June 2017

Real-time Classification of Biomedical Signals, Parkinson's Analytical Model

Abolfazl Saghafi
University of South Florida, asaghafi@mail.usf.edu

Real-time Classification of Biomedical Signals, Parkinson's Analytical Model

by

Abolfazl Saghafi

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Department of Mathematics and Statistics
College of Arts and Sciences
University of South Florida

Major Professor: Chris P. Tsokos, Ph.D.
Ramachandran M. Kandethody, Ph.D.
Lu Lu, Ph.D.
Dan Shen, Ph.D.

Date of Approval:
June 6, 2017

Keywords: Electroencephalography, Machine Learning, Eye state detection,
Classification, Parkinson's disease, Heredity, Phonocardiography

ACKNOWLEDGMENTS

I would like to express my gratitude to my major professor, Dr. Chris P. Tsokos, whose expertise, understanding, and patience, added considerably to my graduate experience. He consistently allowed this research accomplishment to be my own work, but steered me in the right direction whenever he thought I needed it.

I must thank the members of my committee, Dr. Kandethody M. Ramachandran, Dr. Lu Lu, Dr. Rebecca Wooten, and Dr. Dan Shen for their support and assistance at all levels of my research. In addition, I would like to thank Dr. Kamal Alsharif for agreeing to be the chairperson in such short notice.

I would also like to thank my friends in the University of South Florida who helped enrich the experience. Special thanks to Damian and Hannah Torres who helped me significantly in many aspects.

Finally, I must express my very profound gratitude to my parents. This accomplishment would not have been possible without them.

Thank you

Abolfazl Saghafi

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

The reach of technological innovation continues to grow, changing all industries as it evolves. In healthcare, technology is increasingly playing a role in almost all processes, from patient registration to data monitoring, from lab tests to self-care tools. The increase in the amount and diversity of generated clinical data requires development of new technologies and procedures capable of integrating and analyzing the BIG generated information as well as providing support in their interpretation.

To that extent, this dissertation focuses on the analysis and processing of biomedical signals, specifically brain and heart signals, using advanced machine learning techniques. That is, the design and implementation of automatic biomedical signal pre-processing and monitoring algorithms, the design of novel feature extraction methods, and the design of classification techniques for specific decision making processes.

In the first part of this dissertation Electroencephalogram (EEG) signals that are recorded in 14 different locations on the scalp are utilized to detect random eye state change in real-time. In summary, cross channel maximum and minimum is used to monitor real-time EEG signals in 14 channels. Upon detection of a possible change, Multivariate Empirical Mode Decomposes the last two seconds of the signal into narrow-band Intrinsic Mode Functions. Common Spatial Pattern is then employed to create discriminating features for classification purpose. Logistic Regression, Artificial Neural Network, and Support Vector Machine classifiers all could detect the eye state change with 83.4% accuracy in less than two seconds. We could increase the detection accuracy to 88.2% by extracting relevant features from Intrinsic Mode Functions and directly feeding it to the classification algorithms.

Our approach takes less than 2 seconds to detect an eye state change which provides a significant improvement and promising real-life applications when compared to slow and

computationally intensive instance based classification algorithms proposed in literatures. Increasing the training examples could even improve the accuracy of our analytic algorithms. We employ our proposed analytic method in detecting the three different dance moves that honey bees perform to communicate the location of a food source. The results are significantly better than other alternative methods in the literature in terms of both accuracy and run time.

The last chapter of the dissertation brings out a collaborative research on Parkinson's disease. As a Parkinsons Progression Markers Initiative (PPMI) investigator, I had access to the vast database of The Michael J. Fox Foundation for Parkinson's Research. We utilized available data to study the heredity factors leading to Parkinson's disease by using Maximum Likelihood and Bayesian approach. Through sophisticated modeling, we incorporated information from healthy individuals and those diagnosed with Parkinson's disease (PD) to available historical data on their grandparents' family to draw Bayesian estimations for the chances of developing PD in five types of families. That is, families with negative history of PD (type 1) and families with positive history in which estimations provided for the prevalence of developing PD when none of the parents (type 2), one of the parents (type 3 and 4), or both of the parents (type 5) carried the disease.

The results in the provided data shows that for the families with negative history of PD the prevalence is estimated to be 20% meaning that a child in this family has 20% chance of developing Parkinson. If there is positive history of PD in the family the chance increases to 33% when none of the parents had PD and to 44% when both of the parents had the disease. The chance of developing PD in a family whose solely mother is diagnosed with the disease is estimated to be 26% in comparison to 31% when only father is diagnosed with Parkinson's.

CHAPTER 1

INTRODUCTION AND THEORETICAL FRAMEWORK

As the technological innovation continues to grow, adjustment to changes in all industries is inevitable. In healthcare, technology is increasingly playing a role in almost all processes, from patient registration to data monitoring, from lab tests to self-care tools. The increase in the amount and diversity of generated clinical data requires development of new technologies and procedures capable of integrating and analyzing the BIG generated information as well as providing support in their interpretation [37].

To that extent, this dissertation focuses on the analysis and processing of biomedical signals using advanced machine learning techniques. This includes, the design and implementation of automatic biomedical signal pre-processing and monitoring algorithms, the design of novel feature extraction methods, and the design of classification methods for specific decision making on biomedical data. In this context, we develop a much faster method to identify random eye state change using brain signals in chapter 2. Our findings provide a valuable improvement in comparison to slow, computationally intensive instance based algorithms proposed in published literatures. Although, our primary focus is on classification of biomedical signals, the analytics as we see on the next chapter can be applied to other signals and time series data as well.

In chapter 3, we use the same proposed analytics to a famous classification problem, detection of honey bee dancing patterns. Chapter 4 presents our future work on classification of heart signals to detect various heart dysfunctions. Results of a collaborative work on Parkinson's disease is presented in chapter 5. This chapter presents basic and background information that one needs to know to better understand the following chapters.

1.1 Machine Learning

Machine learning is defined as a set of methods that can automatically detect patterns in data, and then use the uncovered patterns to predict future data, or to perform other kinds of decision making under uncertainty [58]. Machine learning tasks are typically classified into three broad categories, depending on the nature of the learning "signal" or "feedback" available to a learning system [72]. These are

- Supervised learning: The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.
- Unsupervised learning: No output labels are given to the learning algorithm, leaving it on its own to find structure in its input.
- Reinforcement learning: A computer program interacts with a dynamic environment in which it must perform a certain goal for example driving a vehicle or playing a game against an opponent. The program is provided feedback in terms of rewards and punishments as it navigates its problem space.

Our focus is on supervised learning, specifically classification problems in which inputs are classified into distinct groups. More details about supervised and unsupervised learning processes and examples are provided in the following.

1.1.1 Supervised learning

In the predictive or supervised learning approach, the goal is to learn a mapping from inputs x to outputs y, given a labeled set of input-output pairs $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{m}$. Here $\mathcal{D}$ is called the training set and m is the number of training examples [58].

In the simplest setting, each training input x_i is a D-dimensional vector of numbers, representing, say, the height and weight of a person. These are called features, attributes or covariates. In general, however, x_i could be a complex structured object, such as an image, a sentence, an email message, a time series, a molecular shape, a graph, etc.

(a) Setosa (b) Versicolor (c) Virginica

Figure 1.1: Three types of iris flowers: Setosa, Versicolor, and Virginica

Similarly the form of the output or response variable can in principle be anything, but most methods assume that y_i is a categorical or nominal variable from some finite set, $y_i \in \{1, ..., C\}$ (such as male or female), or that y_i is a real-valued scalar (such as income level). When y_i is categorical, the problem is known as classication or pattern recognition, and when y_i is real-valued, the problem is known as regression. Another variant, known as ordinal regression, occurs where label space Y has some natural ordering, such as grades A-F [58].

Example 1.1. Iris Plants data set. Fisher's paper and dataset on three types of iris plants is perhaps the best known example of classification in the literature [29]. The aim is to build a predicative model to classify the iris types (Setosa, Versicolour, or Virginica) using 4 numeric, predictive attributes (sepal length, sepal width, petal length, and petal width). Data set contains 50 sample measurements of each iris type and is available at UCI Machine Learning Repository [103].

Example 1.2. Combined Cycle Power Plant. The dataset contains 9568 data points collected from a Combined Cycle Power Plant over 6 years (2006-2011), when the power plant was set to work with full load. Features consist of numerical values of hourly average ambient variables Temperature (T), Ambient Pressure (AP), Relative Humidity (RH) and Exhaust Vacuum (V) to predict the net hourly electrical energy output (EP) of the plant. The data set for this regression problem is publicly available at UCI Machine Learning Repository [103].

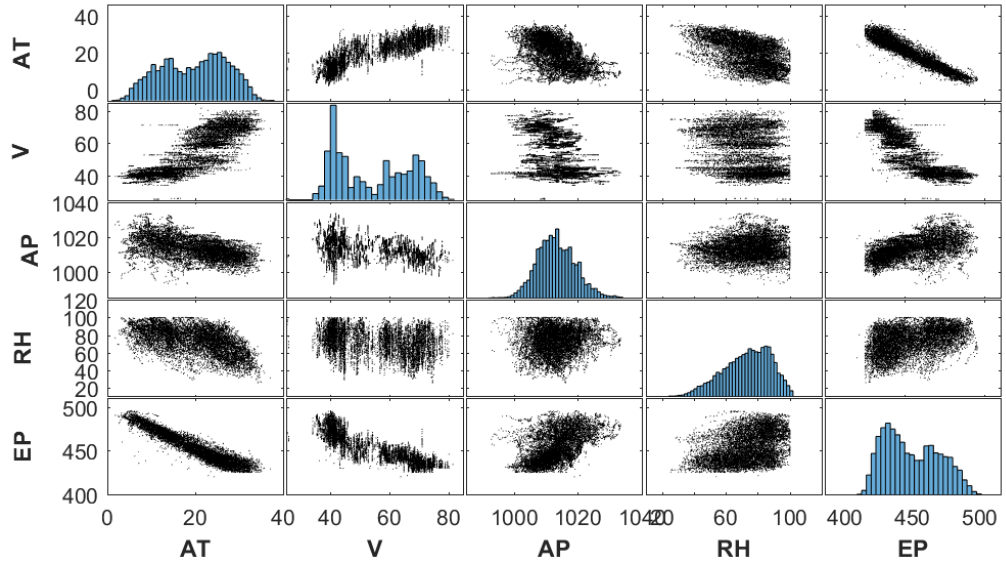
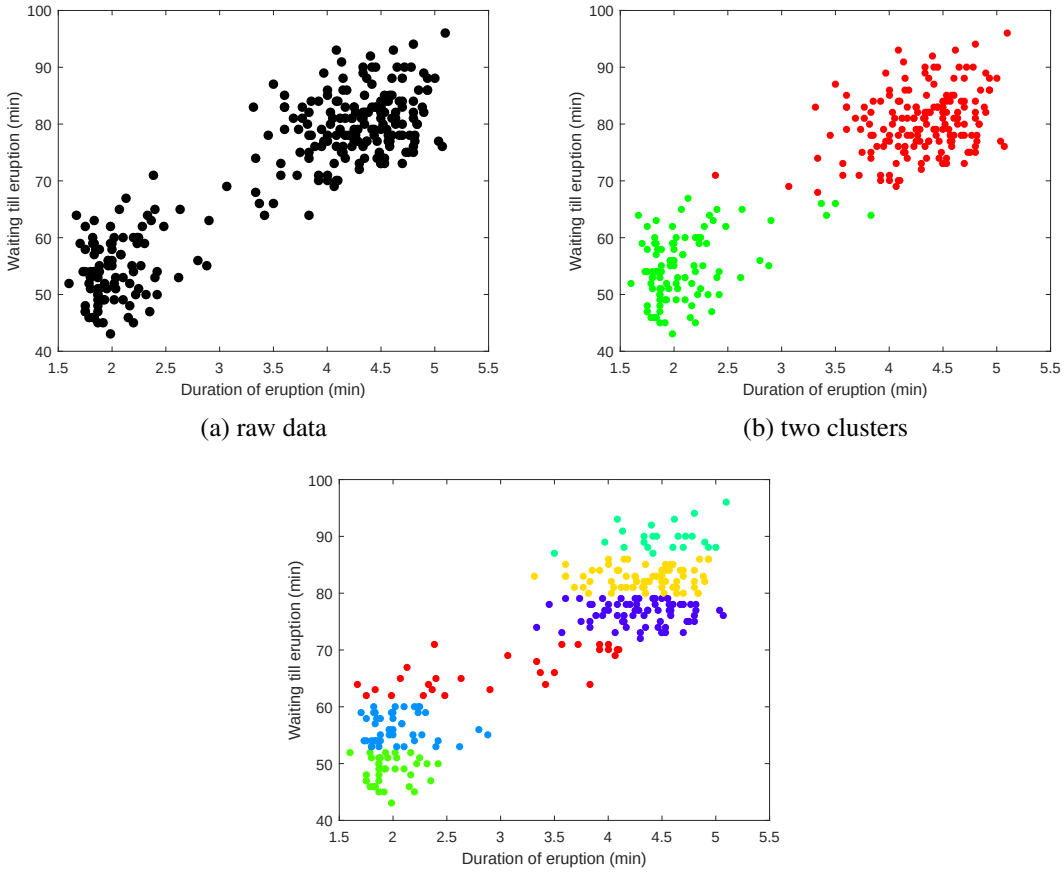


Figure 1.2: Matrix Scatter plot of variables in Combined Cycle Power Plant

1.1.2 Unsupervised learning

Unlike supervised learning, we are not told what the desired output is for each input. The goal is to discover interesting structure in the data. Unsupervised learning is arguably more typical of human and animal learning. It is also more widely applicable than supervised learning, since it does not require a human expert to manually label the data. Labeled data is not only expensive to acquire, but it also contains relatively little information, certainly not enough to reliably estimate the parameters of complex models. Some applications of unsupervised learning include clustering and anomaly detection. Clustering algorithms, in general, aims to find similar groups in a data while the aim of anomaly detection is to find unusual or anomalous cases. Examples are provided in the following [58].

Example 1.3. The Old Faithful data set. A data set with 272 observations on 2 variables, waiting time until the next eruption of the geyser (min) and the duration of the eruption (min) in Yellowstone National Park, Wyoming, USA. The raw data is shown in Figure 1.3.a and is freely available online. The aim is to cluster the observations into similar groups. Clustering results using k-means method for 2 and 6 clusters are shown in Figure 1.3.b and 1.3.c, respectively.



(a) raw data

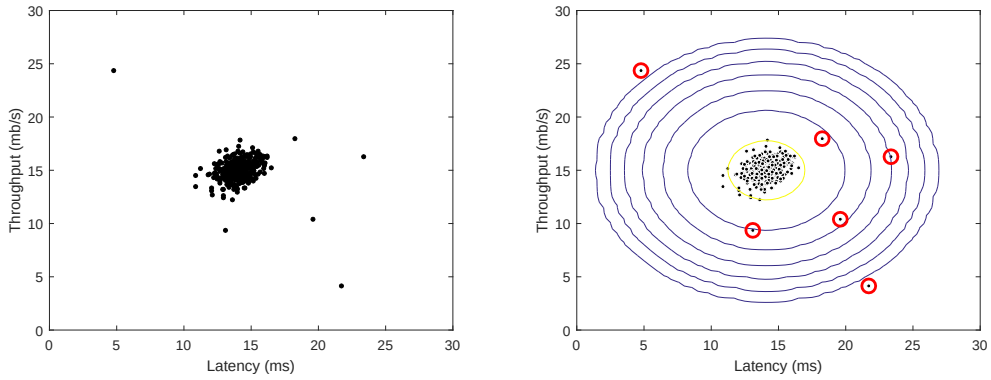
(b) two clusters

(c) six clusters

Figure 1.3: Clustering of waiting time until next eruption versus duration of eruption

Example 1.4. Anomalous Behavior in Server Computers. Figure 1.4 shows latency (ms) and throughput (mb/s) in 307 operating servers. We suspect that the vast majority of the examples are normal (non-anomalous) examples of the servers operating normally, but there might also be some examples of servers acting anomalously within this dataset. Gaussian distribution contours of the distribution fit to the dataset shows possible anomalous servers to investigate further. This data set is available at Machine Learning Course on Coursera [24].

By now, we know a good deal about different types of machine learning problems. On the next section we learn about biomedical signals and the ever increasing importance of machine learning in health care, specifically how classification can be used to automate the decision making process.



(a) raw data (b) possible anomalous servers

Figure 1.4: Raw server computers data and possible anomalous servers using estimated Gaussian distribution contours

1.2 Biomedical Signals

Biomedical signals are observations of physiological activities of organisms, ranging from gene and protein sequences, to neural and cardiac rhythms, to tissue and organ images [17]. Decades ago, the primary focus of biomedical signal processing was on filtering signals to remove noise [50, 64]. Sources of noise arise from imprecision of instruments to interference of power lines. Other sources are due to the biological systems themselves under study. Organisms are complex systems whose subsystems interact, so the measured signals of a biological subsystem usually contain the signals of other subsystems. Removing unwanted signal components can then underlie subsequent biomedicine discoveries. Another analysis framework derives from statistical signal processing in which the processing, e.g. Wiener filtering or Kalman filtering, utilizes statistical characterizations of the signals to extract desired signal components [23, 20].

Thanks to the development of various novel biomedical instruments, the field of biomedical signal processing continues to expand. The advancement of medical instruments such as ultrasound, magnetic resonance imaging (MRI), positron emission tomography (PET), electro and magnetoencephalogram (EEG/MEG), electrocardiogram (ECG), and many more has enabled practitioners to save millions of lives. Cellular imaging such as uorescence tagging and cellular MRI assists biologists monitoring the distribution and

evolution of live cells that is essential in detecting cancer [18]; diagnosis of patients with early-stage breast cancer is made possible using PET/CT scans [11]; early detection of Alzheimer's and Parkinson's disease is possible using EEG/MEG signal processing [42, 43]. These examples show how signal processing techniques has contributed significantly to the advancement of healthcare.

A recent emerging, important need in many biomedical experiments is classication [17]. In clinics for example, the goal of classication is to distinguish pathology from normal. For instance, monitoring physiological recordings of EEG, clinicians judge if patients suffer from illness [39]; watching cardiac MRI scans, cardiologists identify which region the myocardium experiences failure [19]; analyzing gene sequences of a family, geneticists infer the likelihood that the children inherit disease from their parents [16]. These examples illustrate that automatic decision making can be an important step in medical practice especially in the era of BIG data. In particular, incorrect disease identication will not only waste diagnosis resources but also delay treatment or cause loss of patients' lives.

However, classication in biomedical science faces several difculties, among them are the following.

1. Experts need to take a long time to accumulate enough knowledge to reliably distinguish between different related cases, say, normal and abnormal.

2. Manual classication among these cases is labor intensive and time consuming.

3. Sometimes the signal characteristics are not prominent and hence not easily discernible by experts.

Automated methods for classication in biomedical signals hold the promise for overcoming some of these difculties and to assist biomedical decision making. An automatic classier can learn from a database the categorial information, replace human operators, and classify indiscernible features without bias. To that extent, the focus of this dissertation is automated classification of biomedical signals, in particular brain and heart signals, to achieve some defined study goals. In the next two sub-sections we describe some important information and background study that is required to know before stating the main study goal.

1.2.1 Brain Signals

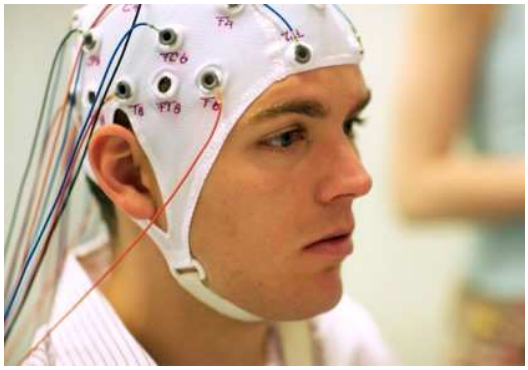
It is estimated that human brain has about 86 billion neurons in average [6] making up a complex integrated information processing and control system that regulates all the conscious and unconscious daily activities. The ability to recognize biological motion perceptions and physiological changes through brain signals is of great importance especially in Medical Diagnosis and Brain-Computer Interface (BCI) [3]. MEG (MagnetoEncephaloGraphy) and EEG (ElectroEncephaloGraphy) are non-invasive electrophysiological techniques for recording brain activity. EEG is based on measuring voltage sensed by an array of electrodes placed on the scalp. MEG is based on measuring the magnetic field outside the head using an array of very sensitive magnetic field detectors (See Figure 1.5). The signals recorded by EEG and MEG directly reflect current flows generated by neurons within the brain. The temporal frequency content of these signals ranges from less than 1 Hz (one cycle per second) to over 100 Hz (100 cycles per second).

Signals such as those illustrated in the Figure 1.5.d are recorded from each MEG sensor or EEG electrode pair. From MEG/EEG signals its easy to build a surface map reflecting ongoing brain activity, however, building 3D maps of brain activity from these signals requires the solution of an inverse source localization problem that finds the locations in the brain of sources of electromagnetic activity [94].

The EEG/MEG cannot pick the electrical potential generated by a single neuron. In fact, the EEG/MEG recording is a reflection of the summation of the synchronous activity of a big group of neurons, probably thousands or millions, having a similar spatial orientation. A similar spatial orientation is necessary for the ions to line up and create waves that will be strong enough to pass the detection threshold. It is known that the voltage fields fall off with the square of the distance, and hence the pyramidal neurons of the cortex that are well aligned and fire together are thought to produce most EEG/MEG signals. The brain activity from deep sources is difficult to detect as compared to the activity that happens near the skull [44].

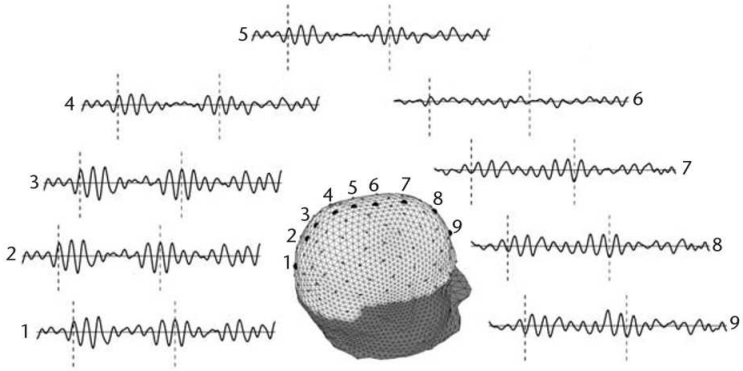
Neural oscillations are observed throughout the central nervous system. These are generated by large groups of neurons and can be characterized by the frequency, amplitude,

(a) MEG



(b) EEG with 32 channels

(c) EEG with 128 channels



(d) brain signals

Figure 1.5: Examples of MEG and medical grade EEG devices

and phase of the oscillations. Cognitive functions such as information transfer, perception, motor control, and memory are in one way or another related to neural oscillations and synchronization [44]. EEG recordings are commonly used to investigate neural oscillations. Neurons can generate action potentials or spikes in a rhythmic pattern. Some neurons have the tendency to fire at particular frequencies and are called resonators. Spiking patterns that are the result of bursting are considered fundamental for information coding. In many neurological disorders, the cause is excessive neural oscillation. For example, in seizures, excessive synchronization has been observed. Similar phenomena have been observed in tremor patients in Parkinson's disease. Oscillatory activity can also be used in BCIs to control external devices.

The human brain can produce five major brain waves, classified by their frequency ranges. These five major waves can range from low frequency to high frequency. These are known as alpha (α), theta (θ), beta (β), delta (δ), gamma (γ), and mu (μ). Berger, in 1929, discovered alpha and beta waves; gamma waves were discovered by Jasper and Andrews in 1938, while delta and theta waves were discovered by Walter in 1936, both of which represent waves with a frequency below the alpha range [10]. Below you can find more details about these frequency band signals.

- **Delta** waves fall within the range of 0.5-4.0 Hz. These waves have the highest amplitude among the other waves but have the lowest frequency. These occur frontally in adults and posteriorly in children. These waves are primarily associated with deep sleep. These delta rhythms may also be associated with subcortical lesions, deep midline lesions, or metabolic encephalopathy hydrocephalus. Figure1.6 shows delta waves in the interval 0 to 1 second.

- **Theta** waves fall in the range of 4-8 Hz. Theta waves are normally observed in young children. However, in older children and adults, these waves are observed in a state of drowsiness or arousal, as well as in meditation. These waves may occur anywhere and are not related to the tasks at hand. An excess of theta waves may represent abnormal activity.

- **Alpha** waves lie in the frequency range of 8-13 Hz. These waves were discovered by Dr. Hans Berger in 1908. Because these waves were the first to be discovered, they are called alpha waves (first waves). Alpha waves are associated with wakefulness, closing the eye, effortless alertness, and creativity. These waves normally appear in the posterior half of the head and have higher amplitude over the occipital areas.

- **Beta** waves lie in the frequency range 14-26 Hz. Found only in normal adults, these waves are correlated with active attention, active thinking, solving critical problems, or focusing on the outside world and, therefore, are also known as sensory motor rhythm. Rhythmical beta waves are experienced mainly in the frontal and central regions. Beta waves are low in amplitude and are normally under 30μV.

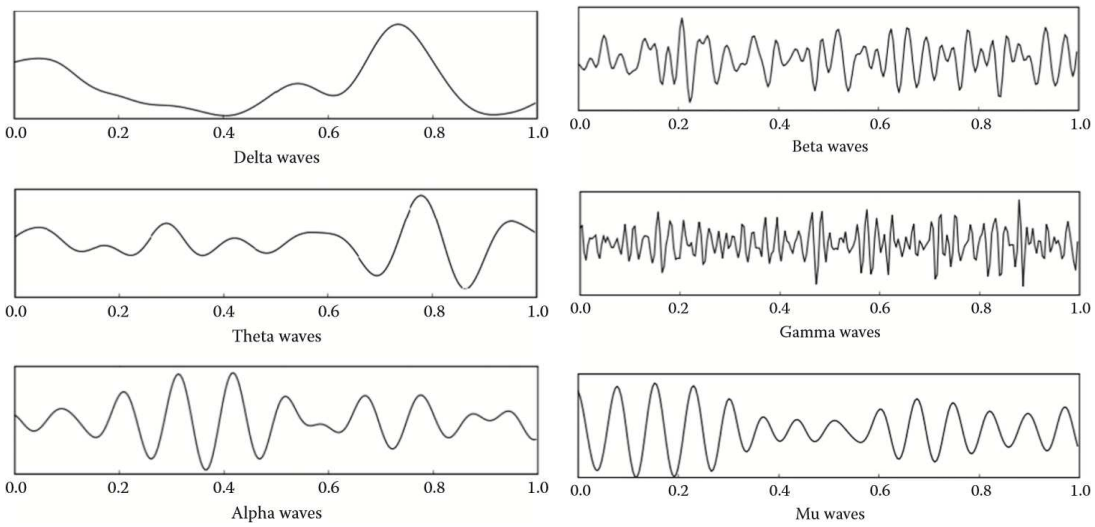


Figure 1.6: Different brain rhythms [103]

- **Gamma** waves lie in the ranges above 30Hz (up to 100Hz). These waves help to determine the binding of different populations of neurons together. They occur rarely in the human brain. They occur only during crossmodal sensory processing, that is, the process of combining different senses such as sound and sight.
- **Mu** wave ranges from 8 to 13Hz. These waves are mixed with other waves and sometimes partly overlap other rhythms. It shows the synchronous firing of motor neurons over the sensorimotor cortex. The different brain rhythms are shown in Figure1.6.

An EEG recording could be noninvasive or invasive. The noninvasive procedures use surface electrodes and hence are safe and painless. Human EEG can be measured using special electrodes with a typical diameter of 0.4cm to 1.0cm. The electrodes are held on the scalp with a paste (wet or dry), depending on how these electrodes are designed [103]. In some applications such as BCI and ERP studies, a minimal number of electrodes (usually 14) is often used and placed at the movement-related area or selection area in which the signals are strong using the conventional electrode positioning system. In advanced research or neural laboratories, more than 16 electrodes (often 64 to 258 recording electrodes) are used to get more detailed data. However, adding more electrodes may generate less useful data unless computer algorithms are used to manage the raw EEG data. Often, the potential at each location can be measured by taking the average of all the potential differences when

a large number of electrodes are used [103].

In chapter 2 we use the signals of the brain recorded in 14 different location on the scalp using an EEG device to detect real-time eye state change.

1.2.2 Heart Signals

Listening to the heart is perhaps the most important, basic, and effective clinical technique for evaluating a patient's cardiovascular function. A skilled practitioner can quickly evaluate common complaints that may be quite serious. With the advancements in technology and healthcare and assistance of noninvasive devices such as ElecroCardioGram (ECG) and PhonoCardioGram (PCG), a graphic recording of heart function can be obtained, leading to more objective analysis and interpretation. These portable electronic devices allow clinicians to apply both electro and phonocardiography more conveniently. They have also opened the possibilities for the application of advanced signal processing and data analysis techniques in the diagnosis of heart diseases [2].

In recent years, there has been a growing interest in applying machine learning techniques to aid in the detection, analysis and quantification of various aspects of heart signals. Machine Learning can be used to automate the measurement of various signal characteristics and diagnose many heart dysfunctions reducing subjectivity and increasing reliability [56]. Our goal is to develop algorithms to classify heart sound recordings collected from a variety of clinical or nonclinical (such as in-home visits) environments. The aim is to identify, from a single short recording (10-60s) from a single precordial location, whether the subject of the recording should be referred on for an expert diagnosis.

The main purpose of the heart is to pump blood through the body; it does so in a repeating sequence called the cardiac cycle. During the cardiac cycle, the heart firstly generates the electrical activity and then the electrical activity causes atrial and ventricular contractions. This in turn forces blood between the chambers of the heart and around the body. The electrical impulses in the heart produce electrical currents that flow through the body and can be measured on the skin using electrodes. This information can be observed using an electrocardiogram.

The human heart beats over 100,000 times per day. In each cardiac cycle, the heart contracts (systole), pushing out the blood and pumping it through the body (Figure 1.7.c). This is followed by a relaxation phase (diastole), where the heart fills with blood (Figure 1.7.a). The atria contract at the same time, forcing blood through the atrioventricular valves into the ventricles (Figure 1.7.b). Closing of the atrioventricular valves produces a monosyllabic ”lup” sound. Following a brief delay, the ventricles contract at the same time forcing blood through the semilunar valves into the aorta and the pulmonary artery (which transports blood to the lungs). Closing of the semilunar valves produces a monosyllabic ”dup” sound [92]. These vibrations are audible at the chest wall and recordable using phonocardiogram. Listening for specific heart sounds can give an indication of the health of the heart. Figure 1.8 illustrates a short section of a PCG/ECG recording.

Four locations are most often used to listen to the heart sounds, which are named according to the positions where the valves can be best heard:

- Aortic area, centered at the second right intercostal space.
- Pulmonic area, in the second intercostal space along the left sternal border.
- Tricuspid area, in the fourth intercostal space along the left sternal edge.
- Mitral area, at the cardiac apex, in the fifth intercostal space on the mid-clavicular line.

Fundamental heart sounds (FHSs) usually include the first (S1) and second (S2) heart sounds. S1 occurs at the beginning of iso-volumetric ventricular contraction, when the mitral and tricuspid valves close due to the rapid increase in pressure within the ventricles. S2 occurs at the beginning of diastole with the closure of the aortic and pulmonic valves. While the FHSs are the most recognizable sounds of the heart cycle, the mechanical activity of the heart may also cause other audible sounds, such as the third heart sound (S3), the fourth heart sound (S4), systolic ejection click (EC), mid-systolic click (MC), diastolic sound or opening snap (OS), as well as heart murmurs caused by the turbulent, high-velocity flow of blood.

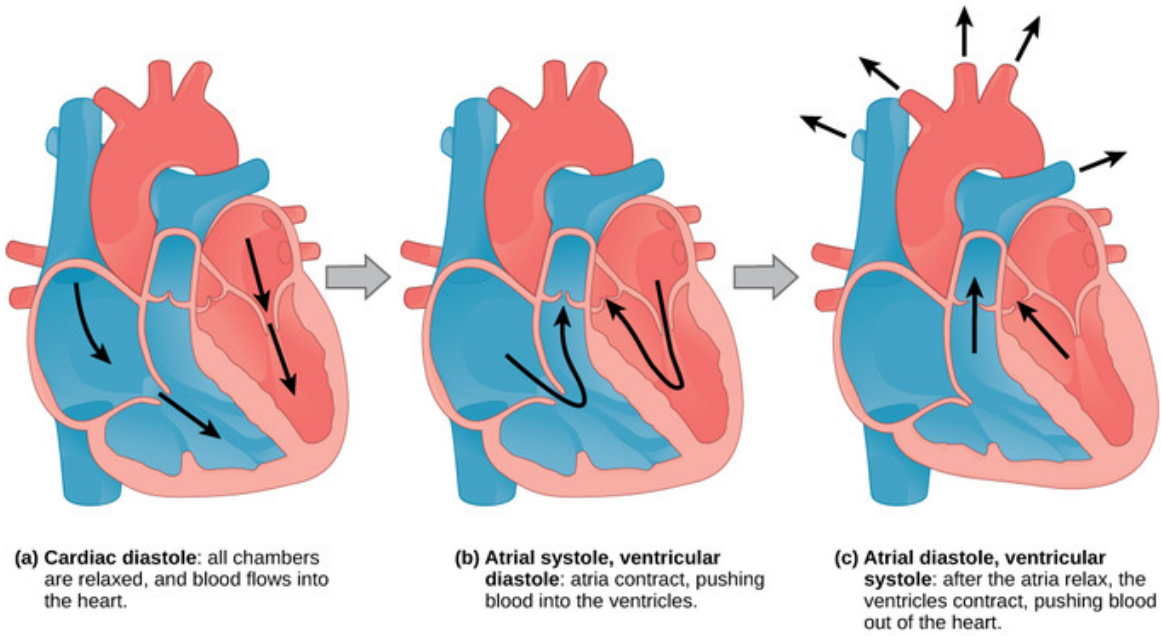


Figure 1.7: Diastole and Systole

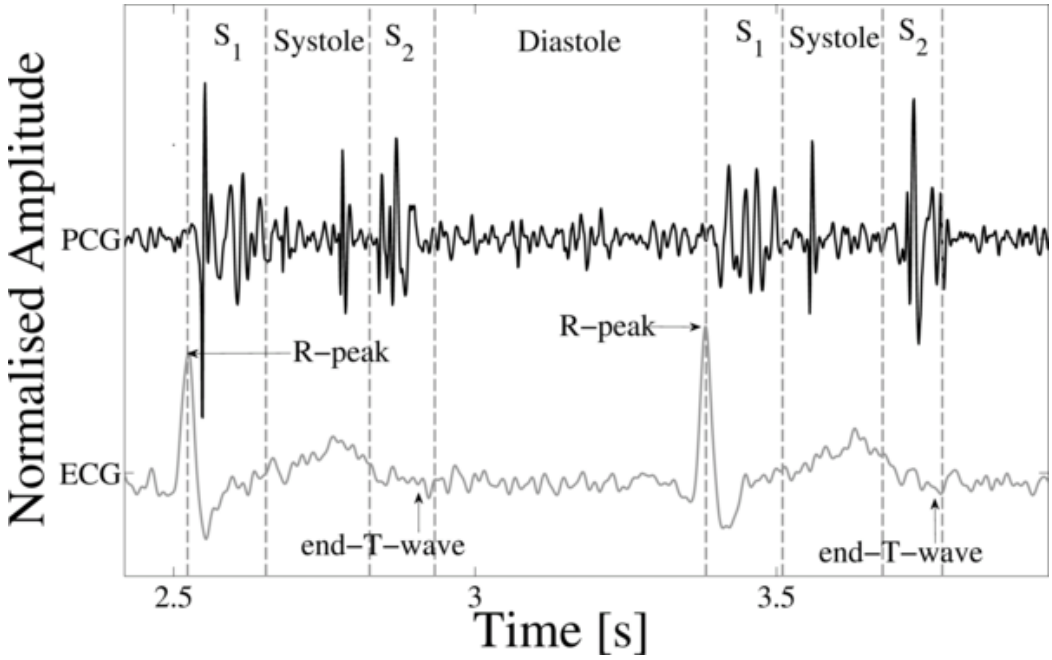


Figure 1.8: A PCG with simultaneously recorded ECG

The automated classification of ECG and PCG signals has been performed for over 50 years, but still presents challenges. Gerbarg et al. were the first researchers to attempt the automatic classification of pathology in PCGs using a threshold-based method [35], motivated by the need to identify children with rheumatic heart disease (RHD). Artificial Neural Networks (ANNs) have been the most widely used machine learning-based approach for heart activity classification. Typical relevant studies grouped by the signal features as the input to the ANN classifier include: using wavelet features [48], time, frequency and complexity-based features [82], and time-frequency features [27]. A number of researchers have also applied support vector machines (SVM) for heart sound classification in recent years. The studies can also be divided according to the feature extraction methods, including wavelet [4, 5], time, frequency and time-frequency feature-based classifiers [49]. Hidden Markov Models (HMM) have also been employed for pathology classification in PCG/ECG recordings [60, 77]. Clustering-based classifiers, typically the k-nearest neighbors (kNN) algorithm [9, 52, 62], have also been employed to classify pathology in PCG/ECGs. In addition, many other techniques have been applied, including threshold-based methods, decision trees, discriminant function analysis, and logistic regression [2].

Although, a number of the current studies for heart signal classification are flawed because of

- Good performance on carefully-selected data,
- Lack of a separate test dataset,
- Failure to use a variety of PCG/ECG recordings,
- Validation only on clean recordings,

In chapter 4, we propose our future work on accurate classification of normal and abnormal heart sound signals. We aim to utilize a dataset provided by PhysioNet in a recent challenge on the largest public collection of PCG recordings from a variety of clinical and nonclinical environments [101].

1.3 Methodologies

In this section we present some of the methods used throughout the dissertation with details. Examples are provided when necessary to demonstrate how methods work. Most of the codes to perform the techniques are written in MATLAB and are available either at the cited references or the appendix of the dissertation.

1.3.1 Classification

This section presents three classification algorithms that has been used in this dissertation in details. In classification problems, in general, the variable y that we would like to estimate is discrete valued. Some examples include deciding whether an email is spam/not–spam, wether an online transaction is fraudulent/not–fraudulent, or wether a tumor is malignant/benign. In all of these examples the variable y that we would like to estimate takes two values, either 0 (negative class, e.g. benign) or 1 (positive class, e.g. malignant). The assignment of two classes to positive or negative class is somewhat arbitrary but often the intuition is to assign a negative class in absence of a certain characteristic. There are cases where y takes more than two class but it is not discussed here.

To see how we develop a classification algorithm take a look at Figure 1.9. Figure 1.9.a shows training set examples for a classification task. The task is to find a decision boundary which classifies a tumor to malignant or benign using only one feature, tumor size. Figure 1.9.b shows the same example but with two input features, X_1 and X_2. Figure 1.9.c is the same example but a more complicated decision boundary is required to classify the tumor. The methods to achieve the goal of the task is presented in the following subsections.

1.3.1.1 Logistic Regression

Logistic regression is one of the most popular and most widely used classification algorithms. The term logistic regression rises from the fact that this method uses the sigmoid or logistic function $g(z) = \frac{1}{1+exp(-z)}$ in its core decision algorithm. One of the benefits of the

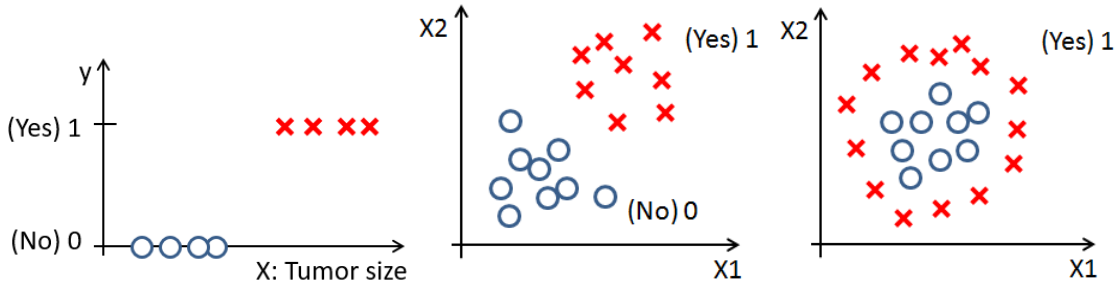
logistic function which makes it a good candidate for binary classification problems is that $0 \leq g(z) \leq 1$. Thus, by assigning a threshold for the value of this function one can decide whether a sample fits to class 0 or 1. An unbiased threshold of 0.5 is more commonly used than other values.

To formulate the task of finding a decision boundary that separates the two values of the response variable Y, suppose a set of m training samples $\{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{m}$ with n features $\mathbf{x}^{(i)} = (x_1^{(i)}, x_2^{(i)}, \ldots, x_n^{(i)})$ is given. Finding a decision boundary means estimating a set of parameters $\boldsymbol{\theta} = (\theta_0, \theta_1, \ldots, \theta_n)$ by minimizing a well defined cost function. Assume that the decision boundary is linear, i.e. it is in the form:

$$\boldsymbol{\theta}^T X = \theta_0 + \theta_1 X_1 + \cdots + \theta_n X_n. \tag{1.1}$$

We first define the hypothesis function $h_{\boldsymbol{\theta}}(\mathbf{x}) = g(\boldsymbol{\theta}^T \mathbf{x})$ where $g(z) = \frac{1}{1+exp(-z)}$. This function estimates the chance of a sample belonging to class 1, i.e. it estimates the probability $P(y = 1|\mathbf{x}; \boldsymbol{\theta})$. A reasonable cost function considers a zero cost for correct predictions and a positive penalty for incorrect predictions. More importantly, it is convex to guarantee a global minimum. The following cost function satisfies these conditions and is widely used in logistic regression classifiers:

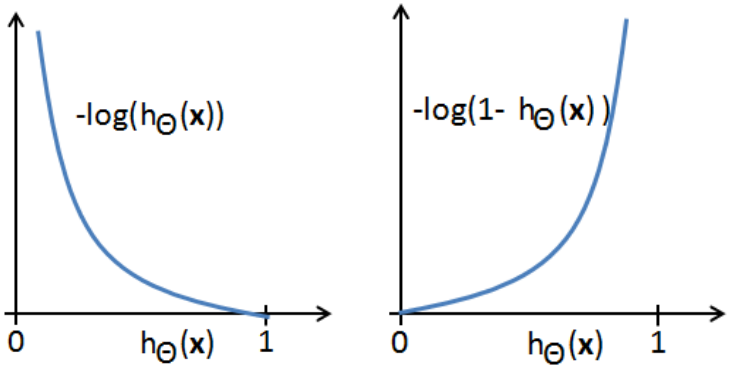
$$Cost(h_{\boldsymbol{\theta}}(\mathbf{x}), y) = \begin{cases} -\log(h_{\boldsymbol{\theta}}(\mathbf{x})) & \text{if } y = 1 \\ -\log(1 - h_{\boldsymbol{\theta}}(\mathbf{x})) & \text{if } y = 0 \end{cases} \tag{1.2}$$



(a) (b) (c)

Figure 1.9: Examples of training sets for a classification task

The intuition behind this cost function is straightforward as is shown in Figure 1.10. If $y = 1$ and we predict it correctly, that is we predict $h_{\boldsymbol{\theta}}(\mathbf{x}) = 1$, then $cost = 0$ as shown in the plot of the cost function. But as our prediction deviates from 1, $h_{\boldsymbol{\theta}}(\mathbf{x}) \to 0$, the cost increases $cost \to \infty$. On the other hand, if $y = 0$ and we predict $h_{\boldsymbol{\theta}}(\mathbf{x}) = 0$ then the cost is zero and the cost increases gradually by wrong prediction.



(a) if $y = 1$ (b) if $y = 0$

Figure 1.10: Logistic cost function

Putting these two cases together, the logistic regression classifier estimates the set of parameters $\boldsymbol{\theta}$ by minimizing the following average cost over all the m training samples:

$$\begin{aligned} J(\boldsymbol{\theta}) &= \frac{1}{m}\sum_{i=1}^{m} Cost(h_{\boldsymbol{\theta}}(\mathbf{x}^i), y^{(i)}) \\ &= \frac{-1}{m}\sum_{i=1}^{m} \left(y^{(i)} \log h_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}) + (1 - y^{(i)}) \log(1 - h_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}))\right) \end{aligned} \tag{1.3}$$

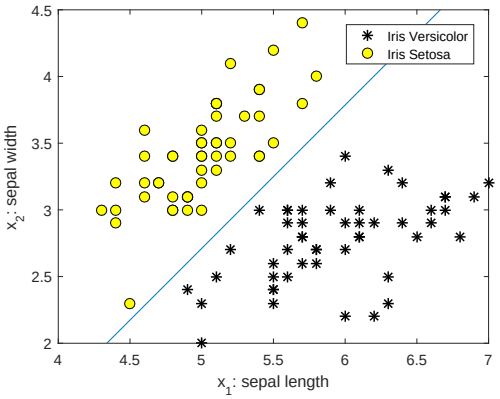
One of the many ways to minimize $J(\boldsymbol{\theta})$ is an iterative numerical technique known as gradient descent. Starting with an initial state $\boldsymbol{\theta}^{(0)}$ that is usually chosen randomly over the parameter space, the gradient descent simultaneously updates all θ_j through

$$\begin{aligned} \theta_j &:= \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\boldsymbol{\theta}) \\ &:= \theta_j - \alpha \sum_{i=1}^{m} (h_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}) - y^{(i)}) x_j^{(i)} \end{aligned} \tag{1.4}$$

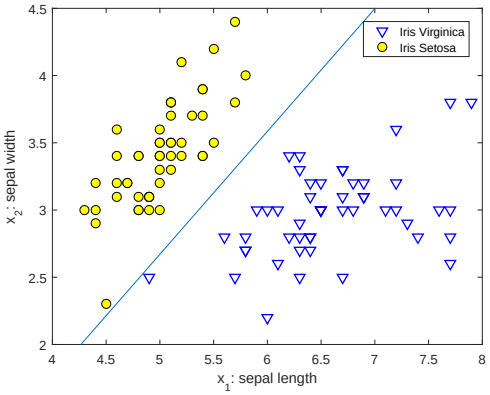
where α is a learning rate which is usually a constant value set to 0.5. The stopping criteria

for the algorithm is when the number of iterations reach a certain pre-specified limit, say 50, or when the difference between answers in two iterations is less than a pre-specified threshold, say 10^{-5}, whichever comes first.

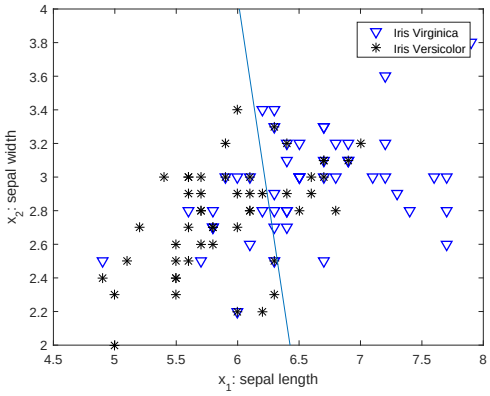
Example 1.5. Iris Data. As a toy problem to show how the logistic regression algorithm works we selected iris data in two features, sepal length and width. Figure 1.11.a shows the scatter plot and fitted decision boundary using logistic regression method for iris Setosa versus Versicolor. Figure 1.11.b shows the fitted decision boundary for iris Setosa versus Virginica. Figure 1.11.c shows the fitted decision boundary for iris Versicolor versus Virginica. See how two features can separate iris Setosa from Versicolor and Virginica but fail to efficiently discriminate Versicolor from Virginica. Gradient descent with zero values as the initial parameters has been used to estimate the class boundaries.



(a) $\hat{\boldsymbol{\theta}} = [-53.0, 21.4, -19.9]$



(b) $\hat{\boldsymbol{\theta}} = [-27.4, 13.2, -14.5]$



(c) $\hat{\boldsymbol{\theta}} = [-12.9, 1.9, 0.4]$

Figure 1.11: Iris data fitted boundaries using logistic regression

In many situations we have to add higher order interactions of features to achieve a non-linear decision boundary in logistic regression classifier. However, by increasing the dimension of the input features n, the dimension of the input features with quadratic terms increase by a factor of $n^2/2$. It is computationally expensive to find and represent all these features and it is not an efficient way to learn complex non-linear decision boundaries in logistic regression classifiers. Artificial Neural Networks are a much better way to learn complex non-linear decision boundaries when the input feature space is large.

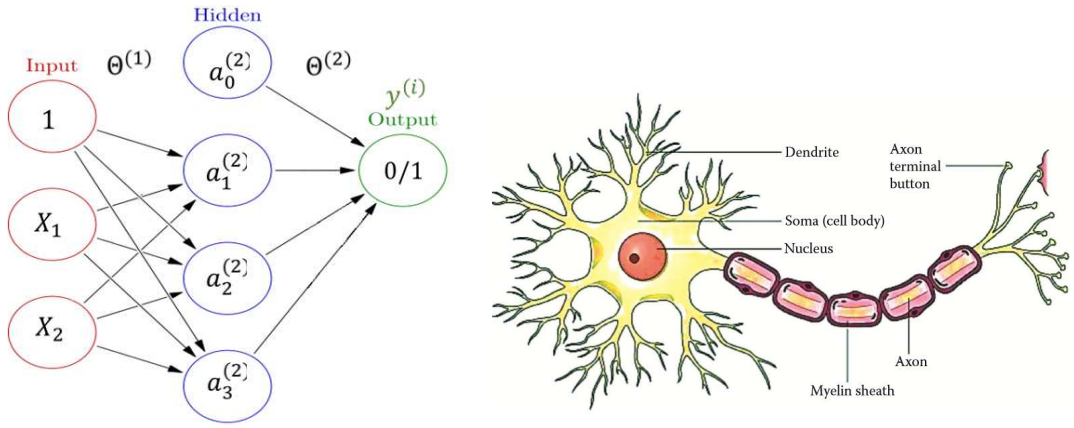
1.3.1.2 Artificial Neural Networks

The term 'neural network' has its origins in attempts to find mathematical representations of information processing in biological systems [55]. Figure 1.12 shows how a neural network resembles the information passing and processing in human brain. This figure shows a $2 \times 3 \times 2$ neural network. There is one input layer with two units, one hidden layer with three units, and a binary output. The layers may be described as

- Input layer: accepts the data vector or pattern.
- Hidden layer(s): They accept the output from the previous layer, weight them, and pass through a, normally, non-linear activation function.
- Output layer: takes the output from the final hidden layer, weights them, and possibly pass through an output, nonlinearity, to produce the target values.

The nature of the decision boundaries that may be produced varies with the network topology. However, any decision boundary can be approximated arbitrarily closely by a two layer (one hidden layer) network having sigmoidal activation functions [32]. Generally, the number of inputs is determined by the number of input dimensions. In choosing the number of hidden units one must consider that with too few hidden units, the network will not model complex decision boundaries and with too many, the network will have poor generalisation. The number of output units is determined by the number of output classes.

Let us go through some notations to explain the specific computation represented by a neural network. We only show the computations for a network with one hidden layer.



(a) A sample neural network

(b) A human neuron

Figure 1.12: Network diagram for a two-layer neural network and its resemblance to a neuron

Suppose we have a set of m training samples $(\mathbf{x}, y)$ with n features $\mathbf{x} = (x_1, x_2, \ldots, x_n)$. To find a decision boundary, we have to estimate matrices of parameters $\Theta^{(1)}$ and $\Theta^{(2)}$ where $\Theta^{(j)}$ is matrix of weights controlling function mapping from layer j to layer $j+1$. Suppose the hidden layer has K units. The information processing step in unit i ($i = 1, \ldots, K$) of hidden layer is shown by $a_i^{(2)}$ that is called activation of unit i in layer 2 and is defined as

$$a_i^{(2)} = g(\Theta_{i0}^{(1)} + \Theta_{i1}^{(1)} x_1 + \cdots + \Theta_{in}^{(1)} x_n), \tag{1.5}$$

where $g(.)$ is usually the sigmoid activation function and $a_0^{(2)} = 1$. Similar to logistic regression, class prediction is made via setting a threshold on the hypothesis function computed in the output layer as

$$h_{\mathbf{\Theta}}(\mathbf{x}) = a_1^{(3)} = g(\Theta_{10}^{(2)} a_0^{(2)} + \Theta_{11}^{(2)} a_1^{(2)} + \cdots + \Theta_{kn}^{(2)} a_k^{(2)}), \tag{1.6}$$

where it estimates the chance of a sample belonging to class 1, i.e. $P(y = 1|\mathbf{x}; \mathbf{\Theta})$. Given a set of initial values for the parameters $\Theta^{(1)}$ and $\Theta^{(2)}$, the process of computing $h_{\mathbf{\Theta}}(\mathbf{x})$ is called forward propagation. Backward propagation on the other hand includes the process of updating the initial parameters via an iterative method to minimize the following average

cost of the m given samples

$$J(\mathbf{\Theta}) = \frac{-1}{m}\sum_{i=1}^{m}\left(y^{(i)}\log h_{\mathbf{\Theta}}(\mathbf{x}^{(i)}) + (1-y^{(i)})\log(1-h_{\mathbf{\Theta}}(\mathbf{x}^{(i)}))\right). \tag{1.7}$$

The average cost looks the same as the logistic regression average cost but if one looks at equation (1.3) realizes that the equation (1.7) is more complex and that more parameters are now included. Backward propagation is a complicated process resembling a black-box. The intuition behind the steps is that

1. We move from the output layer to input and compute error terms. That is, starting from the output layer, we compute an error term, $\delta^{(3)} = h_{\mathbf{\Theta}}(\mathbf{x}^{(i)}) - y^{(i)}$. Then, we propagate the error in the hidden layer via the following in which $.*$ is an inner product of two vectors.

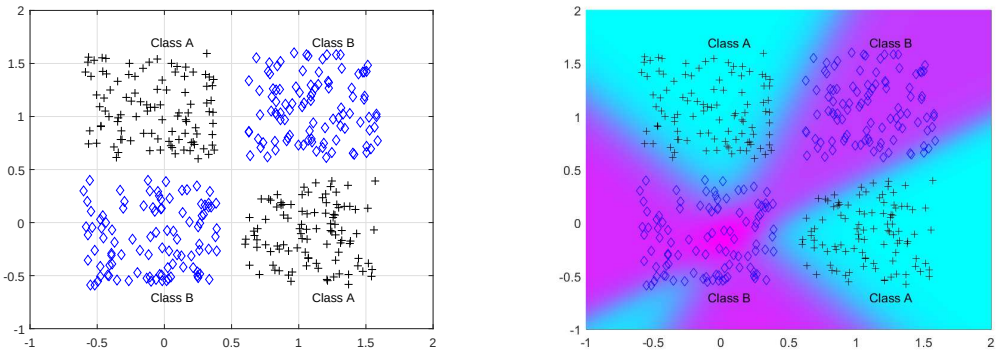
$$\delta^{(2)} = (\Theta^{(2)})^T\delta^{(3)}. * g\prime(\Theta_{i0}^{(1)} + \Theta_{i1}^{(1)}x_1 + \cdots + \Theta_{in}^{(1)}x_n). \tag{1.8}$$

2. We use the computed error terms $\delta^{(l)}$ in conjunction with gradient decent to iteratively update the parameters toward achieving a minimum average cost using

$$\begin{aligned}\Theta_{ij}^{(l)} &:= \Theta_{ij}^{(l)} - \alpha\frac{\partial}{\partial\Theta_{ij}^{(l)}}J(\mathbf{\Theta}) \\ &:= \Theta_{ij}^{(l)} - \alpha\ a_j^{(l)}\delta_i^{(l+1)}\end{aligned} \tag{1.9}$$

Similar to logistic regression, the process of forward and backward propagation is repeated until the algorithm reaches a pre-specified iteration limit or improvement threshold.

Example 1.6. XOR problem. Four clusters of data are defined in a 2-dimensional input space, see Figure 1.13. A and B clusters represent Exclusive OR (XOR) classification problem (XOR is a logical operation that outputs true only when inputs differ). The task is to define a neural network to classify the XOR problem [65]. Notice that the decision boundary in this example is not linear and simple binary logistic regression classifier fails to efficiently classify the XOR problem. However, a simple $2 \times 3 \times 2$ Neural Network successfully classifies the four clusters of data.



(a) Classical XOR problem with two class (b) Decision boundary using using ANN

Figure 1.13: Complex decision boundary using a $2 \times 3 \times 2$ Neural Network

Covering up the input layer and its matrix of parameters in Figure 1.12, we see that the remaining neural network is the same old logistic regression but instead of the initial input data it uses a weighted set of non-linear transformations of input learned through examples. This helps in computing more complex features of the input for the task of predicting the class output.

1.3.1.3 Support Vector Machine

In practice, we are usually faced with non-linear class boundaries and clearly simple logistic regression will perform poorly in such situations. In that case, we could consider enlarging the feature space using functions of the predictors, such as quadratic and cubic terms, in order to address this non-linearity. Support vector classifiers, could address the problem of possibly non-linear boundaries between classes in a similar way, by enlarging the feature space using kernel trick [14]. Every kernel holds a non-linear kernel function that helps to build a high dimensional feature space. Some standard kernels are

- Homogeneous Polynomial Kernel, can be represented by the expression $k(x_i, x_j) = (x_i, x_j)^d$, where $k(.)$ is the kernel function, x_i and x_j are two feature vectors and d is the degree of the polynomial function.
- Non-homogeneous Polynomial Kernel, given by $k(x_i, x_j) = (x_i^T x_j + c)^d$ where c is a constant term. The constant term, also known as a free parameter influences the

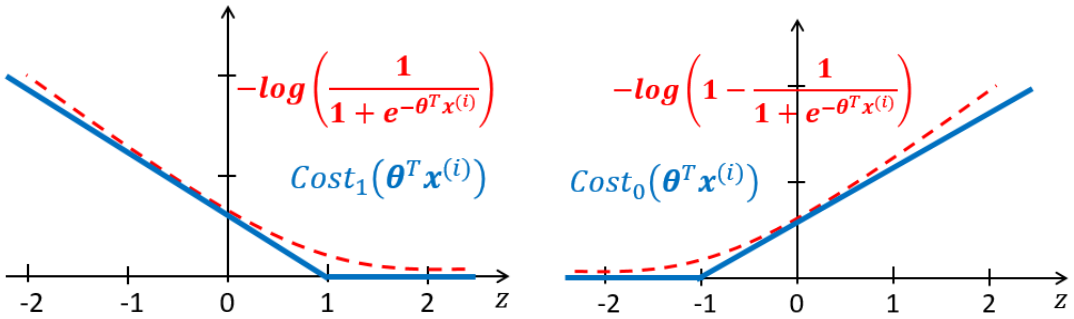
combination of features x_i and x_j.

- Radial Basis Function Kernel, also known as RBF kernel is one of the most popular kernels. For distance metric, squared Euclidean distance is used as $k(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$, where σ is a free parameter. Selection of parameters is a critical choice as using a typical value of the parameter can lead to overfitting.

However, the computations would become unmanageable in logistic regression by increasing the dimension of the feature space. The support vector machine, allows us to enlarge the feature space used by the support vector classifier in a way that leads to efficient computations. The trick is how it defines a simplified cost function and regularization term which prevents overfitting. If you look at the cost function for logistic regression, you see that each example contributes to the cost according to the following equation:

$$\begin{aligned} Cost(h_{\boldsymbol{\theta}}(\mathbf{x}^i), y^{(i)}) &= -\left(y^{(i)} \log h_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}) + (1 - y^{(i)}) \log(1 - h_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}))\right) \\ &= -y^{(i)} \log(\frac{1}{1 + e^{-\boldsymbol{\theta}^T \mathbf{x}^{(i)}}}) - (1 - y^{(i)}) \log(1 - \frac{1}{1 + e^{-\boldsymbol{\theta}^T \mathbf{x}^{(i)}}}) \end{aligned} \tag{1.10}$$

Following the computations in Figure 1.14, we can see that when $y = 1$ the cost is a decreasing logarithmic function of the term $z = \boldsymbol{\theta}^T \mathbf{x}^{(i)}$ that assigns a small cost for large values of z. What Support Vector Machine does is that it replaces the smooth decreasing cost with a bi-linear obtuse function that assigns a zero for cost when $z \geq 1$. If $y = 0$, the SVM cost is a bi-linear obtuse function that assigns a zero for $z \leq -1$.



(a) when $y = 1$ (b) when $y = 0$

Figure 1.14: Comparison of Cost function for logistic regression (dashed) and SVM (solid)

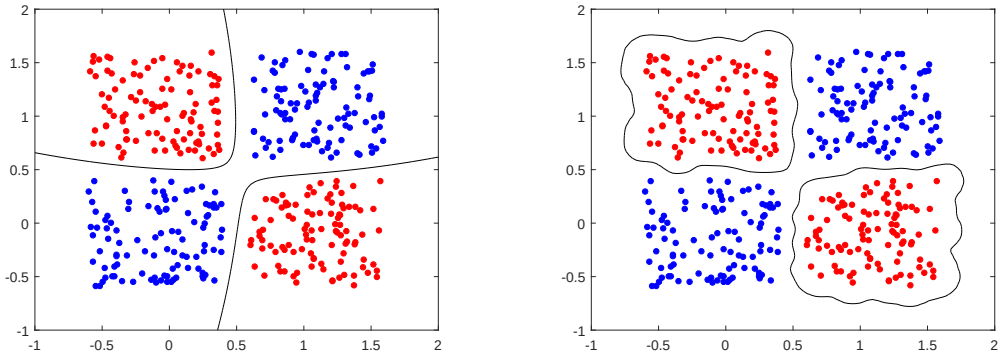
This approach gives computational advantage to SVM classifiers. Overall, the average cost for SVM classifier is written as

$$J(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)}\, cost_1(\boldsymbol{\theta}^T\mathbf{x}^{(i)}) + (1-y^{(i)})\, cost_0(\boldsymbol{\theta}^T\mathbf{x}^{(i)})\right) + \frac{\lambda}{2m}\sum_{j=1}^{N}\theta_j^2, \qquad (1.11)$$

where N is the size of the new feature space and $\lambda \geq 0$ is a tuning parameter that controls the new added regularization term. When $\lambda = 0$, the penalty term has no effect, and all the features are included in the model. However, as $\lambda \to \infty$, the impact of the shrinkage penalty grows, and the features with smaller coefficients and lesser impacts will be removed from the model. Apparently, selecting a good value for λ is critical and it is usually set through cross-validation.

Similar to the previous classification methods, an iterative optimization technique is used to estimate the set of model parameters that leads to estimating the decision boundary. The regularization term is usually added to logistic regression and artificial neural network to control model over-parametrization.

Example 1.7. XOR problem. We refer back to our XOR problem and this time we use two SVM classifiers to find optimal decision boundaries. The first SVM uses Polynomials of order three with $\lambda = 1$ and the second SVM uses RBF kernel with $\lambda = 1$ and $\sigma = 0.1$. The decision boundaries are shown in 1.15.



(a) Polynomial kernel with $d = 3$, $\lambda = 1$

(b) RBF kernel with $\lambda = 1, \sigma = 0.1$

Figure 1.15: Two SVM classification boundaries

While SVM seems to be a better choice in this example, it might not always provide the best decision boundaries. The biggest limitation of the support vector approach lies in choice of the kernel. A second limitation is speed and size, both in training and testing which they can make it abysmally slow in test phase. Discrete data presents another problem [15].

1.3.2 Classification Accuracy

After building a classification model using a training set, one always want to find a measure of model goodness by looking at its accuracy as the number of correct predictions in a test set, that is

$$\text{Accuracy} = \frac{\#\text{of correct predeictions in the test test}}{\text{total number of test set samples}} \times 100. \tag{1.12}$$

However, accuracy alone is typically not enough to show model goodness. The breast cancer dataset is a standard machine learning dataset to showcase this point [103]. The dataset contains 9 attributes describing 286 women that have suffered and survived breast cancer and whether or not breast cancer recurred within 5 years. Of the 286 women, 201 did not suffer a recurrence of breast cancer, leaving the remaining 85 that did. A model that only predicts no recurrence of breast cancer would achieve an accuracy of $(201/286) * 100 = 70.28\%$. This is a high accuracy but a terrible model as it would send home 85 women with incorrectly thinking their breast cancer was not going to reoccur (high False Negatives).

Thus, we should look for alternative evaluation methods that are valid even in skew class sizes. A clean and unambiguous way to present the prediction results of a classifier is to use a confusion matrix. As Figure 1.16 shows, it is a 2×2 table for a binary classification problem. Each cell contains the number of predictions made by the classifier that fall into that cell. In breast cancer case, a perfect classifier would correctly predict 201 no recurrence and 85 recurrence which would be entered into the top left cell no recurrence/no recurrence (True Negatives) and bottom right cell recurrence/recurrence (True Positives). However, the confusion matrix itself can be ambiguous and misleading. It is always good to have a number that evaluates the goodness of a classifier. Two important evaluation measures are

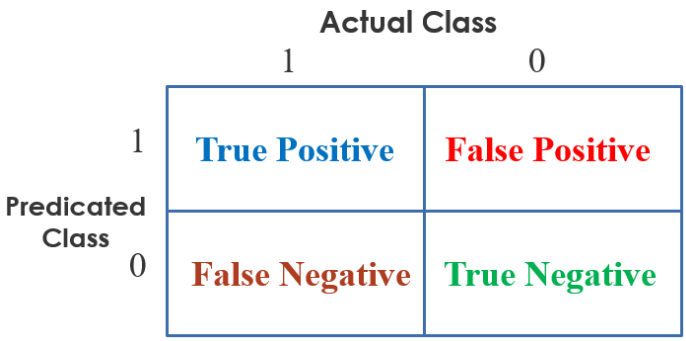


Figure 1.16: Elements of a confusion matrix

know as Precision and Recall that are defined as

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive + False Positive}} \times 100. \tag{1.13}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive + False Negative}} \times 100. \tag{1.14}$$

Obviously, having a high precision and recall is desirable in classification models. By computing precision and recall we have a better sense of how our classifiers are doing. In particular, if we have an algorithm that predicts $y = 0$ all the time, then this classifier will have a recall of 0 that is a quick way to recognize its efficiency. Overall, a classifier with high precision and recall is a good classifier.

If we were looking to select a model based on a balance between precision and recall however, the following measure known as F-measure or F-score conveys a balanced value between the precision and the recall

$$\text{F-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision + Recall}} \times 100. \tag{1.15}$$

1.3.3 Cross validation

Residual evaluations do not provide an indication of how well a learner algorithm will perform in light of new data. One way to overcome this problem is to keep some of the data in a storage and to not use the entire data set for training a learner. Then, when training is done, the data that was removed can be used to test the performance of the learned model on ”new” data. This is the basic idea for a whole class of model evaluation methods called cross validation. The following are some of the many ways the cross validation is utilized in machine learning algorithms [12].

- **Holdout method.** In this approach, the data set is separated into two sets, called the training set which usually includes 70% of the data and the testing set which includes the remaining 30%. The machine learning algorithm fits a model using the training set only, then it predicts the output values for the data in the testing set that has never seen. The model efficiency is evaluated using the mean absolute test set error. This approach is preferable to the residual method and takes no time to compute. However, its evaluation can have a high variance. Moreover, the evaluation depend heavily on which data points end up in the training set and which end up in the test set, and thus the evaluation may be significantly different depending on how the division is made. Random selections is performed to remedy this issue.

- **K-fold Cross Validation.** One way to improve the holdout method is to use k-fold cross validation, commonly 5 or 10-fold. Here, the data set is divided into k subsets, and the holdout method is repeated k times. Each time, one of the k subsets is used as the test set and the other k-1 subsets are put together to form a training set. Then the average error across all k trials is computed. One advantage of this method is that it matters less how the data gets divided. Every data point will be in a test set exactly once, and will be in a training set k-1 times. The variance of the resulting estimate is reduced as k is increased. The disadvantage of this method is that the training algorithm has to be rerun from scratch k times, which means it takes k times as much computation to make an evaluation. A variant of this method is to randomly divide

the data into a test and training set k different times. The advantage of doing this is that you can independently choose how large each test set is and how many trials you average over.

- **Leave-one-out method.** The K-fold cross validation is called Leave-one-out when it is performed for k= N, N being the total number of data points. As before, the average error is computed and used to evaluate the model. The evaluation given by leave-one-out cross validation error (LOO-XVE) is good, but at first pass it seems very expensive to compute. However, learner algorithms can make LOO predictions just as easily as they make regular predictions.

CHAPTER 2

RANDOM EYE STATE CHANGE DETECTION IN REAL-TIME USING EEG SIGNALS

In this chapter we propose two analytic methods to detect the eye state change in real-time using Electroencephalogram signals. One article is published based on findings of this chapter [74] and one is under revision in a reputable peer-reviewed journal [75].

2.1 Introduction

The average human brain is estimated to have about 86 billion neurons [6] making up a complex integrated information processing and control system that regulates all the conscious and unconscious daily decisions. The ability to recognize biological motion perceptions and physiological changes through brain signals has been studied for many reasons especially medical diagnosis and Brain-Computer Interface (BCI) [3]. Electroencephalogram (EEG) provides a non-invasive tool to record the brain's electrical potential along the scalp at multiple locations over the scalp [34]. It measures voltage fluctuations resulting from ionic current flows within the neurons of the brain. Discriminating between the open and closed state of the eyes using EEG signals in real-life situations is a challenging research goal not only crucial to medical care but also significant to several daily life tasks.

The significance of eyes state identification using EEG signals is shown through related studies in scientific literatures. Some of the applications include automated classification of sleep-waking states in infants [28] and hospitalized patients, stress feature identification [59], human-computer interface design [46], alertness of pilots especially fighter jet

pilots, and detection of driving drowsiness [54, 33, 25] where it estimated to be responsible for at least 72,000 crashes, 44,000 injuries, and 800 deaths in 2013 [93]. Various methods have been used in literatures to achieve higher classification accuracy; among them are embedded hidden Markov models [40], time series classification [89], pattern recognition [28], and machine learning methods [70, 73]. In this research innovation a fully data-driven method which can handle non-linear, non-stationary, and multivariate signals is fused with spatial filtering and common machine learning classification methods to achieve a fast real-time classification analytics with a very high prediction accuracy.

Aside from neural activities, EEG captures potential fluctuations of non-neural origins such as technical and biological artifacts. DC noise, power line, improper placements, and high impedance of EEG electrodes create technical artifacts. Eye movements and blinks, muscle contractions, and heart activity are among the biological artifacts that a person creates. Eye blink artifacts are generated by the movement of the eyelid along the cornea. By friction between lid and cornea, this movement results in charge separation, with a dominant dipolar charge distribution, and the dipole moment pointing in up-down-direction. In the EEG, this effect is recorded as a positive peak that lasts a few tenths of a second, is most visible in the frontopolar region, but propagating to all the electrodes of the montage, becoming weaker with distance from the frontal part. The frequency bands that are more affected by this artifact are δ (0.5-4.0 Hz) and θ (4-8 Hz), with the main portion of energy below the 5 Hz [81]. Among neural activities, α band signals that are associated with wakefulness, closing the eyes, effortless alertness, and creativity are triggered normally in the posterior half of the brain in the frequency range of 8-13 Hz [44].

The dataset used in this study is publicly available via UCI Machine Learning Repository [103]. It was recorded by Emotiv EPOC headset (see Figure (2.1)) that records brain signals with 128 Hz sampling frequency [95] in 14 locations on the scalp. Using this dataset, Rösler and Suendermann [70] have applied various instance based classification methods and achieved 97.3% accuracy using R-Star algorithm. But their algorithm took at least 2 hours to train and 20 minutes to predict the state of the eye for a new instance. In another study, Wang et al. [9] have extracted standard deviation and average of channels as

new features and used Incremental Attribute Learning to achieve an accuracy of 76.6% for eye state classification. They did not report the running time of their procedure. In a more recent study, Sabancı and Koklu [73] have employed k-Nearest Neighbors and multilayer perception neural networks algorithm to classify the eye state. Their highest achieved success rate was 84.05% by three nearest neighbors' method yet the runtime was not reported. Considering the data recording speed, these three studies follow slow and computationally intensive instance based algorithms. Besides, the nature of the experiment in which the eye state randomly changes in time, does not require going through classification procedures for every instance.

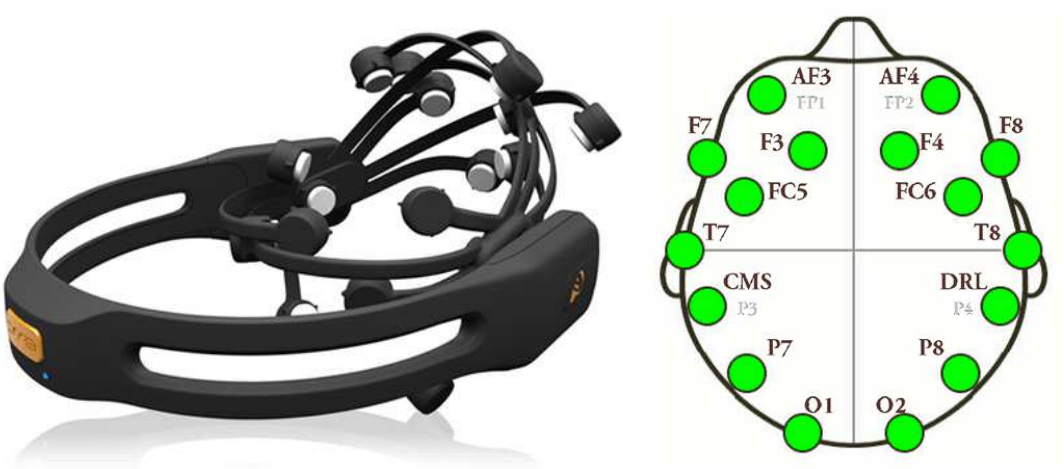


Figure 2.1: Emotiv EPOC neuroheadset and its channel locations

In our research, we employ cross-channels maximum and minimum to monitor recording instances in real time. Upon detection of a possible change, Multivariate Empirical Mode Decomposition analytics are applied to the last two seconds of the signal to extract narrow-band Intrinsic Mode Functions. Then, selected Intrinsic Mode Functions are passed through Common Spatial Pattern to extract relevant features. Finally, features are fed into classification analytics to decide whether a change in the eye state has occurred. Furthermore, we propose a second procedure that uses analytical statistical features from Intrinsic Mode Functions which is then feeded directly to the classification analytics. Both procedures take less than two seconds to predict a change in the eye state and could be employed in real-time eye state detectors.

2.2 Data Cleaning and Exploration

The EEG eye state corpus from UCI Machine Learning Repository created by Frank and Asuncion in 2010 [103] is utilized in our research. The dataset was recorded using Emotiv EPOC headset in an experiment conducted in 2013 by Rösler and Suendermann [70] from Baden-Wuerttemberg Cooperative State University (DHBW), Germany. During the experiment which was conducted in two sessions of one to two minutes each, a participant closed/opened his/her eyes at will. Brain signals recorded from 14 different locations on the scalp namely AF3, AF4, F3, F4, F7, F8, FC5, FC6, T7, T8, P7, P8, O1, and O2 at 128 Hz sampling frequency. The eye state added to the data after investigating a camera feed which was focused on the eyes. After dumping the first few seconds of the recording, only 117 seconds of the signals, a total of 14,980 instances in 15 attributes, were made available online at UCI machine learning repository [103].

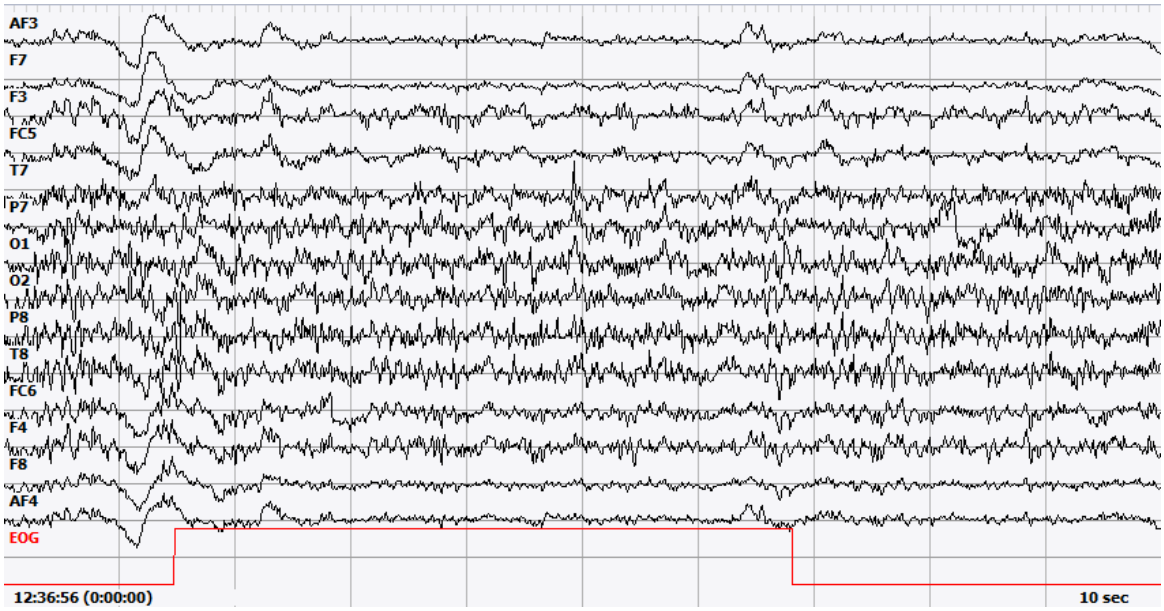


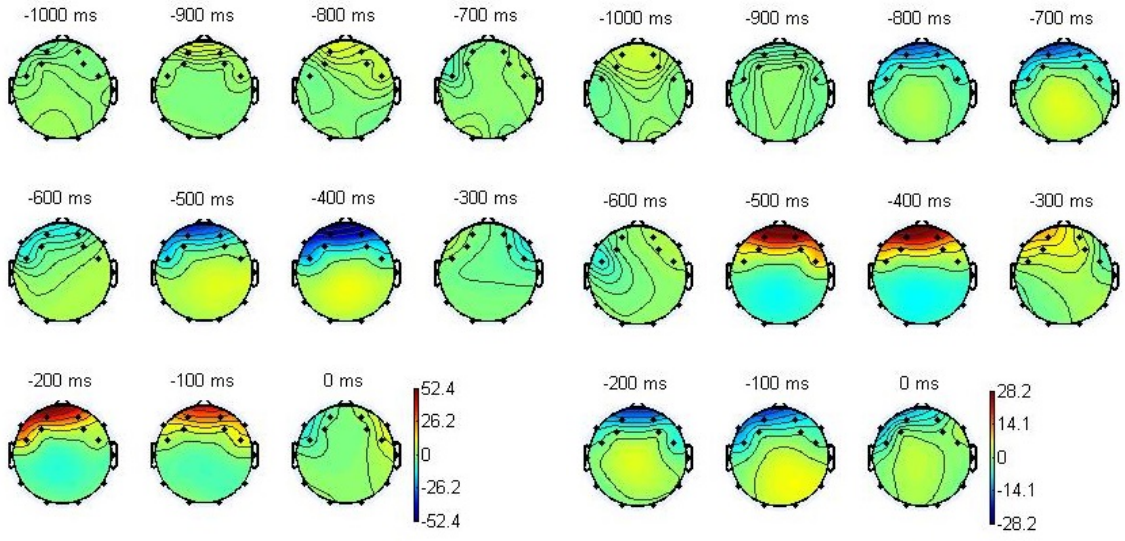
Figure 2.2: First 10 seconds of the filtered signal for all channels plus eye state (redline)

The performance of the EPOC headset in this specific task has been compared to medical grade BrainAmp Standard device by Rösler et al. [71] and the difference was insignificant. In comparison to the BrainAmp Standard, the EPOC headset has a significantly lower price, a higher usability due to its wireless connection, and a faster and easier setup

which emphasizes its advantageous applicability.

Upon visual inspection of the dataset, instances 899, 10387, 11510, and 13180 have been removed from the data since they tackled an outlier in some of the channels. A high-pass Butterworth filter of 0.5 Hz is then used to remove the DC offset. Figure 2.2 shows the first 10 seconds of the normalized signal which includes closing eye (1) and opening eye (0) events. The rhythmic activity of Event-Related Synchronization (ERS) and Event-Related De-synchronization (ERD) is detectable by eye right before changing eye state in most of the channels. The cleaned signal will be shown by $X^{ch \times t}$ hereinafter where $ch = 14$ is the number of channels and $t = 14,976$ is the time instances.

A total of 12 closed eyes and 11 open eyes intervals with various lengths are available in the dataset. After cutting these 23 intervals into one-second epochs right before change of the eye state, the average amplitude of epochs for each eye state were computed and localized using EEGLAB [26]. Some of the intervals were not disjoint. Figure 2.3 shows the scalp map of the brain activity for the average amplitude over epochs. ERS/ERD is noticeable especially in the frontal part of the head where channel AF3 and AF4 are located. This charge is due to muscle activity.



(a) before closing eyes (b) before opening eyes

Figure 2.3: Scalp maps for average epochs

2.3 Methods

2.3.1 Monitoring and Change Detection

Considering the sampling frequency of 128 Hz, each second 128 instances are recorded in 14 channels. Running instance-based classification algorithms is slow and computationally intensive. Instead, we propose monitoring the instances in real time until a possible change of the eye state is observed. After going through some statistical characteristics of the channels such as variance, average, minimum, maximum, range and midrange, the maximum and minimum computed across channels defined by

$$\max(t) = \max_{ch}(X^{ch \times t}), \tag{2.1}$$

and

$$\min(t) = \min_{ch}(X^{ch \times t}), \tag{2.2}$$

were selected as monitoring functions. This selection is compatible with the results of Figure 2.3 where the signals amplitude decreases then increases significantly right before closing eyes (a) and it increases then drops significantly right before opening eyes (b). Figure 2.4 illustrates that the eye state change happens in synch with significant increase of both (2.1) and (2.2). This increase is more notable in changing the eye state from open to closed. There are also a few noticeable peaks in monitoring functions when the eyes are open which are probably due to unregistered blinking.

There is no need to compute the monitoring (2.1) and (2.2) for every instance as every four instances are good enough to detect a change of eye state efficiently. That is, a 32 Hz input signal for the monitoring performs well. This will reduce the computational cost for the penalty of losing the exact instance the change happens. However, considering the speed of recording instances and the nature of the experiment, the gain is worth the cost.

The eye state change time is random since the participant closes/opens the eyes without further notice. The change of eye state affects the brain signals due to eye muscle movement artefact and alpha band signal activity. The monitoring $\max(t)$ and $\min(t)$ given by (2.1) and (2.2) computed at 32 Hz can pick up this change, thus applying a threshold for

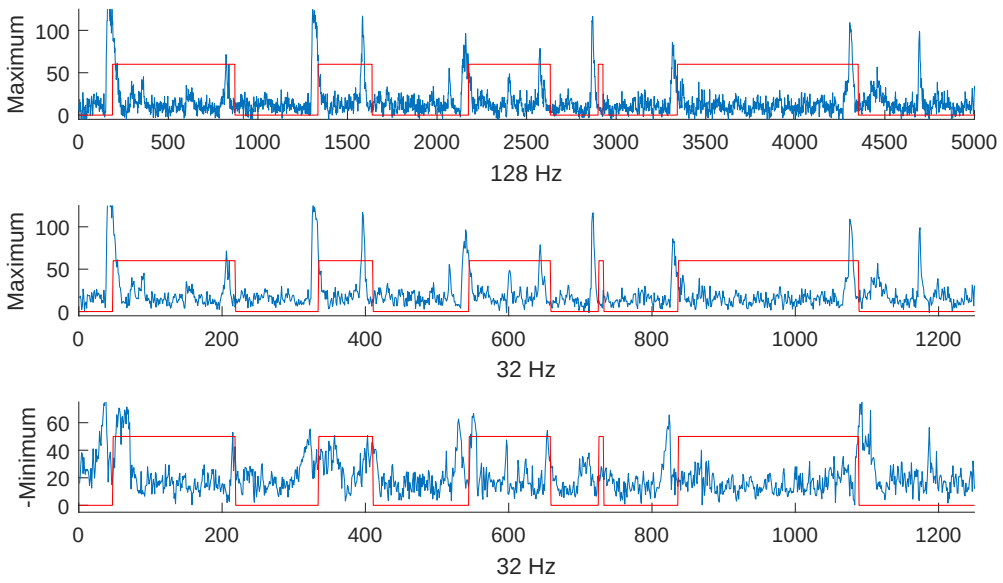


Figure 2.4: Cross-channel Maximum at 128 Hz frequency (top) Maximum at 32 Hz frequency (middle) -Minimum at 32 Hz frequency (bottom)

these two functions is an efficient way to detect possible change of eye state.

The utilized closing eye indicator was dropping the $\min(t)$ less than -45 μv and then rising above 55 μv within 7-9 instances. This indicator did not pick up three blinking epochs that originally lasted for 28, 40, and 52 instances and signaled one false alarm. The utilized opening eye indicator was raising the $\max(t)$ above 45 μv and then dropping under -40 μv within 5-9 instances. This indicator did detect all the opening eye epochs. The thresholds have been assigned by experimenting. Although the monitoring results did not match exactly with the changing class label times, they provided a close range of (-12, 12) instances from the actual class labels.

2.3.2 Multivariate Empirical Mode Decomposition (MEMD)

Multivariate Empirical Mode Decomposition (MEMD) is a data driven method that localizes time-frequency information in multivariate, non-stationary, low signal-to-noise ratio, and closely spaced frequency bands EEG signals [63, 51]. The open source MATLAB codes that are available through [63] have been used in this dissertation to decompose the

filtered signals into narrow-band components. Empirical Mode Decomposition (EMD) introduced by Huang et al. [41], decomposes a one-dimensional input signal, $x(t)$, into a finite set of Intrinsic Mode Functions (IMF), $c_i(t), i = 1, \ldots, M$, that is,

$$x(t) = \sum_{i=1}^{M} c_i(t) + d(t), \tag{2.3}$$

by using the following analytic precedure:

1. Find the locations of all the exterma of $x(t)$.
2. Interpolate between all the minima (respectively maxima) to obtain the lower (upper) signal envelop $e_{min}(t)$ (respectively $e_{max}(t)$).
3. Compute the local mean $m(t) = (e_{min}(t) + e_{max}(t))/2$.
4. Subtract the mean from the signal to obtain the "oscillatory mode" $s(t) = x(t) - m(t)$.
5. If $s(t)$ obeys the stopping criteria, then define $c(t) = s(t)$ as an IMF, otherwise set $x(t) = s(t)$ and repeat the process from step 1.

Once the first IMF is obtained, the same procedure is applied iteratively to the residual $d(t) = x(t) - c(t)$ to extract the remaining IMFs. The stopping criteria require that the number of exterma and zero crossings differ at most by one.

Every IMF is a narrow-band signal reflecting a different temporal scale intrinsic to the input signal. The IMFs are used to obtain a localized time-frequency spectrogram using Hilbert transform. However, analyses of EEG signals which are recorded simultaneously by channels require multivariate techniques that capture the cross-channel interdependence. Multivariate EMD, [68], uses a vector-valued form to simultaneously decompose a p-variate signal $\mathbf{x}(t)$ to

$$\mathbf{x}(t) = \sum_{i=1}^{M} \mathbf{c}_i(t) + \mathbf{d}(t), \tag{2.4}$$

where $\mathbf{c}_i(t),\ i = 1, \ldots,\ M$, are the p-variate IMFs containing scale-aligned intrinsic joint rotational modes and $\mathbf{d}(t)$ is the residue. This is accomplished via the following

analytic procedure. Consider the sequence of p-variate signal $\{\mathbf{x}(t)\}_{t=1}^{T}$, and suppose $\mathbf{s}_{\theta_k} = \{s_k^1, \ldots, s_k^p\}$ denote a set of direction vectors along the directions given by angles $\theta_k = \{\theta_k^1, \ldots, \theta_k^{(p-1)}\}$ on a (p-1) sphere. Then,

1. Choose a pointset for sampling on a $(p-1)$ sphere, let $\theta_k = \{\theta_k^1, \ldots, \theta_k^{(p-1)}\}_{k=1}^{K}$ be the angles of the direction vectors.

2. Calculate the projections $\{p_{\theta_k}(t)\}_{t=1}^{T}$ of the input signal $\{\mathbf{x}(t)\}_{t=1}^{T}$ along the direction vector s_{θ_k}, for $k = 1, \ldots, K$ (the whole set of direction vectors) giving the set of projections $\{p_{\theta_k}(t)\}_{k=1}^{K}$.

3. Find the time instants $\{t_{\theta_k}^i\}_{k=1}^{K}$ corresponding to the maxima of the set projected signals $\{p_{\theta_k}(t)\}_{k=1}^{K}$.

4. Interpolate $[t_{\theta_k}^i, \mathbf{x}(t_{\theta_k}^i)]$ to obtain the multivariate envelope curves $\{\mathbf{e}_{\theta_k}(t)\}_{k=1}^{K}$.

5. Calculate mean of the K multidimensional envelope curves $\mathbf{m}(t) = \frac{1}{K}\sum_{k=1}^{K}\mathbf{e}_{\theta_k}(t)$

6. Extract the detail $\mathbf{c}(t) = \mathbf{x}(t) - \mathbf{m}(t)$. If $\mathbf{c}(t)$ fulfills the stopping criterion for a multivariate IMF, apply the above procedure to $\mathbf{x}(t) - \mathbf{c}(t)$, otherwise apply it for $\mathbf{c}(t)$.

The analytic procedure is fully data-driven with no prior assumptions which give it an upper hand for analyzing nonlinear and non-stationary phenomena such as EEG signals. See Figure 2.5 for an example that shows how MEMD decomposes a three dimensional signal (x_t, y_t, z_t) into a collection of narrow-band frequency signals. The signal is given by

$$
\begin{aligned}
x_t &= \begin{cases} sin(2\pi f_1 t) + WN, & t = 1, \ldots 512, \\ sin(2\pi f_2 t) + WN, & t = 513, \ldots, 1024, \end{cases} \\
y_t &= sin(2\pi f_1 t) + sin(2\pi f_2 t) + WN, \quad t = 1, \ldots, 1024, \\
z_t &= sin(2\pi f_2 t) + sin(2\pi f_3 t) + WN, \quad t = 1, \ldots, 1024,
\end{aligned}
$$

where $f_1 = 5/f_s, f_2 = 11/f_s, f_3 = 23/f_s$, $f_s = 128$ and WN represents white noise.

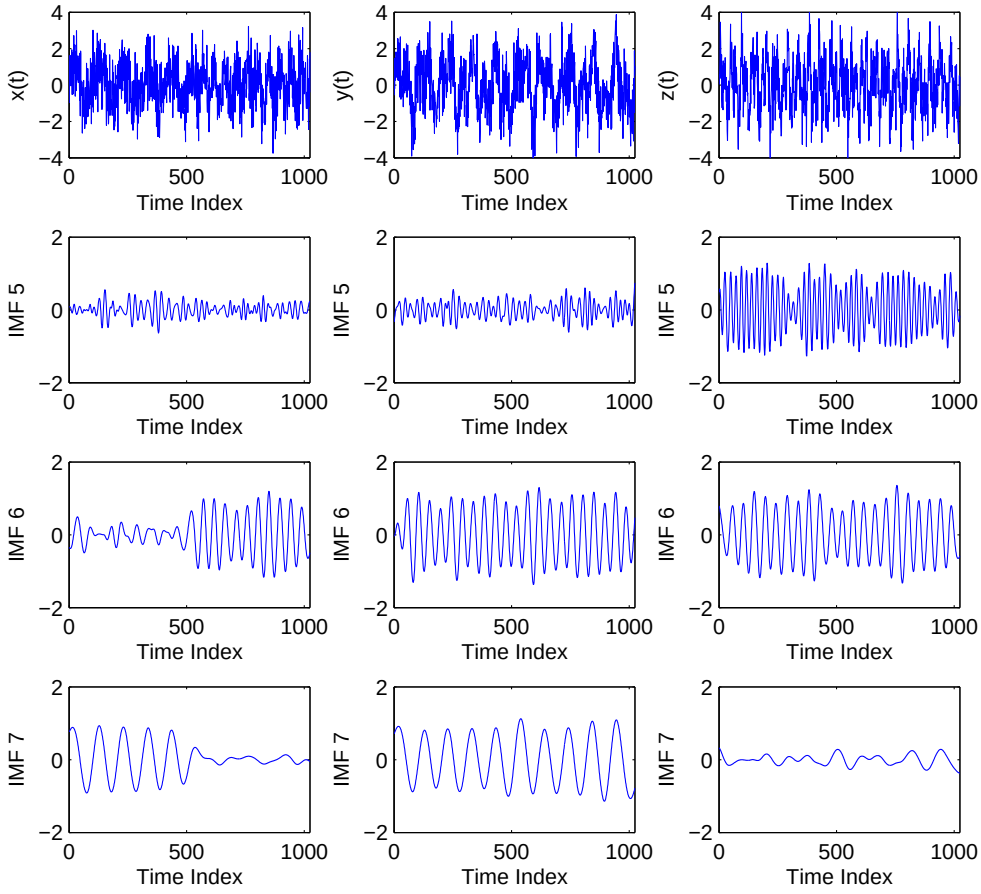


Figure 2.5: IMF5-7 of three dimensional signal (x_t, y_t, z_t) derived using MEMD

2.3.3 Common Spatial Pattern (CSP)

Common Spatial Pattern which is widely used in BCI applications [45, 87] creates spatial features that maximize the variance of the signals in one class while minimizing the variance of signals in the other class. This leads to creating discriminant features to classify two populations of EEG signals. It has been used in this study to design features to classify open/closed eye states. The CSP analytic driven algorithm is described as follows. Suppose $S^{(c)} \in \mathbb{R}^{ch \times ch}$ and $S^{(o)} \in \mathbb{R}^{ch \times ch}$ are the pooled estimates of the covariance matrices of the band-pass filtered EEG signal $X^{ch \times t}$ in the two conditions, i.e. closed and open eyes, given

by

$$S^{(j)} = \frac{1}{|I_j|} \sum_{i \in I_j} X_i^{ch \times t} \left(X_i^{ch \times t}\right)^T, \qquad j \in \{c, o\}, \tag{2.5}$$

where I_j is the set of indices corresponding to trials belonging to each condition and $|I|$ denotes the size of the set I. The CSP analysis seeks to find a matrix $W \in \mathbb{R}^{ch \times ch}$ and diagonal matrices $\Lambda^{(c)}$ and $\Lambda^{(o)}$, such that,

$$W^T S^{(c)} W = \Lambda^{(c)}, \quad W^T S^{(o)} W = \Lambda^{(o)}, \quad \text{and} \quad \Lambda^{(c)} + \Lambda^{(o)} = I. \tag{2.6}$$

This is accomplished by solving a generalized eigenvalue problem [13]. Having W, it allows projecting the EEG signals as

$$Z = W^T X. \tag{2.7}$$

For discriminating between two tasks, the variances of the spatially filtered signals by (2.7) are used as feature. The row vectors z_p from Z that maximize the difference in the variance between two groups are associated with the largest eigenvalues in $\Lambda^{(c)}$ and $\Lambda^{(o)}$. These signals are contained in the m first $(p = 1, \ldots, m)$ and last $(p = ch - m + 1, \ldots, ch)$ rows of Z in (2.7). The spatial features can then be obtained as

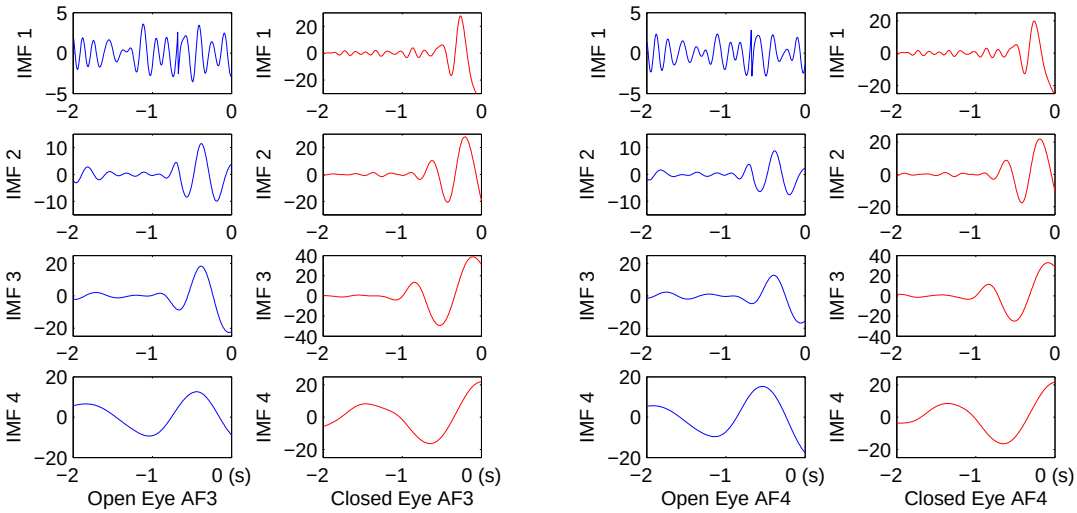
$$f_p = \frac{\mathrm{var}(z_p)}{\sum_{\substack{i=1,\ldots,m \\ i=ch-m+1,\ldots,ch}} \mathrm{var}(z_i)}, \tag{2.8}$$

where $\mathrm{var}(.)$ denotes the variance. For more information on CSP filters, see [13].

2.4 Frequency Band Selection

The cleaned signal $X^{ch \times t}$ is first passed through a low-pass filter of 8 Hz to include only the δ and θ band signals. The signal is then divided into 17 intervals of variable sizes based on the monitoring results. The average over all the epochs for each eye state has been computed for the last two seconds of the signals before changing eye state and the average passed through MEMD. If a signal was shorter than two seconds, the available instances have been used in the average. Figure 2.6 shows the first four IMFs of the open and closed

eye state for channels AF3 and AF4.



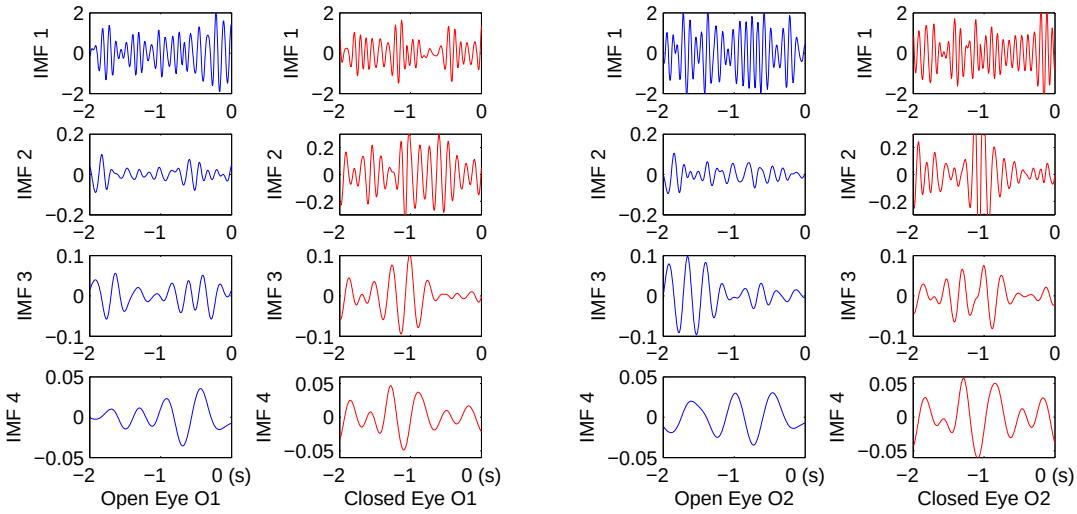
(a) channel AF3

(b) channel AF4

Figure 2.6: The first four IMFs of open/closed eye state averaged epochs after 0.5–8 Hz filter

Obviously, there is a change in behavior of the IMF signals right before the change of the eyes state. Typically, the closed eye has more energy especially when getting close to the event of changing the eye state [47]. The signals for channel AF3 have higher amplitude. Similar patterns observed in IMF signals of other channels and the signal strength decayed by moving further from the frontal part of the head.

The process then repeated using a band-pass filter of 8-13 Hz to investigate the effect of α band frequency. Figure 2.7 shows the first four IMFs of the open and closed eye state for channels O1 and O2. The IMF signals were stronger in these two channels and decayed by moving further away from posterior part of the head. The two channels have roughly the same patterns and characteristics such as power spectral, variance, and range. Notice the amplitude of signals which is a lot less than signals in Figure 2.6 but the frequencies are higher. The difference between the two states is detectable but the numerical values tested for features such as power spectrum, variance, maximum, and range is not significantly different. Considering Figure 2.6 and statistical characteristics of IMF signals in the two eye state, only band-pass filter of 0.5-8 Hz which is related to the eyes muscle activity is used for further analysis.



(a) channel O1 (b) channel O2

Figure 2.7: The first four IMFs of open/closed eye state averaged epochs after 8–13 Hz filter

2.5 Feature Extraction and Classification

In this section we develop two different feature extraction methods and perform classification procedures on each of them separately. In the first method a Common Spatial Pattern which is discussed in chapter 1 is used. In the second method statistical properties of signals in the two eye state that provide better discriminant features are employed to extract features. The results of the two method then is compared.

2.5.1 Using Common Spatial Pattern

In this section, features relevant to the two eye state were extracted using CSP algorithm. CSP filters should be calculated from training data and the resulting filters need then to be applied to the test set [13]. For this purpose and to evaluate the efficiency of the predictive analytic model, 10-fold cross-validation has been used. That is, the band-pass filtered (0.5-8 Hz) data including 9 open eye and 8 closed eye epochs has been divided into ten subsets where each time four subsets used for training the classifier and one for testing the performance.

The test and training signals first passed though MEMD to extract oscillatory IMFs. The first IMF of the training signals which provided the best discriminant statistics is then used to compute the spatial filters W using CSP. The first two and last two spatial filters then applied to both the test and training IMFs as described in (2.7) and their f_p features were calculated using (2.8). Finally, three classification methods have been used to assess the accuracy of the analytic model. Table 1 below, provides average results of the five-fold classification using logistic regression, artificial neural networks, and support vector machine classifiers.

Table 2.1: Classification Results

Features	Logistic Regression		ANN $4 \times 4 \times 2$		SVM RBF Kernel $\sigma = 0.5$	
	Accuracy	F-score	Accuracy	F-score	Accuracy	F-score
IMF1: f_2, f_3 IMF2: f_1, f_3	0.834	0.854	0.834	0.802	0.834	0.854
IMF1: f_1, f_3 IMF2: f_1, f_4	0.834	0.828	0.834	0.828	0.766	0.820
IMF1: f_1, f_3 IMF2: f_1, f_2	0.784	0.828	0.766	0.750	0.716	0.794
IMF1: f_2, f_3 IMF2: f_1, f_4	0.768	0.788	0.766	0.668	0.784	0.728

The set of features were selected upon visual inspection of the features' matrix scatter plots. The combination of four features, two from IMF1 and two from IMF2, provided higher accuracy than features from one IMF alone. Features from IMF3 were not used because of their low test accuracy. The best classification performance for Simple Logistic Regression, Artificial Neural Network ($4 \times 4 \times 2$), and Support Vector Machine with Gaussian Kernel ($\sigma = 0.5$) classifiers achieved by utilizing features f_2, f_3 from IMF1 and f_1, f_3 from IMF2, accuracy of 83.4%. The average F-score which is the ratio of precision to recall was 0.854 for both Simple Logistic Regression and Artificial Neural Network, and was 0.802 for Support Vector Machine. The default value of 0.5 has been used for Gaussian kernel standard deviation in SVM method since the training size was not large enough to perform model parameters' estimation.

Real-time prediction follows the procedure in Figure 2.8. First, the signal is passed through a high-pass Butterworth filter of 0.5 Hz to remove the DC offset. The cleaned signal is monitored in real time by $\min(t)$ and $\max(t)$ computed at 32 Hz. Upon detection of a change in the eye state, the last two seconds of the 128 Hz signal is passed though a low-pass filter of 8 Hz to include only the eye movement artefact. The filtered signal is then passed through Multivariate Empirical Mode Decomposition and four features (f_2, f_3 from IMF1, f_1, f_3 from IMF2) are extracted using CSP method. The features are then fed to a logistic regression classifier to predict the eye state. The monitoring then continues. The procedure needs calibration to estimate the spatial filters W before applying.

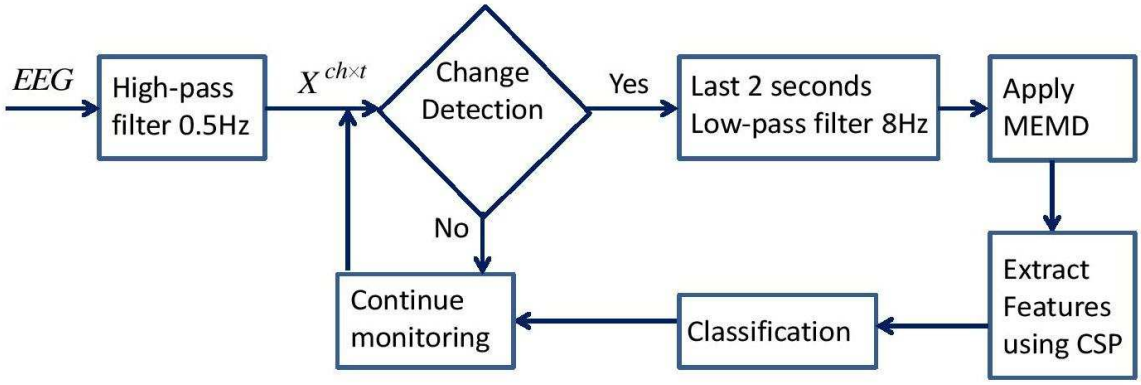


Figure 2.8: Prediction process using CSP

2.5.2 Using Discriminant Functions

Investigating Figure 2.6, a change in variance of the IMF signals is obvious right before change of the eyes state. In addition, looking at the range of the y-axis, the first three IMFs clearly show a higher amplitude in the close than open eye states. This motivates using the variance and maximum of the first three IMFs as discriminant features. The similarity between the IMFs of the AF3 and AF4 channels suggests averaging their computed features for higher reliability. Therefore, variance and maximum of the last one second of the first three IMFs were computed for channels AF3 and AF4 then averaged and marked as potential features. These features presented in Table 2.2 provide the most discriminant values between the two eye states, especially Max1 and Var2 which are the maximum of first and variance of the second IMF, respectively.

Table 2.2: Extracted Features from relevant IMF number

Max1	Max2	Max3	Var1	Var2	Var3	Class
13.03	34.09	34.60	31.51	293.56	356.22	C
13.18	46.47	36.97	31.88	478.27	451.94	C
11.22	13.68	45.59	28.85	59.08	709.03	C
10.92	28.40	34.86	20.31	208.63	410.00	C
13.98	21.20	46.59	44.21	168.20	773.56	C
8.99	27.01	35.72	17.83	187.53	459.14	C
17.96	26.30	23.06	44.87	179.57	173.45	C
7.33	39.57	59.16	11.97	413.06	885.72	C
28.30	44.56	26.34	140.33	481.84	170.15	C
10.74	11.28	20.21	17.47	31.90	164.18	O
5.02	28.50	34.81	5.02	173.97	439.28	O
19.02	7.93	24.27	62.11	18.97	259.55	O
8.31	23.89	37.62	12.71	127.14	671.07	O
12.10	13.23	7.12	27.94	37.49	19.02	O
9.96	18.06	20.78	20.58	81.80	204.45	O
5.49	3.55	3.38	6.61	3.95	4.18	O
10.96	8.24	15.72	19.39	21.15	83.34	O

To evaluate the efficiency of the predictive analytic model, five-fold cross-validation has been used. That is, the data set including 9 open eye and 8 closed eye epochs has been divided into five subsets where each time four subsets used for training the classifier and one for testing the performance. Table 2.3 provides average results of the five-fold classification using Logistic Regression, Artificial Neural Networks, and Support Vector Machine classifiers.

Simple Logistic Regression classified the two eye states with an accuracy of 88.2% using only two features, Max1 and Max2. The average F-score which is the ratio of precision to recall was 0.882. Accuracy of Artificial Neural Network classifier with three hidden layers was 82.4% with F-score of 0.824 for the same features. Using Max1 and Var2 features ANN achieved a higher accuracy, 88.2%. Support Vector Machine classifier's average accuracy was 70.1% for the two sets of features. Linear Kernel has been used for SVM classifier. These performances could be improved by further training the classifiers with more samples. More training examples also provide enough data to utilize regularized logistic regression which can improve the accuracy by adjusting a regularization factor. In addition to the reported accuracies that are based on cross-validation of 17 eye state changes, the

instance based accuracies were computed that count the percentage of correct predicted instances. These accuracies were 0.877 (Max1, Max2) and 0.786 (Max1, Var2) for Logistic Regression and 0.815 (Max1, Max2) and 0.877 (Max1, Var2) for Artificial Neural Network classifiers.

Table 2.3: Classification Results

Features	Logistic Regression		ANN $2 \times 3 \times 2$		SVM Linear Kernel	
	Accuracy	F-score	Accuracy	F-score	Accuracy	F-score
Max1, Max2	0.882	0.882	0.824	0.824	0.706	0.706
Max1, Var2	0.765	0.760	0.882	0.882	0.706	0.700

Real-time prediction follows the procedure in Figure 2.9. First, the signal is passed through a high-pass Butterworth filter of 0.5 Hz to remove the DC offset. The cleaned signal is monitored in real-time by (2.1) and (2.2) computed across channels at 32 Hz. Upon detection of a change in the eye state, the last two seconds of the 128 Hz signal is passed though a low-pass filter of 8 Hz to include only the eye movement artifact. The filtered signal is then passed through Multivariate Empirical Mode Decomposition and two features Max1 and Max2 are extracted that are based on the first and the second IMFs. The features are then fed to a logistic regression classifier to predict the eye state. The monitoring then continues. The proposed algorithm detects an eye state change in less than two seconds from its happening and continues monitoring right after a decision is made regarding the eye state. Thus, a delay of two seconds at most in detection is expected but the monitoring catches up afterwards. The procedure needs calibration before utilization.

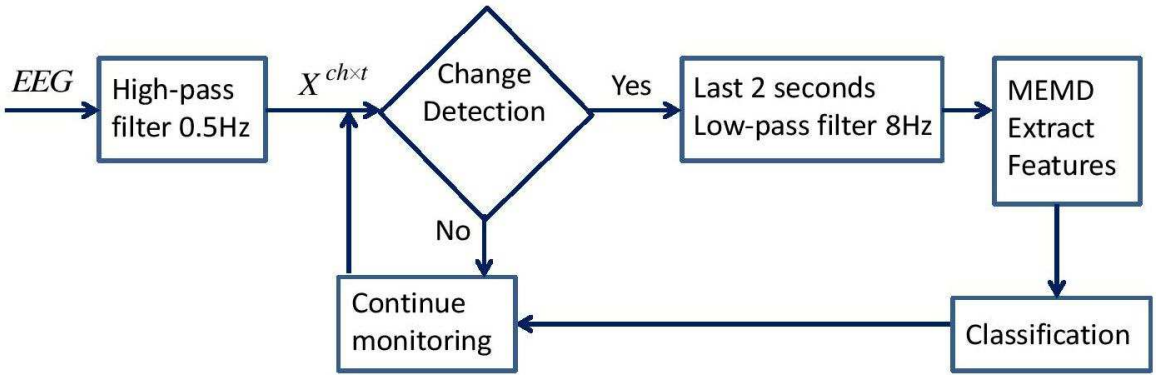


Figure 2.9: Prediction process using Discriminant Functions

2.6 Contribution

While Rösler and Suendermann [70] have used this dataset and achieved 97.3% accuracy using k-star classifier, their algorithm needs an average of 20 minutes to predict the eye state for new instances. Wang et al. [89] and Sabanci and Koklu [73] achieved accuracy of 76.6% and 84.05%, respectively through state of the art instance based algorithms. However, considering the sampling frequency of 128 Hz, the computations these classification procedure have to perform is huge, time consuming, and energy inefficient.

The methods described in this study take the random nature of the eye state change into account and classification is applied only when a possible change of the eye state is detected using cross-channel information. The two procedures we proposed in this study take less than 2 seconds to predict an eye state change in real-time with 83.4% and 88.2% accuracy using a personal notebook with dual-core CPU 2.30Hz and 2GB RAM. A higher number of training examples could improve the accuracy of the classification procedures.

Having access to only one participant was a study limitation. However, this does not belittle the study and applicability of its results. The proposed analytical procedure could also be used with medical grade EEG devices and for new participants after adjustments for monitoring thresholds. Real applications require recording and investigation of the signals under real-life situations so that the signals include environmental artifacts. However, the controlled experimental situations are not simplistic trials and considered a step toward achieving the ultimate goal.

CHAPTER 3

APPLICATION IN MODELLING HONEY BEE DANCE

Honey bees perform special dances within the beehive to communicate the location of food sources. Usually, each dance pattern consists of two phases. The waggle phase during which the bee walks roughly in a straight line while rapidly shaking its body from left to right and the turning phase at the endpoint of a waggle dance in which the bee typically returns to the starting location of the waggle dance by turning in a clockwise (right) or counterclockwise (left) direction. The direction and duration of waggle dance conveys the direction and distance of the food source, respectively. For example, flowers that are located directly in line with the sun are represented by waggle runs in an upward direction on the vertical combs, and any angle to the right or left of the sun is coded by a corresponding angle to the right or left of the upward direction; see Fig. 3.1.

The Australian scientist Karl von Frisch was the first who discovered how honeybees communicate the location of the food source through dancing [31] and was honored with a Nobel Prize in 1973 for his discovery. Honey bees even adjust their flight path to compensate for being blown off course by the wind and for changing position of the sun through time [69]. However, their course is seldom so precise that they can find the food without the aid of vision and/or smell as they approach it [66].

The aim of this chapter is to use the approach discussed in chapter 2 to classify the dance patterns with high accuracy in real-time. That is, to utilize a monitoring function, then apply classification upon observing a potential change in the dance move.

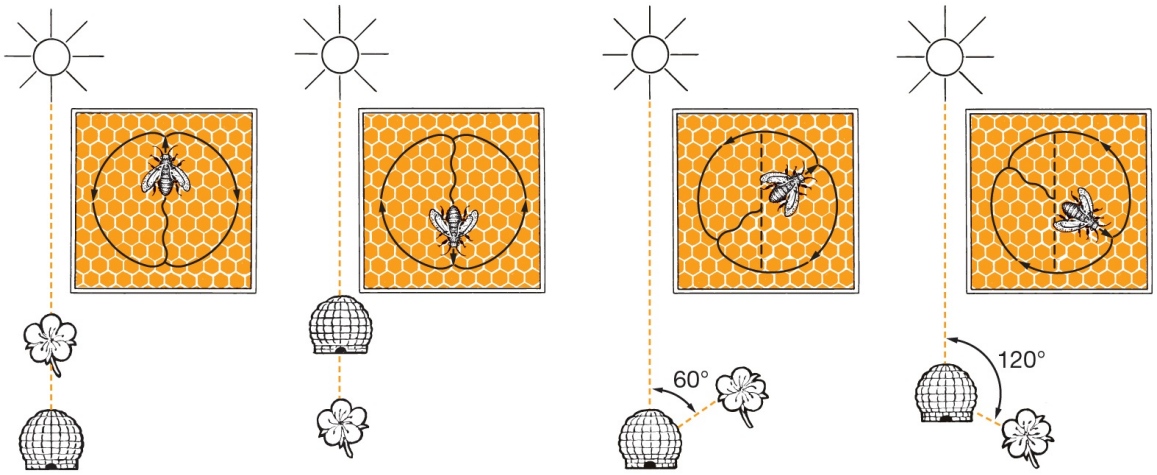
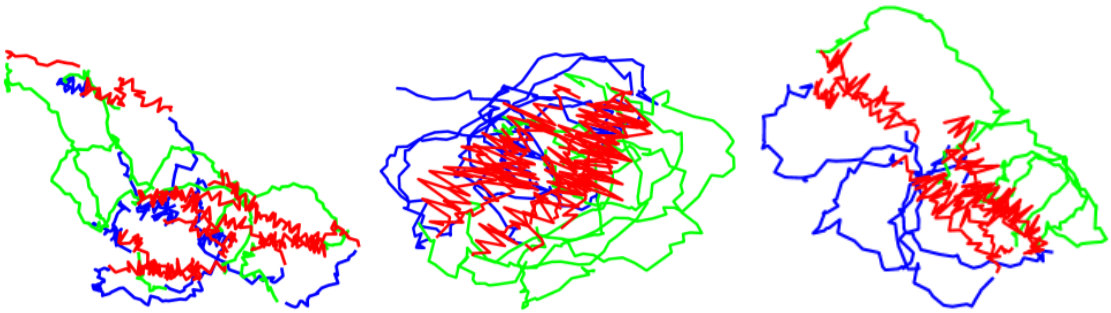


Figure 3.1: Honeybee dance to indicate location of a food source [91]

3.1 Materials and Methods

A dataset for honey bee dance can be found in [61] where there are six set of measurements $\mathbf{x}_t = (x_t, y_t, \theta_t, c_t)$ in which (x_t, y_t) denotes the 2D coordinates of the bee's body in x and y axis at time t, θ_t represents the bee's head angle, and c_t shows the ground truth or label of the dance move; $c = -1$ for turn-right, $c = 0$ for turn-left, $c = 1$ for waggle. The dance label has been added later by investigating a video feed that was focused on the honey combs. Figure 3.2 displays three such sequences of honey bee dances. Researchers have tried to model the honey bee dance patterns using various methods in order to understand the communication aspect and to mimic that pattern in building intelligent robots. Yet after 60 years of discovering the waggle dance it is unknown how a follower bee decodes the information contained in the dance. Moreover, researchers still use the time-consuming and error-prone process of human labeling to segment dance patterns.

Oh et al. [61], Xuan and Murphy [88], and Fox [30] have used switching dynamical models to analyze these honey bee dances. In particular, Oh et al. (2006) have used Switching Linear Dynamic System (SLDS) in a supervised hold-one-out formulation and Parameterized Segmental SLDS which requires additional supervising during the learning process to achieve high model accuracy. Xuan and Murphy (2008) have segmented the overall signal into patterns of dance moves using a first-order Auto-Regressive AR(1) model, using



(a) sample honey bee dance #1 (b) sample honey bee dance #5 (c) sample honey bee dance #6

Figure 3.2: Honeybee dance patterns, turn-right (blue), turn-left (green), waggle (red)

independent features or with a full covariance model. Fox (2010) have used sticky Hidden Markov Model (HMM) with Hierarchical Dirichlet processes (HDP) and switching Vector Autoregressive (VAR) in both unsupervised and partially supervised setting to model the dance patterns. Table 3.1 shows accuracy of these approaches for each dancing bee.

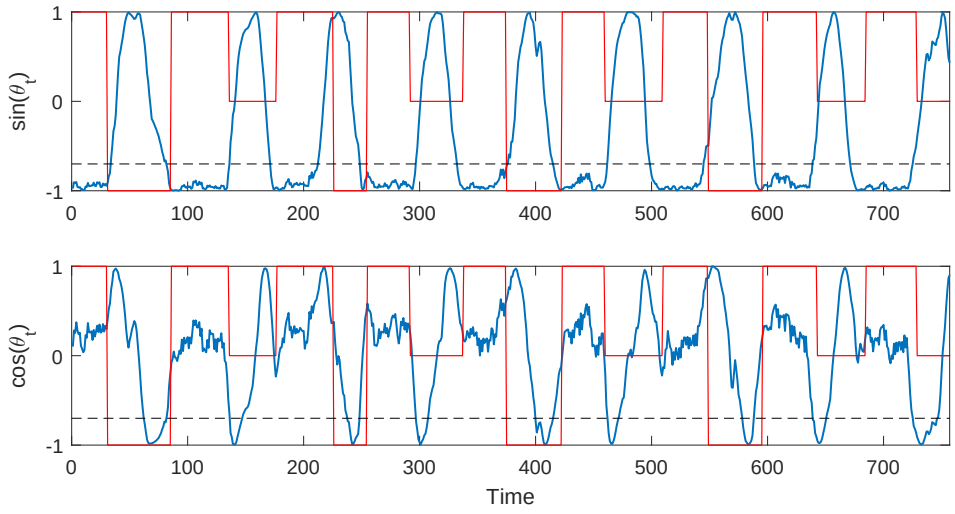
Table 3.1: Model accuracy

Bee #	1	2	3	4	5	6
HDP–VAR(1)–HMM unsupervised	45.0	42.7	47.3	88.1	92.5	88.2
HDP–VAR(1)–HMM partially supervised	55.0	86.3	81.7	89.0	92.4	89.6
SLDS supervised MCMC	74.0	86.1	81.3	93.4	90.2	90.4
PS–SLDS supervised MCMC	75.9	92.4	83.1	93.4	90.4	91.0

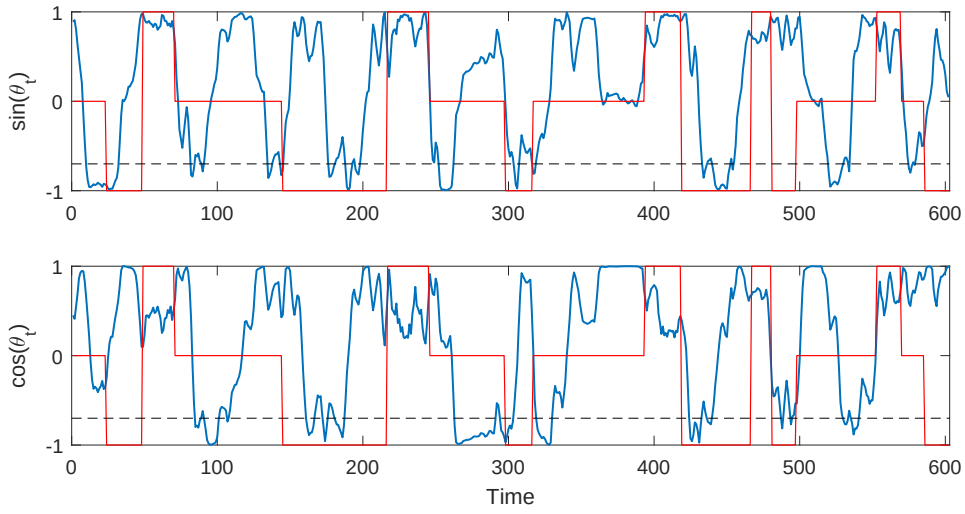
Although, the above methods seems to achieve a high model accuracy through sophisticated modeling, we believe a simpler model that runs fast and does not require extensive training is favorable in real applications. This can be achieved through monitoring and then classifying only when a potential change is observed.

After investigating some statistical characteristics of the the dance signals such as variance, average, minimum, maximum, moving mean and variance we arrived at the conclusion that a moving average of $\sin(\theta_t)$ with window size 3 can efficiently signal for a change of dance pattern. The moving average of 3 smooths the sharp edges of a signal and helps with detecting steady increase/decrease. Figure 3.3.a shows how setting a threshold on $\sin(\theta_t)$ can segment the overall dance signal into intervals of patterns for bee #4. We set the threshold as -0.7 which could detect all the change dance patterns with some lags at

the time they happened. The same pattern could be seen in bee #5 and #6 but it was less tractable in bees #1, #2, and #3. The utilized head angle signals have been normalized to reflect the direction toward 6:00.



(a) bee #4



(b) bee #3

Figure 3.3: Monitoring functions for the honeybee dance

By comparing the dance patterns of bee #3 and bee #4 in Figure 3.3 we realize that there are some intervals that logically does not match with the provided label for bee #3. This issue has been mentioned in [30] as well. We could not find out how the labels have been assigned to the overall signal but we could find the videos related to the dance moves.

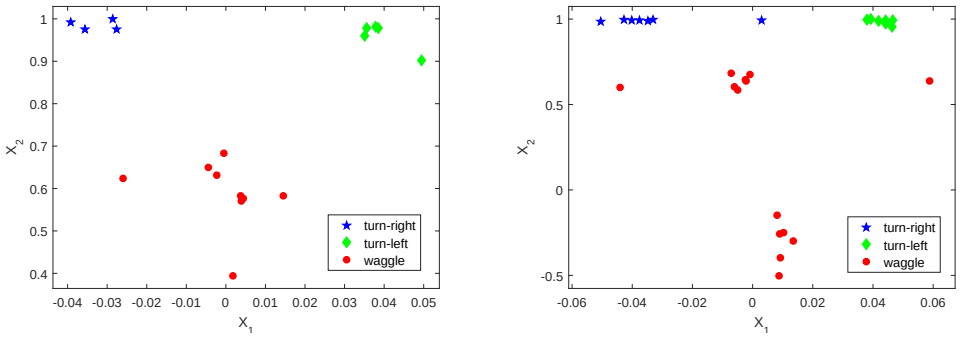
However, the files were not the original recordings and low quality of the videos made it impossible to find the true dance patterns. Thus, we opt to analyze only the dance patterns of bee #4 to #6 which seems that have little to no discrepancy.

Next step is the feature extraction in which we find discriminant features to classify three different dance moves. We started by bee#4 and after positive feedback applied the same technique to other bees. For bee #4, after segmenting the continuous signal into 18 intervals of various sizes based on the results of the monitoring section, we extracted various numbers from each interval as potential features. For the head angle θ_t, these features included variance, mean, minimum, and maximum of $\sin(\theta_t)$, $\cos(\theta_t)$, $MovAve(\sin(\theta_t), 3)$, $MovAve(\cos(\theta_t), 3)$, $Diff(MovAve(\sin(\theta_t), 3))$, $Diff(MovAve(\cos(\theta_t), 3))$, etc. computed over a given interval $t \in T$. After investigating the matrix scatter plot of the extracted features, we decided to use X_1: mean of the $Diff(MovAve(\cos(\theta_t), 3))$ and X_2: maximum of $MovAve(\cos(\theta_t), 3)$ as features. Figure 3.4.a best shows the reason for this decision. Obviously, this choice significantly discriminates the three dance patterns.

Going over the same process of monitoring using moving average of $\sin(\theta_t)$, utilizing a threshold of -0.7 for detecting potential change of dance pattern, segmenting the continuous recording signal using the monitoring section, and extracting the two features X_1 and X_2 for bees #5 and #6 will result in scatter plot of features illustrated in Figures 3.4.b and 3.4.c, respectively. Like before, the segmented intervals are well discriminated using the two utilized features. This will result in great classification performance in the next step.

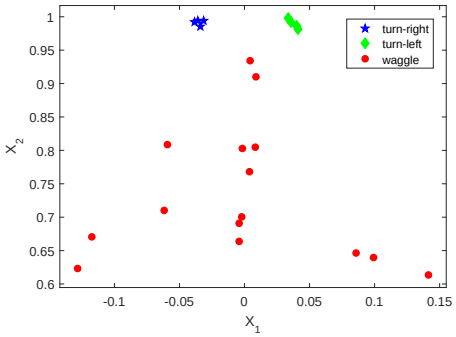
3.2 Results

In this step, the features are fed to three classification methods known as Logistic Regression, Artificial Neural Network, and Support Vector Machine. Weka software has been used to perform the classifications with a 5-fold cross validation. Table 3.2 provides the accuracy and F-score of classifying the bee dance patterns using our approach for different bees. The advantages of this approach are



(a) bee #4

(b) bee #5



(c) bee #6

Figure 3.4: The two selected features and their values

- High classification accuracy that reaches to approximately 100% for some cases that beats alternate approaches in the literature.
- Fast run time of the procedure, it runs in fractions of a second, which makes it suitable for real-time and real-life predictions.
- Simplicity of the proposed process
- Possibility of pooling the samples together for higher classification accuracy.

As seen in Table 3.2, we can pool the extracted features together and then perform classification. This improves the reliability of the results since there are more training samples.

Table 3.2: Classification Accuracy and F-score

	Bee #	4	5	6	4–6
Logistic Regression	Accuracy	94.4	100.	87.0	97.1
	F-score	94.6	100.	87.0	97.2
ANN $2 \times 3 \times 3$	Accuracy	100.	100.	73.9	92.9
	F-score	100.	100.	69.3	93.0
SVM RBF Kernel	Accuracy	100.	81.5	66.2	98.6
	F-score	100.	76.3	52.5	98.6

3.3 Contribution

We propose utilizing the decision circuit process given in Figure 3.5. First, the normalized head angle signal θ_t is monitored in real time by $\sin(\theta_t)$ computed at 30 Hz frequency. Upon passing a threshold of -0.7, the available signal is used to extract two features X_1, X_2. The features are then fed to a selected classifier, we recommend ANN, to predict the type of dance movement. The monitoring then continues for the remainder of the signal. The procedure needs calibration before applying.

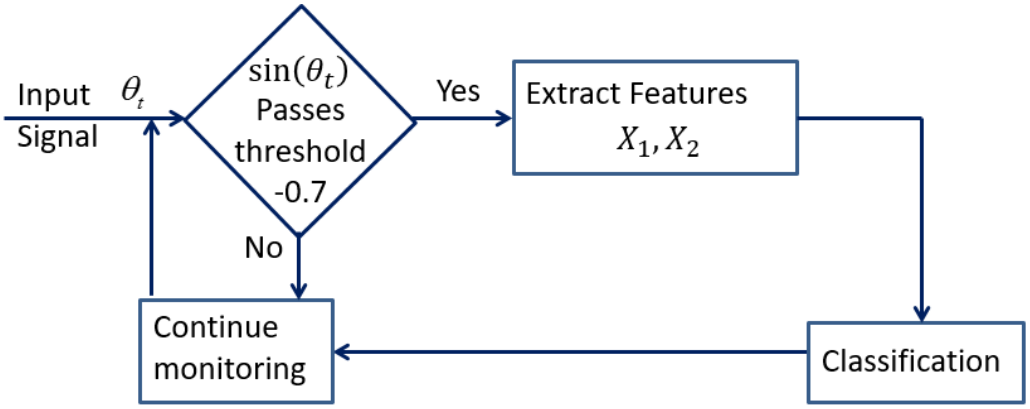


Figure 3.5: Real-time application circuit

Studying honey bee dance patterns requires expert-labeling of videotapes which is a time-consuming and error prone process. As seen in this application, the monitoring-action process significantly improves the processing time of classifying signals and could replace the time-consuming and error-prone stage of expert labeling. We are currently working to present the results of this chapter to a peer reviewed journal.

To shed light on the importance of the results we would like to mention that decoding honey bee waggle dance aids in in commercial crop pollination and could be followed by other larger studies. An obvious example is the California almond crop, for which approximately one million bee colonies are rented each year during spring bloom [57] and which according to FAO report in 2015 had an annual value of $5.7 billion [96].

CHAPTER 4

FUTURE RESEARCH

This chapter brings out our future project on classification of heart sound signals (PCG). Moreover, we are in the process of gathering a BIG collection of ECG/PCG signals related to various heart dysfunctions to perform classification of heart diseases. The results will allow us to accurately predict different heart dysfunctions using a minute of heart EEG/PCG.

In the first part of this chapter, we introduce a variety of methods that is used to classify time series data. These methods are somehow different and more developed than the initial classification approaches mentioned in the chapter 1. The second part is dedicated to bringing forward the two important questions that needs to be answered in future attempts.

4.1 Time Series Classification

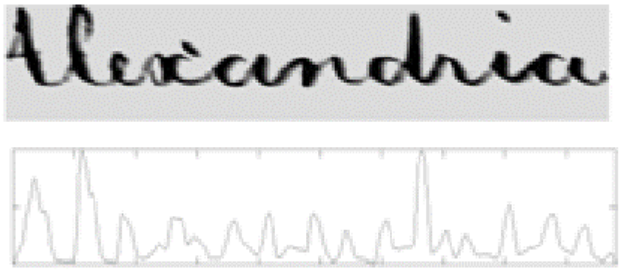
Time series (TS) are a collection of values sequentially recorded over time. Everything we do is a function of time, thus it is not a surprise to see TS tangled in every day activities. Some examples of TS data include brain and heart signals, measurements of wearable devices, stock market price, home censors, or even shapes; see Figure 4.1. TS emerge in many applications, like weather observations, industry automation, mobility tracking, etc. Time Series Classification (TSC) describes the task of predicting a class label for a TS whose label is unknown. To achieve that, a classifier has to train a model using labeled TS [79]. TSC problems are differentiated from traditional classification problems because the attributes are ordered. Whether the ordering is by time or not is in fact irrelevant. The important characteristic is that there may be discriminatory features dependent on the ordering [7].

(a) brain and heart signals

(b) wearable devices

(c) shapes

(d) stock market price



(e) handwriting in x and y axis

Figure 4.1: Some Time Series examples

The University of California, Riverside (UCR) time series classication and clustering archive [102] is often used as basis for benchmarking and comparing TS research results. UCR contains a representative sample of 85 univariate TS datasets and widely used by researchers to test TS algorithms. TS classifiers can be divided into six groups [7, 79]:

1. **Whole Series:** A distance measure is applied to the whole TS data and two TS are in the same class if their distance is close. Elastic distance measures compensate for small differences in the TS such as warping in the time axis.

2. **Intervals:** Rather than using the whole series, selects one or more (phase dependent) intervals of the series to compute a similarity distance. At its simplest, this involves a feature selection of a contiguous subset of attributes.

3. **Shapelets:** Focuses on finding short patterns that can appear anywhere in the series to define a class. These phase independent patterns are commonly called shapelets. A class is then distinguished by the presence or absence of one or more shapelets somewhere in the whole series.

4. **Bag–of–Features/Bag–of–Patterns:** Distinguished by the frequency or repetition of subseries or patterns, then build classifiers based on the resulting histograms.

5. **Ensembles:** combines two or more of the above approaches into a single classifier. Each classifier produces a label and a majority vote is performed. These classifiers have shown to be highly accurate at the cost of increased runtime.

6. **Model based:** Fits a generative model such as Auto-Regressive, Hidden Markov, Kernel, etc. to each series, then measures similarity between series using similarity between model parameters.

Bagnal et al. [7] have implemented 18 recently proposed TSC algorithms in a common Java framework and compared their classification results by performing 100 resampling experiments on each of the 85 datasets in UCR TS database. They show that one classifier, the Collective Of Transformation Ensembles (COTE), is signicantly more accurate than all of the other algorithms [8]. That is not a surprise since COTE is an ensemble classifier that includes a combination of algorithms in time, autocorrelation, power spectrum and shapelet domains. It pools 35 classifiers into a single ensemble with votes weighted by train set cross validation accuracy.

However, there is a drawback to Ensemble classifiers, the issue of run time. Schäfer [79] has recently compared 12 state–of–the–art time series classifiers based on prediction

and classication times. He observed that most of these classifiers including COTE require extensive train and classification times, and might not be applicable for some applications. Apparently, there is a trade off between classification accuracy and computational complexity. To obtain high accuracy, time series classifiers have to perform extensive training. Schäfer then suggests using Bag–of–SymbolicFourierApproximation–Symbols (BOSS VS) since it offers a good trade off between classication accuracy and runtime [80]. Future research in time series classification could lead to producing fast and accurate classifiers.

4.2 Dataset

As an example, we refer to a dataset from a recent challenge proposed by PhysioNet [101]. Heart sound recordings were collected at either a clinical or nonclinical environment, from healthy subjects and pathological patients, both including children and adults. Each subject have contributed between one to six heart sound recordings. All recordings have been resampled to 2,000 Hz and have been provided as .wav format. The training set consists of five databases (A through E) containing a total of 3,126 heart sound recordings, lasting from 5 seconds to just over 120 seconds. The training and test sets have each been divided so that they are two sets of mutually exclusive populations.

The heart sound recordings were collected from different locations on the body. The typical four locations are aortic area, pulmonic area, tricuspid area and mitral area, but could be one of nine different locations. In both training and test sets, heart sound recordings were divided into two types: normal and abnormal heart sound recordings. The normal recordings were from healthy subjects and the abnormal ones were from patients with a confirmed cardiac diagnosis, typically heart valve defects and coronary artery disease. There are recordings corrupted by various noise sources, such as talking, stethoscope motion, breathing and intestinal sounds. Thus, they are very difficult or even impossible to classify as normal or abnormal. Therefore, the classifications for the heart sound recordings could be three types: normal, abnormal and unsure (too noisy to decide). Figure 4.2 shows an example of Normal/Abnormal PCG TS graphed in the interval of 10 to 15 seconds.

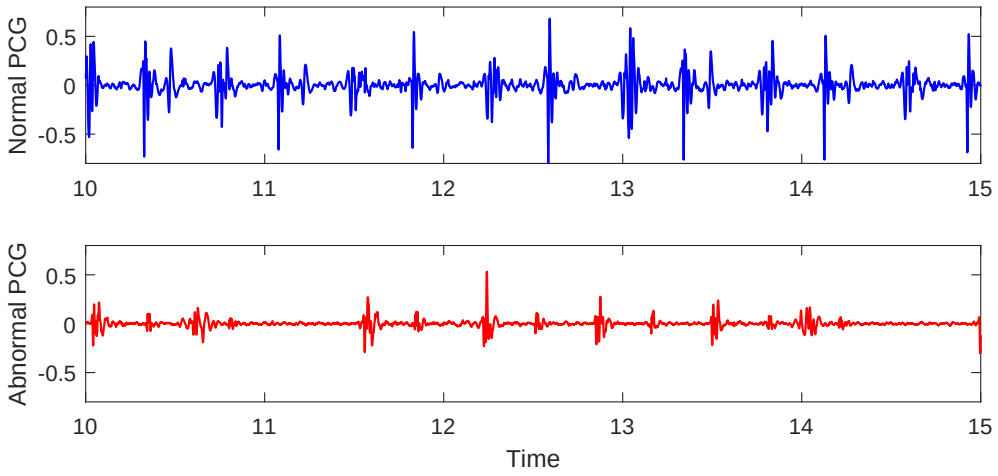


Figure 4.2: Sample PCG TS for Normal/Abnormal subjects

4.3 Classification Approach

By looking at the dataset and considering characteristics such as variable length, different signal patterns and strength even within one subject, and intervals of not so useful recordings we decided to first segment the heart sounds into intervals of S1, Systole, S2, and Diastole in each subject and compute average heart sound for each subject. Then perform an Ensemble classification algorithm to classify subjects into three mentioned groups which aligns with the importance of accurate predictions. Thus, there are two questions that has to be answered

- How to automatically segment the heart sound into intervals of S1, Systole, S2, and Diastole?
- What algorithms to include in the ensemble classification approach?

The challenge proposes Springer's improved version of Schmidt's method [83, 84], which uses a Hidden Markov Model trained using database "A" of the training set. This is certainly a good strategy to segment the signals but as it is with HMM models, it sure takes a long time to run. that we are working on. What we propose to use is described in the following section.

4.4 Future Work

One of the drawbacks of employing machine learning techniques upfront is that most of the state-of-the-art algorithms require extensive train and prediction times, thus they are not suitable for real-time predictions which limits their real-life applications. However, many problems do not require going through decision making using machine learning algorithms upfront.

The methods that we utilized in modeling multivariate brain signals can be adopted to analyze other biomedical signals and time series data in real-time. The process control monitors the input for potential change. Upon detection of a possible change, a fully data-driven method steps in to extract discriminant features from the segmented input signal. Then, machine learning algorithms such as Logistic Regression, Artificial Neural Network (ANN), and Support Vector machine (SVM) are employed to verify the change. In chapter 3, we employ our proposed method in detecting the three different dance moves that honey bees perform to communicate the location of a food source. The results are significantly better than other alternative methods in the literature. Other possible applications of our approach are

- Real-time epileptic seizure prediction using brain signals where early detection and prediction might be able to abort seizures through targeted therapies or could prevent accidents and limit injury.
- Monitoring Market price for a significant change of patterns, then employing machine learning for making decisions upon detecting potential change.
- Detecting Change of Motion through video or biomedical signals achievable by monitoring for a possible change and then using machine learning when a possible change happens.
- Autonomous cars to continue a former pattern unless a significant change in the conditions has taken place.

CHAPTER 5

ON HEREDITY FACTORS OF PARKINSON'S DISEASE: A PARAMETRIC AND BAYESIAN ANALYSIS

It is believed that hereditary is one of the key risk factors of Parkinson's disease (PD) and the children of individuals with Parkinson's disease carry a two-fold risk for Parkinson's [99]. The aim of this chapter is to study the prevalence of Parkinson's disease in five types of families. That is, families with negative history of PD (type 1), families with positive history where none of the parents (type 2), one of the parents (type 3 and 4), or both of the parents (type 5) carried the disease. We use Maximum Likelihood and Bayesian Inference to estimate and compare the chance of developing Parkinson's in the five mentioned family types. Historical data on the number of people in grandparents' family who suffered from the PD in the form of prior information is incorporated with the data from individuals to draw Bayesian estimations.

The Data used in this chapter is obtained from the Parkinson's Progression Markers Initiative (PPMI) database [100]. PPMI is a public-private partnership that is funded by the Michael J. Fox Foundation for Parkinson's Research and funding partners, including AbbVie, Avid Radiopharmaceuticals, Biogen, Bristol-Myers Squibb, Covance, GE Healthcare, Genentech, GlaxoSmithKline, Lilly, Lundbeck, Merck, Meso Scale Discovery, Pfizer, Piramal, Roche, Servier, and Union Chimique Belge (UCB).

The role of heredity in developing Parkinson's is verified in this study through different estimations based on the status of parents with respect to Parkinson's disease. Currently, our results are under review by a journal geared towards medical professionals [76].

While many physicians have provided medical opinions on this topic, our study is one of the first to provide statistical analysis to support the conclusions.

5.1 What is Parkinson's disease?

Parkinson's disease (PD) is a chronic and progressive movement disorder, meaning that symptoms continue and worsen over time. Nearly one million people in the US are living with Parkinson's disease and approximately 60,000 Americans are diagnosed with Parkinson's disease each year, and this number does not reflect the thousands of cases that go undetected. The cause is unknown, and although there is presently no cure, there are treatment options such as medication and surgery to manage its symptoms [99].

The diagnosis of PD depends upon the presence of one or more of the four most common motor symptoms of the disease. That is, the physical symptoms of PD, such as balance, rigidity, tremor, freeze attacks, uncontrollable movements, slowness of movement, walking/gait, postural difficulties, writing difficulty, swallowing difficulty, muscle pain, masked face, speech problems and general loss of motor skills. In addition, there are other secondary and non-motor symptoms that affect many people and are increasingly recognized by doctors as important to treating Parkinson's. These symptoms contribute to severe disability and impaired quality of life in advanced Parkinson's disease. Symptoms include anxiety, depression, cognitive mood swings, dementia, constipation, pain, genitourinary problems, sudden drop in blood pressure upon standing, excessive sweating, sleep disturbances, sense of smell, vision, memory, weight loss, psychosis, hallucinations and loss of energy, among others [1, 99]. The present problem in developing statistical analysis and modeling of PD is the lack of data to identify the significant risk factors and interactions so that we can rank this attributable variables as a function of their present contribution to PD and be able to predict the upcoming of PD.

There are a number of research centers and foundations which study Parkinson's disease with the aim of providing education to the society about Parkinson's, providing facilities for people with Parkinson's, understanding the Parkinson's disease better, reducing its effect in patients, and finally finding a cure for the Parkinson's. Among them are Na-

tional Parkinson Foundation, Parkinson's Disease Foundation, American Parkinson Disease Association, Davis Phinney Foundation, and Michael J. Fox Foundation for Parkinson's Research.

Through contact with The Michael J. Fox Foundation for Parkinson's Research [97] we were granted access to the vast database of Parkinson's Progression Markers Initiative (PPMI) [100] on different factors related to registered people with PD. Our aim is to study the heredity factors leading to Parkinson's by statistically comparing the existing data on healthy individuals in control group to patients with Parkinson's disease. The available information include whether biological parents, paternal and maternal grandparents, paternal and maternal aunts/uncles, full siblings, and children of test subjects suffered from the disease. The total sample size in our study is 1258. Our results depend on the legitimacy and quality of the provided data. The primary objectives of the present study is to

- Increase the knowledge on the importance of hereditary factors of Parkinson's.
- Test whether the disease rate is the same between families with history of PD and families without history.
- Test whether the disease rate is the same for individuals whose none, one, or both of the parents had Parkinson's.

5.2 Data Exploration

Figure 5.1 shows the number of healthy and diagnosed with Parkinson's disease cases with respect to family history of Parkinson and gender. Out of 1248 valid cases, 503 (40.3%) were females and 745 (59.7%) were males. The total number of cases diagnosed with Parkinson was 898 (72.0%) in comparison to the control healthy group of 350 (28.0%). Overall, in 433 (34.7%) cases there was family history for PD which means at least one of the grandparents, parents, aunts, uncles, full siblings or children had Parkinson.

Figure 5.2, shows the age of the subjects with respect to their disease status and gender. The average age of the people in the healthy group is 61 with a standard deviation

of 12.7 years, in contrast to the 66 with standard deviation of 10.0 years in the Parkinson's disease group. The age difference is significant between the healthy and control group (p-value$<$0.001) with 95% confidence. The age difference is also significant between genders (p-value=0.002).

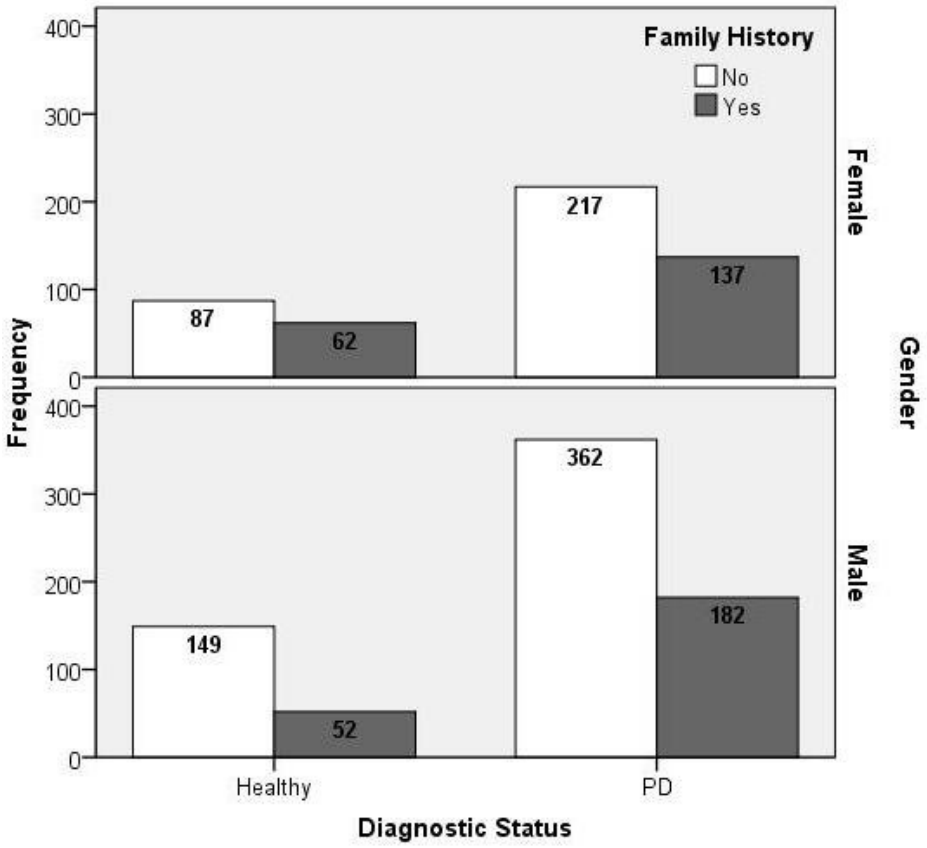


Figure 5.1: Total number of subjects in each category

Breaking the age on disease status shows that in healthy group, female and male test subjects were 59 (S.D.= 12.7) and 63 (S.D.= 12.3) years old in average while in the Parkinson's group females and males were 66 (S.D.= 10) and 66 (S.D.= 9.9) years old in average. The age difference between these four groups was significant using Kruskal-Wallis test (p-value$<$ 0.001). Moreover, Levene's test of variance homogeneity showed four non-homogeneous age groups with respect to both gender and disease status. Non-parametric tests have been used to make comparison between different age groups since the normality tests did not support the normality assumption.

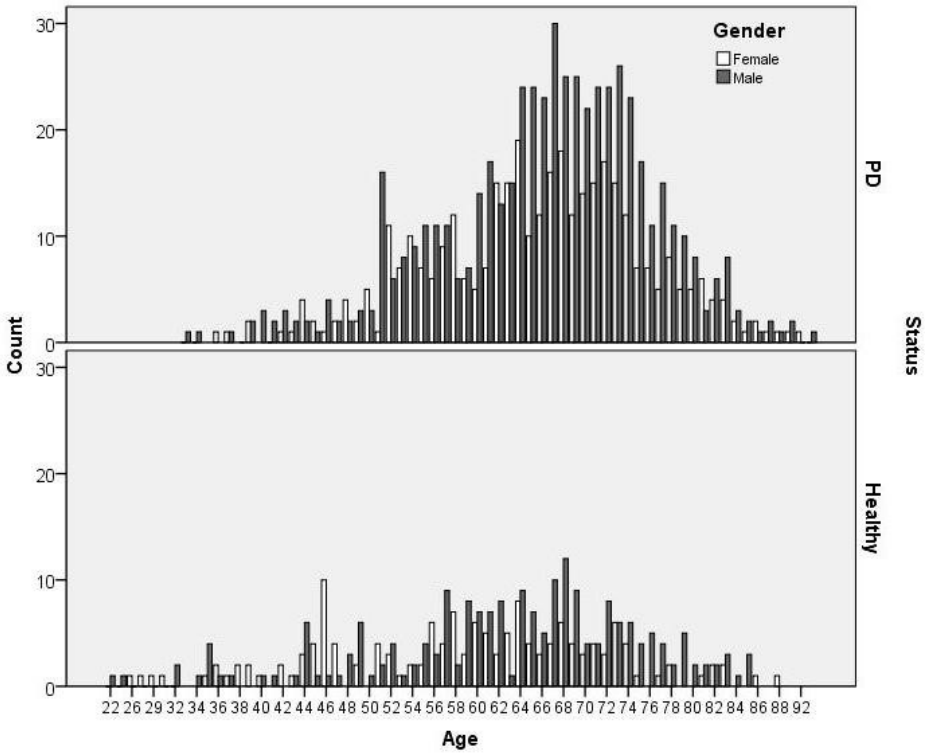


Figure 5.2: Age of the test subjects with respect to Gender and Disease group

Next, we study the extent to which the three variables in Figure 5.1 are associated by testing various hypotheses of complete and partial independence using log-linear model selection. For notations, we will use three subscripts to identify the cells in a $2 \times 2 \times 2$ table; i indexing family history (H), j indexing gender (G) and k indexing diagnostic status (D). Let π_{ijk} denote the probability that an observation falls in cell $(i,\ j,\ k)$. These probabilities define the joint distribution of the three variables. We will use y_{ijk} as the observed count in cell $(i,\ j,\ k)$, which is treated as a realization of random variable Y_{ijk} having a Poisson distribution. Thus, we will treat the Y_{ijk} as independent Poisson random variables with means μ_{ijk}, and we will fit log-linear models to the expected counts.

Table 5.1 provides the results of a backward elimination model selection. In each step, the effect with the highest significance level in the Likelihood Ratio Change is deleted, provided the significance level (Sig.) is greater than 0.05. Statistics are displayed for the best model at each step after step 0. For "Deleted Effect", this is the change in the Chi-Square after the effect is deleted from the model. Based on the results the final model

Table 5.1: Log-linear Model Selection results

Step		Effects	Chi-Sq	df	Sig.	Iterations
0	Generating Class	I*J*K	0	0	.	
	Deleted Effect	I*J*K	3.222	1	0.073	3
1	Generating Class	I*J, I*K, J*K	3.222	1	0.073	
	Deleted Effect	I*J	8.948	1	0.003	2
		I*K	1.16	1	0.282	2
		J*K	1.221	1	0.269	2
2	Generating Class	I*J, J*K	4.382	2	0.112	
	Deleted Effect	I*J	8.763	1	0.003	2
		J*K	1.035	1	0.309	2
3	Generating Class	I*J, K	5.417	3	0.144	
	Deleted Effect	I*J	8.763	1	0.003	2
		K	249.027	1	0	2
4	Generating Class	I*J, K	5.417	3	0.144	

includes all main effects and only the two-way effect of $I \times J$ meaning that 1) all the three variables of family history, gender, and diagnostic status significantly affect the observant; 2) family history and gender are associated but are jointly independent of diagnostic status.

5.3 Modeling

The result in the previous section verifies the importance of the family history and gender in prevalence of Parkinson's. This has been confirmed in other literatures as well [22, 78]. Figure 5.3 shows the outline of the available data to carry out this study. The information included was whether either one of the paternal/maternal grandparents had PD (0 for none, 1 for either one, and 2 for both), whether the biological father/mother carried PD (0 for no and 1 for yes), the number of paternal/maternal aunts/uncles with PD and in total, the number of full/half siblings with PD and in total, and the number of children so far diagnosed with PD. Note that the person himself/herself could be healthy or diagnosed with PD. The numbers in parenthesis shows the number of cases in each category. Unfortunately, we did not have enough information on gender to be able to perform gender related tests and studies. This limitation has been mentioned in discussion and suggestions have been proposed to build a more detailed database for gender studies.

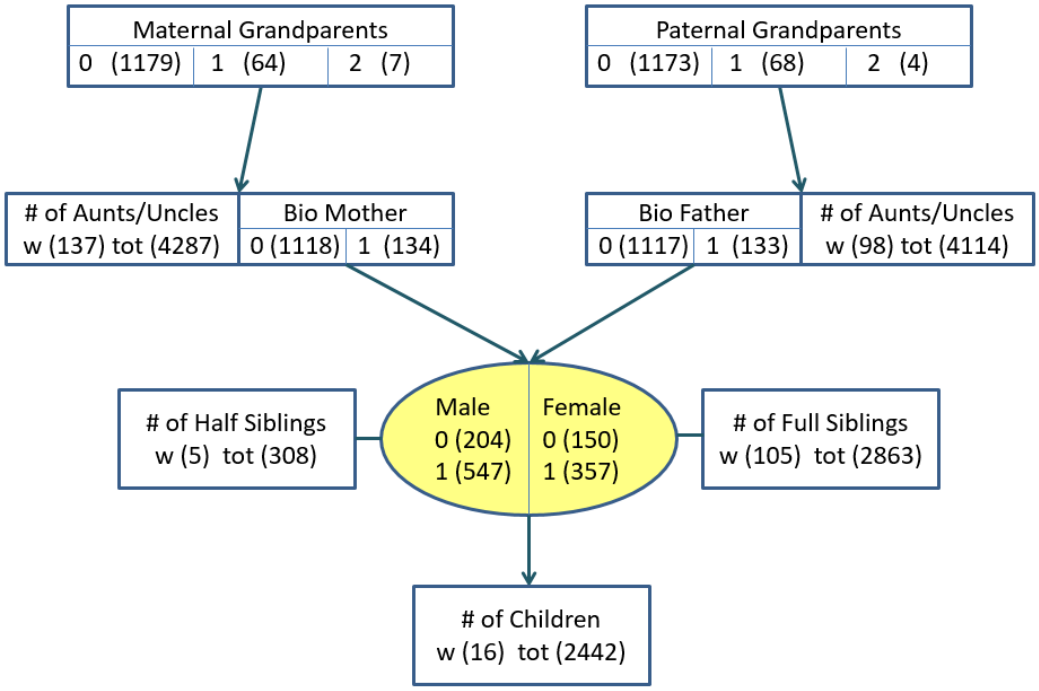


Figure 5.3: Schematic diagram of available data and the counts

Technically, one can follow different approaches using the available data. In order to answer the research questions, we decided to use the approach shown in Figure 5.4 which emphasizes on discovering the hereditary importance of the PD. Therefore, the data is first divided into two exclusive groups based on the heredity factor; negative heredity and positive heredity, $H = 0$ and $H = 1$, respectively. Heredity is considered positive if at least one individual out of grandparents, parents, aunts/uncles, or full siblings carried the PD. Then, cases in each group categorized based on the disease status of parents; for case i, $(F_i, M_i) = (0, 0)$ when none of the parents carried Parkinson's, $(F_i, M_i) = (0, 1)$ when father was healthy and mother was diagnosed with Parkinson's, etc.

In this approach, the number of cases with Parkinson's in each one of the five categories follows a Binomial distribution with two parameters, total number of siblings in the family (n_i), including the person himself/herself, and probability of having Parkinson's (θ). Generally, for case i one can write

$$X_i^{(j,k,l)} | \, (H_i = j, F_i = k, M_i = l) \; \sim \; Bin\left(n_i^{(j,k,l)}, \theta_{jkl}\right), \quad j, k, l = 0, 1, \tag{5.1}$$

where $H_i = j$ with $j = 0, 1$ shows the negative/positive heredity group, $(F_i = k, M_i = l)$ with $k, l = 0, 1$ shows the status Healthy/PD of the parents, $n_i^{(j,k,l)}$ shows the total number of siblings in the family, and θ_{jkl} represents the probability of having PD. The likelihood function can then be written as

$$L(\theta_{jkl}|\mathbf{X}^{(j,k,l)}) = \prod_{i=1}^{k_{jkl}} \binom{n_i^{(j,k,l)}}{x_i^{(j,k,l)}} \theta_{jkl}^{x_i^{(j,k,l)}} (1-\theta_{jkl})^{n_i^{(j,k,l)} - x_i^{(j,k,l)}}, \tag{5.2}$$

where k_{jkl} is the number of cases in each of the five family types represented by $(H_i = j, F_i = k, M_i = l)$. It is easy to arrive at the following maximum likelihood estimator

$$\hat{\theta}_{jkl} = \frac{\sum_i X_i^{(j,k,l)}}{\sum_i n_i^{(j,k,l)}}. \tag{5.3}$$

Table 5.2(a) provides maximum likelihood estimations for parameters θ_{jkl} in each of the five family types plus the number of valid cases (k_{jkl}) in the dataset. The results show that the probability of having PD in families with negative heredity is equal to 0.214. This estimation is based on 824 case subjects. As expected, this probability is higher in families with positive heredity. When none of the parents carry PD the prevalence of PD for an offspring is 0.324. With a drop in chance for families whose one of the parents carry PD, the prevalence increases by transitioning from a mother with PD to a father with PD and increases even more when both parents carry Parkinson's disease.

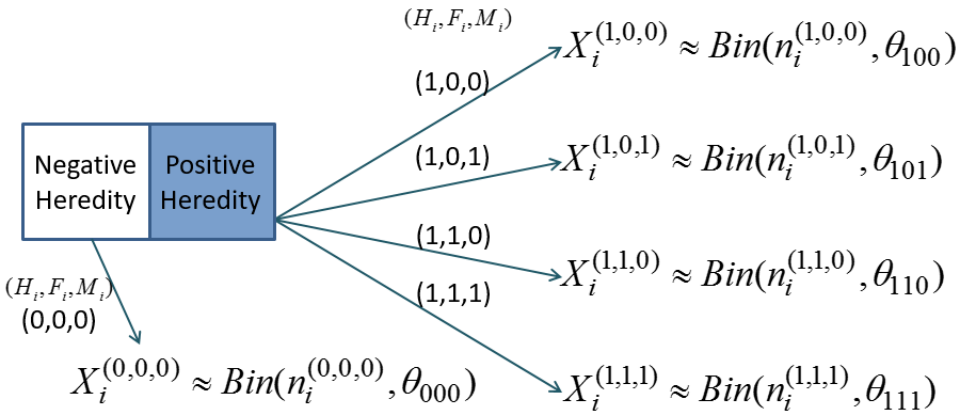


Figure 5.4: Flowchart of modeling approach

Table 5.2: Maximum Likelihood estimations for θ_{jkl} and the number of cases (a) using the parents-individuals link (b) using combined information with grandparents' family

(a)	θ_{000} (k_{000})	θ_{100} (k_{100})	θ_{101} (k_{101})	θ_{110} (k_{110})	θ_{111} (k_{111})
	$\frac{584}{2729} = 0.214$ (824)	$\frac{189}{584} = 0.324$ (165)	$\frac{104}{380} = 0.274$ (125)	$\frac{113}{385} = 0.294$ (124)	$\frac{12}{29} = 0.414$ (9)
(b)	θ_{000} (k'_{000})	θ_{100} (k'_{100})	$\theta_{101/110}$ $(k'_{101/110})$	θ_{111} (k'_{111})	
	$\frac{732}{11728} = 0.062$ (2993)	$\frac{376}{1196} = 0.314$ (280)	$\frac{239}{887} = 0.269$ (253)	$\frac{19}{72} = 0.264$ (20)	

In deriving estimations of Table 5.2(a) only the link between parents and the individual plus his/her siblings have been used. One can use the link between the person's grandparents and parents to achieve higher number of samples, thus more consistent estimations. The estimations in Table 5.2(b) use the combined likelihood, one from parents-children link and the other from grandparents-parents. Clearly, estimations are significantly different in negative heredity group and where both parents carried PD. This could trigger changing the prevalence through time. Moreover, since no information was provided on the gender of the grandparents with PD, a combined probability has been estimated for the case of θ_{101} and θ_{110}. This combined probability shows a state where either one of the parents carried PD.

The chance of having PD in families with positive PD history is five times more than that of with no history of the disease when none of the parents carry PD. It is about four times more when one or both parents carry the disease. Surprisingly, the chances for developing PD when none of the parents are diagnosed with PD are significantly higher than the case where one or both parents are diagnosed with the disease (p-value=0.00014 for binomial test). This could suggest a dormant gene effect for the Parkinson's.

5.4 Bayesian Approach

The chance of passing PD to next generations depends on many factors and could vary from family to family. This random nature justifies using Bayesian approach for estimations. Moreover, we can use sets of hierarchical information as likelihood-prior and update prior information anytime new observations are added to the dataset.

To conduct a Bayesian approach data in Table 5.2(a) that utilizes the link between individuals plus full siblings and their biological parents is used as likelihood. Notice that we have information on whether paternal/maternal aunts/uncles are diagnosed with PD and whether grandparents carried the disease. We utilize this information as prior to derive Bayesian estimations for the model parameters θ_{jkl} using two different approaches. In the first method the frequency of PD in each of the paternal and maternal grandparents' family is used as discrete prior. In the second method this data is mixed with the information regarding the individual's family and a uniform prior is utilized to derive estimations.

5.4.1 Discrete Prior

Suppose we are interested in selecting a prior for θ_{100}. We first select the cases with positive family history of PD (decided based on the status of grandparents, aunts, and uncles) whose none of the paternal grandparents carried PD ($H = 1, F = 0, M = 0$). Next, in each case we count the total number of paternal aunts/uncles along with the diagnosed with PD cases. Then, we add the status of father as a valid count. The following expression provides frequency of disease using paternal family for a given case

$$\frac{\text{Father's status} + \#\text{ of paternal aunts/uncles with PD}}{1 + \text{total }\#\text{ of paternal aunts/uncles}}. \tag{5.4}$$

Following the same procedure in the maternal family yields frequency of disease using maternal family

$$\frac{\text{Mother's status} + \#\text{ of maternal aunts/uncles with PD}}{1 + \text{total }\#\text{ of maternal aunts/uncles}}. \tag{5.5}$$

These two separate frequencies when computed for each case provide a priori information to be used in estimating θ_{100}. Likewise, we can gather prior information for θ_{110} by frequency of disease in the paternal and maternal families where the grandfather did and grandmother did not have PD. However, the only information we have about the grandparents' families is the sum of PD status of grandmother/grandfather. In that case, we still count the number of PD diagnosed cases but the prior for θ_{101} and θ_{110} would be the same.

Prior information for θ_{111}can be derived using the same technique but in different families with respect to grandparents' status. The same approach is used to derive prior for θ_{000}. Table 5.3 provide the frequency of PD occurrence in the grandparents' family.

Table 5.3: Frequency of PD in the parents/aunts/uncles for

(a) Negative Heredity group

	Frequency	Percent
.000	2021	93.18
.091	2	0.09
.111	4	0.18
.125	3	0.14
.143	9	0.41
.167	7	0.32
.200	15	0.69
.250	20	0.92
.333	35	1.61
.500	31	1.43
1.000	22	1.01
Total	2169	100

(b) Positive Heredity group when ONE of the grandparents had PD

	Frequency	Percent
.000	54	42.19
.111	2	1.56
.125	2	1.56
.167	1	0.78
.200	2	1.56
.250	6	4.69
.333	13	10.16
.400	2	1.56
.429	1	0.78
.500	19	14.84
.571	1	0.78
.667	7	5.47
.750	4	3.13
.875	1	0.78
1.000	13	10.16
Total	128	100

(c) Positive Heredity group when NONE of the grandparents had PD

	Frequency	Percent
.091	2	1.74
.100	2	1.74
.111	2	1.74
.125	5	4.35
.143	10	8.70
.167	11	9.57
.182	1	0.87
.200	6	5.22
.250	10	8.70
.286	5	4.35
.333	21	18.26
.400	6	5.22
.500	16	13.91
.600	3	2.61
.667	6	5.22
.750	4	3.48
.833	2	1.74
1.000	3	2.61
Total	115	100

(d) Positive Heredity group when BOTH of the grandparents had PD

	Frequency	Percent
.000	7	63.64
.167	1	9.09
.250	1	9.09
.500	1	9.09
1.000	1	9.09
Total	11	100

To use these information as discrete priors, The set of $\{0.000, 0.001, 0.002, \ldots, 0.999, 1.000\}$ with 101 values has been used as the support and a weight equal to frequency

values of Table 5.3 has been given to the respective points. Other values that had zero frequency have been given a weight of 0.001 then the probabilities have been assigned by dividing the frequencies using the total summation of the weights. This does not change the mean of the priors significantly and provides a nonzero chance for other points in support when mixed with likelihood. The prior then would be in the following form

$$P\left(\theta_{jkl} = \frac{m}{100}\right) = p_m^{jkl} \quad m = 0, 1, \ldots, 100,\ j, k, l = 0, 1, \tag{5.6}$$

where p_m^{jkl} is derived from Table 5.3 for $m = 0, \ldots, 100$ after adding nonzero weights as described above. Combining the prior with the likelihood given in (5.2) produces the following discrete posterior distribution for the five model parameters θ_{jkl}

$$P\left(\theta_{jkl} = \frac{m}{100}\right) = \frac{p_m^{jkl} m^{\sum_{i=1}^{k_{jkl}} x_i^{(j,k,l)}} (100-m)^{\sum_{i=1}^{k_{jkl}} n_i^{(j,k,l)} - \sum_{i=1}^{k_{jkl}} x_i^{(j,k,l)}}}{\sum_{m=0}^{100} p_m^{jkl} m^{\sum_{i=1}^{k_{jkl}} x_i^{(j,k,l)}} (100-m)^{\sum_{i=1}^{k_{jkl}} n_i^{(j,k,l)} - \sum_{i=1}^{k_{jkl}} x_i^{(j,k,l)}}}. \tag{5.7}$$

Table 5.4 provides the estimated parameters using posterior mean and the credible sets accompanied by their percent coverage. Estimation for θ_{000} is 0.200 whereas for θ_{100} it is equal to 0.3280. The relative risk of having PD in positive heredity families whose none of the parents carried PD is $\frac{0.32801}{0.20012} = 1.64\%$ higher than families with negative heredity. The estimation for θ_{101} and θ_{110} are 0.2649 and 0.3148 respectively both with 99% credible set of [0.25, 0.33]. The chance of developing PD increases to 0.4422 when both parents carry the PD which is 1.35% higher than the families where none of the parents are diagnosed with PD. These estimations are close to the maximum likelihood estimations in Table 5.2.

Table 5.4: Bayesian estimations of the model parameters with discrete prior

Parameter	**Mean**	**S.D.**	**% Coverage**	**Credible Set**
θ_{000}	0.20012	0.001657	99.353	{0.20}
θ_{100}	0.32801	0.008845	94.451	{0.33}
θ_{101}	0.26487	0.031059	98.649	[0.25, 0.33]
θ_{110}	0.31477	0.031213	98.779	[0.25, 0.33]
θ_{111}	0.44222	0.100040	90.687	[0.17, 0.50]

5.4.2 Uniform Prior

In this section the available data from grandparents' family is considered as Binomial counts and is mixed with the data from the individual's family in the form of likelihood and Bayesian estimations are derived by using non-informative uniform priors. The posterior distribution could be written as

$$f\left(\theta_{jkl}\right) = \frac{\theta_{jkl}^{\sum_{i=1}^{k'_{jkl}} x_i^{(j,k,l)}} (1-\theta_{jkl})^{\sum_{i=1}^{k'_{jkl}} n_i^{(j,k,l)} - \sum_{i=1}^{k'_{jkl}} x_i^{(j,k,l)}}}{\int_0^1 \theta_{jkl}^{\sum_{i=1}^{k'_{jkl}} x_i^{(j,k,l)}} (1-\theta_{jkl})^{\sum_{i=1}^{k'_{jkl}} n_i^{(j,k,l)} - \sum_{i=1}^{k'_{jkl}} x_i^{(j,k,l)}} d\theta_{jkl}} \tag{5.8}$$

where k'_{jkl} accounts for the new sample cases in families when $(S_i = j, F_i = k, M_i = l)$ for fixed j, k, l. Since no information regarding the gender of grandparents with Parkinson's was available, the information from this link has been copied for both θ_{101} and θ_{110}. When combined with the primary likelihood, this gives us distinct estimations for θ_{101} and θ_{110}.

The Bayesian computations in this section have been carried out using WinBUGS and Monte Carlo Simulations where three simultaneous chains have been utilized to arrive at convergent and stable estimations. A burn in of 110001 with threads of 150000 long has been used for this part of the analysis. Table 8 provides the results of the estimations.

The model parameter θ_{000} is estimated to be 0.0625 with 95% credible bound of (0.0582, 0.0669). This estimation is close to the maximum likelihood estimation of 0.062. For positive heredity group, θ_{100} through θ_{111} were estimated to be 0.3147, 0.27, 0.2785, and 0.2702, respectively. As expected, all estimations are close to their respective maximum likelihood estimations provided in Table 2(b) since a non-informative uniform prior has been used. The chance of having Parkinson's for an offspring in positive heredity family when none of the parents carry PD is about five times higher than a family with negative heredity. This can be deduced from the estimated relative risk of $\theta_{100}/\theta_{000} = 5.042$ and its significant 95% credible interval (4.51, 5.61). The relative risk of $\theta_{110}/\theta_{101}$ was not statistically significant since its prediction interval included 1. Interestingly, where both of the parents have PD the children were less likely to have PD than the situation where

Table 5.5: Bayesian estimations of the model parameters with uniform prior

Parameter	Mean	S.D	MC Error	CB 2.5%	Median	CB 97.5%
θ_{000}	0.0625	0.00223	5.821E-06	0.0582	0.0625	0.0669
θ_{100}	0.3147	0.01341	3.490E-05	0.2887	0.3145	0.3414
θ_{101}	0.2700	0.01490	3.882E-06	0.2411	0.2698	0.2996
θ_{110}	0.2785	0.01500	3.799E-05	0.2495	0.2783	0.3085
θ_{111}	0.2702	0.05119	1.314E-04	0.1758	0.2682	0.3756
$\theta_{100}/\theta_{000}$	5.042	0.2802	7.246E-04	4.51	5.037	5.61
$\theta_{110}/\theta_{101}$	1.035	0.0800	2.078E-04	0.8873	1.032	1.201
$\theta_{111}/\theta_{000}$	0.8601	0.1672	4.265E-04	0.5543	0.8528	1.207

none of the parents is diagnosed PD. This might suggest the effect of dormant genes or lack of adequate data for the case of both parents positive PD status. This estimation is in accordance with some research studies [38, 67].

Environmental factors and the occupational choices act as one of the most important attributable variables in the PD [67, 98, 86] but we should not ignore the role of hereditary factors. Scientific data over the past several years supports the emerging belief that genetics plays a more significant role in Parkinson's disease than was previously thought [38].

5.5 Contribution

The chance of developing PD in families with negative heredity and in four family types with positive heredity has been estimated using four different approaches, two Maximum Likelihood and two Bayesian. Table 5.6 presents all four estimations and their standard deviation. The information for grandparents and their families date to first and second generations back and might not be as exact as it should be. There were registered cases having 18 and 21 aunts/uncles which might be due to registration error or might represent an extreme case which could affect the entire analysis to some degree. For this reason the first and second generation information of 47 cases that had more than 11 aunts/uncles has been excluded from the present study. It is more reasonable to use former less exact information as prior knowledge and let the more recent and authentic information shape it to more reliable estimations. Thus, we opt to report the Bayesian estimations with discrete prior as the more reliable ones.

Table 5.6: Comparison of the estimations

Parameters	ML		Bayes	
	Primary using 1st link	Secondary using 2nd links	Discrete Prior	Uniform Prior
θ_{000} (SD)	0.214 (0.0079)	0.016 (0.0022)	0.20012 (0.0017)	0.0625 (0.0022)
θ_{100} (SD)	0.324 (0.0194)	0.314 (0.0134)	0.32801 (0.0088)	0.3147 (0.0134)
θ_{101} (SD)	0.274 (0.0229)	0.269 (0.0149)	0.26487 (0.0311)	0.2700 (0.0149)
θ_{110} (SD)	0.294 (0.0232)	0.269 (0.0149)	0.31477 (0.0312)	0.2785 (0.0150)
θ_{111} (SD)	0.414 (0.0915)	0.264 (0.0519)	0.44222 (0.1000)	0.2702 (0.0514)

For negative heredity group estimations of θ_{000} vary from 0.016 to 0.214, both extreme estimations are ML estimations based on sample sizes of 2169 and 824. Increasing sample size should increase the consistency and efficiency of the ML estimations but one has to consider the authenticity of information as well. This difference could also point out the change in prevalence of Parkinson's through generations. The Bayesian method with discrete prior provides an estimation of 0.20012 meaning that a child in this family has a 20% chance of developing Parkinson.

Estimations for θ_{100} is less volatile among four different methods. In this case, Bayesian method with discrete prior estimates a chance of 33% for developing Parkinson's for the children. When compared to θ_{000} a relative risk of 1.59 is derived suggesting 1.59 times more chance of having PD if there is a positive Parkinson history in the family although none of the parents have PD. This estimation is with accordance with findings of a community based study in 1996 [53].

The chance of developing PD in a family whose mother is diagnosed with the disease is estimated to be 0.26487 in comparison to 0.31477 when father has Parkinson's; suggesting that the chance of passing Parkinson's from father to child is slightly higher than passing it from mother to child [53]. Finally, there is 44% chance of developing Parkinson in a family whose both parents have Parkinson's disease.

To build a more detailed database for gender studies we suggest taking note of the gender of cases with PD in relatives. Lack of enough gender related information prevented deriving separate estimations for men and women. Study on twins would also be beneficial to understand the Parkinson's disease better.

5.6 Future Research

Although a primary cause for Parkinson's disease is yet to be identified, a number of risk factors are known to be contributing to the disease. Among them are advancing age, sex, family history, declining estrogen levels, environmental factors, levels of B vitamin, and head trauma. Multiple research have been done about the importance of each of these risk factors [90]. However, there has not been a single research on the degree of significance that each risk factor has on the disease.

A proposed future research is to gather data and perform a fundamental analysis to identify statistically the attributable risk factors of the PD. This is extremely important in controlling the disease: prevention, progression, and minimization of developing PD. Ranking the various risk factors according to their percentage of contribution across Parkinson's enable us to determine the most important attributable variables. Thus we propose to develop such a statistical model to identify and rank the risk factors and their interactions. In addition, by having such a model we will be able to use it for the accurate prediction of the response which is PD. We will also be able to apply surface response analysis to identify the contribution of the risk factors to minimize the risk of the Parkinson's disease in a given patient.

There are no lab tests that can diagnose Parkinson's disease. If a doctor isn't sure a patient has Parkinson's, he/she may do certain tests to see if patient has another condition with similar symptoms. For instance, blood tests may be done to check for abnormal thyroid hormone levels or liver damage. An imaging test such as a CT scan or an MRI may be used to check for signs of a stroke or brain tumor. PET tests sometimes may detect low levels of dopamine in the brain, a key feature of Parkinson's, but PET scanning isn't commonly used to evaluate Parkinson's because it is very expensive, it is not available in many hospitals, and it is only used experimentally.

We propose a research in using Electroencephalogram (EEG) to detect Parkinson's disease and keep track of its progress. EEG has several strong advantages as a tool for exploring brain activity. Beside its ease of use, portability and low set up cost in comparison to other brain study tools, it can detect changes within a millisecond time frame which is

excellent considering an action potential takes approximately 0.5-130 milliseconds to propagate across a single neuron, depending on the type of neuron. EEG measures the brain's electrical activity directly, while other methods record changes in blood flow (e.g., SPECT, fMRI) or metabolic activity (e.g., PET), which are indirect markers of brain electrical activity.

Parkinson's disease is a neurological disorder that affects brain activity in one or more ways. Investigation of EEG abnormalities in the early stage of Parkinson's disease shows detectable changes in different frequency bands of brain activities [21] making potential use of EEG as biomarkers of PD in the early stage.

REFERENCES

[1] Aarsland, D., Kramberger, GM. (2015) Neuropsychiatric Symptoms in Parkinson's Disease, Journal of Parkinson's Disease. 5(3), 659–667.

[2] Abbas, A.K., Bassam, R., Phonocardiography Signal Processing, Morgan and Claypool Publishers, 2009.

[3] Abdulkader, S.N., Atia, A., Mostafa, M.M. (2015) Brain computer interfacing: Applications and challenges, Egyptian Informatics Journal. 12(2), 213–230.

[4] Addison, P.S. (2005) Wavelet transforms and the ECG: a review, Physiological Measurement. 26(5), 155-199.

[5] Ari, S., Hembram, K., Saha, G. (2010) Detection of cardiac abnormality from PCG signal using lms based least square svm classier, Expert Syst Appl. 37, 8019–8026.

[6] Azevedo, F.A., Carvalho, L.R., Grinberg, L.T., Farfel, J.M., Ferretti, R.E., Leite, R.E., Jacob Filho, W., Lent, R., Herculano-Houzel, S. (2009) Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled-up primate brain, Journal of Comparative Neurology. 513(5), 532–541.

[7] Bagnal, A., Lines, J., Bostrom, A., Large, J., Keogh, E. (2017) The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. 71(3), 606–660.

[8] Bagnall, A., Lines, J., Hills, J., Bostrom, A. (2015) Time-series classification with COTE: the collective of transformation-based ensembles, IEEE Trans Knowl Data Eng. 27, 2522-2535.

[9] Bentley, P.M., Nokia, R.D., Camberley, U.K., Grant, P.M., McDonnell, J.T.E. (1998) Time-frequency and time-scale techniques for the classification of native and bioprosthetic heart valve sounds, IEEE Trans Biomed Eng. 45, 125–128.

[10] Bernard, S.C., Donald, L.S., Ernst, N., Normal EEG and sleep: Adults and elderly, In Niedermeyer's Electroencephalography: Basic Principles, Clinical Applications and Related Fields, L. S. Donald and H. L. Fernando, Eds., 6th ed., Philadelphia, PA: Lippincott Williams & Wilkins, 2011, 183–214.

[11] Bernsdorf, M., Berthelsen, A.K., Wielenga, V.T., Kroman, N., Teilum, D., Binderup, T., Tange, U.B., Andersson, M., Kjær, A., Loft, A., Graff, J. (2012) Preoperative PET/CT in early-stage breast cancer, Annals of ncology. 23(9), 2277–2282.

[12] Bishop, C.M, Pattern Recognition and Machine Learning, Springer, 2006.

[13] Blankertz, B., Tomioka, R., Lemm, S., Kawanabe, M., Muller, K.R. (2008) Optimizing Spatial filters for Robust EEG Single-Trial Analysis, IEEE Signal Processing Magazine. 25(1), 41–56.

[14] Boser, B.E., Guyon, I.M., Vapnik, V.N., A training algorithm for optimal margin classifiers, Proceedings of the fifth annual workshop on Computational learning theory (COLT92), Pittsburgh, PA, USA, July 1992.

[15] Burges, C.J.C. (1998) A Tutorial on Support Vector Machines for Pattern Recognition, Data Min Knowl Discov. 2(2), 121-167.

[16] Carter, B.S., Beaty, T.H., Steinberg, G.D., Childs, B., Walsh, P.C. (1992) Mendelian inheritance of familial prostate cancer, Proceedings of the National Academy of Sciences of the United States of America. 89, 3367-3371.

[17] Chang, H.H., Moura, J.M.F., Biomedical Signal Processing, in Biomedical Engineering and Design Handbook, Volume 1, McGraw Hill, 2010, 559–579.

[18] Chang, H.H., Moura, J.M.F., Wu, Y.L., Ho, C., Immune cells detection of in vivo rejecting hearts in USPIO-enhanced magnetic resonance imaging, Proceedings of IEEE International Conference of Engineering in Medicine and Biology Society, New York, US, August 2006.

[19] Chang, H.H., Moura, J.M.F., Wu, Y.L., Ho, C. (2008) Automatic detection of regional heart rejection in USPIO-enhanced MRI, IEEE Transactions on Medical Imaging. 27(8), 1095-1106.

[20] Charleston, S., Azimi-Sadjadi, M.R. (1996) Reduced order Kalman ?ltering for the enhancement of respiratory sounds, IEEE Transactions on Biomedical Engineering. 43, 421-24.

[21] Chun-Xiao, H., Jiang, W., Guo-Sheng, Y., Yan-Qiu, C. (2013) Investigation of EEG abnormalities in the early stage of Parkinson's disease, Cognitive Neurodynamics. 7(4): 351–359.

[22] Cilia, R., Siri, C., Rusconi, D., Allegra, R., Ghiglietti, A., Sacilotto, G., Zini, M., Zecchinelli, A.L., Asselta, R., Duga, S., Paganoni, A.M., Pezzoli, G., Seia, M., Goldwurm, S. (2014) LRRK2 mutations in Parkinson's disease: Confirmation of a gender effect in the Italian population, Parkinsonism & Related Disorders. 20(8), 911–914.

[23] Clancy, E.A., Morin, E.L., Merletti, R. (2002) Sampling, noise-reduction and amplitude estimation issues in surface electromyography, Journal of Electromyography and Kinesiology. 12, 1-16.

[24] Machine Learning, an online non-credit course authorized by Stanford University and offered through Coursera, Accessed on 2015

[25] Correa, A.G., Orosco, L., Laciar, E. (2014) Automatic detection of drowsiness in EEG records based on multimodal analysis. Medical Engineering & Physics. 36(2), 244–249.

[26] Delorme, A., Makeig, S. (2004) EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics, Journal of Neuroscience Methods. 134, 9–21.

[27] De Vos, J.P., Blanckenberg, M.M. (2007) Automated pediatric cardiac auscultation. IEEE Trans Biomed Eng. 54, 244–252.

[28] Estévez, P.A., Held, C.M., Holzmann, C.A., Perez, C.A., Prez, J.P., Heiss, J., Garrido, M., Peirano. P. (2002) Polysomnograhic Pattern Recognition for Automated Classification of Sleep-Waking States in Infants, Medical and Biological Engineering and Computing. 40(1), 105–113.

[29] Fisher, R.A. (1936) The use of multiple measurements in taxonomic problems, Annals of Eugenics. 7(2), 179-188.

[30] Fox, E.B., Bayesian Nonparaetric Learning of Complex Dynamical Phenomena, Doctoral Thesis, Massachusetts Institute of Technology, July 2009.

[31] Frisch, K.V., The Dance Language and Orientation of Bees, Cambridge, MA: Harvard University Press, 1967.

[32] Gales M., Multi-Layer Perceptron: Introduction and Training, University of Cambridge Engineering Part IIB & EIST Part II, 2001.

[33] Garcés Correa, A., Orosco, L., Laciar, E. (2014) Automatic detection of drowsiness in EEG records based on multimodal analysis, Medical Engineering & Physics. 36(2), 244–249.

[34] Genuth, I. (2015) All in the mind [EEG], Engineering & Technology. 10(5), 37–39.

[35] Gerbarg, D.S., Taranta, A., Spagnuolo, M., Hofler, J.J. (1963) Computer analysis of phonocardiograms, Progress in Cardiovascular Diseases. 5, 393–405.

[36] Grauman, K., Betke, M., Lombardi, J., Gips, J., Bradski, G.R. (2003) Communication via eye blinks and eyebrow raises: video-based human-computer interfaces, Universal Access in the Information Society. 2(4), 359–373.

[37] Grime, S.L. (2011) Using 80001 to manage medical devices on IT network, IT Horizons. 6, 23–26.

[38] Gwinn, K. enetics and Parkinson's disease: What Have We Learned? Winter 2009 Newsletter, News & Review, Parkinson's Disease Foundation, Inc., 2009.

[39] Hazarika, N., Tsoi, A.C., Sergejew, A.A. (1997) Nonlinear considerations in EEG signal classification, IEEE Transactions on Signal Processing. 45, 829-836.

[40] Huabiao, Q., Jun, L., Tianyi, H., An Eye State Identification Method Based on the Embedded Hidden Markov Model, IEEE International Conference on Vehicular Electronics and Safety, Istanbul, Turkey, July 2012, 255–260.

[41] Huang, N.E., Shen, Z., Long, S.R., Wu, M.C., Shih, H.H., Zheng, Q., Yen, N.C., Tung, C.C., Liu, H.H. (1998) The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis, Proc. R. Soc. A. 454, 903–995.

[42] Hulbert, S., Adeli, H. (2013) EEG/MEG- and imaging-based diagnosis of Alzheimer's disease, Reviews in Neurosciences. 24(6), 563–576.

[43] Jumeily, D.A., Iram, S., Hussain, A.J., Francois, V., Fergus, P. (2014) Early Detection Method of Alzheimers Disease Using EEG Signals, International Conference on Intelligent Computing (ICIC2014), Taiyuan, China, August 2014

[44] Kamel, N., Malik, A.S., EEG/ERP Analysis Methods and Applications, CRC Press, 2015.

[45] Koles, Z.J. (1991) The quantitative extraction and topographic mapping of the abnormal components in the clinical EEG, Electroencephalography and Clinical Neurophysiology. 79(6), 440–447.

[46] Królak, A., Strumio, P. (2012) Eye-blink detection system for human–computer interaction, Universal Access in the Information Society. 11(4), 409–419.

[47] Li, L., The Differences among Eyes-Closed, Eyes-Open and Attention States: An EEG Study, Proc. 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM), Chengdu City, China, September 2010.

[48] Liang, H., Hartimo, I., In A feature extraction algorithm based on wavelet packet decomposition for heart sound signals, Proceedings of the IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis, Pittsburgh, PA, 1998.

[49] Maglogiannis, I., Loukis, E., Zafiropoulos, E., Stasis, A. (2009) Support vectors machine-based identification of heart valve diseases using heart sounds. Comput Methods Programs Biomed. 95, 47–61.

[50] Majdalawieh, O., Gu, J., Bai, T., Cheng, G., Biomedical signal processing and rehabilitation engineering: A review. Proceedings of IEEE Paci?c Rim Conference on Communications, Computers and Signal Processing, Victoria, Canada, August 2003.

[51] Mandic, D., Rehman, N., Wu, Z., Huang, N. (2013) Empirical Mode Decomposition-based Time-Frequency Analysis of Multivariate Signals, IEEE Signal Processing Magazine. 30(6), 74–86.

[52] Manrique, A.F., Godino-Llorente, J.I., Blanco-Velasco, M., Castellanos-Dominguez, G. (2010) Selection of dynamic features based on time-frequency representations for heart murmur detection from phonocardiographic signals, Ann Biomed Eng. 38, 118–137.

[53] Marder, K., Tang, M.X., Mejia, H., Alfaro, B., Cote, L., Louis, E., Groves, J., Mayeux, R. (1996) Risk of Parkinson's disease among first-degree relatives, Neurology. 47(1), 155–160.

[54] Mardi, Z., Ashtiani, S.N.M., Mikaili, M. (2011) EEG-based drowsiness detection for safe driving using chaotic features and statistical tests, Journal of Medical Signals and Sensors. 1(2), 130–137.

[55] McCulloch, W.S., Pitts, W. (1943) A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics. 5(6), 115–133.

[56] Millette, V., Baddour, N. (2011) Signal processing of heart signals for the quantification of non-deterministic events, BioMedical Engineering. 10(10), 1–23.

[57] Morse, R.A., Calderone, N.W. (2000) The value of honey bees as pollinators of US crops in 2000, Bee Culture. 128, 1-15.

[58] Murphy, K.P., Machine Learning: A Probabilistic Perspective, MIT Press, 2012

[59] Norizam, S., Taib, M.N., Mohd Aris, S.A., Abdul Hamid, N.H., Lias, S., Murat, Z., Stress features identification from EEG signals using EEG Asymmetry & Spectral Centroids techniques, Proc. IEEE EMBS Conference on Biomedical Engineering and Sciences, Kuala Lumpur, Malaysia, 2010, 417–421.

[60] Novák, D., Electrocardiogram Signal Processing using Hidden Markov Models, PhD dissertation, Czech Technical University, Prague, Czech Republic, 2003.

[61] Oh, S.M., Rehg, J.M., Balch, T., Dellaert, F. (2008) Learning and Inferring Motion Patterns using Parametric Segmental Switching Linear Dynamic Systems, International Journal of Computer Vision (IJCV) Special Issue on Learning for Vision. 77(1-3), 103–124.

[62] Palreddy, S., Hu, Y.H., Mani, V., Tompkins, W.J., A multiple-classifier architecture for ECG beat classification, Proceedings of the IEEE Workshop on Neural Networks for Signal Processing, September 1997.

[63] Park, C., Looney, D., Rehma, N., Ahrabian, A., Mandic, D. (2013) Classification of Motor Imagery BCI using Multivariate Empirical Mode Decomposition, IEEE Transactions on Neural Systems and Rehabilitation Engineering. 21(1), 10–21.

[64] Philips, W. (1996) Adaptive noise removal from biomedical signals using warped polynomials, IEEE Transactions on Biomedical Engineering. 43, 480-492.

[65] Potocnik, P., Neural Networks Practical Examples, Laboratory of Synergetics (LASIN), University of Ljubljana, 2012.

[66] Preece, K., Beekman, M. (2014) Honeybee waggle dance error: adaption or constraint? Unravelling the complex dance language of honeybees, Animal Behaviour. 94, 19–26.

[67] Priyadarshia, A., Khudera, S.A., Schauba, E.A., Priyadarshi, S.S. (2001) Environmental Risk Factors and Parkinson's Disease: A Meta-Analysis, Environmental Research. 86(2), 122–127.

[68] Rehman, N., Mandic, D.P. (2009) Multivariate Empirical Mode Decomposition, Proc. R. Soc. A. 466(2117), 1291–1302.

[69] Riley, J.R., Greggers, U., Smith, A.D., Reynolds, D.R., Menzel, R. (2005) The flight paths of honeybees recruited by the waggle dance, Nature. 435, 205–207.

[70] Rösler, O., Suendermann, D., First step towards eye state prediction using EEG. Proc. International Conference on Applied Informatics for Health and Life Sciences (AIHLS 13), Istanbul, Turkey, September 2013.

[71] Rösler, O., Bader, L., Forster, J., Hayashi, Y., Heßler, S., Suendermann, D., Comparison of EEG Devices for Eye State Classification, International Conference on Applied Informatics for Health and Life Sciences (AIHLS 14), Kusadasi, Turkey, 2014.

[72] Russell, S., Norvig, P., Artificial Intelligence: A Modern Approach (2nd ed.). Prentice Hall, 2003

[73] Sabancı, K., Koklu, M. (2015) The Classification of Eye State by Using kNN and MLP Classification Models According to the EEG Signals, International Journal of Intelligent Systems and Applications in Engineering. 3(4), 127–130.

[74] Saghafi, A., Tsokos, C.P., Goudarzi, M., Farhidzadeh H. (2017) Random Eye State Change Detection in Real-Time using EEG Signals, Expert Systems With Applications. 72(15), 42–48.

[75] Saghafi, A., Tsokos, C.P., Farhidzadeh H. (2017) A Common Spatial Pattern Method for Real-time Eye State Identification by using EEG Signals, IET Signal Processing. Under Revision

[76] Saghafi, A., Tsokos, C.P., Wooten, R.D., On Heredity Factors of Parkinson's disease: A Parametric and Bayesian Analysis, Under Review

[77] Saracoglu, R. (2012) Hidden markov model-based classification of heart valve disease with pca for dimension reduction, Eng Appl Artif Intell. 25, 1523–1528.

[78] Saunders, R., Stanley, K., San Luciano, M., Barrett, M.J., Shanker, V., Raymond, D., Ozelius, L.J., Bressman, S.B. (2012) Gender differences in the risk of familial parkinsonism: beyond LRRK2, Neurosci Lett. 496(2) 125–128.

[79] Schäfer, P., Leser, U., Benchmarking Univariate Time Series Classifiers, Lecture Notes in Informatics (LNI), 2017.

[80] Schäfer, P. (2016) Scalable time series classification, Data Mining and Knowledge Discovery. 30(5), 1273-1298.

[81] Schalk, G., Mellinger, J., A Practical Guide to Brain-Computer Interfacing with BCI2000, Springer Science & Business Media, 2010.

[82] Schmidt, S., Graebe, M., Toft, E., Struijk, J. (2011) No evidence of nonlinear or chaotic behavior of cardiovascular murmurs. Biomed Signal Process Control. 6, 157–163.

[83] Schmidt, S.E., Holst-Hansen, C., Graff, C., Toft, E., Struijk, J.J. (2010) Segmentation of heart sound recordings by a duration-dependent hidden markov model, Physiol Meas. 31, 513–529.

[84] Springer, D.B., Tarassenko, L., Clifford, G.D. (2016) Logistic regression-hsmm-based heart sound segmentation, IEEE Trans Biomed Eng. 63(4), 822–832.

[85] Sulaiman, N., Mohd, N.T., Siti, A.M.A., Noor, H.A.H., Lias, S., Murat, Z.H., Stress features identification from EEG signals using EEG Asymmetry & Spectral Centroids techniques, IEEE EMBS Conference on Biomedical Engineering and Sciences (IECBES), Kuala Lumpur, Malaysia, 2010.

[86] Sullivan, K.L., Mortimer, J.A., Wang, W., Zesiewicz, T.A., Brownlee, H.J., Borenstein, A.R. (2015) Occupational Characteristics and Patterns as Risk Factors for Parkinson's Disease: A Case Control Study, Journal of Parkinson's Disease. 5(4), 813–820.

[87] Tan, D.S., Nijholt, A., Brain-Computer Interfaces, Springer-Verlag London, 2010.

[88] Xuan, X., Murphy, K., Modeling changing dependency structure in multivariate time series, Proc. of International Conference on Machine Learning, June 2007.

[89] Wang, T., Guan, S.U., Man, K.L., Ting, T.O. (2014) EEG Eye State Identification Using Incremental Attribute Learning with Time-Series Classification, Mathematical Problems in Engineering. 2014(2014), 158–161.

[90] Weyhenmeyer, J., Hernandez, M.E., Lainscsek, C., Sejnowski, T.J., Poizner, H. (2014) Muscle artifacts in single trial EEG data distinguish patients with Parkinson's disease from healthy individuals. Conf Proc IEEE Eng Med Biol Soc. 3292–3295.

[91] Britannica Online Dictionary, Honeybee: dance to indicate location of a food source, Accessed May 2017, www.britannica.com/animal/honeybee

[92] The Cardiac Cycle, Boundless Biology Boundless. Accessed April 2017, www.boundless.com/biology/textbooks/boundless-biology-textbook/the-circulatory-system-40/mammalian-heart-and-blood-vessels -226/the-cardiac-cycle-852-12097.

[93] Center for Disease Control and Prevention, Drowsy Driving: Asleep at the Wheel, Accessed January 2016, www.cdc.gov/features/dsdrowsydriving

[94] Components of the Human Connectome Project: EEG MEG, Website Accessed on April 2017, www.humanconnectome.org/about/project/MEG-and-EEG.html

[95] EMOTIV EPOC Headset, Accessed January 2016, www.emotiv.com

[96] FAO (2015) Agricultural Statistics, Accessed May 2017, www.faostat.fao.org/site/567/default.aspx

[97] Michael J. Fox Foundation for Parkinson's Research, Accessed on March 2015, www.michaeljfox.org

[98] Michael J. Fox Foundation for Parkinson's Research, Questions and Answers, Accessed on March 2015, www.michaeljfox.org/understanding-parkinsons/i-have-got-what.php

[99] Parkinson Disease Foundation, Accessed on March 2015, www.pdf.org/about_pd

[100] Parkinson's Progression Markers Initiative (PPMI) database, Accessed on March 2015, www.ppmi-info.org/data

[101] Classification of Normal/Abnormal Heart Sound Recordings: the PhysioNet/Computing in Cardiology Challenge 2016, Accessed December 2016, www.physionet.org/challenge/2016/

[102] Chen, Y., Keogh, E., Hu, B., Begum, N., Bagnall, A., Mueen, A., Batista, G., The UCR Time Series Classification Archive, www.cs.ucr.edu/ẽamonn/time_series_data, 2015.

[103] Frank, A., Asuncion, A., UCI Machine Learning Repository, http://archive.ics.uci.edu/ml, 2010.

APPENDIX A

MATLAB CODES

```
%% Example 1.5. Iris data. Classification using Logistic Regression
data=importdata('iris.txt');
filter=data(:,5)~=3; \% filter out Virginica
X=data(filter,[1,2]);
y=data(filter,5)~=1;
[m,n]=size(X);
X = [ones(m, 1) X];

initial_theta=zeros(n+1,1);
[cost,grad]=costFunction(initial_theta,X,y);
options=optimset('GradObj','on','MaxIter',400,'Display','iter');
[theta,cost]=fminunc(@(t)(costFunction(t,X,y)),initial_theta,options);
fprintf('\%f\n',theta);
plotDecisionBoundary(theta,X,y);

% The costFunction is given by
function[J,grad]=costFunction(theta,X,y)
m=length(y); J=0;
grad=zeros(size(theta));
hyp=sigmoid(X*theta);
J=-sum((y.*log(hyp))+((1-y).*log(1-hyp)))/m;
grad=((hyp-y)'*X)./m;
grad=grad';
end
```

```
%% Example 1.6. XOR problem. Classification using ANN
K=100;     \% number of examples in each cluster
q=0.6;      \% offset of classes
A=[rand(1,K)-q; rand(1,K)+q];
B=[rand(1,K)+q; rand(1,K)+q];
C=[rand(1,K)+q; rand(1,K)-q];
D=[rand(1,K)-q; rand(1,K)-q];

plot(A(1,:),A(2,:),'k+')
hold on
grid on
plot(B(1,:),B(2,:),'bd')
plot(C(1,:),C(2,:),'k+')
plot(D(1,:),D(2,:),'bd')

a=-1; c=-1; b=1;  d=1;
P=[A B C D];
T=[repmat(a,1,length(A)) repmat(b,1,length(B)) ...
repmat(c,1,length(C)) repmat(d,1,length(D))];

net=feedforwardnet(3);
net.divideParam.trainRatio = 1; % training set [%]
net.divideParam.valRatio = 0; % validation set [%]
net.divideParam.testRatio = 0; % test set [%]
[net,tr,Y,E] = train(net,P,T);
view(net)        % show network

span = -1:.005:2;
[P1,P2]=meshgrid(span,span);
pp=[P1(:) P2(:)]';
aa=net(pp);  % simulate neural network on a grid
mesh(P1,P2,reshape(aa,length(span),length(span))-5);
colormap cool
```

```
%% Example 1.7. XOR problem. Classification using SVM
P=P'; T=T';
figure;
gscatter(P(:,1),P(:,2),T);
hold on

% Using RBF Kernel
svmstruct = svmtrain(P,T,'Kernel_Function','rbf','rbf_sigma',0.1, ...
    'showplot','true')

% Using Polynomial Kernel
svmstruct = svmtrain(P,T,'Kernel_Function','polynomial', ...
    'Polyorder',3,'showplot','true')
```