Rapport de stage long

CHARBEL-RAPHAËL SEGERIE

ELÈVE INGÉNIEUR

DOUBLE DIPLOME À L'ENS PARIS-SACLAY

MASTER MATHÉMATIQUES VISION APPRENTISSAGE

Revealing

THE NEUROANATOMICAL SIGNATURES OF TIME IN WORKING MEMORY

with statistical learning

Fiche de synthèse

- Type de stage : Stage de recherche

- Année : 2021

- Auteur (Nom, prénom) : Charbel-Raphaël Segerie

- Formation 2ème année (IMI, GI, SEGF, etc.) : IMI

- Titre du rapport : The neuroanatomical signatures of time in working memory

- Titre en français : Les signatures neuroanatomiques du temps dans la mémoire de travail

- Organisme d'accueil : Inria Parietal / Neurospin

- Pays d'accueil : France

- Responsable de stage : Eric Duceau.

- Mots-clés caractérisant votre rapport (4 à 5 mots maximum) : Mémoire de travail, apprentissage statistique, magnétoencéphalographie, analyse temps-fréquence, signaux multivariés.

Résumé

L'évaluation de la durée est l'une des capacités fondamentales du cerveau. Cependant, nous ne savons pas comment cette information est stockée en mémoire. Ici, nous utilisons un paradigme de reproduction temporelle récemment développé pour étudier la dynamique neuronale du temps dans la mémoire de travail. Les participants ont écouté des séquences non rythmiques composées d'intervalles temporels, qu'ils devaient ensuite reproduire le plus fidèlement possible. Nous avons fait varier la longueur totale des séries temporelles ainsi que le nombre d'intervalles (n-item) les composant. L'expérience a été menée et enregistrée par magnétoencéphalographie ($N = 24$). Au cours de mon stage, j'ai développé des méthodes mathématiques pour analyser ces enregistrements. Nous procédons en deux étapes : 1. Nous procédons à une analyse temps-fréquence des données et nous trouvons les principaux clusters informationnels. 2. Nous utilisons les clusters temps-fréquence identifiés pour analyser les données dans l'espace source. Nos résultats suggèrent que la mémoire de travail temporelle est stockée principalement dans la bande alpha (8-14 Hz), et qu'elle est localisée dans l'aire somatosensorielle complétée par le cortex occipital.

Mots clés : Mémoire de travail, apprentissage statistique, magnétoencéphalographie, analyse temps-fréquence, signaux multivariés.

Abstract

Evaluating duration is one of the fundamental capacities of the brain. Yet, we do not know how this information is stored in memory. Here, we use a recently developed temporal reproduction paradigm to investigate the neural cynamic of time in working memory. Participants listened to non-rhythmic sequences composed of temporal intervals, which they then had to reproduce. We varied the total length of the temporal series as well as the number of intervals (n-item) composing them. The experiment was conducted and recorded by magnetoencephalography ($N = 24$). During my internship, I developed mathematical methods to analyze these recordings. We proceed in two steps: 1. We proceed to a time-frequency analysis of the data, and we find the main informational clusters. 2. We use the identified time-frequency clusters to analyze the data in the source space. Our results suggest that temporal working memory is stored predominantly in the alpha band (8-14 Hz), and that it is located in the somatosensory area complemented by the occipital cortex.

Keywords: Working memory, Statistical learning, Magnetoencephalography, Time-frequency analysis, Multivariate signals.

Contents

List of Figures

List of Tables

Host organizations

My internship was a bit particular because it took place between two research institutes. I worked in parallel with :

- Inria, within the Parietal team

- and Neurospin, in the Cognition and Brain Dynamics team.

Inria

The French National Institute for Research in Computer Science and Control (Inria) is a French national research institution focused on computer science and applied mathematics. Inria employs 3800 people, including 2800 researchers and doctoral students located all over France.

Inria - Parietal

The parietal team is a laboratory of about 30 researchers dedicated to learning brain structure, function and variability from neuroimaging data. The parietal team focuses on mathematical methods for statistical modeling of brain functions from neuroimaging data (fMRI, MEG, EEG), with a particular interest in machine learning techniques, applications to human cognitive neuroscience and scientific software development.

Within Inria parietal, I have particularly worked with the team of developers now **MNE-Python**, and **MNE-BIDS-Pipeline**, especially with Richard Höchenberger and Alexandre Gramfort. But I was able to be present at the team presentations every Tuesday afternoon during the parietal talks. Every Tuesday afternoon, we had team presentations, which gave us the opportunity to follow the work of the other members of the parietal team, and to present our work.

Neurospin

Neurospin is a brain imaging center in Saclay, south of Paris. The INSERM-CEA Cognitive Neuroimaging Unit, comprises five teams:

- Languages of the Brain, which attempts to answer the question: Why are we the only species with a sophisticated communication system?

- Neuroimaging of Development, which studies human cognitive development in infants and children, both structurally and functionally, and aims to develop new imaging techniques appropriate for human infants.

- Neuromodulation which focuses on brain function in primates.

- Computational Brain, which studies the different functions of the human brain from the perspective of computation and information coding.

- **Cognition and Brain Dynamics** which focuses on the processing of multisensory information, their temporal organization and in particular the representation of temporal information in the human brain, using magnetoencephalography (MEG) as the main methods.

I did my internship in this last team: Cognition and brain dynamics with Sophie Herbst, but I was also able to interact with some people from the other teams, especially during the Friday afternoon meetings that allow the 5 teams to get together.

Introduction

I divided this report into four parts.

The first chapter gives the scientific context of my internship. This part is an account of the environment in which I am inserted and of the knowledge required to understand my work hereafter. Science is no longer a solitary activity, especially in neuroscience, where the techniques used derive from years of collaborative work. I explain in this chapter the ins and outs of the pilot study paper [6] on which I rely. The goal of my internship is to complete this pilot study by analyzing magnetoencephalography (MEG) data, by finding appropriate mathematical methods to analyze the MEG data, and to implement these mathematical methods in an open source automatic analysis pipeline, the MNE-BIDS-Pipeline. This pipeline aims to allow future cognitive science researchers to analyze electroencephalogram data in a snap, while promoting replicability and open science. This part could constitute the introduction part of a future paper.

The second chapter presents the more technical aspects of my contribution to the reproduction of the pilot study. I detail the method used to analyze the parts of the brain in action during the use of the working memory. We used the MNE-BIDS-Pipeline to analyze the data. But we had to adapt the MNE-BIDS-Pipeline by using new algorithms to meet the requirements of our experimental paradigm. An important part of my work has been allocated to implement in open-source these methods. This part could constitute the method part of a future paper.

The third part presents the obtained results. The beginning of a discussion is also outlined.

An appendix provides more information on the algorithms used and on some mathematical subtleties. An important point is made on the optimization of the computation time. Indeed, even if the optimization of the computation time does not change the results, it was critical in the practical feasibility of our project.

Chapter 1

Scientific Context

1.1 The Time in Working Memory (TiWM) paradigm

This part presents the *Time in Working Memory* paradigm, which was exposed in the paper [6]. My internship is a continuation of this paper. We take up the experimental paradigm proposed in the pilot study, but this time we collect the data using MEG.

1.1.1 Description of the Pilot Study

Planning for the future relies on the ability to remember the duration of events, but it is unknown how durations are stored in the brain's working memory. The working memory is often described as a short-term information system, limited in capacity, and is considered to be a core cognitive process that underpins a range of behaviors [9]. The paper *abstracting time in working memory* proposes a new *n-item delayed duration reproduction task* to assess whether elapsed time is stored in memory as a continuous feature or as an abstract element.

The participants listened to sequences composed of empty time intervals that they had to reproduce as precisely as possible. For each sequence, one can manipulate the number of time intervals and the overall duration of the sequence to separate their effects on recall accuracy. The figure 1.2 presents the paradigm. Temporal reproduction accuracy decreased systematically with an increasing number of items. **These results suggested that the number of time intervals, not their duration, determines recall precision**. According to the brain clock models proposed by cognitive science, these results are interpreted as evidence of the existence of a symbolic representation of duration in working memory.
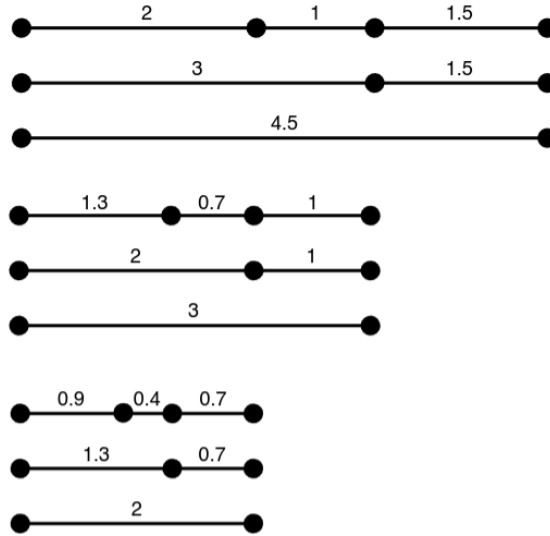
Figure 1.1: n-item delayed reproduction task, figure taken from [6]. The number of items and the sequence duration were orthogonalized. A sequence was composed of 1, 2 or 3 items with sequence duration fixed to 2 s, 3 s, or 4.5 s irrespective of the number of items composing it. Black dots represent 1kHz tones. All numbers are durations in seconds.
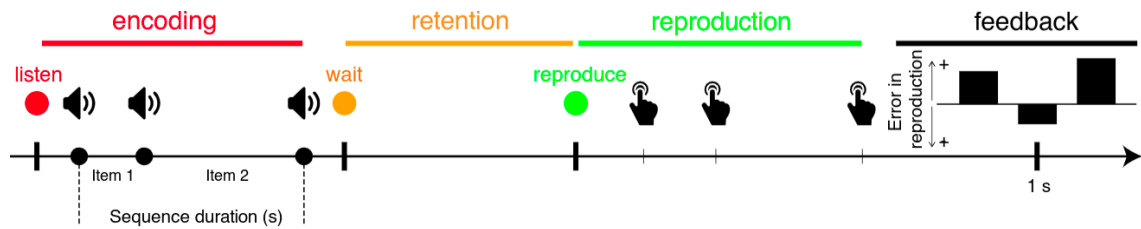
Figure 1.2: Example trial. In all experiments, each trial was composed of four phases: First, participants listened to a sequence of pure tones demarcating time intervals of varying duration. Following a retention interval, participants were asked to reproduce the temporal sequence as precisely as possible. In the example depicted here, the sequence was composed of two items. Following the reproduction (green), feedback was provided showing the relative temporal reproduction error for teach item in the sequence.

1.1.2 Reproducing the pilot study using MEG data

Continuing the pilot study Thus, according to this first study, we know that durations are not represented in a continuous way, but in a symbolic way. But we still do not know in which part of the brain the information is stored. In order to visualize in which part of the brain the working memory is encoded, we reproduced the experiment this time by recording the cerebral data with magnetoencephalograms. In this new experiment, we no longer concentrate on the analysis of the accuracy of the restitution of the duration of the time intervals, but we will concentrate more on the visualization of the frequencies and cerebral areas at stake for the working memory.

Working hypothesis The pilot study showed that the accuracy of temporal reproduction decreased systematically with the increase of the number of elements. We will therefore consider that two sequences of the same total duration, but composed of a greater number of items, load the working memory differently: the sequence with the greater number of items loads the working memory more. Thus, we will try to visualize the difference in working memory load for different numbers of items.

Differences in the experiment Compared to the original study, we changed two main points:

- We did not repeat the 3 experiments using MEGs, but only Experiment 2, which orthogonalizes the number of items and the total sequence duration.

- In the original Experiment 2, three sequences durations were tested (2 s, 3 s and 4.5 s) and the number of items was 1, 2 or 3 items for each sequence 1.1. In our replication of this experiment, we use only sequences of either 1 or 3 items, which allows to maximize the effect size.

1.1.3 Glossary

In the context of MEEG, the vocabulary used is very specific, and is essential to the understanding. The table 1.1 gives the main vocabulary of the neural oscillation theory. The table 1.2 gives the usual vocabulary used in experimental protocols.

1.2 EEG, MEG

EEG and MEG measure the electrical activity of our brain using electrodes placed on the scalp. It tells us, from the surface measurements, how active the brain is. It is possible to measure both EEG and meg data, which is why the acronym MEEG is used to designate these data.

For our experimental paradigm, it is much more important to have high temporal precision than

EEG-MEG	The Electro-magnetoencephalography is a non-invasive method for studying brain function that reflect the electrical activity of neuronal populations with millisecond temporal resolution.
Local field potential (LFP)	electric potential in the extracellular space around neurons. LFP is a widely available signal in many recording configurations, ranging from single-electrode recordings to multi-electrode arrays
Neuronal oscillations	prominent feature of spontaneous and task-related brain activity that occur at the level of single units, local field potentials (LFPs), and EEG/MEG recordings. The traditional view is that neuronal oscillations reflect inhibition-based fluctuations of neuronal activity that emerge from the synchronous activation of large neuronal ensembles.
Spectral power	reflects the amplitude of neural oscillations computed through a time–frequency transformation (TFT).

Table 1.1: Glossary of the Neural Oscillation and of the Theory of the working memory as defined in [12]

high spatial precision. This is why we choose to use MEEG rather than fMRI to study this paradigm.

Indeed, the temporal sampling rate of MEG is high, with more than 1000 Hz. This high temporal resolution contrasts with that of functional magnetic resonance imaging, which essentially detects changes in the concentration of oxygen in the blood, a system with a much slower response in the human brain, with a lag of several seconds, and therefore not suitable for our temporal reproduction paradigm.

1.2.1 MEG in comparison to EEG

MEG is a cutting-edge functional brain imaging technology. MEG is extremely sensitive and measures very weak magnetic fields produced by the electromagnetic activity of neurons. MEG is the safest of the various brain imaging technologies: no energy is deposited in the individual being monitored. The machine does not even touch the head. MEG is particularly important in basic and clinical research, and especially in studies with young child populations.

The advantage of measuring magnetic fields, rather than electric fields as in electroencephalography (EEG), is that they pass through the skull and other tissues between the active neurons and the MEG detectors without distortion, unlike EEG, where the signal is less accurate.

1.2.2 Equipment: MEG in Neurospin

MEGs are extremely rare in France, and MEG equipment in France can be counted on the fingers of one hand. But Neurospin is one of them. Our center has a shielded chamber protecting a MEG recording device from electromagnetic noise.

Session	A logical grouping of neuroimaging and behavioral data collected consistently across participants. A session includes the time involved in completing all experimental tasks. This begins when a participant enters the research environment until he/she leaves it.
Run	An uninterrupted period of continuous data acquisition without operator involvement.
Event	An isolated occurrence of a presented stimulus, or a subject response recorded during a task
Epoch	In the MEEG literature, the term epoch designates the outcome of a data segmentation process. Typically, epochs in event-related designs (for analysis of event related potentials or event related spectral perturbations) are time-locked to a particular event (such as a stimulus or a response)
Evoked data	Evoked objects typically store an EEG or MEG signal that has been averaged over multiple epochs, which is a common technique for estimating stimulus-evoked activity.
Sensors	Sensors are the physical objects or transducers that are used to perform the analogue recording, i.e., EEG electrodes and MEG magnetometers/ gradiometers. Sensors are connected to amplifiers, which not only amplify, but also filter the MEEG activity.
Channels	Channels refer to the digital signals that have been recorded by the amplifiers. It is thus important to distinguish them from sensors. A 'bad channel' refers to a channel that is producing a consistently artifactual or low-quality signal.
Sensor space	Sensor space refers to a representation of the MEEG data at the level of the original sensors, where each of the signals maps onto the spatial location of one of the sensors.
Source space	Source space refers to MEEG data reconstructed at the level of potential neural sources that presumably gave rise to the measured signals (according to an assumed biophysical model). Each signal maps onto a spatial location that is readily interpretable in relation to individual or template-based brain anatomy.

Table 1.2: Glossary of MEEG terminology commonly used to describe stimulation and task parameters and protocols, as defined in [11].

The MEG at Neurospin is an Elekta Neuromag device. The MEG helmet consists of 102 sensors-triplets (1 triplet = 2 orthogonal planar gradiometers and 1 magnetometer). The MEG data I manipulatied during the internship are composed of 204 gradiometer channels and 102 magnetometer channels (Figure 1.3) This organization between gradiometer and magnetometer will be important when we try to speed up the algorithms in the section A.1.1.

1.2.3 Frequency band

The high temporal accuracy of MEG data allows the study of signals in the frequency domain. The main frequency bands used are defined in Table 1.3.
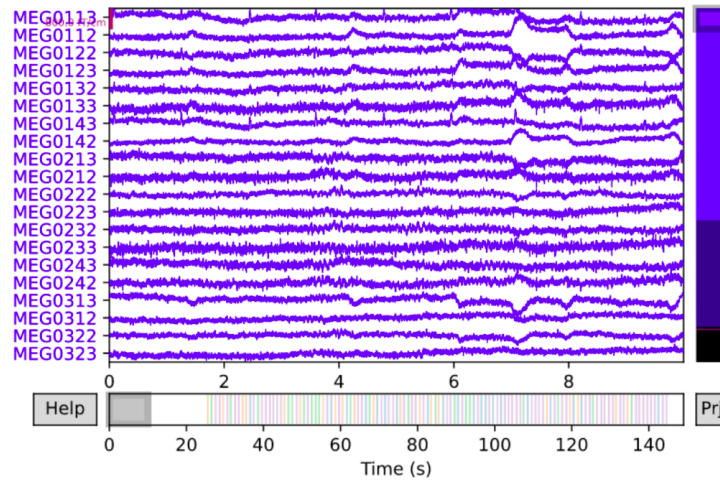
Figure 1.3: Example of an MEG recording of a passive session. Here, 20 of the 306 channels are represented.

Table 1.3: Comparison of EEG bands for a typical adult

Band	Frequency (Hz)	Location	Normally
Delta	≤ 4	frontally in adults, high-amplitude waves	slow-wave sleep
Theta	4–8	Found in locations not related to task at hand	Drowsiness or idling. Associated with inhibition of elicited responses.
Alpha	8–14	posterior regions of head	relaxed, inhibition control
Beta	14–30	mainly frontally, low-amplitude waves	active thinking, focus
Gamma	≥ 30	Somatosensory cortex	Cross-modal sensory processing

1.3 The MNE ecosystem

The MNE ecosystem is composed (among other things!) of three different libraries, nested by increasing dependency:

- **MNE-Python**, which is an open-source Python package for exploring, visualizing, and analyzing human neurophysiological data mainly in EEG and MEG format. This library contains everything needed to manipulate EEG signals, from visualization to machine learning.

- **MNE-BIDS**. The BIDS format is a convention for structuring MEEG datasets. MNE-BIDS is a library that facilitates the use of this convention in python.

- **MNE-BIDS-Pipeline** is a library allowing to automate the analysis of MEEG data for datasets using the Bids format.

1.3.1 MNE-Python

MNE-Python [5] is an open-source python library allowing to analyze brain data from EEG or MEG. The figure 1.4 allows to visualize the flow of the data from the collection of the anatomical information (T1) and the raw data collected by MEEG to the estimation of the tridimensional Sources.

For the comprehension of this report, the most important steps are on the left of the figure 1.4:

- Raw data: The recording of the data is done in a continuous way during runs of 10 minutes. The recording is not stopped during this period. A run is composed of about 30 events.

- The data is pre-processed: defective sensors are removed, a low-pass filter is applied to keep only the useful brain signals, and an artifact removal technique such as ICA is used. The artifact rejection pipeline is presented in section 2.1.2.

- Epoch data: After cleaning the run, we split the run into different events. In our study, the events are either composed by intervals of three items or intervals of one item.

- Evoked data: We average the events in order to obtain exploitable results. In our case, we mainly calculate two averages, one for each number of items.

- Source Estimate: All the previous steps are carried out in the *sensor space*, that is to say in the space of the 306 sensors of the MEG, which represent 306 temporal series. On the other hand, the calculation of the *source estimate* allows to convert these temporal series (sensor space) into a 3D or 4D representation (source space).

1.3.2 the Brain Imaging Data Structure (BIDS) format

The Brain Imaging Data Structure was created to provide a simple way to organize neuroimaging data. The goal of this structure is to be a standard format to make the data accessible to everyone and avoid confusion. Using a Python package called MNE-BIDS [1], datasets can easily be converted to BIDS format. Datasets in BIDS format are supported by many tools. BIDS is the result of a collaborative effort to standardize brain data for easy analysis and sharing.

Figure 1.5 shows an example of a file architecture in BIDS format. The figure 1.6 shows an example of use in python of the MNE-BIDS library. In the absence of a standard format, to access a file, the full path must be written. A path is specific to a file, so if you want to access another file or even another dataset, you will have to change the full path. If the dataset is in BIDS format, it is possible to use the BIDSPath function of MNE-BIDS to call a file. In this case, if we want to call another file or another BIDS-compatible dataset, we only need to change a few arguments of this
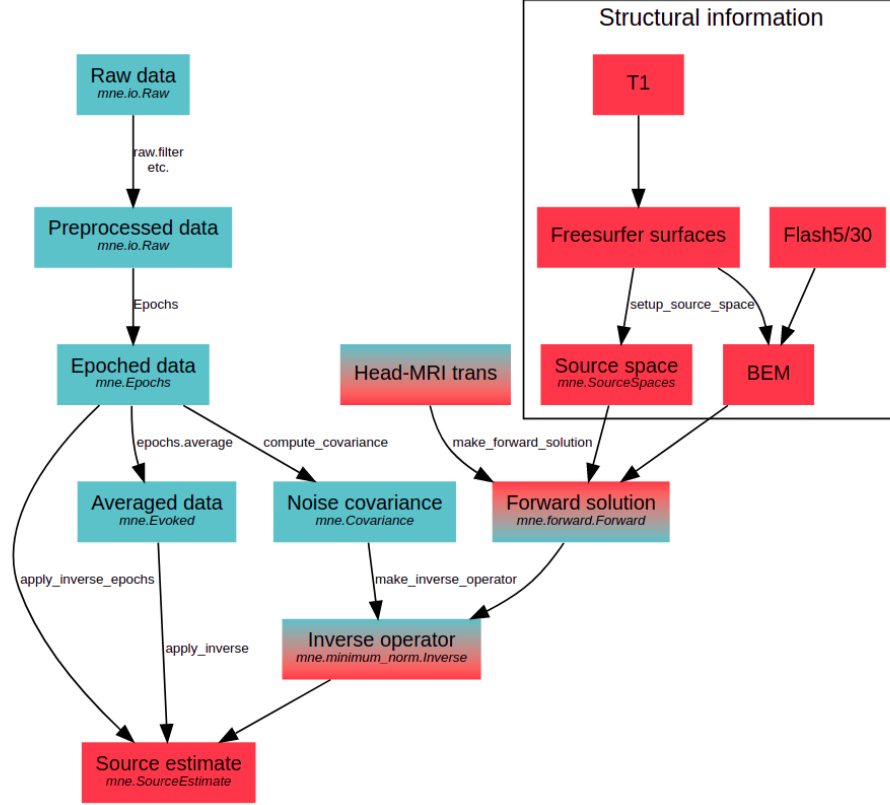
Figure 1.4: Workflow of the MNE software [5]

function, such as the subject name or the file root, without having to worry about the full path, which makes it much more modular.

Thus, by implementing the BIDS format support, it is possible to switch from one BIDS-compatible dataset to another in the blink of an eye by changing only a few elements of the scripts.

1.3.3 Automatic preprocessing with the MNE-BIDS-Pipeline

Using BIDS-compatible datasets enables us to apply the MNE-BIDS-Pipeline to the preprocessing part of the model. The MNE-BIDS-Pipeline is an automatic preprocessing and processing pipeline for MEG and EEG data stored following the BIDS format [4]. To apply the preprocessing steps wanted, the MNE-BIDS-Pipeline only requires to set the corresponding parameter in a configuration file. Once it is done for one dataset, it can be used on other datasets just by changing one line in the configuration file.

The following reasons justify the use of this pipeline:

- It improves reproducibility, as only a config file contains all the information required to reproduce the results.

- The pre-processing and analysis steps of MEEG studies are common to most studies.
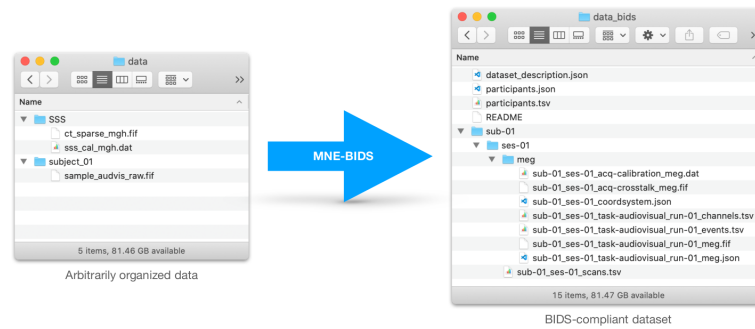
Figure 1.5: An example of the use of the BIDS format.

Figure 1.6: Example of the use of MNE-BIDS in Python: we transform the initial long path into a modular function.

- Many of the meta parameters are hard to calibrate, especially for cleaning and going to the source space. Without expertise, it is better to use the default parameters used in the pipeline.

- The pipeline allows to process the different participants in parallel, which allows to considerably accelerate the calculation times.

Chapter 2

Method

Our exploration of the MEG data for the paradigm follows the classical steps of a MEG study:

- Pre-Processing of the data. Our study started with the verification and cleaning of the data.

- Analysis in the sensor space. In this step we visualized the Evoked response of the different conditions. In order to find out when and how often the information is stored at the time of retrieval, we implemented a new procedure, based on Common Spatial Patterns (CSP). After determining the most salient time and frequency, we check the statistical significance of the response using cluster permutation tests.

- Source analysis. After having determined the frequency and the moment of maximum activation of the working memory, we visualize the 3D response.

In order to pre-process the data, and to analyze the data in the sensor and source space, I based myself on the MNE-BIDS-Pipeline, which is still under development. As I wanted to use it for the time in working memory paradigm, I had to help implement the missing features I needed. The list of features to which I contributed is presented in the table 2.1.

This chapter aims to detail the list of algorithms I implemented or improved in the pipeline to analyze our MEG data.

2.1 Preprocessing of the data

Before this internship, I had already worked on EEG data for the "Dream, sleep apnea" challenge and for the Inria - Brain Computer Interface Kaggle Challenge. But on these two occasions, the data I had manipulated had already been cleaned by the competition organizers. This internship allowed me to realize the complexity of preprocessing data in brain imaging.

Type	Title
New feature	Add possibility to exclude runs from the analysis via the new exclude run setting.
Code health	Files docstrings in the preprocessing steps were updated.
Behavior changes	Warn if using ICA and no EOG- or ECG-related ICs were detected.
New feature	Added the possibility to have different runs for different subjects.
Behavior changes	Check that the baseline interval falls into the epoch interval.
Behavior changes	ica_reject now also applies to ECG and EOG epochs.
Bug fix	The sanity check comparing the rank of the experimental data and the rank of the empty-room after Maxwell-filtering did not use the maxfiltered data.
Bug fix	epochs_tmin and epochs_tmax were named incorrectly in some test config files.
Bug fix	We now reject bad epochs by using ica_reject before producing the "overlay" plots that show the evoked data before and after ICA cleaning in the 'proc-ica_report'.
New feature	It is now possible to analyze the contrast using the Common Spatial Patterns in the time-frequency domain using the new script: 03b-time_frequency_csp.py. We also test the significance of the contrast between the two conditions using cluster permutation statistics.

Table 2.1: List of my contributions to the MNE-BIDS-Pipeline on GitHub.

2.1.1 The different steps of the preprocessing

The different phases of the data preprocessing are as follows:

- Finding bad channels: some channels of MEG recordings may be noisy or flat. They need to be identified and marked as bad so they are not taking into account during the analysis.

- Applying Maxwell filter: help to remove part of the sensors noise.

- Frequency filter: to remove non desired frequency bands.

- Creating epochs: epochs are data structures to represent equal-duration chunks of MEG signals. When the recording session includes stimuli, epochs are often defined from 0.2 seconds before the stimulus to 0.5 seconds after. Otherwise, like in rest session, epochs are fixed and overlapping frames of the MEG signal.

- Computing evoked data: they are created by averaging MEG signal over several epochs. It is useful to study stimulus-evoked brain activity.

The figure 2.1 summarizes the preprocessing step in the pipeline. At the beginning of my internship, I created this table in order to familiarize myself with the different scripts of the pipeline. Then, I added the main problems that occur in practice associated with each step, as well as the main ideas of sanity checks to be performed after the execution of the pipeline.

mne-bids-pipeline						
preprocessing						
	======================== 01. MaxWell-filter MEG data ========================	======================== 02. Apply freq filter ========================	======================== 03. Construct epochs ========================	=========== 04. Run ICA ===========	=============== 05. Apply ICA ===============	======================== 06. Remove bad ptp epochs ========================
"""description"""	Import data, find bad channels, apply MaxWell filter	Bandpass filter, resample	Construct the epochs, decimate	Fits ICA on epoched data filtered with 1 Hz highpass	EOG/ECG artifacts detected and corresponding ICA components are removed	Epochs containing peak-to-peak above the thresholds defined in the 'reject' parameter are removed from the data
Main mne method	mne.preprocessing.maxwell_filter(raw)	raw.filter()	mne.Epochs(raw)	ica.fit(epochs)	ica.apply(epochs.copy())	epochs.drop_bad(reject=cfg.reject)
Main parameters	▶ bids_root ▶ deriv_root ▶ runs ▶ subjects	▶ l_freq ▶ h_freq ▶ decim ▶ resample_sfreq	▶ epochs_tmin ▶ epochs_tmax ▶ conditions ▶ baseline	▶ ica_reject	▶ ica_reject	▶ reject ▶ conditions ▶ contrasts
File output	*_proc-sss_raw.fif *_bads.tsv	*_proc-filt_raw.fif	*-epo.fif	*_ica.fif *_proc-ica_components.tsv	*_proc-clean_epo.fif	*_proc-clean_epo.fif
Report output	NaN	NaN	NaN	*_proc-ica+components_report.html	*_proc-ica_report.html	*task-*-report.html
How to inspect the data?	mne_bids.inspect()	mne.io.read_raw_fif()	mne.read_epochs("*_epo.fif") epochs.get_data().ptp()	Check the reports	Check the reports	Check the reports
Main figure in the report						
What to look for?	- Check the number of events, number of runs - Check the number of runs in your Bids dataset	- Check out bad channels, they are easily spotted in the psd plot	- Epochs should not overlay	- Check the peak to peak amplitude: the parameter ica_reject should only reject (massive) non biological artifacts	- There shouldn't be massive artifacts anymore here	-The evoked data and the timing of the roc-auc decoding should make sense with the paradigm
Potential problems	- Config file not set properly and path incorrect -Take a moment to mark bad channels with the mne-bids inspector	- The pipeline takes a long time to run. Do not hesitate to use decimate and resample to speed up the calculations. Parallelize with N_JOBS	- Baseline not set properly - epochs_tmax too big (we see the beginning of the following event)	- Not enough iterations: augment ica_max_iterations - No IC related component detected for EOG or ECG or too much variance in the IC plot : that's probably because of artifacts: lower ica_reject		

Figure 2.1: Summary of the preprocessing of the pipeline.

2.1.2 Highlight on the management of artifacts
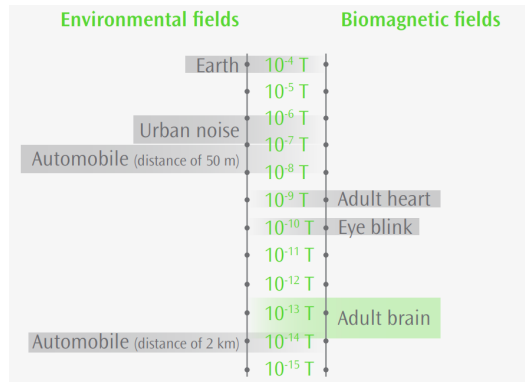
Figure 2.2: Scale of magnitude of different magnetic fields

An important point during the preprocessing is the order in which the data is rejected. Indeed, a major difficulty with raw data coming from cortical signals is the fact that these signals are extremely small. Although the data is recorded in a shielded chamber, the slightest noise can totally overwhelm the signal. Figure 2.2 shows the scale of orders of magnitude, which makes it clear that the external noise is of an order of magnitude immeasurably larger than the neurological signals of interest.

Even after removing all the noise from the external environment, some of the so-called biological artifacts, such as heartbeats and blinks, must be removed by either discarding the contaminated time interval or separating the sources by using independent component analysis (ICA), which separates the heartbeat and blink sources from the rest of the signal.

The figure 2.3 allows to visualize the pipeline rejection of the MNE-BIDS-Pipeline.

Here are the details of the procedure:

Figure 2.3: Flow of the BIDS-Pipeline rejection process.

- At the beginning the signal contains both biological artifacts (eye blink, heartbeat), and non-biological artifacts. The latter are of an order of magnitude larger than the biological artifacts.

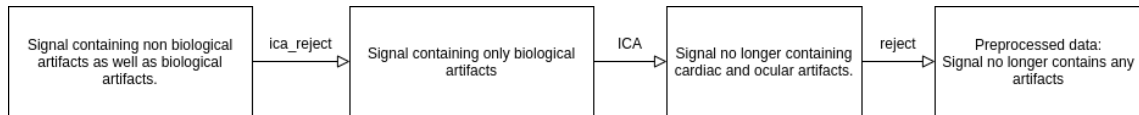- We can start by filtering the different epochs, rejecting the epochs whose peak-to-peak amplitude exceeds the amplitude specified by the "ica_reject" parameter. This parameter allows filtering signals that are 50 times larger than the nominal amplitude and therefore allows to make a first screening.

- The resulting signal does not contain any more redhibitory non-biological artifact, which allows to improve the convergence of the ICA (Independent Component Analysis), allowing to separate the sources for the ocular and cardiac artifacts.

- The ICA source separation does not work perfectly, so an additional step is required to reject the last artifacts using the "reject" parameter, which allows to filter the signals with a peak-to-peak amplitude 10 times higher than the nominal amplitude. The result is the preprocessed signal.

Even if the global pipeline rejection method was already established before my arrival, the experimental data revealed problems in the order of the filtering operations, and we thus improved a lot the automatic data preprocessing especially for difficult data that contains simultaneously biological and non-biological artifacts such as phase inversion fields. Figure 2.4 shows an example of ICA enhancement before and after our pull request that changes the order of operations, in particular by filtering out non-biological signals before the ICA.

2.2 Sensor Space: Finding the main time-frequency cluster

2.2.1 Insufficiency of the pipeline to analyze in a relevant way the MEG

At the beginning of my internship, the MNE-BIDS-Pipeline allowed us to quickly obtain figures such as 2.5 and such as 2.6. These figures show that the preprocessing went well since the results are clean. But these two figures also present some limits in the framework of our paradigm:

- The figure 2.5 presents the evoked data for each of our 6 different items. The three top figures are temporal sequences containing only one interval (1 item), while the three bottom

(a) ICA correction before amelioration

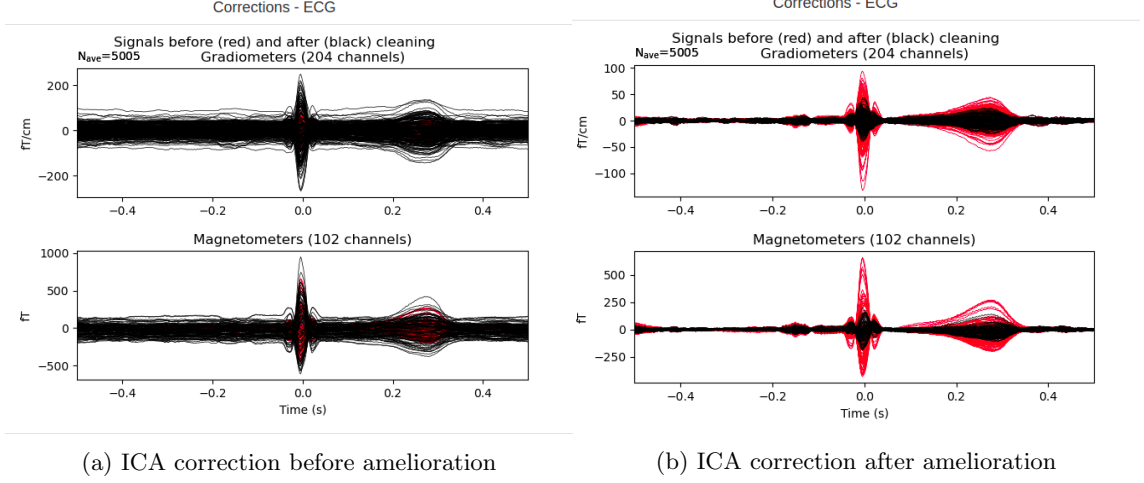(b) ICA correction after amelioration

Figure 2.4: Automatic processing of cardiac artifacts before and after our upgrade. We can see that after our pull request, the ICA is capable of reducing almost by 4 the peak to peak amplitude of the ECG artifact.

(a) 1_item_short

(b) 1_item_medium

(c) 1_item_long

(d) 3_item_short

(e) 3_item_medium

(f) 3_item_long
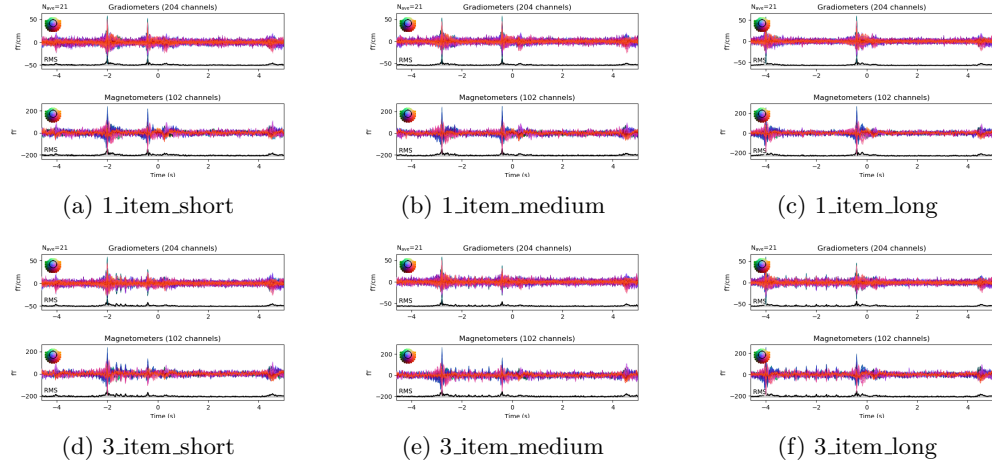
Figure 2.5: Evoked data for our 6 different items.

figures contain sequences of 3 intervals (3 items). When there is only one item, we observe two very clear peaks before $t \leq 0$. These two peaks correspond to the pure tone, which give the beginning and the end of the interval. After $t \geq 0$ begins the restitution phase, i.e., the phase during which the participant must recall the sequence in his head as precisely as possible. We see that during this restitution phase, the evoked data does not present any particular phenomenon, there is no peak that we could exploit.

- In the same way for the figure 2.6, even if we can see that the decoder used in the pipeline manages to decode the signals before $t = 0$ in a very satisfactory way, especially at the location of the auditory stimuli, the performance of the decoders is null during the restitution phase.

In this study, we are not interested in the auditory cortex, but in the working memory. We therefore do not concentrate on the listening phase before $t \leq 0$, but rather on the restitution
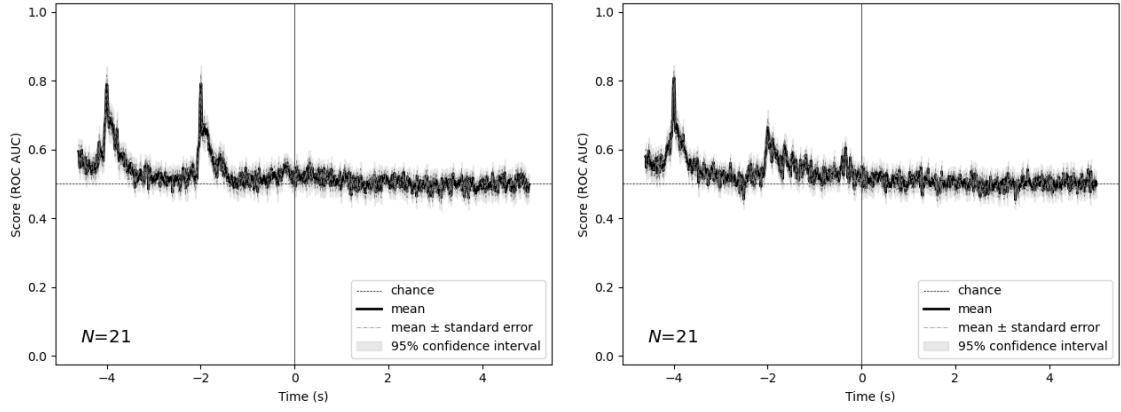
(a) Decoding between 1_item_short and 1_item_long. (b) Decoding between 3_item_short and 3_item_long.

Figure 2.6: Decoding originally in the pipeline at the beginning of my internship. We can see that after $tgeq0$, the decoding is not performing well.
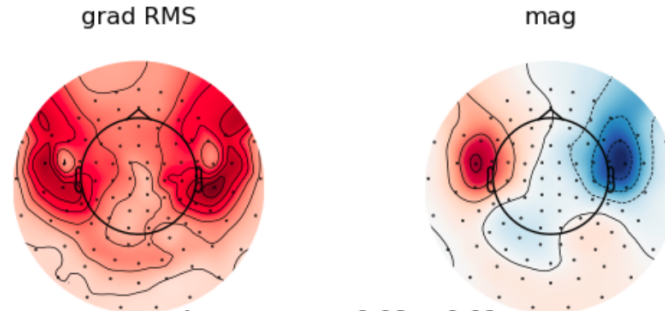
Figure 2.7: Topographic map of the first peak from 1_item_short evoked data 2.5a. We can see that during these peaks which appear on the figures 2.5, it is above all the auditory cortex which is involved.

phase after $t \geq 0$. This means that the algorithms that were currently in place in the pipeline were not sufficient at the beginning of my internship, and that we need to find an adequate method to study working memory.

2.2.2 Strategy: analyzing the contrast in time-frequency space

Classically the sensor space is decomposed in the following steps:

- Computation of the evoked data, by averaging different conditions (i.e., 1 item vs 3 item for us, see figure 2.5 in the results section)

- Using a sliding estimator with a logistic regression model for every time point (see figure 2.6 in the results section).

- Time frequency decomposition: the epoched data is transformed to time-frequency domain.

- Computation of the group average results.

This order of operations is fine for most studies. But in the case of our study, it is crucial to obtain

results in the time-frequency space. We seek here to identify the most salient time-frequencies in order to increase the signal-to-noise ratio when analyzing the contrast in source space. To fill this gap, I implemented a new script in the pipeline to visualize which time-frequency ranges are the most informative with respect to working memory. Concretely, we train a classifier to distinguish the sequences of one item and the sequences of 3 items, and we look at its performance for different frequency ranges, and different time ranges.

We remind you that our working hypothesis is that 3 items load the working memory much more than a single item. The reason for contrasting 3 items with 1 item and not 3 item with 0 item (rest) is that the contrast should not be based on a difference in the nature of the tasks, but rather a difference in load.

My pull request adds a new script to the pipeline, designed to analyze in the time-frequency domain a contrast between two conditions. There are two main steps in this script:

1. **Building the time-frequency map using CSP**: for each time-frequency bin, we use a CSP classifier in order to distinguish between the two conditions. We compute the ROC-AUC score for each time-frequency bin.

2. **Permutation statistics at the group level**: we try to answer the following question: is the difference between the two conditions statistically significant? We use the classic permutations cluster tests on the time-frequency ROC-AUC map.

2.2.3   Building the time-frequency ROC-AUC map using CSP

Common Spatial Pattern transformation is based on a geometrical analysis of the brain response, which is called a *pattern*. CSP is used to identify the most salient time-frequencies. We will use CSP to classify between 1-item events and 3-item events. To be able to classify between these two items is to be able to recognize a working memory with two different loads.

We use CSP to identify the most salient times and frequencies for our task: we just have to look at which frequency interval and which time interval our classifier performs best. A good discrimination between the two conditions (1 item vs. 3 items) is characterized by a ROC-AUC close to 1 while a ROC-AUC close to 0.5 means that the neural code that encodes the working memory is not detectable by high level geometric features.

2.2.4   Decoding with Common Spatial Patterns (CSP)

CSP is a technique to analyze multichannel data based on recordings from two classes that was introduced in the EEG context by [7].
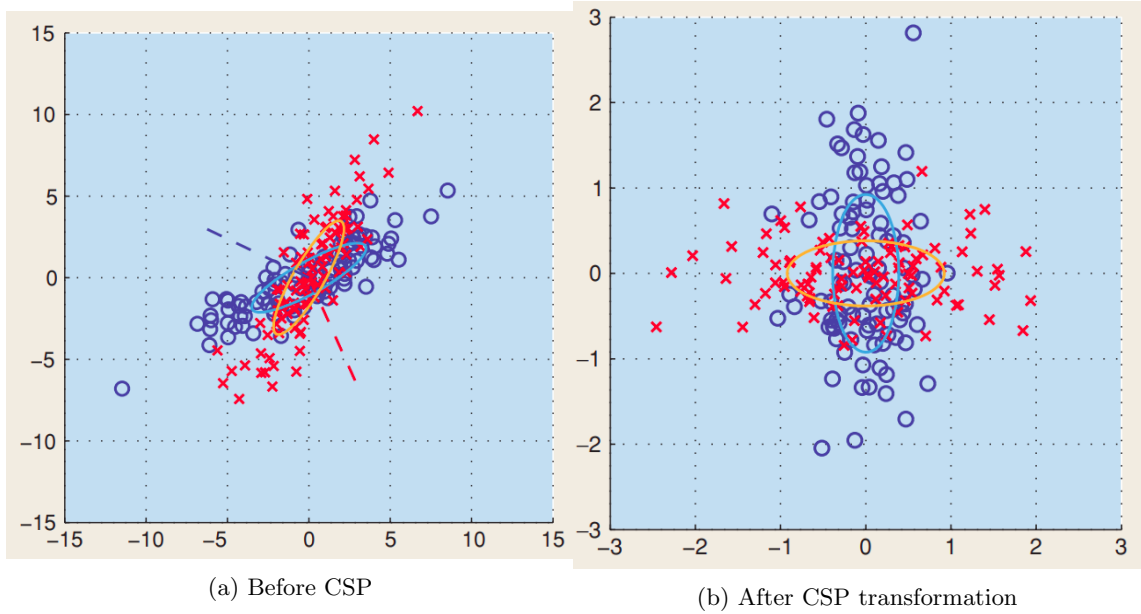
(a) Before CSP   (b) After CSP transformation

Figure 2.8: Toy example showing the functioning of the CSP transformation.

Intuitive explanation

I took the figure from [2] which also contains a comprehensive tutorial on CSP.

If we have two Gaussian signals, both centered on the zeros, but with different principal directions, we can try to transform the two point clouds so that the principal directions of the two point clouds in CSP space are orthogonal in order to maximize the ratio of variances along the principal directions.

For example on the figure 2.8, we start from two point clouds, red and blue, very correlated, which we transform into decoupled red' and blue' point clouds.

The main vector of the red' cloud is aligned on the x-axis, while the main vector of the blue' cloud is aligned on the y-axis.

By this process, we maximize the variance ratio along the x-axis between the red' and the blue' clouds.

In our usage, a point cloud corresponds to an event, and each color corresponds to a different condition. Here, we do not try to classify points individually as in machine learning, but we classify a whole point cloud. If we try to guess the color/condition of a point cloud, we just have to transform the point cloud with the same transformation as before (i.e., apply the unmixing matrix as we will see later), and take the projection on the x-axis of the new point cloud. We can then calculate the variance along the x-axis and then train a linear classifier to discriminate among the different variances. By combining the CSP algorithm, and for example a logistic regression, we can classify between the two conditions.

Technical Description (to be skipped in a first reading)

The above explanation only explains intuitively the algorithm for the first component of the CSP. But it does not explain how to obtain the other orientations. An efficient formulation to obtain all components of the CSP while being computationally reasonable is to formalize the problem as a generalized eigenvector problem.

General Eigenvalue Problem Formulation   In this section, I use the same notation as in [8]:

Let $X \in \mathbb{R}^{C \times T}$ be a matrix containing C channels and T time points. The data at a single time point is denoted by $x(t) \in \mathbb{R}^C$. Common spatial pattern (CSP) is a transformation that projects the signal in the original sensor space to CSP space:

$$x_{\text{CSP}}(t) = W^T x(t) \tag{2.1}$$

where each column of $W \in \mathbb{R}^{C \times C}$ is a spatial filter and each row of $x_{\text{CSP}}(t)$ is a CSP component. Let $\Sigma^+ \in \mathbb{R}^{C \times C}$ and $\Sigma^- \in \mathbb{R}^{C \times C}$ be the respective covariance matrices of the two different conditions. CSP analysis is given by the simultaneous diagonalization of the two covariance matrices.

$$W^T \Sigma^+ W = \lambda^+ \tag{2.2}$$

$$W^T \Sigma^- W = \lambda^- \tag{2.3}$$

Where the two λ are diagonal matrices whose entries are the eigenvalues of the following generalized eigenvalue problem in $(w, \lambda) \in (\mathbb{R}^C, \mathbb{R})$:

$$\Sigma^+ w = \lambda \Sigma^- w \tag{2.4}$$

Large entries in the diagonal matrix corresponds to a spatial filter which gives high variance in one class but low variance in the other. Thus, by projecting the cloud of points on the first components, the filter facilitates discrimination between the two classes.

The unmixing matrix   Another way to read the two equations 2.2 and 2.3 is to remember that $w^T \Sigma w$ computes the variance of the covariance Σ along the w direction. Thus, $W^T \Sigma^+ W$ and $W^T \Sigma^- W$ just compute diagonal matrices, whose diagonal elements are the variances of the covariances along the different columns of W, namely the variances along the different spatial filters.

By finding all the w which satisfy the equation 2.4, we construct the matrix W also called *unmixing matrix*.

After having found all the couples (w_i, λ_i) which satisfy the generalized eigenvalue equation 2.4, we can order the solutions by decreasing eigenvalues. The first filter corresponds to the direction maximizing the ratio of the variances between the two scatter plots. Indeed, if we replace in the equation 2.2, 2.3, we obtain:

$$w_i^T \Sigma^+ w_i = \lambda_i^+ \tag{2.5}$$

$$w_i^T \Sigma^- w_i = \lambda_i^- \tag{2.6}$$

And it is then enough to replace in 2.4 then to multiply on the left by w^T to obtain

$$\lambda_1 = \lambda_1^+ / \lambda_1^- \tag{2.7}$$

2.2.5 Significance of time-frequency bins with Permutation statistics

We try to answer the following question: is the difference between the two conditions statistically significant? We use the cluster permutations tests on the time-frequency ROC-AUC map in order to check the significance of the activation.

Overview of the cluster permutation statistics method
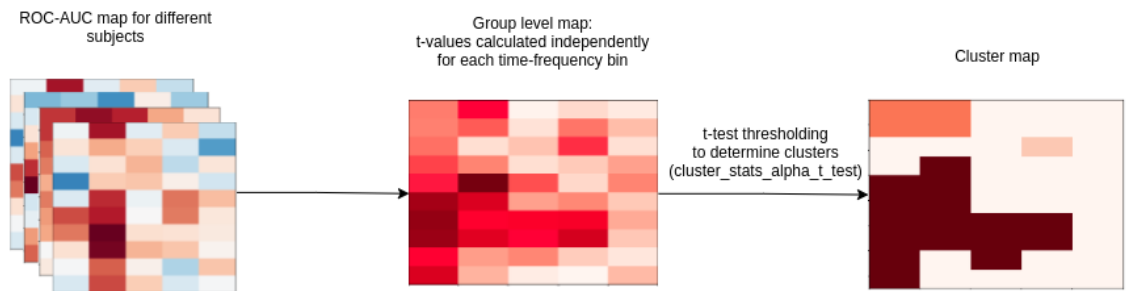
Figure 2.9: Overview of the permutation statistics pipeline.

The figure 2.9 gives an overview of the cluster creation pipeline.

1. We start by collecting all the time-frequency ROC-AUC maps of all the subjects.

2. For each time-frequency bin, we compute a t-value, independently for each bin (the details of the computations can be found in the Appendix). We then obtain the t-value map.

3. Then, in order to find the location of the clusters, we use a threshold, for example corresponding to a chance level of 0.05 or 0.01. This threshold is not very important mathematically,

but in practice, it allows to control the size of the clusters. Once the threshold is applied, we potentially get one or more clusters.

In order to assign a p-value to each cluster, we have to use the permutation mechanism. The permutation mechanism consists in computing a metric on our clusters: which can be either the size of the cluster, or the maximum of t-values within the cluster, or the sum of t-values within the cluster. In our case, we have chosen to sum the t-values on the cluster because we are not interested in the spread of the cluster, but rather we want to target precisely a region located in the time-frequency map. Then we compare this metric to the distribution of this metric simulated for the permutation (simulation of the null hypothesis): We reverse the sign of the difference for each bin in each subject. We obtain the distribution of the metric in the null hypothesis, where the distribution of spatial cluster sizes is independent of the sign of the data.

2.3 Source Space Analysis

Following the extraction of the most informative time and frequency ranges, we can now study the anatomical location of the brain response. To do this, we must visualize the contrast in a three-dimensional space corresponding to a model of the brain, which is called the source space.

2.3.1 Contrast calculation for one subject

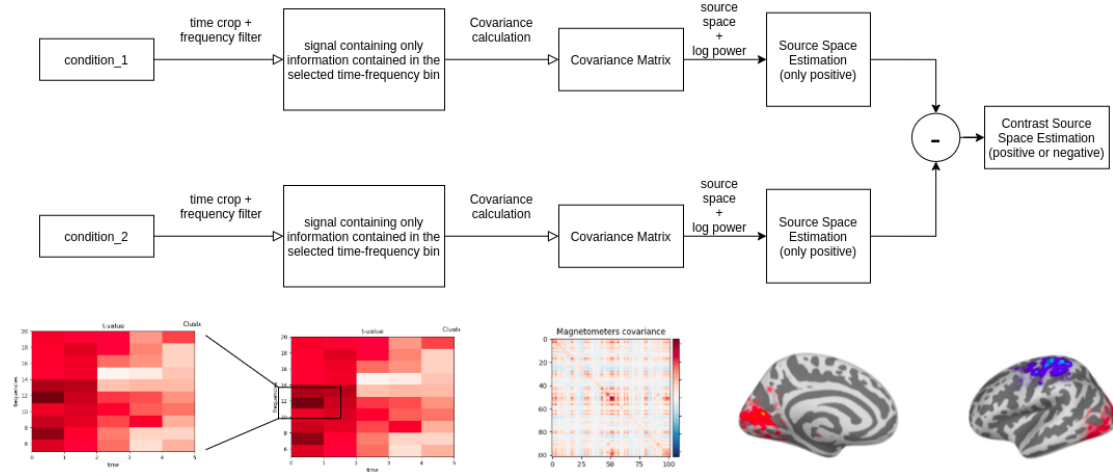Figure 2.10: Schema of the procedure to visualize the contrast in source space.

The figure 2.10 explains the procedure to obtain the contrast in the source space for one subject:

- We start with two different conditions, i.e., the sequences of one item and three items.

- We filter the data to keep only the most informative times and frequencies, in order to increase our signal-to-noise ratio.

- We compute the covariance matrix of each condition.

- We project the covariance matrix of each condition in the source space (see dSPM method, in section [?]). We then obtain a three-dimensional activation map called a *source estimate*.

- We then subtract the log of the power associated with the two source estimates.

2.3.2 Group average

Once we have the contrast for a subject, we need to average the 3D contrast for different subjects. But we cannot take the average between the different subjects immediately because the different subjects do not have the same head shape. Before taking the average of a subject, we morph all the head shapes to an average head shape, which we call *fsaverage* (fs stands for freesurfer, the software that segments the fMRI images, which allows to create the electromagnetic conductivity model of the brain tissue, and the boundary element model, see annex B.1). We can then calculate the average contrast in the source space.

Chapter 3

Results

In this chapter, we present the results of the method explained previously.

1. Sensor Space: We start by studying the data in the time-frequency space to find the most important times and frequencies ranges:

   (a) The classification by CSP allowed us to obtain a time-frequency map, which allows associating to each time and frequency a ROC-AUC. This ROC-AUC indicates to what extent the information stored in the working memory is decodable by geometric information.

   (b) The permutation statistics allow to extract the most informative time-frequency cluster, as well as to check the significance of the results.

2. Source space: After having verified the statistical significance of our cluster, we use this cluster to project the information in a three-dimensional space, (aka the source space), in order to visualize the contrast between the working memory loaded with 3-items versus 1-item. It is mainly this step that allows us to extract the neuroanatomical information.

3.1   Sensor space results

3.1.1   CSP results

In the figure 3.1, we observe the ROC-AUC score of the classifier composed of (CSP, logistic regression) for different :

- frequency ranges distributed linearly between 5 Hz and 20 Hz as well.

- time ranges in the reproduction interval, i.e., between $t = 0$ and $t = 5$ seconds (the figures
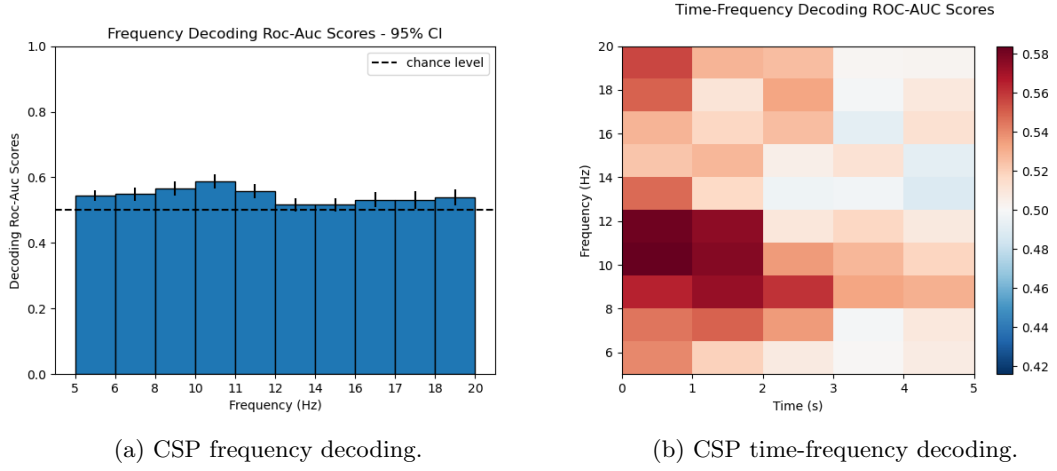
(a) CSP frequency decoding.　　　　(b) CSP time-frequency decoding.

Figure 3.1: CSP results for the average subject.

presented the times $t \leq 0$ only for pedagogical purposes).

We can see that the CSP algorithm succeeded in highlighting the alpha frequency band between 8 and 12 Hz, and the results are consistent between the frequency map 3.1a and the time-frequency map 3.1b, with both a maximal score around 10 Hz.

3.1.2　Cluster Permutation Test results
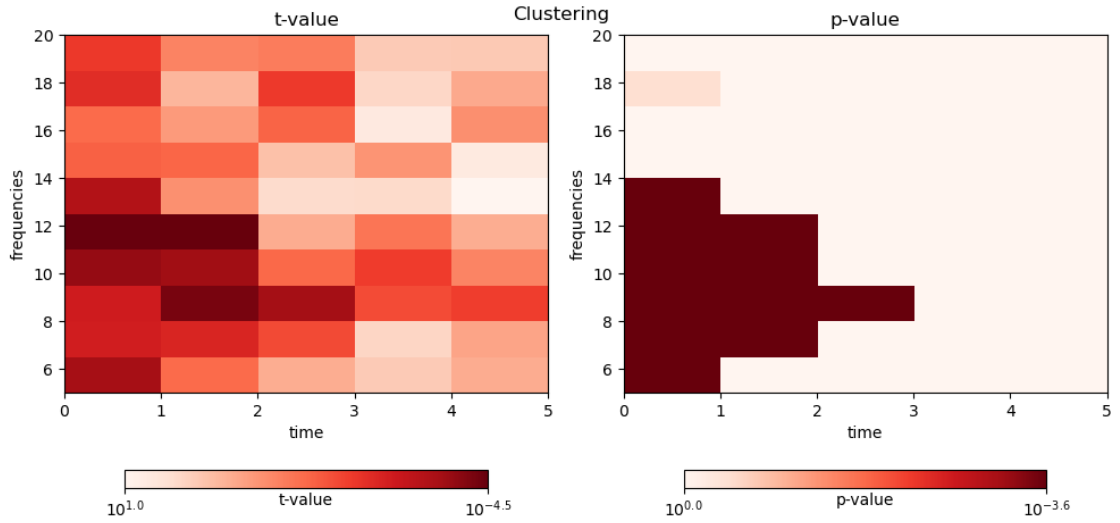
Figure 3.2: Permutation statistics results.

The figure 3.2 presents the results of the permutation test cluster. We obtain two clusters of p-values, in low alpha frequency (6 Hz to 14 Hz) and beta (17 Hz-18.5 Hz), of p-values $10^{-3.6}$ and 0.12 respectively. These clusters indicate that it is necessary to filter on the alpha band at the beginning of the reproduction phase in order to maximize the signal-to-noise ratio when studying the contrast in source space.

3.2 Sensor results discussion

Quantitative analysis An AUC of 0.6 is not high in absolute terms, but not bad in EEG analysis on intervals of only 0.5 seconds. If we combine all the bins by bagging, we would get a much better AUC. Furthermore, statistics by clusters permutation show that the results are very significant at the group level.

Qualitative analysis Our main cluster is centered on the alpha band. We will discuss the alpha band in more detail in section 3.3. But we already know from the literature that the alpha band is key to the working memory loads [10], which is encouraging.

A surprising phenomenon is that the algorithm can decode much more easily at the beginning of the restitution phase than at the end of the restitution phase. Given that the CSP encodes geometric information, this means that:

- either the information in the working memory is transformed in the course of restitution into information disseminated in the brain in a non-geometrical way.

- Or, given the very large number of repetitions of the experiment, the subject gets tired after a while and forgets the sequence "instantly" after having repeated it in order to save mental energy.

3.3 Source space results
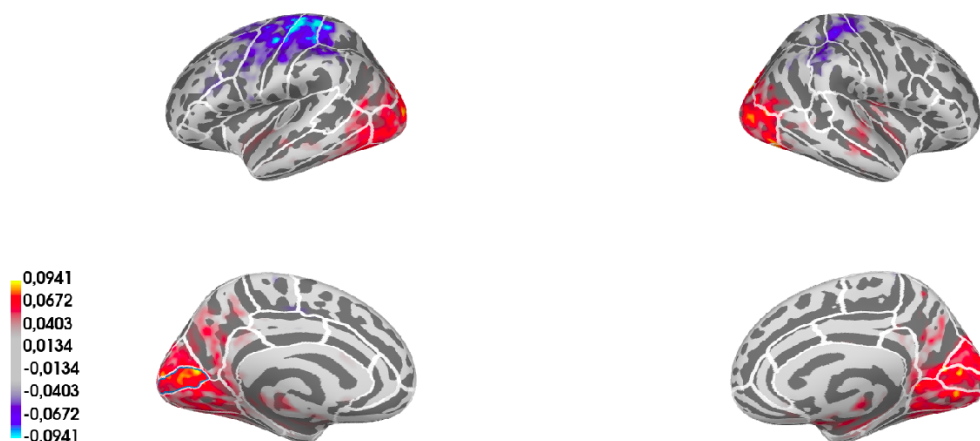
Figure 3.3: Contrast results in source space for 0s-1s, and 8Hz-14Hz (alpha).

The figure 3.3 presents the results of the contrast in source space for the most significant time-frequency bin at the group level, i.e., 0s-1s, and 8Hz-14Hz. In this figure, the red regions are the regions activated for 3 items, while the blue regions are more activated for 1 item. We recall that 3 items are supposed to load the working memory, more than 1 item.
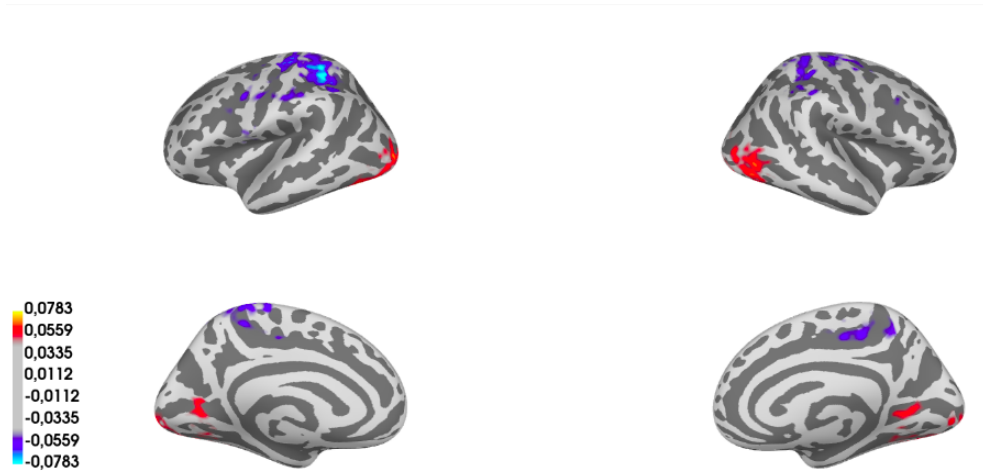
Figure 3.4: Contrast results in source space for 0s-1s, and 15Hz-20Hz (beta).

Occipital cortex (visual)   The red region activated here belongs to the visual area. The fact that we see the visual area here is not surprising: we have to remember that we are visualizing here the alpha band, which is a classically inhibitory band in the literature. Thus, the classical interpretation of an activation in the alpha band corresponds to an inhibition of the function of the associated region. This is confirmed by the fact that when we look at the activation in the beta 3.4 of our contrast, the activation of the visual cortex (the red region) disappears almost completely.

An increase in the alpha band was an expected classical effect: the literature presents other results similar to this one. For example, [10] shows that "Enhanced alpha oscillations (8-13 Hz) during retention of items in working memory are often interpreted to reflect increased demands on storage and inhibition".

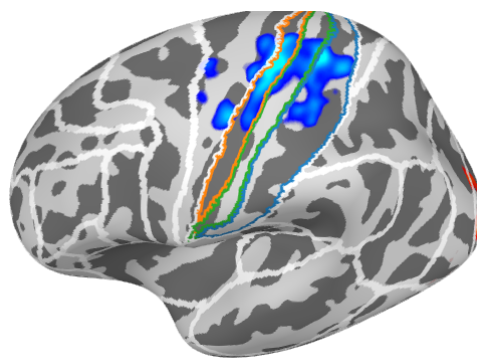Figure 3.5: Focus on the Brodmann areas 1 (green), 2 (blue) and 3 (orange), from the alpha band in the left hemisphere.

Sensorimotor Areas   The blue region activated at the top of the brain is the sensorimotor area, centered on the postcentral gyrus (Brodmann areas 1, 2 and 3). Before the MEG experiment, subjects were trained to reproduce the task by pressing a button, which explains the activation

here. And the region of the somatosensory cortex activated here corresponds to the location of the hand receptors 3.5. This is also supported by the fact that the activation is stronger in the left hemisphere, which is consistent with the fact that the subjects are right-handed.

A possible interpretation is that time is represented by the preparation of the movement of one of the body parts, here the hand. We can also reason by analogy by noting that musical children learn to master rhythms by beating the beat or walking to the rhythm of the pulse. The rhythmic information is embodied. Therefore, even if here the analogy has limits since the temporal signal is not organized in a rhythmic and regular way, this interpretation seems to us to be the most natural one.

Auditory area There is no evidence of activation in the visual regions while the cue is an auditory one. There are several interpretations: — To see an activation in the auditory zones one must place oneself at $t \leq 0$, as on the figures 2.7. But here, during the reproduction phase ($t \geq 0$), the auditory cortex is no longer involved. — The signal is abstracted from the auditory region to transform it into a purely rhythmic information, thus no longer restricted to the auditory areas. — The auditory region is activated, but we do not see it in this group figure. As a matter of fact, the auditory regions are visible in the CSP components of the subjects as in the figure A.2b. But there is a lot of inter-subject variability: After looking at the CSP components for the different subjects, it appears that only subjects with a marked beta peak show activation in auditory cortex, and this peak is not restricted to $0 \leq t \leq 1$ but are spread across all subjects between $0s - 5s$. In order to see an auditory signal at the group level, one would have to align and average the first extracted components for each peak from each subject. But averaging the components presents mathematical difficulties (see in annex A.1.3), and could be a research topic in itself.

Conclusion

In conclusion, we exploited a new experimental paradigm using minimalist sequences of empty intervals with MEG recordings to investigate the neuroanatomy and neurodynamics of working memory durations. Our analyses highlight the importance of the alpha band, somatosensory areas and occipital cortex in working memory. To obtain these results, we integrated the algorithms for time-frequency analysis and visualization of the contrasts in the source space into an open-source pipeline. In order to make the pipeline usable and practical, considerable work has been done in optimizing the running time of the pipeline, combining advanced software engineering techniques with techniques from statistical learning theory. The implemented algorithms allow to combine the performance needed to analyze data of such dimensions with the interpretability required for the analysis of brain signals.

References

[1] Stefan Appelhoff et al. "MNE-BIDS: Organizing electrophysiological data into the BIDS format and facilitating their analysis". In: *The Journal of Open Source Software* 4.44 (2019).

[2] Benjamin Blankertz et al. "Optimizing spatial filters for robust EEG single-trial analysis". In: *IEEE Signal processing magazine* 25.1 (2007), pp. 41–56.

[3] Inderjit S Dhillon. "Alan Kaylor Cline". In: ().

[4] Krzysztof J Gorgolewski et al. "The brain imaging data structure, a format for organizing and describing outputs of neuroimaging experiments". In: *Scientific data* 3.1 (2016), pp. 1–9.

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

[6] Sophie Herbst et al. "Abstracting time in memory". In: (2021).

[7] Zoltan J Koles, Michael S Lazar, and Steven Z Zhou. "Spatial patterns underlying population differences in the background EEG". In: *Brain topography* 2.4 (1990), pp. 275–284.

[8] Zoltan Joseph Koles. "The quantitative extraction and topographic mapping of the abnormal components in the clinical EEG". In: *Electroencephalography and clinical Neurophysiology* 79.6 (1991), pp. 440–447.

[9] Wei Ji Ma, Masud Husain, and Paul M Bays. "Changing concepts of working memory". In: *Nature neuroscience* 17.3 (2014), pp. 347–356.

[10] Jonas Obleser et al. "Adverse listening conditions and memory load drive a common alpha oscillatory network". In: *Journal of Neuroscience* 32.36 (2012), pp. 12376–12383.

[11] Cyril Pernet et al. "Best practices in data analysis and sharing in neuroimaging using MEEG". In: (2018).

[12] Frédéric Roux and Peter J Uhlhaas. "Working memory and neural oscillations: alpha–gamma versus theta–gamma codes for distinct WM information?" In: *Trends in cognitive sciences* 18.1 (2014), pp. 16–25.

Appendix A

Sensor annex

A.1 CSP algorithm

A.1.1 CSP running time optimization

The optimization of the computation time does not change the neuroanatomical results presented, and therefore is not integrated in the "method" part, but still required a considerable amount of work.

My first implementation of the CSP algorithm took a whole day to compute for a single subject, which corresponds to a 15 days computation for our cohort. It was therefore crucial to find ways to greatly accelerate the calculations, which will then be reiterated at each commit of the pipeline as part of the continuous integration with Circle CI.

A great deal of research and engineering effort was then put into reducing the calculation time.

Classic software engineering techniques Classic software engineering techniques have been key to massively reduce computation times: factoring of all steps, use of tools such as the line-profiler — which allows to visualize the execution time of each line — use of multi-processing. On the other hand, even if mathematically some operations are commutative, in practice, the choice of the order of the operations allows saving computation time: for example even if the operations (time crop, frequency filter) are commutative, it is better to start by cropping the data and then filtering, because the filtering operation is very expensive.

Dimensionality reduction Once all the software improvements have been made, we can also make improvements coming from machine learning techniques by using dimensionality reduction techniques: We can either select the type of channels, or use a PCA.
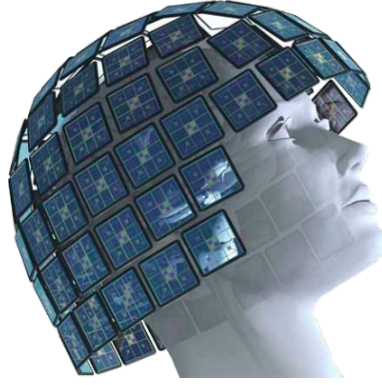
Figure A.1: MEG Sensors [Elekta documentation]

- Reduction of dimensionality by selection of the channel type: The MEG data comes from 102 sensors, which are distributed as shown in the figure A.1. Each sensor consists of 2 gradiometers measuring the gradiometric field (Tesla/meter) in the direction tangent to the sensor plane, and a magnetometer measuring the electric field in the direction normal to the sensor plane. Magnetometers are robust to external noise, while magnetometers are more exposed. But magnetometers have two crucial advantages: 1. they record localized spatial information, which is perfectly suited to the CSP algorithm, which subsequently favors the interpretability of patterns from the CSP. 2. The magnetometers are half as numerous as the gradiometers.Thus, by selecting only the magnetometers, the number of channels is divided by three. The CSP algorithm being limited by an SVD whose complexity in $\mathcal{O}(Tn^2)$ evolves as a function of the square of the number n of channels [3], we thus accelerate the algorithm by a factor 9. (T is the number of time point, in our case bigger than the number of channels. The complexity of the SVD internal to the CSP is in the square of the minimum between T and n).

- The following point applies only to data from a MEG, but not to EEGs. Indeed, EEGs have only one channel type. We can therefore reduce the dimensionality using a simple PCA.

Sampling The last way to accelerate the computation time is to use sampling, i.e. to reduce the sampling frequency. This is also called decimation. For example, $decim = 5$ consists in taking only one point on 5. But sampling can introduce numerical instabilities. Indeed, the CSP algorithm must start by estimating the covariance matrix of the signal. However, the estimation of a covariance matrix of dimension n requires at least approximately n points. With a sampling frequency of 100 Hz, over a time window of 0.5 seconds, we have 50 points. It is therefore necessary to reduce the dimensions to at most 50 or to use covariance regularization methods, presented in section A.1.2.
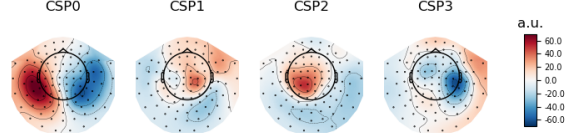
(a) Example of CSP components (alpha band)
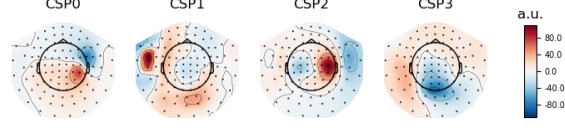
(b) Example of CSP components (beta band)

Figure A.2: Example of CSP components from the alpha and beta band.

CSP running time optimization results The choice of the order of commutativity, the reduction of the dimensionality and the resampling allow us today to obtain results in less than three hours with 8Cross validation for all subjects. These three techniques allow us to respectively accelerate by $2 * 10 * 5$ on our dataset while keeping similar performances in all points. It even seems that the PCA increases the numerical stability of the results.

A.1.2 CSP Regularization

In order to regularize the estimation of the covariance matrix, many strategies exist, notably based on Riemannian geometry. But in our case, it is enough to use a regularization of diagonal elements of the covariance matrix, in order to obtain strictly positive eigenvalues. Thus, it is sufficient to replace the empirical estimate C of the covariance matrix by :

$$C' = C + \sum_k \varepsilon_k \bar{\sigma}_k^{\ 2} I^{(k)} \ , \tag{A.1}$$

with k, the index of the considered channel type, ε_k the regularization coefficient, σ_k the average standard deviation of the considered channel type, $I^{(k)}$ the matrix with 1 diagonal matrices containing ones at the positions corresponding to the channels contained in each channel group.

We used $\varepsilon_k = 0.1$.

topographic map, right-hand rule

The visualization of CSP patterns makes it one of the most interpretable algorithms. But in the context of MEGs, one must keep in mind that the activations are done by magnetic dipole. Thus, in the CP0 of the figure, by using the right-hand rule, we deduce that the magnetic field enters inside the right ear.

A.1.3   Mathematical subtleties

Why not aiming for the best classifier?

Mathematically, we are not looking here to create the best classifier, and to optimize the ROC-AUC score, we are only looking to obtain unbiased scores that can be used later in the permutation test. This is why it is not a problem to optimize the execution time.

Moreover, in neuroscience, interpretability is much preferred to performance. And the CSP algorithm is ultra interpretable thanks to the visualization of patterns. Since our results are already significant with CSP, there is no need to struggle any further. The CSP algorithm is already one of the best compromise between interpretability and performance.

Choosing the t-value threshold

The fact that you can choose the size of the clusters simply by adjusting the threshold of the t-value is disturbing at first sight. But in reality, it is a good sign. We only tested two different t-values (0.01 and 0.05) and the fact that we get two different cluster sizes simply means that our 0.05 cluster has a significant difference as well as our 0.01 cluster, which is also significant. One does not exclude the other.

Choosing a t-value threshold of 0.01 we obtain the figure 3.2, with a cluster having a p-value of $10^{-3.6}$. I chose this restrictive threshold of 0.01 for my final image in order to limit the area of my cluster and to limit it to the most significant areas. But it is possible by choosing a threshold of 0.05 to obtain a figure with a cluster going through the whole alpha band from 0 to 5 seconds.

However, the obtained cluster has no physical reality and could be extended to the whole image if we had more subjects. We must therefore take a cluster of reasonable size in order to maximize the signal-to-noise ratio when moving to the analysis of sources.

Difficulties concerning the calculation of an average of the CSP components

We have thought about the feasibility of calculating the averages of the CSP components in order to obtain "average" components shared by all subjects. But some difficulties make these averages unfeasible:

- Although the CSP components are ordered by decreasing eigenvalue, there is a great risk of mixing CSP components across different anatomical regions for different subjects. To mitigate this risk, one would have to do a CSP decomposition for each brain area, which would complicate the operation.

- Using the same CSP components for everyone would necessarily lower the decoding results.

Considering that our subjects probably use different memorization strategies, these different strategies would be "mixed" in a group decoding, and would harm the interpretability.

A.2   Cluster permutations statistics algorithm

A.2.1   t-values calculation

This test rejects the null hypothesis that the mean of a population of independent observations is equal to a value M.

Hypothesis   The t-test is a parametric test that assumes the Gaussianity of the underlying values. This assumption of Gaussianity is not always true for data from brain imaging because of the many preliminary filters. But in our case, we compute t-values from the difference between the ROC-AUC score and the average chance level, which allows us to restore the Gaussianity assumption.

Intuition   The calculation of p-values is done for each time frequency bin independently.

We consider the list of differences $(X_i)_{i \in [\text{Nb Subjects}]}$ for all subjects between the ROC-AUC and the chance level $M = 0.5$ for a bin of temporal frequency. We obtain the set of differences between the ROC-AUC and the chance level. A normal distribution of these differences is made. If the normal distribution is sufficiently to the right of 0, then we reject the null hypothesis.

Algorithm   We want to compare the mean μ of a population with a normal distribution and an unknown standard deviation σ to 0. To do this, we compute the empirical mean $\overline{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$ and the unbiased estimator S_n^{*2} of variance σ^2

$S_n^{*2} = \frac{1}{n-1} \sum_{i=1}^{n} (X_i - \overline{X}_n)^2.$

According to the null hypothesis, the sampling distribution of this mean is also normally distributed with a standard deviation $\frac{\sigma}{\sqrt{n}}$.

The test statistic:

$Z = \sqrt{n} \frac{\overline{X} - \mu_0}{S_n^*}$ then follows a Student's law of $n - 1$ degrees of freedom under the null hypothesis.

We choose a risk α, generally 0.05 or 0.01 and we calculate the realization of the test statistic :

$z = \sqrt{n} \frac{\overline{x}_n - \mu_0}{s_n^*}$, where $s_n^* = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \overline{x}_n)^2}$

A.2.2 Choosing the number of time-frequency bin

Choosing the number of Bin is somewhat arbitrary. One just has to respect some conditions such as Nyquist, plus the fact of having enough points to estimate the covariance matrix. But here, it is not a problem to choose arbitrary the number of points, since the result of the sensor space is only used to increase the signal-to-noise ratio when going into the source space.

Appendix B

Source annex

B.1 Dynamic statistical parametric mapping (dSPM)

Method Dynamic statistical parametric mapping allows to obtain a three-dimensional image of the sources by taking into account the activation of the baseline. For each three-dimensional point, a normalization by the noise variance is performed. The resulting image gives the activation map, made of positive intensities only.

Choice of covariance matrix In order to calculate the source estimate by dSPM, we need to choose a covariance matrix, in order to know the covariance of the signal not carrying the information. The choice of the covariance matrix is therefore not a mathematical problem, but a choice that follows from our experimental paradigm. We must answer the question: what constitutes noise? Here are the two main choices of covariance matrix:

- Use the empty room covariance. This is the covariance matrix associated with sensor noise when there is no one in the room. This is the default choice in the MNE-BIDS-Pipeline.

- Use the pre-stimulus covariance. Indeed, before the cue, the participant does not yet know the duration of the cue. This is the method we recommend.

B.2 Boundary Element Model (BEM)

Figure B.1 presents the boundary element model used to model one of our subject's heads. Our participants were recorded by MRI before their MEG recording in order to obtain T1 anatomical images. These T1 images are used to create a 3D model of the brain and to model each area of the brain with different electromagnetic conductivity. This model is then used to calculate the inverse
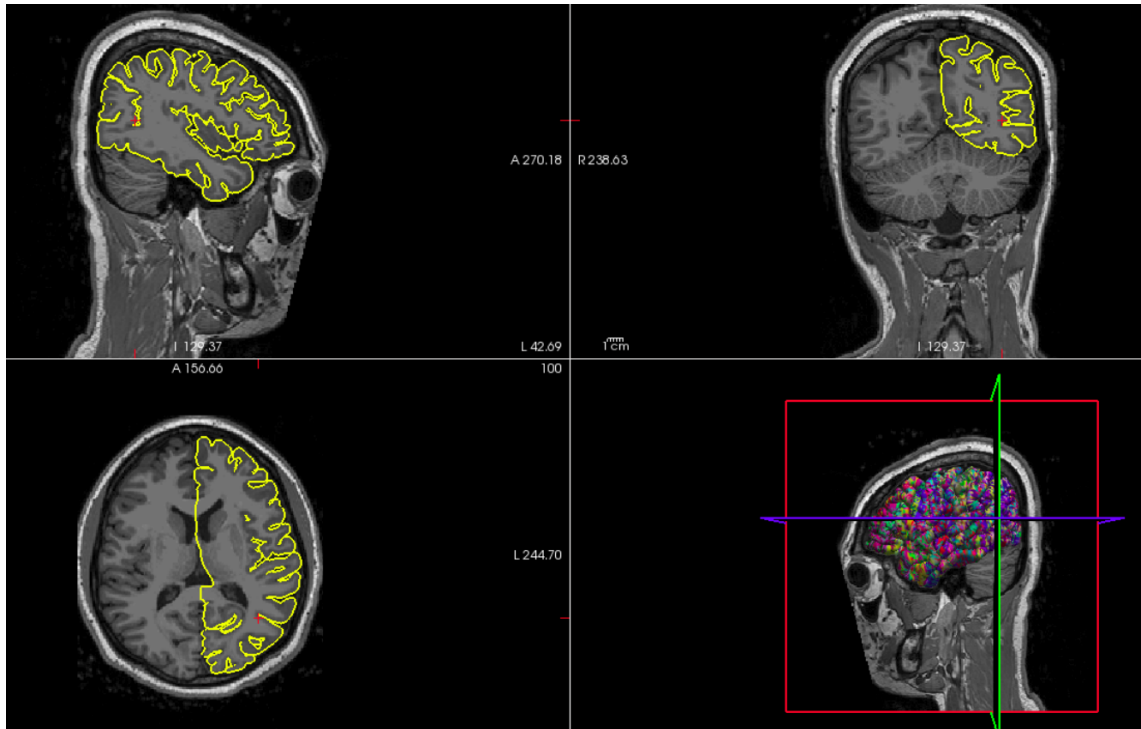
Figure B.1: BEM model used for one of our anonymized subjects.

operator to find the location in the brain of the electromagnetic sources from the MEG recording.

We can see on the figure B.1 that the face of the person has been cut in order to anonymize the data. We can see on the figure the result of the segmentation between the white matter and the gray matter. The results of the algorithm, although satisfactory, are not perfect. But this is of little importance because MEG is a relatively robust technique with respect to the imperfections of electromagnetic conductivity modeling. For example, for EEG we use generally 3 layers (inner skull, outer skull, and skin) while for MEG 1 layer (inner skull) is enough: The MEG is said to be "transparent".