POLITECNICO DI TORINO

Master's Degree in
BIOMEDICAL ENGINEERING

Master's Degree Thesis

IMPROVING EFFICIENCY AND ROBUSTNESS IN INFANT EEG PREPROCESSING

An Optimized Pipeline

Supervisors

Prof. Gabriella OLMO

Prof. Ghislaine DEHAENE-LAMBERTZ

PhD Jhunlyn LORENZO

PhD Ana FLÓ

Candidate

Nicolò FORMENTO MOLETTA

JULY 2025

Acknowledgments

I want to express my deepest gratitude to everyone who supported me throughout my master's thesis journey.
Firstly, I am profoundly grateful to M.D., Ph.D. Ghislaine Dehaene-Lambertz, my supervisor at NeuroSpin. Thank you for welcoming me into your research team and for your exceptional guidance. Your extensive knowledge of infant cognition and innovative ideas for exploring neuro-development significantly enriched my learning experience. My sincere thanks also go to Ph.D. Jhunlyn Lorenzo, whose consistent patience and invaluable guidance were instrumental throughout the majority of this internship. I am also thankful to Ph.D. Ana Flò for her invaluable help in reviewing this manuscript and assessing its clarity. Finally, I extend my appreciation to Pr. Gabriella Olmo, M.D., Surgeon, for her remote support during these past months of work.
I would like to express my kind gratitude to the Erasmus+ program, which allowed my exchange between the Polytechnic University of Turin and Léonard de Vinci Graduate School of Engineering. This experience was truly transformative, expanding my horizons and providing invaluable lessons. I am also immensely grateful to the teaching and administrative teams at both the universities which welcomed me in these two years. Thank you for assisting with all formalities and for consistently monitoring my progress throughout my master's thesis path.
I also want to thank all my colleagues with whom I "colonized" the "monkey open-space" during these months. Our Friday runs kept us active, and the no-layer Tiramisu was always a delight.
Outside of academia, I want to extend my gratitude to the community I found in Paris, especially the choir members. Engaging with you in the real world provided a wonderful balance and much-needed respite.
Finally, a big thank you to my family and friends for their support and encouragement in all my career choices. Your belief in me has been a constant source of strength.

Nicolò
June 2025, Paris

Abstract

Infant Electroencephalogram (EEG) preprocessing presents unique challenges due to the characteristics of the signal and the higher presence of artifacts. To address these difficulties, the MATLAB-based APICE pipeline was developed. Recognizing the limitations of a proprietary platform, APICE-Py was developed as its fully open-source Python counterpart. This translation aims to enhance accessibility, foster collaboration among research teams, and integrate seamlessly with modern machine learning tools.

The shift in programming languages necessitated key modifications, notably the parallelization of the Spherical Spline Interpolation (SSI), which dramatically improved computational efficiency for large datasets compared to the original MNE-Python library function. While numerical differences in filter implementations led APICE-Py to be more conservative in the number of retained epochs, validation demonstrated comparable performance. Using the Standardized Measurement Error (SME), no significant statistical differences were found in the extracted Evoked Response Potential (ERP) between the Python and MATLAB versions. Computational time analyses confirmed that the parallelized Python version achieves similar processing speeds.

APICE-Py is the fully open-source Python counterpart of the APICE pipeline. It democratizes robust infant EEG preprocessing, fosters collaboration, and promotes broader research and clinical applications. The pipeline is openly available in the NeuroKidsLab GitHub repository: https://github.com/neurokidslab/apice-py.

Keywords: Preprocessing, Artifact correction, EEG, ERP, Infant, Newborns, Development

Contents

List of Figures . . . . . **I**

List of Tables . . . . . **III**

1 Introduction **1**

2 Theoretical background **3**
2.1 Historical parenthesis 3
2.2 Action potential 4
2.3 EEG 7
2.4 Evoked Response Potential 10

3 Related work **13**
3.1 Literature Review 14

4 Methodology **19**
4.1 Pipeline Overview 19
4.1.1 Raw data import 19
4.1.2 Filtering 20
4.1.3 Artifacts detection 21
4.1.4 Correction of artifacts 24
4.1.5 Bad epochs definition 25
4.1.6 Preprocessing report 26
4.2 Python Implementation 26
4.2.1 Translation Process 26
4.2.2 Challenges & Workarounds 27
4.3 Datasets 27
4.3.1 Dataset 1: neonates auditory task 27
4.3.2 Dataset 2: 5-months old infants visual task 28
4.3.3 Dataset 3: 7-years old children and their parents collaborative task 28
4.4 Validation and Results 29
4.4.1 Comparison with MATLAB Outputs 29

5 Discussion **37**

6 Conclusion and Future Work **39**

A Appendix A . . . . . **VII**

Bibliography . . . . . **IX**

List of Figures

2.1 An example of the first human EEG signal. Image taken from Berger 1929. 4

2.2 Three drawings of cortical lamination by Santiago Ramon y Cajal, each showing a vertical cross-section, with the surface of the cortex at the top, with explicit layers identification. Left: Nissl-stained visual cortex of a human adult. Middle: Nissl-stained motor cortex of a human adult. Right: Golgi-stained cortex of a 1.5 month-old infant. The Nissl stain shows the cell bodies of neurons; the Golgi stain shows the dendrites and axons of a random subset of neurons. 5

2.3 Drawing of Purkinje cells (A) and granule cells (B) from pigeon cerebellum by Santiago Ramón y Cajal, 1899. Instituto Santiago Ramón y Cajal, Madrid, Spain. 6

2.4 Computational model of neuron proposed by McCulloch and Pitts. The dendritic trees are represented by the synaptic weights, the soma as the summing junction, the membrane potential threshold by the activation function. 7

2.5 The action potential of a neuron cell. If the thresholds reached (a), the depolarization inverts the polarity of the cell membrane. It follows a re-polarization and hyper-polarization of the cell before the rest potential is achieved again. In case the stimulus is not strong enough, the action potential doesn't happen (b). Image adapted from Chris et al. 2007. 8

2.6 The influence of the cortical orientation of the neurons on the recorded signal amplitude. The optimal recording is when the neurons under the electrodes are perpendicular to the scalp and consequently to the electrode. Image taken from Brienza et al. 2019. 9

2.7 The average ERP of a 21-week-old infant in response to a central visual stimulus. The mid-occipital cluster of the electrodes used is presented below the plots. Image taken from Adibpour et al. 2018. 11

4.1 Overview of the proposed preprocessing pipeline usually applied on the infants' data. The preprocessing steps are grouped according to the functions implemented in the software pipeline. 20

4.2 Overview of the proposed preprocessing pipeline usually applied on the infants' data. 24
4.3 Grand average ERPs obtained using the Python and MATLAB with pipeline with the same parameters for artifact rejection. 31
4.4 SME with Wilcoxon Signed-Rank test to detect a possible statistical difference. 32
4.5 Percentage of retained epochs with Wilcoxon Signed-Rank test to detect a possible statistical difference. 32
4.6 Summary of the artifacts detected before (top row) and after (bottom row) the correction methods on the continuous data for Dataset 1. 33
4.7 Summary of the artifacts detected before (top row) and after (bottom row) the correction methods on the continuous data for Dataset 2. 33
4.8 Summary of the artifacts detected in the children (a) and parents (b) before and after the correction methods on the continuous data for Dataset 3. . . 34
4.9 Interpolation time requested by the pipeline to perform the SSI. The results are from a single subject taken from the Dataset 2 which is supposed to be the noisier among the three. It's evident that the MNE-Python method is unoptimized for large dataset, whereas the new implementation is reducing the computational time approaching the MATLAB as the cores used in the computation increases. 35

List of Tables

3.1 Summary of the filter parameters implemented by different pipelines. . . . 14
3.2 Summary of the main parameters and preprocessing step chosen by different pipelines. 17

1 Introduction

The concept of children as a "small adult" is increasingly discarded thanks to precious findings on their biological and cognitive development during their transition from infancy to adulthood. Their evolving cognitive functions are divided in several small steps which are the results of the interaction of exogenous stimuli which constitute the personal experience with endogenous genetically determined characteristics (Larcher 2015). This new perspective on neurodevelopment has generated significant interest, driving the research to better understand the relative contributions and their influential order.

During the years different behavioral studies were done to achieve a better understanding of the different stages spanning the extrema of human development, but only recently a quantitative and objective measure has been introduced in the study of this specific target population: the Electroencephalogram (EEG). This technology represents a valuable tool for investigating the high level brain functions with higher temporal resolution compared to other technologies such as the functional Magnetic Resonanse Imaging (fMRI) and at lower cost compared to Magnetoencephalography (MEG) remaining non-invasive. The downside of the EEG, however, is its high sensitivity to noise, especially the biological artifacts generated by subject movements, which can corrupt the signal, hiding the extractable information. When dealing with infants, the percentage of artifacts increases significantly when compared to the adult population due to their inability to remain completely still and the number of the trials is reduced for their lack of patience for long recording session.

Consequently, when dealing with infant EEG data preprocessing becomes a mandatory and pivotal step to be done before any further analysis to ensure the correct extraction of information. While the complexity of the experimental paradigms on cognition is increasing, requiring a higher number of trials, infants' inherent difficulty in remaining still persists. This makes a robust data recovery an essential need.

Different attempts were made over the last years to build the more suitable pipeline to identify and correct the data before more depth analysis. Up to now, the majority of proposed solutions to deal with infant EEG signals are MATLAB-based. Since it is a proprietary software, it influences negatively their broader applicability in the research field.

The aim of this work is to validate the open-source fully-automated pipeline Python-based to preprocess infant EEG data. The fulcrum of the new implementation is the MNE-Python library which is a massive collaborative effort, very popular for EEG analysis. Adapted from the MATLAB version, its innovation lies in increased age-scalability, free-availability and easier integration with machine learning due to the selected programming language. This work aims to democratize the access to robust infant EEG preprocessing, fostering broader research and clinical applications.
The different programming languages might lead to a different output even when the input data are the same. To objectively evaluate any possible difference with the new implementation, the comparison was made using the same metrics used to validate the MATLAB implementation. Since the objective is to avoid any performance decrease, other parameters, such as the computational time, were evaluated as well.
This work first provides a brief background, followed by a general description of the implemented pipeline and existing solutions. Finally, the results of the validation are presented.

2 Theoretical background

2.1 Historical parenthesis

The EEG played one of the major technological advancements which enabled a better understanding of the brain along with its underlying activity. The discovery of the electrical current as the so far invisible hand driving the brain function was achieved by small steps, starting with Galvani's discovery of muscle contraction when an electrical current was applied in 1828. From this peripheral point of view, in the mid-19^{th} century the concept swam upstream following the nerves which are the link between the central nervous system and the periphery. This was evident with the work of Du Bois, who described a negative wave propagating along the nerves and then to the muscle, which, nowadays, it is called action potential (Edu 2024), and Von Helmholtz, who provided a better characterization of this new finding by computing its speed which, with surprise, was smaller than the speed of the current ($\approx 25/40\ \frac{m}{s}$) (Edu n.d.). Finally, the brain electrical sensitivity was explored using the same idea that guided Galvani and the muscle: Fritsch and Hitzig applied a direct electrical stimulation of the cerebral cortex. The electrical stimulation on some areas of the cerebral cortex provoked some contraction in the opposite part of the body proving once again the contra laterality of the brain and the nature of the communication method employed by the cells. Playing on the current intensity, they managed to see also the correlation between the amplitude of the stimulation current and the extension of the contraction of the body. They detected the motor area of the cerebral cortex (Hagner 2012). The relationship between the current and the brain became stricter, but it's only with Caton in 1875 that the first electric signal of the brain was recorded, demonstrating that the gray matter of the cerebral cortex produces small currents. His experiences moved even further, allowing him to be the first to detect the visual Evoked Response Potential (ERP) in the occipital cortex of a rabbit when some light was directly pointed to its eyes (Haas 2003). In a single shot both the electroencephalographic and the evoked potential are born.

Up to now the substantial link between the brain activity and the current was established. The missing tile to complete the mosaic was the ability to measure this electro-magnetic

variable with a non-invasive method. The progress must be credited to Hans Berger who in 1924 placed some electrodes on the scalp of his son and recorded the electrical activity of the brain (Fig. 2.1), marking it as the first EEG done on a human subject (Haas 2003).

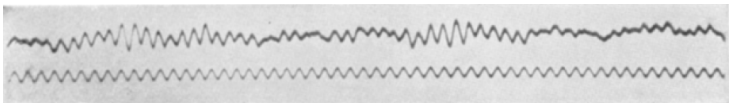


Figure 2.1: An example of the first human EEG signal. Image taken from Berger 1929.

He discovered the change in the waves frequency correlated with the mental work load required by the subject while the EEG was recorded, coining the two different conditions as "alpha" and "beta" waves and establishing a standard protocol for the acquisition of the differential signal. His work was furthered by Herbert H. Jasper in the 20^{th} century. With him, the clinical relevance of these non-invasive exams was introduced thanks to a proper definition of the electrodes' placement. With a better methodological setup, he was able to start exploring the state of consciousness, the developmental learning, along with some clinical pathologies such as the epilepsy (Avoli 2012). He also managed to monitor the activity of a single neuron, with the use of extracellular or intracellular microelectrodes. The power of the newly introduced technology was perfectly interpreted by William Grey Walter who redefined the interpretations of the different rhythms, being able to analyze them in real-time. He also introduced the localization of a brain tumor through the observation of the slow frequency rhythms and made progress on the epilepsy (Bladin 2006). The technological development of the last decades of the 20^{th} century and the beginning of the 21^{th} century allowed the EEG to increase its performances and reliability. With the digitalization of the acquired data, their manipulation became easier opening the method to be used outside the clinical environment. An increased availability of raw observations and new technical achievements allowed to define better experimental paradigms and thus to make inferences on explanatory concepts on the way the brain works. This field of research is known under the name of cognitive neuroscience. In children and infants, the use of fMRI or MEG is harder to implement and thus limited because movement severely disrupts the acquired signal, making EEG more suitable for these conditions.

2.2 Action potential

The fundamental cells of the brain are called neurons, and they were firstly described and drawn by Ramon Y Cajal after being stained by the "reazione nera" of Golgi (Fig. 2.3). The cell is made by the soma which is where the nucleus is stored, the dendrites which

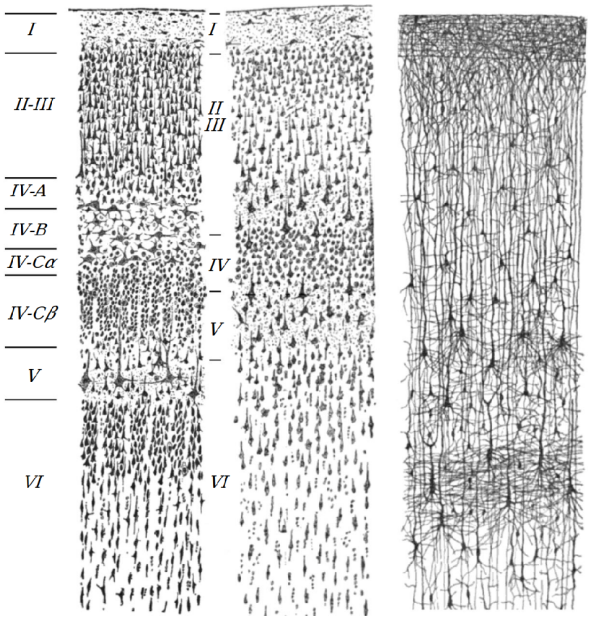


Figure 2.2: Three drawings of cortical lamination by Santiago Ramon y Cajal, each showing a vertical cross-section, with the surface of the cortex at the top, with explicit layers identification. Left: Nissl-stained visual cortex of a human adult. Middle: Nissl-stained motor cortex of a human adult. Right: Golgi-stained cortex of a 1.5 month-old infant. The Nissl stain shows the cell bodies of neurons; the Golgi stain shows the dendrites and axons of a random subset of neurons.

allow capturing the signals coming from the other neurons and the axon which is the road along which the response of the single neuron travels to the dendrites tree of the next neuron. As all the cells of the human body its membrane maintain a resting potential of -70 mV.
The electro-magnetic field captured by the EEG consists of the sum of all the depolarizations that occurs in the 6 layers of the pyramidal neurons under the probe electrode. These layers are numbered following a decreasing order from the innermost part (layer VI), close to the white matter, to the outermost part (layer I), just before the pia mater (Fig. 2.2) and in each layer a different type of neurons can be found. The results of their activity are all the high-level brain functions: cognition, sensory perception, spatial reasoning, generation of motor commands and language.

The communication between neurons is performed following different consequential steps. Following the information flow in a single brain cell, the different stages are:

- The structure that allows the exchange of information between the sender and the receiver is called a synapse and two general classes can be identified based on the nature of the medium used:

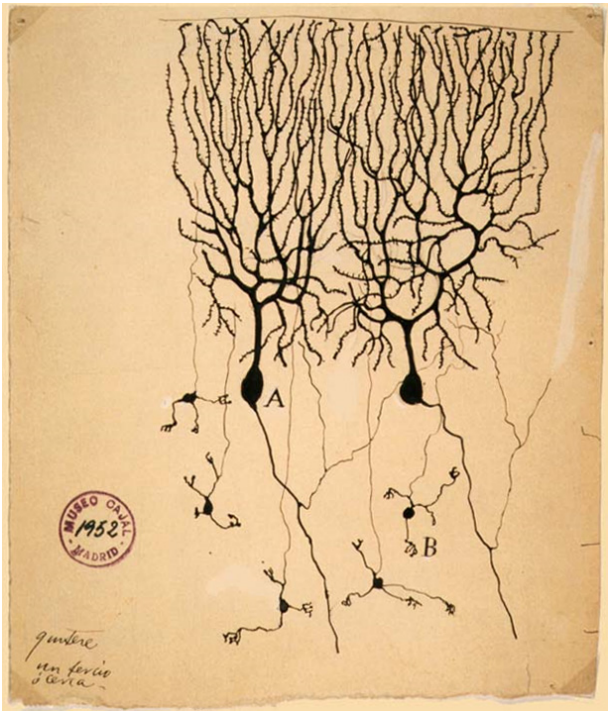


Figure 2.3: Drawing of Purkinje cells (A) and granule cells (B) from pigeon cerebellum by Santiago Ramón y Cajal, 1899. Instituto Santiago Ramón y Cajal, Madrid, Spain.

1. **Chemical synapses**:
   The cell to cell communication is guaranteed with the secretion and release of neurotransmitters in the synaptic gap or synaptic cleft which has a dimension between 20 and 40 μm. The neurotransmitters are then absorbed in the dendritic level of the receiver neuron. This kind of communication is slower, but allows modulating the information between an inhibitory and excitatory stimuli.

2. **Electrical synapses**:
   In this situation a direct, passive flow of electrical current flows from one neuron to another. This exchange is faster, and it depends on the existence of gap junctions, dedicated membrane channels connecting two neurons.

- The role of the dendritic tree is to assimilate all the different signals that comes from the sending neurons to the receiving neuron.

- At the level of the soma the veritable neuronal response is generated. The stimuli that came from the others neurons can increase or decrease the cell membrane potential. If the depolarization of the membrane increases to more than -55 mV a response is released with no more possibility to stop it following an all or none law. The response is called action potential.

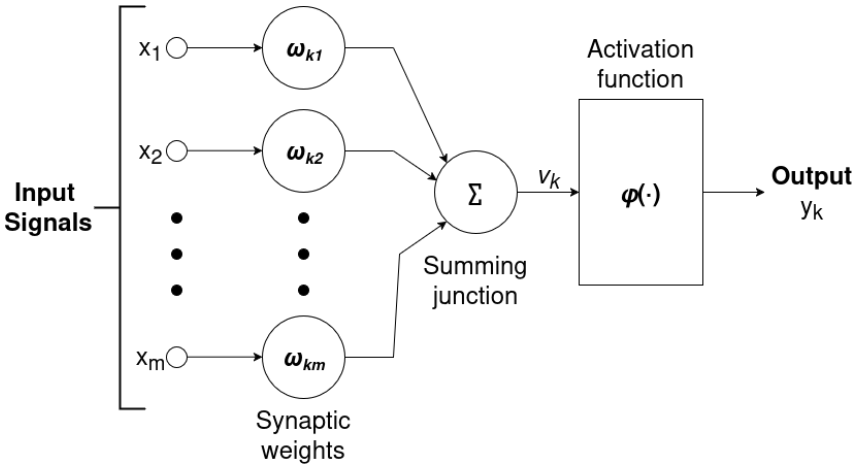


Figure 2.4: Computational model of neuron proposed by McCulloch and Pitts. The dendritic trees are represented by the synaptic weights, the soma as the summing junction, the membrane potential threshold by the activation function.

- The action potential moves from the soma to the axon of the neuron with a saltatory mechanism.
- When the action potential reaches the end tip of the axon, the information is given to the linked neurons' dendritic tree using one of the two synapses structures.

An oversimplification which allows a better understanding of the information elaboration of a neuron was given by Warren McCulloch and Walter Pitt in 1943 with the first computational model of a neuron (McCulloch et al. 1943) (Fig. 2.4).

The shape of an action potential is very well known, and it is always the same every time it occurs (Fig. 2.5). When the excitatory inputs which come from the dendrites reach the threshold, the cell membrane potential depolarizes rapidly up to 40 mV, followed by a hyperpolarization period that takes the membrane potential at a lower point ($\approx$ -90 mV) than the rest potential. During the hyperpolarization the neurons can't be excited again, this phase is called refractory period.

The depolarization follows the axon, and it happens at each node of Ranvier, points where the myelin is not present making possible for the ions to be exchanged between the inner and the outer part of the cells. At the tip of the axon, the depolarization causes the release of the neurotransmitter in the synaptic gap.

2.3 EEG

Even if it's possible to measure the activity of a single neuron (Avoli 2012), the information from its activity is generally not relevant enough to extract a correlation with a

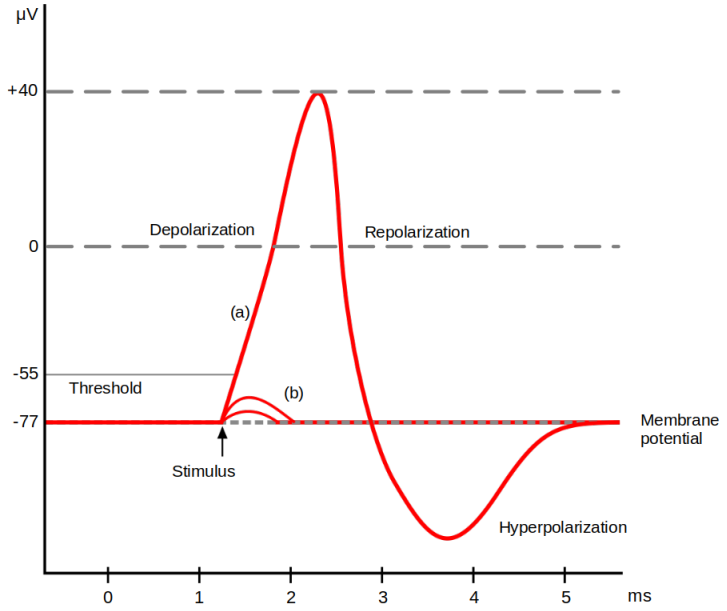


Figure 2.5: The action potential of a neuron cell. If the thresholds reached (a), the depolarization inverts the polarity of the cell membrane. It follows a re-polarization and hyper-polarization of the cell before the rest potential is achieved again. In case the stimulus is not strong enough, the action potential doesn't happen (b). Image adapted from Chris et al. 2007.

cognitive function. Moreover, the method is invasive and thus this kind of electrodes can be placed only on subjects suffering from epilepsy.
The corresponding non-invasive method suitable for research purpose, in particular when infants are involved, is the EEG. The method consists in placing electrodes on the scalp of the subject whose brain activity needs to be recorded. The number of electrodes can differ from a low-density electrodes nets, when the number of electrodes is smaller than 32 electrodes, to the high-density which are more technological advanced reaching up to 256 electrodes. The electrodes themselves can be grouped in two main categories.

1. **Gel-based Electrodes**
   A conductive gel or paste is used to achieve a better electrical contact between the electrodes and the scalp reducing the impedance. This solution maintains a lower noise providing a better recorded signal. The use of gel requires a large amount of time to correctly prepare the subject and achieve optimal performance, but the setup time can be faster when a saline solution is used instead of the gel.

2. **Wet Electrodes**
   These electrodes use an alternative medium to gel: a saline, saltwater, solution. The advantage introduced by this method is a faster setup while preserving the

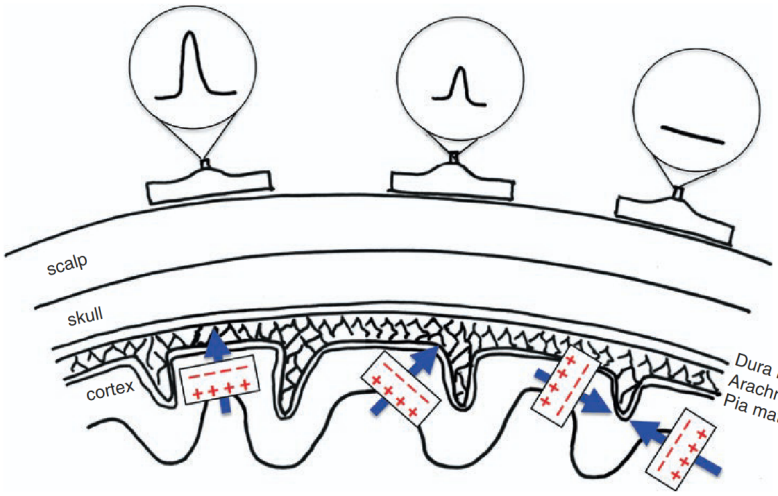


Figure 2.6: The influence of the cortical orientation of the neurons on the recorded signal amplitude. The optimal recording is when the neurons under the electrodes are perpendicular to the scalp and consequently to the electrode. Image taken from Brienza et al. 2019.

performances of the gel-based electrodes, with a low impedance and a good quality signal. The issue of choosing this method is the lower stability over long recording sessions compared to the gel-based method.

3. **Dry Electrodes**
   This solution is more commonly implemented outside clinical and research environments. Their appeal stems from the absence of a conductive medium, making them exceptionally user-friendly and facilitating the adoption of this technology in real-world settings. Since these electrodes are in direct contact with the scalp without an intermediate substance, the impedance is typically higher, which results in poorer signal quality when compared to wet electrode methods.

The recorded signal can be then analyzed using different wave patterns grouped by their frequency bands: delta (0.5-4 Hz), theta (4-8 Hz), alpha (8-12 Hz), beta (12-30 Hz), and gamma (30 Hz and above). Over the years, a different subject state has been associated with each of these.
As with all the imaging techniques, two metrics are used to evaluate a method: temporal resolution and spatial resolution. The former is perfectly represented by the EEG thanks to its high sampling frequency offering a ms resolution. This makes it particularly useful for studying dynamic processes such as sensory perception, motor responses, and cognitive tasks. The resolution duo is inversely related, meaning that the spatial resolution is low compared to its high counterpart. Indeed, when compared to other technique, such as fMRI, the signal recorded are the result of the volume conduction of electrical activity,

making it challenging to pinpoint the exact location of brain activity.
It is important to remark that the signal captured by an EEG electrode, no matter how small, is the result of the thousands of neurons which are under it.

2.4 Evoked Response Potential

In cognitive neuroscience, it is important to be able to record an objective response to a specific stimulus from the subject. In the context of the EEG, the brain response to specific stimuli or events is called ERP. The spontaneous activity of millions of neurons communicating unpredictably under a single electrode gives rise to a signal associable with a casual process. In the middle of all these voices, the ERPs are difficult to be directly seen after the stimulus presentation because of its low amplitude, especially if the presented stimulus is not visual. To increase the chances of detection of the ERP, the best extraction method of the stimulus answer consists in the averaging technique. The key idea of behind this strategy is that the ERP are consistent along the same stimulus which is not the case for the underlying brain activity. Several sensory stimuli are regularly presented to the subject multiple time during which the EEG activity is recorded. The signal is then preprocessed and segmented in smaller time windows containing the stimuli. Each extracted segment is called epoch. Then each epoch is realigned with the others and the average of all the recorded epochs is computed. This time-locked analysis of the signal increases ERP amplitude to the detriment of the casual activity underneath making the Signal-to-Noise Ratio (SNR) increase and the ERP detection possible. From a biological point of view, the ERPs are produced by the synchronous firing of a huge amount of similarly oriented cortical pyramidal neurons (Sur et al. 2009). Two different kinds of ERPs can be classified in humans:

- **Early waves**
  When the response peak is reached before 100 *ms* after the stimulus presentation, they are mainly dependent on the physical parameters of the stimuli, and they are usually called "sensory" or "exogenous".

- **Later waves**
  When the response peak occurs after this time interval, it is usually caused by a higher information processing. This kind of ERPs is usually used with the terms "cognitive" or "endogenous".

Other features are used to provide a better description such as latency, amplitude and polarity. Their high reproducibility among different subjects leads to the definition of a nomenclature. The capital letter with which the ERP waveform name begins define the

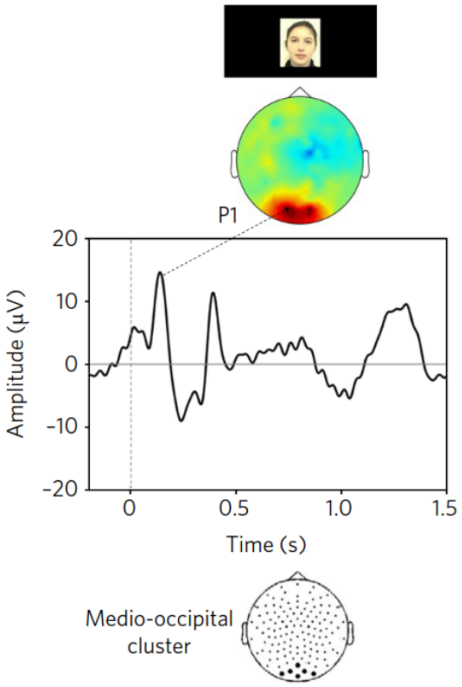


Figure 2.7: The average ERP of a 21-week-old infant in response to a central visual stimulus. The mid-occipital cluster of the electrodes used is presented below the plots. Image taken from Adibpour et al. 2018.

positivity (P) or negativity (N) of the peak. The following number define the latency in *ms* of the onset of the peak for the ERP. An example of visual ERPs can be seen in Figure 2.7. In detail, it is possible to see the stimulus presentation at time 0 *s*, followed by the P100, an early wave of the stimulus response, and the P400, a later wave response. The ERP is computed using the electrodes from the occipital region.

3 Related work

The EEG represents a valuable tool for investigating the high level brain functions with higher temporal resolution compared to other technologies such as fMRI at lower cost compared to MEG, and remaining non-invasive. To the scalp electrophysiological potential recorded with this instrument it is possible to apply different analysis to extract the meaningful information of the underlying cognitive processes hidden by the casual signal originated by all the activity running in parallel in the brain. The common requirement among the existing approaches, like time-frequency analysis or ERP analysis, is the availability of a high number of trials with a low variability. Once several trials are extracted and realigned with the stimulus presentation, their average allows unveiling the cognitive process signature from the intrinsic chaotic nature of the EEG signal. The reproducible shape of the neural activity time-locked to the stimulus is defined as ERP. The downside of this solution is the high sensitivity to the artifacts that can corrupt the signal, especially when their magnitude is bigger than the signal itself.
The preprocessing of the recorded signal becomes a fundamental step to be applied before the extraction of the ERP to avoid any distortion on the final shape that can lead to a misinterpretation of the results. When the experimental population is young, i.e., infants or young babies, the challenge's difficulty increases even more. Indeed, if for the adults a longer attention focus and stillness during the experiment allows longer and less artifacted recordings, when dealing with younger populations, their lack of patience and the impossibility for them to follow the request of avoid movement results in shorter and more artifacted signals. Along with movements and physiological phenomena, the detriment of the signal quality is as well caused by environmental factors, like the power line noise. The difference between the adults and infants is not only marked by the higher presence of artifacts, but also by the different properties of the EEG signal itself which depend on the age of the subject and its sleep-wakefulness stage during the acquisition: the power spectrum and the characteristics of the ERP change during the development with the maturational changes the brain undergoes (Naik et al. 2023). This introduces an uncorrectable intrinsic variability in the signal, affecting the extracted ERP.
The heterogeneous combination of all the sources of noise generates non-stereotypical artifacts, thereby removing some denoising methods, like Independent Component Analysis

(ICA), from the spectrum of effective techniques. Since the number of segments to average is limited, their rejection based in the artifact presence should be cautiously evaluated and avoided as much as possible because it sacrifices relevant aspects of the EEG signal necessary to extract a correct ERP and it reduces the experimental power. The necessity of preprocessing the raw data is in general accepted, but there is no unique definition on the consequential steps that should be applied to the data, leading to a high variability on the implemented actions among different labs. Among these stages sometimes, the user is required to select manually some parameters or even remove some bad electrodes based on subjective decisions increasing even more the already non-negligible preprocessing variability.
Another aspect to not discard is the necessity of longer recording period to be able to investigate more complex cognitive activities. In this scenario, the ability to reconstruct the corrupted signal which is highly redundant by the diffusion of the electric field allows the implementation of different signal denoising techniques (Jiang et al. 2019). This possibility leads to an increase in the retained data without losing data quality.

3.1 Literature Review

With the aim of reducing the problem, different automatic pipelines were proposed to detect and correct the raw recorded signal efficiently from large datasets: NEAR (Kumaravel et al. 2022), MADE (Debnath et al. 2020), HAPPE (Gabard-Durnam et al. 2018). Since APICE-Py is the translation from the MATLAB version and its steps do not differ from the original implementation, its discussion is omitted for an in-depth analysis in section 4.1.

	NEAR	**HAPPE**	**MADE**
Highpass filter (Hz)	from 0.15 to 0.3	1	0.3
Lowpass filter (Hz)	40	X	before 50

Table 3.1: Summary of the filter parameters implemented by different pipelines.

In all the pipelines, filtering of the recorded signal represents the initial step of preprocessing. In general two filter are used in cascade to bandpass filter the signal. While there is a consensus over the necessity to apply a high-pass filter to remove the drifts and slow activity in the data that doesn't have a neural origin but related to other external or biomechanical sources, the selection of the cutoff frequency is not univocally defined by the literature. From the different pipelines analyzed, a broad range of values were

proposed, from 1 *Hz* of HAPPE to 0.15 *Hz* of NEAR (Tab. 3.1). When from the experimental paradigms an ERP is expected, the spectrum of the frequency value to be chosen is reduced: the ERP shape can be dangerously modified if a frequency higher than 0.3 *Hz* is picked (Luck 2014). Another issue with high pass filtering is distortion in the data, especially in the presence of fast artifacts (Cheveigné et al. 2019). In general, the data might be used for different analyses and having a pipeline that can be elastic enough to preprocess the data destined to different analyses is better.
The signal is also contaminated by the environmental noise caused by the electrical frequency at 50 or 60 *Hz*. Since applying a notch filter or a more advanced method to remove this noise are not perfectly able to maintain the brain activity components, the more effective solution is to low-pass the data with a lower frequency. Another reason to apply this criterion is that the Electromyography (EMG) frequency components are more represented in the higher frequencies.
For all the pipeline (except HAPPE which didn't provide any details), the filters applied are non-causal Finite Impulse Response (FIR) filters. In details, for MADE the non-causal FIR filter with latency shift correction integrated in the FIRfilt plugin is used on the continuous data (Widmann et al. 2015) whereas for NEAR the EEGLAB's default filter is used.
Once the data are filtered, they undergo the process of artifact detection in all the pipelines except NEAR, where the segmentation is applied before any further analysis to reduce the computation to only the segments of interest. At this stage each pipeline uses different methods to identify the artifacts.

- **MADE**
  From FASTER EEGLAB plugin, the *channel_properties* is used to extract the metrics used for the artifact detection. The long range dependence of the time series is evaluated via the Hurst exponent, the signal similarity among neighbor channels, which should be similar but not identical because of volume conduction, is investigated with the channel correlation and the variability which should not be too high with the channel variance. For all the metrics, z-score normalization is applied and a threshold of 3 define the presence of artifacts.
  It is possible to remove a priori the outlier rings of electrodes from the recorded signal. This choice can be done based on the assumption that these channels are more susceptible to noise and thus marked as bad. If it is applied, they are removed before the application of *channel_properties* to not influence the standardized values.

- **HAPPE**
  A subset of 19 electrodes based on the 10-20 system along with the channels of interest is extracted from the whole number of recorded data. This is made to avoid

any possible data overlearning of the ICA decomposition that could happen in high-density recordings.
The normed joint probability of the average logarithmic power from 1 to 125 *Hz* is computed over the extracted subset. All the values that are further than three times the Standard Deviation (SD) from the mean are marked as bad. The method is applied twice to achieve a better result.

- **NEAR**
The *clean_flatlines* function from the EEGLAB *clean_rawdata* detects a bad channels if a flat signal is present for more than 5 seconds.
To detect artifacts corrupting temporary the signal, a density-based data-driven approach is implemented. The Local Outlier Factor (LOF) defines the values for each channel which should be considered outliers when compared to its k neighbors. The optimal value for the hyperparameter k is chosen from the application of a Natural Neighbors algorithm to the data and the threshold applied to mark as an outlier one channel is defined from a Training dataset on the basis of a standard scoring of the bad channels.
An optional periodogram analysis can be applied to identify motion related artifacts predominantly corrupting a channel. If a decrease in the Delta and Alpha band power is associated with an increase of the Beta band power, the channel is labeled as bad.

HAPPE is the only pipeline which does not use a lowpass filter to remove the line noise component, relying on the multi-taper regressiion approach implemented in the CleanLine program.
In both MADE and HAPPE, the artifacts correction is made using ICA or a derived method, making the pipelines not suitable for infants' recordings which can be corrupted by not stereotyped artifacts.
The NEAR pipeline avoids the use of the ICA, implementing the Artifacts Subspace Reconstruction (ASR) to get remove the artifacts. It makes use of an empirically optimal value for the parameter of the ASR obtained from a Training dataset. The performances of this solution are dependent on the choice of these parameters which should be defined on the quality of the collected data which on their turn depends on the EEG setup.
The NEAR pipeline is the only among the presented that performs the analysis on the segmented data. It enhances its computational speed since a lower amount of data need to be processed, but it reduces also its generalizability avoiding analysis outside the epochs.
Channel rereferencing to the average is allowed for all the pipelines, whereas for the rereferencing channel-wise is not allowed only by MADE (Tab. 3.2).
The common limitation between all these pipelines is the required MATLAB license which

is an obstacle to their widespread use in the scientific community. To address this limitation, APICE-Py represent the first fully open-source EEG pipeline to preprocess infant EEG data without any licensing cost restrictions based on an existing pipeline. The choice of the programming language lies on its native integration with machine learning and ease of new module or function integration. The full python code is distributed openly, and it is compatible with all major existing EEG libraries, such as MNE-Python or SciPy. APICE-Py implementation was validated against its counterpart in MATLAB.

	NEAR	**HAPPE**	**MADE**
Segmentation	Applied after filter	Optional	Done
Rereferencing average	Yes	Yes	Yes
Rereferencing channel	Yes	Yes	No

Table 3.2: Summary of the main parameters and preprocessing step chosen by different pipelines.

4 Methodology

4.1 Pipeline Overview

APICE-Py relies on MNE-Python (Gramfort et al. 2013) to perform the most of the standard processing steps, like importing the data, filtering and epoching. For more specific tasks, a personalized implementation was done, also modifying some MNE-Python functions to achieve faster computation. The structure of the pipeline is highly modular giving to the user the freedom to edit the order of the different functions or even introduce a personal function as a new step. The processing flow that the data follows is the same as described by APICE (Fig. 4.1).

1. Raw data import
2. Filtering
3. Artifacts detection
4. Definition of bad samples and channels
5. Correction of artifacts
6. Bad epochs' definition
7. Preprocessing report

4.1.1 Raw data import

Since APICE-Py is based on MNE-Python (Gramfort et al. 2013), the file is imported automatically using the correct reader if one of the following file formats is provided: .fif, .mat, .vhdr, .bdf, .cnt, .edf, .set, .egi, .mff, .nxe, .gdf, .data, .raw and .lay. The available formats are quite numerous, allowing the software to read the most common EEG raw data file formats.

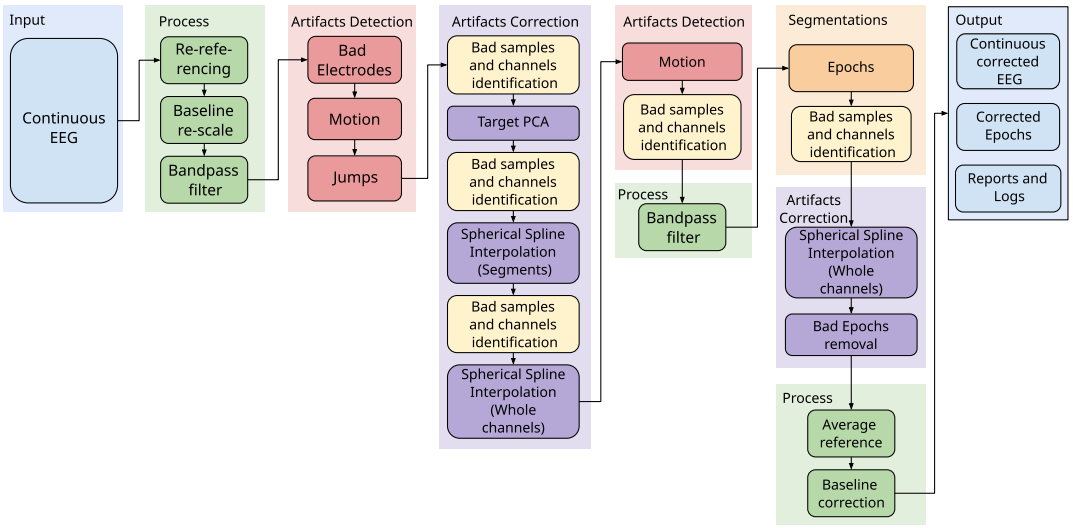


Figure 4.1: Overview of the proposed preprocessing pipeline usually applied on the infants' data. The preprocessing steps are grouped according to the functions implemented in the software pipeline.

4.1.2 Filtering

In signal processing, filtering is at the same time essential and critical because it allows to reduce the undesired noise coming from the non-relevant frequencies but with the potential of introducing signal distortions if not correctly applied.
The primary aim of a filter is to attenuate frequencies primarily associated with noise and outside the band of interest for the analysis. In the case of infants/newborns the frequency range of interest spans between 0.1 to 40 *Hz*, and they are extracted applying the cascade of a low-pass filter followed by a high-pass filter. Selecting the low-pass cut frequency at 40 *Hz* enable to attenuate higher frequency noise coming from the muscle activity which are typically up to far more than 100 *Hz* and moreover efficiently remove the line noise at 50 *Hz* or 60 *Hz*. The choice of a high-pass cutoff frequency is more discussed in the literature. The standard choice for the presented pipeline is set at 0.1 *Hz*: while it removes the slow drifts of the signal without introducing distortions is still leaving the freedom to the user for an harsher definition for different successive analysis on

the preprocessed data. It is worth noticing that a high-pass filtering frequency higher than 0.5 *Hz* introduce substantial distortions in the shape of the ERP (Kappenman et al. 2010). The application of the high-pass filtering should be applied only on the continuous data to avoid edge effects, whereas the low-pass filter can be applied even after the following steps.
Both the low-pass and high-pass filters are FIR for their easier control, their stability and can be corrected to zero-phase without additional computations (Widmann et al. 2015). The default parameters of the MNE-Python library is used.

4.1.3 Artifacts detection

The detection of artifacts corrupting the signal represents a crucial step before performing any kind of correction. First of all the differences in the signal properties between the adults and the infants/children make ineffective the tools already implemented and tested for the former age group. Second the number of artifacts is higher in pediatric recordings: the young subjects lacks of patience moving more their head or body during the experiments causing electrodes displacement. Finally to increase the recordings length, it is common practice to reduce as much as possible the gap time between the EEG sensors placement and the beginning of the recording leading to imperfect electrode contact, poor stability and a variability of the electrode impedance. For these different reason the high presence of artifacts in the EEG data required a robust artifact detection.
In the previous proposed pipelines except APICE, the signal re-referencing is performed only at the last step. This choice is made because the mean which is subtracted from the signal to obtain an overall scalp potential equal to zero can be influenced by the artifacts, leading to a wrong re-reference. At the same time, in unipolar recordings the signal is the potential difference between the electrodes with a selected electrode which is the reference. This measurement system makes the use of fixed threshold biased for artifact detection: since the signal amplitude increases as the distance from the reference increases, a same level of noise can be labeled differently just because of the farther position of the electrode respect to the reference electrodes. Different solutions were proposed: the LOF operates the detection based on the "distance" between each channel (Kumaravel et al. 2022), the normed joint probability of the mean log power in the 1-125 *Hz* frequency band for user-selected channels (Gabard-Durnam et al. 2018) or a combination of Hurst exponent, correlation with other channels, and channel variance (Debnath et al. 2020). Another workaround of this circular dependency between computing the most accurate mean and the re-referencing of the signal is to detect bad period in the recordings applying thresholds on extracted electrodes characteristics and using amplitude independent techniques. After the first detection, the first mean is computed and removed from the

data without considering the detected bad periods. Only after this step is performed, the re-referenced data are used to successive artifacts' detection. The choice of replicating APICE in python, it's mostly due to its scalability through the use of adaptive thresholds which are not defined at a protocol or population level. It brings with it an increase in the processing standardization enhancing data comparability among different studies and across subject.
To increase the data retention, the artifacts' identification is done on the continuous data. Indeed, if a transient artifact corrupt an epoch, its correction is no more possible and its remotion from the available epochs should be done to not corrupt the ERP, whereas in case of the continuous data is possible to correct them retrieving more information. Besides this, the estimation of the thresholds and the EEG characteristics are more accurate when done over a bigger amount of data. Lastly, since more complex paradigms that are more and more implemented nowadays to explore infant cognition require different types of analysis, a common preprocessing pipeline is more flexible.
All the measures used by APICE-Py to detect the artifacts are computed from a sliding time window or from each timestamp. The artifact label is assigned to a timestamp when the extracted value exceed the relative thresholds which are set by default. The limit of acceptance for each value is based on the analysis of the usually normal distribution, and it is conceived on a variation of the simple graphical method based on box-plot proposed by Tukey in 1977. Despite its straightforward appearance, this approach effectively calculates a measure of spread by design, specifically avoiding the use of extreme potential outliers, which could otherwise skew the calculation and diminish the measure's resilience to anomalous data. For each distribution the threshold is computed using $Thresh_+ = Q_3 + k \cdot IQR$ and $Thresh_+ = Q_1 - k \cdot IQR$ where the inter-quartile range (IQR) is computed as $(Q_3 - Q_1)$ with Q_1 and Q_3 being the first and the third quartiles of the distribution. The parameters k can be modified, by default set to 3. Even if the use of adaptive thresholds is in general a better idea, the pipeline gives to the user the freedom to work using fixed absolute threshold, using a single threshold for all the electrodes or individuals thresholds per electrodes. More flexibility is provided with the option to run the artifact detection on average reference data or data z-scored per channel. The algorithms that are applied to the continuous data can be grouped in three groups.

1. **Algorithms to detect non-working electrodes**
   It groups two complementary approaches: one analyzes the activity correlation along the channels marking the ones with a low correlation, the other confronts the channels' power spectrum across frequency bands.

2. **Algorithms to detect motion artifacts**
   This kind of controls relies on the abnormal increase of the running average, the

signal's amplitude and the temporal variance in case of motion artifacts.

3. **Algorithms to detect discontinuities in the signal**
   When the signal presents abrupt changes or jumps, this last detection group is able to label them as artifacts.

The results of each detection step are stored in a boolean matrix of the same shape of the recorded signal, i.e., *epochs* x *channels* x *samples*. The Bad Channel and Time (BCT) matrix marks with True the presence of an artifact with the same resolution of the original signal. Before proceeding, the BCT is refined via some analysis performed on the matrix without looking at the signal: a mask of 50 *ms* is applied to all bad segments, good data that are in the middle of bad data and shorter of 2 *s* are likely a transient period of the same artifacts and thus rejected and bad segments that are too small, default 20 *ms*, are included again.
Even if the group of algorithms used to detect the artifacts proved to be effective in the detection of the majority of the artifacts, more considerations are done based on the BCT matrix itself. Since the key aspect of the pipeline is the correction of the detected artifacts, the first control that should be done to get a good quality interpolation is to have enough good data in a specific timestamp. Indeed, if most of the channels are identified as bad, the interpolation based on small number of electrodes is likely to be poor. A percentage threshold of 30% on the total number of working electrodes is used to define a timestamp as Bad Times (BT). To increase the method specificity the BT label is removed for the segments shorter than 100 *ms*. Due to the possible earlier onset of the artifact compared to the actual timestamp when it is detected, a mask of 500 *ms* before and after any bad time segment and short segments of less than 1 *s*. At the end of these steps a BT boolean matrix is obtained with a shape of *epochs* x *samples*.
Thanks to the high redundancy of the information in the EEG signal, a channel-wise additional control is made on the proportion between good data and identified bad data in the BCT matrix. A default limit of 30% is applied on the computed proportion values to define a whole channel as bad. The boolean matrix Bad Channel (BC) has shape of *epochs* x *channels*.
The same steps to detect artifacts is applied after the artifacts correction and to the segmented epochs to ensure a good data quality. The only difference consists in the marking of bad channel all the channels that presents a bad tags for more than 100 *ms*. It is at this stage that the *epochs* dimension in the boolean matrixes is different from 1. At the end of the process the boolean matrix stores all the detected artifacts (Fig. 4.2).

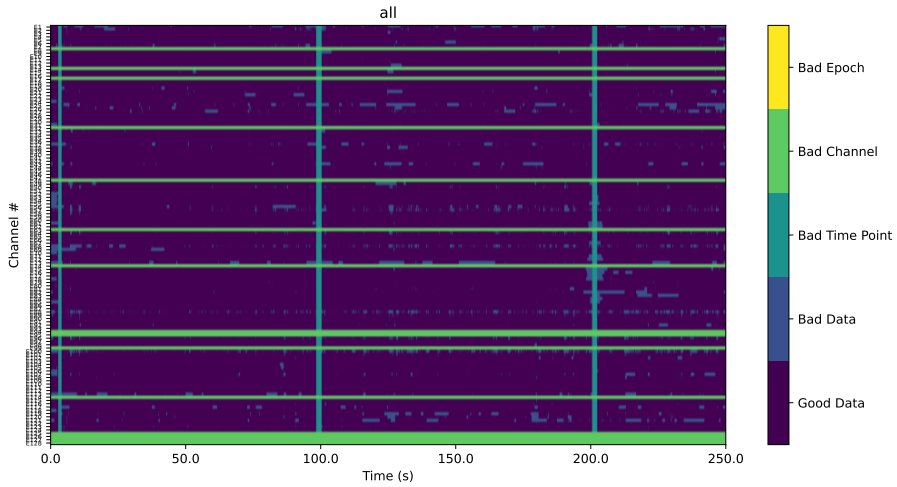


Figure 4.2: Overview of the proposed preprocessing pipeline usually applied on the infants' data.

4.1.4 Correction of artifacts

The detected artifacts corrupting the signal are corrected through data cleaning using artifact estimation coupled with its subtraction or data reconstruction using the interpolation. A sufficient number of good electrodes is crucial for effectively correcting timestamps, particularly for longer artifacts. However, even if all channels are affected, shorter artifacts (less than $100ms$) can still be corrected by removing the first components, even if they aren't explicitly marked by the BT criterion.
For the other types of artifacts, BC and BCT, the correction is performed. For APICE-Py the use of ICA is avoided because the success of the technique in separating neural and non-neural sources is reduced when applied to infants/children data due to the ERP increased variability in amplitude and time and to the subtle similarity between signal and artifacts. This signal characteristics reduce the decomposition quality making difficult the automatic classification of the noise components. The data reconstruction is performed using two strategies:

- **Principal Components Analysis (PCA)**
  This method targets transient artifacts like jumps or discontinuities which are very limited in time. It identifies and removes the initial PCA components that account for 90% of the total variance, effectively eliminating most of the signal's variance. This process is applied simultaneously to all relevant data segments; the PCA is performed on a concatenated dataset of all segments showing "jumps" and "glitches". Assuming these transient artifacts are associated with high variance, the first computed component is removed. While some neural activity might be removed along

with the artifact, this negative effect is considered negligible. This is because the method is applied only to signal segments with a maximum length of 100ms, and only if the first principal component's variance exceeds 90% of the total channel variance. The benefit of reducing artifacts and avoiding the loss of longer data segments outweighs the minimal downside.

- **Spherical Spline Interpolation (SSI)**
  After PCA is applied, the pipeline addresses artifacts affecting longer time segments and small number of electrodes. In this scenario, the Spherical Spline Interpolation (SSI) (Perrin et al. 1989) is made to recover as much data as possible. Usually this technique is used to interpolate electrodes labeled as bad during the whole recording. The innovation of the pipeline is to apply the same idea also to recover the data for channels which stops working for a limited period. A good interpolation of this transient artifacts can be done only if enough good channels are present. The default percentage used by APICE-Py is set to 30% and if it is not reached the timestamp is marked as bad time. Since an interpolation is done, no new information is added, but it helps for a between-subjects analysis and to recover data segments that other approaches end up rejecting.

To summarize, artifact correction is applied considering an increasing order of artifact time length corrupting the signal: from the PCA which deals bad times up to 100 *ms* to the SSI which is applied in a first run to the bad segments and finally to whole bad channels.

4.1.5 Bad epochs definition

At this stage of the preprocessing, the corrected continuous data undergoes the segmentation process. The software asks the user to provide the event label that should be used to extract the epochs. The same artifacts detection is applied to the segmented signal and the SSI is applied to correct the artifacts affecting a channel during an epoch. After the interpolation, the epochs are analyzed on three criteria to detect the epochs which should not be used to compute the ERP:

1. More than 30% of the epoch's channels are labeled as bad
2. More than 50% of the data is interpolated
3. Any BT are contained

All the percentage are the default values implemented in the pipeline, but the user is free to modify them.

4.1.6 Preprocessing report

At the end of the preprocessing different output files are generated. First of all the preprocessed EEG data are saved along with the different segmented epochs and computed ERP with a .fif extension. For a more descriptive view over the preprocessing, the log report contains the output of all the different processing steps, and it is saved in a .txt format, whereas for a more informative summary of the data an HTML file contains the visual representation of the artifact matrix before and after the correction, the Power Spectrum Density (PSD) and the plot of the raw signal, the epochs and the computed ERP. More than that CSV files summarizes the different percentage of detected artifacts in the raw and corrected timestamps in the continuous and segmented data.

4.2 Python Implementation

4.2.1 Translation Process

The APICE MATLAB implementation is based on EEGLAB, a free toolbox developed by Delorme et al. 2004. The counterpart for APICE-Py is MNE-Python (Gramfort et al. 2013) which is used in several step of the pipeline.
The first correspondence to be found between the two software is the filtering. In MATLAB the *pop_eegfiltnew* function based on the FIRfilt plugin is used to filter the raw data using a non-causal FIR filter. In the Python version, the MNE-Python library provides the method *.filter()* which by default applies a zero-phase FIR filter to the data. To increase the performance the computation is parallelized through the use of *n_jobs* parameters which define the number of jobs to run in parallel and can be defined by the user based on its resources. The method is based on the *SciPy* library.
For the artifacts' detection, the functions used by APICE-Py use *NumPy* library to manipulate the data stored in vectors or matrixes.
The artifact correction, in particular the SSI which was implemented in MATLAB was in a first time applied using the MNE-Python method *interpolate_bads*, replaced afterward with a parallelized version of the same function to enhance computational speed. The method automatically applies the spherical splines to interpolate the bads EEG data. For the MATLAB version, the function was coded from scratch following Perrin et al. 1989.

The segmentation and the extraction of the ERPs is performed through the use of the built-in methods of MNE-Python.

4.2.2 Challenges & Workarounds

The interpolation of the bad segments and whole channels was initially done by the MNE-Python method *interpolate_bads*. This solution provided good results when compared with MATLAB, but the performance in time was significantly decreased leading the Python pipeline to be unusable for large datasets. The slowdown is mainly caused by the repeated calling of the method which manipulates the MNE *Raw_object* which is not computationally efficient. A workaround consisted in the manual implementation of the interpolation function with the foresight to enable it to run the interpolation of the different bad segments in parallel using a user-defined number of cores. The achievement is possible by avoiding the object structure and using the extracted information from it, computing the interpolation and storing back the output data to the object for the following steps.
Parallel processing splits a task into smaller, independent chunks that multiple processors can work on at the same time enabling a faster computation. The *joblib.Parallel* is used to separate processes along with its special mechanism of memory mapping to reduce the overhead that a serialization of the data would imply. It automatically generates a temporary file on the filesystem. This removes the need to generate multiple copies of the data since each worker can rely on the reference to the memory-mapped file as input.

4.3 Datasets

To assess APICE-Py's scalability in preprocess EEG data from different experimental conditions, three datasets were used.

4.3.1 Dataset 1: neonates auditory task

The auditory task dataset is composed by the neural activity of 24 healthy full-term infants (11 males) with normal pregnancy and birth. The data were recorded at a sampling frequency of 250 Hz using 128-electrodes net (Electrical Geodesics, Inc) in a soundproof booth during quite rest or while sleeping at the Port Royal Maternity (AP-HP) in Paris, France. The electrical activity of the brain was measured with respect to the vertex

electrode. The experiment consisted in 216 trials for each infant. During a single trial, the participants heard between 4 and 5 syllables lasting 250 ms every 600 ms. The informed consent of the parent was provided.

4.3.2 Dataset 2: 5-months old infants visual task

The visual task dataset consists of 26 (12 males) infants with an average age of 22.98 weeks, SD 1.41, maximum age of 27 weeks and minimum age of 20.86. A 128-electrodes net (Electrical Geodesics, Inc) was used to capture the scalp electrophysiological activity referred to the vertex with a sampling frequency of 500 Hz. The recordings were made in a soundproof shielded booth in which the infant was held by their parent on their lap at NeuroSpin, in Gif-sur-Yvette, France. The experiment had not a fixed duration, lasting until the subject patience allowed. Around 80 up to 160 trials for all participants were collected (mean $\pm$ SD: 126.35 ± 26.07). A successful trial consisted in an attention grabber shown at the center of the screen for 0.6 s, then an image appears lasting 1 s followed by a second image lasting 1.2 s. Another attention grabber for 1.0-1.2 s marked the ending of the trial. The informed consent of the parent was provided.

4.3.3 Dataset 3: 7-years old children and their parents collaborative task

The collaborative task dataset is made by 32 pairs of children-parents (12 male children, 13 male adults). Two synchronized 32 electrodes gel-based Smarting Pro system (mBrainTrain) assured the recording of the scalp electrophysiological activity in both participants simultaneously at a sampling frequency of 500 Hz, reference at the frontal electrodes, ground on central. The experimental environment was a shielded booth located at NeuroSpin, in Gif-sur-Yvette, France. The experimental task consisted in a numerical puzzle, with a paradigm included a naturalistic interaction between the couple of participants with free movement and verbal communication allowed. All the dyads completed three trials each of them did not have a defined duration set a priori, with a mean of 5.6 ± 2.07 minutes (mean $\pm$ SD). The informed consent of the parent was provided.

4.4 Validation and Results

The pipeline itself was already proven to be effective in its MATLAB version (Fló et al. 2022). The validation of its python counterpart is done taking in account different metrics. The first and the more important is to avoid a decrease in the performance between the two implementation. To assess the same performances, the Standardized Measurement Error (SME), proposed by Luck et al. 2021, was used to make a comparison with the results shown in the original paper. Its magnitude is correlated with the variability across the different epochs: the larger it is, the more distant the single epochs' responses are from each other. On the contrary, the smaller it is with increased trials, the stronger is the power of an eventual statistical analysis. To resume, it quantifies the data quality in terms of precision, and it can be applied to any metrics extracted from the ERP waveform. In this cse the metrics used was the mean amplitude in time and space.
The validation implied also the exploration of other metrics such as the retained epochs, the percentage of detected artifacts before and after the correction and, finally, the computational time. The former metrics analysis should be coupled with the results obtained from the SME computation since a smaller number of retained epochs influence the overall SME value. The two latter parameters are used to evaluate the consistency of the results among the three datasets.
To increase the robustness of the analysis and to prove the scalability of the pipeline on the number of electrodes and age of the participants, the three different dataset were used. The first dataset comes from an auditory paradigm with sleeping neonates passive listening to syllables. In this dataset the software is expected to detect a few artifacts since the participants even if infants is sleeping. The second dataset is expected to be more noisy since it involves 5-month-old infants awake. The visual paradigms consist in watching sequence of images running on the screen. The third and last dataset consist of paired data from parents and children involved in solving a cognitive task. The artifacts should be in a range between the two previous situation since the movement is allowed and at the same time the subjects are older. Since no stimuli were given to the subjects, no ERP can be computed leaving this dataset outside the SME computation.

4.4.1 Comparison with MATLAB Outputs

The first comparison between the two programming languages is defined by the SME's results. The computation for both dataset was identical: a same amount of the retained epochs is randomly sampled with replacement for each subject. The average is computed along the selected time window and selected electrodes' subspace dimension. This computation is repeated 1000 times and the SD of the collected measure corresponds to the

SME of each subject ERP. For the Dataset 1, the central electrodes were used to compute the SME over the auditory response (Dehaene-Lambertz et al. 2001) between 250 and 350 *ms* (Fig. 4.3a), whereas for the Dataset 2 the occipital region was used in a period between 550 and 650 *ms* to evaluate the quality of the visual P400 (Nelson et al. 2001) (Fig. 4.3b).

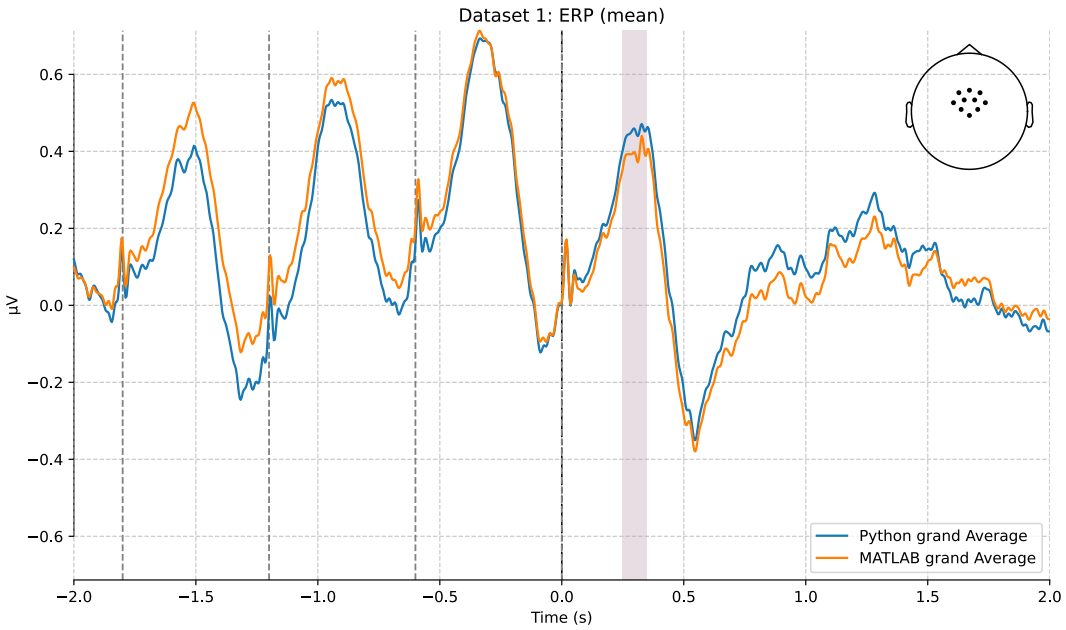
From the Figure 4.3 is possible to see that the two methods return a slight different grand average ERP. This difference was statistically tested using a running one-sample t-test (False Discovery Rate (FDR) corrected by the number of sample) between the subjects' ERP: even if the distance between the two signal appears large for some timestamps, no significant difference ($p > 0.05$) where found. The discrepancy between the results is non-consistent across subjects, but might be driven by a single subject. The Python implementation doesn't introduce any important biases.
Since no statistical differences were detected between the two implementations, the SME was evaluated to have a more robust metrics of the equivalence between the methods (Fig. 4.4). For both dataset, the SME distribution was not normal so the Wilcoxon Signed-Rank test was used to define any statistical difference between the paired values. For both Dataset 1 (Fig. 4.4a) and Dataset 2 (Fig. 4.4b) no statistical difference were found.

A necessary metric to be coupled with the SME is the proportion of retained epochs, computed as the ratio between the number of epochs after the rejection and the total number of epochs. A good pipeline should result in a high number of retained epochs coupled with a small SME's value meaning a high accuracy on the ERP of each subject.

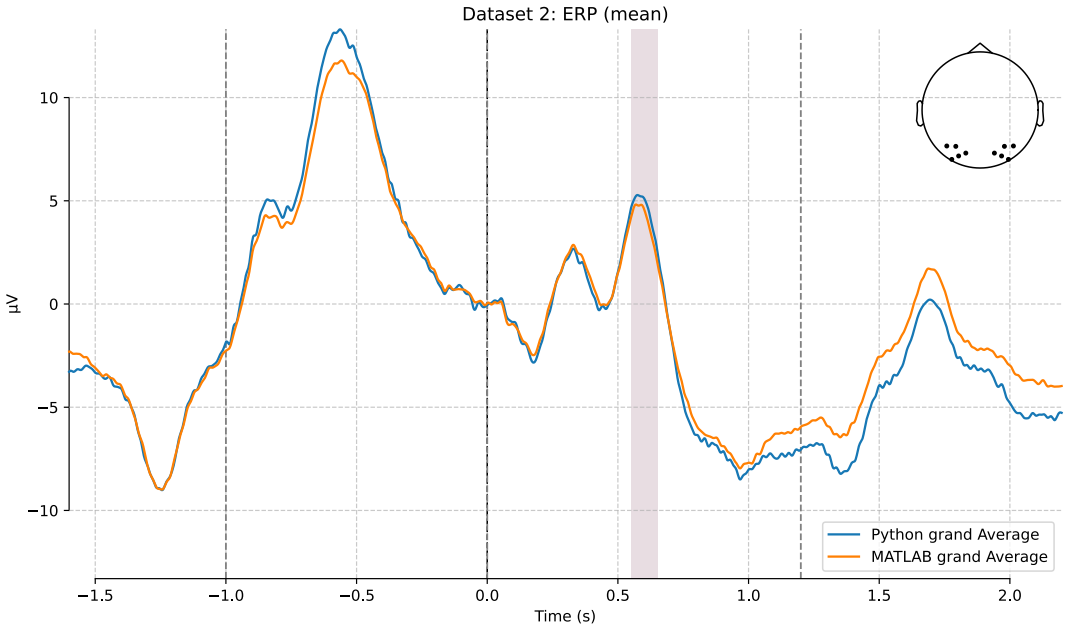
Results show that the python implementation is more conservative compared to its counterpart in MATLAB with a significant smaller number of retained epochs (Fig. 4.5).
The pipelines' performance are also assessed across different datasets by looking at the percentage of detected artifacts both before and after interpolation. Specifically, it is a comparison of the percentage of bad data, bad channels, and bad times between the two implementations.

- **Dataset 1**
  As expected the number of artifacts is not highly corrupted since the participants, even if infants, are asleep (Fig. 4.6).

- **Dataset 2**
  In this other situation the participants (5-months-old) are awake during the recordings, increasing significantly the number of artifacts detected (Fig. 4.7).

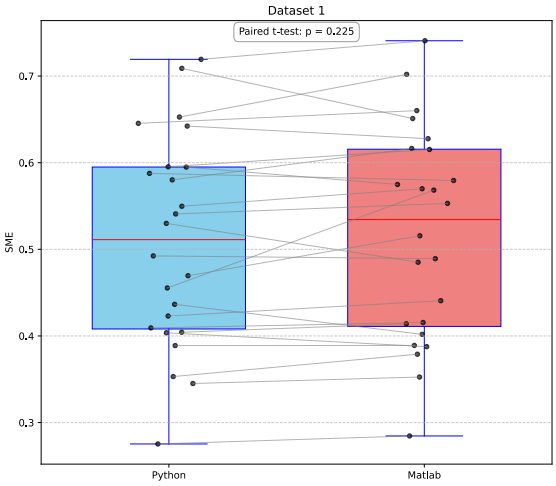


(a) This plot displays the grand average ERP computed on the central electrodes (top right of the figure) for Dataset 1, shown in the top right of the figure. The peaks appearing after the dotted lines (which mark the syllables' onset) represent the auditory ERP following each syllable. The shaded area highlights the time window where the SME was calculated (250-350 ms), corresponding to the peak of the auditory response evoked by the last syllable of the epoch.

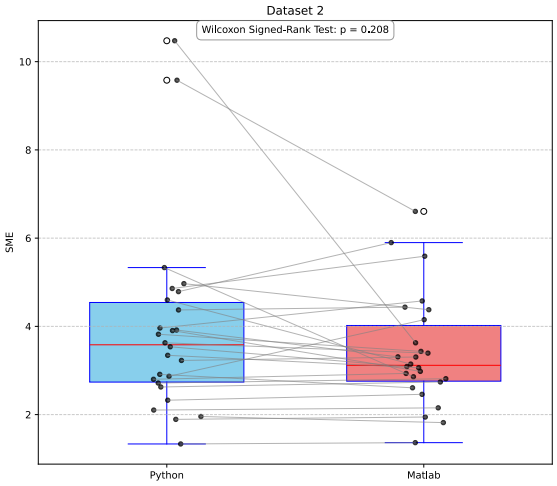


(b) This plot displays the grand average ERP computed on the occipital electrodes (top right of the figure) for Dataset 2. The first two dotted lines indicate the onset of two images, and the third dotted line marks the appearance of the attention grabber. P1 and P400 components are visible after the onset of the images, followed by the visual response to the attention grabber. The shaded area highlights the time window where the SME was computed (550-650 ms), corresponding to the P400 evoked by the second image.

Figure 4.3: Grand average ERPs obtained using the Python and MATLAB with pipeline with the same parameters for artifact rejection.

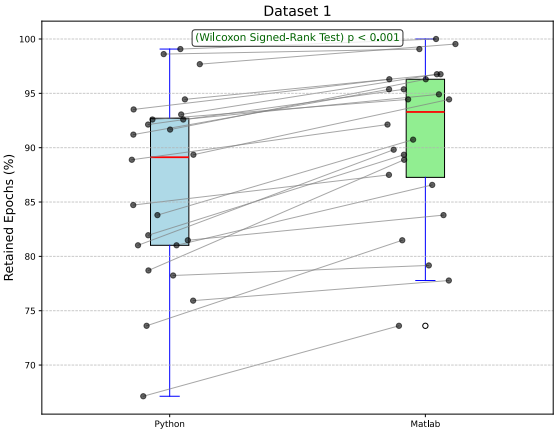


(a) SME for Dataset 1

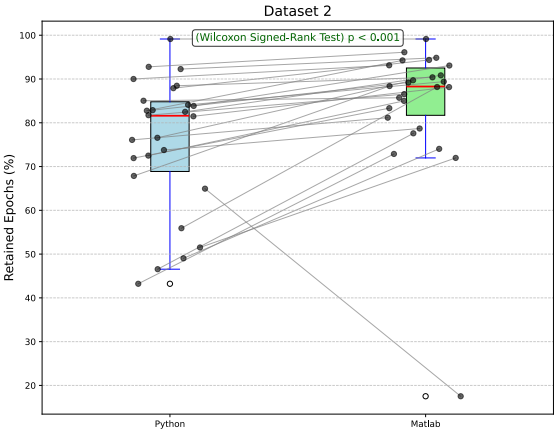


(b) SME for Dataset 2

Figure 4.4: SME with Wilcoxon Signed-Rank test to detect a possible statistical difference.



(a) Percentage of retained epochs for Dataset 1



(b) Percentage of retained epochs for Dataset 2

Figure 4.5: Percentage of retained epochs with Wilcoxon Signed-Rank test to detect a possible statistical difference.

- **Dataset 3**
  This last dataset identifies an intermediate number of artifacts because even if the subjects are older (children with parents), the movement and the talking is allowed during the recording (Fig. 4.8).

In all the three conditions the software is able to interpolate the bad channels and to reduce the overall number of bad data. When the percentage of detected artifacts are compared between the two implementation, a difference is found: the Python version appears to be more conservative detecting an overall higher amount of artifacts. For Dataset 3, the percentages of detected artifacts exhibit a consistent order of magnitude across both

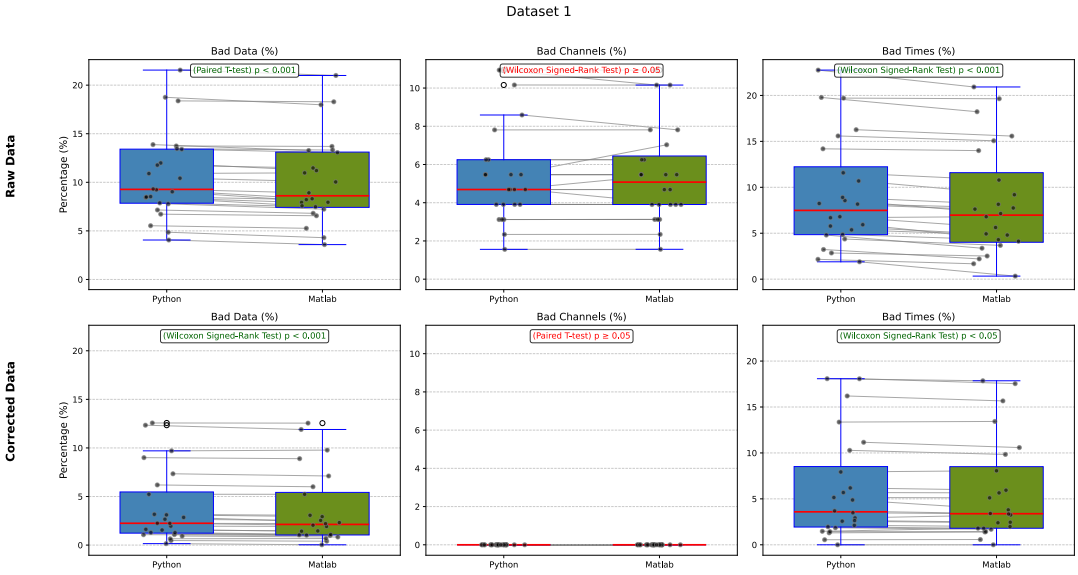


Figure 4.6: Summary of the artifacts detected before (top row) and after (bottom row) the correction methods on the continuous data for Dataset 1.

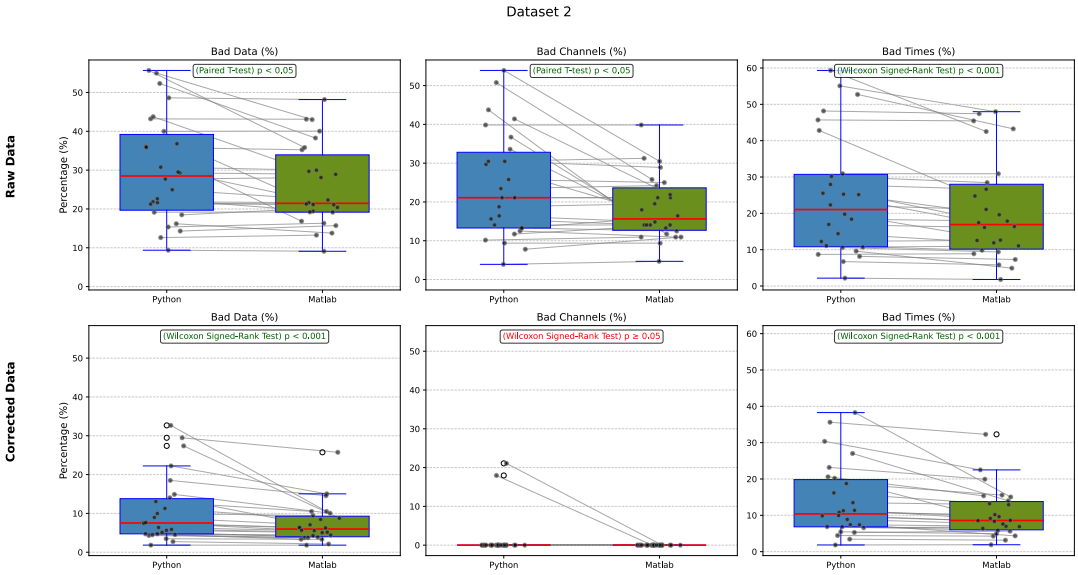
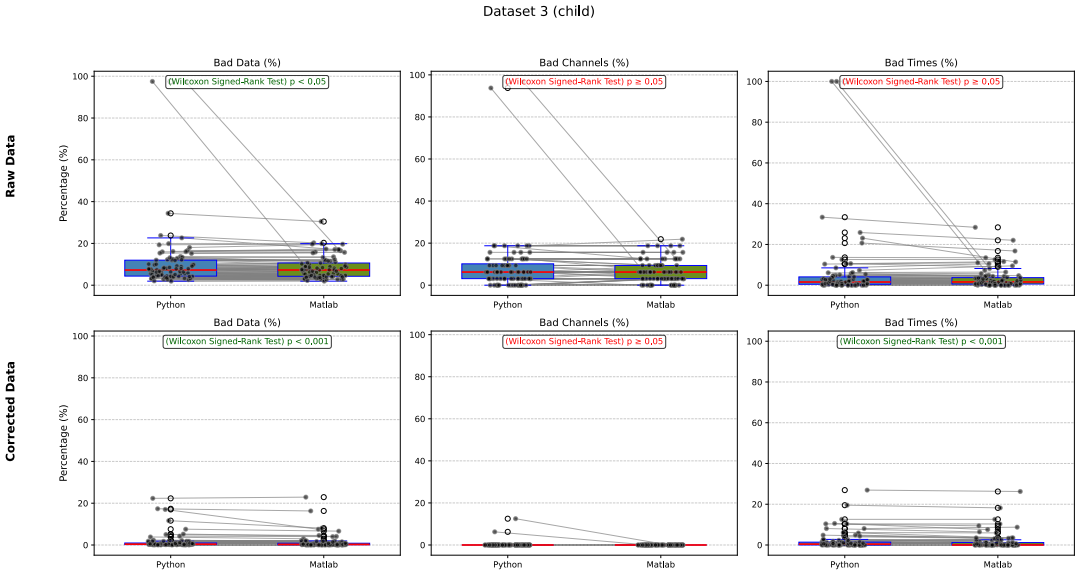


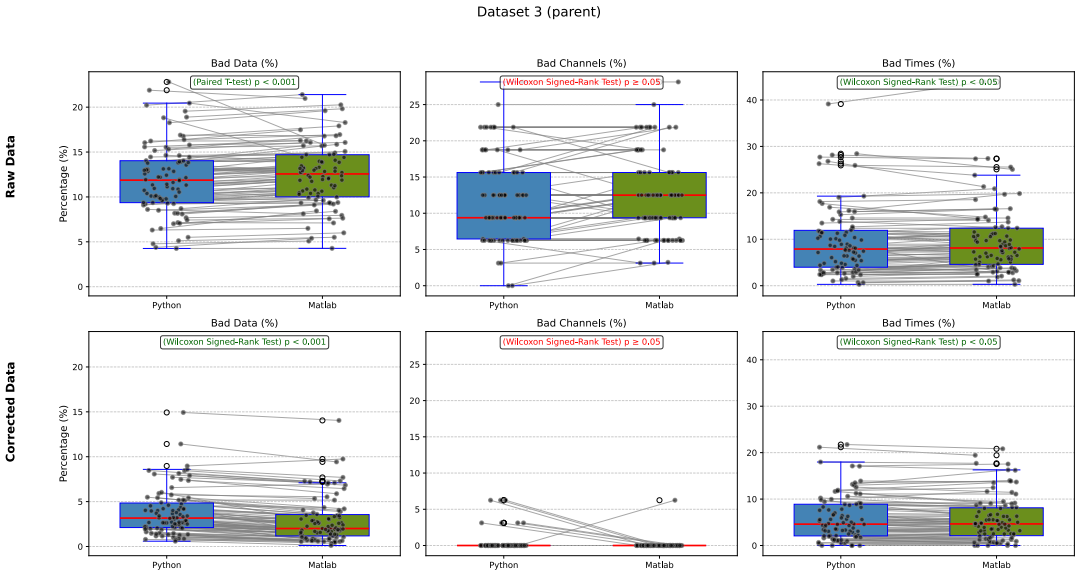
Figure 4.7: Summary of the artifacts detected before (top row) and after (bottom row) the correction methods on the continuous data for Dataset 2.

populations (children and adults), thereby demonstrating the software's age-versatility in population analysis.

The computational time was a side parameter to provide another proofs on the maintained performances of the Python implementation. Before the implementation of the parallelized computation for the SSI, the inefficiency of the Python implementation made it not suitable for the everyday use or for the preprocessing of large datasets. The parallelized pipeline's computational time is evaluated with a variable number of cores (Fig. 4.9).



(a) Summary of the artifacts detected before (top row) and after (bottom row) the correction methods on the continuous data for the children of Dataset 3.



(b) Summary of the artifacts detected before (top row) and after (bottom row) the correction methods on the continuous data for the parents of Dataset 3.

Figure 4.8: Summary of the artifacts detected in the children (a) and parents (b) before and after the correction methods on the continuous data for Dataset 3.

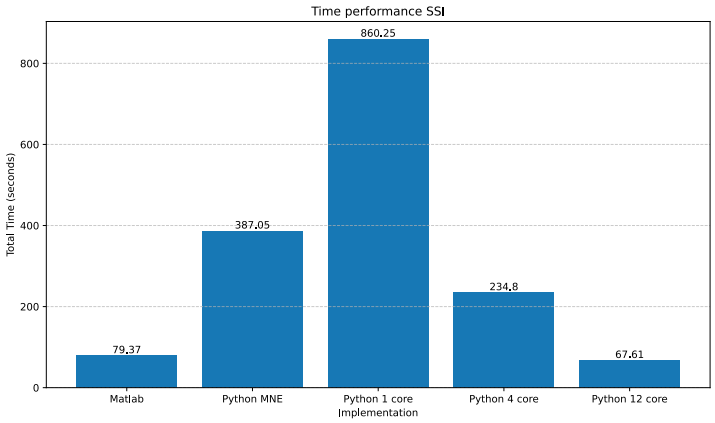


Figure 4.9: Interpolation time requested by the pipeline to perform the SSI. The results are from a single subject taken from the Dataset 2 which is supposed to be the noisier among the three. It's evident that the MNE-Python method is unoptimized for large dataset, whereas the new implementation is reducing the computational time approaching the MATLAB as the cores used in the computation increases.

5 Discussion

To evaluate the correct translation of the pipeline between the two programming languages, the same metrics of the original pipeline was followed (Fló et al. 2022). Multiple random selection with repetition of computed ERP were used to evaluate their mean values distribution. This technique is sensible to detect remains of unwanted noise in the epochs that can affect the averaged results and thus affecting the distribution of the means. Indeed, a single epoch with an artifact, which, when leading with infants' recordings, it is characterized by large value of noise, can introduce a large change in the ERP shape even if the number of epochs is large. This concept proposed for the first time by Luck et al. 2021 is called SME and can be applied to a selected metrics that should be extracted from an ERP to define its quality. The lower is it, the better. The results of this analysis can be better interpreted if the number of retained epochs is considered along: a higher percentage of retained epochs with a low SME value means a more stable results which leads to a higher statistical result. Another measure for the pipeline evaluation is the overall percentage of detected artifacts which was compared in two successive steps: before and after the artifact correction algorithms. The last validation was made over the computational time required by the two software making different trial using a variable number of cores.

Both pipelines were used in their standard implementations using their default parameters. Even if implemented with the same parameters, the Python implementation consistently showed higher rejection levels in the three datasets compared to the MATLAB counterpart. This difference could be linked to the difference in the filter numerical implementation between them. In details, a lighter low frequencies filter can lead to an overall increase in the signal amplitudes making the thresholds to be more sensible to mark a timestamp as an artifact. The threshold can be increased to match better the Python version and the MATLAB version.

The filtering in both pipelines is made using two cascaded FIR filters, specifically a highpass and a lowpass filter. The application of the highpass filter after each interpolation is crucial because interpolating bad data segments with PCA or SSI can introduce signal drifts due to the necessary realignment of different data segments. To counteract these drifts, the data must be highpass filtered following the application of PCA and

segment-based SSI. In the MATLAB implementation it is the *pop_eegfiltnew* EEGLAB function, which uses the FIRfilt plugin, whereas in Python the standard MNE-Python is used. Even if there is a direct correspondence on the frequencies pass edges and the possibility to specify the transition band, they compute the filter length using a different algorithm. If in MNE-Python the value is defined incrementing by one the order of the filter, the *popeegfiltnew* estimate the order automatically based on the frequency provided and sampling rate. This low-level implementation difference is at the origin of the output difference between programming languages. Therefore, the impact of the small differences in filter characteristics could be amplified by the repeated highpass filtering for signal realignment.

The interpolation process is made in parallel in the Python implementation to achieve a comparable speed of the MATLAB counterpart. The number of cores affects the preprocessing speeds, but using more than two is enough to achieve a higher computation speed than the not-parallelized version proposed by MNE-Python. The proposed MNE-Python function is more optimized for a single run (for the whole channel interpolation), but since it is dealing with an MNE Raw object, when introduced in the for loop for the correction of small segments its speed is lowered significantly. An attempt was made to parallelize the interpolation using the MNE-Python function, but pickling is not designed for stable persistence, making the attempt fail (MNE-Python Developers n.d.).

A difference from the MATLAB pipeline is that it doesn't provide the possibility to use ICA or the Denoising Source Separation (DSS) to denoise the data since it is shown to be not effective for infant population. Nevertheless, the clear structure implementation of the pipeline gives the freedom to insert personalized preprocessing function.

APICE-Py allows the user to preprocess the raw signal without performing any kind of segmentation and even if the segmentation is applied, the continuous preprocessed data are saved by default in case different analysis on the signal should be performed.

6 Conclusion and Future Work

This pipeline aims to introduce a standardized methodology for preprocessing EEG data of pediatric population, in both clinical settings and neurodevelopmental studies. Its full open-source nature makes it suitable to enhance an inter-laboratory collaboration. The mutual help between research teams enables to develop a better and more general solution by implementing new functions, suggesting modification or reporting issues via the GitHub repository.

The ability of APICE-Py to be adapted to preprocess data from a wide range of age and for diverse analyses, eased by its use of adaptative thresholds, makes it a suitable choice for preprocessing data from various experimental paradigms. The recovery of the corrupted data increase the data quality and quantity, which is crucial for the ERP analysis. By the punctual identification and subsequent correction, APICE-Py ensure the restoration of the underlying neural activity, leading to more reliable results. These characteristics become even more desirable with challenging experimental populations such as infants, where movement artifacts and short patience critically affect long paradigms.

The presented pipeline is not addressing all kind of artifacts, but just the more common. Future enhancements include identification and correction of physiological artifacts, i.e., eye blinks or eye movements. However, the existing methods that tackle this gap are grouped under the category of Blind Source Separation (BSS) which are not accurate enough to discriminate between neural activity and artifact component. This inaccuracy can potentially lead to the removal of genuine signal and the necessity of implementing such techniques is not suggested by the already achieved pipeline performance. Nevertheless, the modularity of the pipeline allows for future integration if the physiological artifact needs to be managed.

No statistical difference in the SME performance of APICE-Py is detected when compared to the MATLAB implementation, meaning that the shift in programming language does not affect the reliability and precision of the original implementation. The consistent difference on the retained epochs could be attributed to the subtle variation in the numerical implementation of the high-pass filter between languages. This statistical difference suggests a further investigation to achieve a tuning between the two implementations, achieving the same data retention without loosing in data quality.

In the future, the development of a Graphic User Interface (GUI) for APICE-Py could significantly increase its accessibility and application across laboratory. A user-friendly GUI could reduce the barrier of adoption for researcher who are not familiar with command-line interfaces, promoting more standardized preprocessing methods. This, in turn, would enhance the inter-laboratory comparability of results, a critical step to achieve stronger knowledge in neurodevelopental research.
The pipeline is available at in the NeuroKidsLab GitHub repository:

A Appendix A

Spherical Spline Interpolation proposed by Perrin et al. 1989

The interpolation method involves several steps. First, the electrodes are projected onto a unit sphere using 3D coordinates. Then, a matrix is computed to map the good channels to the bad channels. Finally, this matrix is used to interpolate the bad channels.
The method assumes that the potential $V(r_i)$ at all points r_i on the surface of the sphere can be represented by Eq.A.1:

$$V(r_i) = c_0 + \sum_{j=1}^{N} c_i g_m(cos(r_i, r_j)) \tag{A.1}$$

where:

- $C = (c_1, ..., c_N)^T$ are constants to be determined;
- $g_m(\cdot)$ is a function of order m given by

$$g_m(x) = \frac{1}{4\pi} \sum_{n=1}^{\infty} \frac{2n+1}{(n(n+1))^m} P_n(x) \tag{A.2}$$

  with $P_n(x)$ as the Legendre polynomials of order n.

The estimation of the constants C involves solving the following system of two equations simultaneously:

$$\begin{cases} G_{ss}C + T_s c_0 \\ T_s^T C = 0 \end{cases} . \tag{A.3}$$

with

- $G_{ss} \epsilon R^{N \times N}$ is a matrix with entries $G_{ss}[i, j] = g_m(cos(r_i, r_j))$
- $X \epsilon R^{N \times 1}$ corresponds to the potentials $V(r_i)$ of the good channels
- $T_s = (1, 1, ..., 1)^T$ is a 1-column vector of dimension N

The first equation of the system A.3 is the matrix formulation of Eq. A.1, while the second equation effectively applies an average reference to the data. Combining the equations of the system A.3:

$$\begin{bmatrix} c_0 \\ C \end{bmatrix} = \begin{bmatrix} T_s^T & 0 \\ T_s & G_{ss} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ X \end{bmatrix} = C_i X \tag{A.4}$$

C_i corresponds to the matrix with its first column deleted, having a shape of $(N+1) \times N$. The estimation of the potentials $\hat{X} \epsilon R^{M \times 1}$ for the bad channels is done using Eq. A.5.

$$\hat{X} = G_{ds} C + T_d c_0 \tag{A.5}$$

where $G_{ds} \epsilon R^{M \times N}$ maps the bad and good channels using $g_m(r_i, r_j)$. Combining Eq. A.4 and Eq. A.5:

$$\hat{X} = \begin{bmatrix} T_d & G_{ds} \end{bmatrix} \begin{bmatrix} c_0 \\ C \end{bmatrix} = \begin{bmatrix} T_d & G_{ds} \end{bmatrix} C_i X \tag{A.6}$$

Bibliography

Adibpour, P., J. Dubois, and G. Dehaene-Lambertz (2018). "Right but not left hemispheric discrimination of faces in infancy". In: *Nat Hum Behav* 2, pp. 67–79. DOI: 10.1038/s41562-017-0249-4.

Avoli, Massimo (2012). "Herbert H. Jasper and the Basic Mechanisms of the Epilepsies". In: *Jasper's Basic Mechanisms of the Epilepsies*. Ed. by Jeffrey L. Noebels, Massimo Avoli, Michael A. Rogawski, et al. 4th. National Center for Biotechnology Information (US). URL: https://www.ncbi.nlm.nih.gov/books/NBK98150/.

Berger, Hans (1929). "Über das Elektrenkephalogramm des Menchen". In: *Archives für Psychiatrie* 87, pp. 527–70.

Bladin, Peter F. (2006). "W. Grey Walter, pioneer in the electroencephalogram, robotics, cybernetics, artificial intelligence". In: *Journal of Clinical Neuroscience* 13.2, pp. 170–177. ISSN: 0967-5868. DOI: https://doi.org/10.1016/j.jocn.2005.04.010. URL: https://www.sciencedirect.com/science/article/pii/S096758680500398X.

Brienza, Marianna and Oriano Mecarelli (2019). "Neurophysiological Basis of EEG". In: *Clinical Electroencephalography*. Ed. by Oriano Mecarelli. Cham: Springer International Publishing, pp. 9–21. ISBN: 978-3-030-04573-9. DOI: 10.1007/978-3-030-04573-9_2. URL: https://doi.org/10.1007/978-3-030-04573-9_2.

Cheveigné, Alain de and Israel Nelken (2019). "Filters: When, Why, and How (Not) to Use Them". In: *Neuron* 102.2, pp. 280–293. DOI: 10.1016/j.neuron.2019.02.039.

Chris, Diberri, and tiZom (June 14, 2007). *Schematic of an action potential without labels*. Wikimedia Commons. Original by Chris 73, updated by Diberri, converted to SVG by tiZom. URL: https://commons.wikimedia.org/wiki/File:Action_potential.svg.

Debnath, Ranjan et al. (2020). "The Maryland analysis of developmental EEG (MADE) pipeline". In: *Psychophysiology* 57.6, e13580. DOI: https://doi.org/10.1111/psyp.13580. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/psyp.13580. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/psyp.13580.

Dehaene-Lambertz, G. and M. Pena (2001). "Electrophysiological evidence for automatic phonetic processing in neonates". In: *Neuroreport* 12.14, pp. 3155–3158.

Delorme, Arnaud and Scott Makeig (2004). "EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis". In: *Journal of Neuroscience Methods* 134.1, pp. 9–21. ISSN: 0165-0270. DOI: https://doi.org/10.1016/j.jneumeth.2003.10.009. URL: https://www.sciencedirect.com/science/article/pii/S0165027003003479.

Edu, World History (n.d.). *Hermann von Helmholtz - World History Edu — worldhistoryedu.com*. https://worldhistoryedu.com/hermann-von-helmholtz/. [Accessed 12-05-2025].

– (2024). *Emil du Bois-Reymond: The German Physiologist who discovered nerve action potential - World History Edu — worldhistoryedu.com*. https://worldhistoryedu.com/emil-du-bois-reymond-the-german-physiologist-who-discovered-nerve-action-potential/. [Accessed 12-05-2025].

Fló, Ana et al. (2022). "Automated Pipeline for Infants Continuous EEG (APICE): A flexible pipeline for developmental cognitive studies". In: *Developmental Cognitive Neuroscience* 54, p. 101077. ISSN: 1878-9293. DOI: https://doi.org/10.1016/j.dcn.2022.101077. URL: https://www.sciencedirect.com/science/article/pii/S1878929322000214.

Gabard-Durnam, L. J. et al. (2018). "The Harvard Automated Processing Pipeline for Electroencephalography (HAPPE): Standardized Processing Software for Developmental and High-Artifact Data." In: *Frontiers in neuroscience* 12.97. DOI: https://doi.org/10.3389/fnins.2018.00097.

Gramfort, Alexandre et al. (2013). "MEG and EEG Data Analysis with MNE-Python". In: *Frontiers in Neuroscience* 7.267, pp. 1–13. DOI: 10.3389/fnins.2013.00267.

Haas, L F (Jan. 2003). "Hans Berger (1873-1941), Richard Caton (1842-1926), andelectroencephalography". en. In: *J. Neurol. Neurosurg. Psychiatry* 74.1, p. 9.

Hagner, Michael (2012). "The electrical excitability of the brain: toward the emergence of an experiment". In: *Journal of the History of the Neurosciences* 21.3, pp. 237–249. DOI: 10.1080/0964704X.2011.595634.

Jiang, Xiaotian, Gui-Bin Bian, and Zaojian Tian (2019). "Removal of Artifacts from EEG Signals: A Review". In: *Sensors* 19.5, p. 987. DOI: 10.3390/s19050987.

Kappenman, Emily S. and Steven J. Luck (2010). "The effects of electrode impedance on data quality and statistical significance in ERP recordings". In: *Psychophysiology* 47.5, pp. 888–904. DOI: https://doi.org/10.1111/j.1469-8986.2010.01009.x. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1469-8986.2010.01009.x. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1469-8986.2010.01009.x.

Kumaravel, Velu Prabhakar et al. (2022). "NEAR: An artifact removal pipeline for human newborn EEG data". In: *Developmental Cognitive Neuroscience* 54, p. 101068. ISSN:

1878-9293. DOI: https://doi.org/10.1016/j.dcn.2022.101068. URL: https://www.sciencedirect.com/science/article/pii/S1878929322000123.

Larcher, Vic (2015). "Children Are Not Small Adults: Significance of Biological and Cognitive Development in Medical Practice". In: *Handbook of the Philosophy of Medicine*. Ed. by Thomas Schramme and Steven Edwards. Dordrecht: Springer Netherlands, pp. 1–23. ISBN: 978-94-017-8706-2. DOI: 10.1007/978-94-017-8706-2_16-1. URL: https://doi.org/10.1007/978-94-017-8706-2_16-1.

Luck, Steven J. (2014). *An Introduction to the Event-Related Potential Technique*. 2nd. Cambridge, MA: MIT Press, p. 227.

Luck, Steven J et al. (2021). "Standardized measurement error: A universal metric of data quality for averaged event-related potentials". In: *Psychophysiology* 58.6, e13793. DOI: 10.1111/psyp.13793.

McCulloch, Warren S and Walter Pitts (1943). "A logical calculus of the ideas immanent in nervous activity". In: *The bulletin of mathematical biophysics* 5.4, pp. 115–133.

MNE-Python Developers (n.d.). *Why is it dangerous to pickle my MNE-Python objects and data for later use?* URL: https://mne.tools/1.0/overview/faq.html#why-is-it-dangerous-to-pickle-my-mne-python-objects-and-data-for-later-use (visited on 06/17/2025).

Naik, Shruti et al. (2023). "Event-related variability is modulated by task and development". In: *NeuroImage* 276, p. 120208. ISSN: 1053-8119. DOI: https://doi.org/10.1016/j.neuroimage.2023.120208. URL: https://www.sciencedirect.com/science/article/pii/S1053811923003592.

Nelson, Charles A and Christopher S Monk (2001). "9 The Use of Event-Related Potentials". In: *Handbook of developmental cognitive neuroscience*, p. 125.

Perrin, F. et al. (1989). "Spherical splines for scalp potential and current density mapping". In: *Electroencephalography and Clinical Neurophysiology* 72.2, pp. 184–187. ISSN: 0013-4694. DOI: https://doi.org/10.1016/0013-4694(89)90180-6. URL: https://www.sciencedirect.com/science/article/pii/0013469489901806.

Sur, S. and V. K. Sinha (2009). "Event-related potential: An overview". In: *Indian Journal of Psychiatry* 18.1, pp. 70–73. DOI: 10.4103/0972-6748.57865.

Widmann, Andreas, Erich Schröger, and Burkhard Maess (2015). "Digital filter design for electrophysiological data - a practical approach". In: *Journal of Neuroscience Methods* 250. Cutting-edge EEG Methods, pp. 34–46. ISSN: 0165-0270. DOI: https://doi.org/10.1016/j.jneumeth.2014.08.002. URL: https://www.sciencedirect.com/science/article/pii/S0165027014002866.