BIOMEDICAL ENGINEERING MSc PROGRAM
UNIVERSITY OF CRETE
TECHNICAL UNIVERSITY OF CRETE
FOUNDATION FOR RESEARCH AND TECHNOLOGY-HELLAS

Application Grade Thesis

Title: Design of a clinical trial simulator for studying infant seizures

Student's Name: Torakis Ioannis

Supervisor's Name: Prof. Zervakis Michalis

Date of completion:

26/04/2022

This dissertation is submitted as a partial fulfilment of the requirements for the Master's degree of Biomedical Engineering M.Sc. program

Acknowledgements

To begin with, I would like to thank my parents Dimitris and Chrysanthi for their continuous support, help and patience in every decision that I have made in all these years. I would also like to thank my brothers Sotiris and Stefanos for their patience and tolerance towards my strange character. Special thanks to my supervisor, Prof. Michael Zervakis for his useful guidance and valuable help, due to which I decided to take this MSc program, during a tough period in my life. Additional gratitude for involving me in such an interesting research subject. I would also like to thank Dr. Katerina Bei for her useful insights during the search of bibliography and her continuous support.

Last, but not least, I would like to express my special gratitude and appreciation to Dr. Marios Antonakakis for all his help and support throughout all these months. His insights, mentoring, suggestions, expertise and experience were key factors for me in order to accomplish this thesis and working under his guidance was an honour.

Abstract

Neonatal seizures are a condition happening in early childhood years and it is accounting for several deaths and severe problems on newborn neonates. Despite the early advancements on the treatment of this condition, the main problem concerning the physicians is the difficulty to identify and characterize a seizure, as one a small percentage gets detected in neonatal intensive care units (NICU). Multi-channel EEG signal analysis is the gold standard for seizure detection. However, the interpretation of such signals presents a great challenge, since only experienced pediatric neurologists who have emphasized in neonatal EEG analysis can perform this task. Machine learning methods can become a useful tool in the interpretation of EEG signals and in the assignment of seizure classification and regression tasks.

Various studies exist in the literature that have also employed supervised machine learning methods for neonatal seizure classification. However, an important step before proceeding with seizure classification, is rejecting the multiple artefacts that exist throughout the whole EEG signal. Especially in neonatal EEG analysis, where there are more artifacts compared to adult EEG signals, further steps of preprocessing need to be considered.

In our study, we included an extra step, besides the basic frequency filtering steps proposed in the literature, of a signal decomposition to its independent signal sources, by using independent component analysis (ICA). This way, and by computing some statistical measures as thresholds for component rejection, we managed to isolate the independent noise sources that were present throughout the whole frequency spectrum and reject them upon confirming their noisy nature.

Having artefact-free signal sources, we performed wavelet analysis to extract features both in time and frequency domain, which would serve as classifiers for the supervised classification models. The basic brain rhythm frequency bands were extracted, along with some additional statistical measures, as suggested by the literature.

Two seizure classification models were trained on two-class labeled datasets, containing seizure and non-seizure windows. An SVM and a random forest classifier were cross validated and used for the classification step and the features were finally reduced by performing feature selection to remove the redundant ones. The whole process was repeated in four different trials, where seizure and non-seizure windows of varying length were used to observe the impact of the different window size on our models.

Both classification models were tested on independent datasets and yielded great accuracy scores of more than 82% for SVM and more than 95% for random forest. This thesis contributes two classification models for neonatal seizure detection, as well as six selected features (delta_meanEnergy, gammaLow_meanEnergy, gammaHigh_meanEnergy, Shannon_entropy, Renyi_entropy and Kurtosis) which yielded high accuracy scores.

The importance of thorough artefact rejection is discussed, as well as the differences between the two classification models and the impact of the varying window size on their performance.

Περίληψη

Τα επιληπτικά επεισόδια νεογνών είναι μια κατάσταση που εμφανίζεται στα πρώτα παιδικά χρόνια και ευθύνονται για αρκετούς θανάτους και σοβαρά προβλήματα σε νεογέννητα βρέφη. Παρά τις τελευταίες εξελίξεις στη θεραπεία της πάθησης, το κύριο πρόβλημα που απασχολεί τους γιατρούς είναι η δυσκολία εντοπισμού και χαρακτηρισμού ενός επεισοδίου, καθώς ένα μικρό ποσοστό ανιχνεύεται σε μονάδες εντατικής θεραπείας νεογνών (NICU). Η ανάλυση σήματος EEG πολλαπλών καναλιών είναι η βασικότερη μέθοδος ανίχνευσης επιληπτικών κρίσεων. Ωστόσο, η ερμηνεία τέτοιων σημάτων αποτελεί μεγάλη πρόκληση, καθώς μόνο έμπειροι παιδονευρολόγοι με εξειδίκευση στην ανάλυση ηλεκτροεγκεφαλογραφήματος νεογνών μπορούν να κάνουν αυτήν την εργασία. Οι μέθοδοι μηχανικής μάθησης μπορούν να γίνουν ένα χρήσιμο εργαλείο στην ερμηνεία των σημάτων EEG και ταξινόμηση / πρόβλεψη αν αναφέρονται σε επιληπτικό επεισόδιο.

Υπάρχουν διάφορες μελέτες στη βιβλιογραφία που έχουν επίσης χρησιμοποιήσει μεθόδους μηχανικής μάθησης για την ταξινόμηση των κρίσεων νεογνών. Ωστόσο, ένα σημαντικό βήμα πριν προχωρήσουμε στην ταξινόμηση των επιληπτικών κρίσεων, είναι η απόρριψη των πολλαπλών πηγών θορύβου που υπάρχουν σε ολόκληρο το σήμα EEG. Ειδικά στην ανάλυση EEG νεογνών, όπου υπάρχουν περισσότερες πηγές θορύβου σε σύγκριση με τα σήματα EEG ενηλίκων, πρέπει να ληφθούν υπόψη περαιτέρω βήματα προεπεξεργασίας.

Στη μελέτη μας, συμπεριλάβαμε ένα επιπλέον βήμα, εκτός από τα βασικά βήματα φιλτραρίσματος συχνότητας που προτείνονται στη βιβλιογραφία, μιας ανάλυσης σήματος στις ανεξάρτητες πηγές σήματος του, χρησιμοποιώντας ανάλυση ανεξάρτητων πηγών (ICA). Με αυτόν τον τρόπο, και υπολογίζοντας ορισμένα στατιστικά μέτρα ως κατώφλια για την απόρριψη πηγών θορύβου, καταφέραμε να απομονώσουμε τις ανεξάρτητες πηγές θορύβου που υπήρχαν σε όλο το φάσμα συχνοτήτων και να τις απορρίψουμε επιβεβαιώνοντας τη θορυβώδη φύση τους.

Έχοντας πηγές σήματος χωρίς θόρυβο, πραγματοποιήσαμε ανάλυση wavelets για να εξαγάγουμε χαρακτηριστικά τόσο στο πεδίο του χρόνου όσο και της συχνότητας, τα οποία αξιοποιήθηκαν ως classifiers για τα μοντέλα ταξινόμησης. Εξήχθησαν οι βασικές ζώνες συχνοτήτων του εγκεφαλικού ρυθμού, μαζί με ορισμένα πρόσθετα στατιστικά μέτρα, όπως προτείνεται από τη βιβλιογραφία.

Δύο μοντέλα ταξινόμησης επιληπτικών κρίσεων εκπαιδεύτηκαν σε σύνολα δεδομένων δύο τάξεων, που περιείχαν παράθυρα με επεισόδιο και χωρίς. Ένας SVM και ένας Random Forest αξιολογήθηκαν και χρησιμοποιήθηκαν για το βήμα ταξινόμησης και τα χαρακτηριστικά μειώθηκαν τελικά μέσω του βήματος της επιλογής χαρακτηριστικών (feature selection) για την αφαίρεση των περιττών χαρακτηριστικών. Η όλη διαδικασία επαναλήφθηκε σε τέσσερις διαφορετικές δοκιμές, όπου χρησιμοποιήθηκαν παράθυρα με επεισόδια και μη, σπασίματος διαφορετικού μήκους για να παρατηρηθεί ο αντίκτυπος του διαφορετικού μεγέθους παραθύρου στα μοντέλα μας.

Και τα δύο μοντέλα ταξινόμησης δοκιμάστηκαν σε ανεξάρτητα σύνολα δεδομένων και κατάφεραν να πετύχουν ποσοστά επιτυχίας άνω του 82% για το SVM και άνω του 95% για τον RF. Αυτή η διατριβή συνεισφέρει δύο μοντέλα ταξινόμησης για την ανίχνευση κρίσεων νεογνών, καθώς και έξι επιλεγμένα χαρακτηριστικά (delta_meanEnergy, gammaLow_meanEnergy, gammaHigh_meanEnergy, Shannon_entropy, Renyi_entropy και Kurtosis) τα οποία πέτυχαν υψηλά ποσοστά επιτυχίας.

Σχολίαζεται επίσης η σημασία της επισταμένης απόρριψης ανεξαρτήτων πηγών θορύβου, καθώς και οι διαφορές μεταξύ των δύο μοντέλων ταξινόμησης και ο αντίκτυπος του ποικίλου μεγέθους παραθύρου στην απόδοσή τους.

Table of contents

Table of Contents

List of figures

List of tables

Chapter 1: Introduction

Seizures in childhood is a major unsolved problem which affects a big percentage of neonates born (1,8-3,5/1000 in the United States). It appears that there is a greater risk for seizure during the neonatal period. Despite the existing treatment of phenobarbital, recent studies suggest that we cannot be sure for the use of anticonvulsants in the neonatal period. (Silverstein, 2007)

Although there have been advancements in neonatal seizure treatment, another serious problem is that we can only identify about one third of the occurred seizures, while the other cases remain undetected in Neonatal Intensive Care Units (NICU). (Temko, 2014)

The only available method for neonatal seizure detection that has been used till today, is studying of multichannel EEG signals. The interpretation of these signals cannot be performed by any physician, but only by experienced pediatric neurologists with expertise in neonatal EEG. These experts can annotate the parts of the signal where a seizure occurred, by evaluating the frequency and the amplitude of each channel. (Temko, 2014)

To tackle this problem of limited availability of experts in neonatal EEG interpretation, different methods have been developed. Amplitude integrated EEG (aEEG) is one of these methods, which involves a simpler form of EEG monitoring. It is computed from two EEG channels, each one taken from each hemisphere, and it is a logarithmically scaled, compressed, and temporally smoothed version of the EEG. These EEGs also need to be interpreted by neurologists, possibly with less expertise in neonatal seizure detection. (Temko, 2014)

As an alternative to aEEG, many studies exist on the development of algorithms to automate the detection of neonatal seizures on multichannel EEGs. Various methods have been proposed, but there are some limitations blocking their transition to clinical use. The first limitation is the proof of concept, because it contains specific targeted segments of the EEG signal. In addition, the training and testing data are unrealistic, since they have been carefully selected as subsets of the whole signal, based on their performance, rejecting the bad performing channels or signal segments. Lastly, algorithm provided diagnosis or prognosis is currently unacceptable from the clinical settings. (Temko, 2014)

While these factors act as a barrier for developing algorithms as the only tool for seizure detection and prognosis, they do not prevent them for acting as a useful tool set for neurologists, who can use them as a complementary tool, before they come up with their verdict on seizure characterization.

There are two major approaches in neonatal seizure detection. The first creates a set of heuristic riles and then decides by thresholding from clinical prior knowledge. The resulting features can contribute to a decision which is made by implementing empirically derived thresholds, resulting to binary classification decisions. (Celka, 2002)

The second approach focuses on inductive learning by utilizing statistical classifier methods or model-based parameterization. These methods are used in the stages of feature extraction, in order to determine the features that will be used as classifiers, using a data-driven decision rule. (Aarabi, 2007)

Thesis contribution

As mentioned before, an automated neonatal seizure detection algorithm cannot be solely used as a detection method, but it has been proven to be a useful tool in assisting the clinician with their verdict. Our study contributes two machine learning methods which will be used for neonatal seizure detection, on independent data. Despite the many years of the involvement of machine learning in various fields of medicine, only a small number of algorithms has been developed for seizure detection in neonates. (SR, 2016) This is due to the difficulty of signal acquisition in neonates, since they tend to make a wide range of different movements. Thus, some unusual repetitive spikes in the signal might be misinterpreted as seizure, making it challenging for the clinicians to give an accurate diagnosis. (Minetti, 2020)

Taking into consideration the later, we try to tackle this problem by implementing analytical signal preprocessing before we proceed with the seizure detection algorithms.

In a nutshell, our approach contains the following steps, which are analytically described in Chapter 3. We start by preprocessing the signals, by implementing filtering in time-domain, and using independent component analysis (ICA) for artifact removal. We then proceed with feature extraction, using wavelet analysis (both in time and frequency domain), which results to a set of computed features. These features are used as classifiers in the classification step, where we train an SVM and a Random Forest classifier for the classification of the signals. Lastly, we perform feature selection, using the maximum relevance- minimum redundancy algorithm, to result to a lower dimension feature space with the best scoring features. The models are cross validated and tested on independent datasets.

Thesis overview

Chapter 2 describes the necessary background for this thesis. Signal preprocessing, ICA, statistical metrics, and machine learning methods used for this study are extensively described. The related work is also discussed and how this contributes to the literature. Chapter 3 we describe in detail the research methodology, and we present our work and our models' implementation from a technical point of view, explaining all the steps that were followed during the study. In Chapter 4 we present the results of our proposed

models. Finally, in Chapter 5 we discuss the results of this thesis and in Chapter 6 we suggest some possible future research enhancements and directions.

Chapter 2: State-of-the-art

Preprocessing

One of the greatest challenges in signal processing is the existence of noise and artifacts. These artifacts may include instrumental noise, environmental noise, or powerline noise (introduced by the network). Especially in EEG signals, the challenge is greater, since the signal is also mixed with artifacts from muscle activity, heart activity, eye movement or blinking. During an EEG recording, the patient needs to stay still and try to make the less possible moves in order to minimize the inserted noise. The case is different with neonates, where it becomes a greater challenge to hold the neonate still during the recording. (Cheveigné, 2019)

Thus, a preprocessing step appears to be crucial in order to reduce the added noise from the previous various sources and increase the signal to noise ratio (SNR).

Digital filtering is a widely used preprocessing step when working with EEG data. The typical practice is to first apply low pass filtering to filter out high frequencies that may be correlated with a spike (noise) (usually greater than 50 Hz). Low pass filtering is also applied to remove slow frequencies that contain a great amount of random, such as baseline changes and artifacts. (Awnish Kumar, 2020) (usually less than 1Hz).

For both high pass and low pass filtering, we used zero phase finite impulse response filters.

As a general definition, a filter is a function

A zero-phase filter is a subcategory of linear phase filters, where the phase slope is zero, resulting an even impulse response h(n). $h(n) = h(-n), n \in Z$

Zero phase filtering with IIR filters is achieved with forward-backward filtering, as implemented in Matlab's filtfilt function. The resulting total frequency response is the squared magnitude of the original IIR filter's frequency response. Since the squared magnitude is real-valued, the resulting filter is a zero-phase filter.

A FIR of order N, is described by:

$$y[n] = \sum_{i=0}^{N} bi \;\; x[n-i]$$

Where x[n] is the input signal,

Y[n] is the output signal,

N is the filter order,

b_i is the value of the impulse response at the i'th instant for $0 \leq i \leq N$ of an N^{th} -order FIR filter.

Next, we also have to reject the powerline noise which is inserted by the network. The best practice to remove noise of a specific frequency range, is to use a band reject filter. The powerline interference is analyzed in a narrow band (48-52) of harmonic signals. In order to remove this narrow band of frequencies, we can use a highly selective notch filter, at a specific frequency. (Verma, 2015)

A Notch frequency filter is defined by:

$$H(s) = \frac{s^2 + \omega z^2}{s^2 + \left(\frac{\omega p}{Q}\right) s + \omega p^2}$$

where ωz is zero circular frequency and ωp is the pole circular frequency. Zero frequency is the cutoff frequency and ωp sets the type of the notch filter: standard notch when ωz = ωp, low-pass notch (ωz > ωp) and high-pass notch (ωz < ωp) filters. Q denotes the Q-factor.

Independent Component Analysis (ICA)

Independent component analysis (ICA) is one of the most commonly used techniques for blind source separation. It is a real challenge when trying to separate the useful information of a signal from the various independent noise sources that are introduced. These noise sources can have a great impact on the measured signal. In EEG signals, the main sources of noise are muscle and cardiac activity, respiratory activity, and movement artifacts. However, while we have managed to remove some noise sources from the signal by applying frequency filters and rejecting specific frequency bands, this can not be the case here, as these noise sources are spread throughout the whole frequency spectrum.

Thus, it is crucial to find a technique that can distinct the signal contribution coming from unrelated sources, without rejecting the signal parts containing useful information. This process of separating mixed signals, is known as blind source separation. One of the most common examples to describe the problem of blind source separation, is the cocktail party problem. The later, describes the interference of the voice signals of different people speaking at the same time, while in a cocktail party. The aim of this problem is to extract the original voice of a single person, amongst the mixed signal of different voices and ICA is one

of the most commonly used techniques for dealing with this problem. The problem can also be described by the following figure. (Tharwat, 2018)

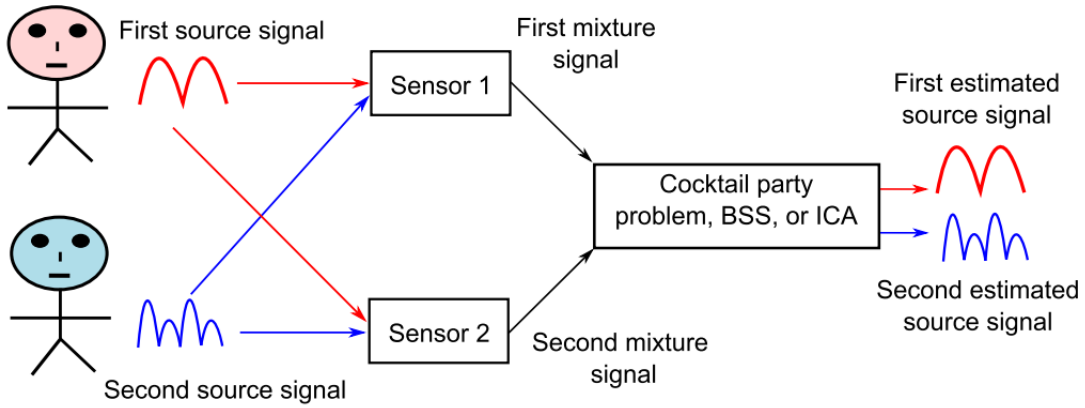

Figure 1: Cocktail Party Problem (Tharwat, 2018)

ICA works by optimizing higher-order statistics (i.e. kurtosis) in order to extract independent signal sources. It uses many algorithms such as Infomax, projection pursuit and FastICA. These algorithms aim on extracting the independent components by :

1. Maximization of the non-gaussianity
2. Minimization of the mutual information
3. Applying the maximum likelihood estimation method

The mixture of two different source signals can be described as follows:

$$X = \begin{pmatrix} x1 \\ x2 \end{pmatrix} = \begin{pmatrix} as1 + bs2 \\ cs1 + ds2 \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} s1 \\ s2 \end{pmatrix} = As$$

Where $X \in R^{n\,x\,N}$ is the defined space from the mixtures and n is the number of mixtures. A is the mixing coefficient matrix, where $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$.

The process of extracting independent signals from the mixture, is by applying the abovementioned algorithms and can be also described by the following figure where the independent source signals are extracted from two mixture signals, by using the unmixing matrix **W**. (Tharwat, 2018)

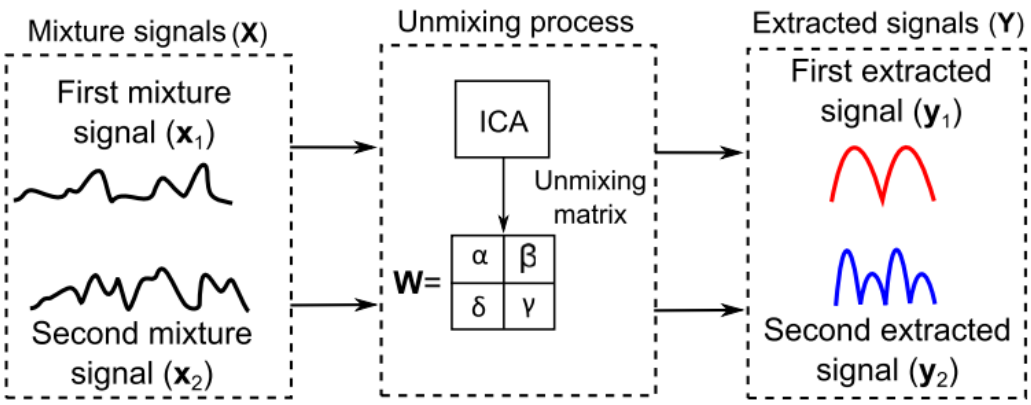

Figure 2 Unmixing Matrix W (Tharwat, 2018)

Feature Extraction

Multichannel EEG signals contain large amounts of data, coming from different channels located throughout the scalp. This makes it impossible to insect and analyze the raw EEG data visually. Thus, there is a high demand to extract relevant information from the signal, which will give us insights for the evaluation and understanding of the cognitive processes. (Amin, 2015)

Extracting useful information from the signal, is a crucial step in order to proceed to the machine learning analysis (classification or regression). The process of feature extraction is the link between raw data and machine learning methods, as it creates the input data (features) which will be used by the ML models. It goes without saying that the impact of this step is crucial to the performance of the classification, since the machine learning models depend strongly on the quality of their input data (garbage in → garbage out). (Amin, 2015). If the computed features are not directly related to the analysis we are interested (in our case seizure classification), the performance results of the classification step will not be satisfactory, as they may yield poor classification results, due to the lack of strongly related features with the study (Amin, 2015).

Different approaches exist for feature extraction in EEG signals, with the top performing being time domain, frequency domain and wavelet transform analysis. From these three, wavelet transform features appear to be more effective, since they tackle better the non-stationarity of EEG signals, than the compared methods.

The discrete wavelet transform (DWT), is the most commonly used method when analyzing EEG signals, due to its non-stationary characteristics. The DWT uses short duration windows for high frequencies and longer duration windows for lower frequencies, which results to a good time-frequency analysis. It uses successive high pass and low pass filters of the input time series, followed by two down samplers by a factor of 2. The mother wavelet

corresponds to the high pass filter g(n), while the low pass filter h(n) is its mirror version (Amin, 2015).

There is a variety of mother wavelets which can be employed, with some of the most common being Daubechies wavelet (db4), Morlet, Mayer, Mexicanhat, Gaussian and Haar.

In our study we chose to work with the Morlet wavelet, as proposed by the bibliography for EEG analysis. The Morlet wavelet is a complex exponential which is tuned by a Gaussian function which depends on a parameter called "number of oscillations" and is user decided. This parameter is determined by the standard deviations σ_t and σ_f of the time and frequency resolutions respectively. (D'Avanzo, 2009). This wavelet tuneable parameter is tuned for each frequency band, in order to achieve better time-frequency resolution. The Morlet wavelet can be described by the following equation:

$$C(\alpha,\tau) = < \chi, \psi_{\alpha,\tau} > = \int_R x(t)\psi *_{a,t} (t)dt$$

Where ψ(t) is the mother wavelet, a is the scale, C(α,τ) is the wavelet coefficient and * is the conjugate complex operator (D'Avanzo, 2009).

We considered CWT analysis on all main EEG frequency bands (delta, theta, alpha1, alpha2, beta, gamma low, gamma high). On each frequency band, we calculated the mean energy and the standard deviation of the wavelet.

We also added the Shannon entropy, renyi entropy and kurtosis to our features.

Shannon entropy describes the distribution of signal components and is a widely used feature in EEG analysis. Shannon wavelet entropy is the Shannon entropy applied on the calculated wavelets and describes the signal variation on different frequency scales. Shannon entropy is used to extract the periodicity in the signal, and it has been applied for the detection of weak signals in the past. (Ling, 2007)

Shannon wavelet entropy is described by:

$$s^s = -\sum_j p_j \, log p_j$$

Renyi Entropy mainly serves as an index of diversity. It is an automorphic function and it can describe the degree of randomness in a given signal. (Liang, 2015) It is described by:

$$S_a^{(R)} = \frac{1}{1-a}\log[\sum_j (p_j)^a]$$

Kurtosis is the last computed feature that we have included in our study. Kurtosis is a metric that can measure the peakyness of the signal. Kurtosis has been related with specific activity distributions in EEG signals. It is positive for peaked activity distributions (i.e. eye blink, movement, muscle and cardiac activity) and it is negative for flat activity distributions (i.e. flat added noise). (Giuseppina Inuso, 2007) Kurtosis can be described by:

$$k = m_4 - 3m_2^2$$

$$m_n = E\{(x - m_1)^n\}$$

Where m_1 is the mean and m_n is the n-order central moment of the variable (Giuseppina Inuso, 2007).

Classification methods

Supervised classification is the process of developing models in order to classify the input data to labeled classes. Binary classification is a special case where only two classes exist, and the classification problem is transposed to two-class decision. This is the case in our study, where we only have two output classes that describe our model's decision, seizure and non-seizure. The trained models undergo a training process, where they are trained based on the features of the training dataset and then their accuracy is evaluated against a testing dataset. In order to evaluate the performance of the trained models, we also compute some additional metric which give further information about the quality of our models. The most commonly used metrics are ROC curve, sensitivity (true positive rate), specificity (true negative rate) and mean square error (MSE). We included specificity and sensitivity in our study, as they give us enough information for the evaluation of our models.

- Sensitivity, or true positive rate, is defined as the ratio of positive samples that are correctly predicted as positive, with respect to all positive data samples.

$$Sensitivity = \frac{True\ positive}{True\ positive + False\ negative}$$

- Specificity, or true negative rate, is defined as the ratio of negative samples that are correctly predicted as negative, with respect to all negative data samples.

$$Specificity = \frac{True\ negative}{False\ positive + True\ negative}$$

SVM

Support vector machines (SVMs) is the first classification method we used. Support vector machine is a powerful tool used for two-class classification and it targeted to be used as a non-linear mapping of the input vectors into a high-dimensional feature space. It relies on the idea of finding the maximum geometric margin between the two classes. One of the simplest types of support vector machines is linear classification, which attempts to set

a straight-line separating data with two dimensions. A linear classifier is also referred to as hyperplane. Various hyperplanes achieve the same target, to separate the two-class data, but only one can achieve the maximum separation. (Pirooznia, 2008)

The basic principle of the learning procedure in SVM is to find a hyperplane which will separate the data into two classes, and then try to maximize the margin between the two classes and the separating hyperplane, whilst ensuring the accuracy of correct classification. The final binary classifier that is produced, is called optimal separating hyperplane. It does not suffer from local optima problem, i.e., it works without a convex optimization problem. (Rabia Musheer, 2015)

In case of linearly separable data, the principle of SVM is described as follows. The main goal of the training phase is to find the linear function :

$$f(x) = W^T X + b$$

which will be the plane that will divide the data and the space to two different classes according to the condition:

$$W^T X + b > 0$$
$$W^T X + b < 0$$

These functions define the separating plane, and the distance between the two parallel hyperplane equals to: $\frac{2}{|w||^2}$. This quantity is referred to as the classification margin, as described in figure 3.

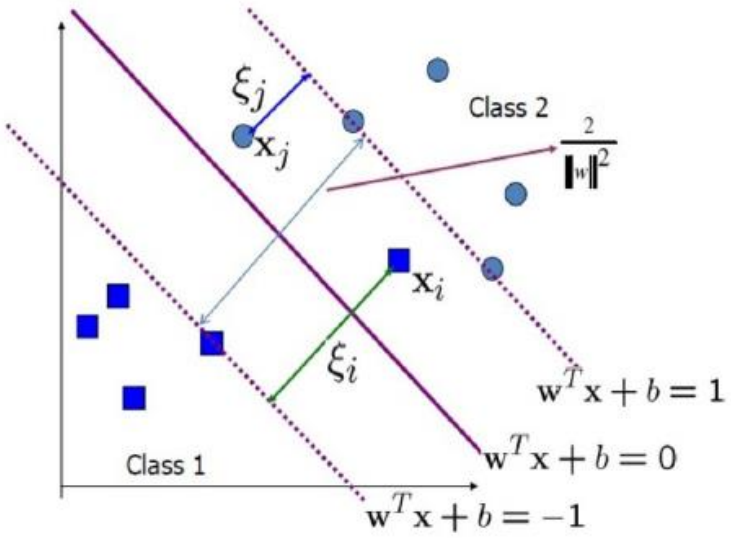

Figure 3: SVM Hyperplane definition

In order to maximize the classification margin, the algorithm is required to solved the following optimization problem:

- Minimize $\frac{1}{2\,|w||^2}$
- Subject to $Y_i(W^T X_i + b) \geq 1$

In case of non-linearly separable data, SVM will have to work with more than two dimensions, and therefore will have to map the data from the input space into a high-dimensional feature space. The classes will then be separated by an optimal hyperplane. (Rabia Musheer, 2015) In order to perform this mapping, we will use a function called a kernel function. While several kernel functions exist, we decided to work with the polynomial, since it yielded the best results. The polynomial kernel is defined as:

$$K(x_i, x_j) = (\gamma x_i^T, x_j + r)^d, \gamma > 0$$

For non-linearly separable data, SVM requires the solution of the following optimization problem:

- minimize $\frac{1}{2\,|w||^2} + C\sum_{i=1}^{n}\xi_i$
- Subject to $Y_i(W^T X_i + b) \geq 1 - \xi_i$
- $\xi_\iota \geq 0$

The kernel's goal is to minimize the distance of each sample xi from its center, which is achieved by calculating the value weights in each run. The successfulness of SVM strongly depends on the choice of the kernel function K, and of course the hyper parameters therefore in order to adjust optimally these parameters we should perform a cross-validation procedure. (Zekic-Sušac, 2014)

Random Forest

Random forest is the second machine learning method we used for classification of the two-class data. Random forest was firstly developed in 2001 and it has proven to be a powerful tool with great accuracy scores in classification tasks. Random forest works by utilizing an ensemble of classification trees. Input data are bootstrapped and are built into trees, which may be built by bagging or random variable selection. A variable candidate set is selected randomly at each split, from the whole input dataset. Each tree is grown on different random subsamples, ensuring the randomness of the method. In order to have a low bias, all trees are fully grown. Low correlation between individual trees is also ensured by both bagging and random variable selection. An ensemble forest is produced by averaging the large ensemble of high-variance, low-bias and low correlation trees. Thus, the problem of overfitting is tackled.

The algorithm of random forest can be described by the following steps:

Input:

T: Training set $(x1, y1), (x2, y2) \ldots (xn, yn)$

N_{tree} : the number of the built trees

M_{try} : the number of chosen variables to split at each node

Training:

for each k in N_{tree} :

create a bootstrap sample from the input training dataset T.

at each node of the tree, pick M_{try} random variables and decide the best split between these M_{try} variables

create an unpruned tree from these bootstrapped samples

end

Classification:

Use majority voting for classification among the N trees

Compute

$$f_{avg}(X) \coloneqq \big(p_1(X), \dots, p_k(X)\big) \coloneqq \frac{1}{N}\sum_{b=1}^{N} f_i(X)$$

$$f_{RF}(X) \coloneqq argmax_k\{p_1(X), \dots, p_k(X)\}$$

An important step which plays a crucial role in achieving the desired performance, is parameter optimization in the random forest model. Two major parameters need to be decided during the training of the model. The first is the number of trees that will grow on each forest (N_{tree}). The second is the number of variables that will be tried on each split (M_{try}). The later is a real parameter, meaning that its optimal value is strongly dependent on the input data. The square root of the number of variables is often chosen as the default value for M_{try}. For larger number of candidate classifiers M_{try}, it is desired to have a large number of trees in the forest N_{tree}, in order to give enough chances for each classifier to be selected. A common approach on deciding a good value of N_{tree} is gradually increase the value of N_{tree} and stop when the prediction error becomes stable. (Chen, 2013)

Cross Validation

In order to validate our classification methods described earlier, it is necessary to have some test datasets, independent from the training datasets, that will be used to measure the classification error. However, since our datasets our significantly limited and hard to find, it is difficult to obtain independent datasets for testing, or weaken our training datasets by keeping out some samples for testing. A technique that will give a solution to this problem is K-fold cross validation. K-Fold cross validation also prevents the problem of overfitting in our dataset, which happens when the classifiers are computed multiple times from the same samples. (Pirooznia, 2008)

The concept behind cross-validation is the same as with a single holdout validation set, to estimate the model's predictive ability and performance on unseen data. Its basic principle is that it repeats the experiments multiple times by dividing the training dataset in "V" different parts every time, keeping one of them out for validation and using the others for learning. It does not require separate test datasets, and it also does not reduce the training dataset. The training dataset is partitioned into "V" smaller datasets, called "folds". The default number of "V" is 10. In each repetition, 1 subset is kept out for testing and the

remaining "V-1" are used for training. This procedure is repeated "V" times, resulting to a bigger test dataset and taking advantage of the full spectrum of the training dataset. It is worth mentioning that cross validation does not prevent overfitting in itself, but it may help in identifying a case of overfitting caused by the classification method. (Pirooznia, 2008)

Feature Selection

Feature selection is the process of sub-setting a dataset with relevant and redundant features, in order to improve the performance of the classification methods, regarding accuracy and time to construct the model. (Aziz, 2017) It differs from the feature extraction process, as it selects a subset from already selected features, thus avoiding the drawback of the output interpretability. The feature selection methods are classified as filters, wrappers and embedded, depending on the methods used to evaluate the feature subsets. (Rabia Musheer, 2015)

Filter methods are widely used on gene ranking, as they have computational efficiency. They select the best subset by variable ordering, using variable ranking methods, implementing heuristic methods. They also use a ranking criterion of statistics, in order to score the variables and define a threshold value, discarding the variables under it. Their main drawback is that they are independent of the specific required prediction task. That means that they will select the features even if the latter don't fit in the classification model, thus making them unreliable. One of the most commonly used filter feature selection method is Maximum Relevance Minimum Redundancy (mRMR). (Rabia Musheer, 2015)

Wrapper methods on the other hand, don't use feature relevant criteria like the filter methods. Instead, they depend on the performance of classifiers to obtain a feature subset. They use the predictive accuracy of a data mining method, to determine the fitness of a selected subset, by integrating the data mining method as a black box. The aim of this method is to find the subset with the maximum evaluation, by following a trial and error method. This approach forces the method to execute cross validation on small datasets in order to find the most accurate estimation, resulting in better overall performance. (Rabia

Musheer, 2015) On the downside, wrapper methods are very expensive regarding time and computations, when implemented on high dimensional feature space. (Pirooznia, 2008)

The embedded methods were inspired as an attempt to combine the advantages of both filter and wrapper methods. Unlike the two previous ones, which separate the feature selection and training process, the embedded methods integrate the feature selection methods into the construction process of the classifier or regression model. (Rabia Musheer, 2015) More specifically, embedded methods incorporate the feature selection as a part of the training process, while significantly reducing the computational time. The consider both relations between input and output features, and also search for features which allow better local discrimination. They use the independent statistical criteria used by filter methods, in order to obtain the optimal subsets of a known group of classifiers. After that, the classification method is used to select the optimal subset among the group of optimal subsets produced by the previous step. They can be categorized into three sub methods, namely pruning method, built-in mechanism, and regularization models. In the pruning method, all features are included in the training process initially, and then the ones with the smaller correlation coefficient values are recursively removed (pruned), using an SVM algorithm. In the built-in mechanism method, the features are selected by some supervised learning algorithms, in the training phase, while in the regularization method, the objective functions are used to minimize fitting errors and near zero regression coefficient features are eliminated. Various feature selection techniques are suggested in the literature. LLDA based Recursive Feature Elimination (LLDA-RFE), kernel-penalized SVM (KP-SVM), discriminative least squares regression (LSR), Support Vector Data Description (SVDD) and Support Vector Machine - Recursive Feature Elimination (SVM-RFE) are some of the most significant ones. Feature selection methods are widely used in EEG signal analysis due to their conceptual simplicity. However, as every algorithm, they come with some drawbacks. During the feature selection process, many useful features may be rejected, thus resulting in loss of useful information, while correlations between variables are not taken into consideration. These problems can be overcome by selecting the optimal subsets according to a quality criterion instead of filtering out the redundant features. However, these methods will not perform as well on independent testing datasets, since they suffer from overfitting, and also implement some computational heavy algorithms, which are difficult to integrate and interpret. (Aziz, 2017)

Minimum redundancy maximum relevance (mRMR)

The Minimum redundancy maximum relevance algorithm is amongst the best feature selection methods for dimensionality reduction, due to its high accuracy. It was firstly introduced when trying to tackle the problem of high dimensionality of DNA microarray data, where there is a high number of features and a small number of samples. However,

the number of features have a strong impact on its computational cost. It comes to tackle the problem of machine learning methods poor performance, when faced with many irrelevant and redundant features. During the previous years, the mRMR method has gained great popularity, despite its computational cost, mainly because of its high accuracy. (Ram´ırez-Gallego, 2016).

The mRMR method works by scaling quadratically based on the number of features and growing linearly depending on the sample size. It is worth mentioning that the mRMR has been accused of not enclosing conditional redundancy in its computations. However, it has been proven that the mRMR offers a great trade-off between stability and accuracy. (Brown, 2012)

mRMR ranks the features by their importance for a given classification task. The relevance of the features to the target is evaluated, and a penalty is given for each redundant feature. Its main goal is to find the maximum relevance between a class *c* and a feature set *X*, using their mutual information, which is defined as:

$$\mathrm{I(A;B)} = \sum_{b\in B}\sum_{a\in A} p(a,b)\log\left(\frac{p(a,b)}{p(a)p(b)}\right)$$

Where p(a) and p(b) are the marginal probabilities and p(a, b) is the joint probability between these two features.

However, applying the mRMR in high dimensional spaces is not an easy task, where the samples may be insufficient, and the computational cost may be high. A way to solve this problem is by applying the maximum relevance criterion, which searches for the features that satisfy the following equation:

$$\max D(\boldsymbol{X}, c); D = \frac{1}{|X|}\sum_{Xi\in X} I\,(Xi\,;c)$$

If we only apply the maximum relevance criterion, this will result in a large amount of redundancy in our feature set. Thus, the criterion of minimum redundancy also needs to be applied, which is defined as:

$$\min(R(X); R = \frac{1}{|X|}\sum_{Xi,Xj\in X} I\,(Xi, Xj)$$

The combination of these criteria leads to the mRMR algorithm, which is essentially a greedy algorithm, described in the following equation:

$$max_{Xi\in S}\left[I(Xi;c) - \frac{1}{|S|}\sum_{Xj\in S} I\,(Xi;Xj)\right]$$

Where S is the set of the selected features. (Ram´ırez-Gallego, 2016)

Related Work

Seizures in neonates can have serious consequences in the patients. More specifically, they may lead to brain injury, deterioration of respiratory or circulatory systems, and in severe cases they may result in the death of the patient. (Açikoğlu, 2019) Therefore, the early diagnosis and prognosis of neonatal seizures is of crucial importance, in order to improve prognosis and long-term impact on the patients.

In this context, many studies have been implemented which employ machine learning methods, as an assistance to the physician for the improvement of accurate seizure diagnosis and prognosis.

It is well established that the gold standard for detecting neonatal seizures is through the analysis of EEG signals. (Açikoğlu, 2019) All the studies that we are referencing, have worked with EEG signals from neonates.

Moreover, several studies have shown that the use of support vector machines (SVMs) in neonatal seizure classification tasks, yields great results. (Açikoğlu, 2019) uses SVM and KNN (k nearest neighbors) for seizure classification, with accuracy scores over 95%. (Raghu S, 2018) also used SVM (amongst other algorithms) and showed that the SVM outperformed the other models. (Siddiqui1, 2020) used SVMs for seizure classification and scored accuracy over 95%. (Tanveer, 2021) also used SVMs, managing to outperform several older studies in seizure classification accuracy. Finally, Temko et al, in both their studies (Temko, 2011) (Temko, 2016) used SVM classifiers and achieved high classification scores.

The use of random forest is also common when dealing with neonatal seizure classification tasks. Ensemble trees have been employed by (Siddiqui1, 2020), managing to outperform other classification algorithms. (Tanveer, 2021) also used random forest classifiers for training and evaluation and achieved high accuracy scores as well. (Chen, 2013) also used an ensemble classifier for neonatal seizure classification, which yielded an accuracy score over 92%.

It goes without saying that all the above-mentioned studies have used cross validation during the training process, in order to avoid overfitting. Again, the gold standard is 10-fold cross validation.

Feature selection techniques have also been suggested in the literature, as the demand to reduce the irrelevant features appears to be crucial. Many algorithms have been employed for the feature selection step, with the most commonly used being Neighborhood component analysis (NCA), Infinite Latent Feature Selection (ILFS), Feature Selection via Concave Minimization (FSV), Laplacian Score (LS), Multi-Cluster Feature Selection (MCFS), Correlation-based Feature Selection (CFS), Unsupervised Feature Selection with Ordinal Locality (UFSOL), Least Absolute Shrinkage and Selection Operator (LASSO) and Minimum Redundancy Maximum Relevance (MRMR). (Açikoğlu, 2019) (Siddiqui1, 2020) (Temko, 2016) also mentions Principal Component Analysis (PCA) as a good method for feature selection in seizure classification.

In our study, we decided to use the MRMR feature selection algorithm, as suggested by (Mohammad Reza Mohammadi, 2016). MRMR is a two-step algorithm, where a set of candidate features is selected in the first step and a compact subset of features is selected in the second step. As suggested by the literature, the MRMR appears to select the most relevant features with great accuracy, as the selected features yielded classification performance of 92% (Mohammad Reza Mohammadi, 2016).

Based on these previous studies, we decided the algorithms we are going to use for the classification and feature selection steps. However, we spotted that the most challenging part in order to achieve good classification results, is feature extraction.

While several papers exist on classification and feature selection, only a small number of studies exists on the steps before classification (data preprocessing and feature extraction).

Our work was mainly focused on signal preprocessing of the input EEG signals and several aspects were considered. The main goal was to properly filter out the artifacts and the added noise on the signal, an important step that has a great impact on the classification accuracy. This thesis contributes an approach of signal preprocessing steps, each one corresponding to different types of noise, as an important step before feature extraction. Independent component analysis is also employed to help with artifact rejection, by evaluating selected statistical metrics on each component. Since the EEG signals from neonates are quite more challenging compared to adult EEG signals, we have focused our study on preprocessing the signals optimally and then extracting features by using wavelet analysis.

Our thesis comes to contribute to the literature a pipeline of preprocessing steps, artifact rejection by ICA, feature extraction by wavelet analysis and finally two classification models for neonatal seizures, including feature selection.

Chapter 3: Research methodology

The aim of this thesis is to propose some machine learning methods for the classification of seizures in neonatal EEG recordings.

The steps that were followed are briefly described in Figure 4. Each step is described in detail, in their dedicated sections. The whole process of signal analysis and machine learning was implemented in Matlab.

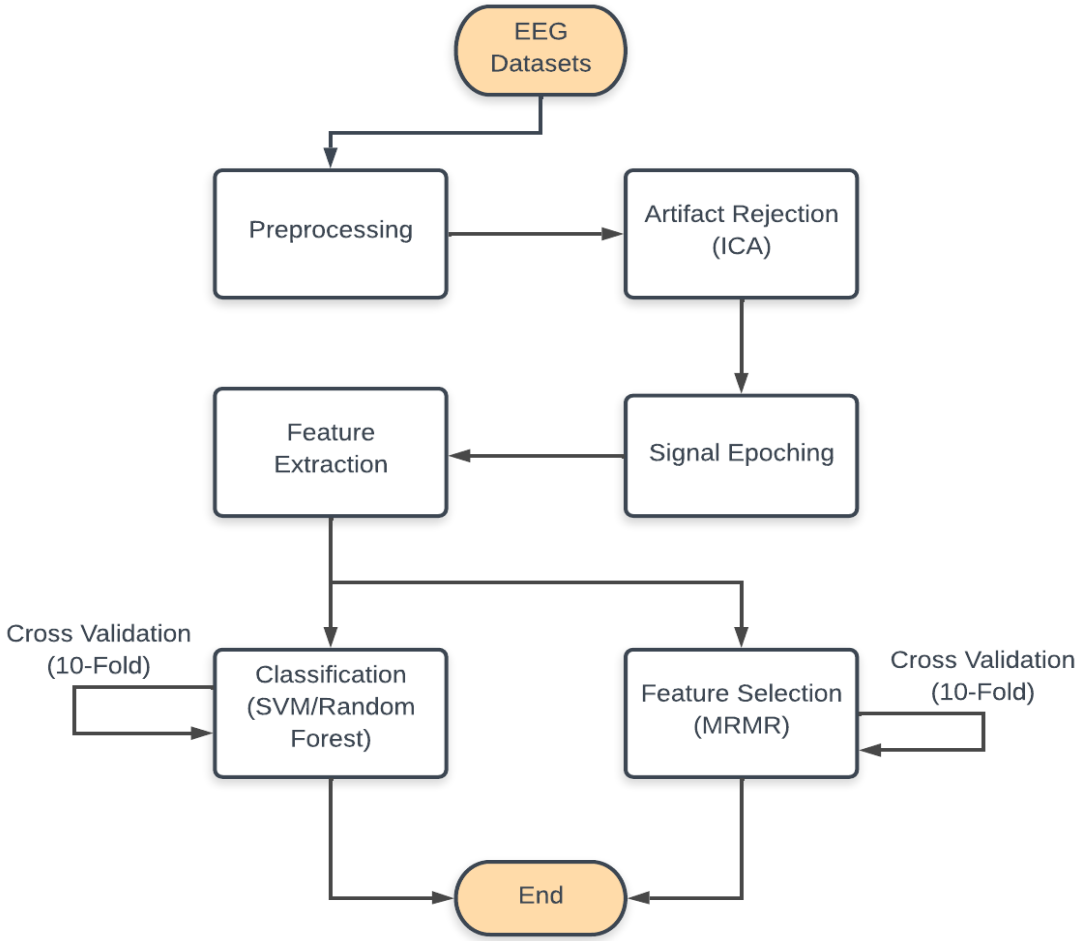

Figure 4 : Methodology Pipeline

Signal Preprocessing

For our study, we used Multi-channel EEG signals, recorded from 79 term neonates admitted to the NICU at the Helsinki University Hospital. The median recording duration was 74 min (IQR: 64 to 96 min). The EEGs were annotated for the presence of seizures by three experts. An average of 460 seizures were annotated per expert in the dataset, while 22 neonates were seizure free by consensus. (Stevenson, 2019)

Since the datasets are raw signal data, containing various sources of noise, an initial step of preprocessing appears to be crucial. Multi-channel EEG signals mainly suffer from artifacts which are caused by instrument noise, powerline frequency, muscle activity, cardiac and respiratory activity, patient movement etc. All these noise sources need to be extracted and removed, so that the actual signal containing useful information can be studied.

The abovementioned common EEG artifacts have been encountered in many previous studies and have known statistical characteristics. Thus, we proceed frequency filtering (high pass, low pass and notch filtering) as proposed by the literature, to reject common artifact sources and increase SNR. We also apply independent component analysis in order to isolate and reject the noise sources that are entangled in the signal frequency spectrum.

Channel Locations

As a preliminary step, we need to import the channel locations onto the signals, in order to be able to visualize the channels and extract useful information about them.

The EEG signals were recorded using a NicOne EEG amplifier with sampling frequency of 256 Hz and EEG caps with 19 electrodes. The international 10-20 standard was employed for the positioning of the channels, including a recording reference at midline. The standard longitudinal bipolar layout (a.k.a. 'double banana') was used to generate a bipolar montage for annotation:

standard longitudinal bipolar layout (a.k.a. 'double banana'): Fp2-F4, F4-C4, C4-P4, P4-O2, Fp1-F3, F3-C3, C3-P3, P3-O1, Fp2-F8, F8-T4, T4-T6, T6-O2, Fp1-F7, F7-T3, T3-T5, T5-O1, Fz-Cz, Cz-Pz (Figure 2)

The two extra channels *ECG EKG* (heart rate monitor) and *Resp Effort* (respiratory effort) were removed, resulting in datasets of 19 channels. Common Average Referencing was also added for each channel.

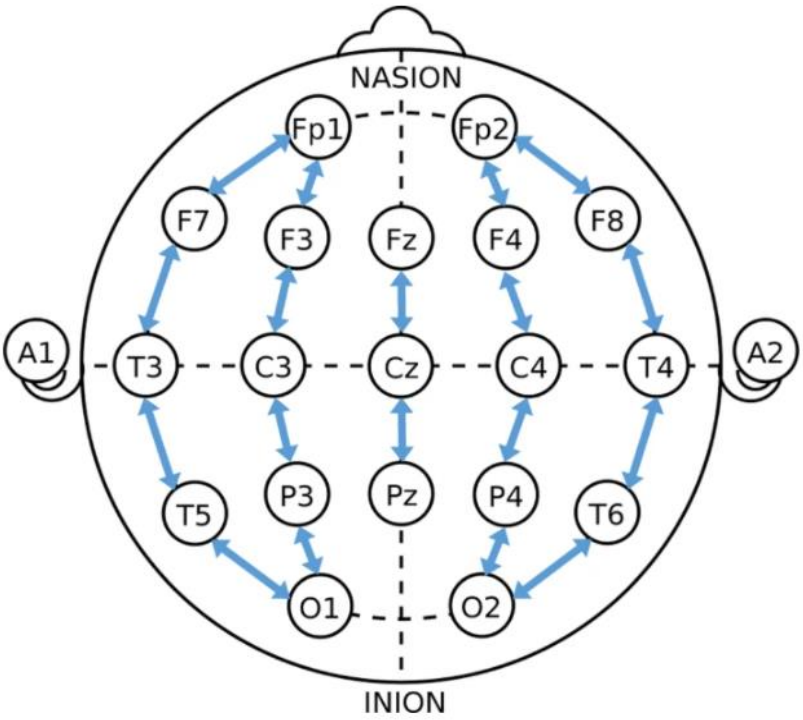

Figure 5 : Channel Locations

High Pass Filtering

As a first preprocessing step, high pass filtering was applied on the signals.

A great amount of random noise is contained in low frequencies, such as baseline changes and artifacts. (Awnish Kumar, 2020)

The purpose of high pass filtering is to suppress the low frequency interference and increase the SNR. It also removes linear trends, slow and possibly large amplitude drifts and contributes to obtaining good quality ICA decompositions. A zero phase FIR filter was applied with a low frequency cutoff at 0.5 Hz. The output signals were cleansed from low-frequency drifts, resulting in artifact-free signals.

A Finite Impulse Response (FIR) Filter is employed to perform high pass filtering. Its main characteristic is that is only uses the delayed version of the input signal x(t) to filter the signal, without taking into consideration the previous output values (requiring no feedback).

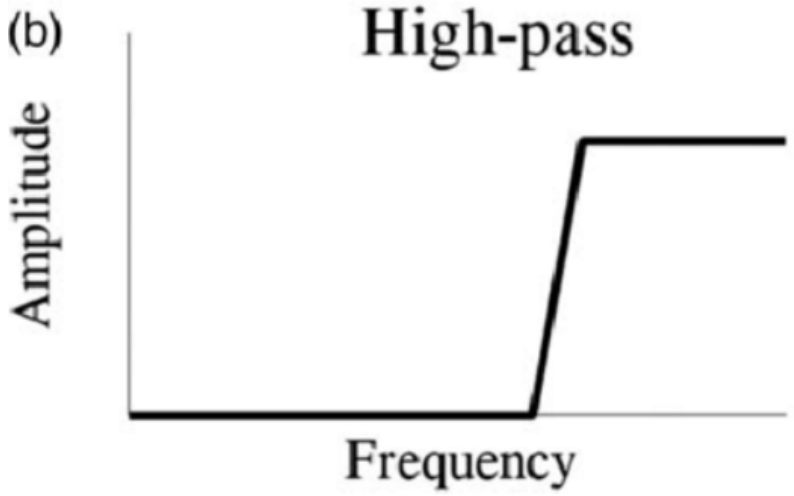

Figure 6: High Pass Filtering

A zero phase filter is a subcategory of linear phase filters, where the phase slope is zero, resulting an even impulse response h(n). $h(n) = h(-n), n \in Z$

A FIR of order N, is described by:

$$y[n] = \sum_{i=0}^{N} bi \;\; x[n-i]$$

Where x[n] is the input signal,

Y[n] is the output signal,

N is the filter order,

b_i is the value of the impulse response at the i'th instant for $0 \leq i \leq N$ of an N^{th} -order FIR filter.

Low Pass Filtering

Artifacts can also be found in higher frequencies where brain activity interferes with other body activities, for example muscle activity (20-300 Hz). This activity introduces the so-called muscle artifacts in our signals, which needs to be removed. (Muthukumaraswamy, 2013)

In general, we want to examine only the bands of our signal where the gamma band ends, in order to avoid all other types of high frequency oscillations that only contribute artifacts to our signals, rather than useful information.

Likewise high pass filtering, a Finite Impulse Response (FIR) Filter is employed to perform low pass filtering. A zero phase FIR filter was applied with a high frequency cutoff at 70 Hz.

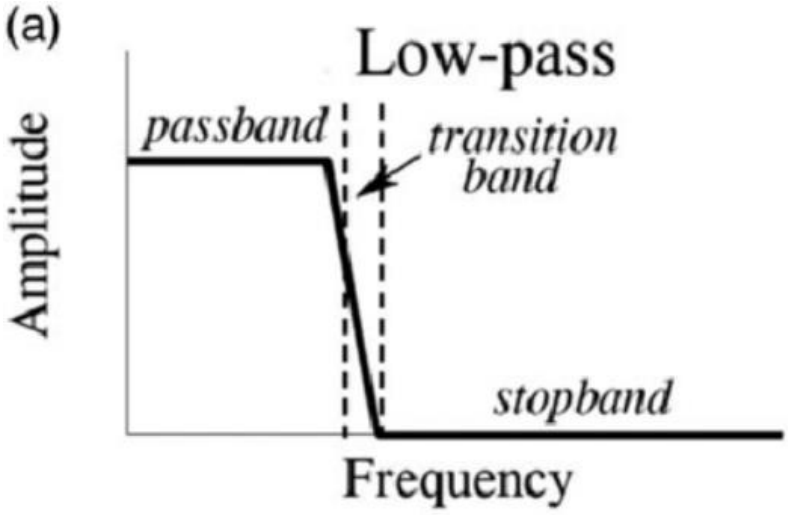

Figure 7: Low Pass Filtering

Notch Filtering

Finally, we need to reject the noise added by the powerline. In Europe, the powerline frequency is at ~50 Hz, while in the US at ~60 Hz. This powerline noise is added to the monitoring equipment and is also recorded onto the signal, resulting in artifact.

One of the most commonly used band reject filters is the Notch Filter. In a Notch Filter, we need to define a low and a high frequency, which refer to the limits of the band to be rejected. In our study we define F_L = 48 Hz and F_H = 52 Hz, in order to ensure that the powerline noise is attenuated.

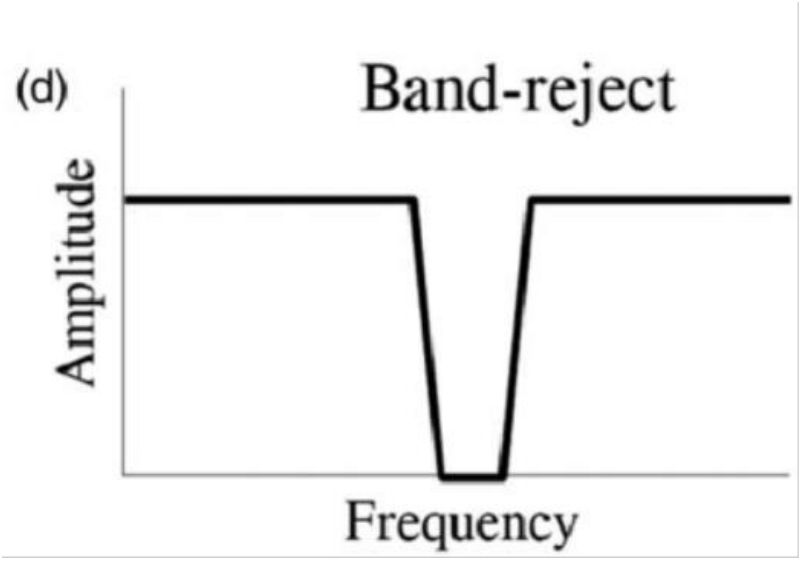

Figure 8: Notch Filtering

In figure 9 and figure 10 we can see the signal before and after applying notch filtering, respectively. We can observe the attenuation around 50 Hz, which was the desired result.

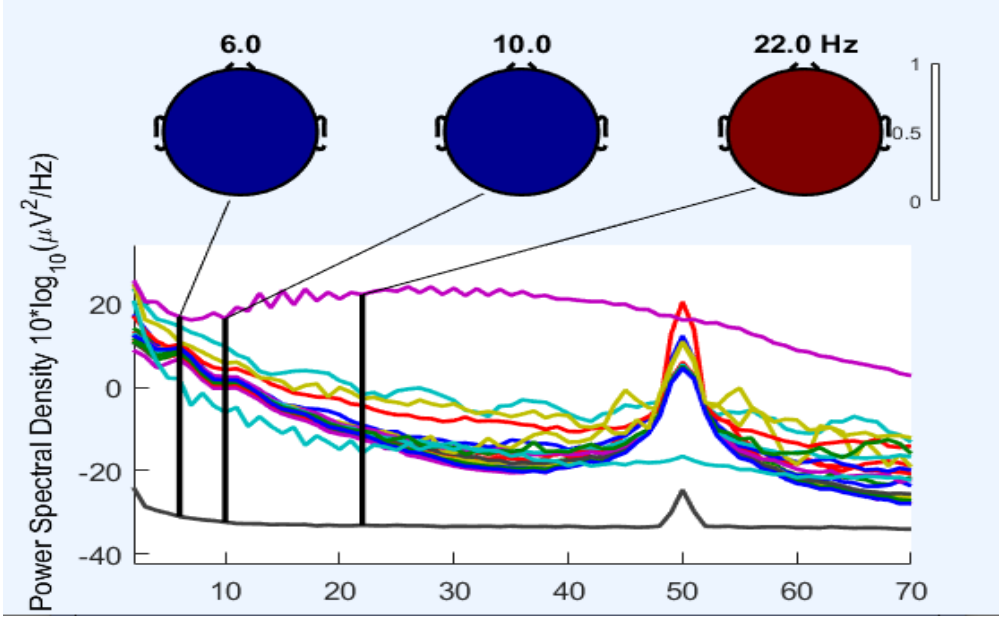

Figure 9: Before applying Notch Filtering

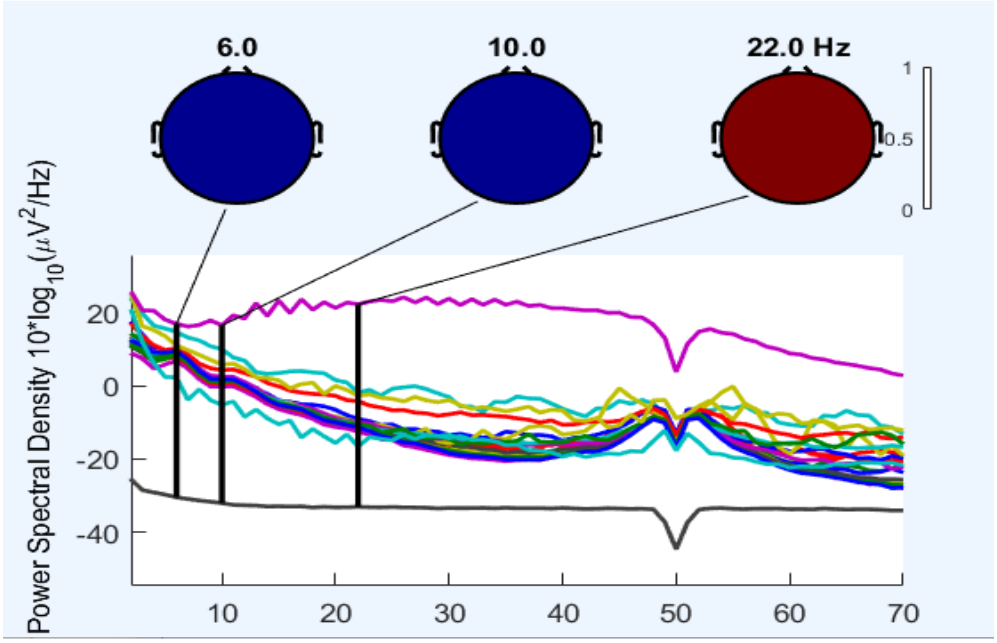

Figure 10: After applying Notch Filtering

Artifact Removal

While the initial steps of filtering manage to remove some artifacts based on frequency, they are not enough.

Artifacts can appear into EEG signals from various sources. It is a fact that EEG signals contain a mixture of brain and non-brain activity, with the latest contributing irrelevant signal information (artifact) which needs to be removed. Common artifacts are pulse, cardiac, sweat, eye movement, respiratory and muscle and movement artifacts. (Britton JW, 2016)

In order to identify and reject the noise coming from different sources, we used one of the most commonly used decomposition techniques, Independent Component Analysis.

ICA

Independent Component Analysis (ICA) is a linear decomposition technique, aiming to reveal the underlying independent statistical sources of mixed signals. It can help on multi-channel EEGs, where the detection of different signal sources cannot be performed on raw data level, even when applying common statistical techniques.

Amongst the various algorithms that exist for the application of ICA, we decided to use Infomax ICA, as it is one of the most commonly used ones with great discrimination ability between independent components. Infomax ICA aims on finding the independent components by maximizing entropy. That means that the algorithm tries to minimize the number of mutual information between two observations X and Y, thus searching for the observations (components) which are maximally independent.

We applied Infomax ICA on all raw signals. Each signal was analyzed in 19 independent components, as the number of different channels. ICA was applied on the full signal duration. The ICA was a time-consuming process, since it attempts to find the best split of components throughout the whole signal duration. The generated components are maximally independent and they can reveal different noise sources, which will be characterized as artifacts based on statistical criteria, and then rejected.

In Figure 7 we can visualize the independent components generated from the analysis of dataset 1, and the different parts of the brain that they are correlated to. Matlab matches the analyzed components with a signal activation map by colorizing the component activations on different brain regions.

An unmixing matrix is also generated, which is the inverse of the mixing matrix and can be used to recover the original signal sources from the preserved components.

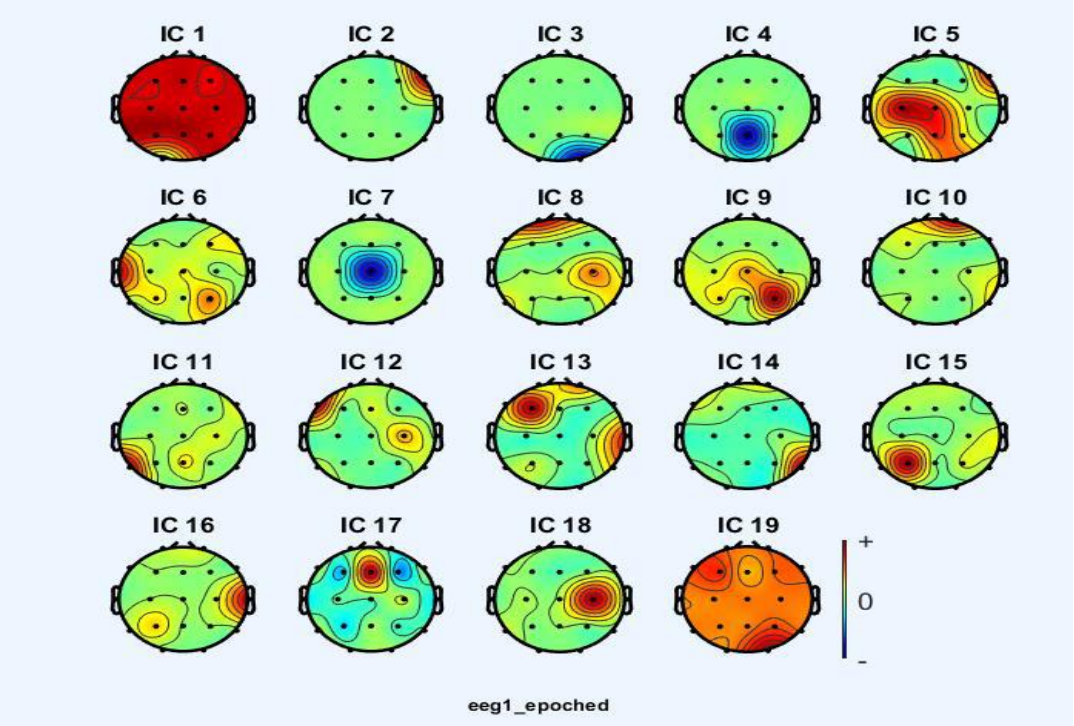

Figure 11: Visualization of Component Activations

Global Metrics

Having extracted the independent components of each signal, we then need to decide which ones are considered as artifacts. We could rely on the graphical representation of Figure 7, as estimated by eeglab, but we saw that these estimations are not always accurate, resulting to loss of useful information.

In order to decide which components will be considered as artifacts, we calculated some metrics on each component, along the whole signal duration (global approach).

More specifically, we calculated the kurtosis, Shannon entropy and Renyi entropy for each component.

Kurtosis has been used in different studies for the identification of different artifacts, including the cardiac artifact (CA) and ocular artifact (OA). Kurtosis is defined as

$$kurtosis = \frac{m_4}{(m_2)^2} - 3,$$

Where $m_n = E\{(x - E\{x\})^n\}$.

Shannon Entropy and Renyi entropy, are two metrics that have also been used in artifact detection algorithms. Shannon Entropy is defined as

$$H_{sh} = -\sum_{i} p_i(x)\log[p_i(x)]$$

Renyi Entropy of order α, where $\alpha \geq 0$ and $\alpha \neq 1$, is defined as

$$H_\alpha(X) = \frac{1}{1-\alpha}\log\left(\sum_{i=1}^{n} p_i^\alpha\right)$$

Entropy is a measure of disorder, randomness, or uncertainty. Higher entropy values mean more irregular signals, which can be translated into bigger drifts in a component's values, thus considering it an artifact. (L. Lee, 2003)

We then defined a threshold for each metric, which was given by Chevyshev's inequality. The latest guarantees that for a class of unknow probability distributions, only a certain fraction of values can be greater than a certain distance from the mean. It is defined as

$$\Pr(|x-\mu| \geq k\sigma) \leq \frac{1}{k^2}$$

After calculating the threshold for each metric, which gave a list of components over each threshold, we took the union of the three and resulted with the final list of components that we want to reject:

$$CP_{artifact} = Kur \cup H_{sh} \cup H_{RE}$$

The same process was repeated on all datasets. An average of 69% of the components was kept, while the remaining 31% was considered as artifact and was removed.

In Figure 7 we present the calculated metrics thresholds and the components that exceed them, for dataset 1.

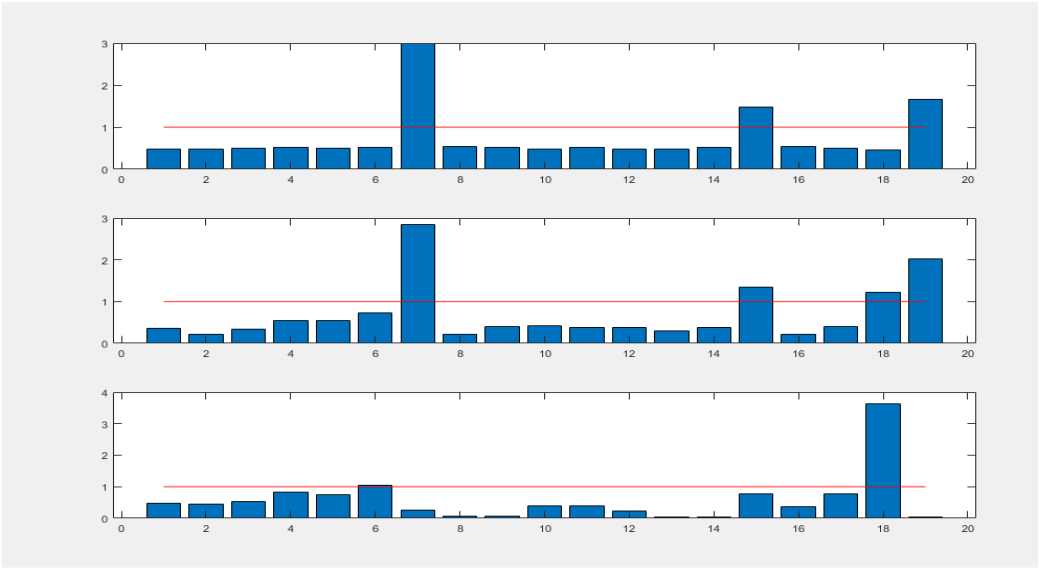

Figure 12: Global metrics and thresholds on ICs

The next step, after we have decided which components to reject, is to back project the remaining components to the channels x timepoints. This process is performed by projecting the ICA component activations through the associated weight matrices to reconstitute the observed data using only the selected ICA components.

Import of Annotations

With the steps of signal filtering and artifact rejection with ICA, we have completed the step of preprocessing. Following up, we want to import the annotations of the three experts onto the signals and then define our regions of interest.

Since we want to examine the presence of seizures in neonatal EEG signals, it is appropriate to define the regions where the seizures occurred.

The three experts annotated the signals for seizure, mainly by visual detection in the EEG. They were not presented with the clinical details of the infants. The annotations were based solely on the EEG signals, while there was an additional channel of ECG for interpretation.

It is expected that the opinion of the three experts would not converge in all cases. More specifically, the 65% of the seizures was annotated by all three reviewers, the 21% was annotated by two reviewers and 14% was annotated by one reviewer. In order to ensure that the annotated regions refer to seizures, we took the consensus of the three experts and only used these annotations for seizure characterization on each dataset.

Signal Epoching

After importing the annotations on all datasets, we need to epoch our data. We are only interested in the regions of the signals when the seizures occurred, so we need to extract these time windows. In addition, we extract time windows where no seizure occurred, to use them in the classification step as control samples.

The duration of seizures varies, and the mean duration is 92 sec. We decided to also extract non-seizure windows of the same duration (92 sec), so that we have comparable sizes.

This process results to a new dataset, containing only a number of seizure and non-seizure windows, (with all 19 channels), for each dataset.

In the next step of feature extraction, these time windows are the only part of the signal that we will be using, and they will provide us with the features that we need to extract.

However, since the seizure windows have a big duration variance, we decided to examine if the window size will affect the next steps, so we created four trials of different window sizes. In each trial, each seizure was divided into overlapping windows of different size. The non-seizure windows followed the partitioning of the seizure windows. The trials partitioning is presented in Table 1. The results from the partitioning trials are quite interesting and they are presented in the Results Chapter.

Table 1: Window partition trials

Seizure size (sec) / overlap (sec)	Non seizure size (sec)
Full size / no overlap	92 / no overlap
45 / 10	45 / 10
20 / 5	20 / 5
10 / 2	10 / 2

Feature Extraction

EEG records multi-frequency non-stationary brain signal from channels localized in a specific topology. Feature extraction is the first – and a very important – step, towards machine learning (classification or regression). It refers to the process of transforming raw data into numerical features that can be processed while preserving the information in the original data set. It yields better results than applying machine learning directly to the raw data.

Several feature extraction methods, specifically for signal analysis, exist. Features can be extracted from time domain, frequency domain or both (wavelet analysis). Especially for EEG signals, wavelet analysis has proven to yield great results, since it performs better on non-stationary signals, compared to time-domain and frequency-domain analysis. Wavelet analysis of EEG signals considers measurement of mean spectral magnitude or power for some given frequency bands. (Varsha K. Harpale, n.d.)

We decided to use the discrete wavelet transform, which is one of the most commonly used methods for feature extraction, as it has non-stationary characteristics, like multi-channel EEG signals. DWT can be used with various mother wavelets with the latest referring to linear transformations where the base functions are shifted and scaled versions of an initial function. The Morlet wavelet has been associated with similar EEG studies, so we decided to use this one.

We implemented the DWT analysis on all main frequency bands which are:

Delta (0.5−4 Hz), Theta (4−8 Hz), Alpha 1 (8−11 Hz), Alpha 2 (11−15 Hz), Beta (15−30 Hz), Gamma Low (30−45 Hz), Gamma High (45−63 Hz).

These frequency bands are also called brain rhythms. These rhythms have been studied upon for decades and there has been established some characteristic behavior related to each one of the brain rhythms.

- Delta: Delta waves range from 0.5 to 4Hz. They are the lower-frequency waves, and they are related with the state of deep sleep and continuous attention tasks. Due to similar nature, they are sometimes confused with movement artefacts.

- Theta: Theta waves range from 4-8Hz. They are related to sleepiness or deep meditation states.
- Alpha: Alpha waves range from 8-15Hz. They are mostly occurring on the occipital lobe of the brain. Alpha waves have been linked with relaxation states of mind, and states of closed eyes. Any state of sudden distraction or stress reduces the alpha waves.
- Beta: Beta waves range from 15-30Hz. As the frequency arises and the brain gets more active, beta waves are linked with active thinking, states of high alert and anxiety or high brain focus.
- Gamma: Gamma waves range from 30-63Hz. They are considered to participate in higher brain functionality, like combining information from different input sensory sources in order to produce a complex outcome. They are also related with certain brain diseases. (Bajaj, 2020)

We calculated the mean energy and the standard deviation for each of the beforementioned frequency bands. We also used the full spectrum frequency and calculated the mean energy and the standard deviation as well. Finally, we also estimated the three statistical measures which were also used for artifact rejection (kurtosis, Shannon entropy and Renyi entropy) and added them to our features.

The final list of estimated features (19 in total), is the following:

1. delta_meanEnergy
2. theta_meanEnergy
3. alpha1_meanEnergy
4. alpha2_meanEnergy
5. beta_meanEnergy
6. gammaLow_meanEnergy
7. gammaHigh_meanEnergy
8. delta_stdEnergy
9. theta_stdEnergy
10. alpha1_stdEnergy
11. alpha2_stdEnergy
12. beta_stdEnergy
13. gammaLow_stdEnergy
14. gammaHigh_stdEnergy
15. allFrequencies_meanEnergy
16. allFrequencies_stdEnergy
17. shannon_entropy
18. renyi_entropy, 19. kurtosis

The 19 mentioned features were estimated for all datasets, for all seizure and non-seizure windows. The process was repeated four times, one for each different trial. As expected, the trial with the smallest overlapping window (10 sec / 2 sec overlap) had about

$9 * Num_{windows}$, thus $9 * Num_{features}$ of the full-size window (92 sec average).

Having estimated all our features, we can migrate from the signal domain to the machine learning domain. These features, which are essentially numerical values, will be used as data for training classification models and then testing their accuracy.

Classification

Several algorithms have been used for seizure classification on human EEG signals. (Murugavel, 2013) establishes SVM classifier to perform EEG classification. (Kumar, 2012) analyzes EEG based on fuzzy set. (Zhou, 2018) introduces CNN to analyze EEG signals. (Damodar ReddyEdla, 2018) approaches the problem with ensemble learning, by using classifiers such as random forest.

We decided to implement SVM (support vector machines) and Random Forest, since they are some of the most widely used classifiers. The SVM classifier is a useful tool in two class classification, which works by trying to define the optimal hyperplane that will better separate the data in two classes. Random forest on the other hand, utilizes an ensemble of classification trees. The input data are organized into trees, and the optimal variable candidate set is decided recursively. After the best split in trees has been decided, majority voting is used for classification amongst the trees.

Before training our models, we need to prepare our training and testing datasets, which are the time windows of features estimated in the feature extraction step.

We merged the seizure and non-seizure features in one dataset and labeled the data with two classes, ‘Seizure’ and ‘NonSeizure’, matching each feature to its corresponding class.

In order to avoid training a biased model, we divided the dataset into training and testing datasets. 70% of the data were used for training, while 30% were used for independent testing. This way, we can train our model on the majority of samples and test them on an independent testing dataset.

All models were cross validated by 10-fold cross validation. Cross-validation is a technique for evaluating machine learning models by training several models on subsets of the training data and evaluating them on the complementary subset of the data. 10-fold cross validation divides the data into 10 equal sets and trains a model on 9 of them as input and tests it on the remaining subset. The process is repeated 10 times and then the average accuracy result is kept. Cross validation is used to avoid overfitting the data, which means the danger of generalizing a data specific pattern.

The two models that we trained were SVM and random forest. Before training the SVM classifier, we performed parameter optimization, in order to achieve better results. More specifically, hyperparameter optimization tries to find the optimal values of box-constraint and kernel-scale. Box-constraint is a value of allowed misclassification in the training set, when the data is not perfectly separable. The higher the box-constraint the higher the cost of the misclassified points, leading to a stricter separation of the data. Kernel-scale on the other hand, is a scaling parameter for the input data.

SVM typically follows the following steps:

1. A hyperplane separates the data in two classes is found.
2. The algorithm runs recursively in order to maximize the margin between the data and the hyperplane.
3. The mapping of the input data to the high-dimensional feature space is performed by a kernel function.
4. The kernel function is tuned by kernel parameters.
5. The tuning parameters are optimized by the process of cross-validation.
6. After the optimal tuning parameters are derived, class predictions are made for all samples.
7. Total accuracy level is estimated by computing the average classification rate from all repetitions.

Random Forest on the other hand, typically follows the following steps:

1. N number of random records are taken from the data set having k number of records
2. Individual decision trees are constructed for each sample
3. Each decision tree will generate an output
4. Final output is considered based on Majority Voting or Averaging for Classification and regression respectively

Both models were trained with 10-fold cross validation. We repeated the process for all four trials of different window size that we had (92 sec, 45 sec, 20 sec, 10 sec). The accuracy scores are presented in the results chapter.

Feature Selection

As a final step, we performed feature selection on our estimated features from the feature extraction chapter.

Feature selection is the process of reducing the number of input variables (features) when developing a classification or regression model. It is desirable to reduce the number of input variables to both reduce the computational cost of modeling and, in some cases, to improve the performance of the model. Feature selection aims on reducing the original set of features, by removing irrelevant and noisy or redundant features that possibly have a negative impact on the model. (Jianyu Miao, 2016)

Several algorithms exist for feature selection, like SVM RFE (recursive feature elimination), Pearson Correlation, Chi-squared, Tree based etc.

We decided to use the Maximum Relevance Minimum Redundancy (mRMR) algorithm for the selection of features. mRMR selects a subset of features which have the most correlation with a class (maximum relevance) and the least dependency among themselves (minimum redundancy). Relevance is calculated by the F-statistic (continuous features) or the mutual information (discrete features), while redundancy is calculated by the Pearson

correlation coefficient (continuous features) or mutual information (discrete features). (Milos Radovic, 2017)

mRMR sorts all features by their maximum relevance and minimum redundancy score and then the user can decide how many to select. After many trials, we decided to select the top six features on all trials, since they managed to achieve the higher classification scores.

The top six features that were selected from the mRMR algorithm are the following:

1. delta_meanEnergy
2. gammaLow_meanEnergy
3. gammaHigh_meanEnergy
4. Shannon_entropy
5. Renyi_entropy
6. Kurtosis

We then trained again all our models, with the subset of the selected features, for all different window size trials. The results are presented in the results chapter.

Chapter 4: Research findings / results

In this chapter we present the results from the stages of our study. The results from the pre-processing stages (filtering, ICA) have already been presented. Here we present the results from the classification and feature selection steps.

Classification

Tables 1-4 contain information about the mean accuracy of the 10-fold validated models and the variance between them. Sensitivity and specificity, two important metrics have also been included. Sensitivity (true positive rate) is used to determine the proportion of positive cases which got predicted correctly as positive, while specificity (true negative rate) is the proportion on negative cases which were predicted as negative.

$$Sensitivity = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

$$Specificity = \frac{True\ Negative}{True\ Negative + False\ Positive}$$

The addition of these metrics can give us more information about the quality of our models, and how they perform on the decision of each class. For instance, a high sensitivity rate

could imply that the model is able to distinguish the first class with high accuracy, while a low specificity would imply a weakness in classifying samples in class two.

The two classification models where first trained and tested on the same datasets used for training. Afterwards, they were tested on the independent testing datasets. The process was repeated four times, each time having a different window partition trial. The results of each trial are presented in the following tables.

Table 2: Full window size / 92 sec non seizure window

	training		testing	
(%)	SVM	RF	SVM	RF
Mean accuracy	82.01	91.86	81.65	77.12
Variance	0.001	0.02	2.24E-28	0.24
Sensitivity	0.97	0.98	NaN	0.35
Specificity	0.99	0.90	1	0.87

Table 3: 45 sec – 10 sec overlap seizure / non seizure window

	training		testing	
(%)	SVM	RF	SVM	RF
Mean accuracy	67.81	98.90	67.77	97.12
Variance	5.9E-05	0.21	1.34E-06	0.52
Sensitivity	0.65	0.99	0.75	0.96
Specificity	0.99	0.99	1	0.92

Table 4: 20 sec – 5 sec overlap seizure / non seizure window

	training		testing	
(%)	SVM	RF	SVM	RF
Mean accuracy	68.58	97.81	68.52	97.33
Variance	1.51E-05	0.3	1.15E-07	0.09
Sensitivity	0.79	0.98	0.5	0.95
Specificity	0.99	0.99	1	0.98

Table 5: 10 sec – 2 sec overlap seizure / non seizure window

	training		testing	
(%)	SVM	RF	SVM	RF
Mean accuracy	70.47	97.76	75.32	94.81
Variance	3.78E-06	0.07	2.7E-05	0.04
Sensitivity	0.82	0.99	0.16	0.97
Specificity	0.99	0.99	0.99	0.99

Feature Selection

In this section we present the results after the selection of the six most relevant features (delta_meanEnergy, gammaLow_meanEnergy, gammaHigh_meanEnergy, Shannon_entropy, Renyi_entropy, kurtosis). Following the classification step, the models have been tested both on the training datasets themselves, and the independent testing datasets. The process was repeated four times, each time having a different window partition trial. The results of each trial are presented in the following tables.

Table 6: Full window size / 92 sec non seizure window (with selected features)

	training		testing	
(%)	SVM	RF	SVM	RF
Mean accuracy	81.64	87.82	81.64	75.45
Variance	1.14E-06	0.01	2.24E-28	0.38
Sensitivity	0.5	0.82	NaN	0.22
Specificity	1	0.98	1	0.90

Table 7: 45 sec – 10 sec overlap seizure / non seizure window (with selected features)

	training		testing	
(%)	SVM	RF	SVM	RF
Mean accuracy	67.70	98.35	67.75	97.83
Variance	1.6E-03	4.01	2.24E-28	0.1
Sensitivity	0.2	0.99	0.14	0.98
Specificity	0.99	0.99	0.99	0.99

Table 8: 20 sec – 5 sec overlap seizure / non seizure window (with selected features)

	training		testing	
(%)	SVM	RF	SVM	RF
Mean accuracy	68.50	97.58	68.51	97.73
Variance	1.12E-06	0.57	2.24E-28	0.04
Sensitivity	0.28	0.99	0	0.98
Specificity	0.99	0.99	1	0.99

Table 9: 10 sec – 2 sec overlap seizure / non seizure window (with selected features)

	training		testing	
(%)	SVM	RF	SVM	RF
Mean accuracy	70.39	97.80	70.39	95.23
Variance	7.23E-05	0.05	2.7E-05	0.01
Sensitivity	0.07	0.99	0	0.98
Specificity	0.99	0.99	1	0.99

Chapter 5: Discussion and analysis of findings

We conducted a series of steps analyzing 79 neonatal seizure multi-channel EEG signals in order to train seizure classification algorithms. The study can be divided into four core parts. The first part contains the data preprocessing and artifact rejection steps. The second part contains the wavelet analysis and the feature extraction. The third part contains the training of the classification models, and the fourth part contains the feature selection process.

Preprocessing and Artifact Rejection

The frequency filtering step is a preprocessing step integrated in every EEG study. The low pass filters usually filter out frequencies above 40-60 Hz, while high pass filters attenuate frequencies below 1 Hz. Since we know that these frequencies contain only added noise and not useful signal, it is a common practice to proceed with filtering them out.

The greater challenge arises when we want to analyze the noise components in the spectrum of useful signal. There, it is important to have an objective criterion that will be able to distinguish between artifacts and useful signal amongst the examined independent components. Although eeglab provided us with a visualization map containing estimations on the activation of different brain sections for each component, we did not confirm the accuracy of these estimations as we observed a significant loss of useful signal when rejecting these components. The more objective criterion that we used for artifact removal (estimation of kurtosis, Shannon entropy and Renyi entropy) yielded good results, as it managed to accurately reject the noisy components. The latter were visually inspected by examination of the signal data for each rejected component and confirming that they are related to noise spikes from their signal characteristics (frequency, amplitude, drift).

Feature Extraction

One of the most crucial steps for machine learning is always the extraction of signal-related features, that will be able to provide the model with sufficient and strongly event-related information in order to take accurate classification decisions. As we are working with epochs of a multi-channel EEG signal, where seizures occurred, we expect these parts to have different statistical characteristics that the normal EEG signal epochs.

Wavelet analysis has proven to obtain the most out of each epoch window, as it analyzes the signal both in time and frequency domain. They are also the preferred method compared to STFT (Short Time Fourier Transform), because the wavelet analysis using different wavelet functions can enrich the study with more details. (Bajaj, 2020)

The extracted wavelet features which were based on the brain rhythms, gave us sufficient information about the event related brain activity, which we later used as input for the

classification step. These frequency bands (brain rhythms) contain information about various brain functionality and characteristics and are good indicators to highlight abnormal (seizure related) activity.

The three statistical measures that were also computed and extracted as features (Shannon entropy, Renyi entropy, kurtosis), provided us with useful information about the distribution and the characteristics of the signal parts, with and without seizure events. Their importance is confirmed, as all three of them were selected in the feature selection step.

Classification

We trained two classification models on segmented EEG datasets containing seizure events and control window samples. Since we expected our models to perform differently based on window size, we decided to extract our features from different window sizes on each trial and see how the different window sizes will affect the accuracy of our models.

We performed four different partitioning trials on our datasets. The first, contains the full window size of the seizures and a 92 sec time window for the control samples (which is the average duration of the seizures). In the second trial, we split our seizure and non-seizure windows to 45 sec windows with 10 sec overlap between them and we extracted the features. The extracted features were then used for training and testing our classification models. The third trial contained 20 sec windows with 5 sec overlap, and the fourth trial contained 10 sec windows with 2 sec overlap, and the process of feature extraction and classification was repeated for each trial respectively. It is reasonable that the number of the extracted features would be different on each trial, with the smaller window size trial having the greatest number of features.

The accuracy scores showed that the varying window size indeed affected the performance of our models. As we partitioned the window to smaller sizes, the SVM seemed to perform more poorly compared to the full-size window. It started from 81.67% on the full-size window and was reduced to 70.39% while the window size was getting smaller. Random forest on the other hand, seemed to work better when presented to more features of smaller window size, than the smaller number of full-size window features. It started from 91.86% on the full-size window and was increased to 97.76% on the 10 sec window trial.

This impact of the varying window size on our models, can be related to the different algorithms that the two machine learning models used for training. As we all know, support vector machines try to define the optimal hyperplane that will optimally separate the input data to two classes. The optimal hyperplane is defined by trying to maximize the distance (margin) between the observations and the hyperplane itself. This means that the marginal values that are closer to the hyperplane will be more prone to be falsely classified, compared to the values that have larger distance from the hyperplane, where the classification task is easier. That is the main reason why the SVM models perform worse as we proceed with smaller window size trials, as the largest number of features will inevitably

produce some marginal values which can be easily classified on the wrong classification class. The following figure describes the mentioned problem.

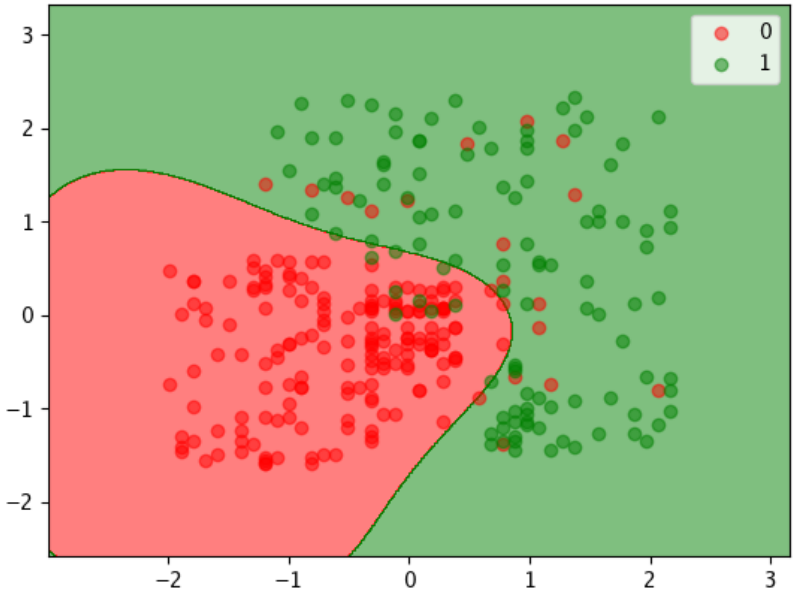

Figure 13 – Values closer to the hyperplane are more prone to be misclassified compared to values with greater distance from the hyperplane

Random forest on the other hand, seems to perform better when it is trained with more feature datasets of smaller window length. Random forest works by selecting random samples from the provided dataset and then create a decision tree for each selected sample. Each decision tree will be fully grown, and a prediction or classification result will be obtained from each decision tree. Majority voting is used to decide the final outcome. A large number of uncorrelated decision trees is the key to making the best data-related decisions and it is the main advantage of ensemble training compared to other models. Thus, it is deducted that the more we increase the number of uncorrelated feature samples, the better accuracy results we will yield, since more independent forests will be grown, and more consistent decisions will be made. This justifies the increase in the accuracy of random forest, when the window size was decreased, generating more feature samples.

Another important parameter in achieving acceptable accuracy scores, is hyper parameter optimization. SVM hyperparameters include box-constraint and kernel-scale and they are directly involved in the performance of the model. The box-constraint is the degree of allowed misclassification when the training data is not perfectly separable, with higher values meaning stricter data separation. Kernel scale is a scaling parameter for the input data. For random forest, the tunable hyperparameters include the number of decision trees and the number of chosen variables to split at each node.

We trained our models both with and without parameter optimization and the results were obvious. Parameter optimization played a crucial role in both SVM and random forest classifiers, as it managed to yield higher accuracy scores of >30%, compared to the absence

of parameter optimization. This is justified by the fact that there is a strong dependance between choosing the right kernel function and setting its parameters appropriately, with achieving high classification results. (Syarif, 2016)

The process of cross validation also played an important role in avoiding overfitting and providing our models with more samples to train and validate. Cross validation was also important in choosing the optimal hyper parameters for our classification models.

The two models were tested on both training and independent testing datasets. As expected, the accuracy scores were slightly higher when tested on the datasets used for training, rather than on the independent testing datasets. This happens due to the correlation of the model parameters with the training data, as the model already has knowledge of the data characteristics and is prone to becoming biased. However, the models yielded high accuracy rates on the independent testing datasets as well, confirming their contribution in neonatal seizure classification tasks.

In supervised machine learning, we generally have to keep a balance between variance and bias in order to train a good model. The variance generally increases on SVM classifiers when we decrease the margin, which also results in decreasing the bias. Respectively, when we increase the margin of the hyperplane, the variance decreases and the bias increases. We also included the variance in our study to ensure that a balance is kept while trying to define the optimal hyperplane by adjusting the constraints of the margin. Low values in variance are desirable (but non-zero), as we want to optimally distinguish two close data points, without considering them similar.

The performance of machine learning models can be evaluated by several metrics. Besides classification accuracy which we have already discussed, we also included sensitivity and specificity, two widely used measures that describe the ability to correctly decide and classify data to each class. Sensitivity, or true positive rate, describes the ability to correctly classify the positive class, while specificity, or true negative rate, describes the ability to correctly classify the negative class. The results showed us that the negative class (non-seizures) was correctly identified in most cases. The specificity rates, that were >95% for both SVM and RF in all window splitting trials, imply that the models were able to classify data in the negative class (non-seizures) with great accuracy, independently of the window size and number of features.

However, the same did not apply in the case of sensitivity. From the testing of our models, we observe that SVM had lower sensitivity rates in the window splitting trials than the full-size window. This means that while the seizures were split in smaller windows, the SVM models could not predict the positive class (seizures) with such accuracy, compared to the full-size windows. This can be explained by the fact that SVM performs poorly when introduced with many marginal values which make the definition of the hyperplane harder, compared to higher margin values, where the classification task is easier.

Random forest on the other hand, had the reverse behavior on choosing the positive class (seizures), as the results show that the sensitivity was increased while we were proceeding with smaller window size partitioning trials. This behavior is also justified by the fact that larger training datasets result to more created uncorrelated random forests, which results in majority voting to choose the positive (or negative) class optimally.

Summing up, we can deduct that both models managed to handle the neonatal seizure classification task successfully. The wavelet analysis was a good technique for feature extraction, as suggested by the literature, since its strong relevance with brain functionality was confirmed by the extracted features that were successfully employed for seizure classification. The varying window size affected the decisions of our models, with smaller windows of larger training datasets favoring random forest over SVM. Overall, random forest outperformed SVM, especially in terms of sensitivity, where the latter appeared to struggle in correctly deciding the positive class, while the window size was decreasing.

Feature Selection

The feature selection process confirmed the remarks we made in the classification step. The MRMR algorithm managed to associate the features with the classification result and rank the most important ones by relevance score. The goal was to obtain the minimum-optimal subset, where the most relevant features would be included, while preserving the highest accuracy rate. The final subset of selected features managed to train classifiers that yielded accuracy scores similar to the full feature set and in some trials, even higher. The fluctuation of the accuracy and sensitivity/specificity metrics followed the one from the full feature set classification, where the different number of predictor values (on smaller window size trials) also affected the models, and the sensitivity difference between SVM and random forest was present as well. The whole feature selection process managed to filter out the less relevant features, while preserving the high accuracy scores in both models, pinpointing the features which are contributing the most for correct seizure classification.

Chapter 6: Conclusion and recommendations

Neonatal seizures are a major problem in infants, being responsible for many serious diseases and in some cases, even death. One of the main problems is that it remains undetected in most of the cases, resulting in false diagnosis and insufficient early treatment. Even though many studies exist on the analysis of multi-channel EEG signals for neonatal seizure classification with the involvement of machine learning methods, which is the gold standard for seizure detection, less studies have focused on thorough artefact rejection, before proceeding with the classification task. Neonatal EEG signals, besides the various noise sources that affect all EEG signals, also suffer from added artefacts due to the nature of the examination (infants are more prone to move, cry, etc.). The lack of experienced pediatric neurologists with expertise in neonatal EEG, is also a great challenge in accurate seizure detection in neonates.

Our study contributes a set of extracted features based on wavelet analysis and two supervised classification models, trained on these features. ICA was involved in the step of artefact rejection, in order to filter out the irrelevant to the signal noise sources and extract features that contained the highest possible SNR. Four trials were performed, where the feature window size was changed, and the feature extraction and classification/ feature selection steps were repeated. The variance of the seizure window size appeared to have an impact of the classification models, favoring random forest and making SVM struggle while the number of marginal samples increased.

The proposed models (SVM and random forest) were tested on independent datasets, yielding accuracy scores of > 80% and >95% respectively. These models can also be used for classification tasks on different multi-layer EEG signals and present the physician with accurate and useful insights for the absence or presence of seizures in a neonatal EEG.

However, before the process of feature extraction, which is a time-consuming process, the correct preprocessing steps need to be implemented. Filtering in frequency and artefact rejection through ICA and estimation of global metrics, are important steps for extracting artefact-free EEG signals. Then, the classification models can be used on the proposed computed features (delta_meanEnergy, gammaLow_meanEnergy, gammaHigh_meanEnergy, Shannon_entropy, Renyi_entropy, Kurtosis).

Since the important step of ICA has been successfully integrated in the preprocessing phase, this study can be expanded in various ways. More supervised machine learning could be trained for the classification task, such as KNN, neural networks, etc.) These different algorithms can be compared to SVM and random forest, and their dependance on the varying window size could be examined. Moreover, the study could be expanded to a deep learning analysis, provided that there is an adequate number of training samples.

In addition, besides the extracted features that are used for classification, we could also use the preprocessed signal epochs for regression tasks. A softmax function could be employed for the estimation of the probability function over a time window on the signal. The distribution of the probability on overlapping moving windows, could give a prediction value for the presence or absence of a seizure on the signal. These probabilities could work as predictors for regression tasks on neonatal seizure detection, advancing the study even further and providing the physician with an estimation both on the diagnosis and prognosis of this severe condition.

References

Aarabi, A., 2007. A multistage knowledge-based system for EEG seizure detection in newborn infants.

Açikoğlu, M., 2019. Incorporating Feature Selection Methods into a Machine Learning-Based Neonatal Seizure Diagnosis. *Medical Hypotheses.*

Amin, H. U., 2015. Feature extraction and classification for EEG signals using. *Australasian College of Physical Scientists and Engineers in Medicine.*

Awnish Kumar, R. T. A. G., 2020. Low Frequency Noise Remove from EEG Signal. *International Journal of Recent Technology and Engineering (IJRTE),* 9(1), pp. 1510-1513.

Aziz, R., 2017. Dimension reduction methods for microarray data: a review.. *AIMS Bioengineering,.*

Bajaj, N., 2020. *Wavelets for EEG Analysis.* s.l.:s.n.

Britton JW, F. L. H. J. a., 2016. Electroencephalography (EEG): An Introductory Text and Atlas of Normal and Abnormal Findings in Adults, Children, and Infants. *American Epilepsy Society.*

Brown, G., 2012. Conditional likelihood maximisation: a unifying framework for information theoretic feature selection. *Mach Learn Res.*

Celka, P., 2002. A cinoyter aided detection of EEG seizures in infants.

Chen, W., 2013. A random forest model based classification scheme for neonatal amplitude-integrated EEG. *IEEE International Conference on Bioinformatics and Biomedicine.*

Cheveigné, A., 2019. Filters: When, Why, and How (Not) to Use Them.

D'Avanzo, C., 2009. A wavelet Methodology for EEG Time-frequency. *International Journal of Bioelectromagnetism .*

Damodar ReddyEdla, K. M., 2018. Classification of EEG data for human mental state analysis using Random Forest Classifier. *Procedia Computer Science,* Volume 132, pp. 1523-1532.

Edla, D. R., 2018. Classification of EEG data for human mental state analysis using Random Forest Classifier. *Procedia Computer Science,* Volume 132, pp. 1523-1532.

Giuseppina Inuso, 2007. Brain Activity Investigation by EEG Processing:. *Proceedings of the 2007 International Conference on Information Acquisition.*

Jianyu Miao, L. N., 2016. A Survey on Feature Selection. *Procedia Computer Science,* Volume 91, pp. 919-926.

Kumar, Y. D. M. L. a. A. R. S., 2012. Relative wavelet energy and wavelet entropy based epileptic brain signals classification. *Biomed. Eng. Lett. 2,* pp. 147-157.

L. Lee, M. L. H. a. A. M., 2003. The functional brain connectivity workshop: report and. *NeuroImage,* Volume 19, pp. 457-465.

Liang, Z., 2015 . EEG entropy measures in anesthesia.

Ling, W.-K., 2007. Shannon Entropy. *Nonlinear Digital Filters.*

Milos Radovic, M. G. N. F. Z. O., 2017. Minimum redundancy maximum relevance feature selection approach for temporal gene expression data. *BMC Bioinformatics.*

Minetti, C., 2020. Deep learning for neonatal seizure detection: a friend rather than foe.

Mohammad Reza Mohammadi, 2016. EEG Classification of ADHD and Normal Children Using Non-linear Features and Neural Network. *The Korean Society of Medical & Biological Engineering and Springer.*

Murugavel, A. M. R. S. M. U. a. S., 2013. Combined seizure index with adaptive multi-class SVM for epileptic EEG classification. *International Conference on Emerging Trends in VLSI, Embedded System, Nano Electronics and Telecommunication System (ICEVENT).*

Muthukumaraswamy, S. D., 2013. High-frequency brain activity and muscle artifacts in MEG/EEG: a review and recommendations. *Hum. Neurosci.*

Pirooznia, M., 2008. A comparative study of different machine learning methods on microarray gene expression data. *BMC Genomics.*

Rabia Musheer, 2015. tindependent component analysis for svm classification of dna- microarray data.. *International Journal of Bioinformatics Research.*

Raghu S, S. N., 2018. Classification of focal and non-focal eeg signals using neighborhood component analysis and machine learning algorithms.. *Exp Syst Appl.*

Ram´ırez-Gallego, S., 2016. Fast-mRMR: Fast Minimum Redundancy Maximum Relevance Algorithm for High-Dimensional Big Data. *INTERNATIONAL JOURNAL OF INTELLIGENT SYSTEMS.*

Siddiqui1, M. K., 2020. A review of epileptic seizure detection using machine learning classifiers. *Brain Informatics.*

Silverstein, F. S., 2007. Neonatal Seizures. *Neurological Peogress.*

SR, M., 2016. Validation of an automated seizure detection algorithm for term neonates.

Stevenson, N. T. K. L. L. e. a., 2019. A dataset of neonatal EEG recordings with seizure annotations.. *nature.com.*

Syarif, I., 2016. SVM Parameter Optimization using Grid Search and Genetic Algorithm to Improve Classification Performance. *TELKOMNIKA (Telecommunication Computing Electronics and Control).*

Tanveer, M. A., 2021. Convolutional neural networks ensemble model for neonatal seizure detection. *Journal of Neuroscience Methods.*

Temko, A., 2011. EEG-based neonatal seizure detection with Support Vector Machines. *Clinical Neurophysiology.*

Temko, A., 2014. Decision Support Systems.

Temko, A., 2016. Detecting Neonatal Seizures With Computer Algorithms.

Tharwat, A., 2018. Independent component analysis: An introduction. *Applied Computing and Informatics.*

Varsha K. Harpale, V. K. B., n.d. Time and Frequency Domain Analysis of EEG Signals for.

Verma, A., 2015. Adaptive Tunable Notch Filter for ECG Signal Enhancement.

Zekic-Sušac, M., 2014. A comparison of machine learning methods in a high-dimensional classification problem. *Business Systems Research Journal.*

Zhou, M., 2018. Epileptic seizure detection based on EEG signals and CNN. *Front. Neuroinform.*