

Linguistic transcription of EEG responses to sequences of visual stimuli

Arne Van Den Kerchove

Thesis voorgedragen tot het behalen
van de graad van Master of Science
in de ingenieurswetenschappen:
computerwetenschappen

Promotoren:

Prof. dr. ir. H. Blockeel
Prof. dr. ir. M. Van Hulle

Assessoren:

dr. ir. G. Marra
Prof. dr. A. Simeone

Begeleider:

dr. ir. B. Wittevrongel

© Copyright KU Leuven

Without written permission of the thesis supervisors and the author it is forbidden to reproduce or adapt in any form or by any means any part of this publication. Requests for obtaining the right to reproduce or utilize parts of this publication should be addressed to the Departement Computerwetenschappen, Celestijnenlaan 200A bus 2402, B-3001 Heverlee, +32-16-327700 or by email info@cs.kuleuven.be.

A written permission of the thesis supervisors is also required to use the methods, products, schematics and programmes described in this work for industrial or commercial use, and for submitting this publication in scientific contests.

Zonder voorafgaande schriftelijke toestemming van zowel de promotoren als de auteur is overnemen, kopiëren, gebruiken of realiseren van deze uitgave of gedeelten ervan verboden. Voor aanvragen tot of informatie i.v.m. het overnemen en/of gebruik en/of realisatie van gedeelten uit deze publicatie, wend u tot het Departement Computerwetenschappen, Celestijnenlaan 200A bus 2402, B-3001 Heverlee, +32-16-327700 of via e-mail info@cs.kuleuven.be.

Voorafgaande schriftelijke toestemming van de promotoren is eveneens vereist voor het aanwenden van de in deze masterproef beschreven (originele) methoden, producten, schakelingen en programma's voor industrieel of commercieel nut en voor de inzending van deze publicatie ter deelname aan wetenschappelijke prijzen of wedstrijden.

Preface

*“The world makes much less sense than you think.
The coherence comes mostly from the way your mind works.”*

— Daniel Kahneman

I have always been fascinated by the human mind, by its flaws and capabilities. Computer Science can play a major role in understanding the way our brain works and in improving the quality of life for those who have been affected by neurological disorders. Therefore, I hope this thesis, or further work, can contribute new ideas to the field of Computational Neuroscience. It certainly has introduced me to brain-computer interfaces, their applications, and the possibilities of new algorithmic techniques, machine learning, and data analysis, all the while learning about neuroscience.

I would like to thank my promotor at the Departement of Computer Science, prof. Hendrik Blockeel for guiding the progress of my thesis. Furthermore, I would like to thank my promotor and supervisor at the Department of Neurosciences, prof. Marc Van Hulle for his very involved supervision and for the work he has done in helping me apply for a Ph.D. position to possibly continue this work, and Benjamin Wittevrongel for teaching me how to work with EEG data and for his guidance in producing this text. I would also like to thank Valentina Pergher for providing the data from her research, as well as all others at the Computational Neurosciences research group, for the many new insights they offered me. Finally, I would like to thank my friends and family for their constant support throughout ordinary class weeks and extraordinary pandemics, and for catching my typos.

Arne Van Den Kerchove

Contents

Preface	i
Abstract	iv
List of Figures	v
List of Tables	viii
Image Credit	ix
List of Acronyms and Symbols	x
1 Introduction	1
2 Problem Statement	5
2.1 BCI and ERPs	5
2.2 Linguistic transcription	6
2.3 The <i>N</i> -back task and BCI paradigm	7
2.4 Requirements and challenges	9
3 Literature Review	11
3.1 Related work	11
3.2 Single-trial ERP analysis and classification	13
3.3 Spatio-temporal beamforming	14
4 Methods	19
4.1 Recording & preprocessing	19
4.2 Multi-component beamforming	21
4.3 Quantization	27
5 Experiments	33
5.1 Exploratory study	33
5.2 Results	35
5.3 Single-trial classification	36
5.4 Clustering	40
6 Conclusion	45
6.1 Summary	45
6.2 Future work	46
A Multi-component Activation Patterns	51
B Symbol Representatives	57

Bibliography

67

Abstract

This work constructs an algorithm to transcribe event-related potentials (ERPs) recorded by EEG in response to a sequence of single-trial stimuli, to a sequence of symbols. Each symbol represents a distinct category of ERP shapes. The resulting sequence-of-symbols representation is a noise-reduced format for sequences of ERPs and can be utilized by pattern mining algorithms to analyze sequential and contextual structure in sequences of ERPs. This information would otherwise be lost when averaging over multiple EEG trials, which is currently the most popular method for reducing noise in ERPs. The algorithm is developed using available EEG recordings from N -back experiments, a visual paradigm which produces a sequence of ERPs.

In order to cope with the remarkably high signal-to-noise ratio, spatio-temporal beamforming is applied as a linear filter to extract noise-reduced features from single-trial ERPs. This work extends the spatio-temporal beamformer to the spatio-temporal multi-component beamformer to capture more information from each ERP. Two multi-component beamforming approaches are developed. The first method is based on time-window truncated activation patterns, while the second method refines the component activation patterns to isolate specific ERP components. Subsequently, features extracted by the multi-component-beamformer are categorized in groups, which in turn can be represented by a transcribed symbol, using k -medoids clustering. The methods are validated by testing the informativeness of extracted features in several classification tasks and by inspecting the representative ERPs for each symbol, obtained by averaging over all trials in the symbol's cluster.

Results show that the multi-component beamformer has the potential to improve state-of-the-art performance in target/non-target classification for ERP based Brain-Computer Interface paradigms. Visual inspection of symbol representatives shows apparent differences in ERP shape for each cluster, indicating that the algorithm can capture different types of ERPs in different symbols. However, this method is still hindered in its performance and applicability by the large inter-subject variability of ERPs, as well as the inherently high noise levels present in single-trial ERPs.

List of Figures

1.1	An example of an EEG device.	2
1.2	A visualization of an EEG output.	2
1.3	Ideal event-related potential in one channel with annotated components. Components are named after the sign of the amplitude of the deflection (P for positive and N for negative) and the time after the stimulus at which they occur (e.g., N2 or N200 for a negative component appearing around 200ms after the stimulus onset. Conventionally, the potential axis is reversed.	2
2.1	Average of all target trials for 1 subjects in a 1-back experiment, showing multiple ERP components. A trial window starting 200ms before stimulus onset and ending 1000ms after is used. Channels corrupted by noise are indicated in gray.	7
2.2	Graphical rendition of a visual <i>2-back</i> task with one target (T) preceeded by two non-targets (N).	8
3.1	Illustration of separate spatial and temporal activation patterns. Colors at the electrodes represent the values in the spatial activation pattern for the source equivalent dipole of the corresponding color.	14
3.2	Data-driven spatio-temporal activation pattern for the P300 component and related components that are activated by the <i>N</i> -back experiment, constructed as the difference wave of 1-back target and non-target averages for <i>Subject 1</i> . Corrupted channel Cz will be removed from the data before beamforming.	16
4.1	Recording electrode scalp positions. Reference electrodes are indicated in blue.	20
4.2	Visualization of average 1-back target and non-target trials for each subject.	22
4.3	Focus window component activation patterns for <i>Subject 1</i>	24
4.4	Refined component activation patterns for <i>Subject 1</i>	28
4.5	Pearson correlation coefficients of features extracted by the multi-component beamformer from 1-back target trials of <i>Subject 1</i>	30
4.6	Pairwise principal component plots of focus window beamformer scores for all trials of <i>Subject 1</i>	31

5.1	Contrasts of length 1 trial subsequences compared with T/N contrast for 1-back trials averaged over all subjects. Corrupted channels were set to 0 before averaging.	34
5.2	Classifier ROCs for all models per subject for experiments 1), 2) and 3).	37
5.3	Refined component activation patterns for <i>Subject 4</i>	39
5.4	Calinski-Harabasz index for k -medoids clustering of beamformer scores.	42
5.5	Averaged cluster representative trial distance matrix norm divided by k^2 for k -medoids clustering of beamformer scores.	43
A.1	Focus window component activation patterns for <i>Subject 1</i>	52
A.2	Refined component activation patterns for <i>Subject 1</i>	52
A.3	Focus window component activation patterns for <i>Subject 2</i>	53
A.4	Refined component activation patterns for <i>Subject 2</i>	53
A.5	Focus window component activation patterns for <i>Subject 3</i>	54
A.6	Refined component activation patterns for <i>Subject 3</i>	54
A.7	Focus window component activation patterns for <i>Subject 4</i>	55
A.8	Refined component activation patterns for <i>Subject 4</i>	55
B.1	Average of target trials per cluster for <i>Subject 1</i> processed with mcBF1 and clustered using $k = 6$	58
B.2	Average of non-target trials per cluster for <i>Subject 1</i> processed with mcBF1 and clustered using $k = 6$	58
B.3	Average of target trials per cluster for <i>Subject 2</i> processed with mcBF1 and clustered using $k = 6$	59
B.4	Average of non-target trials per cluster for <i>Subject 2</i> processed with mcBF1 and clustered using $k = 6$	59
B.5	Average of target trials per cluster for <i>Subject 3</i> processed with mcBF1 and clustered using $k = 6$	60
B.6	Average of non-target trials per cluster for <i>Subject 3</i> processed with mcBF1 and clustered using $k = 6$	60
B.7	Average of target trials per cluster for <i>Subject 4</i> processed with mcBF1 and clustered using $k = 6$	61
B.8	Average of non-target trials per cluster for <i>Subject 4</i> processed with mcBF1 and clustered using $k = 6$	61
B.9	Average of target trials per cluster for <i>Subject 1</i> processed with mcBF2 and clustered using $k = 6$	62
B.10	Average of non-target trials per cluster for <i>Subject 1</i> processed with mcBF2 and clustered using $k = 6$	62
B.11	Average of target trials per cluster for <i>Subject 2</i> processed with mcBF2 and clustered using $k = 6$	63
B.12	Average of non-target trials per cluster for <i>Subject 2</i> processed with mcBF2 and clustered using $k = 6$	63
B.13	Average of target trials per cluster for <i>Subject 3</i> processed with mcBF2 and clustered using $k = 6$	64

B.14 Average of non-target trials per cluster for <i>Subject 3</i> processed with mcBF2 and clustered using $k = 6$	64
B.15 Average of target trials per cluster for <i>Subject 4</i> processed with mcBF2 and clustered using $k = 6$	65
B.16 Average of non-target trials per cluster for <i>Subject 4</i> processed with mcBF2 and clustered using $k = 6$	65

List of Tables

4.1	Number of available target (T) and non-target (N) trials per subject. Trials corrupted by noise are counted separately (C).	21
4.2	Time windows for focus window component activation patterns.	25
4.3	Time and channel windows for refined component activation patterns. .	27
5.1	Wilcoxon rank sum test p -values for comparison of 300ms-600ms window beamformer output for 1-back trial subsequences.	35
5.2	Wilcoxon rank sum test p -values for comparison of 100ms-300ms window beamformer output for 1-back trial subsequences.	35
5.3	Classification performance (mean area under ROC \pm standard deviation) for 10 repetitions of 5-fold cross validation of classification of 1-back targets and 1-back non-targets.	36
5.4	Classification performance (mean area under ROC \pm standard deviation) for 10 repetitions of 5-fold cross validation of classification of 1-back targets and 2-back targets.	38
5.5	Classification performance (mean area under ROC \pm standard deviation) for 10 repetitions of 5-fold cross validation of classification of 1-back targets and 3-back targets.	38

Image Credit

Figure 1.1 licensed under the Creative Commons Attribution 2.0 Generic license (<https://creativecommons.org/licenses/by/2.0/deed.en>), photographed by Chris Hope.

Figure 1.2 and Figure 1.3 licensed under the Creative Commons Attribution-Share Alike 3.0 Unported license. (<https://creativecommons.org/licenses/by-sa/3.0/deed.en>).

Figure 3.1 adapted from from [56].

List of Acronyms and Symbols

Acronyms

MCBF1	Focus window multi-component beamformer feature extraction model
MCBF2	Refined multi-component beamformer feature extraction model
STBF	Spatio-temporal beamformer feature extraction model
AP	Activation pattern
BCI	Brain-computer interface
CoI	ERP Component of Interest
ECG	Electrocardiogram
EEG	Electroencephalogram
EOG	Electrooculogram
ERP	Event related potential
P300	A positive event related potential component appearing 300ms after stimulus onset
PDA	Push-down automaton
SD	Symbolic dynamics
SNR	Signal-to-noise ratio

Symbols

\bar{a}	Grand average activation pattern, difference wave of 1-back targets and non-targets
ϕ	The sampling rate of the EEG signal
Σ	Spatio-temporal covariance matrix
Σ_{sp}	Spatial covariance matrix
x	

a	Spatio-temporal activation pattern
a_{sp}	Spatial activation pattern
c	The number of components in multi-component beamforming
CH	Calinski-Harabasz Index
$DIST$	Symbol representative diversity metric
k	Number of clusters, number of symbols in the dictionary
m	The number of channels in the EEG signal
n	The number of time samples in an EEG trial
r	The number of trials in an EEG experiment
S	A set of EEG trials
S_T	All target trials in a recording
S'_i	Single-channel temporal trial after spatial beamforming
S^*	Preprocessed trials
S_{1-back}	All 1-back target trials in a recording
S_i	An EEG trial
w	Spatio-temporal beamformer
w_{sp}	Spatial beamformer
X	An EEG signal
Y	Feature matrix after multi-component beamforming
NT	Target trial preceeded by a non-target trial
N	Non-target trial
T	Target trial

Chapter 1

Introduction

Brain-computer interfaces (BCIs) are devices that establish a communication channel between the brain and a computer. The field of BCI research, computational neuroscience, lies on the intersection of neuroscience, computer science and engineering. Specifically, the computational neuroscientist utilizes signal processing, mathematical modeling, and other algorithmic techniques to decode the brain's hidden internal state. Recently, BCIs have sparked interest in the machine learning community, which have led to an influx of new techniques and results. Advances in computational neuroscience research have had an essential influence on the fields of neuroscience and clinical neurology, with applications such as detecting epileptic seizures or communication devices for paralyzed patients who suffer from the so-called locked-in syndrome.

While a plethora of devices exists to capture brain activity and translate it to a signal which can be interpreted by a computer, the most widespread of these measure the electromagnetic activity in the brain. The brain consists of neurons that communicate with each other by passing an electrical current. When a large population of neurons exhibits similar activations, changes in the external electromagnetic field can be detected. These changes are tiny yet measurable, even on the scalp of a subject. A cheap, non-invasive and proven method of capturing this activity is the electroencephalogram (EEG), invented by Hans Berger in 1924. EEG works by attaching electrodes to the scalp of a subject that measure changes in the electrical potentials at different locations on the scalp. The output of an EEG device is a multi-channel time-series consisting of the voltage measured by each electrode.

EEG analysis often focuses on time-locked features in this time-series. When the subject observes external stimuli via its senses, the EEG signal will exhibit detectable behavior in the time and frequency domain, within a specific time frame of the stimulus. These signal deflections are called event-related potentials or ERPs and consist of multiple components (deflections), each of which can vary based on the stimulus (see [Figure 1.3](#)) or other factors, such as the subject's attention. Analysis of these EEG potentials can grant insight into the human brain's inner workings and the processes involved in perception and reasoning. While ERP based BCI paradigms often use state-of-the-art ERP analysis techniques that aim to increase classification

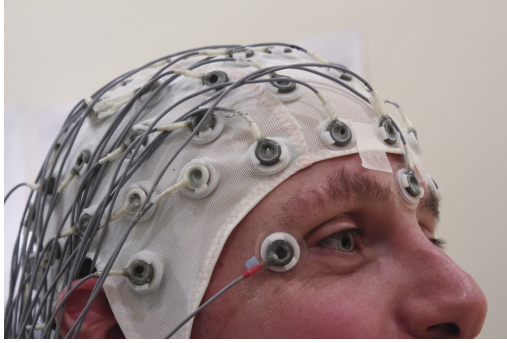


Figure 1.1: An example of an EEG device.



Figure 1.2: A visualization of an EEG output.

performance, there are still opportunities for improvement. An especially compelling case in which earlier research is currently lacking is the analysis of ERPs in the context of their surrounding ERPs, i.e., sequential ERP analysis. Most current techniques process the averages of ERP responses to multiple repetitions of a stimulus in order to decrease the influence of independent noise on the EEG signal. This averaging necessarily means that differences caused by the context in which the singular ERPs occur are lost, or large sections of stimuli must be repeated, thus preserving the context. The latter is rather impractical for BCI applications. A new technique is required to take into account the differences in components caused by the influences of earlier ERPs or stimuli.

This thesis proposes linguistic transcription as a new data format for a sequence

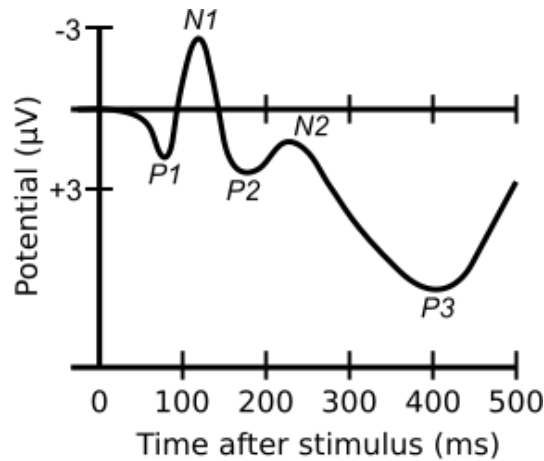


Figure 1.3: Ideal event-related potential in one channel with annotated components. Components are named after the sign of the amplitude of the deflection (P for positive and N for negative) and the time after the stimulus at which they occur (e.g., N2 or N200 for a negative component appearing around 200ms after the stimulus onset. Conventionally, the potential axis is reversed).

of ERPs with reduced noise, thus avoiding ERP averaging and facilitating sequential ERP analysis. In this format, a symbol represents each ERP. Each symbol describes a different category of ERPs that are relatively similar. A symbolic transcription has some clear advantages, which include the following: 1) A structured format for researching sequential structures in the EEG. 2) Techniques in data mining, text mining, pattern recognition, and bio-informatics often work with symbol sequences. A sequence-of-symbols representation for ERPs could open an entirely new window in BCI research to the application of techniques that have proven to be powerful in the past. 3) In general, the sequence could provide a workable representation format for the training of ERP classifiers and predictors. 4) The transcription is in itself an automatic annotation of ERPs or EEG subsequences. Automatic annotation has possible medical applications, such as assisting experts in EEG pattern analysis. 5) A sequential data format has the representation power necessary to capture brain activity caused by context buildup. If a series of stimuli is presented to a subject, the transcription can be utilized to analyze the influence of earlier ERPs and stimuli on current ERPs, without the need for averaging over entire sequences. This principle is especially powerful in text-based BCIs. When a subject receives textual stimuli, e.g., in the form of words on a screen, each word will elicit an ERP. If a sequence of these words carries a semantic meaning and forms a coherent sentence, this sequence of ERPs will, according to some research indications, represent the brain's activity when processing sentences. Currently, however, there is no direct method to capture these ERPs sequentially. The development of such a method is the main motivation for this thesis and could have multiple applications in text and speech processing BCIs.

This last application is of particular interest for neuroscientific research in understanding brain processes and the development of text-based BCIs that aim to predict or analyze words based on their response in brain activity. Analyzing responses to individual words does not provide reliable results, since a significant portion of the semantic meaning of a word can be attributed to the context in which it appears. When a sample word is not extremely concrete, its meaning only resolves once it is placed in a sentence (i.e., a sequence of other words). Currently, BCIs can only take into account one word, for instance, the end word of a sentence. The linguistic transcription approach developed in this work could clear the road for ERP analysis in sentences.

This work first defines the necessary EEG concepts and research goals, followed by a specification of the dataset and the requirements and challenges for developing the linguistic transcription algorithm. Next, an overview of related work and background knowledge required for the development of the algorithm is provided. An exploratory analysis studies whether a sequential structure is present in the data, which could later be leveraged to verify the algorithm output. Finally, feature extraction and validation methods are developed for the linguistic transcription algorithm and their effectiveness verified by experiments in a supervised and unsupervised setting.

Chapter 2

Problem Statement

Problem statement *Develop an automatic annotation of ERP responses elicited when presenting rapid sequences of visual stimuli, leveraging recent developments in spatio-temporal EEG filters to tackle the low signal-to-noise ratio.*

This thesis aims to develop an algorithm to aid in uncovering sequential structures in EEG signals. The algorithm will transform an EEG signal into a sequence of symbols. These symbols can form the basic elements of a language, hence the title *linguistic transcription* (section 2.2). After transcription, the symbols can be grouped to form words or n-grams, and their distributions studied. The linguistic representation allows other algorithms to analyze and exploit possible sequential information in the EEG traces.

In practice, the algorithm will be implemented for EEGs recorded while presenting test subjects with visual stimuli. These stimuli elicit ERPs (see section 2.1) when shown. ERPs are deflections in EEG activity that appear within a short, fixed period after a sensory event (the stimulus). For data availability reasons, the algorithm will be applied to the EEG responses acquired during earlier EEG experiments, described in section 2.3.

This thesis focuses mainly on the algorithmic, mathematical and machine learning techniques used, and not on the neuroscientific results obtained. Furthermore, the focus lies mainly on the extraction and transcription of relevant features from ERPs, rather than on the analysis and interpretation of the obtained transcription.

2.1 BCI and ERPs

Event-related potentials are positive and negative deflections of the EEG signal which are time-locked to a specific stimulus and usually appear within 1 second after that stimulus. Extensive research has shown that they can carry semantical and contextual information about the presented sensory stimuli[4, 12, 24, 25, 32]. Their main benefit lies in the fact that they are time-locked to a stimulus. Hence, it is possible to define EEG trials by setting a fixed time window, starting at the stimulus onset with a length of about 1 second, and recording the EEG within these trials.

Trials with a fixed length constrain the dataset’s dimensionality and can be processed by algorithms without the necessity of searching through the entire EEG signal.

Definition 1 *An EEG trial $S_i \in \mathbb{R}^{mn}$ is a columnwise flattened subsection of the EEG signal with m channels and n samples in the trial window, a trial duration of $n\phi$ with sampling rate ϕ and $i \in 1, \dots, r$ with r the number of trials in the experiment.*

Definition 2 *The trials S_i for $i \in 1, \dots, r$ form the rows of $S \in \mathbb{R}^{mn \times r}$ containing all flattened trials in a recording.*

Definition 3 *An EEG signal $X \in \mathbb{R}^{m \times nr}$ is a concatenation of all (non-flattened) trials for a recording duration $nr\phi$.*

In general, however, ERPs have an unfavorable signal-to-noise ratio (SNR). In order to tackle this problem, trials are often repeated and averaged together to extract the ERP[38]. Because this thesis is interested in individual ERP responses and their context (surrounding ERPs), averaging cannot always be applied. After all, when repeating a stimulus, its context may have changed. Averaging sequences of multiple responses to retain context would increase the length of experimental recordings, which is impractical. Instead, this work will tackle the SNR problem by applying spatio-temporal filtering using beamforming (see [section 3.3](#)).

An ERP consists of multiple components with a fixed naming scheme. For instance, when the ERP shows a positive deflection 300ms after the stimulus onset, this is denoted as the P300 component (P=positive). Research has shown that different components carry information about different brain processes involved in processing the stimulus. Of specific interest for the linguistic transcription is the P300 component, which is related to the working memory[5] and which is the most prominently activated component in the experiments that provide the data for the development of this algorithm.

2.2 Linguistic transcription

The goal of the linguistic transcription algorithm is to map a sequence of ERPs to a sequence of symbols as defined by Definition 4.

Definition 4 *The linguistic transcription algorithm learns a transcription function $\mathcal{T} : \mathbb{R}^{mn} \rightarrow \mathcal{D} : S_i \mapsto \mathcal{T}(S_i)$ which maps an EEG trial S_i to a symbol $\gamma_j \in \mathcal{D}$ with \mathcal{D} a dictionary (set of symbols) of variable size, such that trials that map to the same symbol will contain an ERP response with similar morphology.*

When performing the BCI experiment described below, the brain consistently produces ERPs. The core idea of the algorithm is to extract numerical features from the components of each ERP and then assign a symbol based on those features.

Linguistic transcription allows for easy sequential analysis, both manually and automatically. Experts and algorithms can detect structures in the transcribed

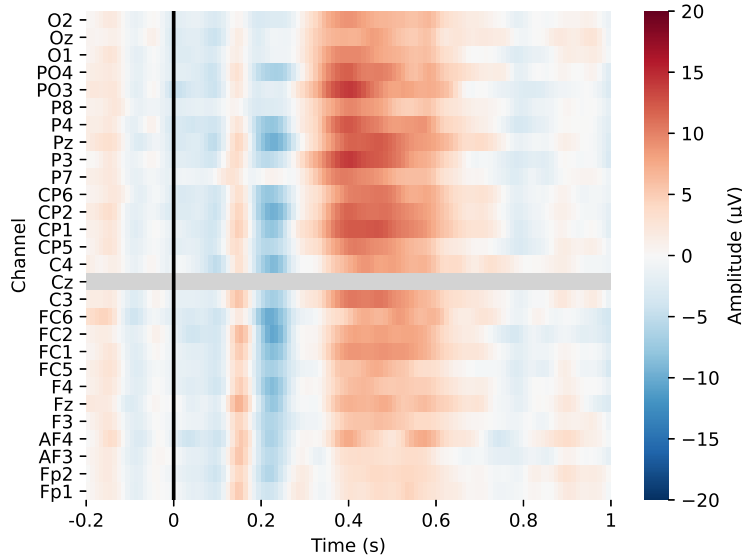


Figure 2.1: Average of all target trials for 1 subjects in a 1-back experiment, showing multiple ERP components. A trial window starting 200ms before stimulus onset and ending 1000ms after is used. Channels corrupted by noise are indicated in gray.

words by looking for sets of co-occurring symbols or relating symbols to the stimuli presented.

A related use case for the algorithm is to transcribe responses to linguistic stimuli for textual BCIs. Instead of analyzing ERP responses to single words in a sentence[12], entire sentences can now be analyzed. The transcription could then directly reflect semantic and lexical concepts of the brain functions, such as context build-up. When all n -grams of symbols are aggregated in a dictionary, statistical analysis can confirm hypotheses about the distribution of words and patterns.

2.3 The N -back task and BCI paradigm

In order to develop the transcription algorithm, EEG data gathered in another BCI study was used. This research by Pergher et al.[47, 48] focused on the so-called N -back training task and its transfer effects to other cognitive tasks.

The N -Back task was introduced in multiple variants in 1958[31] and 1959[39], initially as a working memory training task, and later introduced in neuroscience and EEG studies[20]. In the N -Back task, a screen shows visual stimuli to a test subject for a short time at fixed intervals. The subject is asked to press a button if a stimulus matches another stimulus, shown N intervals back with N a fixed number for the task. If the currently shown stimulus matches the N -back stimulus, it is denoted as a *target* stimulus. An example is shown in Figure 2.2. Common N -Back

2. PROBLEM STATEMENT

levels are *1-back*, *2-back* and *3-back*. *0-back* is sometimes used as a baseline or for validation.

The *N*-Back BCI paradigm is suitable for the initial development of the linguistic transcription hypothesis because of two reasons. Firstly, when executing the *N*-back task, the brain consistently produces ERPs at a fixed time after stimulus onset, resulting in a dataset of non-overlapping EEG trials containing ERPs. Secondly, a concrete hypothesis can be attached to the transcribed results to study the algorithm’s capability of extracting sequential information from a series of ERPs, as described in [section 5.1](#).

The specific *N*-Back tasks used in this research are visual *1-back*, *2-back* and *3-back* tasks as defined by Pergher et al. (2018)[47]. On a screen, stimulus pictures are shown for a duration of 1000ms, followed by a 2000ms inter-stimulus interval. In this interval, the subject is asked to focus on a fixation cross in the center of the screen. The shown pictures consist of easily recognizable objects (e.g., a dog, an apple, ...). Each recording consists of 180 *1-back* level trials followed by 180 *2-back* trials and 180 *3-back* trials.

Each time a subject observes a target or non-target stimulus, their brain evokes an ERP response. Studies regarding the *N*-Back paradigm often focus on the P300 component of the ERP. Several studies have shown that task complexity, information transmitted to the subject, and training level influence the amplitude[27,28,32–34,45] and latency[18,50] of the P300 potential in *N*-Back and similar tasks, like the Sternberg task. The general morphology of the P300 ERP component has been studied extensively[58]. The P300 component in an ERP response to an *N*-back trial stimulus encodes useful information for the algorithm to transcribe. Other

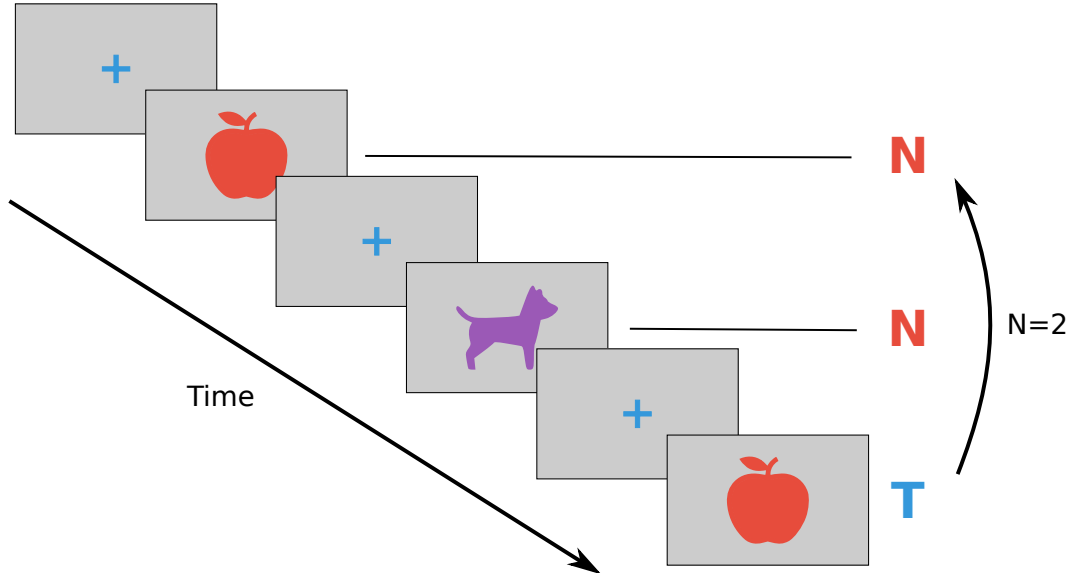


Figure 2.2: Graphical rendition of a visual *2-back* task with one target (T) preceded by two non-targets (N).

components may also be of interest.

To simplify notation, trials containing target and non-target responses will be marked as T and N in the rest of this work, as defined below:

Definition 5 *The sets containing trials of target and non-target trials in a recording will respectively be denoted as S_T and S_N . Similarly, N -back level trial sets are denoted as S_{1-back} , S_{2-back} and S_{3-back} .*

Finally, since interest lies in sequential analysis of ERPs, a sequential notation is defined as follows: target trials directly following another target trial are marked with TT. Similarly, target trials following a non-target trial are marked with NT. The same notation can also mark non-trial targets following a target trial or another non-target trial respectively as TN and NT.

2.4 Requirements and challenges

The algorithm developed in this thesis should meet two basic requirements: 1) The resulting sequence of symbols should capture sequential information present in ERPs, meaning that it should be possible to measure the effects of stimuli and ERPs that appear earlier in the sequence have on any given ERP. It should also allow for comparison between single ERPs that occur in series and make it possible to analyze differences in ERPs arising from their position in the sequence relative to other ERPs or stimuli (i.e., their context). 2) Modelling techniques used in the transcription algorithm should be transparent or white-box models. Transparent models allow for the interpretation of results obtained using data mining or pattern analysis methods on the transcribed sequence. A researcher should be able to directly link the shape of components in the ERP to a transcribed symbol. This requirement restricts the algorithm from leveraging black-box machine learning methods such as neural networks.

A transcription algorithm that fulfills the above requirements should tackle several challenges : 1) EEG signals inherently are subject to very high noise levels. The signal-to-noise ratio in the available EEG data is too low to directly make inferences about ERP features without first filtering out noise. The noise consists of electromagnetic signals that are not of interest and carry no information but are also picked up by the EEG recording hardware and combined with the signal of interest. Noise signals can originate from electrical appliances in the environment, from neurological activity or muscle movement of the subject, which is not of interest. The power of these combined noise signals is considerably higher than the power of the signal of interest. The EEG scalp recording picks up a mixture of noise signals and the signal of interest, making it hard to isolate relevant features from noise. 2) Due to the cost (both time and money) of carrying out medical EEG experiments, data availability is low. No EEG experiments were designed and carried out for this thesis. Instead, recordings for an experiment with another purpose, described in [section 2.3](#), were used to develop and test the transcription algorithm. The data consists of EEG recordings of limited length for a few test subjects. 3) Due to the

2. PROBLEM STATEMENT

nature of the N -back task, the ERP dataset is imbalanced since the number of targets is significantly lower than that of non-targets. 4) Finally, due to the lack of techniques for sequential ERP analysis until now, very little earlier research directly applicable to this topic is available.

Chapter 3

Literature Review

In the past, there have been few attempts at crafting a linguistic transcription of an EEG signal. Most research that transcribes the EEG signal to a sequence of symbols focuses on symbol dynamics or direct annotation of the signal (section 3.1), and is not able to isolate features with signal-to-noise ratios of the same level as ERPs. A set of features for a single-trial ERPs needs to be defined to attribute a symbol to an ERP. Current ERP feature extraction techniques are reviewed (see section 3.2). Finally, as it will become clear that these techniques are still vulnerable to low SNR, the spatio-temporal beamformer is introduced to cope with some of the drawbacks (see section 3.3).

3.1 Related work

A few studies have transcribed the EEG signal to a sequence of symbols for analysis with symbolic dynamics (SD) [29, 46], with the most notable method described in a 2010 paper by Tupaika et al.[53] Symbolic dynamics is a mathematical technique to model non-linear chaotic systems as a space of sequences of abstract symbols[49]. The idea to apply SD to EEG stems from an earlier successful application of symbolic dynamics to construct a classifier for the electro-cardiogram (ECG)[57]. When applying SD, results achieved with ECG data (electromagnetic heartbeat activity) are generally better than those obtained from EEG data because of the lower complexity of the ECG signal. Given an EEG signal X with x_i an individual sample, SD constructs a sequence of symbols by thresholding the signal at different levels, dividing the signal into discrete categories, which can be assigned a symbol.

In contrast to the problem statement of this thesis, the SD algorithm directly translates each data point to a symbol, not taking into account the trial windows or information about present ERPs. The fact remains that this method is susceptible to noise and is not suitable for annotating ERPs, which have a particularly low SNR. Single EEG trial amplitudes can vary significantly due to contamination of other signals. Another issue is the fixed alphabet size required by SD. In the study mentioned above, the alphabet contains only four symbols. If the goal is for the transcription to reflect sequential structures in the ERPs, it is very restrictive. It

might be possible that more symbols are needed to represent the required information. A data-driven solution to determine the number of symbols would be more suited to cope with this problem, since little prior knowledge about the sequential information present is available. In conclusion, directly thresholding the amplitude of the EEG signal on a priori fixed intervals is a relatively simple technique that might be suitable for SD analysis, but would probably not result in an informative sequence for other analysis techniques. The SD research does, however, provide useful insights in analysis methods for a transcribed sequence, like the statistical analysis of the distribution of n -grams in the transcribed sequence.

More recent research has been carried out in the field of automatic EEG clustering and annotation. Multiple frameworks aim to either automatically annotate EEG recordings for relevant clinical brain events, or assist an expert in doing so. In general, they follow a similar approach. First, continuous EEG signals are split into time segments according to some metric which defines homogeneous subsequences in the signal. Next, several features are extracted from each segment. While older algorithms[19, 37, 44] mostly use wavelet transform as a method to extract features, newer frameworks also take into account morphological features of EEG segments in the time and frequency domain[9, 26, 51]. These features have been identified to contain relevant EEG information[1, 7]. Finally, some form of clustering or quantization assigns a categorical descriptor to each segment. Each category can trivially be described by a symbol to obtain a linguistic transcription from such a method.

The one other symbolic EEG transcription result, developed in 2017 by Durand[15], applies a similar method. This work aims to transcribe the EEG signal in order to process it with push-down automata (PDA) to allow for easier waveform detection and classification. While PDAs can efficiently process a sequence of symbols, detection of sequential structure by the use of PDAs would only shift the problem to the construction of an appropriate PDA, which in itself is a hard task. EEG signal segmentation of the signal is carried out by applying a function that splits the time series into homogeneous subsequences, which are then transcribed based on the clustering of features extracted from the segments. Some of the above algorithms also use a similar segmentation approach but do not explicitly construct a symbolic sequence. They focus on the clinical applications of automatic waveform annotation, in particular epileptic seizure detection.

Since the data obtained from the N -back experiments is already segmented into trials, it is only necessary to find a method to extract informative features from the ERPs in the trials. These features can then be mapped to symbols from a dictionary D using classic unsupervised machine learning techniques. A shared problem with the above approaches is the fact that they are designed to detect large scale EEG events and deflections in the signal and will probably not be able to cope with signals with a much lower SNR like ERPs. Without sufficient filtering or feature extraction, ERP signals would be lost amongst the other signals originating from inside the brain and from the environment.

Finally, methods exist to calculate noise-reduced, sparse approximations of the EEG signal. In sparse approximation, each signal trial is represented as a linear

combination of some basic signal elements which are part of a dictionary of signals. These dictionaries can be either fixed Fourier or wavelet dictionaries, or dynamic dictionaries determined by dictionary learning[3, 22, 42]. The latter is a data-driven approach to determine the optimal dictionary of elements representing the different basic shapes of EEG (and ERP) features and is related to blind source separation. After successful dictionary learning for ERPs, it should be possible to quantize the sparse representations of the EEG trials, thus obtaining a categorical and transparent transcription of the ERPs. Furthermore, learning dictionaries per subject could cope with high inter-subject ERP variability. While state-of-the-art multivariate dictionary learning results look promising, they have not been applied to a similar use case, and more experience in EEG and ERP modeling is required to implement correct dictionary models successfully.

3.2 Single-trial ERP analysis and classification

Several single-trial ERP analysis methods have been developed to improve the speed and performance of visual P300 based BCIs, decreasing the need for trial repetition and averaging. Most of these methods focus on the classification of the P300 ERP component in applications like P300 speller BCIs for motor-impaired patients[17]. Single-trial ERP analysis models can be split up into two categories.

On the one hand, more advanced machine learning techniques like multilayer feed-forward neural networks[41], convolutional neural networks[10, 40] and linear support vector machines[11, 35] have been applied directly to measured trials. These methods consider measured amplitudes in a subset or transformation of channels and time samples of the trial as features for classifiers. Since these models are black-box classifiers, the relevant component activation and other ongoing brain activity and external noise are jointly modeled.

In some cases, better performance can be achieved using prior knowledge of the expected ERP, like scalp distribution and timing information. This approach leads to the development of a second class of models that can leverage this information to model signal and noise separately. Some models develop a spatial[6] or spatio-temporal[36, 56] filter, which can be applied to the multivariate EEG signal before the classification layer. Crucial to the development of these models is the correct estimation of the subject-specific multivariate covariance matrix[6], which can be a hard problem due to the presence of excessive noise or lack of data since a sample covariance matrix is highly sensitive to outliers. Therefore, regularization techniques such as shrinkage for covariance matrix estimation may be appropriate. The spatio-temporal beamformer[56, 59] achieves state-of-the-art classification performance.

There is an ongoing debate in the computational neurosciences community whether linear or non-linear models are best suited for ERP classification[43]. In low data availability cases, linear models have proven to be more successful[43]. Linear methods like linear discriminant analysis have also shown to be effective for P300 ERP classification[41].

3.3 Spatio-temporal beamforming

In signal processing, a beamformer is a linear spatial filter that focuses a receiving array of sensors (antennas) on a signal of interest, originating from a specific location while attenuating signals originating from other sources (noise). A spatial beamformer produces a single channel output signal as a linear combination of the multi-channel signal received at the sensor array and a set of weights, one for each sensor[54]. Formally, the beamformer is defined by its weights $w_{sp} \in \mathbb{R}^{m \times 1}$, which can be applied to signal $S_i \in \mathbb{R}^{m \times 1}$ resulting in the measured signal of interest y by calculating

$$y_{tmp} = w_{sp}^T S \quad (3.1)$$

for m sensors. The spatial filter reduces a multi-sensor/channel signal to a single-channel signal so that only the signal of interest is present in the output signal. While this can be entirely implemented in software, the sensor array is, in a sense, forming a ‘beam’ to the source of the signal of interest and only picking up electromagnetic activity from that direction.

In BCI, spatial beamforming can be used to isolate a specific signal of interest, originating from a source equivalent dipole within the brain, from other recorded signals and noise[23]. In order to calculate an appropriate beamformer, an activation pattern is required. The activation pattern (AP) is a template of the signal of interest as measured at the beamformer antennas, or, in the BCI case, the electrodes at the subject’s scalp. **Figure 3.1** shows examples of spatial activation patterns, isolating a signal originating from a source in the brain, and temporal activation patterns isolating a source which exhibits the specified behavior over time.

In ERP analysis, the spatial beamformer can be extended to a spatio-temporal beamformer, which is a recent technique first developed by Van Vliet et al. (2015)[56].

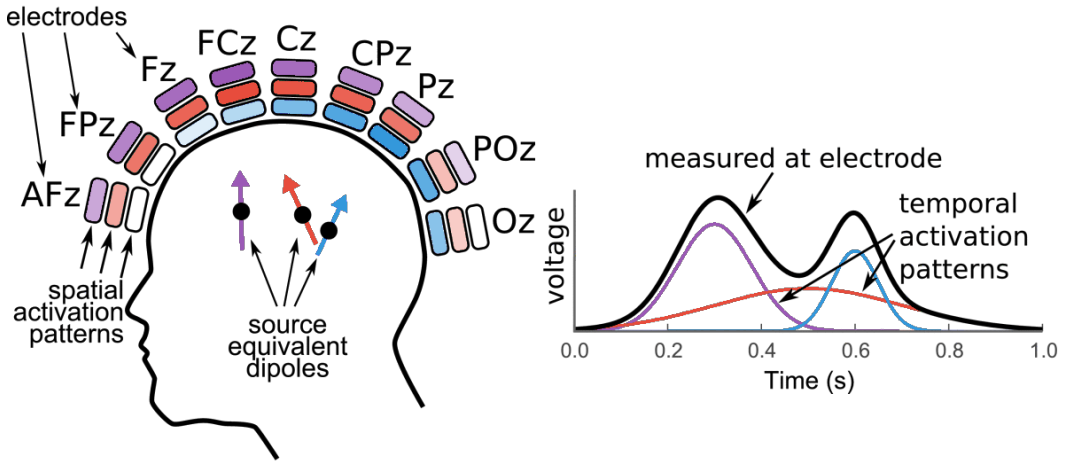


Figure 3.1: Illustration of separate spatial and temporal activation patterns. Colors at the electrodes represent the values in the spatial activation pattern for the source equivalent dipole of the corresponding color.

This beamformer no longer isolates a specific ERP signal from a spatial origin but scores a specific component within the ERP, the component of interest (CoI). The main benefit of spatio-temporal beamforming is that it allows for the extraction of relevant features, like the amplitudes of CoI's, from single ERPs, thus circumventing the need for ERP averaging. The construction of a spatio-temporal beamformer defines the CoI by a spatio-temporal activation pattern, a template matrix containing amplitudes for each channel and time sample within an ERP time window.

While multiple approaches to calculate beamformers exist, Van Vliet et al. make use of the linearly constrained minimum variance (LCMV) beamformer, which has also proven useful in isolating EEG activity in Van Veen et al. (1997) [55]. Van Veen et al. define the original spatial LCMV beamformer by its weights (Equation 3.2) and a linear constraint (Equation 3.3) to avoid trivial solutions.

$$w_{sp} = \arg \min_{w_{sp}} w_{sp}^T S (w_{sp}^T S)^T = \arg \min_{w_{sp}} w_{sp}^T \Sigma_{sp} w_{sp} \quad (3.2)$$

$$a_{sp}^T w_{sp} = 1 \quad (3.3)$$

$\Sigma_{sp} \in \mathbb{R}^m$ is the spatial covariance matrix of the signal S and $a^T \in \mathbb{R}^{m \times 1}$ represents the spatial activation pattern. Using Equation 3.2 and Equation 3.3 and the method of Lagrange multipliers, the weights can be calculated as

$$w_{sp} = \frac{\Sigma_{sp}^{-1} a_{sp}}{a_{sp}^T \Sigma_{sp}^{-1} a_{sp}} \quad (3.4)$$

Similarly, a temporal beamformer can be constructed, that calculates weights for every time point within an ERP time window. This thesis, however, is interested in the spatio-temporal case, where a spatial and temporal beamformer are combined into one spatio-temporal filter. The spatio-temporal filter defines weights $w \in \mathbb{R}^{mn \times 1}$ for every spatial channel and time point within an ERP response window with m the number of channels and n the number of time samples, and can be applied to an EEG trial $S \in \mathbb{R}^{mn \times 1}$. Similar to the spatial beamformer, the resulting output $y \in \mathbb{R}$ can be calculated with Equation 3.5 and Equation 3.6.

$$y = w^T S \quad (3.5)$$

$$w = \frac{\Sigma^{-1} a}{a^T \Sigma^{-1} a} \quad (3.6)$$

with $\Sigma \in \mathbb{R}^{mn \times mn}$ the covariance matrix of S and $a \in \mathbb{R}^{mn \times 1}$ a spatio-temporal activation pattern.

In summary, the spatio-temporal beamformer is a linear filter that, given a template of one or more ERP CoIs, returns a single score for the amplitude of the specified CoIs, suppressing all other activity or noise. The question remains on how to construct these templates. An activation pattern should indicate in which channels a CoI has the highest gain, and at which time points the component reaches

3. LITERATURE REVIEW

a maximum or a minimum value. In principle, an expert neuroscientist with prior knowledge of the brain activity and regions that cause the component, and of the propagation of the electrical brain signals to the EEG sensor electrodes can construct an activation pattern manually. Clearly, this is a complicated and inconvenient task. Furthermore, brain activity responses can vary significantly between test subjects and EEG recording sessions, which poses the need to recalculate the activation pattern for the analysis of each recording.

Another possibility is a data-driven approach to determining the activation pattern of a CoI.[59]. If some EEG trials, of which it is known that the CoI is present, are available, these can be used to determine an approximation of the ideal activation pattern of the CoI for a given subject and recording session. First, the trials containing the CoI and the trials not containing the CoI are separately averaged together. Averaging removes independent background noise, which varies per trial. Next, the activation pattern that captures the CoI is constructed as the difference between the averages of CoI and non-CoI trials. This subtraction is denoted as contrasting and removes components present in both averages but not of interest (e.g., visual baseline activity). Figure 3.2 shows an example of a data-driven spatio-temporal activation pattern for the N -back dataset.

In the N -back experiment, the P300 ERP component is present in all target trials (and, to a lesser extent, in non-target trials). If prior knowledge is available indicating which trials contain responses to targets and non-targets, an activation

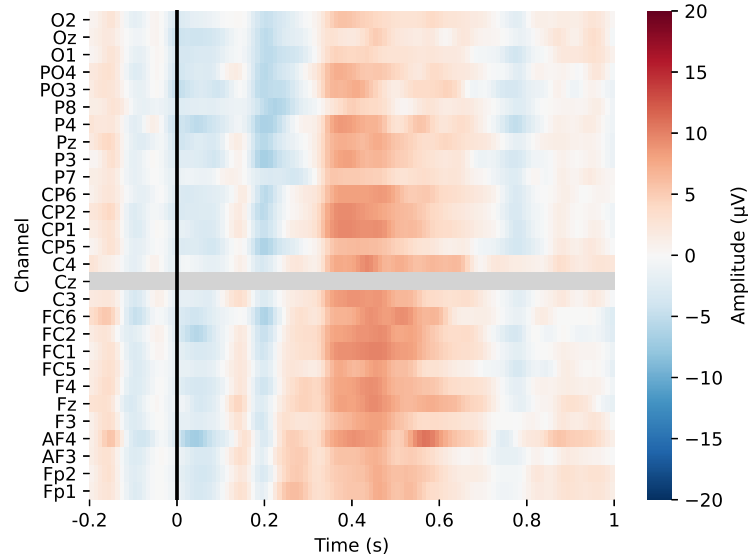


Figure 3.2: Data-driven spatio-temporal activation pattern for the P300 component and related components that are activated by the N -back experiment, constructed as the difference wave of 1-back target and non-target averages for *Subject 1*. Corrupted channel **Cz** will be removed from the data before beamforming.

pattern containing all activated components in a target trial, most notably the P300 component, can be constructed.

Because the spatio-temporal beamformer allows the ERP analysis to focus on specific components in single trials, faster and more efficient ERP classifiers can be built[59, 60]. However, these classifiers are highly dependent on the activation pattern and the amount of data available to estimate the covariance matrix.

Chapter 4

Methods

This chapter first gives an overview of the recording and preprocessing method and the resulting N -back dataset (see [section 4.1](#)). Next, methods for linguistic transcription are developed. The linguistic ERP transcription procedure consists of two steps. First, spatio-temporal beamformers extract features from the individual trials (see [section 4.2](#)). An ideal feature extraction technique that can simultaneously account for the high SNR and inter-recording variability should balance the data-driven approach to determine the activation of ERP components with prior knowledge about the expected timing and spatial distribution of these components. Afterward, each trial is assigned a symbol based on these features and an unsupervised learning algorithm ([section 4.3](#)).

4.1 Recording & preprocessing

This study uses N -back experiment data from 4 recordings of different subjects. For each subject, the EEG was recorded as described in Pergher et al. (2018):

“EEG was recorded continuously from 32 Ag/AgCl electrodes at a sampling rate of 2 kHz using a SynampsRT device (Neuroscan, Australia). The electrodes were placed at O1, Oz, O2, PO3, PO4, P8, P4, Pz, P3, P7, TP9, CP5, CP1, CP2, CP6, TP10, T7, C3, Cz, C4, T8, FC6, FC2, FC1, FC5, F3, Fz, F4, AF3, AF4, Fp1, Fp2. The reference was placed at AFz and the ground at CPz. Additionally, four electrodes were placed around the eyes, on the upper and lower side of the left eye (vertical) and near the external canthus [corner] of each eye (horizontal), for electro-oculogram recording (EOG, bi-polar recording).” [\[47\]](#)

[Figure 4.1](#) shows the location of these electrodes on the scalp. Analysis of recorded data was carried out in MATLAB using the EEGLAB toolbox[\[14\]](#), preprocessing and spatio-temporal beamforming software of the Computational Neuroscience group at KU Leuven, and custom MATLAB code.

In order to analyze ERPs in EEG trials, the continuous EEG signals first need to be preprocessed to remove some types of noise and then cut into non-overlapping

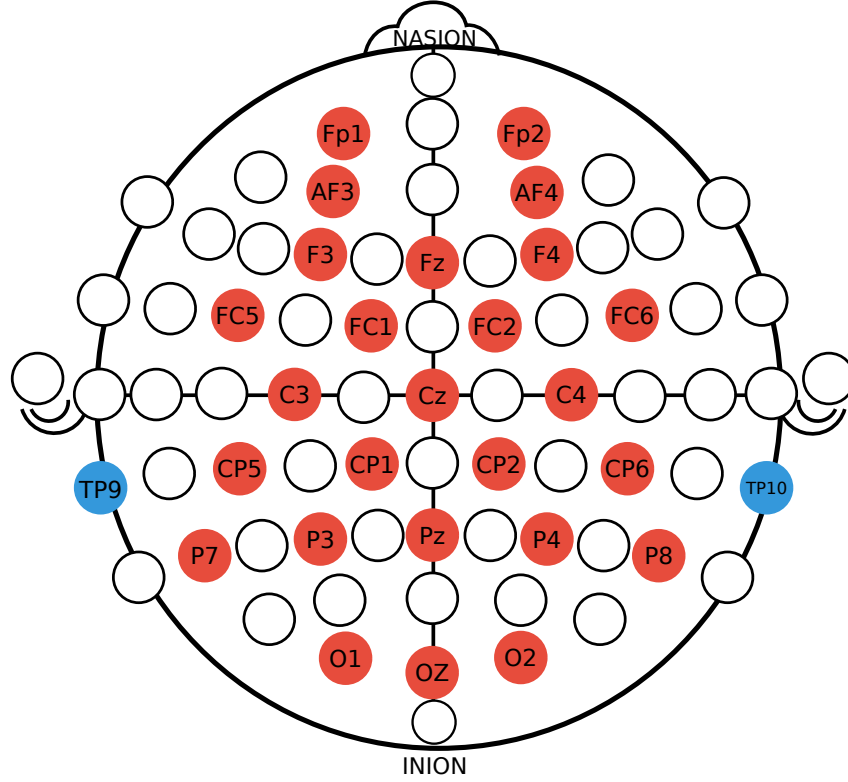


Figure 4.1: Recording electrode scalp positions. Reference electrodes are indicated in blue.

trial windows. First, the signal X is re-referenced to the mean of electrodes TP9 and TP10, since brain signals are weakest at these electrodes due to bony structures in the skull. Next, the signal is band-pass filtered between frequencies 0.5Hz and 15Hz using a fourth-order Butterworth filter. For each subject, visual inspection determines the channels that are corrupted by noise. These channels are dropped from the subject's data. The filtered signal is then cut into trials starting 100ms before each stimulus and ending 1000ms after. The 100ms pre-stimulus window contains no ERP information but is used to center each trial by subtracting the average pre-stimulus baseline activity from the trial. Finally, each trial is downsampled to 100Hz to decrease dimensionality, resulting in processed trials $S_i \in \mathbb{R}^{mn}$ with m the number of remaining channels and n the number of time samples in a trial.

For each subject, trials that contain noise that is several orders of magnitude larger than the signal are identified by visual inspection and marked. When a trial has a noticeably noisy impact on an average in which it is included, it is marked as corrupted. These trials are not yet dropped from the dataset but are excluded when averaging over trials and estimating sample covariance matrices. In applications outside of this experimental setting, where manual trial rejection is not appropriate, the electrooculogram recording can be used for ocular artifact correction with the RAAA method[13] to mitigate the impact of noisy trials. Table 4.1 gives an overview

of the available data after channel and trial rejection.

4.2 Multi-component beamforming

In order to construct a beamformer for feature extraction, a single, data-driven spatio-temporal activation pattern \bar{a} can be constructed for each subject using Equation 4.1.

$$\bar{a} = \frac{\sum S_{1\text{-back},T}}{\#S_{1\text{-back},T}} - \frac{\sum S_{1\text{-back},N}}{\#S_{1\text{-back},N}} \quad (4.1)$$

This activation pattern is denoted as the *grand average activation pattern*, adopting the terminology of Van Vliet et al.[56] It is essential this pattern correctly capture the informative ERP components in the N -back task. N -back ERP responses are strongest in the 1-back task. Furthermore, informative components are mainly activated in target trials and to a much lesser extent in non-target trials. Lastly, the averages used to determine the data-driven activation pattern should not be corrupted by trials with extremely high noise levels. Hence, the data-driven activation pattern is constructed from 1-back trials that are not marked as containing extreme noise, while contrasting trials against non-trials to eliminate brain activity that is not of interest.

Tuning the grand average activation pattern using prior information about the timing at which the CoI is expected to appear can increase the spatio-temporal beamformer’s performance for a specified task[59]. A specialized activation pattern can be derived from \bar{a} by specifying a time window and setting all values outside that window to zero. Applying a window allows the beamformer to focus only on the components present in this time window of the trial while ignoring other components present in the grand average activation pattern. Consequently, the use of windows makes it possible to obtain a specific score for the components in the specified window, instead of a single score for all the components in the trial.

A beamformer with a window for a single component is probably not sufficient to pick up contextual differences in ERP responses, which are smaller than component

Table 4.1: Number of available target (T) and non-target (N) trials per subject. Trials corrupted by noise are counted separately (C).

	1-back			2-back			3-back			noisy channels
	T	N	C	T	N	C	T	N	C	
<i>Subject 1</i>	45	127	8	44	122	14	43	116	21	Cz
<i>Subject 2</i>	37	112	31	44	127	9	37	106	37	O2,P04,P4,Pz, CP1,FC1,FC5, Fp1
<i>Subject 3</i>	43	127	10	45	129	6	48	126	6	Oz,FC2
<i>Subject 4</i>	47	131	2	48	131	1	45	132	3	-

4. METHODS

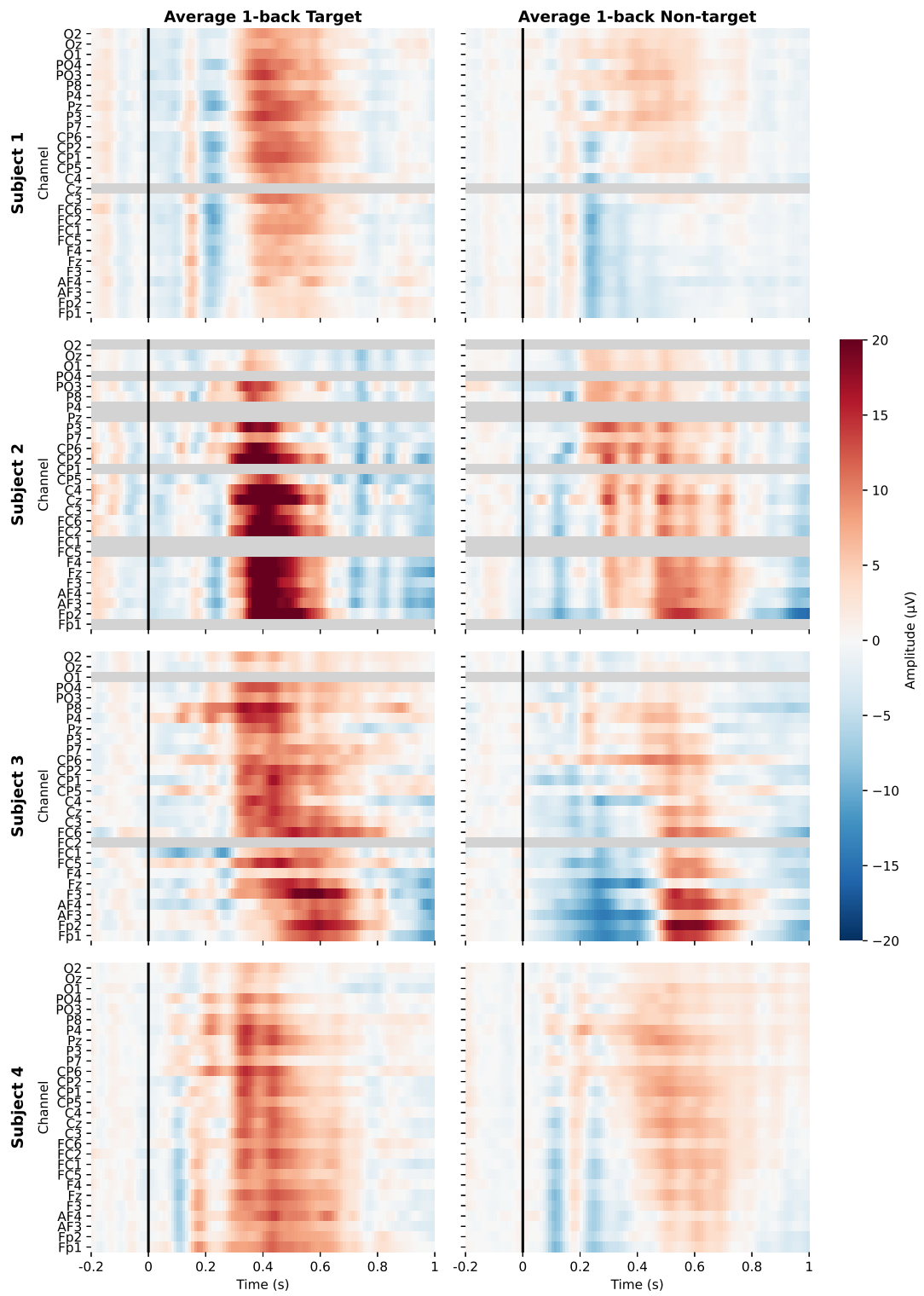


Figure 4.2: Visualization of average 1-back target and non-target trials for each subject.

amplitude or latency differences between targets and non-targets. In turn, a single beamformer with a larger window that covers more ERP components has the disadvantage that it cannot pickup sufficient detail, and can only score a trial as, e.g., ‘more target-like’ or ‘less target-like’ for an activation pattern constructed as the T-N difference wave. Therefore, an ensemble method in which multiple beamformers each focus on a specific window or component could produce better results. The features derived by such a method should contain more information to feed into a transcription algorithm than the the unidimensional output of a single spatio-temporal beamformer.

To counter the drawbacks of using a single spatio-temporal beamformer, multiple beamformers based on different activation patterns are trained for each subject. Each beamformer will extract a different feature from the ERP present when applied to a trial, resulting in a feature vector $y_i \in \mathbb{R}^c$ for trial S_i with c the number of beamformers, or a feature matrix $Y \in \mathbb{R}^{r \times c}$ for all trials. The activation patterns of the beamformer determine what the features represent.

Initially, multiple activation patterns were crafted based on difference waves between different trial/non-trial subsequences. However, this approach was abandoned due to low data availability for some subsequences, and because it presumes that these activation patterns capture relevant information, of which no proof is available. Instead, each activation pattern in the multiple beamformers approach corresponds to one or more specific ERP components, leading to an ensemble denoted as *multi-component beamforming*.

4.2.1 Focus window component activation patterns

By definition, the grand average activation pattern includes ERP components that are present in targets and not, or to a lesser extent, present in non-targets.

By setting all time samples outside a specified time window to 0, the spatio-temporal beamformer will only focus on components occurring within that time window, ignoring earlier and later components. [Figure 4.3](#) shows an example of these component activation patterns. When one of these component activation patterns is correctly constructed to capture only the desired component, the output of the beamformer applied to a given trial for this activation pattern can be interpreted as the score of that component in the trial (cfr. [section 5.1](#)). The latter is a desirable property, since it makes the process transparent, in the sense that an extracted feature can directly be linked to a neuroscientific concept that can be of interest in analysis.

Since the N -back task is a visual attention based task, the relevant components in the ERP responses are P100, N100, P200, N200 and P300 components[\[52\]](#). The negative component occurring between 700ms and 900ms as visible in [Figure 3.2](#) is also included. Ideally, a distinct time window would be assigned to each of these components. However, this is not possible since components can overlap in time and scalp distribution. Therefore, some component activation patterns will capture multiple components. The output of the beamformer then no longer directly correlates to a component amplitude, but rather represents a combined score for the

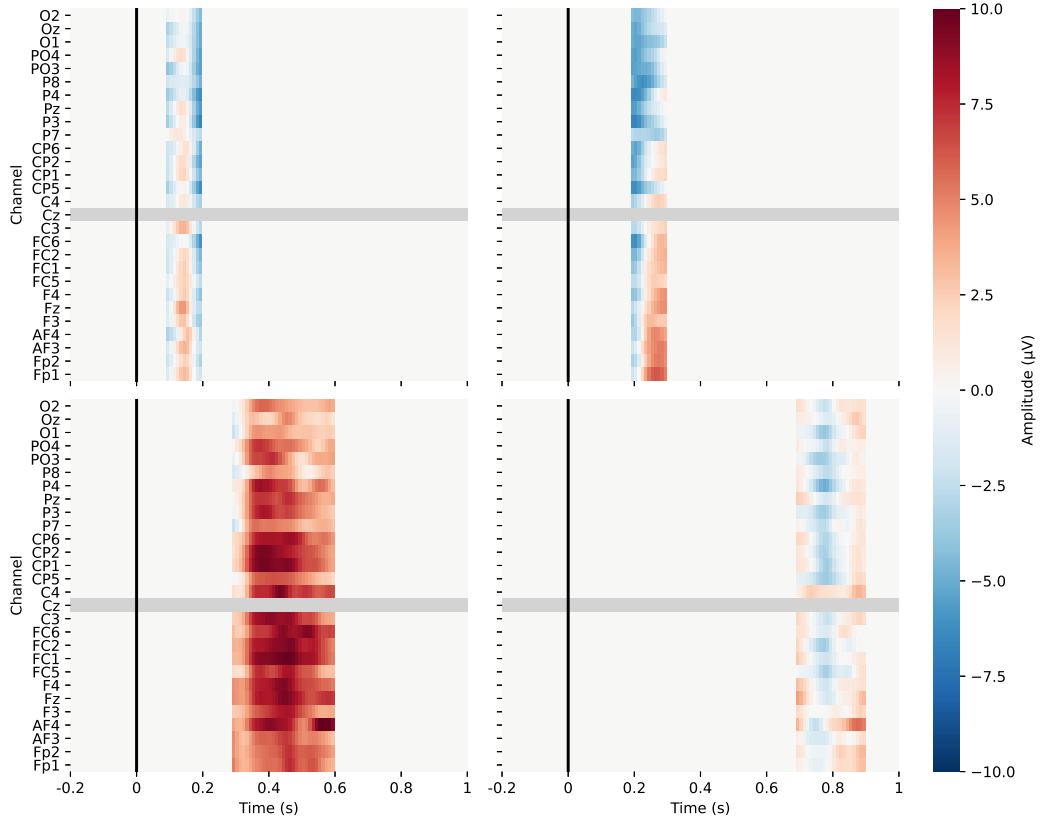


Figure 4.3: Focus window component activation patterns for *Subject 1*.

components present in the window. This combined score results in information loss since components can no longer be evaluated separately. These problems will be addressed in [subsection 4.2.2](#).

[Table 4.2](#) lists the time windows used in the multi-component beamformer, and the components captured by each activation. Once these activation patterns are created, the spatio-temporal beamformer can trivially be extended to a multi-component spatio-temporal beamformer by learning a filter vector for each of the activation patterns and applying these filters separately to a given trial. The spatio-temporal covariance matrix Σ can be estimated once from all available data and reused in the calculations of all w_j . This procedure leads to the procedure in [Algorithm 1](#) which returns feature matrix Y from trials S as a vector y_i of component scores for each trial S_i .

4.2.2 Refined component activation patterns

As mentioned above, the component activation patterns created with focus windows have several drawbacks. Overlapping components that reach their peak at around the same time can obscure each other’s amplitudes. To achieve a pure activation

Table 4.2: Time windows for focus window component activation patterns.

AP	time window	channel window	ERP components
1	100-200ms	all	P100
2	200-300s	all	N100,P200
3	300-600ms	all	N200,P300
4	600-900ms	all	

Algorithm 1 Focus window multi-component beamforming trial feature extraction procedure

```

1: procedure MCBF1FEATURES( $X, \phi_0, \text{windows}$ )
2:    $S, \phi \leftarrow \text{PREPROCESS}(X, \phi_0)$ 
3:    $S^* \leftarrow S \setminus \text{BADPOCHS}(S)$ 
4:    $\bar{a} \leftarrow \frac{\sum S_{1\text{-back},T}^*}{|S_{1\text{-back},T}^*|} - \frac{\sum S_{1\text{-back},N}^*}{\#S_{1\text{-back},N}^*}$ 
5:   for  $j \in 1, \dots, c$  do
6:      $t_{\text{start}}, t_{\text{end}} \leftarrow \text{windows}_j$ 
7:      $a \leftarrow \bar{a}$ 
8:      $a[:, 1 \dots t_{\text{start}}\phi] \leftarrow 0$ 
9:      $a[:, t_{\text{end}}\phi \dots m] \leftarrow 0$ 
10:     $w_j = \text{CALCULATESTBF}(a, S^*)$ 
11:     $y_{i,j} = w_j^\top S_i$  for  $S_i \in S$ 
12:   return  $Y$ 

```

pattern that is only focused on a single component of interest, the activity from other components that are active in the window for the CoI needs to be subtracted from the component activation pattern. Since the data-driven grand average activation pattern \bar{a} is measured as an additive combination of components, it is non-trivial to separate these components from each other without constructing an anatomical forward model for their independent activation as measured at the scalp using a priori knowledge concerning the electromagnetic activity in the source equivalents and the conductivity of the tissue between the generators and the electrodes. This mixture of component activations leads to a cocktail party problem, in which the individual CoI activation patterns have to be derived from \bar{a} .

Therefore, a second approach to component activation patterns is developed in addition to the windowing method in [subsection 4.2.1](#). Since the data-driven approach to determine activation patterns avoids using prior information about the source equivalents in the brain and only leverages information about the timing and spatial distribution as measured on the scalp, no source separation model is developed. Instead, an approximate method is developed to refine component activation patterns determined by a window in such a way that activity from other components than the CoI is suppressed. This approach is based on the procedure to construct a CoI activation pattern proposed by Van Vliet et al. (2015)[56], with some adaptations to

better tailor the resulting activation patterns to other components than P300. The refinement procedure for a CoI activation pattern, as described by Algorithm 2, starts from the grand average activation pattern \bar{a} and a prior estimate of the time window and set of channels in which the CoI reaches its peak. These time and channel windows can uniquely define a component, even when overlapping components are present. The procedure starts by identifying the time point t_{peak} in the time window at which the CoI reaches its maximum or minimum in \bar{a} . At this time, most of the activity in \bar{a} can be assumed to originate from the CoI. Hence, the activation at t_{peak} , $\bar{a}[:, t_{peak}\phi]$ with sampling rate ϕ can be regarded as a spatial activation pattern $a_{sp} \in \mathbb{R}^m$ for the CoI, leading to the construction of a spatial beamformer w_{sp} using Equation 3.1. Before calculating the beamformer weights w_{sp} , all channel values in a_{sp} with opposite sign as the amplitude of the CoI peak are set to 0, in order to exclude other components that are active at the same time. The spatial beamformer for the CoI w_{sp} can be applied to the trials in S^* using Equation 3.1. The spatial beamformer maps each trial S_i^* to a single-channel trial $S_i' \in \mathbb{R}^n$ containing the temporal activation of the CoI during that trial.

Analogous to the construction of the grand average activation pattern, a new, temporal activation pattern $a_{tmp} \in \mathbb{R}^m$ for the CoI can be constructed using the difference wave of 1-back target and non-target spatially filtered trials in S' . Each temporal activation pattern reaches its peak when the respective CoI is most active. However, earlier or later components with a similar spatial distribution may also be present in a_{tmp} . Therefore all values in a_{tmp} outside the zero crossings, minima or maxima surrounding t_{peak} are set to zero. Finally, the refined activation pattern $a \in \mathbb{R}^{mn}$ is the columnwise flattened product $a_{sp}a_{tmp}^T$.

Algorithm 2 Single component activation pattern refinement procedure.

```

1: procedure REFINEACTIVATIONPATTERN( $\bar{a}, S^*, \text{time\_window}, \text{channel\_window}$ )
2:    $t_{start}, t_{end} \leftarrow \text{time\_window}$ 
3:    $t_{peak} \leftarrow \arg \max_t |\sum \bar{a}[\text{channel\_window}, t\phi]|$  with  $t_{start} \leq t \leq t_{end}$ 
4:    $a_{sp} \leftarrow \bar{a}[:, t_{peak}\phi]$ 
5:   for  $i \in 1, \dots, m$  do
6:     if  $\text{sign}(a_{sp}[i]) \neq \text{sign}(\sum \bar{a}[\text{channel\_window}, t_{peak}\phi])$  then
7:        $a_{sp}[i] \leftarrow 0$ 
8:    $w_{sp} \leftarrow \text{CALCULATESPATIALBF}(a_{sp}, S^*)$ 
9:   for  $S_i^* \in S^*$  do
10:     $S_i' = w_{sp}^T S_i^*$ 
11:     $a_{tmp} \leftarrow \frac{\sum S_{1\text{-back},T}'}{\#S_{1\text{-back},T}'} - \frac{\sum S_{1\text{-back},N}'}{\#S_{1\text{-back},N}'}$ 
12:     $t'_{start} \leftarrow \text{index of first minimum of } |a_{tmp}| \text{ before } t_{peak}$ 
13:     $t'_{end} \leftarrow \text{index of first minimum of } |a_{tmp}| \text{ after } t_{peak}$ 
14:     $a_{tmp}[1 \dots t'_{start}\phi] \leftarrow 0$ 
15:     $a_{tmp}[t'_{end}\phi \dots n] \leftarrow 0$ 
16:     $a \leftarrow a_{sp}a_{tmp}^T$ 
17:   return  $a'$ 

```

This ap refinement procedure gives rise to a second ERP feature extraction algorithm, listed by Algorithm 3. Table 4.3 defines time windows and channel regions as input for the refinement procedure. Values for time windows and channel regions were experimentally determined to match the correct component in all subjects but can be tuned with additional prior information about the spatial and temporal activation of the desired component. In total, 5 component activation patterns can be extracted, as illustrated in Figure 4.4.

Algorithm 3 Refined multi-component beamformer trial feature extraction procedure.

```

1: procedure MCBF2FEATURES( $X, \phi_0, \text{time\_windows}, \text{channel\_windows}$ )
2:    $S, \phi \leftarrow \text{PREPROCESS}(X, \phi_0)$ 
3:    $S^* \leftarrow S \setminus \text{BADEPOCHS}(S)$ 
4:    $\bar{a} \leftarrow \frac{\sum S_{1\text{-back},T}^*}{\#S_{1\text{-back},T}^*} - \frac{\sum S_{1\text{-back},N}^*}{\#S_{1\text{-back},N}^*}$ 
5:   for  $j \in 1, \dots, c$  do
6:      $a \leftarrow \text{REFINEACTIVATIONPATTERN}(\bar{a}, S^*, \text{time\_windows}_j, \text{channel\_windows}_j)$ 
7:      $w_j = \text{CALCULATESTBF}(a, S^*)$ 
8:     for  $S_i \in S$  do
9:        $y_{i,j} = w_j^\top$ 
return  $Y$ 

```

4.3 Quantization

The extracted trial features are categorized in discrete groups to complete the linguistic transcription algorithm. A symbol can then be assigned to each group. There are multiple ways to perform this categorization. The simplest method is to define a fixed grid in the feature space and assign a symbol to each cell in the grid. This approach will often lead to many cells without or with only a few trials and is very sensitive to noise. It also does not take into account the internal structure in the data, for instance, clusters of similar trials or dense regions, which ideally would be

Table 4.3: Time and channel windows for refined component activation patterns.

AP	time window	channel window	ERP component
1	100-175ms	Cz, FC1, FC2, Fz, AF3, AF4, Fp1, Fp2, C3, FC5, F3, C4, FC6, F4	P100
2	200-275ms	O2, Oz, O1, P04, P03, Pz, CP1, CP2, P7, P3, CP5, P4, P8, CP6	N100
3	200-275ms	Fz, AF3, AF4, Fp1, Fp2, F3, F4	P200
4	300-600ms	all	P300
5	725-800ms	Cz, FC1, FC2, Fz, AF3, AF4, Fp1, Fp2, C3, F3, CP6, C4, F4	

4. METHODS

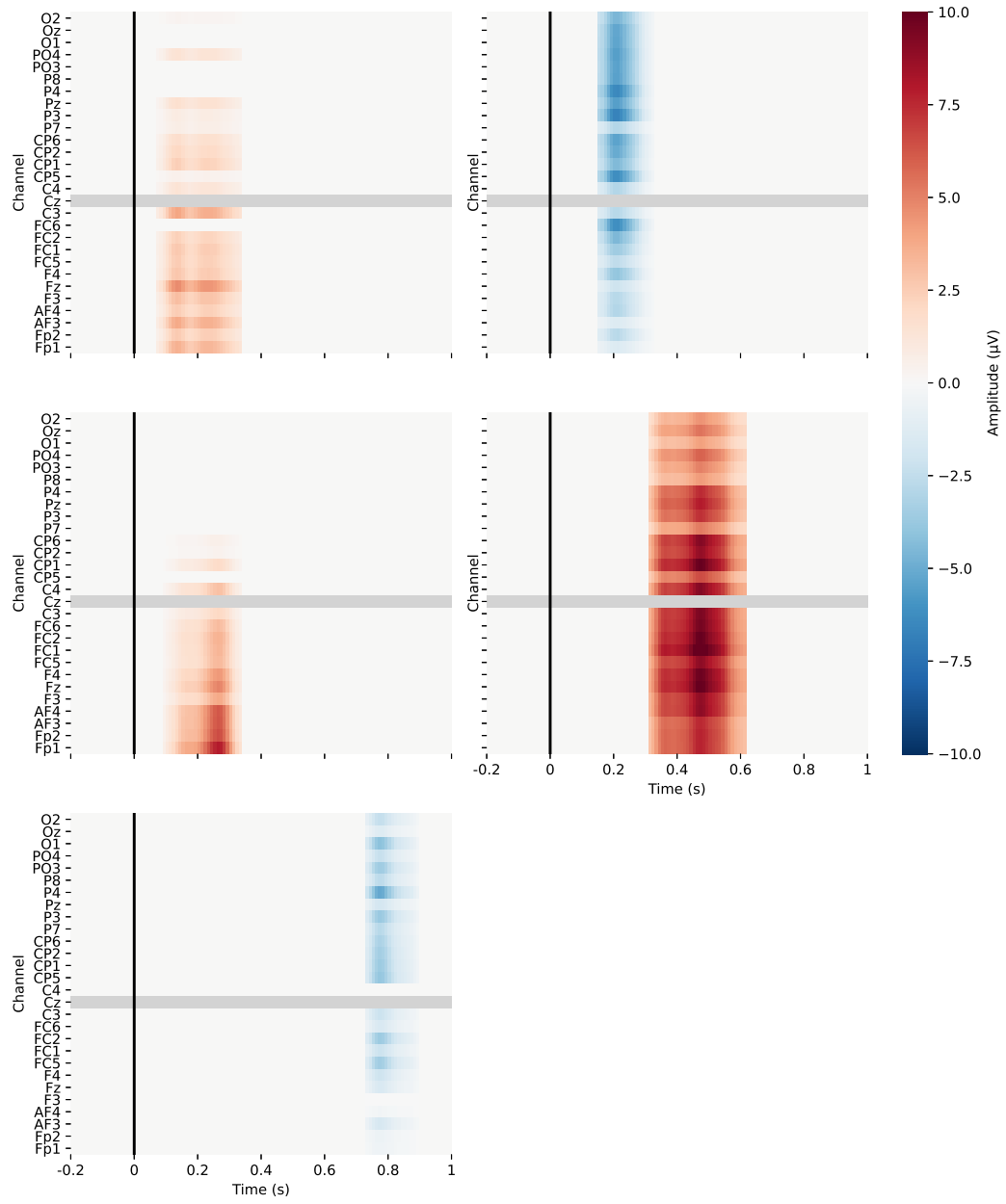


Figure 4.4: Refined component activation patterns for *Subject 1*.

assigned the same symbol. Another approach to mitigate some of these drawbacks is defining a dynamic grid instead of a fixed grid. Each feature dimension can be divided into dense regions based on its marginal distribution, and these divisions can be combined to form a grid. Finally, partitioning clustering was selected to optimally exploit the density information present in the data, decreasing the number of symbols required.

Multiple clustering methods were considered. DBSCAN[16] has some desirable properties, like the ability to automatically indicate outliers as a separate category. However, due to the lack of clearly pronounced separated clusters in the beamformer scores, it is hard to use density based clustering methods that automatically decide the number of clusters. Grid-based subspace clustering techniques with fixed[2] or adaptive[21] grid size were also considered since they have the ability to transparently determine clusters located in only a subset of the features. The latter is a desirable property since it is not a priori known if all constructed features are informative. Subspace clustering methods could prove useful in extensions of this work intending to extract more features from each trial, e.g., by constructing multiple localized beamformers per time window to better capture the spatial distribution of single-trial ERPs.

Ultimately, k -medoids clustering was selected as a starting point for its simplicity. k -medoids is suitable since the features are of relatively low dimensionality, and because it is more robust to extreme outliers (which might be present due to noise) than k -means clustering. The use of the Mahalanobis distance accounts for the correlation between the component beamformer scores (see Figure 4.5). MATLAB implements k -medoids using the Partitioning Around Medoids[30] (PAM) algorithm, minimizing the Mahalanobis distances between cluster members and the representative medoid of their respective clusters. After ten repetitions of the PAM algorithm with random cluster initializations, k -medoids retains the clustering with the best distance evaluation.

k -medoids requires a specified number of clusters k . Depending on the application for which the linguistic transcription algorithm is applied as a processing step, this number k can be set manually to determine the desired dictionary size, potentially with prior knowledge about the number of clusters present in the data. In the general unsupervised setting, however, k is not known a priori. An optimal k can be selected from a range of values according to some criteria.

The first of the criteria used in this work is the Calinski-Harabasz Index. The Calinski-Harabasz Index is defined as the ratio of the between-clusters dispersion mean and the within-cluster dispersion[8], given by Equation 4.2.

$$CH = \frac{tr(B_c)}{tr(W_c)} \cdot \frac{r - k}{k - 1} \quad (4.2)$$

with B_c the between cluster dispersion matrix for a given clustering c with k clusters, and W_c the corresponding within-cluster dispersion matrix. Both dispersion matrices are calculated from beamformed scores Y using the Mahalanobis distance.

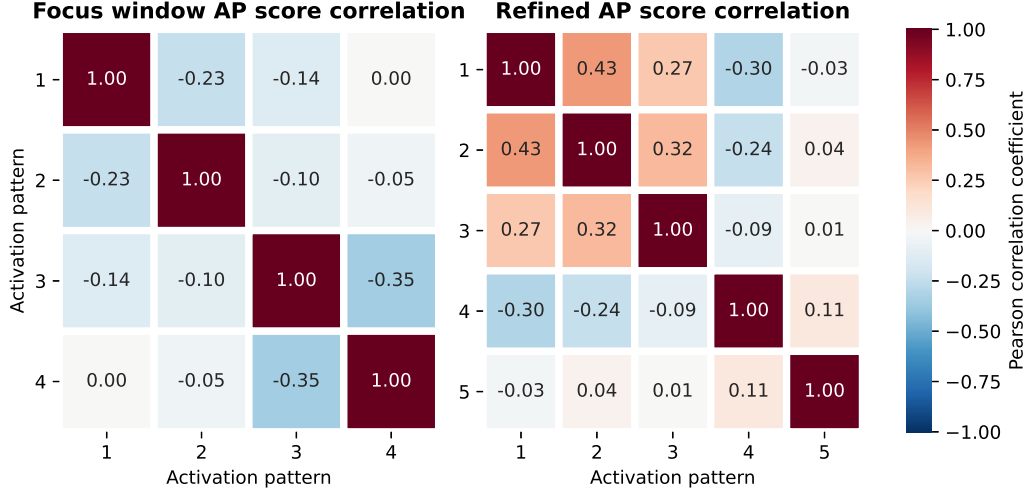


Figure 4.5: Pearson correlation coefficients of features extracted by the multi-component beamformer from 1-back target trials of *Subject 1*.

The Calinski-Harabasz criterion is suited to determine the optimal value for k if there are distinct clusters present in the beamformer scores for a given set of trials. However, this is not always the case. For instance, the N -back trials are not clearly separated in clusters after beamforming. As [Figure 4.6](#) shows, trials and not trials are not even clearly separated. However, this does not mean these trials cannot meaningfully be divided into groups and no transcription symbols can be defined.

Another proposed evaluation metric tackles this problem. As stated in the problem statement, the symbols assigned to trials should represent different ERP morphologies. Therefore, a clustering instance can be evaluated by comparing the between-cluster spread of the trials. If a reliable grouping is achieved, the shapes of the ERPs in any given cluster should be sufficiently different from trials in other clusters. Due to the noise present in single trials, this spread cannot be determined directly from the trials in the cluster. Hence, all trials in a cluster are first averaged to form a representative trial for each cluster. Next, a measure of this spread can be defined as follows:

$$DIST = \frac{1}{k^2} \|D_c\|_2 \quad (4.3)$$

for a given clustering c with k clusters. D_c is the pairwise Euclidean distance matrix between the average trial for each cluster. Since the 2-norm of D_c is proportional to the number of elements in D_c , division by k^2 is used as normalization. Elements in D_c can be calculated as

$$d_{i,j}^c = \frac{\sum S_{c_i}}{\#S_{c_i}} - \frac{\sum S_{c_j}}{\#S_{c_j}} \quad (4.4)$$

with S_{c_i} all trials in cluster i .

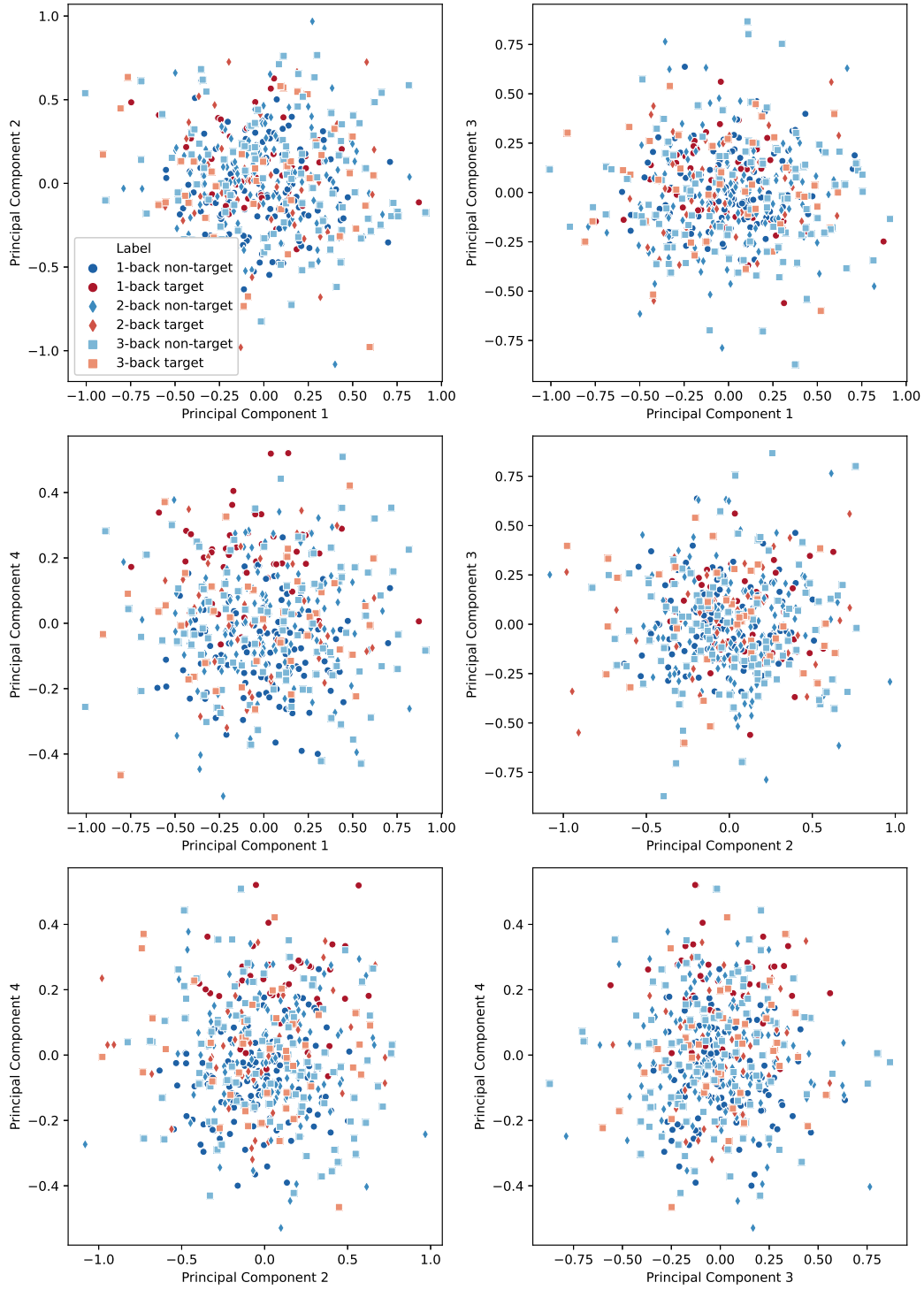


Figure 4.6: Pairwise principal component plots of focus window beamformer scores for all trials of *Subject 1*.

This metric is only a measure of between-cluster difference and does not measure within-cluster similarity. Since the power of noise signals in a single trial is often higher than the power of the ERP signal, it is hard to directly measure the within-cluster similarity from the trials without feature extraction. As a consequence, the developed metric *DIST* cannot be used as an optimization cost function for the k -medoids algorithm. A good *DIST* value is a necessary yet not sufficient condition for a good transcription result.

Chapter 5

Experiments

First, an experiment using the spatio-temporal beamformer tests if the available data contains sequential/contextual structure present in the ERP components activity, and whether the beamformer can capture this information (see [section 5.1](#)). Additionally, if this experiment produces convincing results, the uncovered sequential structure can be used as a validation method for the final algorithm’s ability to capture context in the transcribed symbol sequence. Next, experiments are designed and carried out to test the performance and expressiveness of the multi-component beamformer for ERP feature extraction (see [section 5.3](#)), and to evaluate the clustering transcription process of these features (see [section 5.4](#)). These experiments do not aim to prove neuroscientific hypotheses but rather study the effectiveness of the methods described above.

5.1 Exploratory study

This experiment aims to investigate whether contextual information is present in the available sequences of ERPs and to show that this information can be extracted by measuring scores for component activation patterns with the spatio-temporal beamformer. Therefore, this study tries to verify Hypothesis 1 regarding the N -back experiment:

Hypothesis 1 *Does the fact of whether the current stimulus is a target stimulus or not affect the processing of future stimuli in the brain?*

In order to verify Hypothesis 1, trials will be divided into classes based on whether they are a target or non-target, and on a fixed-length history of target and non-targets that preceded that target. For simplicity, these classes are assumed to be independent. If the hypothesis is correct, the difference in the processing of these classes in the brain will result in different ERP shapes per class (e.g., higher or lower component amplitudes or differences in latency). However, these ERPs can not directly be compared, since noise present in the signal would obscure the differences. Therefore, two techniques are leveraged to uncover sequential differences: averaging and the spatio-temporal beamformer with focus window.

5. EXPERIMENTS

Hypothesis 2 *Does the fact of whether the current stimulus is a target stimulus or not affect the processing of the next stimulus in the brain?*

To validate Hypothesis 2, four classes of trials are constructed. Target trials preceded by a target trial (indicated by TT) are compared to target trials that appear after a non-target trial (NT). Similarly, TN and NN trial subsequences are compared. First, the averages of these classes in the 1-back case are visually compared. Figure 5.1 shows the difference between all TT and NT trials and the difference between all TN and NN as compared to the difference of all T and N trials.

A second experiment aims to verify whether these differences are significant and whether they can be captured by spatio-temporal beamforming. Two spatio-temporal beamformers are constructed: the first one focuses on the P300 component while the second one focuses on components leading up to the P300. For each subject, the beamformers filter all 1-back trials. The covariance matrix of the

Figure 5.1: Contrasts of length 1 trial subsequences compared with T/N contrast for 1-back trials averaged over all subjects. Corrupted channels were set to 0 before averaging.

beamformers is estimated from all 1-back trials. The resulting scores are compared across subsequences.

5.2 Results

The above experiment is run with time windows 100ms-300ms and 300ms-600ms after stimulus onset. Table 5.1 and Table 5.2 compare the resulting beamformer scores per subject for different target/non-target trial subsequences. Similarly to Figure 5.1, responses to targets following a non-target are compared with targets following another target and responses to non-targets following other non-targets are compared to non-targets following a target. As a reference, targets are also directly compared to non-targets.

First and foremost, it should be noted that the results show little significance, possibly due to low data availability (cfr Table 4.1) and high SNR, even after beamforming. However, some results do seem to hint towards a relation between the previous stimulus and the current response. Especially the scores measured by the 100ms-300ms window beamformer for targets following another target are generally lower than scores for targets following a non-target, but not enough samples are available to confirm this. While the results are generally inconclusive, this experiment seems to indicate that possible contextual variations cannot be picked up by a single beamformer, motivating the introduction of multi-component beamforming.

Finally, since the experiment did not produce consistently significant results, the presence of TT-NT and TN-NN differences cannot be used as a validation method for the expressiveness of the linguistic transcription.

Table 5.1: Wilcoxon rank sum test p -values for comparison of 300ms-600ms window beamformer output for 1-back trial subsequences.

	<i>Subject 1</i>	<i>Subject 2</i>	<i>Subject 3</i>	<i>Subject 4</i>
T-N	< 0.0001	< 0.0001	< 0.0001	< 0.0001
TT-NT	0.5056	0.0341	0.7392	0.6743
TN-NN	0.5893	0.1951	0.1060	0.9490

Table 5.2: Wilcoxon rank sum test p -values for comparison of 100ms-300ms window beamformer output for 1-back trial subsequences.

	<i>Subject 1</i>	<i>Subject 2</i>	<i>Subject 3</i>	<i>Subject 4</i>
T-N	< 0.0001	0.0006	< 0.0001	< 0.0001
TT-NT	0.2673	0.2731	0.1458	0.3463
TN-NN	0.0796	0.8393	0.5578	0.0688

5.3 Single-trial classification

5.3.1 Experiment

The next experiment constructs multiple supervised classifiers to predict known labels in the dataset from the beamformer scores in order to evaluate the expressiveness of the features extracted by multi-component beamforming. Three feature extraction models are compared: the classic spatio-temporal beamformer (STBF), the focus window multi-component beamformer (MCBF1) and the multi-component beamformer with refined activation patterns (MCBF2). Each model is combined with an LDA classifier and evaluated per subject using ten repetitions of stratified five-fold cross validation. Finally, these models are compared in three classification tasks: 1) distinguishing 1-back targets from 1-back non-targets, 2) distinguishing 1-back targets from 2-back targets, 3) distinguishing 1-back targets from 3-back targets. These classes were selected since ground truth for these labels is known in the dataset, and there are proven differences between ERP responses in each compared class. For instance, the P300 component has a larger amplitude in 1-back targets than in 1-back non-targets. P300 also has a lower amplitude and higher latency in 2-back and 3-back targets than in 1-back targets[47]. Since the linguistic transcription algorithm should ultimately be able to capture such categorical differences in the symbol groups, these experiments will appropriately evaluate the feature expressiveness for transcription.

Each cross-validation fold, the activation pattern and covariance matrix of the beamformer are determined from the training trials, as well as the training of the LDA classifier. For experiments 1), 2) and 3), calculating the covariance matrix from respectively 1-back trials, 1-back and 2-back trials, and 1-back and 3-back trials in the training set yielded the highest performance. Epochs marked as corrupted by noise were not taken into account.

5.3.2 Results

Figure 5.2 and Tables 5.3, 5.4 and 5.4 show classifier performance for experiments 1), 2) and 3) using ROCs and the respective areas under the curve.

The obtained results show that the MCBF2 feature extraction performs worst in almost every case. This lack in performance might be due to several reasons, most notably that the pattern refinement procedure cannot capture all components

Table 5.3: Classification performance (mean area under ROC \pm standard deviation) for 10 repetitions of 5-fold cross validation of classification of 1-back targets and 1-back non-targets.

	<i>Subject 1</i>	<i>Subject 2</i>	<i>Subject 3</i>	<i>Subject 4</i>
STBF + LDA	0.73 \pm 0.09	0.80 \pm 0.08	0.62 \pm 0.11	0.69 \pm 0.80
MCBF1 + LDA	0.74 \pm 0.09	0.80 \pm 0.08	0.64 \pm 0.10	0.67 \pm 0.09
MCBF2 + LDA	0.68 \pm 0.10	0.70 \pm 0.10	0.55 \pm 0.10	0.59 \pm 0.12

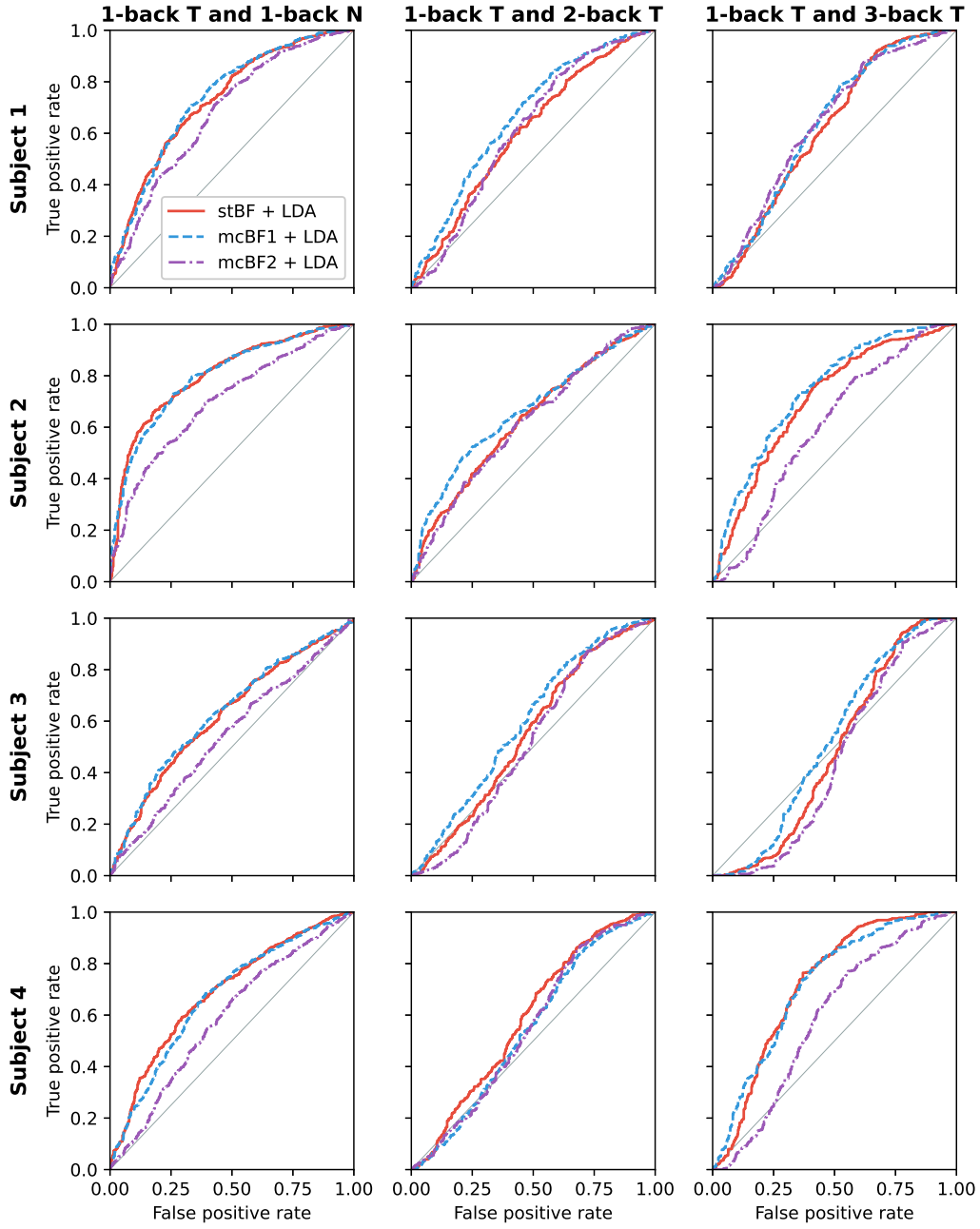


Figure 5.2: Classifier ROCs for all models per subject for experiments 1), 2) and 3).

5. EXPERIMENTS

Table 5.4: Classification performance (mean area under ROC \pm standard deviation) for 10 repetitions of 5-fold cross validation of classification of 1-back targets and 2-back targets.

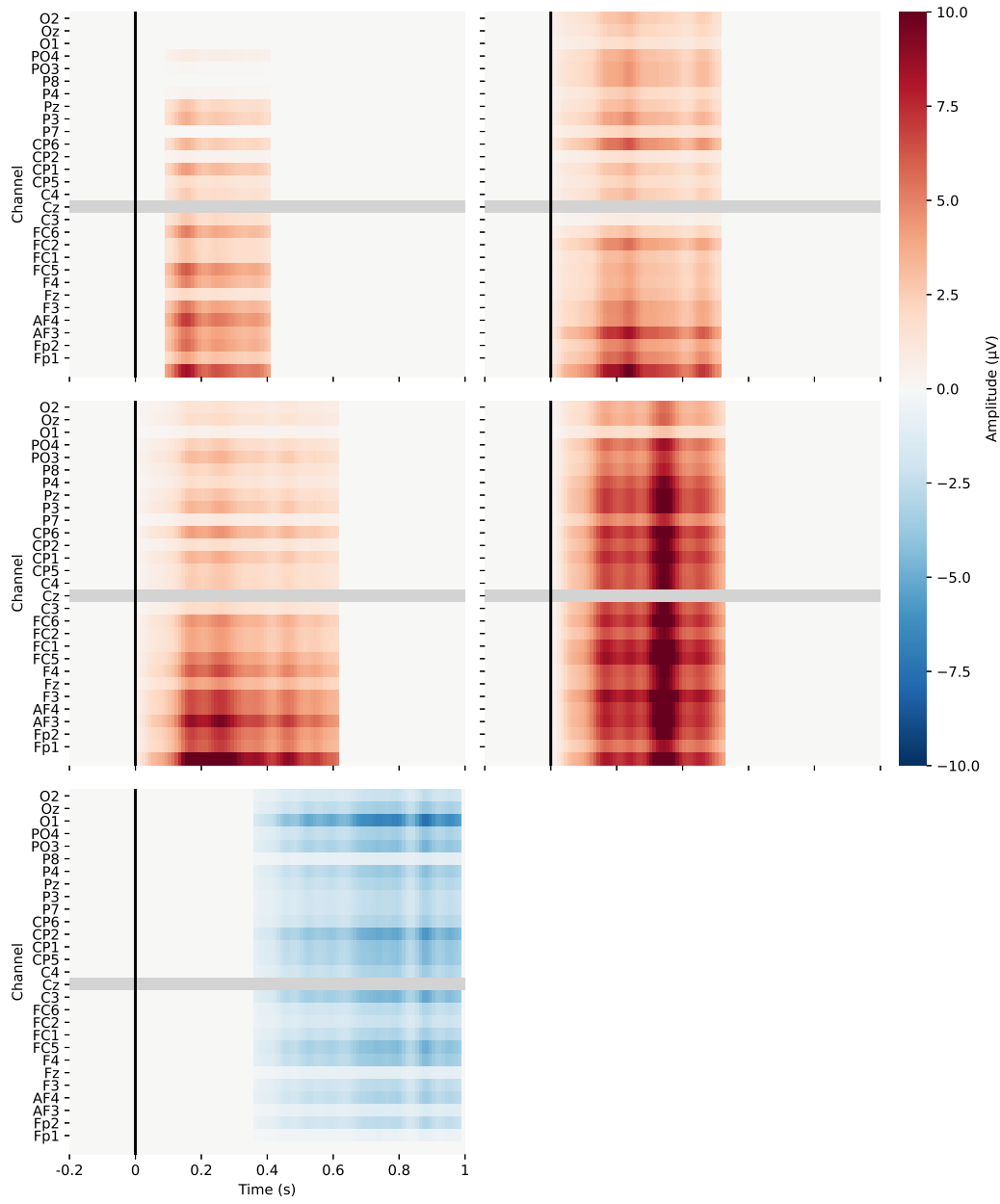
	<i>Subject 1</i>	<i>Subject 2</i>	<i>Subject 3</i>	<i>Subject 4</i>
STBF + LDA	0.61 ± 0.14	0.63 ± 0.13	0.56 ± 0.12	0.59 ± 0.12
MCBF1 + LDA	0.67 ± 0.13	0.67 ± 0.13	0.60 ± 0.13	0.55 ± 0.12
MCBF2 + LDA	0.61 ± 0.15	0.62 ± 0.13	0.53 ± 0.15	0.56 ± 0.14

Table 5.5: Classification performance (mean area under ROC \pm standard deviation) for 10 repetitions of 5-fold cross validation of classification of 1-back targets and 3-back targets.

	<i>Subject 1</i>	<i>Subject 2</i>	<i>Subject 3</i>	<i>Subject 4</i>
STBF + LDA	0.62 ± 0.12	0.70 ± 0.13	0.49 ± 0.11	0.73 ± 0.11
MCBF1 + LDA	0.63 ± 0.12	0.74 ± 0.14	0.54 ± 0.14	0.73 ± 0.11
MCBF2 + LDA	0.64 ± 0.13	0.62 ± 0.16	0.45 ± 0.14	0.60 ± 0.14

specified in [Table 4.3](#) for all subjects. An example of this can be observed when comparing MCBF2 performance between *Subject 1* and *Subject 4*. In *Subject 1*, the performance of MCBF2 + LDA is just slightly lower than the performance of the other models. For *Subject 4*, there is a significantly larger gap in performance between MCBF2 + LDA and other models. [Figure 5.3](#) shows that the second refined activation pattern for *Subject 4* does not capture the negative N200 component as intended, in contrast to the correct behaviour seen in [Figure 4.4](#) for *Subject 1*. Hence, features for *Subject 4* are less informative, leading to a drop in performance. Similar activation pattern construction faults can be seen in the refined activation patterns for *Subject 2* and *Subject 3*. Due to the high inter-subject variability of ERPs, this problem is hard to tackle. Presumably, the overgeneralization of the refined activation patterns is another cause of poor MCBF2 performance, meaning the refinement procedure cancels some of the relevant structures in the unrefined, windowed activation patterns.

The windowed multi-component beamformer MCBF1 performs roughly similar, or, in some cases, better than the original spatio-temporal beamformer STBF for the presented classification tasks. Relative to STBF, MCBF1 performs better in tasks 2) and 3) in comparison to task 1). In task 1), the goal is to differentiate 1-back targets from 1-back non-targets. Because of the nature of the grand average activation pattern, STBF should be more suited to perform task 1). The generally higher performance of STBF for task 1) as compared to task 2) confirms this claim.

Figure 5.3: Refined component activation patterns for *Subject 4*.

5.4 Clustering

5.4.1 Experiment

This last experiment evaluates the performance of the linguistic transcription algorithm with k -medoids clustering while simultaneously finding suitable dictionary sizes for the ERPs in the N -back data. To ensure the algorithm never attributes the same symbol to targets and non-targets, the data labeling is used to cluster targets and non-targets separately. Since k -medoids clustering is relatively robust to extreme outliers, the transcription algorithm should be able to handle transcribing corrupt trials, as long as they are not included in the calculation of the activation patterns and the estimation of the covariance matrix of all subjects. For this experiment, however, they are discarded to enable better analysis of the structure present in the data.

For each subject, features are extracted with either MCBF1 or MCBF2. The covariance matrix for both beamformers is estimated per subject from all available trials. Subsequently, k -medoids clustering is performed multiple times for k ranging from 2 to 10. For each k , 10 repetitions of k -medoids clustering are calculated, each repetition selecting the best of 10 PAM repetitions. Finally, the evaluation metrics defined in [section 4.3](#) are applied.

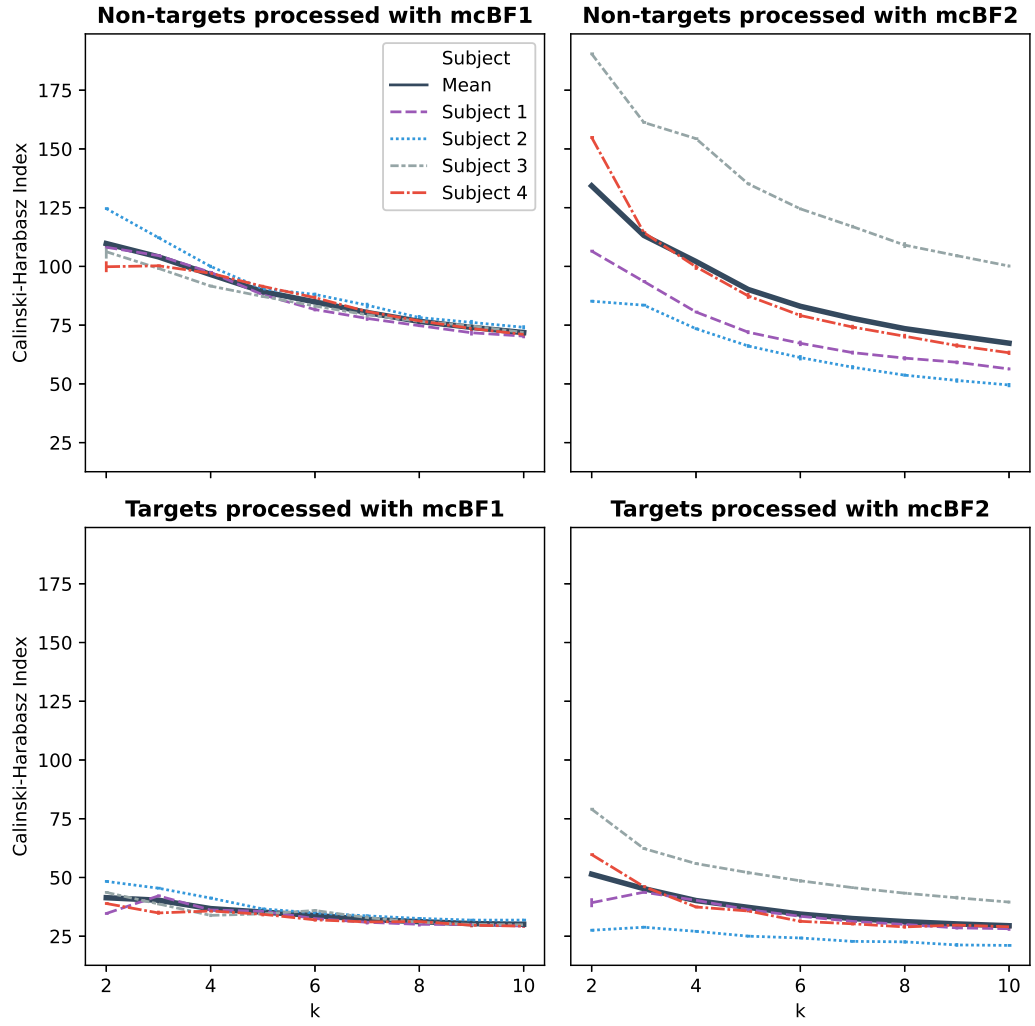
5.4.2 Results

[Figure 5.4](#) shows the Calinski-Harabasz clustering validity index results. For both MCBF1 and MCBF2, as well as for targets and non-targets, the Calinski-Harabasz Index consistently decreases as the number of clusters k increases, without peaking at an optimal k value. The absence of a peak value is an indication that no clearly separated clusters are present in the data, as indicated above in [section 4.3](#). This absence could either be caused by the fact that there is little categorical structure present within target and non-target ERPs and variations lay on a continuum with no clear variations in density, or this structure is present in features that are not picked up by the multi-component beamformer, or noise contamination is too high to observe the clusters. No information which could verify either of these claims is available for the N -back dataset.

Calinski-Harabasz values for MCBF2 are generally higher than for MCBF1. This could indicate that MCBF2 can pick up more informative features for clustering than MCBF1, but this contradicts the results from supervised classification results. In the unsupervised setting, it is impossible to directly determine whether the constructed clusters reflect contextual structure in the data. Therefore, visual inspection of the average cluster trials is appropriate to draw conclusions, as well as quantifying the differences between these cluster representatives, as calculated by the *DIST* metric.

[Figure 5.5](#) indicates that, for targets, $k = 3$ clusters is an appropriate value to capture the variance in ERP shapes for most subjects. This might be related to the fact that the dataset contains 3 N -back complexity levels. For non-targets, the optimal k value is less pronounced. Notice that *DIST* values for targets cannot

be compared to *DIST* values for non-targets since the number of non-targets is significantly lower. Hence, the representatives will be averaged over fewer trials per cluster, and the averages will contain more extreme values originating from noise, increasing the euclidean distance between averages. [Appendix B](#) shows averages of cluster members and indicates different clusters contain different ERP shapes. $k = 6$ was chosen for these clusters in order to produce results that are visually differentiable.

Figure 5.4: Calinski-Harabasz index for k -medoids clustering of beamformer scores.

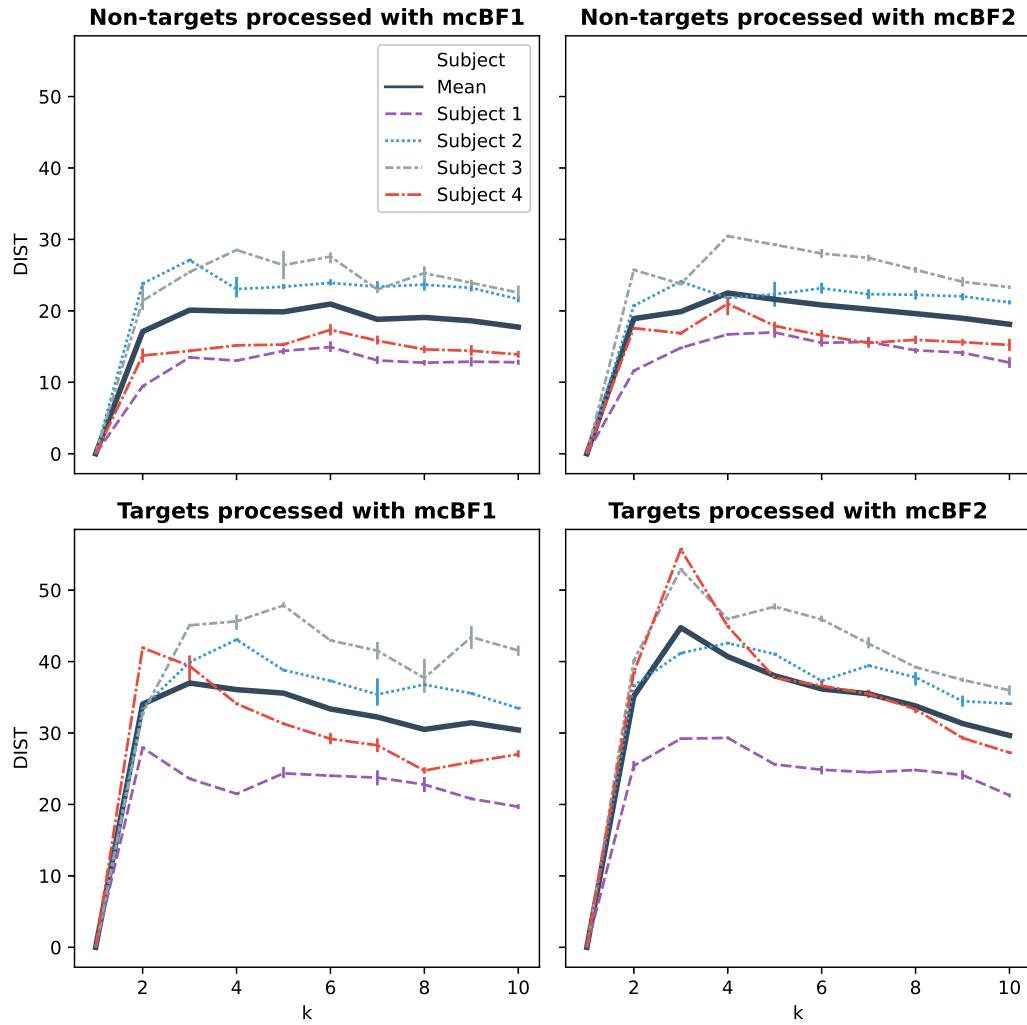


Figure 5.5: Averaged cluster representative trial distance matrix norm divided by k^2 for k -medoids clustering of beamformer scores.

Chapter 6

Conclusion

6.1 Summary

As indicated by earlier computational neuroscience and BCI research, ERP analysis is an inherently hard problem due to high noise levels present in the recorded EEG signal, which obscure the ERPs. Currently, this problem is often solved by repeating stimuli several times and averaging the ERP responses together to cancel out noise independent from the response. However, averaging imposes a penalty on the speed of ERP-based BCIs and creates an obstacle for techniques wishing to analyze ERPs in a sequential setting, in which averaging over subsequent trials would filter out the sequential information present.

For this reason, the field could benefit from a preprocessing step and corresponding representation format for ERPs that allows algorithms to analyze ERP responses in their context of surrounding stimuli and responses, without the need for averaging. The proposed linguistic transcription algorithm reduces a single EEG trial containing an ERP response and added noise to a categorical symbol representing some characteristics of the ERP response present in the trial. A sequential analysis algorithm can then process a sequence of these transcribed symbols.

A review of the related research shows a lack of techniques aiming to categorize or quantize EEG signal trials containing ERPs. Presumably, this can largely be explained by the unfavorable SNR of ERPs. The available techniques require extensive forward anatomical modelling. Techniques exist that can accurately evaluate ERPs while only averaging over a few trials, but these techniques are mostly aimed toward target/non-target classification and have not yet been applied in the unsupervised case. The spatio-temporal beamformer is a relatively new state-of-the-art transparent method for analyzing ERPs in a few repeated trials. Prior knowledge about timing and spatial distribution can be combined with data-driven methods to fit the subject when determining the spatio-temporal activation patterns.

The data acquired from the N -back experiments provides a reliable sequence of ERPs. Several studies have shown that the amplitudes and latencies of ERP components in N -back experiment trials encode information about the stimulus and the subject. First, a sequential analysis of these N -back trials was performed to

study whether a possible categorization of sequential effects in the N -back responses based on the previous response could yield a verification method to check if the transcription algorithm can capture sequential information. This exploratory study has not found any significant effects of the previous trial on the current trial. Hence, the results cannot be used to verify the algorithm’s ability to capture possible contextual structure. The results also indicate that single-component beamforming cannot extract information of desired granularity.

The final algorithm combines an extension of the spatio-temporal beamformer to multiple ERP components with k -medoids clustering to translate the extracted beamformer scores into symbols. If sufficiently pronounced, sequential effects in ERP sequences will result in different ERP shapes, which will, in turn, be expressed in different beamformer scores, resulting in the attribution of different symbols in the transcribed sequence. The beamformer approach has the added benefit of separately modeling signal and noise, thus providing a transparent feature extraction method. Beamformer scores for ERPs can directly be linked to specific ERP component shapes expressed by activation patterns. Furthermore, a transcribed symbol’s meaning can also be examined by inspecting the average of all trials being represented by that symbol.

Supervised classification experiments show that multi-component beamforming can possibly improve state-of-the-art performance in ERP classification tasks. The relatively good performance proves that the features extracted by multi-component beamforming are informative and form a reasonable basis for the transcription algorithm, especially in the case of the unrefined focus window-based multi-component beamformer. No convincing cluster structure is present in the features extracted by the beamformers. It is therefore hard to evaluate the goodness of the method that transforms these features in symbols, since no ground truth for possible sequential effects is known. However, visual inspection shows distinct categories of ERP signals. It is not yet clear how these should be interpreted in a neuroscientific context.

6.2 Future work

The logical next step for this research is to apply the algorithm to a text-based ERP dataset, which can benefit from the linguistic transcription to study context build-up by analyzing the distribution of symbol subsequences. The developed method can then also be evaluated based on expected results. Further research with a more extensive target/non-target based dataset also has yet to confirm the classification performance of the multi-component beamformer and test it in combination with different linear and non-linear classifier models, like, for instance, decision trees. If multi-component beamforming can consistently exceed the state-of-the-art performance of the spatio-temporal beamformer, this approach can play a role in the development of ERP target/non-target based BCIs, like communication devices for patients suffering from locked-in-syndrome.

The developed algorithm can still be improved in several ways. The construction of the component activation patterns shows the most potential for improvement. Ac-

tivation patterns that completely isolate a specified ERP component can potentially be constructed using blind source separation techniques like principal component analysis, independent component analysis, or dictionary learning. Multi-scale or latency invariant activation pattern models might also better handle the high variability in ERP responses. Lastly, activation patterns can be constructed by an expert with more prior knowledge to better match specific informative components. Another possibility for improvement is constructing the activation patterns in such a way that scores can be compared across subjects. Currently, data-driven component activation patterns differ significantly per subject due to inter-subject variability in ERP response and noise. More data and per subject tuning are required to smooth out these differences. Finally, classifiers and unsupervised methods based on the multi-component beamformer can benefit from feature selection schemes to determine which components are informative and which introduce noise.

Appendices

Appendix A

Multi-component Activation Patterns

A. MULTI-COMPONENT ACTIVATION PATTERNS

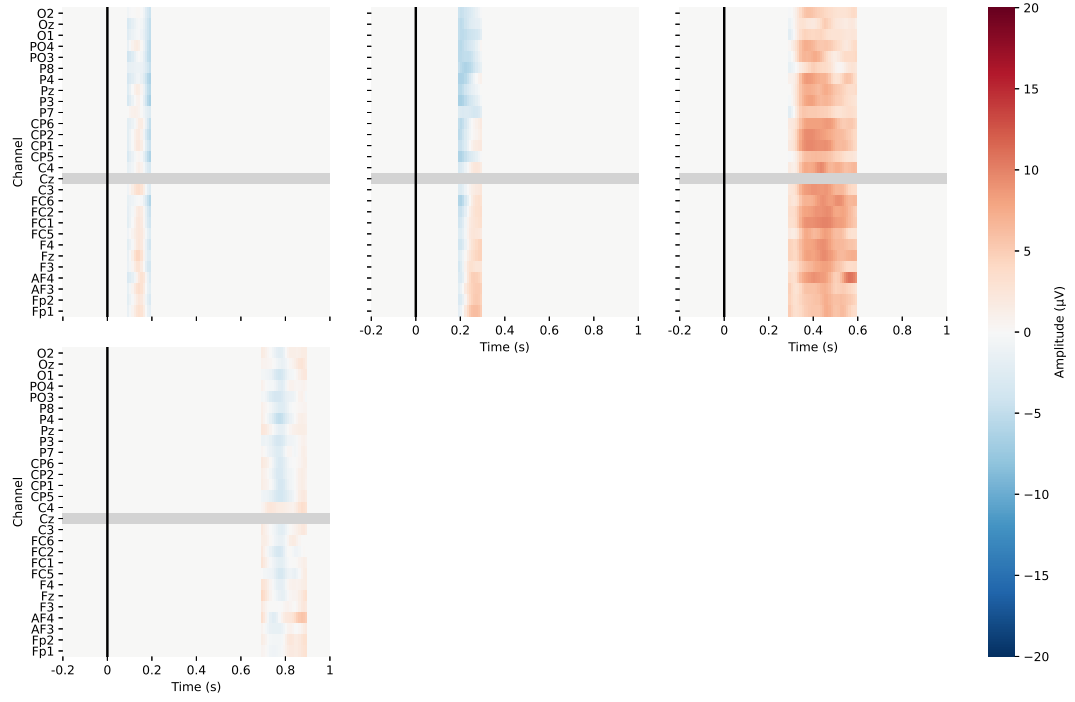


Figure A.1: Focus window component activation patterns for *Subject 1*.

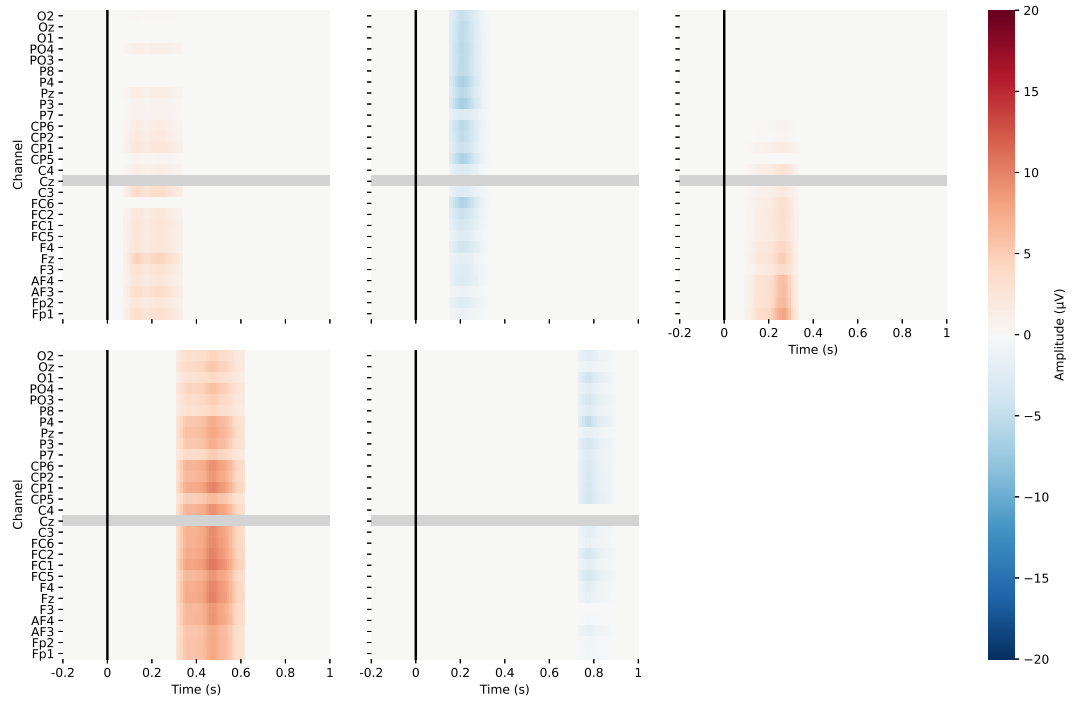


Figure A.2: Refined component activation patterns for *Subject 1*.

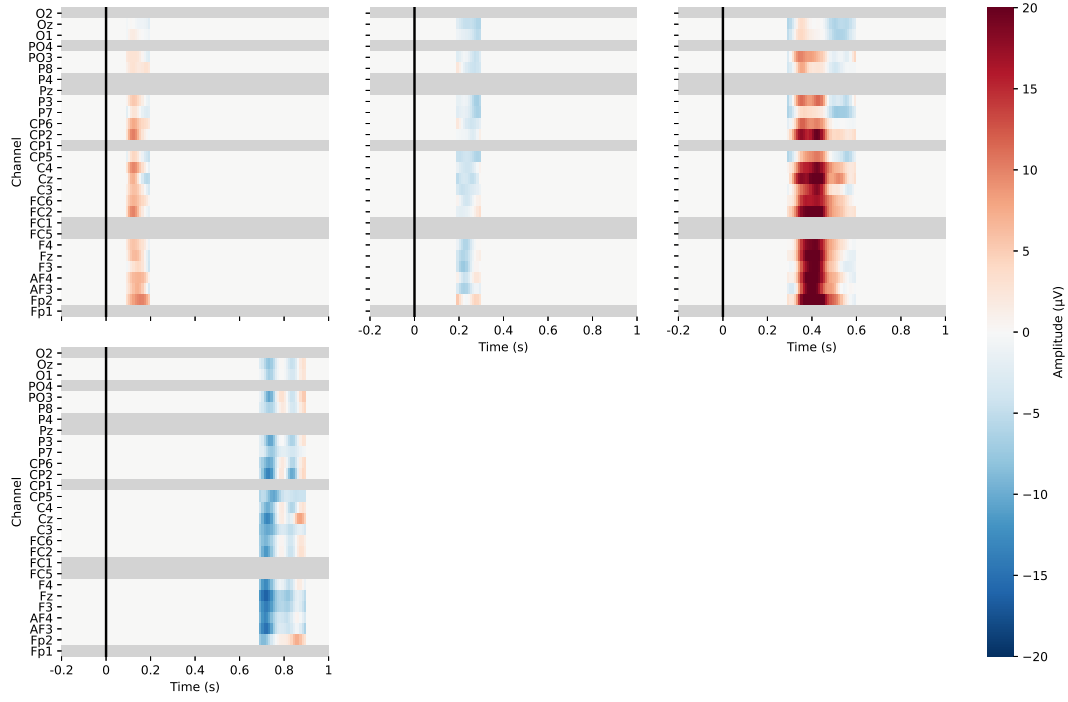


Figure A.3: Focus window component activation patterns for *Subject 2*.

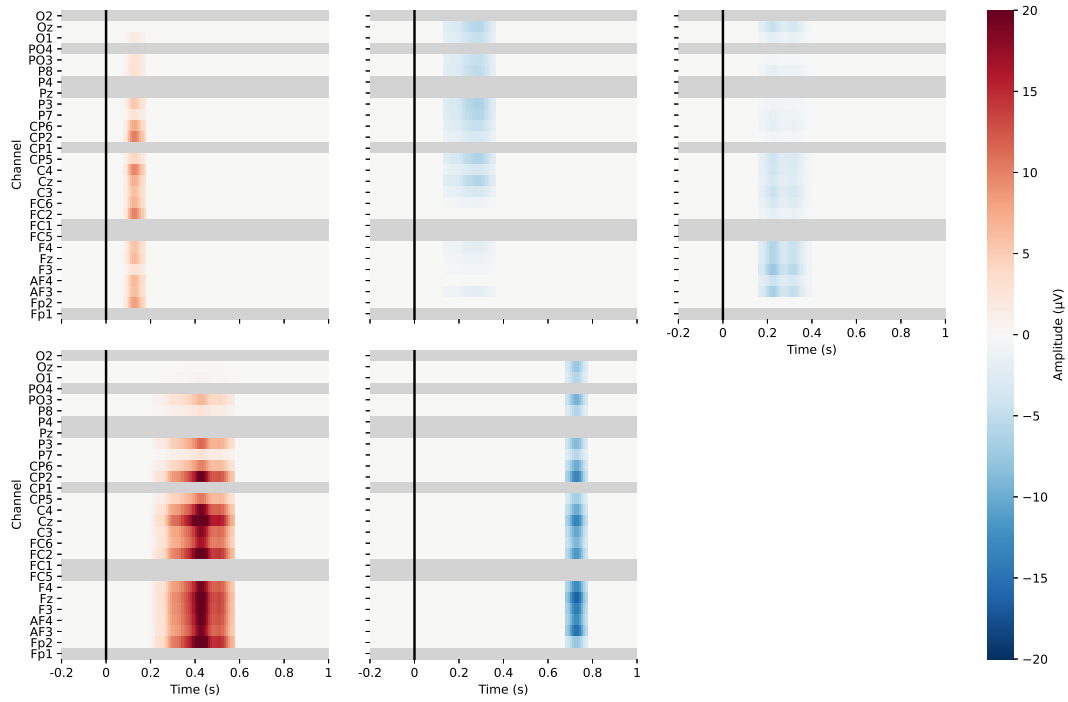


Figure A.4: Refined component activation patterns for *Subject 2*.

A. MULTI-COMPONENT ACTIVATION PATTERNS

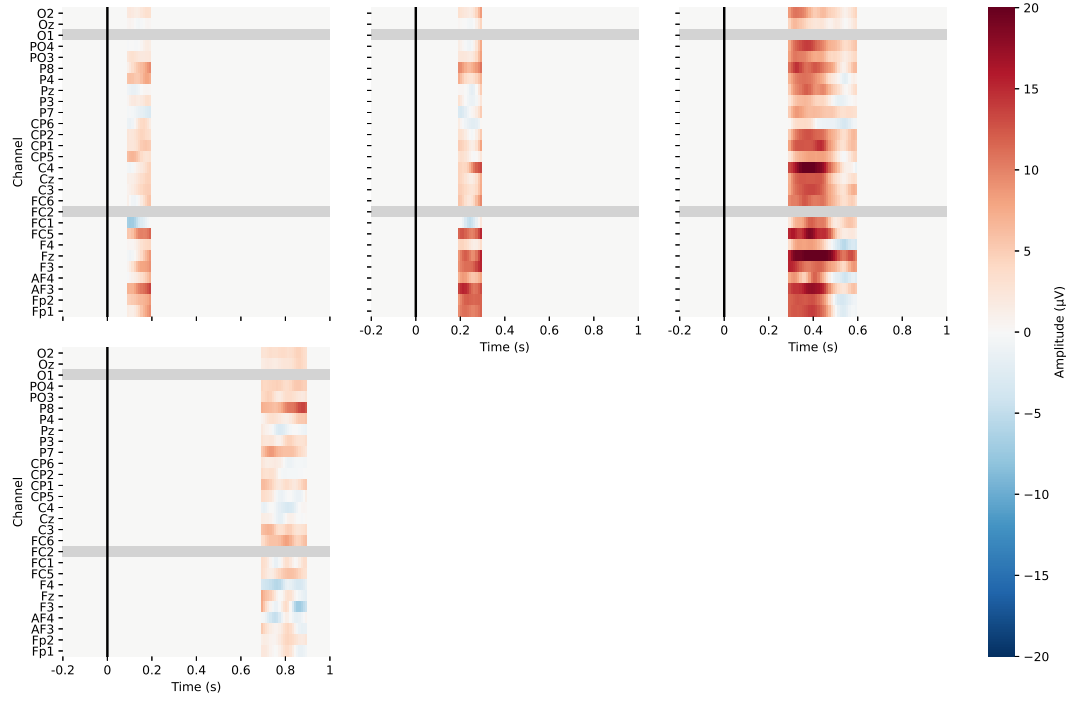


Figure A.5: Focus window component activation patterns for *Subject 3*.

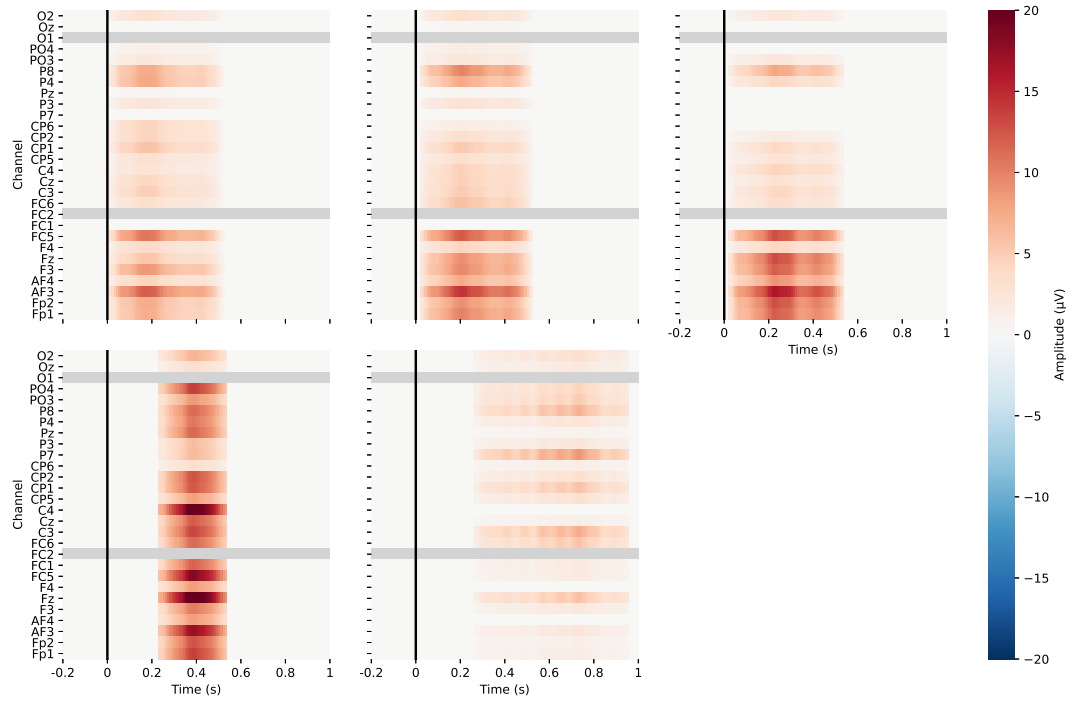


Figure A.6: Refined component activation patterns for *Subject 3*.

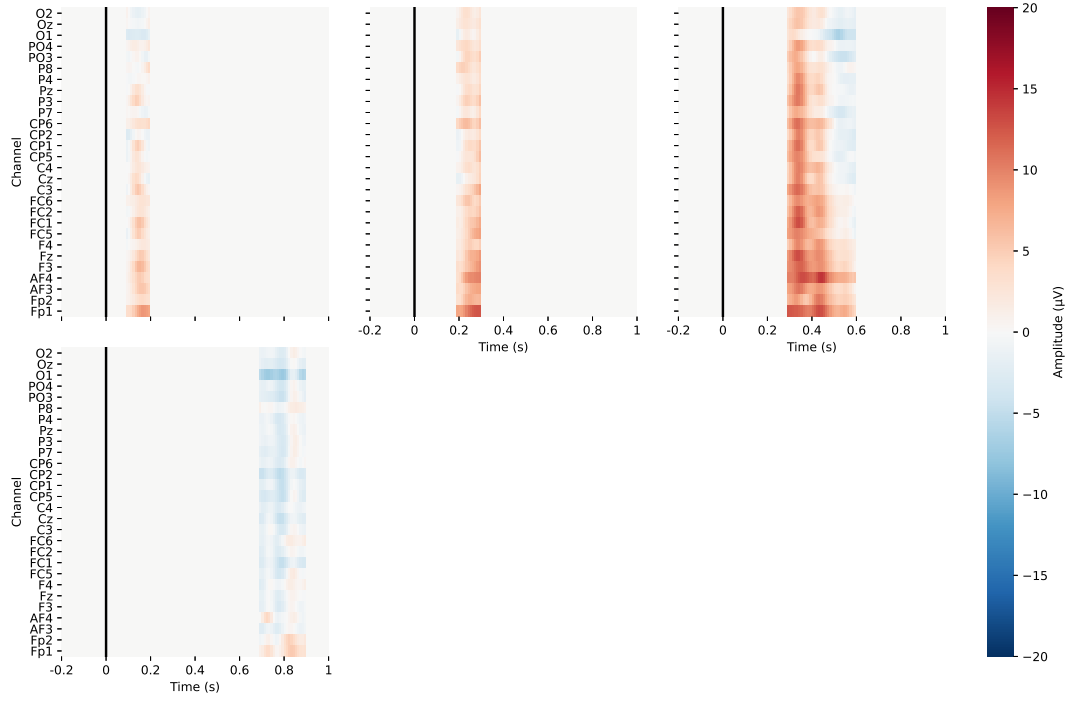


Figure A.7: Focus window component activation patterns for *Subject 4*.

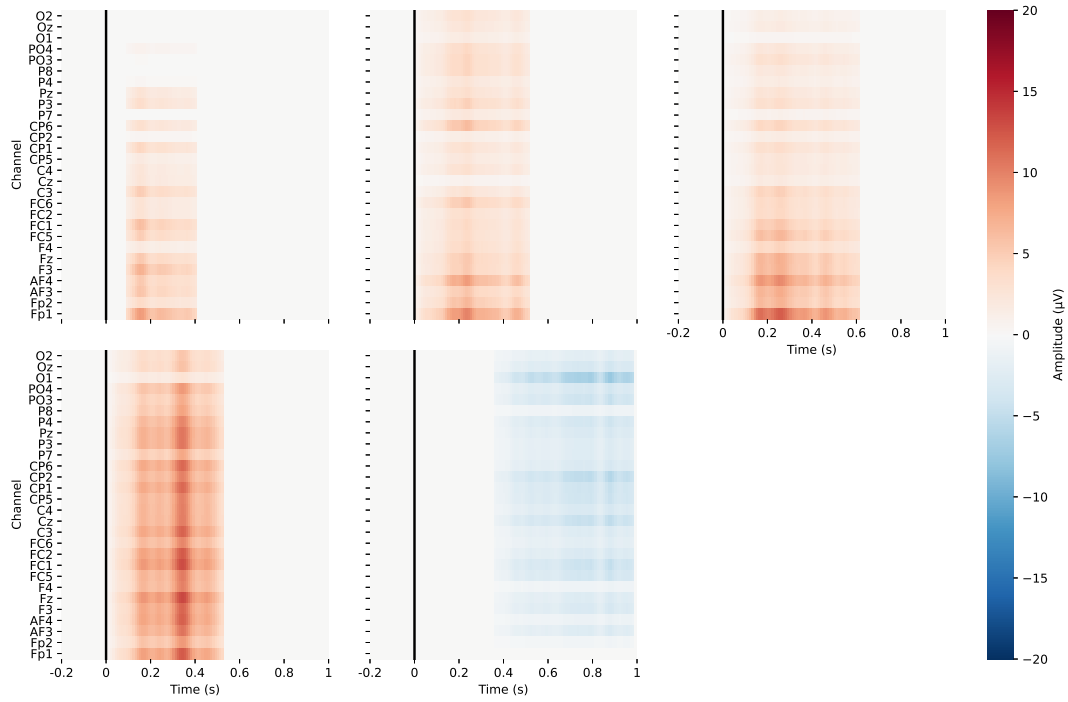


Figure A.8: Refined component activation patterns for *Subject 4*.

Appendix B

Symbol Representatives

B. SYMBOL REPRESENTATIVES

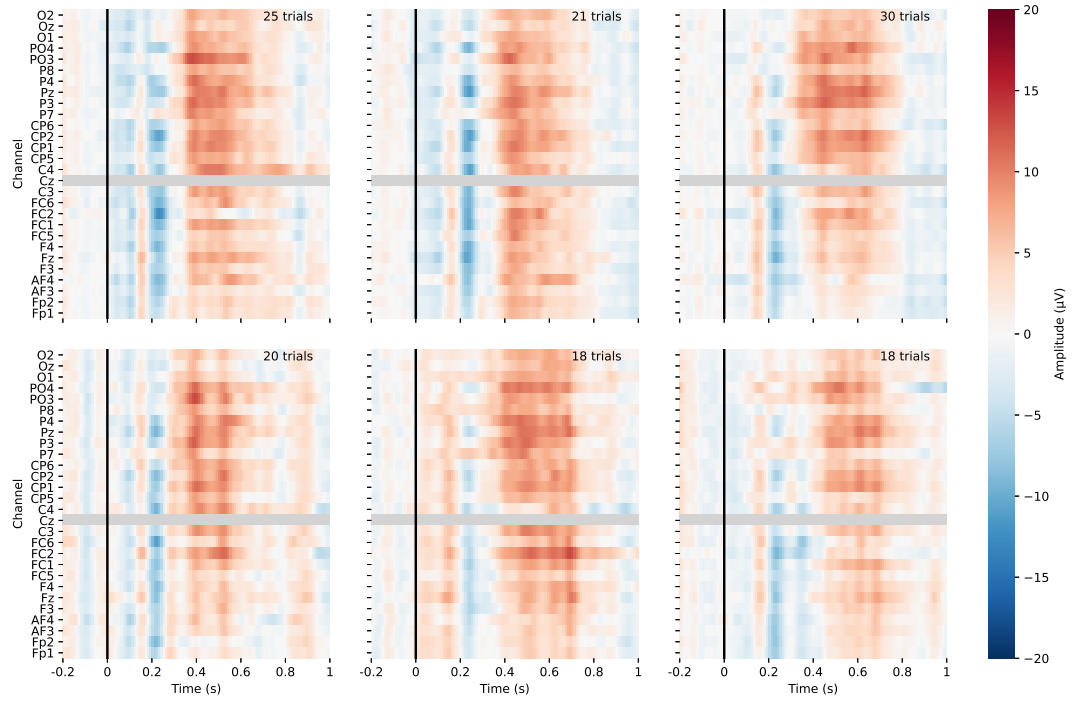


Figure B.1: Average of target trials per cluster for *Subject 1* processed with mcBF1 and clustered using $k = 6$.

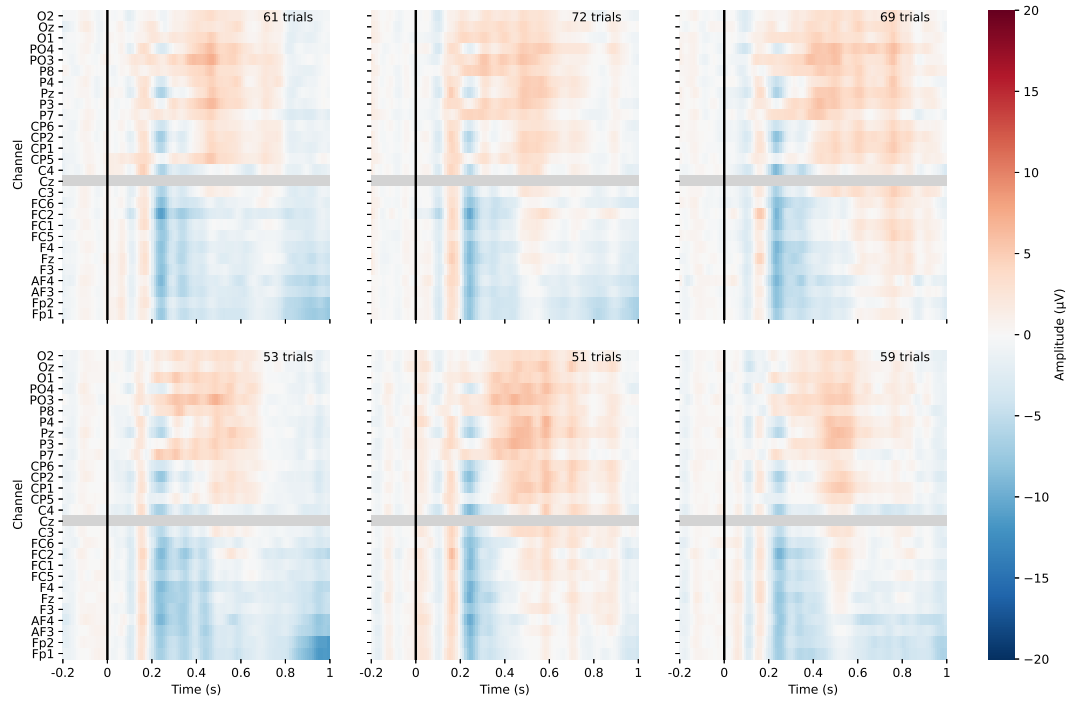


Figure B.2: Average of non-target trials per cluster for *Subject 1* processed with mcBF1 and clustered using $k = 6$.

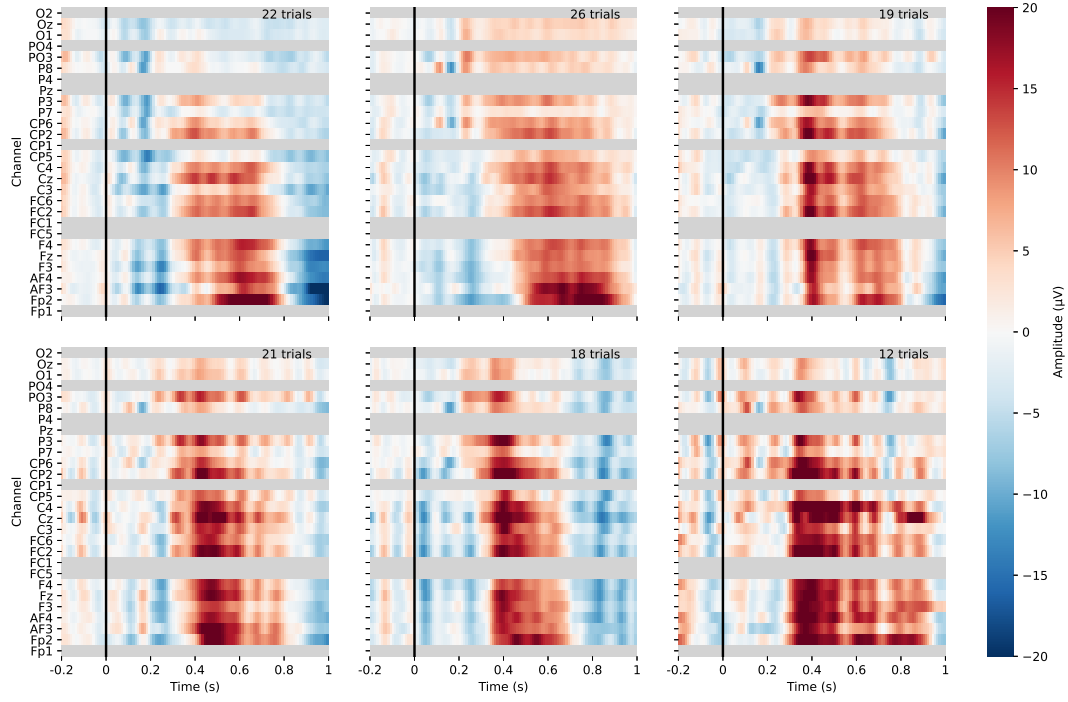


Figure B.3: Average of target trials per cluster for *Subject 2* processed with mcBF1 and clustered using $k = 6$.

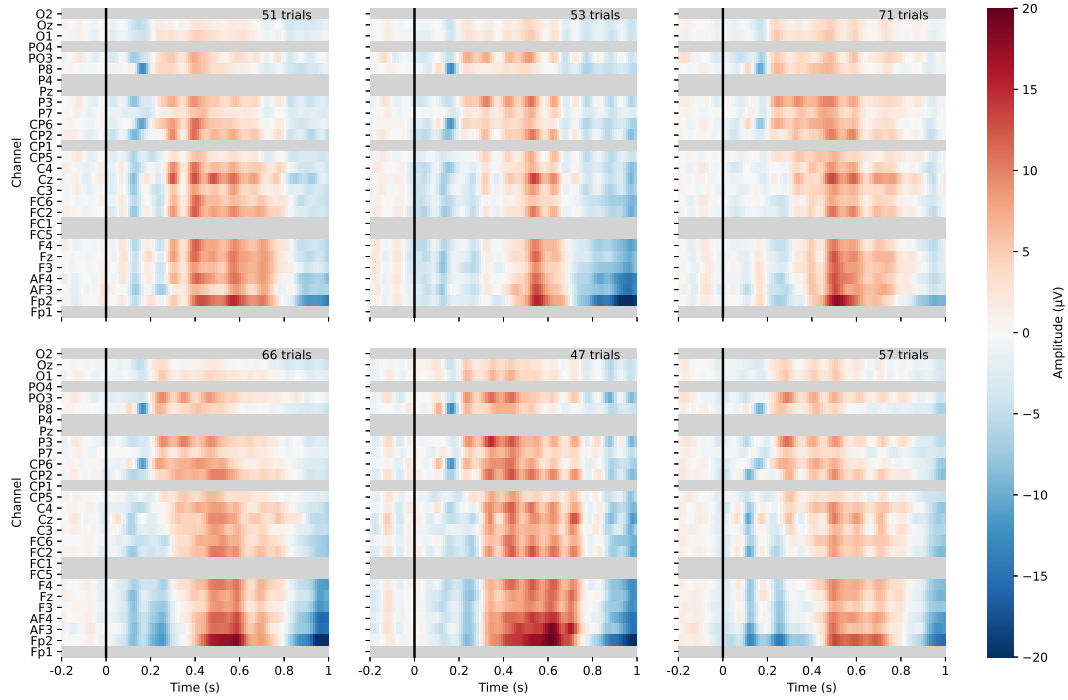


Figure B.4: Average of non-target trials per cluster for *Subject 2* processed with mcBF1 and clustered using $k = 6$.

B. SYMBOL REPRESENTATIVES

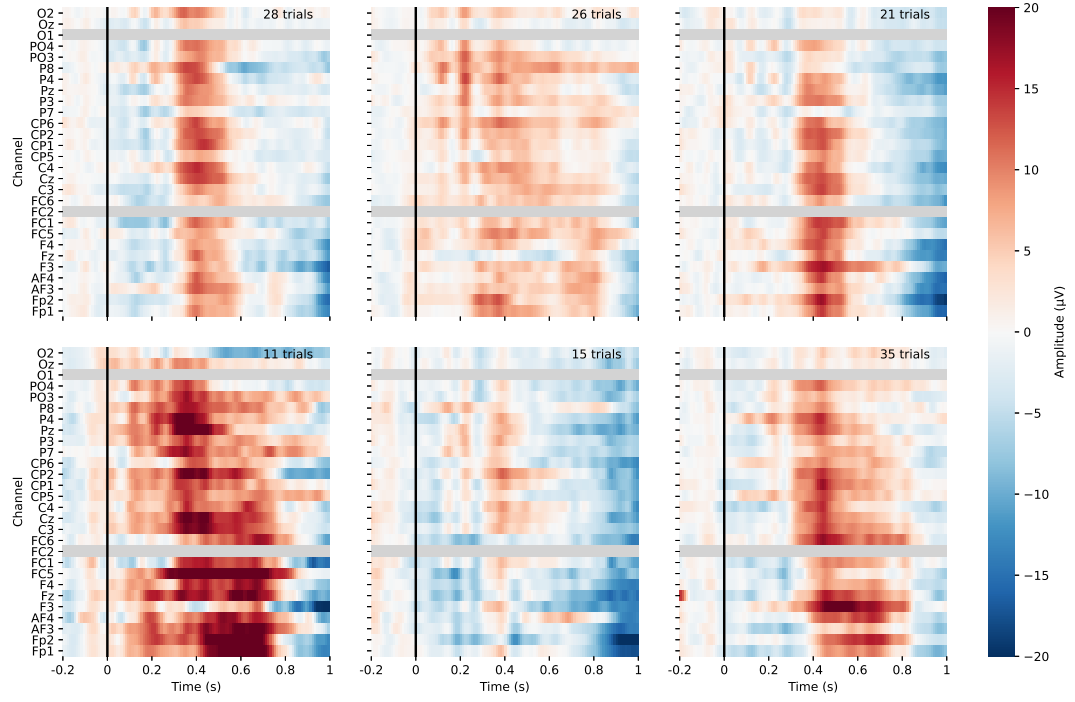


Figure B.5: Average of target trials per cluster for *Subject 3* processed with mcBF1 and clustered using $k = 6$.

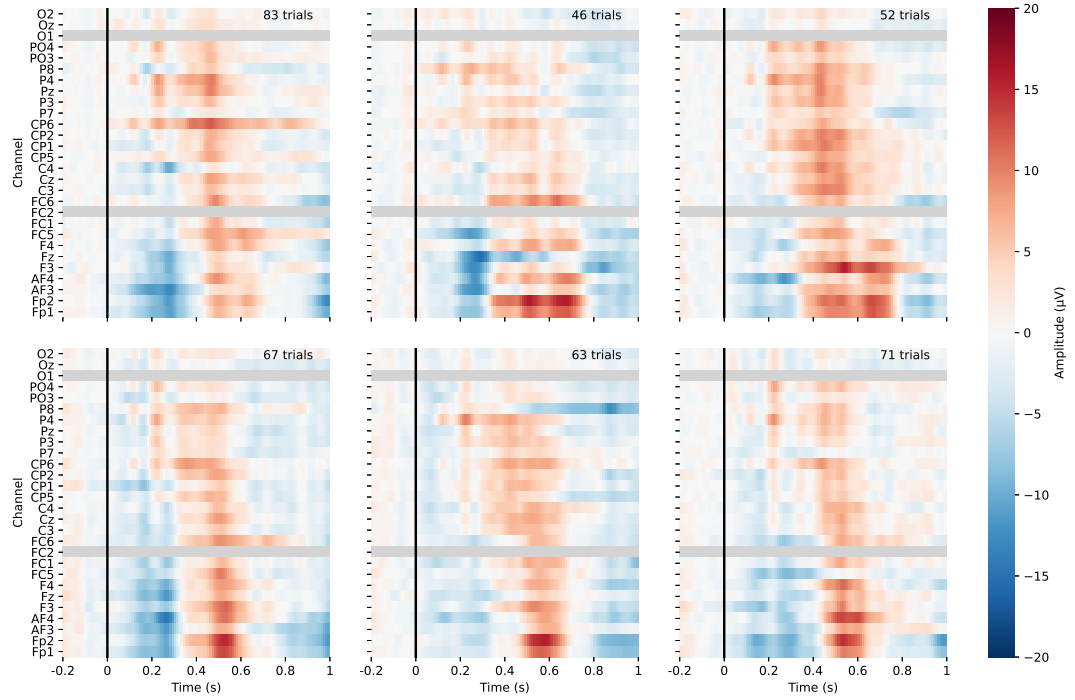


Figure B.6: Average of non-target trials per cluster for *Subject 3* processed with mcBF1 and clustered using $k = 6$.

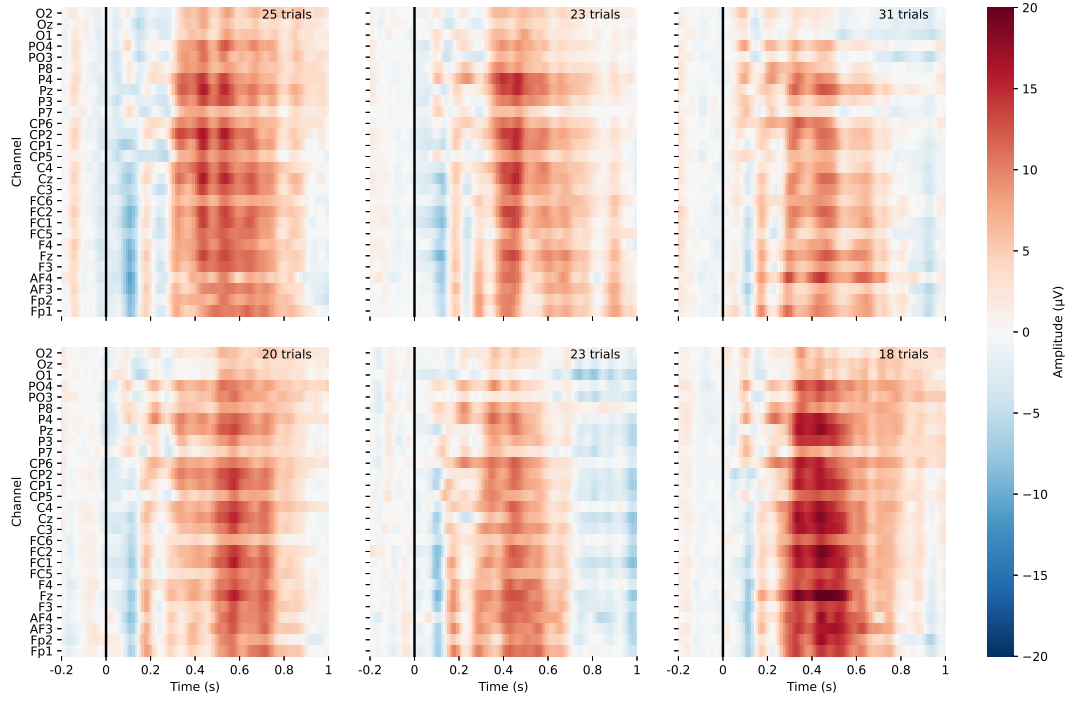


Figure B.7: Average of target trials per cluster for *Subject 4* processed with mcBF1 and clustered using $k = 6$.

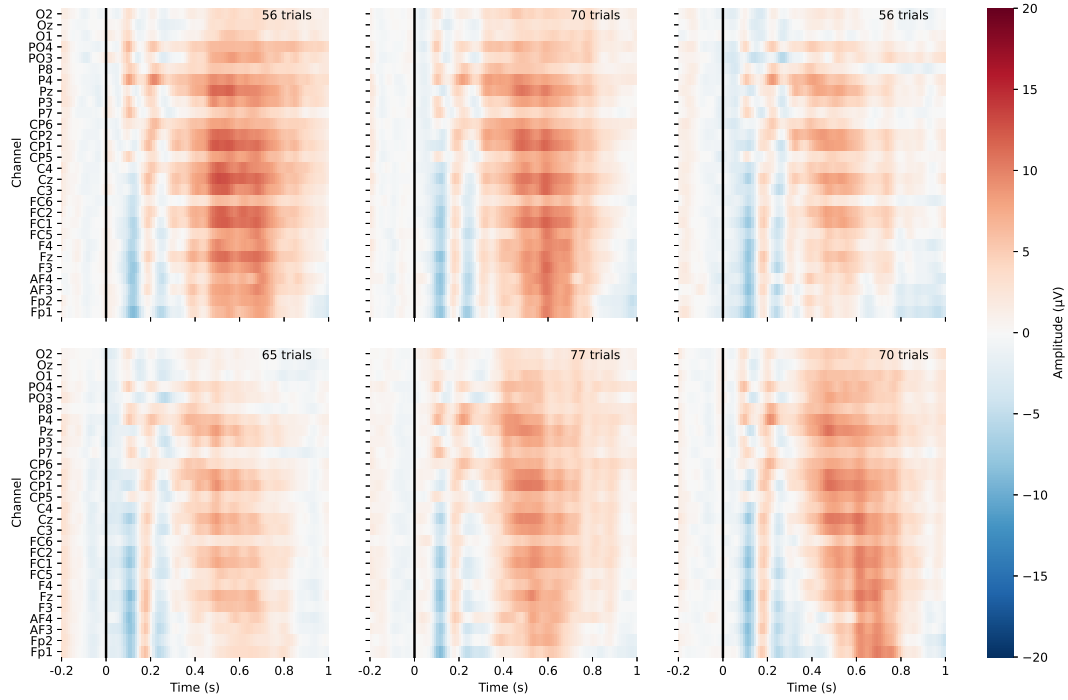


Figure B.8: Average of non-target trials per cluster for *Subject 4* processed with mcBF1 and clustered using $k = 6$.

B. SYMBOL REPRESENTATIVES

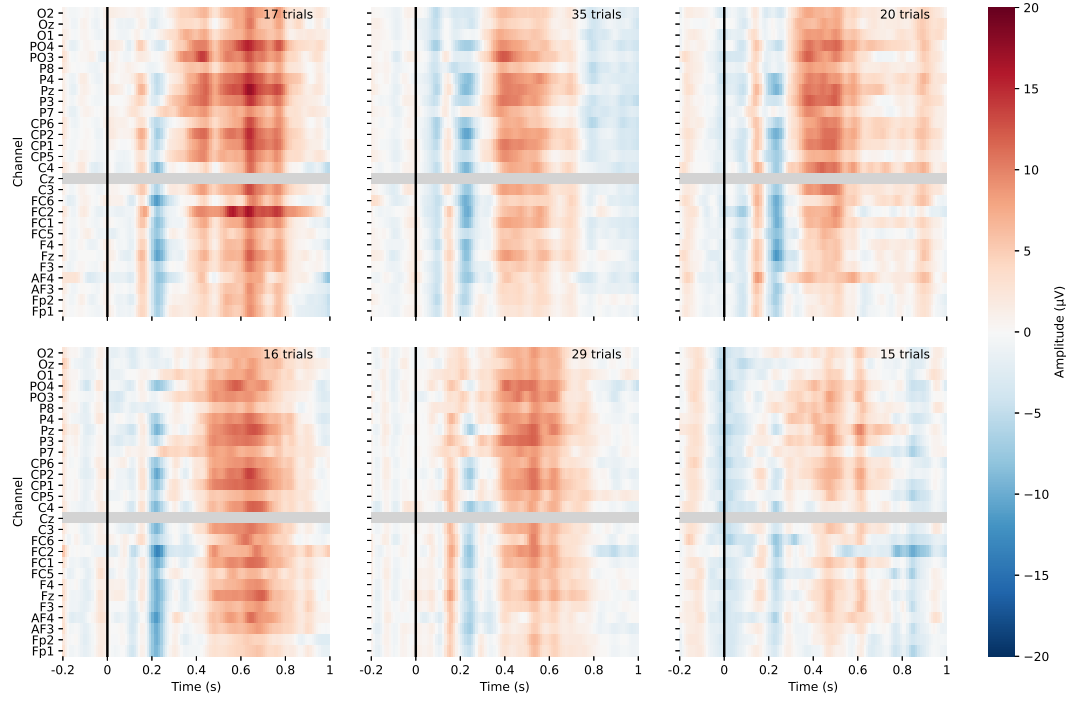


Figure B.9: Average of target trials per cluster for *Subject 1* processed with mcBF2 and clustered using $k = 6$.

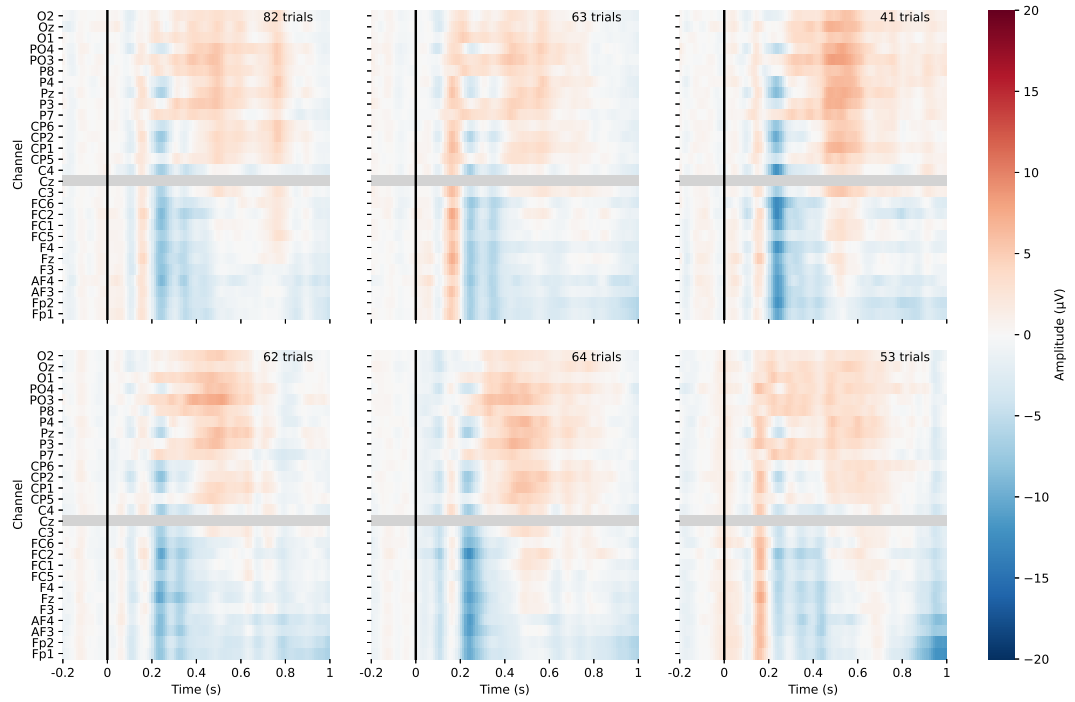


Figure B.10: Average of non-target trials per cluster for *Subject 1* processed with mcBF2 and clustered using $k = 6$.

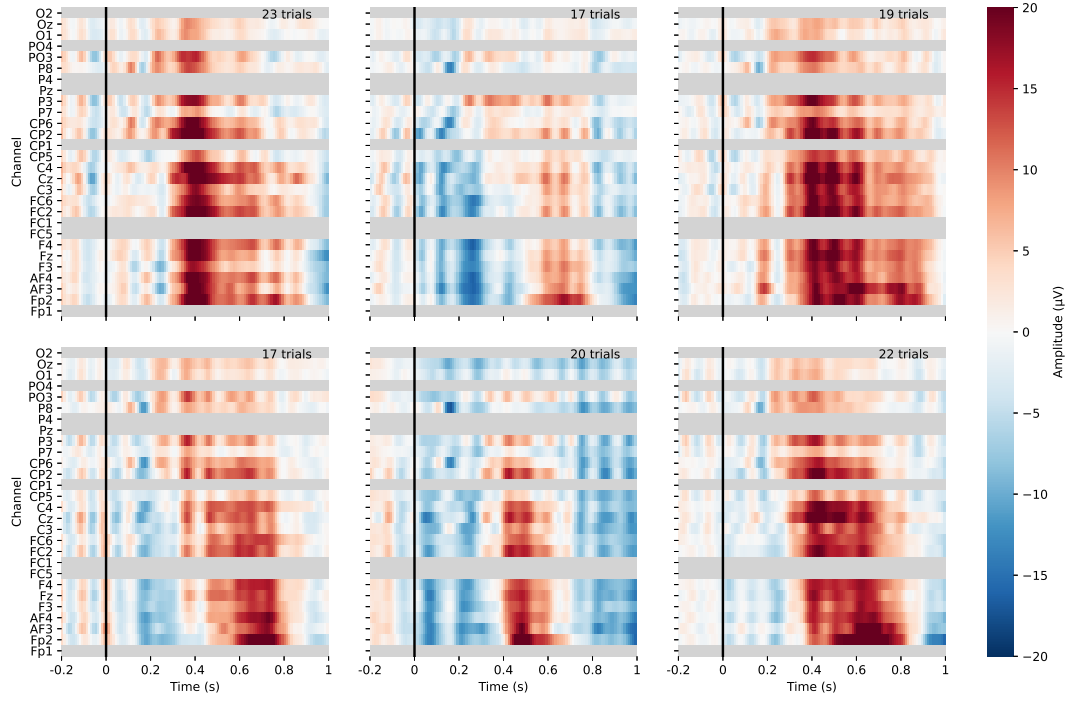


Figure B.11: Average of target trials per cluster for *Subject 2* processed with mcBF2 and clustered using $k = 6$.

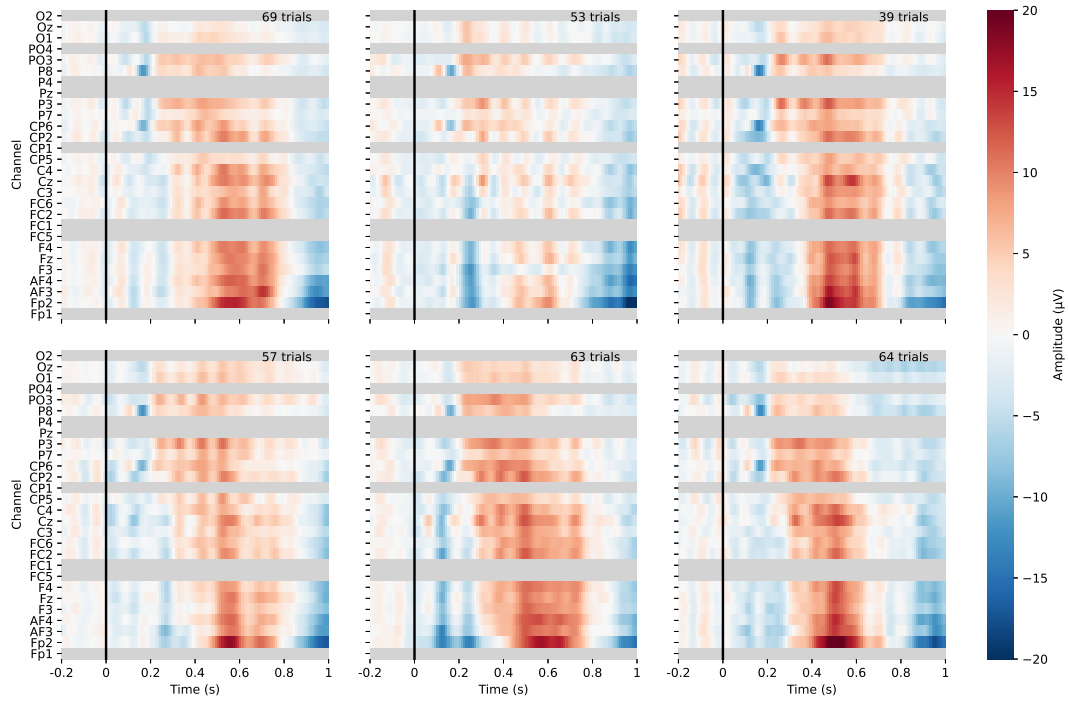


Figure B.12: Average of non-target trials per cluster for *Subject 2* processed with mcBF2 and clustered using $k = 6$.

B. SYMBOL REPRESENTATIVES

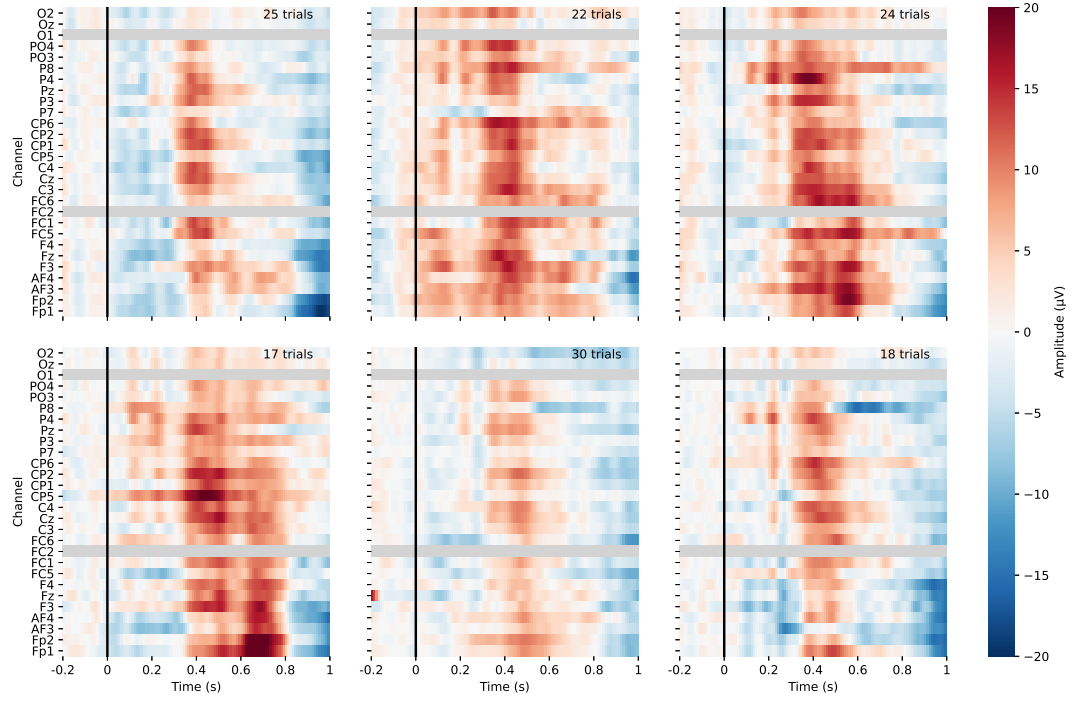


Figure B.13: Average of target trials per cluster for *Subject 3* processed with mcBF2 and clustered using $k = 6$.

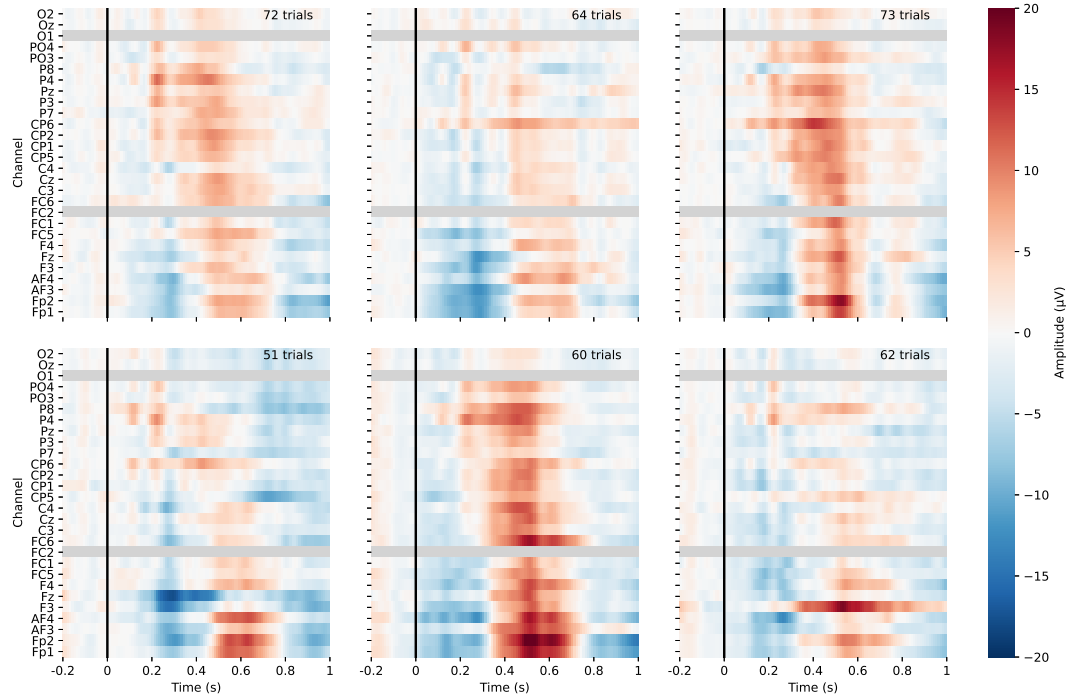


Figure B.14: Average of non-target trials per cluster for *Subject 3* processed with mcBF2 and clustered using $k = 6$.

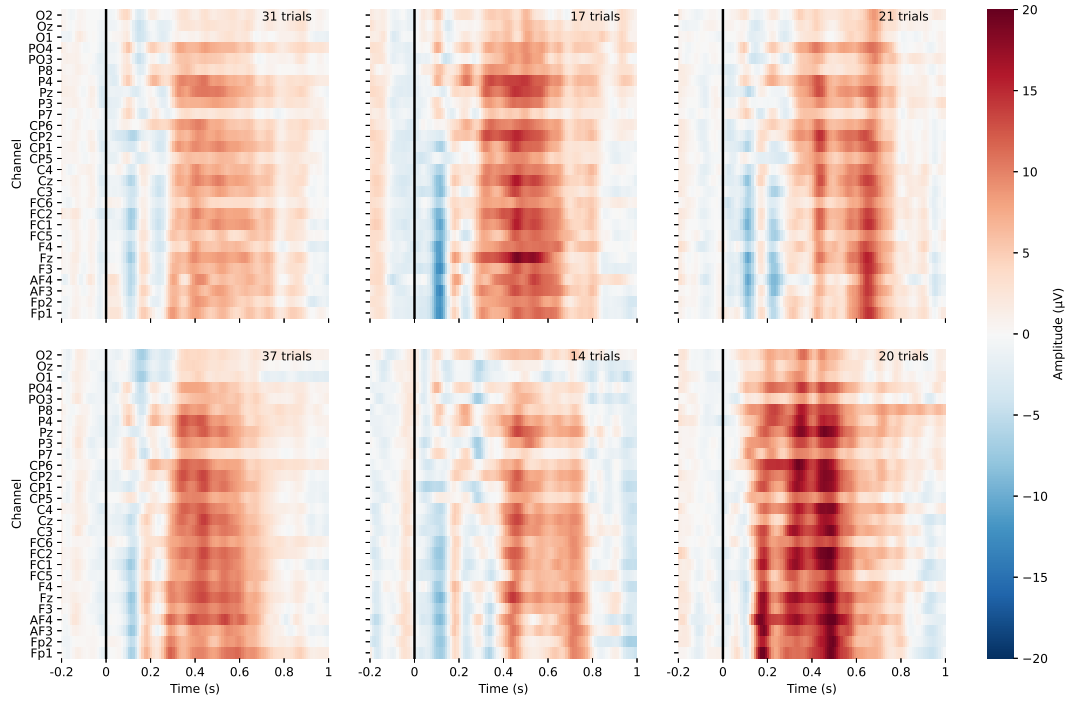


Figure B.15: Average of target trials per cluster for *Subject 4* processed with mcBF2 and clustered using $k = 6$.

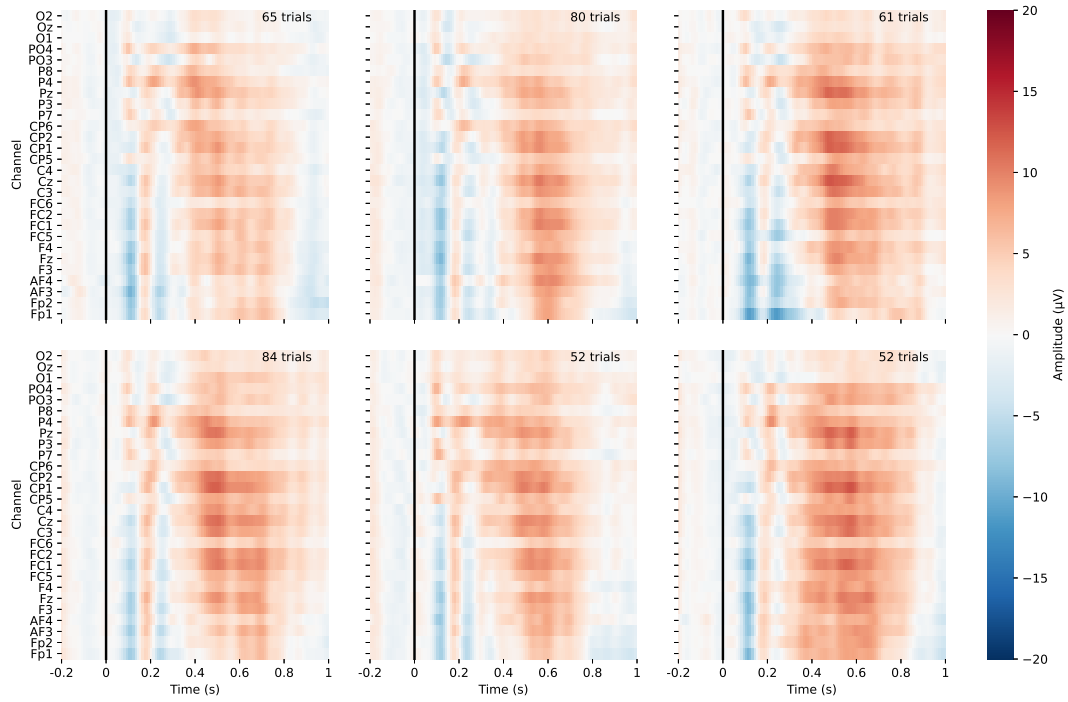


Figure B.16: Average of non-target trials per cluster for *Subject 4* processed with mcBF2 and clustered using $k = 6$.

Bibliography

- [1] V. Abootalebi, M. H. Moradi, and M. A. Khalilzadeh. A new approach for eeg feature extraction in p300-based lie detection. *Computer Methods and Programs in Biomedicine*, 94(1):48 – 57, 2009.
- [2] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. In *Proceedings of the 1998 ACM SIGMOD international conference on Management of data*, pages 94–105, 1998.
- [3] Q. Barthélemy, C. Gouy-Pailler, Y. Isaac, A. Souloumiac, A. Larue, and J. I. Mars. Multivariate temporal dictionary learning for eeg. *Journal of neuroscience methods*, 215(1):19–28, 2013.
- [4] S. Bentin, G. McCarthy, and C. C. Wood. Event-related potentials, lexical decision and semantic priming. *Electroencephalography and clinical Neurophysiology*, 60(4):343–355, 1985.
- [5] L. Bherer, A. F. Kramer, M. S. Peterson, S. Colcombe, K. Erickson, and E. Becic. Training effects on dual-task performance: are there age-related differences in plasticity of attentional control? *Psychology and aging*, 20(4):695, 2005.
- [6] B. Blankertz, S. Lemm, M. Treder, S. Haufe, and K.-R. Müller. Single-trial analysis and classification of erp components — a tutorial. *NeuroImage*, 56(2):814 – 825, 2011. Multivariate Decoding and Brain Reading.
- [7] G. Bodenstern and H. M. Praetorius. Feature extraction from the electroencephalogram by adaptive segmentation. *Proceedings of the IEEE*, 65(5):642–652, 1977.
- [8] T. Caliński and J. Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27, 1974.
- [9] L. Cao, W. Tao, S. An, J. Jin, Y. Yan, X. Liu, W. Ge, A. Sah, L. Battle, J. Sun, et al. Smile: A system to support machine learning on eeg data at scale. *Proceedings of the VLDB Endowment*, 12(12), 2019.
- [10] H. Cecotti and A. Graser. Convolutional neural networks for p300 detection with application to brain-computer interfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(3):433–445, 2011.

- [11] A. Combaz, C. Chatelle, A. Robben, G. Vanhoof, A. Goeleven, V. Thijs, M. M. Van Hulle, and S. Laureys. A comparison of two spelling brain-computer interfaces based on visual p3 and ssvep in locked-in syndrome. *PloS one*, 8(9):e73691, 2013.
- [12] J. F. Connolly and N. A. Phillips. Event-related potential components reflect phonological and semantic processing of the terminal word of spoken sentences. *Journal of cognitive neuroscience*, 6(3):256–266, 1994.
- [13] R. J. Croft and R. J. Barry. Removal of ocular artifact from the eeg: a review. *Neurophysiologie Clinique/Clinical Neurophysiology*, 30(1):5–19, 2000.
- [14] A. Delorme and S. Makeig. Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis. *Journal of neuroscience methods*, 134(1):9–21, 2004.
- [15] D. Durand. A novel approach based on the push down automata (pad) for the automated detection and a classification of waveforms in eeg, especially for spike and wave discharges (swd). 2017.
- [16] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996.
- [17] L. A. Farwell and E. Donchin. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and clinical Neurophysiology*, 70(6):510–523, 1988.
- [18] J. M. Ford, W. T. Roth, R. C. Mohs, W. F. Hopkins III, and B. S. Kopell. Event-related potentials recorded from young and old adults during a memory retrieval task. *Electroencephalography and clinical Neurophysiology*, 47(4):450–459, 1979.
- [19] A. B. Geva and D. H. Kerem. Forecasting generalized epileptic seizures from the eeg signal by wavelet analysis and dynamic unsupervised fuzzy clustering. *IEEE Transactions on Biomedical Engineering*, 45(10):1205–1216, Oct 1998.
- [20] A. S. Gevins, S. L. Bressler, B. A. Cutillo, J. Illes, J. C. Miller, J. Stern, and H. R. Jex. Effects of prolonged mental work on functional brain topography. *Electroencephalography and clinical neurophysiology*, 76(4):339–350, 1990.
- [21] S. Goil, H. Nagesh, and A. Choudhary. Mafia: Efficient and scalable subspace clustering for very large data sets. In *Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, volume 443, page 452. ACM, 1999.
- [22] F. Goksu, N. F. Ince, V. A. Tadipatri, and A. H. Tewfik. Classification of eeg with structural feature dictionaries in a brain computer interface. In *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 1001–1004. IEEE, 2008.

-
- [23] M. Grosse-Wentrup, C. Liefhold, K. Gramann, and M. Buss. Beamforming in noninvasive brain–computer interfaces. *IEEE Transactions on Biomedical Engineering*, 56(4):1209–1219, April 2009.
 - [24] R. I. Herning, R. T. Jones, and J. S. Hunt. Speech event related potentials reflect linguistic content and processing level. *Brain and language*, 30(1):116–129, 1987.
 - [25] P. J. Holcomb and W. B. Mcpherson. Event-related brain potentials reflect semantic priming in an object decision task. *Brain and cognition*, 24(2):259–276, 1994.
 - [26] J. Jing, E. d’Angremont, S. Zafar, E. S. Rosenthal, M. Tabaeizadeh, S. Ebrahim, J. Dauwels, and M. B. Westover. Rapid annotation of seizures and interictal-ictal continuum eeg patterns. In *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 3394–3397. IEEE, 2018.
 - [27] R. Johnson. A triarchic model of p300 amplitude. *Psychophysiology*, 1986.
 - [28] R. Johnson Jr. On the neural generators of the p300 component of the event-related potential. *Psychophysiology*, 30(1):90–97, 1993.
 - [29] C. Kamath. Symbolic time-series analysis to unravel gait dynamics in aging and disease. *Science Postprint*, 1(2):e00057, 2016.
 - [30] L. Kaufman and P. J. Rousseeuw. *Finding groups in data: an introduction to cluster analysis*, volume 344. John Wiley & Sons, 2009.
 - [31] W. K. Kirchner. Age differences in short-term retention of rapidly changing information. *Journal of experimental psychology*, 55(4):352, 1958.
 - [32] J. Kounios and P. J. Holcomb. Concreteness effects in semantic processing: Erp evidence supporting dual-coding theory. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 20(4):804, 1994.
 - [33] A. F. Kramer and D. L. Strayer. Assessing the development of automatic processing: an application of dual-task and event-related brain potential methodologies. *Biological psychology*, 26(1-3):231–267, 1988.
 - [34] A. F. Kramer, D. L. Strayer, and J. Buckley. Task versus component consistency in the development of automatic processing: A psychophysiological assessment. *Psychophysiology*, 28(4):425–437, 1991.
 - [35] D. J. Krusienski, E. W. Sellers, F. Cabestaing, S. Bayoudh, D. J. McFarland, T. M. Vaughan, and J. R. Wolpaw. A comparison of classification techniques for the p300 speller. *Journal of neural engineering*, 3(4):299, 2006.
 - [36] R. Li, J. C. Principe, M. Bradley, and V. Ferrari. A spatiotemporal filtering methodology for single-trial erp component estimation. *IEEE Transactions on Biomedical Engineering*, 56(1):83–92, 2009.

- [37] R. Lin, R.-G. Lee, C.-L. Tseng, Y.-F. Wu, and J.-A. Jiang. Design and implementation of wireless multi-channel eeg recording system and study of eeg clustering method. *Biomedical Engineering: Applications, Basis and Communications*, 18(06):276–283, 2006.
- [38] S. Luck. An introduction to the event-related potential technique. 2005. *Massachusetts: The MIT Press Google Scholar*, 2005.
- [39] J. F. Mackworth. Paced memorizing in a continuous task. *Journal of experimental psychology*, 58(3):206, 1959.
- [40] R. Manor and A. B. Geva. Convolutional neural network for multi-category rapid serial visual presentation bci. *Frontiers in Computational Neuroscience*, 9:146, 2015.
- [41] N. V. Manyakov, N. Chumerin, A. Combaz, and M. Van Hulle. Comparison of classification methods for p300 brain-computer interface on disabled subjects. *Computational Intelligence and Neuroscience*, 2011:1–12, 2011.
- [42] M. Miao, A. Wang, and F. Liu. A spatial-frequency-temporal optimized feature sparse representation-based classification method for motor imagery eeg pattern recognition. *Medical & biological engineering & computing*, 55(9):1589–1603, 2017.
- [43] K.-R. Muller, C. W. Anderson, and G. E. Birch. Linear and nonlinear methods for brain-computer interfaces. *IEEE transactions on neural systems and rehabilitation engineering*, 11(2):165–169, 2003.
- [44] U. Orhan, M. Hekim, and M. Ozer. Eeg signals classification using the k-means clustering and a multilayer perceptron neural network model. *Expert Systems with Applications*, 38(10):13475 – 13481, 2011.
- [45] R. Parasuraman, R. Johnson, and J. W. Rohrbaugh. *Event-related brain potentials: Basic issues and applications*. Oxford University Press, 1990.
- [46] L. Paternoster, M. Vallverdú, U. Melia, F. Claria, A. Voss, and P. Caminal. Analysis of epileptic eeg signals in children by symbolic dynamics. In *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 4362–4365. IEEE, 2013.
- [47] V. Pergher, B. Wittevrongel, J. Tournoy, B. Schoenmakers, J. Arsenault, and M. Van Hulle. N-back training and transfer effects in healthy young and older subjects gauged using eeg: A preliminary study. *International journal on advances in life sciences*, 10(1&2):54–54, 2018.
- [48] V. Pergher, B. Wittevrongel, J. Tournoy, B. Schoenmakers, and M. M. Van Hulle. N-back training and transfer effects revealed by behavioral responses and eeg. *Brain and behavior*, 8(11):e01136, 2018.

-
- [49] C. Robinson. *Dynamical systems: stability, symbolic dynamics, and chaos*. CRC press, 1998.
 - [50] D. L. Strayer, C. D. Wickens, and R. Braune. Adult age differences in the speed and capacity of information processing: Ii. an electrophysiological approach. *Psychology and aging*, 2(2):99, 1987.
 - [51] K.-m. Su, W. D. Hairston, and K. Robbins. Eeg-annotate: automated identification and labeling of events in continuous signals with applications to eeg. *Journal of neuroscience methods*, 293:359–374, 2018.
 - [52] M. S. Treder and B. Blankertz. (c) overt attention and visual speller design in an erp-based brain-computer interface. *Behavioral and brain functions*, 6(1):1–13, 2010.
 - [53] N. Tupaika, M. Vallverdu, M. Jospin, E. W. Jensen, M. M. Struys, H. E. Vereecke, A. Voss, and P. Caminal. Assessment of the depth of anesthesia based on symbolic dynamics of the eeg. In *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pages 5971–5974. IEEE, 2010.
 - [54] B. D. Van Veen and K. M. Buckley. Beamforming: a versatile approach to spatial filtering. *IEEE ASSP Magazine*, 5(2):4–24, April 1988.
 - [55] B. D. Van Veen, W. Van Drongelen, M. Yuchtman, and A. Suzuki. Localization of brain electrical activity via linearly constrained minimum variance spatial filtering. *IEEE Transactions on biomedical engineering*, 44(9):867–880, 1997.
 - [56] M. Van Vliet, N. Chumerin, S. De Deyne, J. R. Wiersema, W. Fias, G. Storms, and M. M. Van Hulle. Single-trial erp component analysis using a spatiotemporal lcmv beamformer. *IEEE Transactions on Biomedical Engineering*, 63(1):55–66, 2015.
 - [57] A. Voss, J. Kurths, H. Kleiner, A. Witt, N. Wessel, P. Saperin, K. Osterziel, R. Schurath, and R. Dietz. The application of methods of non-linear dynamics for the improved and predictive recognition of patients threatened by sudden cardiac death. *Cardiovascular research*, 31(3):419–433, 1996.
 - [58] S. Watter, G. M. Geffen, and L. B. Geffen. The n-back as a dual-task: P300 morphology under divided attention. *Psychophysiology*, 38(6):998–1003, 2001.
 - [59] B. Wittevrongel and M. M. Van Hulle. Faster p300 classifier training using spatiotemporal beamforming. *International journal of neural systems*, 26(03):1650014, 2016.
 - [60] B. Wittevrongel and M. M. Van Hulle. Frequency-and phase encoded ssvep using spatiotemporal beamforming. *PloS one*, 11(8):e0159988, 2016.