UNIVERSIDAD
DE GRANADA

TESIS DOCTORAL

PROGRAMA DE DOCTORADO EN TECNOLOGÍAS DE LA
INFORMACIÓN Y LA COMUNICACIÓN

Procesado avanzado de señales de electroencefalografía y resonancia magnética en Neurociencia Cognitiva

DOCTORANDO

David López García

DIRECTORES

María Ruz Cámara, Juan Manuel Górriz Sáez

Enero, 2023

Editor: Universidad de Granada. Tesis Doctorales
Autor: David López García
ISBN: 978-84-1117-740-5
URI: https://hdl.handle.net/10481/80692

Agradecimientos

Quisiera dedicar unas palabras de agradecimiento a todas las personas que han contribuido a que esta tesis llegue a su fin. De manera directa e indirecta, durante todos estos años he recibido ayuda, opinión experta y consejo, además de mucho cariño y ánimo de numerosas personas.

Muchas gracias también a toda mi familia, por confiar en mí y en mi trabajo.

Finalmente, agradecer especialmente a la Dra. María Ruz Cámara y al Dr. Juan Manuel Górriz Sáez, mis directores de tesis, por su confianza, apoyo, ánimos y enseñanzas durante todos estos años.

Gracias a todos.

Table of contents

List of tables

List of Figures

Acronyms

AC	*Alternate Current*
ACC	*Accuracy*
ADHD	*Attention Deficit Hyperactivity Disorder*
AI	*Artificial Intelligence*
AUC	*Area Under the Curve*
BCI	*Brain-Computer Interface*
BIDS	*Brain Imaging Data Structure*
BOLD	*Blood-oxygen-level-dependent imaging*
CRP	*Corneoretinal Potential*
CSF	*Cerebrospinal Fluid*
CTI	*Cue-Target Interval*
CV	*Cross-Validation*
DA	*Discriminat Analysis*
DST	*Demand-Selection Task*
ECG	*Electrocardiogram*
EEG	*Electroencephalogram*
EMG	*Electromyogram*
EOG	*Electrooculogram*
EPI	*Echo Planar Imaging*

EPSP	*Excitatory Postsynaptic Potential*
ERP	*Event-Related Potential*
FDR	*False Discovery Rate*
FFT	*Fast Fourier Transformation*
FIR	*Finite Impulse Response*
FMRI	*Functional Magnetic Resonance Imaging*
FN	*False Negative*
FP	*False Positive*
FPR	*False Positive Rate*
FRF	*Frequency Response Function*
FWHM	*Full Width at Half Maximum*
GLM	*General Lineal Model*
GM	*Grey Matter*
GUI	*Graphic User Interface*
HRF	*Hemodynamic Response Function*
HT	*Hilbert's Transformation*
ICA	*Independent Component Analysis*
IIR	*Infinite Impulse Response*
IPSP	*Inhibitory Postsynaptic Potential*
LCD	*Liquid-crystal Display*
LDA	*Linear Discriminant Analysis*

LDC	*Linear Discriminant Contrast*
LFP	*Local Field Potential*
LR	*Logistic Regression*
LSVM	*Linear Support-Vector Machine*
MEG	*Magnetoencephalogram*
ML	*Machine Learning*
MR	*Magnetic Resonance*
MRI	*Magnetic Resonance Imaging*
MVCC	*Multivariate Cross-Classification*
MVPA	*Multivariate Pattern Analysis*
NB	*Naïve Bayes*
NMR	*Nuclear Magnetic Resonance*
NT	*Neurotransmitter*
PCA	*Principal Component Analysis*
PD	*Proton Density*
PLS	*Partial Least Squares*
PSD	*Power Spectral Density*
QDA	*Quadratic Discriminant Analysis*
RAM	*Random Access Memory*
RDM	*Representational Dissimilarity Matrix*
RF	*Radiofrequency*

ROC	*Receiver Operating Characteristic*
RSA	*Representational Similarity Analysis*
RT	*Reaction Time*
SAM	*Statistical Agnostic Mapping*
STD	*Standard Deviation*
SEM	*Standard Error of the Mean*
SNR	*Signal-to-Noise Ratio*
SPM	*Statistical Parametric Mapping*
SPSS	*Statistical Package for the Social Sciences*
SVM	*Support Vector Machine*
TDT	*The Decoding Toolbox*
TE	*Time to Echo*
TG	*Temporal Generalization*
TGM	*Temporal Generalization Matrix*
TN	*True Negatives*
TP	*True Positives*
TPR	*True Positive Rate*
TR	*Repetition Time*
WM	*White Matter*

Section I.

Introduction

Chapter 1. Introduction

1.1. Motivation

The study of brain cognitive function has experienced remarkable progress in the past decades (Posner & Raichle, 1994). The incorporation of modern non-invasive neuroimaging modalities to the study of cognition is a fundamental catalyst of these major advances. Two of the most employed techniques in the study of brain cognitive function are the high-density Electroencephalography (EEG), which measures the macroscopic bioelectrical activity in the surface layer of the brain (Luck, 2014), and the functional Magnetic Resonance Imaging, a technique that allows to detect changes in brain activations associated with blood flow (Huettel et al., 2004). These bioelectrical or hemodynamic changes reflect brain activity associated with different cognitive tasks, such as perception, memory, decision-making, control, etc. Thus, Cognitive Neuroscience is the scientific discipline conducting the study of how brain structures or its dynamics are related to human cognitive function.

Cognitive neuroscientists employ different neuroimaging techniques to record the participants' brain activity while they perform simple tasks, which are previously designed to measure a specific cognitive function (e.g., cognitive control, memory, visual perception, etc.). Then, the recorded signal is preprocessed, cleaned and analyzed with the purpose of understanding the basis of the cognitive process of interest. However, establishing a direct relation between the experimental manipulation and the recorded neuroimaging signal is not a straight-forward exercise, and it usually relies on different aspects, such as an adequate experimental design or the use of powerful analysis techniques.

Modern neuroimaging techniques provide a vast amount of information, which fluctuates in time and is always contaminated with noise coming from sources of very different nature. Additionally, these dynamic fluctuations correlate imperfectly with the cognitive functions of interest, since the recorded signal only reflects a portion of the total neural processing. However, this intrinsic richness of neuroimaging signals fades when employing classic analytical approaches to extract information from them. Although classical univariate approaches have been broadly employed in the field and have allowed a myriad of scientific and medical breakthroughs, they present a potential limitation trying to understand complex and dynamic systems as such the human brain. Historically, these classical analytic approaches mostly focus on the study of amplitudes, power and time delays in different frequency bands and electrodes of the average of several evoked EEG recordings, commonly called Event-Related Potentials (ERPs) (Luck, 2014). Meanwhile, classical fMRI studies rely on univariate analyses of the BOLD (Blood-oxygen-level-dependent) response (Huettel et al., 2004), the gold standard in the fMRI literature for years.

More recently, the constant progress of science and technology has allowed researchers and engineers to develop and apply more advanced signal processing techniques to multivariate data in the Cognitive Neuroscience field, such as Machine Learning-based algorithms (ML). This trend started with studies by Haxby and Norman (Haxby, 2001, 2012; Norman et al., 2006) and other significant contributions (T. Davis & Poldrack, 2013; Haynes & Rees, 2006; Kriegeskorte et al., 2006; Lemm et al., 2011; Mur et al., 2009; Pereira et al., 2009), opening novel avenues of research on brain function and pattern recognition. Although multivariate approaches are already broadly extended in fMRI ambits, the poor temporal resolution of the fMRI signal prevents an accurate study of how cognitive processes unfold in time. Thus, its application in electro and magnetoencephalography studies is growing exponentially, offering an exceptional explicative potential compared to classical univariate approaches.

Despite its intrinsic potential, the human brain is still a very complex system, and cognitive functions encompass a welter of neural activity with specific temporal and spatial dynamics: oscillations, transient signals, quick transfers of information between different areas and so forth. It seems obvious, then, that for a better understanding and characterization of cognitive processes, neural activity should be identified simultaneously in time and in space. However, due to the nature of the signal of current non-invasive neuroimaging modalities, neuroscientists are only allowed to characterize cognitive processes either in a time or space-resolved way, but not in both. While M/EEG signals provide exceptional temporal resolution but lack spatial resolution, functional Magnetic Resonance Imaging localizes brain activity changes at millimetric levels but blurred in time. Overcoming this dichotomy is one of the major current challenges in the field of Neuroscience. Some trends bet on multimodal data fusion as a promising solution (Radoslaw M. Cichy & Oliva, 2020). Fusing methods combine information of non-concurrent recordings from different neuroimaging modalities, preserving their individual strengths while overcoming their weaknesses.

The present PhD thesis has been developed within the previously described scope. These new advanced analytic methodologies are not simply explored but are also applied into the study of the human brain function, more specifically in the cognitive control framework. As a result, a complete software package has been developed, The MVPAlab Toolbox, a MATLAB-based, flexible but powerful tool including different statistical and machine learning routines to analyze multivariate neuroimaging data.

1.2. Aims and objectives

The present PhD thesis explores several multivariate pattern recognition methodologies that help to elucidate how the human brain implements cognitive control mechanisms. We aim at collecting both electroencephalography and functional magnetic resonance data from human volunteers performing specifically designed experiments to study such mechanisms.

More specifically, the main objectives of this thesis are listed below:

1. The implementation of advanced Multivariate Pattern Analysis (MVPA) techniques in electroencephalography data for a better understanding of the neural mechanisms involved in effort avoidance. These methodologies allow to study how different brain patterns differ in time and across different frequency bands and electrodes.
2. The implementation of new analytical frameworks based on pattern similarity that allow to fuse information from different non-concurrent neuroimaging modalities, the electroencephalography and the functional magnetic resonance imaging. This approach provides high-resolution information about brain function in time and space.
3. The development of a new analytic software package compiling the implemented multivariate and fusion methodologies. The developed software package should be highly flexible, maintainable, powerful and easy-to-use for a wide range un users, not just for those with advanced coding skills.

In order to achieve the aforementioned goals, we divided our work in three main studies:

1. A first work studying the neural underpinnings of interference-effects in effort avoidance situations. Here we designed a classical Demand-Selection Task (DST) experiment and we collected, cleaned, and preprocessed the EEG data. We finally implemented machine-learning-based analytical approaches in EEG data, such as multivariate pattern analyses. We studied how the analyzed brain patterns differ and generalize across time or in which frequency bands the neural processes of the interference effect relied on.
2. In our second work we present the MVPAlab Toolbox, including an extensive set of computational resources to design, configure and execute the complete pipeline of different pattern recognition analyses for multidimensional M/EEG data. We also provide a visualization software for data representation and an easy-to-use graphic user interface.
3. The third and final work extends the MVPAlab Toolbox capabilities. We recorded the participant's brain activity using non-concurrent EEG and fMRI modalities and, based on the representational similarity of the analyzed signals, we developed multimodal fusion methods. This methodology allowed to fuse data from different neuroimaging modalities in order to characterize cognitive processes in time and in space.

1.3. Structure and content of the thesis

This PhD. thesis is divided into three main sections, each of them containing several chapters organized as follows:

- Section I: Introduction: This first part consists of three chapters: In *Chapter 1* we present the motivation and the main goals of this thesis. *Chapter 2* describes the basic principles of two of the most employed non-invasive neuroimaging techniques in Neuroscience, the Electroencephalography and the functional Magnetic Resonance Imaging. This chapter also includes a detailed description of several preprocessing techniques that are usually performed in most preprocessing pipelines. Finally, in *Chapter 3* we examine the state of the art of multivariate pattern analyses applied to neuroimaging data, including classification analyses (decoding), the Representational Similarity Analysis, multimodal fusion methods and how to draw statistical inferences at a group level in our data.

- Section II: In this second part we present the contributions of this thesis. In *Chapter 4* we present a complete study including a set of multivariate, and machine-learning based routines for computing time-resolved analyses in EEG data. *Chapter 5* describes The MVPAlab Toolbox, a MATLAB based application encapsulating not only the previously developed functionalities but a complete integrated environment for computing multivariate analyses in EEG data. Finally, in *Chapter 6* we describe how we significatively extended the capabilities of The MVPAlab Toolbox, including the Representational Similarity Analysis not only for EEG but for fMRI dataset. In addition, we developed multimodal fusion analysis, an analytic framework that allows to characterize cognitive processes in time and in space.
- Section III: The last part of this thesis includes a general discussion of the obtained results and conclude discussing the possible implications in the Neuroscience field of the developed work.

Chapter 2. Non-invasive neuroimaging techniques

2.1. The electroencephalography signal

The electroencephalography is a non-invasive electrophysiological technique that records the electrical activity arising from the brain. The first EEG recording in humans was performed by the German psychiatrist Hans Berger in 1924 (R. Jung & Berger, 1979). Due to its outstanding temporal resolution, the EEG technique has been broadly employed in last decades to evaluate the dynamics of cerebral functioning. The EEG allows physicians to evaluate sleep quality, to detect sleep disorders or to evaluate epilepsy and other seizure disorders. The EEG has also been adopted for other clinical indications: from monitoring the depth of anesthesia during surgical procedures to augmenting the diagnosis of certain psychiatric conditions, such as the Attention Deficit Hyperactivity Disorder (ADHD). In addition to its clinical value, the EEG technique has also been employed for decades in research applications. Neuroscientists have used the averaged EEG waveform related to certain stimulus (also known as Event-Related Potential, ERP) to investigate the neural correlates of visual, auditory, somatosensory, and higher cognitive functions, such as emotion, language, attention, and learning. Another important application of the EEG is the development of Brain-Computer Interfaces (BCI). These technologies transform brain activity into action by connecting the brain to different external devices, such as computers or robotic arms, which allows, for example, to restore or enhance lost capabilities of paralyzed patients.

The EEG signal is a direct measure of the electrical activity of the brain. It is versatile and low in cost, with an exquisite temporal resolution. These features make EEG one of the most important and widely employed neuroimaging techniques in the past decades. Throughout this chapter we describe the basis of the electroencephalography signal, how it is generated, the origin and function of brain oscillations and several aspects related to the recording equipment, such as the types of electrodes and their placement over the scalp. In addition, we meticulously describe the preprocessing stage of the EEG signal, which plays a pivotal role in the analytical process since it attenuates noise from the recorded data to get closer to the true neural signals.

2.1.1. Basic principles of the electroencephalography signal

Pyramidal cells

The electrical activity recorded by non-invasive electroencephalography techniques is mostly generated by a group of cortical cells known as pyramidal neurons (Elston, 2003; García-López et al., 2006). These multipolar cells were first discovered by Santiago Ramón y Cajal, a Spanish neuroscientist who won the Nobel Prize in Medicine in 1906 and who is considered the pioneer of modern Neuroscience (Ramón y Cajal, 1988). Pyramidal neurons are located in several areas of the brain, including the cerebral cortex, the amygdala and the hippocampus, being one of the larger neurons in the human brain. Its name comes from the conical shape of its cell body, one key structural characteristic of pyramidal cells. Its soma (cell body) is, in average 20μm in length and its multiple basal dendrites are usually several hundred micrometers each, branching abundantly. The apical dendrite is a single and thick branch that extends to the cortical surface. They only have one single axon which is even longer, reaching several centimeters in total (Megías et al., 2001).

Neurophysiological mechanisms of the electroencephalography

Nerve cells (i.e., pyramidal neurons) communicate via a combination of electrical and chemical transmission mechanisms that occur at nanometric gaps, called synapses. Here specialized parts of the two cells (i.e., the presynaptic and postsynaptic neurons) release and receive a chemical substance known as neurotransmitter (NT). The neurotransmitter molecules bind to the receptor proteins and alter postsynaptic neuronal function. The neurotransmitter is released into the synaptic cleft when an action potential (rises and falls in the membrane potential of a specific location of the cell) arrives at the presynaptic terminal, which then diffuses to reach membrane receptors on the postsynaptic terminal and trigger an Excitatory or Inhibitory Postsynaptic Potential (EPSP and IPSP respectively). The electrical activity occurring in the brain is generated by billions of neurons, and it is based on this chemical exchange principle (Herculano-Houzel, 2009). However, neurons rarely act alone, particularly in the neocortex, and the electric potential of a single neuron cannot be recorded by any non-invasive technique. The EEG technique measures the combined and synchronous activity of millions of cerebral cortical neurons with the same spatial orientation near the scalp, where the recording electrodes are placed.

The inhibitory or excitatory postsynaptic potentials generated by groups of pyramidal neurons can be represented as an electrical field (dipole) parallelly aligned with the orientation of those neurons (Figure 1). Thus, positive or negative deflections in the signal are generated by both excitatory and inhibitory afferents. Negative (upward) deflections are due to superficial excitatory or deep inhibitory inputs, whereas positive (downward) deflections represent deep excitatory or superficial inhibitory inputs (Kirschstein & Köhling, 2009).

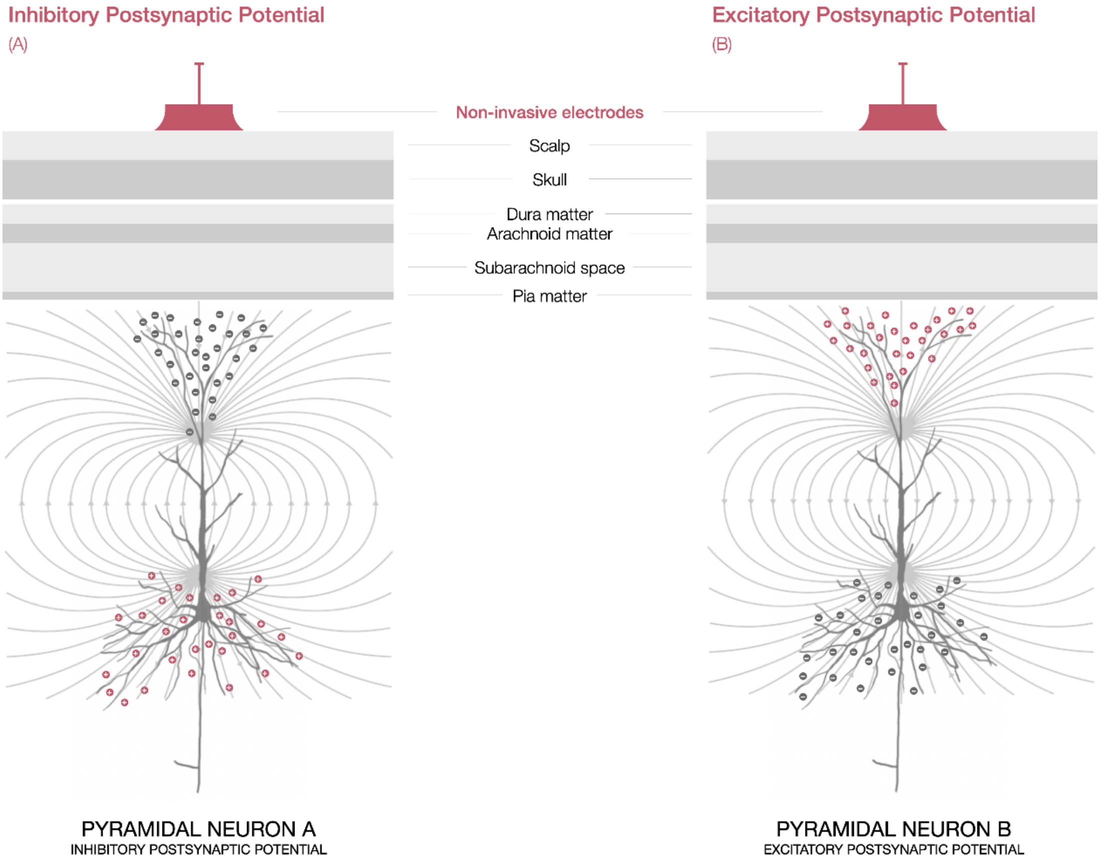

Inhibitory Postsynaptic Potential
(A)
Excitatory Postsynaptic Potential
(B)
Non-invasive electrodes
Scalp
Skull
Dura matter
Arachnoid matter
Subarachnoid space
Pia matter
PYRAMIDAL NEURON A
INHIBITORY POSTSYNAPTIC POTENTIAL
PYRAMIDAL NEURON B
EXCITATORY POSTSYNAPTIC POTENTIAL

Figure 1. Neurophysiological basis of the electroencephalography

The EEG signal represents the electrical activity in the brain generated by the combination of the synchronous activation of groups of pyramidal neurons in the cerebral cortex. Inhibitory and Excitatory postsynaptic potentials (IPSPs, EPSPs) generate electric dipoles perpendicular to the surface of the cortex. Most EEG signals are generated by pyramidal neurons located in layers III and V. These neurons are typically aligned perpendicular to the cortical surface, which yields a dipole layer orthogonal to the surface of the scalp.

In short, the principle that allows researchers and medical doctors to record electrical activity from the human brain using non-invasive electrodes is based on enormous populations of neurons firing in concert, producing waves of electrical impulses powerful enough to be detected by external electrodes. Unfortunately, the EEG signal does not represent the full picture of brain activity. The electrical activity which the scalp electrodes record is very limited as it depends on the orientation of the electric dipoles generated by groups of neurons in the cortical surface. More specifically, perpendicular dipoles to the cortical surface are seen clearly on EEG, but those parallel or tangential to the scalp (neurons located in brain fissures), or those with complex metrics and multiple dipoles, are very difficult to see or missed completely by scalp EEG. Furthermore, electrical activity coming from deep sources is more difficult or impossible to detect due to voltage fields gradients decay with the square of distance. Additionally, as shown in Figure 1 the EEG signal crosses several layers of non-neural tissues with different conduction properties that attenuate the signal before it reaches the scalp electrodes.

Brain oscillations and rhythms

The electrical activity in the brain is strongly oscillatory at a variety of frequencies (Begleiter & Porjesz, 2006) and it represents synchronized activity of different neuron populations. Electric oscillations play an important role in most fundamental aspect of brain function, and they have been extensively studied by hundreds of research groups around the world. Regardless of the species, these oscillations are involved not only in the low-level neurobiological functions but also in many high-level psychological aspects such as perception (E. Başar et al., 2000), memory (Düzel et al., 2010; Hanslmayr et al., 2019; Hanslmayr & Staudigl, 2014), emotion (Schubring & Schupp, 2021; Schutter & Knyazev, 2012), language (Benítez-Burraco & Murphy, 2019; Weiss & Mueller, 2012) and action (De Lange et al., 2008; Tomassini et al., 2017). Additionally, brain oscillations also play an important role in clinical environments for studying Parkinson' Disease (Foffani & Alegre, 2022; Meissner et al., 2018; Solís-Vivanco et al., 2018), Alzheimer' Disease (Chan et al., 2021; Hanson et al., 2020; Jafari et al., 2020; Osipova et al., 2005), neuropsychiatric disease (Erol Başar et al., 2016; Buzsáki & Watson, 2012; Herrmann & Demiralp, 2005), epilepsy (Bragin et al., 2002; Matsumoto et al., 2013; Worrell et al., 2004; Zijlmans et al., 2012), several sleep disorders (Crowley, 2011; Weiner & Dang-Vu, 2016), and more besides.

The importance of oscillations in the correct function of the brain seems obvious, but why is the electrical activity of the brain constantly oscillating? As mentioned in previous paragraphs, excitatory and inhibitory neurons, which are densely interconnected with each other, generate polarizations and de-polarizations in the cortical surface which are sensed by the scalp electrodes. In response to an external stimulus (e.g., the opening the eyes activates neurons in the visual cortex) the excitatory neurons start activating each other which generates a massive increase in excitatory activity. These excitatory neurons are also connected to inhibitory ones, which also cause an increase in inhibition. This increasing inhibition leads to a decay in excitation, and as a result, the inhibition decreases too, which in turn leads back to an increase in excitatory activity, starting the cycle indefinitely. This rhythmic mechanism is one of the most important generators of neural oscillations (Buzsáki & Wang, 2012) but not the only one. There are several other mechanisms which produce brain oscillations which do not require the interaction of excitatory or inhibitory cells (Buzsáki, 2006; Buzsáki et al., 2012; X.-J. Wang, 2010). Not even a complex neuronal system or sophisticated biophysical mechanisms are required for producing oscillations. These can be found on small portions of brain tissue or even in mathematical and computational models of neural networks. Although the basic principle and function of brain oscillations is widely studied and understood, much remains to be done to achieve a complete understanding of how these complex and dynamic fluctuations are related to cognition and a healthy functioning of the brain.

Frequency ranges of oscillation

Brain oscillations are not randomly distributed across the frequency spectrum but mostly occur at very specific frequency bands. There are some biological bases that can explain these *preferred* oscillation speeds. For example, certain groups of neurons have conductance channels which activate with a time constant of around 150ms, which produces an action potential every 100-150ms (Cohen, 2016). If the stimulation increases those neurons will fire a bit faster but never faster than one action potential every 100ms (10Hz). Following the same principle, there are several groups of neurons with different time constants firing at different speeds. These preferred speeds of oscillation are obliquitous in human individuals but also in mammals, insects and mollusks species (Kay, 2015). Thus, researchers and medical doctors usually distinguish different frequency bands in the human brain according to its oscillation speed. As shown in Figure 2, in cognitive neuroscience the frequency spectrum is commonly divided in five bands which are strongly related with cognition: delta (~2Hz), theta (~6Hz), alpha (~10Hz), beta (~30Hz) and gamma (~50Hz).

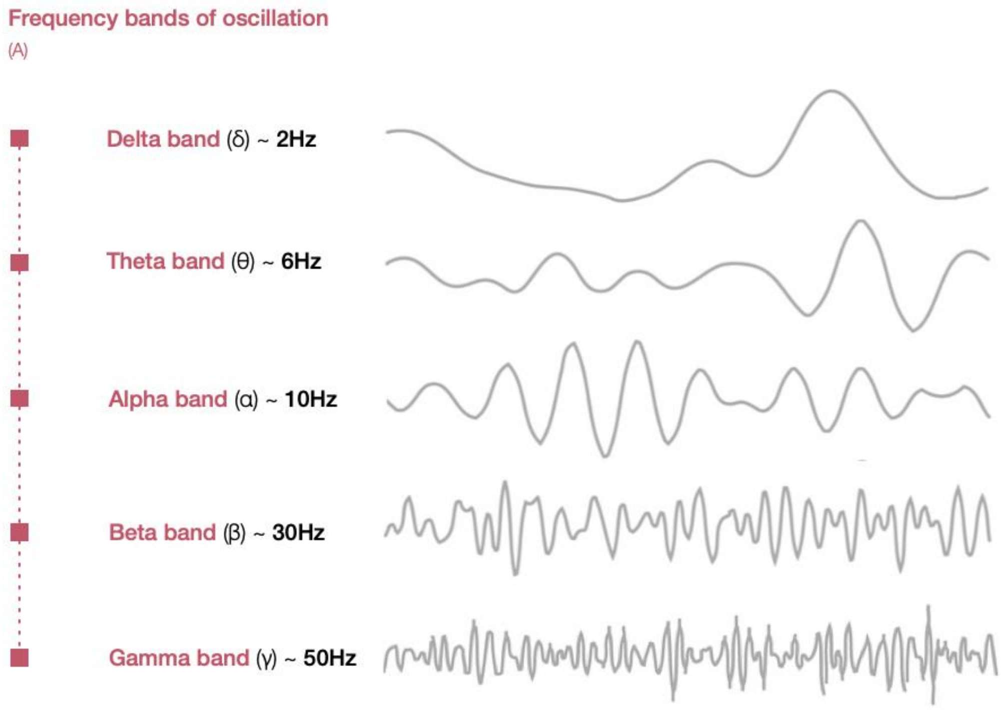


Figure 2. Brain oscillations classified in different frequency bands

Brain oscillation at different speeds observed in human brains. An important research area in the Neuroscience field (neurodynamic), try to determine how these oscillations are generated and the role they play in human brain function. Several studies linked these neural oscillations with many cognitive processes, such as perception (Baldauf & Desimone, 2014), memory (Fell & Axmacher, 2011), motor control and information transfer (Fries, 2005; Schnitzler & Gross, 2005). Brain oscillations have been widely studied in the past decades, and despite the recent neuroimaging techniques there is not a unified interpretation.

The relation between successive oscillation frequencies is close, but not exactly integer multiples of each previous frequency. More specifically, classical frequency bands can be described as a geometric series with a ratio of 1.618, the golden mean (Pletzer et al., 2010). Thus, the synchronization of two oscillation frequencies is mathematically impossible when the ratio between them is the golden mean. Indeed, a brain with oscillation frequencies that are perfect integer multiples of other oscillation frequencies would be very inefficient and only could process a small amount of information at a time. For example, if one oscillation frequency perfectly fits another oscillation frequency (e.g., 5Hz and 10Hz) the action potentials which occur in the peak of both oscillations are synchronized and highly distinguishable.

Recording electrodes

The synchronized electrical activity occurring in the brain crosses several layers of non-neural tissues before it reaches the scalp surface. At this point, the EEG signal measured by the scalp sensors is very attenuated and needs to be amplified. Thus, these sensors act as an interface between the body and the recording hardware. Depending on its design properties, the EEG sensors are subdivided in different categories. The EEG signal is usually acquired as the voltage difference between a fixed *reference electrode* and the actual *recording electrode*s. Thus, the recording electrodes are placed over the specific scalp locations we want to measure, and the signal recorded by the reference electrode is subtracted from each of them. Furthermore, the *ground electrode* matches the potential of the amplifier and the body of the participants, reducing artifacts caused by the common mode interference of the amplifier. Additionally, in order to reduce noise and artifacts sensitivity, a high-conductivity (or low impedance) interface between the sensor and the subject's epidermis is usually desired. To achieve this conductivity levels an electrolytic substance such as conductive gel, a saline solution or even tap water, is applied between the sensor surface and the skin. These types of sensors are known as *wet* or *semi-dry electrodes* (see Figure 3a). As clear disadvantages, wet electrodes require the usage of additional equipment (syringes, gel, etc.) and more tedious preparation and cleaning protocols. On the other hand, *dry* electrodes are directly in contact with the skin, with no electrolytic substances (Figure 3b). These electrodes are fast to place and clean, but the recorded signal is noisier and artifact-prone, which is usually compensated by employing optimized and high-performance amplifiers. Finally, according to sensor shielding, the EEG electrodes are divided in *active* and *passive* electrodes. Passive electrodes (Figure 3d) simply sense and transmit the electric potential fluctuations in the surface of the scalp to the amplifier through a conductive wire. The contact surface is usually plated with silver-silver chloride (Ag/AgCl) or gold.

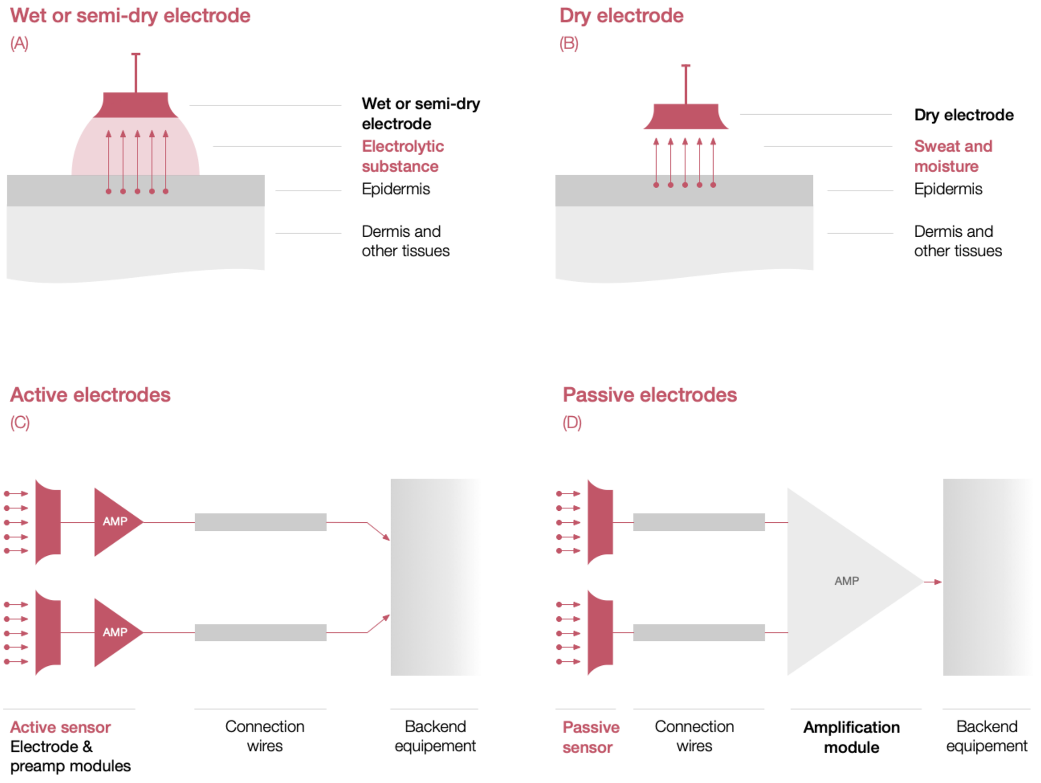


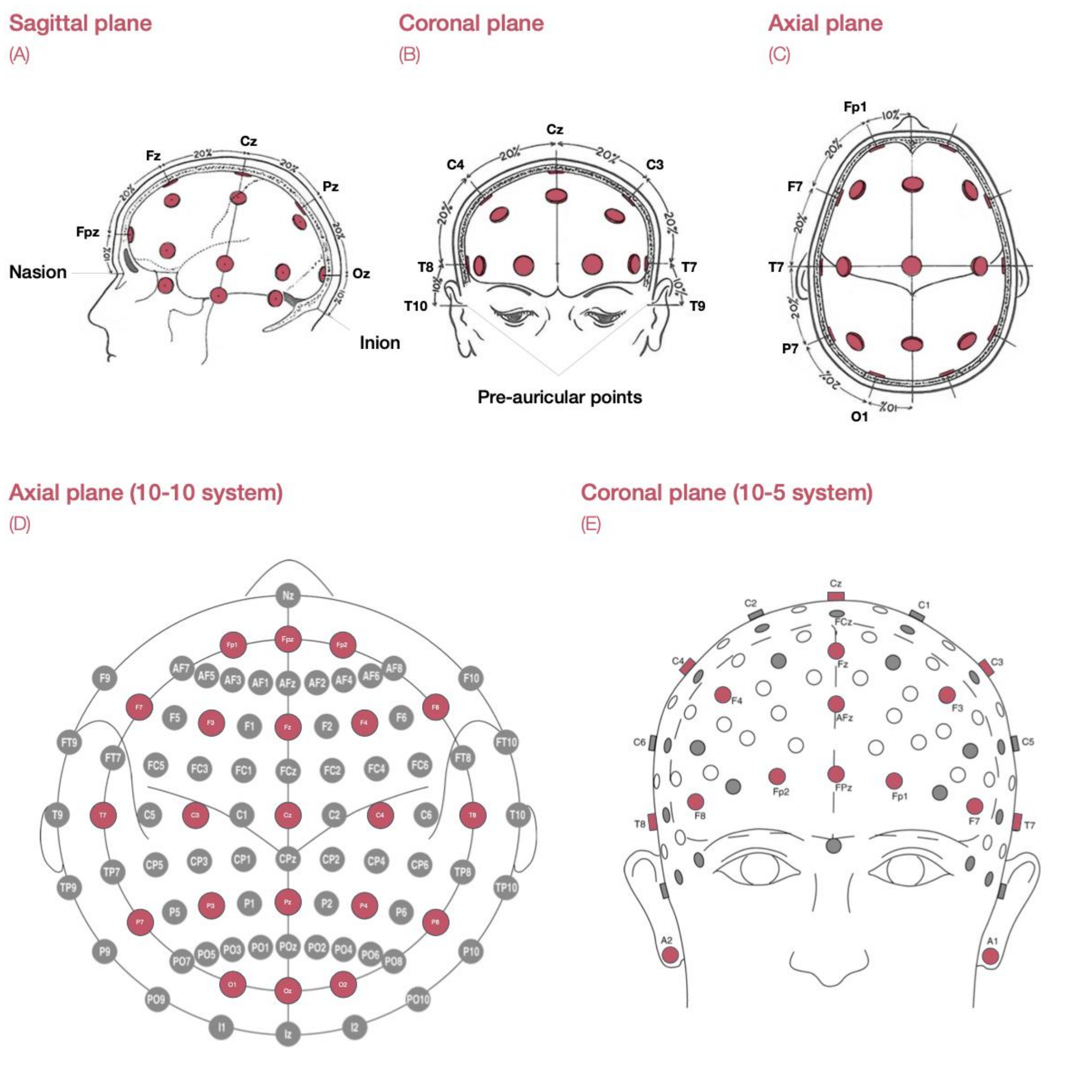
Figure 3. Different types of recording electrodes

Wet or semi-dry electrodes (a) require an electrolytic substance such as conductive gel, saline solution or tap water to achieve a low impedance interface between the sensor and the skin. Dry electrodes (b) are easier to place and clean, but also noisier than wet or semi-dry electrodes. Active electrodes consist of a sensor plate and an individual pre-amplification module, which makes the signal more robust and less sensitive to artifacts since it is amplified before traveling through the connection wires. Finally, in passive electrodes (d) the sensory part and the connection wires are welded together. These electrodes are cheaper and easier to manufacture but the recorded signals are usually noisier.

These electrodes are cheap and easy to manufacture but, in order to achieve a relatively low impedance (<10kΩ) between the skin surface and the sensor, the use of electrolytic substances and skin abrasion is usually required. Active electrodes (Figure 3c) emerged as a response of passive electrodes disadvantages (Taheri et al., 1994). The idea behind active electrodes is to move part of the amplification stage as close as possible to the skin. In doing so, the signals transmitted through the cables are strongly driven by the pre-amplification stage in the sensor, which make them less susceptible to electromagnetic contaminations while traveling, hence improving the signal to noise ratio. In addition, active electrodes can also be used in dry setups, thereby compensating the high impedance and noise levels usually associated with these types of electrodes and obtaining an equivalent performance with no skin preparation. To obtain the best of both worlds, active electrodes can also be used in wet scenarios, applying conductive gel in the sensor-skin interface, obtaining a low impedance channel and all the benefits derived from the active technology. Different studies (Laszlo et al., 2014; Mathewson et al., 2017) demonstrate a superior performance of active electrodes in several scenarios, obtaining better signal quality especially when the impedance values are above 2kΩ. As a main disadvantage, active electrodes are more expensive and complex to manufacture than passive electrodes, since they incorporate an individual pre-amplification electronic module embedded in each sensor, in addition to more wires (power and ground) for enabling the functioning of the pre-amplification circuit.

Electrode layout

In 1947 a committee was formed to define a standard procedure for the measurement of brain activity employing the electroencephalography technique. This committee developed a standardized position layout and labeling for the recording electrodes, which was crucial for compiling, comparing, reproducing, sharing and analyzing different EEG datasets. *The International 10-20 System* is the name that was given to this internationally recognized method for describing and positioning the scalp electrodes (Jasper, 1958). The idea behind this methodology was to equally distribute the recording electrodes over the scalp surface delimitated by four anatomical landmarks: the inion, the nasion, and both preauricular points. The nasion is a depressed area located just on top of the bridge of the nose, while the inion is located back in the skull and is identified as a bump (Figure 4). The preauricular point, also known as *tragus*, is a portion of cartilage located in front of both ears. Thus, the distance between contiguous electrodes is 10% | 20% | 20% | 20% | 20% | 10% of the total inion-to-nasion or tragus-to-tragus distance, measured over the sagittal and coronal central reference curves. This configuration leads to a 21 electrodes montage, which could be insufficient for some applications, especially for the identification of intracranial sources of scalp-recorded EEG signals. The separation distance protocol can be modified in order to obtain high-density montages, filling the intermediate sites halfway between those of the existing 10-20 system. Depending on the selected distance between electrodes we can generate new layouts, such as the *10-10 System*, leading to an 81 electrodes montage or the *10-5 System* with 320 electrodes (Jurcak et al., 2007; Oostenveld & Praamstra, 2001). Note that modern EEG systems, often having 128-256 channels available are derived from those standards.



Sagittal plane (A)

Coronal plane (B)

Axial plane (C)

Axial plane (10-10 system) (D)

Coronal plane (10-5 system) (E)

Figure 4. International location system for EEG recordings

(A-C) Original electrode placement layout of the International 10-20 system for EEG recordings. *Figure modified from Seeck et al., 2017*. (D) High-density montage according to the 10-10 system layout. Red circles represent the location of the original 10-20 system and the grey circles are the additional electrodes introduced in the 10-10 extension. *Figure modified from Oostenveld & Praamstra, 2001*. (E) Coronal plane showing the extended 10-5 system. White circles indicate the additional electrodes introduced in the 10-5 extension. *Figure modified from Oostenveld & Praamstra, 2001*.

The labeling system of the recording electrodes also follows an international standard. Each electrode is labeled with a letter indicating the brain lobe over which it is located: ***Fp*** for frontopolar, ***F*** for frontal, ***C*** for central, ***P*** for parietal, ***T*** for temporal and ***O*** for occipital electrodes. Electrodes located in the midline (zero line) are labeled with a ***Z***. Additionally, electrodes located over the right hemisphere are labeled with even numbers (2,4,6,8) while odd numbers (1,3,5,7) refer to those electrodes located over the left hemisphere. This standard organization and sensor nomenclature allows us to obtain comparable result in the EEG literature and to easily determine the specific brain area that is covered by each electrode.

Sources of noise in EEG recordings

During an EEG experiment the signal is usually collected in controlled environments where we try to minimize external interactions which affect the signal. However, several sources of both physiological and environmental noise inevitably interact with the electrical activity recorded by the scalp electrodes. Any component of the EEG signal recorded by the sensors which is not directly originated in the brain is called an artifact (Urigüen & Garcia-Zapirain, 2015). Physiological artifacts are caused (voluntarily or involuntarily) by the body of the participant while non-physiological artifacts emerge from the environment. One of the most common sources of physiological artifacts is the ocular activity, including blinks and lateral movements (Clark, 1998; Islam et al., 2016; Sörnmo & Laguna, 2005). Due to the corneoretinal potential (CRP) the eye acts as a moving electric dipole, changing the surrounding electric field (Electrooculogram, EOG). These field perturbations are mostly recorded by frontal electrodes leading to a signal component one order of magnitude larger than the EEG signal (100-200μV). Muscle activity is also another important source of noise in EEG recordings since contracting muscles generate high-frequency electrical activity (Electromyogram, EMG) which could mask brain activity in beta and gamma bands. Low-amplitude and rhythmic distortions of the EEG signal are sometimes observed in the recorded signal. This artifact is caused by the electrical activity of the heart (Electrocardiogram, ECG) and generates a rhythmic pattern that overlaps the EEG signal. Additionally, slow drifts (low frequency components) in the recorded EEG signal are also commonly observed in perspiring participants. This perspiration caused by the sweat glands on the skin changes the baseline of the electrodes and causing shorts in more extreme scenarios. Finally, respiration artifacts are found especially in sleep recordings since inhale and exhale movements modify the skin-electrode contact specially if the participant is lying on a bed. On the other hand, cable sway is one of the main causes of external motion artifacts in EEG recordings, corrupting the signal and masking the neural signal of interest (Symeonidou et al., 2018). In addition, a poor electrode grounding or insufficient wire shielding could cause AC electromagnetic interferences, generating high-frequency components (50-60Hz and its harmonics, depending on the region) coupled to the EEG signal. This artifact is easily detectable, since it generates a large frequency spike of around 50-60Hz in the frequency spectrum of the signal. Other causes of external artifacts encompass bad electrode-skin contact, mainly due to body movements, causing abrupt and high amplitude interferences localized in individual electrodes. In the section 2.1.2 Signal preprocessing we describe in detail how to deal with different sources of unwanted noise and artifacts which are inevitably present in EEG recordings.

Advantages and limitations

The electroencephalography presents important advantages which have made it one of the most employed techniques in clinical, research or even commercial applications. The most important is its outstanding temporal resolution. The recording hardware is capable of collecting data from hundreds of sensors simultaneously at a millisecond levels. It is also relatively non-invasive and tolerant of subject movement (O' Regan et al., 2010). It is totally silent, so participants are able to respond to auditory stimulus. It can be used in patients with metal implants or claustrophobia, since they are not exposed to high intensity magnetic fields, radiation, or closed and confined spaces. Additionally, this technique is cheaper than most neuroimaging modalities, the recording equipment is relatively small and easy to move, which makes this technique highly adaptable to different contexts compared with other brain assessment devices. Finally, the recording process of the EEG signal is safe and entails almost no risks or side effects and little discomfort for participants.

Due to all of its advantages, the EEG technique also presents some important limitations compared with other neuroimaging modalities. The more significant ones are related with the spatial resolution of the technique. Several factors lead to this poor source location. For example, the electrical activity occurring in deeper regions of the brain are rarely captured by scalp electrodes. Even in the cortex, depending on the location and the orientation of the electrical dipole which generates the EEG signal, there may be a misinterpretation of the original source due to the inverse problem (Grech et al., 2008). Despite the tremendous effort made by researchers to obtain acceptable estimations of sources of brain activity, it is mathematically impossible to reconstruct a specific intracranial current from the recorded EEG signal, since some currents generate potentials that cancel each other out (Niedermeyer & da Silva, 2005). Another important disadvantage of the EEG technique is that the signal-to-noise ratio of the recorded signal is often low, therefore requiring a large amount of subject to obtain a decent statistical power in an EEG study.

2.1.2. Signal preprocessing

Like many other physiological recordings, the raw EEG data usually contains different types of noise and artifacts which obscure the actual neural signals and need to be removed or attenuated for the subsequent data analysis. This cleaning and signal preparation process is known as the preprocessing stage. Although this procedure is always performed before any analysis, there is not a standard pipeline that is broadly accepted by the community (nonetheless, we can find in the literature some standardization attempts (Bigdely-Shamlo et al., 2015)). The researcher is, based on his experience, how decides how to transform and clean the recorded raw data. Here we present a set of preprocessing routines which are commonly performed in order to account for different sources of noise. Note that, depending on researcher’s preference, these routines are sometimes done in a different order, or maybe not always computed.

Filtering

A digital filter attenuates a specific frequency band of a signal, suppressing all the frequency components that are below or above certain cutoff value. There are several types of filters according to their behavior. *Low-pass filters* attenuate all the frequency components above the filter cutoff frequency. For example, a low-pass filter of 128Hz attenuates all the frequencies above 128Hz while frequency components below that value remain unaltered. Following the same approach, *high-pass filters* attenuate frequency components below its cutoff frequency. Additionally, *band-pass filters* are a combination of the previous two, which keep the frequency components between an upper and a lower bound unaltered while attenuate the rest.

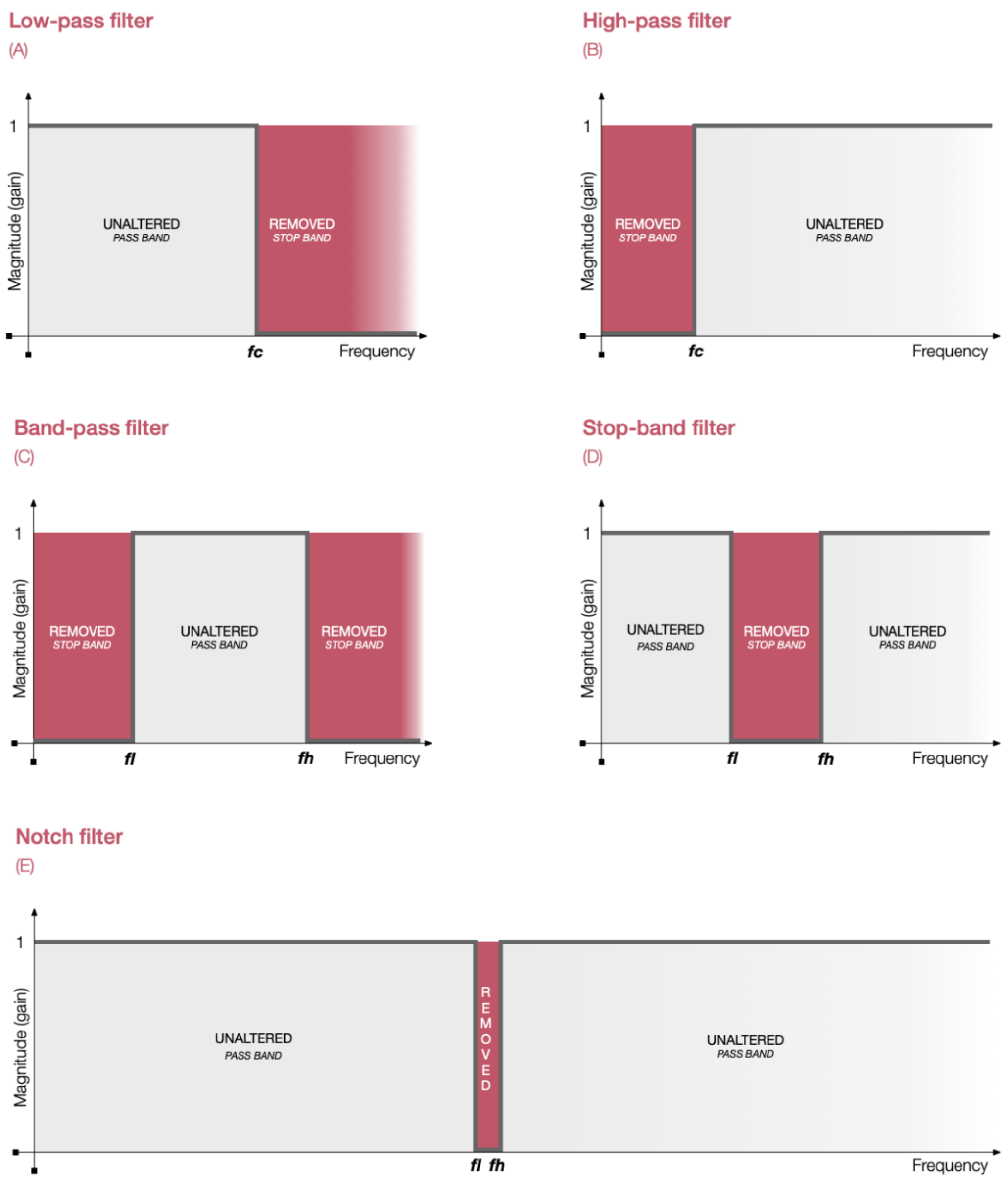


Figure 5. Ideal magnitude response of different digital filters

The frequency response of a filter indicates how the frequencies of the filtered signal will be affected. As shown, red areas represent the rejection band of each filter. Frequency components on this range will be attenuated. Contrary, grey areas represent the pass band of the filter. Thus, frequency components on this range will remain unaltered.

The opposite filter is known as *band-stop* or *band-rejection filter*, which attenuates the frequency components between an upper and a lower bound. Finally, a *notch filter* is a special type of a band-stop filter with a very narrow rejection band. The behavior of any filter is determined by its *Frequency Response Function* (FRF). The frequency response function is a complex signal which yields the gain and phase-shift of the filter as a function of frequency. Figure 5 shows the ideal frequency response function (magnitude) of the previously mentioned type of filters.

The mathematics of filtering and design decisions are extensive and out of the scope of the present thesis (Ifeachor & Jervis, 2002; Parks & Burrus, 1987; Widmann et al., 2015). However, some aspects are worthy of consideration. Firstly, filtering is a technique that distorts the signal in a considerable way, so it must be employed carefully and consciously. In order to minimize the effects of filtering, an adequate filter design should be selected. For example, filters can be classified into two main categories: Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters. Although IIR filters are computationally more efficient and are characterized by steeper cutoffs than FIR equivalents, in EEG preprocessing FIR filters are always recommended. This is mainly because FIR filters can be easily designed to have linear phase. This characteristic implies that all frequencies are shifted in time by the same amount maintaining a constant *group* and *phase delay*.

Although the use of filters in EEG preprocessing pipelines is open to discussion (de Cheveigné & Nelken, 2019; Rousselet, 2012; van Driel et al., 2021; VanRullen, 2011; Widmann & Schröger, 2012), it is true that filters are commonly applied in most preprocessing pipelines in the literature. The reason is that most of the EEG studies in humans focus on brain oscillations ranging from 1 to 30Hz. Thus, if an adequate filter is carefully selected and frequency components beyond the frequency band of interest are attenuated without introducing filtering artifacts, the signal-to-noise ratio could increase in a significantly. For example, perspiration on the scalp or cable movements can lead to slow drifts in the recorded EEG signal below 1Hz. Additionally, other sources of noise such as muscle artifacts, or electromagnetic interferences lead to higher frequency artifacts. Figure 6 shows the resulting Power Spectral Density (PSD) function of an EEG signal contaminated with a high frequency artifact (power line noise and its harmonics) after applying different filters.

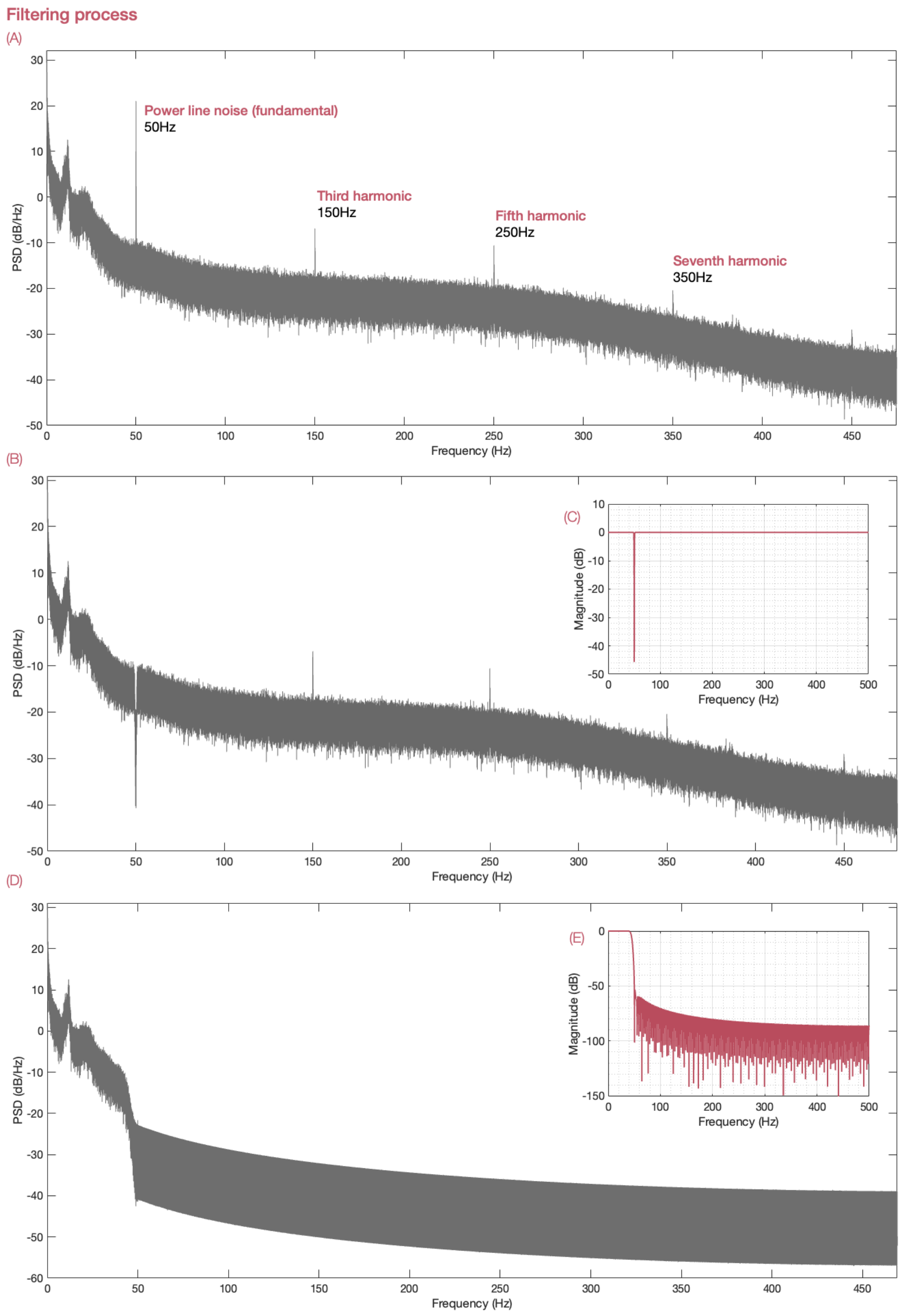


Figure 6. Removing power line noise with filters

(A) The PSD function of a raw EEG signal contaminated with power line noise (50Hz) and its harmonics. (B) The resulting PSD function after filtering the signal employing a notch filter with a cutoff frequency of 50Hz (C). As shown, the fundamental frequency has been attenuated, remaining the rest of the spectrum unaltered. (D) The resulting PSD function after filtering the signal with a low-pass filter (50Hz cutoff) (E). Frequencies above 50Hz have been attenuated.

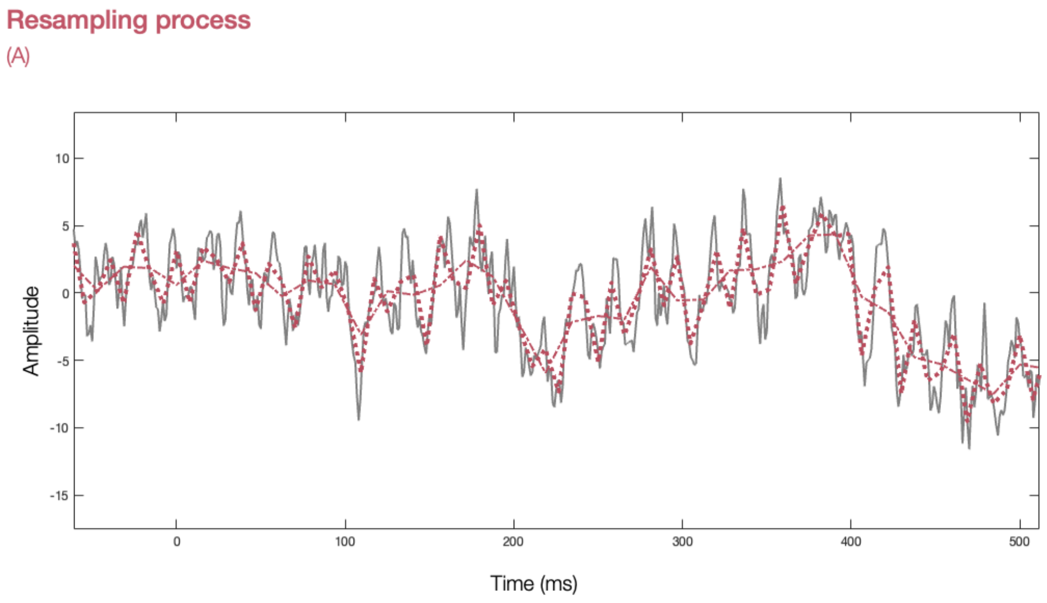


Figure 7. Resampling an electroencephalography signal

This figure shows the same EEG signal sampled at different frequencies. The sampling frequency of the grey, red dotted and red dashed lines are 1000Hz, 128Hz and 64Hz respectively.

Resampling

As mentioned in previous sections, the electroencephalography technique is noted for its remarkable temporal resolution. Indeed, the EEG signals are usually recorded at sampling frequencies around 2000Hz. This means that each electrode records, transmits and stores two thousand values of brain activity per second. This sampling frequency leads to a tremendous amount of data, especially in long experiments and high-density montages. For example, if the signal is acquired at a sampling frequency of 2KHz with a high-density setup of 128 electrodes, and the data is represented as a 32-bit float number, the total bitrate will be 8.192.000 bits per second (1MB/s). If the total duration of the experiment is 40 mins and the total sample is 48 subjects, the complete EEG dataset will be more than 115GB. This amount of data can be easily stored in modern storage devices but, the computational load required to process and analyze this high-resolution data can be ridiculously high. In fact, these high values of sampling frequencies are not usually required to study brain oscillations. According to Nyquist-Shannon theorem (Shannon, 1949), the minimum sampling frequency $\boldsymbol{f_s}$ required in order to capture all the information of a continuous signal of a certain bandwidth $\boldsymbol{B}$ is $\boldsymbol{f_s = 2B}$Hz. Thus, according to this rule, if the highest frequency of interest for a specific study is 40Hz, a sampling frequency of $\boldsymbol{f_s} = 80$Hz is sufficient to capture all the information beyond this bandwidth. Frequencies above this value will be confounded or *aliased* to lower ones (Figure 7). This downsampling process will accelerate the analyses considerably, obtaining similar results. There is not a golden rule about how and when to resample the raw data, but resampling is usually recommended after filtering and epoching the data, making the epochs long enough so that edge effects from the filtering do not affect the temporal span of the epoch.

Blind source separation: The Independent Component Analysis

Imagine that three microphones, placed in different locations, record the sound of three musical instruments while playing music. Thus, the track recorded by each microphone is a mix of the individual sounds of each musical instrument and the ambient noise. The question then is whether it is possible to separate individual contributing sources (the sound of a single instrument) from the observed total signal (the mix of each instrument and the ambient noise). The blind source separation techniques decompose the observed signal in a linear combination of other signals according to some mathematical considerations such as independence or orthogonality. Mathematically, these techniques represent a linear change of basis from data collected at single scalp channels into a spatially transformed *virtual channel* basis. One of the most employed demixing techniques in EEG preprocessing is the Independent Component Analysis (ICA) (Choi, 2004; Comon, 1994), which leads to better results than simpler counterparts such as Principal Component Analysis (PCA). One of the main advantages of ICA over PCA is that orthogonality or gaussian behavior of the original sources are not assumed. In EEG preprocessing the ICA is used to separate components of the original signal in order to remove artifacts such as eye blinks, eye movements, muscle activity or heartbeats. There are several tools and algorithms available to compute the ICA from an EEG dataset, including Fieldtrip or EEGlab. Figure 8 depicts the process for removing one of the most common artifacts in EEG recordings, eye blinks and lateral movements employing the ICA technique. First using a preprocessing software (e.g., EEGlab) the ICA algorithm should be computed.

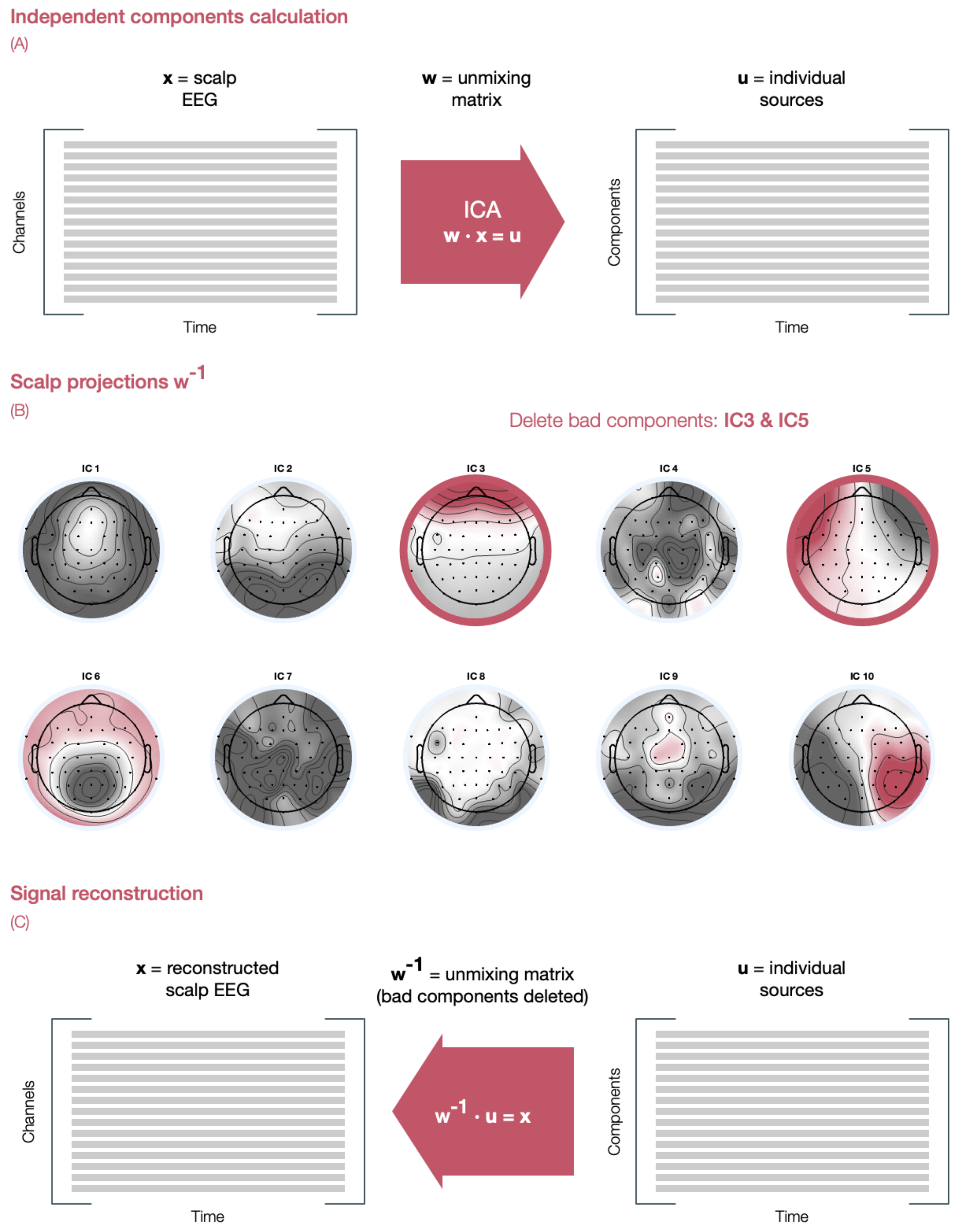


Figure 8. Independent Component Analysis diagram

(A) Independent Component calculation. When the ICA is computed, the unmixing matrix **w** is first calculated. This matrix transforms the scalp EEG data **x** in its independent components. (B) Scalp projection of each component. Noisy components should be deleted before the final signal reconstruction. This picture shows two clear examples of artifactual components: IC3 and IC5, accounting for eye blinks (strong frontal activations) and eye lateral movement (strong frontal and polarized activations). (C) The final EEG signal is then reconstructed from the individual sources **u** and the inverse unmixing matrix **w,** subtracting the projection of the artifactual components.

According to EEGlab website, researchers can select different implementations of the ICA algorithm, including *Infomax ICA* (Makeig et al., 1996), *JADE* or *SOBI* (A Belouchrani et al., 1997; Adel Belouchrani et al., 1993), among others.

Before computing the ICA decomposition, each row of the original EEG data matrix $\boldsymbol{X}$ contains the time courses of mixed voltage differences between sources and a reference potential recorded by a specific electrode. When ICA decomposition is computed, each row of the source matrix $\boldsymbol{S} = \boldsymbol{W} \cdot \boldsymbol{X}$ (activation matrix) represents the activity of one component process spatially filtered from the original data channel. The $\boldsymbol{W}$ is the weight matrix to go from the $\boldsymbol{X}$ space to the $\boldsymbol{S}$ space. This decomposition is performed to produce the maximally temporally independent signals available in the original data channel. These signals are the individual information sources that have been mixed by volume conduction and recorded by the scalp electrodes. Analyzing these individual sources (its activity power spectrum, the continuous time course or its projection in a scalp map), we can note that they may represent activity from non-cortical sources, such as eye blinks, eye movements, muscle activity or heartbeats. Learning to recognize and identify different types of artifactual components is not a straightforward exercise, it requires experience and usually, there is no consensus as regards how to act when these artifactual components are identified. There is a myriad of resources when it comes to learning how to recognize and when to remove these artifactual activations, visually or even automatically employing machine learning techniques such as *MARA* (Winkler et al., 2011) or *IClabel* (Pion-Tonachini et al., 2019). Here, in Figure 8 we presented two of the most common and easily recognizable artifactual components, one that accounts for the eye blinks, with a prominent frontal activation (IC3 in the figure) and the other accounting for the eye lateral movements, characterized by prominent bi-polar activations in frontal areas (IC5 in the picture). Once the artifactual components have been spotted, the contribution of each component to the original data matrix should be removed. To do that, the projection of each artifactual component ($\boldsymbol{S_c}$) in the electrode space $\boldsymbol{X_c}$ should be calculated as $\boldsymbol{X_c} = \boldsymbol{W}^{-1} \cdot \boldsymbol{S_c}$ and subtracted from the original data matrix $\boldsymbol{X}$. Note that $\boldsymbol{W}^{-1}$ is the inverse matrix to go from the source space $\boldsymbol{S}$ to the electrode space $\boldsymbol{X}$. Figure 9a shows a real example in which the initial EEG signal is clearly contaminated by eye blinks artifacts. After computing the ICA procedure, the source activations can be inspected and selected for their removal (Figure 9b). As a result, the reconstructed EEG signal is perfectly clean and free of artifacts (Figure 9c).

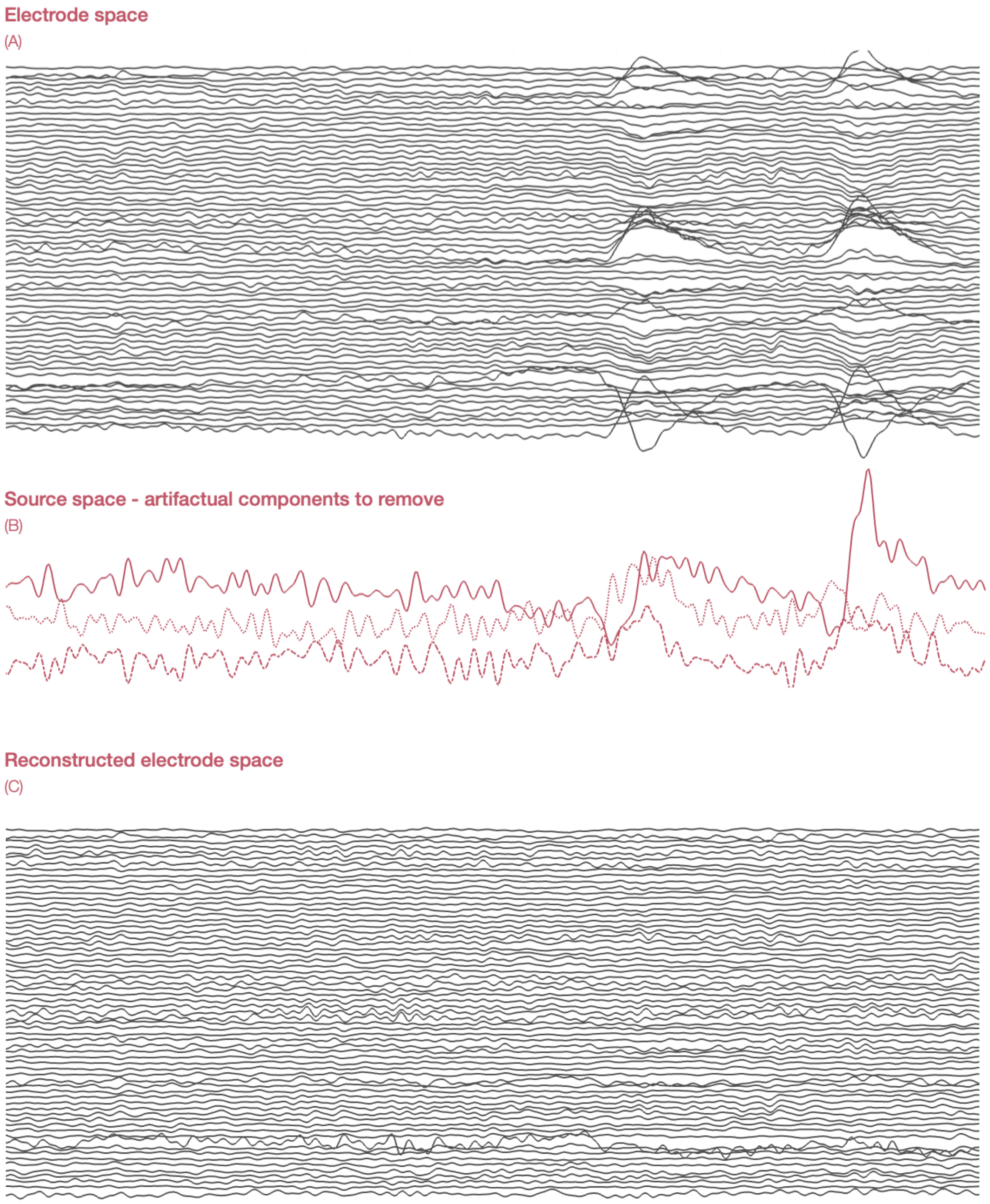


Figure 9. Artifact removing procedure using ICA

This picture depicts the artifact removing procedure employing the Independent Component Analysis technique. (A) The initial EEG dataset is clearly contaminated by eye blinks and movements. (B) Three artifactual independent components were manually inspected and selected to be removed. (C) After the removing process, the EEG dataset is reconstructed, and the eye blinks were removed from the original data.

Trial rejection

Another usually performed step in most preprocessing pipelines accounts for spurious artifacts in the data. This process is commonly called *Trial rejection* and it tries to reduce the variance in the data which is not related to the designed experimental conditions. During the trial rejection procedure, the researcher visually or automatically inspects the data searching for recognizable artifactual patterns and flag them for further rejection. Although the manual process may be highly arbitrary and time consuming (since it depends on the researcher's judgement, and the amount of data to visualize and inspect is typically significant), it is commonly performed in many laboratories. Contrary, the automatic trial rejection procedure is faster and more objective, since it is based on quantitative properties of the signal, such as its variance, the magnitude of voltage increases, abnormal spectra, etc. There are several automatic routines implemented in most preprocessing software packages which allow to automatically detect artifactual trials based on predefined parameters. Although this procedure can be easily computed, it is important to truly understand the selection criteria and the configuration parameters of the rejection algorithms, as well as the nature of the artifacts to deal with. Otherwise, these automated routines could lead to undesired scenarios in which real neural data of interest will be removed. It is usually recommended to visually inspect the automatically selected trials for removing. As in earlier instances, there is not a standard procedure that accounts for bad data. The selection of the rejection technique (manual/automatic), the rejection criteria and the configuration parameters always depend on several factors, such as the type of analysis to compute, the quality of the recorded data and the experience and judgment of the researcher.

Bad channels interpolation

When an electrode is not recording the EEG signal property over time is termed a bad channel. These electrodes usually provide data that is too noisy (or not even neural data) to be usable. Bad channels are easily detectable by visual inspection although several automatic tools are available in almost every preprocessing software (Bigdely-Shamlo et al., 2015). Several reasons cause an electrode to malfunction. Sometimes the electrode is not correctly placed, making no contact with the scalp, the electrolytic solution applied to the electrode dries out and therefore increasing the electrode impedance, or two adjacent electrodes are bridged. Whatever the problem is, once a bad channel is detected, it can be removed or interpolated in order not to contaminate further analyses. It is important to mark as bad only those channels with abnormal behavior over time, and not those which are periodically noisy. However, it is not always a straightforward exercise to decide whether a channel should be marked as bad or not, since the concept of *noisy channel* is always arbitrary. For example, should a correctly placed electrode containing muscle activity be marked as a bad channel? This decision should be made based on the researcher's experience and judgment. Once a bad channel is spotted, removing it from the dataset could be sufficient in some scenarios, but if maintaining the rank of the data is important for further analysis, this bad channel should be interpolated. The interpolation is a reconstruction process based on the spherical spline method (Perrin et al., 1989). This methodology projects the sensor locations onto a unit sphere (representing the head) and interpolates the signal at the bad sensor locations based on the signals at the good locations.

Re-referencing

The voltage recorded for each electrode in EEG is always relative to the voltage recorded by another electrode (or combination of electrodes) termed as reference. Thus, the electrical activity sensed by the reference electrode will be reflected in the rest of the recording electrodes, contaminating the resulting signal. For that reason, it is important to select a reference electrode located far away from the region of interest, or, once the signal is recorded, re-reference the entire dataset to another electrode or combination of electrodes. Some common choices are electrodes placed in the mastoids, the earlobes or the central electrode (Cz). Another common practice is to select the common average reference, which means that the signal recorded by each electrode is re-referenced to the mean voltage of all the electrodes. This re-reference technique reduces the influence of a single reference electrode, and it is widely employed in the EEG literature.

2.2. Magnetic Resonance Imaging

The Magnetic Resonance Imaging (MRI) is an extremely versatile imaging modality that can be used to study both brain function and brain structure. Both structural and functional images can be acquired using the same scanner, and different types of brain images can be generated to emphasize different tissue characteristics. In this section we present the physics principles of the MRI signal and the most common preprocessing routines to identify and remove artifacts and validate model assumptions.

2.2.1. Basic principles of the Magnetic Resonance Imaging

The hydrogen atom: Rotation and precession movement

The effect of magnetic resonance is based on the nuclear magnetic resonance (NMR) phenomenon, a physical property of some atomic nuclei experimentally described in 1946 for both Bloch and Purcell (Bloch, 1946; Purcell et al., 1946), who won the Novel Prize in physics in 1952. The hydrogen atom (H+) consists of a single proton, a positively charged particle that spins on its own axis constantly. This spin generates an electrical current on its surface which induces a magnetic moment and a local magnetic field μ (dipole) with a specific direction and amplitude. Each proton rotates without a fixed direction but, interestingly enough, in presence of an external magnetic field, those protons align with the external field and start an *asynchronous precession movement*. The precession movement is a gyroscopic rotation around the axis of the external magnetic field, and its angular frequency ω_0 (the Larmor frequency) depends on the intensity of that external field B_0 (Figure 10). Since hydrogen atoms represent a high percentage of our body, it would seem convenient to induce magnetic changes on these particles in order to obtain the MR signal.

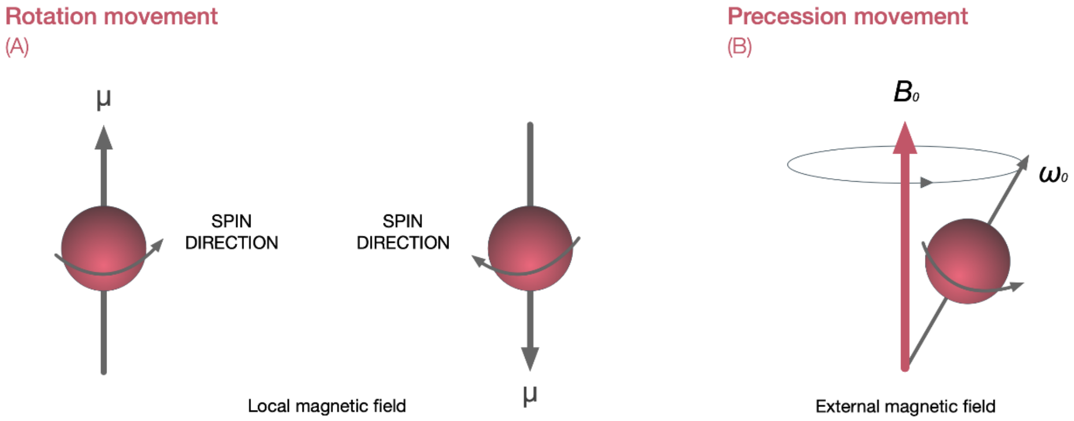


Figure 10. Nuclear spin and precession movement

The spinning particle induces a local magnetic field depending on the spinning direction. In the presence of an external magnetic field, the spinning particle aligns with the direction of the external field but due to its rotation around its own axis, this particle starts a gyroscopic movement also known as precession movement.

External magnetic field

MRI scanners generate a strong magnetic field B_0 in order to obtain the MR signal. The intensity of this field varies from 1.5 to 7 Teslas (T). In order to put this number into perspective, the Earth magnetic field is 0.000005T, which means that a 7T MR scanner generates a magnetic field 1.400.000 times stronger. As mentioned before, in the presence of an external magnetic field B_0, the hydrogen protons *longitudinally align* with that magnetic field. However, those protons can be aligned with a parallel (low-energy state) or antiparallel orientation (high-energy state) to the magnetic field (Figure 11). The magnetic moments induced by particles with parallel and antiparallel orientations cancel each other out, which lead to a null net magnetization vector ($M = 0$). However, in solid and liquids, the number of particles in low-energy states usually predominates when a strong magnetic field is present, and this unbalance is directly proportional to its intensity. This excess of particles in low-energy states contributes to the net magnetization vector, so the more intense the external magnetic field, the better the resulting MR signal. As an example, under the influence of a magnetic field of 1.5T, only nine H+ per million contributes to the net magnetization vector.

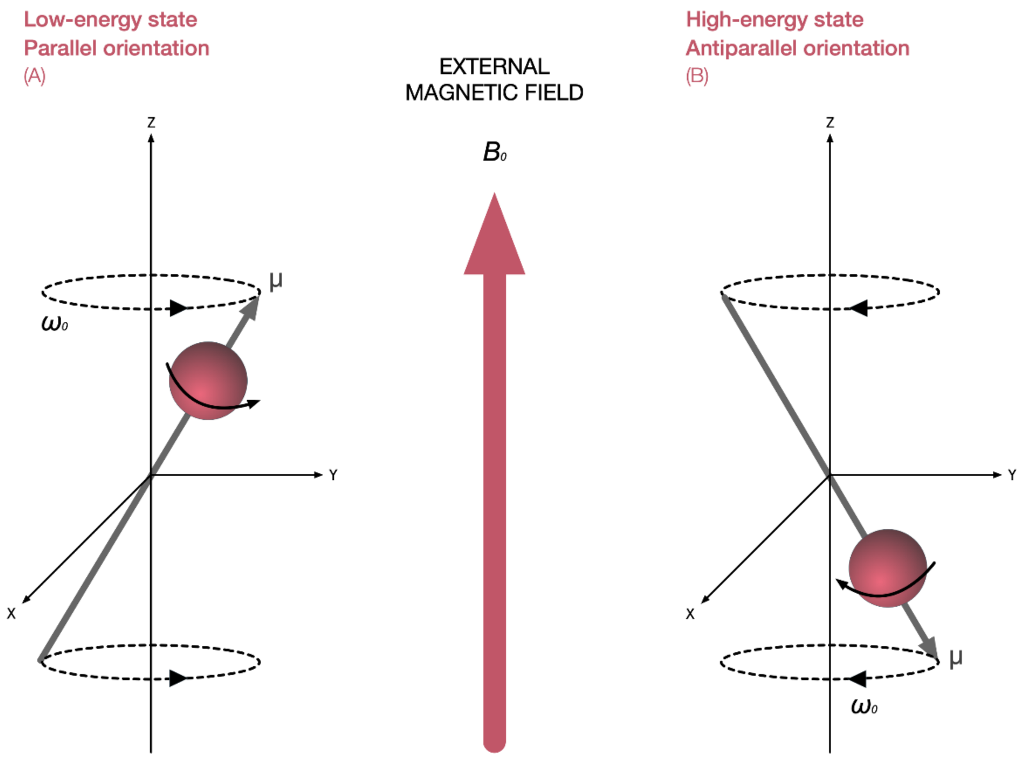


Figure 11. Longitudinal alignments in presence of an external magnetic field

Parallel and antiparallel longitudinal alignments of H+ particles, corresponding with low and high energy states respectively.

Radiofrequency pulse

When particles longitudinally aligned with the external magnetic field B_0 are perturbed by a perpendicular radio frequency (RF) pulse of the same frequency as their precession frequency ω_0, the longitudinal alignment becomes transverse to the field. This phenomenon is called the *resonance process* and refers to the ability of nucleus to absorb energy emitted at the same frequency of their resonance frequency ω_0. This energy absorption changes the energy state of the particles from low to high. Additionally, the asynchronous precession movement becomes phase synchronous (Figure 12). The MR scanner has antennas which emit these radiofrequency pulses "by slices".

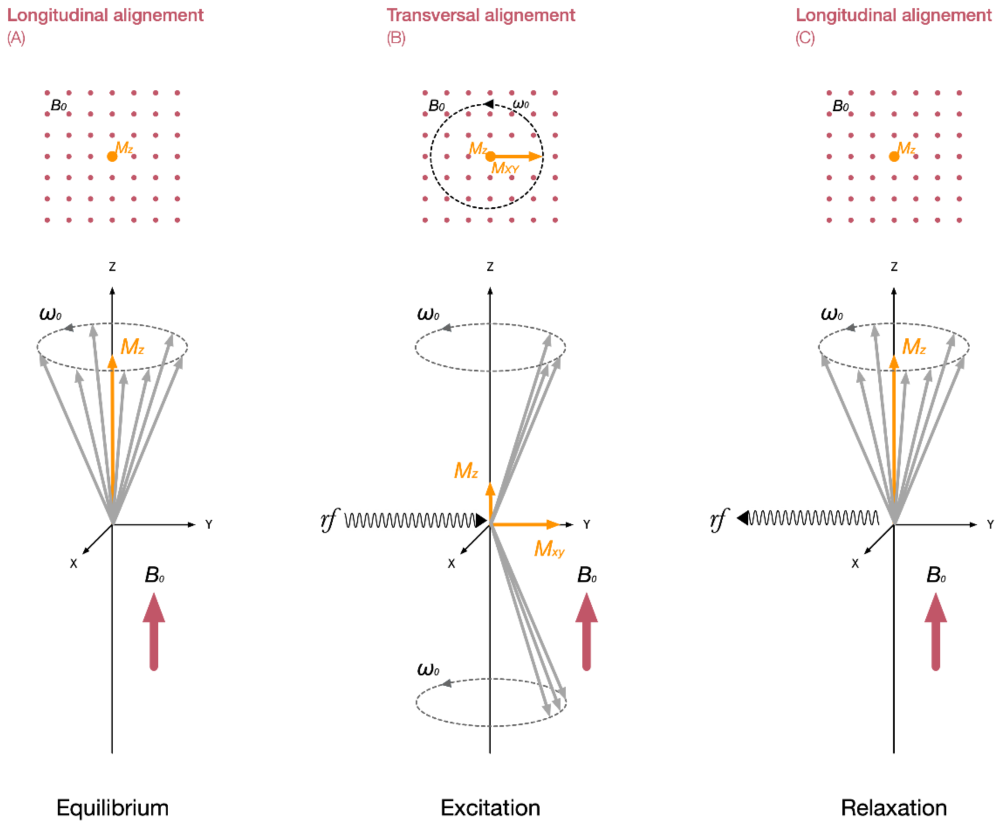


Figure 12. Excitation and relaxation processes.

When the RF pulse is applied to the system the longitudinal magnetization component decays while the transverse magnetization increases. Once the RF transmitter is turned off, relaxation occurs. The protons re-radiate the absorbed energy and the coils of the scanner receive the signal in the transverse plane due to variations of the transverse magnetization vector. This signal is oscillating at resonance frequency and signal envelope is a decay curve described as an exponential curve.

For example, for 1.5T fields, the resonance frequency is around 64MHz. Unlike X-rays, these low-frequency and low energy signals emitted by the scanner are innocuous to the human body. When the influence of the pulse stops, each particle recovers its longitudinal alignment with the external magnetic field through *longitudinal relaxation processes*, re-radiating the absorbed energy. The scanner coils can collect the emitted signal, whose intensity is directly proportional to the number of protons found in each region. Thus, thanks to reconstruction algorithms we can map these recorded intensity differences in grayscale images.

Relaxation processes

The term relaxation refers to the process by which a nuclear spin returns to thermal equilibrium after absorbing the radiated RF energy. Longitudinal and transverse relaxation processes are described by exponential curves and the time constants, T_1 and T_2, respectively. The *longitudinal relaxation time* is described by an exponential growth defined by the time constant T_1. This is the required time for the system to reach thermal equilibrium levels (63%). On the other hand, the *transversal relaxation time* is described by an exponential decay defined by the time constant T_2. In that case, T_2 is the time required for the transverse magnetization to fall to approximately 37% (1/e) of its initial value. Interestingly, T_1 and T_2 differ across different tissues. In other words, the protons contained in White Matter (WM), Grey Matter (GM) and Cerebrospinal Fluid (CSF) behave differently, leading to different relaxation curves (e.g., $T_1 = 600, 1000, 3000$ms for WM, GM and CSF respectively).

Structural images

The MRI scanner builds images that represent spatial distribution of a certain property of the protons, registered by the receiving coils. This property can be the intensity of the signal, which, as mentioned before, is proportional to the number of protons found in a certain area of the brain. This image modality is also known as *Proton Density* (PD) image. Additionally, the MR scanner can be tuned to highlight certain types of tissues according to the relaxation time of the tissues in which these protons are found. This is so because the intensity of the received signal follows the following exponential function:

$$I = M_0 \cdot \left(1 - e^{-\frac{TR}{T_1}} \cdot e^{-\frac{TE}{T_2}}\right)$$

Equation 1

where TR is the *Repetition Time*, which indicates how frequently the protons of the system are excited (how often the RF pulse is emitted) and the TE (*Time to Echo*) value represents the time between the delivery of the RF pulse and the receipt of the echo signal.

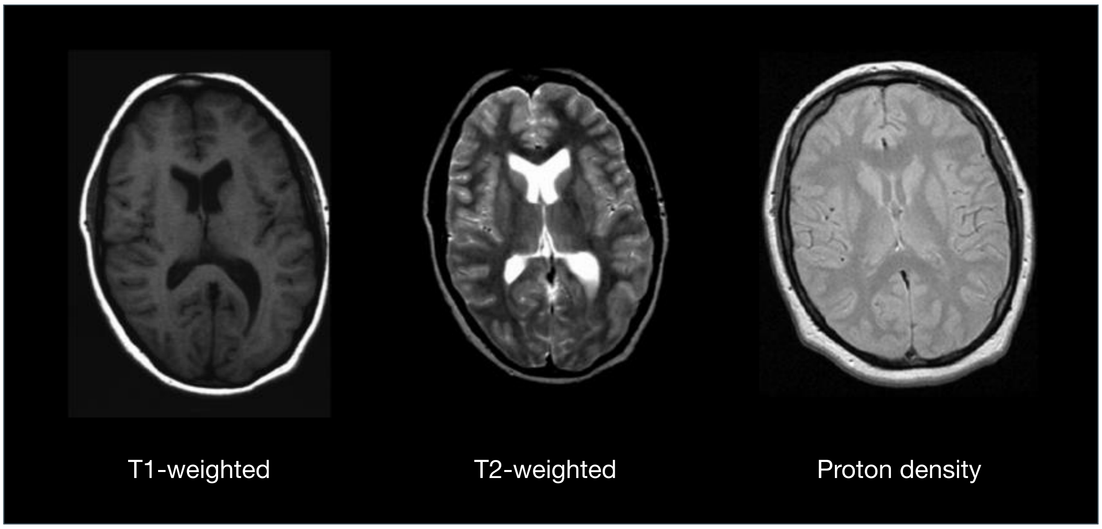


Figure 13. Different MRI pulse sequences

Three of the most common MRI pulse sequences: the T1-weighted, the T2-weighted and the proton density images.

Thus, according to Equation 1 and the fact that different tissues are characterized by different relaxation times, we can tune the TR and TE values to obtain images highlighting different brain tissues. The most common MRI sequences are T_1-weighted and T_2-weighted images. According to Equation 1, using short TE and TR times we obtain T_1-weighted images, where the contrast and brightness of the resulting image are predominately determined by the first exponential (T_1 properties of tissue). Conversely, T_2-weighted images are produced by using longer TE and TR times. In this case, the contrast and brightness are predominately determined by the second exponential (T_2 properties of tissue). Those images are easily distinguishable by looking to the cerebrospinal fluid, which is dark in T_1-weighted images and bright in T_2-weighted images (Figure 13). Finally, using a short TE and a longer TR we cancel the contribution of both exponential terms, and the contrast and brightness of the images are predominately determined by the proton concentration across different tissues.

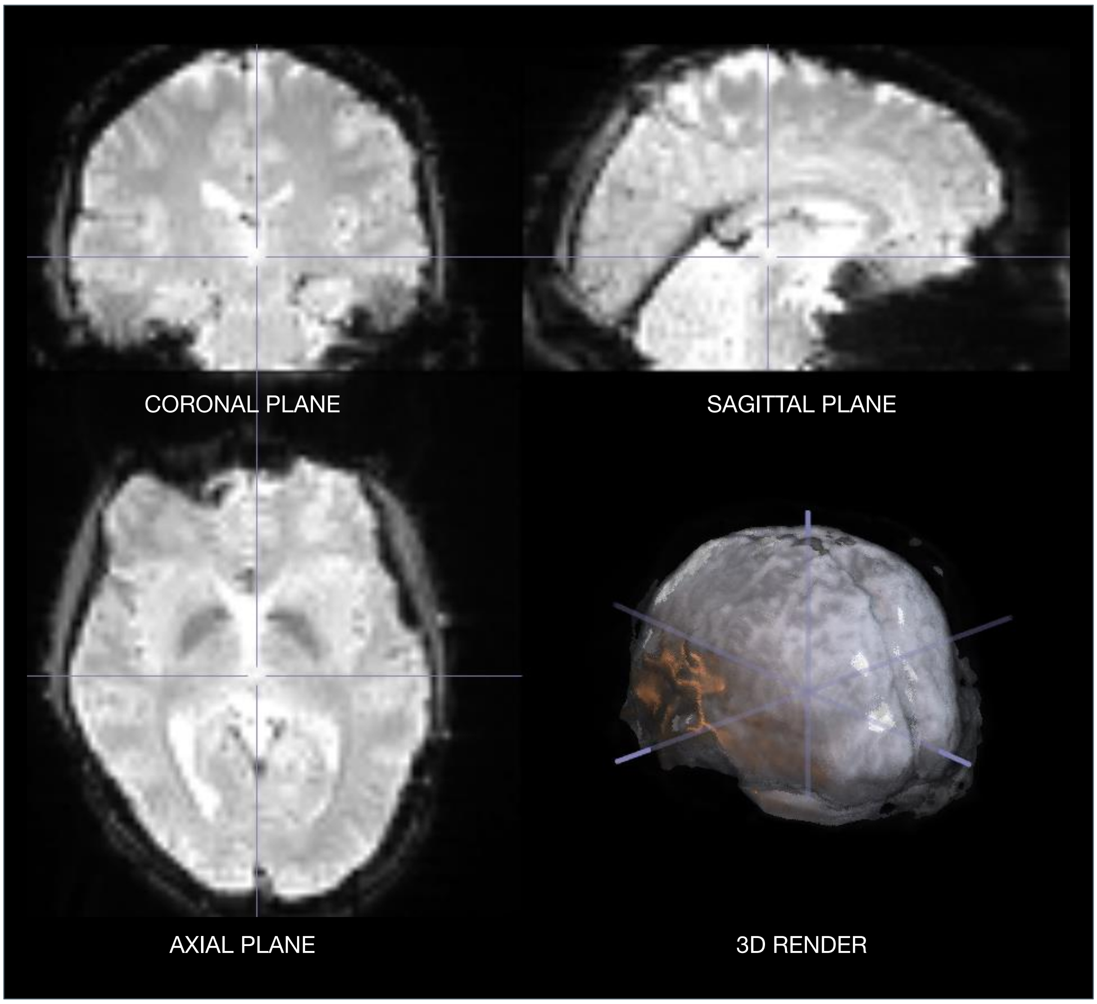


Figure 14. Functional MRI volume

Three planes of an fMRI volume: the coronal plane, the sagittal plane and the axial plane. A 3D render of the functional volume is also shown. These images have been generated using the MRIcroGL software.

Functional images

Another interesting MRI modality is the functional MRI, which generally refers to the imaging of brain activation detectable by changes in regional cerebral blood flow. This image modality combines the effect of T_2 with local inhomogeneities in the magnetic field caused by the presence of other particles in the brain. More specifically, the functional MRI is based in the concentration differences of oxyhemoglobin and deoxyhemoglobin in arterial and venous blood. The oxyhemoglobin is a diamagnetic substance, which means that it is not affected by magnetic fields. On the contrary, deoxyhemoglobin is a paramagnetic substance that perturbates the external magnetic field locally. The presence of paramagnetic deoxyhemoglobin creates local magnetic field distortions in and around blood vessels. These local field disturbances cause nearby stationary and slowly moving spins to have different resonance frequencies and phase shifts. Thus, the regional T_2 relaxation times of brain tissues decrease as the fraction of deoxyhemoglobin increases, weakening the received MR signal. This MRI modality is known as T_2*-weighted images.

Interestingly, the concentration of deoxyhemoglobin varies from 2% in arterial to 40% in venous blood and, in response to brain activations, more oxygenated blood is supplied to the area than is required for the immediate metabolic needs of the brain. This phenomenon causes a local decay in concentration of deoxyhemoglobin, which mitigates the local magnetic field disturbances leading to the subsequent amplification of the recorded MR signal. Thus, the variations in signal intensity in a fMRI image result from this so-called *Blood Oxygen Level Dependent* (BOLD) contrast.

The BOLD signal

The BOLD response is then an indirect measure of brain activity. It is widely accepted that changes in the BOLD are followed by a specify temporal course known as the canonical *Hemodynamic Response Function* (HRF) (Lindquist et al., 2009) (Figure 15). When a certain brain region presents an increased neuronal activity, the oxygen consumption in that region also increases. Due to this instant demand of oxygen the concentration of deoxyhemoglobin in the region increases momentarily with the subsequent attenuation of the BOLD signal (*initial dip*). The origin of this initial dip is not totally clear (X. Hu & Yacoub, 2012), some studies opt for an increment in oxygen consumption (Röther et al., 2002) while others point to an increased concentration of deoxygenated hemoglobin caused by changes in neural activity (X. Hu & Yacoub, 2012). This phenomenon is overcompensated by increasing the oxygenated blood flow to that region in order to cover immediate metabolic needs, increasing the oxyhemoglobin to deoxyhemoglobin ratio which in turn increases the BOLD response (*peak*). This response is sluggish and delayed in time, occurring 6-8 seconds after the initial neural activation. Finally, it is often observed a signal *undershoot* (Logothetis & Wandell, 2004) which origin is also unclear. One possible explanation is the differences in blood flow and blood volume recoveries, increasing the concentration of deoxyhemoglobin present in the vessels (Buxton et al., 1998). Contrary, this fMRI signal undershoot could also be caused by a sustained oxygen consumption after the initial stimulus that caused the neural activation (van Zijl et al., 2012).

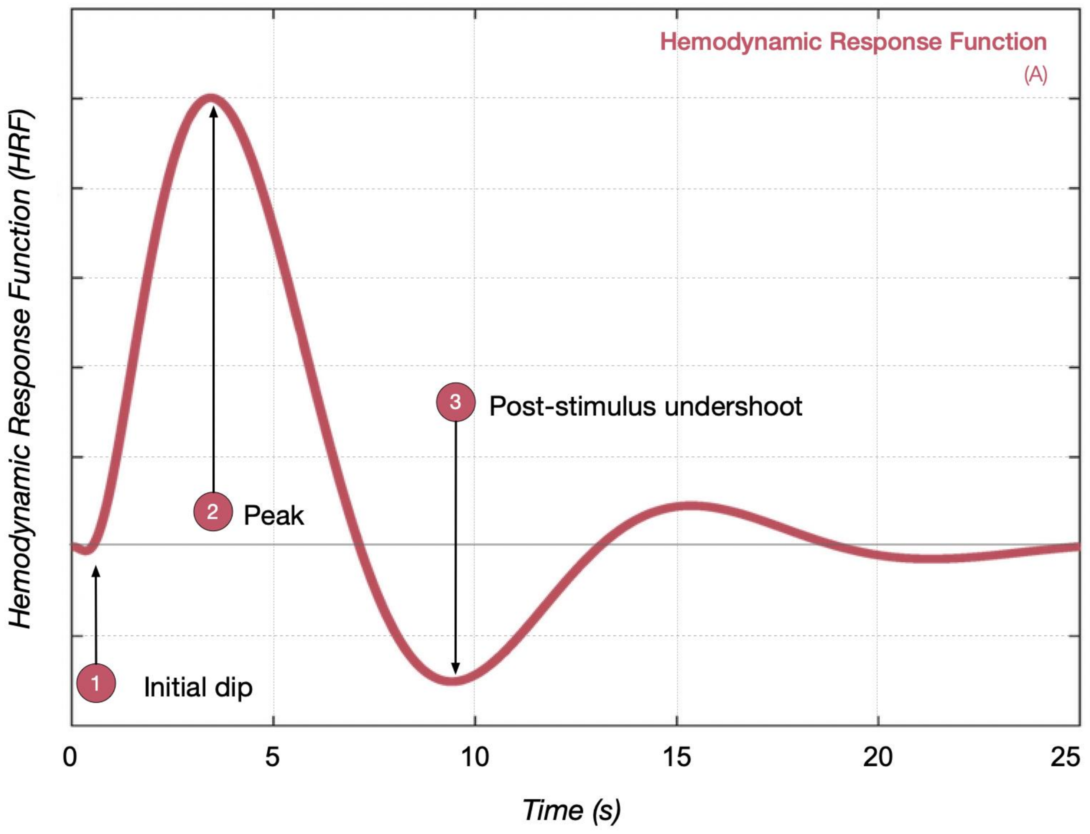


Figure 15. Hemodynamic Response Function to a brief stimulus

The Hemodynamic Response Function represents the transfer function linking neural activity with the functional MRI signal, modeling neurovascular coupling.

Sources of noise in fMRI

The fMRI signal, just as other physiological recordings, is very noisy. The fMRI scanner is sensible to several sources of unwanted noise that need to be controlled in most cases. This noise is usually classified in three categories: the system noise, movement artifacts and physiological noise. The system noise is always present in fMRI recordings. It is caused by the movement of the electrons due to the participant' and the equipment temperature. The higher the intensity of the magnetic field and the temperature, the higher the noise. If the temperature increases, electrons move faster, and more energy is lost. Additionally, the scanner always presents some instabilities, which generates low frequency artifact (signal drift) changing the intensity of the voxels over time. Participants' movements are one of the most concerning sources of noise since in excess, movement artifacts lead to unusable data. During the preprocessing stage moderate motion effects can be partially mitigated but not fully removed. Thus, is extremely important to minimize participants' movements during the acquisition by using head restraints and avoiding long recording sessions with no resting periods. Finally, physiological noise is also generated by the participants, and it is always present in fMRI recordings. Respiration, eye movements and heart beats are common sources of this kind of noise at certain frequencies.

In the following section we describe several preprocessing routines that help to mitigate the effects of this unwanted noise in our data.

2.2.2. Signal preprocessing

As described for electroencephalography signals, the functional magnetic resonance data needs to be prepared and preprocessed for the subsequent statistical and multivariate analyses. Before those analyses, the recorded fMRI data undergoes a series of preprocessing routines aimed at identifying and removing artifacts and validating model assumptions. Thus, the main goals of preprocessing are to minimize the influence of data acquisition and physiological artifacts, to transform the data to meet assumptions of subsequent statistical analyses and to standardize the location of brain regions across subjects to achieve validity and sensitivity in group analyses. The following paragraphs of this section are focused in describing the basis of the standard preprocessing pipeline of fMRI data.

Signal saturation

To begin with, it is common to remove the first three or four volumes of the recorded data to allow the saturation of the signal (Soares et al., 2016). This is because the scanner requires some time to stabilize the gradients as well as the brain tissues require some seconds to reach the excitation.

Slice timing correction

As described previously, one brain volume is acquired a slice at a time and the time it takes is determined by the TR. This basically means that for the same brain volume, the data collected by the scanner reflects brain activations at different time points. This presents a problem since typical multivariate and statistical analyses assume that the data of each volume was recorded at the same time. The slice time correction is a procedure applied to fMRI data to account for these differences in the acquisition time of different slices of the same volume. Temporal interpolation is used to solve these temporal discrepancies in acquisition times, so the amplitude of the recorded signal is estimated at a point when it was not originally collected employing nearby data. This changes the data in such a way that it would seem that the whole volume would have been measured at the same moment in time. The simplest approach to compute slice timing correction is linear interpolation, which can be mathematically expressed as follows:

$$y_n^{(r)} = \frac{y_n\big(t(r) - t(n-1)\big) + y_{n-1}(t(n) - t(r))}{t(n) - t(n-1)}$$

Equation 2

where y is the time series of the slice n at a timepoint $t(n)$ and $t(r)$ is the timepoint of a slice selected as a reference. The reference slice depends on the acquisition order (Figure 16). For ascending/descending acquisition orders, the first/last slice is collected at the beginning and the rest are collected consecutively. In this case, the first/last slice is selected as the reference. For interleaved acquisition order, in which the odd numbered slices are recorded first and then followed by the even numbered slices, the reference slide is usually the center one (e.g., the 15th in a 30-slice volume). Note that this correction introduces temporal smoothing (Sladky et al., 2011) which could lead to undesired artifacts. However, this is not the only approach to compute the slice timing correction, a $sinc$ linear interpolation is usually preferred. This correction uses the Fast Fourier Transformation (FFT) and is mathematically described below:

$$y_n^{(r)} = \sum_{i=-\infty}^{\infty} x_i \cdot sinc\left(\frac{\pi}{TR}(r - iTR)\right)$$

Equation 3

Although this temporal correction presents some undesirable side effects such as worsening artifacts (smearing spikes) or unpredictively interacting with other preprocessing steps such as motion correction, the effectiveness of slice timing correction has been proved in several scenarios (Parker et al., 2017) and is commonly included in fMRI preprocessing routines and software packages such as SPM (Welcome Centre for Human Neuroimaging, 2018), FLS (Jenkinson et al., 2012) or BrainVoyager (Goebel et al., 2006).

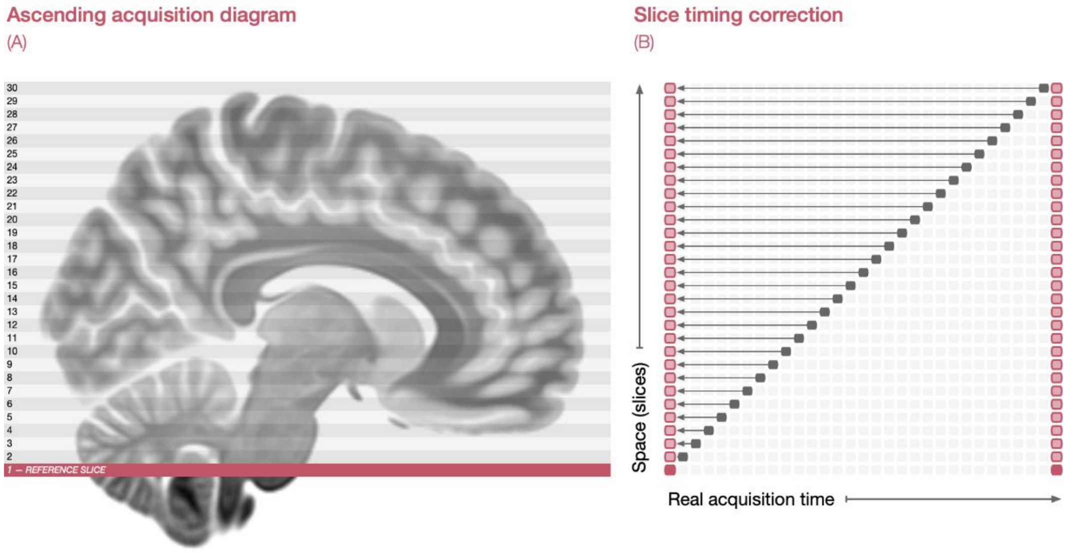


Figure 16. Ascending acquisition order diagram and slice timing correction

The whole brain is not recorded at the same time but in successively individual 2D slices. The required time for a complete volume recording is TR, which is usually around 3 seconds. This functioning presents a problem: for a functional volume of 30 slices and a volume TR of 3 seconds, activations recorded of the last slice are measured almost 3 seconds later than the activations of the first one. Despite the sluggishness of the hemodynamic response, an imprecise specification of time in the order of 3 seconds will lead to suboptimal statistical analysis.

Motion detection and correction

One of the major problems during fMRI acquisitions are the substantial head displacements since the quality of fMRI data is strongly affected by movements in the range of 1-2mm. Some experts recommend discarding datasets including head displacements larger than 5mm. For this reason, the use of head restraint is required, which reduces the participant's head motion considerably, but it does not completely eliminate it. These head movements disturb the homogeneity of the magnetic field, which has been finely-tuned previously to the functional scans for a given head position. The aforementioned would therefore lead to displacements in anatomic brain features (Haller & Bartsch, 2009), temporal variation of the voxel time course (van Dijk et al., 2012) and differences in contrast and BOLD sensitivity (Deichmann et al., 2002; Yarach et al., 2016).

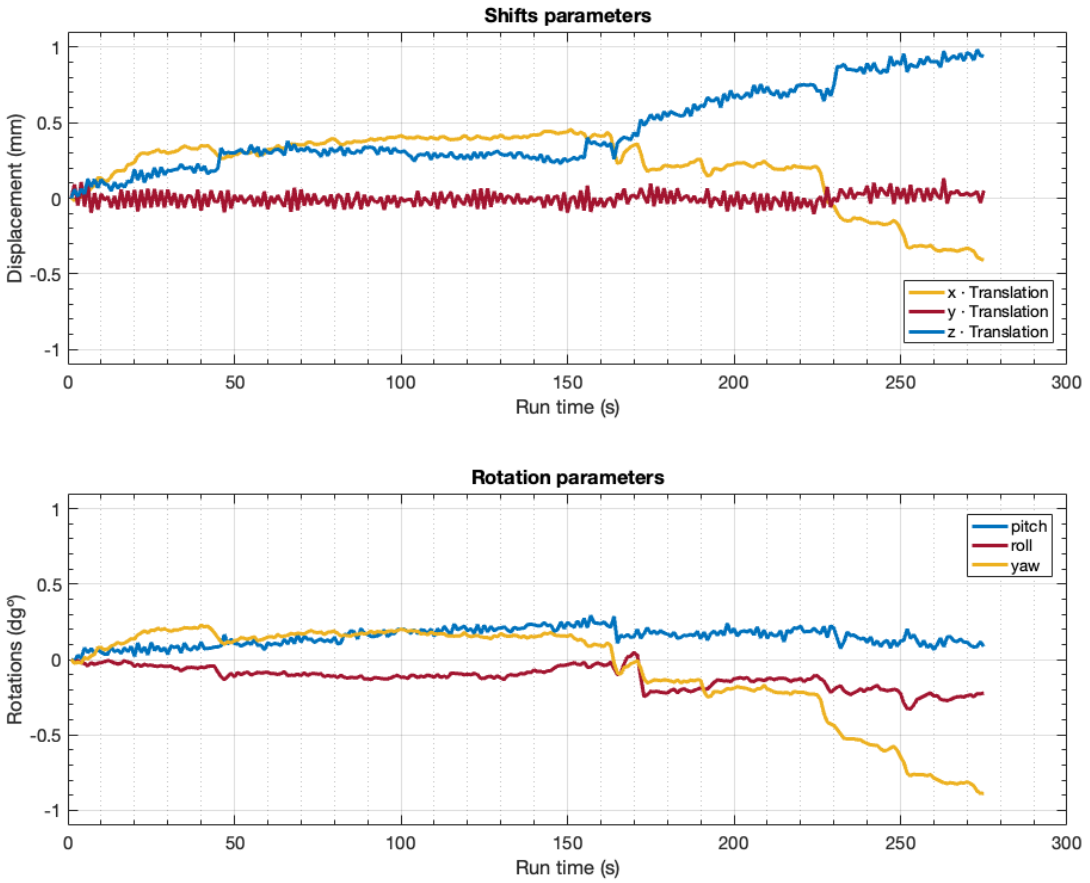


Figure 17. Movement correction parameters during an fMRI experiment

Parameters associated with the motion correction transformation applied to an individual run of fMRI data. Shift parameters represent displacements of the three axes in mm (x, y, and z) and rotation parameters indicate head rotations in degrees (pitch, roll and yaw).

There are several algorithms describing head displacements through six parameters of rotation and translation (Figure 17). Using these six parameters (three of them describe the translation along the x, y and z axes and the other three the rotation around those axes) the motion of any rigid body can be characterized. In order to compute the motion correction procedure a functional volume of a run (or an average of volumes) is selected as a reference and the rest of functional volumes are aligned to it. The aforementioned six parameters are iteratively estimated by an optimization algorithm in order to achieve the translation and rotation which provides the better alignment between the source volume and the reference volume. The iterative algorithm stops when no further improvements are obtained, in other words, when a minimum of the registration error is found.

However, this assumption of the rigid body is not always valid, since abrupt movements can occur at any moment in time, and even in the middle of a volume acquisition, leading to intra-volume displacements. In order to account for these intra-volume misalignments, several fine-grained approaches have been proposed, performing a slice-by-slice registration and considering the brain tissue to which each voxel belong to (Beall & Lowe, 2014; Bhagalia & Kim, 2008). Fortunately, head movements from volume to volume are typically small and the assumption of a moving rigid body is largely valid. Finally, including these six-motion parameter timeseries as covariates when the subsequent General Lineal Model is estimated is another common practice to mitigate the effects of head displacement. This approach increases the sensitivity for detecting the desired effect while the absolute error in the model estimation decreases (Johnstone et al., 2006), provided that there is no correlation between the task design and head displacements. In this case modeling the motion parameter could remove task-associated activations.

Spatial normalization to a common space

Brain shapes and sizes differ greatly among individuals. Therefore, establishing a spatial correspondence between different brains is a key step in order to generalize and average results across subjects, since this procedure reduces their brain variabilities in shape and size. Spatial normalization is not only important for whole-brain group analyses but also for reporting individual activations in a common spatial coordinate system. Two of the most employed standard templates are the MNI space and the Talairach space. Firstly, the Talairach space was defined as a three-dimensional cartesian space based on several anatomical landmarks (Talairach & Szikla, 1980) and then an atlas from Brodmann's areas was created to improve the location in the new coordinate system (Talairach, 1988). Nowadays the Talairach space does not play an important role in modern neuroscience anymore since it has several limitations (since it was constructed from a single subject, it does not represent the population). On the contrary, the MNI space, which was created from 305 transformed scans (Brett et al., 2001) to match an average brain computed from 241 scans (Evans et al., 1993), is commonly used in most modern fMRI studies (Figure

18). Brain normalization in the volume space is usually computed by warping each brain into a common space. To do so, several algorithms first compute an affine transformation based on twelve parameters such as translation, rotation, scale, squeeze, etc., and then a series of nonlinear transformations. After computing spatial normalization, a point in the common space identified by its x, y, z coordinates is assumed to refer to a similar region in any brain normalized volume.

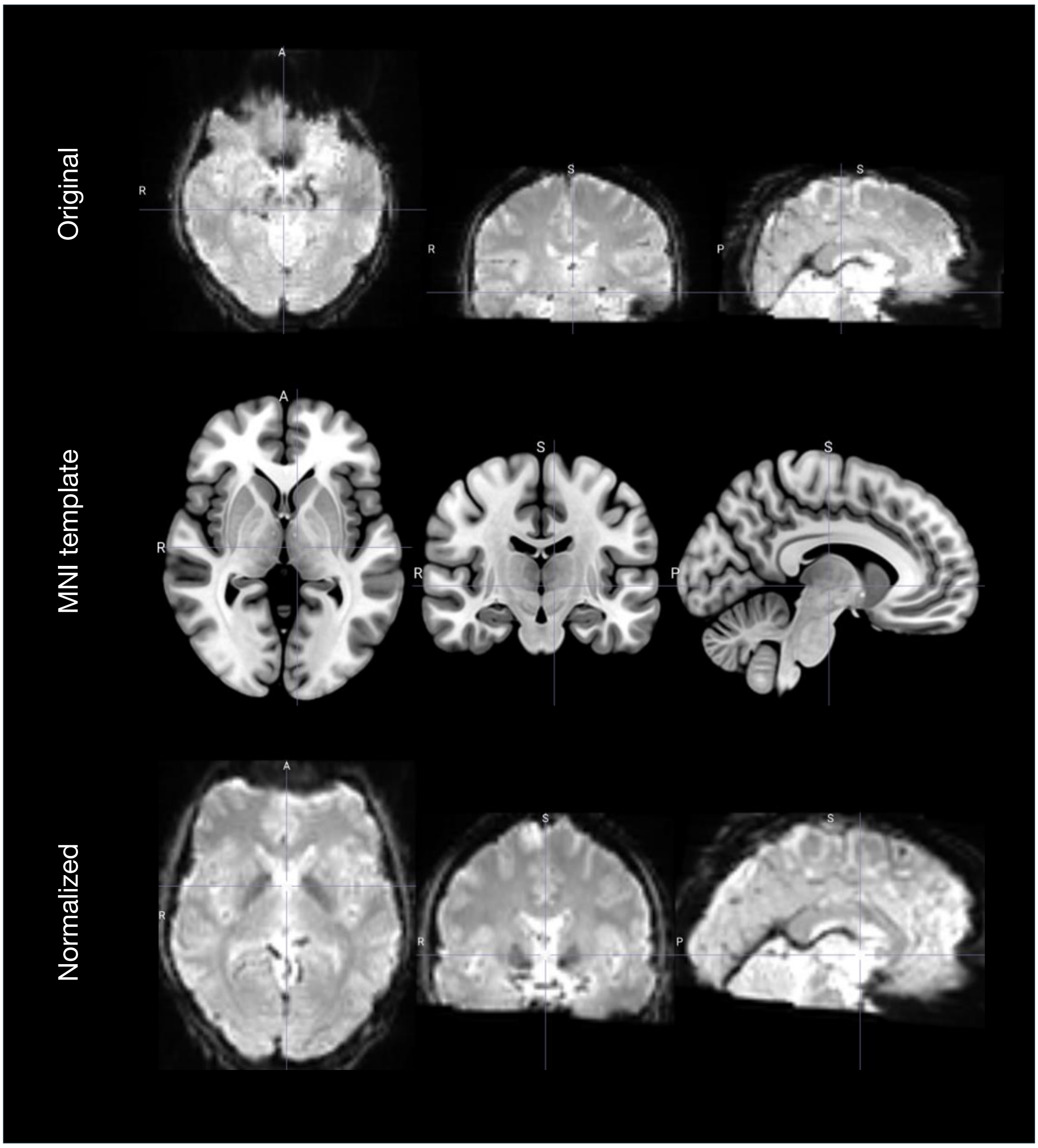


Figure 18. Normalization process of a subject’s functional image

Different images during the normalization process in SPM12: the original volume after the slice timing correction and corregistration, the MNI152 template (extracted from MRIcroGL), and the normalized image.

Spatial smoothing

The spatial smoothing step is also included in most fMRI preprocessing pipelines and it is broadly accepted in the neuroscience community. This is because fMRI data inherently show spatial correlations due to functional similarities of adjacent brain regions and the blurring of the vascular system. When a spatial smoothing routine is applied, fMRI datapoints are averaged with their neighbors, leading to a low-pass filter effect which blur the higher frequencies while enhancing the lower ones, and also increasing the spatial correlation of the data.

The computation of this preprocessing procedure presents several advantages. Firstly, the smoothing can reduce the thermal noise recorded during the acquisition, and thereby increasing the signal-to-noise ratio (Molloy et al., 2014; Triantafyllou et al., 2006; Welvaert & Rosseel, 2013) and the sensitivity. In addition, the spatial smoothing reduces the anatomical and functional differences between subjects, which also increases the overlapping of activated brain regions between them, even after computing a spatial realignment and normalization to a common space. The validity of statistical analysis is also improved after the spatial smoothing since this procedure normalizes the error distribution. Additionally, spatial smoothing modifies the data to satisfy the Gaussian Random Field Theory (Worsley et al., 1992), enabling hypothesis testing and the subsequent multiple comparisons correction (Benjamini & Hochberg, 1995). Despite all these benefits, spatial smoothing could reduce the decoding performance in classification analysis (Hendriks et al., 2017). In that case, this preprocessing routine is usually avoided.

The spatial smoothing is usually computed as a convolution of the MR signal with a specific with gaussian kernel, which results in a weighted sum of the value of adjacent voxels in the original volume. According to this approach, voxels closer to the center of the filter contribute largely to the final value of the smoothed voxel. The width of this gaussian function kernel determines how much the data is smoothed and is expressed in terms of the Full Width at Half Maximum (FWHM). The larger the width of the kernel, the more blurred the resulting volume (Figure 19). The adequate value for the filter width varies depending on several variables. For example, when the filter width matches the expected signal width the SNR reaches its maximum (according to the matched filter theorem). Therefore, the selected filter width depends on the experimental design and the functional brain areas under investigation, but practically, an optimum value for the FWHM should be at least twice the voxel size (Worsley & Friston, 1995).

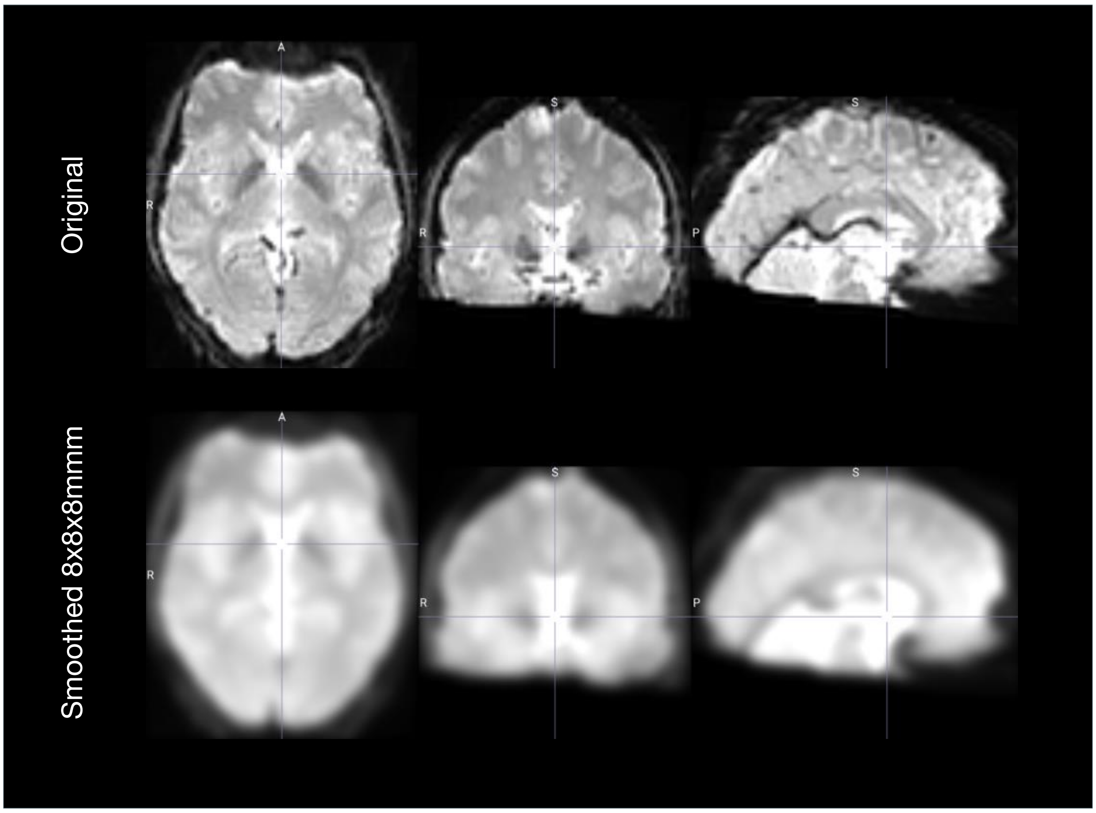


Figure 19. Smoothing process for a subject's functional image

Different images during the smoothing process in SPM12: the original volume after the slice timing correction, corregistration and normalization, and the smoothed image (FWHM = 8mm).

2.2.3. General Lineal Model Analysis

The localization of neural correlates of cognitive, sensory and motor processes is one of the most important challenges in fMRI studies. Therefore, the neuroscientists need statistical tools to identify increments or decrements in brain activation related to certain experimental conditions when compared to other conditions. The General Lineal Model (GLM) plays a very important role in this matter, and it is considered the keystone of the statistical data analysis from the early days of fMRI (Friston et al., 1994). In this section, we briefly describe the mathematics of the univariate GLM analysis in fMRI scenarios. The General Lineal Model is mathematically identical to a multiple regression analysis, in which the variability of a dependent variable (the empirical fMRI timeseries of an individual voxel) is explained in terms of a linear combination (weighted sum) of different reference functions (also called regressors, predictors, explanatory variables, etc.). Thus, the observed fMRI timeseries of an individual voxel is modeled as follows:

$$y_1 = \beta_0 + \beta_1 X_{11} + \beta_2 X_{12} + \cdots + \beta_p X_{1p} + \epsilon_1$$

$$y_2 = \beta_0 + \beta_1 X_{21} + \beta_2 X_{22} + \cdots + \beta_p X_{2p} + \epsilon_2$$

$$\vdots$$

$$y_n = \beta_0 + \beta_1 X_{n1} + \beta_2 X_{n2} + \cdots + \beta_p X_{np} + \epsilon_n$$

Equation 4

These equations can also be expressed in matrix notation:

$$\begin{bmatrix} y_1 \\ \vdots \\ \vdots \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & X_{11} & \cdots & \cdots & \cdots & X_{1p} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & X_{n1} & \cdots & \cdots & \cdots & X_{np} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \vdots \\ \beta_p \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \vdots \\ \vdots \\ \vdots \\ \epsilon_p \end{bmatrix}$$

Equation 5

Or using the simplified notation:

$$\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\epsilon}$$

Equation 6

The variable $\boldsymbol{y}$ (known) at the left corresponds to the observed data (e.g., the measured fMRI signal time course of a single voxel). The matrix $\boldsymbol{X}$ is also called the design matrix or the model and contains the predictor time courses as column vectors. These predictor time courses are the expected (ideal) fMRI responses for different conditions of the experimental paradigm. The beta values $\boldsymbol{\beta}$ (also known as beta maps or beta images) quantify the contribution of each regressor in explaining the observed fMRI time course. Normally, a large positive/negative beta value indicates that this particular voxel presents strong activation/deactivation during the modeled experimental condition when compared to the baseline. The value of β_0 is called the *intercept* or constant and it typically represents the signal level of the baseline condition. It is very important to include the intercept in the design matrix. This allows the other regressors to model small condition-related fluctuations as increases or decreases according to the baseline signal level. However, the absolute value is not informative. Finally, the variable $\boldsymbol{\epsilon}$ represents the residuals of the model, also called noise or prediction error, which account for the remaining unexplained data. In other words, the prediction error is the difference between the observed fMRI data and its estimation. Rearranging the previous equation system:

$$\boldsymbol{\epsilon} = \boldsymbol{y} - \boldsymbol{X\beta} = \boldsymbol{y} - \widehat{\boldsymbol{Y}}$$

Equation 7

The next logical step is to find the beta values that minimize the total error. However, note that the error values can be positive or negative, so the GLM procedure estimates the beta values which minimize the sum of the squared error values. This is the Least-Squared estimation (Aitken, 1936) and there are many variations in fMRI literature (Monti, 2011; Waldorp, 2009). Mathematically, this can be expressed:

$$\boldsymbol{\epsilon^T\epsilon} = (\boldsymbol{y} - \boldsymbol{X\beta})^{\boldsymbol{T}}(\boldsymbol{y} - \boldsymbol{X\beta}) \rightarrow min$$

Equation 8

Finally, according to (Poline & Brett, 2012) the descriptions in the optimal beta values can be obtained non-iteratively as follows:

$$\boldsymbol{\beta} = (\boldsymbol{X^TX})^{-\boldsymbol{1}}\boldsymbol{X^Ty}$$

Equation 9

At this point only one question remains unclear: *how the predictor time courses are modeled?* As mentioned before, these time courses represent the idealized fMRI response expected for each experimental condition and are obtained convolving the canonical Hemodynamic Response Function (HRF) with a box-car timeseries coding when the modeled experimental condition is active (

Figure 20). The box-car timeseries for a specific experimental condition is set to one/zero "1/0" for a specific time point if the experimental condition is active/inactive at this time point.

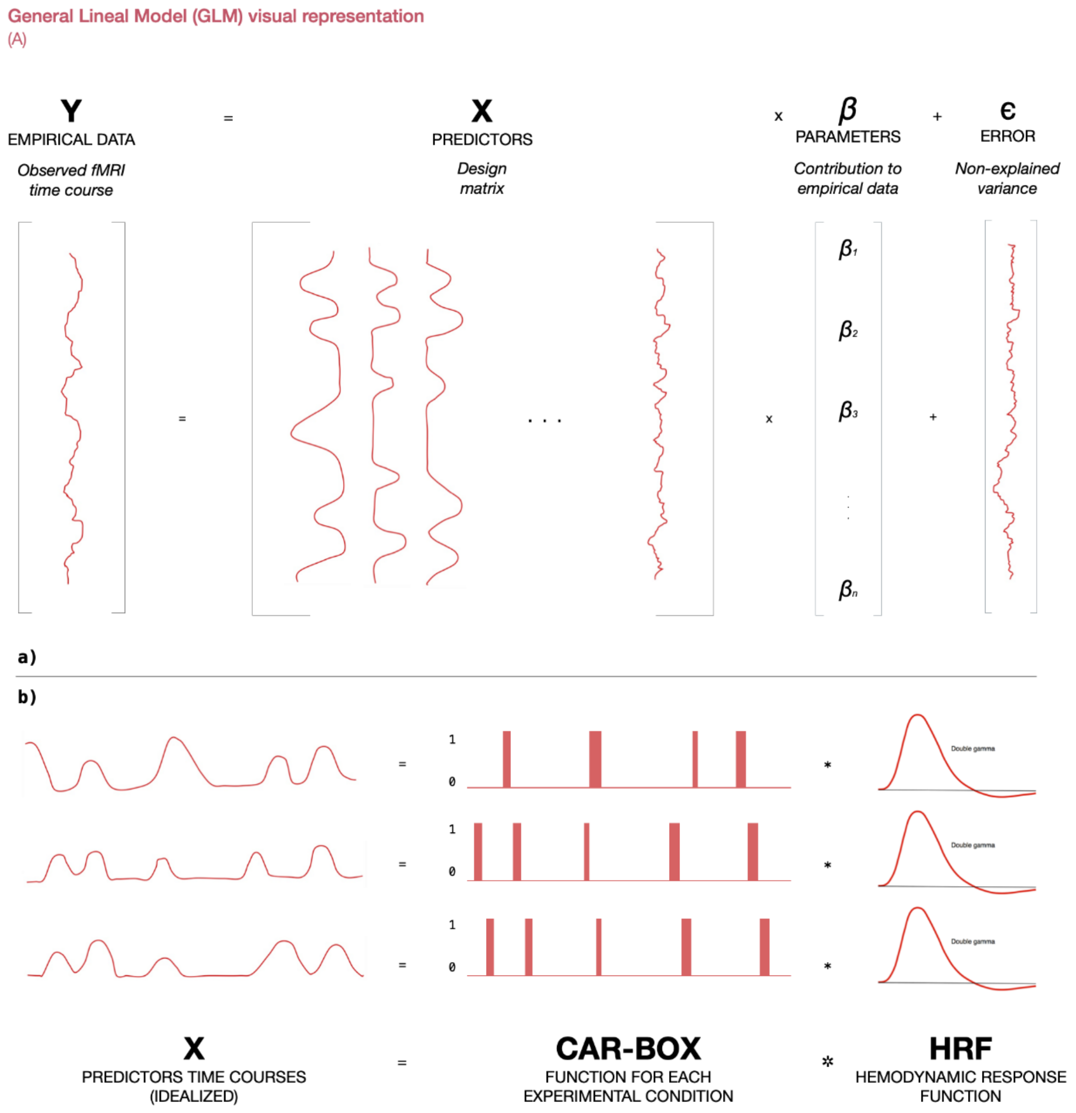

General Lineal Model (GLM) visual representation
(A)
Y
EMPIRICAL DATA
Observed fMRI time course
=
X
PREDICTORS
Design matrix
x
β
PARAMETERS
Contribution to empirical data
+
ϵ
ERROR
Non-explained variance
=
. . .
x
β_1
β_2
β_3
β_n
+
a)
b)
=
1
0
*
Double gamma
=
1
0
*
Double gamma
=
1
0
*
Double gamma
X
PREDICTORS TIME COURSES (IDEALIZED)
=
CAR-BOX
FUNCTION FOR EACH EXPERIMENTAL CONDITION
*
HRF
HEMODYNAMIC RESPONSE FUNCTION

Figure 20. General Lineal Model diagram for a specific voxel

General linear model diagram (a): Beta values are estimated minimizing the sum of squared errors. The predictors time courses are obtained by convolving the canonical HRF with a box-car function of each experimental condition (b).

Chapter 3. Multivariate Pattern Analysis on neuroimaging data

3.1. Introduction

The use of machine learning algorithms in the Neuroscience field has revolutionized the way we study and analyze the brain function. Since 1939, when the first Event-Related Potential was recorded in awake humans at the Harvard Medical School (H. Davis et al., 1939), the study of the brain function mostly relied on univariate paradigms. Univariate approaches have been proved a useful and reliable tool to study and analyze brain data for decades. However, most neuroimaging modalities provide data with an outstanding richness in information, and univariate analyses do not have the potential to unveil this fine-grained information contained in the data. With the recent development of science and technology, several multivariate analytical approaches have emerged to revolutionize the scientific panorama in a cross-cutting basis. The application of these powerful mathematical frameworks to brain imaging is known as Multivariate Pattern Analyses (MVPA, Norman et al., 2006). These MVPA techniques have been successfully applied to several neuroimaging modalities such as Positron Emission Tomography (PET), Single Positron Emission Computed Tomography (SPECT) or Structural Magnetic Resonance Imaging (sMRI) among others. In the Cognitive Neuroscience field, however, they have been commonly applied to functional Magnetic Resonance Imaging (fMRI) and, more recently, to Magneto and Electroencephalography (M/EEG).

Contrary to univariate analyses, the MVPA extracts information from a set of multivariate features, such as the entire set of recording electrodes on EEG studies or the voxels contained in a brain volume on fMRI studies. Thus, multivariate analyses have been proven to be more sensitive in detecting subtle changes in brain activations than their univariate counterparts: the contrast analyses based on the General Lineal Model on fMRI studies or the ERPs and time-frequency analyses on EEG studies (Haxby, 2012).

This chapter provides a general overview of these multivariate approaches, briefly defining two broadly employed approximations, the classification analysis (also known as decoding analysis) and the Representational Similarity Analysis (RSA).

3.2. Classification Analysis

3.2.1. Definition

Classification analyses are also known as decoding analyses. This is because the brain activity recorded by a neuroimaging modality from the participant’s brain is treated as a pattern with encoded information which these methodologies aim at decoding (Yang et al., 2012). Decoding analyses are based on the idea that different cognitive processes are associated with distinct patterns of neural activity across different regions of the brain. These multivariate approaches employ classical machine-learning algorithms such as Support Vector Machines to identify patterns and relationships between neural activity and different cognitive processes. The first prototype of a classification analysis was developed for the investigation of the functional architecture for face and object recognition in ventral temporal cortex (Haxby, 2001, 2012). From that moment, decoding analyses rapidly burst into the Cognitive Neuroscience scene, especially in fMRI studies covering several domains such as perception (Kravitz et al., 2010; Mur et al., 2010) , emotion (Ethofer et al., 2009; Gomez et al., 2011; Kotz et al., 2013), memory (Chadwick et al., 2010; Schultz, 2010) or decision-making (Bode et al., 2011; Haynes, 2011; Soon et al., 2008), among others. Due to its intrinsic potential, time-resolved decoding analyses are also gaining popularity in recent years for analyzing the EEG signal in a wide range of disciplines (Ashton et al., 2022; Berberyan et al., 2021; Carlson et al., 2019; Takacs et al., 2020). The main goal of this section is to provide a general overview of how decoding analyses are defined and implemented. More detailed and specific explanation of decoding analyses are provided in Section II. Contribution of this thesis.

Thus, the basic stages of a classification analysis in a classic Neuroscience experiment are listed below.

- *Data collection*: We need to collect the neural activity from different regions of the brain employing certain neuroimaging techniques. Usually, this data collection is carried out while the participant is engaged in a cognitive task or presented with different stimuli.
- *Feature extraction and selection*: At this stage, patterns of neural activity are extracted and selected to train and test the classifier. These activity patterns are known as *feature vectors* or simply *features*. For example, in EEG experiments a

feature vector usually refers to the raw voltage recorded for each electrode at certain time point.

- *Preprocessing*: Before training the classifier some preprocessing steps such as data normalization or data smoothing are usually performed.
- *Training the classification model*: A machine-learning-based classification algorithm is trained employing the feature vectors previously extracted. This algorithm learns from the input data and creates the best decision boundary that separates this training set in two (or more) classes.
- *Evaluating the effectiveness of the classifier*: The trained classification model is now tested employing unseen data to evaluate its ability to perform correctly in different scenarios.
- *Statistical significance of the results*: In order to assess the statistical significance of the obtained results, a cluster-based permutation test is usually computed. This procedure accounts for the multiple comparisons problem occurring in neuroimage studies due to the large number of independent statistical tests required.
- *Interpreting the result*: Last but not least, the classification results should be interpreted, for example, to identify regions of the brain that are associated with specific cognitive processes or behavioral outcomes.

3.2.2. Classification algorithms

There is a myriad of mathematical algorithms designed to separate or classify data of different nature according to its intrinsic characteristics. In a first stage, supervised classification algorithms are fed with an input dataset containing known data belonging to two or more different classes. In Cognitive Neuroscience these two classes usually represent the brain activity patterns elicited by two different stimuli which the participant was exposed to. Then, the classification model learns the internal structure of the input data and creates the best decision boundary that separates this training set in two or more classes. Finally, once the model is trained and the decision boundary established, it can be fed with new and unknown data that will be automatically classified into one of the two (or more) classes. This decision is made according to the location of the unknown data in relation to the decision boundary established in the training stage.

Classification algorithms are commonly divided into two main categories according to the shape of the decision boundary: the linear and non-linear classifiers. As the name suggests, linear classifiers establish a linear decision boundary (or a hyperplane in high-dimensional spaces) and are the most used in neuroscience (Figure 21). The main reason for that is their simplicity, generalization ability and their relatively low computational load when compared to non-linear counterparts. For example, when employing linear classifiers each feature of the data (electrodes or voxels on EEG and fMRI studies respectively) has a weight value associated indicating the feature contribution to the final classification accuracy. Two of the most common linear classifiers employed in the Neuroscience field are based on Linear Discriminant Analysis (LDA) and Support Vector Machines (SVM, Boser et al., 1992; Cortes & Vapnik, 1995). The LDA defines a decision boundary that optimally discriminates the covariance matrices of the two classes while maximizing the withing-class variance (Bishop & Nasrabadi, 2006). On the other hand, SVM defines a decision boundary trying to maximize the margin between the training patterns. Finally, non-linear classifiers establish more complex decision boundaries (quadratic, polynomic, etc.), which usually leads to optimal classification accuracies but also to a harder interpretation on the results, increased computational load, etc. For these reasons the application of these non-linear approaches is not broadly extended.

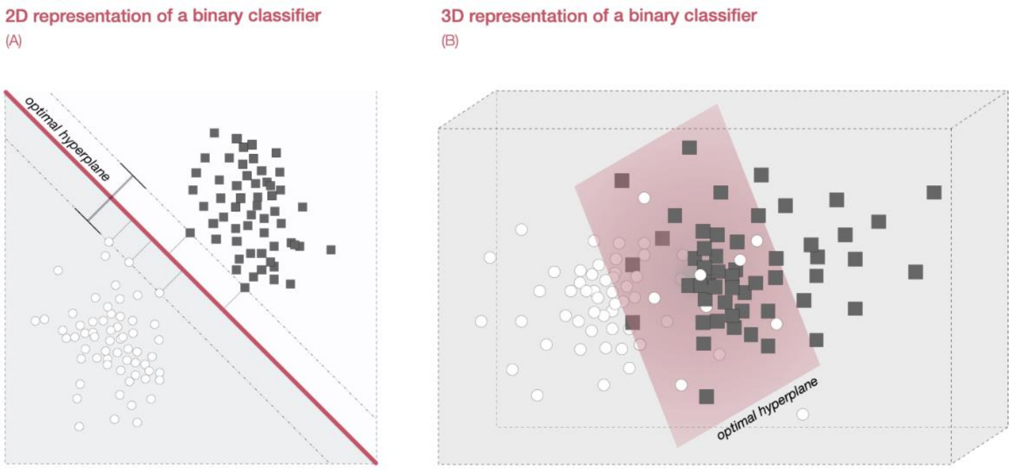

2D representation of a binary classifier
(A)
optimal hyperplane
3D representation of a binary classifier
(B)
optimal hyperplane

Figure 21. Linear SVM classifiers

Visual representation of a classifier. As shown, several simulated observations belonging to two different classes (circles and squares) are separated by a linear decision boundary or hyperplane maximizing the margin between these classes.

3.2.3. Model validation: Cross-validation approaches

As mentioned before, the main goal of classification algorithms is to define a decision boundary that correctly classify new observations based on previous knowledge. To evaluate the performance of a classification model, it should be fed with new observations that have not been previously used for the training stage. This is crucial to assure that the classification performance is not biased. In other words, after training the classification model we need an indication of how well the learner will generalize to an unseen data set. Therefore, one could ask: *which data of a neuroimaging dataset should be used for training and which for testing?* Cross-validation approaches are designed to solve this problem, since they iteratively divide the whole dataset into training and testing subdivisions. There are several techniques of cross-validation specifically designed and adapted to the nature of the neuroimaging dataset. Here we provide a general overview of two of the most employed approaches in neuroscience: the leave-n-out and the k-fold cross-validation.

Leave-N-Out Cross-Validation

As its name suggests, this approach leaves N observations out of the training set. Thus, these N observations will be employed to validate the effectiveness of the classification model. This procedure is iteratively repeated for all combinations in which the original dataset can be subdivided. Therefore, the overall classification performance is calculated by averaging all the correct rates obtained in each iteration. A particular case of this cross-validation approach in which $n = 1$ is shown in Figure 22a. This particular case is known as Leave-One-Out Cross-Validation so only one observation is excluded from the training data in each iteration.

This approach is referred as an exhaustive cross-validation method since all the possible combinations of training and testing subdivision are evaluated. However, it could lead to high computational loads specially for large datasets. Note that the term observation could refer to different ideas depending on the nature of the data and the experimental paradigm we work with. For example, the term observation could represent a single run or an individual participant on fMRI studies. On the other hand, it also could represent an individual trial on an EEG dataset.

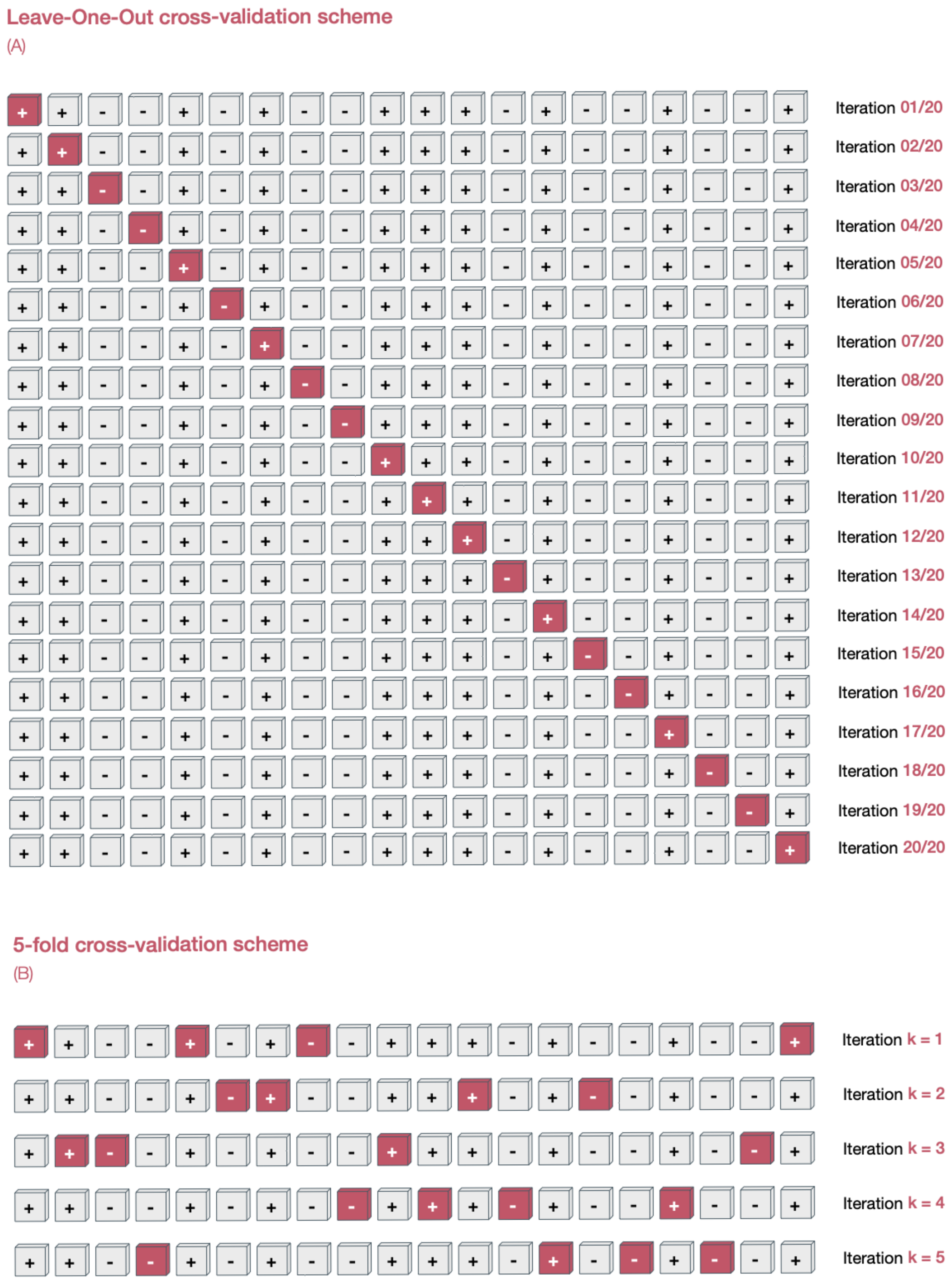


Figure 22. Cross-validation schemes

Leave-one-out and 5-fold cross-classification schemes. Positive and negative observations are represented in the picture using the +/- symbols. For each iteration grey observations represent the training set while the red ones represent the test set.

K-fold Cross-Validation

The K-fold approach divides the data in K partitions. Therefore, the classification model is trained employing $K-1$ partitions and validated with the remaining one. This procedure is then repeated K times, such that each time, one of the K subsets is used as the validation set and the rest form the training set. In this way, each observation will fall in the validation set only in one iteration and $K-1$ in the training set. As before, the overall classification performance is calculated by averaging all the correct rates obtained on each iteration. This procedure reduces the bias and the variance of the results since it uses most of the data for training the model but also for validation. The Figure 22b shows a graphical representation of this cross-validation procedure for $k=5$. Although there is not a golden rule for defining the number of folds, values of $k=5$ or $k=10$ are mostly employed.

If the input dataset is imbalanced, which means that the number of observations belonging to the positive and negative class differs significatively, the k-fold approach can be slightly modified to avoid bias in the results. In this particular case known as stratified K-Fold, the percentage of positive and negative observations is approximately the same for each fold.

3.2.4. Derived analyses

A classification analysis is not an unalterable single-purpose algorithm but a general and easily extendable analytical framework. Therefore, along the Section II of this document we describe several multivariate analyses derived from the original idea of decoding. For instance, *Multivariate Cross-Classification analyses* (MVCC) aim to verify whether the information encoded in brain activity patterns is consistent across different condition or contexts. Thus, classifiers are trained with data belonging to one condition (or task), while the validation stage is performed employing activity patterns observed in another condition (or task). On the other hand, *Temporal Generalization analyses* test the stability of brain activity patterns in time. To do so, on EEG studies the classifier is trained with the EEG activity recorded at certain time point and tested in all the remaining time window. This process is repeated for each time point leading to the so-called *Temporal Generalization Matrix*. Further details about different classification analyses are provided in the Chapter 4 and Chapter 5 of this thesis.

3.3. Representational Similarity Analysis

3.3.1. Definition

The Representational Similarity Analysis (RSA) technique was first proposed by Kriegeskorte et al. in 2008 giving to MVPA analyses a broader scope. The aim of this technique was to bridge the three branches of systems neuroscience: behavioral experimentation, brain-activity experimentation, and computational modeling. The RSA is based on a very simple idea: the patterns of neural activity evoked in the brain should be similar when the brain is processing similar stimuli or performing similar tasks. Thus, the concept of voxel/electrode activation pattern is generalized to a more abstract similarity pattern that can be measured among different conditions and stimuli. To do so, the raw activity patterns are pairwise compared and transformed in dissimilarity matrices which are comparable across brain regions, temporal points, individuals, neuroimaging modalities or even animal species

The basic stages of an RSA in a classic Neuroscience experiment are listed below.

- *Data collection*: We need to collect the neural activity from different regions of the brain employing certain neuroimaging technique. Usually, this data collection is carried out while the participant is engaged in a cognitive task or presented with different stimuli.
- *Extract representations*: The recorded activity patterns induced by a set of stimuli are extracted from the original dataset. These activity patterns are usually known as *representations*. Thus, different stimuli should evoke different representations.
- *Calculate the Representational Dissimilarity Matrices*: Also known as first-level analysis. Representational Dissimilarity Matrices (RDMs) are calculated computing pairwise comparisons between brain patterns evoked by each experimental condition or stimulus. Therefore, this matrix arrangement visually reveal how different or distinguishable different activation patterns are in a specific brain region (fMRI) or time point (EEG).
- *Second Level Analysis*: The main goal of the second-level analysis is to compare the previously calculated RDMs between each other or to theoretical models based on previous knowledge. Since RSA is a very flexible methodology, the second order analysis could be computed differently depending on the experimental paradigm and the tested hypothesis.

- *Statistical significance of the results*: In order to assess the statistical significance of the obtained results, a data permutation-based test is usually performed. Thus, the original data is shuffled, and the analyses are repeated to obtain the empirical null distribution of the dataset. This procedure account for the multiple comparisons problem occurring in neuroimage studies due to the large number of independent statistical tests required.
- *Interpreting the result*: Finally, the obtained results should be interpreted to, for example, identify regions of the brain that are functionally related, or to compare representations across different tasks, stimuli, or even between different groups of participants.

3.3.2. Types of RSA analyses

As stated before, two different levels of pattern analysis are defined in this analytical approach, the fist and the second-level analysis. Here we provide a general overview of both levels, since the intrinsic versatility of the RSA framework allows for a wide range of implementations.

First-level RSA

Representational Dissimilarity Matrices are the cornerstone of the RSA technique. These matrices present pairwise comparisons between brain-activity patterns associated to a set of experimental conditions. Since RDMs are generated comparing the activity patterns elicited by different stimuli, they can be extracted no matter the nature of the original data. To compute this pairwise comparisons several measures are employed, including the Euclidean distance or the Pearson’s correlation among others.

As an example, Figure 23 depicts how RDMs are constructed from different data sources: EEG and fMRI. For temporal-accurate neuroimaging modalities a neural RDM is usually generated for each time point and participant. Contrary, in other neuroimaging modalities such as fMRI there are different approaches to compute this first-level analysis. The simplest approach is known as *ROI analysis,* and it requires *a priori* knowledge or a strong hypothesis about the brain regions being involved in a specific cognitive process. Thus, the brain-activity patterns are masked and used to construct the RDM from each specific brain region of interest. On the other hand, an exploratory brute-force analysis can also be computed. This analysis is known as *whole-brain searchlight RSA*, and it selects certain number of voxels contained inside a small sphere to construct the RDM for the center voxel of the sphere. Then, this sphere iterates across the whole brain volume generating a RDM for each voxel in the brain.

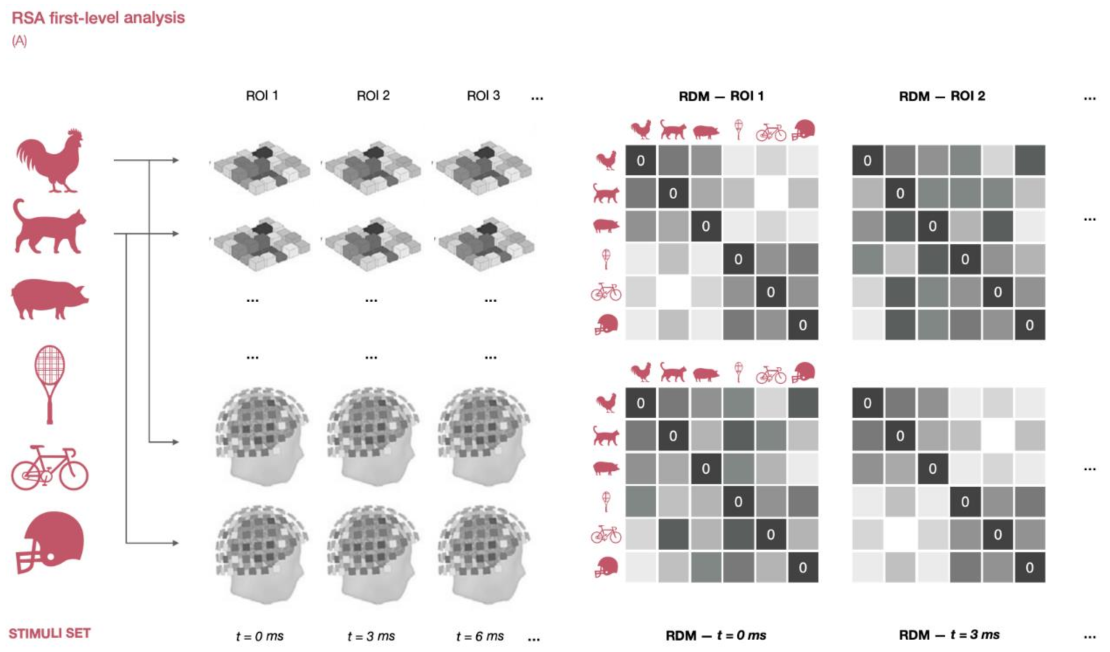


Figure 23. RSA: First-level analysis representation

Visual representation of the RSA first-level analysis. Activity patterns elicited by different stimuli (animated vs inanimate) are extracted from different regions of the brain (fMRI) or different time points (EEG). These representations are then pairwise compared resulting in a Representational Dissimilarity Matrix for each region of the brain and time point.

Additionally, conceptual or theoretical RDMs can also be defined based on a presumed relationship between the experimental conditions. This synthetic RDM reflects the expected differences between the stimuli set along a feature of interest (e.g., faces vs. non-faces, animate vs. inanimate objects, etc.). Thus, the dissimilarity values between stimuli belonging to the same category (e.g., animate-animate or inanimate-inanimate) are coded with a “0”. Contrary, the dissimilarity values between stimuli belonging to different categories (e.g., animate-inanimate) are coded with a “1” (Figure 24).

By definition, RDMs are square matrices horizontally and vertically indexed by the set of N stimuli, leading to a $N \times N$ symmetric matrix along its diagonal. Zero values at the diagonal represent the dissimilarity obtained by comparing each experimental condition to itself while values outside the diagonal result from pairwise comparisons between all possible combinations of two different experimental conditions.

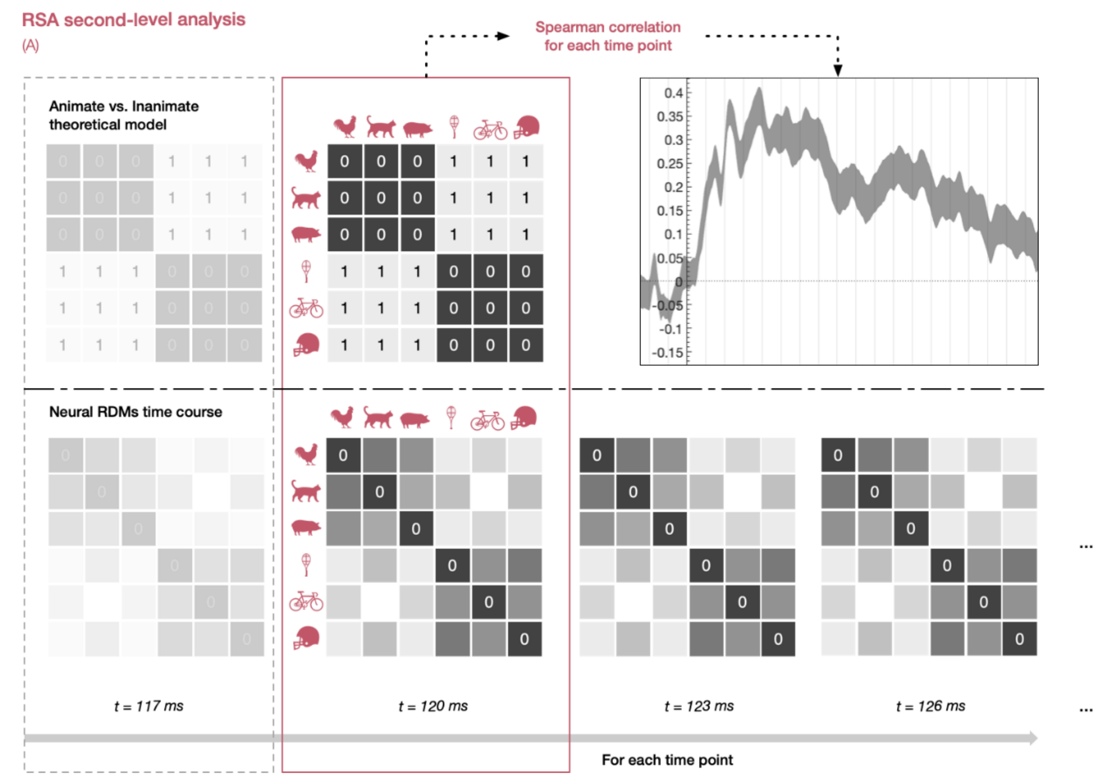


Figure 24. RSA: Second-level analysis representation

Visual representation of the RSA second-level analysis. On this simulated example a theoretical model for object animacy is correlated with the RDMs time course of an EEG experiment to study the temporal dynamics of visual processing.

Second level RSA

The second-level analysis is based on one of the greatest advantages of the RSA: the ability to compare different RDMs. Depending on the researcher’s hypothesis, several approaches have been defined in order to compare those RDMs. For example, Figure 24 depicts a second-level analysis comparing a pre-defined theoretical model to empirical

RDMs extracted from neural data. However, this is not the only possible scenario. Neural RDMs of multiple ROIs can also be compared to see which region better represents a category. For example, an RDM in the inferior temporal cortex would better represent animate as compared to inanimate objects than a neural RDM from early visual cortex. This is due to the fact that the theoretical model RDM for animacy is more similar to the neural RDM in the inferior temporal cortex than in the early visual cortex (Kriegeskorte et al., 2008; Popal et al., 2019). Additionally, this second-level analysis allows to compare data from behavioral measures and neural activations in response to the same stimuli; or even to compare data from different animal species. Indeed, in this thesis we provide an overview of another powerful application of the RSA analysis: the multimodal fusion methods.

3.3.3. Derived analyses: Multimodal fusion methods

One of the reasons why the RSA technique is so powerful is that it allows different types of data to be mapped onto each other. For example, Kriegeskorte et al., 2008 have demonstrated that it is possible to map the function of Inferior Temporal (IT) cortex across humans and monkeys using the RSA technique. Therefore, information extracted from different neuroimaging modalities such as EEG and fMRI can be fused following the same rationale (Radoslaw M. Cichy & Oliva, 2020). Computing the similarity of RDMs across fMRI and M/EEG measurement spaces, we can test the hypothesis that the same neural generators are measured in particular locations at particular time points. In other words, multimodal fusion methods mitigate the weaknesses of individual neuroimaging modalities, which allows to obtain information about the function of the brain with high temporal and spatial resolution.

Multimodal fusion methods have been already used, for example, to investigate the cascade of spatiotemporal processing during visual object processing using a ROI approach (Radoslaw Martin Cichy et al., 2014) or to study the spatiotemporal dynamics underlying face perception (Muukkonen et al., 2020). Further details about RSA and multimodal fusion methods are provided in *Chapter 6 Multimodal fusion methods in MVPAlab* of this thesis. That chapter meticulously describes not just the Representational Similarity Analysis applied to EEG and fMRI but also discusses and tests the EEG-fMRI data fusion paradigm.

Section II.

Contribution of this thesis

Chapter 4. MVPA techniques to study Flanker interference effects

4.1. Introduction

Cognitive control comprises a set of mechanisms that allow humans to behave according to their internal goals while ignoring distracting information (Botvinick et al., 2001). The Flanker task (Eriksen & Eriksen, 1974), where participants respond to the direction of an arrow surrounded by other distracting arrows, is among the most used in the field. The main result of this task is the so-called interference or conflict effect, where responses are slower and less accurate in incongruent (when the direction of the distracters is opposite to the target) vs. congruent trials. In the current study, we employed the described Flanker task in the context of effort avoidance (Kool et al., 2010). Cognitive control involves effort, which is costly and partly aversive, and thus humans usually avoid it if given the chance. In Demand- Selection Tasks (DST) (Kool et al., 2010), participants tend to choose the easy option over the hard one. The tendency to avoid the hard option seems partly due to the cost of overcoming the increased cognitive control required when responding to incongruent situations. However, the neural underpinnings of this effect are not well understood.

The majority of Electroencephalography (EEG) studies of the interference effect have analyzed Event-Related Potentials (ERPs), focusing on the N2 component. Besides, studies employing frequency analyses have shown Theta and Delta band involvement. Other authors (Harper et al., 2014) have proposed a link between the ERPs and modulations in the Delta-Theta band of frequency. These univariate approaches have been the gold standard in the EEG literature for years, not only to study the interference effects but several cognitive processes.

In recent years, newer Multivariate Pattern Analysis (MVPA) techniques based on Machine Learning algorithms, in conjunction with neuroimaging techniques such as functional Magnetic Resonance Imaging (fMRI), Electroencephalography or Magnetoencephalography (MEG), have gained popularity in Cognitive Neuroscience (Etzel et al., 2011; Haxby et al., 2014; Johnson et al., 2009; Kriegeskorte & Bandettini, 2007; Misaki et al., 2010; Norman et al., 2006). These supervised Machine Learning algorithms, particularly Linear Support Vector Machines (LSVM) (Boser et al., 1992; Cortes & Vapnik, 1995) have been also widely applied in clinical settings such as computer-aided diagnosis of Alzheimer's disease (I. Álvarez et al., 2009; Chaves et al., 2009; Ramírez et al., 2013; Salas-Gonzalez et al., 2010), automatic sleep stages classification (Aboalayon et al., 2014; Koley & Dey, 2012) or automatic detection of sleep disorders (López-García et al., 2018b).

One of the most remarkable advantages of these multivariate over univariate approaches is its sensitivity in detecting subtle changes in the patterns of activations associated with specific information content (Kaplan et al., 2015). When applied to fMRI data, the poor temporal resolution of the signal prevents an accurate study of how cognitive processes unfold in time. In contrast, when applied to M/EEG signals (Su et al., 2014), MVPA has been useful to uncover the neural dynamics of face detection (Cauchoix et al., 2014), the process of memory retrieval (Kerrén et al., 2018), the representational dynamics of task and object processing in humans (Hebart et al., 2018a) or the representation of spoken words in bilingual listeners (Correia et al., 2015). In the same line, time-resolved MVPA presents an opportunity to categorize the temporal sequence of the neural processes underlying the interference effect. Furthermore, the relationship between these and Theta frequency modulations reported in previous studies (Cohen & Donner, 2013) can be better understood using this approach.

This study is an extension of previous work (López-García et al., 2019) which adapted a DST to a format that allows measuring concurrent high-density electroencephalography. Our main goal is to present a set of methodological MVPA tools that allow to study and decode the conflict-related neural processes underlying interference effects, in a time-frequency resolved way.

4.2. Materials and methods

4.2.1. Paradigm and data acquisition

Participants

Thirty-two healthy individuals (21 females, 29 right-handed, mean age = 24.65, SD = 4.57) were recruited for the experiment. The sex imbalance reflects the usual distribution of gender in the student pool (Psychology) where participant recruitment took place. Participants had normal or corrected-to-normal vision and no neurological or psychiatric disorders. All of them provided informed, written consent before the beginning of the experiment and received a 10-euro payment or course credits in exchange for their participation. The experiment was approved by the Ethics Committee of the University of Granada.

Stimuli and apparatus

Stimuli presentation and behavioral data collection were carried out using MATLAB (MathWorks) in conjunction with Phychtoolbox-3 (Kleiner et al., 2007). The visual stimuli were presented in an LCD screen (Benq, 1920x1080 resolution, 60 Hz refresh rate) and placed 68.31 ± 5.37 cm away of participants' Glabella, in a magnetically shielded room. Using a photodetector, the stimuli onset lag was measured at 8ms, which corresponds to half of the refresh rate of the monitor. Triggers were sent from the presentation computer to the EEG recording system through an 8-bit parallel port and using a custom MATLAB function in conjunction with inpoutx64 driver (Gibbons, n.d.), a C++ extension (mex-file) that uses native methods to access low-level hardware in MATLAB (I/O parallel ports).

Cues consisted of two squares of two different colors (red/green and yellow/blue, in different blocks) stacked and presented at the center of the screen (visual angle ~5 degree). In forced blocks, a small white indicator (circle 50% or square 50%) appeared on top of the color that had to be chosen. In voluntary blocks, this indicator appeared between the two-colored squares (Figure 25b). Each target stimulus consisted of five arrows pointing left or rightwards, which were displayed at the center of the screen (visual ~6 degree). The color of the target stimulus was the same as the cue previously selected.

Procedure

The Color-Based Demand-Selection Task (DST) (Figure 25a) modified from (Kool et al., 2010), consisted of a cue-target sequence arranged in four blocks (2 forced and 2 voluntary). In voluntary blocks, participants were required to freely choose one of the two colors available, which indicated the difficulty of the upcoming task. In forced blocks, a small white indicator appeared on top of the color that had to be chosen. The color of the target stimulus was the same as the cue previously selected and participants were required to discriminate the orientation (right or left) of a central arrow target surrounded by arrows pointing at the same (compatible distractors) or opposite (incompatible distractors) directions.

Our task was built following a 3-way factorial design, containing the following within-subjects independent variables: (1) Stimulus type (congruent/incongruent); (2) Block type (forced/voluntary) and (3) Context (easy/difficult). The task difficulty manipulation was based on the proportion of congruent and incongruent trials, with the easy contexts presenting 80% of congruent and 20% of incongruent trials, and the difficult task context the opposite proportion. Within forced blocks, half of the trials corresponded to the easy context and the remaining to the difficult one (maintaining, within each condition, the proportion of congruent and incongruent trials). On voluntary blocks, however, participants freely chose the context and no experimental control could be exerted upon this variable.

Participants were instructed to respond as fast and accurately as possible, and to not choose color based on personal preference. They were unaware of the cognitive effort manipulation. To preserve the signals as clean as possible and remove the least number of trials, participants were encouraged to remain as still and relaxed as possible, avoiding face muscle activity and eye movements, but blinking normally. The order of the blocks, cue colors, response keys and color-conflict context mappings were counterbalanced across participants. There were 4 blocks, 240 trials per block, and the total recording session lasted ~90min. Before the experimental session, participants performed a brief practice to familiarize themselves with the task (4 blocks, 20 trials per block, practice duration ~20min). To reduce fatigue, there were rest periods between blocks, with a variable duration depending on participants' choice. During this period, participants were asked to remain seated and rest their eyes and posture before continue with the task. Additionally, block order was counterbalanced across participants, and within each block, trial order was randomized, which effectively prevents confounds due to differential levels of fatigue across conditions.

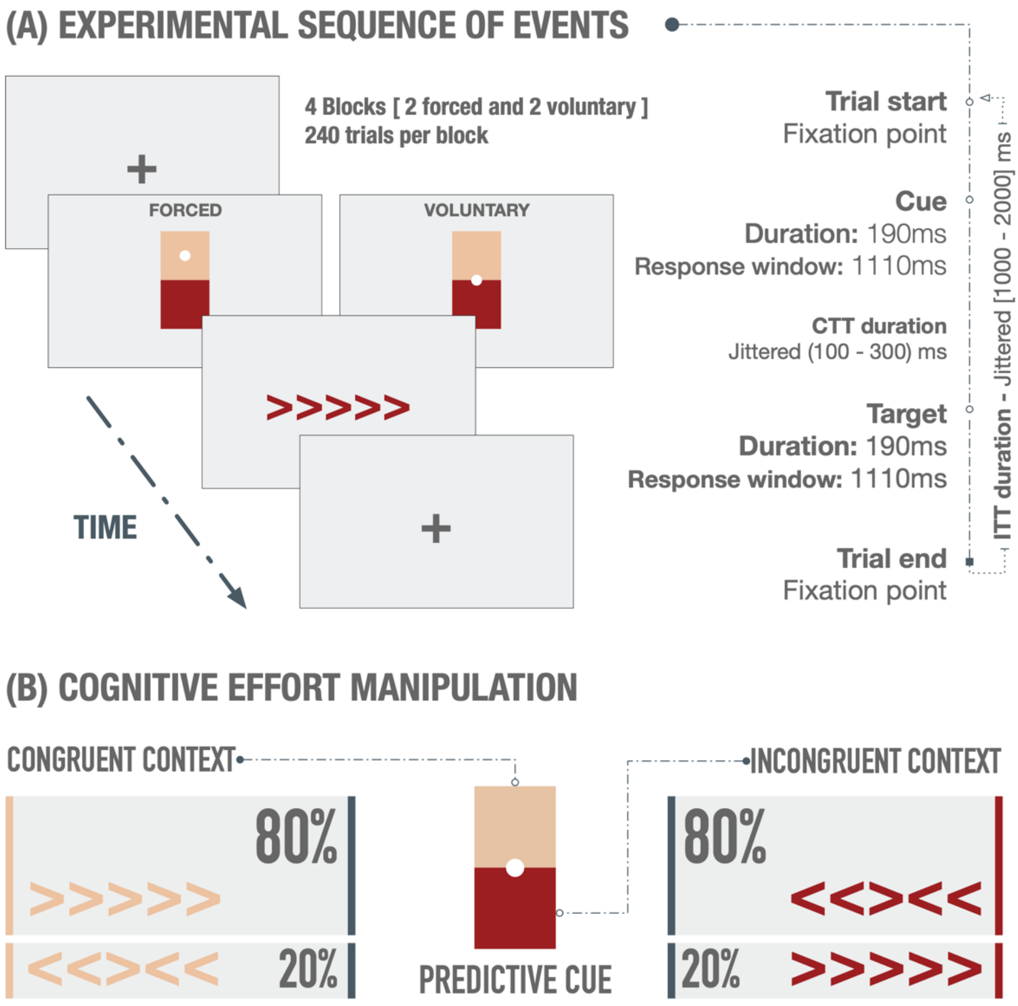


Figure 25. Demand Selection Task

(A) Experimental sequence of events in case of a correct response on both cue and flanker stimuli. Each trial started with a fixation point, followed by a cue, which acted as a selector of the difficulty of the upcoming Flanker target. Participants had to choose (freely or forced, depending on the block type) the possible color of the upcoming target stimulus, which was associated with either high (difficult) or low (easy) probability of incongruent trials. Finally, after a variable time interval (100-300ms) the target stimulus appeared and participants had to respond to the orientation of the central arrow. Another variable time interval appeared before the beginning of the next trial. The cue and the target stimuli remained on screen for 190ms. **(B)** Cognitive effort was manipulated through the percentage of congruent and incongruent trials. Each cue color was associated with the high or low conflict contexts.

EEG acquisition and preprocessing

High-density electroencephalography was recorded from 65 electrodes mounted on an elastic cap (actiCap slim, Brain Products) at the Mind, Brain, and Behavior Research Center (CIMCYC, University of Granada, Spain). The TP9 and TP10 electrodes were used to record the electrooculogram (EOG) and were placed below and next to the left eye of the participant. Impedances were kept below 5kΩ, as recommended by the amplifiers manufacturer. EEG activity was referenced online to the FCz electrode and signals were digitized at a sampling rate of 1KHz.

Electroencephalography recordings were average referenced, downsampled to 256Hz, and digitally filtered using a low-pass FIR filter with a cutoff frequency of 120Hz, preserving phase information. The recording amplifiers have an intrinsic lower cutoff frequency of 0.016Hz (time constant $\tau = 10s$).

No channel was interpolated for any participant. EEG recordings were epoched [-1000, 2000ms centered at onset of the target arrows] and baseline corrected [-200, 0ms], and data were extracted only from correct trials. To remove blinks from the remaining data, Independent Component Analysis (ICA) was computed using the *runica* algorithm in EEGLAB (Delorme & Makeig, 2004), excluding TP9 and TP10 channels. Artifactual components were rejected by visual inspection of raw activity of each component, scalp maps and power spectrum. Then, an automatic trial rejection process was performed, pruning the data from non-stereotypical artifacts. The trial rejection procedure was based on (1) abnormal spectra: the spectrum should not deviate from baseline by ±50dB in the 0-2 Hz frequency window, which is optimal for localizing any remaining eye movements, and should not deviate by -100dB or +25dB in 20-40Hz, useful for detecting muscle activity (~1% of the total sample was rejected); (2) improbable data: the probability of occurrence of each trial was computed by determining the probability distribution of values across the data epochs. Trials were thresholded, in terms of ±6SD, and automatically rejected (~6% of the total sample); (3) extreme values: all trials with amplitudes in any electrode out of a ±150μV range were automatically rejected ~3% of the total sample). See (Chaumon et al., 2015; Gross et al., 2013; Keil et al., 2014) for similar EEG preprocessing routines.

Final dataset description

The final dataset for our binary classification problem is shown in Table 1, where $\boldsymbol{N}$ is the initial number of trials per participant and class, N_r represents the number of remaining correct trials after the trial rejection stage and N_b is the final number of trials used for classification per participant (after downsampling the majority class in order to get balanced datasets).

Table 1. Number of observations of the final dataset

Observations per participant	$\boldsymbol{N}$	N_r	N_b
Congruent class	480	426±49	359±52
Incongruent class	480	368±59	359±52
Total number of observations		N_{Tr}	N_{Tb}
Congruent class		13644	11505
Incongruent class		11782	11505

Behavioral data analysis

Reaction time (RT) and error rates were registered for each participant. Before the statistical analysis, the first trial of each block, trials with choice errors and trials after errors were filtered out (Schouppe et al., 2014). Finally, RT outliers were also rejected using a ±2.5 SD threshold, calculated individually per participant and condition. To analyze behavioral data (accuracy and reaction times) we conducted repeated-measures ANOVAs in SPSS Statistics Software (v.20). Post hoc tests were carried out on the significant interactions using a Bonferroni correction for multiple comparisons.

4.2.2. Multivariate pattern analysis

The MVPA for the decoding analysis was performed in MATLAB by a custom-developed set of linear Support Vector Machines, trained to discriminate between congruent and incongruent target stimuli. To avoid skewed classification results, the datasets were strictly balanced, by downsampling the majority class to match the size of the minority one. In addition, class size was set as a factor of k, the total number of folds in the cross-validation stage. Accordingly, each fold was composed by the same number of observations, avoiding any kind of bias in the results. The rest of the classification parameters remained by default.

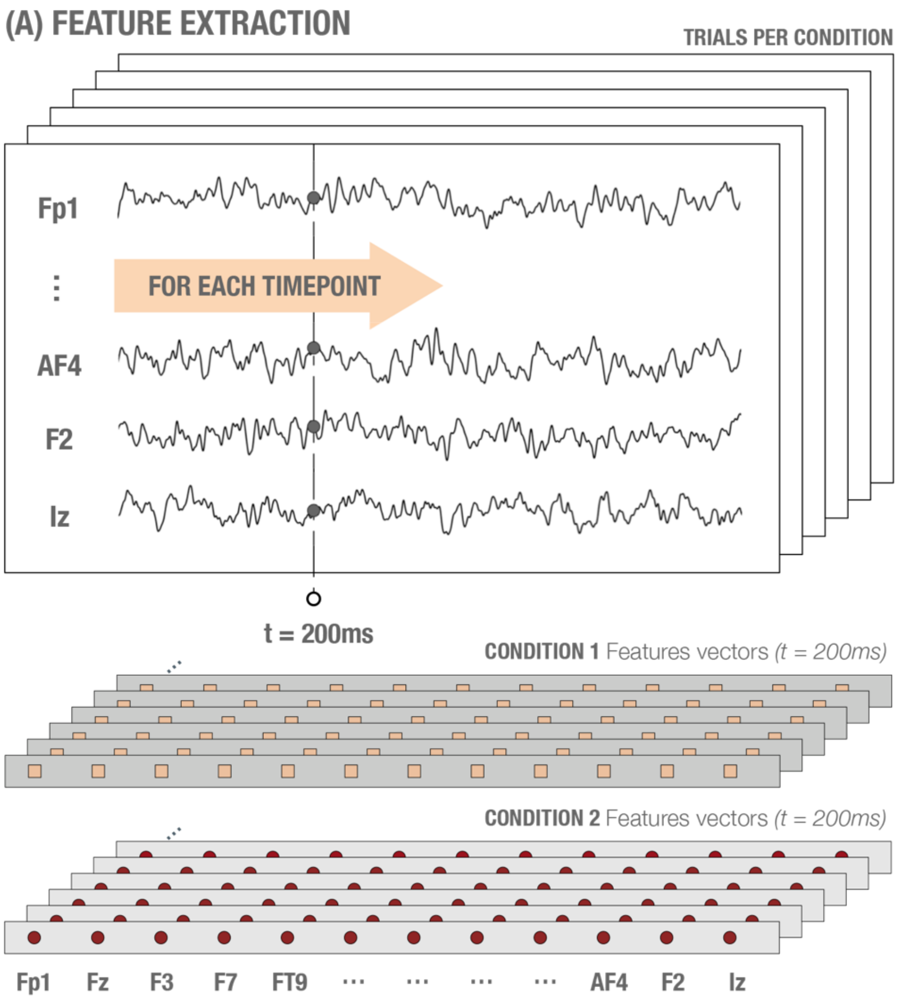


Figure 26. Feature extraction process

The feature vectors of each condition and time point consisted of a z-scored voltage array for all the scalp electrodes. For an improved SNR, several trials were averaged before feature extraction.

Feature extraction

To obtain the classification performance in a time-resolved way, the feature vectors were extracted as shown in Figure 26. The classification procedure, for each participant, ran as follows: (1) For each timepoint and trial, we generated two feature vectors (one for each condition or class) consisting of the raw potential measured in all electrodes (excluding EOG electrodes: TP9 and TP10). (2) Each individual feature vector, containing raw potential values were normalized (z-score, $\mu = 0,\ \sigma = 1$).

Supertrial generation

Due to the noisy nature of the EEG signal, a trial averaging approach was carried out during the feature extraction stage. This approach increases the signal-to-noise ratio (SNR) (Isik et al., 2014a), improves the overall decoding performance and also reduces the computational load. Each participant's dataset was reduced by randomly averaging a number of trials t_a belonging to the same condition. The value of t_a is a trade-off between an increased classification performance (due to an increased SNR) and the variance in the classifier performance, since reducing number of trials per condition typically increases the variance in (within-participant) classifier performance (Grootswagers et al., 2017). Therefore, the optimal number of trials to average depends on the dataset, taking into account that averaging more trials does not increment the decoding performance linearly.

Feature selection

$\mathrm{X}_{\mathrm{n\times p}}$ datasets are generated for each participant and timepoint, where n is the number of trials (observations) and p the total number of electrodes (variables or features). In machine learning, feature selection techniques, also known as dimension reduction, are a common practice to reduce the number of variables in high-dimensional datasets (Figure 27) Principal Component Analysis (PCA) is probably the most popular multivariate statistical technique and it is used in almost all scientific disciplines, (Abdi & Williams, 2010) including Neuroscience (Hebart et al., 2018a).PCA is a linear transformation of the original dataset in an orthogonal coordinate system in which axis coordinates (principal components) correspond to the directions of highest variance sorted by importance. To compute this transformation, each row vector $\boldsymbol{x_i}$ of the original dataset $\boldsymbol{X}$ is mapped to a new vector of principal components $\boldsymbol{t_i} = (\boldsymbol{t_1}, \dots, \boldsymbol{t_l})$, also called *scores*, using a p-dimensional *coefficient* vector $\boldsymbol{w_j} = (\boldsymbol{w_1}, \dots, \boldsymbol{w_p})$**.** For dimension reduction, $\boldsymbol{l} < \boldsymbol{p}$:

$$t_i = x_i \cdot w_j \quad i = 1, \dots, n \quad j = 1, \dots, l$$

Equation 10

To maintain the model's performance as fair as possible, in our study PCA was computed only for training sets $\boldsymbol{X_{training}}$, independently for each fold inside the cross-validation procedure. Once PCA for the corresponding training set was computed and the model was trained, the exact same transformation was applied to the test set $\boldsymbol{X_{test}}$ (including centering, $\boldsymbol{\mu_{training}}$). In other words, the test set was projected onto the reduced feature space obtained during the training stage. According to Equation 10, this projection is computed as follows:

$$\mathrm{T_{test}} = \frac{\mathrm{X_{test}} - \mathrm{\mu_{training}}}{\mathrm{W'_{training}}}$$

Equation 11

Feature selection techniques such PCA usually imply an intrinsic loss of spatial information, e.g. data projected from the sensor space onto the reduced PCA features space. Therefore, PCA presents a trade-off between dimension reduction and the interpretation of the results. If PCA is computed, the spatial information of each electrode is lost, which means that, for example, we cannot directly analyze which electrodes are contributing more to decoding performance.

Evaluation of the performance of the model

To evaluate classification models in neuroscience, performance is usually measured employing mean accuracy (Combrisson & Jerbi, 2015). However, mean accuracy may generate systematic biases in situations with very skewed sample distributions, and overfitting one single class should be avoided. Therefore, nonparametric and criterion-free estimates, such as the Area Under the ROC Curve (AUC) have been proved as a better measure of generalization in these situations (J.-R. King & Dehaene, 2014). The AUC provides a way to evaluate the performance of a classification model. The larger the area, the more accurate the classification model is, and it is computed as follows:

$$AUC = \int_0^1 \mathrm{ROC}(s)ds$$

Equation 12

The ROC curve is one of the most suitable evaluation criteria, as it shows how capable the model is in distinguishing between conditions, by facing the sensitivity (True Positive Rate, TPR) against 1-specificity (False Positive Rate, FPR). In this study, we employed both methods, the mean accuracy, to replicate a common approach in literature, and ROC curves and AUC, to provide a more informative measure.

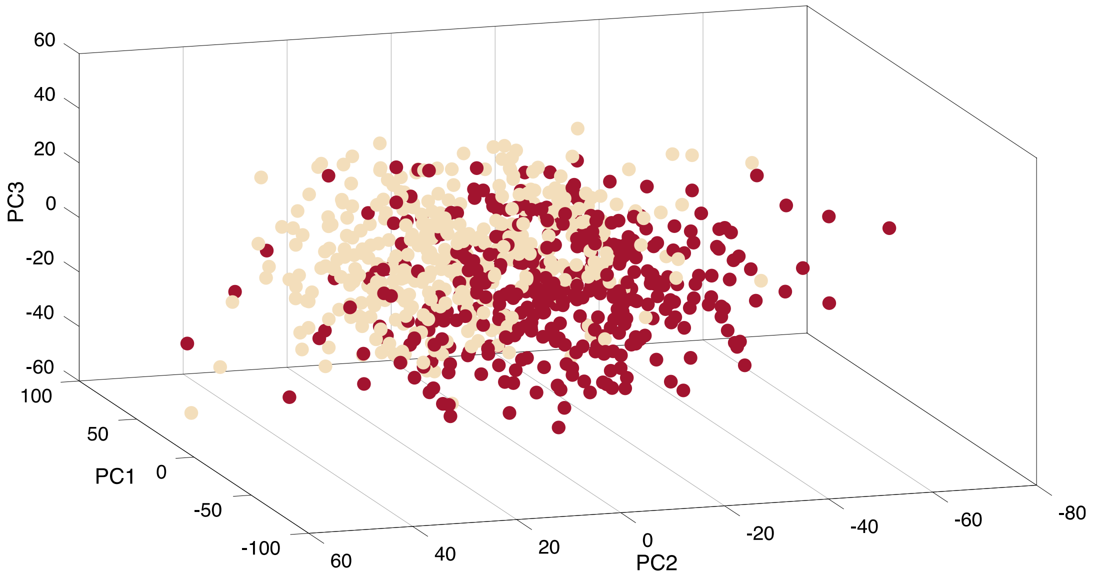


Figure 27. Dimensionality reduction procedure in real data

3D representation of the three first PCA components for congruent vs. incongruent trials [example participant, t = 421ms after Flanker stimulus onset].

To evaluate the performance of our model, LSVMs were trained and validated, resulting in a single performance value for each timepoint and participant. The classification performance at the group level was calculated by averaging these values across participants. The chance level was calculated following the former analysis but using randomly permuted labels for each trial.

The generalization ability of our model was estimated through a Cross-Validation (CV) approach (stratified k-fold, $k = 5$), which is a well-established and a widely implemented technique to preserve complex models from overfitting.

Moreover, some important aspects are worth being highlighted. The use of CV approaches often leads (particularly in Neuroscience) to small sample sizes and a high level of heterogeneity when conditions are split into each fold, causing among other things a large classification variability (Varoquaux, 2018). To address these problems, recent studies (Gorriz et al., 2019; Górriz, Ramirez, et al., 2019) considered the use of the resubstitution error estimate when using LSVM (in small sample sizes and low dimensional scenarios), proposing a novel analytic expression for the upper bound on the actual risk $\gamma_{emp}(l, d)$ for a range of sample sizes l, dimensions d and any significance $\eta < 0.05$ (Figure 32). Therefore, the difference between the actual error and the resubstitution error is bounded by the actual risk γ_{emp}, which is computed as follows:

$$\gamma_{emp} \leq \sqrt{\frac{1}{2l} \ln \frac{N(l,d)}{\eta}}$$

Equation 13

where N is defined as:

$$N(l,d) = 2 \sum_{k=0}^{d-1} \binom{l-1}{k}$$

Equation 14

Resubstitution has been proved competitive in some heterogeneous data scenarios with CV approaches not only in terms of accuracy but also in computational load (Braga-Neto et al., 2004). The proposed solution has been recently applied in clinical settings studying autistic patterns (Górriz, Ramírez, et al.,   2019a) or Alzheimer’s Disease (Martinez-Murcia et al., 2020). The scenario previously mentioned (linear classifiers, small sample size and low dimensional space) seems to fit perfectly with our study setup, therefore, we also used the resubstitution error estimate to evaluate the classification performance.

Optimization of SVM hyperparameters

A search-grid based optimization of the misclassification cost parameter $\boldsymbol{C}$ was carried out using five-fold cross-validation on the training set:

$$||\beta||^2 + C\sum_{i=1}^{l} \xi_i$$

Equation 15

where $\boldsymbol{C}$ is a constant which modulates the trade-off between the training error and the complexity of the model and the vector $\boldsymbol{\beta}$ contains the coefficients that define an orthogonal vector to the hyperplane.

4.2.3. Temporal generalization matrix

Temporal generalization analyses are used to evaluate the stability of the brain patterns along time, by training the model in one temporal point and testing its ability to discriminate between conditions in the remaining temporal window. This process is repeated for every timepoint. In our study, classification performance was assessed through a cross-validation technique (stratified k-fold cross-validation,$k = 5$). For each timepoint, the classifier was trained with $\mathbf{X}_{\mathbf{training}}$ dataset and tested with $\mathbf{X}_{\mathbf{test}}$ in the remaining points of the temporal window. This process was repeated k times, obtaining the final decoding accuracy.

An above-chance discrimination rate outside the diagonal of the matrix suggests that the same activity pattern is sustained in time. However, if there is no evidence of temporal generalization, different patterns of activity can be assumed (Hebart et al., 2018a)

4.2.4. Multivariate cross-classification

The ability of MVPA to detect subtle differences in brain activity patterns can be used to study how these patterns are similar across different cognitive contexts. In other words, the consistency of the information across different sets of data can be analyzed. To this end, classification algorithms are trained with one set of data and the consistency is assessed by testing the model with another dataset, belonging to a different experimental condition. This technique is called Multivariate Cross-Classification (MVCC) (Kaplan et al., 2015) and is growing in popularity in recent years (Etzel et al., 2008; Oosterhof et al., 2010, 2013).

The fact that the training and test sets are different eliminates the need to use cross-validation techniques. However, the classification direction has to be taken into account, that is, which set is used for training and which one for testing. The result of the classification could differ if, for instance, the signal-to-noise ratio is quite different across datasets, that is to say, differences in classes separability across datasets and an asymmetry in the generalization direction (Hurk & Beeck, 2019). For this reason, reporting results in both directions is highly recommended.

In this study, MVCC was used to analyze if the neural patterns associated with the congruency effect are similar across voluntary and forced blocks. For that, classifiers were trained with data of forced blocks and tested in voluntary blocks, and *vice versa*. In addition, a temporal generalization matrix was also computed to study the similarity across block types and time. Feature selection in MVCC analysis also requires some additional considerations, as features selected for the training set could not be optimal for the test set. To avoid possible skewed results, no feature selection was computed for MVCC analyses.

4.2.5. Frequency contribution analysis

The contribution of each frequency band to the overall decoding performance was assessed through an exploratory sliding filter approach. We designed a band-stop FIR filter using ***pop_firws*** EEGLab function (2Hz bandwidth, 0.2Hz transition band, 2816 filter order, Blackman window) and pre-filtered the EEG data (120 overlapped frequency bands, between 0-120Hz and linearly-spaced steps) producing 120 filtered versions of the original EEG dataset. The former time-resolved decoding analysis (congruent vs. incongruent, $t_a = 10$) was repeated for each filtered version and the importance of each filtered-out band was quantified computing the difference maps in decoding performance between the filtered and the original decoding results. Significant clusters were found applying the proposed cluster-based permutation test to filtered-out datasets, generating accuracy null distributions for each time-frequency point.

With the purpose of obtaining better frequency resolution in lower bands, the previous analysis was repeated for frequencies between 0-40Hz in 120 overlapped and logarithmically spaced steps.

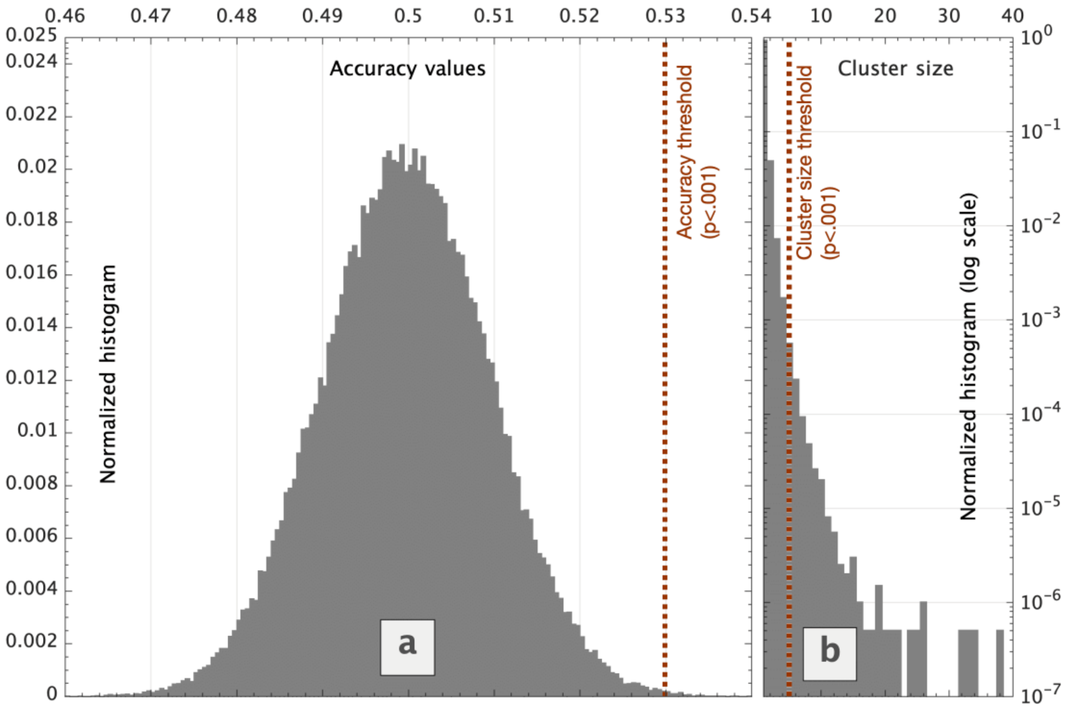


Figure 28. Null distribution of accuracy values and cluster size

Accuracy **(a)** and cluster size **(b)** null distributions. The vertical dotted line represents the threshold corresponding to a very low probability to obtain significant results by chance. This threshold corresponds to a p-value below 0.001 for both distributions.

4.2.6. Statistical analysis

Applying t-test statistics on multivariate results is an unsuitable approach to draw statistical inferences at the group level (Stelzer et al., 2013). For that reason, the use of cluster-based non-parametric permutation methods is widespread, not only in fMRI (Arco et al., 2018; Díaz-Gutiérrez et al., 2018; Nichols & Holmes, 2002; Palenciano et al., 2019) but more recently also in M/EEG studies. (Hubbard et al., 2019; Koenig-Robert & Pearson, 2019; Maris & Oostenveld, 2007; Meconi et al., 2019). In our study, a non-parametric cluster-based permutation approach, proposed in (Stelzer et al., 2013) for fMRI data, was adapted and implemented for the statistical analysis.

We thresholded the decoding accuracy obtained with an empirical accuracy null distribution, calculated by means of a combined permutation and bootstrapping technique. First, at the single-subject level, 100 randomly permuted accuracy maps were generated. To draw statistical inferences at the group level, we randomly drew one of the previously calculated accuracy maps for each participant. This selection was group-averaged and the procedure was repeated 10^5 times, generating 10^5 permuted group accuracy maps.

Next, for each timepoint we estimated the chance distribution of accuracy values and determined the accuracy threshold (99.9th percentile of the right-tailed area of the distribution), which corresponds to a very low probability to obtain significant results by chance (Figure 28a).

Then, we searched and collected clusters of timepoints exceeding the previously calculated threshold in all the 10^5 permuted accuracy maps, generating the normalized null distribution of cluster sizes (Figure 28b). Finally, we applied a correction for multiple comparisons (FDR, False Discovery Rate) at a cluster level to obtain the smallest cluster size to be considered significant (Benjamini & Hochberg, 1995; Benjamini & Liu, 1999; Bennett et al., 2009; Eklund et al., 2016).

4.3. Results

4.3.1. Behavioral results

The behavioral results of reaction times replicate well-known conflict effects linked to context-dependent congruency (Kool et al., 2010; Schouppe et al., 2014), with a significant interaction of Context × Stimulus Type ($F_{(1,31)} = 26.285, p < .004, \eta_p^2 = .459$). Planned comparisons showed significant differences between congruent and incongruent trials for both the easy ($F_{(1,31)} = 272.707, p < .001, \eta_p^2 = .885$) and the difficult contexts ($F_{(1,31)} = 183.109, p < .001, \eta_p^2 = .855$) with larger differences in reaction times in the easy (congruent trials: $M = 0.465, SD = 0.13$; incongruent trials: $M = 0.560, SD = 0.15$), compared to the difficult context (congruent trials: $M = 0.474, SD = 0.13$; incongruent trials: $M = 0.553, SD = 0.14$). The effort-avoidance effect, as expected, was also observed in voluntary decision blocks (percentage of choice of easy 57,11%, $SEM = 2.93$ vs difficult 42,88%, $SEM = 2.93$ contexts; $t = 2.42, p = .021$).

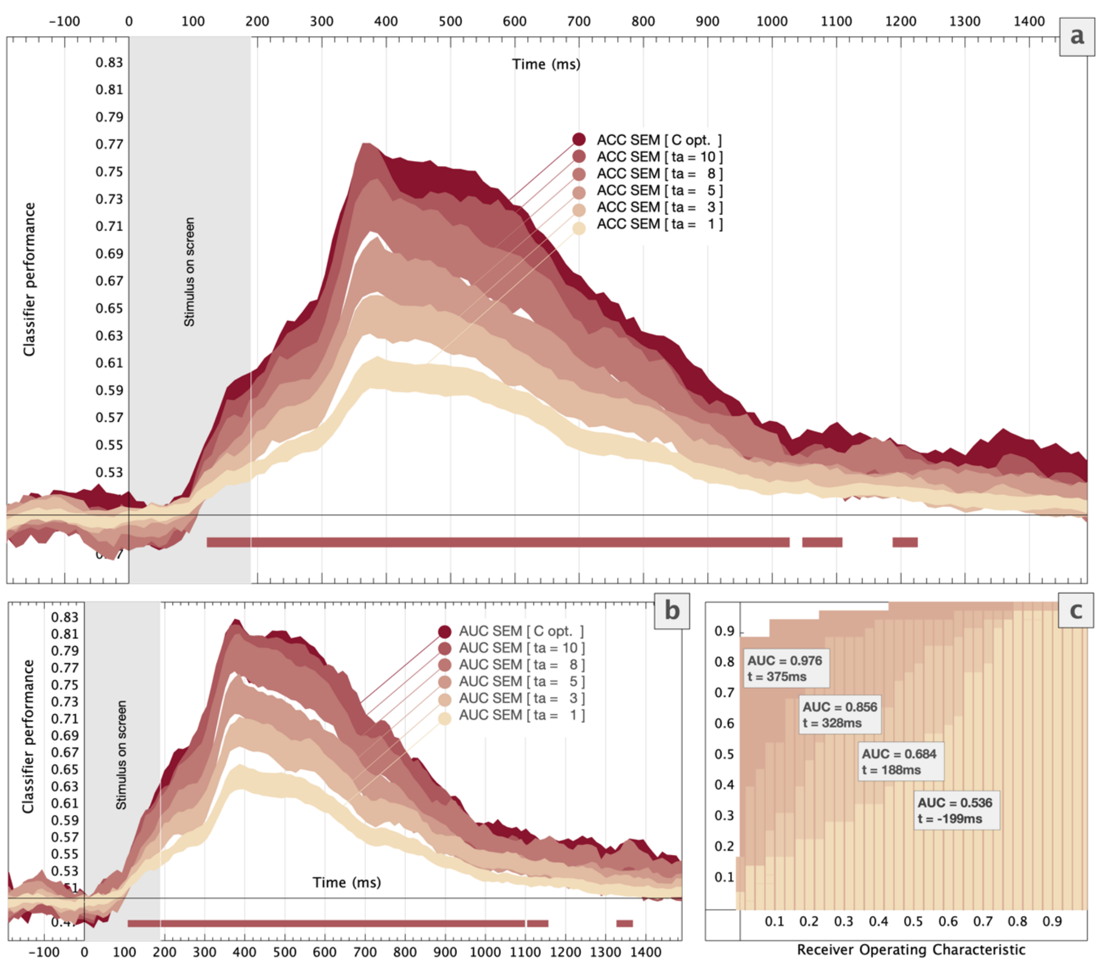


Figure 29. Group level MVPA results

Time-resolved classifier performance when different number of trials were averaged. The standard error of the **(a)** classification accuracy and **(b)** the Area Under the Curve are represented using colored areas. Significant windows ($t_a = 10$) obtained via Stelzer permutation test are highlighted using horizontal bold lines. The stimulus screen time [0-190] ms is shaded. **(c)** Receiver Operating Characteristic curves for different timepoints [example participant, $t_a = 10$, t=-199ms, 188ms, 328ms and 375ms].

4.3.2. Electrophysiological results

The electrophysiological analyses (Figure 29a) show significant differences (p<.001, cluster corrected) in activity patterns for congruent vs. incongruent trials, peaking at 375ms after the stimulus onset. At this point, the classifier accurately predicted (>80%) if participants were responding to congruent or incongruent trials. Table 2 reports the variations in classification performance for averages of different number of trials. The SVM hyper-parameter C was optimized, slightly increasing the decoding performance; however, the computation time required increased significantly.

When ten trials were averaged to generate supertrials, the statistically significant regions extended from 130ms after stimulus onset to 1200ms afterwards. As Figure 29 shows, before the stimulus onset the classification accuracy remained at chance levels (0.5).

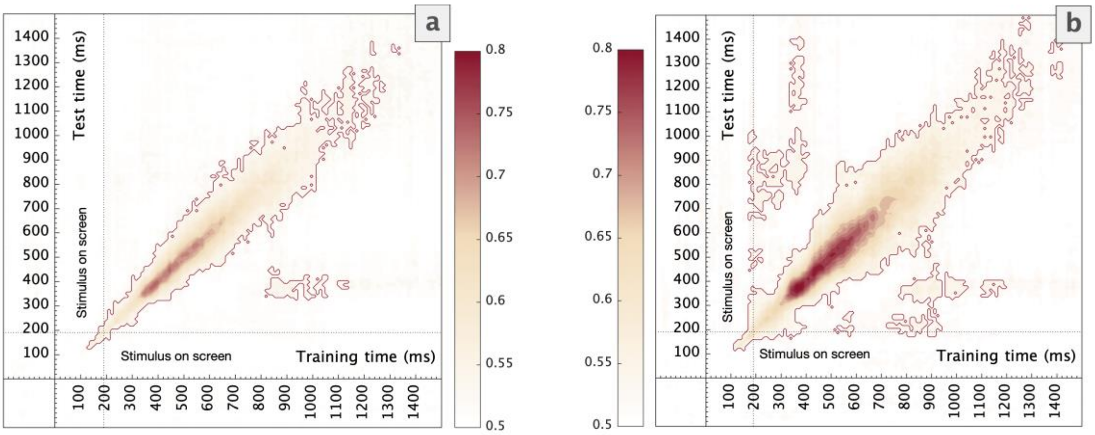


Figure 30. Group level temporal generalization results

Group level temporal generalization results for congruent vs. incongruent trials ($t_a = 10$). Accuracy **(a)** and AUC **(b)** values when the model was trained and tested in each time point of the whole time window. Significant clusters obtained via Stelzer permutation tests are highlighted using red lines.

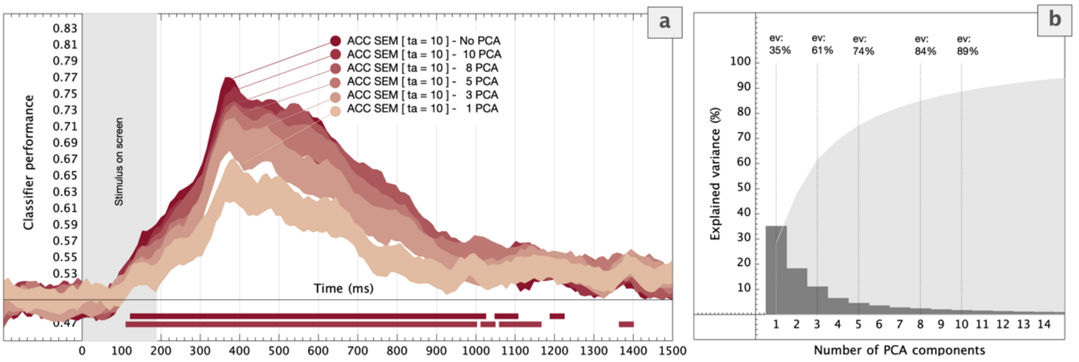


Figure 31. Group level MVPA results for different numbers of components

(a) Time-resolved classifier performance (ACC, congruent vs. incongruent trials) for different number of PCA components. Colored areas represent the ACC standard error. Statistically significant time windows ($t_a = 10$, for both 10 PCA components and when PCA was not computed) are highlighted using horizontal bold lines. The stimulus screen time [0-190] ms is shaded. **(b)** Explained variance for different numbers of PCA components [example participant].

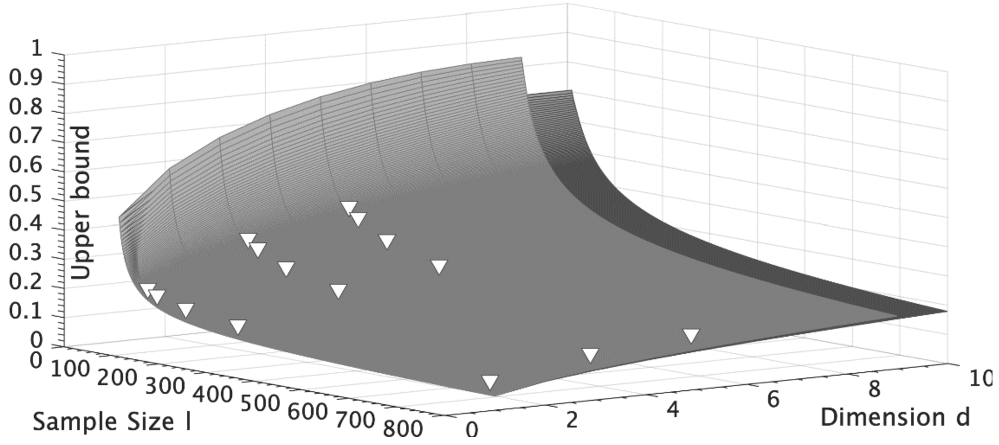


Figure 32. Different upper bound estimations across dimensions and sample size

Different upper bound estimations via the procedure found in (Górriz, Ramirez, et al., 2019) for LSVM across dimension and sample size at a 95% confidence level ($\eta = 0.05$). White markers represent the upper bound values for the experimental conditions tested in our study.

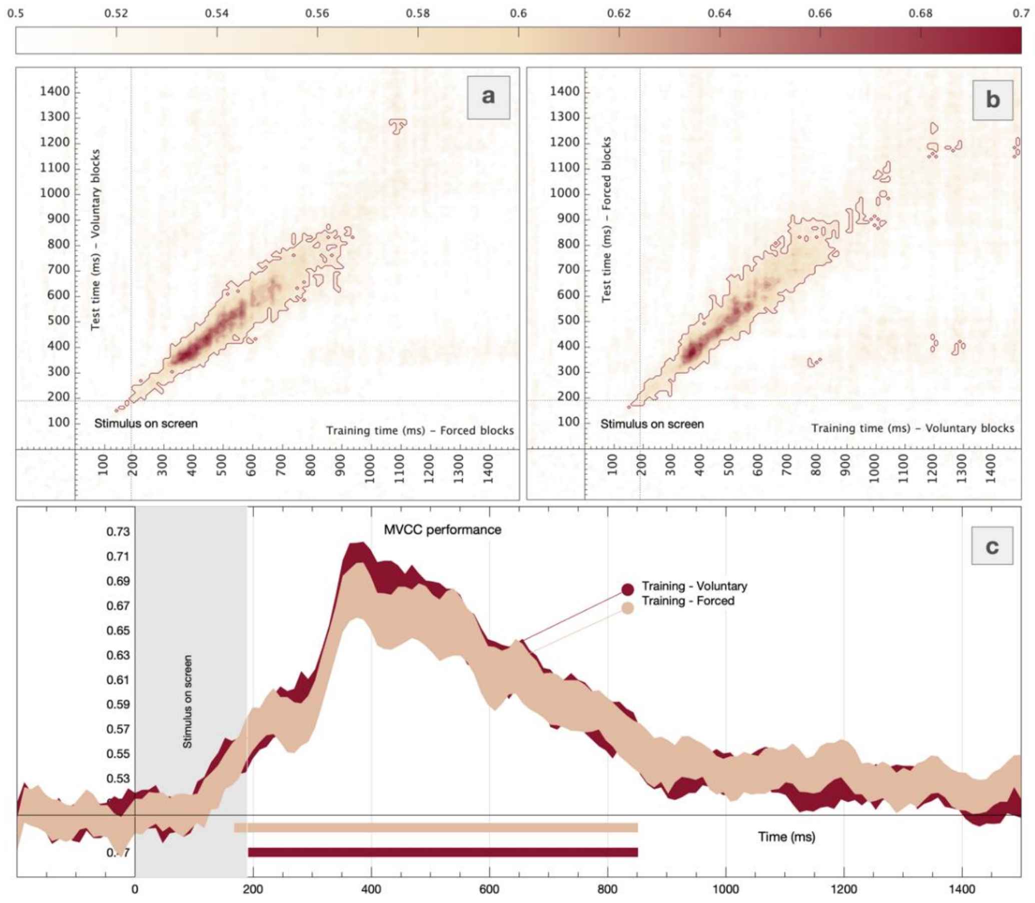


Figure 33. Multivariate Cross-Classification results

(a) Temporal generalization results when the model was trained with forced blocks and tested in voluntary blocks and (b) *vice versa*. (c) Classifier performance (acc) for the former analyses. Colored areas represent the standard error. Significant windows calculated via Stelzer permutation test are highlighted using horizontal lines.

The temporal generalization analysis is shown in Figure 30. First, the AUC proved to be a more sensitive measure. The AUC temporal generalization matrix (Figure 30b) shows a distinct pattern of generalization. Clusters appearing only alongside the diagonal have been associated with a succession of different mechanisms. That is to say, the neural information that allows the classifier to tell apart congruent and incongruent trials is likely the result of a series of distinct events. Moreover, Figure 30b shows a cluster of homogeneous AUC between 200 and 400ms, which theoretically suggests the operation of a single cognitive process maintained in time (J.-R. King & Dehaene, 2014). Such mechanism apparently reappears at ~800-1000ms after the target onset, posterior to the mean RT (513ms).

Table 2. LSVM model peak classification performance at the group level

LSVM model peak classification performance [t=375ms] at the group level. The mean accuracy and AUC are reported for different values of t_a and different numbers of PCA components.

No. of averaged trials (t_a)	ACC± SD	AUC ± SD
$t_a = 1$	$.60 \pm .05$	$.65 \pm .07$
$t_a = 3$	$.65 \pm .07$	$.70 \pm .08$
$t_a = 5$	$.69 \pm .10$	$.74 \pm .10$
$t_a = 8$	$.74 \pm .10$	$.79 \pm .12$
$t_a = 10$	$.76 \pm .11$	$.80 \pm .13$
$t_a = 10$ C-optimized	$.76 \pm .10$	$.81 \pm .13$
No. of PCA components ($t_a = 10$)		
First component	$.64 \pm .14$	$.66 \pm .17$
3 first components	$.71 \pm .11$	$.76 \pm .13$
5 first components	$.72 \pm .11$	$.78 \pm .12$
8 first components	$.73 \pm .12$	$.80 \pm .13$
10 first components	$.74 \pm .11$	$.81 \pm .13$

The actual risk estimation for different sample sizes and dimensions γ_{emp} is shown in Figure 32. The difference between the actual error and the resubstitution error is bounded by γ_{emp}. White markers represent different experimental configurations for both the sample size l and the number of PCA components d analyzed in our study. Performance results obtained by resubstitution (C-optimized, t=375ms) for these experimental configurations are shown in Table 3. The classification accuracy remained above chance despite the conservative estimation of the upper bound of the actual error, preserving our classification model for overfitting and proving that both conditions (congruent and incongruent) are representative of the different underlying activity patterns associated with congruent and incongruent stimuli.

The cross-classification results (Figure 33a,b) showed smaller clusters compared to the MVPA time generalization (Figure 30a, b). However, the main diagonal cluster in the matrix indicates a series of different events that occur in cascade, but shared between n both contexts (J.-R. King & Dehaene, 2014). This mechanism could reflect the interference process itself, previous to the response.

Table 3. Classification performance and the actual risk for a LSVM model

Classification performance and the actual risk γ_{emp} for different values of l and d in a C-optimized LSVM model obtained by the resubstitution approach. [example participant, t=375ms].

$ACC(\gamma_{emp}(l,d))$	$d = 1\text{PCA}$	$d = 3\text{PCA}$	$d = 5\text{PCA}$
$l = 790\ (ta\ = 1)$	.55(.04)	.63(.10)	.65(.13)
$l = 260\ (ta\ = 3)$	.58(.08)	.68(.16)	.71(.20)
$l = 150\ (ta\ = 5)$	.58(.11)	.79(.20)	.81(.25)
$l = 90\ (ta\ = 8)$	.72(.14)	.80(.24)	.83(.30)
$l = 70\ (ta\ = 10)$	.81(.15)	.88(.27)	.92(.34)

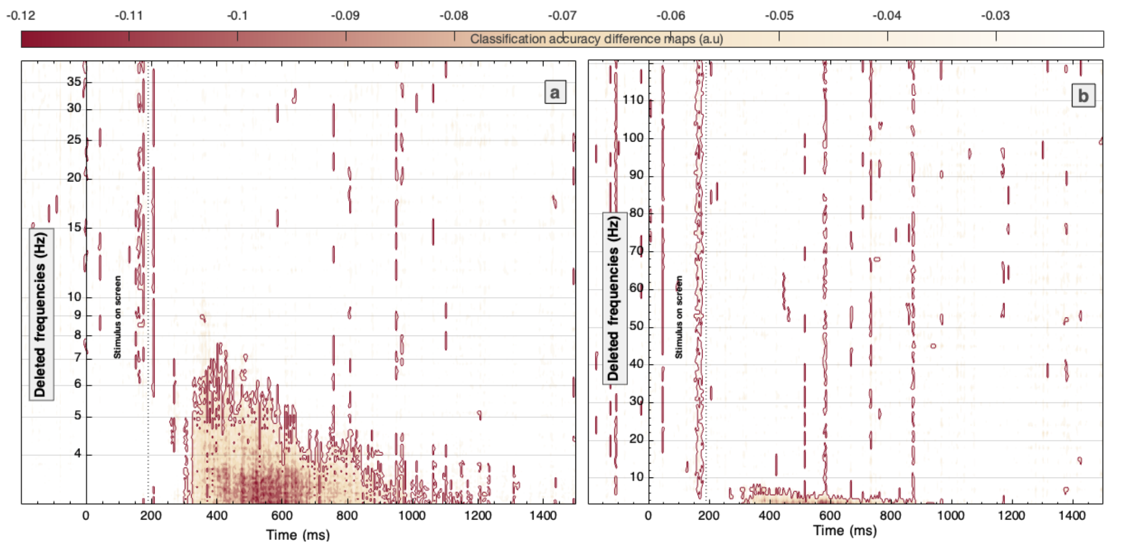


Figure 34. Results of the frequency contribution analysis

Classification accuracy differences when a specific frequency band is filtered out. **(a)** [0-40] Hz logarithmically spaced and **(b)** [0-120] Hz linearly spaced sliding filter approach. Significant clusters obtained via Stelzer permutation test are highlighted using red lines.

4.3.3. Frequency contribution results

A sliding band stop filter approach was followed to study the contribution of each frequency band to the overall decoding accuracy. Results show that the interference effect observed relies on neural processes operating in the Delta and Theta frequency bands. Figure 34a shows how decoding accuracy significantly drops when frequencies up to 8Hz were filtered out.

4.4. Discussion

In the current study, we present a set of multivariate pattern analysis techniques for EEG data. Overall, we effectively classified interference-related electrophysiological signals from a Demand-Selection Task in both a time and frequency-resolved manner.

Previous studies on cognitive control and more specifically, interference processing, have found that slow rhythms (i.e. Theta, Delta) are associated with communication between distant brain regions (Fries, 2005). Our results are in line with those studies, showing that Theta and Delta oscillations are relevant for the brain activity underlying performance in an interference task. Moreover, previous results (Cohen & Donner, 2013) show the relevance of Theta in the first instances of target processing, which changed to Delta after the participants' response. These results are supported by the present study, which shows Theta and Delta to be crucial for classification right after the target onset, which evolves into a single Delta-based classification around and after the response time. The meaning of the change from one frequency band to another along time could be due to neuronal activity on the Theta band preventing the distractors to be processed. Once the target is selected, Delta, which arises later, could reflect inhibition of competing and erroneous motor responses (Harmony, 2013). Our results also indicate the existence of a particular brain process involved in the interference effect that intervenes in the initial stages of target processing during an extended time window and reappears after the behavioral response is given. Interestingly, this is the same temporal window where classic Event-Related Potential studies (Van Veen & Carter, 2002) have repeatedly observed the N2 potential, which is taken as the reflection of interference processing. The indication that the same underlying mechanism reappears after the response could reflect the reinstantiation of the interference episode, perhaps reflecting trial event boundaries (Sols et al., 2017). This finding, which could not have been obtained with classic analytical strategies, opens novel avenues of research and aids to better characterize a commonly used neural correlate. Further research will be needed to clarify and extend this phenomenon. To summarize, our behavioral and electrophysiological results add up to existing literature, pointing to an overall robust effect of interference and conflict avoidance, which can be observed in various environments and demographic samples.

Future lines of investigation should address these findings to complement the results found in the current investigation. In addition, to increase our understanding of preparation processes and conflict effects, it would be of interest to continue analyzing the current dataset, focusing not only on the target stimulus, but also on the neural activity triggered by the cues. Further detailed analyses should be carried out to study the activation differences between forced and voluntary blocks or high and low congruency contexts. The use of newer classification algorithms, such as Spiking Neural Networks (Bernert & Yvert, 2019; Ghosh-Dastidar & Adeli, 2007, 2009; R. Hu et al., 2019) should be considered in related studies. They have been demonstrated to be more powerful in some scenarios than linear SVM (Ahmadlou & Adeli, 2010; Rafiei & Adeli, 2017). Nonetheless, in the present study, given the small sample size and our main goal (effective discrimination in time and frequency regardless of the actual accuracy value obtained), we decided to use less complex algorithms, which lead to more easily interpretable results (Hebart et al., 2016; Korjus et al., 2016; Pereira et al., 2009) and prevent model overfitting. For this, SVM-based multivariate techniques represent an opportunity to study the neural basis of complex psychological processes. In addition, the resubstitution error estimate proved that, even in the worst-case scenario when the estimated actual risk is maximum, the classification performance remained over the chance level. This method, which is suitable for small sample sizes and low dimensional scenarios is worthy of consideration in Cognitive Neuroscience studies, opening up a new path that could lead to promising results.

4.5. Conclusion

The current study is an initial approximation to adapt a DST to a format that allows measuring concurrent high-density electroencephalography. While most of previous studies categorize the interference effect through ERP markers such as the N2 potential, (Van Veen & Carter, 2002) we successfully used multivariate pattern analysis (MVPA) to decode conflict-related neural processes associated with congruent or incongruent events in a time-frequency resolved way. Our results replicate findings obtained with other analysis approaches and offer new information regarding the dynamics of the underlying mechanisms.

Chapter 5. MVPAlab: A Machine Learning decoding toolbox

5.1. Introduction

Historically, the study of brain function employing electroencephalography (EEG) data has relied on classical univariate analyses of amplitudes and delays of different peaks of the average of several evoked EEG recordings, commonly called Event-Related Potentials (ERPs). The constant development of science and technology in past decades has allowed researchers and engineers to develop and apply more advanced signal processing techniques, such as time/frequency analyses, phase clustering, Independent Component Analysis (ICA) decompositions (T. P. Jung et al., 2000; Makeig et al., 1996), and others. These techniques have been implemented in excellent analysis and preprocessing tools, such as EEGLAB (Delorme & Makeig, 2004), ERPLAB (Lopez-Calderon & Luck, 2014) or Fieldtrip (Oostenveld et al., 2011), enabling researchers to develop a myriad of studies in a wide range of areas.

More recently, newer Machine Learning-based algorithms (ML), in conjunction with advanced neuroimaging techniques, such as functional Magnetic Resonance Imaging (fMRI) or Magnetoencephalography (MEG), have gained popularity in neuroscience. This trend started with studies by Haxby and Norman (Haxby, 2001, 2012; Norman et al., 2006), and other reference contributions (T. Davis & Poldrack, 2013; Haynes & Rees, 2006; Kriegeskorte et al., 2006; Lemm et al., 2011; Mur et al., 2009; Pereira et al., 2009), which opened novel avenues of research on brain function. For years, ML models have been also successfully employed in medical imaging, mainly in the area of computer-aided diagnosis (Shiraishi et al., 2011). To mention just a few examples, the use of different ML approaches is mainstream in the study and detection of several neurological diseases, such as Parkinson (Ahmadi Rastegar et al., 2019; Gao et al., 2018; Martinez-Murcia et al., 2018), Alzheimer (Martinez-Murcia et al., 2020; Ramírez et al., 2013; Salas-Gonzalez et al., 2010), Autism (Duda et al., 2016; Górriz, Ramírez, et al., 2019b; Wall et al., 2012), or sleep disorders (D. Álvarez et al., 2020; López-García et al., 2018a; Palotti et al., 2019). Even the recently spread COVID-19 can be successfully diagnosed using Artificial Intelligence (AI) in chest radiographies, according to preliminary studies(Arco et al., 2021; S. H. Wang et al., 2021; R. Zhang et al., 2021). However, the recent growth of ML models is not limited to neuroscience or medical applications but is present in a huge range of scientific disciplines in a cross-cutting basis.

5.1.1. Related work

Multivariate Pattern Analysis (MVPA) usually encompasses a set of supervised learning algorithms, which provide a theoretically elegant, computationally efficient, and very effective solution in many practical pattern recognition scenarios. One of the most remarkable advantages of these multivariate approaches over univariate ones is its sensitivity in unveiling subtle changes in activations associated with specific information content in brain patterns. Several MVPA toolboxes, such as SPM (Penny et al., 2011), The Decoding Toolbox (TDT) (Hebart et al., 2015) or Pattern Recognition for Neuroimaging Toolbox (PRoNTo) (Schrouff et al., 2013), particularly designed for fMRI studies have been developed in the past years. Despite the good spatial resolution of the fMRI, the poor temporal resolution of the BOLD signal limits an accurate study of how cognitive processes unfold in time. For that reason, the application of multivariate pattern analyses to other neuroimaging techniques with a higher temporal resolution, such as EEG or magnetoelectroencephalography (MEG), is growing in popularity. With the aim of facilitating the work of researchers from different disciplines, allowing the access to these complex computation algorithms, diverse M/EEG-focused toolboxes have been developed. The Amsterdam Decoding and Modeling Toolbox (ADAM) (Fahrenfort et al., 2018), CoSMoMVPA (Oosterhof et al., 2016), MVPA-light (Treder, 2020), The Decision Decoding Toolbox (DDTBOX) (Bode et al., 2019), BCILAB (Kothe & Makeig, 2013) and The Berlin Brain-Computer Interface (Blankertz et al., 2016) are excellent examples of MATALB-based toolboxes. MNE-Python(Gramfort et al., 2013), Nilearn(Abraham et al., 2014) or PyMVPA (Hanke, Halchenko, Sederberg, Hanson, et al., 2009; Hanke, Halchenko, Sederberg, Olivetti, et al., 2009) are other Python-based and open source alternatives.

5.1.2. MVPAlab: a machine learning toolbox for decoding analysis

Despite the tremendous effort applied in other implementations to facilitate researchers the use of these tools (e.g. high-level functions which compute a complete decoding analysis in a few lines of code), its use is sometimes really challenging, especially for students, newcomers or other researchers with profiles with no coding experience.

Here we present MVPAlab, an easy-to-use decoding toolbox for M/EEG data. So, what makes MVPAlab different from any other existing alternatives? The MVPAlab Toolbox has been designed to include an easy-to-use and very intuitive Graphic User Interface (GUI) for the creation, configuration, and execution of different decoding analysis. Importantly, this friendly GUI provides access to an extensive set of computational resources to design, configure and execute the complete pipeline of different decoding analyses for multidimensional M/EEG data, including visualization software for data representation. MVPAlab implements several decoding functionalities, such as time-resolved binary classification, temporal generalization, multivariate cross-classification, statistical analyses to find significant clusters, feature contribution analyses, and many others. Highly configurable linear and non-linear ML models can be selected as classification algorithms, including Support Vector Machines (SVM) or Discriminant Analysis (DA). Additionally, MVPAlab offers several data preprocessing routines: trial averaging, data smoothing and normalization, dimensionality reduction, among others. This MVPAlab GUI also includes a very flexible data representation utility, which generates really appealing and colorful plots and animations. In addition to this, MVPAlab implements some exclusive analyses and functionalities, such as parallel computation, which divides the computational load in different execution threads, significantly reducing the computation time, or frequency contribution analysis, which allows to estimate how relevant information is distributed across different frequency bands.

Hence, MVPAlab has not been designed for beginners only, as implements several high and low-level routines allowing more experienced profiles to design their own projects in a highly flexible manner. The following sections depict, in as much detail and as descriptively as possible, the main aspects of MVPAlab, including installation, compatibility, data structure, and a complete getting started section.

5.1.3. Installation, compatibility and requirements

The installation of MVPAlab Toolbox is quite simple. First, an up-to-date version of the code is freely available for download in the following GitHub repository:

```
github.com/dlopezg/mvpalab/releases
```

Users should (1) select and download the source code of the desired release, (2) unzip the downloaded source code folder and (3) add it to the MATLAB path with subfolders. Please see MVPAlab wiki for more detailed instructions:

```
github.com/dlopezg/mvpalab/wiki/Installation
```

The MVPAlab Toolbox has been designed to be fully compatible with MATLAB 9.0 (R2016a) and above. This restriction is only applicable to the graphic user interface, which has been developed using App Designer, introduced in the 9.0 version. Custom MVPAlab scripts can be executed under older MATLAB versions. Other toolboxes include several function names overlapping the MATLAB (or other external packages) built-in functions, causing in some cases errors and malfunctioning. To avoid this type of problems, MVPAlab uses a specific suffix in their function names. Since this software has been developed using MATLAB and has no external dependencies, the MVPAlab Toolbox is fully supported by GNU/Linux, Unix, Windows and macOS platforms. Hardware requirements depend on the size of the analyzed dataset. While the CPU specifications only affects to the computation time, enough RAM capacity is required to store and process M/EEG data. For almost any process, the recommended RAM capacity is at least the double of the size of the dataset (measured in gigabytes). For more memory demanding processes, such as frequency contribution analysis, MVPAlab splits and stores EEG data on the hard drive, importing it again when needed. Since MVPAlab only uses the CPU for computation, the GPU specification does not affect to the toolbox performance.

Some MATLAB built-in packages and functions are required for a correct functioning of this software. For the statistical analysis, the Image Processing Toolbox is required to find clusters in significant masks. The Statistics and Machine Learning Toolbox provides functions to train and validate classification models, dimensionality reduction, feature selection, etc. The Signal Processing Toolbox is required for extracting M/EEG envelopes as features. The Parallel Computation Toolbox is not required but recommended to drastically reduce the computation time as it allows to divide the computational load in different processing threads. Finally, MVPAlab greatly benefits from other open-source M/EEG toolboxes such as EEGlab and FieldTrip: some filtering functions require the EEGlab Toolbox installed and initiated for a correct operation. If MVPAlab finds an EEGlab installation it will initiate it automatically. Because of all of this, users should ensure that these dependencies are included in their MATLAB installation.

5.1.4. Dataset structure and format

MVPAlab is designed to read and work with epoched data from two of the most employed preprocessing toolboxes: EEGLAB and FieldTrip. For a correct operation of MVPAlab Toolbox, epoched data should be previously saved on one independent file for each subject using `.mat` format. EEGlab format `.set` is also supported. The data structure and format should remain unaltered. If EEGlab was used for the data preprocessing, users should save the entire `EEG` structure for each participant, not only the `EEG.data` matrix. MVPAlab collects additional information from the data file, such as sampling frequency (`EEG.srate`), the location of the electrodes (`EEG.chanlocs`) or data time points (`EEG.times`). In the same way, if FieldTrip is used, users must save the entire data structure, as MVPAlab reads the required subject's data from `data.trial`, `data.time` and `data.fsample`.

5.1.5. MVPAlab Toolbox architecture

The complete architecture of MVPAlab Toolbox is shown in Figure 35, including several of the configuration parameters, processes and routines employed for a complete decoding analysis. The complete architecture and its configuration parameters are resumed in the following stages:

Initialization stage. During the initialization stage, MVPAlab generates a default configuration structure. This variable is required for a correct operation of the toolbox.

Import data and feature extraction stage. Here, M/EEG data is imported, preprocessed, and prepared for the decoding analysis. During this stage, some specific configuration is required: the participants' files to import, identifiers for binary classes, the complete path to the dataset, and others. Additionally, users can select which M/EEG feature will be extracted for classification (raw signal voltage or its envelope); enable or disable and configure several preprocessing procedures, such as trial averaging, data normalization, balanced class sizes, and others. All these preprocessing procedures are computed during this stage. Finally, the feature vectors are extracted and prepared for the multivariate analysis.

Evaluation stage. During the evaluation stage, several classification models can be trained and validated using cross-validation approaches. Dimensionality reduction, if enabled, is also computed during this stage. Users can specify different classification models, linear and non-linear kernel functions, different cross-validation techniques, different model's performance metrics, etc. The results of the decoding analysis, the configuration file and other analysis-related files will be hierarchically stored in the project's directory. This directory is the folder containing the main analysis script.

Statistical significance stage. If permutation test is enabled, statistically significant clusters are extracted from the result via non-parametric cluster-based permutation testing. For this stage, users can specify the total number of permutations at a participant and group level to be computed, the p-value thresholds for a data point or cluster size to be considered significant and other relevant information.

Graphical representation stage. Last but not least is the graphical representation stage. MVPAlab has fully integrated high-level plotting tools, allowing researchers to easily design and generate high quality and highly customizable result representations.

5.1.6. Getting started

Computing a multivariate analysis in MVPAlab Toolbox is quite simple for all type of users. Researchers with no coding experience can use the integrated graphic user interface, which allows to create, save, configure, execute and plot the results of any supported multivariate analysis in a very intuitive way. Not a single line of code is needed. However, users with coding experience looking for a faster and more flexible way to interact with the toolbox can create their own scripts. Be that as it may, MVPAlab also includes several easy-to-understand and well-documented demo scripts for different types of analyses, making this tool very convenient not just for experienced users but also for newcomers. This section includes a general introduction to the functioning of MVPAlab Toolbox, either by using the GUI or building custom scripts.

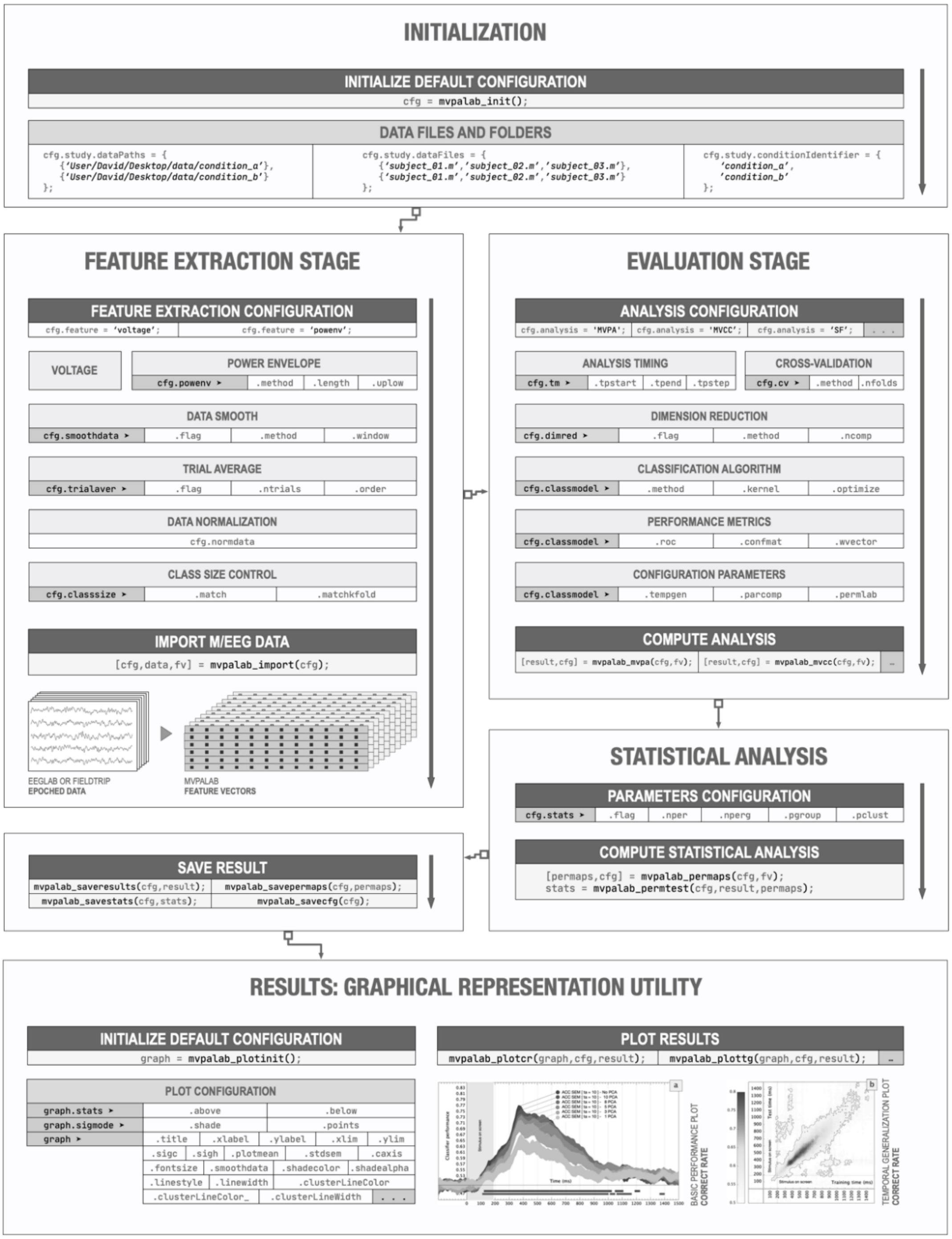
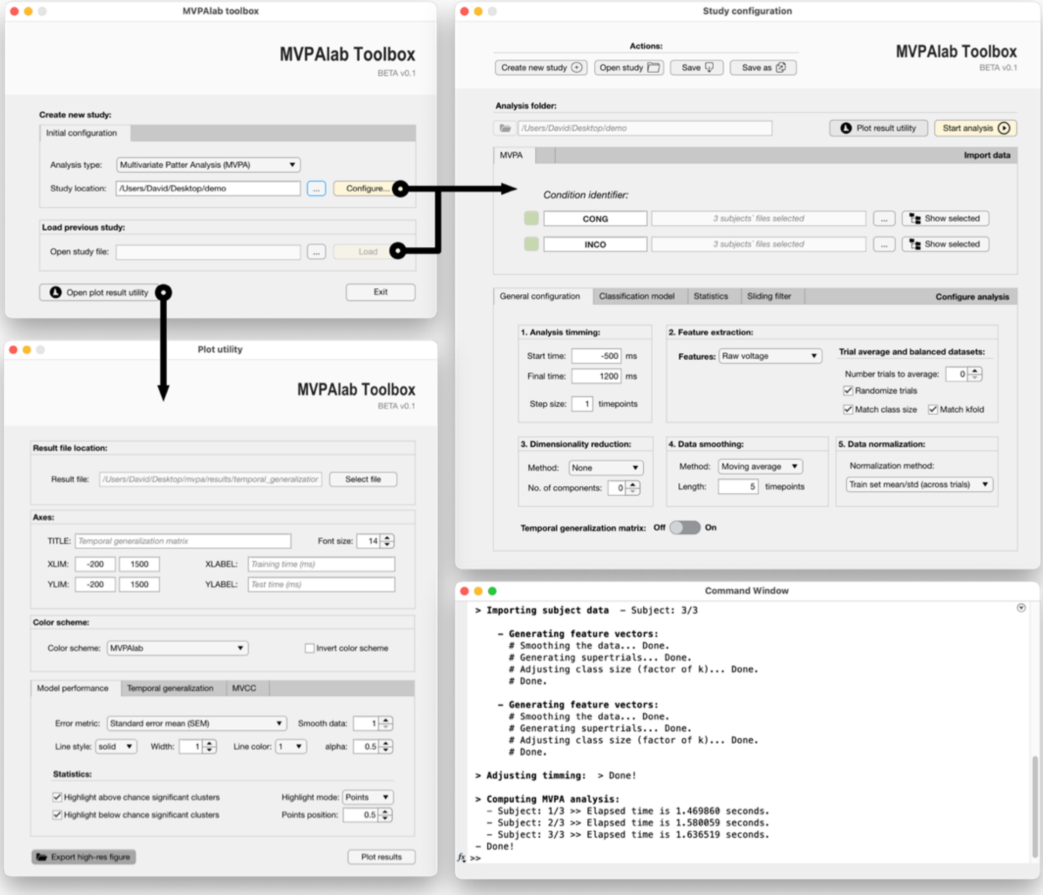


Figure 35. Complete architecture and configuration parameters of the MVPAlab

Graphic User Interface

Once MVPAlab is installed, the graphic user interface can be launched by typing the following command in the MATLAB command line:

```
>> mvpalab
```



Analysis type: Multivariate Patter Analysis (MVPA)
Study location: /Users/David/Desktop/demo
Open study file:

/Users/David/Desktop/demo
CONG 3 subjects` files selected
INCO 3 subjects` files selected

1. Analysis timming:
Start time: -500 ms
Final time: 1200 ms
Step size: 1 timepoints

2. Feature extraction:
Features: Raw voltage
Trial average and balanced datasets:
Number trials to average: 0
☑ Randomize trials
☑ Match class size ☑ Match kfold

3. Dimensionality reduction:
Method: None
No. of components: 0

4. Data smoothing:
Method: Moving average
Length: 5 timepoints

5. Data normalization:
Normalization method: Train set mean/std (across trials)

Temporal generalization matrix: Off On

Result file: /Users/David/Desktop/mvpa/results/temporal_generalizatior
TITLE: Temporal generalization matrix Font size: 14
XLIM: -200 1500 XLABEL: Training time (ms)
YLIM: -200 1500 YLABEL: Test time (ms)
Color scheme: MVPAlab □ Invert color scheme
Error metric: Standard error mean (SEM) Smooth data: 1
Line style: solid Width: 1 Line color: 1 alpha: 0.5
☑ Highlight above chance significant clusters Highlight mode: Points
☑ Highlight below chance significant clusters Points position: 0.5

```
> Importing subject data  - Subject: 3/3

    - Generating feature vectors:
      # Smoothing the data... Done.
      # Generating supertrials... Done.
      # Adjusting class size (factor of k)... Done.
      # Done.

    - Generating feature vectors:
      # Smoothing the data... Done.
      # Generating supertrials... Done.
      # Adjusting class size (factor of k)... Done.
      # Done.

> Adjusting timming:  > Done!

> Computing MVPA analysis:
  - Subject: 1/3 >> Elapsed time is 1.469860 seconds.
  - Subject: 2/3 >> Elapsed time is 1.580059 seconds.
  - Subject: 3/3 >> Elapsed time is 1.636519 seconds.
- Done!
fx >>
```

Figure 36. MVPAlab graphic user interface

(a) Initial view. (b) Analysis configuration view. (c) Plot utility view. (d) MATLAB command window.

Creating new analyses: If the MVPAlab folder is correctly added to the MATLAB path as described in Section 5.1.3, the initial MVPAlab window should appear as shown in Figure 36 (a). Using this interface, users can create new analyses, open previously created analyses or open the plotting utility. Creating new analyses in MVPAlab using the GUI is very simple and intuitive. Researchers only need to specify the type of analysis required from the dropdown menu and select the location folder. Results, configuration and other analysis-related files will be hierarchically stored in this directory. Once everything is selected, clicking the configuration button will create the project folder structure and launch the analysis configuration window, as shown in Figure 36 (b).

Configuring the decoding analysis: Before computing a multivariate analysis, additional details of configuration are required. Users must specify the locations of the epoched datasets and label each condition with a condition identifier. All the relevant parameters of the decoding analysis are set to its default value and can be modified within this configuration window. These configuration parameters include a wide range of processes that can be executed during the decoding analysis, such as: data normalization, data smoothing, trial averaging, analysis timing, dimensionality reduction, balance datasets and others. Additionally, the employed classification models can also be designed here. Users can choose between different classification algorithms, kernel functions, cross-validation strategies and select several output performance metrics. They can enable the computation of the temporal generalization matrix, activate parallel computation or configure statistical analyses. All MVPAlab toolbox functionalities are perfectly detailed in Section 5.2 Materials and Methods.

Computing the decoding analysis: Once the configuration parameters are correctly specified, the computation of the multivariate analysis can be started by clicking the *Start analysis* button. Depending on the size of the dataset and the selected configuration, this process may be time-consuming and CPU/memory demanding. Anyhow, during the computation of the entire analysis pipeline, as shown in Figure 36 (c), MVPAlab prompts in the MATLAB command window detailed information of the processes being executed.

Plotting the results: For the graphical representation of the results, MVPAlab also offers an intuitive plot utility that can be opened by clicking on *Open plot utility* button Figure 36 (d). This tool enables users to open, plot, combine and compare results of different analyses without dealing with cumbersome lines of MATLAB code. The most common configuration parameters such as titles, labels, line styles, transparencies, color palettes, axes limits, data smoothing or highlighting can be easily configured for time-resolved analysis, temporal generalization matrices, frequency contribution analyses, and others. In addition, with this interface users can create animated temporal representations of feature weights distribution over scalp templates.

All this combined allows researchers with no or little coding experience to prepare and compute multivariate decoding analyses of M/EEG data; create high quality and ready-to-publish figures, all of this without witting a single line of code.

Building custom scripts

The intuitive and easy-to-use GUI is not the only way to utilize this software. For those researchers looking for flexibility and automation, MVPAlab implements several high-level functions to easily set up a custom decoding analysis. The complete analysis pipeline can be divided into five main steps, including the statistical permutation test and plotting functions, and runs as follows:

```
% [1] - Initialize MVPAlab toolbox:
cfg = mvpalab_init();
```

```
% [2] - Run the configuration file:
run cfg_file.m
```

```
% [3] - Import data and extract feature vectors:
[cfg,data,fv] = mvpalab_import(cfg);
```

```
% [4] - Compute a multivariate analysis:
[result,cfg] = mvpalab_mvpa(cfg,fv);
```

```
% [5] - Plot the results:
run plot_file.m
```

First, the function `mvpalab_init()` initializes the toolbox. This function returns a default configuration structure `cfg`, which consist of all the required configuration parameters for an analysis. Please see the *Material and Methods* section for a detailed description of each field of the configuration variable.

Users should modify this configuration variable to set up the desired configuration for a specific decoding analysis. For the sake of clarity and for maintaining a clean code organization, all this configuration code should be placed in an external configuration file `cfg_file.m`. This file will be executed after the toolbox initialization.

Once the MVPAlab toolbox is initialized and a specific analysis configured, the function `mvpalab_import(cfg)` imports and preprocess the datasets provided, according to the configuration file `cfg`. This function returns a copy of the preprocessed data (`data`), which can be omitted to save memory, and the extracted feature vectors (`fv`), which will be the input for the classification models. Please see Section 5.2.3 Importing data and feature extraction for a more detailed explanation of the feature extraction process.

Next, the function `mvpalab_mvpa(cfg,fv)` computes the multivariate pattern analysis. Other functions are available for different analyses, such as `mvpalab_mvcc(cfg,fv)` for cross-classification and `mvpalab_sfilter(cfg,fv)` for frequency contribution analysis.

These functions return the variable `result`, which includes the time-resolved decoding performance for every performance metric enabled in the configuration file. In addition, the result files are automatically saved in separate folders in the project directory.

To compute the statistical analysis and draw statistical inferences at the group level, one additional step should be added to the former execution pipeline:

```
% Compute permutation test:
[permaps,cfg] = mvpalab_permaps(cfg,fv);
stats = mvpalab_permtest(cfg,result,permaps);
```

These functions implement a non-parametric cluster-based permutation test, returning the variable `stats`, which includes statistically significant clusters found in the data. Please, see Section 5.2.5 *Cluster-based permutation testing* for an exhaustive description of this test.

Finally, in addition to the graphic user interface, MVPAlab includes several plotting routines, allowing users to design customizable and ready-to-publish figures and animations. Please see Section 5.2.6 *Result representation pipeline* for more details. Several demo scripts for different types of analyses and result representations are included in the MVPAlab Toolbox folder.

5.2. Materials and Methods

5.2.1. Sample EEG dataset

A sample EEG dataset has been compiled to test all the MVPAlab main functionalities. It is freely available in the following repository:

```
https://osf.io/du6fa/
```

Here, three different EEG data files have been selected from the original work (López-García et al., 2019, 2020). For each participant, two different main conditions (*condition_a* vs. *condition_b*) have been selected for the MVPA analysis. Additionally, four subconditions (*condition_1, condition_2,* vs. *condition_3 and condition_4*) have been selected for the multivariate cross-classification analysis. Readers interested on the experimental details of these data should refer to the original publication (López-García et al., 2019, 2020).

During the original study, high-density EEG was recorded from 65 electrodes. The TP9 and TP10 electrodes were used to record the electrooculogram (EOG) and were removed from the dataset after the preprocessing stage. Impedances were kept below 5kΩ and EEG recordings were average referenced, downsampled to 256 Hz, and digitally filtered using a low-pass FIR filter with a cutoff frequency of 120 Hz, preserving phase information. No channel was interpolated for any participant. Continuous data were epoched [−1000, 2000ms centered at onset of the stimulus] and baseline corrected [−200, 0ms]. Independent Component Analysis (ICA) was computed to remove eye blinks from the signal, and the artifactual components were rejected by visual inspection of raw activity of each component, scalp maps and power spectrum. Finally, an automatic trial rejection process was performed, pruning the data from non-stereotypical artifacts. For more details please see (López-García et al., 2020).

The final compiled dataset consists of an EEGlab data structure per subject and condition with `[63 x 768 x ntrials]` EEG data matrices. The number of trials per condition and participant is shown in the following table:

Table 4. Total number of trials per subject and condition in the demo dataset

	subject_01.mat	subject_02.mat	subject_03.mat
condition_a	468	413	434
condition_b	403	399	396
condition_1	212	193	190
condition_2	218	202	212
condition_3	191	206	206
condition_4	250	211	222

5.2.2. Defining a configuration file

For the sake of clarity and code organization, we recommend including all the configuration code for a specific decoding analysis in an external configuration `.m` file.

This file should be executed before the computation of the multivariate decoding analysis. This recommendation, however, is not mandatory and more experienced users can design their own scripts according to their needs and preferences. For both scenarios, all the available configuration parameters in MVPAlab Toolbox will be described in detail during this section. For the sake of clarity and code organization, we recommend including all the configuration code for a specific decoding analysis in an external configuration `.m` file. This file should be executed before the computation of the multivariate decoding analysis. This recommendation, however, is not mandatory and more experienced users can design their own scripts according to their needs and preferences. For both scenarios, all the available configuration parameters in MVPAlab Toolbox will be described in detail during this section.

Participants and data directories

The first required information that should be specified by the user is the working directory and the location of the dataset to be imported and analyzed. This includes, for each class or condition, the name of each individual subject data file and the complete path of the class folder. These parameters can be defined in the configuration file as follows:

```
% Working directory:
cfg.location = pwd;

% Conditions data paths:
cfg.dataPaths{1,1} = 'C:\...\class_a\';
cfg.dataPaths{1,2} = 'C:\...\class_b\';


% Subjects data files:
cfg.dataFiles{1,1} = {
      ‘subject_01.mat’,
      ‘subject_02.mat’,
      ‘subject_03.mat’
};
cfg.dataFiles{1,2} = {
      ‘subject_01.mat’,
      ‘subject_02.mat’,
      ‘subject_03.mat’
};
```

Before computing the multivariate decoding analysis, the MVPAlab Toolbox can be used to execute several preprocessing procedures that may improve the final results in different ways (e.g. increasing accuracy, avoiding skewed results, data normalization, data smoothing, etc.). The default configuration of each of these procedures is initialized when MVPAlab toolbox is launched. However, these procedures and their configuration parameters can be adjusted by the users to meet the required specific analysis conditions. During this section, all of these preprocessing procedures and their configuration parameters will be meticulously described.

Trial averaging

If enabled, this approach randomly or sequentially averages a certain number of trials $\mathbf{n_{trials}}$ belonging to the same condition for each participant. This procedure creates *supertrials* and usually increases the signal-to-noise ratio (SNR) which improves the overall decoding performance and reduces the computational load. Since reducing the number of trials per condition typically increases the variance in the decoding performance, this procedure imposes a trade-off between the increased variance/accuracy. It should be noted that increasing $\mathbf{n_{trials}}$ does not increase the decoding performance linearly. Please see (Grootswagers et al., 2017; Isik et al., 2014a) for more details.

The default parameters for this procedure can be modified in the MVPAlab configuration file as follows:

```
cfg.trialaver.flag       = true;
cfg.trialaver.ntrials    = 5;
cfg.trialaver.order      = 'rand';
```

Trial averaging can be enabled or disabled by setting the configuration variable (.flag) to true or false. The number of trials to average can be modified in (.ntrials). Finally, the order in which the trials are selected for averaging can be modified setting the variable (.order) to ‘rand' or 'sequential'.

Balanced dataset

Unbalanced datasets can lead to skewed classification results (Sun et al., 2009). To avoid this phenomenon, the number of trials per condition should be the same. MVPAlab can be used to define strictly balanced datasets by downsampling the majority class to match the size of the minority one (`cfg.classsize.match`). In addition, each class size can be set as a factor of k, the total number of folds in the cross-validation (CV) procedure. Thus, during CV each fold will be composed by exactly the same number of observations, avoiding any kind of bias in the results (`cfg.classsize.matchkfold`).

These features are disabled by default but can be enabled in the MVPAlab configuration structure as follows:

```
cfg.classsize.match       = true;
cfg.classsize.matchkfold  = true;
```

Data normalization

In machine learning, data normalization refers to the process of adjusting the range of the M/EEG raw data to a common scale without distorting differences in the ranges of values. Although classification algorithms work with raw values, normalization usually improves the efficiency and the performance of the classifiers (Singh & Singh, 2020). Four different (and excluding) data normalization methods are implemented in MVPAlab. A commonly used normalization approach (J. R. King et al., 2013) is computed within the cross-validation loop. Hence, the training and test sets are standardized as follows:

$$\mathrm{X}_{\mathrm{train}} = \frac{\mathrm{X}_{\mathrm{train}} - \mu_{\mathrm{train}}}{\sigma_{\mathrm{train}}} \qquad \mathrm{X}_{\mathrm{test}} = \frac{\mathrm{X}_{\mathrm{test}} - \mu_{\mathrm{train}}}{\sigma_{\mathrm{train}}}$$

Equation 16

where μ_{train} and σ_{train} denote the mean and the standard deviation of each feature (column) of the training set. Other normalization methods implemented in MVPAlab are: z-score ($\mu = 0$; $\sigma = 1$) across time, trial or features. To compute these normalization strategies MVPAlab uses the MATLAB built-in function *zscore*, included in the Statistics and Machine Learning Toolbox.

Data normalization method, which is disabled by default, can be modified as follows:

```
cfg.normdata = 4;  % 0 - Disabled
                   % 1 - ZSCORE across features
                   % 2 - ZSCORE across time
                   % 3 - ZSCORE across trials
                   % 4 - Nested in CV loop
```

Data smoothing

Data smoothing is a procedure employed in recent M/EEG studies (Isik et al., 2014b; Kerrén et al., 2018; LaRocque et al., 2013; Shatek et al., 2019) to attenuate unwanted noise. MVPAlab implements an optional data smoothing step that can be computed before multivariate analyses. This procedure is based on MATLAB built-in function *smooth*, included in the Curve Fitting Toolbox, which smooths M/EEG data points using a moving average filter.

The length of the smoothing window can be specified in the variable (`cfg.smoothdata.window`) and should be an odd number. For a window length of 5 time points, the smoothed version of the original signal is computed as follows:

$$
\begin{aligned}
&\mathrm{y}_{\mathrm{smoothed}}(1) = \mathrm{y}(1) \\
&\mathrm{y}_{\mathrm{smoothed}}(2) = \frac{\left(\mathrm{y}(1) + \mathrm{y}(2) + \mathrm{y}(3)\right)}{3} \\
&\mathrm{y}_{\mathrm{smoothed}}(3) = \frac{\left(\mathrm{y}(1) + \mathrm{y}(2) + \mathrm{y}(3) + \mathrm{y}(4) + \mathrm{y}(5)\right)}{5} \\
&\mathrm{y}_{\mathrm{smoothed}}(4) = \frac{\left(\mathrm{y}(2) + \mathrm{y}(3) + \mathrm{y}(4) + \mathrm{y}(5) + \mathrm{y}(6)\right)}{5} \\
&\ldots
\end{aligned}
$$

Equation 17

Data smoothing is disabled `(.method = 'none')` by default and can be enabled and configured in the MVPAlab configuration file as follows:

```
cfg.smoothdata.method  = 'moving';
cfg.smoothdata.window  = 5;
```

Analysis timing

By default, MVPAlab computes the time-resolved decoding analysis for each timepoint across the entire M/EEG epoch. However, the user can define a specific region of interest (time window) and a different step size as follows:

```
cfg.tm.tpstart   = -200;
cfg.tm.tpend     = 1500;
cfg.tm.tpsteps   = 3;
```

This way, the temporal decoding analysis will be computed from -200ms (`.tpstart`) to 1500ms (`.tpend`) not for each timepoint but for every three (`.tpsteps`) timepoints. Note that increasing the step size decreases the processing time but also causes a reduction in the temporal resolution of the decoding results.

Dimensionality reduction

In machine learning, dimension reduction techniques are a common practice to reduce the number of variables in high-dimensional datasets. During this process, the features contributing more significantly to the variance of the original dataset are automatically selected. In other words, most of the information contained in the original dataset can be represented using only the most discriminative features. As a result, dimensionality reduction facilitates, among others, classification, visualization, and compression of high-dimensional data (Van Der Maaten et al., 2009). There are different dimensionality reduction approaches but Principal Component Analysis (PCA) is probably the most popular multivariate statistical technique used in almost all scientific disciplines (Abdi & Williams, 2010), including neuroscience (Hebart et al., 2018b). PCA in particular is a linear transformation of the original dataset in an orthogonal coordinate system in which axis coordinates (principal components) correspond to the directions of highest variance sorted by importance. To compute this transformation, each row vector $\mathbf{x_i}$ of the original dataset $\mathbf{X}$ is mapped to a new vector of principal components $\mathbf{t_i} = (\mathbf{t_1}, \dots, \mathbf{t_l})$, also called scores, using a p-dimensional coefficient vector $\mathbf{w_j} = (\mathbf{w_1}, \dots, \mathbf{w_p})$:

$$\mathrm{t_i = x_i \cdot w_j \qquad i = 1, \dots, n \qquad j = 1, \dots, l}$$

Equation 18

For dimension reduction: $\mathbf{l < p}$.

To maintain the model's performance as fair and unbiased as possible, PCA is computed only for training sets $\mathbf{X}_{\mathbf{training}}$, independently for each fold inside the cross-validation procedure. Once PCA for the corresponding training set is computed and the model is trained, the exact same transformation is applied to the test set $\mathbf{X}_{\mathbf{test}}$ (including centering, $\boldsymbol{\mu}_{\mathbf{training}}$). In other words, the test set is projected onto the reduced feature space obtained during the training stage. According to the former equation, this projection is computed as follows:

$$\mathrm{T}_{\mathrm{test}} = \frac{\mathrm{X}_{\mathrm{test}} - \mu_{\mathrm{training}}}{\mathrm{W}'_{\mathrm{training}}}$$

Equation 19

To compute this nested implementation of the PCA algorithm, MVPAlab uses the MATLAB built-in function *pca*, included in the Statistics and Machine Learning Toolbox. However, dimensionality reduction techniques such PCA endorse a trade-off between the benefits of dimension reduction (reduced training time, reduced redundant data and improved accuracy) and the interpretation of the results when electrodes are used as features. When PCA is computed, the data is projected from the sensor space onto the reduced PCA features space. This linear transformation implies an intrinsic loss of spatial information, which means that, for example, we cannot directly analyze which electrodes are contributing more to decoding performance. The default parameters for this procedure can be modified in the MVPAlab configuration file as follows:

```
cfg.dimred.flag    = true;
cfg.dimred.method  = 'pca';
cfg.dimred.ncomp   = 5;
```

Classification algorithms

Classification algorithms are the cornerstone of decoding analyses. These mathematical models play the central role in multivariate analyses: detect subtle changes in patterns in the data that are usually not detected using less sensitive approaches. Different classification algorithms have been used to achieve this goal, from probabilistic-based models such as Discriminant Analyses (DA), Logistic Regressions (LR) or Naïve Bayes (NB) to supervised learning algorithms such Support Vector Machine (SVM).

For the time being, MVPAlab Toolbox implements two of the most commonly employed models in the neuroscience literature, Support Vector Machines and Discriminant Analysis in their linear and non-linear variants.

The classification model employed for the decoding analysis can be specified in the configuration file as follows:

```
cfg.classmodel.method = 'svm';
cfg.classmodel.method = 'da';
```

Both classification approaches are based on MATLAB built-in libraries for support vector machines and discriminant analyses. A brief mathematical description for both models can be found below. Please see the MATLAB documentation of `fitcsvm` and `fitcdiscr` functions for further details.

Support Vector Machine: Support Vector Machine (SVM) provides a theoretically elegant, computationally efficient, and very effective solution for many practical pattern recognition problems (Boser et al., 1992; Cortes & Vapnik, 1995; Cristianini et al., 2000). For that reason, SVM is broadly employed in M/EEG studies. Intuitively, for binary classification problems, during the training stage this algorithm searches for an optimal hyperplane maximizing the separation between this hyperplane and the closest data points of each class. These data points are called *support vectors*. The separation space is called *margin* and is defined as $2/\|\mathbf{w}\|$, and it does not contain any observation for separable classes, as shown in Figure 37(a). Thus, the linear SVM score function is defined as follows:

$$f(\mathrm{x}) = \mathbf{x}^{\mathsf{T}}\mathbf{w} + b$$

Equation 20

where the input vector $\mathbf{x}$ is an observation, the vector $\mathbf{w}$ contains the coefficients that define an orthogonal vector to the hyperplane and b is the bias term. To formalize the optimization problem (that is, to find the optimal hyperplane that maximizes the margin), several constraints should be defined. Therefore, any given sample will be correctly classified as long as:

$$
\begin{array}{ll}
\mathbf{x}^{\mathsf{T}}\mathbf{w} + b \geq +1 & \text{for positive (+) samples} \\
\mathbf{x}^{\mathsf{T}}\mathbf{w} + b \leq -1 & \text{for negative (-) samples}
\end{array}
$$

Equation 21

Introducing $\mathrm{y_j} = \{+1, -1\}$ for positive and negative samples, respectively, the two former equations can be rewritten for mathematical convenience as follows:

$$
\mathrm{y_j} f(\mathrm{x_j}) \geq 1 \ \text{ for any training sample } j \in \{1, \dots, \mathrm{n}\}
$$

Equation 22

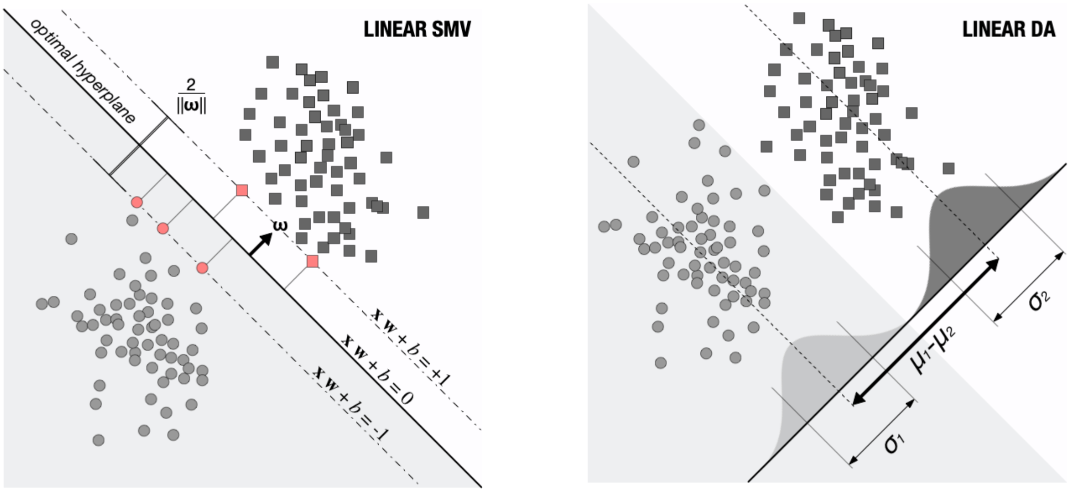


Figure 37. 2D representation of different classification models

Graphical representation of (a) LSVM and (b) LDA classifiers for simulated data. Red points represent the support vectors, the closest data points to the decision boundary (hyperplane).

This is the decision rule for separable classes. When the classes are not perfectly separable, the algorithm imposes a penalty introducing positive slack variables $\xi_\mathrm{j} > 0$ for each observation on the wrong side of the hyperplane. For those observations that are correctly placed: $\xi_\mathrm{j} = 0$. Consequently, non-separable data impose a trade-off between margin maximization and the total number of constraint violations. Now, the optimization problem reads as follows:

$$\arg_{\mathbf{w}} \min \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{\mathrm{j}=1}^{\mathrm{n}} \xi_{\mathrm{j}}$$

Equation 23

with respect to **w** and *b* and subject to:

$$\forall j\colon y_j f\left(\mathrm{x}_j\right) \geq 1 - \xi_{\mathrm{j}} \quad \text{and} \quad \forall j\colon \xi_{\mathrm{j}} \geq 0$$

Equation 24

The parameter *C* is a constant which modulates the trade-off between the training error and the complexity of the model. A search-grid-based optimization of the misclassification cost parameter *C* can be enabled and computed using five-fold CV for the training set on the configuration file as follows:

```
cfg.classmodel.optimize.flag = true;
```

For some classification scenarios, it is not always possible to find an optimal criterion for class separation using linear classifiers. To solve this problem, original data from the input space $\mathcal{N}$ can be mapped into a high dimensional feature space $\mathcal{F}$ using a mapping function $\phi\colon \mathcal{N} \longmapsto \mathcal{F}$. Therefore, the decision equation is now defined as follows:

$$f(\mathrm{x}) = \phi(\mathbf{x}^{\top})\mathbf{w}_{\phi} + b$$

Equation 25

However, the application of the transformation function ϕ is not explicitly needed. Since the hyperplane optimization problem depends on nothing but pairwise dot products (e.g. $\mathbf{x}_1 \cdot \mathbf{x}_2$), we only need a set of kernel functions that meet the following property: $K(\mathbf{x}_1, \mathbf{x}_2) = \langle \phi(\mathbf{x}_1), \phi(\mathbf{x}_2) \rangle$.

This class of function includes, among others, polynomial or gaussian kernels:

$$\mathrm{G}(\mathbf{x}_1, \mathbf{x}_2) = (1 + \mathbf{x}_1 \mathbf{x}_2)^{\mathrm{p}}$$
$$\mathrm{G}(\mathbf{x}_1, \mathbf{x}_2) = \mathrm{e}^{-\|-\mathbf{x}_1 - \mathbf{x}_2\|^2}$$

Equation 26

The mentioned variant of the initial mathematical approach for non-linear classifiers is known as *kernel trick* (Figure 38) and it retains nearly all the simplicity and benefits of linear approaches, making data linearly separable in the feature space $\mathscr{F}$. However, in decoding analyses, linear approaches are normally preferred not just for their simplicity, but also for yielding comparable performance results in several applications (Misaki et al., 2010).

MVPAlab uses linear classifiers for decoding analysis by default, but other kernel functions for non-linear classification can be specified in the MVPAlab configuration file as follows:

```
cfg.classmodel.kernel = 'linear';
cfg.classmodel.kernel = 'gaussian';
cfg.classmodel.kernel = 'rbf';
cfg.classmodel.kernel = 'polynomial';
```

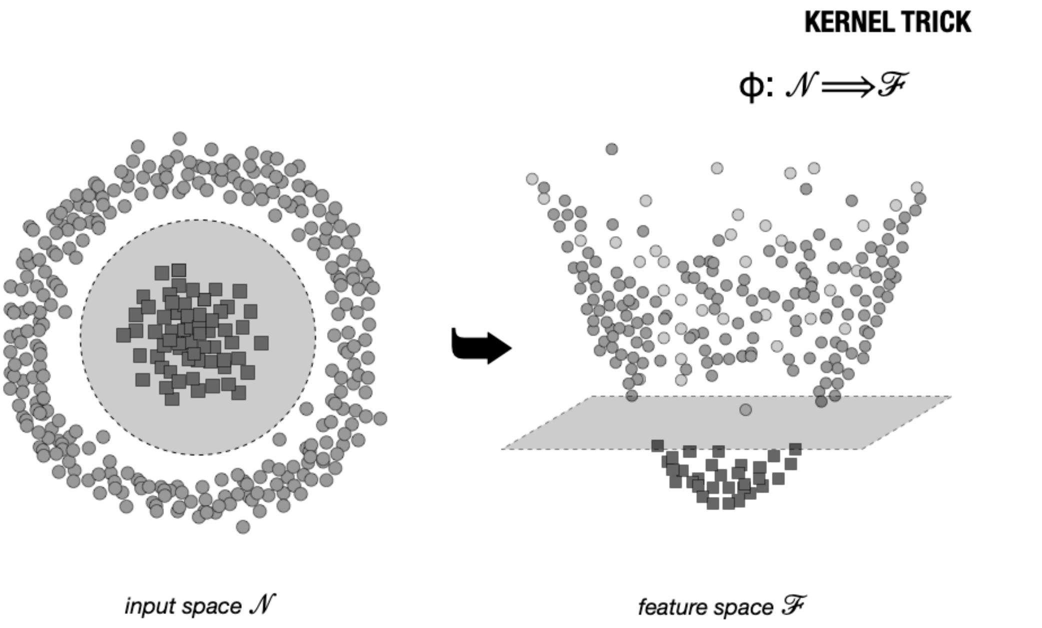

KERNEL TRICK
$\phi: \mathcal{N} \Longrightarrow \mathcal{F}$
input space $\mathcal{N}$
feature space $\mathcal{F}$

Figure 38. Graphical representation of the kernel trick

The original data in the input space is not linearly separable. This data points can be projected into a high-dimensional space using the mapping function ϕ. In this new feature space, classes became separable using linear approaches.

Discriminant analysis: Prediction using Discriminant Analysis (DA), see Figure 37b, is based in three different metrics: posterior probability, prior probability and cost. Thus, the classification procedure tries to minimize the expected classification cost:

$$\hat{\mathrm{y}} = \arg\min \sum_{\mathrm{k}=1}^{\mathrm{K}} \hat{\mathrm{P}}(\mathrm{k}|\mathrm{x})\mathrm{C}(\mathrm{y}|\mathrm{k})$$

Equation 27

where $\hat{\mathbf{y}}$ is the predicted classification, $\mathbf{K}$ corresponds to the number of classes, $\hat{\mathbf{P}}(\mathbf{k}|\mathbf{x})$ is the posterior probability of class **k** for observation **x** and $\mathrm{C}(\mathrm{y}|\mathrm{k})$ is the cost of classifying an observation as **y** when its true class is **k**.

Being $\mathbf{P}(\mathbf{k})$ the prior probability of class **k**, the posterior probability that an observation **x** belongs to class **k** is:

$$\hat{\mathbf{P}}(\mathbf{k}|\mathbf{x}) = \frac{\mathrm{P}(\mathrm{x}|\mathrm{k})\mathrm{P}(\mathrm{k})}{\mathrm{P}(\mathrm{x})}$$

Equation 28

where:

$$\mathbf{P}(\mathbf{k}|\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^{d}|\Sigma_{\mathrm{k}}|}} \exp\left(-\frac{1}{2}\,(\mathrm{x}-\mu_{\mathrm{k}})\Sigma_{\mathrm{k}}^{-1}(\mathrm{x}-\mu_{\mathrm{k}})^{\mathrm{T}}\right)$$

Equation 29

is the multivariate normal density function, being $\mathbf{\Sigma_k}$ the ***d***-by-***d*** covariance matrix and $\boldsymbol{\mu}_{\mathbf{k}}$ the 1-by-***d*** mean. Please see the MATLAB documentation for further details.

While Linear Discriminant Analyses (LDA) assumes that both classes have the same covariance matrices $\mathbf{\Sigma_k}$ and only the means $\boldsymbol{\mu}_{\mathbf{k}}$ vary, for Quadratic Discriminant analyses (QDA), both means and covariance matrices may vary. Thus, decision boundaries are determined by straight lines in LDA and by conic sections (ellipses, hyperbolas or parabolas) for QDA.

Linear Discriminant analysis is configured by default in MVPAlab Toolbox but, as for SVM, this kernel function can be modified in the configuration file as follows:

```
cfg.classmodel.kernel = 'quadratic';
```

Cross-validation

In prediction models, cross-validation techniques are used to estimate how well the classification algorithm generalizes to unknow data. Two popular approaches for evaluating the performance of a classification model on a specific data set are *k-fold* and *leave-one-out* cross validation (Wong, 2015). In general, these techniques randomly split the original dataset into two different subsets, the training set $\mathbf{X_{training}}$**:** $\mathbf{1 - 1/K}$ percent of the exemplars, and the test set $\mathbf{X}_{\mathbf{test}}$: $\mathbf{1/K}$ percent of the exemplars. This procedure is repeated $\mathbf{K}$ times (folds), selecting different and disjoint subsets for each iteration. Thus, for each fold, the classification model is trained for the training set and evaluated using exemplars belonging to the test set. The final classification performance value for a single timepoint is the mean performance value for all iterations.

When $\mathbf{K}$ and the total number of exemplars (instances) are equal, this procedure is called *leave-one-out* cross-validation. Here, the classification model is trained with all but one of the exemplars and evaluated with the remaining exemplar. By definition, this approach is computationally demanding and time consuming for large datasets, and for that reason it is usually employed only with small sets of data. Additionally, the leave-one-out procedure has been proved to yield unstable and biased results, which makes random splits methods the preferred alternative (Varoquaux et al., 2017).

The cross-validation procedure can be tuned in the MVPAlab configuration file as follows:

```
cfg.cv.method  = 'kfold';
cfg.cv.nfolds  = 5;
```

If (`.method = 'loo'`) the number of folds is automatically updated to match the total number of exemplars for each participant.

Performance metrics

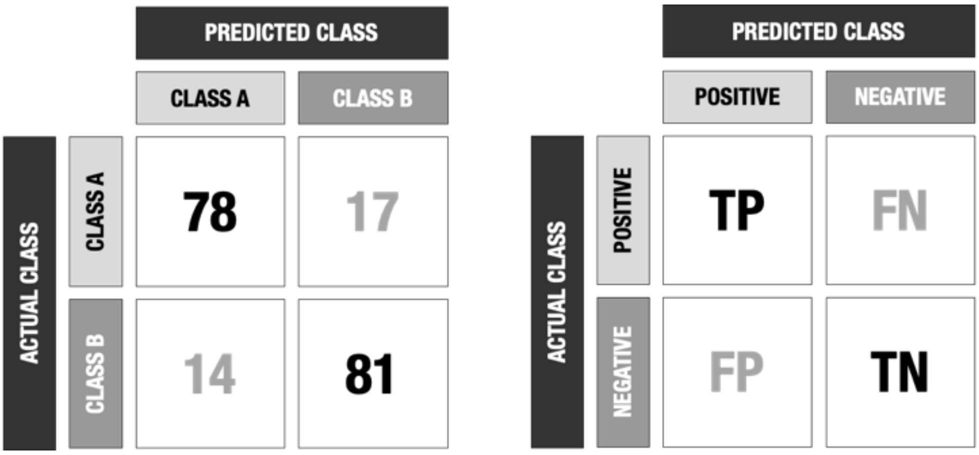
(1) Mean accuracy is usually employed to evaluate decoding models' performance in neuroscience studies (Combrisson & Jerbi, 2015). This metric is fast, easy to compute and is defined as the number of hits over the total number of evaluated trials. By default, MVPAlab Toolbox returns the mean accuracy value as a measure of decoding performance. Nevertheless, in situations with very skewed sample distributions, this metric may generate systematic and undesired biases in the results. Other performance metrics, such as the balanced accuracy have been proposed to mitigate this problem (Brodersen et al., 2010).

Accuracy values can be complemented with the *(2) confusion matrices*, which are very useful for binary classification but even more so for multiclass scenarios. In machine learning, a confusion matrix allows the visualization of the performance of an algorithm (see Figure 39), reporting false positives (FP), false negatives (FN), true positives (TP), and true negatives (TN). To this end, a confusion matrix reflects the predicted versus the actual classes. Rows correspond to true class and columns to predicted classes. Thus, the element $\mathbf{CM_{i,j}}$ indicates the number (or the proportion) of exemplars of class $\mathbf{i}$ classified as class $\mathbf{j}$. Other interesting and more informative performance metrics available in MVPAlab are derivations of the confusion matrix:

(3) *Precision* $\mathbf{PR = TP/(TP + FP)}$: proportion of trials labeled as positive that actually belong to the positive class.

(4) *Recall* (also known as *sensitivity*) $\mathbf{R = TP/(TP + FN)}$: proportion of positive trials that are retrieved by the classifier.

(5) *F1-score* $\mathbf{F1 = 2TP/(2TP + FP + FN)}$: combination of precision and recall in a single score through the harmonic mean.



		PREDICTED CLASS	
		CLASS A	CLASS B
ACTUAL CLASS	CLASS A	78	17
	CLASS B	14	81

		PREDICTED CLASS	
		POSITIVE	NEGATIVE
ACTUAL CLASS	POSITIVE	TP	FN
	NEGATIVE	FP	TN

Figure 39. Confusion matrix

Example of a confusion matrix returned by MVPAlab Toolbox for a binary classification scenario.

Nonetheless, nonparametric, criterion-free estimates, such as the Area Under the ROC Curve (AUC), have been proved as a better measure of generalization for imbalanced datasets (J.-R. King & Dehaene, 2014). This curve is used for a more rigorous examination of a model's performance. The AUC provides a way to evaluate the performance of a classification model: the larger the area, the more accurate the classification model is. This metric is one of the most suitable evaluation criteria, as it shows how well the model distinguishes between conditions, by facing the sensitivity (True Positive Rate (TPR)) against 1-specificity (False Positive Rate (FPR)), defined as follows:

$$\mathbf{AUC} = \int_{\mathbf{0}}^{\mathbf{1}} \mathbf{ROC}(\boldsymbol{s})\mathbf{d}\boldsymbol{s}$$

Equation 30

To compute the AUC and the ROC curve MVPAlab utilizes the MATLAB built-in function `perfcurve`, included in the Statistics and Machine Learning Toolbox.

By default, MVPAlab only returns the mean accuracy, although other performance metrics can be enabled in the configuration file as follows:

```
cfg.classmodel.roc       = false;
cfg.classmodel.auc       = false;
cfg.classmodel.confmat   = false;
cfg.classmodel.precision = false;
cfg.classmodel.recall    = false;
cfg.classmodel.f1score   = false;
```

Users should be aware that enabling several performance metrics will significantly increase the computation time and memory requirements to store the results.

Parallel computation

The MVPAlab Toolbox is adapted and optimized for parallel computation. If the Parallel Computing Toolbox (MATLAB) is installed and available, MVPAlab can compute several timepoints simultaneously. Therefore, the computational load is distributed among the different CPU cores, significantly decreasing the processing time. This feature becomes critical specially when the user is dealing with large datasets and needs to compute several thousand of permutation-based analyses. Parallel computation is disabled by default but can be enabled in the MVPAlab configuration file as follows:

```
cfg.classmodel.parcomp  = true;
```

5.2.3. Importing data and feature extraction

To obtain the classification performance in a time-resolved way, the epoched M/EEG data must be prepared for the classification process. During the feature extraction step, feature vectors are defined as a selection/combination of variables of the original dataset. Typical multivariate analyses use the raw voltage of the signal as a feature for the classification, but other characteristics, such the power envelope of the signal, can also be used as features. For each participant, time point and trial, two feature vectors (one for each condition or class) are generated, consisting of the raw potential (or any other feature such the power envelope) measured in all electrodes.

Once MVPAlab is initialized and the analysis configuration parameters are defined in `cfg_file.m`, the function **`mvpalab_import(`**`cfg`**`)`** imports the original dataset and returns an updated version of the configuration structure (`cfg`), the preprocessed data (`data`) and feature vectors (`fv`):

```
% Initialize MVPAlab toolbox and run cfg file:
cfg = mvpalab_init();
run cfg_file.m

% Import data and extract feature vectors:
[cfg,data,fv] = mvpalab_import(cfg);

...
```

The feature vector and data variables are cell arrays structured as follows: `[1 x subjects]`. Each cell in `fv` contains a data matrix (`X`) with the feature vectors of individual subjects `[trials x features x timepoints]` and a logical vector (`Y`) including the true labels of the subject`s dataset. The data variable contains, for each condition, a data matrix including the preprocessed dataset `[features x timepoints x trials]`.

5.2.4. Type of analysis

The MVPAlab Toolbox computes two main analyses: time-resolved Multivariate Pattern Analysis (TR-MVPA) and time-resolved Multivariate Cross-Classification (TR-MVCC). Different types of analyses such the Temporal Generalization, the Feature Contribution Analysis or the Frequency Contribution Analysis are derived from them.

Time-Resolved Multivariate Pattern Analysis

Multivariate Pattern Analyses, also known as decoding analyses, comprise a set of machine learning models that extract information patterns from multi-dimensional data. One of the most remarkable advantages of these multivariate over univariate techniques is its sensitivity in detecting subtle changes in the patterns of activations, considering information distributed across all sensors simultaneously.

To compute a time-resolved Multivariate Pattern Analysis, a classification model is trained and cross-validated for each time point and participant individually, extracting different performance metrics according to the cfg structure (see Figure 40)**.** All this process is coded in the function mvpalab_mvpa(cfg,fv), which computes the decoding analysis completely:

```
...

% Import data and extract feature vectors:
[cfg,data,fv] = mvpalab_import(cfg);

% Compute MVCC analysis:
[result,cfg] = mvpalab_mvpa(cfg,fv);

...
```

This function returns an updated version of the configuration structure (`cfg`) and the result variable (`result`). Performance values are stored in data matrices `[1 x time x subject]` inside the result variable as shown in the following figure:

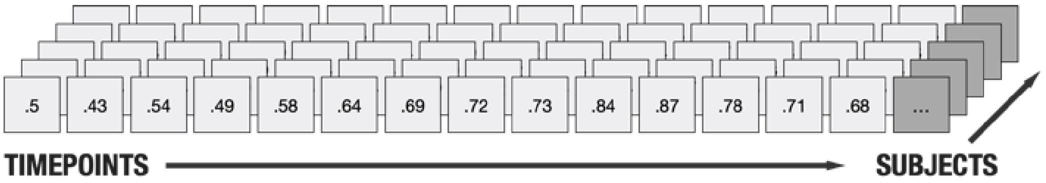


Figure 40. Data structure of the results file

Performance values are stored in 1 x timepoint x subject matrices. Group-level performance values can be calculated computing the mean across the third dimension.

For example, the time-resolved accuracy values can be extracted from `result.acc`. Other class-specific performance metrics such as f1-score, recall or precision are stored for each condition in:

```
result.f1score.condition_1
result.f1score.condition_2
result.f1score.mean
```

Time-Resolved Multivariate Cross-Classification

As mentioned before, the former MVPA technique has the ability to detect subtle differences in brain activation patterns. Thus, this powerful capacity could be used to study how these patterns are consistent across different cognitive contexts. In general, the consistency of the information across different sets of data can be analyzed with these techniques. To this end, classification models are trained with one set of data and the consistency is assessed by testing these models with another data sets, belonging to a different experimental condition. This technique is called Multivariate Cross-Classification (MVCC) (Kaplan et al., 2015) and is growing in popularity in recent years (Etzel et al., 2008; Oosterhof et al., 2010, 2013).

It is important to stress that different results can be obtained depending on which set is used for training and which one for testing (Train: $\mathbf{A}$ → Test: $\mathbf{B}$ or Train: $\mathbf{B}$ → Test: $\mathbf{A}$). This is called classification direction. The observation of classification direction asymmetries in MVCC can be explained by several and very different phenomena, including complex neurocognitive mechanisms or a simple signal-to-noise ratio difference across datasets. For this reason, reporting results in both directions is highly recommended (Hurk & Beeck, 2019). By default, MVPAlab computes and reports both directions separately.

To compute the MVCC analysis, the function `mvpalab_mvcc` should be called after the feature extraction stage:

```
...

% Import data and extract feature vectors:
[cfg,data,fv] = mvpalab_import(cfg);

% Compute MVCC analysis:
[result,cfg] = mvpalab_mvcc(cfg,fv);
```

Similar to previous analysis, this function returns an updated version of the configuration structure and the results variable. In this case, time resolved accuracy values are stored for both classification directions in:

```
result.acc.ab
result.acc.ba
```

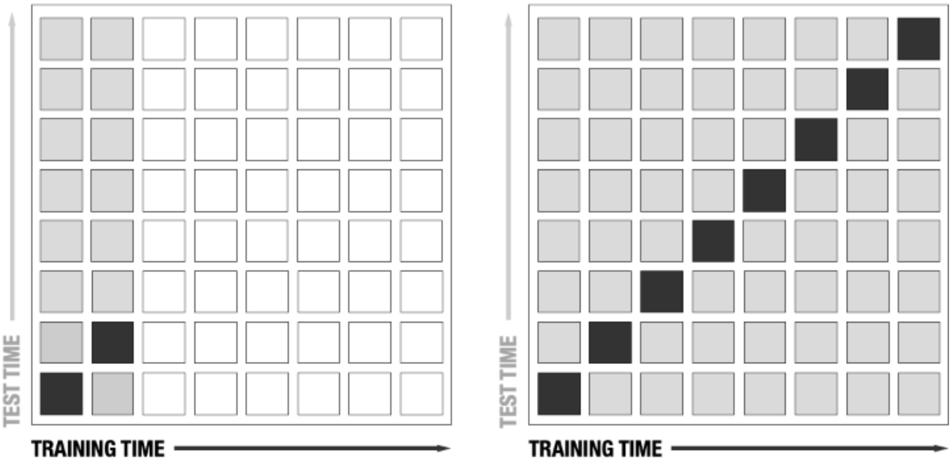


Figure 41. Temporal generalization routine

The classification model is trained with data at certain time point (black square). This model is then tested along the remaining time points (grey square), repeating this process for each time point inside the epoch.

Temporal generalization matrix

To evaluate the stability of brain patterns along time, temporal generalization analyses are commonly used. To obtain the temporal generalization matrix, the model is trained in a specific temporal point, testing its ability to discriminate between conditions in the whole temporal window. This process is then repeated for every timepoint thus obtaining the final decoding accuracy matrix (see Figure 41). An above-chance discrimination rate outside the diagonal of the matrix suggests that the same activity pattern is sustained in time. This phenomenon is usually interpreted as a reactivation of neural representations (J.-R. King & Dehaene, 2014). Therefore, if there is no evidence of temporal generalization, different patterns of activity can be inferred (Hebart et al., 2018b). However, a recent study demonstrated that this interpretation is not always valid, suggesting that this phenomenon can be explained as an artefact of the manner in which the decoding accuracy provided by different components of the signal combine to bring about the overall decoding accuracy (Vidaurre et al., 2020).

Regardless of the previously selected type of analysis (MVPA or MVCC), the calculation of the temporal generalization matrix can be enabled in the MVPAlab configuration structure as follows:

```
cfg.classmodel.tempgen = true;
```

Feature contribution analysis

Usually, classification algorithms are treated as black boxes. However, highly useful information can be extracted out under specific circumstances. For example, the value of a feature weight, obtained after the training process of SVM models, is sometimes correctly interpreted as a measure of its contribution to the model decision boundary. In other words, it is a measure of its importance. As shown in Figure 37, the feature weight vector represents the coefficients of ω, which is an orthogonal vector to the separation hyperplane. However, as mentioned above, this is valid under certain scenarios (e.g., linear classifiers, use of the same scale for all features, no data transformations such PCA, etc.). Even meeting all these requirements, the interpretation of raw feature weights can lead to wrong conclusions regarding the origin of the neural signals of interest. A widespread misconception about features weights is that channels with large weights should be related to the experimental condition of interest, which is not always justified (Haufe et al., 2014). In fact, large weight amplitudes can be observed for channels not containing the signal of interest and *vice versa*. To solve this problem, Haufe et al. (Haufe et al., 2014) proposed a procedure to transform these feature weights so they can be interpreted as origin of neural processes in space, which leads to more accurate predictions in neuroscience studies.

This useful procedure is implemented in the MVPAlab Toolbox. During any decoding analysis, MVPAlab extracts and saves the raw weight vectors and its Haufe correction in a time-resolved way. Thus, the contribution (or importance) of each electrode to the classification performance can be evaluated at any given timepoint. Additionally, and only if channel location information is available, MVPAlab can create animated plots representing the evolution of the distribution of weights over a scalp template. This analysis can be computed at group level or only for a specific participant. Please, see the *Result* section for further details.

Feature contribution analysis is disabled by default but can be enabled in the configuration file as follows:

```
cfg.classmodel.wvector = true;
```

Frequency contribution analysis

The contribution of different frequency bands to the overall decoding performance can be assessed in MVPAlab through an exploratory sliding filter approach. To this end, the original EEG signal can be pre-filtered using a band stop sliding FIR filter. Therefore, different frequency bands can be filtered-out of the original EEG data, producing new filtered versions of the original dataset. The former time-resolved multivariate analysis is now computed for each filtered-out version of the data. The importance of each filtered-out band is quantified computing the difference maps in decoding performance between the filtered and the original decoding results. Accordingly, if the classification performance at any given point is higher for the original signal compared to the filtered-out version, then the removed frequency band contains relevant information used by the classification algorithms to discriminate between conditions. This procedure is illustrated in Figure 42.

By definition, this analysis can be computed in a time-resolved manner (without temporal generalization) and using only the mean accuracy or the AUC as performance metric.

Several parameters should be defined in the MVPAlab configuration structure to compute the sliding filter procedure:

```
cfg.sf.flag   = true;
cfg.sf.metric = ‘auc’;
cfg.sf.lfreq  = 0;
cfg.sf.hfreq  = 40;

cfg.sf.fspac  = 'log';
cfg.sf.nfreq  = 40;
```

Sliding filter analysis can be enabled or disabled setting the configuration variable (`.flag`) to true or false. The (`.lfreq`) and (`.hfreq`) variables define the frequency limits in which the analysis will be computed. As mentioned before, mean accuracy (`.metric = ‘acc`) or AUC (`.metric = ‘auc’`) can be selected as performance metrics for this analysis. The number of individual frequency bands that will be removed from the original dataset (frequency resolution) is defined by (`.nfreq`).

Each of these frequency bands can be linear (`.fspac = 'lin'`) or logarithmically (`.fspac = 'log'`) spaced as shown in Figure 43. On the one hand, if the frequency bands are linearly spaced, the frequency resolution is equally distributed across the entire spectrum. On the other hand, a higher frequency resolution is found in the low part of the spectrum if the frequency bands are logarithmically spaced. This is especially interesting for investigations focusing on the study of the lower part of the M/EEG spectrum (α, β and θ frequency bands).

The filter design parameters such as filter type (`.ftype`), filter bandwidth (`.bandwidth`), window type (`.wtype`), filter order (`.order`), and others, can also be tuned in the configuration file as follows:

```
cfg.sf.ftype = 'bandstop';
cfg.sf.wtype = 'blackman';
cfg.sf.bw    = 2;
cfg.sf.hbw   = cfg.sf.bw/2;
cfg.sf.order = 1408;
```

The filter design and filtering process employed by MVPAlab is based on the EEGlab built-in function `pop_firws`.

Digital filters usually affect brain signals and are commonly applied at many stages from the data acquisition to the final publication. Many undesired events including temporal blurring or signal delays may occur, which may lead to incorrect interpretation of the results. Therefore, an appropriate filter design becomes crucial to prevent (or mitigate) these signal distortions. Please see (de Cheveigné & Nelken, 2019; VanRullen, 2011) for a deeper understanding of how brain signals can be affected by filtering processes.

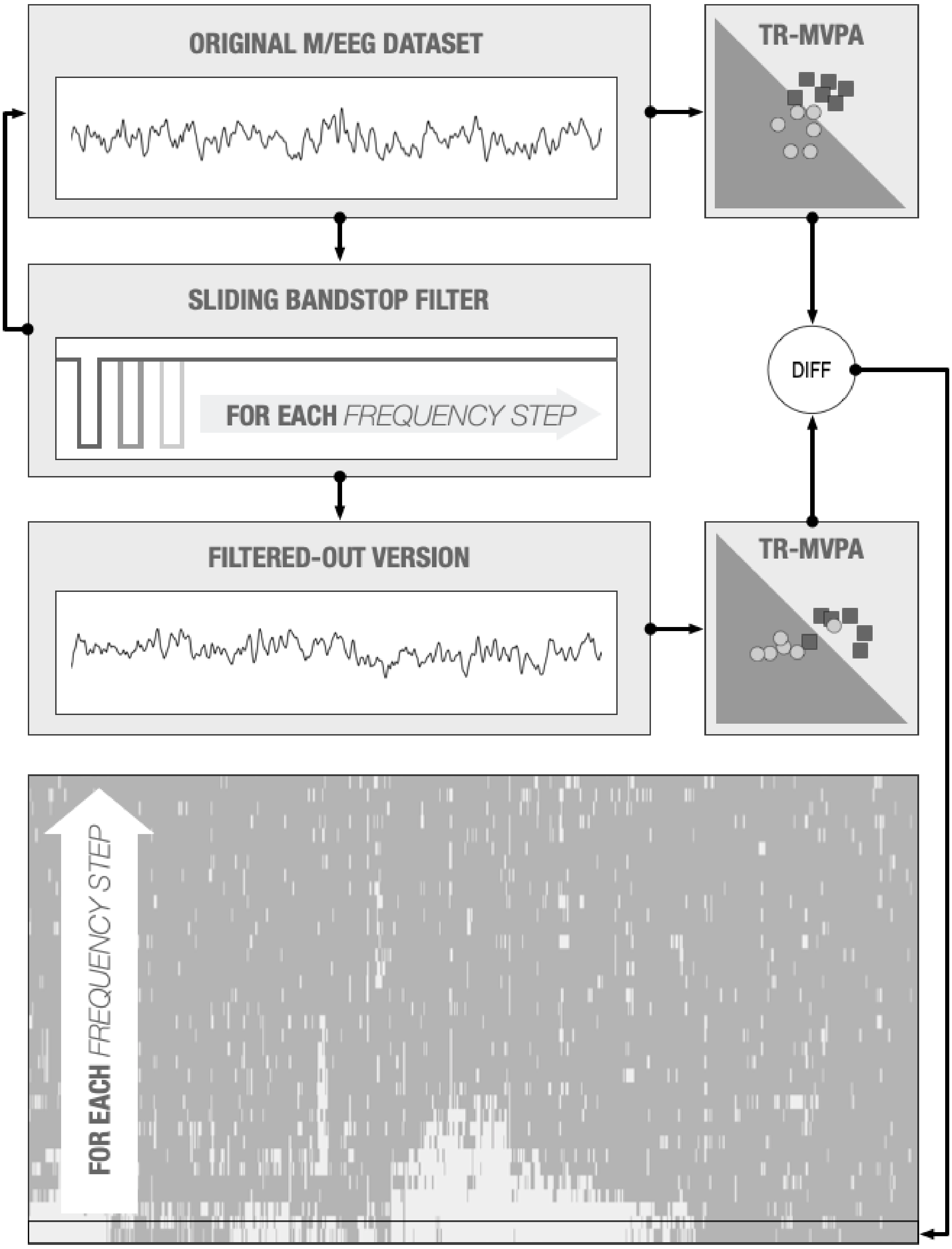


Figure 42. Diagram of the frequency contribution analysis

The classification model is trained with data at certain time point (black square). This model is then tested along the remaining time points (grey square), repeating this process for each time point inside the epoch.

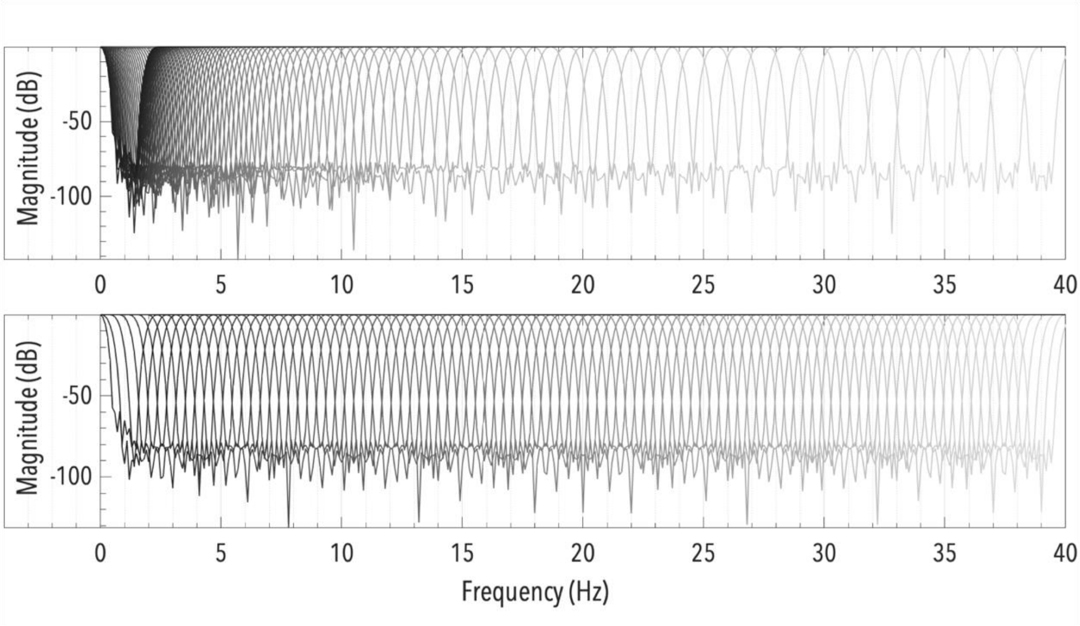


Figure 43. Magnitude responses of the sliding filter

Magnitude response for both linear and logarithmically spaced band-stop sliding filters. 60 frequency bands, 1408 filter order, Blackman window, 2Hz overlapped bandwidth.

The complete sliding filter analysis pipeline is coded in both **mvpalab_import(**cfg**)** and **mvpalab_sfilter(**cfg**)** functions:

```
cfg = mvpalab_import(cfg);

% Compute sliding filter analysis:
[cfg,diffMap,stats] = mvpalab_sfilter(cfg);
```

Due to the elevated RAM requirements of this analysis, the import function stores each filtered versions of the original dataset in a specific folder of your hard drive for each participant individually. The user should consider using an external hard drive for this high-demand analysis.

Then, as explained before, the function **mvpalab_sfilter()** computes and compares the decoding performance of different metrics between the original dataset and each filtered version, returning a difference map structure `diffMap`. The result matrices `[freqs x timepoints x subjects]` for specific performance metrics can be extracted using dot notation (e.g. `diffMap.auc`). Only the mean accuracy and the area under the curve are implemented for this analysis.

Additionally, if enabled, this function also implements the statistical permutation analysis, returning the `stats` variable, which includes the statistically significant clusters (Please see Section 5.2.5 *Cluster-based permutation testing* for a detailed explanation).

5.2.5. Cluster-based permutation testing

In order to draw statistical inferences at the group level, MVPAlab implements a non-parametric cluster-based permutation approach, as proposed by Stelzer (Stelzer et al., 2013) for fMRI studies. This method has been adapted to electroencephalography data and can be computed for different performance metrics: mean accuracy, area under de curve, F1 score, recall and precision.

Using a combined permutation and bootstrapping technique, the null distribution of the empirical decoding accuracy is obtained. By default, at the single-subject level, 100 randomly permuted accuracy maps are generated. Then, one of the previously calculated accuracy maps for each participant is randomly drawn. This selection is group-averaged and the procedure is repeated 10^5 times, generating 10^5 permuted group accuracy maps. Next, for each timepoint, the chance distribution of accuracy values is estimated. The above and below chance thresholds are determined (99.9th percentile of the right and left-tailed area of the distribution), which correspond to a very low probability of obtaining significant results by chance (Figure 44). Then, clusters of time-points exceeding the previously calculated threshold in all the 10^5 permuted accuracy maps are collected, generating the normalized null distribution of cluster sizes. Finally, a correction for multiple comparisons (False Discovery Rate (FDR)) is applied at a cluster level to obtain the smallest cluster size to be considered significant.

The default parameters for this analysis can be modified in the MVPAlab configuration file as follows:

```
cfg.stats.nper   = 100;
cfg.stats.nperg  = 1e5;
cfg.stats.pgroup = 99.9;
cfg.stats.pclust = 99.9;

cfg.stats.shownulldis = true;
```

Two different functions coded the beforementioned pipeline:

```
...

% Compute MVCC analysis:
[result,cfg] = mvpalab_mvpa(cfg,fv);

% Compute permutation maps:
[permaps,cfg] = mvpalab_permaps(cfg,fv);

% Run statistical analysis:
stats = mvpalab_permtest(cfg,result,permaps);

…
```

First, the function `mvpalab_permaps()` computes the required permuted accuracy maps for each subject, randomly shuffling the original class labels. Then, `mvpalab_permtest()` generates the null distributions, determines the significance thresholds, collects significant clusters, computes cluster size distributions and corrects for multiple comparisons (FDR) to obtain the smallest cluster size to be considered significant. The variable `stat` is returned containing, among others, below and above chance significant clusters:

```
stats.clusters.sig   % Above chance clusters
stats.clusters_.sig  % Below chance clusters
```

Above and below chance clusters are extracted using the MATLAB built-in function `bwconncomp` included in the Image Processing Toolbox.

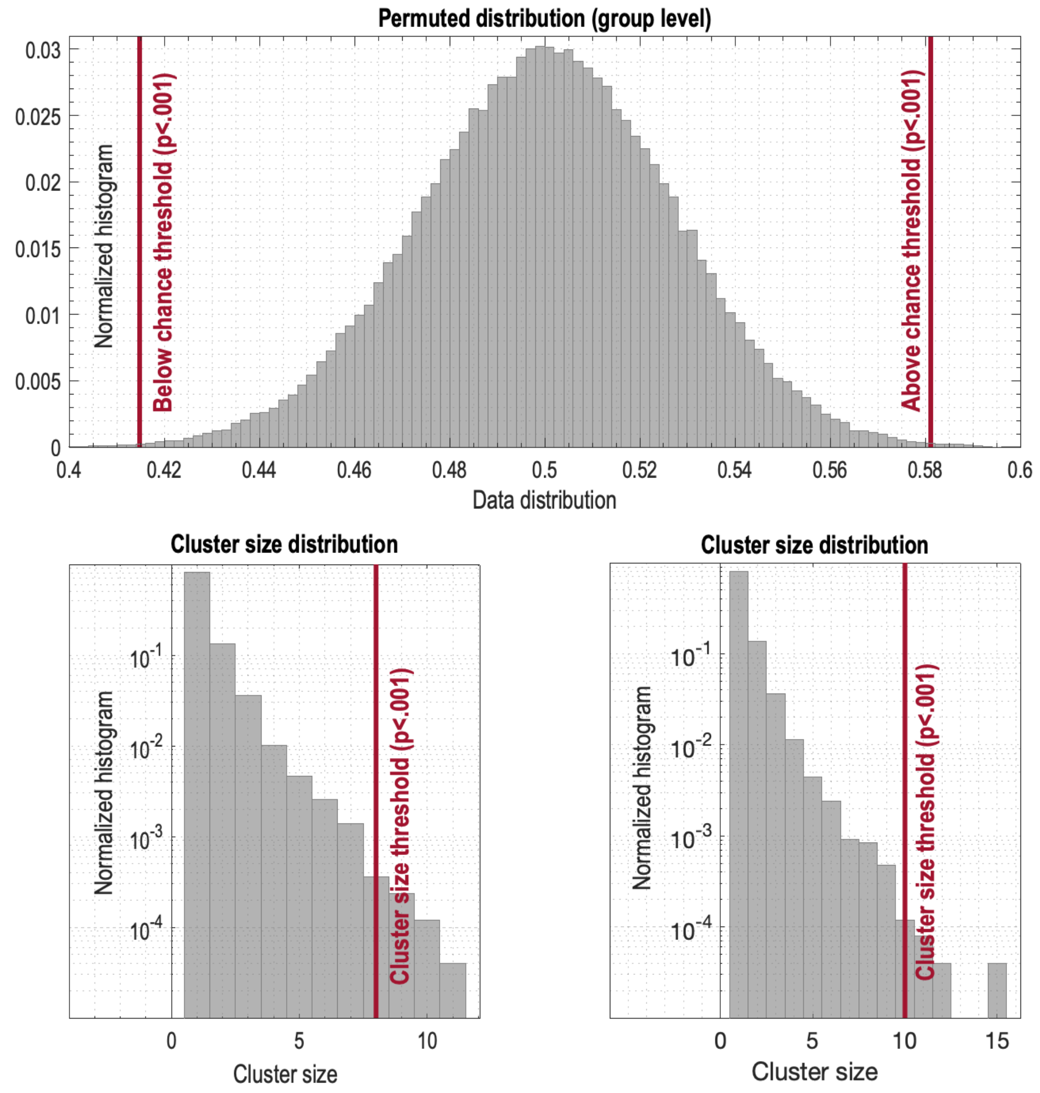


Figure 44. Accuracy and cluster size null distributions

The vertical line represents the threshold corresponding to a very low probability to obtain significant results by chance. These thresholds correspond to a p-value below 0.001 for both distributions.

5.2.6. Result representation pipeline

In addition to the graphic user interface, MVPAlab implements different high-level functions to generate highly customizable graphical representation of the results. Once the decoding analysis is completed and the results files are saved, the graphical representation pipeline runs as follows:

```
graph = mvpalab_plotinit();
```

First, the function `mvpalab_plotinit()` generates and returns a default configuration structure (`graph`) containing all the required configuration parameters. Then, the specific result file to be plotted should be loaded:

```
load results/time_resolved/acc/result.mat
```

Finally, the high-level plotting function returns the graphical representation of the selected result file:

```
mvpalab_plotdecoding(graph,cfg,result,stats);
```

The variable `stats` is optional and contains, among others, the statistically significant clusters. If this variable is not omitted, significant results will be highlighted in the resulting figure.

Several plotting functions are available for different types of analysis:

```
mvpalab_plotdecoding(graph,cfg,result,stats);
mvpalab_plottempogen(graph,cfg,result,stats);
mvpalab_plotfreqcont(graph,cfg,result,stats);
mvpalab_plotfeatcont(graph,cfg,wvector,result);
```

The `mvpalab_plotdecoding()` function generates time-resolved performance plots, `mvpalab_plottempogen()` is used for the graphical representation of temporal generalization matrices, `mvpalab_plotslidfilt()` function generates the graphical representation for the sliding filter analysis and `mvpalab_plotfeatcont()` can generate topological representations and temporal animations of features contribution to the decoding performance.

To get the best of the MVPAlab Toolbox plotting capabilities the use of the graphic user interface is highly recommended. This is a fast, flexible and very intuitive manner to design high-quality plots. Even so, the same results can be obtained by hand coding several configuration parameters included in the graph configuration structure. A complete selection of the most useful configuration parameters and a short explanation is listed below:

```
% Time-resolved decoding analysis:
% --------------------------------------------------
graph.plotmean   = true; % Plot group average
graph.smoothdata = 5;    % Window size for smooth
graph.stdsem     = true; % Plot STD or SEM
graph.linestyle  = '-';  % Line style
graph.linewidth  = 1;    % Line width
```

```
% 2D decoding analysis (TGM or SFILTER):
% --------------------------------------------------
graph.clusterLineColor = [0 0 0]; % Cluster color.
graph.clusterLineWidth = 1;       % Cluster width.
graph.caxis = [.4 .9];            % Color range.
```

```
% Feature contribution analysis:
% --------------------------------------------------
graph.weights.type  = 'raw'; % Raw or corrected
graph.weights.anim  = true;  % Animated/static plot
graph.weights.speed = 0.1;   % Animation speed
graph.weights.start = 400;   % Start time (ms)
graph.weights.end   = 450;   % End time (ms)
graph.weights.sub   = 1;     % Individual subject
```

```
% Highlight significant result:
% --------------------------------------------------
graph.sigmode.points = true; % Points/shade plot
graph.stats.above    = true; % Above chance clusters
graph.stats.below    = true; % Below chance clusters
graph.sigh           = 0.4;  % Sig. points height
```

```
% Font, titles, labels and axes limits:
% --------------------------------------------------
graph.fontsize = 14;
graph.title    = 'MVPAlab - default figure';
graph.ylabel   = 'Classifier performance';
graph.xlabel   = 'Time (ms)';
graph.xlim     = [-200 1500];
graph.ylim     = [0 1];
```

```
% Individual subject plots:
% --------------------------------------------------
graph.subject = 3; % Subject idx (individual plot)
```

5.3. Results

During this section we present the results obtained after testing all the MVPAlab main functionalities with the sample EEG dataset presented in Section 5.2 *Materials and Methods*. As mentioned, we compiled this sample dataset for illustration purposes, including the EEG data of two main conditions (or classes) and four subconditions of three different participants. Readers interested on the results obtained for the entire sample should refer to the original publication (López-García et al., 2020).

Time-resolved decoding analysis. Figure 45a depicts the result of a time-resolved decoding analysis comparing the classification performance of two models, linear support vector machine and linear discriminant analysis. Shaded areas represent the Standard Error of the Mean (SEM) of the averaged performance across participants. Additionally, single-subject plots are depicted in dashed and dotted lines. Statistically significant areas for each classification model are highlighted using horizontal color bars. As shown, SVM outperforms LDA by obtaining higher performance and a wider significant window.

To compute this MVPA analysis, classification models were trained using smoothed (5 timepoint moving average) and normalized supertrials (8 trials randomly averaged). No PCA was computed, so raw voltage values were extracted from the 64 electrodes as features in balanced datasets.

Dimensionality reduction. Figure 45b shows the time-resolved classification performance (f1-score) averaged across participants of an SVM classifier, using different number of PCA components as features. As shown, the f1-score increases with the number of features. Significant results were obtained employing just the first PCA component. When only the first nine PCA components were employed as features, the classification model showed comparable performance results to those obtained when no PCA is computed, as depicted in Figure 45a. Computation time is in fact reduced when the dimension of the feature space is smaller, however, when PCA transformation is computed, the original spatial information is lost.

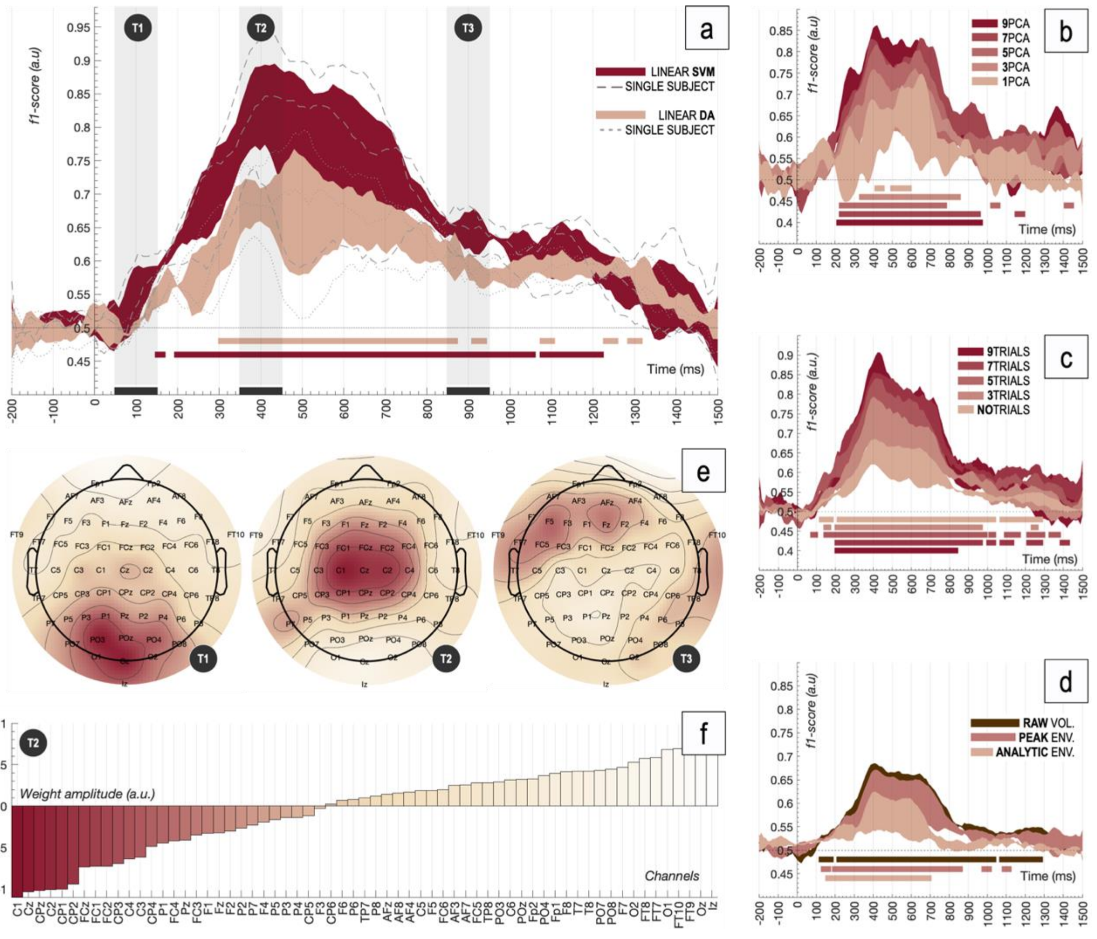


Figure 45. Time-resolved MVPA results

(a) Decoding performance (f1-score) for different classification models at a group-level: support vector machine vs. linear discriminant analysis. Single subject plots are represented in dashed and dotted lines. Significant clusters are highlighted using horizontal colored bars. Shaded areas represent the standard error of the mean. **(b)** Group-level decoding performance for different number of features when PCA is applied. **(c)** Group-level decoding performance as a function of the selected number of trials to average. **(d)** Group-level decoding performance when different power envelopes are extracted and employed as features instead of the raw voltage. **(e)** Group-level weight distribution (corrected) for three different time windows: T1: 50-150ms, T2: 350-450ms and T3: 850-950ms. **(f)** Weights' amplitude for each channel sorted by importance.

Supertrial generation. Figure 45c depicts the classification performance when the input dataset was reduced by randomly averaging different numbers of trials belonging to the same condition. This trial averaging process generates supertrials with an increased signal-to-noise ratio. As shown, the SVM model performance increases with the number of trials averaged, however, the variability of the data (the standard error of the mean) also does due to the reduced input dataset. Thus, this figure shows wider significant windows when no or few trials are averaged.

Power envelope as feature. The comparison between the performance of classifiers using different EEG signal characteristics as features is showed in Figure 45d. First, the peak and analytic upper envelopes of the EEG signal were calculated (5 timepoints window). Then, feature vectors were extracted from these power signals. Significantly lower performance rates were obtained for the analytic power envelope. Although the main goal of this article is not to address this type of questions, there seems to be a plausible cause favoring this outcome: the phase of the EEG signa may contain critical information to discriminate between the two experimental conditions employed. This is due to the fact that the instantaneous phase information contained within the original EEG signal is discarded during the analytic power envelope computation (see *Appendix B* for further details). This approximation is employed in recent literature (Vidaurre et al., 2020)(O'Neill et al., 2015) to remove instantaneous phase from certain brain oscillations and to study how this phase information contributes to decoding performance.

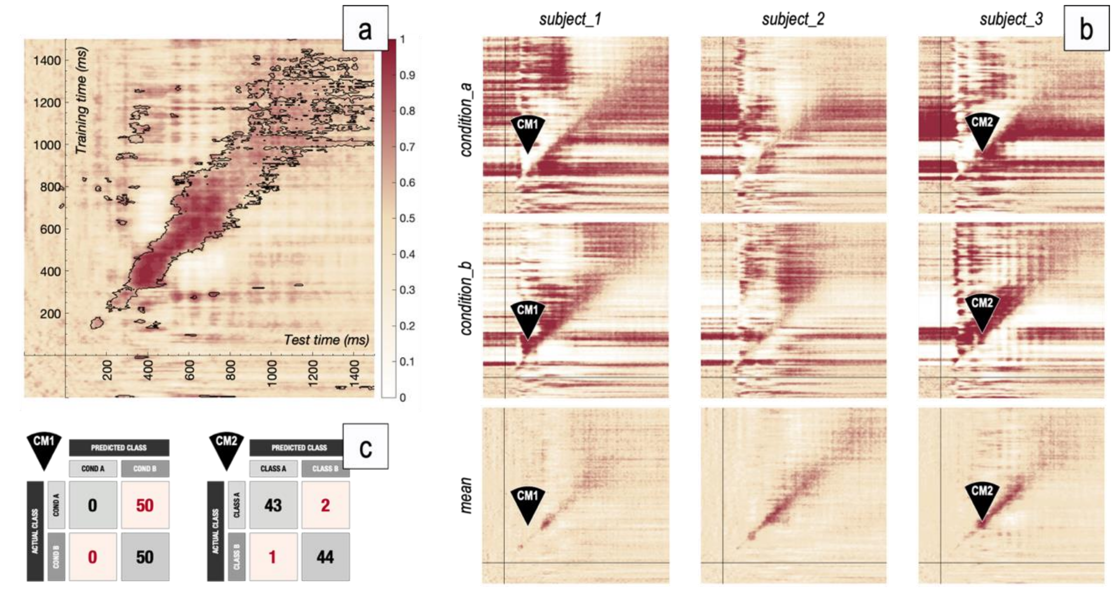


Figure 46. Temporal generalization results

(a) Group-level temporal generalization matrix (area under the ROC curve) for an SVM classifier when 8 trials were averaged to generate the input dataset. Above-chance significant clusters are highlighted using black lines. **(b)** Single subject generalization patterns (sensitivity), individually calculated for each condition. **(c)** Confusion matrices CM1 and CM2 for two different timepoints marked in (b).

Feature contribution analysis. During the training process of the previous linear SVM model, the feature weights were calculated for each timepoint and subject and corrected according to Haufe's method (Haufe et al., 2014). In order to show the activity distribution contributing to decoding accuracy, the feature weights were averaged across participants and three different temporal windows. First, when the slope of the decoding curve becomes positive, between 50-150ms. Then, between 350-450ms, when decoding performance peaks, and finally between 850-950ms, at the end of the significant window for LDA. A corrected version of the training weights distribution for these three different time windows is depicted in Figure 45e. Finally, Figure 45f shows the weight amplitude of each channel sorted by its importance, averaged across participants during the 350-450ms temporal window.

*Temporal generalization analysis***.** Figure 46a shows the temporal generalization matrix of the first MVPA analysis, Figure 45a, representing the performance value (AUC) for each combination of training-test time points. Above-chance significant clusters are highlighted using black lines. This approach is an extension of time-resolved decoding, which is an indication of how EEG patterns vary or persist in time. Different performance metrics, such as the area under the curve or the mean accuracy are usually reported, generating temporal generalization patterns that resembles those shown in Figure 46a. This way, above-chance performance clusters outside the diagonal of the matrix are interpreted as a sign of temporal stability of certain activity patterns along time.

However, in-depth examinations revealed interesting behaviors of classification models, providing extra information about how individual conditions are classified, especially in those areas in which no temporal generalization occurs. Figure 46b depicts the sensitivity (recall) of the classification model for each condition and subject. Complementary generalization patterns are observed for individual conditions, revealing extreme sensitivity values especially when no temporal generalization occurs. Some examples are presented and analyzed using the corresponding confusion matrices. As seen in Figure 46 (c), the confusion matrix **CM1** indicates that, for this specific temporal point, no test samples belonging to *condition_a* were correctly predicted as *condition_a*, leading to a sensitivity value of 0 for this condition. By contrast, all samples belonging to *condition_b* were correctly labelled (in addition to all samples belonging to *condition_a* incorrectly predicted as *condition_b)* which leads to a sensitivity value of 1. This behavior is frequent across subjects and timepoints, reflecting the inability of the classifier to correctly predict information in several areas, which is a clear sign of the absence of temporal persistence of patterns.

Multivariate Cross-Classification analysis. Figure 47a depicts the result of a time-resolved multivariate cross-classification analysis. The classification model was trained with *condition_1* vs. *condition_2* and *condition_3* vs. *condition_4* were used for testing. This process was repeated inversely, generating two different decoding performance curves corresponding to both classification directions (train: A, test: B and *vice versa*). Additionally, single subject curves were added to the figure for each classification direction. As shown, windows of significant differences are obtained between 200-800ms, indicating that this technique successfully shows the consistency of patterns across different sets of data.

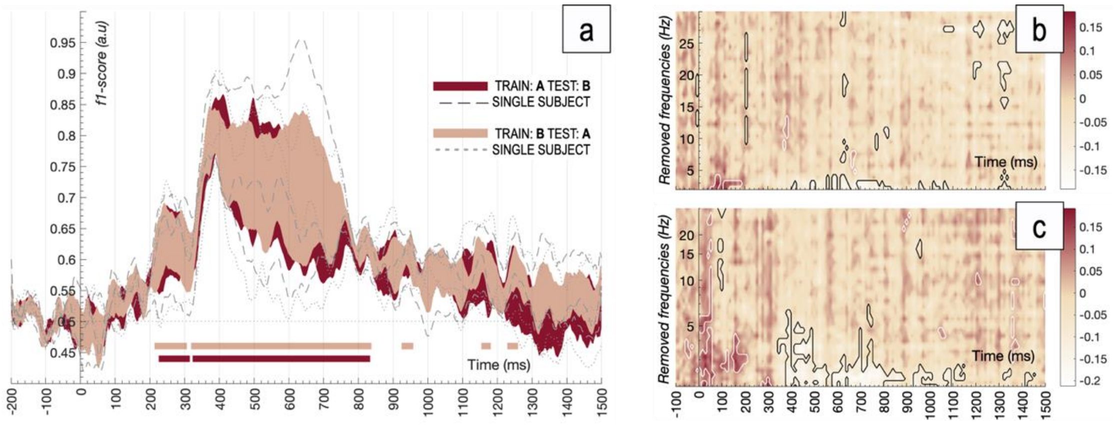


Figure 47. Time-resolved MVCC and frequency contribution analysis results

(a) Group-level decoding performance (F1-score) for both cross-classification directions. Single subject plots are represented in dashed and dotted lines. Significant clusters are highlighted using horizontal colored bars. Shaded areas represent the standard error of the mean. (b-c) Decoding performance maps when different frequency bands are removed from the original datasets in a linear and logarithmically spaced steps.

Frequency contribution analysis. A sliding band-stop filter approach was followed to study the contribution of each frequency band to the overall decoding accuracy. The band-stop FIR filter was designed using the EEGLAB `pop_firws` function (2Hz bandwidth, 0.2Hz transition band, 2048 filter order, Blackman window). The original EEG dataset was pre-filtered (32 overlapped frequency bands, between 0–30Hz in linear and logarithmically-spaced steps) producing 32 new filtered versions of the original signals. The former time-resolved decoding analysis (*condition_a* vs. *condition_b*) was conducted for each filtered version and the importance of each filtered-out band was quantified computing the difference maps in decoding performance between the filtered and the original decoding results. Figure 47b and c shows the results of the sliding filter analysis for linear and logarithmically-spaced steps respectively. As shown, decoding accuracy significantly dropped when frequencies up to 6Hz were filtered-out, suggesting that the studied phenomenon relies on processes operating in the Delta and Theta frequency bands. Significant clusters were calculated applying the proposed cluster-based permutation test to filtered-out datasets, generating accuracy null distributions for each time-frequency point.

5.4. Discussion

Despite the MVPAlab Toolbox is freely available, an important limitation is that it needs the MATLAB core to be executed, which is a proprietary and expensive software. We are aware of the recent growth of free software alternatives, such Python, in academic environments. Nevertheless, we built this software under MATLAB due several reasons, including the huge amount of available and well-documented functionalities for this platform, their active user community and its wide implementation in neuroscience labs. Even so, there are excellent open-source alternatives for those users with no access to a MATLAB license.

Additionally, the MVPAlab Toolbox is not yet compatible with BIDS-EEG (Pernet et al., 2018) format, which is a recently developed project for electroencephalography studies, extending the original Brain Imaging Data Structure (Gorgolewski et al., 2016) (BIDS). Both projects are an excellent effort to standardize the way data is stored, increasing accessibility, usability and reproducibility of neuroimaging data. We favor these principles and we are planning to integrate BIDS-EEG format in the MVPAlab Toolbox in future releases.

Classification algorithms are the cornerstone of multivariate decoding analyses. However, these powerful techniques suffer from hyperparameter overfitting, which usually leads to invalid result. A recent study refers to this phenomenon as "*overhyping*"(Hosseini et al., 2020) and proposes several strategies to avoid this problem.

Regular cross-validation approaches are commonly employed to mitigate spurious result in classification accuracies, but it has been proved that, in some cases, they are not sufficient (Hosseini et al., 2020). Several strategies, such as pre-registration, nested cross-validation (Cawley & Talbot, 2010), lock box and blind analyses are presented as reliable alternatives to prevent or mitigate *overhyping*. Unfortunately, the MVPAlab toolbox does not currently implement those strategies, but we are further investigating these issues for future releases. Additionally, recent studies (Gorriz et al., 2021; Górriz, Ramirez, et al., 2019) proposes the Statistical Agnostic Mapping (SAM) as an interesting alternative to the cross-validation procedures. Particularly in neuroscience, these approaches usually leads to small sample sizes and high levels of heterogeneity when conditions are split into each fold, causing among other things, a large classification variability (Varoquaux, 2018). To address these problems, SAM considered the use of the resubstitution error estimate as a measure of decoding performance. The difference between the actual error and the resubstitution error (which is a very optimistic measure) is upper-bounded by a novel analytic expression proposed in the original article. See (Gorriz et al., 2021; Górriz, Ramirez, et al., 2019) for further details. Future releases of the MVPAlab Toolbox are planned to include this novel classification paradigm, which at the moment is under development.

Furthermore, dimensionality reduction is a crucial step in neuroimaging studies to select the most relevant predictor variables, reducing the experimental noise and mitigating the *small-n-large-p* problem. These techniques prevent the classification model from overfitting, leading to a better predictions and increasing its generalization capability (Mwangi et al., 2014). Although MVPAlab implements Principal Component Analysis, which is one of the most popular dimensionality reduction approaches in neuroscience studies, there are different algorithms which have not been implemented yet. The integration with some of these feature reduction approaches, such as Partial Least Square (PLS)(Krishnan et al., 2011), is currently under development.

Regarding to the classification stage, the MVPAlab Toolbox implements probably two of the most commonly employed classification algorithms in neuroscience literature: Support Vector Machines and Discriminant Analysis, in their linear and non-linear versions. However, this configuration may not be enough in certain situations. In fact, different software alternatives include many other classification models, such as Logistic Regressions, Naïve Bayes or ensembles methods. As mentioned, the MVPAlab Toolbox is in constant development, these functionalities are planned to be implemented in near future.

The MVPAlab Toolbox was initially developed for M/EEG analysis. Due to its nature, M/EEG signals provide exceptional temporal resolution, but lack spatial resolution. Contrary, other non-invasive techniques, such as fMRI, can identify brain activity changes at millimetric levels but suffer from poor temporal resolution. To overcome this dichotomy, recent trends in the neuroimaging field opt for the multimodal data fusion(S. Wang et al., 2021; Y. D. Zhang et al., 2020), which is a step forward towards a better understanding of brain function. These fusion approaches combine data from different neuroimaging techniques (M/EEG-fMRI), preserving their strengths while overcoming their weaknesses (Radoslaw M. Cichy & Oliva, 2020). Extending the MVPAlab functionality from multivariate M/EEG analyses to multimodal data fusion represents one of the most important lines of development on the MVPAlab roadmap.

There are a myriad of new analyses and techniques that can be employed to analyze data of different nature in neuroscience, which is a clear indicator of the fast growth of the field. As mentioned, MVPAlab Toolbox was initially designed to work with epoched M/EEG data, extracting the raw potential of the signal and computing time-resolved classification analyses. The latest release supports different signal characteristics as features, such as the power envelope or the instantaneous phase of the signal. Recent studies (Syrjälä et al., 2021) implement different feature engineering techniques, concatenating data from different frequency bands, to improve the classification result. Currently, MVPAlab does not implement these strategies. However, MVPAlab can be used as a general-purpose classification tool. Users can adapt and import their own datasets, regardless of its nature (source space data, connectivity data or not even M/EEG related signals), and easily perform time-resolved classification analyses.

5.5. Conclusions

MVPAlab is a very flexible, powerful and easy-to-use decoding toolbox for multi-dimensional electroencephalography data, including an intuitive Graphic User Interface for creation, configuration, and execution of different decoding analysis. Not a single line of code is needed. For those users with more coding experience, MVPAlab implements high and low-level routines to design custom projects in a highly flexible manner.

Different preprocessing routines, classification models and several decoding and cross-decoding analyses can be easily configured and executed. MVPAlab also implements exclusive analyses and functionalities, such as parallel computation, significantly reducing the execution time, or frequency contribution analyses, which studies how relevant information is coded across different frequency bands. MVPAlab also includes a flexible data representation utility, which generates ready-to-publish data representations and temporal animations. All of this combined makes MVPAlab Toolbox a compelling option for a wide range of users.

5.6. Supplementary material

Supplementary material such as temporal animations of feature contributions are available to download from an Open Science Framework project:

```
https://osf.io/qrfgk/
```

The Supplementary Material folder includes different video files (.mov) recording the temporal distribution of channels contributing to the decoding accuracy. Raw and corrected feature weights animations for individual participants and group-averaged are included.

5.7. Appendix

5.7.1. Software information

An up-to-date version of the toolbox is freely available in the following GitHub repository:

```
https://github.com/dlopezg/mvpalab
```

We use semantic versioning (e.g. X.Y.Z) to denote different releases, the most recent being the v1.0.0 version, which is our first public release including a stable version of the toolbox. The software documentation can also be found in our GitHub repository:

```
https://github.com/dlopezg/mvpalab/wiki
```

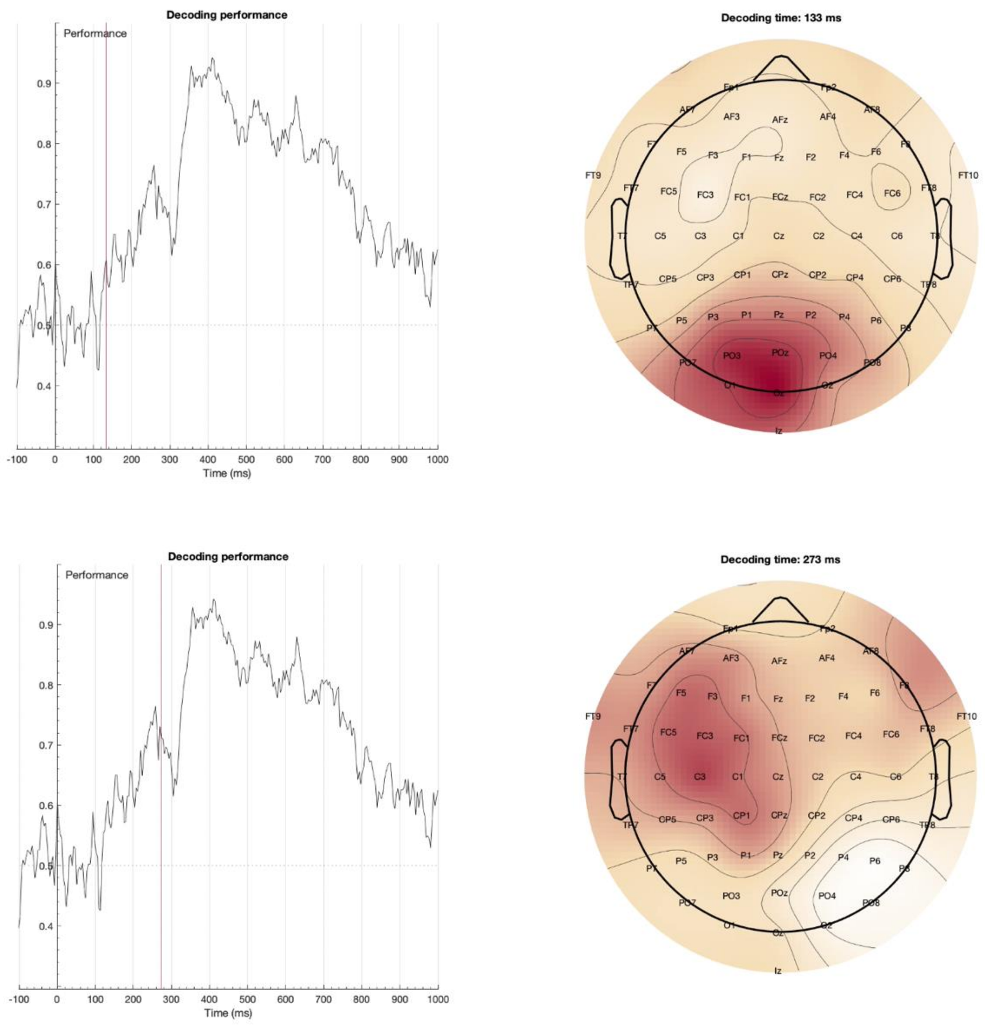


Figure 48. Supplementary materials

This figure shows different frames of a video animation generated with MVPAlab. These frames depict the weights distribution over the scalp in two different timepoints. As mentioned, these animations can be downloaded from the following link: https://osf.io/qrfgk/

MVPAlab toolbox is released under a GNU General Public License (GPL) v.3.0, which allows users to freely use, change and share this software. For further license details please see:

```
https://gnu.org/licenses/quick-guide-gplv3
```

We encourage all users to collaborate in MVPAlab Toolbox development by submitting their own contributions and improvements via *pull request*. To suggest new features, bug report or any other related issues, please use the MVPAlab issue tracker available in GitHub in the following link:

```
https://github.com/dlopezg/mvpalab/issues
```

The sample EEG dataset used in this article is hosted in the Open Science Framework project:

```
https://osf.io/du6fa
```

5.7.2. Benchmarks and processing time

The performance comparison between different implementations of several classification libraries is out of the scope of this article. However, processing time for different analysis have been measured in Windows and macOS and are reported in the following table:

Table 5. Processing time in seconds for different task and platforms.

Time (s)	Windows 10 (64 bits)		MacOS 11.3 (64 bits)	
	Single	Parallel	Single	Parallel
T1: *TR-SVM*	15.58	**4.03**	15.18	**5.27**
T1: *TR-LDA*	8.63	**1.95**	10.24	**3.04**
T1: *TG-SVM*	120.88	**21.70**	102.70	**26.42**
T1: *TG-LDA*	302.72	**58.79**	279.34	**92.37**
T2: *TR-SVM*	10.73	**2.28**	10.30	**4.04**
T2: *TR-LDA*	3.80	**1.03**	4.08	**1.43**
T2: *TG-SVM*	53.24	**11.48**	49.98	**16.43**
T2: *TG-LDA*	155.69	**25.77**	127.61	**38.49**

Task 1 (T1) consist of a single subject time-resolved decoding analysis and a five-fold cross validation stage, when only the mean accuracy was calculated, ten trial averaging and no dimensionality reduction was computed. In this scenario, different classification algorithms (SVM and LDA) were trained and validated for 256x256 timepoints using 80 observations (trials) and 63 features (electrodes).

Task 2 (T2) consist of a single subject time-resolved cross-decoding analysis, when only the mean accuracy was calculated, five trial averaging and no dimensionality reduction was computed. Both classification directions were calculated. In this scenario, different classification algorithms (SVM and LDA) were trained and validated for 256x256 timepoints using 80 observations (trials) and 63 features (electrodes).

These tests were computed in two different setups. First, in a 6-Core workstation (Intel Core i7-5820K CPU @ 3.30GHz, 32GB RAM DDR4 @ 2400MHz) running Windows 10 (64 bits) and MATLAB 2020a (9.8.0.1323502) and finally in a cuad-core MacBook Pro (Intel Core i7-6820HQ CPU at 2,7GHz, 16GB RAM LPDDR3 @ 2133MHz) running macOS Big Sur (64 bits, version 11.3) and MATLAB 2020a (9.8.0.1323502).

5.7.3. Power envelope and instantaneous phase calculation

Different signal characteristics, such the instantaneous amplitude or phase, can be easily calculated and extracted in the complex plane. In order to extract this information from a real-valued signal *x(t)* (e.g. the electroencephalogram), the following transformation can be applied:

$$z(t) = x(t) + \mathrm{j}\mathbf{HT}[x(t)]$$

Equation 31

Here, *z(t)* is the complex form of *x(t)*, also known as the '*analytic signal*', and **HT** denotes the Hilbert's Transformation of the real-valued signal, defined as:

$$\mathbf{HT}[x(t)] = P.V\left[\frac{1}{\pi}\int_{-\infty}^{+\infty}\frac{x(\tau)}{t-\tau}d\tau\right]$$

Equation 32

where P.V denote the *Cauchy Principal Value* of the integral, which is required for assigning values to improper integrals values that would otherwise be undefined (the singularity occurs when t = τ). Thus, the instantaneous amplitude, also known as power envelope $e(t)$, or the instantaneous phase $\phi(t)$, can be easily extracted from the analytic signal as follows:

$$e(t) = |z(t)| = \sqrt{x^2(t) + (\mathbf{HT}[x(t)])^2}$$

$$\phi(t) = \angle z(t) = \arctan\frac{\mathbf{HT}[x(t)]}{x(t)}$$

Equation 33

Chapter 6. Multimodal fusion methods in MVPAlab

6.1. Introduction

The vast progress of science and technology occurred in the past few decades has witnessed the use of Machine Learning-based techniques in a wide range of scientific disciplines, including neuroscience (Górriz et al., 2020). Nowadays, the study of brain functioning employing magneto/electroencephalography (M/EEG) signals relies mostly on multivariate techniques, leaving behind classic univariate approaches such as Event-Related Potentials (ERPs). Multivariate approaches outperform univariate ones in terms of sensitivity, detecting subtle changes in activations associated with specific information content in brain patterns. As a result, several Multivariate Pattern Analysis (MVPA) toolboxes, specifically designed for M/EEG signals, have recently emerged (e.g. The Amsterdam Decoding and Modeling Toolbox (Fahrenfort et al., 2018), MVPA-light (Treder, 2020), The Decision Decoding Toolbox (Bode et al., 2019) , or The MVPAlab Toolbox (López-García et al., 2022) , among others (Gramfort et al., 2013; Hanke, Halchenko, Sederberg, Olivetti, et al., 2009; Oostenveld et al., 2011)).

However, due to the nature of the signal of current non-invasive neuroimaging techniques, these tools help neuroscientists to characterize cognitive processes either in a time or space-resolved way. While M/EEG signals provide exceptional temporal resolution but lack spatial resolution, functional Magnetic Resonance Imaging (fMRI), localizes brain activity changes at millimetric levels but with poor temporal resolution. Overcoming this dichotomy is one of the major current challenges in the field, with some trends betting on multimodal data fusion as a promising solution (Radoslaw M. Cichy & Oliva, 2020). Fusing methods combine information of non-concurrent recordings from different neuroimaging modalities, preserving their individual strengths while overcoming their weaknesses.

Nevertheless, relating activity patterns between different modalities of brain-activity measurement is not a straightforward exercise, since those techniques measure signals of different nature (e.g., hemodynamic response vs. electrical activity), that have no direct correspondence. Therefore, a methodology that abstracts from the activity patterns *per se* is required to solve this correspondence problem. The main goal of this chapter is to implement Representational Similarity Analysis (RSA) (Kriegeskorte et al., 2008) on electroencephalography and fMRI signals, a methodological approach that abstracts from the signal space to a common similarity space, generating commensurable dissimilarity matrices for different experimental conditions and thus allowing the fusion of different signals. These analyses have been fully integrated in MVPAlab, an easy-to-use decoding toolbox for multidimensional M/EEG data. The inclusion of these analyses increases the MVPAlab Toolbox versatility and lays the foundation for the multimodal fusion methods, which represents one of the most important lines of development on the MVPAlab roadmap.

6.2. Materials and Methods

First, high-density electroencephalography signals were recorded from forty-eight participants to study differences in preparation between perceptual expectation and selective attention (Peñalver et al., 2022). To do so, we designed the Attention-Expectation task, which is described in the following section. Then, we adapted the task to fMRI modality and collected brain data from another forty-eight participants. It is noteworthy that EEG and fMRI data was not collected simultaneously but in separate sessions and from different participants.

To evaluate the performance of RSA and fusion analyses we compiled a sample dataset including five different EEG and fMRI data files corresponding to five participants randomly selected from the original sample. This demo dataset and the implemented code for the RSA and fusion analyses are publicly available in the GitHub repository of the MVPAlab Toolbox. As shown in Table 6, eight different experimental conditions were extracted for each participant. The experimental details of this study are specified in the following paragraphs.

Table 6. Extracted conditions from the attention-expectation task

Target-locked trials were grouped into eight different experimental conditions based on their category (name vs. face), block type (attention vs. expectation) and validity (valid vs. invalid).

Valid trials		Invalid trials	
Attention	Expectation	Attention	Expectation
val_name_att	val_name_exp	inv_name_att	inv_name_exp
val_face_att	val_face_exp	inv_face_att	inv_face_exp

6.2.1. Experimental design: Attention-Expectation task

Electroencephalography task

The main task (Peñalver et al., 2022) consisted on a cue-target sequence as shown in Figure 49, where participants were required to discriminate the sex (male or female) of the upcoming target stimuli (a *face* or a *name*). Each participant repeated this sequence 640 times in total, divided into 32 blocks of 20 trials each. Two different types of blocks were defined: *attention* and *expectation*. For each participant, and to avoid perceptual confounds, two cue shapes (counterbalanced across participants) were associated with faces and two with names. At the beginning of each block, the participants were informed about the block type (*Attention* or *Expectation*), the target sex/gender (e.g., "*Is the target male?*"), and the two cues (one for faces and one for names).

Each trial started with the presentation of this visual cue. The purpose of this task was to indicate the sex/gender of the target (male or female). Participants pressed a key ("a", "l", counterbalanced across participants) to respond whether the target belonged to the gender stated at the beginning of each block. Participants were verbally instructed to use the cues in the two blocks to respond as fast as possible while avoiding mistakes.

Importantly, and since attention and expectation are involved in almost any act of visual perception, we aimed at manipulating one process while keeping the other constant.

In attention blocks, the shape of the cue indicated the category of the upcoming stimulus to respond to (50% of *validity*). Only if the stimulus belonged to the relevant category (50% trials), the participant had to perform the gender discrimination task on the target. Otherwise, participants had to answer ‘no’ regardless of the stimulus sex/gender (50% trials). Note that this manipulation of relevance, where further processing has to be applied only to selected stimuli, is similar to that employed in previous literature (Baldauf & Desimone, 2014; Saenz et al., 2002; Summerfield et al., 2006; Womelsdorf et al., 2006). Importantly, both relevant and non-relevant targets were matched in expectation, as by design they appeared with 50% probability after each attention cue.

In expectation blocks, the shape of the cue predicted the category of the upcoming stimulus (75% of *validity*). Thus, after a cue associated with faces, a face appeared with 75% of probability. Here, participants had to perform the gender discrimination task in all trials. This way, both the expected and unexpected targets were equally relevant.

As shown in Figure 49 in every trial of the main task, the sequence of events was as follows: a 50 ms cue was followed by a fixed Cue-Target Interval (CTI) of 1500 ms and then the target appeared for 100ms. Trials were separated by 2500ms intervals. The attention and expectation blocks appeared in a fully counterbalanced order. Cues and target stimuli were also fully counterbalanced across participants. In total, the whole experimental session lasted approximately 80 minutes, with additional practice and EEG preparation time.

Functional Magnetic Resonance task

The Attention-Expectation task originally designed for EEG environments was adapted to meet fMRI requirements *(pending publication)*. Therefore, the fMRI experiment was divided into 8 runs (4 attention and 4 expectation runs) with 48 trials each, adding a total of 384 trials. The cue and the stimuli set remained unaltered (four geometrical shapes: circle, square, raindrop and diamond; male or female faces from the Chicago face database and Spanish male or female names). The timing of the experimental sequence of events was also adapted to fMRI time standards (cue duration: 0.5s, target duration: 0.5s, jitter: 2.5s : 0.7s : 6s).

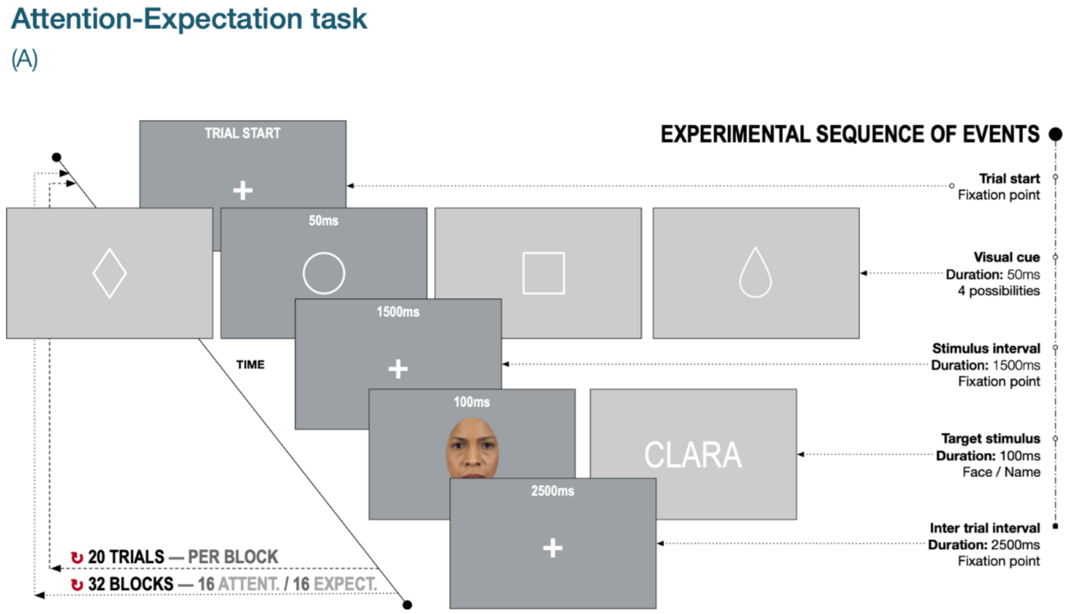


Figure 49. Attention-expectation task: experimental sequence of events

The complete sequence is repeated 640 times, grouped into 32 blocks. After the presentation of the stimulus, participants had to respond *"yes"* or *"no"* (by pressing the key A or L) to the sex discrimination task. Block order, cue shape, stimulus category and response keys are fully counterbalanced across participants to avoid potential confounds. Each sequence was categorized based on its validity (valid or invalid trials), block type (attention or expectation trials) and its category (names or faces). Face images were extracted from *The Chicago face database* (Ma et al., 2015).

6.2.2. Data acquisition and preprocessing

Electroencephalography data acquisition and preprocessing

The stimulus presentation and behavioral data collection were done in a Microsoft PC with The Psychophysics Toolbox on MATLAB (v.2020a). Stimuli were presented on an LCD screen (Benq, 1920x1080 resolution and 60 Hz refresh rate) over a grey background. We employed 160 male and female faces, 50% each, with ~6ºx9º visual angle, extracted from The Chicago Face Database (Ma et al., 2015) plus 160 unique Spanish male and female names (50% each, with ~8ºx2º visual angle). In the main task four different geometrical shapes (circle, square, raindrop and diamond with thin black outlines, unfilled, ~2ºx2º visual angle) were used as cues.

High-density electroencephalography was recorded from 64 electrodes mounted on an elastic cap (actiCap slim, Brain Products) in a magnetically shielded room at the Mind, Brain and Behavior Research Center (CIMCYC) of the University of Granada.

Impedances were kept below 10k. EEG activity was referenced to the FCz electrode during the recording session and signals were digitalized at a sampling rate of 1KHz.

Following previous literature (López-García et al., 2019, 2020), electroencephalography recordings were downsampled to 256Hz and digitally filtered between 0.1 and 120Hz using a high and low-pass FIR filters, preserving the phase information. Power line interference and its harmonics [50 and 100Hz] were removed using a notch filter. All channels were visually inspected and, on average, 1.85% of them were excluded due to excessive noise. EEG recordings were epoched [-1000, 2000ms locked at the target presentation] extracting data only from correct trials. To remove blinks and eye movements from the remaining data, Independent Component Analysis (ICA) was computed using the *runica* algorithm from EEGLAB (Delorme & Makeig, 2004). Artifactual components were rejected by visual inspection of the raw activity for each component, scalp maps and power spectrum. Then, an automatic trial rejection process was performed, pruning the data from no stereotypical artifacts. The trial rejection procedure was based on (1) abnormal spectra: the spectrum should not deviate from baseline by $\pm$50dB in the 0-2 Hz frequency window (which is optimal for localizing any remaining eye movement) and should not deviate by -100dB or +25dB in 20-40Hz (useful for detecting muscle activity); (2) improbable data: the probability of occurrence of each trial was computed by determining the probability distribution of values across trials, with a rejection threshold set at ±6 SD; and (3) extreme values: all trials with amplitudes in any electrode out of $\pm 150\mu V$ range were automatically rejected. On average, a total of 8% of the trials were automatically removed in the trial rejection stage. Finally, previously excluded channels were reconstructed by spherical interpolation and the entire dataset was average re-referenced and baseline corrected [-200, 0ms].

Functional MRI data acquisition and preprocessing

Functional MRI data was recorded in a 3T Siemens Trio scanner at the Mind, Brain and Behavior Research Centre (CIMCYC) in Granada (Spain). Functional images were obtained with a T2*-weighted echo planar imaging (EPI) sequence, with a TR of 1.73s.

Fifty interleaved slices with a thickness of 2.5 mm were obtained (TE = 30 ms, flip angle = 66°, voxel size of 2.5 mm°). The sequence was divided in 8 runs, consisting of 166 volumes each. After the functional sessions, a structural image of each participant with a high-resolution T1-weighted sequence (TR = 2250ms; TE = 4.18ms; flip angle = 9°, voxel size of 1mm^3) was acquired and subsequently defaced to preserve participants anonymity. We used SPM12 (Welcome Centre for Human Neuroimaging, 2018) to preprocess and analyze the neuroimaging data. The first 4 volumes of each run were discarded to allow saturation of the signal. The remaining volumes were realigned and unwrapped to correct head motion, followed by slice-timing correction. Then, T1 images were co-registered with the realigned functional images. Finally, they were normalized to the MNI space (transformation matrices were estimated from EPI images and applied to them in the same step) and smoothed using an 8 mm FWHM Gaussian kernel.

6.2.3. Representational Similarity Analysis: overview

The Representational Similarity Analysis is a multivariate computational technique employed to reveal fundamental insights about how information is represented in the brain. Note that the term *representation* is usually interpreted as the activity pattern induced in the brain by certain experimental conditions or stimuli (Kriegeskorte et al., 2008). First, brain activity is recorded employing a neuroimaging technique while the participant is performing a task. During the task, the participants are exposed to several experimental conditions or stimuli that evoke different spatiotemporal activity patterns across the brain. The main goal of RSA analysis is to understand these spatiotemporal representations by abstracting from the signal space to a high-order and common representational space (Nili et al., 2014). In order to do that, activity patterns associated with each pair of experimental conditions are related and visualized constructing 2-dimensional *Representational Dissimilarity Matrices* (RDMs). One of the most remarkable characteristics of RDMs is that they are comparable no matter the nature of the original data. Therefore, representations across brain regions, temporal points, individuals, neuroimaging modalities or even animal species are now easily comparable in this high-order representational space by computing the correlation between different RDMs (Kriegeskorte & Kievit, 2013). As stated in the Introduction section, this is one of the basic principles of multimodal data fusion: employing a common analysis framework that solves the intrinsic correspondence problem between different neuroimaging techniques. This approach is very flexible and versatile, allowing the comparison of RDMs in different contexts depending on the researcher's question and hypotheses (Popal et al., 2019).

6.2.4. Representational Dissimilarity Matrices

Representational Dissimilarity Matrices are the cornerstone of RSA technique. These matrices present pairwise dissimilarities between experimental conditions revealing how distinguishable are the brain-activity patterns associated with them. Thus, RDMs describes the geometry of the arrangement of these patterns in the representational space (Nili et al., 2014). By definition, RDMs are square matrices horizontally and vertically indexed by the set of stimuli (or experimental conditions), leading to a symmetric matrix along its diagonal (see Figure 51). Zero values at the diagonal represent the dissimilarity obtained by comparing each experimental condition to itself while values outside the diagonal result from pairwise comparisons between all possible combinations of two different experimental conditions.

6.2.5. Extracting RDMs from different neuroimaging modalities

Since RDMs are generated comparing the activity patterns elicited by different experimental conditions, they can be extracted no matter the nature of the original data. As an example, Figure 50 depicts how RDMs could be constructed from different data sources (M/EEG and fMRI). In our case, for temporal-accurate neuroimaging modalities such as EEG or MEG, a neural RDM is generated for each time point and participant. Alternatively, in fMRI data individual RDMs are usually constructed for each voxel or brain region and participant. The following paragraphs describe how representational dissimilarity matrices are extracted from both EEG and fMRI data.

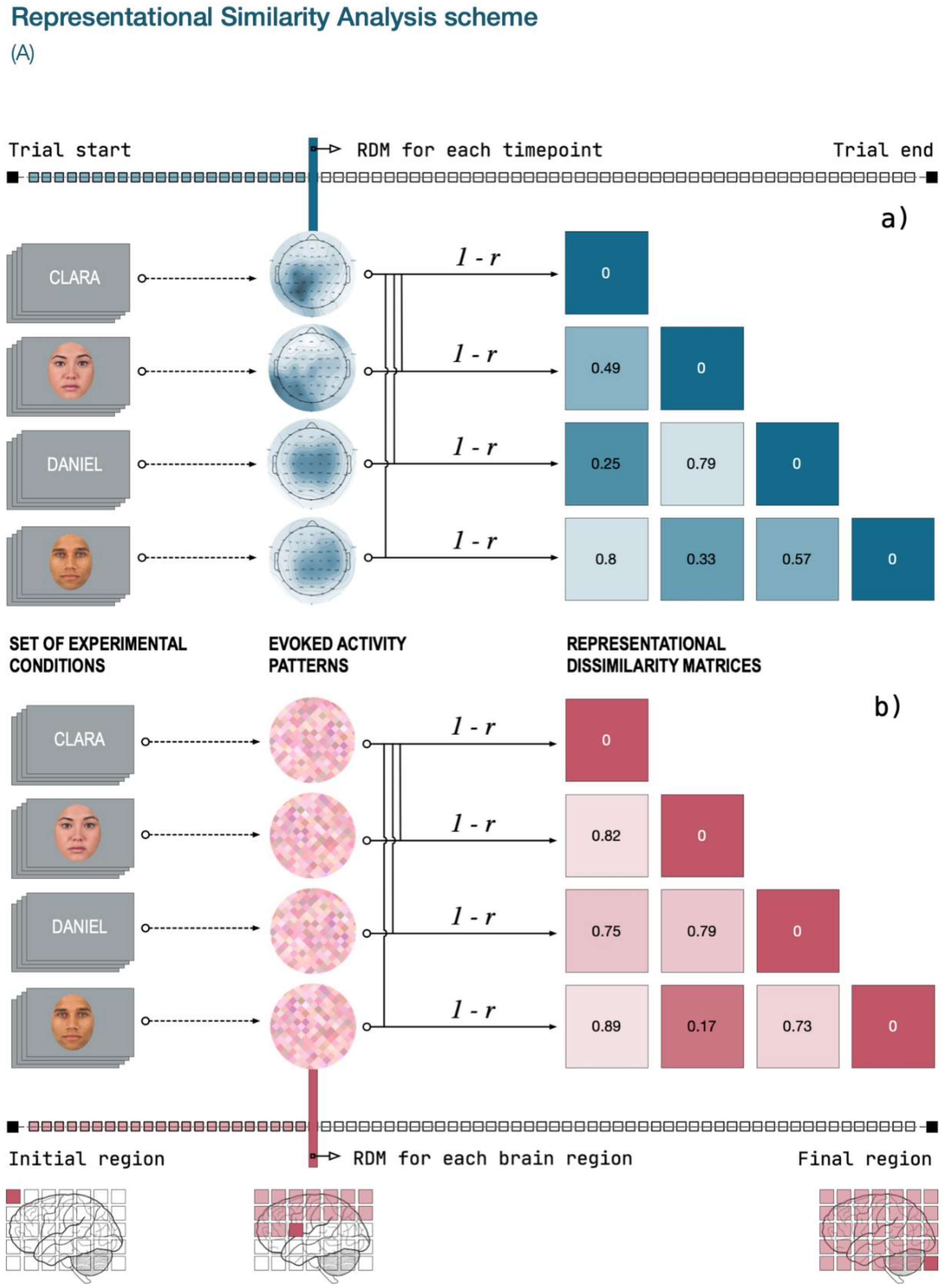
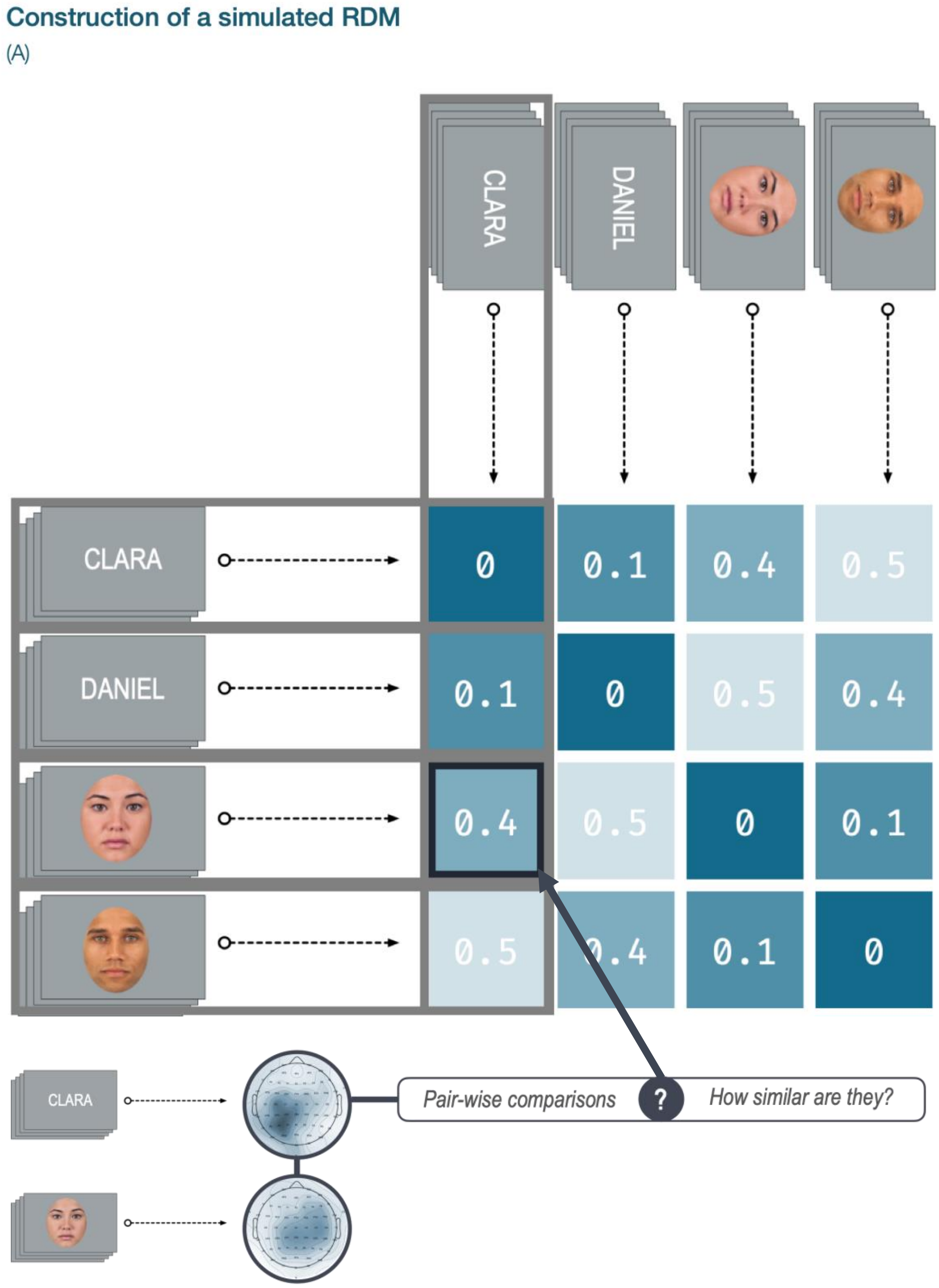


Figure 50. Representational Similarity Analysis scheme

During the experiment, participants are exposed to different stimuli or experimental conditions that elicit different condition-specific activity patterns in the brain. These patterns are then pairwise compared by computing a dissimilarity measure (e.g., $1 -$ Pearson's r) and arranged in the so-called Representational Dissimilarity Matrix (RDM). Depending on the neuroimaging modality employed these RDMs can be generated in **(a)** a time or **(b)** space-resolved manner.



	CLARA	DANIEL		
CLARA	0	0.1	0.4	0.5
DANIEL	0.1	0	0.5	0.4
	0.4	0.5	0	0.1
	0.5	0.4	0.1	0

Figure 51. Construction of a simulated RDM

The first column of the Representational Dissimilarity Matrix is constructed by computing pair-wise comparisons between the brain activity patters elicited by the first stimulus (e.g., a female name: Clara) and the brain activity patterns elicited by the whole stimuli set (e.g., female and male names, female and male faces).

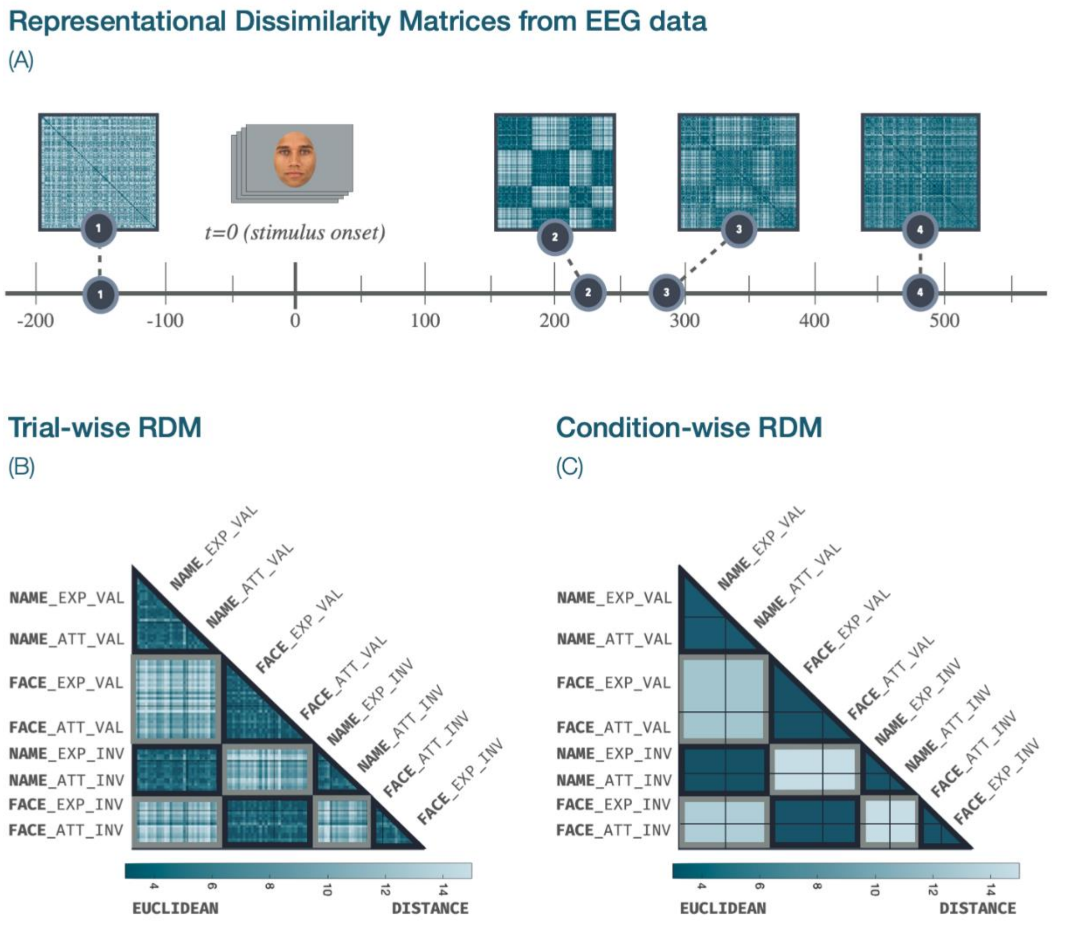


Figure 52. Construction of RDMs from EEG data

For each time point we can construct an RDM from the EEG data **(a)**. As shown, before the stimulus is presented, the dissimilarity values between different brain patters are randomly distributed. This makes sense because at this point there is no evoked activity since the stimulus was not presented yet. Around two hundred milliseconds after the stimulus onset, it is clearly shown that some brain patterns are more dissimilar than others. This effect disappears if we go further in time. The RDM can be constructed following two different approaches: trial-wise **(b)** and condition-wise **(c)**.

Constructing RDMs from M/EEG data

The preprocessed and epoched EEG data for different experimental conditions can be imported to MVPAlab regularly. First, in order to reduce the signal-to-noise ratio and to improve computational efficiency, trials belonging to same condition can be randomly or sequentially averaged in groups. Additionally, these trials are usually normalized (z-score) before constructing the RDM. Then, for each time point and participant, neural RDMs are estimated according to the process described in the Figure 51, employing the selected dissimilarity measure. See section 6.2.6 Computing dissimilarity measures for more detailed information about each measure available. Note that the RDM can be constructed either trial or condition-wise (Figure 52). If the RDM is constructed trial-by-trial, the dimension of the matrix will depend on the total number of trials per class available for each participant. The total number of trials per class does not have to be consistent across participants, which usually leads to different RDM sizes (e.g., trial x trial x timepoints). This is a problem if, in further analyses, we need to correlate several RDMs extracted from different neuroimaging modalities. This inconvenience can be easily solved by transforming the trial-by-trial matrix in a condition-by-condition matrix. To do so, the dissimilarity values belonging to each condition are averaged obtaining just a dissimilarity value per class. Since RDMs are square and symmetrical matrices along their diagonals, both the diagonal and the upper triangle can be removed for computational and data storage efficiency.

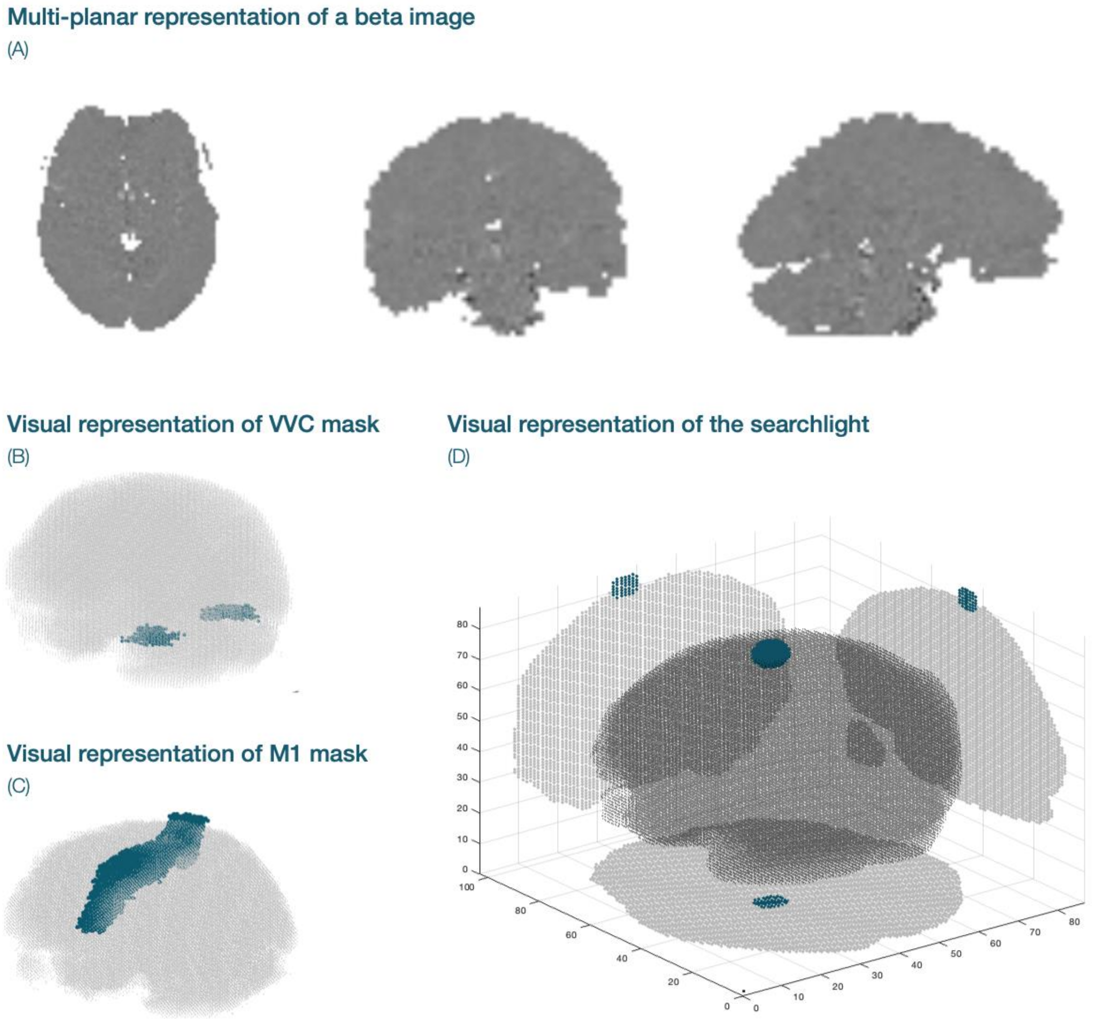
Constructing RDMs from fMRI data

Once the fMRI data is preprocessed, the contribution of each condition to the recorded fMRI time series should be estimated. To do so, a univariate General Linear Model analysis should be defined and computed. Thus, the variability of a dependent variable $\boldsymbol{y}$ (the empirical fMRI timeseries of an individual voxel) is explained in terms of a weighted sum of different reference functions (also called regressors):

$$\begin{bmatrix} y_1 \\ \vdots \\ \vdots \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & X_{11} & \cdots & \cdots & \cdots & X_{1p} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & X_{n1} & \cdots & \cdots & \cdots & X_{np} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \vdots \\ \beta_p \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \vdots \\ \vdots \\ \vdots \\ \epsilon_p \end{bmatrix}$$

Equation 34

The variable $\boldsymbol{y}$ at the left corresponds to the measured fMRI signal time course of a single voxel. The matrix $\boldsymbol{X}$ is also called the design matrix or the model and contains the predictor time courses as column vectors. These predictor time courses are the expected (ideal) fMRI responses for different conditions of the experimental paradigm. The beta values $\boldsymbol{\beta}$ (also known as beta maps or beta images) quantify the contribution of each regressor in explaining the observed fMRI time course. Finally, the variable $\boldsymbol{\epsilon}$ represents the residuals of the model, also called noise or prediction error, which accounts for the remaining unexplained data. After computing the GLM analysis, a single beta image (volume) is obtained for each experimental condition included in the model. As explained before, the intensity values of each voxel in this beta image represents how the condition contributed to the empirical fMRI signal (Figure 53a).



Multi-planar representation of a beta image

(A)

Visual representation of VVC mask

(B)

Visual representation of M1 mask

(C)

Visual representation of the searchlight

(D)

Figure 53. Construction of RDMs from fMRI data

(a) This figure shows the multiplanar representation (axial, coronal and sagittal planes respectively) of the beta image estimated for a specific condition and participant. (b-c) Two different brain masks corresponding to visual (Ventral Visual Cortex, VVC) and motor (Primary Motor Cortex, M1) areas of the human brain. (d) Visual representation of the searchlight analysis. The figure shows the 3D volume of the brain, the sliding sphere and their projections.

The RDMs are constructed from these beta images following two different approaches. Firstly, in the simplest approach, individual RDMs can be extracted from different regions of interest (ROI) based on *a priori* knowledge and on a strong hypothesis about the brain region being involved in a specific process. In this case, all the beta images are masked employing a brain mask, which can be previously extracted from a brain atlas and adapted to fit the beta image of the participant. This mask selects only the voxels that belong to a specific ROI from all the beta images (Figure 53b,c). The selected data is then used to construct the RDM of the region, the dimension of which depends on the number of conditions. Note that, if the stimulus set is the same for different experiments, the dimensions of the resulting RDMs are identical, thus comparable.

On the other hand, in some situations there is not a strong hypothesis about the region involved in specific brain functions. In this case, an exploratory analysis of the whole brain could be more adequate. To do so, a small sphere of radius ***r*** is defined and centered inside the brain volume, selecting only the voxels contained in it. Thus, the RDM of this specific brain region is constructed employing only these selected voxels. Then, the spere is moved voxel-by-voxel across the whole brain, repeating the construction process and obtaining an RDM for each voxel in the brain volume (Figure 53d). This procedure is known as searchlight, and it was proposed by Kriegeskorte et al. in 2006.

6.2.6. Computing dissimilarity measures

Dissimilarity between activity patterns can be assessed employing several dissimilarity measures. Selecting an adequate one is one of the most important decisions in RSA since it usually depends on the inferential aim of the analysis and the original data (Walther et al., 2016). In the MVPAlab Toolbox we implemented three of the most popular measures to estimate neural dissimilarity matrices: the correlation distance (1–Pearson's *r*), the Euclidean distance and the Mahalanobis distance.

Pearson's dissimilarity matrices

Given a data matrix $X_{M\times N}$, where M represents the total number of trials/conditions and N the total number of electrodes/voxels (observations), the neural dissimilarity matrix for any time point or brain region is defined as follows:

$$RDM = 1 - R$$

Equation 35

where R is the *correlation coefficient matrix* of M random variables with N scalar observations each. For $M = 2$ random variables A and B (trials or conditions), the correlation coefficient matrix includes the correlation coefficients for each pairwise variable combination and is defined as follows:

$$R = \begin{pmatrix} \rho(A,A) & \rho(A,B) \\ \rho(B,A) & \rho(B,B) \end{pmatrix}$$

Equation 36

Since A and B are directly correlated to themselves, the values of the diagonal entries are always one:

$$R = \begin{pmatrix} 1 & \rho(A,B) \\ \rho(B,A) & 1 \end{pmatrix}$$

Equation 37

The correlation coefficient ρ of two random variables is an indicator of their linear dependence. If each variable has N scalar observations, the Pearson correlation coefficient is defined as follows:

$$\rho(A,B) = \frac{1}{N-1} \sum_{i=1}^{N} \left(\frac{A_i - \mu_A}{\sigma_A} \right) \left(\frac{B_i - \mu_B}{\sigma_B} \right)$$

Equation 38

where μ_A and σ_A are the mean and the standard deviation of A, and μ_B and σ_B are the mean and standard deviation of B. Alternatively, the correlation coefficient can be defined in terms of the covariance matrix of A and B as follows:

$$\rho(A,B) = \frac{\mathrm{cov}(A,B)}{\sigma_A \sigma_B}$$

Equation 39

The values of the coefficient can range from -1 to 1, with -1 representing a direct, negative correlation, 0 representing no correlation and 1 representing a direct, positive correlation.

Thus, the representational dissimilarity matrix obtained by Equation 35 is a representation of how dissimilar each pair of experimental conditions are at certain time point or brain region.

Distance-based dissimilarity matrices

A distance metric is a function that defines a scalar value of distance between two observations. Given two data matrices $X_{M\times N}$ and $Y_{M\times N}$ where each row is treated as vector (1-by-N) $x_1, x_2, \dots, x_N$ and $y_1, y_2, \dots, y_N$, the various distances between the vector x_j and x_k are defined as follows:

Euclidean distance:

$$d_{j,k} = \sqrt{(x_j - y_k)(x_j - y_k)^T}$$

Equation 40

Mahalanobis distance:

$$d_{j,k} = \sqrt{(x_j - y_k)C^{-1}(x_j - y_k)^T}$$

Equation 41

where C^{-1} is the covariance matrix.

Thus, the distance-based representational dissimilarity matrix can be obtained computing the distance (Equation 40, Equation 41) between each pair of row vectors of a $X_{M\times N}$ data matrix, where M is the total number of experimental conditions (random variables) and N the total number of electrodes (observations).

Cross-validated approaches

Since distances are by definition non-negative and larger than zero for noisy data, cross-validated approaches have been proposed as a more reliable alternative (Walther et al., 2016). As an example, a popular cross-validated measure is the Cross-validated Mahalanobis Distance, also known as Linear Discriminant Contrast (LDC), which is a continuous measure that lacks a ceiling effect, it has a meaningful zero point when two patterns are identical and it includes a cross-validation loop, which makes it less prone to biases.

Given a data matrix $X_{M \times N}$ where row vectors (1-by-N) $x_1, x_2, \dots, x_N$, correspond to each experimental condition/trial and N is the total number of observations, The Linear Discriminant Contrast between each pair of conditions j and k is defined as follows (Bueno & Cravo, 2021):

$$LDC_{j,k,i} = \left(x_{j,i} - x_{k,i}\right)_A \times pS \times {\left(x_{j,i} - x_{k,i}\right)_B}^T$$

Equation 42

where $x_{j,i}$ and $x_{k,i}$ are row vectors containing the total number of observations (neural activity) for the pair of conditions j and k and for a specific time point or brain region i. A and B represent different partitions (folds) of the original data and pS is the pseudo inverse covariance matrix between the residuals of $x_{j,i}$ and the residuals of $x_{k,i}$ from data partition A. The residuals are calculated by subtracting the activity of each trial and electrode for a specific time point from the mean activity for that electrode at that time point. Finally, the obtained distance values are averaged across cross-validation folds. The Linear Discriminant Contrast measure is already under development and will be fully integrated in future versions of MVPAlab.

6.2.7. Theoretical RDM models

Representational Dissimilarity Matrices can be generated not only from neural or behavioral sources but also from theoretical predictions. These are not empirical but conceptual RDMs and they are generated based on expected theoretical dissimilarities between experimental conditions. Hence, to test the RSA technique we built tree different conceptual RDMs based on (1) stimulus type: names vs. faces, (2) block type: attention vs. expectation, and (3) trial validity: valid vs. invalid. As shown in Figure 54 a binary value of expected dissimilarity [0|1] was assigned to each pair of experimental conditions, represented in dark and light green color in the figure. For example, focusing on the stimulus category, we can assume that activity patterns elicited by faces are more similar between them than those elicited by names (Figure 54b). Similarly, attending to the type of the block, we can hypothesize that activity patterns elicited by stimulus in attentional blocks differ from those elicited in expectation blocks (Figure 54c), and the same occurs for validity (Figure 54a).

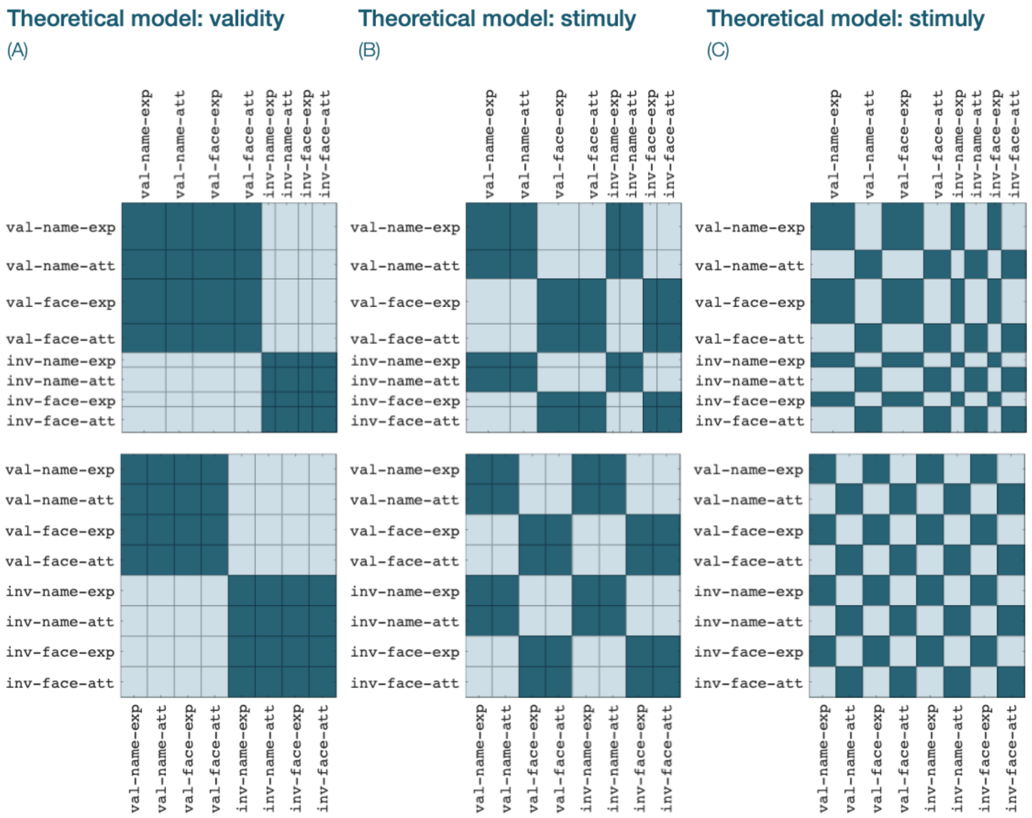


Figure 54. Theoretical RDM models

Three different conceptual RDMs built based on the three main variables in our data: trial validity (valid or invalid), stimulus category (names or faces), block type (attention or expectation). Note that each model can be constructed assuming an identical number of trials per condition (bottom row of the figure), leading to symmetrical cell sizes, or adjusting the size of the cell according to the actual number of trials per condition (top row of the figure).

Estimation of the explained variance for theoretical models

One of the main advantages of RDMs is their versatility, which allows to define several types of analyses to answer different questions. For instance, once the neural matrices and the theoretical models are defined, the share of neural variance explained for each theoretical model can be estimated (Figure 55). To do so, a linear regression is fitted at each time point and for each participant, where the empirical RDM acts as independent variable and the three theoretical models as regressors:

$$Y = X * \beta + \epsilon$$

Equation 43

where Y is the observed data (empirical matrix), X corresponds to regressors, β coefficients are scaling values (unknown) and ϵ is the regression error (residuals). In our case, if we want to estimate the variance explained for each of the three theoretical models:

$$Y = [RDM_1 \quad RDM_2 \quad RDM_3] * \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} + \epsilon$$

Equation 44

$$RDM_{neural} = \beta_{val} * RDM_{val} + \beta_{stim} * DRM_{stim} + \beta_{block} * RDM_{block} + \epsilon$$

Equation 45

If the previous regression is fitted for each time point in electroencephalography data, the obtained time series of β values indicate how well our theoretical models explain the empirical data in a time-resolved way. Equivalently, this regression can also be fitted for each voxel in fMRI data, obtaining a spatial representation of how well our theoretical models explain the empirical data in different brain regions.

6.2.8. Second order dissimilarity analysis

Once the neural and theoretical RDMs are estimated, the so called second order dissimilarity analysis can be computed. Since an RDM is square and symmetrical matrix along its diagonal, both the diagonal and the upper triangle are removed for computational efficiency. Then, for each time point or brain region, participant and theoretical model, the resulting matrices were vectorized and compared employing the Spearman correlation. As a result, we obtained a time series of Fisher's Z transformed correlation coefficients for each theoretical model and participant (see Figure 57a). Finally, we computed a two-tailed cluster-based permutation analysis to test if the obtained coefficients were significantly higher or lower than zero (10^5 permuted iterations and $\alpha = 0.001$ for both group-average and cluster-size levels).

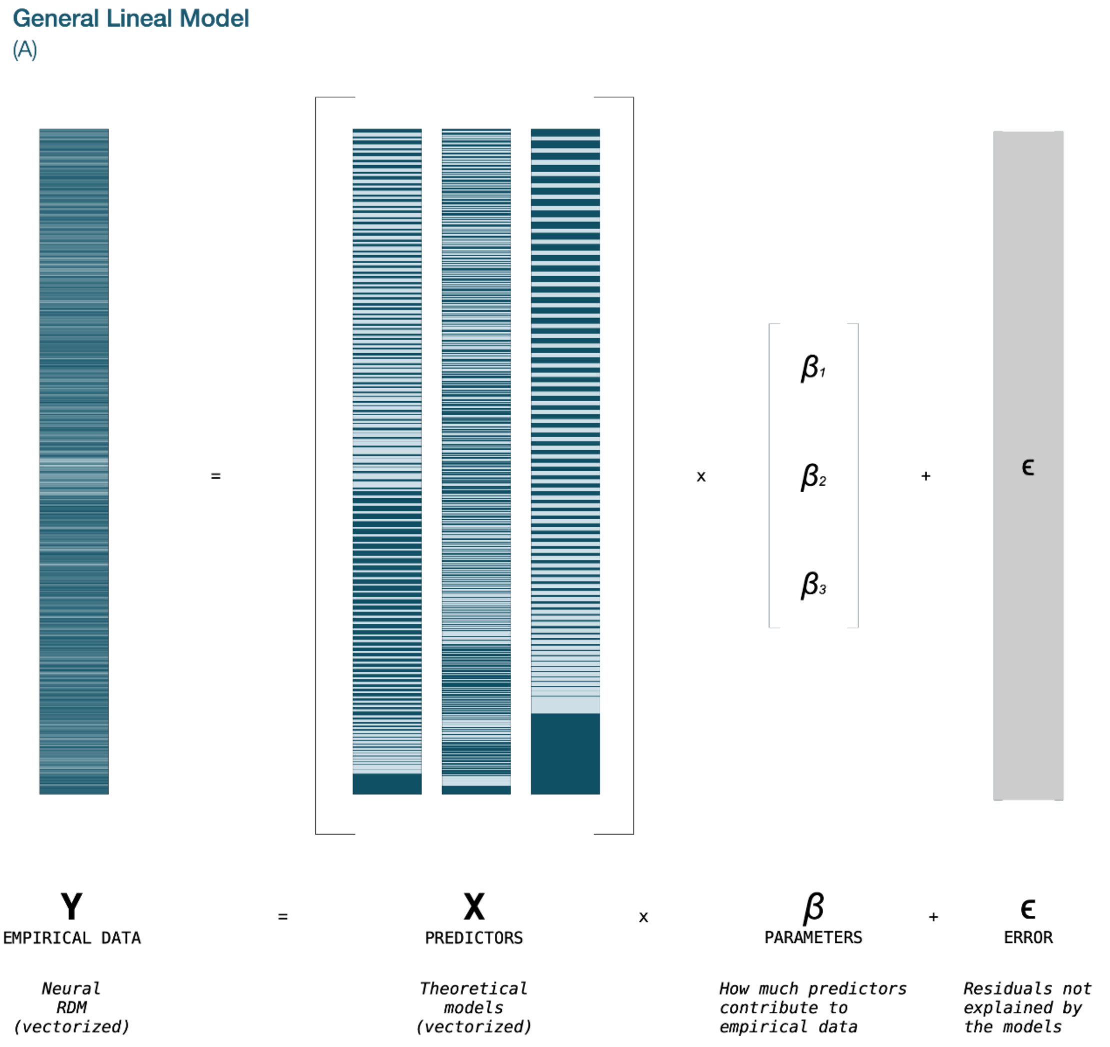


Figure 55. Estimation of the explained variance for theoretical models

The contribution of each predictor X (our theoretical models) to the observed empirical data Y (neural Representational Dissimilarity Matrix) can be estimated according to the General Lineal Model, where β values represent how much predictors contribute to the empirical data and ϵ corresponds to the residuals of the regression (information not explained by the predictors).

6.2.9. Fusing information from different neuroimaging modalities

One of the most remarkable characteristics of different RDMs is that they are comparable no matter the nature of the original data. Therefore, representations across brain regions, temporal points, individuals, neuroimaging modalities or even animal species are now easily comparable. To do so, the correlation between different fMRI and EEG-RDMs should be computed. As shown in Figure 56, regardless of the selected analysis (ROI / searchlight analysis), each fMRI matrix (one matrix per ROI / voxel) can be correlated with all the RDM extracted from the EEG data (one matrix per time point). This procedure leads to correlation values time series that represent the spatiotemporal dynamics of the cognitive process occurring in a particular area of the brain at a particular time point (Radoslaw M. Cichy & Oliva, 2020; Radoslaw Martin Cichy et al., 2014; Muukkonen et al., 2020).

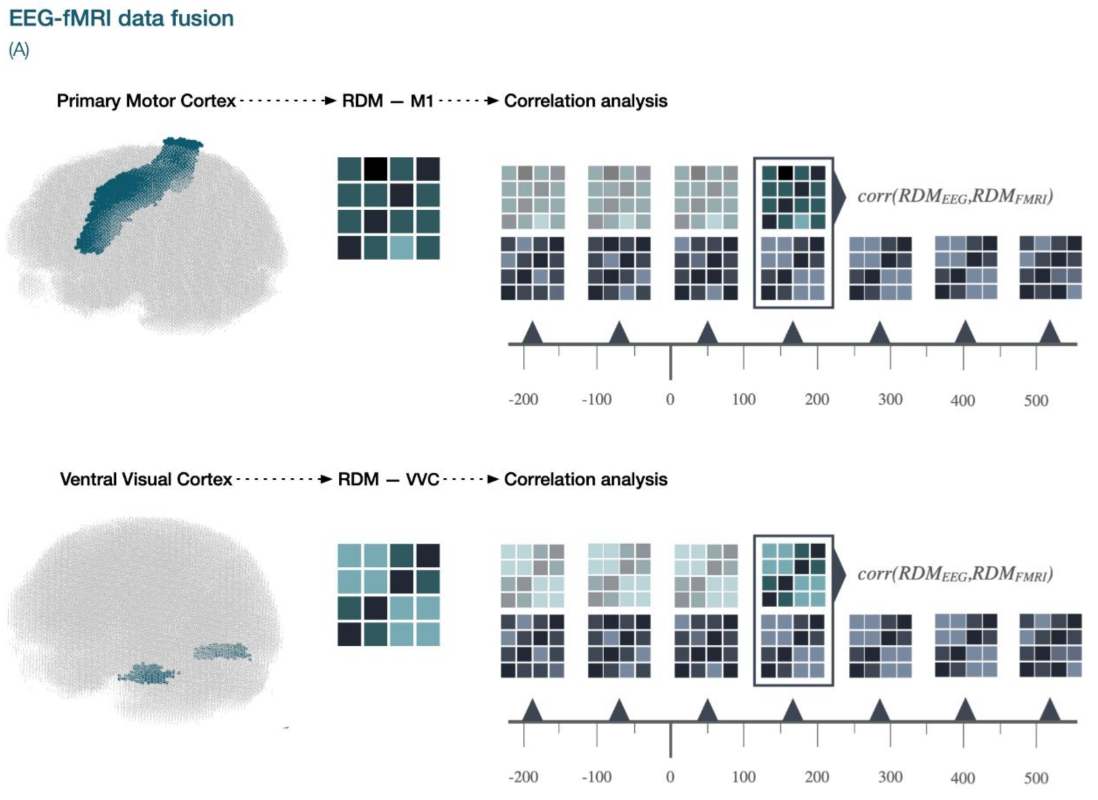


Figure 56. EEG-fMRI fusion analysis representation

To compute the EEG-fMRI fusion analysis, the EEG RDMs should be firstly constructed from each time point. Then, the fMRI RDMs are extracted from different ROIS of the brain. Both EEG and fMRI matrices can be correlated leading to correlation timeseries that represent the spatiotemporal dynamics of the studied cognitive process.

6.3. Results and discussion

The estimated neural dissimilarity matrices time series for a specific participant is shown in the Figure 57a. Three different time points are depicted in the figure, $t = -80$ms, $t = 220$ms and $t = 380$ms. As shown, before the presentation of the stimulus ($t = -80$ms) the neural RDM does not visually present a clear pattern, an indication of low or zero correlation with any of the theoretical models as it will be outlined below. However, 220ms - 3800ms after the presentation of the stimulus the RDMs depict a clear dissimilarity pattern which resembles one of our theoretical models, and leads to positive correlation with that particular model. The Euclidean distance was the selected dissimilarity measure, but the analysis was repeated using other measures such as $1 -$ Pearson's r obtaining equivalent results. Note that the cell sizes are not symmetrical due to the different number of trials per class, which means that the analyses were done in a trial-by-trial manner. Thus, the size of the matrices (and the experimental conditions distributions) is different for each participant, which implies that theoretical RDMs should also be specifically generated and adapted for them. This also means that a direct comparison of RDMs between participants (or neuroimaging modalities) is not allowed, weakening the actual potential of this analytic framework and its applications. This inconvenience can be easily solved by collapsing trial information and computing the analysis in a condition-by-condition manner. This way, if the same set of experimental conditions is employed this approach could combine information from different participants, neuroimaging modalities or even animal species.

The results of the second order Representational Similarity Analysis are depicted in the Figure 57b and c. As shown, the Spearman correlation was computed between the neural RDMs and the three theoretical models for each participant in a time-resolved way. Statistically significant results ($\alpha = 0.001$) were obtained for our three theoretical models. Positive correlations were found practically the entire time windows for the stimulus category model, yielding a correlation coefficient of Fisher(ρ)$\sim$ 0.8 in $t = 220$ms after the stimulus onset, which indicates how differently the brain represents faces vs. non-faces information.

Finally, the preliminarily results of the fMRI-EEG fusion analysis are shown in the Figure 58. To compute this fusion analysis, for each participant, the neural dissimilarity matrices time series for the EEG data were reconstructed in a condition-by-condition manner.

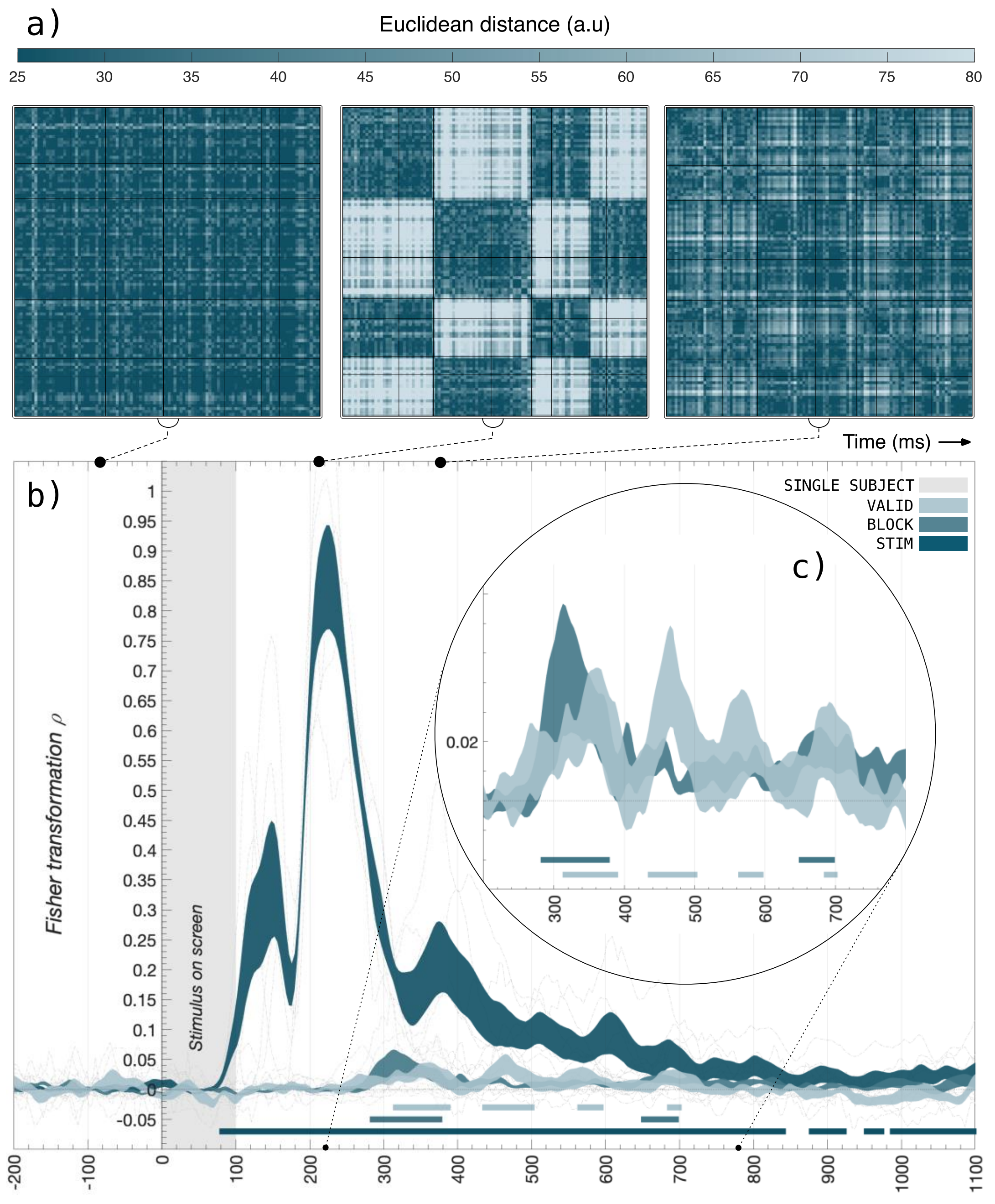


Figure 57. Representational similarity analyses results

(a) Neural RDMs estimated through the Euclidean distance for a specific participant at three different time points: t=-80ms, t=220ms and t=380ms. **(b)** Second order analysis: Time series of the group-averaged Fisher's Z transformation of the ρ correlation coefficient. Shaded areas represent the Standard Error of the Mean for each theoretical model. Dashed grey lines represent the single participant results. Statistically significant time windows are highlighted using horizontal bars. **(c)** Enlarged version of the previous diagram to better appreciate the positive correlations between neural RDMs and theoretical block type and validity models.

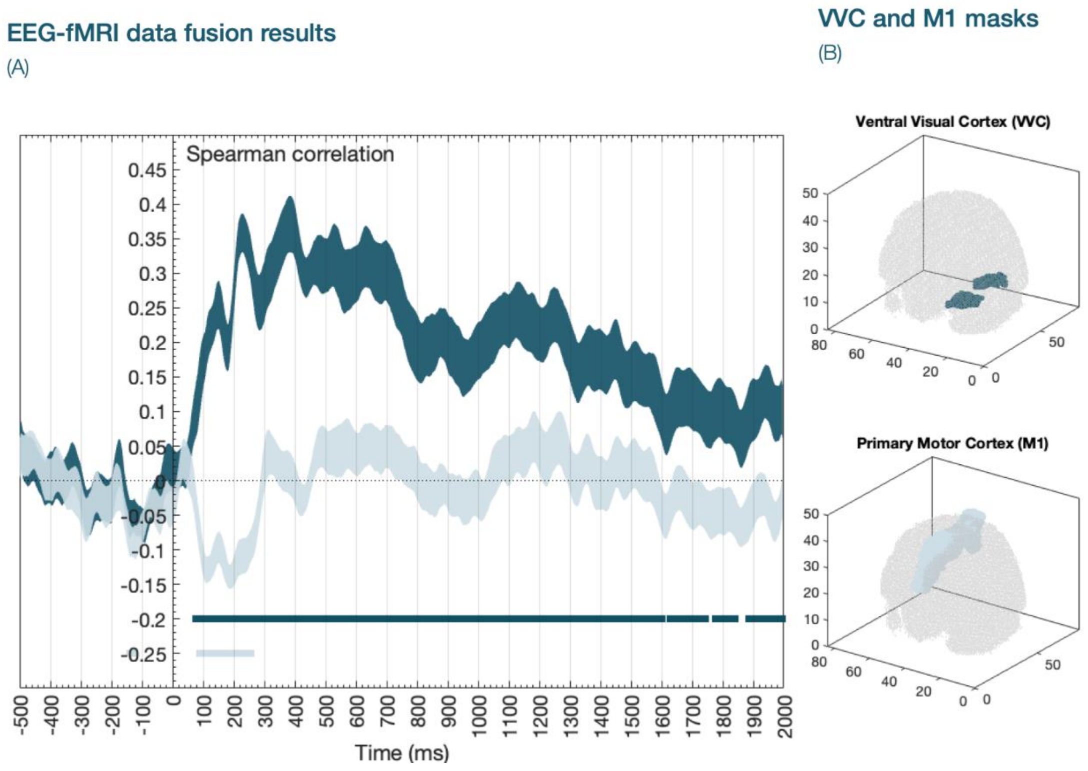


Figure 58. EEG-fMRI fusion results

(a) This picture shows the results of the EEG-fMRI fusion analysis for two different areas of the brain, the Ventral Visual Cortex (VVC) and the Primary Motor Cortex (M1). Each correlation time course represents the group-averaged Fisher's Z transformation of the ρ correlation coefficient. Shaded areas represent the Standard Error of the Mean of the group-averaged time courses for each brain region. **(b)** Visual representation of the employed brain masks for extracting the RDMs from the fMRI data.

Similarly, for each participant, two RDMs were constructed from two different brain areas, the Ventral Visual Cortex (VVC) and the Primary Motor Cortex (M1). We chose these regions to test the reliability of the fusion analysis, since the VVC is usually associated with the visual processing function while M1 is associated with motor functions. The mean RDMs from the VVC and M1 regions were computed by averaging individual RDMs across participants. Then, the mean RDMs of the VVC and M1 regions were correlated with the RDMs extracted for each EEG participant. The correlation time courses obtained for each participant were finally averaged resulting in the fusion analysis depicted in Figure 58.

As expected, this preliminary fusion analysis shows statistically significant positive correlations ($\alpha = 0.001$) between EEG and fMRI data in visual areas (VVC) for almost the entire time window t = [60-2000]ms. On the other hand, we also found a small significant cluster of negative correlation in motor areas after the stimulus onset t = [70-250]ms. This unexpected result could be caused by different aspects, such as some artifact on the data, and requires further investigation.

6.4. Conclusions

The current work is a preliminary approximation to multimodal data fusion techniques based on Representational Similarity Analysis. We developed and tested this analytic framework for electroencephalography and functional magnetic resonance signals employing a sample dataset specifically designed to study differences in preparation mechanisms. One of the many potential applications of RSA analysis is to compute correlations between neural dissimilarity matrices and theoretical models based on our predictions to study how information is represented in the brain. Three theoretical models were designed attending to the three main variables in our experiment: stimulus category, block type and trial validity. Statistically significant positive correlations were found between the empirical and theoretical models, yielding a Fisher(ρ)$\sim$ 0.8 for the stimulus category model 220ms after the stimulus onset. Additionally, we also implemented and tested the EEG-fMRI data fusion analysis, a powerful methodology which combines information from EEG and fMRI data. This analytic strategy allows to study the spatiotemporal dynamics of the cognitive process occurring in a particular area of the brain at a particular time point. We obtained promising results in our preliminary EEG-fMRI fusion analysis. We found positive correlations in visual areas 60ms after the stimulus onset, yielding a Fisher(ρ)$\sim$ 0.4, 400ms after the stimulus onset. In addition, we also found a small cluster of negative correlation in motor areas 70ms after the stimulus onset. Thus, further investigation and programing work is needed for a complete implementation and integration of multimodal fusion methods in our tool.

The main goal of this work was to integrate the Representational Similarity Analysis framework in a user-friendly software tool such as The MVPAlab Toolbox and implement, discuss and test some of its potential applications, not just in M/EEG but also in combination with other neuroimaging techniques such as fMRI, preserving their individual strengths while overcoming their weaknesses. Therefore, this work increments the versatility of the MVPAlab Toolbox and more importantly, it serves as a steppingstone for developing forefront techniques such as multimodal fusion models.

Section III.

General discussion

Chapter 7. General discussion and conclusions

The different contributions made in this PhD thesis have been meticulously discussed in previous chapters. Here, we provide an overview of how this work fits within the present panorama of Machine Learning and Human Neuroscience.

7.1. General discussion

Artificial Intelligence and Machine Learning-based applications have been recently introduced in several areas of our daily lives. A simple search on the internet returns a ton of applications using these technologies to provide services that we use on a daily basis which we are not aware of. E-commerce or entertainment companies invest a vast amount of human and financial resources to develop efficient recommendation algorithms. Based on your personal preferences, they provide a selection of the most interesting TV shows to watch or a suggestion for the next item you need to buy for your kitchen. On occasions more accurately than others. We use artificial intelligence at all times. Just to provide a few more examples, our smartphone detects our face or other biometrical patterns every time we unlock it. When we write a message to our relatives, a machine learning algorithm corrects our spelling and grammar, or predicts the next word to type based on the context. When we take a picture of our dog, several scene-understanding technologies detect objects and depths to optimize contrast, lightning, sharpness, etc., improving the quality of the photo. AI algorithms are also applied in less mundane applications such as online fraud detection, traffic prediction, speech recognition, in the prediction of stock market trends or in computer vision, a crucial technology for the development of self-driving cars. The potential of AI seems endless but also brings concern about wrongful applications such as the creation of synthetic media (deepfakes). The scalation of this fake but realistic content could present a negative impact in the society, promoting fraud or misrepresenting influential personalities in videos. Although some of these scenarios could seem overwhelming or even concerning for some people, when responsibly used, machine learning presents undeniable advantages and direct benefits in our lives. For example, in the field of health care, machine learning algorithms allow to recognize patterns of information in human genome for personalized treatments, and computer-aided diagnosis provides fast and more precise diagnoses allowing early detection and better prognosis of some diseases. Be that as it may, we cannot deny that artificial intelligence has drastically changed the way we interact with the world.

Naturally, these technical advantages are also present in most scientific disciplines, including Neuroscience. The use of machine learning-based algorithms in Cognitive Neuroscience has propelled a more accurate understanding of the human brain function, since these analytical frameworks allow to detect subtle changes in brain signals that classical approaches have usually overlooked. Although the use of these techniques is broadly employed in several scientific fields it is worth highlighting some considerations related to their application when working with neuroimaging data. Machine learning models usually work with thousands or millions of observations and have the aim to obtain the best possible performance in a specific scenario. However, in the Cognitive Neuroscience field things work slightly differently. Firstly, the sample size is way more reduced, reaching a few hundred of samples in best scenarios (it is common to work with a few tens of observations per class, which usually leads to having more features than observations). Additionally, the main objective of most cognitive neuroscience studies is to detect subtle changes in brain patterns that differ significantly and generalize well across the population, and more importantly, how these patterns are related with different human brain functions. Thus, obtaining the highest possible accuracy becomes irrelevant in most cases. For that reason, hyperparameter optimization, a commonly employed procedure in most Computer Science applications, is no longer relevant. This, in conjunction with the use of *simple* algorithms such as linear classifiers, could lead to suboptimal decision boundaries according to the Computer Science perspective. However, these setups are preferred in Neuroscience due to the small sample size, the computational cost saved. They guarantee the reliability, generalization and interpretation of the results.

Being aware of the current expansion of these machine learning-based methodologies, the lack of knowledge and coding skills of non-multidisciplinary research groups, and the importance of a correct application of these methodologies in cognitive neuroscience environments, we decided to develop The MVPAlab Toolbox. This powerful, flexible and easy-to-use MATLAB-based application encapsulates the most recent and widely extended methodologies for data analysis in the Cognitive Neuroscience panorama. In the earlier stages of its development, we tested the suitability of multivariate and machine learning-based methods when applied to the electroencephalography signal in a real Neuroscience experiment, more specifically in relation of Cognitive Control. In order to truly understand the nature of a Cognitive Neuroscience experiment and all the stages of its development, we decided to start from scratch by designing our own experiment, setting the recording equipment, piloting, collecting, cleaning and preprocessing the recorded data. The complete process is fully described in Chapter 4. To analyze the recorded dataset, we developed a set of computational routines for feature extraction, feature selection, data normalization, data smoothing, etc. We also developed different multivariate pattern recognition analyses (based on linear classifiers) such as time-resolved classification and cross-classification algorithms, temporal generalization matrices, frequency contribution analyses and non-parametric permutation tests to address the statistical significance of the results. We also studied the effect of different classification parameters (e.g., the number of averaged trials) in the performance of the model, which was obtained employing different metrics, such as the raw accuracy, the balanced accuracy or the Area Under the Curve. This first implementation showed significant differences (p<.001, cluster corrected) in activity patterns for congruent vs. incongruent trials, allowing us to accurately predict (>80%) if participants were responding to congruent or incongruent trials. Thanks to these methodologies we decoded conflict-related neural processes associated with congruent or incongruent events in a time-frequency resolved way. Our results replicated findings obtained with classical approaches and added up to existing literature, offering new information regarding the dynamics of the underlying mechanisms.

Probed the feasibility of using these methodologies in Cognitive Neuroscience contexts, we decided to improve and extend the previous analyses, giving rise to the first version of the MVPAlab Toolbox. In Chapter 5 we meticulously described all the implemented functionalities. We also developed an extensive online documentation of the toolbox and a detailed theoretical explanation of each implemented analysis. With the aim of differentiating The MVPAlab Toolbox from other existing software alternatives we implemented not only the aforementioned set of computational resources, but we developed a complete integrated environment for computing multivariate analyses in M/EEG and fMRI data. We included a complete and easy-to-use graphic user interface and provided a visualization software for data representation generating ready-to-publish graphics and video animations.

Although the developed toolbox presents a huge potential analyzing brain signals in temporal terms, the spatial dimension of brain function has not been considered so far. Different software packages implement multivariate analyses for space-resolved signals (e.g., the fMRI) or time-resolved ones, but not simultaneously for both. In Chapter 6 we addressed one of the major challenges in the Cognitive Neuroscience field: to characterize cognitive processes in time and space. To do so, we expanded de capabilities of the MVPAlab Toolbox developing and testing multimodal fusion methods, which combine information of non-concurrent recordings from different neuroimaging modalities, preserving their individual strengths while overcoming their weaknesses.

In summary, with this work we proved the feasibility of machine learning-based multivariate analysis in the study of brain function, more specifically employing M/EEG and fMRI data. We also provided a complete tool that helps neuroscientists with different coding skills to push forward their studies, using innovative methodologies that allow to extract richer information from their data.

7.2. General conclusions

Finally, in this last section we provide a solid overview of this PhD thesis:

- First, we successfully employ multivariate and machine learning-based pattern recognition techniques in electroencephalography data to decode conflict-related neural processes associated with congruent or incongruent events in a time and frequency resolved way. Our results replicate findings obtained with other classical analytical approaches and offer new information regarding the dynamics of the underlying mechanisms. Thus, multivariate classification techniques represent an opportunity to study the neural basis of complex psychological processes.
- Given the recent and constant increase of multivariate analyses and its application in cognitive Neuroscience, we decided to develop The MVPAlab Toolbox, a very flexible, powerful and easy-to-use decoding toolbox for multi-dimensional electroencephalography data, including an intuitive GUI for creation, configuration, and execution of different classification analyses. We also developed a flexible data representation utility, which generates ready-to-publish data representations and temporal animations.

- We implemented and tested several classification functionalities, such as time-resolved binary classification, temporal generalization, multivariate cross-classification analyses, feature contribution analyses, non-parametric permutation based statistical analyses to find significant clusters in our results, and so forth.
- We also implemented exclusive analyses and functionalities, such as parallel computation, which significantly reduced the execution time, or frequency contribution analyses, which study how relevant information is coded across different frequency bands.
- Additionally, we offer several data preprocessing routines such as trial averaging, which increase the SNR of the EEG data, different modalities of data smoothing and data normalization, dimensionality reduction techniques such as PCA, among others.
- Despite the intrinsic potential of multivariate analyses when applied to EEG data, the neural activity should be identified simultaneously in time and in space to accomplish a better understanding of cognitive processes. For that reason, we decided to extend the capabilities of the MVPAlab Toolbox including support not only for EEG but for fMRI data.
- We developed and tested multimodal fusion methods based on the Representational Similarity Analysis. These techniques combine information of non-concurrent recordings from EEG and fMRI neuroimaging modalities, preserving their individual strengths and overcoming their weaknesses.
- The Representational Similarity Analysis framework abstracts from the signal space to a common representational space, generating commensurable dissimilarity matrices for different experimental conditions and thus allowing the fusing of different neuroimaging data.
- We implemented fusion analysis for both whole-brain (voxel-wise) and for specific brain regions of interest (ROIs). To evaluate this analytic framework, we built a demo dataset from pre-recorded EEG and fMRI experiments designed to study preparation to process faces vs. words. Preliminary results showed early coding in visual and motor regions, which demonstrate the potential of these new techniques in the future of Neuroscience.

Summary in Spanish

La Inteligencia Artificial y el aprendizaje máquina conforman el motor de incontables aplicaciones que, en los últimos años, se han hecho un hueco en nuestro día a día. Una simple búsqueda en Internet puede servirnos para darnos cuenta de la ingente cantidad de servicios de los que, aun usándolos diariamente, no somos conscientes que su funcionamiento se basa en inteligencia artificial. Por dar un ejemplo muy sencillo, tanto las compañías de comercio electrónico como las distribuidoras de contenidos multimedia a la carta invierten una gran cantidad de recursos, tanto económicos como humanos, en el desarrollo de algoritmos eficientes de recomendación. Estos algoritmos aprenden de nuestros hábitos y preferencias para ofrecernos una selección de contenido que puede ser de nuestro interés o, en caso de las webs de comercio electrónico, el último utensilio que puede hacernos la vida más fácil en la cocina. Lo cierto es que utilizamos la inteligencia artificial casi todo el tiempo. Cuando desbloqueamos nuestro teléfono móvil, este escanea nuestra cara o algún otro rasgo biométrico para comprobar que efectivamente somos los propietarios. Cuando enviamos un mensaje a nuestros familiares por una aplicación de mensajería un algoritmo basado en inteligencia artificial nos propone la siguiente palabra a escribir, además de corregirnos faltas de ortografía y gramática. Cuando le hacemos una foto a nuestra mascota, los algoritmos de detección de escena analizan, entre otros, los objetos presentes en la fotografía, así como la profundidad de la escena. Gracias a estos algoritmos nuestro teléfono mejora aspectos de las fotografías que hacemos, como el contraste, los efectos de luz, la nitidez, el enfoque, los colores, etc.

Todas estas tecnologías basadas en inteligencia artificial también se implementan en otros ámbitos mucho menos mundanos, como la detección de fraude, la predicción del tráfico, el reconocimiento del habla, la predicción de mercados financieros o la visión por ordenador, una de las tecnologías principales en el desarrollo de coches autónomos. El potencial de la inteligencia artificial parece no tener límites. Sin embargo, a pesar de todas sus virtudes, estas tecnologías no se encuentran exentas de polémica, ya que en los últimos años han aparecido aplicaciones que las usan con fines poco éticos, como la creación de contenido multimedia falso (*deepfakes*). Este y otros usos malintencionados de la inteligencia artificial han traído consigo un debate sobre qué consideraciones éticas deberían tenerse en cuenta en el desarrollo de estas tecnologías. A pesar de que muchos de los citados escenarios puedan resultar abrumadores o incluso preocupantes para muchas personas, lo cierto es que cuando la inteligencia artificial se aplica de forma responsable, los beneficios directos en nuestra vida diaria son innegables. En el campo de la medicina, por ejemplo, la inteligencia artificial se usa para detectar patrones en el genoma humano y así desarrollar tratamientos personalizados. Además, el diagnóstico por ordenador permite la detección e intervención temprana de muchas enfermedades, mejorando el pronóstico del paciente. Esto son solo dos ejemplos, ente otros muchos, de cómo la inteligencia artificial aplicada a la medicina tiene un beneficio directo sobre nuestras vidas. Sea como sea, no podemos negar que estas tecnologías han cambiado la forma que tenemos los humanos de interactuar con el mundo.

Naturalmente, estos avances tecnológicos se encuentran presentes en todas las disciplinas científicas, no solo en medicina. Esta tesis está enmarcada en el ámbito de la inteligencia artificial y el aprendizaje máquina aplicados a la Neurociencia Cognitiva. Junto con el desarrollo de potentes técnicas de neuroimágen, la inteligencia artificial ha revolucionado el campo de la Neurociencia en los últimos años, ya que nos ha permitido estudiar y entender mucho más en detalle el funcionamiento de algo tan complejo como el cerebro. Gracias a la inteligencia artificial hemos sido capaces de desarrollar estrategias de análisis que nos permiten detectar cambios muy sutiles en patrones cerebrales, cambios que hasta ahora nos eran indetectables cuando empleábamos métodos de análisis clásicos. Lo cierto es que estos algoritmos de inteligencia artificial se han venido empleando en numerosos ámbitos y disciplinas científicas desde hace años, como las Ciencias de la Computación, por lo que su adaptación al campo de la Neurociencia no parece sorprendente.

Aunque los principios básicos de funcionamiento sean similares, existen algunas consideraciones que merecen la pena resaltar cuando analizamos datos de neuroimagen mediante técnicas de aprendizaje máquina. En primer lugar, cuando estos algoritmos se entrenan para, por ejemplo, detectar correos basura, suelen trabajar con millones de observaciones, teniendo siempre como objetivo lograr el mejor desempeño posible en la tarea para la que fueron entrenados. Este objetivo de intentar lograr siempre el mejor desempeño posible es común a muchas aplicaciones. En Neurociencia Cognitiva, en cambio, esto no siempre ocurre así. La cantidad de datos de los que disponemos en este campo es normalmente muy reducida. Disponer de unos cientos de observaciones es un escenario bastante favorable cuando se trata de datos de neuroimagen, ya que normalmente se suele trabajar con unas decenas de observaciones por clase, lo que hace que dispongamos de más características que observaciones en muchas ocasiones. Esto es debido a que recoger datos cerebrales de participantes humanos (o no humanos) es una tarea que requiere mucho tiempo y dinero. Por otro lado, uno de los objetivos principales de muchos estudios de Neurociencia es detectar cambios minúsculos en patrones cerebrales que difieren de forma significativa entre condiciones y que estos sean generalizables al resto de la población. Además, la Neurociencia Cognitiva se encarga de estudiar cómo estos cambios se relacionan o explican distintas funciones cerebrales. Por lo tanto, obtener el mejor rendimiento posible no siempre es necesario (o incluso deseable). Por este motivo, muchas técnicas para la optimización de los modelos que son ampliamente aplicadas en otros campos (como la optimización de los hiperparámetros) juegan un papel irrelevante en el ámbito de la Neurociencia. Esto, junto al empleo de algoritmos simples, como los clasificadores lineales, suelen proporcionar resultados subóptimos de acuerdo con los estándares de otros campos de la ciencia. Sin embargo, debido al tamaño reducido de los sets de datos, así como a la carga computacional ahorrada, estas configuraciones son preferibles cuando trabajamos con datos de neuroimágen. Además, así garantizamos la fiabilidad, generalización e interpretación de los resultados.

Debido a la reciente explosión de las metodologías de análisis basadas en aprendizaje máquina, a las capacidades y experiencia necesaria para su correcto desarrollo, así como a las anteriormente mencionadas consideraciones cuando se pretende trabajar con datos de neuroimagen, decidimos desarrollar una herramienta a la que hemos llamado *The MVPAlab Toolbox*. Este software desarrollado en MATLAB es una herramienta flexible y fácil de utilizar que encapsula las más recientes y ampliamente extendidas metodologías de análisis en el ámbito de la Neurociencia Cognitiva.

En las primeras fases de desarrollo decidimos mostrar la idoneidad de varios métodos multivariados de aprendizaje máquina en el análisis de datos de neuroimagen, más concretamente, en datos de electroencefalografía. Para entender realmente la naturaleza de un experimento de Neurociencia y todas las fases de su desarrollo, en lugar de apoyarnos en una base de datos pública recopilada por terceros, decidimos comenzar desde el principio. Planificamos, diseñamos y desarrollamos un experimento de control cognitivo. El experimento consistió en la presentación de distintos estímulos visuales a los participantes (flechas con orientaciones congruentes o incongruentes), a los que debían responder lo más rápido posible evitando errores. Estos estímulos, de acuerdo con su dificultad, demandaban distinta carga cognitiva por parte del participante. Tras el reclutamiento de los participantes, recogimos, limpiamos y preprocesamos todos los datos de electroencefalografía para ser analizados. Todo este proceso se encuentra descrito en detalle en el Capítulo 4 de esta tesis. A continuación, procedimos al desarrollo y programación de las técnicas multivariadas de análisis. En un primer acercamiento desarrollamos una serie de rutinas para, entre otras, la extracción y selección de características, la normalización y el suavizado de los registros de electroencefalografía. Posteriormente, desarrollamos e implementamos metodologías para el análisis y reconocimiento de patrones temporales en la señal de electroencefalografía, basados en aprendizaje máquina, más específicamente en máquinas de vectores de soporte lineales. Estos análisis incluían: (1) La clasificación resuelta en el tiempo de los datos de distintas condiciones experimentales, lo que nos permitió estudiar las diferencias en los patrones de actividad cerebral inducidos por los estímulos presentados durante el experimento; (2) la clasificación cruzada resulta en el tiempo nos permitió estudiar los distintos patrones de actividad cerebral inducidos por los estímulos presentados en distintos contextos cognitivos; (3) las matrices de generalización temporal, que nos permitieron estudiar la estabilidad de esos patrones a lo largo del tiempo; y (4) los análisis de contribución de frecuencia, que nos permitieron determinar qué frecuencias de la señal de electroencefalografía contribuían de mayor manera a la correcta clasificación, lo que nos indica que en esas bandas de frecuencia están codificados los mecanismos de procesamiento asociados al estímulo presentado. Además, estudiamos el efecto en la clasificación de distintos parámetros del modelo, como la optimización de los hiperparámetros o el número de ensayos promediados para incrementar la relación señal-ruido. Para ello empleamos distintas medidas de rendimiento como la precisión, la precisión balanceada, o el área bajo la curva. Finalmente, para dar validez estadística a los resultados obtenidos, desarrollamos un análisis de permutaciones no paramétrico, especialmente adaptado a datos de electroencefalografía. Estos análisis revelaron diferencias significativas (p<.001) en los patrones de actividad asociados a estímulos congruentes e incongruentes, lo que nos permitió predecir con una tasa de acierto superior al 80% si los participantes estaban respondiendo a un estímulo congruente o incongruente. Además, estudiamos estos efectos a lo largo del tiempo y las frecuencias, replicando los resultados existentes en la literatura mediante análisis clásicos y proporcionando más información sobre la dinámica de los mecanismos subyacentes.

Una vez demostrada la idoneidad y utilidad de estas metodologías en el ámbito de la electroencefalografía, decidimos ampliar los análisis previamente implementados, dando lugar a la primera versión pública de nuestra herramienta de análisis: *The MVPAlab Toolbox*. El Capítulo 5 de esta tesis doctoral describe en detalle todas las mejoras y novedades, así como una detallada explicación de todos los análisis implementados. Como resumen, implementamos la computación en paralelo, lo que permitió una reducción drástica de los tiempos de cómputo, un aspecto realmente importante cuando trabajamos con cantidades considerables de información. Implementamos, además, un análisis que nos permitió estudiar la contribución de los distintos electrodos a los resultados de la clasificación. Incluimos también más algoritmos de análisis, como los clasificadores no lineales, o el Análisis Discriminante Lineal, implementando nuevas medidas de rendimiento, como la precisión, la sensibilidad, el *f1-score* o la matriz de confusión. Con el objetivo de diferenciar este software de las demás alternativas existentes no solo desarrollamos las distintas funcionalidades, sino que las acompañamos de una interfaz gráfica de usuario intuitiva y fácil de usar, lo que permite a usuarios con poca o nula experiencia de programación diseñar y ejecutar una gran variedad de análisis multivariados para datos de electroencefalografía. Además de todas estas características, en esta primera versión publica *de The MVPAlab Toolbox* desarrollamos una herramienta de visualización de resultados que permite crear gráficas y animaciones, de una forma muy sencilla y listas para su publicación. Todo esto sin la necesidad de escribir una sola línea de código. Esta primera versión completa se encuentra disponible para su descarga en un repositorio público de GitHub (https://github.com/dlopezg/mvpalab) así como una extensa documentación donde explicamos desde su instalación y requisitos previos, al funcionamiento y significado de cada uno de los parámetros de configuración y de las rutinas de análisis implementadas. Así mismo ofrecemos una pequeña base de datos de prueba con la que se hicieron los análisis presentados en el Capítulo 5 de esta tesis (https://osf.io/du6fa).

Todo lo descrito anteriormente hace de esta herramienta una opción sólida y con mucho potencial para analizar señales cerebrales en términos temporales. Sin embargo, la dimensión espacial de la señal aún no la hemos considerado. Para intentar comprender mejor el funcionamiento cerebral, idealmente deberíamos estudiar los procesos cognitivos simultáneamente en el tiempo y en el espacio. Sin embargo, la gran mayoría de técnicas de neuroimagen no invasivas disponibles hoy en día no permiten localizar estos procesos de manera precisa y simultánea. Existen soluciones, como el registro concurrente de señales de resonancia magnética funcional y electroencefalografía, pero estas prácticas, sin entrar en mucho detalle, conllevan graves inconvenientes en lo que a ruido de la señal se refiere, así como importantes restricciones en el diseño del paradigma experimental.

En el Capítulo 6 de esta tesis doctoral abordamos un reto importante en el campo de la Neurociencia Cognitiva, la caracterización de los procesos cognitivos simultáneamente en tiempo y espacio. Para lograr esto, expandimos las capacidades de nuestra herramienta, haciéndola compatible con datos de resonancia magnética funcional. En este caso no se pretendió desarrollar los anteriormente mencionados análisis multivariados y aplicarlos a la señal de fMRI, ya que para ello existen numerosas y potentes herramientas como SPM. En este caso pretendimos integrar en nuestra herramienta una metodología de análisis que nos permitiese fusionar información proveniente de distintas técnicas de neuroimagen, conservando sus fortalezas y mitigando sus debilidades. Concretamente, nuestro objetivo fue fusionar datos de electroencefalografía y resonancia magnética funcional. Para ello, como en el caso anterior, decidimos comenzar desde la base y definir y desarrollar nuestro propio paradigma, así como encargarnos del reclutamiento de participantes, recogida, limpieza y preprocesamiento de los datos. De esta forma, diseñamos dos experimentos equivalentes para el estudio de distintos mecanismos de preparación donde pretendíamos analizar las diferencias entre atención selectiva y expectativa perceptual. Para el diseño de ambos experimentos utilizamos el mismo set de estímulos, pero adaptamos la tarea a los estándares y requerimientos de cada técnica de neuroimagen. Esto fue posible debido a que la recogida de datos mediante ambas técnicas no se realizó de forma simultánea, sino en sesiones distintas y con participantes distintos. La metodología que empleamos para desarrollar los análisis de fusión es conocida como Análisis de Similitud Representacional. Esta permite abstraer los datos del espacio de señal, donde electroencefalografía y resonancia no son directamente comparables debido a su distinta naturaleza, a un espacio de representación en el que sí lo son. Este análisis se fundamenta en el hecho de que los distintos patrones de actividad en cierta región cerebral deben ser similares cuando el estímulo que los genera es similar. De esta forma pudimos generar, tanto para cada punto temporal en los datos de electroencefalografía como para distintas regiones cerebrales en los datos de resonancia magnética, Matrices de Disimilitud Representacional. Estas matrices codifican cuantitativamente las diferencias entre los patrones de actividad cerebral para todo el set de estímulos y, puesto que empleamos el mismo set en ambas modalidades, pudimos correlacionarlas entre sí. Así, extrajimos la matriz de una región cerebral específica y la usamos como semilla para correlacionarla con todas las matrices extraídas para cada punto temporal de la señal de electroencefalografía. Como resultado tras computar varios análisis preliminares, obtuvimos curvas temporales de correlación positivas (p<.001) que nos indicaron cómo de parecidas son las representaciones en distintas regiones específicas (visuales y motoras) a lo largo de todo el intervalo temporal. Dicho de otra forma, gracias a este paradigma pudimos obtener información sobre el comportamiento del cerebro con una alta resolución espacial y temporal. Aunque todos los avances pueden consultarse en el repositorio de MVPAlab, actualmente seguimos desarrollando, mejorando e integrando de forma completa toda esta metodología de análisis en nuestra herramienta. Todas estas mejoras estarán oficialmente disponibles en próximas versiones de The MVPAlab Toolbox.

En resumen, con esta tesis hemos demostrado la viabilidad del empleo de técnicas de análisis multivariadas basadas en reconocimiento de patrones y aprendizaje máquina en el estudio de la función cerebral, especialmente mediante señales de electroencefalografía y resonancia magnética. Como resultado, hemos desarrollado una herramienta completa para el análisis de datos de neuroimagen, *The MVPAlab Toolbox*, la cual integra las más novedosas metodologías de análisis empleadas en el ámbito de la Neurociencia Cognitiva. Esta herramienta ayudará a investigadores de cualquier nivel de experiencia o conocimientos de programación a impulsar sus estudios mediante el empleo de la inteligencia artificial, perimiéndoles extraer información más detallada de sus datos.

References

Abdi, H., & Williams, L. J. (2010). Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, *2*(4), 433–459. https://doi.org/10.1002/wics.101

Aboalayon, K. a. I., Ocbagabir, H. T., & Faezipour, M. (2014). Efficient sleep stage classification based on EEG signals. *IEEE Long Island Systems, Applications and Technology (LISAT) Conference 2014*, 1–6. https://doi.org/10.1109/LISAT.2014.6845193

Abraham, A., Pedregosa, F., Eickenberg, M., Gervais, P., Mueller, A., Kossaifi, J., Gramfort, A., Thirion, B., & Varoquaux, G. (2014). Machine learning for neuroimaging with scikit-learn. *Frontiers in Neuroinformatics*, *8*(FEB), 1–10. https://doi.org/10.3389/fninf.2014.00014

Ahmadi Rastegar, D., Ho, N., Halliday, G. M., & Dzamko, N. (2019). Parkinson's progression prediction using machine learning and serum cytokines. *Npj Parkinson's Disease*, *5*(1), 1–8. https://doi.org/10.1038/s41531-019-0086-4

Ahmadlou, M., & Adeli, H. (2010). Enhanced probabilistic neural network with local decision circles: A robust classifier. *Integrated Computer-Aided Engineering*, *17*(3), 197–210. https://doi.org/10.3233/ICA-2010-0345

Aitken, A. C. (1936). IV.—On Least Squares and Linear Combination of Observations. *Proceedings of the Royal Society of Edinburgh*, *55*, 42–48. https://doi.org/10.1017/S0370164600014346

Álvarez, D., Cerezo-Hernández, A., Crespo, A., Gutiérrez-Tobal, G. C., Vaquerizo-Villar, F., Barroso-García, V., Moreno, F., Arroyo, C. A., Ruiz, T., Hornero, R., & del Campo, F. (2020). A machine learning-based test for adult sleep apnoea screening at home using oximetry and airflow. *Scientific Reports*, *10*(1), 1–12. https://doi.org/10.1038/s41598-020-62223-4

Álvarez, I., Górriz, J. M., Ramírez, J., Salas-Gonzalez, D., López, M., Puntonet, C. G., & Segovia, F. (2009). Alzheimer's diagnosis using eigenbrains and support vector machines. *Electronics Letters*, *45*(7), 342. https://doi.org/10.1049/el.2009.3415

Arco, J. E., González-García, C., Díaz-Gutiérrez, P., Ramírez, J., & Ruz, M. (2018). Influence of activation pattern estimates and statistical significance tests in fMRI decoding analysis. *Journal of Neuroscience Methods*, *308*, 248–260. https://doi.org/10.1016/J.JNEUMETH.2018.06.017

Arco, J. E., Ortiz, A., Ramírez, J., Martínez-Murcia, F. J., Zhang, Y.-D., Broncano, J., Berbís, M. Á., Royuela-del-Val, J., Luna, A., & Górriz, J. M. (2021). *Probabilistic combination of eigenlungs-based classifiers for COVID-19 diagnosis in chest CT images*. http://arxiv.org/abs/2103.02961

Ashton, K., Zinszer, B. D., Cichy, R. M., Nelson, C. A., Aslin, R. N., & Bayet, L. (2022). Time-resolved multivariate pattern analysis of infant EEG data: A practical tutorial. *Developmental Cognitive Neuroscience*, *54*, 101094. https://doi.org/https://doi.org/10.1016/j.dcn.2022.101094

Baldauf, D., & Desimone, R. (2014). Neural Mechanisms of Object-Based Attention. *Science*, *344*(6182), 424–427. https://doi.org/10.1126/science.1247003

Başar, E., Başar-Eroğlu, C., Karakaş, S., & Schürmann, M. (2000). Brain oscillations in perception and memory. *International Journal of Psychophysiology*, *35*(2–3), 95–124. https://doi.org/10.1016/S0167-8760(99)00047-1

Başar, Erol, Gölbaşı, B. T., Tülay, E., Aydın, S., & Başar-Eroğlu, C. (2016). Best method for analysis of brain oscillations in healthy subjects and neuropsychiatric diseases. *International Journal of Psychophysiology*, *103*, 22–42. https://doi.org/https://doi.org/10.1016/j.ijpsycho.2015.02.017

Beall, E. B., & Lowe, M. J. (2014). SimPACE: Generating simulated motion corrupted BOLD data with synthetic-navigated acquisition for the development and evaluation of SLOMOCO: A new, highly effective slicewise motion correction. *NeuroImage*, *101*, 21–34. https://doi.org/10.1016/j.neuroimage.2014.06.038

Begleiter, H., & Porjesz, B. (2006). Genetics of human brain oscillations. *International Journal of Psychophysiology*, *60*(2), 162–171.

Belouchrani, A, Abed-Meraim, K., Cardoso, J.-F., & Moulines, E. (1997). A blind source separation technique using second-order statistics. *IEEE Transactions on Signal Processing*, *45*(2), 434–444. https://doi.org/10.1109/78.554307

Belouchrani, Adel, Abed-Meraim, K., Cardoso, J. F., & Moulines, E. (1993). Second-order blind separation of temporally correlated sources. *Proc. Int. Conf. Digital Signal Processing*, 346–351.

Benítez-Burraco, A., & Murphy, E. (2019). Why Brain Oscillations Are Improving Our Understanding of Language. *Frontiers in Behavioral Neuroscience*, *13*. https://doi.org/10.3389/fnbeh.2019.00190

Benjamini, Y., & Hochberg, Y. (1995). Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, *57*(1), 289–300. https://doi.org/10.1111/j.2517-6161.1995.tb02031.x

Benjamini, Y., & Liu, W. (1999). A step-down multiple hypotheses testing procedure that controls the false discovery rate under independence. *Journal of Statistical Planning and Inference*, *82*(1–2), 163–170. https://doi.org/10.1016/S0378-3758(99)00040-3

Bennett, C. M., Wolford, G. L., & Miller, M. B. (2009). The principled control of false positives in neuroimaging. *Social Cognitive and Affective Neuroscience*, *4*(4), 417–422. https://doi.org/10.1093/scan/nsp053

Berberyan, H. S., van Maanen, L., van Rijn, H., & Borst, J. (2021). EEG-based Identification of Evidence Accumulation Stages in Decision-Making. *Journal of Cognitive Neuroscience*, *33*(3), 510–527. https://doi.org/10.1162/jocn_a_01663

Bernert, M., & Yvert, B. (2019). An Attention-Based Spiking Neural Network for Unsupervised Spike-Sorting. *International Journal of Neural Systems*, *29*(08). https://doi.org/10.1142/S0129065718500594

Bhagalia, R., & Kim, B. (2008). Spin saturation artifact correction using slice-to-volume registration motion estimates for fMRI time series. *Medical Physics*, *35*(2), 424–434.

https://doi.org/10.1118/1.2826555

Bigdely-Shamlo, N., Mullen, T., Kothe, C., Su, K.-M., & Robbins, K. A. (2015). The PREP pipeline: standardized preprocessing for large-scale EEG analysis. *Frontiers in Neuroinformatics*, *9*. https://doi.org/10.3389/fninf.2015.00016

Bishop, C. M., & Nasrabadi, N. M. (2006). *Pattern recognition and machine learning* (Vol. 4, Issue 4). Springer.

Blankertz, B., Acqualagna, L., Dähne, S., Haufe, S., Schultze-Kraft, M., Sturm, I., Ušcumlic, M., Wenzel, M. A., Curio, G., & Müller, K. R. (2016). The Berlin brain-computer interface: Progress beyond communication and control. *Frontiers in Neuroscience*, *10*(NOV). https://doi.org/10.3389/fnins.2016.00530

Bloch, F. (1946). Nuclear induction. *Physical Review*, *70*(7–8), 460.

Bode, S., Feuerriegel, D., Bennett, D., & Alday, P. M. (2019). The Decision Decoding ToolBOX (DDTBOX) – A Multivariate Pattern Analysis Toolbox for Event-Related Potentials. *Neuroinformatics*, *17*(1), 27–42. https://doi.org/10.1007/s12021-018-9375-z

Bode, S., He, A. H., Soon, C. S., Trampel, R., Turner, R., & Haynes, J.-D. (2011). Tracking the Unconscious Generation of Free Decisions Using UItra-High Field fMRI. *PLOS ONE*, *6*(6), 1–13. https://doi.org/10.1371/journal.pone.0021612

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. *Proceedings of the Fifth Annual Workshop on Computational Learning Theory - COLT '92*, 144–152. https://doi.org/10.1145/130385.130401

Botvinick, M. M., Carter, C. S., Braver, T. S., Barch, D. M., & Cohen, J. D. (2001). Conflict monitoring and cognitive control. *Psychological Review*, *108*(3), 624–652. https://doi.org/10.1037/0033-295X.108.3.624

Braga-Neto, U., Hashimoto, R., Dougherty, E. R., Nguyen, D. V., & Carroll, R. J. (2004). Is cross-validation better than resubstitution for ranking genes? *Bioinformatics*, *20*(2), 253–258. https://doi.org/10.1093/bioinformatics/btg399

Bragin, A., Wilson, C. L., Staba, R. J., Reddick, M., Fried, I., & Engel Jr., J. (2002). Interictal high-frequency oscillations (80–500Hz) in the human epileptic brain: Entorhinal cortex. *Annals of Neurology*, *52*(4), 407–415. https://doi.org/https://doi.org/10.1002/ana.10291

Brett, M., Christoff, K., Cusack, R., Lancaster, J., & others. (2001). Using the Talairach atlas with the MNI template. *Neuroimage*, *13*(6), 85.

Brodersen, K. H., Ong, C. S., Stephan, K. E., & Buhmann, J. M. (2010). The balanced accuracy and its posterior distribution. *Proceedings - International Conference on Pattern Recognition*, 3121–3124. https://doi.org/10.1109/ICPR.2010.764

Bueno, F. D., & Cravo, A. M. (2021). Post-interval EEG activity is related to task-goals in temporal discrimination. *PLoS ONE*, *16*(9 September), 1–19. https://doi.org/10.1371/journal.pone.0257378

Buxton, R. B., Wong, E. C., & Frank, L. R. (1998). Dynamics of blood flow and oxygenation changes during brain activation: the balloon model. *Magnetic Resonance in Medicine*, *39*(6), 855–864.

Buzsáki, G. (2006). *Rhythms of the Brain*. Oxford university press.

Buzsáki, G., Anastassiou, C. A., & Koch, C. (2012). The origin of extracellular fields and currents-EEG, ECoG, LFP and spikes. *Nature Reviews Neuroscience*, *13*(6), 407–420. https://doi.org/10.1038/nrn3241

Buzsáki, G., & Wang, X.-J. (2012). Mechanisms of gamma oscillations. *Annual Review of Neuroscience*, *35*, 203–225. https://doi.org/10.1146/annurev-neuro-062111-150444

Buzsáki, G., & Watson, B. O. (2012). Brain rhythms and neural syntax: implications for efficient coding of cognitive content and neuropsychiatric disease. *Dialogues in Clinical Neuroscience*, *14*(4), 345–367. https://doi.org/10.31887/DCNS.2012.14.4/gbuzsaki

Carlson, T. A., Grootswagers, T., & Robinson, A. K. (2019). *An introduction to time-resolved decoding analysis for M/EEG*. arXiv. https://doi.org/10.48550/ARXIV.1905.04820

Cauchoix, M., Barragan-Jason, G., Serre, T., & Barbeau, E. J. (2014). The Neural Dynamics of Face Detection in the Wild Revealed by MVPA. *The Journal of Neuroscience*, *34*(3), 846–854. https://doi.org/10.1523/JNEUROSCI.3030-13.2014

Cawley, G. C., & Talbot, N. L. C. (2010). On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, *11*, 2079–2107.

Chadwick, M. J., Hassabis, D., Weiskopf, N., & Maguire, E. A. (2010). Decoding Individual Episodic Memory Traces in the Human Hippocampus. *Current Biology*, *20*(6), 544–547. https://doi.org/https://doi.org/10.1016/j.cub.2010.01.053

Chan, D., Suk, H.-J., Jackson, B., Milman, N. P., Stark, D., Beach, S. D., & Tsai, L.-H. (2021). Induction of specific brain oscillations may restore neural circuits and be used for the treatment of Alzheimer's disease. *Journal of Internal Medicine*, *290*(5), 993–1009. https://doi.org/https://doi.org/10.1111/joim.13329

Chaumon, M., Bishop, D. V. M., & Busch, N. A. (2015). A practical guide to the selection of independent components of the electroencephalogram for artifact correction. *Journal of Neuroscience Methods*, *250*, 47–63. https://doi.org/10.1016/j.jneumeth.2015.02.025

Chaves, R., Ramírez, J., Górriz, J. M., López, M., Salas-Gonzalez, D., Álvarez, I., & Segovia, F. (2009). SVM-based computer-aided diagnosis of the Alzheimer's disease using t-test NMSE feature selection with feature correlation weighting. *Neuroscience Letters*, *461*(3), 293–297. https://doi.org/10.1016/J.NEULET.2009.06.052

Choi, S. (2004). *Blind Source Separation and Independent Component Analysis : A Review*.

Cichy, Radoslaw M., & Oliva, A. (2020). A M/EEG-fMRI Fusion Primer: Resolving Human Brain Responses in Space and Time. *Neuron*, *107*(5), 772–781. https://doi.org/10.1016/j.neuron.2020.07.001

Cichy, Radoslaw Martin, Pantazis, D., & Oliva, A. (2014). Resolving human object recognition in space and time. *Nature Neuroscience*, *17*(3), 455–462. https://doi.org/10.1038/nn.3635

Clark, J. W. (1998). The origin of biopotentials. *Medical Instrumentation: Application and Design*, *3*, 121–182.

Cohen, M. X. (2016). *Cycles in the mind. How brain rhythms control perception and action*. Sinc(X) Press.

Cohen, M. X., & Donner, T. H. (2013). Midfrontal conflict-related theta-band power reflects neural oscillations that predict behavior. *Journal of Neurophysiology*, *110*(12), 2752–2763. https://doi.org/10.1152/jn.00479.2013

Combrisson, E., & Jerbi, K. (2015). Exceeding chance level by chance: The caveat of theoretical chance levels in brain signal classification and statistical assessment of decoding accuracy. *Journal of Neuroscience Methods*, *250*, 126–136. https://doi.org/10.1016/j.jneumeth.2015.01.010

Comon, P. (1994). Independent component analysis, A new concept? *Signal Processing*, *36*(3), 287–314. https://doi.org/https://doi.org/10.1016/0165-1684(94)90029-9

Correia, J. M., Jansma, B., Hausfeld, L., Kikkert, S., & Bonte, M. (2015). EEG decoding of spoken words in bilingual listeners: From words to language invariant semantic-conceptual representations. *Frontiers in Psychology*, *6*, 71. https://doi.org/10.3389/fpsyg.2015.00071

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. https://doi.org/10.1007/BF00994018

Cristianini, N., Shawe-Taylor, J., & others. (2000). *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press.

Crowley, K. (2011). Sleep and Sleep Disorders in Older Adults. *Neuropsychology Review*, *21*(1), 41–53. https://doi.org/10.1007/s11065-010-9154-6

Davis, H., Davis, P. A., Loomis, A. L., Harvey, E. N., & Hobart, G. (1939). ELECTRICAL REACTIONS OF THE HUMAN BRAIN TO AUDITORY STIMULATION DURING SLEEP. *Journal of Neurophysiology*, *2*(6), 500–514. https://doi.org/10.1152/jn.1939.2.6.500

Davis, T., & Poldrack, R. A. (2013). Measuring neural representations with fMRI: Practices and pitfalls. *Annals of the New York Academy of Sciences*, *1296*(1), 108–134. https://doi.org/10.1111/nyas.12156

de Cheveigné, A., & Nelken, I. (2019). Filters: When, Why, and How (Not) to Use Them. *Neuron*, *102*(2), 280–293. https://doi.org/10.1016/j.neuron.2019.02.039

De Lange, F., Jensen, O., Bauer, M., & Toni, I. (2008). Interactions between posterior gamma and frontal alpha/beta oscillations during imagined actions. *Frontiers in Human Neuroscience*, *2*. https://doi.org/10.3389/neuro.09.007.2008

Deichmann, R., Josephs, O., Hutton, C., Corfield, D. R., & Turner, R. (2002). Compensation of susceptibility-induced bold sensitivity losses in echo-planar fMRI imaging. *NeuroImage*, *15*(1), 120–135. https://doi.org/10.1006/nimg.2001.0985

Delorme, A., & Makeig, S. (2004). EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *Journal of Neuroscience Methods*, *134*(1), 9–21.

Díaz-Gutiérrez, P., Arco, J. E., Alguacil, S., González-García, C., & Ruz, M. (2018). Neural representation of social expectations during interpersonal decisions. *BioRxiv*. https://doi.org/10.1101/355115

Duda, M., Ma, R., Haber, N., & Wall, D. P. (2016). Use of machine learning for behavioral distinction of autism and ADHD. *Translational Psychiatry*, *6*(2), 1–5. https://doi.org/10.1038/tp.2015.221

Düzel, E., Penny, W. D., & Burgess, N. (2010). Brain oscillations and memory. *Current Opinion in Neurobiology*, *20*(2), 143–149. https://doi.org/https://doi.org/10.1016/j.conb.2010.01.004

Eklund, A., Nichols, T. E., & Knutsson, H. (2016). Cluster failure: Why fMRI inferences for spatial extent have inflated false-positive rates. *Proceedings of the National Academy of Sciences*, *113*(28), 7900–7905. https://doi.org/10.1073/pnas.1602413113

Elston, G. N. (2003). Cortex, Cognition and the Cell: New Insights into the Pyramidal Neuron and Prefrontal Function. *Cerebral Cortex*, *13*(11), 1124–1138. https://doi.org/10.1093/cercor/bhg093

Eriksen, B. A., & Eriksen, C. W. (1974). Effects of noise letters upon the identification of a target letter in a nonsearch task. *Perception & Psychophysics*, *16*(1), 143–149. https://doi.org/10.3758/BF03203267

Ethofer, T., Van De Ville, D., Scherer, K., & Vuilleumier, P. (2009). Decoding of Emotional Information in Voice-Sensitive Cortices. *Current Biology*, *19*(12), 1028–1033. https://doi.org/https://doi.org/10.1016/j.cub.2009.04.054

Etzel, J. A., Gazzola, V., & Keysers, C. (2008). Testing Simulation Theory with Cross-Modal Multivariate Classification of fMRI Data. *PLoS ONE*, *3*(11), e3690. https://doi.org/10.1371/journal.pone.0003690

Etzel, J. A., Valchev, N., & Keysers, C. (2011). The impact of certain methodological choices on multivariate analysis of fMRI data with support vector machines. *NeuroImage*, *54*(2), 1159–1167. https://doi.org/10.1016/J.NEUROIMAGE.2010.08.050

Evans, A. C., Collins, D. L., Mills, S. R., Brown, E. D., Kelly, R. L., & Peters, T. M. (1993). 3D statistical neuroanatomical models from 305 MRI volumes. *1993 IEEE Conference Record Nuclear Science Symposium and Medical Imaging Conference*, 1813–1817.

Fahrenfort, J. J., van Driel, J., van Gaal, S., & Olivers, C. N. L. (2018). From ERPs to MVPA using the Amsterdam Decoding and Modeling toolbox (ADAM). *Frontiers in Neuroscience*, *12*(JUL). https://doi.org/10.3389/fnins.2018.00368

Fell, J., & Axmacher, N. (2011). The role of phase synchronization in memory processes. *Nature Reviews Neuroscience*, *12*(2), 105–118. https://doi.org/10.1038/nrn2979

Foffani, G., & Alegre, M. (2022). Brain oscillations and Parkinson disease. In A. Quartarone, M. F. Ghilardi, & F. Boller (Eds.), *Neuroplasticity* (Vol. 184, pp. 259–271). Elsevier. https://doi.org/10.1016/B978-0-12-819410-2.00014-X

Fries, P. (2005). A mechanism for cognitive dynamics: Neuronal communication through neuronal coherence. *Trends in Cognitive Sciences*, *9*(10), 474–480. https://doi.org/10.1016/j.tics.2005.08.011

Friston, K. J., Holmes, A. P., Worsley, K. J., Poline, J. -P, Frith, C. D., & Frackowiak, R. S. J. (1994). Statistical parametric maps in functional imaging: A general linear approach. *Human Brain Mapping*, *2*(4), 189–210. https://doi.org/10.1002/hbm.460020402

Gao, C., Sun, H., Wang, T., Tang, M., Bohnen, N. I., Müller, M. L. T. M., Herman, T., Giladi, N., Kalinin, A., Spino, C., Dauer, W., Hausdorff, J. M., & Dinov, I. D. (2018). Model-based and model-free machine learning techniques for diagnostic prediction and classification of clinical outcomes in Parkinson's disease. *Scientific Reports*, *8*(1), 1–21. https://doi.org/10.1038/s41598-018-24783-4

García-López, P., García-Marín, V., & Freire, M. (2006). Three-dimensional reconstruction and quantitative study of a pyramidal cell of a Cajal histological preparation. *Journal of Neuroscience*, *26*(44), 11249–11252. https://doi.org/10.1523/JNEUROSCI.3543-06.2006

Ghosh-Dastidar, S., & Adeli, H. (2007). Improved spiking neural networks for EEG classification and epilepsy and seizure detection. *Integrated Computer-Aided Engineering*, *14*(3), 187–212. https://doi.org/10.3233/ICA-2007-14301

Ghosh-Dastidar, S., & Adeli, H. (2009). A new supervised learning algorithm for multiple spiking neural networks with application in epilepsy and seizure detection. *Neural Networks*, *22*(10), 1419–1431. https://doi.org/10.1016/j.neunet.2009.04.003

Gibbons, L. P. (n.d.). *InpOut32 is an open source windows DLL and Driver to give direct access to hardware ports*. Retrieved February 24, 2019, from http://www.highrez.co.uk/Downloads/InpOut32/default.htm

Goebel, R., Esposito, F., & Formisano, E. (2006). Analysis of Functional Image Analysis Contest (FIAC) data with BrainVoyager QX: From single-subject to cortically aligned group General Linear Model analysis and self-organizing group Independent Component Analysis. *Human Brain Mapping*, *27*(5), 392–401. https://doi.org/10.1002/hbm.20249

Gomez, A., Rothkirch, M., Kaul, C., Weygandt, M., Haynes, J.-D., Rees, G., & Sterzer, P. (2011). Emotion modulates the effects of endogenous attention on retinotopic visual processing. *NeuroImage*, *57*(4), 1542–1551. https://doi.org/https://doi.org/10.1016/j.neuroimage.2011.05.072

Gorgolewski, K. J., Auer, T., Calhoun, V. D., Craddock, R. C., Das, S., Duff, E. P., Flandin, G., Ghosh, S. S., Glatard, T., Halchenko, Y. O., Handwerker, D. A., Hanke, M., Keator, D., Li, X., Michael, Z., Maumet, C., Nichols, B. N., Nichols, T. E., Pellman, J., … Poldrack, R. A. (2016). The brain imaging data structure, a format for organizing and describing outputs of neuroimaging experiments. *Scientific Data*, *3*(1), 160044. https://doi.org/10.1038/sdata.2016.44

Gorriz, J. M., Group, S., & Neuroscience, C. (2019). Statistical Agnostic Mapping: a Framework in Neuroimaging based on Concentration Inequalities. *BioRxiv*. http://arxiv.org/abs/1912.12274

Gorriz, J. M., Jimenez-Mesa, C., Romero-Garcia, R., Segovia, F., Ramirez, J., Castillo-Barnes, D., Martinez-Murcia, F. J., Ortiz, A., Salas-Gonzalez, D., Illan, I. A., Puntonet, C. G., Lopez-Garcia, D., Gomez-Rio, M., & Suckling, J. (2021). Statistical Agnostic Mapping: A framework in neuroimaging based on concentration inequalities. *Information Fusion*, *66*, 198–212. https://doi.org/10.1016/j.inffus.2020.09.008

Górriz, J. M., Ramírez, J., Ortíz, A., Martínez-Murcia, F. J., Segovia, F., Suckling, J., Leming, M., Zhang, Y. D., Álvarez-Sánchez, J. R., Bologna, G., Bonomini, P., Casado, F. E., Charte, D., Charte, F., Contreras, R., Cuesta-Infante, A., Duro, R. J., Fernández-Caballero, A., Fernández-Jover, E., … Ferrández, J. M. (2020). Artificial intelligence within the interplay between natural and artificial computation: Advances in data science, trends and applications. *Neurocomputing*, *410*(May), 237–270. https://doi.org/10.1016/j.neucom.2020.05.078

Górriz, J. M., Ramírez, J., Segovia, F., Martínez, F. J., Lai, M. C., Lombardo, M. V., Baron-Cohen, S., & Suckling, J. (2019a). A machine learning approach to reveal the neurophenotypes of autisms. *International Journal of Neural Systems*, *29*(7), 1–22. https://doi.org/10.1142/S0129065718500582

Górriz, J. M., Ramírez, J., Segovia, F., Martínez, F. J., Lai, M. C., Lombardo, M. V., Baron-Cohen, S., & Suckling, J. (2019b). A Machine Learning Approach to Reveal the NeuroPhenotypes of Autisms. *International Journal of Neural Systems*, *29*(7), 1–22. https://doi.org/10.1142/S0129065718500582

Górriz, J. M., Ramirez, J., & Suckling, J. (2019). On the computation of distribution-free performance bounds: Application to small sample sizes in neuroimaging. *Pattern Recognition*, *93*, 1–13. https://doi.org/10.1016/j.patcog.2019.03.032

Gramfort, A., Luessi, M., Larson, E., Engemann, D. A., Strohmeier, D., Brodbeck, C., Goj, R., Jas, M., Brooks, T., Parkkonen, L., & Hämäläinen, M. (2013). MEG and EEG data analysis with MNE-Python. *Frontiers in Neuroscience*, *7*(7 DEC), 1–13. https://doi.org/10.3389/fnins.2013.00267

Grech, R., Cassar, T., Muscat, J., Camilleri, K. P., Fabri, S. G., Zervakis, M., Xanthopoulos, P., Sakkalis, V., & Vanrumste, B. (2008). Review on solving the inverse problem in EEG source analysis. *Journal of NeuroEngineering and Rehabilitation*, *5*(1), 25. https://doi.org/10.1186/1743-0003-5-25

Grootswagers, T., Wardle, S. G., & Carlson, T. A. (2017). Decoding Dynamic Brain Patterns from Evoked Responses: A Tutorial on Multivariate Pattern Analysis Applied to Time Series Neuroimaging Data. *Journal of Cognitive Neuroscience*, *29*(4), 677–697. https://doi.org/10.1162/jocn_a_01068

Gross, J., Baillet, S., Barnes, G. R., Henson, R. N., Hillebrand, A., Jensen, O., Jerbi, K., Litvak, V., Maess, B., Oostenveld, R., Parkkonen, L., Taylor, J. R., van Wassenhove, V., Wibral, M., & Schoffelen, J. M. (2013). Good practice for conducting and reporting MEG research. *NeuroImage*, *65*, 349–363. https://doi.org/10.1016/j.neuroimage.2012.10.001

Haller, S., & Bartsch, A. J. (2009). Pitfalls in fMRI. *European Radiology*, *19*(11), 2689–2706. https://doi.org/10.1007/s00330-009-1456-9

Hanke, M., Halchenko, Y. O., Sederberg, P. B., Hanson, S. J., Haxby, J. V., & Pollmann, S. (2009). PyMVPA: A python toolbox for multivariate pattern analysis of fMRI data. *Neuroinformatics*, *7*(1), 37–53. https://doi.org/10.1007/s12021-008-9041-y

Hanke, M., Halchenko, Y. O., Sederberg, P. B., Olivetti, E., Fründ, I., Rieger, J. W., Herrmann, C. S., Haxby, J. V., Hanson, S. J., & Pollmann, S. (2009). PyMVPA: A unifying approach to the analysis of neuroscientifi c data. *Frontiers in Neuroinformatics*, *3*(FEB), 1–13. https://doi.org/10.3389/neuro.11.003.2009

Hanslmayr, S., Axmacher, N., & Inman, C. S. (2019). Modulating Human Memory via Entrainment of Brain Oscillations. *Trends in Neurosciences*, *42*(7), 485–499. https://doi.org/https://doi.org/10.1016/j.tins.2019.04.004

Hanslmayr, S., & Staudigl, T. (2014). How brain oscillations form memories — A processing based perspective on oscillatory subsequent memory effects. *NeuroImage*, *85*, 648–655. https://doi.org/10.1016/j.neuroimage.2013.05.121

Hanson, J. E., Ma, K., Elstrott, J., Weber, M., Saillet, S., Khan, A. S., Simms, J., Liu, B., Kim, T. A., Yu, G.-Q., Chen, Y., Wang, T.-M., Jiang, Z., Liederer, B. M., Deshmukh, G., Solanoy, H., Chan, C., Sellers, B. D., Volgraf, M., … Palop, J. J. (2020). GluN2A NMDA Receptor Enhancement Improves Brain Oscillations, Synchrony, and Cognitive Functions in Dravet Syndrome and Alzheimer's Disease Models. *Cell Reports*, *30*(2), 381-396.e4. https://doi.org/https://doi.org/10.1016/j.celrep.2019.12.030

Harmony, T. (2013). The functional significance of delta oscillations in cognitive processing. *Frontiers in Integrative Neuroscience*, *7*, 1–10. https://doi.org/10.3389/fnint.2013.00083

Harper, J., Malone, S. M., & Bernat, E. M. (2014). Theta and delta band activity explain N2 and P3 ERP component activity in a go/no-go task. *Clinical Neurophysiology*, *125*(1), 124–132. https://doi.org/10.1016/j.clinph.2013.06.025

Haufe, S., Meinecke, F., Görgen, K., Dähne, S., Haynes, J. D., Blankertz, B., & Bießmann, F. (2014). On the interpretation of weight vectors of linear models in multivariate neuroimaging. *NeuroImage*, *87*, 96–110. https://doi.org/10.1016/j.neuroimage.2013.10.067

Haxby, J. V. (2001). Distributed and Overlapping Representations of Faces and Objects in Ventral Temporal Cortex. *Science*, *293*(5539), 2425–2430. https://doi.org/10.1126/science.1063736

Haxby, J. V. (2012). Multivariate pattern analysis of fMRI: The early beginnings. *NeuroImage*, *62*(2), 852–855. https://doi.org/10.1016/j.neuroimage.2012.03.016

Haxby, J. V., Connolly, A. C., & Guntupalli, J. S. (2014). Decoding Neural Representational Spaces Using Multivariate Pattern Analysis. *Annual Review of Neuroscience*, *37*(1), 435–456. https://doi.org/10.1146/annurev-neuro-062012-170325

Haynes, J.-D. (2011). Decoding and predicting intentions. *Annals of the New York Academy of Sciences*, *1224*(1), 9–21. https://doi.org/https://doi.org/10.1111/j.1749-6632.2011.05994.x

Haynes, J.-D., & Rees, G. (2006). Decoding mental states from brain activity in humans. *Nature Reviews Neuroscience*, *7*(7), 523–534. https://doi.org/10.1038/nrn1931

Hebart, M. N., Bankson, B. B., Harel, A., Baker, C. I., & Cichy, R. M. (2018a). The representational dynamics of task and object processing in humans. *ELife*, *7*, e32816. https://doi.org/10.7554/eLife.32816

Hebart, M. N., Bankson, B. B., Harel, A., Baker, C. I., & Cichy, R. M. (2018b). The representational dynamics of task and object processing in humans. *ELife*, *7*, 1–21. https://doi.org/10.7554/eLife.32816

Hebart, M. N., GÃ¶rgen, K., & Haynes, J.-D. (2015). The Decoding Toolbox (TDT): a versatile software package for multivariate analyses of functional imaging data. *Frontiers in Neuroinformatics*, *8*, 88. https://doi.org/10.3389/fninf.2014.00088

Hebart, M. N., Schriever, Y., Donner, T. H., & Haynes, J.-D. (2016). The Relationship between Perceptual Decision Variables and Confidence in the Human Brain. *Cerebral Cortex*, *26*(1), 118–130. https://doi.org/10.1093/cercor/bhu181

Hendriks, M. H. A., Daniels, N., Pegado, F., & de Beeck, H. P. O. (2017). The effect of spatial smoothing on representational similarity in a simple motor paradigm. *Frontiers in Neurology*, *8*(MAY), 1–11. https://doi.org/10.3389/fneur.2017.00222

Herculano-Houzel, S. (2009). The human brain in numbers: A linearly scaled-up primate brain. *Frontiers in Human Neuroscience*, *3*(NOV), 1–11. https://doi.org/10.3389/neuro.09.031.2009

Herrmann, C. S., & Demiralp, T. (2005). Human EEG gamma oscillations in neuropsychiatric disorders. *Clinical Neurophysiology*, *116*(12), 2719–2733. https://doi.org/https://doi.org/10.1016/j.clinph.2005.07.007

Hosseini, M., Powell, M., Collins, J., Callahan-Flintoft, C., Jones, W., Bowman, H., & Wyble, B. (2020). I tried a bunch of things: The dangers of unexpected overfitting in classification of brain data. *Neuroscience and Biobehavioral Reviews*, *119*(September), 456–467. https://doi.org/10.1016/j.neubiorev.2020.09.036

Hu, R., Huang, Q., Wang, H., He, J., & Chang, S. (2019). Monitor-Based Spiking Recurrent Network for the Representation of Complex Dynamic Patterns. *International Journal of Neural Systems*, *29*(08), 1950006. https://doi.org/10.1142/S0129065719500060

Hu, X., & Yacoub, E. (2012). The story of the initial dip in fMRI. *NeuroImage*, *62*(2), 1103–1108. https://doi.org/10.1016/j.neuroimage.2012.03.005

Hubbard, J., Kikumoto, A., & Mayr, U. (2019). EEG Decoding Reveals the Strength and Temporal Dynamics of Goal-Relevant Representations. *Scientific Reports*, *9*(1), 9051. https://doi.org/10.1038/s41598-019-45333-6

Huettel, S. A., Song, A. W., McCarthy, G., & others. (2004). *Functional magnetic resonance imaging* (Vol. 1). Sinauer Associates Sunderland.

Hurk, J. van den, & Beeck, H. P. Op de. (2019). Generalization asymmetry in multivariate cross-classification: When representation A generalizes better to representation B than B to A. *BioRxiv*. https://doi.org/10.1101/592410

Ifeachor, E. C., & Jervis, B. W. (2002). *Digital signal processing: a practical approach*. Pearson Education.

Isik, L., Meyers, E. M., Leibo, J. Z., & Poggio, T. (2014a). The dynamics of invariant object recognition in the human visual system. *Journal of Neurophysiology*, *111*(1), 91–102. https://doi.org/10.1152/jn.00394.2013

Isik, L., Meyers, E. M., Leibo, J. Z., & Poggio, T. (2014b). The dynamics of invariant object recognition in the human visual system. *Journal of Neurophysiology*, *111*(1), 91–102. https://doi.org/10.1152/jn.00394.2013

Islam, M. K., Rastegarnia, A., & Yang, Z. (2016). Methods for artifact detection and removal from scalp EEG: A review. *Neurophysiologie Clinique = Clinical Neurophysiology*, *46*(4–5), 287–305. https://doi.org/10.1016/j.neucli.2016.07.002

Jafari, Z., Kolb, B. E., & Mohajerani, M. H. (2020). Neural oscillations and brain stimulation in Alzheimer's disease. *Progress in Neurobiology*, *194*, 101878. https://doi.org/https://doi.org/10.1016/j.pneurobio.2020.101878

Jasper, H. H. (1958). Report of the committee on methods of clinical examination in electroencephalography. *Electroencephalography and Clinical Neurophysiology*, *10*(2), 370–375. https://doi.org/10.1016/0013-4694(58)90053-1

Jenkinson, M., Beckmann, C. F., Behrens, T. E. J., Woolrich, M. W., & Smith, S. M. (2012). FSL. *NeuroImage*, *62*(2), 782–790. https://doi.org/10.1016/j.neuroimage.2011.09.015

Johnson, J. D., McDuff, S. G. R., Rugg, M. D., & Norman, K. A. (2009). Recollection, Familiarity, and Cortical Reinstatement: A Multivoxel Pattern Analysis. *Neuron*, *63*(5), 697–708. https://doi.org/10.1016/j.neuron.2009.08.011

Johnstone, T., Ores Walsh, K. S., Greischar, L. L., Alexander, A. L., Fox, A. S., Davidson, R. J., & Oakes, T. R. (2006). Motion correction and the use of motion covariates in multiple-subject fMRI analysis. *Human Brain Mapping*, *27*(10), 779–788. https://doi.org/10.1002/hbm.20219

Jung, R., & Berger, W. (1979). Hans Bergers Entdeckung des Elektrenkephalogramms und seine ersten Befunde 1924–1931. *Archiv Für Psychiatrie Und Nervenkrankheiten*, *227*(4), 279–300. https://doi.org/10.1007/BF00344814

Jung, T. P., Makeig, S., Humphries, C., Lee, T. W., Mckeown, M. J., Iragui, V., & Sejnowski, T. J. (2000). Removing electroencephalographic artifacts by blind source separation. *Psychophysiology*, *37*(2), 163–178. https://doi.org/10.1017/S0048577200980259

Jurcak, V., Tsuzuki, D., & Dan, I. (2007). 10/20, 10/10, and 10/5 systems revisited: their validity as relative head-surface-based positioning systems. *NeuroImage*, *34*(4), 1600–1611. https://doi.org/10.1016/j.neuroimage.2006.09.024

Kaplan, J. T., Man, K., & Greening, S. G. (2015). Multivariate cross-classification: applying machine learning techniques to characterize abstraction in neural representations. *Frontiers in Human Neuroscience*, *9*, 151. https://doi.org/10.3389/fnhum.2015.00151

Kay, L. M. (2015). Olfactory system oscillations across phyla. *Current Opinion in Neurobiology*, *31*, 141–147. https://doi.org/https://doi.org/10.1016/j.conb.2014.10.004

Keil, A., Debener, S., Gratton, G., Junghöfer, M., Kappenman, E. S., Luck, S. J., Luu, P., Miller, G. A., & Yee, C. M. (2014). Committee report: Publication guidelines and recommendations for studies using electroencephalography and magnetoencephalography. *Psychophysiology*, *51*(1), 1–21. https://doi.org/10.1111/psyp.12147

Kerrén, C., Linde-Domingo, J., Hanslmayr, S., & Wimber, M. (2018). An Optimal Oscillatory Phase for Pattern Reactivation during Memory Retrieval. *Current Biology : CB*, *28*(21), 3383–3392. https://doi.org/10.1016/j.cub.2018.08.065

King, J.-R., & Dehaene, S. (2014). Characterizing the dynamics of mental representations: the temporal generalization method. *Trends in Cognitive Sciences*, *18*(4), 203–210. https://doi.org/10.1016/j.tics.2014.01.002

King, J. R., Faugeras, F., Gramfort, A., Schurger, A., El Karoui, I., Sitt, J. D., Rohaut, B., Wacongne, C., Labyt, E., Bekinschtein, T., Cohen, L., Naccache, L., & Dehaene, S. (2013). Single-trial decoding of auditory novelty responses facilitates the detection of residual consciousness. *NeuroImage*, *83*, 726–738. https://doi.org/10.1016/j.neuroimage.2013.07.013

Kirschstein, T., & Köhling, R. (2009). What is the Source of the EEG? *Clinical EEG and Neuroscience*, *40*(3), 146–149. https://doi.org/10.1177/155005940904000305

Kleiner, M., Brainard, D., Pelli, D., Ingling, A., Murray, R., Broussard, C., & others. (2007). What`s new in Psychtoolbox-3. *Perception*, *36*(14).

Koenig-Robert, R., & Pearson, J. (2019). Decoding the contents and strength of imagery before volitional engagement. *Scientific Reports*, *9*(1), 3504. https://doi.org/10.1038/s41598-019-39813-y

Koley, B., & Dey, D. (2012). An ensemble system for automatic sleep stage classification using single channel EEG signal. *Computers in Biology and Medicine*, *42*(12), 1186–1195. https://doi.org/10.1016/J.COMPBIOMED.2012.09.012

Kool, W., McGuire, J. T., Rosen, Z. B., & Botvinick, M. M. (2010). Decision Making and the Avoidance of Cogntive Demand. *Journal of Experimental Psychology: General*, *139*(4), 665–682. https://doi.org/10.1037/a0020198.Decision

Korjus, K., Hebart, M. N., & Vicente, R. (2016). An Efficient Data Partitioning to Improve Classification Performance While Keeping Parameters Interpretable. *PloS One*, *11*(8), e0161788. https://doi.org/10.1371/journal.pone.0161788

Kothe, C. A., & Makeig, S. (2013). BCILAB: A platform for brain-computer interface development. *Journal of Neural Engineering*, *10*(5). https://doi.org/10.1088/1741-2560/10/5/056014

Kotz, S. A., Kalberlah, C., Bahlmann, J., Friederici, A. D., & Haynes, J.-D. (2013). Predicting vocal emotion expressions from the human brain. *Human Brain Mapping*, *34*(8), 1971–1981. https://doi.org/https://doi.org/10.1002/hbm.22041

Kravitz, D. J., Kriegeskorte, N., & Baker, C. I. (2010). High-Level Visual Object Representations Are Constrained by Position. *Cerebral Cortex*, *20*(12), 2916–2925.

https://doi.org/10.1093/cercor/bhq042

Kriegeskorte, N., & Bandettini, P. (2007). Analyzing for information, not activation, to exploit high-resolution fMRI. *NeuroImage*, *38*(4), 649–662. https://doi.org/10.1016/J.NEUROIMAGE.2007.02.022

Kriegeskorte, N., Goebel, R., & Bandettini, P. (2006). Information-based functional brain mapping. *Proceedings of the National Academy of Sciences of the United States of America*, *103*(10), 3863–3868. https://doi.org/10.1073/pnas.0600244103

Kriegeskorte, N., & Kievit, R. A. (2013). Representational geometry: Integrating cognition, computation, and the brain. *Trends in Cognitive Sciences*, *17*(8), 401–412. https://doi.org/10.1016/j.tics.2013.06.007

Kriegeskorte, N., Mur, M., & Bandettini, P. (2008). Representational similarity analysis - connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*, *2*(NOV), 1–28. https://doi.org/10.3389/neuro.06.004.2008

Krishnan, A., Williams, L. J., Randal, A., & Abdi, H. (2011). NeuroImage Partial Least Squares (PLS) methods for neuroimaging : A tutorial and review. *NeuroImage*, *56*(2), 455–475. https://doi.org/10.1016/j.neuroimage.2010.07.034

LaRocque, J. J., Lewis-Peacock, J. A., Drysdale, A. T., Oberauer, K., & Postle, B. R. (2013). Decoding Attended Information in Short-term Memory: An EEG Study. *Journal of Cognitive Neuroscience*, *25*(1), 127–142. https://doi.org/10.1162/jocn_a_00305

Laszlo, S., Ruiz-Blondet, M., Khalifian, N., Chu, F., & Jin, Z. (2014). A direct comparison of active and passive amplification electrodes in the same amplifier system. *Journal of Neuroscience Methods*, *235*, 298–307. https://doi.org/https://doi.org/10.1016/j.jneumeth.2014.05.012

Lemm, S., Blankertz, B., Dickhaus, T., & Müller, K. R. (2011). Introduction to machine learning for brain imaging. *NeuroImage*, *56*(2), 387–399. https://doi.org/10.1016/j.neuroimage.2010.11.004

Lindquist, M. A., Meng Loh, J., Atlas, L. Y., & Wager, T. D. (2009). Modeling the hemodynamic response function in fMRI: Efficiency, bias and mis-modeling. *NeuroImage*, *45*(1), S187–S198. https://doi.org/10.1016/j.neuroimage.2008.10.065

Logothetis, N. K., & Wandell, B. A. (2004). Interpreting the BOLD Signal. *Annual Review of Physiology*, *66*(1), 735–769. https://doi.org/10.1146/annurev.physiol.66.082602.092845

Lopez-Calderon, J., & Luck, S. J. (2014). ERPLAB: An open-source toolbox for the analysis of event-related potentials. *Frontiers in Human Neuroscience*, *8*(1 APR), 1–14. https://doi.org/10.3389/fnhum.2014.00213

López-García, D., Peñalver, J. M. G., Górriz, J. M., & Ruz, M. (2022). MVPAlab: A machine learning decoding toolbox for multidimensional electroencephalography data. *Computer Methods and Programs in Biomedicine*, *214*, 106549. https://doi.org/10.1016/j.cmpb.2021.106549

López-García, D., Ruz, M., Ramírez, J., & Górriz, J. M. (2018a). Automatic detection of sleep disorders:

Multi-class automatic classification algorithms based on Support Vector Machines. *International Conference on Time Series and Forecasting (ITISE 2018)*, *3*, 1270–1280.

López-García, D., Ruz, M., Ramírez, J., & Górriz, J. M. (2018b). Automatic detection of sleep disorders: Multi-class automatic classification algorithms based on Support Vector Machines. *International Conference on Time Series and Forecasting (ITISE 2018)*, *3*, 1270–1280.

López-García, D., Sobrado, A., González-Peñalver, J. M., Górriz, J. M., & Ruz, M. (2019). Multivariate Pattern Analysis of Electroencephalography Data in a Demand-Selection Task. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 403–411. https://doi.org/10.1007/978-3-030-19591-5_41

López-García, D., Sobrado, A., Peñalver, J. M. G., Górriz, J. M., & Ruz, M. (2020). Multivariate Pattern Analysis Techniques for Electroencephalography Data to Study Flanker Interference Effects. *International Journal of Neural Systems*, *30*(7). https://doi.org/10.1142/S0129065720500240

Luck, S. J. (2014). *An introduction to the event-related potential technique*. MIT press.

Ma, D. S., Correll, J., & Wittenbrink, B. (2015). The Chicago face database: A free stimulus set of faces and norming data. *Behavior Research Methods*, *47*(4), 1122–1135. https://doi.org/10.3758/s13428-014-0532-5

Makeig, S., Bell, A. J., Jung, T.-P., Sejnowski, T. J., & others. (1996). Independent component analysis of electroencephalographic data. *Advances in Neural Information Processing Systems*, 145–151.

Maris, E., & Oostenveld, R. (2007). Nonparametric statistical testing of EEG- and MEG-data. *Journal of Neuroscience Methods*, *164*(1), 177–190. https://doi.org/10.1016/j.jneumeth.2007.03.024

Martinez-Murcia, F. J., Górriz, J. M., Ramírez, J., & Ortiz, A. (2018). Convolutional Neural Networks for Neuroimaging in Parkinson's Disease: Is Preprocessing Needed? *International Journal of Neural Systems*, *28*(10), 7–12. https://doi.org/10.1142/S0129065718500351

Martinez-Murcia, F. J., Ortiz, A., Gorriz, J.-M., Ramirez, J., & Castillo-Barnes, D. (2020). Studying the Manifold Structure of Alzheimer's Disease: A Deep Learning Approach Using Convolutional Autoencoders. *IEEE Journal of Biomedical and Health Informatics*, *24*(1), 17–26. https://doi.org/10.1109/JBHI.2019.2914970

Mathewson, K. E., Harrison, T. J. L., & Kizuk, S. A. D. (2017). High and dry? Comparing active dry EEG electrodes to active and passive wet electrodes. *Psychophysiology*, *54*(1), 74–82. https://doi.org/https://doi.org/10.1111/psyp.12536

Matsumoto, A., Brinkmann, B. H., Matthew Stead, S., Matsumoto, J., Kucewicz, M. T., Marsh, W. R., Meyer, F., & Worrell, G. (2013). Pathological and physiological high-frequency oscillations in focal human epilepsy. *Journal of Neurophysiology*, *110*(8), 1958–1964. https://doi.org/10.1152/jn.00341.2013

Meconi, F., Linde-Domingo, J., Ferreira, C. S., Michelmann, S., Staresina, B., Apperly, I., & Hanslmayr, S. (2019). Autobiographical memory reactivation in empathy. *BioRxiv*.

https://doi.org/10.1101/715276

Megías, M., Emri, Z., Freund, T. F., & Gulyás, A. I. (2001). Total number and distribution of inhibitory and excitatory synapses on hippocampal CA1 pyramidal cells. *Neuroscience*, *102*(3), 527–540. https://doi.org/10.1016/S0306-4522(00)00496-6

Meissner, S. N., Krause, V., Südmeyer, M., Hartmann, C. J., & Pollok, B. (2018). The significance of brain oscillations in motor sequence learning: Insights from Parkinson's disease. *NeuroImage: Clinical*, *20*, 448–457. https://doi.org/https://doi.org/10.1016/j.nicl.2018.08.009

Misaki, M., Kim, Y., Bandettini, P. A., & Kriegeskorte, N. (2010). Comparison of multivariate classifiers and response normalizations for pattern-information fMRI. *NeuroImage*, *53*(1), 103–118. https://doi.org/10.1016/j.neuroimage.2010.05.051

Molloy, E. K., Meyerand, M. E., & Birn, R. M. (2014). The influence of spatial resolution and smoothing on the detectability of resting-state and task fMRI. *NeuroImage*, *86*, 221–230. https://doi.org/10.1016/j.neuroimage.2013.09.001

Monti, M. M. (2011). Statistical analysis of fMRI time-series: A critical review of the GLM approach. *Frontiers in Human Neuroscience*, *5*(MARCH), 1–13. https://doi.org/10.3389/fnhum.2011.00028

Mur, M., Bandettini, P. A., & Kriegeskorte, N. (2009). Revealing representational content with pattern-information fMRI - An introductory guide. *Social Cognitive and Affective Neuroscience*, *4*(1), 101–109. https://doi.org/10.1093/scan/nsn044

Mur, M., Ruff, D. A., Bodurka, J., Bandettini, P. A., & Kriegeskorte, N. (2010). Face-Identity Change Activation Outside the Face System: "Release from Adaptation" May Not Always Indicate Neuronal Selectivity. *Cerebral Cortex*, *20*(9), 2027–2042. https://doi.org/10.1093/cercor/bhp272

Muukkonen, I., Ölander, K., Numminen, J., & Salmela, V. R. (2020). Spatio-temporal dynamics of face perception. *NeuroImage*, *209*, 116531. https://doi.org/https://doi.org/10.1016/j.neuroimage.2020.116531

Mwangi, B., Tian, T. S., & Soares, J. C. (2014). A Review of Feature Reduction Techniques in Neuroimaging. *Neuroinformatics*, *12*(2), 229–244. https://doi.org/10.1007/s12021-013-9204-3

Nichols, T. E., & Holmes, A. P. (2002). Nonparametric permutation tests for functional neuroimaging: a primer with examples. *Human Brain Mapping*, *15*(1), 1–25. http://www.ncbi.nlm.nih.gov/pubmed/11747097

Niedermeyer, E., & da Silva, F. H. L. (2005). *Electroencephalography: basic principles, clinical applications, and related fields*. Lippincott Williams \&amp; Wilkins.

Nili, H., Wingfield, C., Walther, A., Su, L., Marslen-Wilson, W., & Kriegeskorte, N. (2014). A Toolbox for Representational Similarity Analysis. *PLoS Computational Biology*, *10*(4). https://doi.org/10.1371/journal.pcbi.1003553

Norman, K. A., Polyn, S. M., Detre, G. J., & Haxby, J. V. (2006). Beyond mind-reading: multi-voxel pattern analysis of fMRI data. *Trends in Cognitive Sciences*, *10*(9), 424–430.

https://doi.org/10.1016/j.tics.2006.07.005

O' Regan, S., Faul, S., & Marnane, W. (2010). Automatic detection of EEG artefacts arising from head movements. *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, 6353–6356. https://doi.org/10.1109/IEMBS.2010.5627282

O'Neill, G. C., Barratt, E. L., Hunt, B. A. E., Tewarie, P. K., & Brookes, M. J. (2015). Measuring electrophysiological connectivity by power envelope correlation: A technical review on MEG methods. *Physics in Medicine and Biology*, *60*(21), R271–R295. https://doi.org/10.1088/0031-9155/60/21/R271

Oostenveld, R., Fries, P., Maris, E., & Schoffelen, J. M. (2011). FieldTrip: Open source software for advanced analysis of MEG, EEG, and invasive electrophysiological data. *Computational Intelligence and Neuroscience*, *2011*. https://doi.org/10.1155/2011/156869

Oostenveld, R., & Praamstra, P. (2001). The five percent electrode system for high-resolution EEG and ERP measurements. *Clinical Neurophysiology*, *112*(4), 713–719. https://doi.org/https://doi.org/10.1016/S1388-2457(00)00527-7

Oosterhof, N. N., Connolly, A. C., & Haxby, J. V. (2016). CoSMoMVPA: Multi-modal multivariate pattern analysis of neuroimaging data in matlab/GNU octave. *Frontiers in Neuroinformatics*, *10*(JUL), 1–27. https://doi.org/10.3389/fninf.2016.00027

Oosterhof, N. N., Tipper, S. P., & Downing, P. E. (2013). Crossmodal and action-specific: neuroimaging the human mirror neuron system. *Trends in Cognitive Sciences*, *17*(7), 311–318. https://doi.org/10.1016/j.tics.2013.04.012

Oosterhof, N. N., Wiggett, A. J., Diedrichsen, J., Tipper, S. P., & Downing, P. E. (2010). Surface-Based Information Mapping Reveals Crossmodal Vision–Action Representations in Human Parietal and Occipitotemporal Cortex. *Journal of Neurophysiology*, *104*(2), 1077–1089. https://doi.org/10.1152/jn.00326.2010

Osipova, D., Ahveninen, J., Jensen, O., Ylikoski, A., & Pekkonen, E. (2005). Altered generation of spontaneous oscillations in Alzheimer's disease. *NeuroImage*, *27*(4), 835–841. https://doi.org/https://doi.org/10.1016/j.neuroimage.2005.05.011

Palenciano, A. F., González-García, C., Arco, J. E., Pessoa, L., & Ruz, M. (2019). Representational Organization of Novel Task Sets during Proactive Encoding. *The Journal of Neuroscience*, *39*(42), 8386–8397. https://doi.org/10.1523/JNEUROSCI.0725-19.2019

Palotti, J., Mall, R., Aupetit, M., Rueschman, M., Singh, M., Sathyanarayana, A., Taheri, S., & Fernandez-Luque, L. (2019). Benchmark on a large cohort for sleep-wake classification with machine learning techniques. *Npj Digital Medicine*, *2*(1), 1–9. https://doi.org/10.1038/s41746-019-0126-9

Parker, D., Liu, X., & Razlighi, Q. R. (2017). Optimal slice timing correction and its interaction with fMRI parameters and artifacts. *Medical Image Analysis*, *35*, 434–445. https://doi.org/10.1016/j.media.2016.08.006

Parks, T. W., & Burrus, C. S. (1987). *Digital filter design*. Wiley-Interscience.

Peñalver, J. M. G., López-García, D., González-García, C., Górriz, J. M., & Ruz, M. (2022). *Preparatory neural activity differs between Selective Attention and Perceptual Expectations*. 1–31.

Penny, W. D., Friston, K. J., Ashburner, J. T., Kiebel, S. J., & Nichols, T. E. (2011). *Statistical parametric mapping: the analysis of functional brain images*. Elsevier.

Pereira, F., Mitchell, T., & Botvinick, M. (2009). Machine learning classifiers and fMRI: A tutorial overview. *NeuroImage*, *45*(1), 199–209. https://doi.org/10.1016/j.neuroimage.2008.11.007

Pernet, C. R., Appelhoff, S., Flandin, G., Phillips, C., Delorme, A., & Oostenveld, R. (2018). BIDS-EEG: an extension to the Brain Imaging Data Structure (BIDS) Specification for electroencephalography. *PsyArXiv*. https://doi.org/10.31234/osf.io/63a4y

Perrin, F., Pernier, J., Bertrand, O., & Echallier, J. F. (1989). Spherical splines for scalp potential and current density mapping. *Electroencephalography and Clinical Neurophysiology*, *72*(2), 184–187. https://doi.org/https://doi.org/10.1016/0013-4694(89)90180-6

Pion-Tonachini, L., Kreutz-Delgado, K., & Makeig, S. (2019). ICLabel: An automated electroencephalographic independent component classifier, dataset, and website. *NeuroImage*, *198*, 181–197. https://doi.org/10.1016/j.neuroimage.2019.05.026

Pletzer, B., Kerschbaum, H., & Klimesch, W. (2010). When frequencies never synchronize: The golden mean and the resting EEG. *Brain Research*, *1335*, 91–102. https://doi.org/https://doi.org/10.1016/j.brainres.2010.03.074

Poline, J. B., & Brett, M. (2012). The general linear model and fMRI: Does love last forever? *NeuroImage*, *62*(2), 871–880. https://doi.org/10.1016/j.neuroimage.2012.01.133

Popal, H., Wang, Y., & Olson, I. R. (2019). A Guide to Representational Similarity Analysis for Social Neuroscience. *Social Cognitive and Affective Neuroscience*, *14*(11), 1243–1253. https://doi.org/10.1093/scan/nsz099

Posner, M. I., & Raichle, M. E. (1994). *Images of mind.* Scientific American Library/Scientific American Books.

Purcell, E. M., Torrey, H. C., & Pound, R. V. (1946). Resonance absorption by nuclear magnetic moments in a solid. *Physical Review*, *69*(1–2), 37.

Rafiei, M. H., & Adeli, H. (2017). A New Neural Dynamic Classification Algorithm. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(12), 3074–3083. https://doi.org/10.1109/TNNLS.2017.2682102

Ramírez, J., Górriz, J. M., Salas-Gonzalez, D., Romero, A., López, M., Álvarez, I., & Gómez-Río, M. (2013). Computer-aided diagnosis of Alzheimer's type dementia combining support vector machines and discriminant set of features. *Information Sciences*, *237*, 59–72. https://doi.org/10.1016/J.INS.2009.05.012

Ramón y Cajal, S. (1988). *Cajal on the Cerebral Cortex: An Annotated Translation of the Complete*

Writings (J. DeFelipe & Edward G. Jones (eds.)). Oxford University Press.

Röther, J., Knab, R., Hamzei, F., Fiehler, J., Reichenbach, J. R., Büchel, C., & Weiller, C. (2002). Negative dip in BOLD fMRI is caused by blood flow - Oxygen consumption uncoupling in humans. *NeuroImage*, *15*(1), 98–102. https://doi.org/10.1006/nimg.2001.0965

Rousselet, G. (2012). Does Filtering Preclude Us from Studying ERP Time-Courses? *Frontiers in Psychology*, *3*. https://doi.org/10.3389/fpsyg.2012.00131

Saenz, M., Buracas, G. T., & Boynton, G. M. (2002). Global effects of feature-based attention in human visual cortex. *Nature Neuroscience*, *5*(7), 631–632. https://doi.org/10.1038/nn876

Salas-Gonzalez, D., Górriz, J. M., Ramírez, J., López, M., Álvarez, I., Segovia, F., Chaves, R., & Puntonet, C. G. (2010). Computer-aided diagnosis of Alzheimer's disease using support vector machines and classification trees. *Physics in Medicine and Biology*, *55*(10), 2807–2817. https://doi.org/10.1088/0031-9155/55/10/002

Schnitzler, A., & Gross, J. (2005). Normal and pathological oscillatory communication in the brain. *Nature Reviews Neuroscience*, *6*(4), 285–296. https://doi.org/10.1038/nrn1650

Schouppe, N., Demanet, J., Boehler, C. N., Ridderinkhof, K. R., & Notebaert, W. (2014). The Role of the Striatum in Effort-Based Decision-Making in the Absence of Reward. *Journal of Neuroscience*, *34*(6), 2148–2154. https://doi.org/10.1523/JNEUROSCI.1214-13.2014

Schrouff, J., Rosa, M. J., Rondina, J. M., Marquand, A. F., Chu, C., Ashburner, J., Phillips, C., Richiardi, J., & Mourão-Miranda, J. (2013). PRoNTo: Pattern recognition for neuroimaging toolbox. *Neuroinformatics*, *11*(3), 319–337. https://doi.org/10.1007/s12021-013-9178-1

Schubring, D., & Schupp, H. T. (2021). Emotion and Brain Oscillations: High Arousal is Associated with Decreases in Alpha- and Lower Beta-Band Power. *Cerebral Cortex*, *31*(3), 1597–1608. https://doi.org/10.1093/cercor/bhaa312

Schultz, J. (2010). Brain Imaging: Decoding Your Memories. *Current Biology*, *20*(6), R269–R271. https://doi.org/https://doi.org/10.1016/j.cub.2010.02.001

Schutter, D. J. L. G., & Knyazev, G. G. (2012). Cross-frequency coupling of brain oscillations in studying motivation and emotion. *Motivation and Emotion*, *36*(1), 46–54. https://doi.org/10.1007/s11031-011-9237-6

Seeck, M., Koessler, L., Bast, T., Leijten, F., Michel, C., Baumgartner, C., He, B., & Beniczky, S. (2017). The standardized EEG electrode array of the IFCN. *Clinical Neurophysiology*, *128*(10), 2070–2077. https://doi.org/10.1016/j.clinph.2017.06.254

Shannon, C. E. (1949). Communication in the Presence of Noise. *Proceedings of the IRE*, *37*(1), 10–21. https://doi.org/10.1109/JRPROC.1949.232969

Shatek, S. M., Grootswagers, T., Robinson, A. K., & Carlson, T. A. (2019). Decoding Images in the Mind's Eye: The Temporal Dynamics of Visual Imagery. *Vision*, *3*(4), 53. https://doi.org/10.3390/vision3040053

Shiraishi, J., Li, Q., Appelbaum, D., & Doi, K. (2011). Computer-aided diagnosis and artificial intelligence in clinical imaging. *Seminars in Nuclear Medicine*, *41*(6), 449–462. https://doi.org/10.1053/j.semnuclmed.2011.06.004

Singh, D., & Singh, B. (2020). Investigating the impact of data normalization on classification performance. *Applied Soft Computing*, *97*, 105524. https://doi.org/10.1016/j.asoc.2019.105524

Sladky, R., Friston, K. J., Tröstl, J., Cunnington, R., Moser, E., & Windischberger, C. (2011). Slice-timing effects and their correction in functional MRI. *NeuroImage*, *58*(2), 588–594. https://doi.org/10.1016/j.neuroimage.2011.06.078

Soares, J. M., Magalhães, R., Moreira, P. S., Sousa, A., Ganz, E., Sampaio, A., Alves, V., Marques, P., & Sousa, N. (2016). A Hitchhiker's guide to functional magnetic resonance imaging. *Frontiers in Neuroscience*, *10*(November), 1–35. https://doi.org/10.3389/fnins.2016.00515

Solís-Vivanco, R., Rodríguez-Violante, M., Cervantes-Arriaga, A., Justo-Guillén, E., & Ricardo-Garcell, J. (2018). Brain oscillations reveal impaired novelty detection from early stages of Parkinson's disease. *NeuroImage: Clinical*, *18*, 923–931. https://doi.org/https://doi.org/10.1016/j.nicl.2018.03.024

Sols, I., DuBrow, S., Davachi, L., & Fuentemilla, L. (2017). Event Boundaries Trigger Rapid Memory Reinstatement of the Prior Events to Promote Their Representation in Long-Term Memory. *Current Biology*, *27*(22), 3499–3504. https://doi.org/10.1016/j.cub.2017.09.057

Soon, C. S., Brass, M., Heinze, H.-J., & Haynes, J.-D. (2008). Unconscious determinants of free decisions in the human brain. *Nature Neuroscience*, *11*(5), 543–545. https://doi.org/10.1038/nn.2112

Sörnmo, L., & Laguna, P. (2005). *Bioelectrical signal processing in cardiac and neurological applications* (Vol. 8). Academic press.

Stelzer, J., Chen, Y., & Turner, R. (2013). Statistical inference and multiple testing correction in classification-based multi-voxel pattern analysis (MVPA): Random permutations and cluster size control. *NeuroImage*, *65*, 69–82. https://doi.org/10.1016/j.neuroimage.2012.09.063

Su, L., Zulfiqar, I., Jamshed, F., Fonteneau, E., & Marslen-Wilson, W. (2014). Mapping tonotopic organization in human temporal cortex: Representational similarity analysis in EMEG source space. *Frontiers in Neuroscience*, *8*(OCT), 1–14. https://doi.org/10.3389/fnins.2014.00368

Summerfield, C., Egner, T., Greene, M., Koechlin, E., Mangels, J., & Hirsch, J. (2006). Predictive Codes for Forthcoming Perception in the Frontal Cortex. *Science*, *314*(5803), 1311–1314. https://doi.org/10.1126/science.1132028

Sun, Y., Wong, A. K. C., & Kamel, M. S. (2009). Classification of imbalanced data: A review. *International Journal of Pattern Recognition and Artificial Intelligence*, *23*(4), 687–719. https://doi.org/10.1142/S0218001409007326

Symeonidou, E.-R., Nordin, A. D., Hairston, W. D., & Ferris, D. P. (2018). Effects of Cable Sway, Electrode Surface Area, and Electrode Mass on Electroencephalography Signal Quality during

Motion. *Sensors*, *18*(4). https://doi.org/10.3390/s18041073

Syrjälä, J., Basti, A., Guidotti, R., Marzetti, L., & Pizzella, V. (2021). Decoding working memory task condition using magnetoencephalography source level long-range phase coupling patterns. *Journal of Neural Engineering*, *18*(1). https://doi.org/10.1088/1741-2552/abcefe

Taheri, B. A., Knight, R. T., & Smith, R. L. (1994). A dry electrode for EEG recording. *Electroencephalography and Clinical Neurophysiology*, *90*(5), 376–383. https://doi.org/10.1016/0013-4694(94)90053-1

Takacs, A., Mückschel, M., Roessner, V., & Beste, C. (2020). Decoding Stimulus–Response Representations and Their Stability Using EEG-Based Multivariate Pattern Analysis. *Cerebral Cortex Communications*, *1*(1). https://doi.org/10.1093/texcom/tgaa016

Talairach, J. (1988). Co-planar stereotaxic atlas of the human brain-3-dimensional proportional system. *An Approach to Cerebral Imaging*.

Talairach, J., & Szikla, G. (1980). Application of stereotactic concepts to the surgery of epilepsy. In *Advances in Stereotactic and Functional Neurosurgery 4* (pp. 35–54). Springer.

Tomassini, A., Ambrogioni, L., Medendorp, W. P., & Maris, E. (2017). Theta oscillations locked to intended actions rhythmically modulate perception. *Elife*, *6*, e25618.

Treder, M. S. (2020). MVPA-Light: A Classification and Regression Toolbox for Multi-Dimensional Data. *Frontiers in Neuroscience*, *14*(June), 1–19. https://doi.org/10.3389/fnins.2020.00289

Triantafyllou, C., Hoge, R. D., & Wald, L. L. (2006). Effect of spatial smoothing on physiological noise in high-resolution fMRI. *NeuroImage*, *32*(2), 551–557. https://doi.org/10.1016/j.neuroimage.2006.04.182

Urigüen, J. A., & Garcia-Zapirain, B. (2015). EEG artifact removal-state-of-the-art and guidelines. *Journal of Neural Engineering*, *12*(3), 31001. https://doi.org/10.1088/1741-2560/12/3/031001

Van Der Maaten, L. J. P., Postma, E. O., & Van Den Herik, H. J. (2009). Dimensionality Reduction: A Comparative Review. *Journal of Machine Learning Research*, *10*, 1–41. https://doi.org/10.1080/13506280444000102

van Dijk, K. R. A., Sabuncu, M. R., & Buckner, R. L. (2012). The influence of head motion on intrinsic functional connectivity MRI. *NeuroImage*, *59*(1), 431–438. https://doi.org/10.1016/j.neuroimage.2011.07.044

van Driel, J., Olivers, C. N. L., & Fahrenfort, J. J. (2021). High-pass filtering artifacts in multivariate classification of neural time series data. *Journal of Neuroscience Methods*, *352*(January), 109080. https://doi.org/10.1016/j.jneumeth.2021.109080

Van Veen, V., & Carter, C. S. (2002). The anterior cingulate as a conflict monitor: FMRI and ERP studies. *Physiology and Behavior*, *77*(4–5), 477–482. https://doi.org/10.1016/S0031-9384(02)00930-7

van Zijl, P. C. M., Hua, J., & Lu, H. (2012). The BOLD post-stimulus undershoot, one of the most debated issues in fMRI. *NeuroImage*, *62*(2), 1092–1102. https://doi.org/10.1016/j.neuroimage.2012.01.029

VanRullen, R. (2011). Four common conceptual fallacies in mapping the time course of recognition. *Frontiers in Psychology*, *2*(DEC), 1–6. https://doi.org/10.3389/fpsyg.2011.00365

Varoquaux, G. (2018). Cross-validation failure: Small sample sizes lead to large error bars. *NeuroImage*, *180*, 68–77. https://doi.org/10.1016/j.neuroimage.2017.06.061

Varoquaux, G., Raamana, P. R., Engemann, D. A., Hoyos-Idrobo, A., Schwartz, Y., & Thirion, B. (2017). Assessing and tuning brain decoders: Cross-validation, caveats, and guidelines. *NeuroImage*, *145*(October 2016), 166–179. https://doi.org/10.1016/j.neuroimage.2016.10.038

Vidaurre, D., Cichy, R. M., & Woolrich, M. W. (2020). Dissociable components of oscillatory activity underly information encoding in human perception. *BioRxiv*, 1–29.

Waldorp, L. (2009). Robust and unbiased variance of glm coefficients for misspecified autocorrelation and hemodynamic response models in fMRI. *International Journal of Biomedical Imaging*, *2009*. https://doi.org/10.1155/2009/723912

Wall, D. P., Kosmicki, J., Deluca, T. F., Harstad, E., & Fusaro, V. A. (2012). Use of machine learning to shorten observation-based screening and diagnosis of autism. *Translational Psychiatry*, *2*(January). https://doi.org/10.1038/tp.2012.10

Walther, A., Nili, H., Ejaz, N., Alink, A., Kriegeskorte, N., & Diedrichsen, J. (2016). Reliability of dissimilarity measures for multi-voxel pattern analysis. *NeuroImage*, *137*, 188–200. https://doi.org/10.1016/j.neuroimage.2015.12.012

Wang, S., Celebi, M. E., Zhang, Y. D., Yu, X., Lu, S., Yao, X., Zhou, Q., Miguel, M. G., Tian, Y., Gorriz, J. M., & Tyukin, I. (2021). Advances in data preprocessing for bio-medical data fusion: An overview of the methods, challenges, and prospects. *Information Fusion*, *76*(July), 376–421. https://doi.org/10.1016/j.inffus.2021.07.001

Wang, S. H., Govindaraj, V. V., Górriz, J. M., Zhang, X., & Zhang, Y. D. (2021). Covid-19 classification by FGCNet with deep feature fusion from graph convolutional network and convolutional neural network. *Information Fusion*, *67*(September 2020), 208–229. https://doi.org/10.1016/j.inffus.2020.10.004

Wang, X.-J. (2010). Neurophysiological and Computational Principles of Cortical Rhythms in Cognition. *Physiological Reviews*, *90*(3), 1195–1268. https://doi.org/10.1152/physrev.00035.2008

Weiner, O. M., & Dang-Vu, T. T. (2016). Spindle Oscillations in Sleep Disorders: A Systematic Review. *Neural Plasticity*, *2016*, 7328725. https://doi.org/10.1155/2016/7328725

Weiss, S., & Mueller, H. (2012). "Too Many betas do not Spoil the Broth": The Role of Beta Brain Oscillations in Language Processing. *Frontiers in Psychology*, *3*. https://doi.org/10.3389/fpsyg.2012.00201

Welcome Centre for Human Neuroimaging. (2018). *Statistical Parametrical Mapping*. https://www.fil.ion.ucl.ac.uk/spm/software/spm12

Welvaert, M., & Rosseel, Y. (2013). On the definition of signal-to-noise ratio and contrast-to-noise ratio for fMRI data. *PLoS ONE*, *8*(11). https://doi.org/10.1371/journal.pone.0077089

Widmann, A., & Schröger, E. (2012). Filter Effects and Filter Artifacts in the Analysis of Electrophysiological Data. *Frontiers in Psychology*, *3*. https://doi.org/10.3389/fpsyg.2012.00233

Widmann, A., Schröger, E., & Maess, B. (2015). Digital filter design for electrophysiological data – a practical approach. *Journal of Neuroscience Methods*, *250*, 34–46. https://doi.org/https://doi.org/10.1016/j.jneumeth.2014.08.002

Winkler, I., Haufe, S., & Tangermann, M. (2011). Automatic Classification of Artifactual ICA-Components for Artifact Removal in EEG Signals. *Behavioral and Brain Functions*, *7*(1), 30. https://doi.org/10.1186/1744-9081-7-30

Womelsdorf, T., Fries, P., Mitra, P. P., & Desimone, R. (2006). Gamma-band synchronization in visual cortex predicts speed of change detection. *Nature*, *439*(7077), 733–736. https://doi.org/10.1038/nature04258

Wong, T. T. (2015). Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation. *Pattern Recognition*, *48*(9), 2839–2846. https://doi.org/10.1016/j.patcog.2015.03.009

Worrell, G. A., Parish, L., Cranstoun, S. D., Jonas, R., Baltuch, G., & Litt, B. (2004). High-frequency oscillations and seizure generation in neocortical epilepsy. *Brain*, *127*(7), 1496–1506. https://doi.org/10.1093/brain/awh149

Worsley, K. J., Evans, A. C., Marrett, S., & Neelin, P. (1992). A three-dimensional statistical analysis for CBF activation studies in human brain. *Journal of Cerebral Blood Flow and Metabolism*, *12*(6), 900–918. https://doi.org/10.1038/jcbfm.1992.127

Worsley, K. J., & Friston, K. J. (1995). Analysis of fMRI Time-Series Revisited—Again. *NeuroImage*, *2*(3), 173–181. https://doi.org/10.1006/nimg.1995.1023

Yang, Z., Fang, F., & Weng, X. (2012). Recent developments in multivariate pattern analysis for functional MRI. *Neuroscience Bulletin*, *28*(4), 399–408. https://doi.org/10.1007/s12264-012-1253-3

Yarach, U., Luengviriya, C., Stucht, D., Godenschweger, F., Schulze, P., & Speck, O. (2016). Correction of B 0-induced geometric distortion variations in prospective motion correction for 7T MRI. *Magnetic Resonance Materials in Physics, Biology and Medicine*, *29*(3), 319–332. https://doi.org/10.1007/s10334-015-0515-2

Zhang, R., Tie, X., Qi, Z., Bevins, N. B., Zhang, C., Griner, D., Song, T. K., Nadig, J. D., Schiebler, M. L., Garrett, J. W., Li, K., Reeder, S. B., & Chen, G. H. (2021). Diagnosis of Coronavirus Disease 2019 Pneumonia by Using Chest Radiography: Value of Artificial Intelligence. *Radiology*, *298*(2), E88–E97. https://doi.org/10.1148/RADIOL.2020202944

Zhang, Y. D., Dong, Z., Wang, S. H., Yu, X., Yao, X., Zhou, Q., Hu, H., Li, M., Jiménez-Mesa, C., Ramirez, J., Martinez, F. J., & Gorriz, J. M. (2020). Advances in multimodal data fusion in neuroimaging: Overview, challenges, and novel orientation. *Information Fusion*, *64*(July), 149–187. https://doi.org/10.1016/j.inffus.2020.07.006

Zijlmans, M., Jiruska, P., Zelmann, R., Leijten, F. S. S., Jefferys, J. G. R., & Gotman, J. (2012). High-frequency oscillations as a new biomarker in epilepsy. *Annals of Neurology*, *71*(2), 169–178. https://doi.org/https://doi.org/10.1002/ana.22548