NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

SCHOOL OF SCIENCE
DEPARTMENT OF INFORMATICS AND TELECOMMUNICATION

COMPUTER NETWORKING

BSc THESIS

Machine Learning-driven EEG Analysis towards brain-controlled vehicle.

Georgios K. Halios
Theodora L. Panagea

Supervisors: **Athanasia Alonistioti,** Associate Professor

ATHENS

September 2022

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΔΙΚΤΥΩΣΗ ΥΠΟΛΟΓΙΣΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ανάλυση εγκεφαλογραφημάτων με χρήση μηχανικής μάθησης με σκοπό τον έλεγχο οχήματος.

Γεώργιος Κ. Χαλιός
Θεοδώρα Λ. Παναγέα

Επιβλέποντες: **Αθανασία Αλωνιστιώτη,** Αναπληρώτρια Καθηγήτρια

ΑΘΗΝΑ
ΣΕΠΤΕΜΒΡΙΟΣ 2022

BSc THESIS

Machine Learning-driven EEG Analysis towards brain-controlled vehicle.

Georgios K. Halios
S.N: 1115201500173
Theodora L. Panagea
S.N: 1115201400135

Supervisors **Athanasia Alonistioti,** Associate Professor

September 2022

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ανάλυση εγκεφαλογραφημάτων με χρήση μηχανικής μάθησης με σκοπό τον έλεγχο οχήματος.

Γεώργιος Κ. Χαλιός
S.N: 1115201500173
Θεοδώρα Λ. Παναγέα
S.N: 1115201400135

ΕΠΙΒΛΕΠΟΝΤΕΣ: **Αθανασία Αλωνιστιώτη,** Αναπληρώτρια Καθηγήτρια

Σεπτέμβριος 2022

ABSTRACT

Due to the rapid development of technology, the Human Brain and Computers are interfered with by Bio-Electronic devices employing bio-signals, which are detected by a particular class of sensors called bio-sensors. A new emerging research, the study of bio-signals has focused particularly on mind-controlled technology. More specifically, directly controlling a vehicle using brain waves might assist people with impairments regain their driving abilities as well as offer a fresh option for healthy people to operate a vehicle. The current thesis describes a Brain Controlled Vehicle (BCV) that uses Brain Computer Interface (BCI) technology to interpret Electroencephalography (EEG) data, operate a device, and evaluate brain waves, in order to stay as close as possible to the human nature. The system, which is based on Machine Learning techniques, comprises the following features: (a) Processing of EEG data in order to perform various feature extraction methods; (b) make use of a proper dimensionality reduction method that will find correlations in the data and discard non-critical information; (c) implement classification methods that are able to predict the desired motion related labels (left hand, right hand, both feet, tongue); (d) map the predicted motion related labels into real motions (turn left, turn right, accelerate, slow down) and (e) integrate the best models, with the use of a voting method, into a final BCV system.

SUBJECT AREA: Machine Learning

KEYWORDS: Brain Controlled Vehicle (BCV), Electroencephalography (EEG), Classification Algorithms, Feature Extraction Methods

ΠΕΡΙΛΗΨΗ

Με τη ραγδαία ανάπτυξη της τεχνολογίας, ο ανθρώπινος εγκέφαλος και οι υπολογιστές μπορούν να συνεργαστούν με τη βοήθεια βιοηλεκτρονικών συσκευών που χρησιμοποιούν βιο-σήματα, τα οποία ανιχνεύονται από μια συγκεκριμένη κατηγορία αισθητήρων που ονομάζονται βιο-αισθητήρες. Ένας νέος τομέας έρευνας που σχετίζεται με τη μελέτη των βιο-σημάτων έχει επικεντρωθεί ιδιαίτερα στην τεχνολογία ελεγχόμενη από το μυαλό. Πιο συγκεκριμένα, ο άμεσος έλεγχος ενός οχήματος με χρήση εγκεφαλικών κυμάτων μπορεί να βοηθήσει τα άτομα με αναπηρίες να ανακτήσουν τις οδηγικές τους ικανότητες, καθώς και να προσφέρει μια νέα επιλογή για υγιή άτομα να χειριστούν ένα όχημα. Η παρούσα πτυχιακή εργασία περιγράφει ένα όχημα ελεγχόμενο με το μυαλό (BCV) που χρησιμοποιεί την τεχνολογία Brain Computer Interface (BCI) για να ερμηνεύσει δεδομένα Ηλεκτροεγκεφαλογραφίας (EEG), να χειριστεί μια συσκευή και να αξιολογήσει τα εγκεφαλικά κύματα, προκειμένου να παραμείνει όσο το δυνατόν πιο κοντά στην ανθρώπινη φύση. Το σύστημα, το οποίο βασίζεται σε τεχνικές Μηχανικής Μάθησης, περιλαμβάνει τα ακόλουθα χαρακτηριστικά: (α) Επεξεργασία δεδομένων EEG για την ανάπτυξη διαφόρων μεθόδων εξαγωγής χαρακτηριστικών (β) χρήση κατάλληλων μηχανισμών μείωσης των διαστάσεων των δεδομένων, οι οποίοι στοχεύουν στην εύρεση συσχετισμών στα δεδομένα με σκοπό την απομάκρυνση μη κρίσιμων πληροφοριών, (γ) εφαρμογή μεθόδων ταξινόμησης που είναι σε θέση να προβλέψουν τις επιθυμητές ετικέτες που σχετίζονται με την κίνηση (αριστερό χέρι, δεξί χέρι, και τα δύο πόδια, γλώσσα), (δ) αντιστοίχηση των προβλεπόμενων σχετικά με την οδήγηση ετικετών σε πραγματικές κινήσεις (στροφή αριστερά, στροφή δεξιά, αύξηση ταχύτητας, μείωση ταχύτητας) και (ε) ενσωμάτωση των καλύτερων μοντέλων, με τη χρήση της μεθόδου ψηφοφορίας, σε ένα τελικό σύστημα BCV.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Μηχανική Μάθηση

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Όχημα ελεγχόμενο από τον εγκέφαλο, Ηλεκτροεγκεφαλογράφημα (EEG), Αλγόριθμοι κατηγοριοποίησης, Μέθοδοι εξαγωγής χαρακτηριστικών

This page intentionally left blank

ACKNOWLEDGEMENTS

We would like to deeply thank our supervisor prof. Nancy Alonistioti and especially Lina Magoula and Nikolaos Koursioumpas, for the continuous support during the development of this thesis. Their contribution was really valuable for us in order to get results of better quality.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

PREFACE

The current thesis has been conducted for the bachelor’s program degree offered by the department of Informatics and Telecommunication from the National and Kapodistrian University of Athens. The main study of this thesis concerns the development of a brain – controlled vehicle which is based on EEG analysis using machine learning techniques. In the context of the present work, the proposed system has been implemented using Jupyter Notebook along with Python for the related algorithms and methods, as well as for the visualization of the experimental results. The choice of this topic is due to our interest in the field of Machine Learning and its numerous applications.

1. INTRODUCTION

Starting this thesis, it is essential for our readers to fully understand the main problem we are trying to mitigate through this research. Driverless or self-driving cars are on rage, trying to help the disabled or paralyzed people regain their mobility. Brain Controlled Vehicles (BCV) are a new addition to this technological outburst, but unfortunately it is still on experimental stage. The big problem with this is that the only way to understand drivers' attitude is through either gathering and analyzing a large amount of data about different driving situations a driver might face, or questionnaires in which a driver is asked to answer specific questions about their driving skills and their possible reactions during a possible emergency. Collecting and analyzing large volumes of data is usually very effective as a process of discovering a driver's attitude on the street. However, it remains time consuming and requires a large amount of computing resources when one considers the huge number of the current drivers. Additionally, it is rare for users to be willing to spend time on such surveys and procedures.

Naturally, many efforts and research have been made to provide new and innovative solutions to these problems. This work is one of those attempts to alleviate these problems.

The solution to the aforementioned problems is given by the drivers' brain. The brain, in conjunction with their reflexes during driving, evokes one or more desired actions. As a result, brain waves can be used as a dominant axis in our proposed Brain – Controlled Vechicle (BCV) system. The most basic and powerful motions are associated with the directional moving and speed increments. Therefore, they will also be considered and used in this work. In order to do so, data concerning brain activity or otherwise Electroencephalography (EEG) data should be collected, analyzed and processed using Machine Learning techniques and methods which are listed in the following chapters.

1.1 High Level Architecture

At this point, it is considered prudent to make a high level presentation of the architecture that this work will follow. All the components of the architecture are listed below and are discussed in more detail in *Chapter 4*.

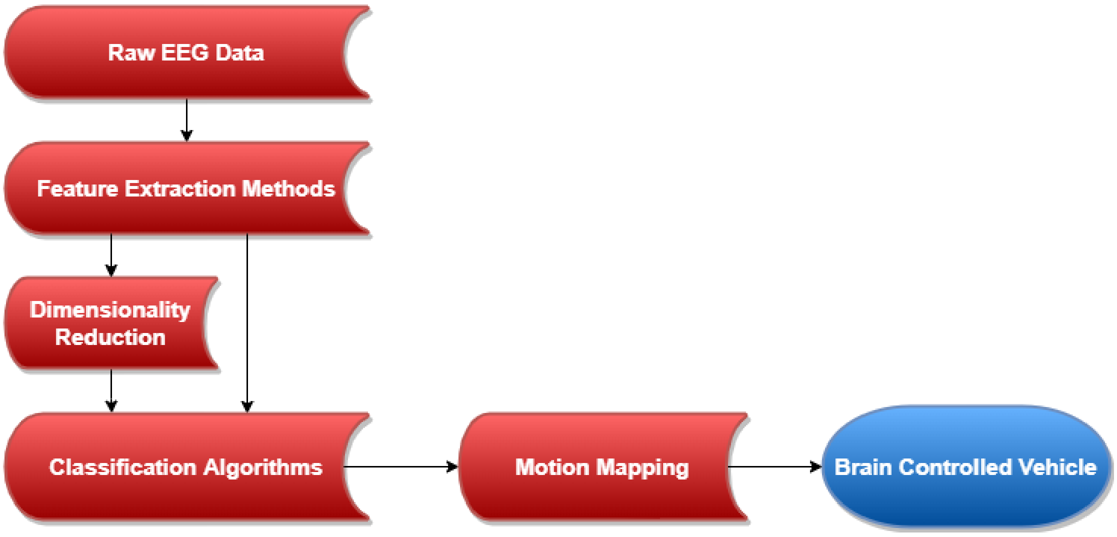

Figure 1: High Level Architecture of the System

RAW EEG Data

The first component of the architecture concerns the gathering of the RAW EEG Data. The current thesis will make use of the database from the BCI Competition 2008, which is also going to be presented in the following chapters.

Feature Extraction Methods

The second component of the architecture concerns the feature extraction stage. This component will be responsible for searching the best features which accurately describe the dataset and are intended to be informative and non-redundant. The methods that are going to be used are listed below:

- Short Time Fourier Transform (**STFT**)
- Power Spectral Density (**PSD**)

Dimensionality Reduction

The feature vectors produced by the feature extraction stage contain a lot of random variables and carry a lot of information. However, it is of major importance to find a correlation between these variables and reduce the amount of randomness under consideration. This procedure can be accomplished by implementing a dimensionality reduction method. In our case, we decide to perform an unsupervised linear transformation technique called Principal Component Analysis (PCA).

Classification Algorithms

This module has a vital role in this research. After identifying the appropriate feature vectors, it is time to perform and validate various classification methods in order to predict the class in which each one of the feature vector belongs to. The classes in our use case are motion related labels. These labels are: left hand (class 1), right hand (class 2), both feet (class 3), tongue (class 4). The classification algorithms that are going to be used are presented below:

- Support Vector Machine (**SVM**)
- k-Nearest neighbors (**kNN**)
- Random Forest (**RF**)
- Multilayer Perceptron - Backpropagation (**MLP-BP**)

Motion Mapping

Emotion mapping is the procedure in which motion related labels predicted by the previous stage (Classification Algorithms) are mapped into the four basic motions that are previously described (turn left, turn right, accelerate, slow down).

Brain Controlled Vehicle

The output of the machine learning framework presented above is a brain – controlled vehicle which is able to be driven with the power of the drivers' brain.

2. VALIDATION METRICS

After implementing all different kind of Machine Learning methods and techniques, it is of major importance to be able to validate the performance. In order to do so, we need to wisely select metrics that are going to be used. In our case, we decide to evaluate our results using Classification Accuracy.

Classification Accuracy: Classification accuracy is the ratio of number of correct predictions to the total number of input samples.

$$Classification\ Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions}$$

3. DISCOVERING KNOWLEDGE

"*We are drowning in information but starved from knowledge*". John Naisbitt's famous quote describes precisely the problem that exists when it comes to discovering knowledge. Sometimes, it is rather easy to come up with a large volume of data. In our case, we discovered the database from the BCI Competition 2008 that contains plenty of data in order to perform our research. The difficult part is trying to identify patterns in this huge volume of information and exploit them towards enhancing decision making. There are three main families that try to discover knowledge through information: Unsupervised Learning, Supervised Learning and Semi-supervised Learning.

3.1 Supervised Learning

The focus of supervised learning methods is to train an algorithm to identify specific patterns apparent in a set of training datasets. The user in this case owns a dataset and knows in advance the patterns and/or trends that appear in it. The main goal is to use this information and train an algorithm so the latter is able to identify similar patterns and/or trends in new datasets. In this case, the validity of the original patterns is assumed a-priori; since, all of our data are labeled, Supervised Learning was the perfect candidate in our thesis.

4. METHODOLOGY

This chapter will provide a detailed analysis of all the methodology used in this thesis. We will analyze all the data sources, all the methods used for Feature Extraction, as well as the way dimensionality reduction mechanisms were performed. In addition, we will analyze all the classification methods used, how the mapping between motion related labels and real motion was achieved, and finally, the brain-controlled vehicle.

4.1 Data Sources Identification

As previously stated, this section is all about identifying and accurately describing the data sources used in the current thesis.

4.1.1 Database of the BCI Competition 2008

This dataset is comprised of EEG data from 9 subjects. Four separate motor imagery tasks were included in the cue based BCI paradigm: the left hand (class 1), right hand (class 2), both feet (class 3), and tongue (class 4). For each subject, two sessions on various days were recorded. There are six runs in each session, separated by brief rest periods. One run consists of 48 trials (12 for each of the four possible classes), yielding a total of 288 trials per session.

4.1.2 Description of Dataset

In this section we will accurately describe the data used from the BCI Competition 2008 in order to perform our experiments.

First, we must describe the raw data used in our case. We have 9 participants and each one of them had to pass 288 trials. Additionally, each participant had a three-second pre-trial relaxation baseline. At the beginning of each session, a recording of approximately 5 minutes was performed to estimate the EOG influence. The recording was divided into 3 blocks: (1) two minutes with eyes open (looking at a fixation cross on the screen), (2) one minute with eyes closed, and (3) one minute with eye movements. The timing scheme of one session is illustrated in Figure 2.

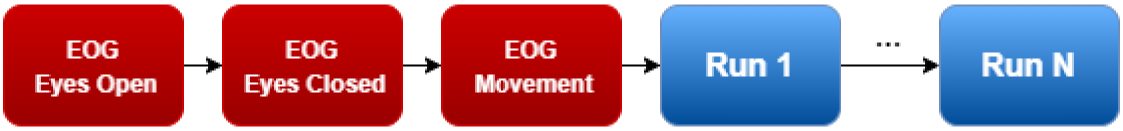

Figure 2: Timing scheme for one session

The subjects were sitting in a comfortable armchair in front of a computer screen. At the beginning of a trial (t = 0 s), a fixation cross appeared on the black screen. In addition, a short acoustic warning tone was presented. After two seconds (t = 2 s), a cue in the form of an arrow pointing either to the left, right, down, or up (corresponding to one of the four classes left hand, right hand, foot, or tongue) appeared and stayed on the screen for 1.25 seconds. This prompted the subjects to perform the desired motor imagery task. No feedback was provided. The subjects were asked to carry out the motor imagery task until the fixation cross disappeared from the screen at t = 6 s. A short break followed where the screen was black again. The paradigm is illustrated in Figure 3.

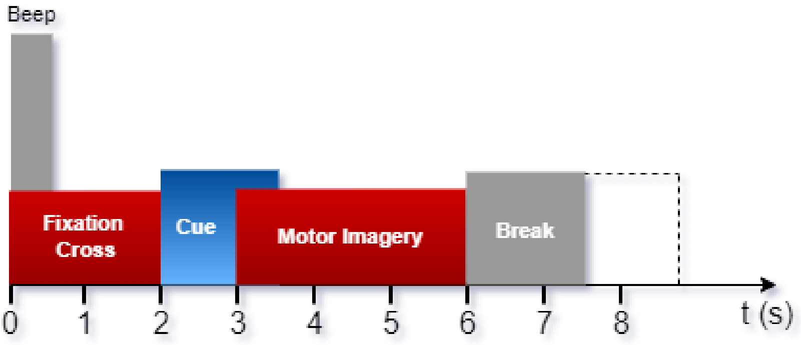

Figure 3: Timing scheme of the paradigm

Finally, twenty-two Ag/AgCl electrodes (with inter-electrode distances of 3.5cm) were used to record the EEG. The signals were sampled with 250Hz and bandpass filtered between 0.5Hz and 100Hz. The sensitivity of the amplifier was set to 100 μV. An additional 50Hz notch filter was enabled to suppress line noise.

Moving on, we have to describe the two different use cases considered in this thesis.

4.1.3 Description Of Use Cases

In this thesis we consider two different use cases in order to perform a more extensive and robust research and reach more mature results.

Use Case 1 (UC1): Subject Independent

In this use case, we are taking advantage of the users as a whole. In other words, we are not aiming at a personalized user experience but a generalized public opinion experience. This means that all raw data from all users are combined in a single file. Furthermore, we created four additional datafiles, one per motion related label as described in *Chapter 4.1.2*.

As a result the datafiles are:

1. Raw Data (EEG Signals) Aggregated from All Participants:

9 Participants	X	288 trials	X	22 Sensors/Channels	X	6s*250 *Hz* Samples per trial

Use Case 2: Subject Dependent

In this use case, we treat each participant as individual. This means that this dataset will provide a personalized experience based on the individual movements that every participant did. Once again, we created additional datafiles for the individual motion related labels for every participant. This means that we have 9 individual datafiles concerning emotion related labels.

As a result the datafiles are:

1. Raw Data (EEG Signals) for the individual participants (**x9 datafiles**):

Participant1	288 trials	X	22 Sensors/Channels	X	6s*250 *Hz* Samples per trial
			⋮		
Participant9	288 Trials	X	22 Sensors/Channels	X	6s*250 *Hz* Samples per trial

4.2 Feature Extraction Methods

In this section all the feature extraction methods examined and deployed during the implementation of the current thesis will be analyzed in depth. The main target of this processing stage is to select and combine variables into features and effectively reducing the amount of data that must be processed, while still accurately and completely describing the original dataset. The following figure is an overview of the Feature Extraction Mechanism, which illustrates all the components needed in order to produce the feature vectors.

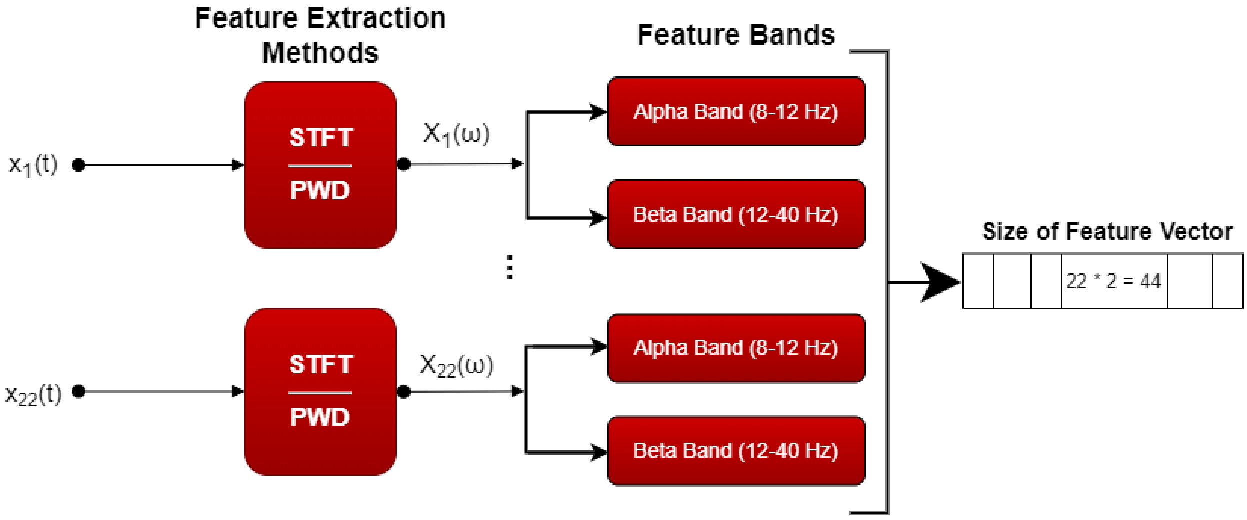

Figure 4: Feature Extraction Methods Mechanism Overview

According to the figure above, for each EEG signal $x_i(t)$ of each channel i (i ∈ {1,..,22}) two feature extraction methods (Short Time Fourier Transform, Power Spectral Density) where applied so as to extract the main frequencies of the human EEG waves. The main frequencies are:

Delta Band (1-4 Hz): The slowest and highest amplitude brainwaves. Delta frequencies are stronger in the right brain hemisphere, and the sources of delta are typically localized in the thalamus.

Theta Band (4–8 Hz): Theta waves can be recorded from all over cortex, indicating that it is generated by a wide-ranging network involving medial prefrontal areas, central, parietal and medial temporal cortices. Theta brainwaves are generally associated with brain processes underlying mental workload or working memory.

Alpha Band (8-12 Hz): Alpha waves are defined as rhythmic oscillatory activity within the frequency range of 8–12 Hz. Alpha waves have several functional correlates reflecting sensory, motor and memory functions.

Beta Band (12-40 Hz): Oscillations within the 12-40 Hz range are commonly referred to as beta band activity. This frequency is generated both in posterior and frontal regions. Active, busy or anxious thinking and active concentration are generally known to correlate with higher beta power.

Gamma Band (≥40 Hz): At the moment, gamma frequencies are the black holes of EEG research as it is still unclear where exactly in the brain gamma frequencies are generated and what these oscillations reflect.

From literature and experimentation, we observed that Alpha and Beta Bands are the ones that contribute the most in our use cases which are related to imagery motion related labels, and that is the main reason we decided to utilize only these two available bands amongst all five.

4.2.1 Short Time Fourier Transform (STFT)

The first method selected and applied to the initial raw EEG data is the Short Time Fourier Transform (STFT). STFT analysis is one of the techniques used in order to reveal the frequency contents of the EEG signals at each time point. STFT, also known as windowed Fourier, is applied to partition the EEG signal into several segments of short-time signals by shifting the time window with some overlapping. This process is called windowing. Therefore, the frequency spectrum was divided into frequency bins, whose size is dependent on the length of the window.

For the current thesis we have selected the “Hann” window. The Hanning window is a suitable STFT windowing function for analyzing EEG signals since it is characterized by its good frequency resolution. Furthermore, this type of window was selected for our thesis due to the fact that it is able to “smooth” data and return a friendly frequency representation of the signal that will be used for further analysis. The spectrogram resolution can be enhanced by modifying the length of the window; a large value of the window length provides a better frequency resolution, but poor time resolution. A shorter window length, however, provides the exact opposite outcome.

4.2.2 Power Spectral Density (PSD)

Finally, Power Spectral Density (PSD) was selected as the second feature extraction method and applied to the initial set of raw EEG Data. PSD is a suitable candidate for EEG signal processing due to the fact that it distributes the signal power over frequency and express the strength of the variations (energy) as a function of frequency. In other words, it shows at which frequencies variations are strong and at which frequencies variations are weak.

In the current thesis, among the various windows for calculating the PSD, soft-behaved Hanning-window was selected to analyze the unpredictable nature of brain signals. The Hanning window with the 1500 samples window's length was chosen to achieve an acceptable frequency resolution. This window selection with a smoothing characteristic was found to be more appropriate because of the different and unpredictable nature of brain signals.

4.3 Dimensionality Reduction

In this section, we are going to discuss about the benefits of performing a dimensionality reduction method. As explained in Chapter 4.2, the feature vector that was produced using the three feature extraction methods has 44 $(2 * 22)$ dimensions. Having so many dimensions increases the likelihood of correlations within the data. These correlations, produce redundancy in the information and reduce the quality of the dataset. Additionally, feature vectors with high dimensions increase the computational complexities. As a result, we chose to perform a dimensionality reduction method in our data set in order to exploit all the advantages mentioned above.

The method that we selected to implement is the Principal Component Analysis (PCA) which is going to be discussed in more detail in the following chapter.

4.3.1 Principal component analysis (PCA)

Principal Component Analysis (PCA) is a widely used method in many research projects related to EEG signal analysis, in order to reduce the dimension of the initial sensors' data. As already mentioned and explained in Chapter 4.4, there is of high importance, for the validity of our experimentation results, to analyze and classify features, to find a balance between the variance of our data and their dimension. The target for the dimensionality reduction of our thesis is the creation of features with the following characteristics:

- **High Variance**: Features with high variance contains a useful information which is a requirement for building an effective Machine Learning Model.
- **Uncorrelated**: Features with high correlation are less useful and in certain cases downright harmful for our study.
- **Low Number of Features**: Too many features relative to observations would not only result in an overfit model that performs poorly out of sample but also in high computational complexity.

Taking all the above into consideration and after our experimentation phase we concluded in selecting PCA as a dimensionality reduction method due to the fact that it totally covers the required characteristics mentioned above. For our UCs we have selected the number of principal components which preserve around 98.8% or 99% of the total variance of the initial feature data. More details about the exact number of the principal components used in our experimentation will be given in Chapter 5.

The figure 4 illustrates in a high level view the process of the dimensionality reduction using PCA. As input data the initial features, derived from the three feature extraction methods analyzed in Chapter 4.2, were fed into the PCA component in order to decide upon the appropriate number of components that best describes our brain signal while ensuring that there will be no violations of the three principal targets presented above.

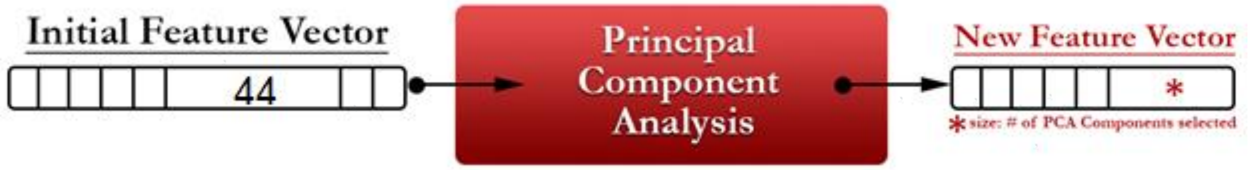
Initial Feature Vector
44
Principal Component Analysis
New Feature Vector
*
* size: # of PCA Components selected

Figure 5 Principal Component Analysis Mechanism Overview

4.4 Classification Algorithms

In the current chapter all the Machine Learning algorithms chosen and deployed during the implementation of the current thesis project will be analyzed in depth. By the term classification we are referring to a technique of categorizing the provided data into a desired and distinct number of classes where we can assign a label to each class. As already mentioned and explained in Chapter 4.1.2, the classification problem that the current thesis addresses is a Supervised Binary Classification problem where for each feature vector (Chapter 4.2) a classification model had to map one by one the 4 motion related labels (left hand, right hand, feet, tongue) to a class. A classifier utilizes some training data, so as to understand how given input variables relate to the class. After this stage, the classifier is ready to predict the class (label) that each new sample belongs to.

Five machine learning algorithms were selected and compared, using the metrics described in Chapter 2, the Support Vector Machine (SVM), k-Nearest Neighbors (kNN), Naive Bayes (NB), Random Forest (RF) and Multilayer Perceptron with Backpropagation (MLP-BP). Each one of them will be explained in detail in the following chapters and the evaluation results will be presented in Chapter 5.

Some General Terminology related to Machine Learning:

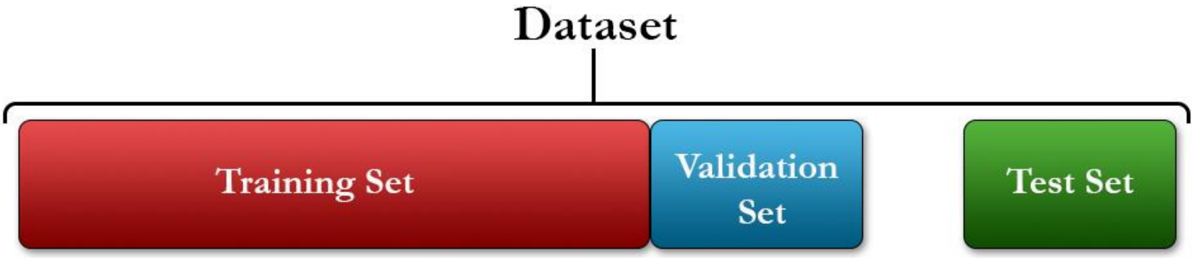

Figure 6 Training, Validation and Test Data Sets

Classifier: An algorithm that maps the input data to a specific category/class

Training Set: A large subset of the input data that is used in order to fit the classification model (most of the times is 80% of the initial dataset).

Test Set: The rest of the data (20%) of the initial dataset that will be used in order to evaluate our classification model is the Test Set. The test set is used to provide an unbiased evaluation of the final model fit on the training dataset.

Validation Set: The sample of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning classification model hyperparameters.

Classification model: A classification model recognizes some patterns in the input values given for training. It predicts the class labels/categories the new data (test set).

Binary Classification: Classification task with two possible outcomes.

Validation: is a method used to tune the hyper-parameters of the model and is done on the validation set.

Evaluation: is a method used to test the final performance of the algorithm and is done on the test set.

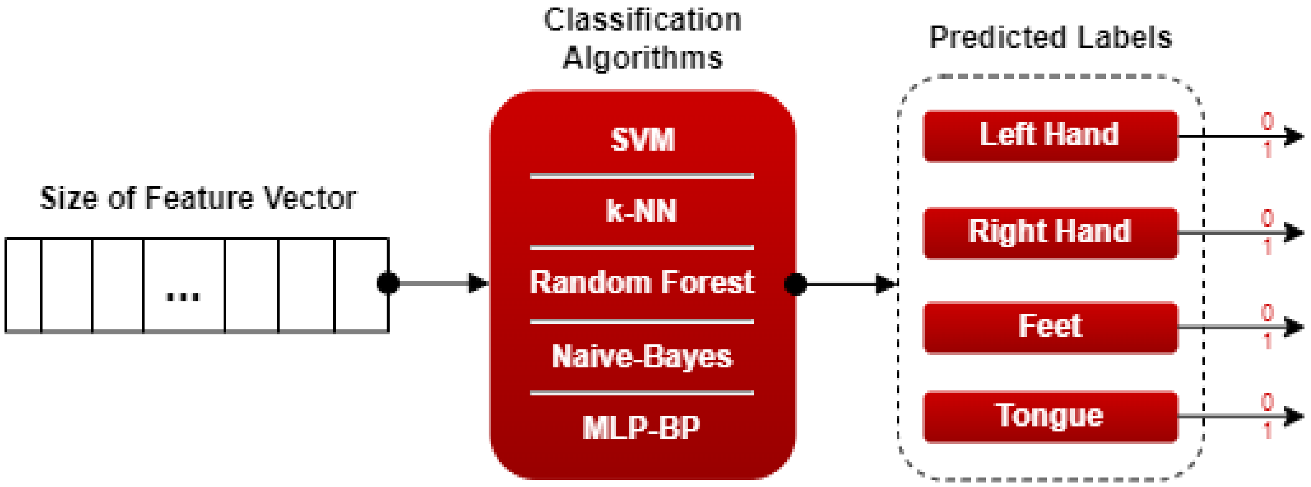

Figure 7: Classification Algorithms Mechanism Overview

4.4.1 Support Vector Machines (SVM)

The first and well-known classification algorithm, examined during the implementation of our thesis, is the Support Vector Machine. Support Vector Machine abbreviated as SVM can be used for both regression and classification tasks.

The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space (where N represents the number of features) that distinctly classifies the data points. By the term hyperplane we are referring to decision boundaries that help classify the data points. To separate the two classes of data points for each one of the emotion related labels, there are many possible hyperplanes that could be chosen. Our objective is to find a plane that has the maximum distance between data points of both classes. In order to better classify future data points we selected to maximize the margin distance.

The following figure illustrates an example of a non-linear classification problem solved using the SVM machine learning algorithm, where the circle with the green chromatic indication represents the hyperplane selected which better separates the data points belong to the two main classes (Class 1, 2).

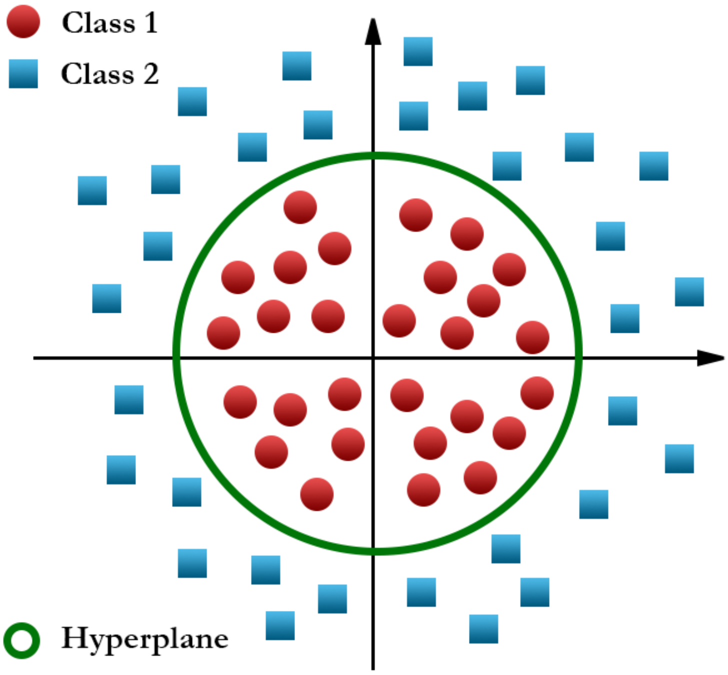

Figure 8: SVM Example Scheme

During the implementation of the SVM we had to decide upon the values of the core parameters related to the algorithm. More specifically, we had to tune the kernel, regularization, gamma and margin of SVM.

- **Kernel:** The function of kernel is to transform the input data into the required form. Different SVM algorithms use different types of kernel functions. These functions can be different types. While implementing SVM using scikit-learn library we experimented with three kernels the linear, polynomial and radial basis function (RBF). The right kernel is crucial, because if the transformation is incorrect, then the model can have very poor results.
- **Regularization:** The Regularization Parameter (in python it's called C) in the SVM optimization expresses the degree of importance that is being given to miss-classified data. If the C parameter value is higher, the optimization will choose smaller margin hyperplane, so training data miss-classification rate will be lower. On the other hand, if the C parameter value is low, then the margin will be large, even if there will be miss classified training data points.
- **Gamma:** The next important parameter for tuning is Gamma. The gamma parameter defines how far the influence of a single training point reaches. This means that higher Gamma value will consider only points close to the hyperplane and lower Gamma values will consider points at greater distance from the hyperplane.
- **Margin:** The last parameter is the margin. This distance from the decision surface (hyperplane) to the closest data point determines the margin of the SVM classifier. Higher margin results in a better classification model due to the fact that makes no low certainty classification decisions. According to the previous state the margin value should be always maximized.

4.4.2 k-Nearest neighbors (kNN)

The k-Nearest Neighbors algorithm or kNN is one of the simplest machine learning algorithms used in classification problems. kNN is based on the elementary state that similar data exist in close proximity. In kNN, K is the number of nearest neighbors. The number of neighbors is the core deciding factor, in order to result in better performance. K is generally an odd number if the number of classes is 2.

In order for a new data point P to be classified to one of the two in total classes, firstly we find the k closest points to P and then classify this point based on the majority vote of its k neighbors. Each one of the closest neighbors, votes for its class and the class with the most votes is taken as the prediction for the point P. For our thesis, in order to find the k closest neighbors we selected the straight-line distance also called the Euclidean distance, which is a popular and familiar choice. To better summarize the kNN algorithm we used the three following simple steps:

- Calculate the Euclidean distance between the new data point and the rest of the data points
- Find the k nearest neighbors
- Vote for the label of the new data point

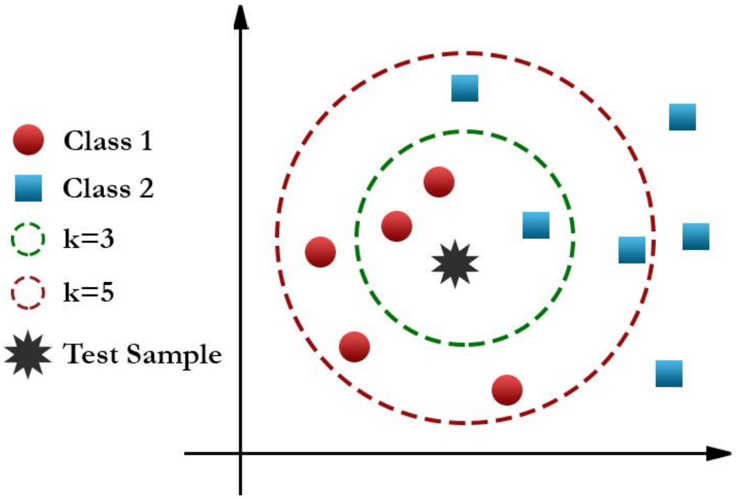

Figure 9: k-NN Example Scheme

Last but not least, the process of deciding upon the value of the K parameter is significant for our results. In the case of selecting a small number of neighbors, the noise will have a higher influence on the result, and a large number of neighbors make it computationally expensive. A small number of neighbors will result in having low bias but high variance. On the other hand, many neighbors will have a smoother decision boundary which means lower variance but higher bias. After experimenting with the K value, we concluded in an even number which is keeping balance between variance and bias and also conduce to a better performance.

4.4.3 Random Forest (RF)

The next machine learning algorithm used in our thesis, is the Random Forest. The random forest is based on a standard machine learning technique called a “decision tree”. A decision tree is a flowchart-like structure that uses a tree-like graph or model of decisions and their possible outputs. In a decision tree each node sets a query on an attribute, a branch represents the output of that condition and the leaf represents a class label. All the paths between root and leaves represent the classification rules. Considering what is already mentioned, decision tree is one way to display an algorithm that only contains conditional control statements. In a decision tree, an input is entered at the top and as it traverses down the tree the data gets bucketed into smaller and smaller sets.

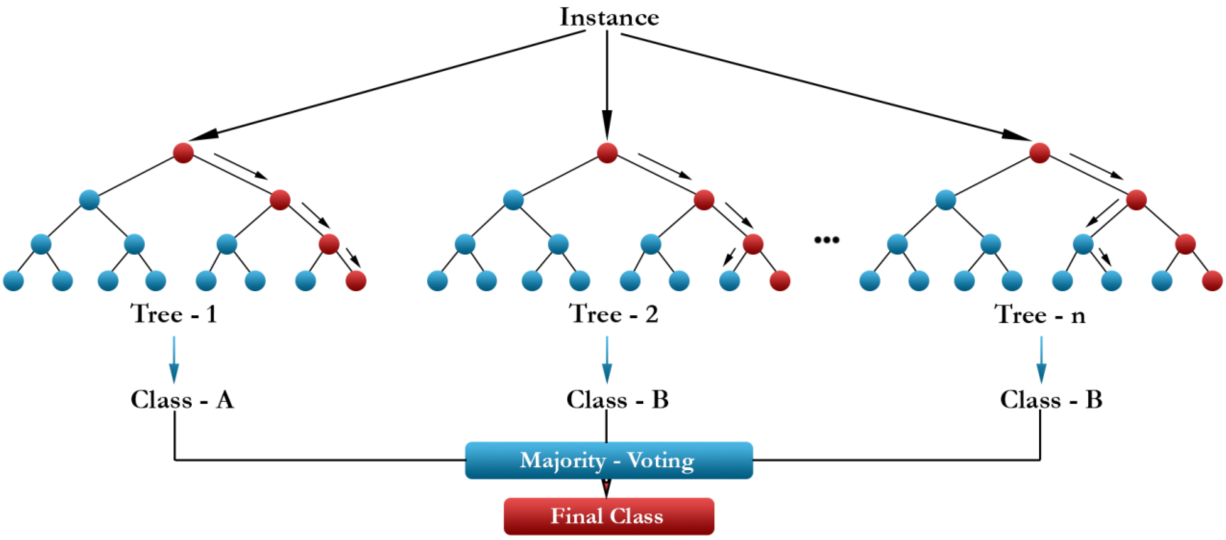

Figure 10 Random Forest Example Scheme

The random forest combines hundreds or thousands of decision trees, trains each one on a slightly different set of the observations, splitting nodes in each tree considering a limited number of the features. The final predictions of the random forest are made by averaging the predictions of each individual tree. The figure above illustrates a high level view of the Random Forest in order to better understand its logical steps. Figure 10 shows a RF which includes n decision trees, each one of them has conditional flows which result in a specific class. Finally, after all the decision trees result in a class, a majority voting concludes to the Final class selected for the feature.

4.4.4 Multilayer Perceptron - Backpropagation (MLP-BP)

The last Machine Learning Method selected is the Multilayer Perceptron with Backpropagation (MLP-BP). A multilayer perceptron (MLP) is a deep artificial neural network. It is composed of multiple layers of perceptrons. An MLP consists of at least three layers of nodes as presented in the figure below. More specifically, there is always an input layer which receives the input signal, an output layer that makes a decision or prediction about the input data, and in between there is a number of hidden layers that are the true computational engine of the MLP.

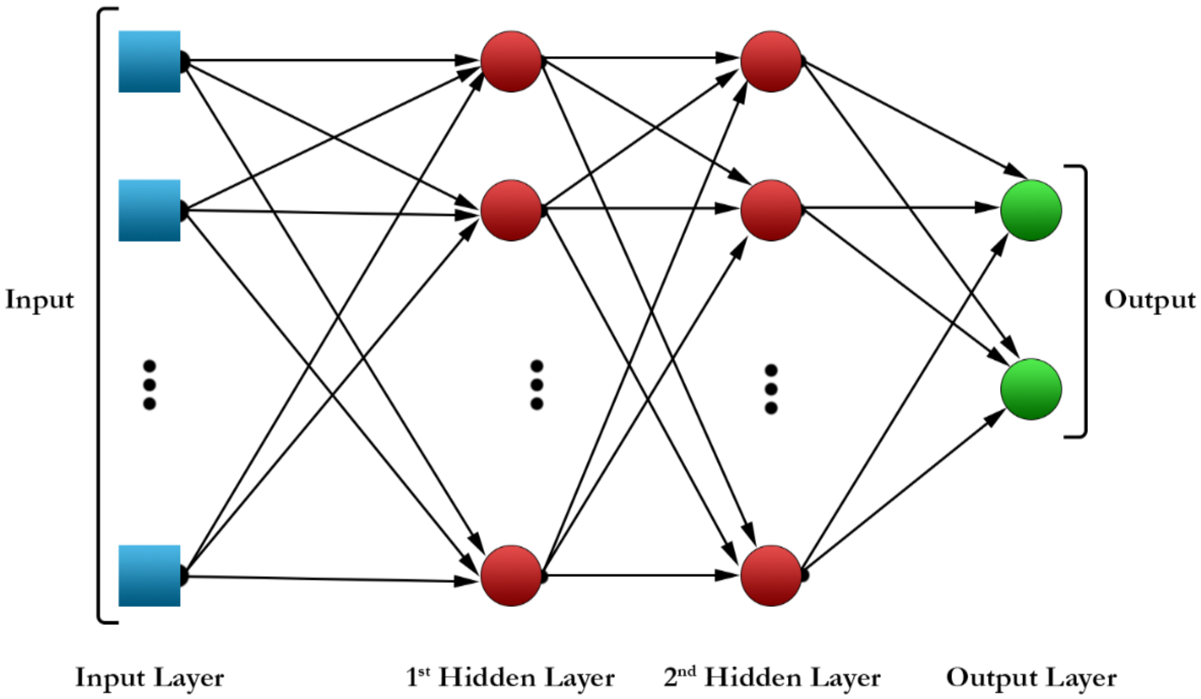

Figure 11 MLP-BP Example Scheme

MLPs are trained and learn to model the correlation (or dependencies) between those inputs and outputs. Training involves adjusting the weights and biases of each neuron (perceptron), of the model in order to minimize the error.
In a supervised classification problem, each input vector is associated with a label (ground truth). The output of the network gives a prediction, for each input fed to the neural network. In order to measure the performance of our classifier, the loss function should be defined. The loss will be high if the predicted class does not correspond to the ground truth class and it will be low otherwise. During the experimentation phase the main target was to better train the network. An optimization procedure was taken place during the current thesis, given the appropriate attention to the loss function and the optimizer. This procedure resulted in finding the values for the set of weights, which minimize the loss function. Backpropagation is used to make those weigh and bias adjustments relative to the error, and the error itself measured using Binary Cross Entropy.

4.5 Motion Mapping

Having introduced and explained all the techniques and methods for feature extraction, data augmentation, dimensionality reduction and classification it is time to explain how to perform the mapping from the imagery motion related labels predicted in Chapter 4.5 to the most basic and powerful motions that presented in Chapter 1. For reminder, these are: turn left, turn right, accelerate and slow down.

Furthermore, to perform the mapping, it is essential to note that each participant must be focused during the trial, because each imagery motion label maps directly to a specific movement.

- If left hand = **1** then: **motion□Turn Left**
- If right hand = 1 then: **motion□Turn Right**
- If Feet = 1 then: **motion□Accelerate**
- If Tongue = **1** then: **motion□Slow down**

5. EXPERIMENTAL RESULTS

After analyzing and describing our complete methodology used for the implementation of this thesis, it is time to present our experimental results in order to validate our claims.

Firstly, it is crucial to present the environment in which these experiments took place. All the experiments have been executed using the Anaconda 2022 combined with Jupyter Notebook 5.5.0 and Python 3.9.8. The Operating Systems used for trials were Debian 11 and Windows 11 .

Moving on, the systems used for these experiments have the specifications mentioned below:

- Operating System: Windows 11
  - AMD Ryzen 7 5800H @ 4.4GHz 8/16 - Core processor
  - 16 GB 3200MHz of DDR4 Ram
  - Nvidia GeForce RTX 3060 graphics card with 6GB GDDR6 of VRam
- Operating System: Debian 11
  - Intel Core I7-7500U @ 2.7GHz 2/4 - Core processor
  - 8 GB 2400MHz of DDR4 Ram
  - Nvidia GeForce 940MX graphics card with 2GB GDDR6 of VRam

The following chapters will present all the experimental results of our research for all the UCs presented in Chapter 4.1.3

5.1 Subject Independent Experimentation and Results

The first results in this chapter are about the UC1. The following figures will present the accuracy (see Chapter 2) for all the algorithms used and described in Chapter 4.5

To begin with, ***Figures 12-14*** present the accuracy for all the algorithms using the Power Spectral Density Feature Extraction Method (see Chapter 4.2.3)

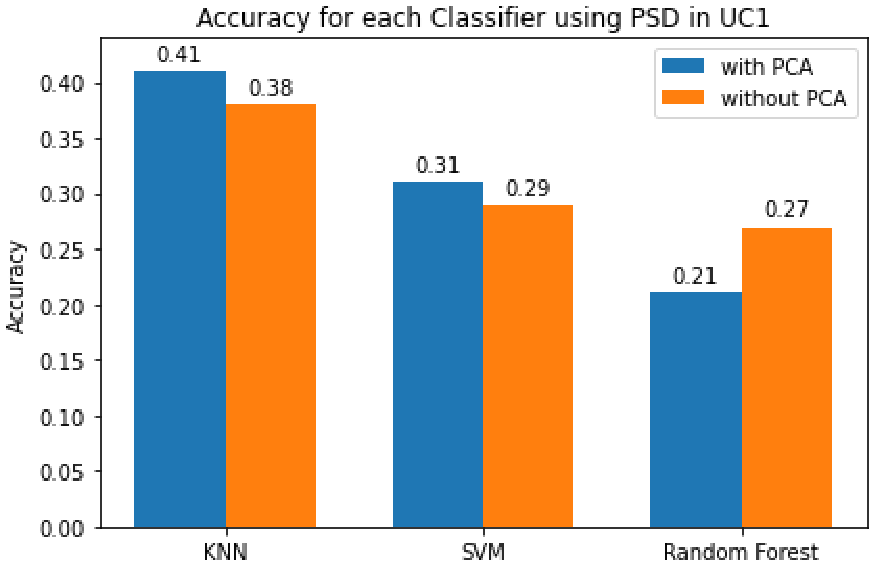

Figure 12: Accuracy using PSD in UC1, with and without PCA, for the kNN, SVM and RF Classification Algorithms

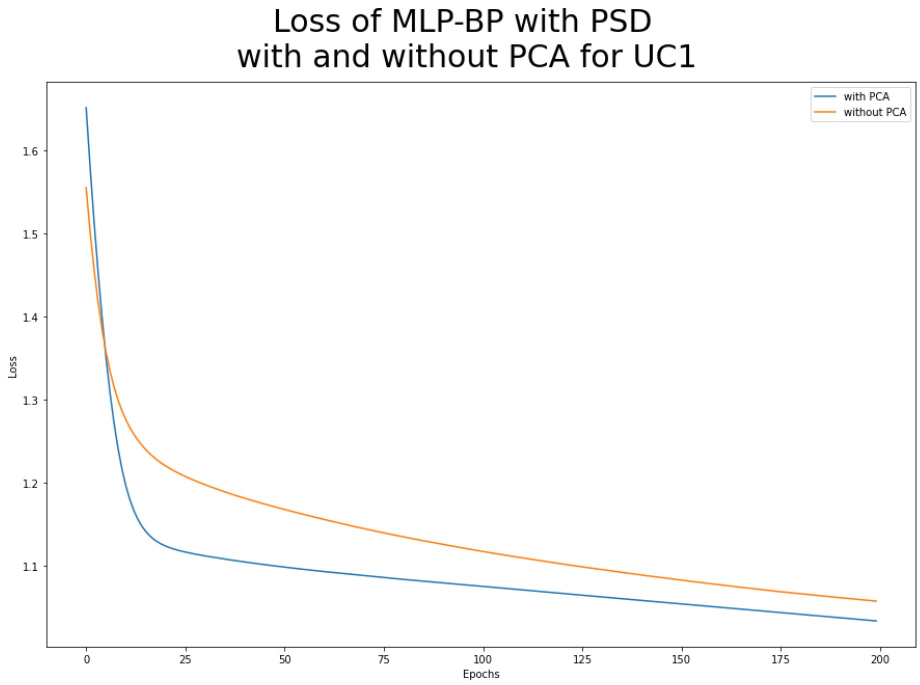

Figure 13:Loss using PSD in UC1, with and without PCA, for MLP-BP as the Classification Algorithm

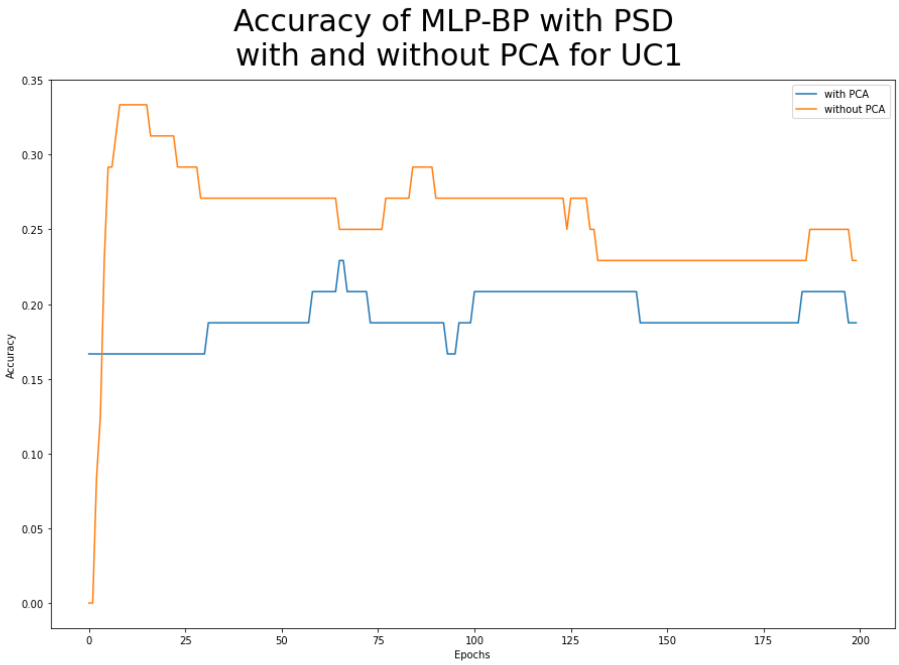

Figure 14: Accuracy using PSD in UC1, with and without PCA, for MLP-BP as the Classification Algorithm

In order to sum up the results for accuracy, we constructed the table seen below:

Table 1: Accuracy using PSD in UC1

Feature Extraction Method: PSD		
	With PCA	**Without PCA**
Algorithm	**Accuracy**	**Accuracy**
SVM	**0.41**	**0.38**
k-NN	**0.31**	**0.29**
RF	**0.21**	**0.27**
MLP-BP	**0.33**	**0.24**

Moving on, ***Figures 15 - 17*** present the accuracy for all the algorithms using the Short Time Fourier Transform Feature Extraction Method (see Chapter 4.2.2)

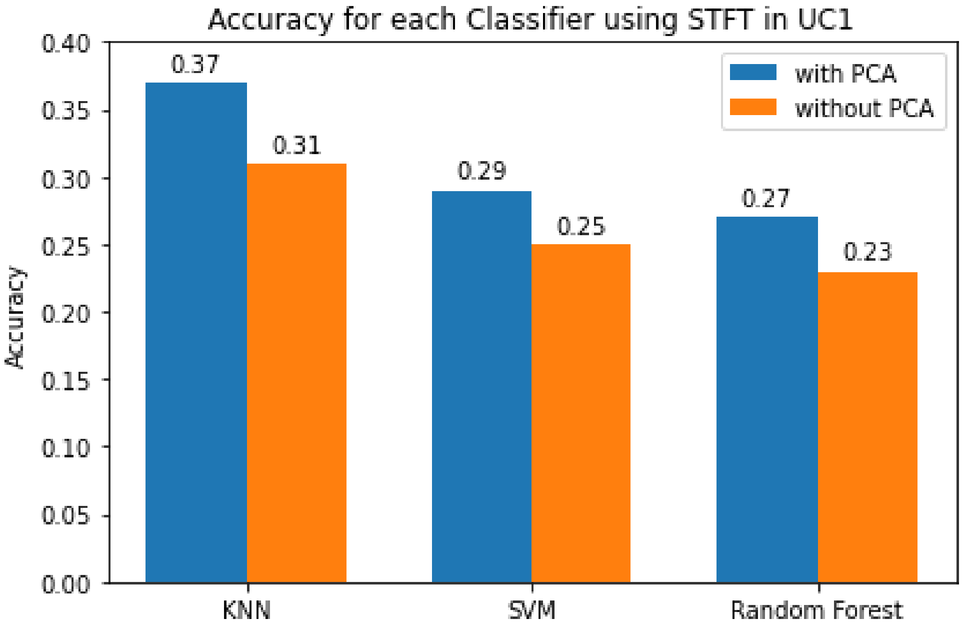

Figure 15: Accuracy using STFT in UC1, with and without PCA, for the kNN, SVM and RF Classification Algorithms

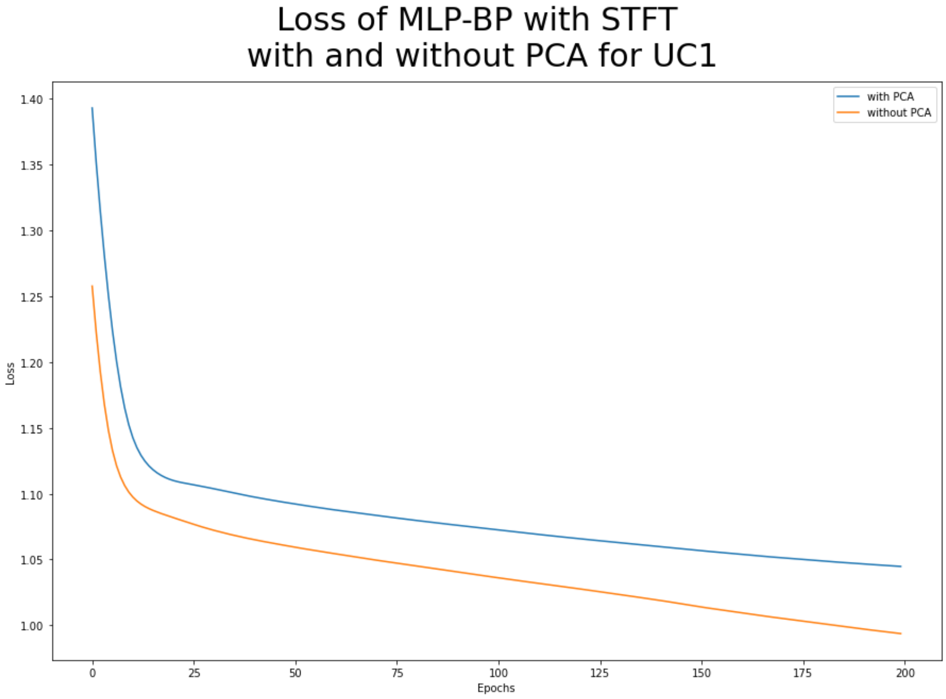

Figure 16: Loss using STFT in UC1, with and without PCA, for MLP-BP as the Classification Algorithm

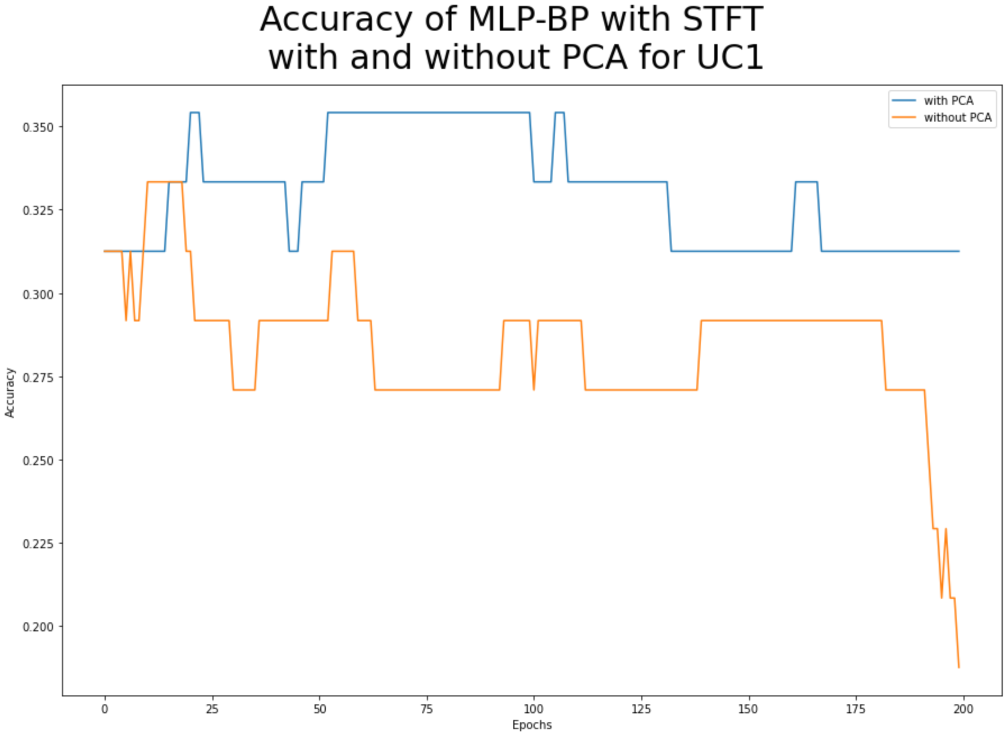

Figure 17: Accuracy using STFT in UC1, with and without PCA, for MLP-BP as the Classification Algorithm

In order to sum up the results for accuracy, we constructed the table seen below:

Table 2 Using STFT in UC1

Feature Extraction Method: STFT		
	No PCA	**With PCA**
Algorithm	**Accuracy**	**Accuracy**
SVM	**0.29**	**0.25**
k-NN	**0.37**	**0.31**
RF	**0.27**	**0.23**
MLP-BP	**0.345**	**0.26**

5.2 Subject Dependent Experimentation and Results

The presented results in this chapter are about the UC2. The following figures will present the accuracy (see Chapter 2) for all the algorithms used and described in Chapter 4.5.

5.2.1 Experimentation Results

The following figures will present the experimental results concerning the UC2, in which we provide the mean accuracies of all subjects, for each algorithm.

Moving on***, Figures 18 - 20*** present the accuracy for all the algorithms using the Power Spectral Density Feature Extraction Method. (see Chapter 4.2.3)

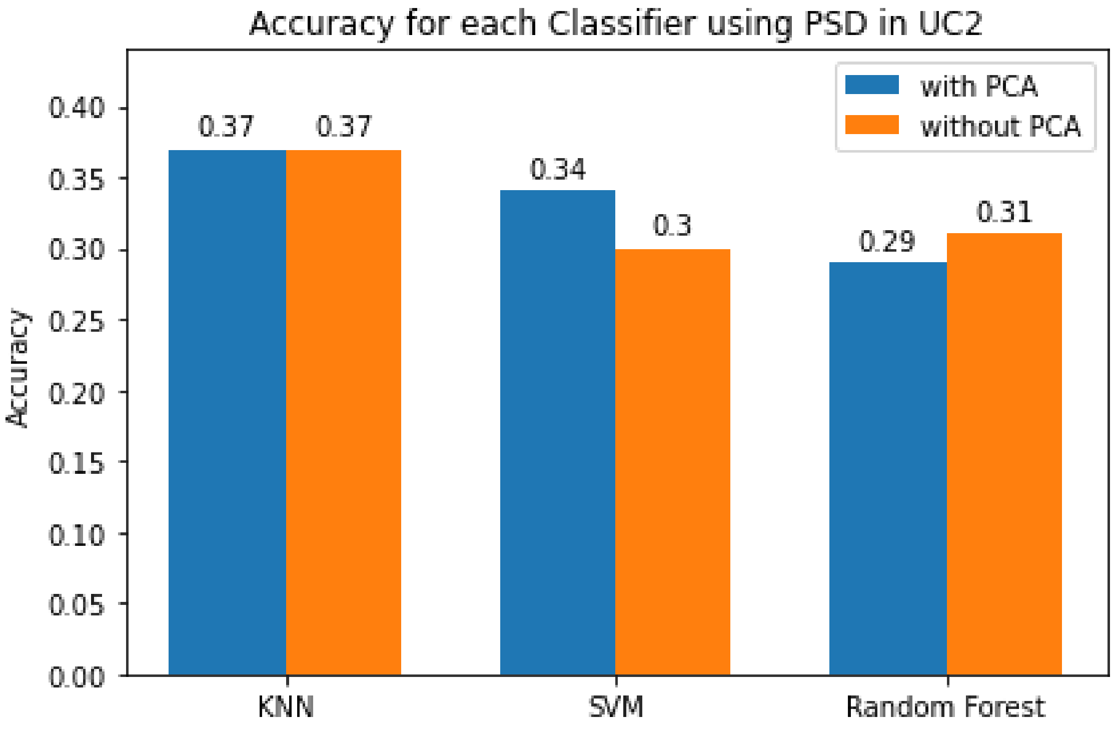

Figure 18: Accuracy using PSD in UC2, for kNN, SVM and RF as Classification Algorithms

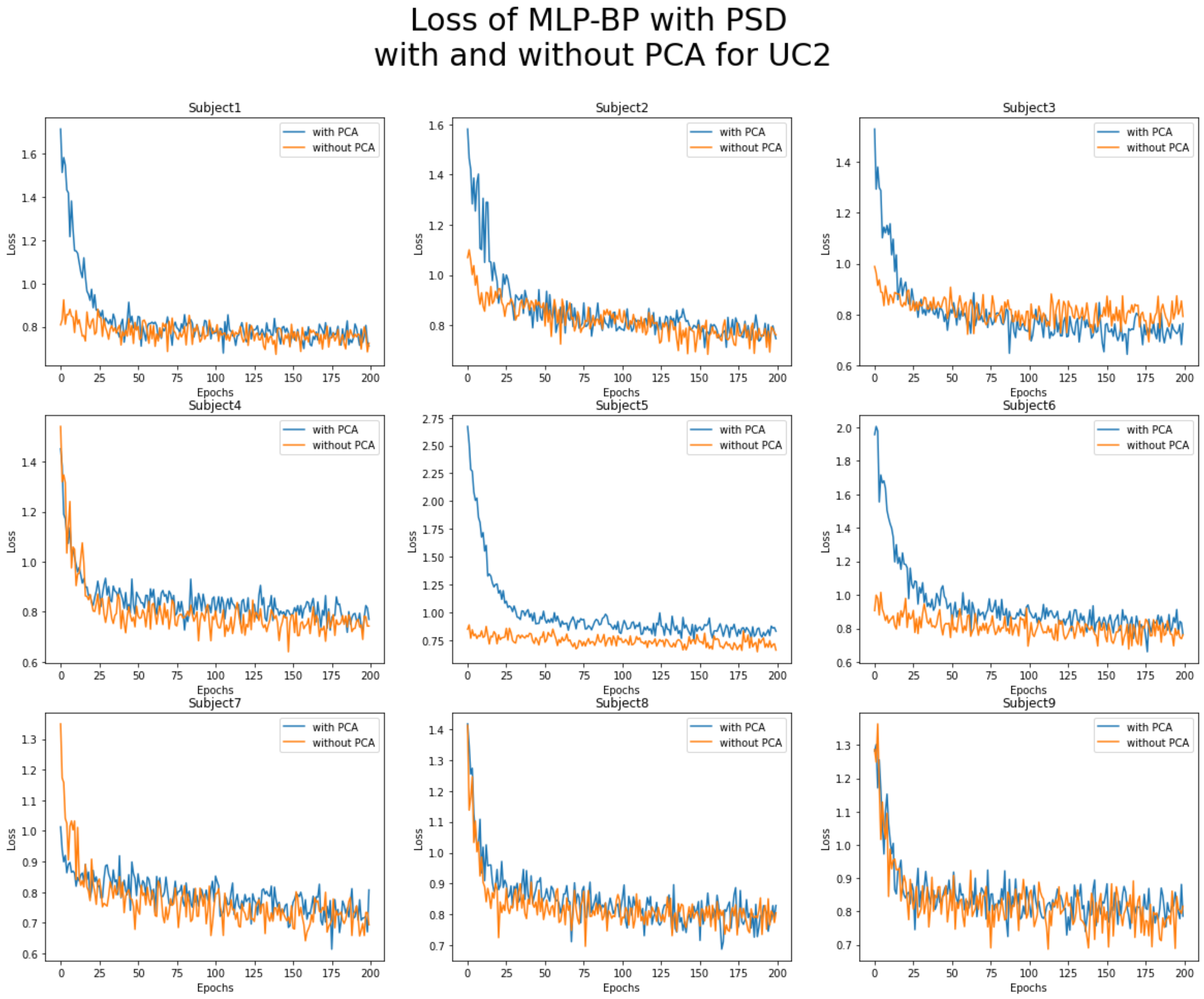

Figure 19: Loss for each Subject, using PSD, for MLP-BP as the Classification Algorithm

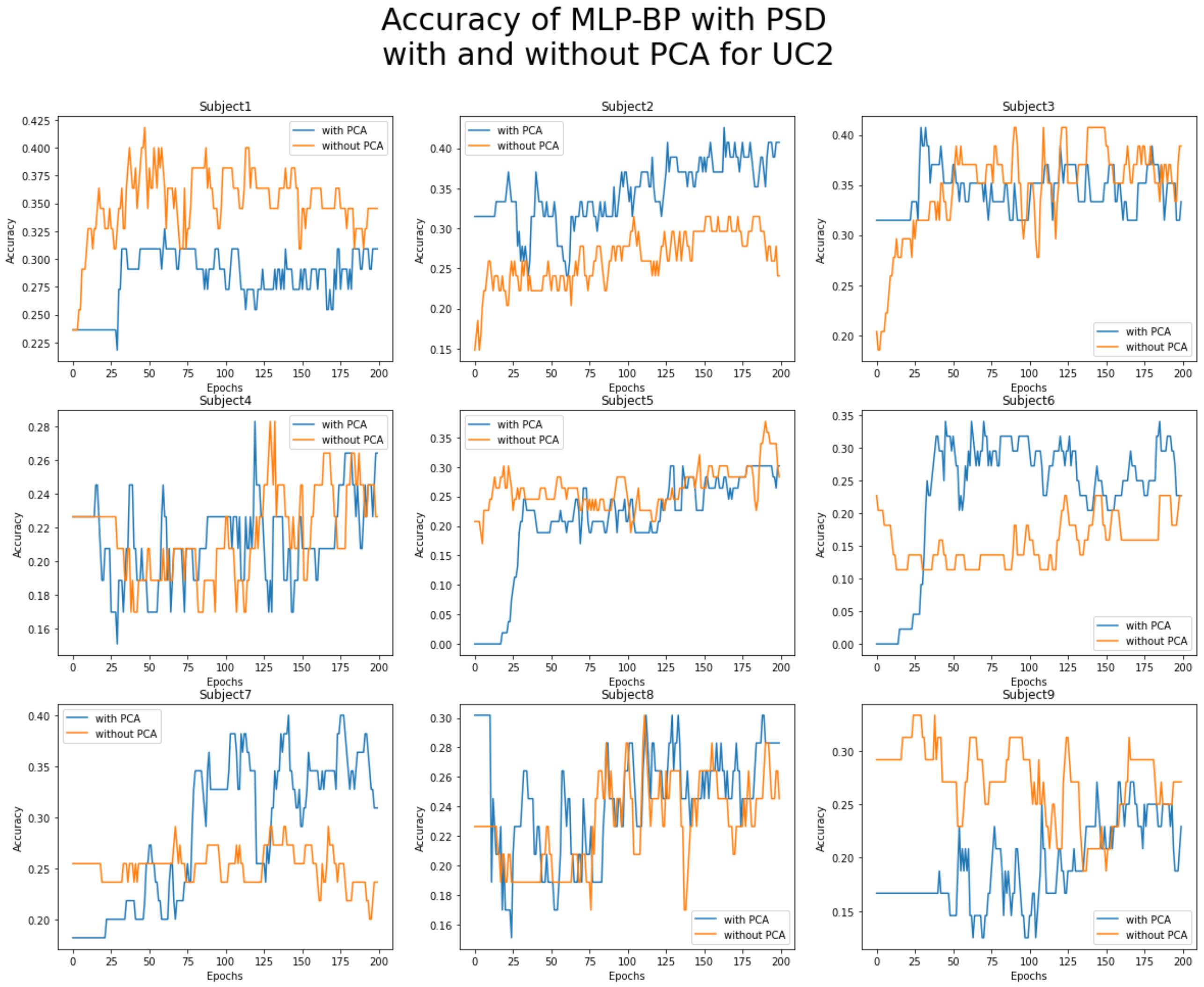

Figure 20: Accuracy for each Subject, using PSD, for MLP-BP as the Classification Algorithm

In order to sum up the mean results for accuracy, we constructed the table seen below:

Table 3 Accuracy using PSD in UC2

Feature Extraction Method: PSD		
	No PCA	With PCA
Algorithm	Accuracy	Loss
SVM	0.34	0.3
k-NN	0.37	0.37
RF	0.29	0.31
MLP-BP	0.35	0.32

Moving on, ***Figures 20 - 22*** present the accuracy for all the algorithms using the Short Time Fourier Transform Feature Extraction Method (see Chapter 4.2.2)

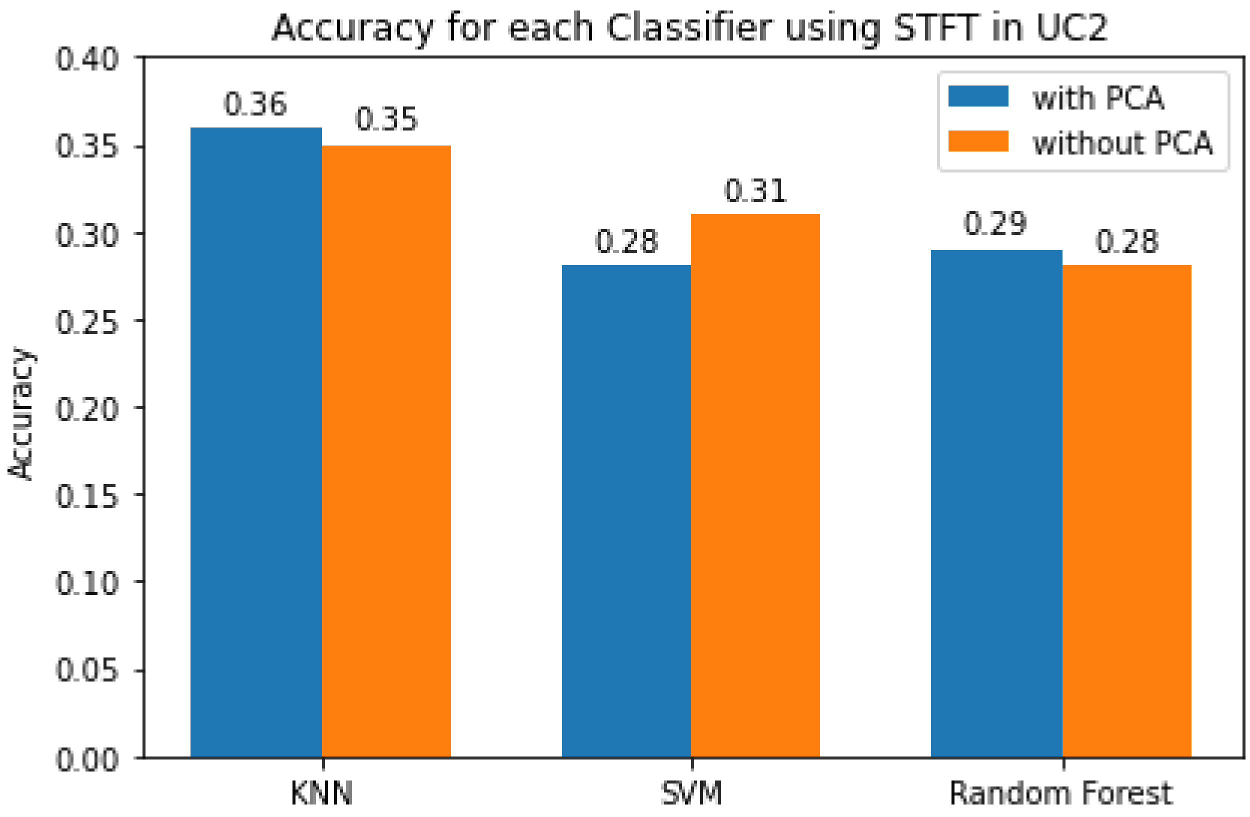

Figure 21: Accuracy using STFT in UC2, for kNN, SVM, RF as Classification Algorithms

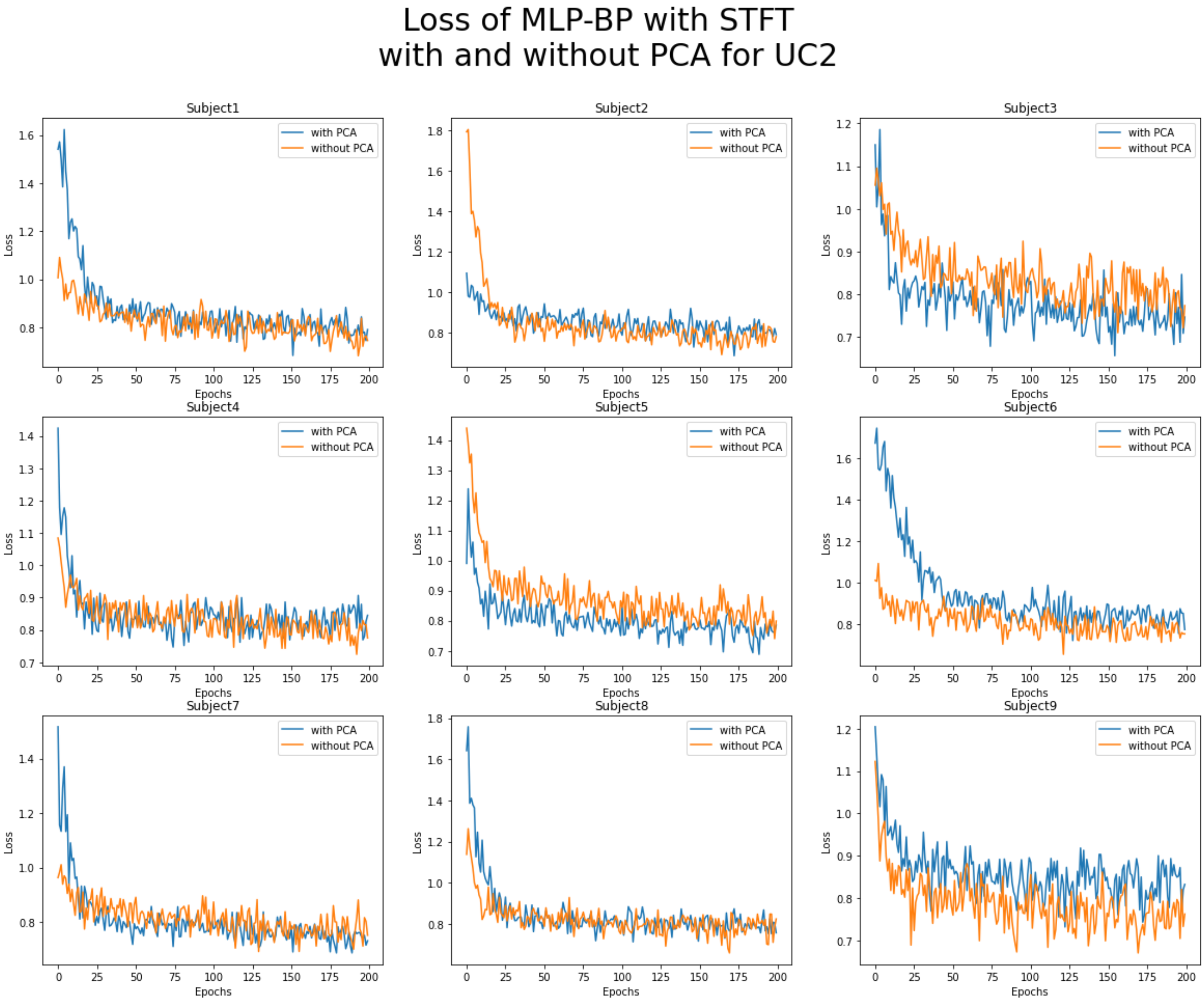

Figure 22: Loss using STFT in UC2, for MLP-BP as the Classification Algorithm for each subject

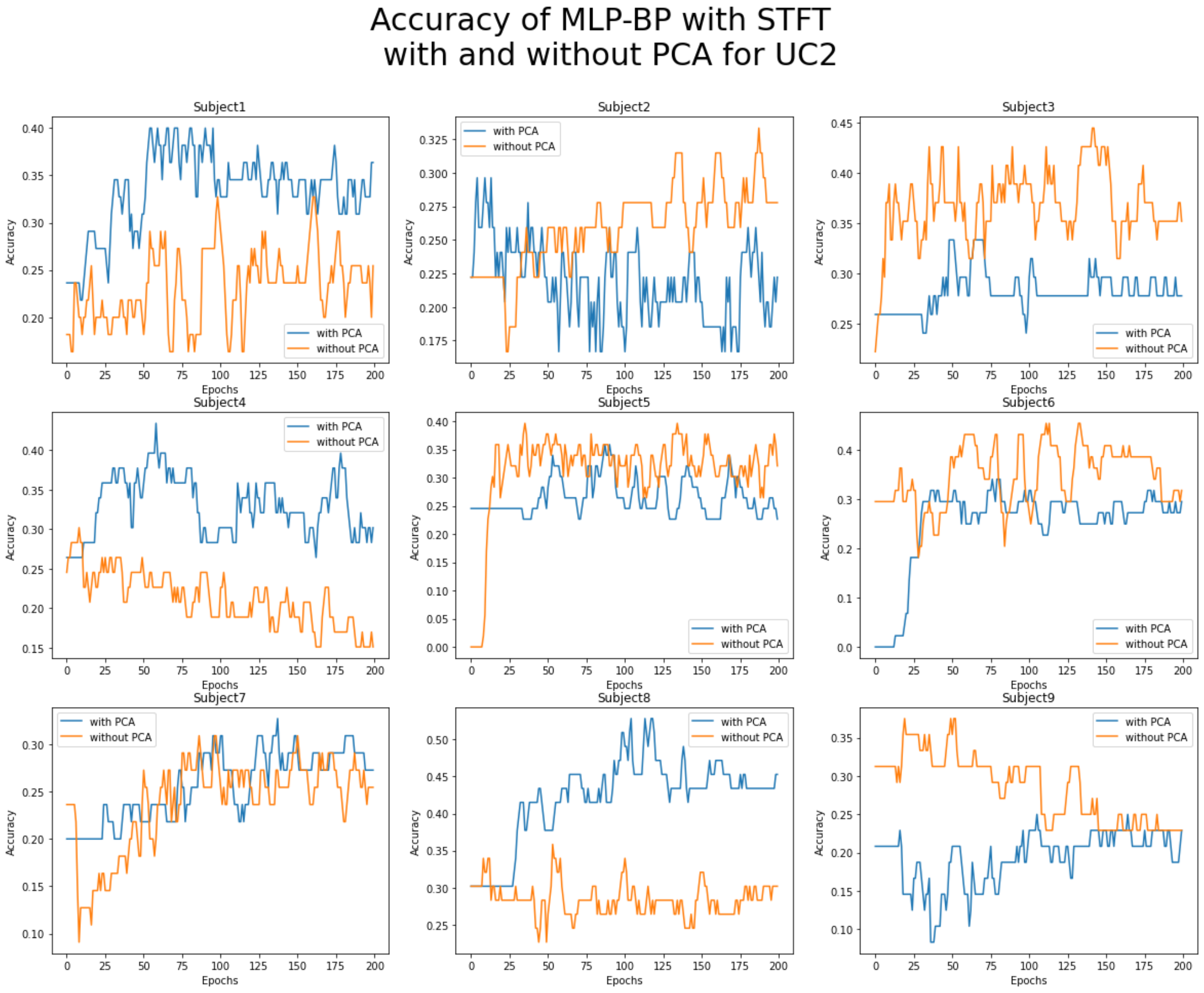

Figure 23: Loss using STFT in UC2, for MLP-BP as the Classification Algorithm for each subject

In order to sum up the results for accuracy and f1 score, we constructed the table seen below:

Table 4: Accuracy using STFT in UC2

Feature Extraction Method: STFT		
	No PCA	With PCA
Algorithm	Accuracy	Accuracy
SVM	0.28	0.31
k-NN	0.36	0.35
RF	0.29	0.28
MLP-BP	0.4	0.35

6. CONCLUSIONS

In the current thesis, we addressed the problem of EEG sentiment analysis targeting in implementing a brain controlled vehicle. One of the main contributions of our work is to express this task as a combinatorial optimization problem, and to propose methods to solve it using Machine Learning Techniques.

Different feature extraction methods and Machine Learning Classifiers have been presented. Our contribution here is twofold. First an experimental comparison related to the performance of each algorithm has been carried out, and second a Voting Classifier, which performed a soft voting between all the Machine Learning Classifiers that were optimized during the experimental period, was developed and validated.

The main focus of our thesis was on the optimization itself. Two Use Cases were examined and for each UC we chose the algorithms that best solve the problem. The experimentation phase included 2 types of Feature Extraction methods and 4 Algorithms for Classification. More specifically STFT and PSD were selected as feature extraction methods and SVM, kNN, Random Forest and MLP as ML Classifiers. Last but not least, we applied a dimensionality reduction method and more precisely the PCA so as to perform a linear mapping of the data to a lower-dimensional space in such a way that the variance of the data in the low-dimensional representation is maximized. By implementing dimensionality reduction we achieve not only lower computational cost but also better performance for the learning algorithm.

From an experimental point of view, our contribution lies in the comparison of the performance of the Machine Learning algorithms for each one of the 2 UCs after selecting the features that most describe the initial data and result in a better outcome. After a long experimental phase, we made several conclusions.

First of all, the Use Case which outperforms the rest of the Use Cases is the Subject Independent, using the PSD as a feature extraction method. On the other hand, the worst Use Case is the User Dependent, which resulted in a lower performance compared to the other Use Case. The final results of the User Dependent Use Case are relevant to the size of the initial data considering that we have only 288 trials (after applying data augmentation) for each one of the participants. As a result, the lack of a larger data set for each participant of the experiment led to inaccurate results.

Additionally, the feature extraction method that result (in most of the cases) in higher metric values and more accurate emotion predictions is the PSD. As for the Machine Learning Classifiers, they all respond similarly to the dataset. Moreover, PCA, as expected, led to significantly better output.

The accuracies resulted from all the Classification Algorithms and the Feature Extraction Methods, are undoubtedly very low. This is due to the fact that the dataset is based on the brainwaves of different subjects, where, during trials, there are distractions, extra noise, or the subjects were not as focused as desired. Surely, one cannot guarantee the quality of the database, which, given our results, might be considered as poor.

Many different adaptations, tests, and experiments have been left for the future due to lack of time. Future work concerns deeper analysis of particular mechanisms, new proposals to try different methods. There are some ideas that we would like to try in the future such as other types of Deep Learning Methods and more precisely Recurrent Neural Networks (e.g Long short-term memory) which best fit time series problems.

ABBREVIATIONS - ACRONYMS

EEG	Electroencephalography
BCV	Brain Controlled Vehicle
STFT	Short Time Fourier Transform
PSD	Power Spectral Density
SVM	Support Vector Machines
k-NN	k - Nearest Neighbors
MLP-BP	Multilayer Perceptron Back-Propagation
RF	Random Forest
PCA	Principal Component Analysis
UC	Use Case
ML	Machine Learning

REFERENCES.

[1] S. Koelstra, C. Muehl, M. Soleymani, J.-S. Lee, A. Yazdani, T. Ebrahimi, T. Pun, A. Nijholt, I. Patras, DEAP: A Database for Emotion Analysis using Physiological Signals, IEEE Transaction on Affective Computing, Special Issue on Naturalistic Affect Resources for System Building and Evaluation, in press.

[2] Rafael Ramirez and Zacharias Vamvakousis, Detecting Emotion from EEG Signals Using the Emotive Epoc Device, Department of Information and Communication Technologies Universitat Pompeu Fabra.

[3] Vangelis P. Oikonomou, Georgios Liaros, Kostantinos Georgiadis, Elisavet Chatzilari, Katerina Adam, Spiros Nikolopoulos and Ioannis Kompatsiaris, Comparative evaluation of state-of-the-art algorithms for SSVEP-based BCIs, technical report - arxiv.org January 2016.

[4] Pallavi Pandey and K. R. Seeja, Subject-Independent Emotion Detection from EEG Signals Using Deep Neural Network, International Conference on Innovative Computing and Communications, Indira Gandhi Delhi Technical University for Woman.

[5] Fang Wang, Sheng-hua Zhong Jianfeng Peng, Jianmin Jiang, and Yan Liu, Data Augmentation for EEG-Based Emotion Recognition with Deep Convolutional Neural Networks, College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518000, People’s Republic of China, Department of Computing, The Hong Kong Polytechnic University, Kowloon, Hong Kong.

[6] Danny Oude Bos, EEG-based Emotion Recognition. The Influence of Visual and Auditory Stimuli, Department of Computer Science, University of Twente P.O. Box 217, 7500 AE, Enschede, The Netherlands.

[7] Tegan Grace Harrison, The Emotiv mind: Investigating the accuracy of the Emotiv EPOC in identifying emotions and its use in Intelligent Tutoring System, Department of Computer Science and Software Engineering.

[8] Salma Alhagry, Aly Aly Fahmy, Reda A. El-Khoribi, Emotion Recognition based on EEG using LSTM Recurrent Neural Network, IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 8, No. 10, 2017.

[9] Prommy Sultana Ferdawoos Hossain, Istiaque Mannafee Shaikat, Fabian Parsia George, Emotion Recognition using Brian Signals based on Time-Frequency Analysis and Supervised Learning Algorithm, A thesis submitted to the faculty of BRAC University in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering, April, 2018.

[10] Soraia M. Alarcao, and Manuel J. Fonseca, Senior Member, IEEE, Emotions Recognition Using EEG Signals: A Survey, IEEE TRANSACTIONS ON AFFECTIVE COMPUTING

[11] J. Preethi, M. Sreeshakthy, A.Dhilipan, A Survey on EEG Based Emotion Analysis using various Feature Extraction Techniques, International Journal of Science, Engineering and Technology Research (IJSETR), Volume 3, Issue 11, November 2014.

[12] Nisha Vishnupant Kimmatkar, Dr. Vijaya B. Babu, Human Emotion Classification from Brain EEG Signal Using Multimodal Approach of Classifier, Department of CSE, K L University, Vaddeshwaram, Guntur.

[13] Priyanka A. Abhang, Bharti w. Gawali, Suresh C. Mehrotra, Introduction to eeg- and speech-based emotion recognition, Department of Computer Science and Information Technology.

[14] Roberto Munoz ,Rodrigo Olivares ,Carla Taramasco ,Rodolfo Villarroel ,Ricardo Soto ,Thiago S. Barcelos , Erick Merino, María Francisca Alonso-Sánchez, Using Black Hole Algorithm to Improve EEG-Based Emotion Recognition, Computational Intelligence and Neuroscience Volume 2018,

[15] Russell, J.A.: A circumplex model of affect. Journal of Personality and Social Psychology, Vol 39(6), Dec 1980, 1161-1178

[16] William Forde Thompson and Lena Quinto, Music and Emotion: Psychological Considerations, Article, January 2012

[17] Wikipedia - https://en.wikipedia.org/wiki/Standard_deviation

[18] Wikipedia - https://en.wikipedia.org/wiki/Approximate_entropy

[19] Towards Data Science - https://towardsdatascience.com/