UCLouvain

École polytechnique de Louvain

Analysis of anaesthesia EEG recordings with machine learning techniques

Prediction of Postoperative Delirium

Author: **Amaury Fierens**
Supervisors: **Mona Momeni, Michel Verleysen**
Readers: **Céline Khalifa, André Mouraux, Dounia Mulders**
Academic year 2021–2022
Master [120] in Computer Science and Engineering

Acknowledgments

First of all, I would like to thank the professors who initiated this thesis, Pr. Michel Verleysen and Pr. André Mouraux who gave me the opportunity to work on this subject combining neurology and Machine Learning. These two subjects have captivated me throughout this year. I would also like to thank them for their availability to answer my questions, their follow-up, their advises and their feedback.

Second, I would like to thank Mrs. Céline Khalifa and Pr. Mona Momeni for their availability to answer my questions. I would also like to thank them, as well as Dr. Dounia Mulders, for having accepted to be part of my jury.

Third, I would like to thank Mrs. Aurélie Deneumoustier, as well as my friend Pierre and my sister for their precious proofreading, and Dr. Cédric Lenoir for his detailed answers to my questions.

Finally, I would like to thank my family and Louise for supporting me throughout this year and throughout all my studies, as well as my close friends Valentin, Takumi, Sarah, Valentine and all my roommates for their unfailing support.

Abstract

Nowadays, Postoperative Delirium is a disorder that affects a large number of people after a surgical operation under general anesthesia. Over the years, evidence seems to suggest that anesthesia is closely related to Postoperative Delirium. Among several findings, one under study shows a link between the frequency space of electroencephalographic signals and postoperative delirium, in particular the frequencies corresponding to the alpha rhythm. The goal of this master thesis is to find Machine Learning models capable of determining the prevalence of a patient to develop Postoperative Delirium. In addition to the alpha frequency band, the beta frequency band proves to be important in the predictive ability of the models for our data set. Among the models and the reconstructed datasets tested, the best model was the Support Vector Machine on a dataset with 13 electrodes located in the front of the brain. It achieves a Fbeta score ($\beta = 1.5$) of 0.70 with a 95% CI of [0.49, 0.72], a recall of 0.85, a precision of 0.50, a AUC score of 0.75 and a specificity of 0.64.

Contents

Introduction

Since 1846, modern anesthesia has been very widely used around the world to practice surgery[1]. Until recently, general anesthesia seemed to be a relatively harmless operation for the human body. However in recent years, research in the domain of complications linked to anesthesia has increased a lot (see table 1). In particular, the research for Postoperative Delirium (POD) as a complication of anesthesia has emerged and it has become one of the main study subjects among the anesthesia complications (4,146 out of 76,740 papers).

1973	**2021**	**Multiplier**	**Research keywords on full data**
640,930	6,360,540	x9.9	*No keywords: all publications*
14,034	147,224	x10.5	Anesthesia
3,554	76,740	x21.5	Anesthesia AND Complications
32	4,146	x129.2	Anesthesia and Postoperative Delirium

Table 1: Evolution of the number of publications from 1973 to 2021[2] for different research equations. Allows to compare the growth of different fields of studies.

Recently, the frequent occurrences of Delirium (9.1% to 10.9% for general hospital population[3]) and especially POD (9% to 87% depending mostly on the patients age and type of surgery[4, 5]) after surgery have been recorded. The research did also highlighted health problems and discomfort due to POD. As reported in the literature, POD has been associated with increased risks of dementia[4] and long-term cognitive[6] and non-cognitive[7] morbidity. In addition, the length of stay in the hospital increased after a POD[8, 9, 10] and some studies have also reported an increase of mortality[11, 12, 13] associated with POD.

The frequency of these POD, the risks associated and the discomfort felt because of them have aroused interest around the world. The objectives behind this interest are to find out what factors would be at the origin of the phenomenon and to find biomarkers to be recorded during the operation that could indicate future appearance of POD. Among the various biomarkers already studied, some quite promising are detectable in the electroencephalogram (EEG) signals of patients during operations. The promising aspect of

these biomarkers is due to the fact that many studies conducted in the field of study of machine learning (ML) have shown convincing results in the prediction of brain diseases using kinds of biomarkers, like Alzheimer[14] or Parkinson[15]. Compared to a majority of medical devices (PET-scan, MRI, CT-scan, ...), EEG devices are inexpensive. Therefore a prediction method could be implemented quickly if doable.

Therefore, being able to predict, via ML models, the potential appearance of POD after a given operation has become a challenge. Indeed, these potential appearance would be able to manage the symptoms and to inform the patient of his current risks.

Other challenges emerging in building ML models for POD are the confirmation of already identified biomarkers and the identification of new potential biomarkers not detected before. For this purpose, analyzing the prediction results achieved for different sets of biomarkers (called features in ML) seems to be a promising way.

Chapter 1

Postoperative delirium : State of the art

1.1 Description

A Delirium can be defined as a significant dysfunction of the brain. Principal symptoms of the Delirium are the development of perceptual disturbance, attention deficits, disorientation, emotional dysregulation, disturbed sleep rhythm or increased pychomotor activity[3, 4, 16]. It can be compared with dementia as the symptoms are quite similar. However, when the symptoms of Delirium are back to normal, it is mostly ameliorated compared to dementia. Several names of Delirium have been used in literature to describe the acute mental status changes related to it. The first effects of delirium that have been described are named *Delirium tremens* which is nowadays used to talk about people experiencing a lack of alcohol. Other names - like *intensive care unit (ICU) psychosis*, related to the fact that most of the symptoms occur in the ICU, or *sundowning Delirium*, describing patients more subject to confusion during periods of "brain inactivity" like night - refer to Delirium in the literature[3].

Actually, Delirium is classified in 3 different subtypes in the literature[17, 18]. It can be hypoactive, hyperactive, or mixed. Even if the definition of those categories varies from author to author, as mentioned by D. Stagno[17], a rough definition could be that hypoactive Delirium implies symptoms less marked than hyperactive Delirium, and that mixed Delirium is somewhere in the middle. Some well defined differences between hypo- and hyperactive have been emphasized by S. Boettger and W. Breitbart[18]. The difference in prevalence of hallucinations - also called perceptual disturbances - and delusions is a criteria to distinguish both, as there are more prevalent in hyperactive than in hypoactive. In one of his papers[3], J. R. Maldonado summarizes in detail what the classical symptoms are, which, put together, reflect a Delirium.

The POD is a Delirium present in patients after surgical operations under anesthesia. As we can see in figure 1.1, POD appears usually from one to three days after the operation[4, 19, 20]. As proposed by J. H. Silverstein[20] and reported by M. Berger[19], POD and postoperative cognitive dysfunction (POCD) could be part of a continuous

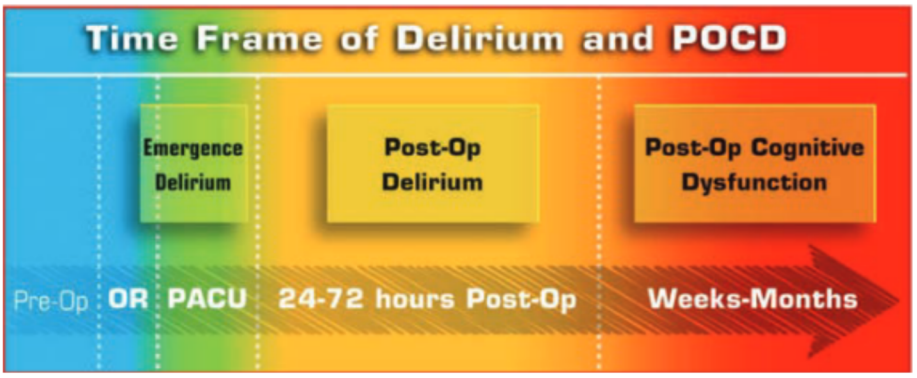


Figure 1.1: This figure shows the different brain complications time frame after a surgical operation under anesthesia. It clearly distinguishes three types of dysfunctions after the operation. Emergence Delirium occurs in the operating room (OR) or in the post-anesthesia care unit (PACU), POD occurs 24h to 72h after surgery and POCD occurs at weeks to months after surgery and anesthesia. Pre-Op means preoperative in this figure. Borrowed from the paper of J. H. Silverstein[20].

spectrum of postoperative dysfunction of the central nervous system. This idea has been proposed because the mechanisms and risk factors behind POD are quite similar to those of POCD and that a lot of patients develop both dysfunctions.

As described by Robinson et Al.[21], the incidence of POD varies from 6% to 87% depending on the patient population and the degree of operative stress. In general, the age is a criteria that increases the risk of developing PO [22].

1.2 Biological mechanisms

Nowadays, the understanding of the physiology of Delirium remains very limited as the field of study of the biological mechanisms responsible for this pathology is relatively recent. The syndrome of Delirium has several aspects and until now, there are not enough studies to assert that a unique mechanism can explain all of these aspects. Some hypothesis, related to biological mechanisms behind the POD, exist to explain the pathology and two of them are actually leading in the literature.

The first hypothesis concerns neurochemical changes for neurotransmitter. Particularly, it affects the cholinergic system[23, 24]. As a reminder, the cholinergic system is a biological system responsible for the synthesis and secretion of Acetylcholine (ACh). ACh is a neurotransmitter that is for example present at neuromuscular junctions and whose disappearance following the degradation of neurons in the cholinergic system leads to a decline in neurocognitive function. In a synthesis of evidence for this hypothesis[23], T. T. Hsieh et Al. summarize the potential pathophysiological mechanisms for Delirium related to ACh. More recently, a study[25] has shown, via neuroimaging techniques (in this case fMRI) differences of connectivity for patients with Delirium in regions producing and/or using ACh during and after an episode of Delirium.

The second hypothesis concerns the role of inflammation[16, 19, 24]. During the surgical operation, some tissues can be injured, leading to the release of cytokines, and thus to inflammation. Two characteristics are then thought to play a role in the likelihood of this inflammation. Firstly, the blood-brain barrier is impacted by cytokines. Since it is has been shown that blood-barrier dysfunction occurs more frequently in older adults[19], it could be a cause of increased frequency of POD in older patients. Secondly, as the cytokine level is known to be increased by chronic stress[16], it appears that chronic stress also affects the blood-brain barrier. Neuroinflammation appears to be responsible for some negative effects in the brain as related by L. Capuron and A. H. Miller[26]. Alone, it can induce memory, cognition and behavior deficits.

1.3 Biomarkers

In addition to the search for POD causes, some research has recently focused on the study of easily acquirable biomarkers which would be able to predict the development of POD for a given patient. Two of them are relatively important for this work. The first one, that will be used in particular in section 1.4 is about plasma neurofilament light (NfL). The second and the third are about EEG and, more specifically, about frontal alpha-band power and about EEG burst suppression and will be used in section 1.4 as well as in order to perform an accurate feature extraction from the EEG signal.

In a paper from C. P. Casey et Al., the association between POD and increased plasma NfL is discussed in details[27]. In order to understand what all this is about, here is an explanation about what Plasma NfL is. Neurons are particular cells of the organism that contain different parts: the cell body (also called soma), the dendrites (receivers of the input messages from other cells), the axon (transmitter of the output messages from this neuron) and the terminal buttons (junctions with other cells). In order for the neurons to hold their size and structure and therefore allowing to efficiently transmit messages to other cells, some proteins are needed. One of those is the NfL protein. It is the smallest amongst the three types of neurofilament (light, medium and heavy) found inside neurons. NfL is mainly located inside axons to maintain its structure and size. When NfL is inside the neuron, it is difficult to measure its concentration, but when a brain injury occurs, or during many neurodegenerative diseases, NfL is released in the cerebrospinal fluid (CSF) in small quantities, and in blood plasma in even smaller quantities. When CSF NfL concentration is easier to measure and is commonly used as a biomarker, pumping CSF is rather more invasive than getting a blood sample. In the recent years, a new technology has been developed (Single Molecular Array) that allows to measure NfL reliably in blood. Therefore, the concentration of NfL in plasma is discussed as a reliable biomarker for POD in the paper cited before[27].

As the study reported, the authors discussed two outcomes they found by analysing their data. The first one is the association between the rise in NfL concentration in plasma and POD. It appears that subjects suffering from POD had a more important rise of NfL level

than the others. The second outcome is related to neuroinflammation discussed earlier. A strong correlation hs been found between inflammation cytokines and delirium severity and another between inflammation cytokines and NfL level rise, all at Day 1 after the surgery. In addition, this study suggests a possible small independent effect of neuronal injury in the pathogenesis of POD, as the change in NfL didn't seem to be completely explained by its relation with neuroinflammation. The plasma NfL is considered to be a relatively accurate biomarker as it reflects at least the neuroinflammation level in the brain, and it could even be involved in an independent effect causing POD.

A second biomarker is EEG burst-suppression. A burst-suppression in general is a pattern usually found in an EEG. It consists of periods of suppression of the signal separated by bursts of activity. Even if it remains controversial[28], it has been emphasized by several studies that appearance of intraoperative EEG suppression was related with appearance of POD[29, 30, 31, 32]. This biomarker could therefore be used as an interesting feature to predict future appearance of POD for a given patient. In a recent paper[33], a ML algorithm has been developed to automatically detect burst-suppression and calculate burst/suppression ration for a given patient. Generating features with this model seems promising to predict POD.

The last biomarker of interest that will be described in details here is the loss of power in frontal α-band of frequency during the operation. This band is going from 8Hz to 12Hz. It is defined in the literature as a physiological band of frequency which is known historically to represent the activity of the visual cortex. In recent papers[32, 34, 35, 36], evidence suggests that this α-band, if monitored during the operation, could be a strong predictor of POD. In a review of the clinical features for the POD[32], the use of α-band as predictor was mentioned. In their study[34], R. Gutierrez et Al. emphasized the potential of this intraoperative α-band to be an efficient biomarker for POD. S. Koch et Al., in their study[35], have shown that this prevalence of decreased α-band power for POD patient was accentuated for older patients suffering of cognitive impairment. It was observed only for absolute intraoperative frontal α-band power (after the anesthesia), not for baseline α-band power (before the anesthesia). This seems to be a feature which is promising enough to try to apply ML algorithms on an intraoperative EEG signal.
Several other biomarkers have been discussed in the litterature that are detectable from EEG as reported by Palanca et Al.[36]. But those markers will not be detailed here as there are no evidence in the litterature about a relation with anesthesia.

1.4 Links with anesthesia

Even if, as said by Whitlock et Al.[4], the links between general anesthesia and POD are not yet fully understood, there are already clues of it. It appears that the biological mechanisms and the biomarkers previously discussed are what mainly connects both.

In the literature, the role of ACh in the mechanisms of anesthesia has already been described. The anesthetic agents mainly act as inhibitors for the ACh receptors as reported by E. Tassonyi[37]. In Alzheimer's disease, a loss of cholinergic neurons results in

profound memory disturbances. In his paper[38], V. Fodale summarizes the knowledge about anesthesia and Alzheimer related to the cholinergic system. He describes the actions of the different anesthetic agents on the cholinergic system in detail, describing each mechanism precisely.

But for the purposes of this work what mainly interests us is the link between general anesthesia and POD on the EEG signals. Both biomarkers related to EEG signals detailed in the previous section seem to be linked with anesthesia.

As reviewed by Pawar et Al.[28], Burst Suppression often occurs when a patient is undergoing general anesthesia. As shown in previous section, there are some clues in the literature that Burst Suppression could be a predictor of POD. But as mention in the same study, if a Burst Suppression is easily detectable visually on EEG, there is no feature documented in the literature that allows an easy detection with a computer. Some methods exist that use Deep Learning as described in the previous section but it requires more than a simple feature extracted from an EEG signal.

As reported by Giattino et Al. and Purdon et Al. [39, 40], when general anesthesia is induced, there is a shift of the alpha rhythm, called anteriorization. The frontal signal increases while the occipital signal decreases. As detailed in section 1.3, a loss of the frontal alpha rhythm is one of the biomarkers of POD. This characteristic of the common EEG signal of a general anesthesia, the anteriorization, seems to be modified for a patient that will suffer from POD. This, added to the fact that all the patients undergoing cardiac operations are operated on general anesthesia, confirms the interest in trying to apply ML algorithms on an intraoperative EEG signal as it could increase the quality of prediction if and when prediction is possible.

Chapter 2

Background

2.1 Electroencephalography (EEG)

2.1.1 EEG principles

In the human body, information is transmitted through neurons via change of membrane potentials. Some of the neurons, called cortical neurons, are part of the central nervous system inside the brain. The changes of potential of those neurons can be recorded at the surface of the brain by using electrodes, as they form electrical signals. The recording of those electrical signals from the brain is called EEG.

This technique of recording have pros and cons. Major advantage is that it is a non-invasive technique, as it requires only to disposed a cap on the scalp of the patient. Major disadvantage is the low spatial localization, as the number of electrodes will never be of the same order of magnitude as the number of emission sources, mainly because the electrodes are bigger that the neurons and far away from them (from the point of view of the neuron). Due to this disadvantage, sources from other parts of the body, such as the heart, are also taken into account. Therefore, signals recorded from this technique are noisy and need to be filtered. Figure 2.1 is an example of such a filtered EEG signal.

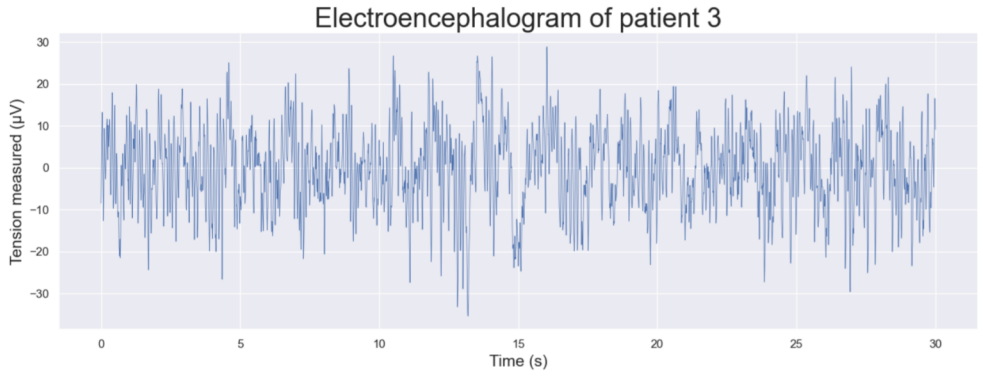


Figure 2.1: Example of filtered EEG signal

To perform such a recording, some recording equipment is needed. It is usually composed of a cap and several electrodes glued to it. The disposition of the electrodes on the cap are called a montage. Several montages exist. Some are used for very specific tasks where other are designed to monitor the whole brain.

Inside the montage, each electrode measures the differences in electric potentials between two recorded voltages. This means that the signals for each channel are the difference with another site of recording. Therefore, in order to be able to compare two electrode signals, one of the electrodes, called the ground electrode, is chosen to be the reference site. But a given reference is sometimes not the best for a given analysis. Therefore, it is possible to re-reference the signal after the recording.

Inside an EEG signal, different waveforms can be found, also called EEG rhythms (see figure 2.2). Each one corresponds to a particular band of frequency and contains information about physiological functions of the brain. In this master thesis, the intervals of frequency used for each band are described in table 2.1.

Name	**Frequencies**
Alpha	8-12Hz
Beta	12-30Hz
Gamma	30-50Hz
Delta	1-4Hz
Theta	4-8Hz

Table 2.1: Physiological bands of frequencies commonly used in the literature[30, 39]

Alpha waves reflect a relaxed mental state for the subject. It occurs for example when the subject is at rest, with their eyes closed, but not tired nor asleep[42]. Some studies have shown the influence of anesthesia on those waves, as described in section 1.4.

Beta waves reflect states of the brain associated with thinking and concentration, at different degrees. They also contain information about muscle contractions and movements in general[43].

Gamma waves seem to vehicle at least a part of the neural consciousness information, and seem to participate to a coherent and unified perception[44].

Delta waves are a way to distinguish between male and female as females have more frequent delta waves than males. In addition to this neurophysiological particularity, the delta waves activity are correlated with the release of several hormones[45].

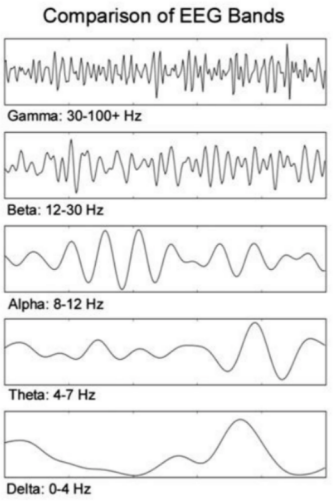


Figure 2.2: Classical EEG waveforms[41]

Theta waves vehicle several aspects of the cognition, like memory, learning and spatial navigation. Particularly, it is related to the hippocampus formation of the brain[46].

2.1.2 Electrode montage

As described in section 2.1.1, when monitoring EEG signals, several types of cap with particular montage exist. The electrode montage used during the operations for each patient consists of 32 electrodes and is called Biosemi 32. The choice of this particular montage is based on the paper of Giattino et Al.[39]. Its objective is to record precise signals from the frontal, central and occipital regions of the scalp. As shown in figure 2.3, the majority of the electrodes are placed in the fronto-occipital axis, with less electrodes placed in the temporal regions.

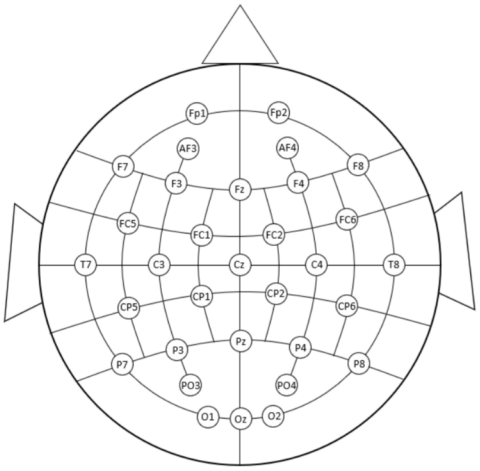


Figure 2.3: Electrode montage used to record the dataset

2.2 Mathematical tools

2.2.1 Power Spectral Density curve

Before using a model in ML, we need to extract features from the dataset. As a reminder, features in ML are the independent variables given to the model.

In the scope of EEG signal, the most commonly used features are the average powers of bands of frequencies. In the case of EEG, the five physiological bands of frequencies commonly used are specified in table 2.1. But other bands of frequencies can also be chosen. For example, it is totally possible to analyze sub-bands of the alpha band from table 2.1.

To get average powers for given intervals of frequencies, a Power Spectral Density (PSD) curve (example given in figure 2.4) needs to be integrated between bounds for each intervals. The method used here is the Welch method and consists in two steps:

1. Cut the signal in a chosen number of segments and to calculate the power of each of those segments. A Discrete Fourier Transform (DFT) of the signal is performed using the Fast Fourier Transform (FFT) algorithm given by the formula 2.1. Then, the formula 2.2 is applied to get power of the given segment.

$$F_s[k] = \sum_{n=0}^{N-1} x[n] e^{-j2k\pi n/N} \tag{2.1}$$

$$P_s(\omega_k) = \frac{1}{M} |F_s[k]|^2 \tag{2.2}$$

   With x the EEG signal in temporal space, ω_k the frequency k and s the segment considered.

2. Average the powers of all the segments (also called the periodogram of the segments) in order to obtain all the power values according to the Welch method. This is done by using the formula 2.3. Then, the Power Spectral Density (PSD) curve can be obtained by plotting the PSD's for each frequency.

$$PSD(\omega_k) = \frac{1}{S} \sum_{s=0}^{S-1} P_s(\omega_k) \tag{2.3}$$

   With S the chosen number of segment.

2.2.2 Simpson integration

The average powers of physiological bands are obtained using integration, and numerically, there are several methods to do this. Here, the Simpson method is used. The idea is to use information from three points from the original signal: the PSD values of the two

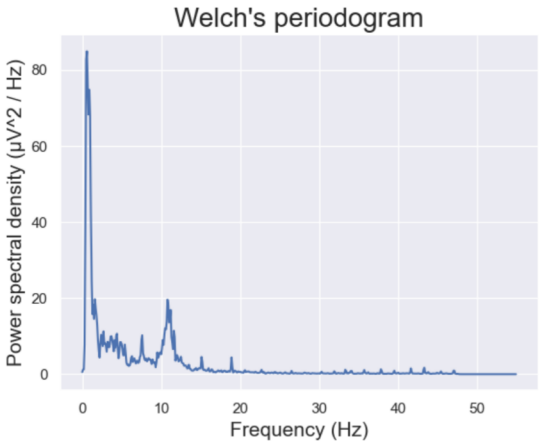


Figure 2.4: Example of Welch's PSD curve

bounds of the band of frequency considered, and the PSD value at the center of the band. The value of the integral of each band is given by the formula 2.4.

$$\int_a^b PSD(\omega)d\omega \approx \frac{b-a}{6}\left[PSD(a) + 4PSD\left(\frac{a+b}{2}\right) + PSD(b)\right] \tag{2.4}$$

With a and b the two bounds of the band of frequency considered.

2.2.3 Synthetic Minority Oversampling Technique (SMOTE)

Classification tasks in medicine often mean that the labels in the dataset will have an imbalanced distribution. Indeed, in the world, there are more people who don't have a disease than people who do have it. Therefore, being able, by using a given method, to deal with imbalanced dataset could be very handy. The SMOTE method is one of the classical methods [47] that allow to deal with those kinds of datasets by oversampling the data artificially.

In order to perform this oversampling, SMOTE principle assume that data from the minor class are close to each other, without looking at the data of the other class. To achieve this, the algorithm looks at one of the observation of the minor class and calculates a chosen number of the closest neighbors for this observation. Then, the algorithm draws lines between it and all the neighbors, and randomly adds an observation somewhere in one of the lines. An example of iteration is shown at figure 2.5.

A last important thing to mention is that SMOTE, like all oversampling methods, must be used on the training data only, avoiding to use it on both validation and testing sets.

2.2.4 Independent Student's t-Test

A classical way to determine if data from two different classes of patients are differently distributed is to look at their mean. When the data is normally distributed, a good

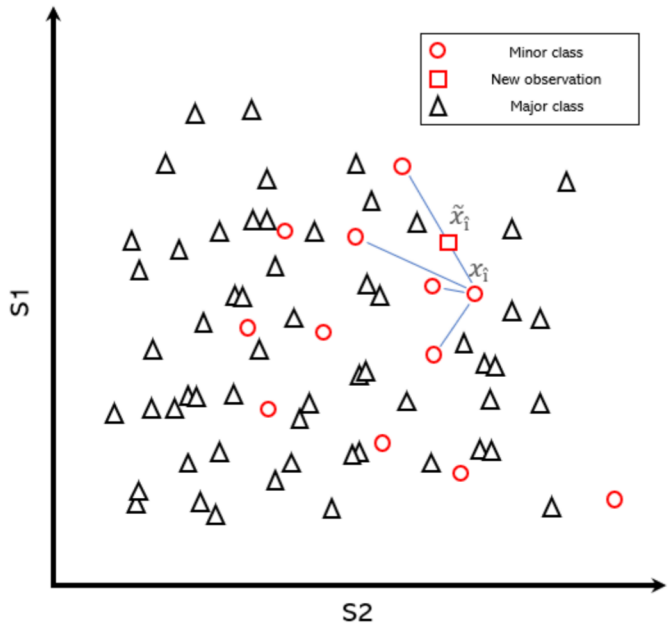


Figure 2.5: Example of new observation generated by SMOTE algorithm

statistical test to achieve this purpose is the Independent Student's t-Test.

The assumptions to use this test are the following:

- The data from both classes must follow normal distributions
- The data follows a continuous or ordinal scale
- The variances of the patient's data must be homogeneous (the standard deviations are approximately the same in both classes)
- Each data value must be independent from others values of the class

In this statistical test, the null hypothesis (H_0) is that the mean (μ_i) of both distributions are equal, while the alternative hypothesis (H_1) is that the means of both distributions are different.

$$H_0 : \mu_1 = \mu_2 \tag{2.5}$$

$$H_1 : \mu_1 \neq \mu_2 \tag{2.6}$$

The t-Test statistic is given by the formula 2.7 and allows us to determine the p-value of the test by looking at a table of Student's t-Test statistics. Usually, we reject the null hypothesis with a significant level of confidence when p-value < 0.05, and with a highly significant level of confidence when p-value < 0.005.

$$t = \frac{\mu_1 - \mu_2}{\sqrt{(\frac{\sigma^2}{n_1} + \frac{\sigma^2}{n_2})}} \tag{2.7}$$

with σ the pooled variance of both classes (as we assume that they are equivalent) and n_1, n_2 the size of classes 1 and 2.

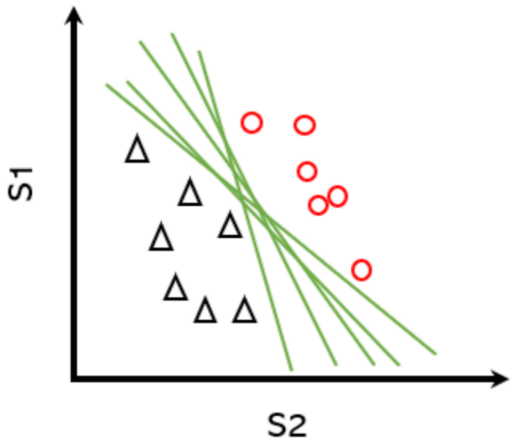


Figure 2.6: Example of possible hyperplanes

2.3 ML models

2.3.1 Support Vector Machines (SVM)

The idea behind Support Vector Machine is to find a hyperplane (a plane in N-dimensions, N being the number of features) that separates the data in two distinct classes. This hyperplane is the decision boundary to classify the samples. But to separate the two classes, there are many hyperplanes that could be chosen, as shown in figure 2.6.

In order to determine the best hyperplane on the training data, the objective is to find a hyperplane with a maximum margin. Both margins are located at a distance equal to the distance to the closest point of each class and are parallel to the original hyperplane, as shown in figure 2.7.

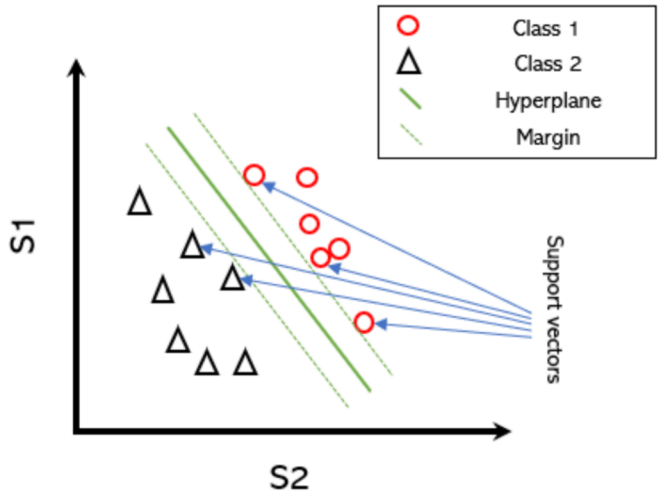


Figure 2.7: Optimal hyperplane with its support vectors

To maximize the margin of the classifier, the SVM algorithm uses what's called support vectors. The support vectors are the data points that are the closest to the hyperplane. Using the total distance from margin to those vectors and trying to minimize it, the SVM algorithm can maximize the distance from the hyperplane to its margins. An additional parameter can be used to smoothen the model and accept a certain number of missclassification, if the data is quasi-linearly separable. The value of this parameter, called C, distinguishes between the smooth-margin algorithm (large C) and the hard-margin one (small C).

To better understand what the algorithm is doing to optimize its margins, a description of the **cost function** is required. The formula of the cost for SVM (see formula 2.8) is composed of two parts.

$$\min_{w} \frac{1}{2} \sum_{i=1}^{n} w_i^2 + C \sum_{j=1}^{m} \max\Big(0, (1 - t_j) \cdot y_j\Big) \tag{2.8}$$

with n the number of features, w the weight parameter corresponding to the orthogonal distances from hyperplane to them, m the number of samples in the data, t the output produced by the model as a product of the weight parameter w and the data input x : $t_j = w^T x_j$.

The first part, $\min_{w} \frac{1}{2} \sum_{i=1}^{n} w_i^2 (= \max_{w} \frac{1}{||w||})$, is the cost of the maximization of the margins.
The second part, $C \sum_{j=1}^{m} \max\Big(0, (1 - t_j) \cdot y_j\Big)$, is called the regularized Hinge loss function, which actually computes a penalty for each misclassed sample. This loss is regularized by the C parameter described earlier.

As the basic method of the SVM only works for linearly separable data and data are not always linearly separable, the SVM algorithm uses a trick to allow passing from a data space to a feature space, where both classes become linearly separable. The transformation is performed by increasing the number of dimensions of the data space, using a kernel function. Figure 2.8 shows how this kernel trick transforms the data space.

Several kernels exist to adapt to different types of non-linearities:

- All polynomial kernels
- Sigmoid kernel
- Gaussian Radial Basis Function (RBF) kernel
- Bessel kernel
- ...

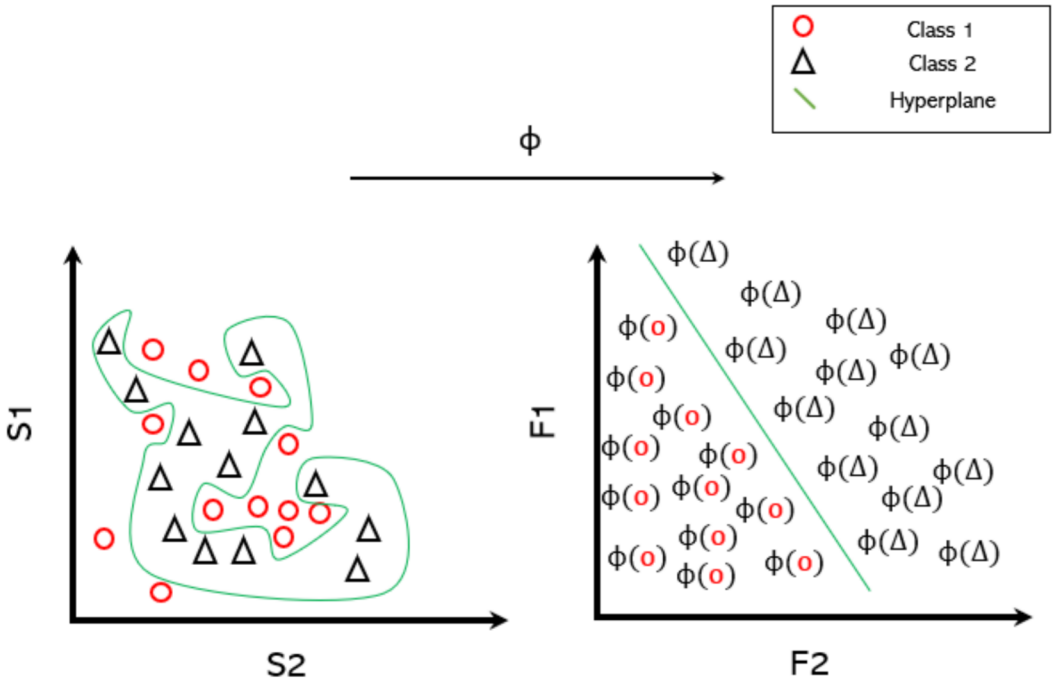


Figure 2.8: Example of kernel trick

In this master thesis, the RBF kernel (see formula 2.9) is the kernel used for the SVM models built. It aims to compute how close two points are from each other using a parameter called γ which sets the spread of the kernel and is actually $\frac{1}{\sigma}$, σ being the unknown variance of the Gaussian kernel.

$$K_{RBF}(\mathbf{x}, \mathbf{x'}) = \exp(-\gamma||\mathbf{x} - \mathbf{x'}||^2) \tag{2.9}$$

2.3.2 K-Nearest Neighbours (KNN)

KNN is a very common algorithm in ML due to its ease of use. The idea behind it is very simple and well explained in the book of Kramer[48] of 2013. The algorithm aims to classify data based on their distances to other points. A majority vote is then performed: if a majority of the k-nearest neighbours around it are from class 1, then the point considered will be labeled as from class 1.

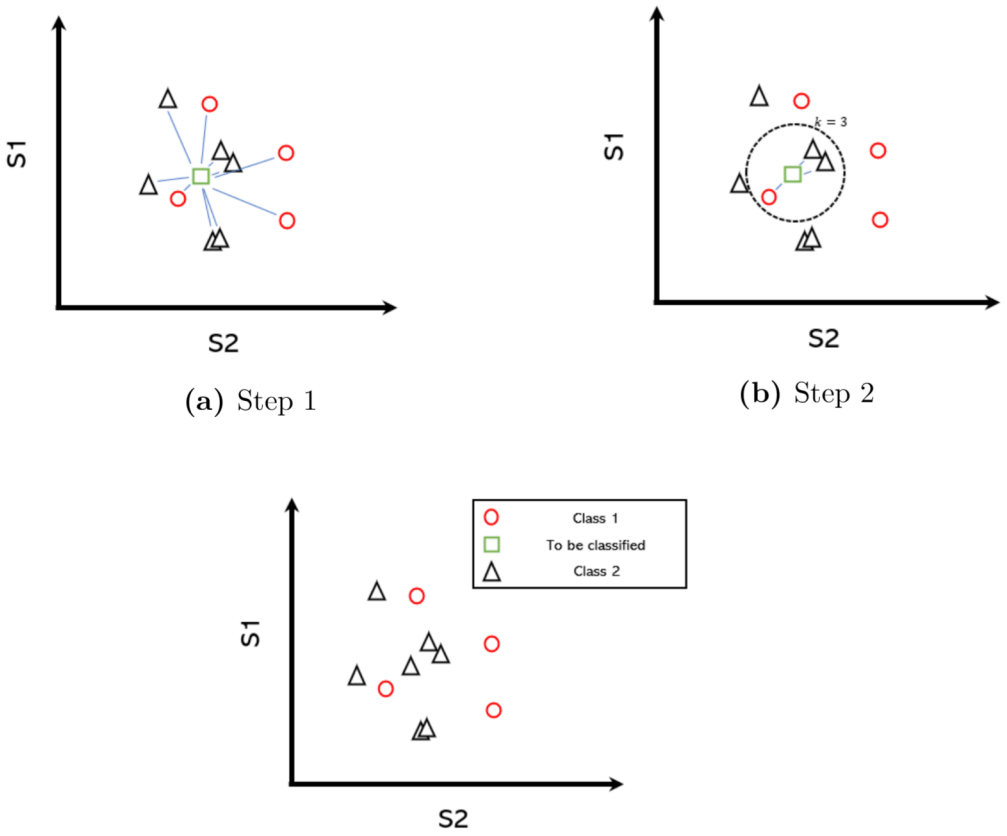
To achieve the algorithm, a distance metric needs to be chosen. The classical metrics that are used here are the Minkowski distances, shown in formula 2.10. Particular cases of these kinds of distances are Manhattan distance when $p = 1$ and Euclidean distance when $p = 2$.

$$d(a, b) = \left(\sum_{i=1}^{n} |a_i - b_i|^p\right)^{\frac{1}{p}} \tag{2.10}$$

The algorithm follows three steps:

1. Calculate distances between the training data and the sample to be classified
2. Sort the training data according to those distances to select the k-nearest samples
3. Classify each sample according to a majority vote between the k-nearest samples chosen in step 2.

The steps are illustrated in figure 2.9.



(a) Step 1

(b) Step 2

(c) Step 3

Figure 2.9: Example of the K-Nearest Neighbours algorithm

2.3.3 Random Forest (RF)

A classical model in biological classification tasks is RF. This model consist in training a lot of Decision Trees (the number is chosen by the user).

To understand how a RF works, it is necessary to understand what is a Decision Tree (DT). A DT is a model in ML that will create decisions on features values. The idea is that after training, when a new data point must be classified, it will go down in the tree and at each stage of the DT, it will either be classified or be submitted to another decision. The reason why DT are not used much in ML is their trends to overfit the training data.

An example of DT is provided in figure 2.10

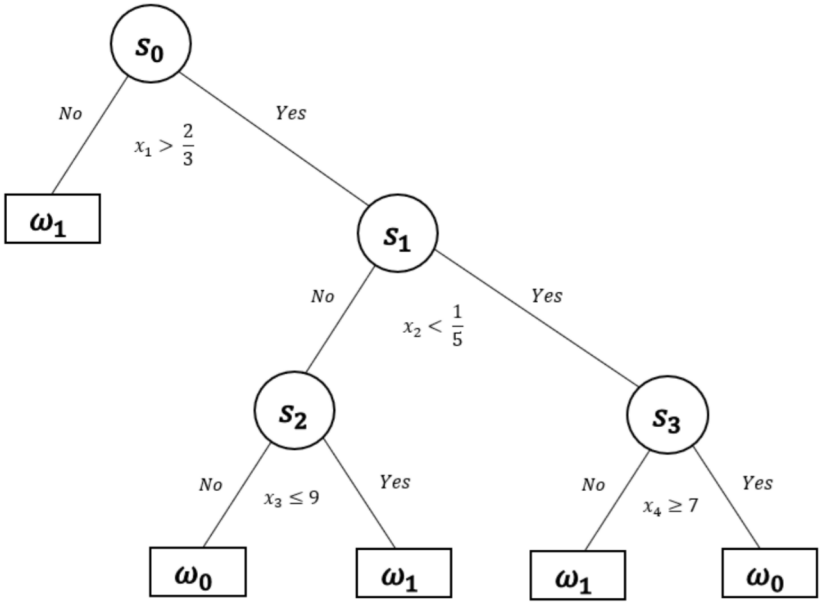


Figure 2.10: Example of Decision Tree where s_i are the nodes where decisions are taken, ω_0 and ω_1 the classes to be predicted.

To build a DT, a recursive algorithm is applied, which, at each step:

1. Create a new node.
2. If the stopping criterion is met (for example, the depth of the tree has reached a fixed maximum), return the model
3. Else, find a split in the dataset that maximizes the impurity decrease, then for each part of this split, go back to step 1.

Here, the optimization problem to find the best model is performed at step 3, where the best split must be found by minimizing the impurity decreases (see formulas 2.11 and 2.12).

$$Tot\_Imp(S^n, \theta) = \frac{k^n_{left}}{k^n_{tot}} * Imp(S^n_{left}(\theta)) + \frac{k^n_{right}}{k^n_{tot}} * Imp(S^n_{right}(\theta)) \tag{2.11}$$

$$\theta^* = \arg\min_{\theta} Tot\_Imp(S^n, \theta) \tag{2.12}$$

with S^n a split S at the node n, k^n_{tot} the total number of samples at node n, k^n_{left} and k^n_{right} the number of samples in the left and right split respectively, θ a given set of parameters and θ^* the one that minimises the impurity.

Two common formulas for the impurity in classification tasks are the Gini impurity (formula 2.13) and the entropy (formula 2.14).

$$Imp(S^n) = \sum_i p_{mi}(1 - p_{mi}) \tag{2.13}$$

$$Imp(S^n) = \sum_i p_{mi} \log(p_{mi}) \tag{2.14}$$

with p_{mi} the proportion of samples from class i in node m.

When the RF algorithm has created a given number of DT's, it will evaluate the class of the data point it wants to classify by passing it through each DT. Then, it will perform a majority vote to determine the class. To train each DT, the algorithm performs two steps. For $k = 1, ..., K$, with K the number of DT's:

1. Sample m training data points called X_k and Y_k, with replacement, from X and Y.
2. Train the k^{th} DT with X_k and Y_k.

Performing this majority vote allows to reduce the drawbacks of the DT and therefore enables for a better classification.

2.3.4 Linear Discriminant Analysis (LDA)

The postulate behind LDA is that the problem is linearly separable. The algorithm aims to retrieve the two classes by fitting Gaussian functions to each classes.

To achieve such a fitting, the LDA algorithm search the probability distribution of each class i by finding the probability distribution that maximises the posterior probability $P(y = i|x)$ of the Bayes theorem (formula 2.15).

$$P(y = i|x) = \frac{P(x|y = i)P(y = i)}{\sum_{c \in classes} P(x|y = c)P(x = c)} \tag{2.15}$$

The conditional posterior $P(x|y = i)$ is modeled as a multivariate Gaussian distribution with density (for a each training sample $x \in \mathbb{R}$:

$$P(x|y = i) = \frac{1}{2\pi^{d/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu_i)^T \Sigma^{-1}(x - \mu_i)\right) \tag{2.16}$$

with d the number of features, Σ the covariance matrix and μ_i the mean of the class i.

To predict the class of a sample, the algorithm seeks the predicted class that maximizes the log of the posterior, as the term $(x - \mu_i)^T \Sigma^{-1}(x - \mu_i)$ is the Mahalanobis distance and computes how close a sample x is to the mean μ_i of the class i, accounting also for the variance of each feature.

$$\begin{aligned} \log P(y = i|x) &= \log P(x|y = i) + \log P(y = i) + Cst \\ &= -\frac{1}{2} \log |\Sigma| - \frac{1}{2}(x - \mu_i)^T \Sigma^{-1}(x - \mu_i) + \log P(y = i) + Cst \end{aligned} \tag{2.17}$$

2.3.5 Extreme Gradient Boosting (XGB)

XGB is the extension of another model called Gradient Boosting. The principle of the algorithm of Gradient Boosting is relatively similar to the RF principle.

The Gradient Boosting is composed of two terms. The Boosting term refers to the combination of multiple weak models (here DT is used as the weak model) to build one strong model. The Gradient term refers to the way the Gradient Boosting works. Indeed, instead of building all the trees at the same time like RF, the Gradient Boosting algorithm will build one tree at each iteration of the Gradient, using the information provided by the previous trees. The number of iterations is the number of estimators chosen by the user. To perform the gradient descent, the algorithm minimizes a loss function. This loss function can be customized, even if, for classification, deviance is the most commonly used. Deviance is described in formula 2.18.

$$D(Y, Y_{pred}) = -\frac{2}{N}\sum_{n=0}^{N-1} Y[n]Y_{pred}[n] - \log\Big(1 + \exp(Y_{pred}[n])\Big) \tag{2.18}$$

with Y the true labels, Y_{pred} the predicted labels and N the number of data points.

Formally, the algorithm initializes the ensemble of model at step 0 M_0 with one model as the constant that minimizes the loss function chosen. Then, at each iteration, it performs three steps:

1. Given the previous ensemble of model, M_{s-1}, and h a candidate tree, compute the best new tree possible.

$$h_s = \arg\min_h \sum_{i=0}^{n-1} l(y_i, M_{s-1}(x_i) + h(x_i)) \tag{2.19}$$

   with $l(y_i, M(x_i))$ being the loss chosen by the user.

2. Th previous calculus needs to have an estimation of $l(y_i, M(x_i) + h(x_i))$. This is done by using Taylor's first order approximation.

$$l(y_i, M_{s-1}(x_i) + h(x_i)) \approx l(y_i, M_{s-1}(x_i)) + h(x_i)\Big[\frac{\partial l(y_i, M(x_i))}{\partial M(x_i)}\Big]_{M=M_{s-1}} \tag{2.20}$$

3. The new tree is added to the ensemble of models.

$$M_s(x) = M_{s-1}(x) + h_s(x) \tag{2.21}$$

At the end of the iterations, to evaluate the class of a new point, the probability for x_i of being in the class i is determined. This probability depends on the loss function used. For example, for the deviance loss, it is estimated by passing the value of the ensemble of model at final step S, M_S, through a sigmoid function.

$$p(y_i = 1|x_i) = \sigma(M_s(x_i)) \tag{2.22}$$

The particularities of the XGB model are not described here, but it mainly consists in a robust and more efficient implementation of a classic Gradient Boosting method.

Chapter 3

Problem statement and objectives of the master thesis

This master thesis aims to extract information from patients' EEG signals in order to predict the occurrence of POD after cardiac surgery. To enable this type of prediction, Machine Learning models will be trained on the data extracted from the patients.

Therefore, the first objective of this master thesis is to build the best models possible given the dataset we have. In this purpose, another objective of this work is to determine the properties that a good model should have in this particular problem and to define specific metrics. In addition, the predictive capacity of a Machine Learning model is limited by the information contained in the chosen features. Therefore, the choice of features used to train the models is also an important part of this work. As seen in the chapter Postoperative Delirium : State of the Art, it exists several features extracted from the EEG signal that have been proven to predict correctly POD state. Therefore, a choice should be made between all those features.

Another objective of this master thesis is to determine where in the EEG signal is the information located. Indeed, several indicators in the literature show that the information is located in some particular bands of frequencies (see section 1.3). In this purpose, having a separation between bands of frequencies is interesting. Another point that we wanted to precise was

About the location of the information, we try to clarify if, as indicated in the literature[36], there was no significant influence in the predictive capacity of a lateralization of the information.

We also try in this master thesis to determine to what extent does the compression of the information given by the different electrodes influence the final prediction capacity. Knowing this would allow in the future to use more additional information without exploding the number of features. To do this, a method considered in this thesis is to generate datasets based on the initial dataset by averaging the information given by several electrodes and thus creating artificial information poles.

Another interesting point that we want to clarify with this work is to what extent the predictive capacity of the information contained in a frequency band is affected by removing a part of the frequency sub-bands.

The last objective we have is to determine if some biomarkers useful for the prediction of POD can be identified that are not well or not at all documented in the literature.

Chapter 4

Dataset description

4.1 Recording context

EEG signals were recorded for 220 patients undergoing cardiac surgery with general anesthesia. The selection of patients undergoing cardiac surgery is due to the fact that this is a major surgery, known to promote the appearance of POD. The recording was realized post-induction of the anesthesia, before and during the operation and the dataset used in this master thesis was the one recorded in intraoperative conditions. In addition to the 32 channels from the cap, 2 electrodes were placed diagonally around the eye in order to record the lateral and horizontal electrical sources coming from the eyes muscular pulses. 2 others were placed onto the arm to record the electrical sources coming from the arm muscular pulses.

The labels were given based on an internationally known scale, the *Confusion Assessment Method for Intensive Care Unit*[49] that was completed by a team of nurses 3 times/day in the ICU, after verifying the level of consciousness of the patient using the *Richmond Agitation and Sedation Scale*. Once the patient leaves the ICU, the following is still performed, and the *Confusion Assessment Method*[50] is completed 2 times/day until the patient leaves the hospital or is transferred to another service.

4.2 Preprocessing

The dataset used in this master thesis was preprocessed upstream. A similar procedure was applied for each patient signal to ensure reproducibility. Here is a summary of what has been applied to the signal recorded by the 32 EEG channels:

1. Remove the slow drift due to direct current component and remove linear trend from the data (center the signals around 0) by removing the mean of the signals.

2. Create a unique reference from the two ocular electrodes (EOG) and a unique reference from the two arm electrodes (EMG). Both will be used to remove artifacts sources from the ICA matrix.

3. Interpolate values of the signals for each defective channel by using neighbouring electrodes. Replace the signal of those channels by this interpolation.
4. Get the frequency domain of the data by using a DFT applying the FFT algorithm.
5. Apply a finite impulse response (FIR) filter using Hanning windows with a low cutoff frequency of 0.5Hz and a high frequency of 47Hz.
6. Segment the signal into chunks of 5 seconds with an interval of 5 seconds between the onset of two successive chunks and then remove visually chunks presenting artifacts.
7. Compute the independent component analysis (ICA) matrix on a signal composed of the clean chunks.
8. Remove sources components considered as artifacts visually (typically muscles, eyes and heartbeats) with the assistance of MARA[51] tool and by comparing ICA components with the two reference computed earlier (EMG and EOG).
9. Apply the previous ICA matrix on the initial signal.
10. Segment the signal acquired after into chunks of 10 seconds with an interval of 1 seconds between the onset of two successive chunks.
11. Reject epochs visually if artifacts are still visible.

The signals obtained after this procedure were the raw signals received for this master thesis. As they were chunked into signals of 10 seconds with an interval of 1 second between the onset of two successive chunks, a reconstruction of the total signal was performed for each patient.

4.3 Groups of electrodes used

In order to distinguish between different types of areas for the brain, to determine the influence of those areas on the predictions of the machine learning models and finally to reduce the set of possible features, regrouping electrodes by averaging their signals has been performed. In addition to the dataset composed of all the electrodes, those different types of groupings have been chosen based on specificity of the prediction task of this thesis or on classical groups used in the literature.

4.3.1 Inspired from literature

In this section, both groups are inspired from commonly used groups of electrodes, corresponding to anatomical and functional separation of the cortex.

Functional

In this group of electrodes, three channels have been determined, each of them representing a function of the brain. This separation in regions of interest is inspired from the literature[52], where it is used to facilitate topographic analysis of EEG bands for example. The 6 regions of interest of this study have been reduced to three as we don't consider lateralization in this dataset.

The first group is the fronto-central one. It is responsible for cognitive functions, like a part of the language. The second group is the sensory-motor one. As the name suggests, it is responsible for the sensitive and the motor system of the body. The last one is the parieto-occipital group, which is responsible for vision and a part of the cognition, like language, mathematics, orientation in space, ... The separation of the electrodes can be seen in figure 4.1. In the rest of this master thesis, this features set will be referred to as the 3 poles set.

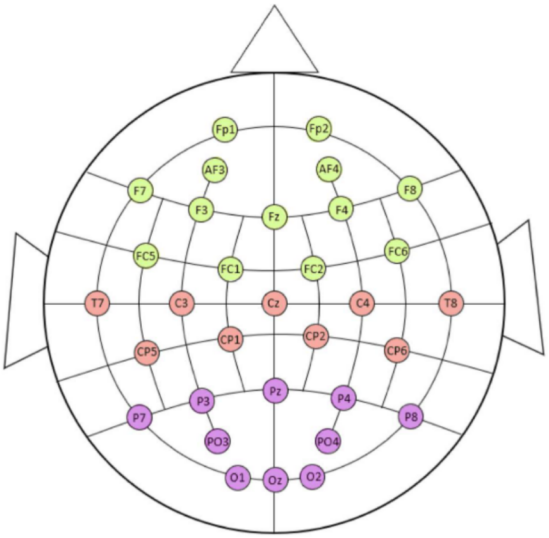


Figure 4.1: Functional groups of electrodes

Anatomic

In those groups of electrodes, four types of channels have been determined, based on four well known lobes of the cortex from an anatomical point of view. The idea of this separation has been inspired from a study of Yang et al.[53] where they used similar anatomical regions of interest in a Machine Learning purpose.

The first group corresponds to the frontal lobe, the second to the central lobe, the third to the parietal lobe and the third to the occipital lobe. Some electrodes situated at the border of two lobes have been used on two groups, as seen in figure 4.2. In the rest of this master thesis, this features set will be referred to as the 4 poles set.

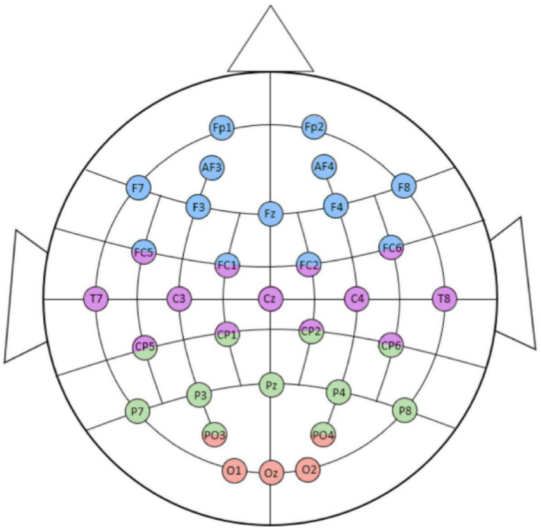


Figure 4.2: Anatomical groups of electrodes

4.3.2 Specific to the task

In this section, all the groups are related to specific knowledge about the EEG signals of patients under anesthesia who will undergo POD.

Anteriorization of alpha rhythm

In those groups of electrodes, nine types of channels have been determined. Our idea behind the choice of those particular groups is due to the Anteriorization of the alpha rhythms, that appears when a patient is under anesthesia, and seems to be attenuated for POD as already described in Chapter 1. To highlight this phenomenon, each line of electrodes from the cap have been regrouped in one mean as we can observe in figure 4.3. In the rest of this master thesis, this features set will be referred to as the 9 poles set.

No influence from lateralization

Those groups of electrodes are composed of five types of channels. The objective of the choice of dataset represented in the figure 4.4 is to show the difference of prediction capacity between left areas (red, green, blue and yellow electrodes), and right areas (red, green, blue and violet electrodes). In the rest of this master thesis, this features set will be referred to as the 5 poles set.

A study using similar groups of electrodes has been found[54] after having chosen to make use of them. This confirms that this separation could be useful to analyze.

Frontal alpha diminution

In this particular separation, only electrodes located in the frontal part are considered, as one of the characteristics of the patient undergoing POD is the frontal alpha rhythm

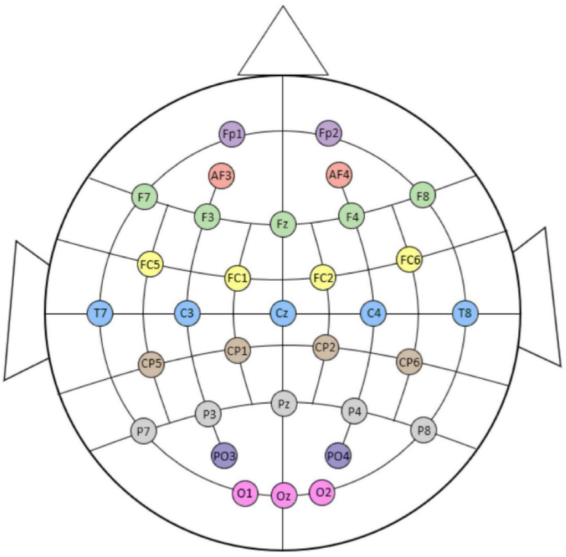


Figure 4.3: Groups of electrodes highlighting the anteriorization of the alpha rhythms

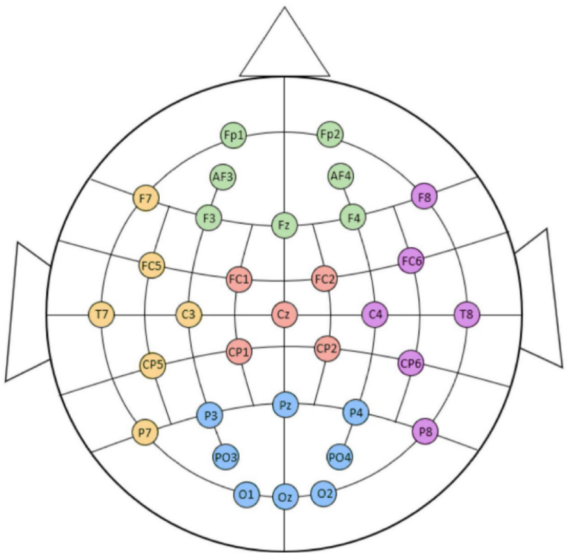


Figure 4.4: Groups of electrodes comparing influence of laterality versus longitudinality

diminution. This separation has been chosen by us because of the clues presented in the litterature for frontal alpha diminution (see section 1.4). No groups are created but a lot of electrodes are removed from the original dataset as we can observe in figure 4.5. The blue electrodes correspond to those who are kept, each as a feature. In the rest of this master thesis, this features set will be referred to as the 13 poles set.

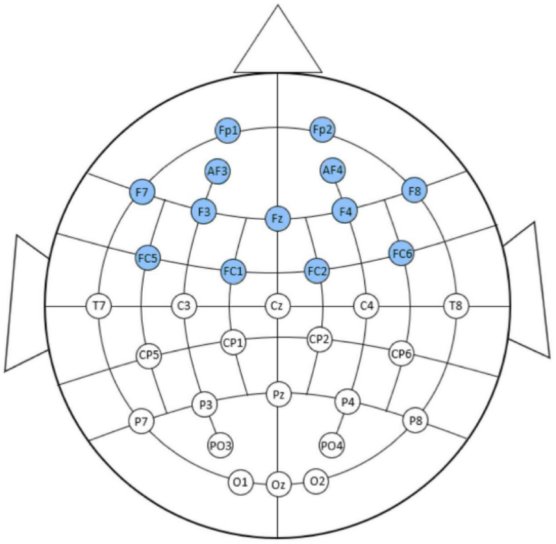


Figure 4.5: Frontal electrodes kept to show the importance of the alpha diminution in the prediction

4.4 Sub-bands analysis

In addition to classical bands of frequencies (see section 2.2.1) used in the literature, an interesting analysis to determine the location of the information in the frequency domain is to segment those bands into sub-bands. In this master thesis, the sub-bands used are only concerning Alpha and Beta bands, for several reasons described in detail in section 6.5. Both bands have been separated into 4 sub-bands of equivalent sizes (see table 4.1).

	sub-band 1 [Hz]	**sub-band 2 [Hz]**	**sub-band 3 [Hz]**	**sub-band 4 [Hz]**
Alpha	[8, 9[	[9, 10[	[10, 11[	[11, 12]
Beta	[12, 16.5[	[16.5, 21[	[21, 24.5[	[24.5, 30]

Table 4.1: Sub-bands separation for Alpha and Beta bands of frequencies.

Chapter 5

Methodology

5.1 Feature choice and extraction

To build models in ML, features need to be chosen and extracted from the original EEG signal. In a paper from Koch et Al.[35], the influence of absolute alpha power in the prediction of cognitive impairment has been highlighted. Therefore, to follow one of the objectives of this master thesis (see chapter Problem Statement and objectives of this master thesis), we decided to take as feature the absolute power of bands of frequencies. To perform this extraction, several steps have been followed.

First, as described in section 4.3, as the signals are separated in chunks of 10 seconds that overlap over 9 seconds, the original signal is reconstructed by merging the first second of each signal. The last is nonetheless completely added.

Once the original signal is reconstructed, different datasets can be obtained by averaging groups of electrodes (see 4.3).

For each dataset, the PSD curve of the signal is then obtained using the method described in section 2.2.1. The average powers are obtained from the Simpson integration of this curve over bounds corresponding either to the five physiological bands described in section or to the sub-bands alpha/beta (see section 4.4).

After this step, a random stratified separation of the datasets between a training and a testing set is performed. It allows a performance assessment after the complete build of the model. We chose to stratify the test set by conserving a similar distribution of the labels in both subsets of the data. Indeed, as the dataset is imbalanced and the hypothesis was made that it is in same proportions than expected for any cardiac operation, we need to conserve those proportions to reflect real conditions in both training and testing sets.

5.2 First look at the data

In order to have a first idea of the distribution of the features for the biggest dataset, density plots have been realized (see figures 5.1 and 5.2). After a quick look at the large majority of them it can be assumed that distribution of both classes are Poisson distributions. As the number of samples is large in terms of statistics (# of samples > 100), the law of the large numbers allows us to approximate this Poisson distribution by a Normal distribution. Therefore to have a first idea about the location of the information in terms of distribution of the data, an independent Student's t-Test has been realized. The p-values between both distributions have been computed to determine which of the features have the more significant changes based on this dataset.

	Alpha	**Beta**	**Gamma**	**Delta**	**Theta**
Fp1	0.011*	0.002**	0.43	0.165	0.053
Fp2	0.015*	0.003**	0.341	0.195	0.057
AF3	0.045*	0.003**	0.232	0.236	0.109
AF4	0.05	0.004**	0.318	0.315	0.136
Fz	0.066	0.003**	0.097	0.491	0.236
F3	0.062	0.002**	0.057	0.344	0.14
F4	0.088	0.005*	0.291	0.547	0.227
F7	0.004**	0.001**	0.362	0.288	0.038*
F8	0.005*	0.0008**	0.321	0.273	0.037*
FC1	0.114	0.026*	0.458	0.673	0.609
FC2	0.182	0.038*	0.409	0.542	0.679
FC5	0.047*	0.002**	0.991	0.861	0.172
FC6	0.156	0.014*	0.518	0.83	0.425
Cz	0.082	0.02*	0.341	0.416	0.153
C3	0.051	0.009*	0.129	0.663	0.144
C4	0.167	0.031*	0.517	0.875	0.338
T7	0.142	0.004**	0.289	0.722	0.244
T8	0.007*	0.0009**	0.117	0.351	0.059
CP1	0.031*	0.011*	0.568	0.366	0.135
CP2	0.013*	0.004**	0.421	0.391	0.101
CP5	0.069	0.002**	0.164	0.453	0.175
CP6	0.014*	0.002**	0.471	0.787	0.137
Pz	0.025*	0.003**	0.494	0.253	0.065
P3	0.04*	0.002**	0.254	0.375	0.167
P4	0.013*	0.001**	0.254	0.335	0.067
P7	0.083	0.002**	0.226	0.379	0.227
P8	0.009*	0.002**	0.233	0.479	0.081
PO3	0.1	0.001**	0.24	0.408	0.211
PO4	0.041*	0.001**	0.196	0.447	0.14
Oz	0.097	0.003**	0.785	0.615	0.342
O1	0.155	0.002**	0.33	0.628	0.342

O2	0.072	0.005**	0.768	0.69	0.259

Table 5.1: P-values of each feature in respect to both classes. Significance levels: * = significant ($\alpha < 0.05$), ** = very significant ($\alpha < 0.005$)

While analyzing the data from this table, some characteristics must be mentioned:

First, there are changes in the frontal alpha rhythms, which confirms what the literature described (see section 1.3). In addition to those variations in the frontal alpha rhythms, it seems that there are also variations in the parietal and occipital parts of the brain for this alpha rhythms.

Looking at the p-values, it is really interesting to see that variations of the beta rhythms are clearly identified by the Student's t-Test used with this purpose. Almost all electrodes show a very significant difference between the beta rhythms of both classes.

Finally for the three other bands of frequency, the p-values do not let us conclude anything about the changes.

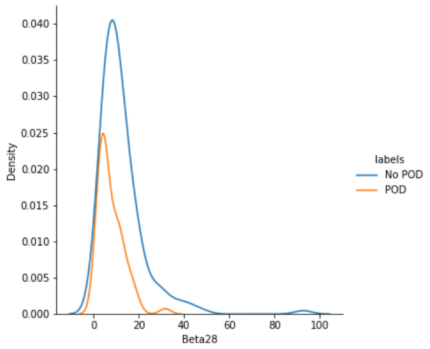
Figure 5.1 shows the density plots using the density kernel estimation technique of the 5 more significant features in terms of p-values (the plots for all features are in Appendix A). We can see on those plots that the peak of density of both distributions are not at the same places. If we compare this with features having a non significant p-value after their t-test, we observe that there is no left-shift of the peak of density for features with low p-values. An example of this is provided in figure 5.2.

5.3 Metrics choice

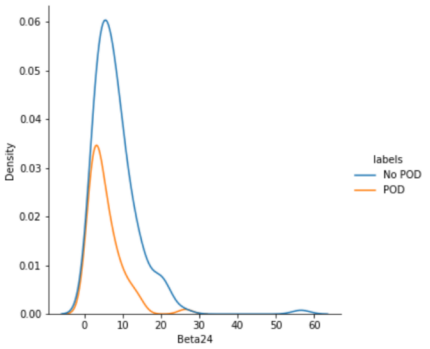
An important decision concerns the metric used to train the models and the metrics used to evaluate the performance of the models on the test set. A metric in ML is a function measuring how well a model is predicting. There are several types of metrics, each having pros and cons, with a large number of them calculated based on the confusion matrix (see figure 5.3). The objective is therefore to identify, for the task of this master thesis, which metric is the best to train the model and which is the best to attest to the quality of the model.

5.3.1 Training metric

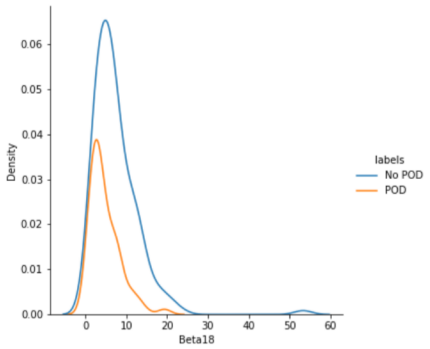
Determining the best training metric is closely related to the objectives of the task. Here, among the objectives, one is about creating a model that correctly identifies all the patients that will have a POD. In other words, it means that the model aims to predict a minimum of false negatives. The metric achieving this is known as the **Recall**. It is defined mathematically by formula 5.1.



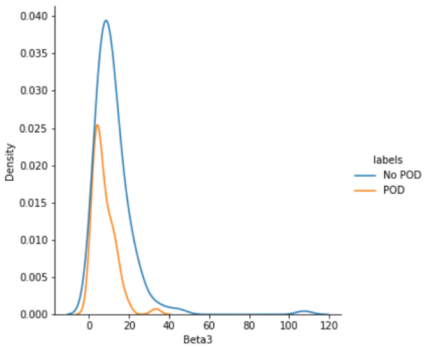
(a) Beta rhythms, Electrode AF4



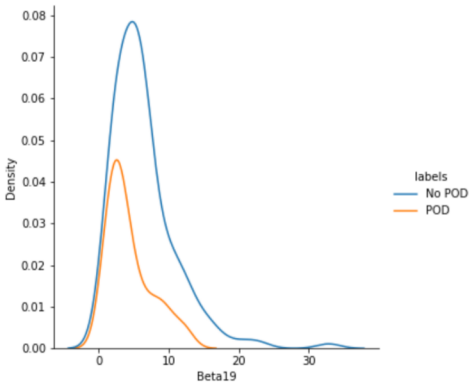
(b) Beta rhythms, Electrode FC6



(c) Beta rhythms, Electrode P4



(d) Beta rhythms, Electrode F3



(e) Beta rhythms, Electrode P8

Figure 5.1: 5 more significant features in terms of p-value differences between both distributions.

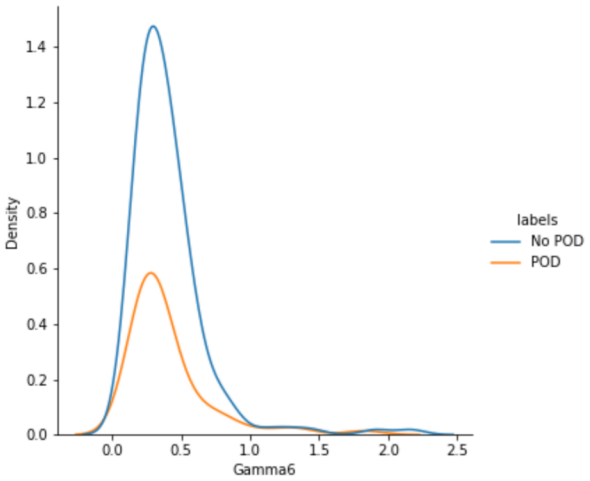


Figure 5.2: Gamma rhythms, Electrode 6: lowest p-value of the dataset

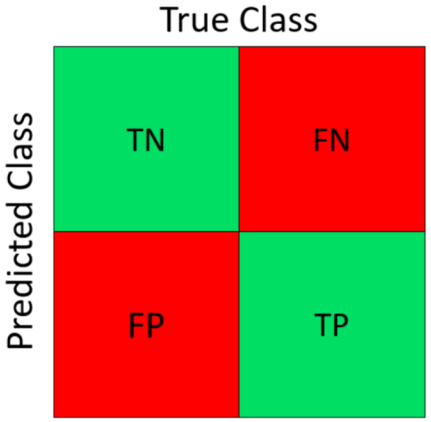


Figure 5.3: Confusion matrix. T and F stand for True and False, P and N stand for Positive and Negative.

$$Recall = \frac{TP}{TP + FN} \tag{5.1}$$

Another objective of the task is to avoid creating a model that leads to a small fraction of patients predicted as positive while they would not have had POD. In other words, it means that the model aims to predict a minimum of false positives. There is a metric achieving this. It is called the **Precision** and it's definition is given in formula 5.2.

$$Precision = \frac{TP}{TP + FP} \tag{5.2}$$

To obtain the best training scores given both metrics, there is a score combining each of them. It is called the f_1-score. This f_1-score is defined as the harmonic mean of both metrics (see formula 5.3). It is maximal when the combination of both metrics is maximum, and minimal when at least one of both metrics is minimal.

$$f_1 score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{5.3}$$

A generalization of this score, called f_β-score (see formula 5.4) is commonly used to change the weight of one metric in comparison to the other. The β value is 1 for the f_1-score. If $\beta > 1$, then more weight is given to the **Recall**, while if $\beta < 1$ more weight is given to the **Precision**.

$$f_\beta score = (1 + \beta^2) * \frac{Precision * Recall}{\beta^2 Precision + Recall} \tag{5.4}$$

Even if both metrics are important in this task, **Recall** is the most important one as predicting false positives is less important than predicting false negatives. In other words, taking more precautions than needed for a given patient is less important than not enough precautions. Therefore, the metric used for the training is this score with a $\beta = 1.5$ to give a little more weight to the **Recall**.

5.3.2 Evaluation metrics

Once the model is trained, a performance assessment will be needed to testify to the quality of the model. To do so, in addition to the training metrics that are **Precision** and **Recall**, the **Specificity** of the model is computed (see formula 5.5). This metric gives the proportion of non-diseased correctly classified.

$$Specificity = \frac{TN}{TN + FP} \tag{5.5}$$

It is also important to have a metric assessing the global performance of the model. The Area Under the Curve (AUC) of the Receiver Operating Characteristic (ROC) curve (see figure 5.4) is a metric widely used in literature for this purpose nowadays. Therefore, this metric will assess the global performance of the model while other metrics will be used to assess specific wanted characteristics. The ROC curve is built by comparing the False Positive Rate, and the True Positive Rate which is in fact the **Recall**. This is possible

for probabilistic classifiers which return a score or probability that reflects the degree to which an instance belongs to a class. The ROC curve is then computed by varying the threshold used for the score.

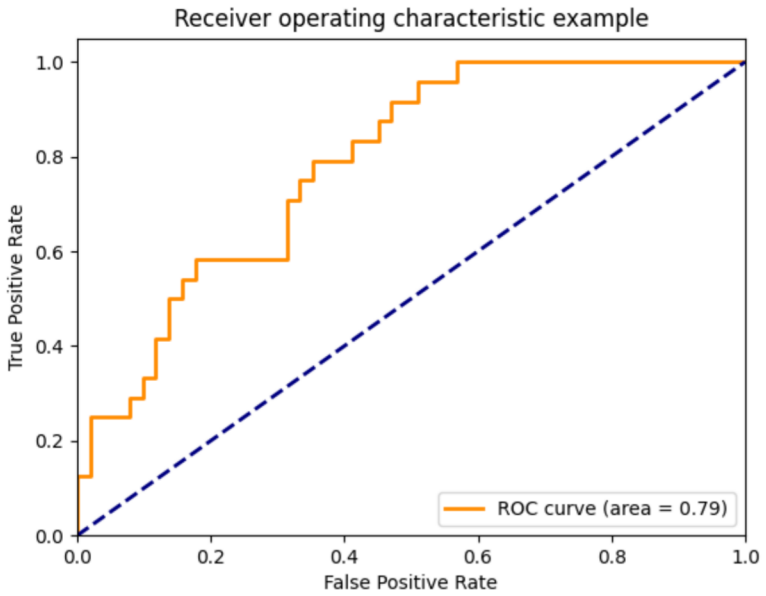


Figure 5.4: AUC-ROC metric example

To see if our models perform better than random, we compare our metrics from the confusion matrix (for the AUC score, the principle is that the score is better than random if larger than 0.5) with to two random estimators called the Weighted Guess Classifier and the Random Guess Classifier from the training set proportions. Weighted Guess Classifier assign randomly $x * 100\%$ of the patients to be predicted to the POD class, and $(1 - x) * 100\%$ to the no-POD class (x being the proportion of patients suffering from POD in the original dataset). Random Guess Classifier randomly assigns 50% of the patients to be predicted to each class. In the training set (80% of the dataset), we have 51 of the 176 patients labelled with POD. A probabilistic estimation of the confusion matrices for both classifiers can therefore be computed and is given in tables 5.2 and 5.3

		Predicted	
		no-POD	**POD**
Actual	**no-POD**	62.5	62.5
	POD	25.5	25.5

Table 5.2: Random Guess Classifier confusion matrix

Based on those confusion matrices, the values of theoretical **Recall**, **Precision** and **Specificity** can be computed and are summarized in table 5.4.

		Predicted	
		no-POD	POD
Actual	no-POD	88.8	36.2
	POD	36.2	14.8

Table 5.3: Weighted Guess Classifier confusion matrix

	Recall	Precision	Specificity	F_β
Random Guess	0.5	0.29	0.5	0.41
Weighted Guess	0.29	0.29	0.71	0.29

Table 5.4: Theoretical scores used as baseline for the performance assessment.

5.4 Feature selection

One of the biggest challenges when doing ML is called the *Curse of Dimensionality*. It occurs when the number of dimensions of the problem (the number of features) is too large in comparison to the number of data values (here patients) available. The problem is that increasing the number of features first increases the performances of the model but after a certain number of features it begins to decrease the performances. In this kind of situation, it can be useful to perform a feature selection.

In this task, only one dataset could really suffer from this *Curse of Dimensionality*. The "big" dataset, containing the data of each of the 32 electrodes for each of the 5 bands of frequency (160 features) has indeed 176 patients in its training set. The ratio of features by patient is close to 1. Two different types of feature selection have been applied to it. For each of the selections, the objective is to be close to a ratio of 1 feature for 3 data values after the feature selection.

5.4.1 Wrapper method using LDA

A classical way to select features in ML is by getting a subset of k features that performs the best for a given model. Here, the strategy followed is to first choose a random subset of 50 features (ratio is about 1/3 as wanted), then to compute the performances of a basic LDA model on it (with a Singular Value Decomposition solver and a tolerance for significance of 0.0001), store those performances, and do the same operations again until the score of all the subsets of k features has been computed. The subset of features that performed the best is finally selected (see figure 5.5).

5.4.2 Manual selection based on t-Test results

The second type of feature selection used in this master thesis is a manual selection of some of the features. In the Student's t-Test results described in section 5.2, the large majority of the alpha and beta rhythms seemed to be relevant whereas the other bands seemed to be useless. Therefore, it has been chosen to keep only those two bands, to see if

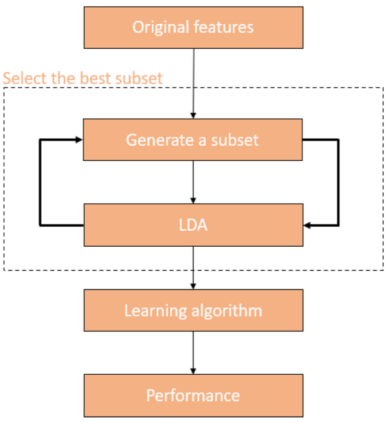


Figure 5.5: Wrapper method using LDA

using it could lead to better performance than keeping all the bands. It allowed to keep features, which is close to the 1/3 ratio desired.

5.4.3 Filtering method using ANOVA results

The last type of feature selection used in this master thesis is a filtering method using a ANOVA F-score to determine the 50 best features. This method is a bit different from the manual selection as:

1. It is an analysis of the variance changes instead of an analysis of the mean changes.
2. In the manual selection, we decide to keep all the alpha and beta because a large majority of them seems to be significant. Here, only the 50 most significant features are kept, meaning that there are no arbitrary choices.

Figure 5.6 describes the steps to perform this feature selection.

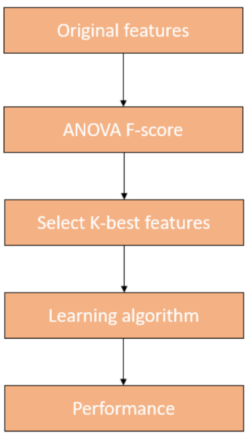


Figure 5.6: Filtering method using ANOVA F-scores

5.5 Model selection

Five types of models have been selected in order to compare them and determine which was the best predictor for patients that would likely suffer from POD. The mathematical characteristics of the selected models have already been discussed in section 2.3. In this section we will discuss the hyperparameters optimized to select the best model of each type for the datasets considered as well as the reasons for using them on those given datasets.

5.5.1 SVM with RBF kernel

The reasons we used this model are its simplicity of use, its frequent usage in classification tasks and its efficiency for small datasets. In addition in this case it does generally not suffer condition of overfitting and generalized therefore well.

The main reason for its simplicity of use is the low number of parameters when using a RBF kernel. Two hyperparameters (described in section 2.3.1) are optimized to achieve good performances:

- **Gamma :** this parameter is tried for 10^{-k} with $k \in [3, 4, 5, 6, 7]$.
- **C :** this parameter is tried for 10^{k} with $k \in [-3, -2, -1, 0, 1, 2, 3]$.

5.5.2 KNN

As this model is simple, it's quite infrequent to achieve big performances with it. This is the reason of its use. It makes a good baseline of the reachable performance for other models.

Three hyperparameters are optimized to achieve good performances:

- **Number of neighbours :** this parameter (described in section 2.3.2) is tried for [5,10,20,30,40,50].
- **Weights :** this parameter can be either set to *uniform* or *distance*. If it is set to *uniform*, then each neighbour will have the same importance. If it is set to *distance*, then closer neighbours will have a greater weight on the vote to classify a new sample.
- **p :** this parameter is the power in the Minkowski distance. It is tried for [1,2,3,4,5,6].

5.5.3 RF

This model has been chosen for two main reasons: its efficiency on small datasets and its frequent usage and efficiency on medical classification tasks.

It is complex with a lot of hyperparameters that can be modified. In this master thesis, five have been considered:

- **Maximum depth :** this parameter is the maximum depth of each DT of the forest. It is tried for [2,5,8,15].
- **Maximum number of features :** this parameter is the maximum number of features a DT can use for each split. The values tried are [3,8,"sqrt"], with sqrt being the square root of the total number of features.
- **Minimum number of samples in a leaf :** this parameter is the minimal number of samples required to be at a leaf node. It is used to control the overfitting of the model. The values tried are [3,4].
- **Minimum number of samples by split :** this parameter allows to control the minimal number of samples required to perform a split. It also controls the overfitting of the model. The values tried are [8,12].
- **Number of estimators** this parameter is the number of DT that should be trained to perform the majority vote (see section 2.3.3). The values tried are [50,100,200,300,400].

5.5.4 LDA

The LDA model is a linear model (see section 2.3.4) which made it the main reason of our choice. Indeed, as all other models are non-linear, it was interesting to have at least one linear model as the data could be linearly separable. The other reason is its simplicity of use as it only requires two hyperparameters.

- **Solver :** the solvers used to estimate the probabilities used in the model. There are three solvers which are tested here. The Singular Value Decomposition (SVD), the Least Square solution and the Eigenvalues Decomposition.
- **Tolerance :** this parameter is only used in association with the SVD solver. It is the absolute threshold for a singular value to be considered significant. The values tried are [0.0001,0.001,0.01,0.1,0.5].

5.5.5 XGB

The XGB model has been chosen mainly for the same reasons as the RF. It performs generally well on small datasets and it has shown good performances in similar classification tasks.

As for the RF, this model is also quite complex and has therefore a lot of hyperparameters to modify. Here, 6 have been chosen.

- **Loss function :** as described in section 2.3.5, the XGB can be used with several loss functions. Here, this parameter is set to binomial negative log-likelihood (also called the deviance).

- **Learning rate :** after each boosting step, we can directly get the weight of new features and the learning rate shrinks the feature weights to prevent overfitting and make the boosting process more conservative. The values tried are [0.001,0.01,0.1].
- **Number of estimators :** this parameter represents the number of boosting stages to process. The values tried are [100, 150, 200, 250, 300].
- **Maximum depth :** this is the same as for RF. The values tried are [2,5,9,15].
- **Minimum child weight :** this parameter tends to reduce overfitting by limiting the minimum sum of instance weight for a child. If the tree partition step results in a leaf node with the sum of instance weight inferior to this minimum, the tree is no longer partitioned. The values tried are [2, 4, 6].
- **Regularization alpha :** this parameter is set to reduce overfitting as it is the alpha value of the L1 regularization term. The values tried are [4, 8, 12].

5.6 Cross-validation

In ML, a model is trained by splitting the dataset into a training and a testing set. Here, the split performed between training set and test set was a 80% - 20%. This split aims to determine the performances of a model after it has been trained. Those 2 sets are however not enough to properly train a ML model. Indeed there are several times in the process of building our models where we want to determine the performance of it. For example, if using a wrapper for the feature selection, we want to know which subset of features gives the best performances on our model. Another example is that in addition to the parameters, the model will train by using the training set. There are parameters, called hyperparameters, that must be chosen by hand. To determine for which hyperparameters a model performs the best, validating the performance of each trained model is useful.

Therefore, the use of a third set, the validation set, is needed in order to assess the performance during the building process. There are three problems with the use of only one validation set.

- By using the validation data to tune our model, we optimize the model hyperparameters (and the selected features from the wrapper) specifically for this validation set, which imply that we overfit to this validation set.
- By using a validation set, we reduce the amount of data that will be used for the training.
- The confidence in the score given by the validation set will be poor, as it is a unique score and not the mean of several scores for which we could compute the variance.

Therefore, instead of using a unique validation set, a technique called K-Fold can be used.

K-Fold idea is to split the data into k blocks, to train the model on each combination of $k-1$ blocks and validate the performance on each remaining block. The mean and the

standard deviation of the k predictions are then computed to get an idea of the stability of the model and of the probable performances on the full dataset. Then, when those performances are acceptable, the model is trained on the fully reconstructed training set. It allows to increase the confidence in the final score estimation, to reduce the overfitting problem of using a unique validation set and to train the model on the full training set.

An illustration of the K-Fold strategy is provided at figure 5.7 with the validation chunk for each fold highlighted. To ensure that the model will not learn from the disposition of the data inside the training set, a shuffle is performed before splitting the training set in chunks.

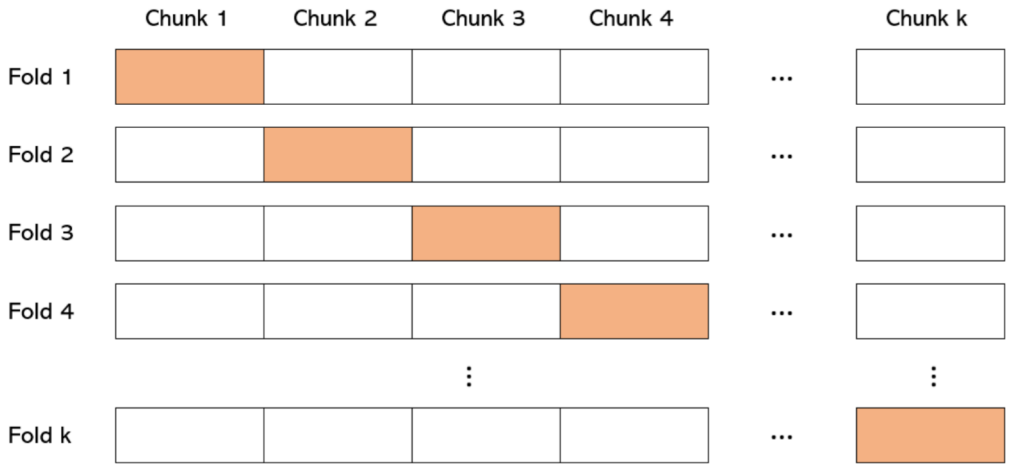


Figure 5.7: K-Fold strategy on the training set.

As the number of patients having POD is significantly lower than the number of patients not having it, an oversampling using SMOTE (see section 2.2.3) is performed for each fold to ensure the balance of the dataset.

5.7 Stability of the models

To assess the stability of the models, several measures have been taken. This section summarizes those measures.

5.7.1 Convergence Curve

A convergence curve shows the performances of the trained model in terms of training scores (score of the model on the training set) and validation scores (score of the model on the validation set) with regards to several numbers of training samples. The goal of this comparison is to determine if an overfitting is plausible by checking if both scores have converged at the same proportion of training/validation as the proportion of training/testing used (80% - 20%). This is a good measure of the stability of prediction of a given model. Indeed, if a model does overfit the predictions of the model on new data are

quite unpredictable.

Figure 5.8 shows an example of a convergence curve. The vertical line represents the proportion, showing here a significant difference between the training scores and the validation scores. The areas around both lines contain 95% of the scores given the cross-validation. Here, an overfitting is quite probable, looking at the difference between both curves at the blue line.

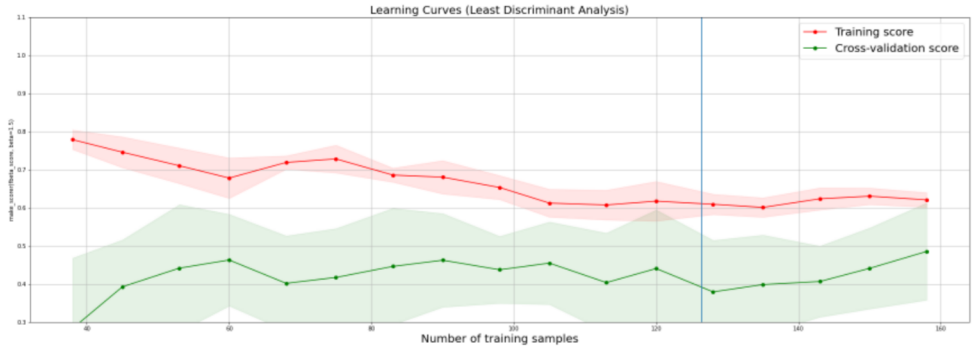


Figure 5.8: Example of Convergence Curve

5.7.2 Confidence Interval

To estimate the potential stability of the model on a new dataset, building the confidence interval (CI) of the cross-validation scores obtained while optimizing the hyperparameters of the model is a useful method. Indeed, if the convergence curve presented before can assess that the model doesn't overfit, than the prediction scores should be close to the validation scores. Therefore, if the CI for validation sets are wide, the model performance assessment risks being inaccurate and the model predictions will be unstable. The CI used here is the normal CI. It can be used if the distribution can be considered as normal with the proportion of each class sufficiently larger than 0 ($\hat{p}_i \cdot n \geq 5$ with $\hat{p}_i$ the proportion of class i and n the number of samples). As an oversampling is performed, both classes have the same proportion (50%). In addition, the number of folds for the cross-validation is 10. Therefore, the 95% normal CI can be computed (see formula 5.6).

$$CI = \mu \pm 1.96\frac{\sigma}{\sqrt{n}} \tag{5.6}$$

with μ the mean of the scores, σ the standard deviation and n the number of scores collected.

5.7.3 Variation of hyperparameters

The last method to assess the stability of a model concerns the variation of the hyperparameters of a given model. Indeed, to determine if a model will not vary a lot in the predictions, a method is to look at how much scores vary when introducing little variations

of the hyperparameters.

If the variation is important then the model is quite unstable and the predictions risk being lower than the validation scores. The good performances of the model on the validation set might therefore be only due to fortune which causes instability.

Chapter 6

Results

6.1 Features selected

In this section, all features referenced come from the 31 channels set of features. As described in the methodology, three different methods have been used and their results are summarized in the following sections.

6.1.1 Selected by hand

Table 6.1 summarizes the features selected by hand. These are all 31 electrodes from the Alpha and Beta bands.

	Alpha	Beta	Gamma	Delta	Theta
Frontal pole	Fp1, Fp2	Fp1, Fp2	/	/	/
Anterio-Frontal	AF3, AF4	AF3, AF4	/	/	/
Frontal	Fz, F3, F4, F7, F8	Fz, F3, F4, F7, F8	/	/	/
Fronto-Central	FC1, FC2, FC5, FC6	FC1, FC2, FC5, FC6	/	/	/
Temporal	T7, T8	T7, T8	/	/	/
Central	Cz, C3, C4	Cz, C3, C4	/	/	/
Centro-Parietal	CP1, CP2, CP5, CP6	CP1, CP2, CP5, CP6	/	/	/
Parietal	Pz, P3, P4, P7, P8	Pz, P3, P4, P7, P8	/	/	/
Parieto-Occipital	PO3, PO4	PO3, PO4	/	/	/
Occipital	Oz, O1, O2	Oz, O1, O2	/	/	/

Table 6.1: Features selected by hand

6.1.2 Wrapper

Table 6.2 summarizes the 50 features selected as those performing the best on a basic LDA.

	Alpha	Beta	Gamma	Delta	Theta
Frontal pole	/	Fp1, Fp2	Fp2	/	/
Anterio-Frontal	/	/	AF4	/	/
Frontal	Fz, F3, F4, F7	F3, F4	Fz, F3	/	F7
Fronto-Central	FC1, FC2	FC1, FC2	FC5	FC5	FC6
Temporal	T7, T8	/	/	/	T7
Central	/	/	C4	C4	Cz, C4
Centro-Parietal	/	CP2	CP1, CP2, CP5, CP6	/	CP1, CP2
Parietal	P4, P7, P8	P3, P4, P8	/	/	P8
Parieto-Occipital	/	PO3	PO3	/	PO4
Occipital	/	Oz, O2	O1	/	Oz, O1, O2

Table 6.2: Features selected by Wrapper method

6.1.3 Filter with ANOVA

Table 6.3 summarizes the features selected by a filter using an ANOVA test, keeping the 50 values with the lowest p-values.

	Alpha	Beta	Gamma	Delta	Theta
Frontal pole	Fp1, Fp2	Fp1, Fp2	/	/	Fp1, Fp2
Anterio-Frontal	/	AF3, AF4	/	/	/
Frontal	F7, F8	Fz, F3, F4, F7, F8	F3	/	F7, F8
Fronto-Central	/	FC1, FC2, FC5, FC6	/	/	/
Temporal	T8	T7, T8	/	/	T8
Central	/	Cz, C3, C4	/	/	
Centro-Parietal	CP1, CP2, CP6	CP1, CP2, CP5, CP6	/	/	/
Parietal	Pz, P4, P8	Pz, P3, P4, P7, P8	/	/	P4
Parieto-Occipital	/	PO3, PO4	/	/	
Occipital	/	Oz, O1, O2	/	/	/

Table 6.3: Features selected by Filter method

6.2 Models

To present the results, for each type of model, the confusion matrix, the metrics values, the best hyperparameters and the CIs of the cross-validation scores (here these are F_β scores) will be summarized into tables for each set of features. The models presented in this section have been built using oversampled data using SMOTE. The ROC curves are shown for the best of each type of model. The other curves are presented in Appendix B.

6.2.1 SVM

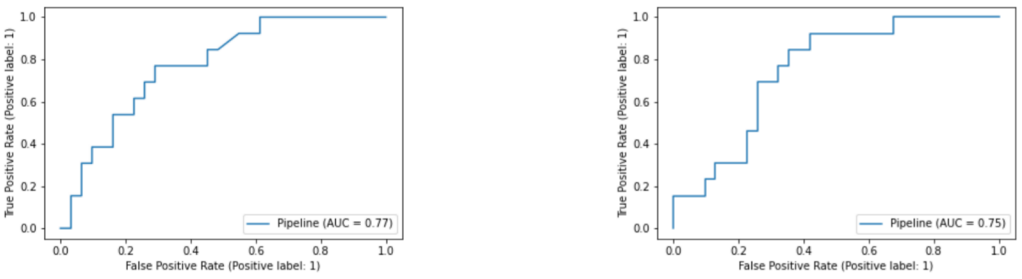
Table 6.4 shows the results for the SVM models. Several sets have a recall of 1.0, a specificity of 0.0, a low precision and a low AUC score. The 0.0 scores in specificity are due to the metric used for the training. Indeed, as a F_β scoring function with $\beta = 1.5$ is used, the recall score have a bigger weight than the precision, which tends to allow the model to always predict the POD class, as it will give them a big recall and a relatively big F_β scoring. Those models are poor in terms of variety as a basic "always POD" model could do the same.

The 13 poles model is the therefore best model here, if the objective is to have the model that performs better in a global fashion. If recall is the priority, with decent values for other metrics, then the 5 poles model should probably be chosen.

Concerning the Random Guess, both models seem to be better, as they have a higher 68% CI for F_β scores, meaning that it is probable for any new dataset to be predicted better than with a Random Guess model.

For the Weighted Guess, both models have a higher 95% CI for F_β scores, meaning that it is very probable for any dataset to be predicted better than with a Weighted Guess model.

Table 6.5 shows the hyperparameters of the best models for each set of features. An interesting thing to observe here is that when the value of C is equal to 10^{-3}, the models have a specificity of 0. Figure 6.1 shows the roc curves of both SVM models.



(a) 5 poles model **(b)** 13 poles model

Figure 6.1: SVM ROC curves

	68% CI	95% CI	Confusion Matrix	F_β Score	Recall	Precision	Specificity	AUC Score
3 poles	[0.45, 0.64]	[0.36, 0.74]	0 31 0 13	0.58	**1.0**	0.29	0.0	0.33
4 poles	[0.45, 0.65]	[0.35, 0.75]	0 31 0 13	0.58	**1.0**	0.29	0.0	0.32
5 poles	[0.37, 0.64]	[0.23, 0.78]	14 17 1 12	0.67	0.92	0.41	0.45	**0.77**
9 poles	[0.40, 0.60]	[0.30, 0.69]	10 21 0 13	0.67	**1.0**	0.38	0.32	0.73
13 poles	[0.49, 0.72]	[0.38, 0.83]	20 11 2 11	**0.70**	0.85	**0.50**	**0.64**	0.75
32 poles	[0.44, 0.74]	[0.29, 0.88]	13 18 1 12	0.66	0.92	0.40	0.42	0.76
32 poles filter	[0.47, 0.65]	[0.38, 0.74]	0 31 0 13	0.58	**1.0**	0.29	0.0	0.29
32 poles wrapper	**[0.53, 0.69]**	**[0.45, 0.77]**	14 17 2 11	0.62	0.85	0.39	0.45	0.69
32 poles by hand	[0.50, 0.69]	[0.41, 0.78]	18 13 4 9	0.57	0.69	0.40	0.58	0.72

Table 6.4: CIs and metric values for SVM. Bold values are the best values for each metric. Green lines are the best models.

	Gamma	C
3 poles	10^{-6}	10^{-3}
4 poles	10^{-6}	10^{-3}
5 poles	10^{-6}	100
9 poles	10^{-7}	100
13 poles	10^{-6}	1000
32 poles	10^{-7}	1000
32 poles filter	10^{-7}	10^{-3}
32 poles wrapper	10^{-6}	100
32 poles by hand	10^{-5}	10

Table 6.5: Hyperparameters of the SVM models. Green lines are the best models.

6.2.2 KNN

Table 6.6 shows the results for the KNN models. The best model here is the 32 poles by hand model. Indeed, the F_β and the recall are the second best values of the table, and all other metric values of this model are the best in their columns.

In addition to those good values on the test set, the CI for cross-validation scores is narrow and the lower bound of the 95% CI is higher than both Random and Weighted Guess. This gives a reasonable confidence in the reachable performances of this model on new datasets.

	68% CI	95% CI	Confusion Matrix	F_β Score	Recall	Precision	Specificity	AUC Score
3 poles	[0.46, 0.64]	[0.37, 0.73]	15 16 / 4 9	0.54	0.69	0.36	0.48	0.67
4 poles	[0.42, 0.66]	[0.31, 0.77]	12 19 / 5 8	0.46	0.61	0.30	0.39	0.65
5 poles	[0.39, 0.62]	[0.28, 0.73]	9 22 / 4 9	0.48	0.69	0.29	0.29	0.59
9 poles	[0.36, 0.72]	[0.18, 0.90]	14 17 / 5 8	0.48	0.61	0.32	0.45	0.58
13 poles	[0.45, 0.61]	[0.37, 0.69]	17 14 / 3 10	0.61	0.77	0.42	0.55	0.71
32 poles	[0.39, 0.66]	[0.25, 0.79]	19 12 / 2 11	**0.68**	**0.85**	0.48	0.61	0.72
32 poles filter	[0.46, 0.68]	[0.36, 0.78]	19 12 / 4 9	0.58	0.69	0.43	0.61	0.74
32 poles wrapper	[0.49, 0.69]	[0.39, 0.79]	15 16 / 4 9	0.54	0.69	0.36	0.48	0.71
32 poles by hand	**[0.51, 0.69]**	**[0.42, 0.77]**	21 10 / 3 10	0.66	0.77	**0.5**	**0.68**	**0.75**

Table 6.6: CIs and metric values for KNN. Bold values are the best values for each metric. Green lines are the best models.

Table 6.7 shows the hyperparameters of the best models for each set of features. Figure 6.2 shows the ROC curve of best KNN model.

	n_neighbors	weights	p
3 poles	50	*uniform*	1
4 poles	40	*distance*	1
5 poles	50	*distance*	1
9 poles	20	*uniform*	2
13 poles	20	*uniform*	4
32 poles	40	*uniform*	6
32 poles filter	40	*uniform*	2
32 poles wrapper	30	*uniform*	1
32 poles by hand	20	*distance*	3

Table 6.7: Hyperparameters for the KNN models. Green lines are the best models.

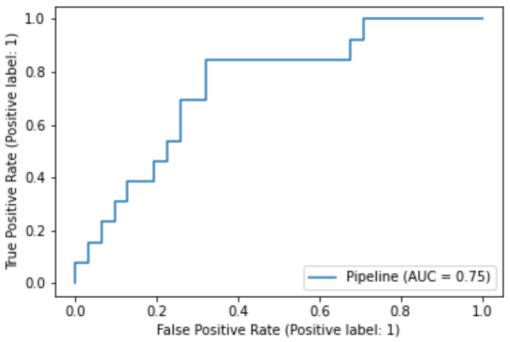


Figure 6.2: KNN ROC curve

6.2.3 RF

Table 6.8 shows the results for the RF models. The best model here is the 13 poles. Even if it is sub-optimal in terms of metrics, the model has the second best AUC score, F_β score, specificity, and the best CI, which is important as the other models performed poorly in terms of CI.

The 95% CI lower bound value is higher than the F_β score of the Weighted Guess and the 68% is higher than the F_β score of the Random Guess. This model should therefore perform well on new datasets.

	68% CI	95% CI	Confusion Matrix	F_β Score	Recall	Precision	Specificity	AUC Score
3 poles	[0.37, 0.63]	[0.24, 0.76]	20 11 / 5 8	0.54	0.61	0.42	0.64	0.73
4 poles	**[0.42, 0.64]**	[0.31, 0.75]	19 12 / 5 8	0.53	0.61	0.4	0.61	0.72
5 poles	[0.42, 0.63]	[0.32, 0.73]	17 14 / 5 8	0.51	0.61	0.36	0.55	0.67
9 poles	[0.28, 0.70]	[0.08, 0.91]	22 9 / 5 8	0.56	0.61	**0.47**	**0.71**	**0.79**
13 poles	**[0.42, 0.64]**	**[0.32, 0.75]**	20 11 / 5 8	0.54	0.61	0.42	0.64	0.75
32 poles	[0.36, 0.64]	[0.23, 0.77]	19 12 / 5 8	0.53	0.61	0.4	0.61	0.73
32 poles filter	[0.33, 0.63]	[0.19, 0.77]	19 12 / 5 8	0.53	0.61	0.4	0.61	0.67
32 poles wrapper	[0.39, 0.66]	[0.26, 0.79]	19 12 / 4 9	0.58	**0.69**	0.43	0.61	0.72
32 poles by hand	[0.37, 0.63]	[0.24, 0.76]	20 11 / 4 9	**0.59**	**0.69**	0.45	0.64	0.70

Table 6.8: CIs and metric values for RF. Bold values are the best values for each metric. Green lines are the best models.

Table 6.9 shows the hyperparameters of the best models for each set of features. Figure 6.3 shows the ROC curve of best RF model.

	max_depth	max_features	min_samples _in_leaf	min_samples _in_split	n_estimators
3 poles	2	3	3	8	100
4 poles	2	*sqrt*	3	8	50
5 poles	2	8	3	8	400
9 poles	2	3	3	8	100
13 poles	2	3	3	12	100
32 poles	2	3	3	8	200
32 poles filter	2	8	3	8	50
32 poles wrapper	2	3	3	8	100
32 poles by hand	2	3	3	8	100

Table 6.9: Hyperparameters for the RF models.
Green lines are the best models.

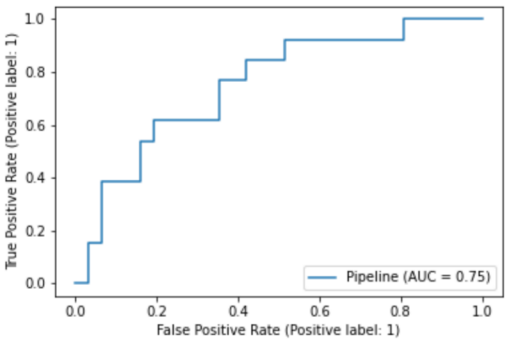


Figure 6.3: RF ROC curve

6.2.4 LDA

Table 6.10 shows the results for the LDA models. Here, the 32 poles wrapper is a particular model as it is trained on a dataset wrapped by a LDA model. Therefore, even if the CIs are good, compared to all other models, the metrics on the test set are quite bad, which is due to a clear overfitting, as the model is an LDA model that got optimal features from another LDA model.

The best model is the 5 poles, as it has good values for all metrics, and a really good recall value. The lower bound of the 68% CI of this model is better than the Weighted Guess, but not better than the Random Guess, which makes it a correct baseline model.

Table 6.11 shows the hyperparameters of the best models for each set of features. Figure 6.4 shows the ROC curve of best LDA model.

6.2.5 XGB

Table 6.12 shows the results for the XGB models. Here, almost all the CIs are pretty wide, meaning that from one model to another, the F_β scores vary a lot. The models with those

	68% CI	95% CI	Confusion Matrix	F_β Score	Recall	Precision	Specificity	AUC Score
3 poles	[0.41, 0.65]	[0.30, 0.76]	13 18 / 2 11	**0.61**	**0.85**	0.38	0.42	0.61
4 poles	[0.37, 0.70]	[0.21, 0.86]	10 21 / 3 10	0.54	0.77	0.32	0.32	0.66
5 poles	[0.36, 0.64]	[0.23, 0.77]	13 18 / 2 11	**0.61**	**0.85**	0.38	0.42	0.70
9 poles	[0.34, 0.61]	[0.20, 0.75]	15 16 / 3 10	0.59	0.77	0.38	0.48	**0.74**
13 poles	[0.36, 0.72]	[0.19, 0.89]	13 18 / 4 9	0.52	0.69	0.33	0.42	0.66
32 poles	[0.32, 0.77]	[0.10, 0.99]	16 15 / 3 10	0.60	0.77	**0.4**	0.52	0.69
32 poles filter	[0.35, 0.66]	[0.20, 0.81]	14 17 / 3 10	0.58	0.77	0.37	0.45	0.72
32 poles wrapper	**[0.53, 0.67]**	**[0.46, 0.74]**	19 12 / 7 6	0.41	0.46	0.33	**0.61**	0.52
32 poles by hand	[0.44, 0.69]	[0.32, 0.81]	14 17 / 4 9	0.53	0.69	0.35	0.45	0.56

Table 6.10: CIs and metric values for LDA. Bold values are the best values for each metric. Green lines are the best models.

	Solver	Tol
3 poles	*SVD*	0.1
4 poles	*SVD*	0.5
5 poles	*SVD*	0.1
9 poles	*SVD*	0.5
13 poles	*SVD*	0.5
32 poles	*SVD*	0.5
32 poles filter	*SVD*	0.5
32 poles wrapper	*SVD*	0.0001
32 poles by hand	*SVD*	0.0001

Table 6.11: Hyperparameters for the LDA models. Green lines are the best models.

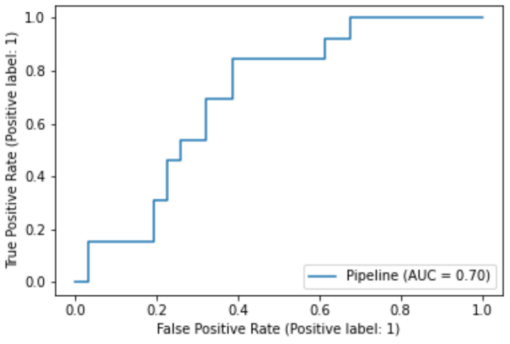


Figure 6.4: LDA ROC curve

large CIs are frequently not better than Random and Weighted Guess, which implies that the models may perform badly on new datasets. The only model with decent CI is the 13 poles one, which outperforms the Weighted Guess in the 68% CI and almost in the 95% CI.

	68% CI	95% CI	Confusion Matrix	F_β Score	Recall	Precision	Specificity	AUC Score
3 poles	[0.31, 0.67]	[0.13, 0.84]	21 10 / 4 9	0.61	0.69	0.47	0.68	0.69
4 poles	[0.30, 0.66]	[0.12, 0.83]	19 12 / 4 9	0.58	0.69	0.43	0.61	0.63
5 poles	[0.38, 0.63]	[0.26, 0.76]	21 10 / 8 5	0.37	0.38	0.33	0.68	0.61
9 poles	[0.28, 0.60]	[0.12, 0.77]	23 8 / 4 9	0.63	0.69	0.53	**0.74**	0.69
13 poles	**[0.40, 0.68]**	**[0.27, 0.81]**	21 10 / 5 8	0.55	0.61	0.44	0.68	0.69
32 poles	[0.32, 0.65]	[0.16, 0.81]	19 12 / 4 9	0.58	0.69	0.43	0.61	**0.70**
32 poles filter	[0.25, 0.69]	[0.04, 0.90]	23 8 / 5 8	0.57	0.61	0.5	0.74	0.69
32 poles wrapper	[0.27, 0.59]	[0.12, 0.74]	19 12 / 6 7	0.47	0.54	0.37	0.61	**0.70**
32 poles by hand	[0.25, 0.71]	[0.03, 0.93]	21 10 / 3 10	**0.66**	**0.77**	**0.5**	0.68	0.65

Table 6.12: CIs and metric values for XGB. Bold values are the best values for each metric. Green lines are the best models.

Table 6.13 shows the hyperparameters of the best models for each set of features. Figure 6.5 shows the ROC curve of best XGB model.

	Loss function	Learning rate	n_estimators	max_depth	min_child_weight	reg_alpha
3 poles	*logloss*	0.01	300	9	6	4
4 poles	*logloss*	0.01	300	5	6	8
5 poles	*logloss*	0.01	150	2	2	12
9 poles	*logloss*	0.01	150	5	6	12
13 poles	*logloss*	0.1	100	5	6	8
32 poles	*logloss*	0.01	150	2	4	12
32 poles filter	*logloss*	0.01	100	5	2	4
32 poles wrapper	*logloss*	0.1	100	2	2	4
32 poles by hand	*logloss*	0.01	300	9	4	8

Table 6.13: Hyperparameters for XGB models. Green lines are the best models.

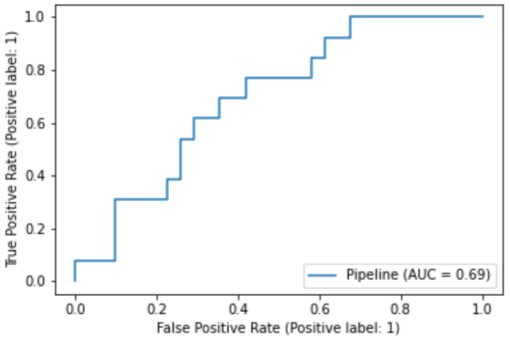


Figure 6.5: XGB ROC curve

6.3 SMOTE

In this section, a small analysis is performed on the best model of each type to justify the use of the SMOTE method. The metrics used for this comparison are the recall, precision and sensitivity as the motivation to use SMOTE was to obtain high recall and precision with correct sensitivity instead of very high sensitivity with a rather low recall, which is easy when working on a very imbalanced dataset.

- **RF :**
  We can see in table 6.14 that precision values of both models are similar, but the recall of the model without SMOTE is bad. The specificity of the model is better without SMOTE, but correct overall. As the objective is to maximize the recall with a good precision and a correct sensitivity, the RF model is better here.

	Recall	Precision	Sensitivity
SMOTE	0.61	0.42	0.64
No SMOTE	0.38	0.55	0.87

Table 6.14: Performances with and without SMOTE (RF model)

- **SVM - 5 poles :**
  This model performs better with SMOTE given the objectives of this master thesis. Indeed, the recall and precision of this model are clearly better with SMOTE, and the sensitivity is lower for SMOTE, but is correct overall (see table 6.15).

- **SVM - 13 poles :**
  For this model, there is no doubt that it performs better with SMOTE. Indeed, the scores of recall, precision, and sensitivity are better for SMOTE (see table 6.16).

- **XGB :**
  In a similar fashion as for the RF model presented before, we can see in table 6.17

	Recall	Precision	Sensitivity
SMOTE	0.92	0.41	0.45
No SMOTE	0.46	0.33	0.61

Table 6.15: Performances with and without SMOTE (SVM - 5 poles model)

	Recall	Precision	Sensitivity
SMOTE	0.85	0.5	0.65
No SMOTE	0.46	0.33	0.61

Table 6.16: Performances with and without SMOTE (SVM - 13 poles model)

that precision values of both models are similar, but the recall of the model without SMOTE is really bad. Its score is under the score of both Random and Weighted Guess. The specificity of the model is better without SMOTE, but correct overall. Here, as the objective is to maximize the recall with a good precision and a correct sensitivity, it is clear that the XGB model is better here.

	Recall	Precision	Sensitivity
SMOTE	0.61	0.44	0.68
No SMOTE	0.23	0.50	0.90

Table 6.17: Performances with and without SMOTE (RF model)

- **LDA :**
  As for the SVM - 5 poles model, this model performs better with SMOTE given the objectives of this master thesis. Indeed, the recall and precision of this model are clearly better with SMOTE, and the sensitivity is largely lower for SMOTE, but is correct overall (see table 6.18).

- **KNN :**
  Like all other models, KNN performs beter with SMOTE. Indeed, the recall is clearly better for SMOTE with an equivalent precision for both. As the objective is to maximize the recall with a good precision and a correct sensitivity, the KNN model is better here with SMOTE (see table 6.19).

6.4 Lateralization

A hypothesis for this problem is that there should be no detectable influence from the lateralization of the rhythms in the prediction as the cognitive functions affected by the POD. Indeed, as described in details in the paper from Palanca et Al.[36], the large majority of the known biomarkers to identify POD appearance are bilaterals. The only

	Recall	Precision	Sensitivity
SMOTE	0.85	0.38	0.42
No SMOTE	0.08	0.25	0.90

Table 6.18: Performances with and without SMOTE (LDA model)

	Recall	Precision	Sensitivity
SMOTE	0.77	0.5	0.68
No SMOTE	0.46	0.6	0.87

Table 6.19: Performances with and without SMOTE (KNN model)

biomarker described in the study that could sometimes be lateralized is not detectable by using absolute powers as features.

To verify this assumption in our dataset, the 5 poles set of features has been separated in two different ways. The first set takes into account four areas, excluding the left area, and the second excludes the right area (see figure 4.4). The metrics used to determine if there is a big difference in prediction capability between both sets are the recall, precision and sensitivity scores, as those metrics should not change if the sets give the same amount of information.

Looking at all values from table 6.20, the majority of the models seems to perform equivalently for both areas. But one model seems to be more sensitive to this lateralization. It is the SVM, which seems to take more information from the right area than from the left area.

	Left area			**Right area**		
	Recall	**Precision**	**Sensitivity**	**Recall**	**Precision**	**Sensitivity**
RF	0.69	0.35	0.45	0.69	0.36	0.48
SVM	0.85	0.34	0.32	0.92	0.41	0.45
XGB	0.54	0.37	0.61	0.61	0.4	0.61
LDA	0.85	0.38	0.42	0.85	0.39	0.45
KNN	0.61	0.30	0.39	0.61	0.29	0.35

Table 6.20: Performances for left and right areas

6.5 Analysis for sub-bands

For the sub-bands analysis, we will look only at the AUC score, which measures the global accuracy of the model, in order to have an idea of the amount of information located in each sub-bands. To determine this amount of information, the AUC score is computed for

a set containing the 4 sub-bands described in section 4.4, and then for all combinations of 3 sub-bands, to look at the loss (or gain, if some sub-bands contain redundant information) of information by not using one of the sub-bands. The choice of analyzing only two bands of frequencies -Alpha and Beta- has been done for reasons described in the sub-bands sections. The electrode feature set used is the 13 poles as it keeps only the frontal values, which are performing well for the majority of the models.

6.5.1 Sub-bands Alpha

The main reasons for analyzing the sub-bands of Alpha are the following:

- **Statistics**
  Looking at the t-test of section 5.2 and at the filter selection of features using ANOVA, it seems that Alpha is related to the POD state in terms of variation of mean and variance.

- **Physiology**
  As it is well described in the literature (see section 1.4), the Alpha band, if monitored during the operation, is believed to be a good predictor of the POD state. Therefore, it could be interesting to see in what sub-bands of this Alpha band the information is located.

We exclude here the SVM values from our conclusions as they strongly overfit (recall: 1.0 and sensitivity: +/- 0.0) the data.

Here, observing table 6.21, the third Alpha sub-band lowers the global scores for each model. In general, it also seems that, except for the KNN model, the models gain information when removing sub-bands, meaning that there should probably be redundant information contained in those sub-bands.

	Alpha	No sub-band 1	No sub-band 2	No sub-band 3	No sub-band 4
RF	0.62	0.70	0.65	0.70	0.63
SVM	0.35	0.26	0.50	0.29	0.27
XGB	0.60	0.70	0.71	0.73	0.72
LDA	0.53	0.69	0.70	0.60	0.56
KNN	0.77	0.76	0.71	0.79	0.64

Table 6.21: AUC scores for Sub-bands of Alpha band

6.5.2 Sub-bands Beta

The reason for analyzing the sub-bands of Beta is the following:

- **Statistics**
  Looking at the t-test of section 5.2 and at the filter selection of features using

ANOVA, it seems that Beta is closely related to the POD state in terms of variation of mean and variance.

Similarly, as for the Alpha sub-bands models, the Beta sub-bands models seem in general, to gain or at least not lose information when removing sub-bands, meaning that there should also probably be redundant information contained in those sub-bands. In particular, SVM, LDA and KNN models seem to work much better when removing one.

	Beta	No sub-band 1	No sub-band 2	No sub-band 3	No sub-band 4
RF	0.69	0.65	0.68	0.64	0.68
SVM	0.67	0.78	0.76	0.76	0.76
XGB	0.68	0.71	0.75	0.66	0.66
LDA	0.68	0.74	0.73	0.70	0.72
KNN	0.58	0.68	0.66	0.65	0.64

Table 6.22: AUC scores for Sub-bands of Beta band

6.6 Convergence curves

In this section, the convergence curves are shown for the F_β scores and commented for the best model of each type identified in previous sections (all curves are in Appendix C).

In those graphs, the blue curves are represented with their 68% distribution estimation. The blue bar is located at a position corresponding to the proportions of training and testing sets. This is done to estimate a measure of the overfitting. Indeed, a hypothesis that we made here is that the distance between the the training and validation score distributions reflect the probable tendency of the model to overfit for the same proportions of training and testing sets in the generalization of the model.

- **SVM - 13 poles :**
  In figure 6.6, we can see that when reaching the blue bar, the distance between training score distribution and testing distribution is relatively large (+/- 0.20) with the training above testing. This means that the SVM model here will have a clear trend to overfit.

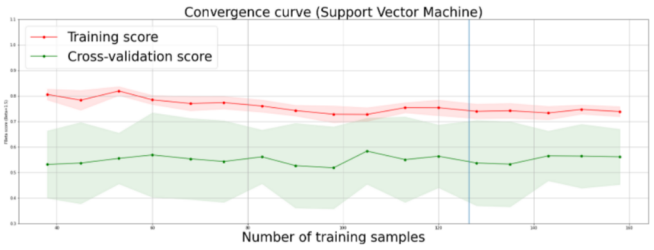


Figure 6.6: Convergence curve of SVM model for 13 poles set

- **SVM - 5 poles :** In figure 6.6, we can see that when reaching the blue bar, the distance between training score distribution and testing distribution is relatively short (+/- 0.10) even if training is above testing. This means that the SVM model will probably have a small trend to overfit.

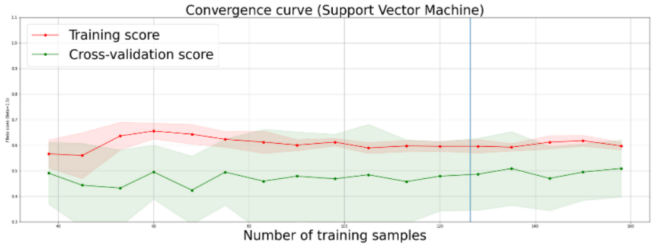


Figure 6.7: Convergence curve of SVM model for 5 poles set

- **KNN :**
  In figure 6.8, we can see that when reaching the blue bar, the distance between training score distribution and testing distribution is almost null, implying that the model will probably not overfit.

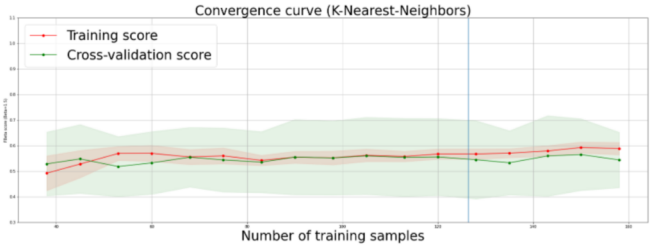


Figure 6.8: Convergence curve of KNN model for 32 poles set

- **RF :**
  In figure 6.9, we can see that when reaching the blue bar, the distance between training score distribution and testing distribution is relatively short (+/- 0.10) even if training is above testing. This means that the RF model here will probably have a small trend to overfit.

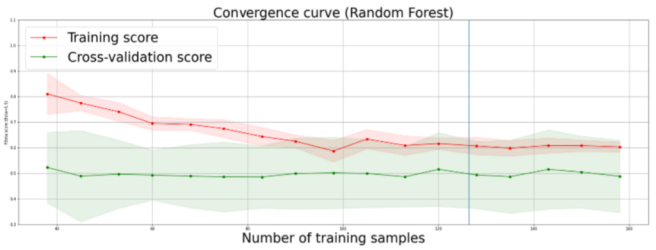


Figure 6.9: Convergence curve of RF model for 13 poles set

- **XGB :**
  In figure 6.10, we can see that when reaching the blue bar, the distance between training score distribution and testing distribution is relatively big (+/- 0.20) with the training above testing. This means that the XGB model here will have a clear trend to overfit.

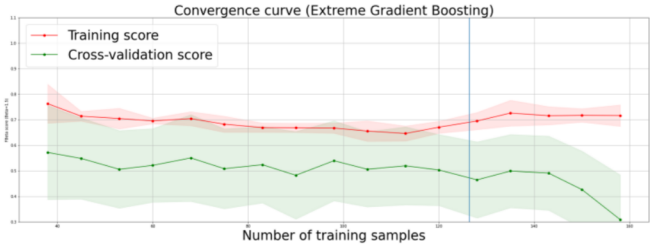


Figure 6.10: Convergence curve of XGB model for 13 poles set

6.7 Stability to hyperparameters

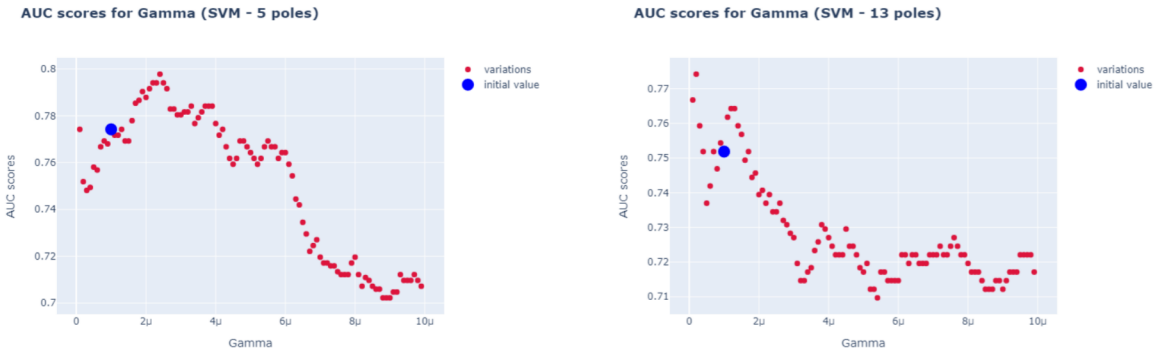
In this section, the objective is to look at the models' variations in terms of scoring when introducing little variations for the different continuous hyperparameters. This analysis is performed on the best model for each type of model. Some of the discrete and all the non-numerical parameters aren't discussed here as their influence on the results should not be continuous in principle, so introducing variations for them could change the scores a lot without telling us anything about the prediction stability of the model.

For continuous hyperparameters and some of the discrete hyperparameters, if the scores vary a lot, then it is quite likely for the model to be unstable. Indeed, the model has therefore learnt much from those precise values of hyperparameters on the dataset and will probably perform poorly on new datasets that will likely be a little bit different. This is mainly because the model could therefore overfit the testing set, with a set of hyperparameters particularly adapted to the testing set but not that much on new datasets.

6.7.1 SVM

For the SVM, both **gamma** and **C** are chosen as those hyperparameters together can lead to large overfitting, as observed in the results of section 6.2.1. In the same section, it is possible to see that two models have been selected as the best models, the model from the 5 poles features set and the one from the 13 poles.

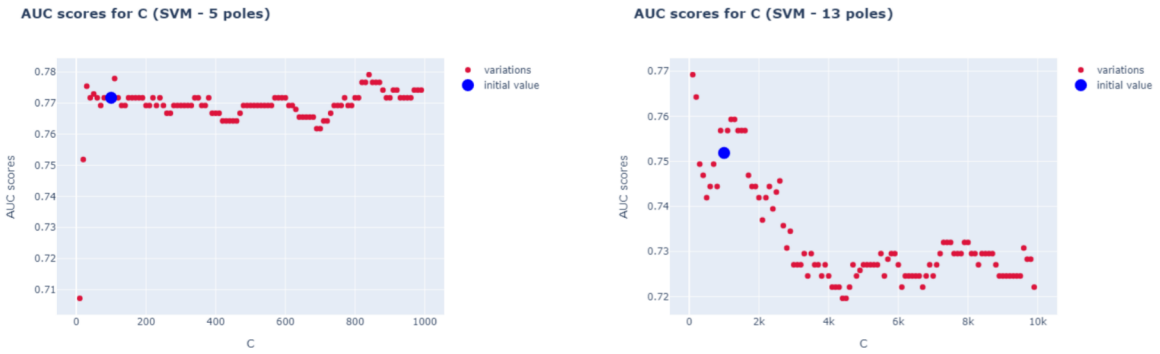
In figure 6.11, we can observe that the variation of the AUC scores seems to be a continuous plot for both models, indicating that the SVM model probably doesn't overfit the testing set in terms of **gamma** hyperparameter.



(a) 5 poles model **(b)** 13 poles model

Figure 6.11: Gamma variations

In figure 6.12, the variation of the AUC scores also seems to be continuous for both models. This indicates that the SVM model also probably doesn't overfit the testing set in terms of **C** hyperparameter.



(a) 5 poles model **(b)** 13 poles model

Figure 6.12: C variations

6.7.2 KNN

For the KNN model, the **number of neighbors** is analyzed as it should not change the results too much if there is no overfitting. As described in section 6.2.2, the features set here is the 32 poles filtered one.

In the figure 6.13, we can observe that the variation of the AUC scores seems to be a continuous plot, indicating that the KNN model probably doesn't overfit the testing set in terms of **number of neighbors** hyperparameter.

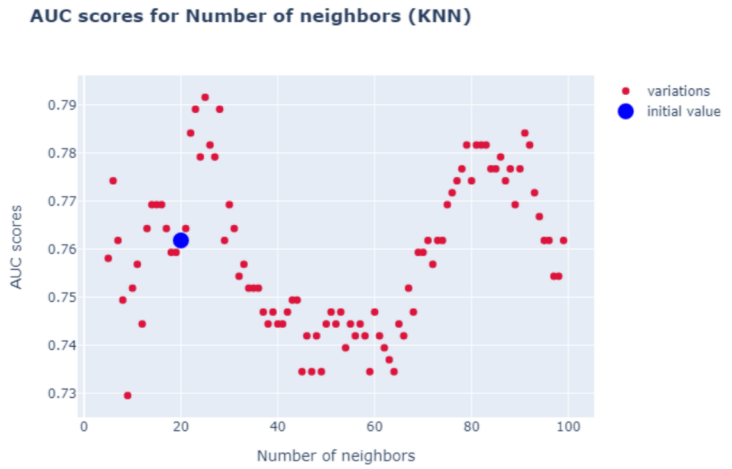


Figure 6.13: Number of neighbors variations

6.7.3 RF

For the RF model, the **number of estimators** is analyzed as it is the only one used that should not highly influence the data. As described in section 6.2.3, the features set here is the 13 poles one.

In the figure 6.14, we can observe that the variation of the AUC scores seems to be a continuous plot looking at the scale of the scores variation, indicating that the RF model probably doesn't overfit the testing set in terms of **number of estimators** hyperparameter.

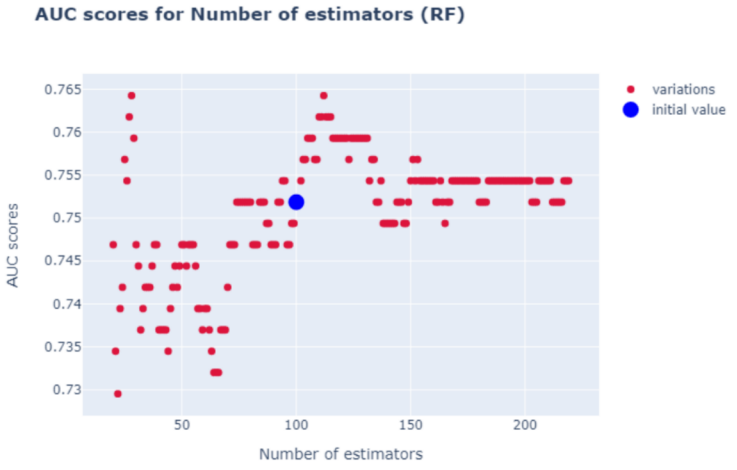


Figure 6.14: Number of estimators variations

6.7.4 LDA

In the LDA models, the only hyperparameters that should not influence the results too much if there is no overfitting is the **tolerance**. But it should influence more than all the other parameters described previously as this is a strict threshold, which implies that there should just be different levels of AUC scores instead of a continuous increase. As described in section 6.2.4, the best features set is the 5 poles one.

The intuition of getting levels instead of continuous plots of the AUC scores is confirmed in figure 6.15. As the AUC scores variations are relatively different from one tolerance to another, we can conclude that the model is highly dependant on the tolerance value.

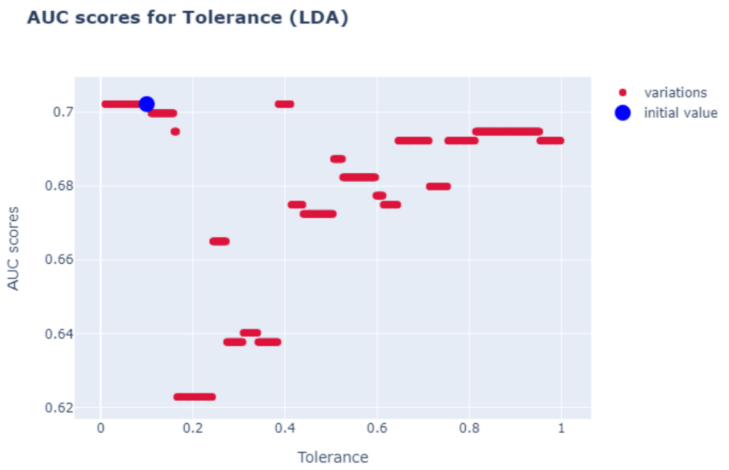


Figure 6.15: Tolerance variations

6.7.5 XGB

For the XGB model, the **learning rate** and the **number of estimators** are analyzed. The **alpha regularization** could have been chosen too, but was not because this parameter is made to prevent overfitting. Therefore, while it can lower the AUC score, it is not possible for it to increase the overfitting of the model. As described in section 6.2.5, the feature set here is the 13 poles one.

The learning rate doesn't seem to imply overfitting from the XGB model, looking at the continuity of the AUC scores in figure 6.16.
In the figure 6.17, we can also observe that the variation of the AUC scores seems to be a continuous plot looking at the scale of the scores variation, indicating that the XGB model probably doesn't overfit the testing set in terms of **number of estimators** hyperparameter.

Figure 6.16: Learning rate variations

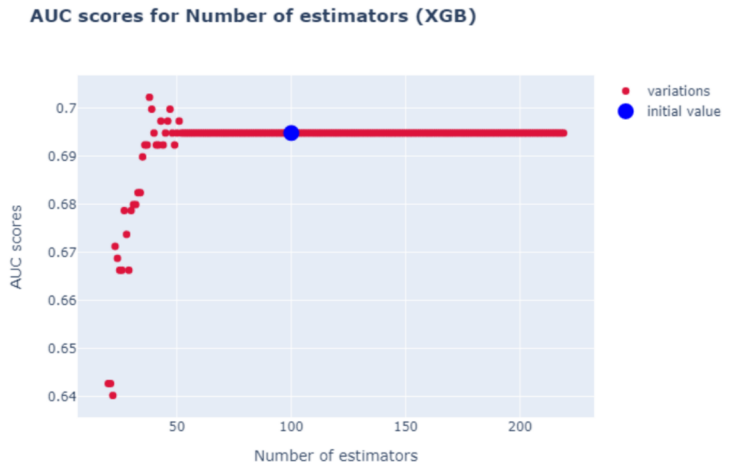


Figure 6.17: Number of estimators variations

Chapter 7

Discussion

7.1 Feature choice

In this master thesis, we chose to take as feature the absolute average power of bands of frequencies based mainly on a paper from Koch et Al.[35]. But this choice could be sub-optimal and other choices might lead to better performances. In light of the Palanca et al. paper[36], a number of different features could potentially be considered. Among others, it seems that the relative average power of frequency bands could be an interesting substitute to the absolute power.

7.2 Feature selection

Some type of feature selection have not been explored. Indeed, the selector using the principle of embedded selection was not used. This choice has been made because of the low performances for almost all models on the datasets with feature selection. However, it is important to mention the selection performed manually on the features. Indeed the KNN model seems to work particularly well on the selected features.

The feature selection was not performed on each feature set. The justification of this choice was the fact that the majority of the feature sets had a sufficiently small number of feature in comparison with the number of patients. The methodology could possibly be improved by trying it for each feature sets to check if some models could perform better on the smaller sets with selected features.

7.3 Stability of the models

In the chapter Results, different indicators were discussed that could influenced the stability of the model when generalizing it. In general, the values of the indicators were decent for the best model of each type. This could mean that their stability is good if the indicators are well-chosen indicators. In this section, the reasons for the choice of those indicators are discussed.

7.3.1 CI for validation folds

The first indicator is the CI of the validation folds scores. As described in section 5.7.2, this confidence interval can give an idea of the scores that the model could reach on new data, even if it will probably be too optimistic.

Normality assumption

To compute the CI of the validation scores, we assume that the scores follow a normal distribution. The prerequisite to make this assumption is met as described in the section 5.7.2. But it is nearly not met as the number of samples is very small (only 10 folds). Therefore in addition to the optimistic assessment inherent in the fact that the calculation of CI is carried out on the validation scores, the lack of data could imply a bad computation of the CI. This could lead to an even more optimistic assessment.

7.3.2 Convergence curves

In the chapter Results, the convergence curves are used to perform an estimation of the potential overfitting when generalizing. But this estimation should be taken with a grain of salt for two reasons.

Firstly, we evaluated a given model to potentially overfit based on the deviation between both training and validation curves and on the deviation between their distributions. But as seen in the previous section, the distribution of both curves could be wrongly calculated, leading to a bad evaluation of the overfitting.

Secondly, the evaluation of this overfitting was performed where the ratio of validation size/training size is the same as the ratio of testing size/training size. This point has been chosen because we assumed that the model will perform equally when both ratios are equal. This hypothesis could be wrong and the evaluation of overfitting could therefore be biased.

7.3.3 Variation of hyperparameters

The variation of the hyperparameters is supposed to give an information about a potential overfitting of the model on the testing set. Indeed, as explained in section 5.7.3 we made the assumption that a model could overfit the data from the testing set if the set of hyperparameters works incredibly well on it, and not that well for little variations of the set. But this hypothesis could be wrong, and our conclusions could therefore be incorrect.

7.4 Analysis of sub-bands

While looking at the results obtained for the sub-bands analysis, an interesting property can be observed. It seems that a majority of the models performed better on the feature sets that contained 3 sub-bands instead of 4, either for Alpha or Beta.

This property could originate from the fact that a part of the information loss comes from redundancies at different frequencies. This is really interesting as at least for this task it could open perspectives for new analyses. Indeed, it could be possible to work on a way to reduce this redundancy on bands of frequency to potentially improve the quality of models.

About the results in general for sub-bands, it is important to mention that certain models seem to get more information for their predictions from Beta sub-bands like LDA and SVM, while the KNN model get more information from the Alpha sub-bands. RF and XGB works similarly on both Alpha and Beta. Therefore it seems that both bands are very useful for the prediction of POD since each model can reach similar performances as those of the best models of each type. This is interesting since this could mean that Beta frequencies in general could be considered as a new biomarker for the prediction of POD. An example of further work could be to determine which part of the Beta band of frequencies is closely related to POD.

7.5 Overall areas for improvement

Overall, there are some weaknesses and areas for scores improvement in this work:

- **Area 1: Lack of data**
  Like a large majority of the prediction models in Biomedical Machine Learning, the number of subjects is limited. We proved the feasibility of predictions by comparing the scores obtained with Random and Weighted Guess. It implies that the performances could be improved by adding subjects as prediction is possible.

- **Area 2: Redundancy**
  As described in the previous section, there could be redundancy inside our bands of frequency that lowers the reachable range of scores. Therefore, it could be feasible to improve our models by trying to reduce this redundancy.

- **Area 3: Other features**
  In general, we saw in section 1.3 that several biomarkers that are considered as good predictor of POD exist in the literature. Having more of those features in addition to the EEG signals could help to improve the models used in this work.

- **Weakness 1: Lack of objective metrics**
  To assess the performances of our models, we used objective metrics, like Recall, Sensitivity, F_β-score, Precision, AUC score and convergence curve. But to assess the quality of a model, we need to interpret those metrics, and this interpretation can be quite subjective. Taking the convergence curves as example, we interpreted the overfitting based on the deviation between both curves, but what deviation begin to be significant? For the general assessment, what is the importance that we should give to each metrics ? In what proportion is the recall more important than the Precision ? Is the $\beta = 1.5$ a good value for the F_β-score ? In general, the evaluation of the models is not based on precise and objective criteria.

- **Weakness 2: SMOTE**
  Even if in general, SMOTE is useful as it allows to work with imbalanced dataset, it implies a weakness in our methodology. SMOTE has a major shortcoming when oversampling. It is called **overgeneralization**: the method generates the data of the minority class without having taking the majority class into account. When the distributions of the class are highly skewed, the minority class is very sparse with respect to the majority class, resulting in a greater chance of class mixture. Therefore, the artificial data generated by SMOTE could possibly decrease the performance in comparison to a downsampling, which was not tried here.

Conclusion

In this master's thesis, we sought to confirm that predicting the occurrence of POD after cardiac surgery is feasible by using Machine Learning. To do so, we tried to create the best models by varying features in order to test the predictive capabilities of the dataset. Another objective we had was to determine in what extent did the compression of the information reduced the overall prediction capabilities by creating several dataset with less poles of information than the original 32 electrodes. We also sought to confirm the effectiveness of an EEG biomarker described in the literature (see section 1.3): *the potential decrease in alpha rhythm anteriorization during anesthesia.* In addition, we attempted to find new biomarkers in EEG signals that could effectively predict the occurrence of POD. Finally, we wanted to determine the location of the information in the observed frequency bands by looking at sub-bands.

To fulfill those objectives, we chose different models (LDA, KNN, SVM, XGB, RF) based on specific criteria. Then, we decided to create 5 datasets based on the initial one by grouping electrodes also based on specific criteria for each. In addition to those 5 datasets, the initial one was also kept and 3 types of feature selection were performed on it to generate 3 additional sets. All the models were trained on those 9 datasets.

The dataset that led to the best results for a majority of the best models was composed of 13 independent electrodes located in the anterior part of the cap. This confirms a significant influence of the anteriorization on the results. Another information we can retrieve is that it is possible to reduce a lot the amount of electrodes used without losing information. This could potentially be useful in the future to allow to use more sources of additional information without increasing the number of features too much.

The best model was the SVM on the dataset presented above as it achieves the best Fbeta score ($\beta = 1.5$): **0.70**. This model had a pretty good 95% CI: [**0.49, 0.72**], a recall of **0.85** and a precision of **0.50**. The AUC score was **0.75** and even the specificity is correct: **0.64**.

For the sub-band analysis, we calculated the scores reachable by each model on the previously described dataset for 4 equally sized Alpha sub-bands and 4 equally sized Beta sub-bands. The decision of keeping only Alpha and Beta sub-bands was led by the literature for Alpha and by the p-values obtained for each Beta electrode when performing the feature selection. This allows to determine two things:
Firstly, as expected when selecting both bands and analyzing the reachable results with

the sub-bands for each of those, the scores obtained were close the the general best scores, probably meaning that a majority of the information comes from both bands.
Secondly, we realized that a possible improvement could be to remove some of the sub-bands from a band of frequency, as for both Alpha and Beta, removing one sub-band led to even better scores. This could be the subject of future works.

In general, the models performed better than guessing randomly, leading to the affirmation that the prediction of POD appearance after cardiac surgery is feasible by using Machine Learning. Knowing this opens up a vast field of research, including for example which features will work best.

Despite the low number of data and the class imbalance, we manage to create decent classifiers. Therefore, the prediction of POD appearance seems to be promising if more data are provided. Other limitations that we identified about our models and the methodology in general are presented in the Discussion chapter.

Furthermore, some possibility remain unexplored and this work raises a number of questions that perhaps future contributions could answer. In addition to those already described in the Discussion, here are some of the questions that could lead to future work.

- As the dataset only concerns patient undergoing cardiac surgery, are the built models applicable to any other type of surgery ?
- The data acquisition was performed using specific devices (such as the acquisition cap), but could the result change based on the devices used ?
- The electrode montage (Biosemi 32) used to record contains 32 electrodes. What would be the influence of a change in the number of electrodes ?
- As described in the section 1.1, there is a large influence from the patient population (several factors, like the age, influences widely the final results). Therefore, is it possible that the population used for training was biased ?
- Features used for this work were only related to physiologic bands of frequency (see section 2.1.1 for details). Could the addition of other features like for instance the age of the patient, or biomarkers identified in the section 1.3 lead to better scores ?

This master thesis lay foundations for future work in Machine Learning applied to POD prediction. Work remains to be done and avenues are still to be explored to allow even more qualitative predictions.

Bibliography

[1] D. H. Robinson and A. H. Toledo, "Historical development of modern anesthesia," *Journal of Investigative Surgery: The Official Journal of the Academy of Surgical Research*, vol. 25, no. 3, pp. 141–149, Jun. 2012.

[2] I. Digital Science & Research Solutions, "Timeline - Overview in Publications - Dimensions." [Online]. Available: https://app.dimensions.ai/analytics/publication/overview/timeline?search_mode=content&year_from=1973&year_to=2021

[3] J. R. Maldonado, "Delirium in the acute care setting: characteristics, diagnosis and treatment," *Critical Care Clinics*, vol. 24, no. 4, pp. 657–722, vii, Oct. 2008.

[4] E. L. Whitlock, A. Vannucci, and M. S. Avidan, "POSTOPERATIVE DELIRIUM," *Minerva anestesiologica*, vol. 77, no. 4, pp. 448–456, Apr. 2011. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3615670/

[5] L. E. Vaurio, L. P. Sands, Y. Wang, E. A. Mullen, and J. M. Leung, "Postoperative delirium: the importance of pain and pain management," *Anesthesia and Analgesia*, vol. 102, no. 4, pp. 1267–1273, Apr. 2006.

[6] J. S. Saczynski, E. R. Marcantonio, L. Quach, T. G. Fong, A. Gross, S. K. Inouye, and R. N. Jones, "Cognitive trajectories after postoperative delirium," *The New England Journal of Medicine*, vol. 367, no. 1, pp. 30–39, Jul. 2012.

[7] M. A. Pisani, S. Y. J. Kong, S. V. Kasl, T. E. Murphy, K. L. B. Araujo, and P. H. Van Ness, "Days of delirium are associated with 1-year mortality in an older intensive care unit population," *American Journal of Respiratory and Critical Care Medicine*, vol. 180, no. 11, pp. 1092–1097, Dec. 2009.

[8] S. R. Markar, I. A. Smith, A. Karthikesalingam, and D. E. Low, "The clinical and economic costs of delirium after surgical resection for esophageal malignancy," *Annals of Surgery*, vol. 258, no. 1, pp. 77–81, Jul. 2013.

[9] E. R. Marcantonio, L. Goldman, C. M. Mangione, L. E. Ludwig, B. Muraca, C. M. Haslauer, M. C. Donaldson, A. D. Whittemore, D. J. Sugarbaker, and R. Poss, "A clinical prediction rule for delirium after elective noncardiac surgery," *JAMA*, vol. 271, no. 2, pp. 134–139, Jan. 1994.

[10] T. N. Robinson, C. D. Raeburn, Z. V. Tran, E. M. Angles, L. A. Brenner, and M. Moss, "Postoperative Delirium in the Elderly: Risk Factors and Outcomes," *Annals of Surgery*, vol. 249, no. 1, pp. 173–178, Jan. 2009.

[11] J. Witlox, L. S. M. Eurelings, J. F. M. de Jonghe, K. J. Kalisvaart, P. Eikelenboom, and W. A. van Gool, "Delirium in elderly patients and the risk of postdischarge mortality, institutionalization, and dementia: a meta-analysis," *JAMA*, vol. 304, no. 4, pp. 443–451, Jul. 2010.

[12] L. J. Krzych, M. T. Wybraniec, I. Krupka-Matuszczyk, M. Skrzypek, A. Bolkowska, M. Wilczyński, and A. A. Bochenek, "Detailed insight into the impact of postoperative neuropsychiatric complications on mortality in a cohort of cardiac surgery subjects: a 23,000-patient-year analysis," *Journal of Cardiothoracic and Vascular Anesthesia*, vol. 28, no. 3, pp. 448–457, Jun. 2014.

[13] G. Bellelli, P. Mazzola, A. Morandi, A. Bruni, L. Carnevali, M. Corsi, G. Zatti, A. Zambon, G. Corrao, B. Olofsson, Y. Gustafson, and G. Annoni, "Duration of postoperative delirium is an independent predictor of 6-month mortality in older adults after hip fracture," *Journal of the American Geriatrics Society*, vol. 62, no. 7, pp. 1335–1340, Jul. 2014.

[14] P. M. Rossini, F. Miraglia, and F. Vecchio, "Early dementia diagnosis, MCI-to-dementia risk prediction, and the role of machine learning methods for feature extraction from integrated biomarkers, in particular for EEG signal analysis," *Alzheimer's & Dementia*, vol. n/a, no. n/a, 2022, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/alz.12645. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/alz.12645

[15] M. I. Vanegas, M. F. Ghilardi, S. P. Kelly, and A. Blangero, "Machine learning for EEG-based biomarkers in Parkinson's disease," in *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, Dec. 2018, pp. 2661–2665.

[16] S.-T. Oh and J. Y. Park, "Postoperative delirium," *Korean Journal of Anesthesiology*, vol. 72, no. 1, pp. 4–12, Feb. 2019. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6369344/

[17] D. Stagno, C. Gibson, and W. Breitbart, "The delirium subtypes: a review of prevalence, phenomenology, pathophysiology, and treatment response," *Palliative & Supportive Care*, vol. 2, no. 2, pp. 171–179, Jun. 2004.

[18] S. Boettger and W. Breitbart, "Phenomenology of the subtypes of delirium: phenomenological differences between hyperactive and hypoactive delirium," *Palliative & Supportive Care*, vol. 9, no. 2, pp. 129–135, Jun. 2011.

[19] M. Berger, N. Terrando, S. K. Smith, J. N. Browndyke, M. F. Newman, and J. P. Mathew, "Neurocognitive Function after Cardiac Surgery: From Phenotypes to Mechanisms," *Anesthesiology*, vol. 129, no. 4, pp. 829–851, Oct. 2018.

[20] J. H. Silverstein, M. Timberger, D. L. Reich, and S. Uysal, "Central nervous system dysfunction after noncardiac surgery and anesthesia in the elderly," *Anesthesiology*, vol. 106, no. 3, pp. 622–628, Mar. 2007.

[21] T. N. Robinson and B. Eiseman, "Postoperative delirium in the elderly: diagnosis and management," *Clinical Interventions in Aging*, vol. 3, no. 2, pp. 351–355, Jun. 2008. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2546478/

[22] M. J. Demeure and M. J. Fain, "The Elderly Surgical Patient and Postoperative Delirium," *Journal of the American College of Surgeons*, vol. 203, no. 5, pp. 752–757, Nov. 2006. [Online]. Available: https://journals.lww.com/journalacs/Citation/2006/11000/The_Elderly_Surgical_Patient_and_Postoperative.22.aspx

[23] T. T. Hshieh, T. G. Fong, E. R. Marcantonio, and S. K. Inouye, "Cholinergic deficiency hypothesis in delirium: a synthesis of current evidence," *The Journals of Gerontology. Series A, Biological Sciences and Medical Sciences*, vol. 63, no. 7, pp. 764–772, Jul. 2008.

[24] G. Androsova, R. Krause, G. Winterer, and R. Schneider, "Biomarkers of postoperative delirium and cognitive dysfunction," *Frontiers in Aging Neuroscience*, vol. 7, p. 112, Jun. 2015. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4460425/

[25] S. H. Choi, H. Lee, T. S. Chung, K. M. Park, Y. C. Jung, S. I. Kim, and J. J. Kim, "Neural network functional connectivity during and after an episode of delirium," *American Journal of Psychiatry*, vol. 169, no. 5, pp. 498–507, May 2012. [Online]. Available: http://www.scopus.com/inward/record.url?scp=84860539606&partnerID=8YFLogxK

[26] L. Capuron and A. H. Miller, "Immune system to brain signaling: neuropsychopharmacological implications," *Pharmacology & Therapeutics*, vol. 130, no. 2, pp. 226–238, May 2011.

[27] C. P. Casey, H. Lindroth, R. Mohanty, Z. Farahbakhsh, T. Ballweg, S. Twadell, S. Miller, B. Krause, V. Prabhakaran, K. Blennow, H. Zetterberg, and R. D. Sanders, "Postoperative delirium is associated with increased plasma neurofilament light," *Brain: A Journal of Neurology*, vol. 143, no. 1, pp. 47–54, Jan. 2020.

[28] N. Pawar and O. L. Barreto Chang, "Burst Suppression During General Anesthesia and Postoperative Outcomes: Mini Review," *Frontiers in Systems Neuroscience*, vol. 15, p. 767489, 2021.

[29] B. A. Fritz, P. L. Kalarickal, H. R. Maybrier, M. R. Muench, D. Dearth, Y. Chen, K. E. Escallier, A. Ben Abdallah, N. Lin, and M. S. Avidan, "Intraoperative Electroencephalogram Suppression Predicts Postoperative Delirium," *Anesthesia and Analgesia*, vol. 122, no. 1, pp. 234–242, Jan. 2016.

[30] M. Momeni, S. Meyer, M.-A. Docquier, G. Lemaire, D. Kahn, C. Khalifa, M. Rosal Martins, M. Van Dyck, L.-M. Jacquet, A. Peeters, and C. Watremez, "Predicting postoperative delirium and postoperative cognitive decline with combined intraoperative electroencephalogram monitoring and cerebral near-infrared spectroscopy in patients undergoing cardiac interventions," *Journal of Clinical Monitoring and Computing*, vol. 33, no. 6, pp. 999–1009, Dec. 2019.

[31] M. Soehle, A. Dittmann, R. K. Ellerkmann, G. Baumgarten, C. Putensen, and U. Guenther, "Intraoperative burst suppression is associated with postoperative delirium following cardiac surgery: a prospective, observational study," *BMC anesthesiology*, vol. 15, p. 61, Apr. 2015.

[32] S. A. Safavynia, S. Arora, K. O. Pryor, and P. S. García, "An update on postoperative delirium: Clinical features, neuropathogenesis, and perioperative management," *Current anesthesiology reports*, vol. 8, no. 3, pp. 252–262, Jul. 2018. [Online]. Available: https://europepmc.org/articles/PMC6290904

[33] G. Narula, M. Haeberlin, J. Balsiger, C. Strässle, L. L. Imbach, and E. Keller, "Detection of EEG burst-suppression in neurocritical care patients using an unsupervised machine learning algorithm," *Clinical Neurophysiology: Official Journal of the International Federation of Clinical Neurophysiology*, vol. 132, no. 10, pp. 2485–2492, Oct. 2021.

[34] R. Gutierrez, J. I. Egaña, I. Saez, F. Reyes, C. Briceño, M. Venegas, I. Lavado, and A. Penna, "Intraoperative Low Alpha Power in the Electroencephalogram Is Associated With Postoperative Subsyndromal Delirium," *Frontiers in Systems Neuroscience*, vol. 13, p. 56, 2019.

[35] S. Koch, I. Feinkohl, S. Chakravarty, V. Windmann, G. Lichtner, T. Pischon, E. N. Brown, C. Spies, and BioCog Study Group, "Cognitive Impairment Is Associated with Absolute Intraoperative Frontal alpha-Band Power but Not with Baseline alpha-Band Power: A Pilot Study," *Dementia and Geriatric Cognitive Disorders*, vol. 48, no. 1-2, pp. 83–92, 2019.

[36] B. Palanca, T. Wildes, Y. Ju, S. Ching, and M. Avidan, "Electroencephalography and delirium in the postoperative period," *BJA: British Journal of Anaesthesia*, vol. 119, no. 2, pp. 294–307, Aug. 2017. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6172974/

[37] E. Tassonyi, E. Charpantier, D. Muller, L. Dumont, and D. Bertrand, "The role of nicotinic acetylcholine receptors in the mechanisms of anesthesia," *Brain Research Bulletin*, vol. 57, no. 2, pp. 133–150, Jan. 2002.

[38] V. Fodale, D. Quattrone, C. Trecroci, V. Caminiti, and L. B. Santamaria, "Alzheimer's disease and anaesthesia: implications for the central cholinergic system," *British Journal of Anaesthesia*, vol. 97, no. 4, pp. 445–452, Oct. 2006.

[39] C. M. Giattino, J. E. Gardner, F. M. Sbahi, K. C. Roberts, M. Cooter, E. Moretti, J. N. Browndyke, J. P. Mathew, M. G. Woldorff, M. Berger, and MADCO-PC Investigators, "Intraoperative Frontal Alpha-Band Power Correlates with Preoperative Neurocognitive Function in Older Adults," *Frontiers in Systems Neuroscience*, vol. 11, p. 24, 2017.

[40] P. L. Purdon, E. T. Pierce, E. A. Mukamel, M. J. Prerau, J. L. Walsh, K. F. K. Wong, A. F. Salazar-Gomez, P. G. Harrell, A. L. Sampson, A. Cimenser, S. Ching, N. J. Kopell, C. Tavares-Stoeckel, K. Habeeb, R. Merhar, and E. N. Brown, "Electroencephalogram signatures of loss and recovery of consciousness from propofol," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 110, no. 12, pp. E1142–E1151, Mar. 2013. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3607036/

[41] "Greek Alphabet Soup – Making Sense of EEG Bands." [Online]. Available: http://neurosky.com/2015/05/greek-alphabet-soup-making-sense-of-eeg-bands/

[42] "Alpha wave," Mar. 2022, page Version ID: 1080145273. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Alpha_wave&oldid=1080145273

[43] "Beta wave," Nov. 2021, page Version ID: 1057429741. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Beta_wave&oldid=1057429741

[44] "Gamma wave," Feb. 2022, page Version ID: 1073289182. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Gamma_wave&oldid=1073289182

[45] "Delta wave," Mar. 2022, page Version ID: 1080125436. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Delta_wave&oldid=1080125436

[46] "Theta wave," Dec. 2021, page Version ID: 1059271818. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Theta_wave&oldid=1059271818

[47] R. Blagus and L. Lusa, "SMOTE for high-dimensional class-imbalanced data," *BMC Bioinformatics*, vol. 14, p. 106, Mar. 2013. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3648438/

[48] O. Kramer, "K-Nearest Neighbors," in *Dimensionality Reduction with Unsupervised Nearest Neighbors*, O. Kramer, Ed. Berlin, Heidelberg: Springer, 2013, pp. 13–23. [Online]. Available: https://doi.org/10.1007/978-3-642-38652-7_2

[49] B. A. Khan, A. J. Perkins, S. Gao, S. L. Hui, N. L. Campbell, M. O. Farber, L. L. Chlan, and M. A. Boustani, "The CAM-ICU-7 Delirium Severity Scale: A Novel Delirium Severity Instrument for Use in the Intensive Care Unit," *Critical care medicine*, vol. 45, no. 5, pp. 851–857, May 2017. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5392153/

[50] S. K. Inouye, C. H. van Dyck, C. A. Alessi, S. Balkin, A. P. Siegal, and R. I. Horwitz, "Clarifying confusion: the confusion assessment method. A new method for detection of delirium," *Annals of Internal Medicine*, vol. 113, no. 12, pp. 941–948, Dec. 1990.

[51] "EEGLAB Extensions." [Online]. Available: https://eeglab.org/others/EEGLAB_Extensions.html

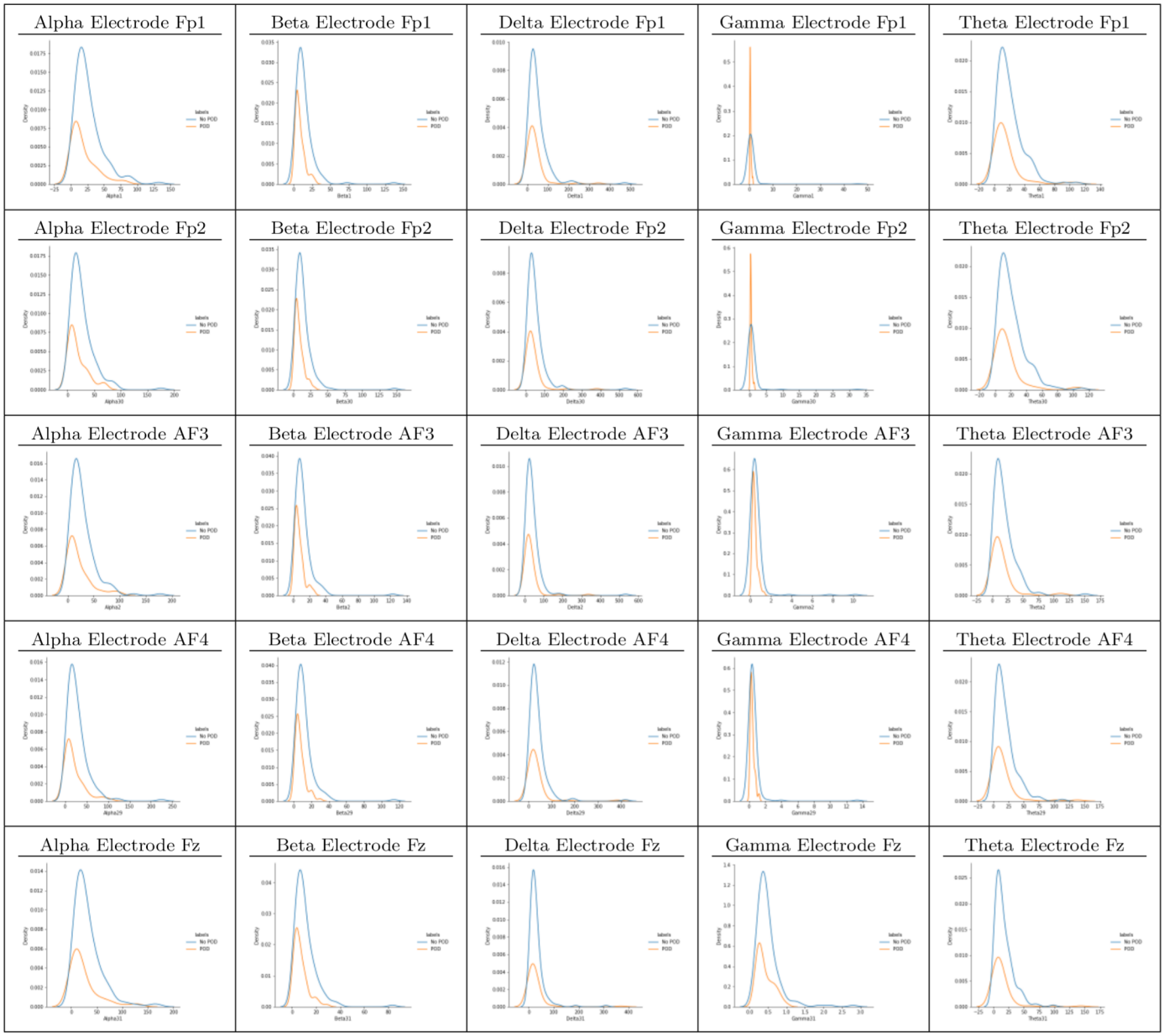
[52] N. Roy, R. Barry, F. Fernandez, C. Lim, M. Al-Dabbas, D. Karamacoska, S. Broyd, N. Solowij, C. Chiu, and G. Steiner, "Electrophysiological correlates of the brain-derived neurotrophic factor (BDNF) Val66Met polymorphism," *Scientific Reports*, vol. 10, Oct. 2020.

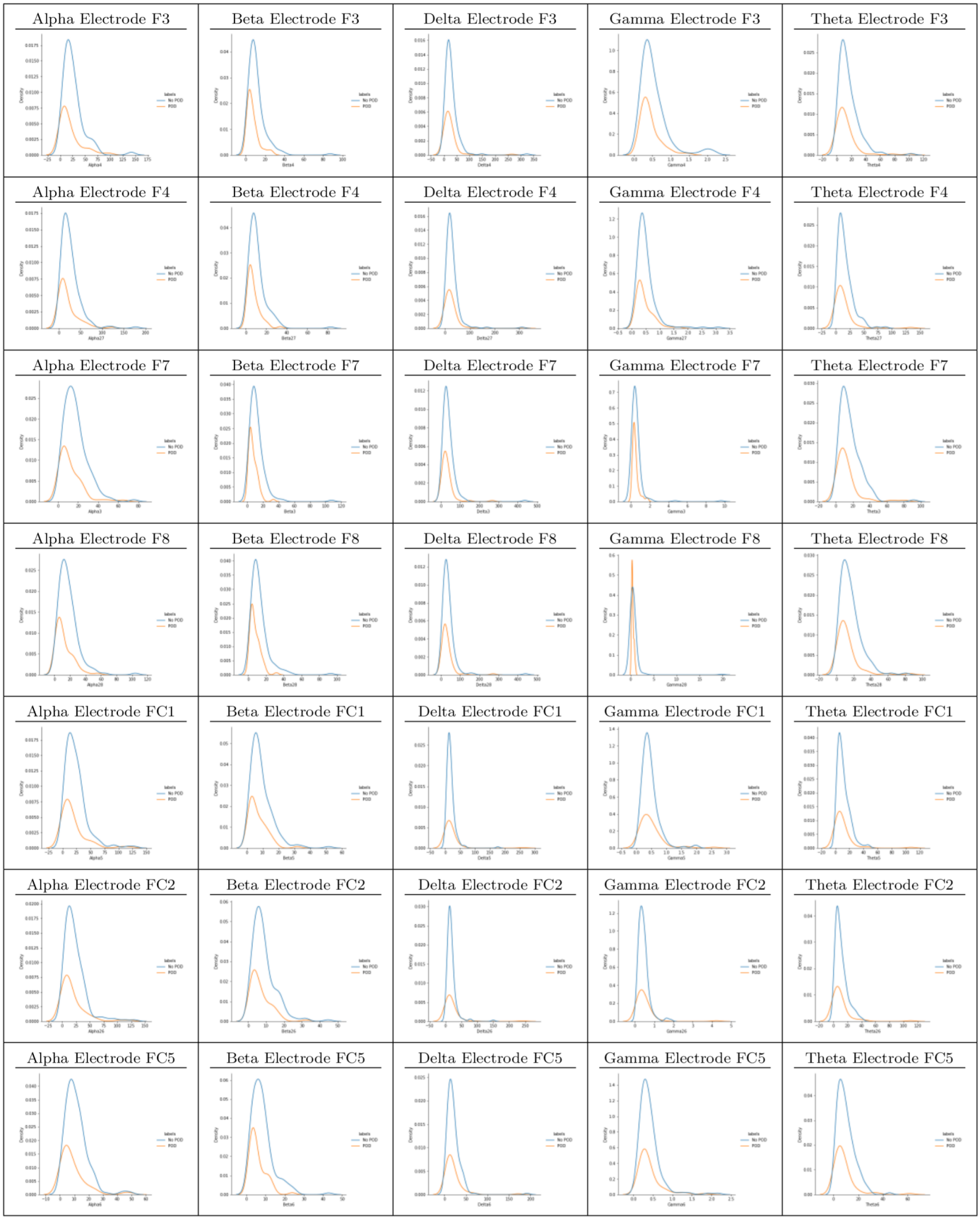
[53] W. Yang, J. Yang, Y. Gao, X. Tang, R. Yanna, S. Takahashi, and J. Wu, "Effects of Sound Frequency on Audiovisual Integration: An Event-Related Potential Study," *PLOS ONE*, vol. 10, p. e0138296, Sep. 2015.

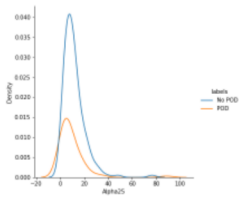
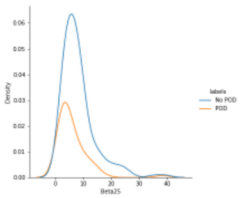
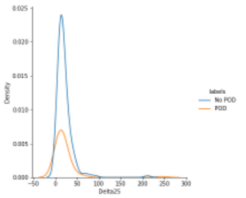
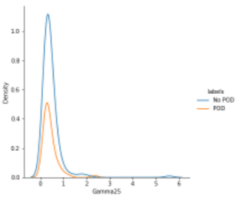
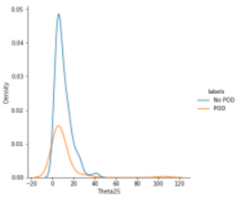
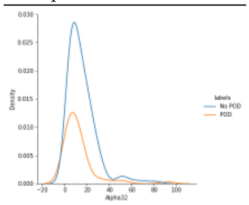
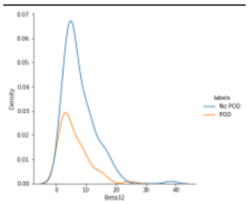
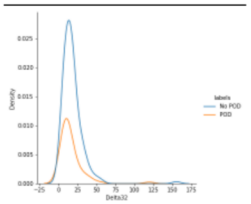
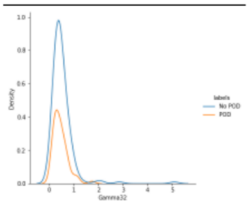
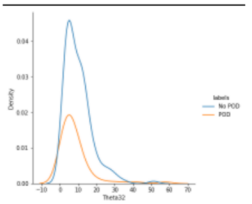
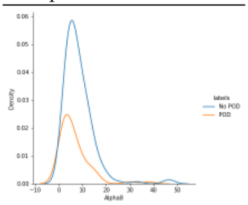
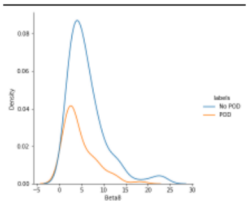
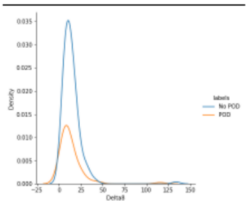
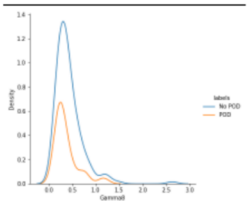
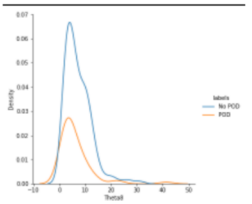
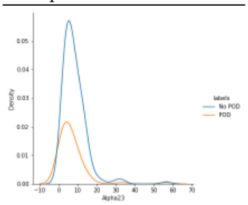
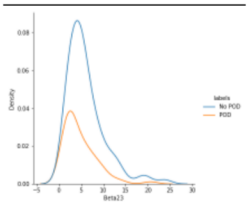
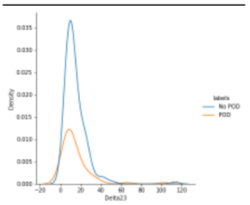
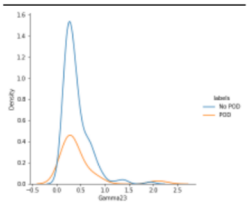
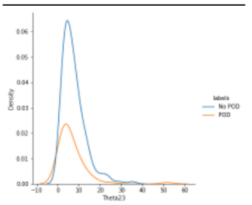
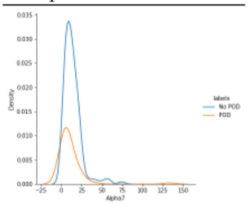
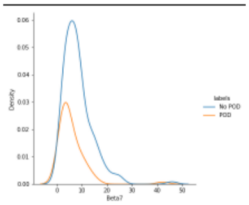
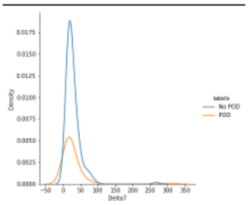
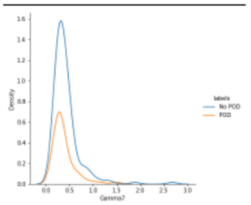
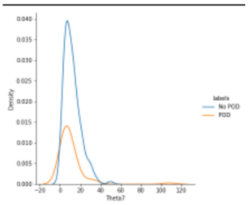
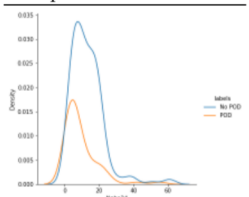
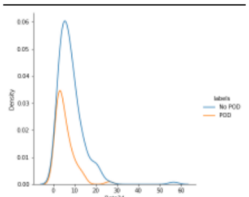
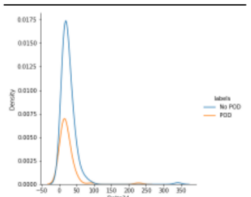
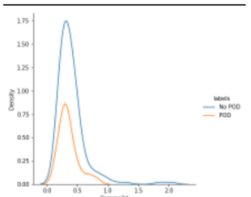
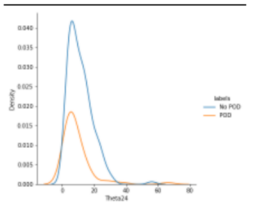
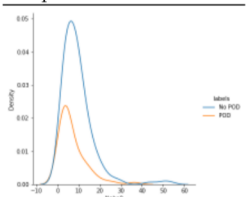
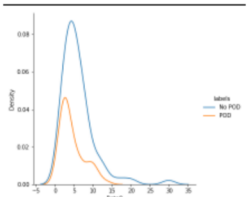
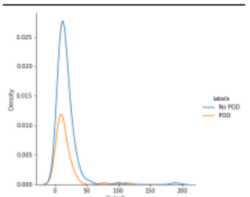
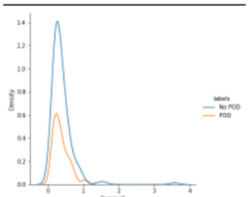
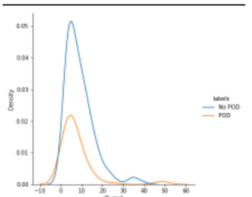
[54] I. Dowding, S. Haufe, and M. Tangermann, "Automatic Classification of Artifactual ICA-Components for Artifact Removal in EEG Signals," *Behavioral and brain functions : BBF*, vol. 7, p. 30, Aug. 2011.

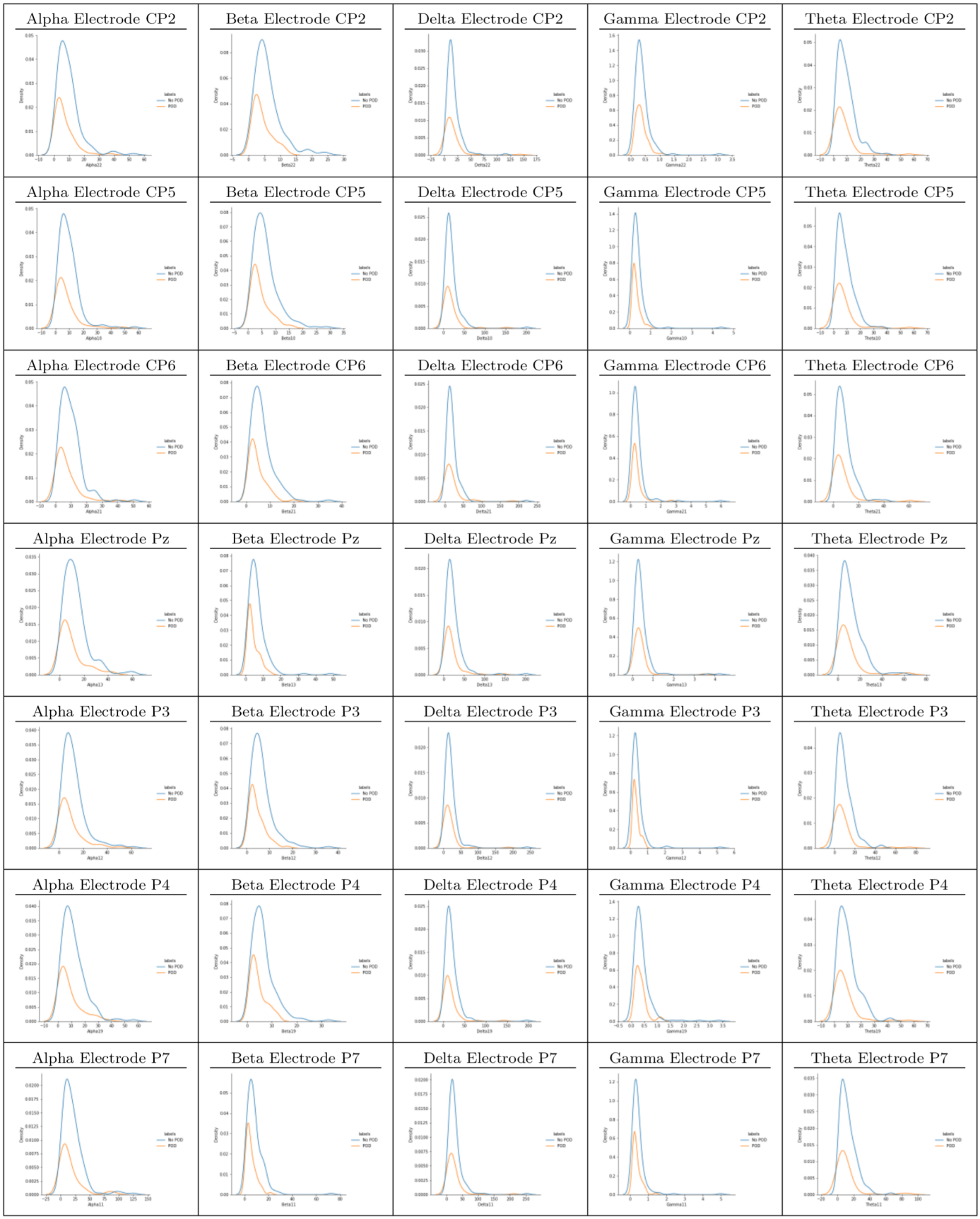
Appendix A

Kernel Density plots




Alpha Electrode F3
Beta Electrode F3
Delta Electrode F3
Gamma Electrode F3
Theta Electrode F3
Alpha Electrode F4
Beta Electrode F4
Delta Electrode F4
Gamma Electrode F4
Theta Electrode F4
Alpha Electrode F7
Beta Electrode F7
Delta Electrode F7
Gamma Electrode F7
Theta Electrode F7
Alpha Electrode F8
Beta Electrode F8
Delta Electrode F8
Gamma Electrode F8
Theta Electrode F8
Alpha Electrode FC1
Beta Electrode FC1
Delta Electrode FC1
Gamma Electrode FC1
Theta Electrode FC1
Alpha Electrode FC2
Beta Electrode FC2
Delta Electrode FC2
Gamma Electrode FC2
Theta Electrode FC2
Alpha Electrode FC5
Beta Electrode FC5
Delta Electrode FC5
Gamma Electrode FC5
Theta Electrode FC5

Alpha Electrode FC6	Beta Electrode FC6	Delta Electrode FC6	Gamma Electrode FC6	Theta Electrode FC6
				
Alpha Electrode Cz	Beta Electrode Cz	Delta Electrode Cz	Gamma Electrode Cz	Theta Electrode Cz
				
Alpha Electrode C3	Beta Electrode C3	Delta Electrode C3	Gamma Electrode C3	Theta Electrode C3
				
Alpha Electrode C4	Beta Electrode C4	Delta Electrode C4	Gamma Electrode C4	Theta Electrode C4
				
Alpha Electrode T7	Beta Electrode T7	Delta Electrode T7	Gamma Electrode T7	Theta Electrode T7
				
Alpha Electrode T8	Beta Electrode T8	Delta Electrode T8	Gamma Electrode T8	Theta Electrode T8
				
Alpha Electrode CP1	Beta Electrode CP1	Delta Electrode CP1	Gamma Electrode CP1	Theta Electrode CP1
				

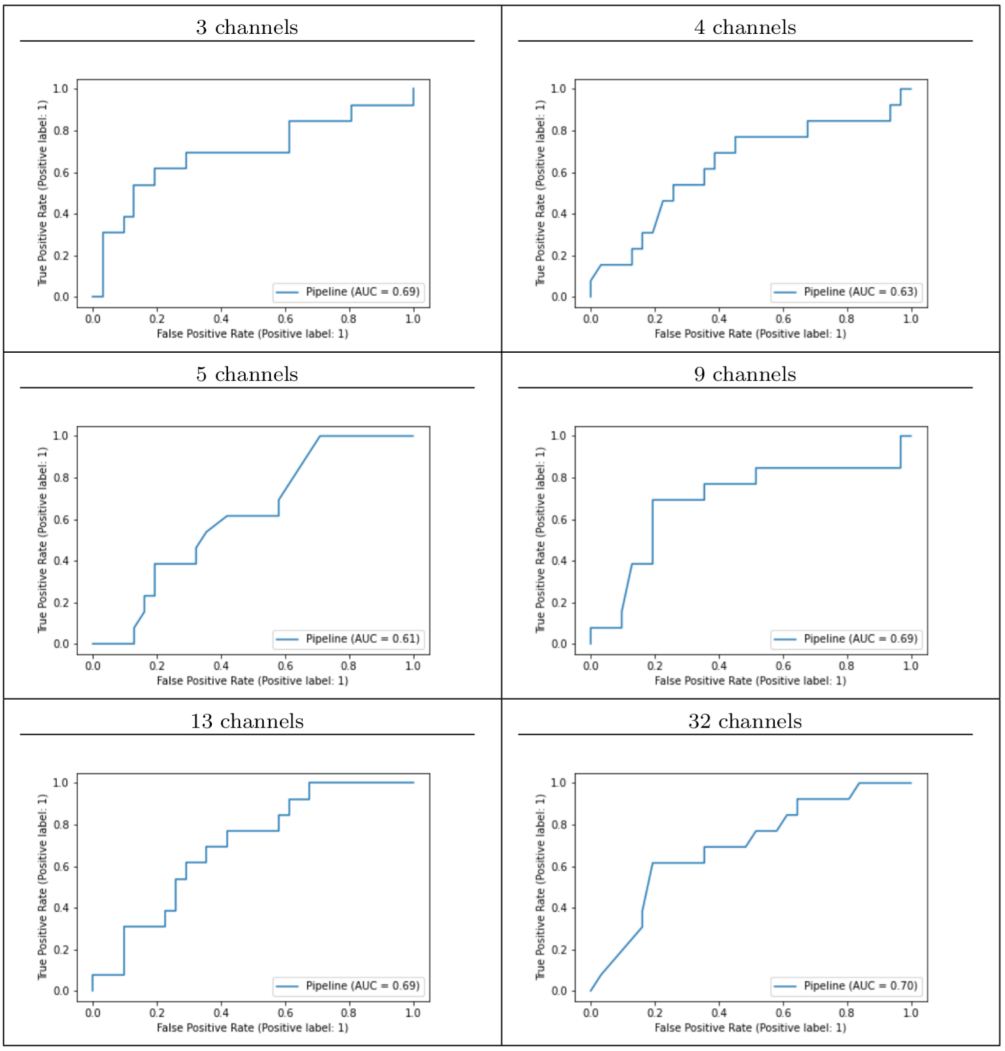

Alpha Electrode CP2
Beta Electrode CP2
Delta Electrode CP2
Gamma Electrode CP2
Theta Electrode CP2
Alpha Electrode CP5
Beta Electrode CP5
Delta Electrode CP5
Gamma Electrode CP5
Theta Electrode CP5
Alpha Electrode CP6
Beta Electrode CP6
Delta Electrode CP6
Gamma Electrode CP6
Theta Electrode CP6
Alpha Electrode Pz
Beta Electrode Pz
Delta Electrode Pz
Gamma Electrode Pz
Theta Electrode Pz
Alpha Electrode P3
Beta Electrode P3
Delta Electrode P3
Gamma Electrode P3
Theta Electrode P3
Alpha Electrode P4
Beta Electrode P4
Delta Electrode P4
Gamma Electrode P4
Theta Electrode P4
Alpha Electrode P7
Beta Electrode P7
Delta Electrode P7
Gamma Electrode P7
Theta Electrode P7
Density
labels
No POD
POD

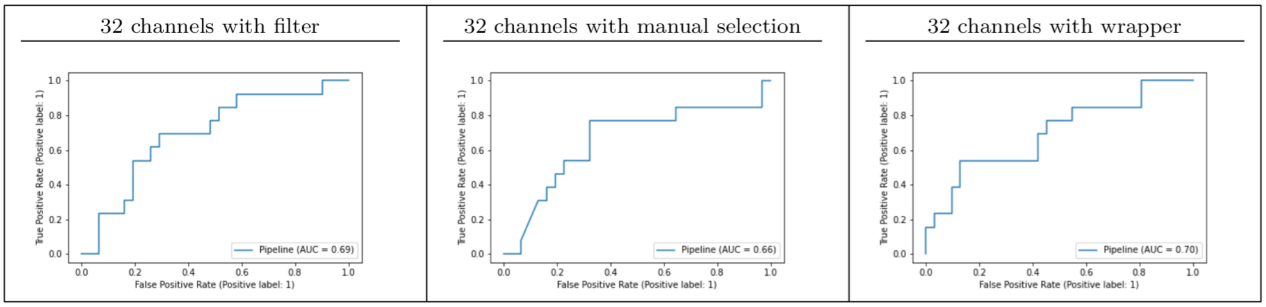
Alpha Electrode P8	Beta Electrode P8	Delta Electrode P8	Gamma Electrode P8	Theta Electrode P8
Alpha Electrode PO3	Beta Electrode PO3	Delta Electrode PO3	Gamma Electrode PO3	Theta Electrode PO3
Alpha Electrode PO4	Beta Electrode PO4	Delta Electrode PO4	Gamma Electrode PO4	Theta Electrode PO4
Alpha Electrode Oz	Beta Electrode Oz	Delta Electrode Oz	Gamma Electrode Oz	Theta Electrode Oz
Alpha Electrode O1	Beta Electrode O1	Delta Electrode O1	Gamma Electrode O1	Theta Electrode O1
Alpha Electrode O2	Beta Electrode O2	Delta Electrode O2	Gamma Electrode O2	Theta Electrode O2

Appendix B

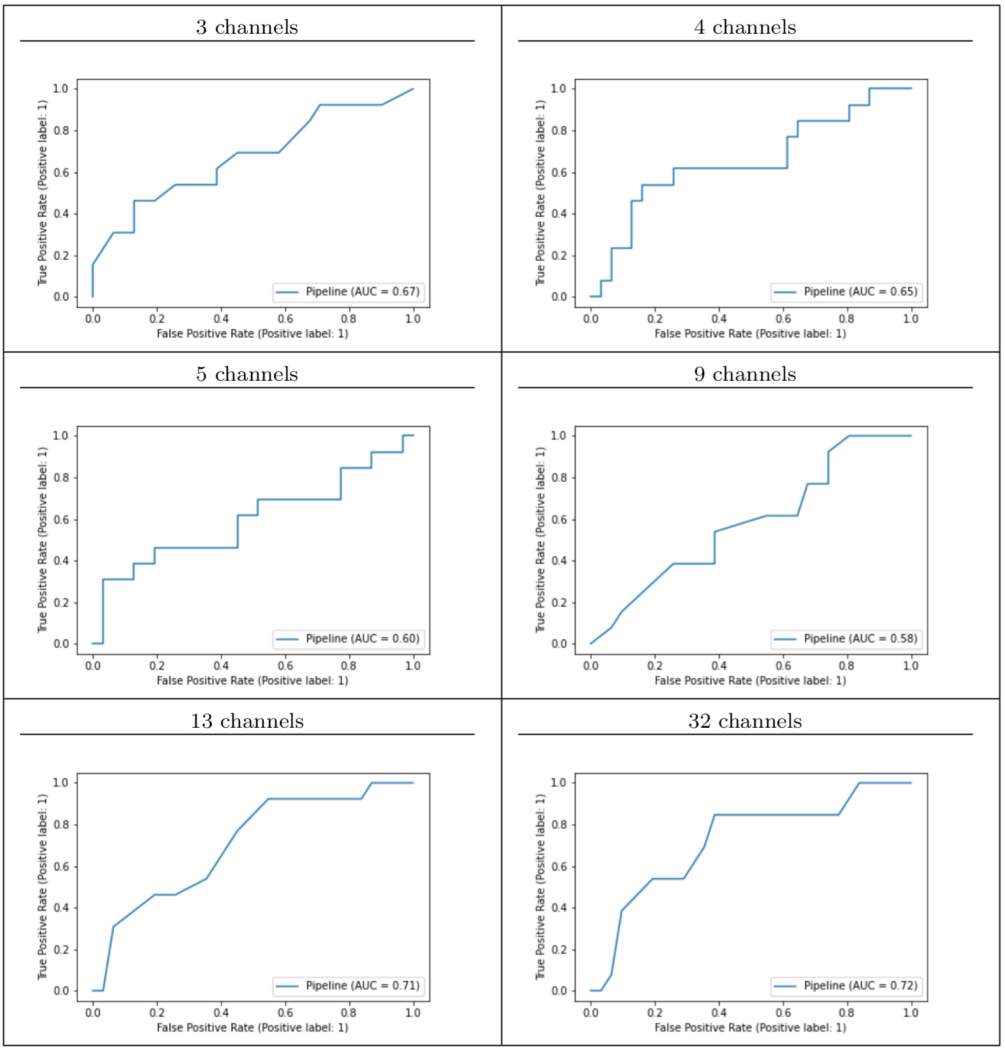
ROC Curves

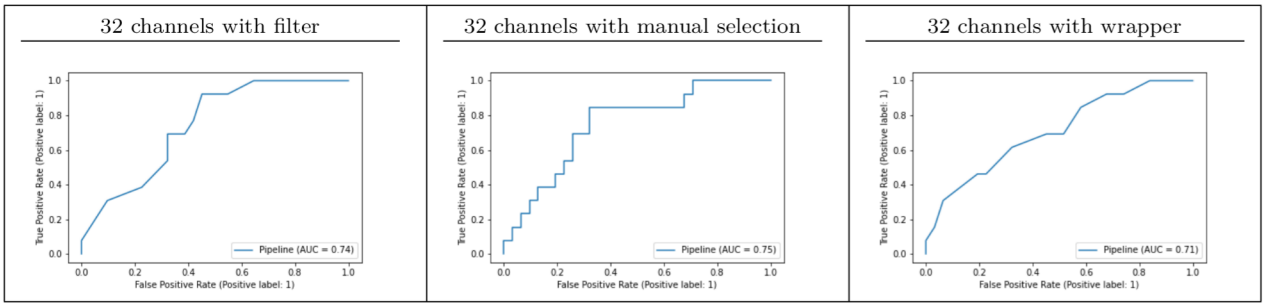
B.1 XGB



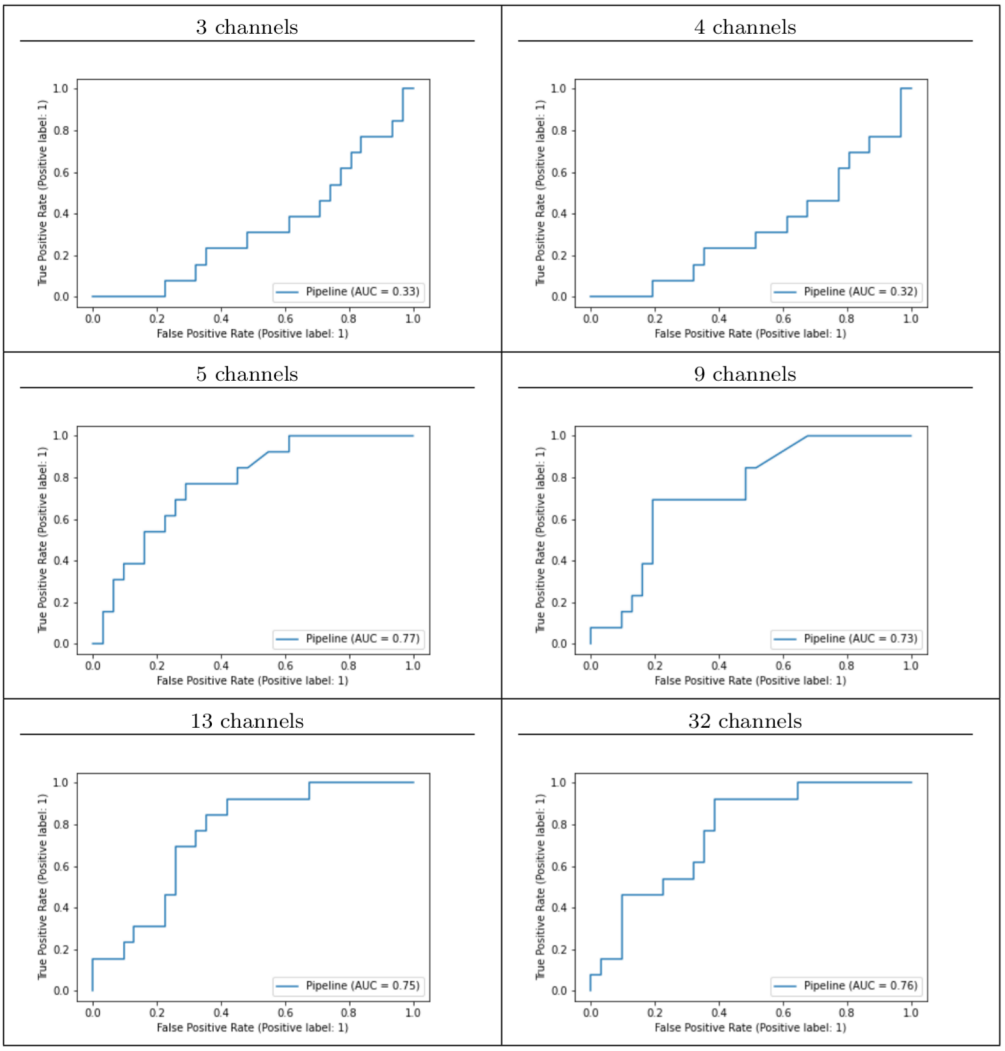

32 channels with filter
Pipeline (AUC = 0.69)
32 channels with manual selection
Pipeline (AUC = 0.66)
32 channels with wrapper
Pipeline (AUC = 0.70)
True Positive Rate (Positive label: 1)
False Positive Rate (Positive label: 1)

B.2 KNN

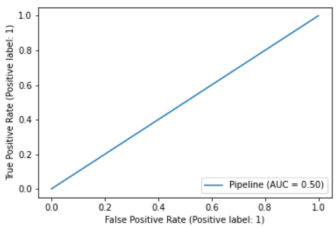

3 channels
Pipeline (AUC = 0.67)
4 channels
Pipeline (AUC = 0.65)
5 channels
Pipeline (AUC = 0.60)
9 channels
Pipeline (AUC = 0.58)
13 channels
Pipeline (AUC = 0.71)
32 channels
Pipeline (AUC = 0.72)
True Positive Rate (Positive label: 1)
False Positive Rate (Positive label: 1)


32 channels with filter
Pipeline (AUC = 0.74)
32 channels with manual selection
Pipeline (AUC = 0.75)
32 channels with wrapper
Pipeline (AUC = 0.71)
True Positive Rate (Positive label: 1)
False Positive Rate (Positive label: 1)

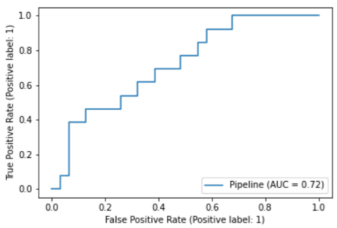
B.3 SVM


3 channels
Pipeline (AUC = 0.33)
4 channels
Pipeline (AUC = 0.32)
5 channels
Pipeline (AUC = 0.77)
9 channels
Pipeline (AUC = 0.73)
13 channels
Pipeline (AUC = 0.75)
32 channels
Pipeline (AUC = 0.76)
True Positive Rate (Positive label: 1)
False Positive Rate (Positive label: 1)

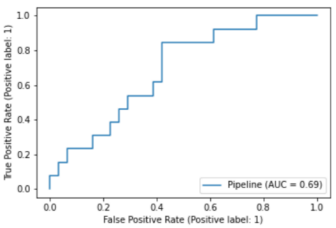
32 channels with filter



32 channels with manual selection

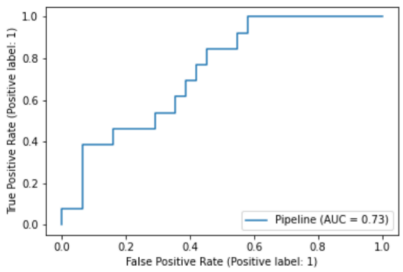


32 channels with wrapper

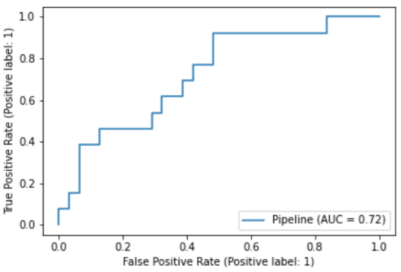


B.4 RF

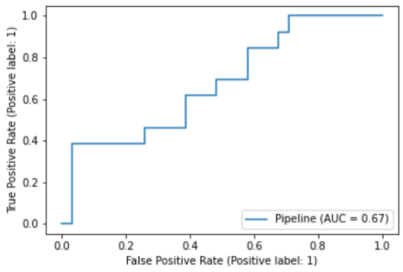
3 channels



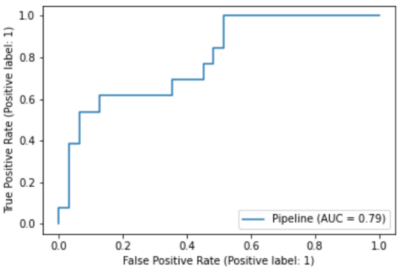
4 channels



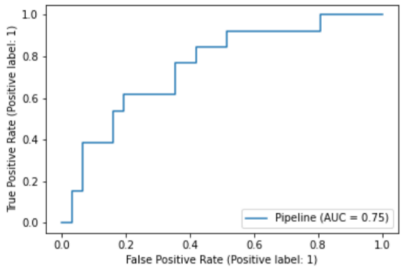
5 channels



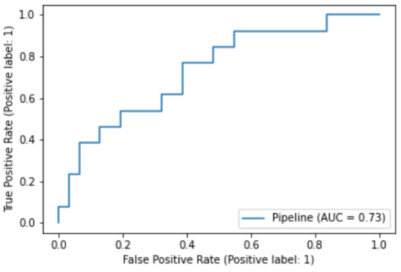
9 channels



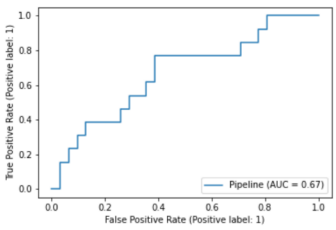
13 channels



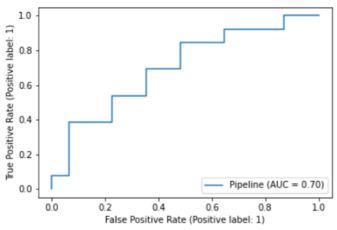
32 channels



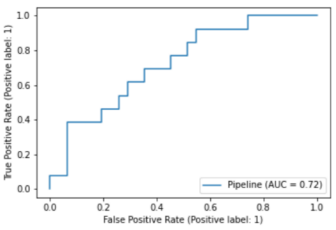
32 channels with filter



32 channels with manual selection

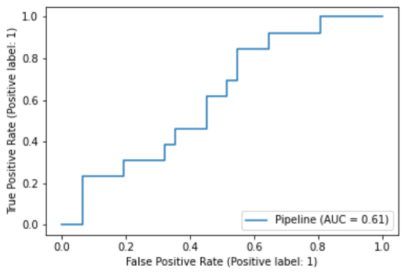


32 channels with wrapper

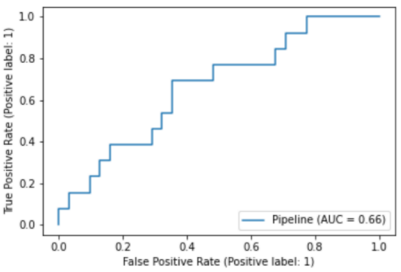


B.5 LDA

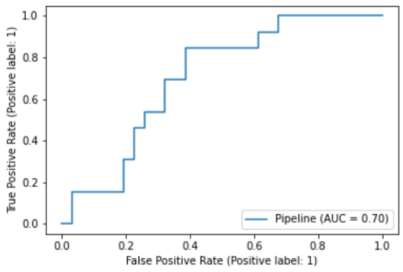
3 channels



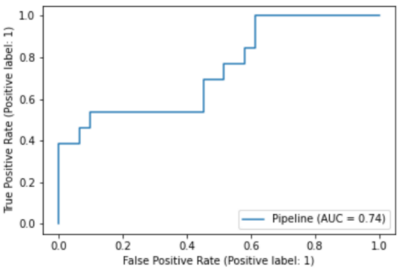
4 channels



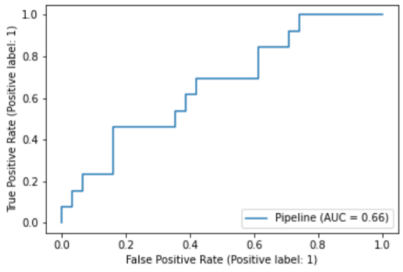
5 channels



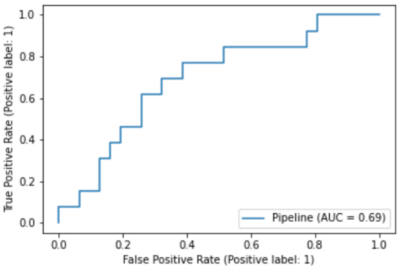
9 channels

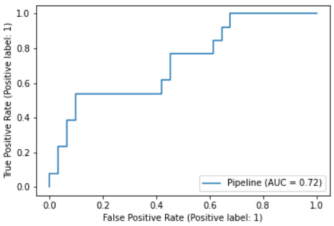
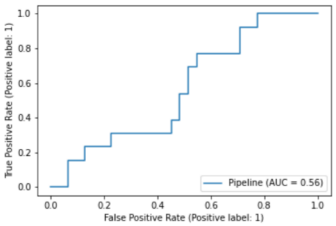
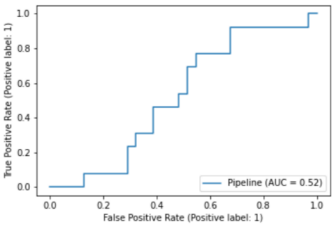


13 channels



32 channels



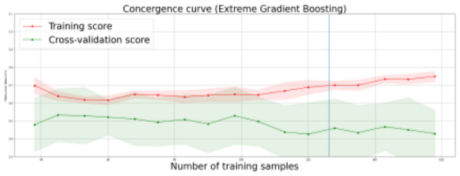
32 channels with filter	32 channels with manual selection	32 channels with wrapper
		

Appendix C

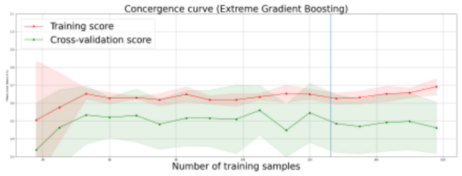
Convergence curves

C.1 XGB

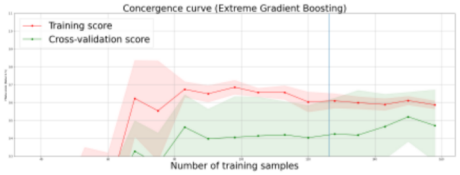
3 channels



4 channels

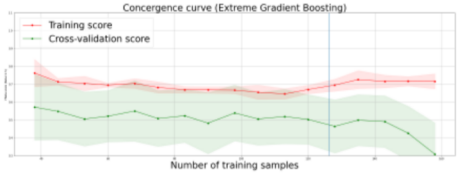


5 channels



9 channels

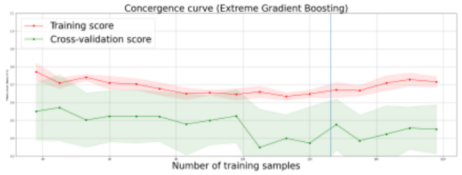
13 channels



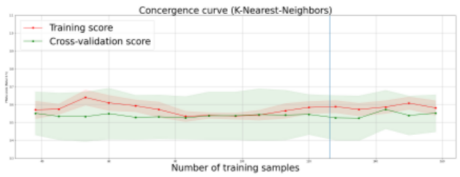
32 channels

32 channels with filter

32 channels with manual selection

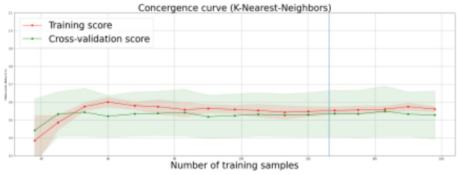


32 channels with wrapper

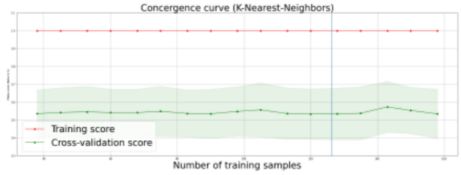


C.2 KNN

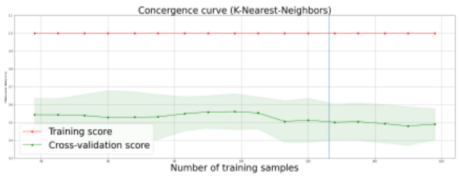
3 channels



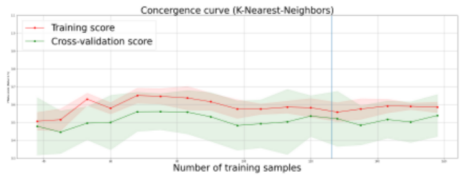
4 channels



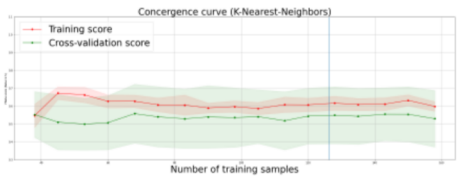
5 channels



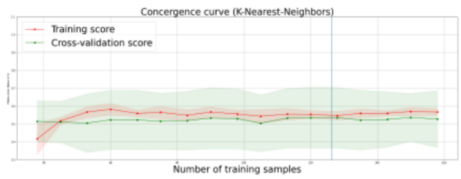
9 channels



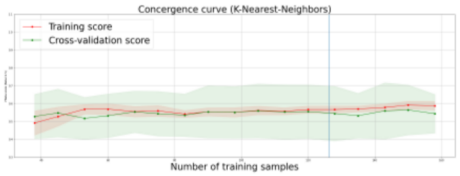
13 channels



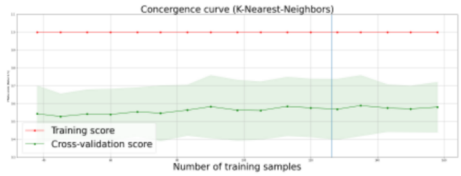
32 channels



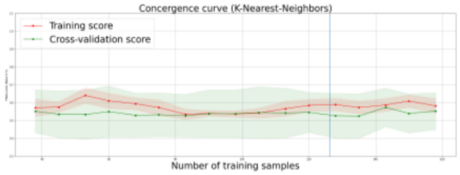
32 channels with filter



32 channels with manual selection

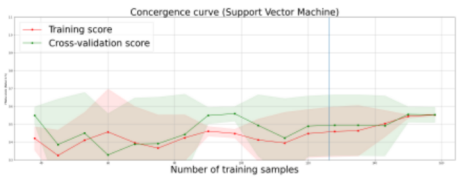


32 channels with wrapper

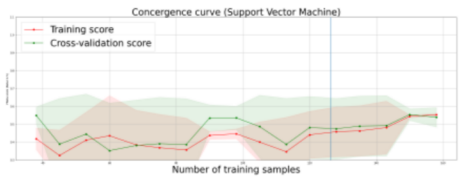


C.3 SVM

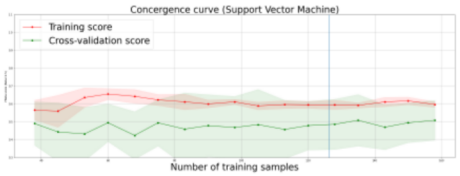
3 channels



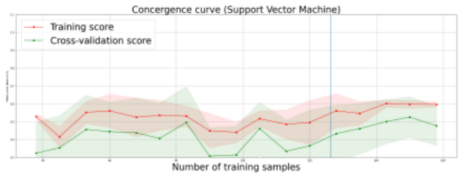
4 channels



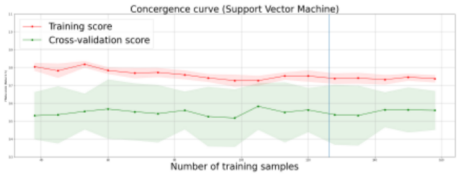
5 channels



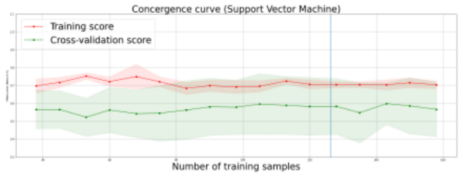
9 channels



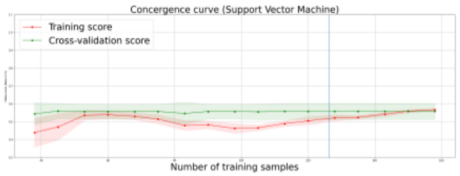
13 channels



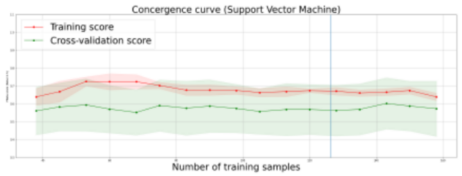
32 channels



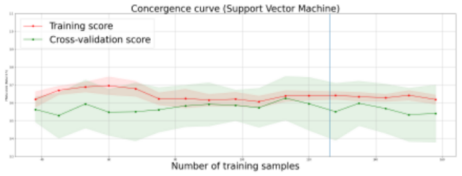
32 channels with filter



32 channels with manual selection



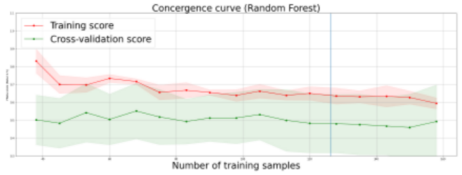
32 channels with wrapper



C.4 RF

3 channels

4 channels



5 channels

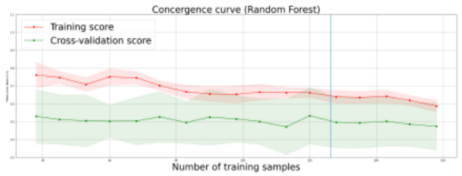
9 channels

13 channels

32 channels

32 channels with filter

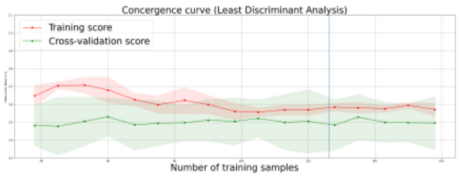
32 channels with manual selection



32 channels with wrapper

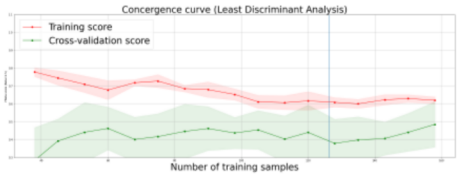
C.5 LDA

3 channels



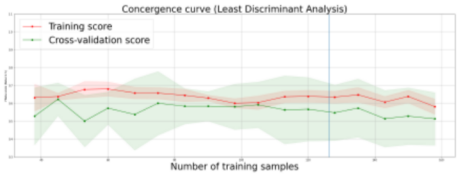
4 channels

5 channels

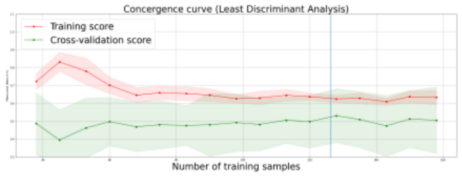


9 channels

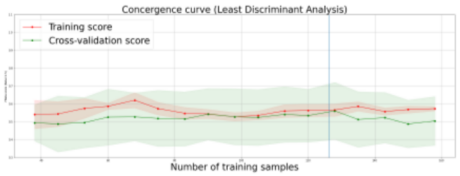
13 channels



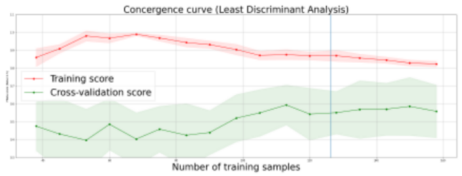
32 channels



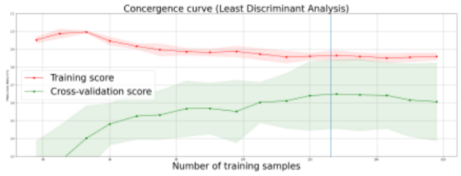
32 channels with filter



32 channels with manual selection



32 channels with wrapper



UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain
Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/epl**