

UNIVERSITY OF SOUTHAMPTON

Faculty of Engineering and Physical Sciences
School of Electronics and Computer Science

Support Vector Machines for Real-Time Seizure Prediction using Neural Data

by

Sameen Islam

September 2021

Supervisor: Nicolas Green
Second Examiner: Mahesan Niranjan

*A dissertation in partial fulfillment
of the requirements for the degree of
Master of Science in Artificial Intelligence*

University of Southampton

Abstract

Faculty of Engineering and Physical Sciences
School of Electronics and Computer Science

Master of Science

Support Vector Machines for Real-Time Seizure Prediction using Neural Data

by Sameen Islam

We investigate the feasibility of predicting seizures using real-time EEG data using two feature extraction methods: autocorrelation moving average (ARMA) and neural oscillatory band power (spectral power). We then use a support vector machine (SVM) with a radial basis kernel (RBF) to generate a prediction signal which is then regularised using two methods: moving average filter (MA) and Kalman filter (KF), to provide a less noisy signal. We compare the performance of these two filters to observe the regularisation performance differences on a prediction signal generated in real-time. Using a stream buffer, our system achieves a prediction frequency of 2 Hz on ARMA mode and 35 Hz on spectral power mode. We have shown in spectral power mode, our system achieves prediction performance above random average which serves as a strong indication that real-time seizure prediction with EEG is feasible.

Acknowledgements

I would like to express my gratitude to my supervisor, Nicolas Green, who has guided me throughout this project and my second examiner, Mahesan Niranjan, who was also my lecturer during the semester, who has given me so much wisdom on machine learning. I also thank the lab staff in the University of Southampton who have supported me by facilitating an excellent working environment amidst a global pandemic.

Declaration of Authorship

I declare that this thesis and the work presented in it is my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. None of this work has been published before submission

Signed: Sameen .

Date: 02.09.2021

Contents

| | |
|--|-------------|
| Acknowledgements | ii |
| Declaration of Authorship | iii |
| List of Figures | vi |
| List of Tables | viii |
| Definitions and Abbreviations | x |
| 1 Introduction | 1 |
| 1.1 What are Seizures? | 1 |
| 1.2 Importance of Real-Time Seizure Prediction | 1 |
| 1.3 Challenges in Seizure Prediction | 2 |
| 1.4 Related Work | 2 |
| 1.5 Comments on Related Work | 3 |
| 1.6 Dissertation Contribution | 4 |
| 1.7 Dissertation Outline | 4 |
| 2 Background | 5 |
| 2.1 Scalp EEG | 5 |
| 2.2 Seizures Within EEG | 6 |
| 2.3 Neural Bands | 6 |
| 2.4 EEG Electrodes | 7 |
| 2.5 Online Prediction | 8 |
| 2.6 Linear Least Squares | 8 |
| 2.7 AR Modelling | 9 |
| 2.8 Evaluation Metrics | 9 |
| 3 Methodology | 11 |
| 3.1 Research Problem | 11 |
| 3.2 Data | 11 |
| 3.2.1 Selection Criteria | 12 |
| 3.2.2 Preictal State | 12 |
| 3.3 Detecting Pre-Seizure States | 13 |
| 3.4 Architecture | 13 |
| 3.5 Feature Extraction | 14 |
| 3.5.1 ARMA | 14 |

| | | |
|----------|----------------------------------|-----------|
| 3.5.2 | Spectral Power | 15 |
| 3.6 | Classification | 15 |
| 3.6.1 | Support Vector Machine | 15 |
| 3.7 | Regularisation | 17 |
| 3.7.1 | MA | 17 |
| 3.7.2 | Kalman Filter | 17 |
| 3.8 | Training | 19 |
| 3.9 | Testing | 19 |
| 4 | Results | 20 |
| 4.1 | Feature Extraction | 20 |
| 4.1.1 | ARMA | 20 |
| 4.1.2 | Spectral Power | 21 |
| 4.2 | Model Comparison | 23 |
| 4.2.1 | ARMA | 23 |
| 4.2.2 | Spectral Power | 27 |
| 4.3 | Model Test | 29 |
| 4.3.1 | ARMA | 29 |
| 4.3.2 | Spectral Power | 30 |
| 4.4 | Regularisation | 33 |
| 4.5 | Summary | 33 |
| 5 | Discussion | 34 |
| 6 | Conclusion | 38 |
| | Appendix A Dataset | 39 |
| | Appendix B Figures | 43 |
| | References | 50 |

List of Figures

| | | |
|--------------|---|----|
| 2.1 | The frontal, parietal, occipital and temporal lobes in the brain. ¹ | 5 |
| 2.2 | Visualising the preictal, ictal and interictal periods in an EEG recording. | 6 |
| 2.3 | Neural band power response for EEG sample. | 7 |
| 2.4 | The international 10-20 system defining the location of scalp electrode placement position. | 8 |
| 2.5 | Linear least squares model fitted on 2-dimensional regressors. | 9 |
| 2.6 | Online estimation of AR parameters using synthetic observation signal. | 10 |
| 2.7 | Confusion matrix for binary classification. | 10 |
| 3.1 | Dimensionality reduction of neural data through ARMA and spectral features. | 14 |
| 3.2 | δ band on a power spectrum. | 16 |
| 4.1 | Class distribution of ARMA features from chb01 training data. | 22 |
| 4.2 | Class distribution of ARMA features from chb06 training data. | 22 |
| 4.3 | Class distribution of ARMA features from chb10 training data. | 22 |
| 4.4 | Class distribution of spectral features from chb01 training data. | 24 |
| 4.5 | Class distribution of spectral features from chb06 training data. | 24 |
| 4.6 | Class distribution of spectral features from chb10 training data. | 24 |
| 4.7 | Prediction with ARMA based RBF SVM using chb01 training set. | 26 |
| 4.8 | Prediction with ARMA based RBF SVM using chb06 training set. | 26 |
| 4.9 | Prediction with ARMA based RBF SVM using chb10 training set. | 26 |
| 4.10 | Prediction with spectral power based RBF SVM using chb01 training set. | 28 |
| 4.11 | Prediction with spectral power based RBF SVM using chb06 training set. | 28 |
| 4.12 | Prediction with spectral power based RBF SVM using chb10 training set. | 28 |
| 4.13 | Prediction with ARMA based RBF SVM using chb01 test set. | 31 |
| 4.14 | Prediction with ARMA based RBF SVM using chb06 test set. | 31 |
| 4.15 | Prediction with ARMA based RBF SVM using chb10 test set. | 31 |
| 4.16 | Prediction with spectral power based RBF SVM using chb01 test set. | 32 |
| 4.17 | Prediction with spectral power based RBF SVM using chb06 test set. | 32 |
| 4.18 | Prediction with spectral power based RBF SVM using chb10 test set. | 32 |
| 4.19 | Comparison of Kalman filter with moving average filter on prediction regularisation. | 33 |
| 5.1 | Performance of proposed algorithm compared against random predictor. | 37 |
| Appendix A.1 | Seizure count by patients in overall dataset. | 39 |
| Appendix A.2 | Channel count by patients in overall dataset. | 39 |

| | | |
|---------------|--|----|
| Appendix B.1 | Interictal and preictal period ARMA response from chb01 training data. | 44 |
| Appendix B.2 | Interictal and preictal period ARMA response from chb06 training data. | 44 |
| Appendix B.3 | Interictal and preictal period ARMA response from chb10 training data. | 44 |
| Appendix B.4 | Interictal and preictal period spectral bandpower response from chb01 training data. | 45 |
| Appendix B.5 | Interictal and preictal period spectral bandpower response from chb06 training data. | 45 |
| Appendix B.6 | Interictal and preictal period spectral bandpower response from chb10 training data. | 45 |
| Appendix B.7 | Prediction with ARMA based linear SVM using chb01 training set. | 46 |
| Appendix B.8 | Prediction with ARMA based Logistic Regression using chb01 training set. | 46 |
| Appendix B.9 | Prediction with ARMA based linear SVM using chb06 training set. | 46 |
| Appendix B.10 | Prediction with ARMA based Logistic Regression using chb06 training set. | 47 |
| Appendix B.11 | Prediction with ARMA based linear SVM using chb10 training set. | 47 |
| Appendix B.12 | Prediction with ARMA based Logistic Regression using chb10 training set. | 47 |
| Appendix B.13 | Prediction with spectral power based linear SVM using chb01 training set. | 48 |
| Appendix B.14 | Prediction with spectral power based Logistic Regression using chb01 training set. | 48 |
| Appendix B.15 | Prediction with spectral power based linear SVM using chb06 training set. | 48 |
| Appendix B.16 | Prediction with spectral power based Logistic Regression using chb06 training set. | 49 |
| Appendix B.17 | Prediction with spectral power based linear SVM using chb10 training set. | 49 |
| Appendix B.18 | Prediction with spectral power based Logistic Regression using chb10 training set. | 49 |

List of Tables

| | | |
|--------------|---|----|
| 2.1 | Neural band frequencies | 7 |
| 3.1 | Patient data considered in this work. | 12 |
| 4.1 | Prediction performance on test data for all patients. | 33 |
| Appendix A.1 | Summary statistics of overall dataset. | 40 |
| Appendix A.2 | chb01 Seizure Times | 40 |
| Appendix A.3 | chb03 Seizure Times | 40 |
| Appendix A.4 | chb06 Seizure Times | 40 |
| Appendix A.5 | chb10 Seizure Times | 40 |
| Appendix A.6 | Training and test set: Part A | 41 |
| Appendix A.7 | Training and test set: Part B | 42 |

*To my wonderful mum and dad, and their success, that has
always made me feel I could reach the unreachable...*

Definitions and Abbreviations

| | |
|------|-------------------------------------|
| AR | Autoregression |
| ARMA | Autoregression Moving Average |
| CNN | Convolutional Neural Network |
| CV | Cross Validation |
| EEG | Electroencephalography |
| iEEG | Intracranial Electroencephalography |
| ECoG | Electrocorticography |
| EXP | Exponential Function |
| FFT | Fast Fourier Transform |
| FIR | Finite Impulse Response |
| FPR | False Positive Rate |
| FP | False Positive |
| FN | False Negative |
| KDE | Kernel Density Estimation |
| KF | Kalman Filter |
| MA | Moving Average |
| MSE | Mean Squared Error |
| PSD | Power Spectral Density |
| RBF | Radial Basis Function |
| SGN | Sign Function |
| STFT | Short Time Fourier Transform |
| SVM | Support Vector Machine |
| TPR | True Positive Rate |
| TP | True Positive |
| TN | True Negative |
| VNS | Vagus Nerve Stimulation |

Chapter 1

Introduction

1.1 What are Seizures?

The word seizure is derived from the ancient Greek, meaning *to take hold*. [Fisher et al. \(2005\)](#) defines an epileptic seizure as a transient occurrence of signs and/or symptoms due to abnormal excessive or synchronous neuronal activity in the brain. For a patient to be epileptic, they must have had more than one epileptic seizure. The transient nature of a seizure means they have a clear start and end time. A partial seizure is when synchronous neuronal activity originates from one of the cerebral hemispheres. They can gradually evolve into a second category, known as generalised seizures where neuronal discharge originates from both hemispheres. Both generalised seizures and a subset of partial seizures, known as partial complex seizures can cause loss of consciousness [Bancaud \(1981\)](#).

1.2 Importance of Real-Time Seizure Prediction

Seizures can come about suddenly, creating a situation where the patient cannot rely on themselves to ensure their own safety. Depending upon the context in which it takes place, a generalised seizure could be fatal. The restrictions and uncertainty this brings to everyday life results can foster negative psychological consequences for the patient. Thus, a real-time prediction device can alleviate anxiety and prepare them to reach safety before a seizure can start. [Schachter and Saper \(1998\)](#) has found that Vagus Nerve Stimulation (VNS) can effectively stop seizures in patients with intractable seizures. Thus, a real-time seizure prediction algorithm holds the potential to know when to initiate the stimulation of VNS which can effectively prevent a seizure from taking hold.

1.3 Challenges in Seizure Prediction

EEG data containing seizures have to be clinically validated by a neurologist which is an expensive task. This has resulted in sparse amounts of open source data being available. However, over the last decade, several open source data sets have been released which has formed the basis of seizure prediction in the scientific community and serves as a standard for comparing algorithm performance. Previous works were often done with privately collected data whose reproducibility proved to be a challenge for peer reviewers. Bancaud (1981) describes that epileptic seizures can arise as part of the sleep-wake cycle and factors such as fatigue, alcohol and emotion. This suggests the endeavour of predicting seizures is highly dependent on the patient at hand. As we shall see in the subsequent section, numerous previous work has focused on seizure prediction using intracranial EEG or iEEG, (also known as Electrocorticography or ECoG), which provide a richer signal from a very specific region in the brain. Although scalp EEG provides a greater region of coverage over the brain, it suffers from greater levels of noise in the signal due to being non-invasive.

1.4 Related Work

Chisci et al. (2010) developed a real-time patient-specific seizure prediction algorithm based on using auto-regression (AR) coefficients as features and a Support Vector Machine (SVM) to perform binary classification of preictal and interictal intervals. A state-space tracking algorithm was applied to the prediction signal generated by the SVM, to filter out sporadic seizure warnings. This was shown to produce a smooth transition in the prediction signal, thereby reducing false positive rate (FPR). The algorithm was tested on 9 patients using invasive iEEG data from the Freiburg dataset. The authors claim their system achieved 100% sensitivity overall by training over 2 seizures per patient, but no patient specific sensitivity data was provided. The average FPR over 9 patients was 0.17 when running on interictal data. We note that although the Freiburg dataset was open-source at the time of publishing, it has since been commercialised and is no longer freely available.

Park et al. (2011) use 9 frequency bands as features by extracting the spectral power of EEG signal, before using an SVM for classifying preictal data. They also show that a Kalman filter is the most effective method of minimising false positive alerts when predictions are generated by the SVM. The authors highlight the use of bipolar and time-differential methods (including their combination) to highlight the robustness of their system's performance, achieving 93.8% sensitivity and a FPR of 0.23 overall in the case of using a combination of bipolar and time-differential preprocessing on 18 patients from the Freiburg iEEG dataset.

Howbert et al. (2014) uses 3 canine subjects with partial epileptic seizures to test the hypothesis that iEEG events preceeding a seizure are not random events. They use spectral power from 6 frequency bands as features to a Logistic Regression classifier with a preictal interval duration of 90 minutes. Since their results are presented in the context of a changing 'Time in warning' parameter, we consider the largest value and find the overall FPR to be 2.713 and sensitivity to be 0.86. Note that these results were the mean from all three canine subjects. Although this study shows a significantly higher FPR when compared to the work previously mentioned, we note that since the sensitivity values are substantially larger than a baseline chance predictor, whose expected sensitivity would be 0.5, the results shown in this study does prove that preictal iEEG holds non-random data which in turn has formed into a basis for seizure prediction.

Similar to the works mentioned so far, more recent developments around seizure prediction methods concentrate on offline prediction (i.e. retrospective analysis) such as the work by Truong et al. (2018), which converts time domain signal into frequency domain features using STFT (Short Time Fourier Transform). This creates an image-like output of fixed size for each channel. The development of convolutional neural networks (CNN) have been inspired by the brain's optical system which excel at extracting hidden features from images. Applying this deep learning CNN model, it is then able to classify interictal and preictal states based on the hidden features it learnt.

The work by Tsiouris et al. (2018) uses a different deep learning architecture, called a recurrent neural network, in combination with many different features, including time domain features such as zero crossings, frequency domain features such as the energy percentage of the six neural spectral bands, as well as autocorrelation and graph theory features. The zero crossing is a simple feature that can presumably tell us how stable the input EEG is, whilst the energy percentage of neural spectral bands have a direct clinical interpretation of brain state, as we shall see later. Graph theory can be used to understand channel placement with respect to the scalp area, which can allow more localised features to be extracted, which could help systems based on these features to identify focal seizures better.

1.5 Comments on Related Work

The path to seizure prediction has been a long one as we have just seen. The overarching idea here is to develop a method which can plausibly predict seizures, then establish the significance of the performance results. To this end, we have found the architecture of SVM combined with AR or neural power to be a popular approach for real-time seizure prediction with iEEG. Another difficult area in seizure prediction

is the use of preictal periods, which apparently occur some time leading up to a seizure. Since there is no agreed definition on how long this period stretches back to, any work claiming to be using preictal data has to establish that their choice of preictal duration is the correct one. This consensus might be reached one day, but more studies — such as the work by [Karoly et al. \(2017\)](#), which establishes a link between the circadian profile and the ability to forecast seizures — needs to be done so we have a more concrete definition of a preictal period.

1.6 Dissertation Contribution

We develop a patient-specific algorithm for real-time prediction of seizure onset using scalp EEG. Once trained on a short period of recorded patient data, our algorithm is able to predict by becoming confident that its observations correspond to preictal data. Our algorithm is capable of two modes of pattern extraction from raw EEG: ARMA and spectral power in neural bands. Using these, we test our system on three patients from the openly available CHB-MIT seizure dataset to infer the quality of our system in the context of existing work and what this means for seizure prediction using scalp EEG.

1.7 Dissertation Outline

The remainder of this dissertation is organised as follows: In Chapter 2 we provide background on scalp EEG, the six neural bands and some context we think is required to understand our work more clearly. In Chapter 3 we describe our algorithm's architecture three core modules: feature extraction, classification and regularisation. Chapter 4 describes the results obtained from testing our prediction algorithm on three paediatric patients from the CHB-MIT dataset. In Chapter 5 we provide a discussion of our results and what they could mean, performing comparison with some of the previous work. Finally, Chapter 6 ends this dissertation with a conclusion and includes comments about future research directions.

Chapter 2

Background

2.1 Scalp EEG

EEG is a medical device which measures the relative potential difference between two electrodes on placed on the scalp. Its unit of measurement is the micro-volt or μV . When bundles of neurons fire synchronously, electrical fields in the mesoscopic to microscopic scales are generated, which are picked up by electrodes placed over various regions of the scalp. The mesoscopic scale is large enough to cover patches of cortex, while the macroscopic scale can cover entire regions of the cortex. Activity at the microscopic level, which may include firing of individual neurons, are weak such that they cannot be captured by EEG. This is due to the nature of electrical fields, whose strength decreases exponentially over distance. Moreover, neurologists advise not to use raw voltage for comparison between patients due to physiological diversity in patients, such as scalp thickness and procedural variations of electrode placement [Cohen \(2019\)](#).

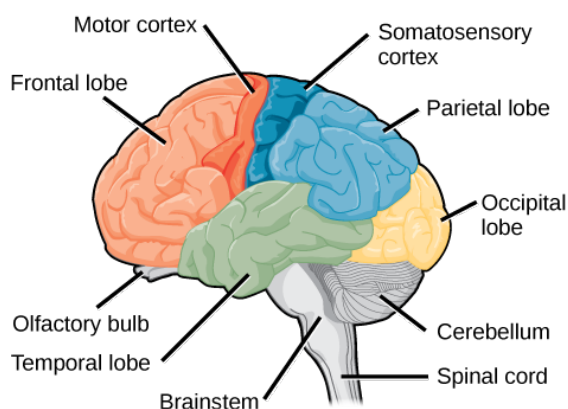


FIGURE 2.1: The frontal, parietal, occipital and temporal lobes in the brain.¹

¹Source: CNX OpenStax, Creative Commons Attribution 4.0 International licence.

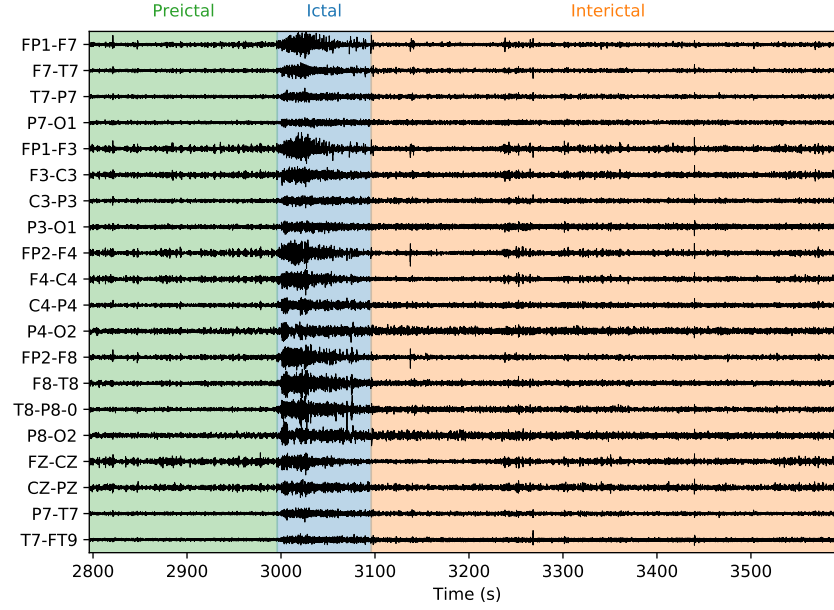


FIGURE 2.2: Visualising the preictal, ictal and interictal periods in an EEG recording.

2.2 Seizures Within EEG

Since seizures are transient events, they have a particular taxonomy which have come to be commonly accepted as debated by [Fisher et al. \(2005\)](#). The interictal period is defined to be between seizures, therefore, we understand this to be the period in which the patient is feeling normal. Ictal is the period in which the patient is having a seizure. Preictal, then, is some period of time immediately preceding a seizure. In seizure prediction literature, we have noted each work defines their own preictal duration, but all occur before the ictal period, indicating a lack of consensus on the precise duration of preictal period. It is important to note that annotation for an ictal period can only be performed by an authorised clinician. Figure 2.2 shows the EEG of a patient showing seizure start at 2996s and ending at 3096s. The seizure period is known as ictal. We use 200s preceeding the seizure as an example of a preictal period. Once the seizure ends, the patient is said to be in the interictal period.

2.3 Neural Bands

These are rhythmic activities emitted by the neural tissues, which themselves are formed of neurons. Activity at the mesoscopic scale gives rise to electrical fields that become visible to EEG. There are five neural frequency bands: $\delta, \theta, \alpha, \beta, \gamma$. In this work, we subdivide the γ band into γ (i.e. low-gamma) and Γ (i.e. high gamma). Table 2.1 shows the frequencies corresponding to each band. Note that since the EEG signal we use has been sampled at $f_s = 256$ Hz, we are only able to see up to the

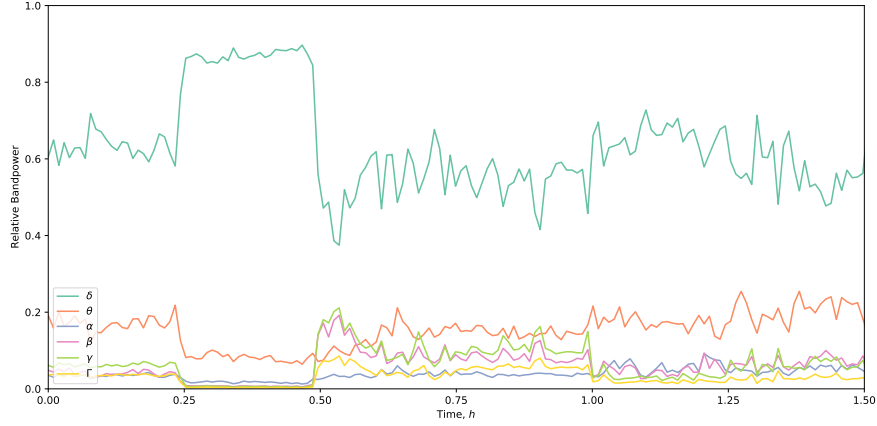


FIGURE 2.3: Neural band power response for EEG sample.

TABLE 2.1: Neural band frequencies

| Band | Frequency (Hz) |
|----------|----------------|
| δ | 0.1 - 4 |
| θ | 4 - 8 |
| α | 8 - 12 |
| β | 12 - 30 |
| γ | 30 - 70 |
| Γ | 70 - 127.9 |

Nyquist frequency $f_s/2 = 128$. Figure 2.3 shows the response of the six bands over some sample EEG data.

2.4 EEG Electrodes

The arrangement of electrodes on the scalp is carried out with respect to an international standard called the "10-20 system", shown in figure 2.4. It ensures that inter-electrode spacing is equal and that electrode placements are proportional to the patient's skull size and shape. The name of an electrode is codified in such a manner that its location can be readily identified. The letters correspond to a brain region; F: Frontal lobe, P: Parietal lobe, T: Temporal lobe, O: Occipital lobe. The numbers following the letters correspond to the hemisphere; Odd number: left hemisphere, even number: right hemisphere. This is with the exception of 'z', which refers to the midline extending from the nasion to the inion. The data used in this study contains 23 channels using 10-20 system.

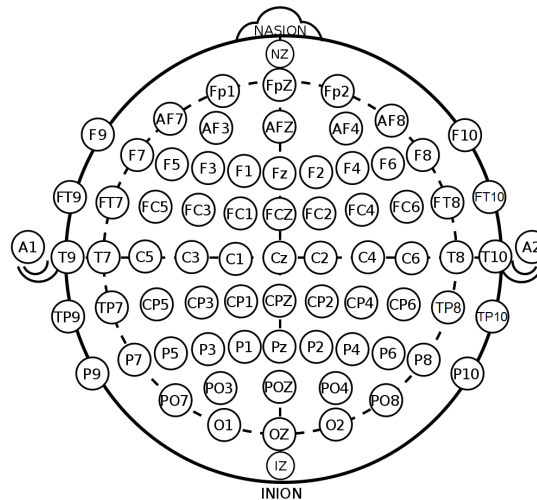


FIGURE 2.4: The international 10-20 system defining the location of scalp electrode placement position.

2.5 Online Prediction

This is a machine learning paradigm where data arrives sequentially, allowing for the opportunity to learn new concepts over time. Online learners, such as the KF, hold a limited memory into the past and use it to make predictions for the future. Since our models are trained on a limited batch of data, our learning algorithms are not online, however, since predictions are generated in real-time, we call our algorithm an online prediction algorithm as it indeed does not have access to the entire EEG signal, but rather receives it through a much smaller buffer window.

2.6 Linear Least Squares

Least square solution is found by minimising the mean squared error (MSE). The final line of (2.1) assumes $X^T X$ is positive definite, meaning X needs to have full column rank. This finds the optimum weights, w , resulting in a line of regression which best fits the data. Figure 2.5 shows the least squares fit of noisy synthetic data.

$$\begin{aligned} E(\mathbf{w}) &= \frac{1}{2} \sum_{i=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 \\ E(\mathbf{w}) &= (\mathbf{t} - \mathbf{X}\mathbf{w})^T (\mathbf{t} - \mathbf{X}\mathbf{w}) \\ E(\mathbf{w}) &= \frac{\delta E(\mathbf{w})}{\delta \mathbf{w}} = \mathbf{w} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{t} \end{aligned} \tag{2.1}$$

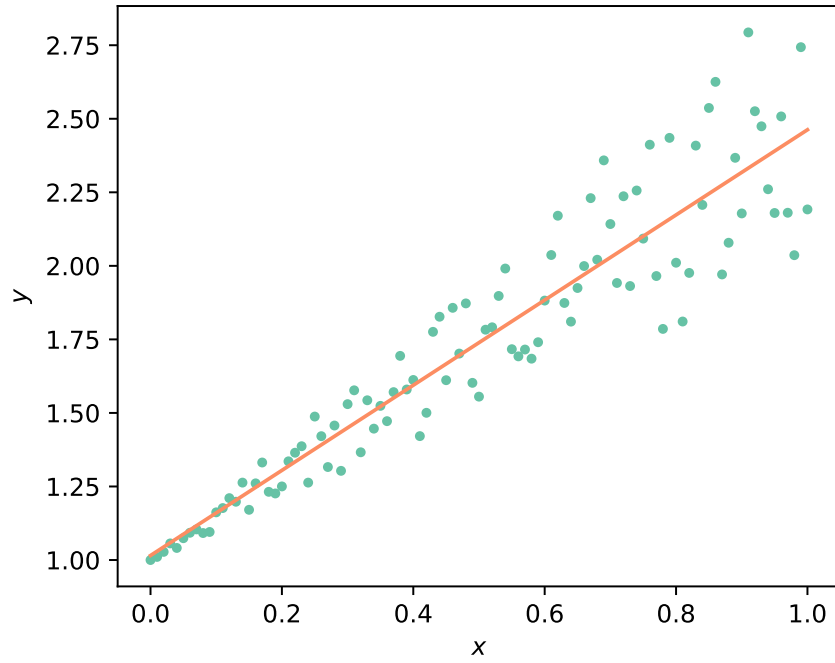


FIGURE 2.5: Linear least squares model fitted on 2-dimensional regressors.

2.7 AR Modelling

Autoregression is a generative model as a weighted sum into the past induces a correlation.

$$s_n = \sum_{i=1}^p a_i s_{n-i} + v_n \quad (2.2)$$

is the p -th order AR model at time n and its model parameters are a_i where $i \in 1, \dots, p$. v_n is a white noise term which adds the assumption that the incoming signal is perturbed with white noise. Figure 2.6 shows updates to the two second order AR(2) model parameters.

2.8 Evaluation Metrics

In binary classification, the metrics that we are interested in are sensitivity and specificity. Sensitivity is computed from the number of true positives, whilst specificity is computed by the number of true negatives. Figure 2.7 shows a confusion matrix, which is a useful tool for understanding the differences between true negatives and false positives. In our study, we use the false positive rate (FPR):

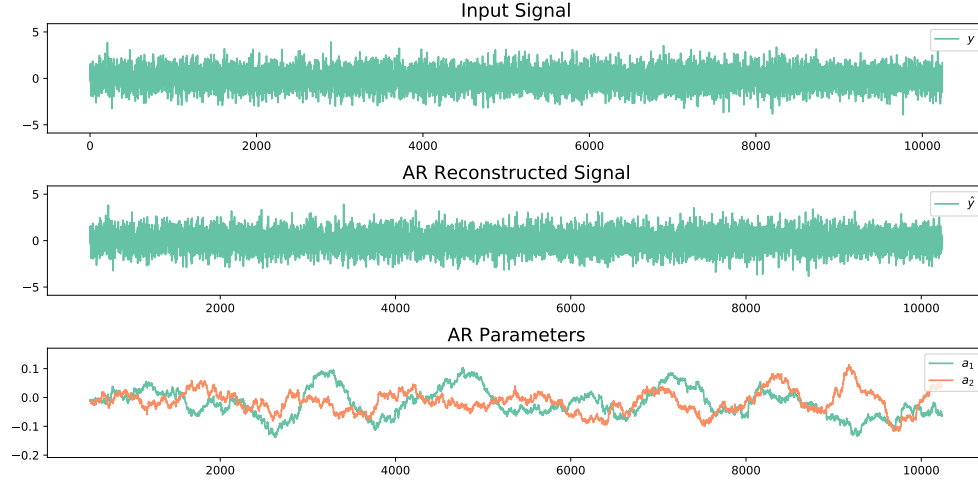


FIGURE 2.6: Online estimation of AR parameters using synthetic observation signal.

$$FPR = \frac{FP}{FP + TN} \quad (2.3)$$

and true positive rate (TPR):

$$TPR = \frac{TP}{TP + FN} \quad (2.4)$$

to evaluate the performance of our algorithm. These metrics are chosen so that our work is comparable to previous related works.

| | | | |
|-----------------|-------------------|-------------------|-------------------|
| Ground Truth | Negative Class | True Negative | False Positive |
| | Positive Class | False Negative | True Positive |
| | | Negative Class | Positive Class |
| | | Predicted | |

FIGURE 2.7: Confusion matrix for binary classification.

Chapter 3

Methodology

3.1 Research Problem

We tackle the following research question: ‘Can seizures be predicted with scalp EEG data in real-time for a patient?’. For this question to be answered, firstly, patient specific EEG data with medically verified annotations are needed. Secondly, we require a basis for seizure prediction: what should we look for in an EEG recording to conclude a seizure is forthcoming? Third, assuming a certain type of pattern exists in the EEG data before a patient is about to have a seizure, can we develop a certain type of algorithm that has the capability to recognise these patterns in the EEG so that it can discriminate between normal and these so-called ‘pre-seizure’ periods? How do we build this algorithm? Lastly, how can we test this algorithm to determine the quality of its predictions on a patient?

3.2 Data

A widely used scalp EEG dataset in the seizure prediction scientific community is the CHB-MIT dataset. It was introduced by [Shoeb \(2009\)](#) during his PhD thesis and is hosted on Physionet by [Goldberger et al. \(2000\)](#) at the time of writing. The dataset contains EEG recording of 23 cases from 22 pediatric subjects with intractable seizures while they were being assessed for surgery. The 23rd case belongs to the same subject as case 1, but 1.5 years later. Additional criteria described in section 3.2.1 are applied before selecting data, a summary of which is provided in table 3.1. All seizure periods in the EEG recording are annotated by clinicians.

TABLE 3.1: Patient data considered in this work.

| Patient ID | Channels | Seizures |
|------------|----------|----------|
| chb01 | 23 | 7 |
| chb03 | 23 | 7 |
| chb06 | 23 | 10 |
| chb10 | 23 | 7 |

3.2.1 Selection Criteria

Appendix A contains table A.1 shows some descriptive statistics for the overall dataset which should be viewed in the context of figures A.1 and A.2. We use two criteria to select a subset of this data to ensure robustness of our results. The first is to eliminate cases whose channel count is not equal to the mode channel count. This eliminates variation in results as more channel implies richer signal and less channel means more attenuated signal from the patient. The second criterion is based around the observation that seizures are relatively rare events. Therefore, we wish to select cases with the highest number of seizure count, which ensures a more robust training and test set for our binary classifier. The dataset provides EEG recording of patients in blocks of fixed time. Since it cannot be guaranteed that there are no gaps between any two blocks, we will only consider preictal data in the same block as the seizure. The tables A.2 to A.5 in appendix A show file blocks of fixed time duration and the starting point of a seizure. This information reveals that for chb01 and chb03, files are supplied in 1 hour blocks, whereas, chb06 comes in 4 hour blocks and chb10 comes in 2 hour blocks. For the purposes of evaluating the generalised performance of our model, we will separate the data into training, test sets. Since class imbalance can affect model performance adversely, we ensure that equal amounts of data are chosen from both classes during training. To reflect a more realistic scenario, we produce an imbalanced dataset for the test set.

3.2.2 Preictal State

From the literature review conducted, we found that neurologists have established a pre-seizure state in EEG recording called the ‘preictal’ state, described in more detail in section 2.2. This serves as the basis for our prediction, as the rationale is if we can successfully predict a preictal state, a seizure (or ‘ictal’ state as known in the literature) will inevitably follow. The purpose of our algorithm, is then to differentiate between a ‘normal’ state (known as ‘interictal’), and a ‘pre-seizure’ state (as ‘preictal’). As we have previously mentioned, a key problem faced with recognising the preictal state in EEG recording is that its duration is not strictly defined. We have found studies using intracranial EEG data using preictal data, whose length is apparently arbitrarily

defined to be ranging between 15 minutes to 90 minutes. In our study, we combine preictal blocks of 15 minutes to show our algorithm can discriminate this period from the interictal one.

3.3 Detecting Pre-Seizure States

Our research problem requires us to develop an algorithm based on modest resources that will serve as baseline for future studies. Thus, we aim to strike a level of complexity that we believe to be sufficient for this purpose. We also believe in clinical interpretability, so any decisions made by the algorithm should be interrogable. Using an overly complex algorithm with an amalgamation of features is both resource constraining and makes clinical interpretation difficult.

Thus, our work is based around two feature extraction methods: autocorrelation and power spectral density, expressed as energy percentage for each of the six neural bands. In subsequent sections of this work, the first approach will be called ‘ARMA’ and the second approach, ‘Spectral Power’.

3.4 Architecture

The algorithm we have developed has two main modules: feature extractor and classifier. The purpose of the feature extractor is to receive signals as they are captured from the brain and cast them to a lower dimensional space. This is because the raw EEG data exists in a high dimensional space, as there are 23 channels, each capturing voltage values over time. This is represented to the left of figure 3.1. Modelling the raw brain signal with ARMA or spectral power in band is a dimensionality reduction step, that aids our algorithm in processing the data in real-time without causing excessive latency. The feature extraction module outputs down-sampled feature vectors as we expect parameter values to change at a slower rate than raw voltage values. In general, the mapping on input signal x at time t is given by $x_t \mapsto a_k$:

$$x_t = \begin{bmatrix} x_t^1 \\ \vdots \\ x_t^{n_c} \end{bmatrix} \mapsto a_k = \begin{bmatrix} a_k^1 & \cdots & a_k^p \end{bmatrix} \quad (3.1)$$

where n_c is the total number of channels and a_k is a p -dimensional feature vector at prediction instant k . The stream buffer size is computed by $I_k = (k - 1)N + 1$ where $N \gg 1$, $N \in \mathbb{Z}^+$ chosen appropriately so that the signal in the stream buffer of duration N/f_s can be assumed to be stationary. Our prediction frequency f_p is then

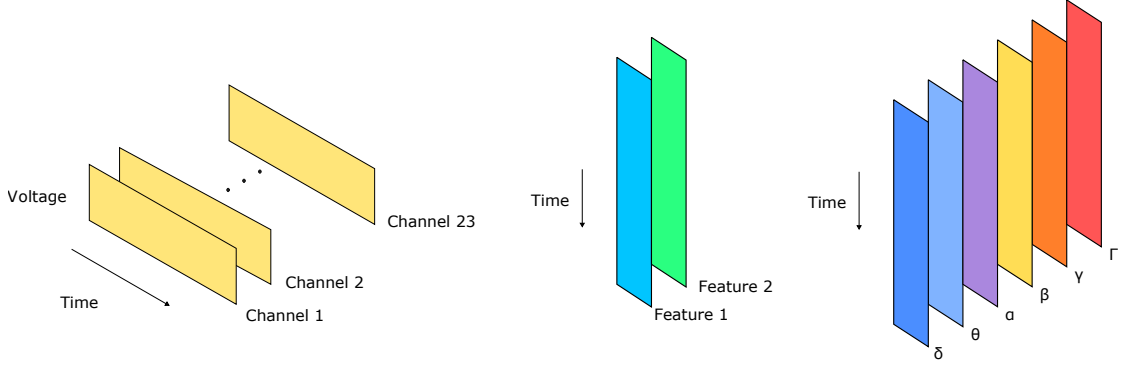


FIGURE 3.1: Dimensionality reduction of neural data through ARMA and spectral features.

given by f_s/N where f_s is the sampling frequency of the EEG signal. For our experiments, we chose $N = 512$ and the sampling frequency of all data is $f_s = 256$ Hz.

With the assumption different patterns exist for both interictal and preictal classes, we perform labelling by pairing each element u_i from the feature vector \mathbf{a}_k with a label l_i , $l_i \in \{-1, 1\}$, $i \triangleq \{1, \dots, P\}$ where P is the number of rows in the training set, $l_i = -1$ is the interictal class and $l_i = 1$ is the preictal class. These labels allow our learning algorithm during training to learn feature patterns associated for a given class.

During testing, our learning algorithm will observe feature vector \mathbf{a}_k and predict a decision y_k using the sign function $\text{sgn}(u_i)$ such that $\text{sgn} : \mathbb{R} \mapsto \{-1, 1\}$.

3.5 Feature Extraction

3.5.1 ARMA

We use the ARMA model as a feature extraction tool by observing parameter changes over time. We use a second order AR model, which has two parameters changing over time as shown in figure 3.1. Or more formally, $AR(2) : \mathbf{x}_t \mapsto \mathbf{a}_k \in \mathbb{R}^2$. Our method for parameter estimation is as follows: let $\boldsymbol{\varphi}_t^i \triangleq [x_{t-1}^i, x_{t-2}^i]^T$ be the two-step time history vector of the input signal (two step as we use a second order model). Then, the AR model is formulated by $x_t^i = \boldsymbol{\varphi}_t^{iT} \mathbf{a}^i + e_t^i$ where $\mathbf{a}^i \triangleq [a_1^i, a_2^i]$ is the model parameter vector and e_t^i is the assumption of noise in the signal. Fitting the input $\boldsymbol{\varphi}_t^i$ with linear least-squares regression which can be estimated using Moore-Penrose pseudo-inverse (2.1):

$$\begin{aligned}
\hat{a}_k^i &= \arg \min_{a^i} \sum_{t \in I_k} \left[x_t^i - (\boldsymbol{\varphi}_t^i)^T a^i \right]^2 \\
&= \left[\sum_{t \in I_k} \boldsymbol{\varphi}_t^i (\boldsymbol{\varphi}_t^i)^T \right]^{-1} \left[\sum_{t \in I_k} \boldsymbol{\varphi}_t^i x_t^i \right]
\end{aligned} \tag{3.2}$$

MA is then performed on \hat{a}_k^i

$$a_k = \sum_{j=0}^m c_j \hat{a}_{k-j} \tag{3.3}$$

where m is a smoothing parameter and c_j are the coefficient parameters such that $\sum_{j=0}^m c_j = 1$.

3.5.2 Spectral Power

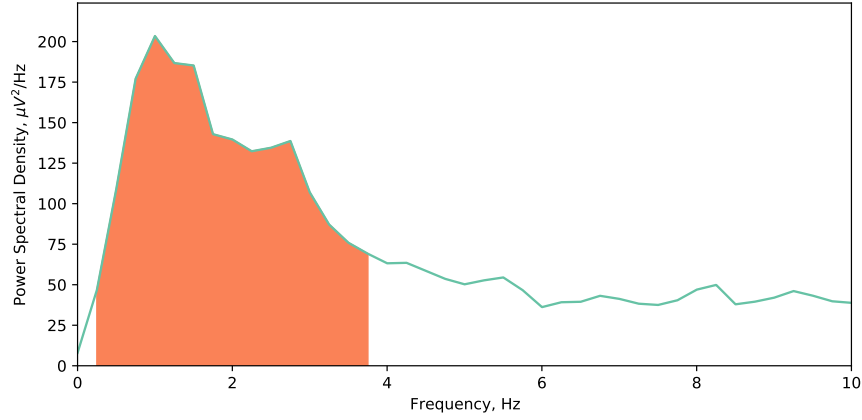
To extract the relative energy produced by each of the six neural bands, the power spectral density of the signal in the stream buffer estimated by first applying STFT (Short Time Fourier Transform) with a Hann window of 2 second length and 50% overlap. These power spectrum, or periodograms, are then time averaged to produce a smoother power spectrum. This is a method of estimating the power spectrum developed by [Welch \(1967\)](#). From the power spectrum, we estimate the bandpower by first performing FIR (Finite-Impulse Response) bandpass filtering, then using Simpson Rule to find the numeric integral approximation. The mathematical details of performing STFT and Welch's Periodogram remain beyond the scope of this work, as we ask the interested reader to refer to the reference for further reading.

Figure 3.2 shows a magnified view of the δ band between 0.1 - 4 Hz highlighted on a power spectrum computed using Welch's method. Once the power for all six bands have been found, we divide each of them by the total power (total area under power spectrum) to compute relative power values in the interval $[0,1]$ for each band. These form a 6-dimensional feature vector for each prediction time k , as visualised in the right side of figure 3.1.

3.6 Classification

3.6.1 Support Vector Machine

For a $n \times p$ data matrix X with n training observations in p -dimensional space and that these are labelled by $y_1, \dots, y_n \in \{-1, 1\}$, a separating hyperplane function $f(x^*)$

FIGURE 3.2: δ band on a power spectrum.

can then classify an unseen test observation $x^* = [x_1^*, \dots, x_p^*]^T$ using the functions:

$$\begin{aligned} f(x^*) &= \beta_0 + \beta_1 x_1^* + \beta_2 x_2^* + \dots + \beta_p x_p^* = 1 \text{ if } f(x^*) > 0 \\ f(x^*) &= \beta_0 + \beta_1 x_1^* + \beta_2 x_2^* + \dots + \beta_p x_p^* = -1 \text{ if } f(x^*) < 0 \end{aligned} \quad (3.4)$$

A hyperplane is a flat affine subspace whose dimension is $p - 1$ for a p -dimensional space. It is given by:

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p = 0 \quad (3.5)$$

where β_0, \dots, β_p are parameters and X_1, \dots, X_p is a point in the p -dimensional space. The idea of a separating hyperplane classifier is extended in maximal margin classifiers which use support vectors to set the position of a hyperplane such that the two classes achieve maximum separation. However, this kind of classification can only work with a linear decision boundary. It is possible to map the feature space to a higher dimension by performing a transformation on the input data such as $x \mapsto x^2$, but this increasing dimensionality is counterproductive to the dimensionality reduction performed in the feature extraction step as it increases computational complexity severely. Using kernels instead of arbitrary polynomial mapping is the method of achieving an efficient way to classify points using a non-linear boundary. The linear SVM is represented by:

$$f(x) = \beta_0 + \sum_{i=1}^n \alpha_i \langle x, x_i \rangle \quad (3.6)$$

where n is the number of training datapoints, α_i are parameters to be estimated and $\langle x, x_i \rangle$ are the inner products of training observations. This inner product can then be

generalised to the form $K(x_i, x'_i)$ which captures the similarity between any two observations. Therefore, the SVM in (3.6) can be expressed more simply as:

$$f(x) = \beta_0 + \sum_{i \in \mathcal{S}} \alpha_i K(x, x_i) \quad (3.7)$$

where the linear kernel produces linear decision boundary:

$$K(x_i, x_{i'}) = \sum_{j=1}^p x_{ij} x_{i'j} \quad (3.8)$$

and the radial basis kernel, which produces radial decision boundaries:

$$K(x_i, x_{i'}) = \exp \left(-\gamma \sum_{j=1}^p (x_{ij} - x_{i'j})^2 \right) \quad (3.9)$$

with the constant $\gamma > 0$. The notation in this section has been adopted from [James et al. \(2013\)](#).

3.7 Regularisation

As there will be noise from the prediction signal, we consider regularisation which acts to smooth out and reduce false positive and false negative classifications. We compare the smoothing performance of MA with Kalman filter as previous works, such as [Chisci et al. \(2010\)](#), who has claimed they are able to perform prediction regularisation effectively.

3.7.1 MA

The moving average (MA) is computed over the last k datapoints p_1, \dots, p_n recursively:

$$MA_{k,k+1} = MA_{k,k-1} + \frac{1}{k} (p_{n+1} - p_{n-k+1}) \quad (3.10)$$

3.7.2 Kalman Filter

The Kalman filter is an online learning algorithm which sequentially receives observations based on some underlying state changes and uses it to estimate a joint

probability distribution $P(\mathbf{x}_n|\mathbf{z}_n)$ where \mathbf{x}_n is the state and \mathbf{z}_n are the observations seen up to time n . Under the Bayesian framework, we can describe the joint posterior as follows:

$$P(\mathbf{x}_n|\mathbf{z}_n) = \frac{P(\mathbf{z}_n|\mathbf{x}_n)P(\mathbf{x}_n|\mathbf{z}_{n-1})}{P(\mathbf{z}_n|\mathbf{z}_{n-1})} \quad (3.11)$$

which is the distribution of states \mathbf{x}_n at time n conditioned on all past observations consisting of $\mathbf{z}_n = \{\mathbf{z}_1, \dots, \mathbf{z}_n\}$. This is formulated under Bayes theorem where $P(\mathbf{z}_n|\mathbf{x}_n)$ is the likelihood and $P(\mathbf{x}_n|\mathbf{z}_{n-1})$ is the prior, normalised using $P(\mathbf{z}_n|\mathbf{z}_{n-1})$. The filter is a generative process as it makes stochastic transitions and generates stochastic observations as time passes, sampling transitions from density $\mathcal{N}(\mathbf{F}\mathbf{x}_{n-1}, \mathbf{Q})$ where \mathbf{F} is the state transition matrix and \mathbf{Q} is the process noise. Observations are sampled from density $\mathcal{N}(\mathbf{H}\mathbf{x}_n, \mathbf{R})$ where \mathbf{H} is the observation matrix which defines the relationship between system's internal state and external instrument measurements, and \mathbf{R} is the measurement noise.

At time $n - 1$, the state given observation is

$$P(\mathbf{x}_{n-1}|\mathbf{z}_{n-1}) = \mathcal{N}(\mathbf{x}_{n-1}|\hat{\mathbf{x}}_{n-1|n-1}, \mathbf{P}_{n-1|n-1}) \quad (3.12)$$

where \mathbf{x}_{n-1} is the probability distribution with mean $\hat{\mathbf{x}}_{n-1|n-1}$ and covariance $\mathbf{P}_{n-1|n-1}$ which models the uncertainty due to noise. Next, a predict step is taken when moving from time $n - 1$ to n :

$$\begin{aligned} P(\mathbf{x}_n|\mathbf{z}_{n-1}) &= \mathcal{N}(\mathbf{x}_n|\hat{\mathbf{x}}_{n|n-1}, \mathbf{P}_{n|n-1}) \\ &= \mathcal{N}(\mathbf{F}\hat{\mathbf{x}}_{n-1|n-1}, \mathbf{F}\mathbf{P}_{n-1|n-1}) \end{aligned} \quad (3.13)$$

At time n , a joint probability distribution is estimated based on:

$$P(\mathbf{x}_n|\mathbf{z}_n) = \mathcal{N}(\mathbf{x}_n|\hat{\mathbf{x}}_{n|n}, \mathbf{P}_{n|n}) \quad (3.14)$$

where the understanding is updated based on the arrival of new data:

$$\begin{aligned} \hat{\mathbf{x}}_{n|n} &= \hat{\mathbf{x}}_{n|n-1} + \mathbf{P}_{n|n-1}\mathbf{H}^T(\mathbf{H}\mathbf{P}_{n|n-1}\mathbf{H}^T + \mathbf{R})^{-1}(\mathbf{z}_n - \mathbf{H}\hat{\mathbf{x}}_{n|n-1}) \\ &= \hat{\mathbf{x}}_{n|n-1} + \mathbf{k}_n(\mathbf{z}_n - \mathbf{H}\hat{\mathbf{x}}_{n|n-1}) \end{aligned} \quad (3.15)$$

Equation (3.15) describes that the estimated state at time n given we have seen states up to time n is the result of our state estimate at time n given we have seen states up to time $n - 1$ and a Kalman gain \mathbf{k}_n which is also used with observations when updating uncertainty:

$$\begin{aligned}\mathbf{P}_{n|n} &= \mathbf{P}_{n|n-1} - \mathbf{P}_{n|n-1}\mathbf{H}^T(\mathbf{H}\mathbf{P}_{n|n-1}\mathbf{H}^T + \mathbf{R})^{-1}\mathbf{H}\mathbf{P}_{n|n-1} \\ &= (\mathbf{I} - \mathbf{k}_n\mathbf{H})\mathbf{P}_{n|n-1}\end{aligned}\tag{3.16}$$

k_n takes a value in range $[0, 1]$ and is found by minimising the expected error in posterior state estimate $\mathbb{E}[||\mathbf{x} - \mathbf{x}_{n|n}||^2]$ in the form of:

$$\nabla_{k_n} \text{Tr}(\mathbf{P}_{n|n})\tag{3.17}$$

minimising the trace of innovation covariance $\mathbf{P}_{n|n}$ with respect to $k(n)$.

3.8 Training

As seizures are relatively rare events, there is less data available for the preictal class. When trained on an imbalanced dataset, a model tends to predict the majority class most of the time on average. To avoid this bias during training, we select equal amounts of data from both interictal and preictal classes. Three 15 minute chunks of random interictal data is sampled from each patient and combined with three more chunks of 15 minute preictal data, totalling 1.5 hours of raw EEG data for each patient that forms the training set.

3.9 Testing

In the real world, our algorithm will not be classifying equal amounts of interictal and preictal data over a fixed time period. Therefore, we test our algorithm on an imbalanced dataset where the ratio of interictal to preictal data is 4:1. Although this is not a completely realistic scenario, it is a one plausible option without having to test our algorithm on live patient data as part of a clinical trial. See table A.6 in appendix A for specific details on which files were included in both training and test sets.

Chapter 4

Results

4.1 Feature Extraction

Ideally, features extracted from the raw signal should give us maximum separation of classes so our learning algorithm can effectively predict the class of an unseen datapoint. The purpose of feature extraction is to reduce the dimensionality of the data. Since each sample in the raw data is represented by μV measured by 23 separate electrodes, the input vector will have 23 dimensions. However, the curse of dimensionality means that sparsity of available data increases due to rapid growth in volume of space. In this study, we consider AR coefficients and relative band power as two possible methods of feature selection, where the AR approach results in 2 features arising from the use of AR(2) model and relative band power results in 6 features derived from neural oscillatory bands.

4.1.1 ARMA

When modelling the signal with AR, we have chosen to use a second-order model for two reasons: 1) relatively faster real-time prediction performance; 2) discriminatory ability of a low-dimensional feature space. ARMA parameter response for each patient is provided in appendix A such as figure B.1. We use a bivariate kernel density estimation (KDE) plot to visualise ARMA samples. The marginal density curves show distribution of both classes according to features. The marginal univariate plots can be interpreted in a similar manner to a histogram, while the bivariate plots highlight clusters of data.

CHB-01

This patient has the clearest separation of data from two classes. Figure 4.1 shows training data having two clusters, of which, the preictal cluster centred at $(-0.55, 1.50)$, has a more well-defined boundary than interictal cluster, centred at $(-0.30, 1.25)$. We also find a greater variance in feature 1 data from interictal period and note that the marginal density curves for both classes and features peak at a similar height.

CHB-06

Our most notable observation from figure 4.2 is that there are two interictal clusters of data, where the first cluster, centred at $(-0.8, 1.75)$, has a greater concentration of data than the second as the peak of the marginal density for this cluster is higher than the second cluster, centred at $(-0.1, 0.5)$. Also, the preictal cluster, centred at $(-0.6, 1.5)$, has greater overlap with interictal data when compared to patient 'chb01'. The marginal density curves for interictal clusters show the second cluster has a 'flattening' effect on the first cluster.

CHB-10

Figure 4.3 shows both clusters from the training data for this patient show near complete overlap. Only the edge of the bivariate distribution for interictal data, near $(-0.8, 1.75)$ appear to be outside of any overlap due to the greater variance in this class, making this distribution the most difficult to discriminate without labels.

4.1.2 Spectral Power

The signal is modelled in terms of the relative band power of six neural bands $\delta, \theta, \alpha, \beta, \gamma, \Gamma$. Note the use of Greek lowercase γ to denote low-Gamma and Greek uppercase Γ to denote high-Gamma. Features are shown using a pair-plot which shows the pairwise relationship of data between spectral bands. The diagonal plots show the marginal distribution of bands in each column. Also note that the pair-plots can be decomposed into an upper and lower triangle (above and below the diagonals with marginal distribution plots). The upper triangle scatter plots are mirrors of lower triangle scatter plots and vice-versa. Since not all bands are expected to share the same discriminatory capability, so we note the bands which appear to provide better discrimination between interictal and preictal classes.

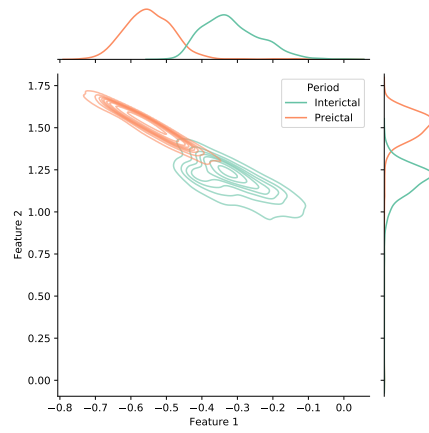


FIGURE 4.1: Class distribution of ARMA features from chb01 training data.

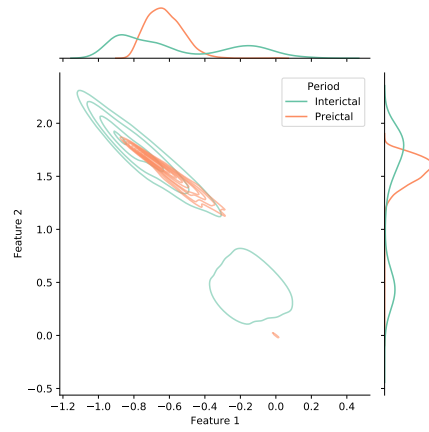


FIGURE 4.2: Class distribution of ARMA features from chb06 training data.

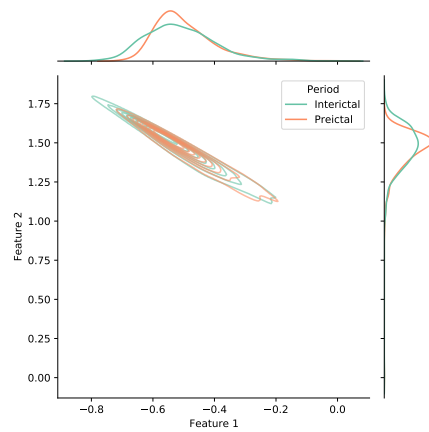


FIGURE 4.3: Class distribution of ARMA features from chb10 training data.

CHB-01

In figure 4.4 we find from the marginal distribution of β and δ features that they have the most overlap between the two classes. γ and Γ on the other hand provides the best separation. We note in the α band there are less preictal datapoints, whilst in the γ and Γ bands, there are less interictal datapoints.

CHB-06

Most interictal data in figure 4.5 is concentrated in the δ, θ and β bands. For every band, preictal data is found to be in a single cluster, where the cluster in Γ has the least variance. Since the marginal distribution of preictal data in the δ band has the highest peak, we say it has the greatest concentration of datapoints.

CHB-10

Figure 4.6 shows δ band contains the majority of the datapoints from both classes, although there are more preictal, since the peak of the marginal distribution is higher. Moreover, as the tail of this marginal distribution is non-symmetric, the boundary of the preictal data cluster is also non-symmetric. Compare this to the marginal distribution of γ , where we see a more symmetric cluster, but this pattern is no better than the δ band as there is a considerable overlap of this cluster with the interictal cluster and both cluster have significantly lower number of datapoints in the γ band which lowers the our confidence of the pattern being seen.

4.2 Model Comparison

Three machine learning models are compared using two feature extraction methods through real-time prediction on training data. These models are: Support Vector Machine with linear kernel (Linear SVM), Support Vector Machine with radial basis kernel (RBF SVM) and Logistic Regression.

4.2.1 ARMA

The real time predictions were performed using balanced classes with distribution discussed previously in section 4.1.1. Plots of real time prediction signal are provided in appendix B.

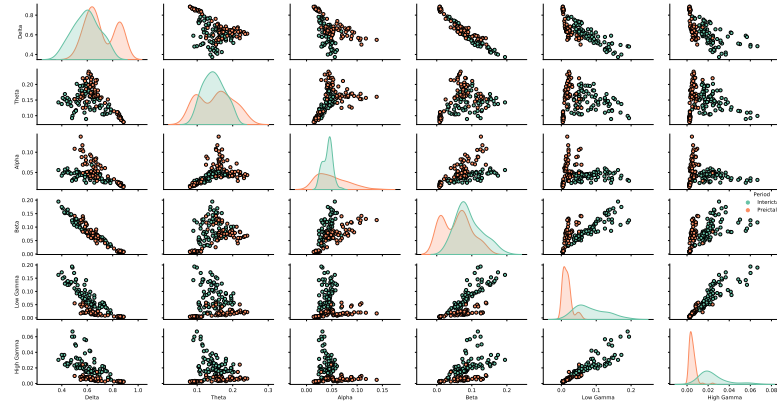


FIGURE 4.4: Class distribution of spectral features from chb01 training data.

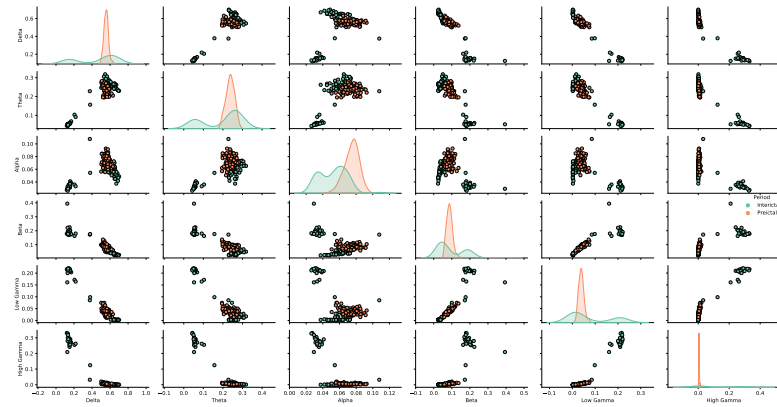


FIGURE 4.5: Class distribution of spectral features from chb06 training data.

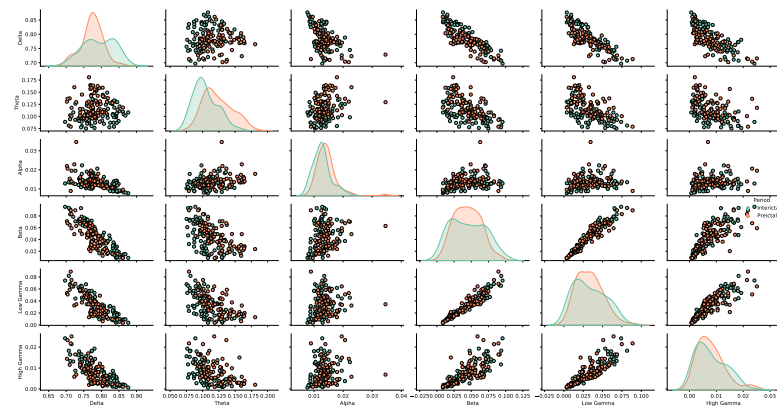


FIGURE 4.6: Class distribution of spectral features from chb10 training data.

CHB-01

All three models generated near identical prediction signals as shown in figures: B.7, 4.7, B.8. Although the raw signals are noisy and contain many false positives in the interictal period as they reach class value of $y_k = 1$, the MA filtered version of the prediction signal never reach $y_k = 1$ and holds the true negative value of $y_k = -1$ for most of the interictal period, or a value relatively close to it. As the preictal period starts at time $t = 0.75$, we find the MA prediction signal values move towards the true positive value of $y_k = 1$ (although it does generate a false negative signal very briefly between time $t = 0.75$ and $t = 1.00$). From $t = 1.125$, we observe the models generating a continuous signal value of $y_k = 1$.

CHB-06

Figure B.9 shows the Linear SVM emitting very noisy prediction signal from $t = 0$ to $t = 0.25$, after which it stabilises into classifying the remaining duration of the recording as preictal. The output generated from $t = 0.25$ to $t = 0.75$ are categorised as false positives while the output generated from $t = 0.75$ to $t = 1.50$ are true positives. The RBF SVM in figure 4.8, produces true negative outputs from $t = 0$ to $t = 0.25$. However from $t = 0.25$, in contrast to the Linear SVM, the RBF SVM begins to output noisy predictions. Although this noisy behaviour continues for the remaining duration of the recording, it does manage to capture more true negatives between $t = 0.50$ and $t = 0.75$. In the preictal period, although the raw prediction signal generates false negatives, MA filtering is successfully able to cancel these out within some tolerance threshold. Logistic Regression, in figure B.10, exhibits a similar behaviour to RBF SVM from $t = 0$ to $t = 0.25$. Similar to the previous two classifiers, it also is not able to combat the false positives for the next 15 minutes ($t = 0.25$ to $t = 0.50$). Between $t = 0.50$ and $t = 0.75$, the MA filtering struggles, more so, than RBF SVM to stay close to $y_k = -1$, however, with a greater tolerance threshold, the MA signal is still able to correctly classify this block of time. During the preictal block, with MA filtering, the whole period is correctly classified in a similar way to the RBF SVM.

CHB-10

Figures B.11, 4.9 and B.12 show prediction signal following a random walk behaviour from all three models as the MA signal does not show any consistency in classifying either the interictal or preictal blocks.

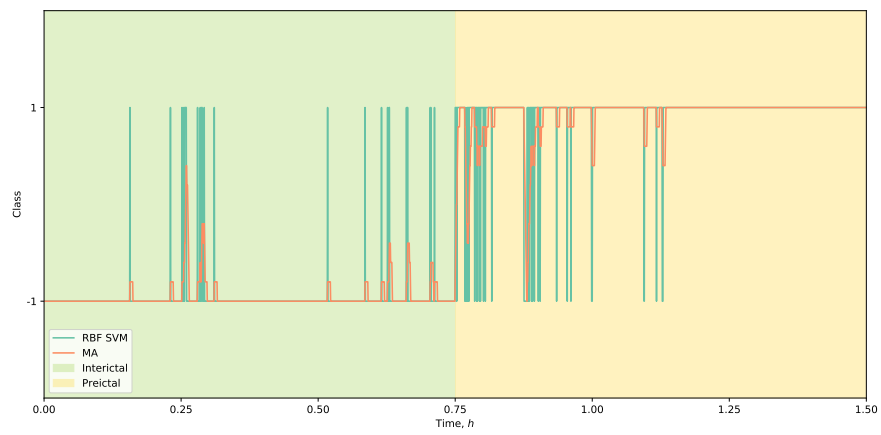


FIGURE 4.7: Prediction with ARMA based RBF SVM using chb01 training set.

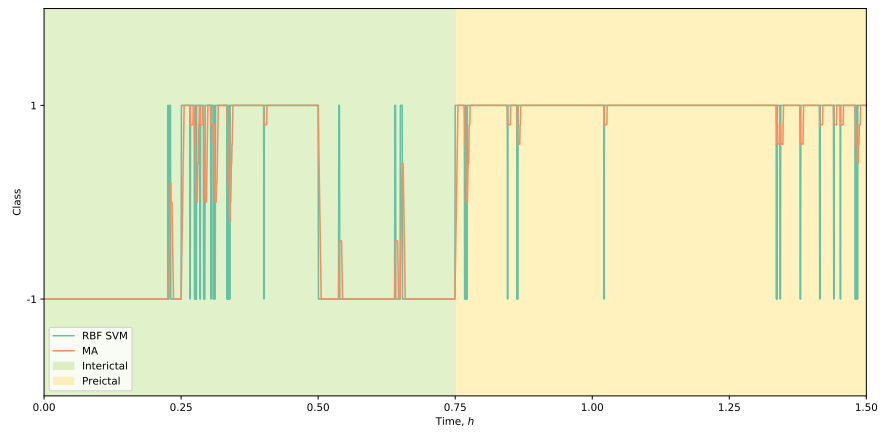


FIGURE 4.8: Prediction with ARMA based RBF SVM using chb06 training set.

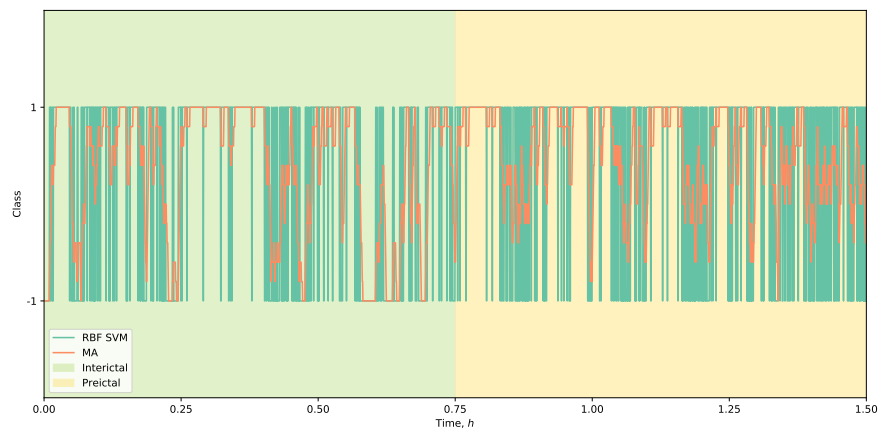


FIGURE 4.9: Prediction with ARMA based RBF SVM using chb10 training set.

4.2.2 Spectral Power

Predictions at 0.029 Hz were obtained by performing FFT with a 2 second Hann sliding window with 50% overlap on the 35 second input signal buffer. The MA filtering parameter was set to 10.

CHB-01

Up to $t = 0.25$, both linear SVM and RBF SVM in figures B.13 and 4.10 generates true negative classifications. For the next 30 minutes, up to the end of the interictal period, both models generate false positives, although at different rates. MA filtering is able to remove false positive classification for both models, although with a lower tolerance in the case of RBF SVM. The Logistic Regression model in figure B.14 generates fewer false positives but the net effect with MA filtering remains roughly the same when MA is applied. For the preictal period, three all models find the input just before $t = 1.25$ difficult to predict correctly, although the RBF SVM model shows the least adverse output. From the start of the preictal period, all models take a similar amount of time to start generating true positives when their MA prediction signals are considered.

CHB-06

Figures B.15 and B.16 show both linear SVM and Logistic Regression begins to generate false positives continuously after $t = 0.25$ and does so for the remainder of the recording. Only the RBF SVM in figure 4.11 is able to filter out false positive classifications between $t = 0.25$ and $t = 0.50$, albeit within a large tolerance with MA filtering. In the preictal period, although the figure 4.11 shows a sudden dip from RBF SVM at $t = 1.25$, MA is able to successfully cancel out this noise. The lack of any noise at all in both interictal and preictal period starting from $t = 0.25$ from the other two models in figures B.15, B.16 could suggest a sort of mode collapse, although this is challenging to verify.

CHB-10

Figures B.17, 4.12 and B.18 show near identical prediction signal both in terms of the raw model output and its MA. The MA predictions seem to follow an oscillatory behaviour with an approximate time period of 0.70 hours. Since the MA signal peaks at around $t = 0.25$ quite close to $y_k = 1$ in the interictal period suggests a struggle to discriminate between two classes, which is in line with our observations made in section 4.1.2.

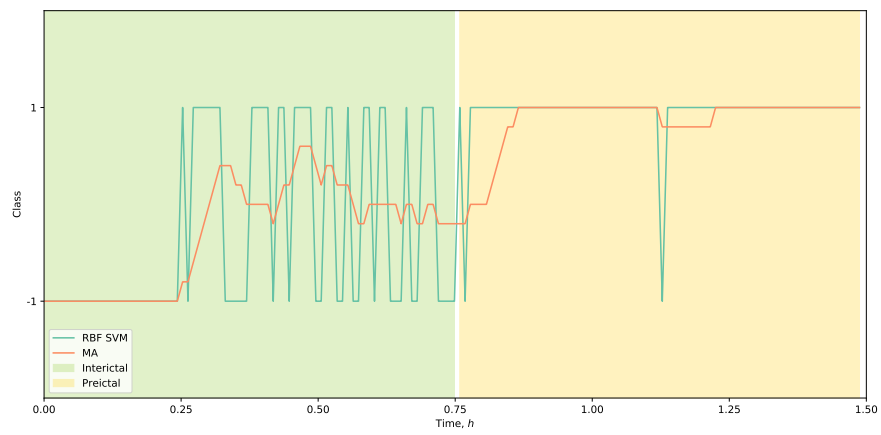


FIGURE 4.10: Prediction with spectral power based RBF SVM using chb01 training set.

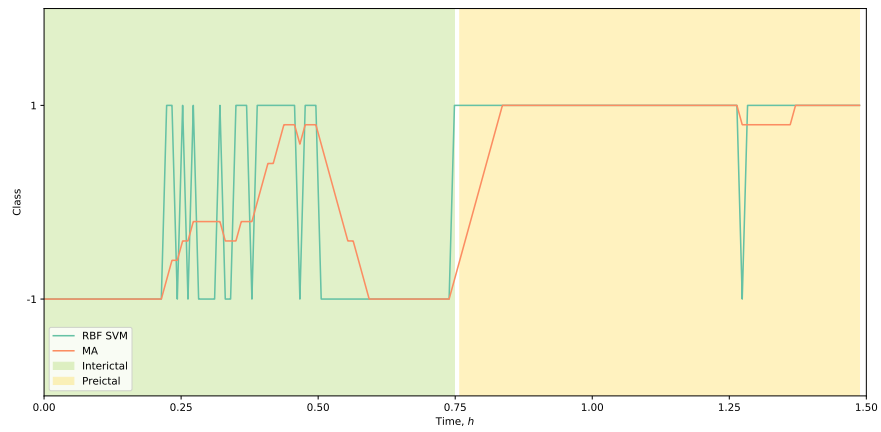


FIGURE 4.11: Prediction with spectral power based RBF SVM using chb06 training set.

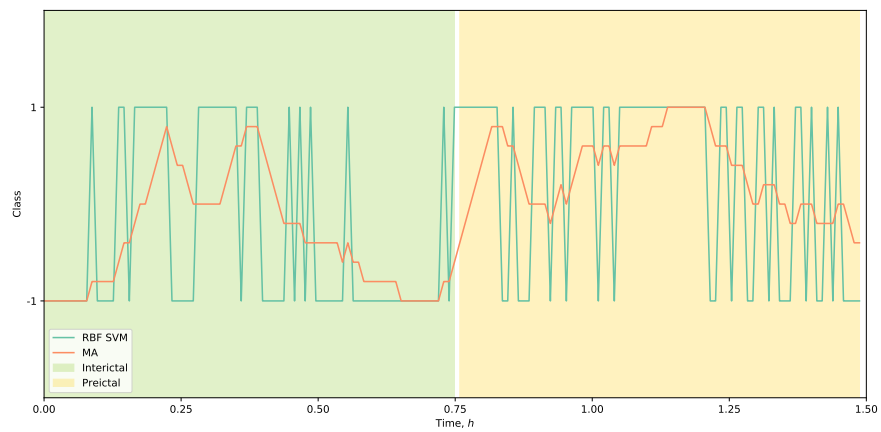


FIGURE 4.12: Prediction with spectral power based RBF SVM using chb10 training set.

4.3 Model Test

We chose the best performing model from the training phase: RBF SVM, to test on an imbalanced dataset that has a interictal to preictal ratio of 4:1. No target labels are supplied to the classifiers, as they perform real-time prediction on the input buffer using the selected feature extraction method.

4.3.1 ARMA

Predictions were generated at 0.5 Hz with MA filtering set to 10 resulting in a real-time update every 2 seconds. Prediction plots for only RBF SVM are provided in this section as the results were identical.

CHB-01

During the interictal period, figure 4.13 shows the two longest duration of true negative classifications were in the interval $t = 0.5$ to $t = 1.0$ and $t = 2.50$ to $t = 3.0$. A very stable false positive classification was predicted from $t = 0.25$ to $t = 0.50$. Starting at $t = 1.0$, the noise in prediction signal became so extreme that MA was unable to counteract it until $t = 1.25$, after which the MA predictions became smoother. We consider the MA signal and find 10 intervals (of length 0.25 hours) out of 12 was correctly classified as true negatives between times $t = 0$ and $t = 3.0$. We consider the patient to have transitioned into preictal period after $t = 3.00$. Here, 2 out of 3 intervals were classified as true positives. Hence, during interictal period, $FPR = \frac{2}{2+10} = 0.17$ and during preictal period $TPR = \frac{2}{2+3} = 0.4$.

CHB-06

The prediction signal for this patient was more noisy than patient chb01 as seen in figure 4.14. We observe the MA signal producing false positives between $t = 0.25$ to $t = 1.0$, then again between $t = 1.50$ to $t = 2.0$. In total, this produces a false positive for 1 hour 15 minutes out of 3 hours of interictal period. Once the preictal period starts at $t = 3.0$ we find false negatives being predicted for approximately 10 more minutes. After this time, it stabilises to producing true positives for another 20 minutes before becoming noisy. For this patient, $FPR = \frac{7}{7+5} = 0.58$ and $TPR = \frac{1}{1+2} = 0.33$.

CHB-10

Figure 4.15 shows an extremely noisy prediction signal where the least amount of false positives are observed between $t = 2.25$ and $t = 3.00$. This patient demonstrates poorest seizure predictability as $FPR = \frac{8}{8+4} = 0.67$ and $TPR = \frac{1}{2+1} = 0.33$.

4.3.2 Spectral Power

Predictions were generated at 0.028 Hz with MA filtering parameter set to 10 resulting in a real-time update every 35 seconds. Prediction plots for only RBF SVM are provided in this section and plots for other models are included in B.

CHB-01

Figure 4.16 shows three sets of false positive classifications, first one being emitted between $t = 0.25$ and $t = 0.50$, the second one at around $t = 1.25$, then again at approximately $t = 2.25$. True positive classification during the preictal period was very stable, although there is a lag between the start of preictal period at $t = 3.0$ and the prediction signal reaching $y_k = 1$. For this patient, the system's $FPR = \frac{3}{3+9} = 0.25$ and $TPR = \frac{3}{3+0} = 1.0$.

CHB-06

This patient shows two periods of false positive predictions compared to patient chb01. As figure 4.17 shows between $t = 0.25$ and $t = 1.0$, then again between $t = 1.50$ and $t = 2.0$ the system predicts a seizure when classifying interictal data. In the preictal period, there is an even longer lag than patient chb01 as it takes almost 20 minutes after $t = 3.0$ for the system to begin emitting true positives. The algorithm's $FPR = \frac{5}{5+7} = 0.42$ and $TPR = \frac{2}{2+1} = 0.67$ for this patient.

CHB-10

Our final patient shows significantly better performance with spectral power features, compared to ARMA features. In the interictal period, no false positives were emitted as the MA was able to effectively smooth out noisy fluctuations from the underlying prediction function. We find that the system started to emit preictal status before the period starts at $t = 3.0$ and continues to maintain it until $t = 3.25$ when it begins to classify false negatives for approximately 10 minutes, before beginning to recover back towards true positive classification, but not quite managing to fully do so. The algorithm achieves $FPR = \frac{0}{0+12} = 0$ and $TPR = \frac{1}{1+2} = 0.33$ for this patient.

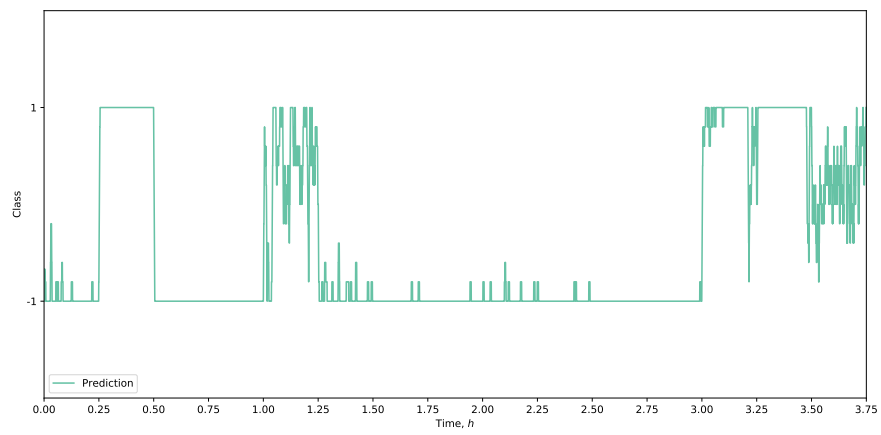


FIGURE 4.13: Prediction with ARMA based RBF SVM using chb01 test set.

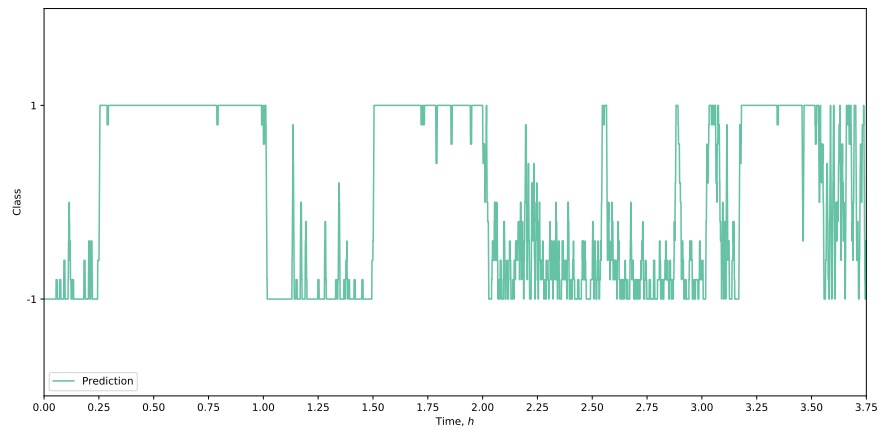


FIGURE 4.14: Prediction with ARMA based RBF SVM using chb06 test set.

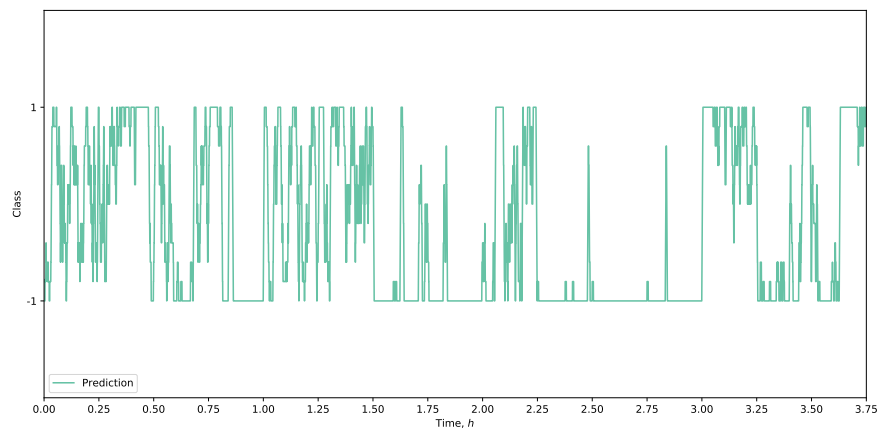


FIGURE 4.15: Prediction with ARMA based RBF SVM using chb10 test set.

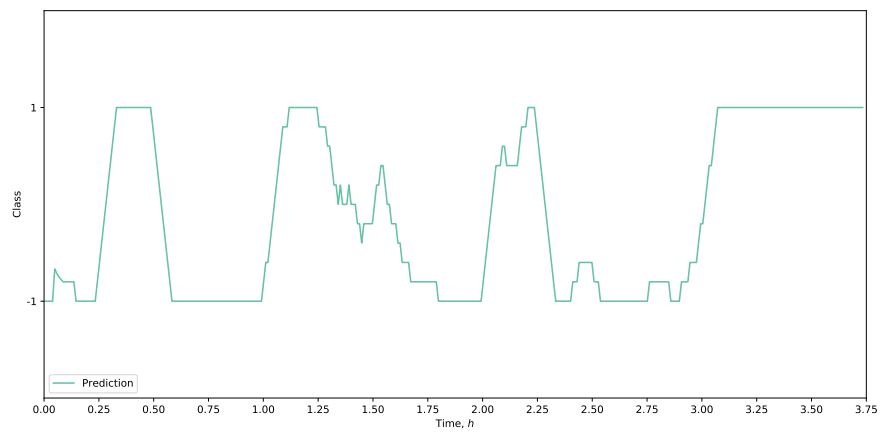


FIGURE 4.16: Prediction with spectral power based RBF SVM using chb01 test set.

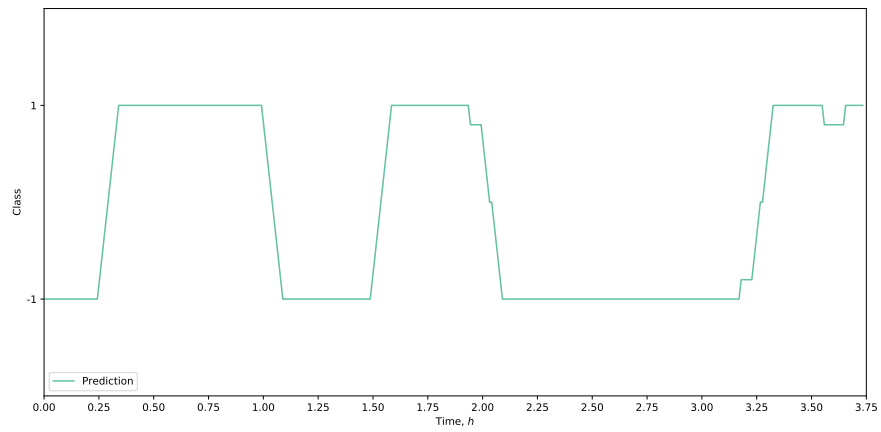


FIGURE 4.17: Prediction with spectral power based RBF SVM using chb06 test set.

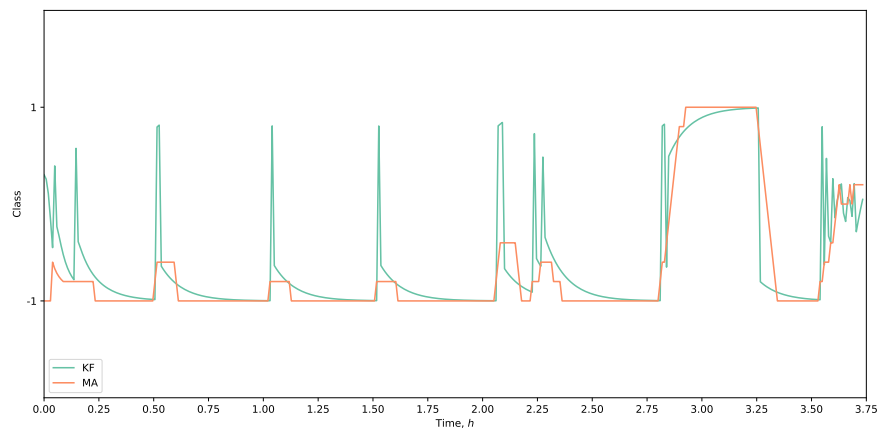


FIGURE 4.18: Prediction with spectral power based RBF SVM using chb10 test set.

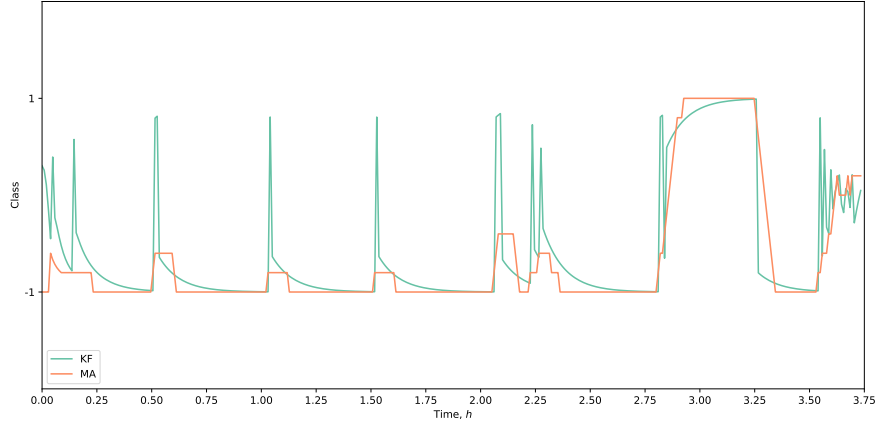


FIGURE 4.19: Comparison of Kalman filter with moving average filter on prediction regularisation.

4.4 Regularisation

We compare the regularisation performance of KF (Kalman filter) against MA on raw prediction signal generated by RBF SVM based on spectral power features using patient chb10 test data. As shown in figure 4.19, we find that in nearly all cases, MA provides a more effective smoothing as effectively suppresses false positives during the interictal period. However, in the preictal period at $t = 3.50$, neither filters are able to prevent false negatives from being emitted. In addition, as explained in section 3.7.2, the KF is an online dynamic system tracking algorithm. Thus, we find a higher level of noise near the beginning of the recording as it tries to learn the position of the raw prediction signal and lock onto it in real-time.

4.5 Summary

We summarise our results in table 4.1a which shows the FPR and TPR for each patient using ARMA feature extraction. In table 4.1b, we present similar information obtained using spectral power features. All of the predictions were performed using an SVM with RBF kernel, with the MA filtered signal considered as the prediction signal.

| Patient | Interictal FPR | Preictal TPR | Patient | Interictal FPR | Preictal TPR |
|---------|----------------|--------------|---------|----------------|--------------|
| CHB-01 | 0.17 | 0.4 | CHB-01 | 0.25 | 1 |
| CHB-06 | 0.58 | 0.33 | CHB-06 | 0.42 | 0.67 |
| CHB-10 | 0.67 | 0.33 | CHB-10 | 0 | 0.33 |

(A) ARMA
(B) Spectral Power

TABLE 4.1: Prediction performance on test data for all patients.

Chapter 5

Discussion

The result of placing the constraint of our selection criteria on the dataset resulted in a severely reduced subset of patient data, which when split into training and test set, resulted in an even smaller subset of data. This relatively small quantity of data makes the result of our study less statistically robust, as without generating further results using more patient data, we have a high relatively degree of uncertainty.

As the dataset we have used contains very few seizures, we decided to combine the 15 minute segments of preictal data from multiple seizures and merge them together to produce a larger preictal class of data which was then used to visualise features in chapter 4. We have done this to capture a larger sample of preictal data from different seizures so that during training time, our machine learning algorithm is exposed to the various patterns which may arise during different seizures. It could be argued that this results in contamination of data between seizures, but we believe that since the preictal period is so loosely defined, the ramifications of this ‘contamination’ is inconsequential.

We are also aware that EEG signals can be contaminated from power line noise at 50 Hz. Many previous studies simply use an FIR notch filter (also known as bandpass filter) to destroy information at this frequency. Whilst we do not want neural data to be contaminated with external noise, this process also destroys any brain signals that could have been emitted in this frequency. While this was the justification for not including this pre-processing step, the modest performance achieved by our model does raise questions about if it could have performed better with noise filtering.

An AR(2) model was chosen for ARMA mode as our assumption was that any correlation the present data has must be with the immediate past. A higher order model takes into account more of the past data at the expense of additional computational complexity. In addition, we wished to explore the impact of performing classification of neural data at very low dimensions to find if the classes

could be separable. However, as we saw in figure 4.3 separation was hardly possible for patient chb10. As expected, real-time prediction on test set produced very noisy prediction as we saw in figure 4.15. On the other hand, when using spectral band power features, our predictor performed significantly better as shown in figure 4.18. The reasons for this could be twofold: first, that the AR(2) model could be forcing high dimensional data into a subspace that severely limits separation, or second, spectral bandpower features better capture the state of the brain than statistical features of the raw signal such as autocorrelation.

If this were to be the case, we would have expected the FPR rate of the study by [Howbert et al. \(2014\)](#) on canine subjects using iEEG to have been much lower than the one by [Chisci et al. \(2010\)](#). Further experimental results using a range of higher-order AR models are needed to verify the efficacy of these two methods.

Further reasons for difference in performance between ARMA and spectral power features could arise from taking the mean of features from all 23 channels of raw data. As we have seen in figure 2.6, a single raw signal modelled with AR(2) produces 2 parameter signals varying over time. Then it goes to show that for 23 channels, we have 46 AR signals to track. Using the mean allowed us to reduce the 46 signals back down to two, but this averaging comes at the cost of smaller variations in parameters which could have the potential to be more sensitive to preictal data. For spectral power in band, a similar smoothing effect occurs as a result of taking the mean of relative neural bandpower from individual channels. This step has caused our system to sacrifice localisation data as each electrode is responsible for capturing neural data over a specific area. Degradation in performance arising from loss of localised data is expected as we have seen other feature extraction methods based on graph theory which precisely tries to capture this information as seen in the study by [Tsiouris et al. \(2018\)](#).

The choice of machine learning algorithm was guided by interpretability as our algorithm has been developed applied for a medical context. To this end, we began our testing with a linear kernel SVM as the predicted class for a given test point could be extracted by finding the side of the hyperplane it lies on. Furthermore, as the support vectors of the hyperplane rely on a smaller subset of datapoints to fix its position, we can compute the distance of the test point from the hyperplane to establish a confidence metric: the further the test point is from the correct side of the margin, the more confident we can be that the test point belongs to that class. If however, the test point lies between the support margin and the hyperplane, we can claim our prediction with low confidence. Notice that we discard the logistic regression model as it was only used as a baseline comparison during training time.

Findings from section 4.2 demonstrated that a linear kernel SVM produces prediction signals that were more noisy than predictions generated using RBF kernel. This result

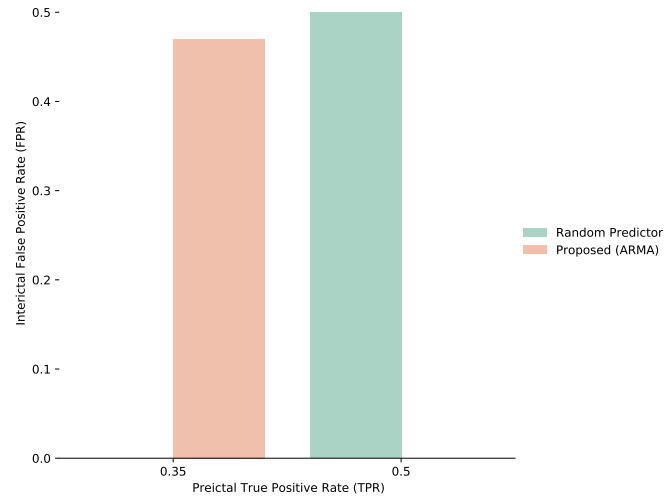
further supports the idea in [Park et al. \(2011\)](#) which claims there is a degradation in performance when switching from a non-linear SVM kernel to a linear one. However we have not been able to directly measure the extent of this performance degradation. We are therefore led to think that our features were not linearly separable in the feature space. A cost parameter could be optimised during training with cross-validation (CV) to improve the performance of linear kernel SVM, but as we have not considered CV in this study, we discarded it from model testing.

A stream buffer of size 512 was used to sample the incoming signal in real-time where we downsample the output such features. For the purposes of this study, we chose to keep the size the same for all subjects. As the data we used were sampled at 256 Hz, a buffer size of 512 translates to $N/f_s = 2$ seconds of input samples. In the case of AR modelling, a stationarity assumption is made about the signal in the buffer. Thus, setting the buffer size to be too large could result in a poor fit by least-squares regression (exemplified in figure 2.5), resulting in poor parameter approximation. A way of verifying the goodness of fit could be to verify if the estimated parameters are able to recreate the input signal in a way shown in figure 2.6. But, as acquire a downsampled parameter estimation, this approach is not possible and an alternative method needs to be explored.

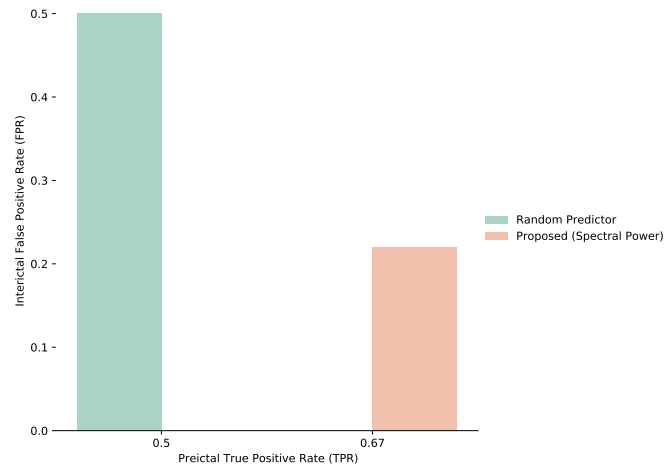
For spectral power in band feature generation, we compute the window size $I_k = f_s \times N$. Since our lowest frequency of interest $f_{min} = 0.1$ Hz which lies in the δ neural band, the lower bound on the size our buffer is given by $2 \times f_{min}^{-1} = 20$ seconds. Thus, we capture 35 seconds of data in our input buffer and use a 20 second Hann window with 50% overlap to compute Welch's PSD. The Hann window function has been chosen because it achieves good apodisation of the sampled signal (i.e. it achieves good tapering). The overlap of 50% is required as the Hann window would otherwise cause some samples in between the windows to be lost due to the tapering. In our experiments, we found that reducing the 35 second input buffer to a shorter duration introduced aliasing.

When we applied the Kalman filter to our prediction signal, we found its smoothing properties did not surpass the one produced by a moving average filter. This result does not match with earlier studies where [Park et al. \(2011\)](#) successfully regularised noisy predictions from their underlying model. The reason for this is not clear as the authors of the previous work has not provided more details on the configuration of their filter but it may have something to do with the way they model the dynamic system's state-space. An obvious next step could be to combine the two filters: MA and KF, but it should be kept in mind that the KF does incur a significantly higher computational complexity than the MA.

Based on the results in chapter 4, in ARMA mode, we found our system's overall $FPR = 0.47$ during interictal period and $TPR = 0.35$ during preictal period. On



(A) ARMA



(B) Spectral Power

FIGURE 5.1: Performance of proposed algorithm compared against random predictor.

spectral power mode, the interictal period had $FPR = 0.22$ and preictal period had $TPR = 0.67$ when all three patients we tested our system on were considered. We first compare the performance of ARMA mode with a baseline random predictor, which is expected to have an $FPR = 0.5$ and $TPR = 0.5$. Figure 5.1a shows that our second-order ARMA feature based model performed worse than random chance. On the other hand, using spectral features, figure 5.1b shows our algorithm has better performance than purely random chance. This indicates that the overall framework we have adopted in this paper has abundant room for further progress.

Chapter 6

Conclusion

In this dissertation, we have proposed a patient-specific seizure prediction system based on non-invasive scalp EEG signals. We have found that the field of seizure prediction is mature and has embraced a variety of approaches to extract different aspects of information from the raw signal. The system we designed could ultimately be used in a wearable device to provide users with an advance warning about a sudden seizure. It has been a challenging endeavour to develop a system for which so little data is available. Our hope is that with newer and larger open access seizure databases, such as the TUH Seizure Dataset, algorithms such as ours could be further refined to the point it has medical relevance in the real world.

Future work should focus on validating our findings on a larger set of patient data and establish which feature extraction method is more powerful with a greater statistical significance. Once this is achieved, an alarm threshold should be empirically set based on the prediction pattern generated by a patient. The alarm threshold could serve as a barrier to prevent further false positives in two ways: by ensuring noisy predictions below the threshold do not generate a warning; tolerating a number violations beyond the threshold to increase certainty about an incoming seizure.

Biologically-inspired feature extraction methods based on deep learning such as the convolutional networks also remain a promising method to automatically learn features invariant to patients. In the medical domain, explainable AI remains an elusive challenge. To this end, exploring game theoretic approaches, such as the Shapley Additive Explanations combined with clinically relevant features such as the spectral power in band or learnt convolutional filters should provide a powerful, explainable system that is interrogable by clinicians.

For now, we believe this work has answered our research question: it is indeed possible to predict seizures in real-time based on EEG data. But despite the promising results, a larger question still remains: are all kinds of seizures predictable?

Appendix A

Dataset

In this appendix we provide summary statistics figures and tables describing the data used in the study.

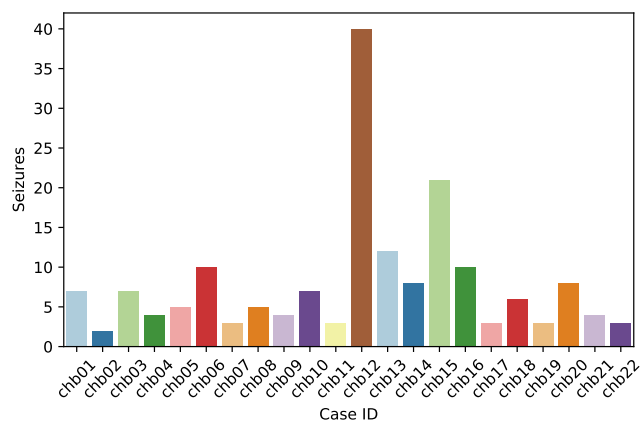


FIGURE A.1: Seizure count by patients in overall dataset.

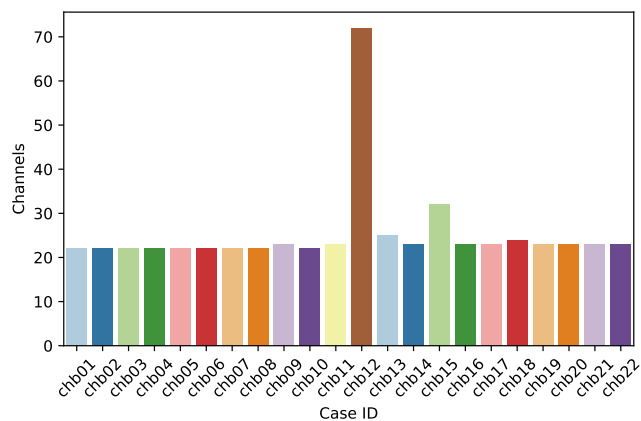


FIGURE A.2: Channel count by patients in overall dataset.

TABLE A.1: Summary statistics of overall dataset.

| | |
|---------------------|------|
| Total Case Count | 22 |
| Total Seizure Count | 175 |
| Mean Seizure Count | 7.95 |
| Mode Channel Count | 23 |

TABLE A.2: chb01 Seizure Times

| Filename | Total Duration (s) | Seizure Start (s) |
|--------------|--------------------|-------------------|
| chb01_03.edf | 3600 | 2996 |
| chb01_04.edf | 3600 | 1467 |
| chb01_15.edf | 3600 | 1732 |
| chb01_16.edf | 3600 | 1015 |
| chb01_18.edf | 3600 | 1720 |
| chb01_21.edf | 3600 | 327 |
| chb01_26.edf | 3600 | 1862 |

TABLE A.3: chb03 Seizure Times

| Filename | Total Duration (s) | Seizure Start (s) |
|--------------|--------------------|-------------------|
| chb03_01.edf | 3600 | 362 |
| chb03_02.edf | 3600 | 731 |
| chb03_03.edf | 3600 | 432 |
| chb03_04.edf | 3600 | 2162 |
| chb03_34.edf | 3600 | 1982 |
| chb03_35.edf | 3600 | 2592 |
| chb03_36.edf | 3600 | 1725 |

TABLE A.4: chb06 Seizure Times

| Filename | Total Duration (s) | Seizure Start (s) |
|--------------|--------------------|-------------------|
| chb06_01.edf | 14400 | 1724, 7461, 13525 |
| chb06_04.edf | 14400 | 327, 6211 |
| chb06_09.edf | 14400 | 12500 |
| chb06_10.edf | 14400 | 10833 |
| chb06_13.edf | 14400 | 506 |
| chb06_18.edf | 14400 | 7799 |
| chb06_24.edf | 14400 | 9387 |

TABLE A.5: chb10 Seizure Times

| Filename | Total Duration (s) | Seizure Start (s) |
|--------------|--------------------|-------------------|
| chb10_12.edf | 7200 | 6313 |
| chb10_20.edf | 7200 | 6888 |
| chb10_27.edf | 7200 | 2382 |
| chb10_30.edf | 7200 | 3021 |
| chb10_31.edf | 7200 | 3801 |
| chb10_38.edf | 7200 | 4618 |
| chb10_89.edf | 7200 | 1383 |

TABLE A.6: Training and test set: Part A

| Case | Filename | Set | Class | Seizure Start | Crop Start | Crop End |
|-------|--------------|-------|------------|---------------|------------|----------|
| chb01 | chb01_03.edf | Train | Preictal | 2996 | 2095 | 2995 |
| chb01 | chb01_04.edf | Train | Preictal | 1467 | 566 | 1466 |
| chb01 | chb01_15.edf | Train | Preictal | 1732 | 831 | 1731 |
| chb01 | chb01_16.edf | Test | Preictal | 1015 | 114 | 1014 |
| chb01 | chb01_18.edf | Test | Preictal | 1720 | 819 | 1719 |
| chb01 | chb01_26.edf | Test | Preictal | 1862 | 961 | 1861 |
| chb06 | chb06_01.edf | Train | Preictal | 1724 | 823 | 1723 |
| chb06 | chb06_01.edf | Train | Preictal | 7461 | 6560 | 7460 |
| chb06 | chb06_01.edf | Train | Preictal | 13525 | 12624 | 13524 |
| chb06 | chb06_09.edf | Test | Preictal | 12500 | 11599 | 12499 |
| chb06 | chb06_10.edf | Test | Preictal | 10833 | 9932 | 10832 |
| chb06 | chb06_18.edf | Test | Preictal | 7799 | 6898 | 7798 |
| chb10 | chb10_12.edf | Train | Preictal | 6313 | 5412 | 6312 |
| chb10 | chb10_20.edf | Train | Preictal | 6888 | 5987 | 6887 |
| chb10 | chb10_27.edf | Train | Preictal | 2382 | 1481 | 2381 |
| chb10 | chb10_30.edf | Test | Preictal | 3021 | 2120 | 3020 |
| chb10 | chb10_31.edf | Test | Preictal | 3801 | 2900 | 3800 |
| chb10 | chb10_38.edf | Test | Preictal | 4618 | 3717 | 4617 |
| chb01 | chb01_01.edf | Train | Interictal | 0 | 0 | 900 |
| chb01 | chb01_02.edf | Train | Interictal | 0 | 0 | 900 |
| chb01 | chb01_05.edf | Train | Interictal | 0 | 0 | 900 |
| chb01 | chb01_06.edf | Test | Interictal | 0 | 0 | 900 |
| chb01 | chb01_07.edf | Test | Interictal | 0 | 0 | 900 |
| chb01 | chb01_08.edf | Test | Interictal | 0 | 0 | 900 |
| chb01 | chb01_09.edf | Test | Interictal | 0 | 0 | 900 |
| chb01 | chb01_10.edf | Test | Interictal | 0 | 0 | 900 |
| chb01 | chb01_11.edf | Test | Interictal | 0 | 0 | 900 |
| chb01 | chb01_12.edf | Test | Interictal | 0 | 0 | 900 |
| chb01 | chb01_13.edf | Test | Interictal | 0 | 0 | 900 |

TABLE A.7: Training and test set: Part B

| Case | Filename | Set | Class | Seizure Start | Crop Start | Crop End |
|-------|--------------|-------|------------|---------------|------------|----------|
| chb01 | chb01_14.edf | Test | Interictal | 0 | 0 | 900 |
| chb01 | chb01_17.edf | Test | Interictal | 0 | 0 | 900 |
| chb01 | chb01_19.edf | Test | Interictal | 0 | 0 | 900 |
| chb01 | chb01_20.edf | Test | Interictal | 0 | 0 | 900 |
| chb06 | chb06_02.edf | Train | Interictal | 0 | 0 | 900 |
| chb06 | chb06_03.edf | Train | Interictal | 0 | 0 | 900 |
| chb06 | chb06_05.edf | Train | Interictal | 0 | 0 | 900 |
| chb06 | chb06_06.edf | Test | Interictal | 0 | 0 | 900 |
| chb06 | chb06_07.edf | Test | Interictal | 0 | 0 | 900 |
| chb06 | chb06_08.edf | Test | Interictal | 0 | 0 | 900 |
| chb06 | chb06_12.edf | Test | Interictal | 0 | 0 | 900 |
| chb06 | chb06_14.edf | Test | Interictal | 0 | 0 | 1800 |
| chb06 | chb06_15.edf | Test | Interictal | 0 | 0 | 1800 |
| chb06 | chb06_16.edf | Test | Interictal | 0 | 0 | 1800 |
| chb06 | chb06_17.edf | Test | Interictal | 0 | 0 | 1800 |
| chb10 | chb10_01.edf | Train | Interictal | 0 | 0 | 900 |
| chb10 | chb10_02.edf | Train | Interictal | 0 | 0 | 900 |
| chb10 | chb10_03.edf | Train | Interictal | 0 | 0 | 900 |
| chb10 | chb10_04.edf | Test | Interictal | 0 | 0 | 900 |
| chb10 | chb10_05.edf | Test | Interictal | 0 | 0 | 900 |
| chb10 | chb10_06.edf | Test | Interictal | 0 | 0 | 900 |
| chb10 | chb10_07.edf | Test | Interictal | 0 | 0 | 900 |
| chb10 | chb10_08.edf | Test | Interictal | 0 | 0 | 900 |
| chb10 | chb10_13.edf | Test | Interictal | 0 | 0 | 900 |
| chb10 | chb10_14.edf | Test | Interictal | 0 | 0 | 900 |
| chb10 | chb10_15.edf | Test | Interictal | 0 | 0 | 900 |
| chb10 | chb10_16.edf | Test | Interictal | 0 | 0 | 900 |
| chb10 | chb10_17.edf | Test | Interictal | 0 | 0 | 900 |
| chb10 | chb10_18.edf | Test | Interictal | 0 | 0 | 900 |
| chb10 | chb10_19.edf | Test | Interictal | 0 | 0 | 900 |

Appendix B

Figures

In this appendix we provide feature responses extracted from the raw EEG signals and prediction signals generated on training dataset.

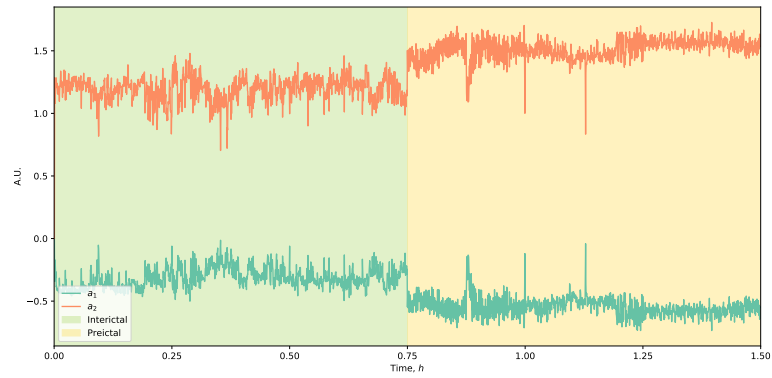


FIGURE B.1: Interictal and preictal period ARMA response from chb01 training data.

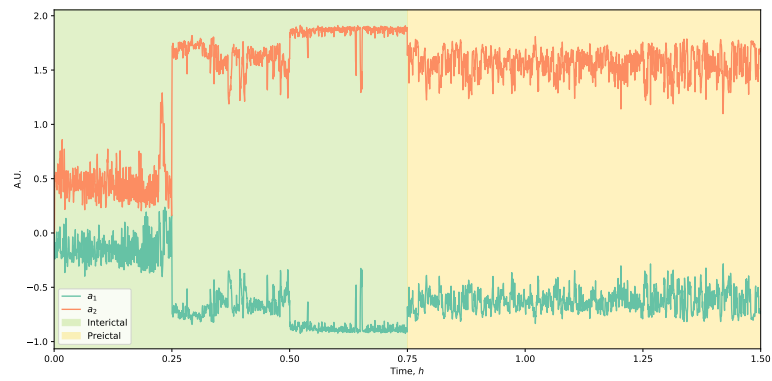


FIGURE B.2: Interictal and preictal period ARMA response from chb06 training data.

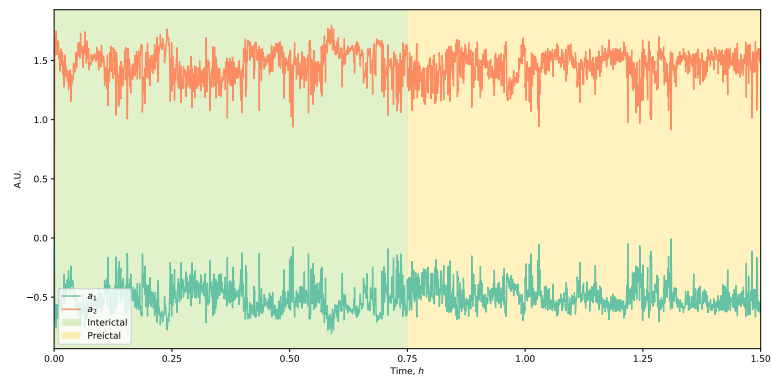


FIGURE B.3: Interictal and preictal period ARMA response from chb10 training data.

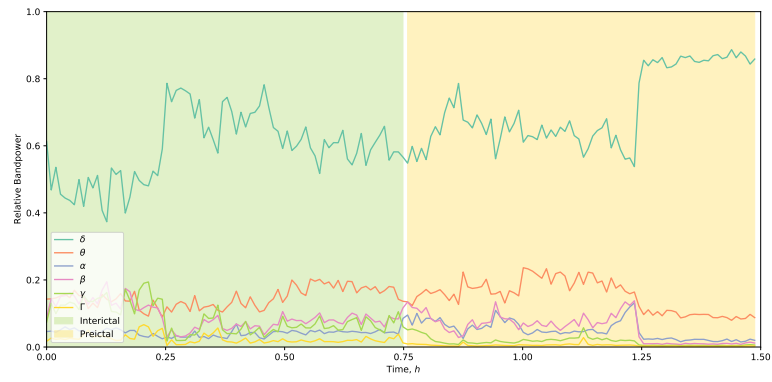


FIGURE B.4: Interictal and preictal period spectral bandpower response from chb01 training data.

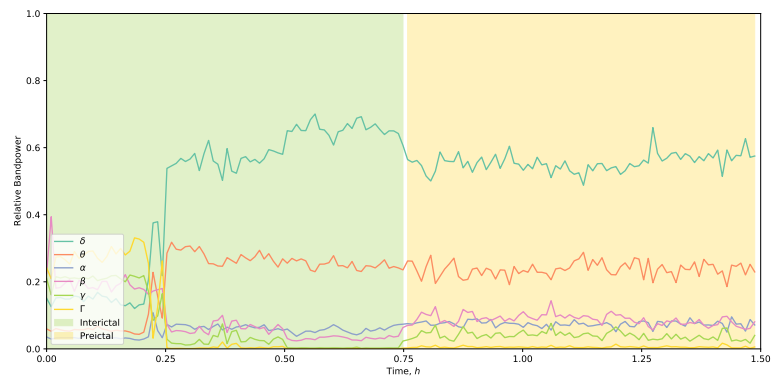


FIGURE B.5: Interictal and preictal period spectral bandpower response from chb06 training data.

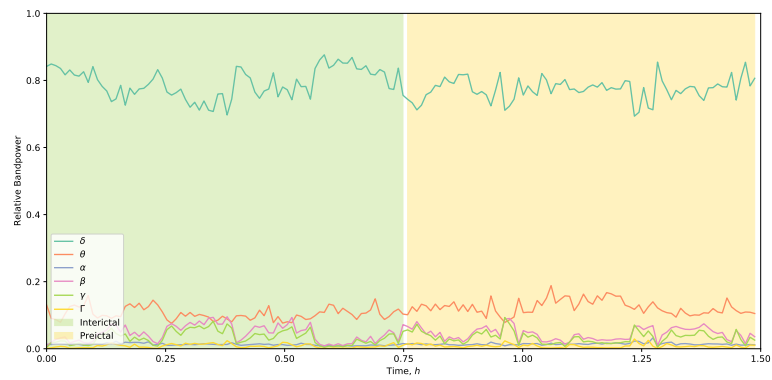


FIGURE B.6: Interictal and preictal period spectral bandpower response from chb10 training data.

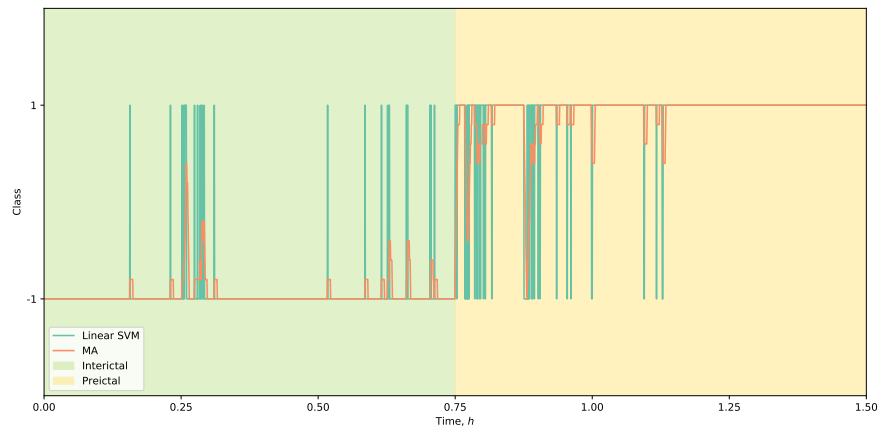


FIGURE B.7: Prediction with ARMA based linear SVM using chb01 training set.

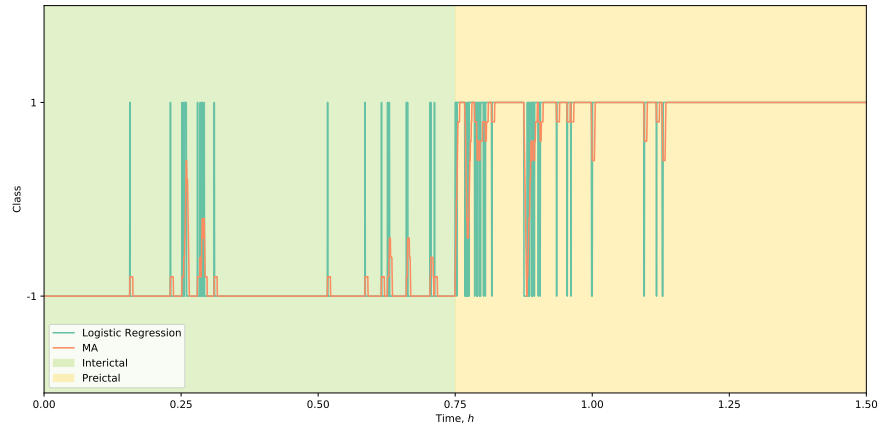


FIGURE B.8: Prediction with ARMA based Logistic Regression using chb01 training set.

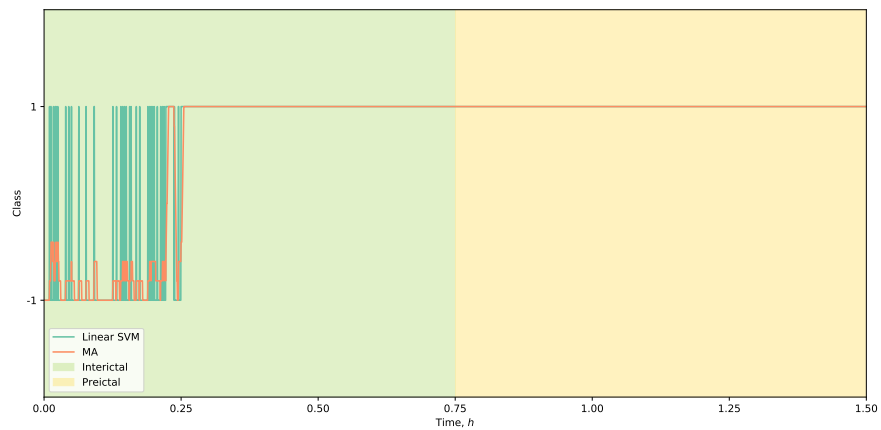


FIGURE B.9: Prediction with ARMA based linear SVM using chb06 training set.

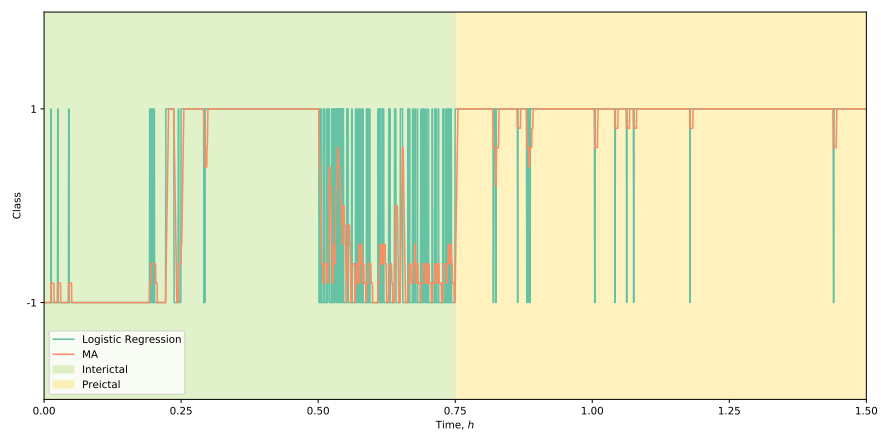


FIGURE B.10: Prediction with ARMA based Logistic Regression using chb06 training set.

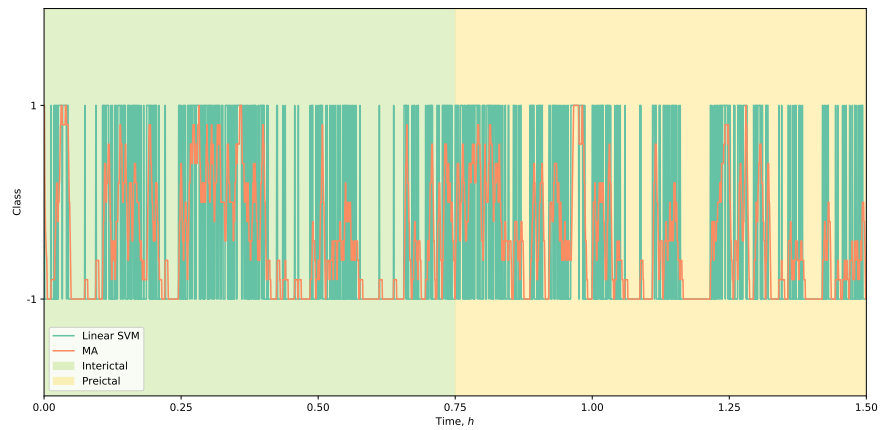


FIGURE B.11: Prediction with ARMA based linear SVM using chb10 training set.

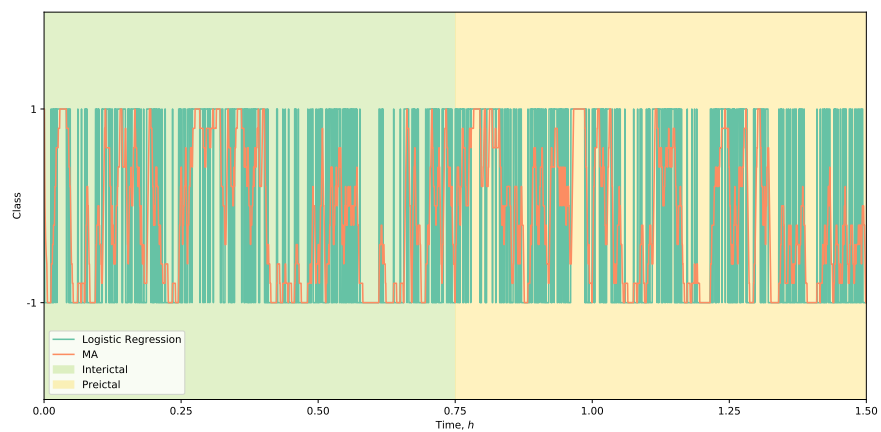


FIGURE B.12: Prediction with ARMA based Logistic Regression using chb10 training set.

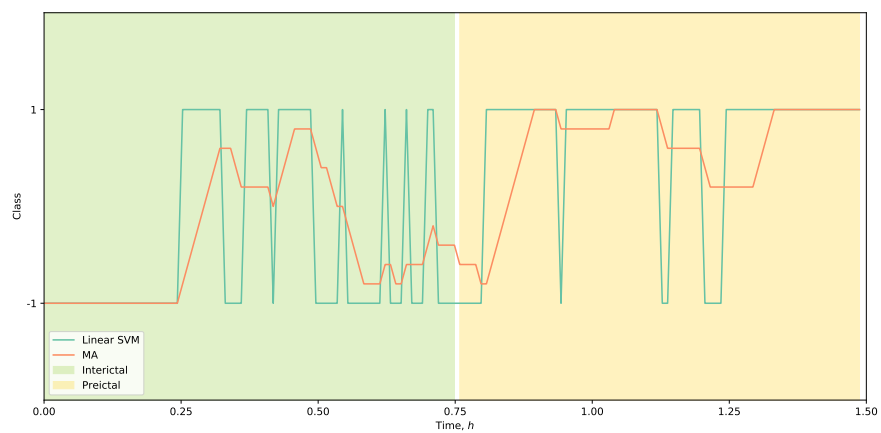


FIGURE B.13: Prediction with spectral power based linear SVM using chb01 training set.

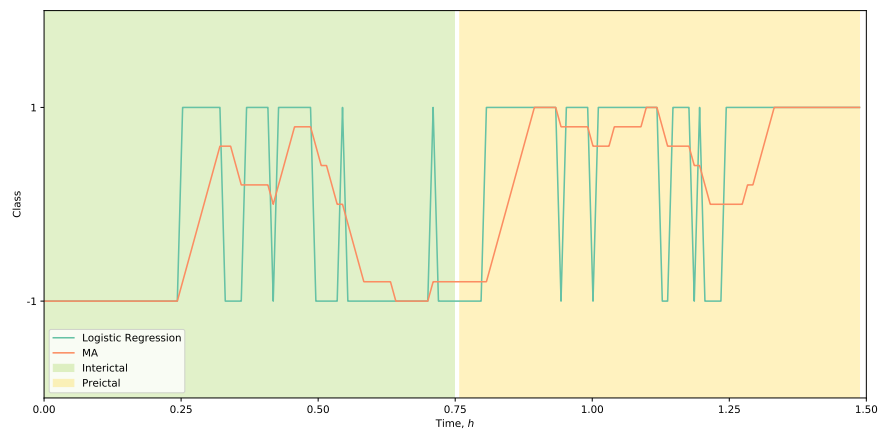


FIGURE B.14: Prediction with spectral power based Logistic Regression using chb01 training set.

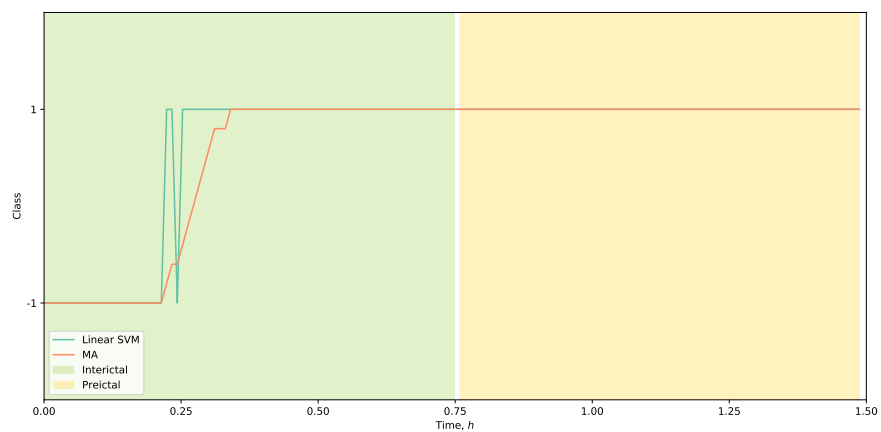


FIGURE B.15: Prediction with spectral power based linear SVM using chb06 training set.

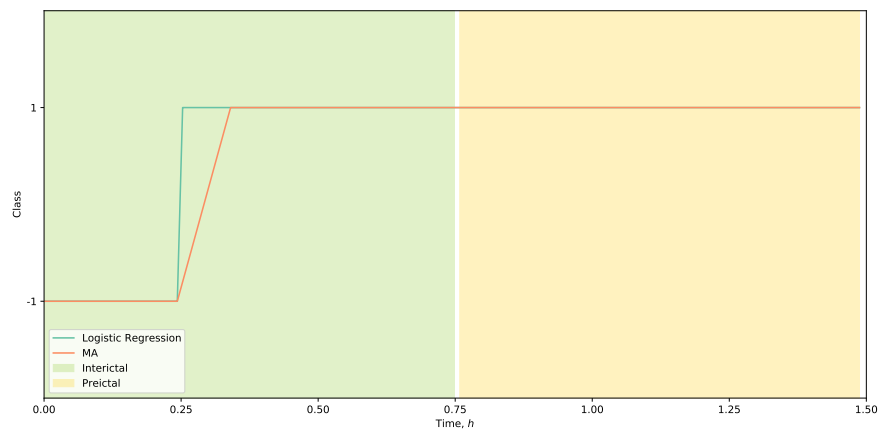


FIGURE B.16: Prediction with spectral power based Logistic Regression using chb06 training set.

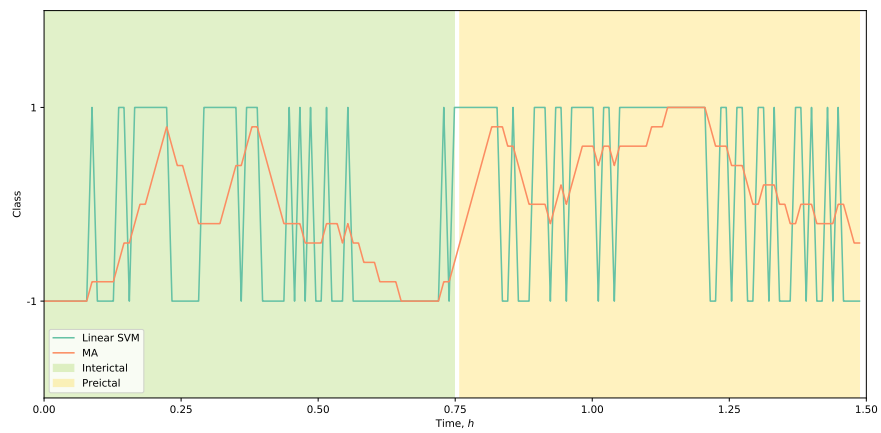


FIGURE B.17: Prediction with spectral power based linear SVM using chb10 training set.

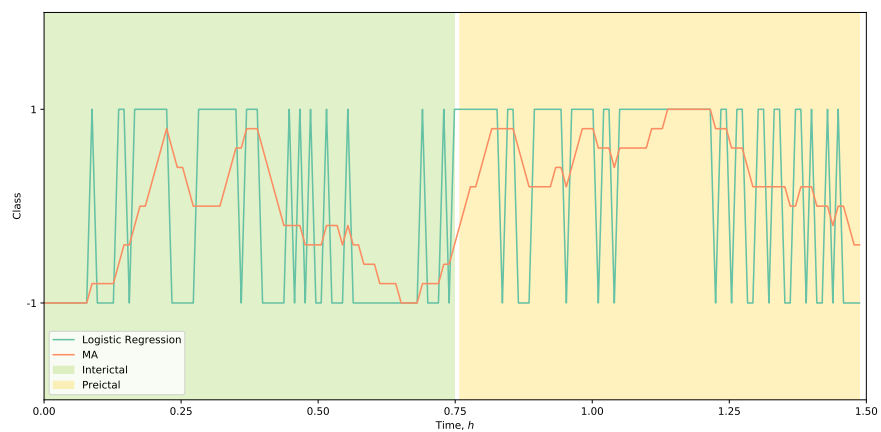


FIGURE B.18: Prediction with spectral power based Logistic Regression using chb10 training set.

References

- J. Kiffin Bancaud, Jean; Penry. Proposal for Revised Clinical and Electroencephalographic Classification of Epileptic Seizures. *Epilepsia*, 22(4): 489–501, 8 1981. ISSN 0013-9580. . URL <https://onlinelibrary.wiley.com/doi/10.1111/j.1528-1157.1981.tb06159.x>.
- Luigi Chisci, Antonio Mavino, Guido Perferi, Marco Sciandrone, Carmelo Anile, Gabriella Colicchio, and Filomena Fuggetta. Real-time epileptic seizure prediction using AR models and support vector machines. *IEEE Transactions on Biomedical Engineering*, 57(5):1124–1132, 2010. ISSN 00189294. .
- Mike X Cohen. *Analyzing Neural Time Series Data*. MIT Press, 2019. .
- Robert S. Fisher, Walter Van Emde Boas, Warren Blume, Christian Elger, Pierre Genton, Phillip Lee, and Jerome Engel. Response: Definitions proposed by the International League Against Epilepsy (ILAE) and the International Bureau for Epilepsy (IBE) [4]. *Epilepsia*, 46(10):1701–1702, 2005. ISSN 00139580. .
- A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C. K. Peng, and H. E. Stanley. PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. *Circulation*, 101(23), 2000. ISSN 15244539. .
- J. Jeffry Howbert, Edward E. Patterson, S. Matt Stead, Ben Brinkmann, Vincent Vasoli, Daniel Crepeau, Charles H. Vite, Beverly Sturges, Vanessa Ruedebusch, Jaideep Mavoori, Kent Leyde, W. Douglas Sheffield, Brian Litt, and Gregory A. Worrell. Forecasting Seizures in Dogs with Naturally Occurring Epilepsy. *PLoS ONE*, 9(1): e81920, 1 2014. ISSN 1932-6203. . URL <https://dx.plos.org/10.1371/journal.pone.0081920>.
- Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibishirani. *An Introduction to Statistical Learning with Applications in R*, volume 11. Springer International Publishing, 2013. ISBN 978-1-4614-7137-0. URL <http://books.google.com/books?id=9tv0taI8l6YC>.

- Philippa J. Karoly, Hoameng Ung, David B. Grayden, Levin Kuhlmann, Kent Leyde, Mark J. Cook, and Dean R. Freestone. The circadian profile of epilepsy improves seizure forecasting. *Brain*, 140(8):2169–2182, 2017. ISSN 14602156. .
- Yun Park, Lan Luo, Keshab K. Parhi, and Theoden Netoff. Seizure prediction with spectral power of EEG using cost-sensitive support vector machines. *Epilepsia*, 52(10):1761–1770, 2011. ISSN 00139580. .
- Steven C. Schachter and Clifford B. Saper. Vagus Nerve Stimulation. *Epilepsia*, 39(7):677–686, 7 1998. ISSN 0013-9580. . URL <https://onlinelibrary.wiley.com/doi/10.1111/j.1528-1157.1998.tb01151.x>.
- Ali Shoeb. Application of machine learning to epileptic seizure onset detection and treatment. *Diss. Massachusetts Institute of Technology*, pages 157–162, 2009. URL <http://dspace.mit.edu/handle/1721.1/54669>.
- Nhan Duy Truong, Anh Duy Nguyen, Levin Kuhlmann, Mohammad Reza Bonyadi, Jiawei Yang, Samuel Ippolito, and Omid Kavehei. Convolutional neural networks for seizure prediction using intracranial and scalp electroencephalogram. *Neural Networks*, 105:104–111, 2018. ISSN 18792782. . URL <https://doi.org/10.1016/j.neunet.2018.04.018>.
- Kostas Tsiouris, Vasileios C. Pezoulas, Michalis Zervakis, Spiros Konitsiotis, Dimitrios D. Koutsouris, and Dimitrios I. Fotiadis. A Long Short-Term Memory deep learning network for the prediction of epileptic seizures using EEG signals. *Computers in Biology and Medicine*, 99(March):24–37, 2018. ISSN 18790534. .
- Peter D. Welch. The Use of Fast Fourier Transform for the Estimation of Power Spectra: A Method Based on Time Averaging Over Short, Modified Periodograms. *IEEE Transactions on Audio and Electroacoustics*, 15(2):70–73, 1967. ISSN 00189278. .