Automatic Assessment of Medication States of Patients with Parkinson’s Disease using Wearable Sensors

Murtadha Dawood Hssayeni
May 19, 2017

A Thesis Submitted
in Partial Fulfillment
of the Requirements for the Degree of
Master of Science
in
Computer Engineering

R·I·T

Department of Computer Engineering
Automatic Assessment of Medication States of Patients with Parkinson’s Disease using Wearable Sensors

MURTADHA DAWOOD HSSAYENI

Committee Approval:

Dr. Behnaz Ghoraani Advisor
Assistant Professor

Dr. Andreas Savakis Professor

Dr. Andres Kwasinski Associate Professor
Acknowledgments

I would like to express my special thanks and gratitude for my advisor Dr. Behnaz Ghoraani for her timely support, optimal guidance throughout my research. Also, I would like to thank Dr. Michelle Burack for providing the datasets used in this work and clarifying any concern I had about Parkinson’s disease. I would like to thank Dr. Andreas Savakis and Dr. Andres Kwasinski for serving as my committee members and for their time and effort in supporting my research. I would like to thank the higher committee for education development in Iraq for support me and my family to complete my thesis and my Master degree.

The first dataset that were analyzed in this work were supported by Small Business Innovation Research (SBIR) grant offered by National Institutes of Health (NIH) to Cleveland Medical Devices (1R43NS071882-01A1; T. Mera, PI). Frank Bonsignore assisted with data collection, and Dr. Roople Unia and Dr. Annie Killoran assisted with video scoring.

The second dataset that were also analyzed in this study were collected under the National Institute on Aging of NIH grant to Great Lakes NeuroTechnologies Inc. (5R44AG044293). Dr. Chris Pulliam and Dr. Dustin Heldman contributed to the study design of this dataset, Charlotte MacGowan, Arita McCoy, Nicole Bonsavage, and Faisal Alerwy assisted with data collection, and Dr. Behnaz Ghoraani supported Nicholas DaCosta to annotate the ADL tasks.

Thanks for all patients who participated in the two datasets, thanks for all the people who supported, assisted with the data collection and participated in scoring and annotating the recorded signals. Thanks for my family and friends who supported me in every moment throughout my Master degree.
To my precious parents, brothers and sisters who support me no matter what the situation is and whether I am close by or far away.

To my lovely wife who stands by my side in every step toward achieving my goals.

To my adorable son who motivates me and gives me hope in the future.
Abstract

Motor fluctuations are a major focus of clinical managements in patients with mid-stage and advance Parkinson’s disease (PD). In this thesis, an automated algorithm is developed to identify those fluctuations (i.e., medication OFF and ON) using wearable sensors while PD patients are engaging in a variety of daily living activities. Four different methods are proposed which are supervised learning using Support Vector machine (SVM) with fuzzy classification, semi-supervised learning using k-means or using Self-organizing Tree Map Algorithm (SOTM) with fuzzy classification, and supervised classification using Long short-term memory (LSTM) as a deep learning method.

A set of temporal and spectral features are extracted from the ambulatory signals of triaxial gyroscope sensors. After performing dimensionality reduction, the features are introduced to SVM or clustering methods using k-means or SOTM. Signals of the gyroscope sensors are passed directly to LSTM network. The developed methods were evaluated on two datasets that included recordings of 19 PD patients. Two scenarios were considered: general training/classification and patient-specific where the former trains and tests the algorithm using subject-based leave-one-out cross-validation, and the latter trains and tests the algorithm for each patient individually. In addition, for patient-specific scenarios, the number and placement of sensors is selected for each patient and this selection is based on the average change in UPDRS score between ON and OFF medication states and the presence of tremor for that patient.

Overall, patient-specific algorithm resulted in a higher classification performance when it based on SVM with fuzzy classification (i.e., 80%, 82% and 78% for accuracy, sensitivity, and specificity of the OFF state, respectively). This algorithm was able to classify the medication states with high confidence (i.e., accuracy 94.86%, sensitivity 91.94% and specificity 96.83%) for the group of patients with a change of more than 15 in their UPDRS score between the OFF and ON medication states. This results are
promising and thus this algorithm can be potentially used in routine clinical practice to improve the quality of this group of PD patients.

In addition to these results, when only one sensor mounted on the ankle was used in the general training scenario, the algorithm based on LSTM performed better than SVM with 74.91%, 69.42%, and 80.55% for accuracy, sensitivity, and specificity, respectively. The promising results of LSTM show the potential outcome of developing deep learning methods in this field.
Contents

Signature Sheet i
Acknowledgments ii
Dedication iii
Abstract iv
Table of Contents vi
List of Figures viii
List of Tables xi

1 Introduction 2
  1.1 Motivation .............................................. 2
  1.2 Literature Review ........................................... 3
  1.3 Contributions ............................................. 6
  1.4 Organization of The Thesis ................................. 8

2 Materials 9
  2.1 Datasets .................................................. 9
    2.1.1 Dataset1 ............................................. 9
    2.1.2 Dataset2 ........................................... 11
  2.2 Methods .................................................. 13
    2.2.1 Support Vector Machine ............................... 14
    2.2.2 Clustering ........................................... 17
    2.2.3 Long Short-Term Memory ............................... 20

3 Proposed Methods 24
  3.1 Sensor Selection and Preprocessing ......................... 24
  3.2 Supervised Classification: SVM ................................ 25
    3.2.1 Segmentation ...................................... 26
    3.2.2 Feature Extraction ................................ 26
    3.2.3 Dimensionality Reduction ........................... 32
    3.2.4 Fuzzy Classification ................................ 35
## CONTENTS

3.3 Semi-supervised Classification: k-means ........................................... 38  
3.3.1 Fuzzy Classification ................................................................. 38  
3.4 Semi-supervised Classification: SOTM ............................................ 41  
3.5 Supervised Classification: LSTM .................................................... 43  

4 Results and Discussions ................................................................. 48  
4.1 Performance metrics ................................................................. 49  
4.2 Generally-trained Classifiers ......................................................... 49  
4.2.1 Supervised Classification: SVM ................................................ 49  
4.2.2 Supervised Classification: LSTM .............................................. 52  
4.3 Patient-specific Classifiers ............................................................ 56  
4.4 Patient-specific vs. Generally-trained Classifiers .............................. 63  
4.5 Comparison to Other Studies ......................................................... 68  
4.6 Directional Information vs. Magnitude .......................................... 74  

5 Conclusion and Future Work ............................................................ 78  
5.1 Future Work .......................................................... 80  

Bibliography ................................................................. 82
## List of Figures

2.1 The locations of the motion units on each body part for the first and second datasets. The signals that were recorded by only these motion units are used in the data analysis. .............................................. 12

2.2 An example of using a separating hyperplane between three dimensional feature vectors of two classes to depict the concept of SVM. The support vectors that specify the largest margin are annotated with dotted circles. ................................................................. 16

2.3 Randomly generated 2-dimensional data before and after clustering using SOTM are shown in part A and B, respectively. During clustering using SOTM, the global $\alpha(r)$ reset upon node generation and the dynamic hierarchical control function $H(r)$ that decays with each iteration ($r$) were saved and shown in Part C and D, respectively. . . 21

2.4 LSTM cell representation in one layer as proposed by [1] during forward propagation at time step $t$. Different colors are used to show the data flow through out different gates. Fully connected and Soft-max layers are shown on top of the LSTM cell. .................................................. 23

3.1 Overall Project Design that shows the different parts of the algorithm and how the data flows through it to make the final assessment of the PD patients’ medication states. .............................................. 25

3.2 Segmenting the recorded signal using a sliding 5-second window with 4-second overlap. ................................................................. 26

3.3 After obtaining the decision values from trained classifier for each feature vector, this figure shows the fuzzy labeling approach proposed in this thesis, and certainty calculation. ................................. 37

3.4 An example of the results of training in semi-supervised classification using k-means. The top part shows the feature vectors using three features and the bottom part relates them to their centroids and strength factors. .................................................. 42
3.5 Many to one architecture of LSTM. LSTM network is unfolded in time for forward computations of a window \((W_k)\) at kth second. Each sample that has depth of n channels is linearly transformed to have a depth of m hidden states. The hidden states from last layer after passing the last sample \((W_K(L))\) is passed through fully connected and softmax layers to assess the medication state for each window. Vertical dotted-blue lines represent a drop-out on the hidden states between the layers. Horizontal continuous-blue and continuous-black lines represent the hidden and internal states that flow through time, respectively. 46

4.1 Testing results of SVM model that was generally training using patient-based leave-one-out cross-validation (12 folds) on dataset 1. The results are the accuracy, sensitivity and specificity for multiple experiments using single or a combination of sensors for each of the patients or the folds. 53

4.2 Testing results of SVM model that was generally training using patient-based leave-one-out cross-validation (7 folds) on dataset 2. The results are the accuracy, sensitivity and specificity for multiple experiments using single or a combination of sensors for each of the patients or the folds. 54

4.3 The average classification accuracy, sensitivity and specificity of the testing data using single or a combination of sensors for each of the patients after training patient-specific models on features extracted from X, Y, and Z signals. The classification models are semi-supervised using k-means or SOTM and supervised using SVM. 58

4.4 Testing results of patient-specific SVM model for each of the patient in dataset 1. The results are the accuracy, sensitivity and specificity for multiple experiments using single or a combination of sensors. 61

4.5 The classification results (1 OFF, 2 ON) with the certainty (continuous red line between 0 and 1) for each round for patient 12 using patient-specific SVM trained and testing on data recorded using ankle sensor in dataset 1. The activities are 1=ambulation, 2=arms resting, 3=cutting, 4=dressing, 5=drinking, 6=unpacking groceries, 7=hair brush with left hand, and 8=hair brush with right hand. 63
4.6 Testing results of patient-specific SVM model for each of the patient in dataset 2. The results are the accuracy, sensitivity and specificity for multiple experiments using single or a combination of sensors.

4.7 The percentage of occurrence of the significant features for at least one of the three axes of sensors mounted on wrist, trunk and ankle to the total number of patients in dataset 1 which is twelve. The features are 1-average jerk, 2-Peak-to-peak, 3-Signal power 1-4 Hz, 4-Signal power 4-6 Hz, 5-Signal power 0.5-15 Hz, 6-Percentage of the powers for frequencies > 4Hz, 7-Temporal Shannon entropy, 8-Standard deviation, 9-The number of autocorrelation peaks, 10-The sum of autocorrelation peaks, 11-First autocorrelation peak, 12-Lag of the first autocorrelation peak, 13-Gini Index, 14-Sample entropy, 15-Mean 16-Skewness, 17-Kurtosis, 18-Spectral entropy, 19-The peak in the power spectral density, 20-Dominant frequency, 21-The second peak in the power spectral density, 22-Secondary frequency, 23-Cross-correlation between X and Y axes, 24-Cross-correlation between X and Z axes, and 25-Cross-correlation between Y and Z axes.

4.8 The box plots of the log of the feature (percentage of the powers for frequencies > 4 Hz) for three patients and two states. The training data used to extract the feature is four activities (ambulation, drinking, arm resting and dressing) from two rounds, one for each MS. The dotted red rectangle shows the overlap between the medians of OFF state for patient 4 and ON state of patient 12.

4.9 The average classification results of the testing data after training patient-specific SVM model on features extracted from X, Y, and Z signals vs. features extracted from their magnitude signal. The results of using single or multiple sensors are presented.
List of Tables

2.1 The demographics information of the analyzed subjects in this thesis for both first and second datasets. DBS: number of subjects with Deep Brain Stimulation, UPDRS: Unified Parkinson’s Disease Rating Scale, PDYS-26: 26-item Parkinson’s Disease Dyskinesia Scale, LEDD: Levodopa Equivalent Daily Dose

4.1 Average testing results of SVM model that was generally training using patient-based leave-one-out cross-validation (12 folds) on dataset 1. The results are the accuracy, sensitivity and specificity for multiple experiments using single or a combination of sensors.

4.2 Average testing results of SVM model that was generally training using patient-based leave-one-out cross-validation (7 folds) on dataset 2. The results are the accuracy, sensitivity and specificity for multiple experiments using single or a combination of sensors.

4.3 The LSTM Hyper-parameters that were explored in this thesis.

4.4 Average testing results of LSTM model that was generally training using patient-based leave-one-out cross-validation (12 folds) on dataset 1. The results are the accuracy, sensitivity and specificity using the ankle or a combination of trunk and ankle sensors.

4.5 Average testing results of LSTM model that was generally training using patient-based leave-one-out cross-validation (7 folds) on dataset 2. The results are the accuracy, sensitivity and specificity using the ankle sensor.

4.6 Average testing results of patient-specific SVM model for dataset 1. The results are the accuracy, sensitivity and specificity using single or a combination of sensors based on the average change in UPDRS score between OFF and ON state for each patient and the presence of rest tremor.

4.7 Average testing results of patient-specific SVM model for dataset 2. The results are the accuracy, sensitivity and specificity using single or a combination of sensors based on the average change in UPDRS score between OFF and ON state for each patient and the presence of rest tremor.
4.8 Average testing results of patient-specific SVM model vs. generally trained SVM and LSTM models for dataset 1. Training patient-specific SVM used single or a combination of sensors based on the average change in UPDRS score between OFF and ON state and the presence of rest tremor for each patient, whereas, training generally-trained SVM and LSTM used two sensor mounted on trunk and ankle, and only one sensor mounted on the ankle, respectively. .......................... 66

4.9 Average testing results of patient-specific SVM model vs. generally trained SVM and LSTM models for dataset 2. Training patient-specific SVM used single or a combination of sensors based on the average change in UPDRS score between OFF and ON state and the presence of rest tremor for each patient, whereas, training generally-trained SVM and LSTM used two sensor mounted on thigh and ankle, and only one sensor mounted on the ankle, respectively. .............................. 67

4.10 The number of the selected features for each patient and for each sensor location. Selected features have p-value < 5% significant level as a result of the statistical analysis on the training data for each patient separately, and all the patients (shown in last row). Number of selected features in bold represents relatively low number of significant feature for that location. ................................................................. 70

4.11 Summary of the approaches in the published literature proposed to classify the medication states, in addition to summary of patient-specific classification using SVM proposed in this thesis. ....................... 73

4.12 The number of significant features that are selected using the statistical analysis on features extracted from X, Y and Z axes or from the magnitude of these axes for each patient and for the sensors mounted on wrist, trunk, and ankle. This table also shows number of features that were significant if they were extracted from X, Y or Z signals for each of the sensors and not significant if they were extracted from their magnitude, and vice versa. ............................... 76
1.1 Motivation

Parkinson’s disease (PD) is a chronic progressive neurological disorder that leads to motor and non-motor impairments [2]. It is the second most common neurodegenerative disease after Alzheimer. Motor impairments include hand and leg tremor while resting, rigidity, reduction in the speed and quantity of spontaneous purposeful movements (bradykinesia and akinesia, respectively), axial disability, and dysarthria (speech difficulty). Non-motor abnormalities consist of sleep disturbances and autonomic dysfunctions [3]. The most potent and effective treatment for motor symptoms is levodopa which is converted to dopamine, the neurotransmitter that is lost with progressive degeneration of the substantia nigra that is caused by PD. At the beginning of using levodopa, PD symptoms may totally disappear, but it does not prevent PD progression. With prolonged treatment, the majority of patients develop motor fluctuations, in which benefit from levodopa wears off between doses, with re-emergence of bradykinesia and tremor (referred to as OFF medication state), and may cause abnormal involuntary movements (dyskinesias) at peak dose effect [4] when tremor, bradykinesia, and other PD symptoms are relatively improved (referred to as ON state). Motor fluctuations are a major focus of clinical management in patients with mid-stage and advanced disease [5].
A variety of assessment methods are used to evaluate response to therapeutic interventions for motor fluctuations to progressively adjust the frequency and dose of levodopa for each PD stage that can be different for each PD patient. The first method is the Unified Parkinson Disease Rating Scale (UPDRS) that includes both objective clinical exam as well as historical information [5]. This information includes side effects of therapy such as motor fluctuations (e.g. percentage of the day spent OFF or ON with dyskinesias). However, for a patient with motor fluctuations, a single clinical exam is not sufficient to evaluate the spectrum of the motor impairments that a patient may experience over the course of a typical day. For example, single clinical exam is not enough to determine how severe the dyskinesias due to the effect of emotional changes in inducing it [3]. In addition, evaluations for invasive treatments such as deep brain stimulation (DBS) surgery typically require repeated examinations in clinic or in hospital before and after onset of medication benefit. These repeated visits are expensive and pose burden to PD patients. Another commonly used assessment tool for patients with motor fluctuations is the motor diary, in which patients log their motor status (OFF, ON without troublesome dyskinesia, and ON with troublesome dyskinesia) every thirty minutes while awake. Motor diaries pose significant burden to patients and they are subjective [6] and typically used in clinical trials rather than for routine clinical care. Therefore, there is a need to automate the assessment of the medication states using a conformable, cheap and objective prediction method.

1.2 Literature Review

The wide availability of wearable inertial sensors (accelerometers and gyroscopes) has led to active interest in developing tools for assessing PD symptoms and tracking patient’s motor fluctuations more comprehensively, accurately, and with less burden on clinicians and patients. PD patients prefer to wear unobtrusive sensors for medication state assessment even in public instead of logging their medication state to a diary [7].
Multiple methods that are based on inertial sensors have been proposed and embedded into systems available in the market to detect PD motor impairments [3, 8]. The methods are proposed to detect and assess the severity of tremor [9–17], bradykinesia [9, 11, 16, 18, 19], and dyskinesia [10, 12, 13, 17, 18, 20–26]. Some of these methods accurately detect PD motor impairments during constrained and unconstrained activities. The results of these methods can be helpful for neurologists if the requirement is individually tracking each one of the symptoms. However, adjusting the medication to reduce motor fluctuations requires the PD physicians to spend a lot of effort and time to further analyze the severity of all PD symptoms and dyskinesias together over time to find the intervals of the ON and OFF states during the day. Therefore, there is a need to detect the medication states in addition to detecting and assessing severity of PD symptoms and dyskinesia.

There are a variety of approaches in the published literature that are applied to classify medication treatment status (ON versus OFF) [9, 17, 24, 27–32]. They use accelerometer and gyroscope sensors that are worn on different parts of the body (wrist [9, 27, 31, 32], leg [9, 27, 28] or/and waist or trunk [9, 17, 24, 27–30]), and they use different windows sizes in the range of 3 seconds to 10 minutes with or without overlap for signal segmentation. For feature extraction, [9, 27] extract temporal features, whereas [17, 24, 28–32] extract both temporal and spectral features. Multiple supervised classification methods are used: linear discriminant (threshold-based) [24, 27–29], Support Vector Machine (SVM) [17, 30]), Artificial Neural Network (ANN) [32], logistic regression [9], and deep learning [31].

Some of these approaches are activity-independent and providing continuous monitoring but they achieved accuracy lower than 80% [9, 17, 27, 31, 32] which is not adequate to be used in the routine clinical practice. Other methods trade continuous monitoring with performance and they achieved accuracy over 90%, but only when patients’ activities are constrained [24, 28–30]. For example, [28] excluded periods of
walking to minimize misclassification and they did use a held out set for validation. That means other periodic activities may be also misclassified and should be excluded (e.g. teeth brushing and cutting groceries). In addition, [24,30] and [29] limit the analysis to period of walking, or both walking to detect bradykinesia and non-walking to detect dyskinesia, respectively. Therefore, algorithms that can accurately classify medication status during the variety of activities that people routinely carry out in their daily lives have thus far been elusive.

In addition, the majority of the approaches to date have involved classification algorithms that are trained using data from a group of well-characterized patients, and then generalized to individual patients for testing [9,17,24,31,32]. In this classification problem, these approaches may not generalize well because the accuracy is likely impacted by the significant variability between patients with respect to:

- **Somatotopy** (which body parts are impaired that may require different sensor’s placement on the patient body and different number of sensors).

- **Phenomenology** (which of the different motor impairments are present, for instance, dyskinesia appears in 50% of PD patients with prolonged treatment with levodopa [3]).

- **Severity of PD symptoms** (e.g. one patient’s best movement speed may correspond to a less advanced PD patient’s worst movement speed).

This variability requires different classification models for each patient. [27–30] adapted their models for specific group of patients or for each patient. [28] categorized their datasets into two groups of patients with and without tremor, calculated specific threshold for the linear discriminant method for each of them, and concluded that the sensor placement on leg for first group and on trunk for second group is more informative. [29] used patient-specific threshold for bradykinesia detection only, whereas [30] and our previous works [33,34] trained patient-specific algorithm to detect the medica-
CHAPTER 1. INTRODUCTION

The advantage of our previous work over [30] is predicting PD medication states without constraining patient’s activities.

In this thesis, patient-specific algorithm is developed for automatic detection of the motor states of patients with Parkinson’s disease without constraining patients’ activities. Four methods that include using shallow or deep learning are proposed to perform the classification. In addition, a comparison is completed between patient-specific and general training algorithms. The algorithm uses gyroscope signals recorded by wearable sensors and can help clinicians to assess different states of PD as part of the routine clinical practice.

1.3 Contributions

The contributions of this thesis are summarized as follows:

1. Developed patient-specific algorithm for automatic detection of the motor states of patients with Parkinson’s disease without constraining patients’ activities.

2. Developed two supervised methods based on Support Vector machine (SVM) with fuzzy classification of feature vectors with temporal dependencies, and based on Long short-term memory (LSTM) that is used for the first time in the detection of PD medication states.

3. Developed two semi-supervised methods based on k-means and Self-organizing Tree Map Algorithm (SOTM).

4. Integrated the three dimensional representations of the gyroscope signal instead of using their magnitude for all the developed methods.

5. Extracted features and developed dimensionality reduction method based on statistical analysis and Principal Component Analysis (PCA).
6. Developed novel quantitative metric denoted as strength factor and used in the semi-supervised methods to perform binary classification.

7. Proposed a novel approach for selecting the number and placement of the gyroscope sensors for each patient specifically to perform medication state classification.

8. Evaluated the proposed methods using two scenarios: general training/classification and patient-specific, and using two datasets of 19 PD patients.


12. Received National Science Foundation (NSF) award for young professionals during IEEE Engineering in Medicine and Biology Society Conference (EMBC 16) for our work in assessing PD medication states.
1.4 Organization of The Thesis

The thesis is organized in five chapters. Brief information about each chapter is mentioned below:

- **Chapter 1. Introduction**: explains the motivation behind the work that has been performed toward completing the thesis, literature review about the proposed tools and methods to assess PD symptoms and medication states, and the contribution of this thesis.

- **Chapter 2. Materials**: describes popular machine intelligence methods that are the basis for the proposed methods in chapter 3, and illustrates the datasets that were used in this thesis in details. The methods are Support Vector Machine (SVM), clustering using k-means and Self-organizing Tree Map (SOTM), and Long Short-Term Memory (LSTM).

- **Chapter 3. Proposed Methods**: describes four proposed method to assess PD medication states in details. These methods are supervised classification using SVM and LSTM, and semi-supervised classification using k-means and SOTM.

- **Chapter 4. Results and Discussions**: explains the metrics that is used to test the proposed methods, feature analysis, results for each of the methods using the two datasets, and comparisons between the methods and between the results of this thesis with other studies.

- **Chapter 5. Conclusion**: mentions a brief summary about the thesis and the obtained results then concludes the outcomes and future work.
Chapter 2

This chapter starts by describing the datasets that are used in this thesis in details. The second section has a brief introduction about machine intelligence and the required steps to build a model to solve a classification or regression problems. In addition, supervised classification method using support Vector Machine (SVM) and Long short-term memory (LSTM), and clustering methods are explained. These methods are the base for proposed methods in next chapter.

2.1 Datasets

2.1.1 Dataset1

Monitoring PD motor impairments during a variety of activities motivated designing a protocol [25, 26] to collect the first dataset that is used in this thesis. Fifteen patients who were diagnosed with idiopathic PD participated in that study and these patients were also showing choreatic peak-dose dyskinesia. The upper part of Table 2.1 includes the demographics of the twelve subjects that are included in the data analysis in this thesis. Three subject are excluded from the analysis because they didn’t show motor fluctuation between the two states during the scheduled time to record the data.

KinetiSense motion sensor unit (Great Lakes NeuroTechnologies Inc., Cleveland,
OH) was used in the study. It is composed of a triaxial accelerometer and triaxial gyroscope. The dimensions of the unit are $22 \times 16 \times 10$ mm and it weights 10 g. The sampling rate of this unit is 128 Hz. In addition to the sensor units, a local computer was used to save the data that was recorded by the sensor units and wirelessly transmitted to the computer.

Before starting the experiment, patients with mild PD were asked to hold their medication after 10 pm of the day before the experiment, whereas patients with more advanced PD were scheduled to do the experiment at the end of a dose interval. Therefore, the experiment started the first round while the patients were in the OFF state. The patient were instrumented with ten of the sensor units that were mounted on the left temple of head, C7 spinous process (trunk), dorsum of each wrist, each ankle, and frontal surface of each thigh and arm. Figure 2.1 shows the locations of the sensors that are included in the data analysis in this thesis.

The experiment was divided into four rounds for each subject. In each round, the subjects were instructed to perform seven daily living activities which were walking, resting while sitting, drinking from a cup for multiple times, dressing by putting on a coat and taking it off, hair brushing, unpacking groceries, and cutting food using a knife and fork. Each of these activities were about 15-60 seconds. At the end of each round, Unified Parkinson’s Disease Rating Scale (UPDRS) was examined by a clinician to categorize each round as OFF or ON. Following the recording in the first round, the subjects took their prescribed anti-parkinsonian medications (without adding booster dose) and went back to their typical medication schedule. After taking the medication, next three rounds were recorded once per hour. All the tasks were videotaped and the videos were used later by two blinded movement disorder experts to score dyskinesias using the modified Abnormal Involuntary Movement Scale (mAIMS).
CHAPTER 2. MATERIALS

Table 2.1: The demographics information of the analyzed subjects in this thesis for both first and second datasets. DBS: number of subjects with Deep Brain Stimulation, UPDRS: Unified Parkinson’s Disease Rating Scale, PDYS-26: 26-item Parkinson’s Disease Dyskinesia Scale, LEDD: Levodopa Equivalent Daily Dose

<table>
<thead>
<tr>
<th>Dataset 1</th>
<th></th>
<th>Dataset 2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong># Analyzed Subjects</strong></td>
<td>12</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td><strong>Sex</strong></td>
<td>9 M 3 F</td>
<td>5 M 2 F</td>
<td></td>
</tr>
<tr>
<td><strong>Age average (yr)</strong></td>
<td>55.8</td>
<td>58.7</td>
<td></td>
</tr>
<tr>
<td><strong>Age range (yr)</strong></td>
<td>42-77</td>
<td>48-68</td>
<td></td>
</tr>
<tr>
<td><strong>Disease duration (yr)</strong></td>
<td>9.4</td>
<td>10.6</td>
<td></td>
</tr>
<tr>
<td><strong>Disease duration range (yr)</strong></td>
<td>4-14</td>
<td>8-15</td>
<td></td>
</tr>
<tr>
<td><strong>DBS</strong></td>
<td>6 with 6 without</td>
<td>3 with 4 without</td>
<td></td>
</tr>
<tr>
<td><strong>% Change in UPDRS</strong></td>
<td>52.8</td>
<td>51</td>
<td></td>
</tr>
<tr>
<td><strong>% Change range</strong></td>
<td>29-69</td>
<td>15-85</td>
<td></td>
</tr>
<tr>
<td><strong>PDYS-26</strong></td>
<td>35.6</td>
<td>36.84</td>
<td></td>
</tr>
<tr>
<td><strong>PDYS-26 range</strong></td>
<td>1-79</td>
<td>11-54</td>
<td></td>
</tr>
<tr>
<td><strong>LEDD (mg)</strong></td>
<td>1225</td>
<td>1380.9</td>
<td></td>
</tr>
<tr>
<td><strong>LEDD range</strong></td>
<td>510-1930</td>
<td>862.5-1975</td>
<td></td>
</tr>
</tbody>
</table>

2.1.2 Dataset 2

The second dataset that is used in this thesis was collected under a protocol [35], which was designed to continuously monitor PD symptoms while PD patients are engaging in daily living activities. Thirteen PD patients participated in the study. These patients had a history of dyskinesias at the peak dose of levodopa. The lower part of Table 2.1 includes the demographics of the seven subjects that are included in the data analysis in this thesis. The other subjects were excluded from the analysis in this thesis because OFF and ON states were not achieved for some of the subjects during recording session, and the data was not finalized for others at the time when
CHAPTER 2. MATERIALS

KinetiSense motion sensor unit (Great Lakes NeuroTechnologies Inc., Cleveland, OH) was used in the study. It is composed of a triaxial accelerometer and triaxial gyroscope. The sampling rate of this unit is 64 Hz. In addition to the sensor units, a local computer was used to save the recorded and wireless-transmitted data.

The experiment was started while the patients were in their OFF state. Therefore, the study was either in the morning before the patients take their normal dose of levodopa or later in the day at the end of a dose for patients with known end of dose time. The patient were instrumented with eight of the sensor units that were mounted on index finger of each hand, dorsum of each wrist, each ankle, and frontal surface of each thigh. Figure 2.1 shows the locations of the sensors that are included in the data analysis in this thesis.

The experiment venue consisted of six stations to mimic activities of daily living (ADL). These stations were: dressing station where subjects put on and take off shoes and coat, entertainment station where subjects talk, read, or watch TV while sitting

**Figure 2.1:** The locations of the motion units on each body part for the first and second datasets. The signals that were recorded by only these motion units are used in the data analysis.
on a couch, snack station where subjects prepare and eat a snack and/or drink a beverage while sitting at a table, hygiene station where subjects comb their hair and brush their teeth, laundry station where subjects fold laundry from a basket, and desk work station where subject write and work on a computer.

After determining that the subjects in the OFF state by clinical exam and physician’s direct observation, the subjects were asked to cycle through all the station and recording the data started. After completing all the stations at least once, the subjects were instructed to take their normal prescribed dose of PD medication. Subjects were left to perform the activities that they prefer before their medication starts its effect. After confirming that they were in the ON state by physician’s direct observation, they were asked again to cycle through all the stations at least once. The total time for the experiment for each subject was about two hours of ADL tasks. In addition to these activities, UPDRS exam was performed by the physician at the beginning of the study for all the subjects. The ADL tasks were videotaped and used for subsequent offline clinical ratings by blinded movement disorder expert, and for annotating the ADL tasks that will be used in the training part of the developed algorithm and in analyzing its performance.

2.2 Methods

In machine intelligence, supervised classification or regression algorithms are constructed and trained on labeled samples (feature vectors) to predict the labels or classes of unknown samples. If only small amount of the training samples are labeled and the majority of these samples are unlabeled, or if unlabeled data are clustered prior to use their labels then the developed algorithms are called semi-supervised learning. If only unlabeled data are used in training the models, then this kind of learning is called unsupervised such as clustering methods.

The first step to solve a problem using machine intelligence is collecting the raw
CHAPTER 2. MATERIALS

data and starting labeling it if that is possible. This data can be images, signals, text, or other type of representation. The second step is filtering or refining the data. The next step is determining the boundaries of patterns that present in the data and relate to the problem which is known as segmentation such as segmenting word or syllables, and segmenting accelerometer signal to cover a specific activity. The last step prior to choose a classification or regression model is dimensionality reduction that includes feature extraction and selection. Feature extraction is the process of deriving new values from the segmented data to form sample or feature vectors that will be used to train, test and validate the classification or regression models. In the next sections, different methods for supervised and unsupervised classification are explained which are Support Vector Machine (SVM), clustering using k-means and Self-organizing Tree Map Algorithm (SOTM), and Long short-term memory (LSTM).

2.2.1 Support Vector Machine

SVM is a supervised classification and regression model [36]. The basic concept of this model is constructing a separating hyperplane that has the maximum margins around it or between the samples of two classes as shown in Figure 2.2. The position of a new sample on each of the hyperplane’s sides decides which class it belongs to.

If the following equation is true for feature vector $x$, then it is called support vector ($X_{sv_{1 \times N_F}}$, where $N_F$ is the number of features) and it bounds the hyperplane (support vectors are denoted with dotted circles in Figure 2.2):

$$|W^T X_{sv} + b| = 1$$  \hspace{1cm} (2.1)

where $W_{1 \times N_F}$ is perpendicular vector to the hyperplane. To optimize the margin that is defined as $\frac{1}{||W||}$ (where $||W||$ is the second norm of $W$), quadratic programming is used to optimize Lagrange formulation and find Lagrange multipliers ($\alpha$). $\alpha$’s that
are calculated and not zero are for $X_{sv}$’s and they are used to find $W$ by applying
the following equation:

$$W = \sum_{i=1}^{n} \alpha(i) y(i) X_i$$  \hspace{1cm} (2.2) $$

where $n$ is the number of support vectors ($X_{n \times NF}$) in the training data, and $y$ is the
predefined labels of $X_{SV}$’s (+1 or -1). The calculated $W$ above is used with any $X_{sv}$
in the following equation to find the bias $b$:

$$y_{sv}(W^T X_{sv} + b) = 1$$  \hspace{1cm} (2.3) $$

The calculated $W$ and $b$ that defined the hyperplane will be used to find the decision
value ($DV$) for the test feature vector $x_{t1 \times NF}$ as follow:

$$DV = W^T x_t + b$$  \hspace{1cm} (2.4) $$

If $DV$ is $\geq 0$ then $x_t$ belongs to the positive class, else it belongs to the negative class.

Classes that are separable in low dimensional space using straight-line hyperplane
can follow the previous method. However, to discriminate between classes in multiple
problems, feature space needs to be rotated to a higher dimensional space. That
means the inner product of the feature vectors in the new high dimensional space while
applying the Quadratic programming will be computationally expensive. However,
the kernel trick can perform the inner product without transforming into or out of the
higher dimensional space and solves the previous issue. Common kernels are linear,
polynomial, sigmoid and Radial Basis Function (RBF) that is defined as:

$$K(x_i, x_j) = e^{-\gamma(x_i-x_j)^2}$$  \hspace{1cm} (2.5) $$

where gamma $\gamma$ is a free parameter that is $> 0$. Another parameter that is used in
this model is the $c$ parameter that works as a regularization parameter in case there
Figure 2.2: An example of using a separating hyperplane between three dimensional feature vectors of two classes to depict the concept of SVM. The support vectors that specify the largest margin are annotated with dotted circles.
are feature vectors that violate the margin. Sufficient high value of \( c \) prevents the feature vectors from violating the margin and leads to including more feature vectors as support vectors.

The implementation of SVM model does not provide posterior probability. Therefore, a parametric form of sigmoid is proposed by Platt et al. [37] for this purpose as follows:

\[
P(y = 1|DV) = \frac{1}{1 + e^{A \times DV + B}}
\]  

(2.6)

where \( P(y = 1|DV) \) is the probability of class \( y \) given the decision value \( DV \), and \( A \) and \( B \) are two scalar parameters that are calculated by minimizing the negative log likelihood that is based on the training set \( \{DV_i, y_i\}_{i=1:N_W} \), where \( N_W \) is the number of windows or feature vectors in the training data).

2.2.2 Clustering

Clustering is unsupervised learning method that is applied on unlabeled data to group the samples based on their similarity. Two clustering methods are employed in this thesis to obtain \( K \) clusters (\( C \)) as explained below:

2.2.2.1 k-means

K-means is a well-known iterative clustering algorithm that assigns \( n \) feature vectors (\( \vec{fv}_{1 \times N_F} \)) to \( K \) pre-defined number of clusters (\( \{C_k\}_{k=1:K} \)) with \( K \) centroids (\( \{C_k \times N_F\}_{k=1:K} \)).

The algorithm consists of the following steps [38]:

1. Initialize the centroids of the \( K \) clusters by randomly choosing \( K \) feature vectors from the training dataset.

2. Compute the Euclidean distance between all the feature vectors to all the centroid.
3. Assign each feature vector to the nearest cluster (i.e. its centroid has the minimum Euclidean distance to $\vec{f}_v$).

4. Update the centroids of the K clusters by calculating the average of all the feature vectors in each cluster.

5. Repeat from step 2 to 4 until one of the following condition:

(a) The assigned $\vec{f}_v$’s to each cluster do not change.

(b) The maximum number of iterations is reached.

In addition to the previous steps, multiple replicates can be used to find a solution that represent a global minimum by starting with different initial $\vec{f}_v$’s for the centroids in each replicate. The selected replicate has the lowest total sum of Euclidean distances between the $\vec{f}_v$’s and the centroids of the clusters to whom they were assigned.

2.2.2.2 Self-organizing Tree Map Algorithm

SOTM is a tree structured neural network in which competitive learning is implemented. The tree starts by adding a root node for the first random feature vector and then it grows by adding new nodes for the $\vec{f}_v$’s that show dissimilarity to the closest node. Node creation is governed by dynamic hierarchical control function $H(r)$ that decays with each iteration ($r$) to explore different level of similarity between the $\vec{f}_v$’s for each node. The resolution of clustering can be imposed by forcing $H(r)$ to stop at a specified level $H(\infty)$. $H(r)$ can also operates in other three modalities proposed by [39]. The exponential $H(r)$ is represented by [39]:

$$H(r) = H(0)(1 - e^{-\frac{r}{\tau_H}})$$  \hspace{1cm} (2.7)
CHAPTER 2. MATERIALS

where $H(0)$ is the sum of the ranges of all the features, and $\tau H = 2 \times N_W$ where $N_W$ is the number of windows in the training dataset. In addition, the weights in each node are adjusted using a learning rate $\alpha(r)$ that decays with each iteration till the weights’ convergence. Different modes of $\alpha(r)$ is explained by [39]. The preferred mode is the global reset upon node generation when $\alpha(r)$ is shared for all the nodes as shown in part C of Figure 2.3.

The SOTM process can be explained as follows [39]:

1. Initialize the root node weights $\vec{w}_1(r)$ with randomly selected $\vec{f}v(r)$, and set the number of nodes ($K$) and iteration $r$ to 1.

2. Take another $\vec{f}v(r)$ randomly at $r + 1$ and find the closest node ($k^*$) based on the minimum Euclidean distance ($d_{k^*}$):

$$
\begin{align*}
  d_{k^*}(\vec{f}v, \vec{w}_{k^*}) &= \min_{k=1}^{K} \| \vec{f}v(r) - \vec{w}_{k^*}(r) \| \\
  \end{align*}
$$

(2.8)

3. If $d_{k^*}$ less than or equal to $H(r)$, then update the node $k^*$ weights using the following equation:

$$
\begin{align*}
  \vec{w}_{k^*}(r + 1) &= \vec{w}_{k^*}(r) + \alpha(r)[\vec{f}v(r) - \vec{w}_{k^*}(r)] \\
  \end{align*}
$$

(2.9)

else $K = K + 1$, spawn a child node to $k^*$ and initialize its weights ($w_K(r)$) from the given $\vec{f}v(r)$.

4. Check the following conditions if any is true, then stop the algorithm and output $\{\vec{w}_k\}_{k=1:K}$ as the clusters’ centroids ($\{\vec{C}_{k \times N_F} \}_{k=1:K}$), else repeat from step 2:

(a) Maximum number of iteration ($MaxItr$) or clusters is reached. $MaxItr$ equals maximum number of epoch multiplied by $N_W$.

(b) The tree structure does not change significantly.
CHAPTER 2. MATERIALS

An example of clustering using SOTM on randomly generated 2-dimensional data is shown in Figure 2.3. Part A and B of this figure show data before and after clustering using SOTM, respectively. Three clusters were detected with the tree that represents the relationship between them. The used modes for $H(r)$ and $\alpha(r)$ were the stepped $H(r)$ with irregular period and the global reset upon node generation, respectively, which are shown in part C and D in the figure.

2.2.3 Long Short-Term Memory

The detection of patterns that have temporal dependencies requires the Artificial Neural Network (ANN) to create hidden states to save the output of the previous time step. ANN with this kind of abilities is known as Recurrent Neural Network (RNN). Adding memory cells to the architecture of the standard RNN enables it to exploit patterns presented in long duration of time. The architecture is known as Long Short-Term Memory (LSTM) [40]. In addition, LSTM solves the vanishing gradient problem that occurs if error gradients are back-propagated through multiple time steps.

Multiple architectures of LSTM have been proposed in recent years to refine the original LSTM. In this thesis, the architecture proposed by [1] is used and is shown in Figure 2.4. This architecture consists of the main part which is the memory cell ($c_t$ at time step $t$), and four gates. These gates are the input gate ($i$), input modulation gate ($g$), forget gate ($f$), and output gate ($o$). Input gate determines and writes the importance of current input at time step $t$ ($x_t$) and past hidden state ($h_{t-1}$), modulation gate summarizes both $x_t$ and $h_{t-1}$ and this summary will be governed by the input gate, forget gate determines which historical data should be reset and when, and output gate determines which part of the memory or internal state ($c_{t-1}$) is important to be output and use for next time step. The operations that are performed
Figure 2.3: Randomly generated 2-dimensional data before and after clustering using SOTM are shown in part A and B, respectively. During clustering using SOTM, the global $\alpha(r)$ reset upon node generation and the dynamic hierarchical control function $H(r)$ that decays with each iteration ($r$) were saved and shown in Part C and D, respectively.
on $x_t$ using $h_{t-1}$ and $c_{t-1}$ from previous time step are defined as follow:

\[ i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \]  
\[ g_t = \phi(W_{xg}x_t + W_{hg}h_{t-1} + b_g) \]  
\[ f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \]  
\[ o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \]

\[ c_t = f_t c_{t-1} + i_t g_t \]  
\[ h_t = o_t \phi(c_t) \]

where $W_{ab}$ is the $ab$ weight matrix ($a = \{x, h\}$ and $b = \{i, g, f, o\}$), $\sigma$ is the logistic sigmoid function, and $\phi$ is the tanh activation function. If multiple LSTM layers are used, then the inputs of top layers are $h_t$ of the layers below them. After updating the internal state of the model and determining $h_t$ of last layer, the output ($y_t$) is calculated using the following linear transformation:

\[ y_t = W_{hy}h_t + b_y \]

After obtaining $y_t$, back-propagation through time (BPTT) is used during training stage. Regularization methods can also be used here to prevent the model from overfitting the training data. These methods are training using mini-batches, dropout layers [1] and weight decay technique (L2 norm regularization).
Figure 2.4: LSTM cell representation in one layer as proposed by [1] during forward propagation at time step t. Different colors are used to show the data flow through out different gates. Fully connected and Soft-max layers are shown on top of the LSTM cell.
Chapter 3

Proposed Methods

After collecting the data, this chapter first describes type of sensors that are selected for data analysis and why, and the preprocessing steps that precede each of the proposed methods. Second, it describes the proposed methods which are: supervised classification using SVM, semi-supervised classification using k-means, semi-supervised classification using SOTM, and supervised classification using LSTM. Figure 3.1 shows the main parts in the first three proposed methods starting from preparing the data, extracting the feature vectors that are used to train and test SVM or semi-supervised classifier, performing dimensionality reduction using two consecutive approaches which are statistical analysis and Principal Component Analysis (PCA), and training and testing fuzzy classifiers. For the last section that describes classification using LSTM, feature extraction and dimensionality reduction are not performed prior to train the deep model.

3.1 Sensor Selection and Preprocessing

Each of the motion units that were used in the previous datasets contained two types of motion sensors which are gyroscope and accelerometer. Gyroscope is less affected by vibration and mechanical noise than accelerometer and it is better reflecting limb rotations that occur during tremor and dyskinesia [25]. Therefore, gyroscope recordings can be used later to accurately detect these PD symptoms. In addition, in
Tsipouras et al. work [23], Experiments 6 and 8 using gyroscope sensors showed better accuracy of levodopa-induced dyskinesia (LID) classification than experiments 5 and 7 using accelerometer sensors, respectively. Therefore, gyroscope is chosen to assess the PD medication states in this thesis and also using one sensor will increase battery life for the system when it will be used in daily life.

Bandpass FIR filter with 3dB cutoff frequency between (0.5-15Hz) is used to filter the three axes of the recorded signal of each sensor to eliminate low and high frequency noise.

![Figure 3.1: Overall Project Design that shows the different parts of the algorithm and how the data flows through it to make the final assessment of the PD patients’ medication states.](image)

**3.2 Supervised Classification: SVM**

The well-known SVM approach (refer to Section 2.2.1) has been used to perform the medication state classification in the first method. SVM has been used before successfully for classification problems related to PD [17, 24] or activity recognition [41–44] using data extracted from inertial sensors. Before training SVM model, signal
Figure 3.2: Segmenting the recorded signal using a sliding 5-second window with 4-second overlap.

segmentation, feature extraction, and dimensionality reduction are performed. The following sections describe each of these steps in details.

3.2.1 Segmentation

The last step of preparing the signal is segmenting the signals into 5-second windows with 4-second overlap between the adjacent windows to do further analysis as shown in Figure 3.2. Using 5-second as window length is found to be suitable to detect tremor, bradykinesia, and dyskinesia as a result of Patel et al. work [45].

3.2.2 Feature Extraction

Different features are extracted from each window (\( \{W_i\}_{i=1:N_W} \), where \( N_W \) is the number of windows) for each of the sensors’ axes (X,Y, and Z) separately to retain directional information in the analysis, and also for a combination of two axes. The features are selected to distinguish the following PD symptoms or levodopa side effects, or between the two states directly: bradykinesia, tremor, and dyskinesia. The peak effect of these symptoms presents in the OFF state with relatively lower or no effect in the ON state. Some of the features were previously identified as relevant to PD symptoms [10–14, 17, 18, 21–25, 28, 33, 34, 46] and others are new developed
Candidate features to detect bradykinesia and dyskinesia are:

1. Average jerk is calculated as the mean of the second derivative of the angular velocity readings in each window ($W_i$) to represent their rate of change. Values of average jerk that are close to zero indicate the presence of bradykinesia and high values indicate the presence of sudden movements that can be dyskinesia.

2. Signal power 1-4 Hz [24,34] is calculated by taking the Fast Fourier Transform (FFT) for each window ($W_i$) to convert them to their representations in the frequency domain ($W_i(f)$) and then calculating the summation of the powers of frequencies in the interval (1 to 4 Hz). High values of this feature are an indication of the presence of dyskinesias and vice versa in case of bradykinesia.

3. Standard deviation ($\sigma$) [17,23,47,48] represents the amount of variation in readings of angular velocity for each window ($W_i$). Low $\sigma$ is an indication of bradykinesia whereas high $\sigma$ is used to predict dyskinesias which are related to a wider range of angular velocities. $\sigma$ is calculated as follows:

$$\{\sigma_{W_i}\}_{i=1:N_W} = \sqrt{\frac{1}{N_S-1} \sum_{n=1}^{N_S} |W_i(n) - \mu_{W_i}|^2} \quad (3.1)$$

where $N_S$ is the number of samples in $W_i$, and $\mu_{W_i}$ is the mean of $W_i$.

4. Temporal Shannon entropy [23,49] finds the complexity and randomness in the readings of angular velocity in a given window $W_i$. High complexity indicates dyskinesias present and low entropy indicates bradykinesia. The entropy is defined as follows:

$$\{H_{W_i}\}_{i=1:N_W} = - \sum_{b \in B_i} (p(b) \ast \log_2(p(b))) \quad (3.2)$$
where \( B_i \) is the histogram bin set \{-400, -396, -392, ..., 400\} for \( W_i \) which are found to be effective in this thesis, and \( p(b) \) is the bin \( b \) probability in \( B_i \).

5. Sample entropy (SampEn) [21, 50] is a measure of noise or dissimilarity in a given window \( (W_i) \). It is calculated by first segmenting \( W_i \) into vectors of length \( m \) and \( m+1 \) that is defined as \( X_m(\tau) = \{ W_i(\tau), W_i(\tau+1), W_i(\tau+2), ..., W_i(\tau + m - 1) \} \) where \( \tau \) is the lag. The next step is calculating SampEn as follows:

\[
\{\text{SampEn}_{W_i}\}_{i=1:N_w} = -\log \frac{\sum_{j=1}^{N_G-1} \sum_{k=1}^{N_G-1} d[X_{m+1}(j), X_{m+1}(k)<r]}{\sum_{j=1}^{N_G-1} \sum_{k=1}^{N_G-1} d[X(m(j)), X(m(k))]<r}}
\]

where \( d[x, y] \) is the Chebyshev distance and every time it is less than \( r \) then increment the nominator or denominator by one, and \( r \) is the tolerance. \( m \) is set to two and \( r \) to 20\% of the standard deviation of the gyroscope signal. The presence of non-dominant segments in \( W_i \) that is related to dyskinesia leads to high SampEn, whereas bradykinesia results in low SampEn that is close to zero.

6. Gini Index is a value between 0 and 1 that also finds moving complexity in a given window \( (W_i) \) and is calculated using the following equation:

\[
\{G_{W_i}\}_{i=1:N_w} = 1 - \sum_{b \in B_i} p_b^2
\]

7. Skewness (\( \gamma_1 \)) [41, 42, 48, 49, 51] is an asymmetry measure about the mean of gyroscope readings in \( W_i \). Readings while patients are suffering from bradykinesia have \( \gamma_1 \) close to zero. \( \gamma_1 \) is calculated using the following equation:

\[
\{\gamma_{1W_i}\}_{i=1:N_w} = \frac{1}{N} \sum_{n=1}^{N_G} (W_i(n) - \mu_{W_i})^3
\]

8. Kurtosis (\( k \)) [41, 42, 48, 49, 51] represents a measure of how much a recorded
signal while patients are performing a specific activity is prone to outliers. The outliers in this case can be dyskinesias. $k$ is defined as:

$$\{k_{Wi}\}_{i=1:Nw} = \frac{1}{N} \sum_{n=1}^{N_S} (W_i(n) - \mu_{Wi})^4$$  \hspace{1cm} (3.6)

The following features are extracted for tremor as follow:

9. Signal power 4-6 Hz [14,34] is calculated by taking the Fast Fourier Transform (FFT) for each window ($W_i$) to convert them to their representations in the frequency domain ($W_i(f)$) and then calculating the summation of the powers of frequencies in the interval (4 to 6 Hz). According to UK Parkinson’s Disease Society Brain Bank clinical diagnostic criteria, Parkinsonian rest tremor is in 4-6 Hz range [52].

10. Percentage of the powers for frequencies $> 4$Hz [28] is calculated by taking the Fast Fourier Transform (FFT) for each window ($W_i$) to convert them to their representations in the frequency domain ($W_i(f)$) then find the power percentage of frequencies $> 4$ Hz.

11. The number of autocorrelation peaks is calculated by first calculating the autocorrelation for a given window $W_i$ as follows:

$$\{R_{Wi}(\tau)\}_{i=1:Nw} = \frac{1}{N_S-1} \sum_{t=1}^{N_S-\tau} (W_i(t) - \mu_{Wi})(W_i(t + \tau) - \mu_{Wi})$$  \hspace{1cm} (3.7)

where $\tau$ is the lag ($\tau = 1 : N_S - 1$). The second step is counting the number of positive peaks in the autocorrelation signal for that window. The presence of tremor in a given window results in high number of peaks related to tremor frequency for each patient.

12. The sum of autocorrelation peaks is calculated by first calculating the auto-
correlation for a given window $W_i$ using equation (3.7). The second step is summing the positive peaks in the autocorrelation signal for that window. This feature helps differentiate between tremor and periodic activities like brushing teeth.

13. Lag of the first autocorrelation peak [10,12,13] is calculated by first calculating the autocorrelation for a given window $W_i$ using equation (3.7). The second step is recording the lag that is associated with the first peak (excluding the peak at the origin) in the autocorrelation signal for that window. Tremor has a small lag because it is a fast periodic symptom.

Additional features are extracted for differentiate between the two state in general as follow:

14. Signal power 0.5-15 Hz is calculated by taking the Fast Fourier Transform (FFT) for each window ($W_i$) to convert them to their representations in the frequency domain ($W_i(f)$) and then calculating the summation of the powers of frequencies in the interval (0.5 to 15 Hz). High values of this feature are an indication that the patients are in their ON state when they move in faster pace relatively than their OFF state.

15. Peak-to-peak is the difference between the means of maximum and minimum 20 samples in each window ($W_i$). Slow movement as a result of OFF state will lead to small range of angular velocity.

16. First autocorrelation peak is calculated by first calculating the autocorrelation for a given window $W_i$ using equation (3.7). The second step is recording the autocorrelation value at the lag that is associated with the first peak (excluding the peak at the origin). The intuition behind this feature is that the specific activities, that the patients are doing in the ON state, flow easily and with more
correlation in comparison to the activities in the OFF state while the patients are suffering from hypokinesia or rigidity.

17. The mean value for each window \( (\mu_{W_i}) \) [18, 41, 42, 48, 51, 53–55] is low in the OFF state due to the slow movement and relatively high in the ON state.

18. Cross-correlation \( (\rho_{W_i^X W_i^Y}) \) [48, 53, 54] a symmetry measure between X and Y axes of the recorded gyroscope signal and is calculated as follows:

\[
\{\rho_{W_i^X W_i^X}\}_{i=1:N_w} = \frac{1}{N-1} \sum_{n=1}^{N_S} (W_i^X(n) - \mu_{W_i^X})(W_i^Y(n) - \mu_{W_i^Y}) \sigma_{W_i^X} \sigma_{W_i^Y} \tag{3.8}
\]

where \( W_i^X \) and \( W_i^X \) are the recorded windows for X and Y axes, respectively.

PD patients tend to bend their hands and/or legs in the ON state in a way that is different than their limbs direction in the OFF state. This and next two features are designed to capture this behavior to differentiate between the two states. In addition, cross-correlation is used before to differentiate between walking and going up stairs [48], so in this thesis, it can also help discriminating between leg dyskinesias and leg periodic activities.

19. Cross-correlation \( (\rho_{W_i^X W_i^Z}) \) between X and Z axes [48, 53, 54].

20. Cross-correlation \( (\rho_{W_i^Y W_i^Z}) \) between Y and Z axes [48, 53, 54].

21. Spectral entropy \( (SH) \) [22, 23, 44, 53, 54] finds complexity in the spectrum \( \{W_i(f)\}_{i=1:N_w} \).

It is calculated by first finding the normalized power spectral density for each window in \( W_i(f) \) for the frequencies \( f \) between 0.5 and 15 Hz as follows:

\[
\{P_{W_i(f)}\}_{i=1:N_w} = \frac{1}{N_S} \sum_{f=1}^{15} \left| W_i(f) \right|^2 \frac{1}{N_S} \sum_{f=1}^{15} \left| W_i(f) \right|^2 \tag{3.9}
\]
and then applying the following equation:

\[ \{SH_{W_i}\}_{i=1:N_{w}} = -\sum_{f=1}^{15} (P_{W_i(f)} \ast \log_2(P_{W_i(f)})) \]  

(3.10)

22. The peak in the power spectral density [14, 25, 46] for each window in \( W_i(f) \) where \( f \) between 0.5 and 15 Hz. With the help of the frequency associated with the peak power (next feature), they can detect the dominant PD symptom or normal activity. In addition, Weiss et al. found that PD patients in the OFF state have lower dominant frequency with lower power than their ON state [46].

23. Dominant frequency [11, 46] is associated with the peak in power spectral density for each window in \( W_i(f) \) where \( f \) is between 0.5 and 15 Hz.

24. The second peak in the power spectral density for each window in \( W_i(f) \) where \( f \) is between 0.5 and 15 Hz. The intuition behind this and next feature is to record the power and frequency of non-dominant PD symptom or normal activity that can help differentiating between the two states.

25. Secondary frequency is associated with the second peak in power spectral density.

3.2.3 Dimensionality Reduction

If it is supposed that only one sensor for each patient is used to extract the features, then sixty nine features will be extracted for the three axes. High dimensional feature space makes classification models overfit the training data and do not generalize well. This is the first reason for applying dimensionally reduction techniques. The second reason is that some of the features may not perform as proposed due to the kind of symptoms shown by the PD patients who participated in the collected dataset, the sensor locations or due to the orientation of the gyroscope’s axes. The third reason
is that some of the extracted features are correlated between each other due to the correlation that can happen between the sensor axes. Statistical analysis is applied first on the extracted features to pick the features that discriminate well between ON and OFF state. After selecting good features, the next step is reducing the correlation between the features which is performed by apply principal component analysis (PCA).

### 3.2.3.1 Statistical Analysis

Statistical analysis approach is used to select relevant features to discriminate between the two states and it is applied on the features that were extracted from the training data only. The second advantage of this approach is determining a probability value of how good each feature that can be used for further analysis of the association between the patients' PD symptoms and the selected features. The first step is determining if the samples that represents the OFF state in each of features are normally distributed using Anderson-Darling test [56], and the same normality check is applied on the samples that represents the ON state. The second step is performing one the following statistical hypothesis tests: Wilcoxon rank sum [57] or unpaired t-test [58]. The rank sum and t-test evaluate the hypothesis that the ON and OFF population in each feature come from a distribution of equal median or from a normal distribution with equal mean and variance, respectively. A p-value is obtained from both tests and $p < 0.05$ values means the ON and OFF samples in each feature are coming from two separable distributions. If one of the ON or OFF populations for a given feature is not normally distributed, then Wilcoxon rank sum is performed, else if both ON and OFF population have normal distribution, then unpaired t-test is performed. Only the significant features ($p$-value < 0.05) were included in the analysis.
CHAPTER 3. PROPOSED METHODS

3.2.3.2 Principal Component Analysis

After performing statistical analysis and in order to reduce the dominance of the features with high numerical values during the classification stage, the selected features \((F_{N_W \times N_F})\) that were extracted from \(\{W_i\}_{i=1:N_W}\) are normalized \((nF)\) by applying the following equation:

\[
\{nF_k\}_{k=1:N_F} = \frac{F_k - \mu_{F_k^T}}{\sigma_{F_k^T}}
\]  \hspace{1cm} (3.11)

where \(N_F\) is the number of the selected features, \(\mu_{F_k^T}\) and \(\sigma_{F_k^T}\) is the mean and standard deviation, respectively, of selected feature \(k\) that is extracted from \(\{W_i\}_{i=1:N_W}\) of the training data only.

Principal Component Analysis (PCA) [59] is applied on the normalized features \((nF)\) to orthogonally transform them to a lower dimension space with uncorrelated features. PCA is summarized in the following steps:

1. Finding the covariance matrix \((S)\) of \(nF\) that are extracted from the training data only as follows:

\[
S_{N_F \times N_F} = nF_{N_W \times N_F}^T \cdot nF_{N_W \times N_F}
\]  \hspace{1cm} (3.12)

where \(nF^T\) is the transpose of \(nF\).

2. Calculating the eigenvalues \((\lambda_{1 \times N_F})\) by solving the characteristic equation:

\[
det(\lambda_{1 \times N_F} I_{N_F \times N_F} - S_{N_F \times N_F}) = 0
\]  \hspace{1cm} (3.13)

where \(det\) is the determinant and \(I\) is identity matrix.

3. Calculating the eigenvectors \((x_{N_F \times N_F})\) by solving the following system of linear equations:

\[
x_{N_F \times N_F} (\lambda_{1 \times N_F} I_{N_F \times N_F} - S_{N_F \times N_F}) = 0
\]  \hspace{1cm} (3.14)
CHAPTER 3. PROPOSED METHODS

4. Sorting the eigenvalues descendingly and accordingly sorting the eigenvectors.

5. Accumulating the sorted eigenvalues or energy and saving the eigenvectors that represent 99% of the features energy ($x_{NF \times NF}$, where $NF$ is the number of saved eigenvectors).

6. Projecting the normalized features that were extracted from all the data on the $x_{NF \times NF}$ to transform them to the new dimensional space ($F_{NW \times NF}$) as follows:

$$F_{NW \times NF} = nF_{NW \times NF} \cdot x_{NF \times NF} \quad (3.15)$$

3.2.4 Fuzzy Classification

3.2.4.1 Training

Before training SVM model on $F_{NW \times NF}$ extracted from the training data, there is a need to tune SVM parameters (linear or RBF kernel, the cost parameter ($c$), and the gamma parameter ($\gamma$) in case RBF kernel is used). Grid search is performed for this purpose based on n-fold cross-validation on the training dataset. The search values of $c$ are $2^{-6, -4, \ldots, 4}$, and the search value of $\gamma$ are $2^{-8, -6, \ldots, 4}$. The folds of cross-validation are activity-based which requires identifying the activities that the patients were doing while recording the training dataset and then labeling each feature vector with specific activity. This method of cross-validation selects the parameters that make the SVM model generalizing well to new activities. The accuracy of the SVM model in each trail in the grid search is calculated and the parameters of the model with highest accuracy are chosen.

If the activities information is not available for the training data or if the SVM model failed to gain more than 65% accuracy on the activity-based cross-validation, then assigning each feature vector to the n-folds of cross-validation are performed randomly. All the parameters’ ranges and steps of grid search, other than feature
vector’s assignment to folds, remain the same as above. After selecting the optimal parameters, SVM model is trained using them on $F_{N_w \times N_x}$ extracted from the training data. After completing training, the calculated $W$ and $b$ that defines the hyperplane will be used to find the decision values ($DV_{1 \times N_w}$) for the training feature vectors in $F_{N_x \times N_w}$. After that, $DV_{1 \times N_w}$ in addition to the ground truth labels ($GT_{1 \times N_w}$) will feed into Platt scaling method to estimate $A$ and $B$ parameters.

3.2.4.2 Testing: Fuzzy labeling

The trained SVM model is used to get a decision value for each feature vector in $F_{N_w \times N_x}$ extracted from the testing data. Because of using sliding window of length 5-second and 4-second overlap, each decision value is for five seconds and four of them are used for the previous and next decision values. Therefore, if each decision value is repeated for 5 times (one for each second), then each second in the gyroscope signal will have five decision values as shown in Figure 3.3, except the four seconds at the beginning and end of the signal. The mean of a vertical window of the five decision values for each second is performed to get a single value for each second. This technique will lower the effect of outliers in the shared decision values.

The medication state does not change rapidly (usually it takes 30 minutes to change from OFF to ON or vice versa), so the adjacent decision values should be similar. Therefore, the decision values are filtered using sliding 40-second averaging filter to eliminate the effect of outliers or incorrect decisions. The last step is labeling the averaged decision values ($DV > 0$) as OFF, or as ON if they are $< 0$ to get the final medication state predictions ($MD$).

3.2.4.3 Testing: Classification Certainty

After providing prediction of the medication state for each second using the gyroscope signals, the certainty of each prediction is found in this section by applying Platt
Figure 3.3: After obtaining the decision values from trained classifier for each feature vector, this figure shows the fuzzy labeling approach proposed in this thesis, and certainty calculation.
CHAPTER 3. PROPOSED METHODS

scaling. Logistic transformation is applied to find the classification certainty (\(P\)) for
the averaged decision values that give OFF or ON predictions using the following two
equations, respectively:

\[
\{P(MD = OFF|s)\}_{s=1:ND|DV(s)>0} = \frac{1}{1 + e^{A \times DV(s) + B}} \tag{3.16}
\]

where \(N_D\) is signal length in seconds, and \(A\) and \(B\) are Platt scaling parameters.

\[
\{P(MD = ON|s)\}_{s=1:ND|DV(s)<0} = 1 - \frac{1}{1 + e^{A \times DV(s) + B}} \tag{3.17}
\]

3.3 Semi-supervised Classification: k-means

One of the problems that can face training SVM model is overfitting the training
data especially if the training data is relatively small with high number of features.
Therefore, semi-supervised classification method using k-means is proposed. In ad-
dition, analysis of the obtained clusters from this method is possible to check if the
feature vectors located in them are related to the PD symptoms or the activities the
patients were doing during data recording. This method is based on semi-supervised
classification with fuzzy labeling proposed by [60].

Segmenting the recorded signals, extracting the PD symptoms-related features,
and applying dimensionality reduction on them are performed in the same way as in
the supervised classification method using SVM (refer to Sections 3.2.1, 3.2.2, and
3.2.3).

3.3.1 Fuzzy Classification

3.3.1.1 Training

Training in this method follows the steps:
1. Choosing the number of clusters \((K)\) is important to prevent k-means from underfitting or overfitting the \(F_{NW\times N_zF}\) extracted from the training dataset. Therefore, optimal \(K\) is selected based on grid search using \(n\)-fold cross-validation on the training dataset. The search values of \(K\) are \(\{2,3,\ldots,30\}\). The folds of cross-validation are either activity-based or random assignments-based as described before in Section 3.2.4.1.

2. Using the optimal number of clusters \((K)\) from the previous step, k-means is used to cluster \(F_{NW\times N_zF}\) extracted from the training dataset. After completing clustering, the \(K\) centroids \((\vec{C}_K\times N_zF)\) are finalized.

3. Each feature vector \((\vec{f}_W_{1\times N_zF} = \vec{F}_i)\) is assigned to the closest cluster \((CC_{\vec{f}_W})\) based on the minimum Euclidean distance to each \(\vec{C}_k\) to form cluster per \(\vec{f}_W\) array \((C_{f_W_{1\times N_w}})\):

\[
\{C_{\vec{f}_W}(i)\}_{i=1:N_w} = CC_{\vec{f}_W} = \arg\min_{k=1}^K \| \vec{f}_W_i - \vec{C}_k \|
\]

(3.18)

where \(\vec{f}_W_i = \vec{F}_i\).

4. Local membership matrix \((LM_{KX2})\) is the percentage of \(\vec{f}_W\)’s being labeled as OFF \((LM(1)_{KX1})\) or ON \((LM(2)_{KX1})\) in each cluster, and is calculated in the same way in [60] using the following equation:

\[
\{LM(m)_k\}_{m=\{1,2\}, k=\{1:K\}} = \frac{j_k(m)}{N_{C_k}^W}
\]

(3.19)

where \(j_k(m)\) is the number of \(\vec{f}_W\)’s that belong to medication state \(m\) (1=OFF or 2=ON) in cluster \(k\), and \(N_{C_k}^W\) is the number of \(\vec{f}_W\)’s in cluster \(k\) \((C_k)\).

5. Global membership matrix \((GM_{KX2})\) is the percentage of \(\vec{f}_W\)’s being labeled as OFF \((GM(1)_{KX1})\) or ON \((GM(2)_{KX1})\) in each cluster to the total number of
ON or OFF $\tilde{f}v$’s in the training data, respectively. It is calculated using the following equation:

\[
\{GM(m)_k\}_{m=\{1,2\},k=\{1,K\}} = \frac{j_k(m)}{N_w^m}
\]  

(3.20)

where $N_w^m$ is the number of $\tilde{f}v$’s being labeled as $m$ in the ground truth of the training data ($GT_{1 \times N_w}$).

6. The strength factor for each cluster ($SF_{1 \times K}$) is a measure of how significant each cluster in discriminating between ON and OFF medication state. It calculated by using the following equation:

\[
\{SF(k)\}_{k=1,K} = (LM(1)_k \times GM(1)_k) - (LM(2)_k \times GM(2)_k)
\]  

(3.21)

The construction of this factor using local and global memberships of each cluster is a novel approach that is used in this thesis. The idea is to give low weights for the clusters with few number of $\tilde{f}v$’s while performing labeling on testing data. These clusters contain feature vectors that are mainly outliers. In addition, it preserves the ability to give low weights for the common clusters that contain $\tilde{f}v$’s that represents both medication states, and high weights for the clusters that mainly contain OFF or ON $\tilde{f}v$’s. The range of this factor is (-1,1). ($SF >$ zero) is for clusters that represent OFF state and higher $SF$ means the cluster is better in representing the OFF state. ($SF <$ zero) is for clusters that represent ON state and lower $SF$ means the cluster is better in representing the ON state.

7. The final step is estimating the parameters ($A$ and $B$) of Platt scaling. This is done by first finding the decision values for all $\tilde{f}v$’s ($DV_{1 \times N_w}$) by assigning $SF$ to each cluster number in $C_{\tilde{f}v_{1 \times N_w}}$ as in the equation below, and second feeding
CHAPTER 3. PROPOSED METHODS

$DV_{1 \times N_W}$ and $GT_{1 \times N_W}$ to Platt scaling method.

$$\{DV(i)\}_{i=1:N_W} = SF(C_{\vec{f}v}(i))$$  \hspace{1cm} (3.22)

An example of the training results is shown in Figure 3.4. This figure shows a sample of the clustered $\vec{f}v$’s from the training data of dataset1 using three features, the clusters and the medication states on top on them using different colors and notations. In addition, it relates some of the clusters to their centroids and strength factors.

3.3.1.2 Testing

Training the semi-supervised classifier results in $K$ centroids $(C_{K \times N_x})$, strength factor for each cluster $(SF_{1 \times K})$, and Platt scaling parameters $(A$ and $B)$. These saved parameters are used in the testing procedure following the steps:

1. Each feature vector in $F_{N_W \times N_x}$ extracted from the testing data is assigned to the closest cluster $(CC_{\vec{f}v})$ based on the minimum Euclidean distance to each $C_k$ to form cluster per $\vec{f}v$ array $(C_{\vec{f}v_{1 \times N_W}})$ using Equation 3.18.

2. Each cluster number in $C_{\vec{f}v_{1 \times N_W}}$ is assigned the corresponding value from $SF_{1 \times K}$ to find the decision values for all $\vec{f}v$’s $(DV_{1 \times N_W})$ using Equation 3.22.

3. The same procedure for fuzzy labeling and classification certainty described in Section 3.2.4.2, and Section 3.2.4.3 are used to complete this method.

3.4 Semi-supervised Classification: SOTM

One of the limitations of using K-means for semi-supervised classification is pre-defining the number of clusters ($K$) before starting clustering. The solution that we proposed before in Section 3.3.1.1 is time and computational power consuming if it
Figure 3.4: An example of the results of training in semi-supervised classification using k-means. The top part shows the feature vectors using three features and the bottom part relates them to their centroids and strength factors.
used for large training dataset. Therefore, SOTM (refer to Section 2.2.2.2 for details) is used in the semi-supervised classification instead of k-means in this method because no need to predefine $K$. SOTM algorithm determines $K$ value based on the data distribution and the predefined resolution of clustering.

The same procedure that is described in semi-supervised classification using k-means (Section 3.3) is used in this method except the first and second steps in the training section 3.3.1.1. The two steps are changed as follow:

1. Setting clustering resolution using SOTM is important to prevent it from overfitting the $\mathbf{F}_{N_W \times N_x F}$ extracted from the training dataset. Therefore, grid search is performed using n-fold cross-validation on the training dataset. It searched the following values of $H(\infty)$: (10%, 20%, ..., 90%) of the maximum standard deviation of features in $\mathbf{F}_{N_W \times N_x F}$. $H(\infty)$ represents the minimum value of $H(r)$ and prevents the SOTM from overfitting the training data. The folds of cross-validation are either activity-based or random assignments-based as described before in Section 3.2.4.1.

2. Using the minimum value of the hierarchical control function $(H(\infty))$ from the previous step, SOTM is used for clustering $\mathbf{F}_{N_W \times N_x F}$ extracted from the training dataset. After completing clustering, the K centroids ($\bar{C}_{K \times N_x F}$) and The strength factors ($SF_{1 \times K}$) are finalized and can be used for testing.

### 3.5 Supervised Classification: LSTM

Nowadays, architectures that are based on deep learning including recurrent neural networks proven to be effective and secure the state of the art in different application. From the applications that based on data recorded using inertial sensor are medication state assessment [31], the detection of bradykinesia [19], and activity recognition [61, 62]. However, deep learning is not fully investigated in the applications of
detecting and assessing PD symptoms and states, and to the best of our knowledge, LSTM network was not used before to discriminate between the medication states. Therefore, deep learning represented by LSTM architecture is implemented in this method to explore the possibility of assess the medication states given that there are a temporal dependencies between PD symptoms and the medications states in general.

Before feeding the signals of the inertial sensors to the LSTM network, two types of segmentation is experimented which are 5-second windows with 4-second overlap as described in Section 3.2.1 or 1-second window without overlap. Constructing the mini-batch follows the method presented by [61]. In this method, \(N_B\) points on the training signal (where \(N_B\) is the number of segments in each mini-batch) is randomly generated to represent the beginning of each segment in the mini-batch. Each of the segments will slide starting from its point toward the end of the signal by 1 second shift with each iteration. If one of the segments reached the end of the signal, then training LSTM model in this epoch will stop and new epoch will start with new \(N_B\) points initialization. This method preserves the temporal dependencies between the signal samples and prevents the model from being biased toward one class.

LSTM model that is large enough may memorize the training signal especially if the training dataset is small that lead to overfitting and low medication state assessment on the testing data. Therefore, the internal states of LSTM after training on each mini-batch is passed over to the next training iteration with a carry-over probability [61], and they are zeroed out otherwise. The other method that is proposed in this thesis is applying the carry-over probability on both internal and hidden sates. These two methods are applied to prevent the LSTM from memorizing the training signals implicitly.

The LSTM cell proposed in [1] is used in this method without using peephole connections. Figure 3.5 shows the different components of many to one output LSTM architecture that is applied in this thesis. Each window of the gyroscope signals is first
linearly transformed to have a depth equal to the number of hidden states in LSTM layer. Second, the transformed signal is passed through LSTM network sample per sample. Finally, the hidden state for the last sample in each window is pass through a fully connected layer of size (\#units or hidden states, 2) and a soft-max layer to obtain a probability for each medication state (Classification Certainty ($P$)) that is used to predict the medication state (OFF or ON). The pseudo-code 1 shows these steps in more details. The weighs of the fully connected layers are initialized using random values from normal distribution of zero mean and 0.1 standard deviation, whereas the biases are initialized to the constant value (0.1).
Figure 3.5: Many to one architecture of LSTM. LSTM network is unfolded in time for forward computations of a window \((W_k)\) at kth second. Each sample that has depth of n channels is linearly transformed to have a depth of m hidden states. The hidden states from last layer after passing the last sample \((W_K(L))\) is passed through fully connected and soft-max layers to assess the medication state for each window. Vertical dotted-blue lines represent a drop-out on the hidden states between the layers. Horizontal continuous-blue and continuous-black lines represent the hidden and internal states that flow through time, respectively.
CHAPTER 3. PROPOSED METHODS

Pseudo-code 1 Training and testing LSTM network with \( m \) hidden states using \( n \)-channel gyroscope signal.

1: Initialize hyper-parameters to build and train LSTM network
2: Build the LSTM network
3: Load the dataset
4: \# Subject-based leave-one-out cross-validation
5: Divide the dataset into \( k \_fold \) \# \( k \)=number of patients
6: \textbf{for} \( Pn \leftarrow 1, k \_fold \textbf{do} \)
7: \# Training LSTM model on all folds except fold \#\( Pn \)
8: \textbf{for} \( Ei \leftarrow 1, \text{max number of epochs} \textbf{do} \)
9: Select \( N_B \) seconds randomly on the training signal
10: Create \( N_B \) segments that start from the \( N_B \) seconds
11: Initialize the hidden and internal states of LSTM network with zeros
12: \textbf{while} no segment reached the end of the signal \textbf{do} 
13: Construct a mini-batch of size \( N_B \) from the segments
14: \# training LSTM network using the mini-batch
15: \textbf{for} \( s \leftarrow 1, \text{last sample in the segments of mini-batch} \textbf{do} \)
16: Linearly transform the samples \( s \) to a depth of \( m \) instead of \( n \)
17: Perform forward propagation on LSTM using the samples \( s \)
18: Linearly transform the hidden states of last sample to a depth of \( 2 \)
19: Compute the loss using softmax cross entropy
20: Perform L2-norm regularization and find mean loss of the mini-batch
21: Perform back-propogation through time using an optimizer
22: \# Finding and saving the training loss and the model from epoch \( Ei \)
23: \textbf{for} \( g \leftarrow 1, \text{last segment in the signal} \textbf{do} \)
24: \textbf{for} \( s \leftarrow 1, \text{last sample in the segment} \ g \textbf{do} \)
25: Linearly transform the sample \( s \) to a depth of \( m \) instead of \( n \)
26: Perform forward propagation on LSTM using the sample \( s \)
27: Linearly transform the hidden states of last sample to a depth of \( 2 \)
28: Compute the loss of segment \( g \) using softmax cross entropy
29: Compute the average training loss
30: \# Selecting and testing LSTM model on fold \#\( Pn \) to find testing loss and
31: \# other performance metrics
32: Select the trained model with lowest training loss
33: \textbf{for} \( g \leftarrow 1, \text{last segment in the signal} \textbf{do} \)
34: \textbf{for} \( s \leftarrow 1, \text{last sample in the segment} \ g \textbf{do} \)
35: Linearly transform the sample \( s \) to a depth of \( m \) instead of \( n \)
36: Perform forward propagation on LSTM using the sample \( s \)
37: Linearly transform the hidden states of last sample to a depth of \( 2 \)
38: Pass them through a soft-max layer
39: Find the medication state for segment \( g \)
Chapter 4

Results and Discussions

This chapter starts by explaining the parameters that were used to evaluate the proposed methods. Next, it reports the evaluation results of the developed methods on the two datasets that included recordings of 19 PD patients. Two scenarios were considered: general training/classification and patient-specific where the former trains and tests the algorithm using subject-based leave-one-out cross-validation for each of the datasets, and the latter trains and tests the algorithm for each patient individually.

In patient-specific scenario, data that were recorded while patients were performing four activities in ON and OFF states was used to train a classifier for each patient, whereas, the rest of the data for the same patient were used for testing. These activities are ambulation, drinking, arm resting and dressing that are used in the routine clinical settings for rating dyskinesia severity in Unified Dyskinesia Rating Scale (UDysRS) and Core Assessment Program for Surgical Interventional Therapies in Parkinson’s disease (CAPSIT-PD). Hence, collecting the required data for training patient-specific classifier will be through the existing PD clinical settings and will not enforce additional burden to patients or clinicians. In addition, for patient-specific scenarios, a new method is proposed to select the number and placement of sensors for each patient.

The results of supervised classification using SVM and LSTM in the first scenario are presented first, and they are compared with the results of patient-specific supervised and semi-supervised approach which reported next. After that, the method with
the highest results is compared with other studies. Lastly, this chapter presents the results of feature analysis to show the advantage of using the directional information in X, Y, and Z axes vs. their magnitude.

4.1 Performance metrics

Three metrics were used to evaluate the trained classifiers to detect the two medication states which are accuracy, sensitivity and specificity of the OFF state. The accuracy is the percentage of correct medication state predictions that classify each second in the recorded signals. Sensitivity of the OFF state is the percentage of correct OFF medication state predictions that classify each second in the recorded data during the OFF state. Specificity of the OFF state is the percentage of correct ON state predictions that classify each second in the recorded data during the ON state.

4.2 Generally-trained Classifiers

4.2.1 Supervised Classification: SVM

Supervised classification method using SVM (refer to section 3.2) was trained and validated using subject-based leave-one-out cross-validation. Equal amount of data for each state for each patient selected in the training folds was used for training and all the data for each patient selected in the validation folds was used for testing. Using equal number of windows for each state prevents the SVM model from being bias toward one of the states. Training and validation were performed separately dataset 1 and 2. LibSVM library [63] was used to train and test the SVM classifier.

For dataset 1, if only one round of activities was recorded in the OFF state for a patient, then it was used with one round of ON state for training, else all the rounds were used for training for the patients selected in the training folds. This dataset contains twelve patients, so 12-fold cross-validation was used. The eleven folds that
were used for training each time had roughly 1.5 hours of recorded data that split equally between the two states. The fold that was used for testing in each time had about 15 minutes of recorded data.

For dataset 2, equal amount of recorded data that represents ON and OFF states are used for training from each patient selected in the training folds. This dataset contains seven patients, so 7-fold cross-validation was used. The six folds that were used for training each time had roughly 5.5 hours of recorded data that split equally between the two states. The fold that was used for testing in each time had about two hours.

The data recorded using each of the sensors and a combination of the sensors were used to train and validate the SVM classifier to experiment the effect of the sensors placement on detecting the ON and OFF state for each patient and for all of them. Three sensors from the first and second datasets were experimented which were (wrist, trunk and ankle) and (wrist, thigh and ankle), respectively. Table 4.1 and 4.2 shows the average testing accuracy, sensitivity and specificity for the first and second datasets, respectively. Medication state detection using data recorded using sensor mounted on the ankle outperformed the detection that was based on data recorded using wrist, trunk, or thigh sensors. Using two sensors increased the accuracy for both first and second dataset, and using the ankle with trunk or with thigh resulted in the highest accuracies which were 71.64 for the first dataset and 78.48 for the second dataset. Using three sensors together reduced the accuracy and specificity and increased the sensitivity.

As a result, using two sensors mounted on ankle and trunk or ankle and thigh had the highest accuracy in general. However, generalizing the same number and placement of sensors was not the case for all the patients. Figure 4.1 and 4.2 shows the results for each the patients for the first and second datasets, respectively. For instance in dataset 1 as shown in Figure 4.1, the highest accuracy that was obtained
Table 4.1: Average testing results of SVM model that was generally training using patient-based leave-one-out cross-validation (12 folds) on dataset 1. The results are the accuracy, sensitivity and specificity for multiple experiments using single or a combination of sensors.

<table>
<thead>
<tr>
<th>Sensors Used</th>
<th>Accuracy</th>
<th>Sensitivity</th>
<th>Specificity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wrist</td>
<td>65.65</td>
<td>72.26</td>
<td>63.41</td>
</tr>
<tr>
<td>Trunk</td>
<td>65.85</td>
<td>63.75</td>
<td>67.09</td>
</tr>
<tr>
<td>Ankle</td>
<td>67.37</td>
<td>66.9</td>
<td>68.17</td>
</tr>
<tr>
<td>Wrist and trunk</td>
<td>71.44</td>
<td>76.71</td>
<td>69.17</td>
</tr>
<tr>
<td>Wrist and ankle</td>
<td>68.67</td>
<td>72.59</td>
<td>67.86</td>
</tr>
<tr>
<td>Trunk and ankle</td>
<td>71.64</td>
<td>70.97</td>
<td>72.72</td>
</tr>
<tr>
<td>Wrist, trunk and ankle</td>
<td>71.59</td>
<td>77.79</td>
<td>68.74</td>
</tr>
</tbody>
</table>

Table 4.2: Average testing results of SVM model that was generally training using patient-based leave-one-out cross-validation (7 folds) on dataset 2. The results are the accuracy, sensitivity and specificity for multiple experiments using single or a combination of sensors.

<table>
<thead>
<tr>
<th>Sensors Used</th>
<th>Accuracy</th>
<th>Sensitivity</th>
<th>Specificity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wrist</td>
<td>64.94</td>
<td>62.69</td>
<td>70.64</td>
</tr>
<tr>
<td>Thigh</td>
<td>74.18</td>
<td>72.78</td>
<td>73.03</td>
</tr>
<tr>
<td>Ankle</td>
<td>77.27</td>
<td>75.99</td>
<td>72.66</td>
</tr>
<tr>
<td>Wrist and thigh</td>
<td>74.05</td>
<td>70.82</td>
<td>79.9</td>
</tr>
<tr>
<td>Wrist and ankle</td>
<td>77.22</td>
<td>75.34</td>
<td>79.45</td>
</tr>
<tr>
<td>Thigh and ankle</td>
<td>78.48</td>
<td>78.13</td>
<td>74.26</td>
</tr>
<tr>
<td>Wrist, thigh and ankle</td>
<td>77.61</td>
<td>86.61</td>
<td>48.92</td>
</tr>
</tbody>
</table>
CHAPTER 4. RESULTS AND DISCUSSIONS

for patient 1, 3 and 7 was for the wrist and trunk sensor, and for patient 2, 4, and 9 was for the ankle sensor. In addition, OFF sensitivity changed significantly if the sensor that was used to training the SVM model was changed as shown for patient 2, 4, 7 and 9 in 4.1 and patient 1, 4 and 5 in Figure 4.2. Therefore, there is a need to train a patient-specific classifier to cope with this variability between patients.

4.2.2 Supervised Classification: LSTM

Supervised classifier using LSTM network was trained and validated using subject-based leave-one-out cross-validation. Datasets were divided for training and testing data in the same way described before in the previous Section 4.2.1. TensorFlow [64] was used to implement, train and test the LSTM network proposed in this thesis. LSTM networks were trained for 300 epochs and the calculated training loss and trained weights were saved after each epoch. Because both datasets are relatively small, no validation set was used to select the best model for testing. Instead, the model with the lowest loss on the training data was selected and validated on the testing folds.

Table 4.3 contains the LSTM hyper-parameters that were examined. They were mainly explored on data recorded using the ankle sensor which was shown in the previous section to provide the highest accuracy if only one sensor was used. In addition, three types of optimizers are tested which are: adagrad [65], momentum [66], and adam [67]. The following parameters were shown to yield the highest accuracy with adam optimizer: 64 hidden states of two LSTM layers, truncated back-propagation length = one second, learning rate = 0.001, min-batch size = 128, out-keep probability = 0.5, carry-out probability = 1. The learning rate and out-keep probability had the highest effect on the LSTM performance. Using carry-out probability lower than 1 on the internal states didn’t help improving the results. In addition, applying carry-over probability on both internal and hidden sates reduced the oscillation in the
Figure 4.1: Testing results of SVM model that was generally training using patient-based leave-one-out cross-validation (12 folds) on dataset 1. The results are the accuracy, sensitivity and specificity for multiple experiments using single or a combination of sensors for each of the patients or the folds.
Figure 4.2: Testing results of SVM model that was generally training using patient-based leave-one-out cross-validation (7 folds) on dataset 2. The results are the accuracy, sensitivity and specificity for multiple experiments using single or a combination of sensors for each of the patients or the folds.
training and testing losses, but at the same time reduced the testing accuracy significantly. Therefore, using carry-over probability less than 1 prevented LSTM model from memorizing long sequences precisely, but at the same time the LSTM was less dependent on patterns flow more than one second in the signal. In other word, this method was forcing the model to learn from each window, so the internal and hidden states of LSTM from the previous window will have small or no effects. To check this behavior, the internal and hidden states was initialized to zero after each mini-batch training and the results were similar to using carry-over probability=0.5. The trained networks did not suffer from exploding gradient, so gradient clipping technique was not applied.

The results of LSTM network are shown in Table 4.4 using data recorded using only ankle sensor or both trunk and ankle in dataset 1. The accuracy of LSTM model using only ankle sensor outperformed SVM classifier accuracy for this sensor by 5%. However, the LSTM network was overfitting the training data very quickly using two sensors which made it poorly generalize to the testing data. The reason for overfitting was adding new channels to relatively small training data.

For the second dataset, the results of the LSTM network are shown in Table 4.5 using data recorded using only ankle sensor. Only ankle sensor was used because it was shown to have the highest performance and also using two sensors for small dataset

<table>
<thead>
<tr>
<th># of Layers</th>
<th>1, and 2</th>
<th># Units</th>
<th>32, 64, 128, and 256</th>
</tr>
</thead>
<tbody>
<tr>
<td>Truncated Back-propagation Length (sec)</td>
<td>1, and 5</td>
<td>Learning Rate</td>
<td>0.1, 0.025, 0.01, and 0.001</td>
</tr>
<tr>
<td>Batch Size</td>
<td>32, 64, 128, and 256</td>
<td>Out-keep Probability</td>
<td>1, 0.7, and 0.5</td>
</tr>
<tr>
<td>Carry-over Probability</td>
<td>1, 0.7, and 0.5</td>
<td># Sensor Used</td>
<td>Ankle, and (ankle and trunk)</td>
</tr>
</tbody>
</table>

Table 4.3: The LSTM Hyper-parameters that were explored in this thesis
Table 4.4: Average testing results of LSTM model that was generally training using patient-based leave-one-out cross-validation (12 folds) on dataset 1. The results are the accuracy, sensitivity and specificity using the ankle or a combination of trunk and ankle sensors.

<table>
<thead>
<tr>
<th>Patient #</th>
<th>Ankle</th>
<th>Trunk and ankle</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>89.14</td>
<td>88.77</td>
<td>89.30</td>
<td>89.95</td>
<td>81.81</td>
<td>93.48</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>82.34</td>
<td>62.2</td>
<td>89.24</td>
<td>68.99</td>
<td>61.04</td>
<td>71.71</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>60.57</td>
<td>67.56</td>
<td>58.21</td>
<td>46.65</td>
<td>94.59</td>
<td>30.47</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>93.85</td>
<td>93.98</td>
<td>93.81</td>
<td>88.85</td>
<td>93.44</td>
<td>87.23</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>80.15</td>
<td>86.52</td>
<td>78.01</td>
<td>69.06</td>
<td>59.06</td>
<td>72.42</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>75.33</td>
<td>24.55</td>
<td>92.36</td>
<td>42.85</td>
<td>76.64</td>
<td>31.52</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>75.33</td>
<td>56.52</td>
<td>93.9</td>
<td>74.62</td>
<td>70.65</td>
<td>78.67</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>54.14</td>
<td>29.91</td>
<td>78.51</td>
<td>50</td>
<td>28.49</td>
<td>71.63</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>80.79</td>
<td>51.47</td>
<td>90.71</td>
<td>69.26</td>
<td>88.72</td>
<td>62.68</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>61.94</td>
<td>71.12</td>
<td>58.59</td>
<td>62.94</td>
<td>87.7</td>
<td>53.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>56.53</td>
<td>80.58</td>
<td>30.76</td>
<td>64.51</td>
<td>93.35</td>
<td>33.61</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>64.46</td>
<td>47.11</td>
<td>84.85</td>
<td>62.78</td>
<td>49.33</td>
<td>78.59</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td><strong>72.86</strong></td>
<td><strong>63.36</strong></td>
<td><strong>78.19</strong></td>
<td><strong>65.87</strong></td>
<td><strong>73.73</strong></td>
<td><strong>63.83</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

will lead to overfitting. LSTM for dataset 2 yielded approximately the same accuracy and sensitivity for SVM, but it outperformed SVM specificity by about 10%.

To sum up, when only one sensor mounted on the ankle was used for general training and testing, the algorithm based on LSTM performed better than SVM with 74.91%, 69.42%, and 80.55% for accuracy, sensitivity, and specificity, respectively, whereas, SVM yielded 72.32% accuracy, 71.44% sensitivity, and 70.41% specificity.

### 4.3 Patient-specific Classifiers

Three methods which are supervised classification using SVM (refer to Section 3.2) and semi-supervised using k-means or SOTM (refer to Section 3.3 and 3.4) were trained for each patient specifically. Data that were recorded while patients were performing four activities in ON and OFF states were used to train a classifier for
CHAPTER 4. RESULTS AND DISCUSSIONS

Table 4.5: Average testing results of LSTM model that was generally training using patient-based leave-one-out cross-validation (7 folds) on dataset 2. The results are the accuracy, sensitivity and specificity using the ankle sensor.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>87.27</td>
<td>84.03</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>86.686</td>
<td>99.4</td>
<td>60.74</td>
</tr>
<tr>
<td>3</td>
<td>80.84</td>
<td>79.91</td>
<td>100</td>
</tr>
<tr>
<td>4</td>
<td>91.56</td>
<td>88.54</td>
<td>100</td>
</tr>
<tr>
<td>5</td>
<td>74.78</td>
<td>63.55</td>
<td>97.35</td>
</tr>
<tr>
<td>6</td>
<td>71.23</td>
<td>81.84</td>
<td>41.07</td>
</tr>
<tr>
<td>7</td>
<td>46.37</td>
<td>31.02</td>
<td>81.17</td>
</tr>
<tr>
<td>Average</td>
<td>76.96</td>
<td>75.47</td>
<td>82.9</td>
</tr>
</tbody>
</table>

each patient. These activities are ambulation, drinking, arm resting and dressing.

For dataset 1, the four activities were selected from two rounds, one for each medication state, and were used to train the classifier for each patient individually, whereas, the rest of the data from the same patient were used for testing. Total time of the recorded data that was used for training and testing for each patient was roughly four and ten minutes, respectively.

Despite the small amount of data used for training the patient-specific classifier (about 250 feature vectors), supervised classification using SVM obtained higher accuracy and sensitivity using single or a combination of sensors for each of the patients as shown in Figure 4.3. However, SVM resulted in classification specificity that is approximately equal to the results of other two methods, or a little bit lower especially if the trunk sensor was used. The second high performance was for the semi-supervised using SOTM that achieved a little bit higher accuracy than using k-means for the semi-supervised classification.

Figure 4.4 shows the results of patient-specific classification using SVM for each patient using single or a combination of sensors. SVM was used because it yielded the highest accuracy as compared before in Figure 4.3. It is clear that specific sensor
Figure 4.3: The average classification accuracy, sensitivity and specificity of the testing data using single or a combination of sensors for each of the patients after training patient-specific models on features extracted from X, Y, and Z signals. The classification models are semi-supervised using k-means or SOTM and supervised using SVM.
placement for each patient plays vital role in medication state prediction with high accuracy. For instance, ankle sensor for patient 12 yielded 96% accuracy while the second highest accuracy was 80% for the trunk sensor. For patient 10, the trunk gave the highest accuracy while for patient 5, a combination of the ankle and trunk had the highest accuracy.

Inspecting UPDRS score for each patient and comparing it with the results in Figure 4.4 showed that for patient with average change in UPDRS score between OFF and ON states more than 20, one sensor mounted on the limb with highest change in tremor score is sufficient to obtain high classification accuracy. An exception for this case is if the patient shows approximately equal rest tremor on the face, upper and lower extremities, then two sensors mounted on ankle and trunk or wrist (if trunk sensor was not used) are preferred. Examples of this case were patient 1 and 12, while patient 9 was an example for the exception. For patient with average change in UPDRS score between OFF and ON states more than 10 and lower than 20, two sensors mounted on upper body (trunk) and lower body (ankle) are sufficient to obtain high classification accuracy. Examples for this case were patient 4, 5, 7, 10 and 11. For other patients with change in UPDRS lower than 10, using three sensors are preferred to get consistent results and to prevent the model from being bias to one of the states.

Based on the correlation between the average change in UPDRS score between OFF and ON state for each patient and the presence of rest tremor, selecting the number of sensors and their placement for each patient was proposed as shown in Table 4.6. The low accuracy for patient 3, 6, 8 and 11 was because they had low benefit from their medication as it is clear from their change in UPDRS score which were 4.7, 6.7, 6.5 and 14.5, respectively. In addition, having dyskinesia during OFF state was another reason for low performance for Patients 6 and 8, whereas, patient 3 showed higher tremor in one of the ON rounds than OFF rounds. The data used
CHAPTER 4. RESULTS AND DISCUSSIONS

for training the model for patient 11 did not include tremor, whereas, this symptom presented in testing data and it was the main reason for misclassification for this patient. An example of the report generated by patient-specific SVM classifier is shown in Figure 4.5 for patient 12 using the data recorded using ankle sensor. This figure also shows the classification certainty for each second and it was low for the transition between the two states and for activity 5 and 8 that were misclassified in round 3.

The proposed method for selecting the number and placement of the sensors was validated on the second dataset. After annotating the activities that the patient were doing in dataset 2, the same four activities used before were selected for the two medication states and were used to train the classifier for each patient individually, whereas, the rest of the data from the same patient were used for testing. Each of the selected activities was about two minutes. Total time of the recorded data that was used for training and testing for each patient was roughly 15 and 105 minutes, respectively.

Table 4.7 shows the detection results for dataset 2 based on the proposed selection of the number of sensors and their placement for each patient. Because sensor mounted on trunk was not used in dataset 2, thus wrist sensor was used instead if required. Patients 1 and 4 showed average change in UPDRS score > 20 and relatively high wrist tremor, thus the sensor mounted on the wrist was used for them instead of ankle sensor. Patient 7 showed average change in UPDRS score > 20 with face, wrist and leg tremor, thus sensors mounted on wrist and ankle were used for this patient. For other patients, the same proposed method for dataset 1 was used. The highest accuracy was associated with patients who showed change in UPDRS > 20, and the lowest accuracy was for patients 2 and 6 who showed low improvement in their UPDRS score after medication in addition to having tremor score in ON state that was higher or equal to tremor score in the OFF state.
Figure 4.4: Testing results of patient-specific SVM model for each of the patient in dataset 1. The results are the accuracy, sensitivity and specificity for multiple experiments using single or a combination of sensors.
Table 4.6: Average testing results of patient-specific SVM model for dataset 1. The results are the accuracy, sensitivity and specificity using single or a combination of sensors based on the average change in UPDRS score between OFF and ON state for each patient and the presence of rest tremor.

<table>
<thead>
<tr>
<th>Patient #</th>
<th>Acc.</th>
<th>Sens.</th>
<th>Spec.</th>
<th>Sensor used</th>
<th>Average Change in UPDRS Score between OFF and ON</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>99.22</td>
<td>100</td>
<td>99</td>
<td>Ankle</td>
<td>22.7</td>
</tr>
<tr>
<td>2</td>
<td>63.26</td>
<td>87.78</td>
<td>58.73</td>
<td>Wrist, trunk and ankle</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>43</td>
<td>100</td>
<td>33.06</td>
<td>Wrist, trunk and ankle</td>
<td>4.7</td>
</tr>
<tr>
<td>4</td>
<td>98.64</td>
<td>100</td>
<td>98.39</td>
<td>Trunk and ankle</td>
<td>15</td>
</tr>
<tr>
<td>5</td>
<td>95.89</td>
<td>90.67</td>
<td>96.77</td>
<td>Trunk and ankle</td>
<td>12</td>
</tr>
<tr>
<td>6</td>
<td>41.18</td>
<td>80</td>
<td>34.02</td>
<td>Wrist, trunk and ankle</td>
<td>6.7</td>
</tr>
<tr>
<td>7</td>
<td>82.54</td>
<td>70.16</td>
<td>95.03</td>
<td>Trunk and ankle</td>
<td>15</td>
</tr>
<tr>
<td>8</td>
<td>64.38</td>
<td>51.17</td>
<td>77.76</td>
<td>Wrist, trunk and ankle</td>
<td>6.5</td>
</tr>
<tr>
<td>9</td>
<td>94.28</td>
<td>85.62</td>
<td>95.66</td>
<td>Trunk and ankle</td>
<td>24</td>
</tr>
<tr>
<td>10</td>
<td>85.47</td>
<td>40</td>
<td>93.85</td>
<td>Trunk and ankle</td>
<td>13.7</td>
</tr>
<tr>
<td>11</td>
<td>63.86</td>
<td>99.68</td>
<td>27.92</td>
<td>Trunk and ankle</td>
<td>14.5</td>
</tr>
<tr>
<td>12</td>
<td>95.45</td>
<td>91.28</td>
<td>100</td>
<td>Ankle</td>
<td>22</td>
</tr>
<tr>
<td>Average</td>
<td>77.26</td>
<td>83.03</td>
<td>75.84</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
50 seconds
Round 0, before medication
OFF
UPDRS=49, mAIMS=4.7
Round 1, 1 hour after medication
ON
UPDRS=30, mAIMS=7
Round 2, 2 hour after medication
ON with Dyskinesia
UPDRS=17, mAIMS=16
Round 3, 3 hour after medication
OFF
UPDRS=42, mAIMS=4

Medication State
OFF
ON
Classification
Certainty

Figure 4.5: The classification results (1 OFF, 2 ON) with the certainty (continuous red line between 0 and 1) for each round for patient 12 using patient-specific SVM trained and testing on data recorded using ankle sensor in dataset 1. The activities are 1=ambulation, 2=arms resting, 3=cutting, 4=dressing, 5=drinking, 6=unpacking groceries, 7=hair brush with left hand, and 8=hair brush with right hand.

To inspect if the selection of sensors was optimal for each patient in dataset 2, patient-specific SVM classifier was trained and tested on all possible combination of wrist, thigh and trunk sensors. The results for this experiment are shown in Figure 4.6. For all the patients, the selected sensors yielded the highest accuracy except for patients 3 and 5, but the difference was not significant.

4.4 Patient-specific vs. Generally-trained Classifiers

To compare the results of generally-trained SVM classifier with patient-specific SVM classifier, the highest results of using generally-trained SVM in Tables 4.1 and 4.2 are used which was based on using trunk and ankle sensors for dataset 1 and using thigh and ankle sensors for dataset 2. In addition, highest results of generally-trained LSTM network are used for the comparison. The comparisons are shown in Tables 4.8 and 4.9 for the first and second datasets, respectively.

Patient-specific SVM outperformed generally-trained SVM and LSTM by about 5% accuracy for both datasets. For the first dataset, the results of the three methods were correlated for most patient. However, for some patients, the results were signifi-
Table 4.7: Average testing results of patient-specific SVM model for dataset 2. The results are the accuracy, sensitivity and specificity using single or a combination of sensors based on the average change in UPDRS score between OFF and ON state for each patient and the presence of rest tremor.

<table>
<thead>
<tr>
<th>Patient #</th>
<th>Acc.</th>
<th>Sens.</th>
<th>Spec.</th>
<th>Sensor used</th>
<th>Average Change in UPDRS Score between OFF and ON</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>96.64</td>
<td>97.93</td>
<td>90.3</td>
<td>Wrist</td>
<td>23</td>
</tr>
<tr>
<td>2</td>
<td>64.37</td>
<td>61.52</td>
<td>71.31</td>
<td>Wrist, thigh and ankle</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>79.38</td>
<td>80.12</td>
<td>52.13</td>
<td>Wrist, thigh and ankle</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>99.43</td>
<td>99.26</td>
<td>100</td>
<td>Wrist</td>
<td>34</td>
</tr>
<tr>
<td>5</td>
<td>78.18</td>
<td>79.1</td>
<td>76.2</td>
<td>Wrist and ankle</td>
<td>13</td>
</tr>
<tr>
<td>6</td>
<td>63.94</td>
<td>59.02</td>
<td>81.58</td>
<td>Wrist and ankle</td>
<td>10</td>
</tr>
<tr>
<td>7</td>
<td>92.68</td>
<td>91.24</td>
<td>96.26</td>
<td>Wrist and ankle</td>
<td>22</td>
</tr>
<tr>
<td>Average</td>
<td>82.09</td>
<td>81.17</td>
<td>81.11</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure 4.6: Testing results of patient-specific SVM model for each of the patient in dataset 2. The results are the accuracy, sensitivity and specificity for multiple experiments using single or a combination of sensors.
CHAPTER 4. RESULTS AND DISCUSSIONS

Table 4.8: Average testing results of patient-specific SVM model vs. generally trained SVM and LSTM models for dataset 1. Training patient-specific SVM used single or a combination of sensors based on the average change in UPDRS score between OFF and ON state and the presence of rest tremor for each patient, whereas, training generally-trained SVM and LSTM used two sensor mounted on trunk and ankle, and only one sensor mounted on the ankle, respectively.

<table>
<thead>
<tr>
<th>Patient #</th>
<th>Patient-specific SVM</th>
<th>Generally-trained SVM</th>
<th>Generally-trained LSTM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>99.22</td>
<td>100</td>
<td>99</td>
</tr>
<tr>
<td>2</td>
<td>63.26</td>
<td>87.78</td>
<td>58.73</td>
</tr>
<tr>
<td>3</td>
<td>43</td>
<td>100</td>
<td>33.06</td>
</tr>
<tr>
<td>4</td>
<td>98.64</td>
<td>100</td>
<td>98.39</td>
</tr>
<tr>
<td>5</td>
<td>95.89</td>
<td>90.67</td>
<td>96.77</td>
</tr>
<tr>
<td>6</td>
<td>41.18</td>
<td>80</td>
<td>34.02</td>
</tr>
<tr>
<td>7</td>
<td>82.54</td>
<td>70.16</td>
<td>95.03</td>
</tr>
<tr>
<td>8</td>
<td>64.38</td>
<td>51.17</td>
<td>77.76</td>
</tr>
<tr>
<td>9</td>
<td>94.28</td>
<td>85.62</td>
<td>95.66</td>
</tr>
<tr>
<td>10</td>
<td>85.47</td>
<td>40</td>
<td>93.85</td>
</tr>
<tr>
<td>11</td>
<td>63.86</td>
<td>99.68</td>
<td>27.92</td>
</tr>
<tr>
<td>12</td>
<td>95.45</td>
<td>91.28</td>
<td>100</td>
</tr>
<tr>
<td>Average</td>
<td>77.26</td>
<td>83.03</td>
<td>75.84</td>
</tr>
</tbody>
</table>

cant using specific method. The significant results were for patient 2, 9, 10 and 12 in dataset 1 (as shown in bold font in Table 4.8). Patient-specific SVM yielded very high accuracy that is more than 90% for patient 12, whereas generally-trained SVM and LSTM classifiers poorly performed. For patient 2, LSTM network yielded the highest accuracy. For patient 9, patient specific SVM and LSTM performed much better than generally-trained SVM. For patient 10, SVM performed better than LSTM.

For the second dataset, the same correlated pattern between the methods results occurred. However, LSTM performed very well for patient 2, but it performed poorly on patient 7. In addition, for patients in dataset 2 with average change in UPDRS > 20, generally-trained SVM classifier did not exceed 89% accuracy whereas Patient-specific SVM yielded accuracy > 92% (as shown in bold font in Table 4.9). The
Table 4.9: Average testing results of patient-specific SVM model vs. generally trained SVM and LSTM models for dataset 2. Training patient-specific SVM used single or a combination of sensors based on the average change in UPDRS score between OFF and ON state and the presence of rest tremor for each patient, whereas, training generally-trained SVM and LSTM used two sensor mounted on thigh and ankle, and only one sensor mounted on the ankle, respectively.

<table>
<thead>
<tr>
<th>Patient #</th>
<th>Patient-specific SVM</th>
<th>Generally-trained SVM</th>
<th>Generally-trained LSTM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>96.64</td>
<td>97.93</td>
<td>90.3</td>
</tr>
<tr>
<td>2</td>
<td>64.37</td>
<td>61.52</td>
<td>71.31</td>
</tr>
<tr>
<td>3</td>
<td>79.38</td>
<td>80.12</td>
<td>52.13</td>
</tr>
<tr>
<td>4</td>
<td>99.43</td>
<td>99.26</td>
<td>100</td>
</tr>
<tr>
<td>5</td>
<td>78.18</td>
<td>79.1</td>
<td>76.2</td>
</tr>
<tr>
<td>6</td>
<td>63.94</td>
<td>59.02</td>
<td>81.58</td>
</tr>
<tr>
<td>7</td>
<td>92.68</td>
<td>91.24</td>
<td>96.26</td>
</tr>
<tr>
<td>Average</td>
<td>82.09</td>
<td>81.17</td>
<td>81.11</td>
</tr>
</tbody>
</table>

The reason behind poor performance of generally-trained SVM for some of the patients is the significant variability between PD subjects as mentioned before in the introduction of this thesis and as will be explained at the end of this section.

LSTM network for both datasets yielded the highest specificity of the OFF state. In addition, it outperformed patient-specific and generally-trained SVM for some patients using only one sensor. This shows its adaptation ability to new patients to cope with the variability between patients if more data were recorded for more patients and used for training.

Feature analysis was performed on the extracted features from X, Y and Z signals for all the patients in dataset 1 to illustrate the variability between patients on feature selection. These signals were used before to train the patient-specific classifiers (refer to Section 4.3). Feature selection was performed based on the statistical analysis proposed in Section 3.2.3.1 on the features extracted from the three axes of sensors mounted on wrist, trunk and ankle. Significant features were the features with p-value
from the statistical analysis < 0.05. To inspect the effect of the variability between patients on feature selection, the percentage of occurrence of the significant features for at least one of the three axes of each sensor to the total number of patients was determined as shown in Figure 4.7. The features were not selected uniformly for all the patient, for instance, features number 1, 11, 16, 18, 23, 24 and 25 were selected for one of the sensors of less than 70% of the patients. Therefore, nonuniform occurrence of the significant features is an indication of the variability between patients. In addition, selecting the features depended on the sensor location, for instance, feature 1, 2 16 had highest occurrence for wrist sensor, whereas, feature 23, 24 and 25 had highest occurrence for ankle sensor.

Table 4.10 shows the number of selected features for each patient and for each sensor location. Number of significant features was about 50 from the 69 features extracted for each sensor. However, about 20 features were selected for patient 2, 11 and 12 using the ankle sensor, and 5 and 26 features were selected for patient 6 and 10, respectively, using the wrist sensor.

Furthermore, Significant feature as proposed by [28] was selected to show the variability between patients in respect to the severity of PD symptoms. This feature was the percentage of the powers for frequencies > 4 Hz. Figure 4.8 shows the box plots of the log of the feature for three patients and two states. For each patient the median of the two state are separable, but the median of ON and OFF states for two patients can overlap such as the ON of patient 12 that overlap with OFF state of patient 1 and 4.

4.5 Comparison to Other Studies

As mentioned in the literature review, many studies were published to discriminate between OFF and ON medication states. Summary of these methods is shown in Table 4.11. This section compares between the results of the published methods
Figure 4.7: The percentage of occurrence of the significant features for at least one of the three axes of sensors mounted on wrist, trunk and ankle to the total number of patients in dataset 1 which is twelve. The features are 1-average jerk, 2-Peak-to-peak, 3-Signal power 1-4 Hz, 4-Signal power 4-6 Hz, 5-Signal power 0.5-15 Hz, 6-Percentage of the powers for frequencies > 4Hz, 7-Temporal Shannon entropy, 8-Standard deviation, 9-The number of autocorrelation peaks, 10-The sum of autocorrelation peaks, 11-First autocorrelation peak, 12-Lag of the first autocorrelation peak, 13-Gini Index, 14-Sample entropy, 15-Mean 16-Skewness, 17-Kurtosis, 18-Spectral entropy, 19-The peak in the power spectral density, 20-Dominant frequency, 21-The second peak in the power spectral density, 22-Secondary frequency, 23-Cross-correlation between X and Y axes, 24-Cross-correlation between X and Z axes, and 25-Cross-correlation between Y and Z axes.
Table 4.10: The number of the selected features for each patient and for each sensor location. Selected features have p-value < 5% significant level as a result of the statistical analysis on the training data for each patient separately, and all the patients (shown in last row). Number of selected features in bold represents relatively low number of significant feature for that location.

<table>
<thead>
<tr>
<th>Patient #</th>
<th># Selected Features for Each Location</th>
<th>Total # Selected Features</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Wrist</td>
<td>Trunk</td>
</tr>
<tr>
<td>1</td>
<td>52</td>
<td>58</td>
</tr>
<tr>
<td>2</td>
<td>61</td>
<td>55</td>
</tr>
<tr>
<td>3</td>
<td>51</td>
<td>46</td>
</tr>
<tr>
<td>4</td>
<td>51</td>
<td>55</td>
</tr>
<tr>
<td>5</td>
<td>41</td>
<td>48</td>
</tr>
<tr>
<td>6</td>
<td><strong>5</strong></td>
<td>50</td>
</tr>
<tr>
<td>7</td>
<td>53</td>
<td>64</td>
</tr>
<tr>
<td>8</td>
<td>25</td>
<td>28</td>
</tr>
<tr>
<td>9</td>
<td>47</td>
<td>30</td>
</tr>
<tr>
<td>10</td>
<td><strong>26</strong></td>
<td>54</td>
</tr>
<tr>
<td>11</td>
<td>56</td>
<td>57</td>
</tr>
<tr>
<td>12</td>
<td>60</td>
<td>61</td>
</tr>
<tr>
<td>All Patients</td>
<td>59</td>
<td>57</td>
</tr>
</tbody>
</table>
Figure 4.8: The box plots of the log of the feature (percentage of the powers for frequencies $>4$ Hz) for three patients and two states. The training data used to extract the feature is four activities (ambulation, drinking, arm resting and dressing) from two rounds, one for each MS. The dotted red rectangle shows the overlap between the medians of OFF state for patient 4 and ON state of patient 12.
that are activity-independent with the proposed patient-specific classification method using SVM that is also activity-independent. The average accuracy, sensitivity and specificity for the this method were 77.26, 83.03, and 75.84, respectively, for the first dataset, and 82.09, 81.17 and 81.11 for the second dataset.

Hoff et al. proposed activity independent method based on 7 accelerometers mounted on upper and lower externalities [27]. They validated their method 24 hours data for 15 patients and they obtained sensitivity (60%-71%) and specificity (66%-76%) which are lower than the results in this thesis.

Salarian et al. published the results of medication state detection method that was based on five accelerometer and gyroscope sensors and was trained and testing using subject-based leave-one-out cross-validation on three to six hours recordings for 13 patients [9]. The minimum average change in UPDRS score was 16. They yielded Sensitivity 90.1% and specificity 76.3%. To make a fair comparison, the results of patients with average change in UPDRS score > 15 for both datasets (8 patients) were determined which are accuracy 94.86%, sensitivity 91.94% and specificity 96.83%. This result shows the correlation between the UPDRS score and good prediction and it is higher than the results of Salarian et al. study.

Khan et al. suggested using a method based on SVM with RBF kernel to detect hand and leg tremor, dyskinesias and ON state with dyskinesia [17]. They used accelerometer sensor mounted on the waist to train the classifier using simulated data and to test on 12 PD patients with mid to late stage disease. They reported 72% accuracy which is lower than the overall classification accuracy in this thesis for dataset 1 and 2.

Hammerla et al. proposed using two sequences of Restricted Boltzmann Machines to detect if the PD patient is asleep, in ON or OFF state or having troublesome dyskinesia based on two accelerometers worn on each wrist [31]. They trained and validated their model on overall 32 PD patients who were performing different activ-
Table 4.11: Summary of the approaches in the published literature proposed to classify the medication states, in addition to summary of patient-specific classification using SVM proposed in this thesis.

<table>
<thead>
<tr>
<th>References</th>
<th># Sensors</th>
<th># Patients</th>
<th>Data Duration for Each Patient</th>
<th>Classification Method</th>
<th>Activity-dependent Model (Yes, No)</th>
<th>Patient-Specific (Yes, No, Partial)</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hoff et al. [27]</td>
<td>7 uni-axial accelerometers</td>
<td>15</td>
<td>24 hours</td>
<td>Linear discriminant</td>
<td>No</td>
<td>Yes</td>
<td>Sens.: 60%-71% Spec.: 60%-76%</td>
</tr>
<tr>
<td>Keijser et al. [28]</td>
<td>One tri-axial accelerometer</td>
<td>23</td>
<td>About 3 hours</td>
<td>Linear discriminant and ANN</td>
<td>Yes</td>
<td>Partial (Tremor, and non-tremor groups)</td>
<td>Sens.: 97% Spec.: 97%</td>
</tr>
<tr>
<td>Salarian et al. [9]</td>
<td>5 sensors (gyroscopes and accelerometers)</td>
<td>11</td>
<td>Three to six hours</td>
<td>Logistic regression</td>
<td>No</td>
<td>No</td>
<td>Sens.: 90.1% Spec.: 76.3%</td>
</tr>
<tr>
<td>Sama et al. [24]</td>
<td>One tri-axial accelerometer</td>
<td>20</td>
<td>About 1 hour</td>
<td>SVM and linear discriminant</td>
<td>Yes (gait)</td>
<td>No</td>
<td>Sens.: 84% Spec.: 90% Acc.: 94%</td>
</tr>
<tr>
<td>Khan et al. [17]</td>
<td>One tri-axial accelerometer</td>
<td>12</td>
<td>About 1 hour</td>
<td>SVM</td>
<td>No</td>
<td>No</td>
<td>Acc.: 72%</td>
</tr>
<tr>
<td>Perez-Lopez et al. [29]</td>
<td>One tri-axial accelerometer</td>
<td>7</td>
<td>About 6 hours</td>
<td>Linear discriminant</td>
<td>Yes (walking and not walking for bradykinesia and dyskinesia detection ,respectively)</td>
<td>Partial (Threshold on Bradykinesia )</td>
<td>Sens.: 99.9% Spec.: 99.9%</td>
</tr>
<tr>
<td>Rodriguez-Molinero et al. [30]</td>
<td>One tri-axial accelerometer</td>
<td>35</td>
<td>About 1.4 - 5.5 hours</td>
<td>Linear discriminant</td>
<td>Yes (walking)</td>
<td>Yes</td>
<td>Sens.: 96% Spec.: 94%</td>
</tr>
<tr>
<td>Hammerla et al. [31]</td>
<td>Two tri-axial accelerometer</td>
<td>32</td>
<td>4 hours (in lab) 1 week (in home)</td>
<td>Restricted Boltzmann Machines</td>
<td>No</td>
<td>No</td>
<td>In home, mean fl-score: 60% In lab, mean fl-score: 76%</td>
</tr>
<tr>
<td>Fisher et al. [32]</td>
<td>Two tri-axial accelerometer</td>
<td>32</td>
<td>4 hours (in lab) 1 week (in home)</td>
<td>ANN</td>
<td>No</td>
<td>No</td>
<td>In home, Sens.: 50% Spec.: 84% In lab, Sens.: 60% Spec.: 83%</td>
</tr>
<tr>
<td>Ramji, Hsayeni et al. [34,68]</td>
<td>Three tri-axial gyroscopes</td>
<td>18</td>
<td>Dataset 1: 15 minutes Dataset 2: about 2 hours</td>
<td>Semi-supervised using k-means and supervised using SVM</td>
<td>No</td>
<td>Yes</td>
<td>Acc.: 80%</td>
</tr>
<tr>
<td>Proposed</td>
<td>1, 2, or 3 tri-axial gyro. based on change in UPDRS for each patient</td>
<td>19</td>
<td>Dataset 1: 15 minutes Dataset 2: about 2 hours</td>
<td>SVM with fuzzy classification</td>
<td>No</td>
<td>Yes</td>
<td>Acc.: 80% Spec.: 82% Spec.: 78%</td>
</tr>
</tbody>
</table>
ities for about 4 hours in the lab and for 1 week in the home. They reported peak mean f1-score 0.76 for 7-fold cross-validation on lab data, and poor generalization to home data. Using stratified 7-fold cross-validation instead of using subject-based cross-validation might lead to high validation accuracy and low testing accuracy. In this thesis, mean f1-score was calculated for each of the patients in dataset 2 using patient-specific method, and it was (0.94, 0.62, 0.5, 0.99, 0.76, and 0.6) with a peak of 0.99.

Fisher et al. reported the results of using ANN to detect if the PD patient is asleep, in ON or OFF state or having troublesome dyskinesia. The same data and validation procedure were used in the study of Hammerla et al. [31] were used in Fisher et al. study. Average 55% sensitivity of OFF and specificity of OFF 82% on lab and home data were obtained. In this thesis average sensitivity and specificity of OFF state for both dataset were 82.1 and 78.48, respectively.

In recent study, Vegnish proposed patient-specific semi-supervised classification using k-means that was based on tensor decomposition of multi-channels of three gyroscope sensors. Dataset 1 and 2 used in this thesis were used in [34]. He reported average accuracy of 80% for both datasets which is approximately equal to 79.68% accuracy in this thesis. The advantage of this thesis is using lower number of sensors for the majority of patients.

### 4.6 Directional Information vs. Magnitude

Analysis of the directional information was performed to check the importance of using X, Y and Z for each sensor instead of using their magnitudes. Training data for the patient-specific classifiers (refer to Section 4.3) was used in this analysis. First, feature selection was performed based on the statistical analysis proposed in Section 3.2.3.1 on the features extracted from X, Y and Z axes or from the magnitude of these axes for each patient and for the sensors mounted on wrist, trunk, and ankle.
After finding the number of significant features, it was found that at least 5 features were significant if they were extracted from X, Y or Z signals for each of the sensors and not significant if they were extracted from the axes magnitude as shown in Table 4.12.

Additional experiment was performed to inspect the effect of extracting significant features from the axes and not from their magnitudes. In this experiment, patient-specific SVM classifier (refer to Section 4.3) was trained and testing separately on significant features extracted from X, Y and Z and from their magnitudes. Figure 4.9 shows the results of this experiment for all the permutations of wrist, trunk and ankle sensors. The average classification sensitivity of the OFF state after training patient-specific SVM model on features extracted from X, Y and Z signals was higher than the sensitivity of training and testing on features extracted from the magnitude of the axes. The only cases where the sensitivity of using the magnitude was higher were using the signals of the ankle sensor or trunk and ankle sensors. The same case for OFF specificity, using the axes individually instead of their magnitude led to higher specificity, except the case where the magnitude of the axes’ signals from wrist or wrist and trunk sensors was used.

Inspecting the accuracy in Figure 4.9 indicates that using X, Y, and Z signals resulted in higher or approximately equal accuracy than using their magnitude for all the sensor combinations, except the case of using only the wrist sensor. As it was mentioned before, higher specificity was obtained using the magnitude and because the time for the ON state was higher than the time for OFF state in the testing data, thus the accuracy was higher despite that the sensitivity using the three axes was higher than using the magnitude. To sum up, maintaining the directional information by extracted features from X, Y and Z directly resulted in better overall sensitivity and specificity.
Table 4.12: The number of significant features that are selected using the statistical analysis on features extracted from X, Y and Z axes or from the magnitude of these axes for each patient and for the sensors mounted on wrist, trunk, and ankle. This table also shows number of features that were significant if they were extracted from X, Y or Z signals for each of the sensors and not significant if they were extracted from their magnitude, and vice versa.

<table>
<thead>
<tr>
<th>Patient #</th>
<th># significant features using x, y and z axes</th>
<th># significant features using x, y or z and not using the magnitude</th>
<th># significant features using the magnitude of the 3 axes</th>
<th># significant features using the magnitude and not using x, y or z</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>177</td>
<td>6</td>
<td>67</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>138</td>
<td>12</td>
<td>22</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>149</td>
<td>6</td>
<td>52</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>175</td>
<td>8</td>
<td>69</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>151</td>
<td>7</td>
<td>62</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>100</td>
<td>13</td>
<td>45</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>180</td>
<td>5</td>
<td>63</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>80</td>
<td>19</td>
<td>27</td>
<td>6</td>
</tr>
<tr>
<td>9</td>
<td>120</td>
<td>21</td>
<td>43</td>
<td>2</td>
</tr>
<tr>
<td>10</td>
<td>118</td>
<td>15</td>
<td>38</td>
<td>2</td>
</tr>
<tr>
<td>11</td>
<td>127</td>
<td>13</td>
<td>14</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>142</td>
<td>16</td>
<td>21</td>
<td>2</td>
</tr>
</tbody>
</table>
Figure 4.9: The average classification results of the testing data after training patient-specific SVM model on features extracted from X, Y, and Z signals vs. features extracted from their magnitude signal. The results of using single or multiple sensors are presented.
Patients with mid-stage and advanced Parkinson’s disease are facing motor fluctuations that significantly affect their way of living and thus they are a major focus of clinical management. In this thesis, four supervised and semi-supervised classification approaches are developed to automatically assess the ON and OFF medication states using wearable sensors while PD patients are engaging in a variety of daily living activities. These methods are supervised learning using SVM with fuzzy classification, semi-supervised learning using k-means or using SOTM with fuzzy classification, and supervised classification using LSTM as a deep learning method.

Multiple temporal and spectral features that are relevant to PD symptoms are extracted from the three axes of gyroscope sensors. After performing dimensionality reduction, the features are passed through to the first three methods, whereas the signals of gyroscope’s axes are used to train and test LSTM networks without applying prior feature extraction. The developed methods were evaluated on two datasets that included recordings of 19 PD patients. Two scenarios were considered: general training/classification and patient-specific where the former trains and tests the algorithm using subject-based leave-one-out cross-validation, and the latter trains and tests the algorithm for each patient individually. Based on the reported results, the following were concluded:

- The results of generally-trained SVM showed that in general using two sensors mounted on ankle and trunk or ankle and thigh had the highest accuracy.
However, inspecting the results of using combinations of gyroscopes mounted on wrist, trunk, thigh, or ankle showed that one sensor or different sensor placement led to higher accuracy for specific patients. Therefore, there is a need for an algorithm that adapts with this variability between patients.

- The results of generally-trained LSTM showed LSTM generalization and adaptation ability using only one sensor mounted on ankle. It obtained significant accuracy for some of the patients for whom generally-trained SVM was not able to classify their medication states correctly.

- The algorithm based on LSTM performed better than SVM with overall 74.91%, 69.42%, and 80.55% for accuracy, sensitivity, and specificity, respectively, when only one sensor mounted on the ankle was used in the general training scenario. In addition, LSTM network for both datasets yielded the highest specificity of the OFF state. Those promising results show the potential outcome of developing deep learning methods in this field.

- Patient-specific SVM outperformed the proposed semi-supervised method using k-means or SOTM despite the small amount of data used for training. The second high performance in patient-specific scenario was for semi-supervised method using SOTM.

- Number of sensors and their placement for each patient that associated with the highest accuracy in patient-specific scenario were correlated with the average change in UPDRS score between OFF and ON state for each patient and the presence of rest tremor. Therefore, a method for selected the number and placement of gyroscopes were proposed and integrated with the patient-specific algorithm using SVM.

- In the comparison between generally-trained SVM and LSTM with
patient-specific SVM classifier, Patient-specific SVM outperformed generally-trained SVM and LSTM by about 5% accuracy for both datasets.

- Performing feature analysis showed the variability between PD patients and also showed retaining the directional information by using X, Y, and Z instead of their magnitude is beneficial for medication state classification.

To sum up, activity-independent classification using patient-specific SVM classifier was selected as the winning method. The selection of the number of sensors and their placement on patient’s body depended on the average change in UPDRS score between ON and OFF medication states and the presence of tremor for that patient. Average accuracy, sensitivity and specificity of OFF state for both datasets were 80%, 82% and 78% using the proposed patient-specific SVM classifier. For group of patients who had change in their UPDRS score between the two states more than 15, classification results were very high which were accuracy 94.86%, sensitivity 91.94% and specificity 96.83%. Comparing the proposed approach with other studies showed it had either the highest performance or equal performance with the advantage of using lower number of sensors. These results are promising and thus this algorithm can be potentially used in routine clinical practice to improve the quality of this group of PD patients.

5.1 Future Work

The limitation of the patient-specific algorithm is the need for prior knowledge about each patient and training data, thus there is a requirement for developing algorithm that can generally adapt to the X, Y, and Z data of gyroscope or accelerometer sensors mounted on the limb that is most affected by PD symptoms. This means training a model that can predict medication states from signals regardless of their sensors placement. In other words, train a model that is independent on patient’s
activities and sensor’s number or placement. LSTM performance on these small datasets shows its ability for adaptation and generalization. Therefore, our future work includes developing generally-trained LSTM model that can predict medication states with significant accuracy.
Bibliography


