Pain-Decoding through fUSI brain signals for more accurate BCI

Human-Computer Interaction, Utrecht University, NERF lab

Author	**Marina Batlló Rius**
Supervisors	Théo Lambert, Egon van den Broek, Alan Urban

Introduction and Literature review

1. Human-Computer Interaction and Brain-Computer Interfaces

Human-Computer Interaction (HCI) is the study of how people interact with computers and other technologies. It is a broad study that involves the design, evaluation, and implementation of interactive systems and technology that are effective and easy to use. The concept was introduced with the first general-purpose computer, the Electronic Numerical Integrator, and Calculator (ENIAC), in 1946. Rising with the appearance of Graphical Computer Interfaces, it has developed into a broad and growing field with applications in nearly all technological devices (Sinha et al., 2010). As technology is nowadays everywhere, the interaction between it and humans plays a very important role. Although this interaction can be usually understood as a person using a PC, many different ways of communication have appeared. On this line, Brain-Computer Interfaces (BCI) have become of interest for different applications.

BCIs are communication systems designed to allow external technological devices, such as computers, to be controlled through brain activity (Nicolas-Alonso & Gomez-Gil, 2012). Many different implementations in HCI have been appearing with the growing interest in the technology. Some remarkable examples of this are BCIs for games (Gezgez & Kaçar, 2021; Glavas et al., 2022; Nijholt et al., 2008), education through or of technology (Gezgez & Kaçar, 2021; Glavas et al., 2022; Hernandez-Cuevas et al., 2020), or user experience evaluation (Cano et al., 2022a; Diya et al., 2019).

However, one of the main characteristics that differentiate BCIs, is that they do not involve motor movement, making them a powerful tool to offer people with motor disorders a new and functional way to use technology (Wolpaw et al., 2000). This opens the door for better adaptation and user interaction (Zickler et al., 2009). BCIs have already been proven to provide people with these types of disorders helpful tools such as movement control of prosthetics (Dhanabalan et al., 2022), technology-mediated communication (Kundu & Ari, 2022), environmental control (Jeyakumar et al., 2022; Leeb et al., 2015), wheelchair locomotion control (Pawuś & Paszkiel, 2022), and even neurorehabilitation (Bamdad et al., 2015).

2. Brain Computer Interfaces' systems

Any BCI system is composed of three main elements. These elements are generally classified between signal acquisition devices, signal processing and translation methodologies, and effector devices (see **Figure 1**) (Ortiz-Rosario & Adeli, 2013). For each of these elements, different technologies and methods are and have been developing, as each of them presents different advantages and drawbacks.

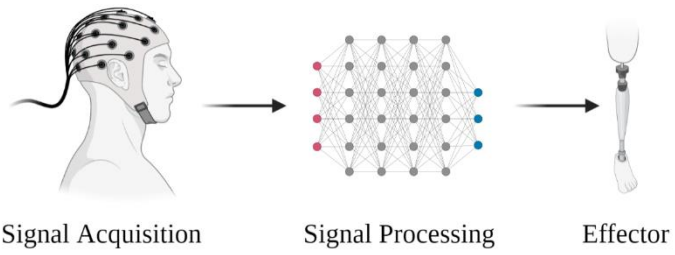

Figure 1: *Basic elements of BCI systems. Created in BioRender.*

2.1. Signal acquisition devices

Signal acquisition is the process of measuring brain activity and translating it into signals. To do this, many approaches can be used, but all of them expect to obtain a sufficient spatial and

temporal resolution to be able to translate these signals to specific actions. Some of the main technologies used in BCIs are electroencephalogram (EEG), magnetoencephalography (MEG), functional Magnetic Resonance Imaging (fMRI), and electrocorticography (ECoG) (Ortiz-Rosario & Adeli, 2013)(Min et al., 2010).

EEG is a non-invasive method that uses metal electrodes on the scalp to measure the electric potentials, that result from neuronal activity. It is the most widely used technology for BCI applications. Its advantages are that it provides very high temporal resolution and is portable, allowing real-world neuroimaging. However, EEG has a low spatial resolution, as each electrode covers a big area of the brain, giving low accuracy in spatial information (Casson et al., 2018).

MEG is similar to EEG, but instead of measuring electric potentials, it measures magnetic fields. Its characteristics are the same as with EEG, although it presents better spatial resolution (Min et al., 2010). However, MEG devices are expensive and big, requiring special settings, which makes them not relevant for real-life applications.

fMRI functions by measuring Blood Oxygenation Level Dependent (BOLD) through magnetic resonance, being a non-invasive technique. This is possible because blood flow levels change depending on brain activity, so, by measuring these changes, it is possible to obtain brain activity information. It has been demonstrated that an increase in neural activity is usually followed by an increase in the blood supply that provides neurons with the required energy (Ladecola, 2017). However, fMRI machines are very expensive and require special facilities and personnel, making it difficult to make them scalable. Apart, subjects need to remain very still, resulting in an approach difficult to be directly implemented in BCI (Min et al., 2010).

Finally, ECoG is a technique that is also based on the placement of electrodes to record brain action potentials. However, contrary to EEG, ECoG is an invasive technique, as the electrode is placed directly on the cortex, requiring surgery. This technique allows the obtention of signals with both high temporal and spatial resolution. However, the field of view is limited to the exposed area of the cortex (Winn, 2022). For other non-invasive used techniques and their characteristics see **Table 1**.

It is to be noticed that other invasive techniques based on intracortical electrophysiology exist, being an important research area (Durand et al., 2022). It has been demonstrated that it can be used to monitor cortical neural populations in freely behaving non-human primates (Yin et al., 2014) and humans (Paulk et al., 2022). As a drawback, the implantation of the needed electrodes must be done through open-brain surgery, causing tissue damage and degradation of the materials and signal quality (Mahajan et al., 2020).

Table 1: *Most extended brain imaging methods for BCIs. Extracted from* (Min et al., 2010)*. Data extracted from* (Kamrani, 2014)

Method	Measurement	Temporal resolution	Spatial Resolution	Limitations	Examples of use
EEG	Electric potentials of cortical activity	Very high (~1 ms)	Coarse (1 cm)	Spatial resolution No information about the deeper layers of the brain	(Abiri et al., 2019; Torres et al., 2020)
MEG	Magnetic fields of cortical activity	Very high (~1 ms)	Coarse (1 – 2 cm)	Spatial resolution No information on the deeper layers of the brain	(Dash et al., 2020; Rathee et al., 2021)
fMRI	BOLD changes in susceptibility-	Low (1–2 s); limited by	Very high (3-6 mm).	Temporal resolution Expensive Needs still subjects	(Du et al., 2022; Sorger &

	weighted MR signal	hemodynamic delays			Goebel, 2020)
Near-infrared Spectroscopy (NIRS)	BOLD changes in the absorption spectrum of near-infrared light	Medium (0.5-1s); limited by hemodynamic delays	Coarse (0.5-2cm)	Spatial resolution No information about brain structure	(Han et al., 2020; Z. Wang et al., 2021)
Functional transcranial doppler sonography (fTCD)	Blood flow velocity associated with neuronal activity	Medium (0.5-1s)	Limited to the vascular territory of the insonated artery (Jansen et al., 2004)	Spatial resolution	(Khalaf et al., 2019b, 2019a)

2.2. Task definition and signal processing

2.2.1. Task definition and understanding

The first step for BCI to function is not only to use the correct technology but also to have a deep understanding of the task that needs to be performed. This is necessary to ensure the best possible performance of the used devices. First, some of these devices do not cover the whole brain, so positioning is key for their acquisition. If the task is known and understood, it is possible to define where to locate the device for obtaining the most relevant information. Apart, for better control of the devices, it is also important to understand where and what kind of activity patterns are happening when performing the tasks, to relate them to specific actions.

To better represent this brain activity, cognitive models were introduced. This understanding of the underlying processes that happen in the human mind dates back to the ancient Greeks. However, it was not until Ulric Neisser published his book 'Cognitive psychology', that this study became a separate area of psychology (Neisser, 1997). However, this discipline had been previously treated by many psychologists, who started to develop cognitive models to explain these processes and try to reproduce and predict the human mind.

Nowadays, these models have evolved into computational cognitive models, which deal with the simulation of human problem-solving and mental processes in a computerized model. These models are largely used in HCI research, as they can create more efficient and accurate user models (Dupret & Piwowarski, 2008), and, specifically in BCI, can help predict user intentions. Apart, cognitive models cannot only be used to understand the needed task, but also for improving the BCI training of the user, and decode the information obtained through brain activity measurement. However, current computational cognitive models are still limited by the technologies, and by the current limited understanding of the human brain, which makes current models normally not able to represent the whole complexity of a process. A better understanding and deeper research on this issue are fundamental to comprehend how and which brain regions are important for specific tasks in BCI.

After the task has been defined and previous knowledge on how the task is processed in the brain has been acquired, the devices and posterior processing can be done with more information leading to better results.

2.2.2 Methods for signal processing

A broad spectrum of methodologies to process and translate the signals exist. Generally, signal processing involves feature extraction and feature classification or translation. Feature extraction techniques are usually divided into time-frequency methods and spatiotemporal techniques.

Inside these groups, many different approaches can be used, and new ones appear frequently (Pawar & Dhage, 2020). Depending on the imaging technique used and the type of application, one method or another is more suitable. Some common examples of signal processing techniques include linear discriminant analysis (Dodia et al., 2019; Fu et al., 2019; Lo et al., 2022), support vector machine algorithms (Maher et al., 2023; Singh & Singh, 2020; Xu et al., 2019) or convolutional neural networks (Fahimi et al., 2019; Roy, 2022; Zhu et al., 2019).

Once feature extraction has been done, there is a need to classify the signals to predict user intentions. Again, there are many ways to perform these classifications, being the most widely used machine learning or deep learning techniques (Aggarwal & Chugh, 2019). Although many different methods have been applied, each individual technique or even the use of different combinations of them still presents several problems that include: not being appropriate for the analysed data; difficult interpretation; problems with overlapping data; obtaining a high enough accuracy of the classification; overfitting of the data; need of ground truth for the evaluation of the method, between many others. Each of these tools also has individual problems that have to be taken into account when being used, which can range from computational burden to the need for pre-defined parameters.

2.3. Effector devices

Once the signal has been processed and translated, the command that is extracted from it needs to be sent to the effector device. Effector devices can have a broad range of shapes depending on their goal. Many different effector devices that use BCI to receive information have been studied, especially from an HCI perspective, where more effective devices are being researched.

Effector devices are not only computers directly controlled through BCI systems (Kumar & Sharma, 2012), but also include a broad range of technologies. In Vasiljevi et al. (Vasiljevic & de Miranda, 2020), a review of different games using consumer-grade devices are made. The devices used in this study as effectors include game consoles, PCs, mobile devices, and websites. The review also includes applications of BCI information into technologies such as adaptive systems, that include cognitive state recognition (Cutrell & Tan, 2008) or emotion recognition (Li et al., 2009). BCI signals have also been applied to evaluate user experience in interactive systems (Cano et al., 2022b) with the effector device not being directly applied, but by evaluating the brain signals when using a certain application, to control prosthetics (Gannouni et al., 2020), to improve VR/AR experience, where the effector device are the VR/AR glasses (Y. Zhang, 2021) or even for musical composition and performance, where the effector device includes a speaker (Williams, 2019).

As reviewed, many different technologies and methods can be used as BCIs. Nonetheless, the existing neuroimaging techniques are still limited and present many problems towards more accurate and functional BCIs such as low temporal resolution, low spatial resolution, invasiveness, low brain coverage, large training periods, or failure on inter-subject functionality between others (Mridha et al., 2021). Because of this, there is a need to develop new technologies that can provide brain imaging with high temporal and spatial resolution. What is more, the development of a validated, reliable, and robust way to process the information extracted from these new techniques needs to be explored. A promising technology that is currently being studied is functional ultrasound imaging, where the Urban Lab at NERF is conducting significant research for improving brain imaging and brain decoding.

3. Functional ultrasound imaging

3.1. Principle of Functional Ultrasound Imaging

Functional Ultrasound Imaging (fUSI) is a hemodynamic neuroimaging technique that measures blood volume through power Doppler imaging. Differently from the beam scanning usually used

in ultrasound techniques, fUSI uses plane-wave illumination, increasing the number of samples per pixel and therefore the sensitivity (Macé et al., 2011). By combining these planar images using compound imaging, it is possible to obtain images with a high signal-to-noise ratio and good temporal resolution (up to 10Hz), also determined by the computing hardware optimization software used to process the raw data. The spatial resolution depends directly on the pitch of the used ultrasound transducer, which also determines the depth of field (usually in the range of 100-300µm[3], and up to several cm in depth). Recording of repeated images over time allows us to track the relative changes in blood volume per voxel, in turn reflecting changes in neuronal activity (Nunez-Elizalde et al., 2022). To avoid the interference of the skull with the ultrasound, all the data is obtained through a cranial window, making fUSI a minimally invasive technique.

When comparing brain imaging techniques many features can be taken into account. The most important ones include spatiotemporal resolution, brain coverage, and restrain method. Although other techniques can overcome fUSI spatiotemporal resolution, it is important to mention that this technology offers full brain coverage, which differentiates it from other techniques. To better compare fUSI to other technologies, we need to observe the whole picture (see **Figure 2**).

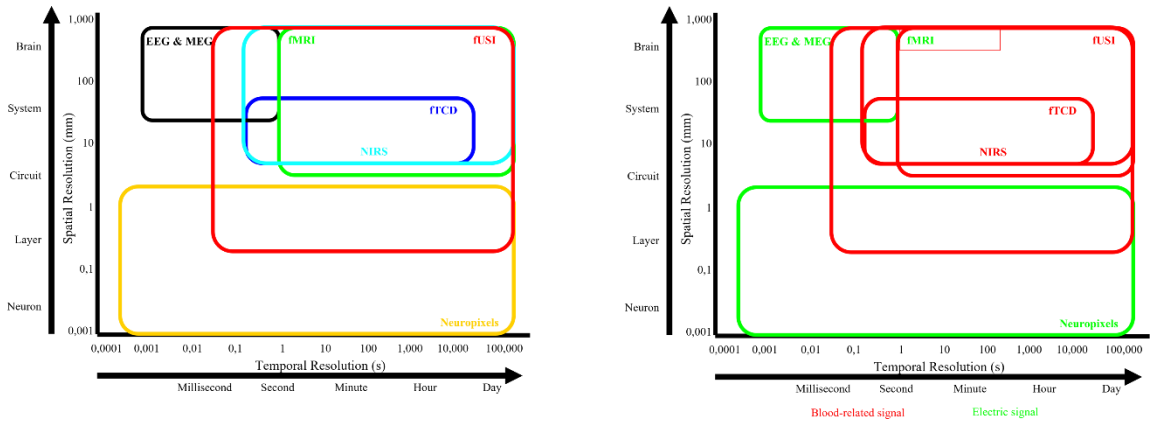

Figure 2: *Comparison of brain imaging techniques in terms of spatiotemporal resolution, applicability conditions and type of signal.*

As can be seen, fUSI covers a broad range of spatiotemporal resolution comparable to many other techniques, reaching temporal resolutions that for instance, fMRI cannot reach, or spatial resolutions that neither fMRI nor EEG can reach. Not only this, but although temporal resolution is lower in fUSI than in EEG or neuropixel, the two later measure electric signals that are much faster than the ones measured through fUSI which are hemodynamic signals. Apart from these benefits, fUSI has also evolved to be a tool applicable to behaving animals (Urban et al., 2015) and has even been scaled to humans (Demene et al., 2016; Soloukey et al., 2020).

3.2. Linear fUSI vs volumetric fUSI

When referring to fUSI, we can now differentiate between two different techniques: linear and volumetric fUSI. Linear fUSI (lfUSI) allows having 2D images up to a depth of several cm, with a spatial resolution of ~100x300x100 µm^3 and a temporal resolution up to 10Hz. The higher-quality images of the brain hemodynamics obtained with linear fUSI have already proved to be useful for single-trial decoding of the timing and goals of an intended oculomotor (Claron et al., 2021; Dizeux et al., 2019) and limbic (Norman et al., 2021) movement in large animals. This demonstrates that the technology has already the potential to be applied in broader BCI applications.

Although linear fUSI has proved to be useful for these applications, it is restricted to cross-sectional images, impeding the direct visualization of the whole brain. To do full brain imaging with linear fUSI, it is necessary to move the probe to multiple positions (Macé et al., 2018). The

problem with this is the need to repeatedly present the stimulus in different positions, requiring a larger acquisition time, and possibly leading to habituation of the animal (Miller et al., 2018). On top of this, the need of moving the probe results in the loose of a substantial part of the brain activity due to its distribution through the brain (Montaldo et al., 2022).

To get over these limitations, a 2D array transducer was used by Rabut et al. (Rabut et al., 2019) to directly obtain 3D images in anesthetized rats. However, this first trial had several limitations such as the decrease in the frame rate, the fact that they used acute conditions, and the anaesthetics, which have been proven to change the hemodynamic response (Schlegel et al., 2015). Because of this further investigation to improve volumetric fUSI (vfUSI) was done by Brunner et al. (Brunner et al., 2020). In their paper, they developed a vfUSI system suitable for real-time 3D imaging of awake subjects with high spatial (220x280x175 μm^3), and temporal (up to 6Hz) resolution that allowed the scanning of almost the entire rodent brain.

3.3. General pipeline for data analysis

With the obtained images using fUSI, referred to as Power Doppler images (**Figure 3**. B), the next step is to register the data. Registration relies on a geometrical transformation to fit the data fit into a reference map of the brain, such as the Paxinos rat brain atlas or the Allen CFF (**Figure 3**.C) (Paxinos & Watson, 2006; Q. Wang et al., 2020a). This process facilitates the comparison of different animals and offers a base for region-averaging procedures. The different trials obtained in an fUSI recording can be used to measure blood volume over time and create an activity map. This activity map can then be averaged per region defined in the used atlas, maximizing the signal-to-noise ratio, and obtaining a temporal map for interpretation (See **Figure 3**.F). Then it is possible to proceed to signal processing and interpretation. In this work, we will explore alternatives to facilitate interpretability and visualization in this step.

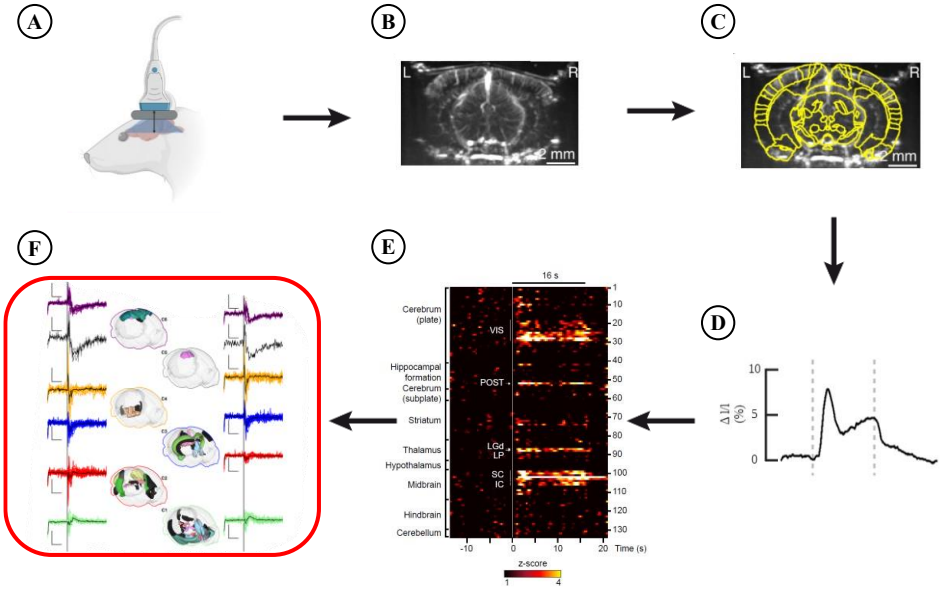

Figure 3: *fUSI process to interpret brain activity.* ***(A)*** *Record fUSI images for a specific experimental setting.* ***(B)*** *Obtain the multiple power-doppler images recorded through the probe.* ***(C)*** *Register each of the images on the decided Atlas to facilitate intra-animal comparison and further steps.* ***(D)*** *Average the signal obtained for each of the pixels on a specific region, to obtain a single image per region.* ***(E)****Create a temporal map containing the temporal trace of activity per each of the brain regions.* ***(F)*** *Proposal of this work for further analysis. It consists of the clustering of the different temporal traces to identify specific activity patterns and facilitate the inspection of brain behaviour, leading to a simplified way to decode specific stimuli. Created in BioRender.*

The problem when having brain-wide images of the brain, especially when using vfUSI is that the amount of information obtained is very large, in the order of ~150 full-wide brain images per experimental trial with a resolution of $\sim 150 \mu m^3$. Although region averaging can provide useful results, it can easily lead to misinterpretation of the data (Poldrack, 2007), or loss of important cues (Constable, 2006).

As a proposal to overcome these limitations, single-voxel clustering has been recently developed. This method relies on the grouping of single voxels in the region of interest (ROI) instead of relying on the averaging of the different brain regions. This results in more specific activity maps with better signal source separation (See **¡Error! No se encuentra el origen de la referencia.**). As a drawback, the outcome of this type of analysis is high-dimensional, meaning that its interpretation is difficult. However, this opens the door to new paradigms that combine the traditional analysis methods with single-voxel clustering for brain-decoding purposes.

4. Animal models in BCI research

Although fUSI is a promising technology for BCI, it is still in its development age, with studies being made specially in animal models. Although the final aim of BCI is to be applied to humans to unlock new ways of communication between them and different technologies, some studies are still not possible directly in human subjects. When doing fundamental research to improve the outcomes of any technology, especially when researching new fields, studies with humans become very limited. Here is where the importance of animal models lies, as different studies can be done *in vivo* for future application to humans considering the ethical issues. Many advantages exist in using animals to do research. First of all, when using animals, it is possible to use invasive techniques that would have ethical issues if used on humans. Not only this, but more advanced recording methods are available for smaller animal models such as rodents. Also, the possibility and affordability of using a larger number of subjects appear, as animal experiments are more available than the ones with human subjects. Apart, when using, for example, rodent animal models, there is an increase in the possibilities and experimental settings. This includes genetic tools that allow to create and research different defects; physical tools such as circuit manipulations; or more invasive histology between others.

It is known that across mammals, species have conserved many properties and advanced brain functions. Thanks to this, animal models have already helped enable the translation of the first BCI technologies into humans. Using animals allows us to understand more specific and possibly relevant issues when it comes to the decoding of brain activity and apply the knowledge directly to humans.

5. Proposal

Through this master thesis, I propose the design of an analysis pipeline to differentiate two related stimuli through fUSI images, which can have a great impact on the decoding of brain signals for BCI. To do so, I will first outline specific areas of interest at a regional brain level, and then I will analyse the chosen regions in order to obtain a classifier for pain decoding and compare it to other region-selection methods. Not only this, but the pipeline also aims to help the interpretation and investigation of whole-brain imaging, where data can become complex and difficult to analyze.

As previously stated, vfUSI data allows us to obtain whole brain images. In this case, it is possible to obtain images at a frame rate of 6Hz and with a resolution of ~200μm. This allows us to combine and compare the information obtained from multiple regions, to understand how specific processes function in the brain. The data obtained through fUSI is usually treated through region-averaging approaches that can provide fast and satisfactory results.

First, we pretend to establish a pipeline to determine which are the brain regions of interest for a specific stimulus. In this case, we will first use a first stimulation, A, applied to rodents and obtain

a view of the circuit using region averaging. Then, we will use a second dataset, B, that will be related to the first one. Through this, we will try to determine which regions are of interest for the detection of the second stimulus. This includes those regions that present more changes between the activity in A and the activity in B. The changes can include more activity in the specific region, less activity, or changes in the type of activation. The identification will be done through the study of different clustering techniques, such as k-means or hierarchical clustering, to classify atlas-based brain regions in different activity patterns. This process aims to accomplish two key goals:

1. **For task definition and understanding:** The development of a pipeline for visualization and interpretation of high-dimensional brain data. This will support further exploration of brain pathways and behaviours, enabling tracking of specific brain regions and activity changes between different stimuli. The potential benefits of this goal include:
    a. **Improved BCI accuracy:** Effective interpretation and visualization of high-dimensional brain data may lead to more accurate decoding of brain signals. Therefore, BCI systems could become more precise, reducing potential errors and improving user experience.
    b. **Enhanced user training:** visualizing activity patterns can help researchers and users to better understand brain signals, potentially leading to easier learning on the use of BCIs and making the technology more accessible.
    c. **Brain activity research tool:** to visualize and better interpret high-dimensionality data could serve as an interesting tool for researchers that want to understand brain activity, especially during BCI.
2. **For fUS data analysis:** A robust methodology to determine the ROIs for further study given specific stimuli. This involves identifying which types of activity are relevant for the presented stimulus and comparing this method to previous knowledge of the stimuli to verify if the process can yield superior results in identifying them.

Following the determination of the interest brain regions through clustering, we will perform a classification between the two stimuli. To validate the usefulness of the proposed pipeline in identifying relevant regions, we will compare the performance of cluster-based selected brain regions to those selected based on prior knowledge. The study will also examine the relevance of whole-brain imaging data versus cortex-only data for the classification of stimuli. This will provide insights into how simpler imaging techniques perform relative to more comprehensive data. The potential benefits of accurate and relevant classification of brain data and comparison of region-selection methods include:

a. **Personalized BCI Systems:** By understanding which areas of the brain better respond to particular stimuli, BCI systems can potentially be tailored to the individual user. This can enhance the efficiency and effectiveness of the BCI, leading to a more personalized and intuitive user experience.
b. **Increased BCI Responsiveness:** If we can identify which types of brain activity are relevant for a given stimulus more accurately, the BCI can respond more quickly and accurately to user commands. This could result in a more responsive and reliable BCI system.
c. **Innovation in BCI Design:** A deeper understanding of how specific stimuli affect different brain regions could stimulate innovation in BCI design. Researchers could use this information to develop new BCI systems that take advantage of this understanding of stimulus-brain region relationships, potentially opening up new possibilities for BCI applications. What is more, understanding the performance of different brain regions in the classification of stimuli, can help to understand the needed complexity of BCI systems.

The whole study proposal can be seen in **Figure 4**.

For this research, the stimuli used to study new ways to interpret and classify fUSI data will consist of two different thermal stimuli directly applied to rodents. The first stimulus will consist of a 40ºC thermal stimulus that is considered to be non-painful. The related stimuli that we will try to decode will be a 50ºC stimuli which is considered to be painful when applied to the subjects. Painful stimulation is known to induce changes, not only in brain activity but also in the behaviour of the animal. These changes and movements will mimic real-life conditions, where brain waves are not perfect, and many different interferences can be found.

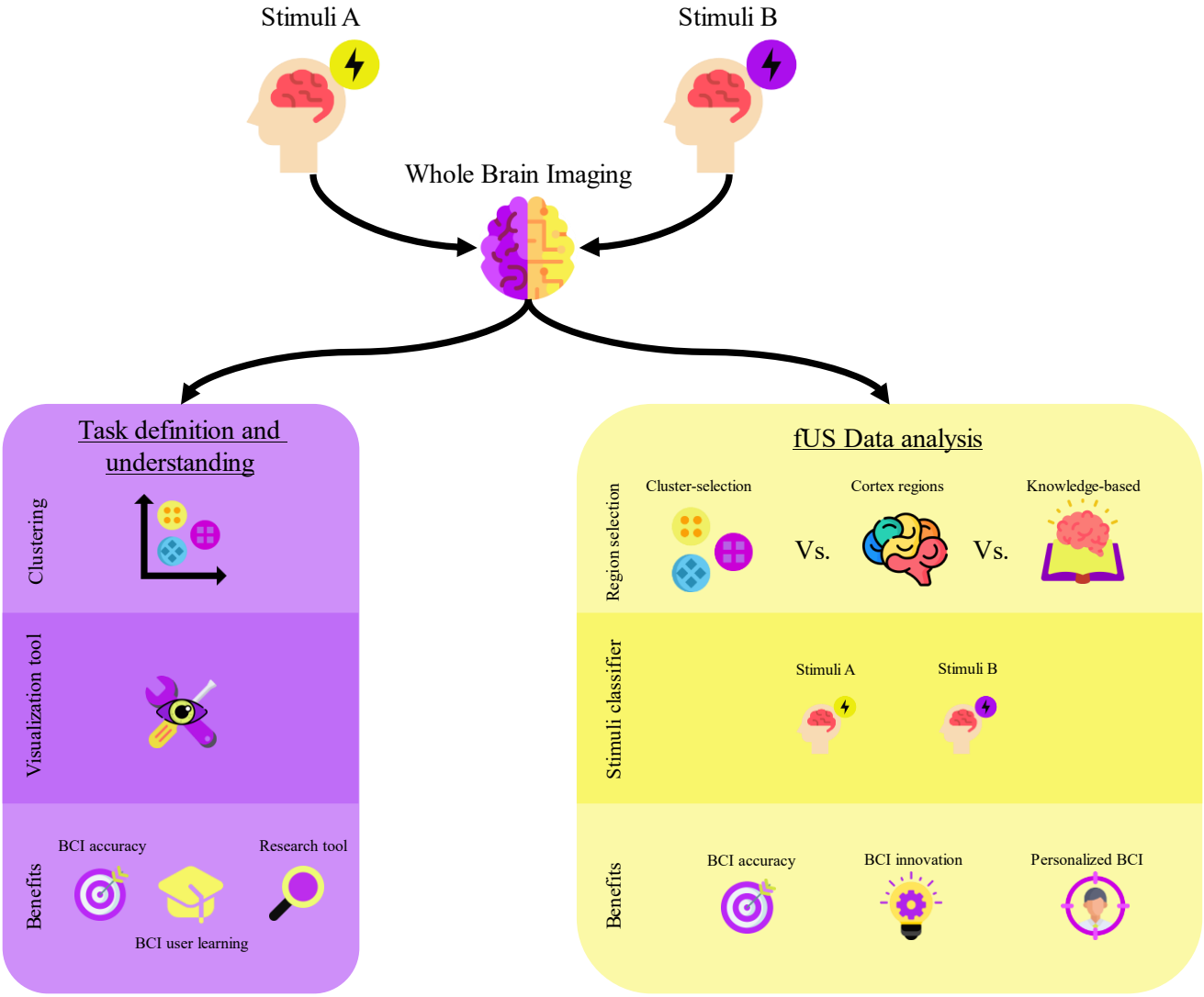

Figure 4: *Research proposal including process, obtained results and potential benefits of every part.*

Skeleton of the Thesis

The general thesis skeleton will be explained in this part. However, the final document could be slightly modified because of corrections or findings, to better express the information. The thesis will consist of:

Introduction and literature review

1. Human-Computer interaction and BCI
2. BCI systems
3. Functional Ultrasound Imaging
4. Animal models in BCI research
5. Thesis proposal and research questions

Research Methodology

1. Dataset
2. Clustering
3. Towards brain-decoding

Results

1. Region-averaging results
2. Single-voxel results
3. Classification results

Discussion

Conclusion

Time Plan for the Second part of the Thesis

At the moment, studies related to region-averaging clustering have already been done. However, it is still needed to define the regions of interest. Therefore, the time plan for the second part of the thesis can be seen in **Table 2**.

Table 2: *Time Plan for the second part of the thesis*

Name	Start Date	End Date	Duration
Extract Regions of Interest	Feb 13	Feb 27	11 days
Single-Voxel Clustering	Feb 27	Apr 21	40 days
Choose Clustering technique	Feb 27	Mar 17	15 days
Perform Clustering of the ROIs	Mar 15	Mar 31	13 days
Interpret Information	Apr 03	Apr 21	15 days
Brain Decoding	May 01	Jun 09	30 days
Classification of Region-level data	May 01	May 12	10 days
Classification of Voxel-level data	May 12	May 26	11 days
Comparison of models	May 29	Jun 09	10 days
Thesis Writing	Mar 06	Jun 30	85 days

Research Methodology

For this study, the first step consists of registering the data obtained through vfUSI, following the typical fUSI pipeline, which includes fitting the data in a brain map. In this case, we will use the Allen Mouse Brain Reference Atlas (Q. Wang et al., 2020b). An average of the brain regions will then be made, leaving a temporal map for analysis. Using clustering techniques, we will analyse the region-averaging data of the two thermal stimuli (painful and not painful), to determine which regions are of interest to differentiate them.

Once these regions have been determined, the next step will be to analyse them at a single voxel level, again using clustering techniques to do so. By doing so, we will be able to determine which information at the single voxel level is useful for decoding the painful stimulus in the last step: classification. For a complete overview of the project see **Figure 5**.

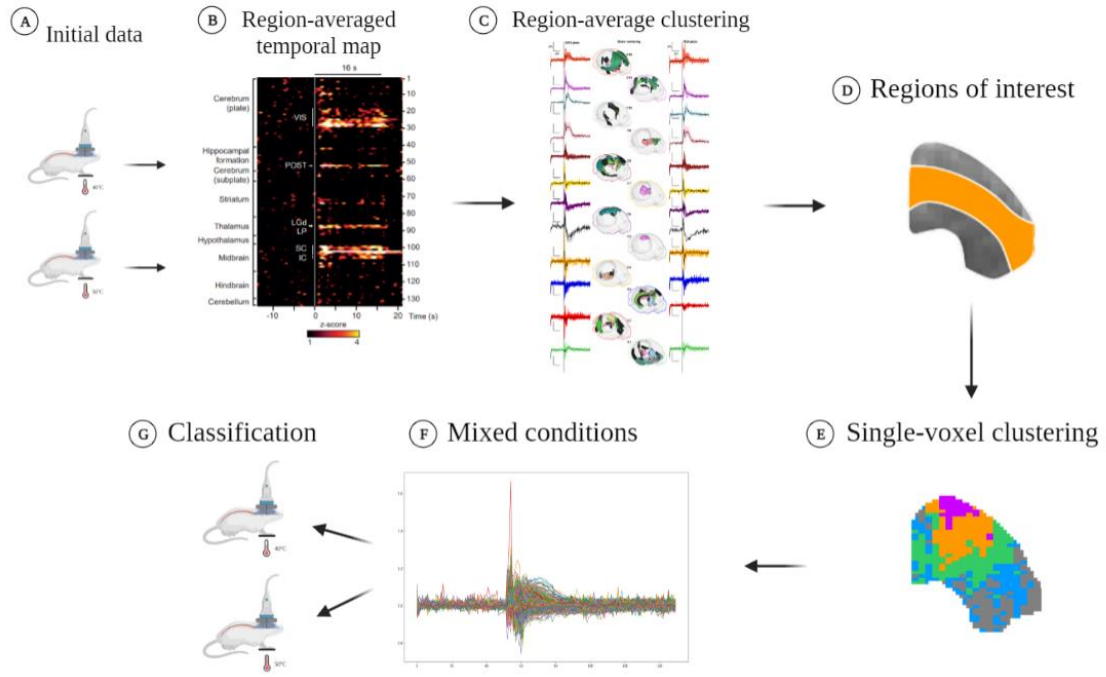

Figure 5: *Process of this Master Thesis.* ***(A)*** *We will get data from two conditions.* ***(B)*** *Region-averaging will be performed in this data after registration to obtain the temporal map.* ***(C)****The region-averaged data will be clustered using the best method in order to find activity patterns.* ***(D)****From the clustering, an analysis will be conducted to find the regions that are of most interest when differentiating both conditions.* ***(E)*** *Single-voxel clustering will be applied to the selected regions.* ***(F)*** *Both data averaged per region and single voxel data will be used separately to train classification models that are able to differentiate between both conditions.* ***(G)*** *We will obtain a classification method that is able to distinguish between the two conditions. Created in BioRender.*

1. Data set

1.1. Volumetric Images

For the dataset used in this work, 5 different subjects of Wild type mice were subjected to three different thermal conditions, directly applied to their right paw, in different sessions. In each session, the animals under came 5 rounds of 5 different thermal stimuli. The presented stimuli included: one time a 30ºC stimuli, three times a 40ºC stimuli, and one time a 50ºC stimuli (which induces pain in the subject), with randomized appearance per round. The 30ºC stimuli is just used as control condition. On the other hand, the 50ºC stimuli was only presented one time in each round to avoid inflammation or burns in the animal that could induce pain sensibilization. In total, the animals were subjected to 27 different sessions, with 5 rounds each, having a total of 675 trials (see **Figure 6**.A).

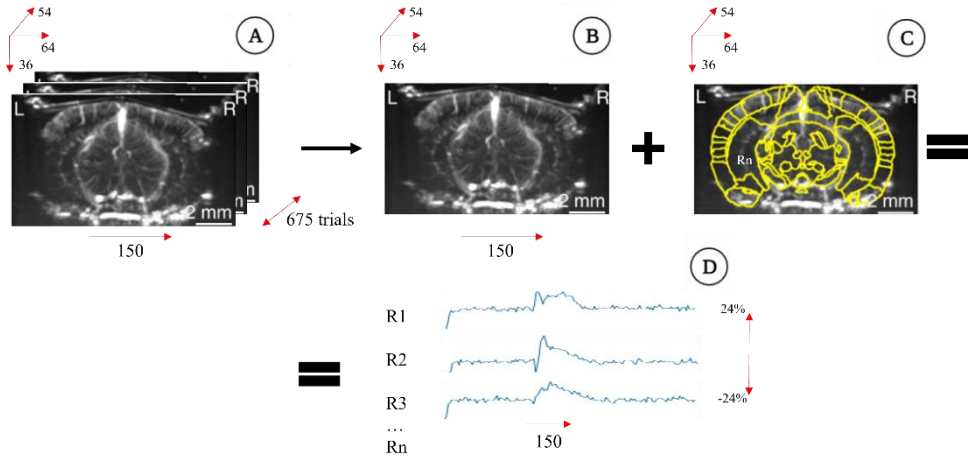

Figure 6: *Dataset used in the work. The numbers represent the number of frames.* ***(A)*** *Original dataset, with all the trials.* ***(B)*** *Averaged dataset.* ***(C)*** *Registration in the Allen brain Atlas.* ***(D)*** *Region Averaged signals.*

In each session, volumetric images of the brain were acquired using a fUSI probe composed of a 2D matrix of ~1,024 piezoelectric elements of 300µm at 15MHz. The probe covered the full volume of the brain (~1cm^3 in rodents). A direct computation of the fUSI images was performed, resulting in images obtained at a frame rate of 2Hz and with a resolution of ~150µm^3. A total of 150 images per stimulus were obtained, meaning a total of 750 images per experimental round.

1.2. Temporal Traces

To simplify intra-animal comparison, the images were then registered in the Allen Mouse Brain Reference Atlas and segmented into 229 regions per hemisphere (see **Figure 6**.C).

A neural signal can then be obtained for each voxel on every image of each stimulus inside a trial. This means that for a round, each voxel had a signal for the 30ºC stimuli, three for the 40ºC, that were then averaged, and one for the 50ºC. Once this was obtained, an average of all the signals per each round inside a session, and then an average of all the sessions was performed (see **Figure 6**.B). This was done to obtain single-trial data and to have signals with lower noise-to-signal ratio. This results in time-series of 150 data points for each of the different stimuli. The signals represent the values of hemodynamic activity ranging an amplitude between ~-24% and ~24% (ΔI/I) (see **Figure 7**), where ΔI/I represents the normalization given a baseline of the doppler signal obtained.

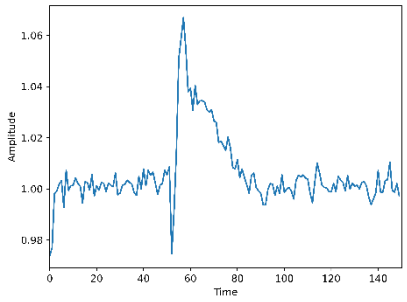

Figure 7: *Example signal of neural activity extracted from a fUSI recording.*

To obtain a more general view of the whole brain and reduce the dimensionality of the data, each voxel was assigned to one of the 229 regions, based on the Atlas. Then, an average of all the signals of the voxels corresponding to a specific region was performed, to obtain a single temporal trace per region. This resulted in 458 temporal traces (229 per hemisphere) for each of the different stimuli (see **Figure 6**.D).

In this case, we will use the data for the stimulus 40ºC and 50ºC. The data will be used as a standard case of decoding an evoked activity, looking for the difference between the two temperatures. The 50ºC pain-induced dataset will also be used as a reproduction of real-world applications, where there are no perfect signals, as the behavioural changes in the animal are expected to infer with the signals. The study of this data will provide an insight on the contribution of disturbances to normal activity and help identify which regions can be used for fine decoding.

2. Clustering

Because of the high dimensionality of data obtained (~120,000 voxels), analysing fUSI signals is not an easy process. Not only the volume but the fact that they are time series can make their comparison and interpretation particularly hard. In this work, we propose a cluster-based approach, which relies on grouping the data depending on its activity pattern to facilitate its interpretation. This method makes it possible to find differences in the activity between regions and conditions in a more straightforward way with further reduced dimensionality (d=K).

However, the clustering of any type of data depends on the used methodology, which needs to be selected and adapted to the analysed dataset. The clustering process is generally divided into three different steps (see **Figure 8**) (Guyon et al., 2009):

1. **Feature selection:** here the parameters and information that will be used for clustering are set.
2. **Clustering:** different algorithms can be used to separate the data into different groups.
3. **Validation of the results:** different measures are used to quantify how relevant is the grouping of the data.

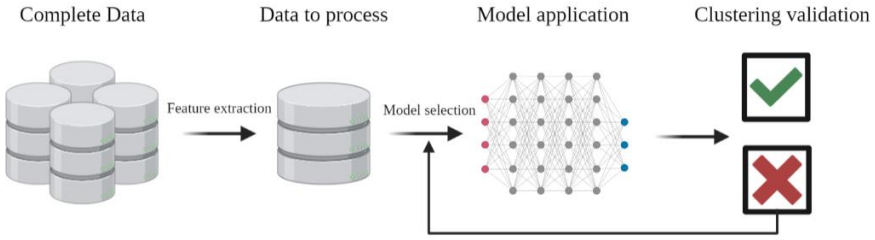

Figure 8: *Generalized clustering process for a dataset. The model selection needs to be validated and changed if results are not relevant.*

2.1. Feature selection

We call feature each input variable that will be given to the clustering model. Although using the raw data is possible, sometimes it can be problematic as the model can use not-important features to group the data and learn from noise. Because of this, the important parameters of the data need to be selected. A common method to do so in time series analysis is to rely on certain signal parameters, such as amplitude or time-to-peak (Krishnan & Athavale, 2018). However, these types of parameters are difficult to determine in fUSI signals because of the variability/noise in the signal (see **Figure 9**).

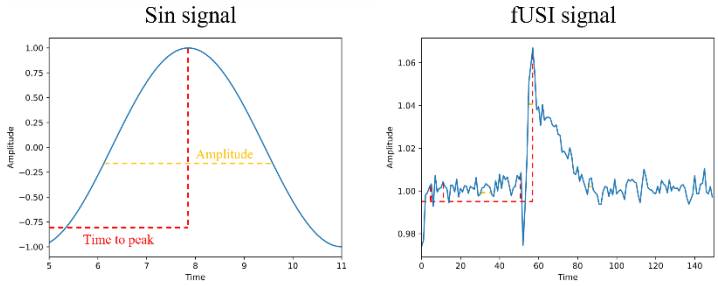

***Figure 9**: Comparison of measuring signal features in a sin signal vs a fUSI signal. It can be observed that in a fUSI signal is very complex to choose which or where to measure possible features.*

That is why it is necessary to investigate different approaches that minimally imply the use of engineered features and observe their advantages and drawbacks.

2.1.1. Dimensionality Reduction

When processing large amounts of data, it is a well-known procedure to use dimensionality reduction techniques to improve the computational time and amount of data needed to properly sample the input space. It is well known that an increase in data dimensionality results in an exponential increase in the data needed to sample the input space properly (Karanam Shashmi, 2021). Not only this but reducing the dimensionality of our data can help eliminate the non-important features, resulting in better outputs (van der Maaten et al., 2009). It is to be mentioned that dimensionality reduction is not useful in all data analysis cases. In this work, we will try two different dimensionality reduction techniques, apart from the raw data, to compare their results. These two methods have been chosen as they are largely used methods that have demonstrated good performance. However, in case of not obtaining satisfactory results with them or the raw data, other dimensionality reduction techniques will be applied such as Factor analysis (Khosla, 2004), Non-negative matrix factorization (Hoyer, 2004), or latent Dirichlet allocation (Blei et al., 2003).

2.1.1.1. Principal Component Analysis (PCA)

PCA is a dimensionality reduction technique that transforms the data into a set of orthogonal components that maximize the variance explained in the dataset. PCA applies an orthogonal transformation to the data, which consists of a linear transformation represented with orthogonal matrices that are composed of orthogonal vectors with the unit norm. This results in the projection of our original features onto the space determined by the principal components (Kurita, 2019).

When applying PCA, we need to choose how many principal components will be used for further investigation. This is generally done through the observation of how much variability within the data is presented by each component. The plot of this information results in a curve, where the number of components is presented on the X axis, and the variance explained on the Y. Generally, the inflection point of this curve indicates the appropriate number of components to choose.

2.1.1.2. Independent Component Analysis (ICA)

ICA tries to decompose the time series into the maximum number of independent components. Although it is usually used for blind source separation, it can also be used as a dimensionality reduction method (Hastie et al., 2009). To do this decomposition, ICA relies on the central limit theorem that states that any linear mixture of independent random variables will have a higher gaussian distribution than the original variables.

In other words, ICA tries to rotate the axis of the data, while minimizing the Gaussianity of the data projected on the axis. When the Gaussian distribution is at its lowest, ICA can recover the independent components of the data. Normally, the number of components to be chosen is determined through PCA.

1.2 Clustering techniques

The clustering of data can be done using different machine learning algorithms that find hidden patterns in unlabelled data to group it into different clusters. This is known as unsupervised machine learning. Depending on the data that is needed to be analysed, one algorithm or another may be used. In this case, we need algorithms that are suitable for a large amount of data. In Macé et al. (Macé et al., 2018) a K-means algorithm was successfully used to identify activity patterns, demonstrating that fUSI data can be analysed without using too intricate methodologies. By reason of this, in this work, we decided to try three different clustering algorithms to compare their results.

2.2.1. K-means Clustering

K-means clustering is one of the more extended clustering algorithms because of its characteristics. This includes implementation speed and simplicity; good performance with large amounts of data; and flexibility in the distance measures used (Sinaga & Yang, 2020). K-means works by first randomly selecting *k* centroids, with *k* being selected a priori as the number of resulting clusters. Then, it performs iterative calculations to optimize the position of the centroids. Finally, it calculates the distance of the data points to the centroids and assigns each point to the nearest centroid (see **Figure 10**) (Hastie et al., 2009).

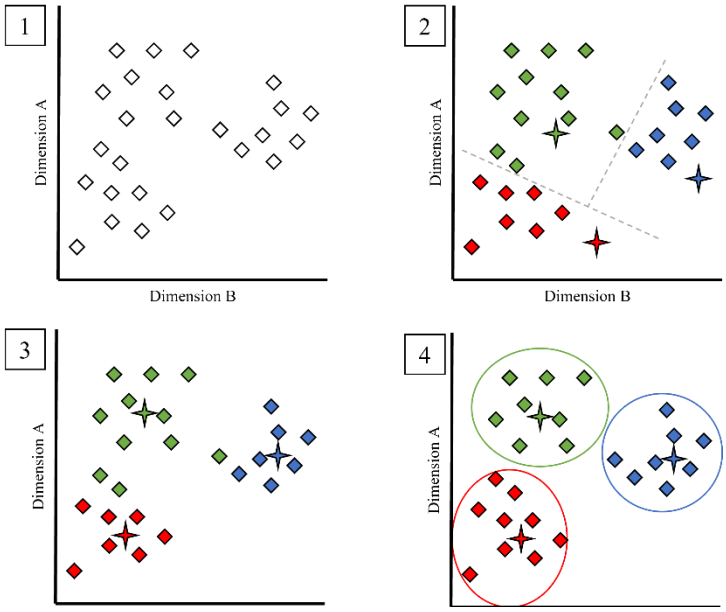

Figure 10: *K-means process.* ***(1)*** *The data to be processed in the n-dimensional space.* ***(2)*** *The centroids are randomly decided, and each datapoint is assigned to the closest centroid.* ***(3)*** *The centroid is corrected so it is in the centre of the datapoints assigned.* ***(4)*** *The datapoints are assigned to each centroid depending on the new distances to them. This process is repeated iteratively.*

The main drawback of this method is that the number of clusters needs to be selected a priori. To better determine this, we will several methods that will be further explained in 2.2.1.

2.2.2. Agglomerative clustering

Agglomerative hierarchical clustering is also a useful method for large amounts of data. This algorithm works by first establishing each data point as a cluster of its own, to then take the two nearest clusters to join them into a new single group. This process is iteratively repeated until the

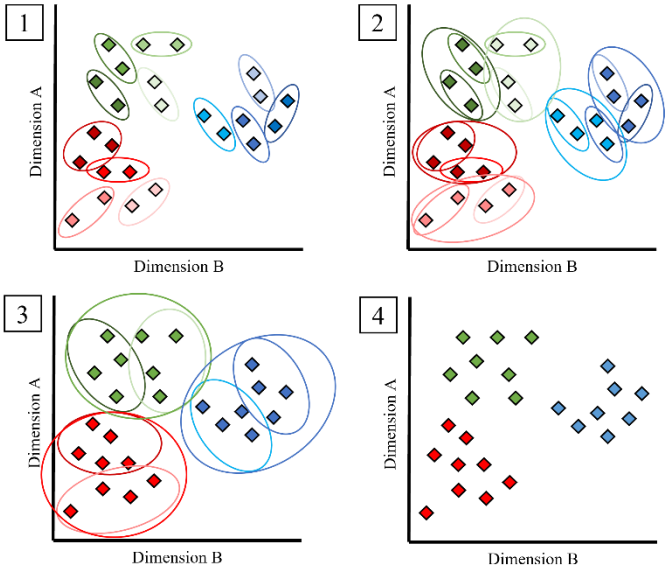

Figure 11: *Hierarchical clustering process.* ***(1)*** *The nearest points on the data are clustered together.* ***(2-3)*** *The nearest clusters are merged in an iterative process until meeting the number of clusters specified.* ***(4)*** *Final result.*

desired number of clusters is reached (see **Figure 11**). This algorithm suffers from the same problem as k-means, as the number of clusters has to also be set a priori. However, it can also be replaced by setting a distance threshold. This technique is not only interesting because of the output clusters, but also because it is possible to extract a dendrogram that graphically represents the order in which factors are merged (Pedregosa et al., 2011). This can be useful for obtaining more information from the analysis.

2.2.3. K-shape clustering

K-shape clustering is a clustering algorithm built upon the basis of K-means algorithm. However, K-shape method was designed specifically for clustering time series based on their shape. To do so, it uses as a distance measure a normalized measure of the cross-correlation method, that gives a shape-based distance between two time series. Note that this means that K-shape clustering works in the same way as K-means but changes the distance measure. As happens with K-means and agglomerative clustering, K-shape also needs the number of clusters to be set prior to the grouping (Paparrizos & Gravano, 2015).

2.3. Performance evaluation

As proposed, we will evaluate the different combinations between dimensionality reduction techniques and clustering algorithms described above. First, we will evaluate the feature selection to decide which data is appropriate to use for the clustering. Once this has been decided, the different clustering algorithms can be evaluated to choose the more significant combination.

2.3.1. Feature selection evaluation

Before establishing which combination is the most appropriate, it is necessary to study the performance of dimensionality reduction techniques. To do so, the explained variance plot is first

used to select the optimal number of components using the inflection point of the curve. Once this is done, it is possible to plot the principal components of the data to observe how it is distributed in the space, and if the partition of the components makes sense (Hastie et al., 2009).

Here, it is important to take into account that, if the minimum number of components that seem to explain a high variance has not a clear partition, or does not seem significant, it is not useful to increase these components.

2.3.2. Clustering algorithm evaluation

Once the data to input has been evaluated and selected, the different clustering algorithms can be applied to it to evaluate the outputs. Different metrics can be used to evaluate the quality of a clustering, by studying the similarity inside a cluster and with the other clusters. Some of these evaluation measures can and will also be used to determine the optimal number of clusters within a clustering algorithm. As in this case we are using data without known labelling, also known as ground truth, the evaluation needs to be done using the model itself.

2.3.2.1. Silhouette coefficient

The silhouette coefficient is a number defined by two different scores for each data point. The silhouette score *s* for a single sample is defined as:

$$s = \frac{b - a}{\max(a, b)}$$

Where *a* is the mean distance between the datapoint and all the other samples within the same cluster; and *b* is the mean distance between the datapoint and all the samples of the nearest cluster. To obtain a score for each clustering algorithm, the mean of the different scores is used. The result is a number between -1 and +1, with 0 indicating overlapping clusters. This coefficient is also used to validate the correct number of clusters that should be used as input for the clustering algorithms (Pedregosa et al., 2011).

2.3.2.2. Calinski-Harabasz Index

This index measures how similar is an object to its own cluster compared to other clusters. It does this based on the distances between the samples of a cluster and its centroid to determine the intra-cluster similarity. For the inter-cluster similarity, it measures the distance between the cluster centroids and the global centroid of the dataset. The Index is calculated through the following equation:

$$CH = \frac{\frac{\sum_{k=1}^{K} n_k ||c_k - c||^2}{K - 1}}{\frac{\sum_{k=1}^{K} \sum_{i=1}^{n_k} ||d_i - c_k||^2}{N - k}}$$

Where c_k and n_k are the centroids and datapoints of the k^{th} cluster; N is the total number of datapoints, and c corresponds to the global centroid. The higher the Calinski-Harabasz Index is, the better the clustering separation is (Pedregosa et al., 2011).

2.3.2.3. Davies-Bouldin index

This measure is used to determine the average similarity of an individual cluster with the cluster most similar to it. The clusters are better separated when this average is low, meaning that in this case, we look for the smallest value. First, the similarity is defined as:

$$R_{ij} = \frac{s_i + s_j}{d_{ij}}$$

With *s* being the average distance between each datapoint in a cluster and the clusters' centroid, and d_{ij} being the distance between the centroids of clusters *i* and *j*. Once this is done, the Davies-Bouldin index can be calculated:

$$DB = \frac{1}{k}\sum_{i=1}^{k} \max R_{ij^{(\text{with i} \neq \text{j})}}$$

With *k* being the number of clusters (Pedregosa et al., 2011).

2.3.2.4. Spearman's correlation coefficient

This coefficient is used to measure the similarity of the samples within a cluster. It measures the strength of a monotonic relationship between paired data, being a generalized tool for correlation analysis. Different to other correlation measures, Spearman's coefficient does not rely on the linear relationship of the data. This is because Spearman's correlation is not calculated directly through the numerical values of a dataset, but by its relative order, known as rank. Its value can be defined by the following formula:

$$s_c = \frac{cov(R(X), R(Y))}{\sigma_{R(X)}\sigma_{R(Y)}}$$

Where R(X) and R(Y) refer to the rank values of each compared dataset and σ to the standard deviation. The coefficient can range from -1 to 1, with 0 meaning that there is no relationship between the two time-series. This can be applied to find the difference between two individual signals, but if the signals inside a cluster are averaged, it is possible to also find the similarity between clusters (Hastie et al., 2009).

2.3.2.5. Euclidean distance

The Euclidean distance measures the difference between two signals point by point. To do so, it requires the signals to be of equal length, as it is the vector norm between two time series. The advantage of this method is that is very fast to compute, but it has to be taken into account that its application is limited, as many differences in time series cannot be found using this metric. The distance is defined by:

$$d = 2\sqrt{(a^2 + b^2)}$$

Its values range from 0 to infinity, where the bigger the number, the less similarity between the time series exists (Hastie et al., 2009).

Using these different quantification methods and examining them, we will determine the best number of clusters and the best clustering technique to group our dataset. It is important to mention that any of these techniques are not always fully reliable when it comes to the outcomes, so they have to be studied and decide if they can be used as a valid quantification for the clustering validity.

2.4. Region-wise analysis

To decide the regions of interest that will be further analysed with single-voxel clustering, we will first need to cluster the region-averaged dataset (see **Figure 6**.D), using the previously mentioned techniques. The first step will be to decide on the use of the raw dataset or to use a dimensionality reduction technique. Then, the number of clusters and the clustering technique will be chosen to obtain *k* number of groups representing different activity patterns.

The data from conditions 40ºC and 50ºC will be clustered together to ensure that there is a common basis to interpret the data. This means that each cluster will contain a set of regions both from the 40ºC response and the 50ºC one, that will later be divided for visualization purposes. If we clustered the conditions separately, we would obtain different patterns for each that would be very difficult to compare. This section aims to find those activity patterns that are exclusive or stronger to condition 50ºC, as they will be composed of those regions that codify pain response. The analysis will be based on observation and quantification through similarity measures between the time series. We will use different visualization methods for this purpose, which will be chosen in a later stage depending on the grouping and complexity of the outcome. For the quantification of how similar or not are the different regions, we will use different parameters to make specific cluster characterization. Although we have mentioned that these parameters are difficult to apply to fUSI signals, by using the average of the signals included in a cluster, the outcomes can remain significant. The parameters that we will include are:

- Time to peak: time point corresponding to the first inflection point after the stimulus onset.
- Response length: last time point of the longest sequence of negative derivatives after stimulus onset.
- Full width at half maximum: the duration of the longest consecutive series of negative slopes following the onset of the stimulus.
- Number of positive vs negative segments:
- Changes in the derivatives
- Time point at half maximum: the point when the signal exceeds half the maximum value after the stimulus has begun.

3. Towards brain-decoding

The next step relies on the findings of the previous session for inter-condition comparison. Here we will try to classify the region-averaged data into two different conditions, the non-painful stimulus and the painful one. Different region selection methods will be used to determine and compare their performance.

3.1. Region selection methods

Three different methods will be used to compare and decide which regions perform better to classify the different stimuli. This will help us understand the necessary process for brain data classification and will help in the understanding of the need (or lack of) for whole brain imaging for BCI.

3.1.1. Data-based selection

The first approach we will use will rely on finding the individual accuracy scores for each region, and then mixing that region with the rest of the highest-scoring ones. The process will be repeated until the classifier starts to get confused with the addition of new regions. This approach will help us to find a mix of regions with a high accuracy score, and also to identify the best number of regions to be used before the accuracy decreases due to confusion of the classifier. However, due to a large number of regions, we will not be using all the combinations possible, but only mix the best-performing regions between them so the final result may not be the group of regions with the best performance.

3.1.2. Cluster-based region selection

The first approach we will use will be based on the previous clustering. Through the observation of the variations in activity patterns between conditions, we will select the regions with higher differences between the stimuli, expecting that those regions contain important information for the identification of each one.

3.1.3. Cortex regions

The goal of this selection method is to compare the performance of internal brain regions compared to those in the cortex, to understand the implications of whole-brain imaging in BCI. To do so, the individual performances of each individual region for a chosen classifier will be investigated. Not only this, but through mixing the regions with better accuracy we will aim to find the region combination that better performs in the classification of the stimuli. Then, we will observe the position of those regions that show better performance and compare them to the performance of solely cortex regions.

3.2.3. Knowledge-based selection

Finally, we will study the performance of regions that are known to be involved in the specific stimuli. In this case, we will be studying non-painful vs painful stimuli, so we will use regions involved in pain processing. Although the process of how the pain is processed in the brain is not completely understood, a common biological pathway related to it is the Somatosensory pathway (Tan & Kuner, 2021) . Therefore, we will use regions that are highly related to this process. (Maybe include list of regions that will be used)

3.2. Classification algorithms

To approach the classification, we will use supervised learning, meaning that, in this case, the data entered will have a label: painful or not painful. However, due to the large dimensionality of our data, using the raw signals for training can be problematic, as overfitting may happen, and the models would not perform well. To study this, we will choose a set of parameters from the signals that will be used as features to input to the classification algorithm, which will be selected after the clustering, where we will see what features better differentiate both conditions. Then, we will compare the performance of the raw signals vs the extracted features. Also, for classification, we will use the individual trials of each mouse, as we cannot use the average because we would be left with a sole input.

For binary classification problems, many different algorithms can be used. These algorithms are first trained with the labeled data to predict the label of new data that has not been used for training. New algorithms are constantly being designed. However, in this work we are not looking to find the highest possible accuracy, but we are rather looking to better understand brain signals and how the processing of stimulus works. Because of this, we will not explore complex algorithms or approaches, but we will use common explainable models for better understanding of te outputs. The algorithms that we will explore in this work are the following ones.

3.1.1. Binary logistic regression

Logistic regression is used for binary classification problems and is based on probability. Logistic regression uses the *Sigmoid function* to map each datapoint into a probability value between 0 and 1, instead of directly using a linear function that would not properly fit the data. This function can be described as:

$$\sigma(x) = \frac{1}{1 - e^{-(x)}}$$

This function will be used to predict the sigmoid function from the linear function $y = ax + b$ to obtain the function that bests fits our data. This means that for an input of x features represented as a matrix X, with each feature having a weight w, we will be able to obtain a prediction value using the function:

$$\hat{y} = \sigma(W^T X + b)$$

Later, a threshold for the prediction value can be set to choose if a specific data point is assigned to one condition or the other (Hastie et al., 2009).

3.2.2. Support vector machine

Support vector machine (SVM) algorithms have as objective finding a hyperplane in the data dimensionality that can distinctly classify the datapoints, maximizing the distance of points in different classes. To maximize this margin, the algorithm uses support vectors, which are the datapoints that are closer to the set hyperplane. This results in a non-linear optimization problem. Given input vectors of the features $x_i \in \mathbb{R}^p$ and a label vector $y \in \{1, -1\}^n$ the algorithm searches for $w \in \mathbb{R}^p$and $b \in \mathbb{R}^p$ where the prediction given by $sign(w^T \emptyset(x) + b)$ is correct for the largest number of samples. The optimization problem that SV for classification solve can be described as:

$$\min_{w,b,\zeta} \frac{1}{2} w^T w + C \sum_{i=1}^{n} \zeta i$$

$$subject\ to\ y_i(w^T \phi(x_i) + b) \geq 1 - \zeta_i,$$

$$\zeta_i \geq 0, i = 1, \ldots, n$$

Different than in logistic regression, SVM uses values higher than one and lower than -1 to set one label or the other, being this the reinforcement range known as margin. To obtain this output, the decision for a specific datapoint *x* is:

$$\sum_{i \in SV} y_i \alpha_i K(x_i, x) + b$$

The output sign of this equation will give the corresponding label for the sample (Hastie et al., 2009).

3.2.3. Decision tree

Decision trees are machine learning models that use a set of hierarchical decision boundaries based on the input data. Although it is very useful to understand the outputs, as it can be easily interpreted, it is important to keep in mind that they are prone to major overfitting. To solve this issue, decision tree models use *pruning*, a process that removes the unnecessary structures from the decision tree, reducing its complexity. To determine these decision boundaries, the algorithm iteratively tries different split points for our dataset and tries to minimize the cost function given by:

$$E = \sum (p_k * (1 - p_k))$$

Where $p_k$are the proportion of training instances of a specific class given an individual prediction node. If the value equals 0, it means that the split outputs a single class 100% of the class, which would be an ideal case. To stop this process from being infinite, some criterion to stop the construction of tree boundaries needs to be set (Hastie et al., 2009).

In case these algorithms do not provide satisfactory results, we will use other classification methods such as Naïve Bayes methods (H. Zhang, 2004), ensemble methods (Breiman, 1996), or supervised neural networks (Sperduti & Starita, 1997).

3.3. Training the algorithm

In supervised machine learning, we need to train the algorithm before it can actually predict a label for a data point. To do so, we need to split the data into training data, validation data and testing data. This needs to be done before any data input, to ensure that the splitting does not influence the model. The training data will be used to train the model, while the validation data will be used for tuning the hyperparameters. Then, test data can be used to evaluate the

classification. From our data, 70% will be used for training (80%) and validation (20%). To avoid biases in the model and ensure that the results are not dependent on a simple partition, we will use 5-fold cross-validation, which means that different groups will be randomly made from this stack of data and each time four will be used for training while one will be used for validation. Finally, the remaining 30% will be used to evaluate the classification with data that will have never been used for training.

The data we will use as input in the classification algorithm will come from the 675 single trials (see **Figure 6**.A). For each of these trials, we will select the regions and parameters previously defined in sections **2.4.** Region-wise analysis and **¡Error! No se encuentra el origen de la referencia.**. This will leave us with a dataset composed of a limited number of features and characteristics of the signals corresponding to the selected regions of interest. The same process will be followed using the raw data, as, by choosing specific regions, the data dimensionality can be heavily reduced, and it is possible to avoid feature-selection, which may be unsuccessful in complex signals.

3.4. Classification evaluation

To determine how good a classification algorithm works for our given data, we can compare the output label for a specific number of samples to their actual label, given by the ground truth. To compare these two values, we can use different methods. To do so, we first need to understand that a binary model produces four kinds of values:

- **True positives (TP)**: predictions of A (in our case painful stimulus) that are correct according to ground truth.
- **True negatives (TN)**: predictions of B (in our case non-painful stimulus) that are correct according to ground truth.
- **False positives (FP)**: predictions of A (in our case painful stimulus) that are incorrect according to ground truth.
- **False negatives (FN)**: predictions of B (in our case painful stimulus) that are incorrect according to ground truth.

3.4.1. Classification matrix

This matrix refers to the representation of how many samples of each type were classified with each label. The columns and rows of the classification matrix include the classification's actual values and the predicted ones. In the X axis we can find the actual label, while in the Y axis, the predicted label is represented. The matrix then shows the amount of true and false negative and positive predictions. An example can be seen in **Figure 12**.

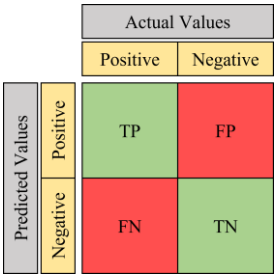

Figure 12: *Classification matrix*

3.4.2. AUC-ROC

The AUC-ROC curve measures the performance of a classification algorithm at different thresholds. ROC is a probability curve, while AUC is a measure of separability between the labels.

The higher the AUC, the better the model is at classifying correctly. To draw the ROC curve, the False Positive Rate (FPS) is plotted on the X axis, while the True Positive Rate is plotted on the Y axis (Hoo et al., 2017). The TPR or recall is calculated through the:

$$TPR = \frac{TP}{TP + FN}$$

While the FPR needs first the calculation of specificity:

$$\text{specificity} = \frac{TN}{TN + FP}$$

$$FPR = 1 - specificity$$

An example of an AUC-ROC curve can be seen in **Figure 13.**

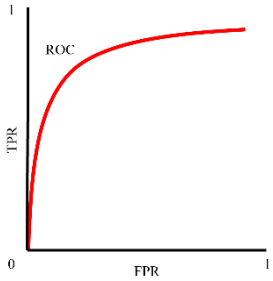

Figure 13: *ROC curve*

3.4.3. Performance measurement values

There is a set of values that are commonly used to determine how well a model is performing. The most common ones include accuracy, precision, recall, and f-1 score (Ping Shung, 2018).

Accuracy describes the correct predictions over the total number of predictions:

$$accuracy = \frac{True\ positives + True\ negatives}{Nº\ of\ predictions}$$

Precision is defined as with the total numbers of positive predictions are correct. So:

$$precision = \frac{True\ positive}{Nº\ of\ correct\ predictions}$$

Recall measures the number of positive cases that were correctly predicted over the true number of positive cases. This is defined as:

$$recall = \frac{True\ positive}{True\ positives + False\ negatives}$$

F-1 score is a measure that combines recall and precision to provide a single measure that weights the two scores:

$$2 * \frac{Precision * Recall}{Precision + Recall}$$

To all these performance measures previously stated, we can apply statistical tests such as the *McNemar test* or other appropriate tests to decide on one model or another.

With these different measures, we will be able to compare the different models and determine if they are good at distinguishing between the different conditions presented. Through this, we will be able to conclude the differences between studies at regional level and studies at voxel-level, and their application in related condition classification. This work will help to establish a basis for brain-wide data analysis and will help define how to analyse fine-grained data in order to obtain more accurate decoding systems for fUSI and get a step closer to BCI applications.

Results

4. Clustering

4.1. Feature selection

Both PCA and ICA were used to reduce the dimensionality of our data In the case of PCA, an explained variance plot was generated to determine the minimum number of principal components required to account for at least 70% of the total variance in the data. The outcomes can be seen on **Figure 14**.

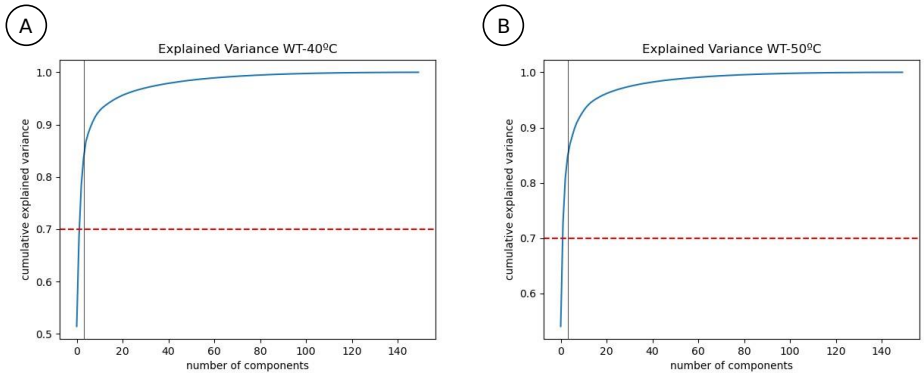

Figure 14: *Explained variances for all the experimental conditions. A)WT-40ºC; B)WT-50ºC.*

Based on these graphs it can be concluded that a minimum of three principal components are sufficient to achieve the desired threshold for explained variance across all experimental conditions. However, it is important to note that achieving the desired level of explained variance does not necessarily ensure satisfactory results for the clustering process. In order to ascertain the relevance of the identified principal components for the clustering algorithm, a plot depicting the three principal components for each experimental condition was generated. The purpose of this plot was to assess whether these components could be partitioned in a meaningful way, thereby ensuring that the information from each of the components would be relevant to the clustering process. The results can be seen on **Figure 15**.

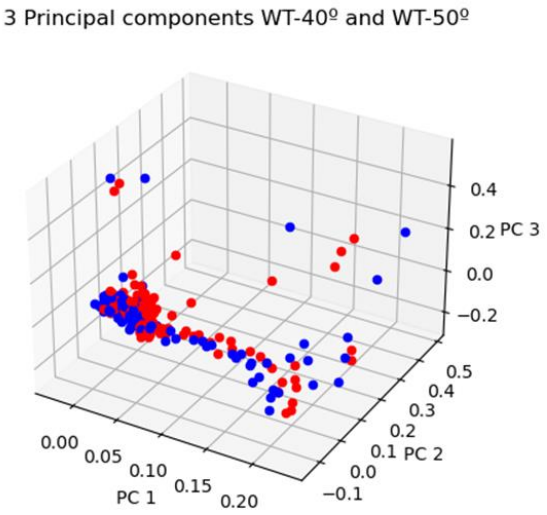

Figure 15: *First 3 Principal components for each region and the different experimental conditions. WT-40º in blue and WT50º in red*

The plots generated from the application of PCA on the data indicate that despite achieving an explained variance of over 70%, there is no clear delineation between the datapoints.

Consequently, the suitability of PCA as a dimensionality reduction technique for our data was considered unsatisfactory and subsequently discarded.

An alternative approach was using ICA to isolate the independent components within the data. To determine the feasibility of this method, ICA was visually inspected to assess the distribution of the data. The results can be seen in **Figure 16**

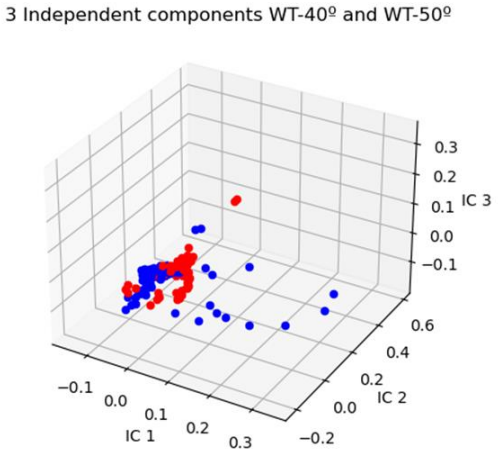

Figure 16: *First 3 Independent components for each region and the different experimental conditions. WT-40º in blue and WT50º in red*

However, the visual inspection of the distributions of the first three independent components once again failed to reveal a clear separation of the data. In view of this, the clustering process was carried out using the original raw data in an attempt to optimize the performance.

4.2. Clustering techniques

For this project 3 main clustering techniques were evaluated: K-means clustering, K-shape clustering and Hierarchical clustering. It is noteworthy that all of these techniques require the determination of the optimal number of clusters prior to their application. Therefore, to ascertain the optimal number of clusters, a comprehensive evaluation was conducted using three distinct tests for each of the clustering techniques. The evaluated range for each of the techniques was between 2 and 12 clusters.

The first test utilized was the silhouette score, that compares the inter-cluster distance to the intra-cluster distance. The highest, the better the clusters are differentiated. The second test employed was the Calinski-Harabasz Index. Similar to the silhouette score, the highest the value, the better the clustering should be. Finally, the Davies-Bouldin was computed. Contrary to the previous tests, a lower index indicates a better clustering. A detailed presentation of the results obtained for the different clustering techniques used in this study is depicted in **¡Error! No se encuentra el origen de la referencia.**.

The analysis of the clustering techniques demonstrates that the three algorithms are suggested different number of clusters. Moreover, discrepancies between the algorithms and evaluation methods are evident. Notably, the option of K=2 was not considered for our study as it would result in a very rudimentary division, and our aim is to identify different activity patterns.

For instance, (**¡Error! No se encuentra el origen de la referencia.** A) indicates that K = 4 would be most appropriate option, while the Calinski-Harabasz Index recommends K = 3, and the Davies-Bouldin Index suggests 3,6 or even 10 clusters. Although both the K-Means and Hierarchical clustering produce similar results, when evaluating the K-shape algorithm yields considerably different outcomes in comparison, and the optimal value for K seems to be around

9 or 10 (**¡Error! No se encuentra el origen de la referencia.** G-I). These differences also appear between clustering techniques. These variations between clustering techniques make it challenging to identify the optimal value of K based solely on the evaluation scores.

Therefore, to determine the optimal number of clusters, we decided to compare different K values on a single technique and evaluate them. We selected Agglomerative Clustering for this purpose as it is straightforward to follow and compare the divisions due to its operational mode. We decided to first study the clustering in the range K = 6 to K = 12 applied to condition 40ºC, as our first approach was to cluster this condition and then predict the labels for condition 50ºC based on it. Two different visualization techniques were used for each K value. The first one with the temporal traces in each cluster and their average (see **Figure 17**) . The second one, using a colormesh of the different values of the temporal traces ordered by their Area Under the Curve (AUC) (see **Figure 18**).

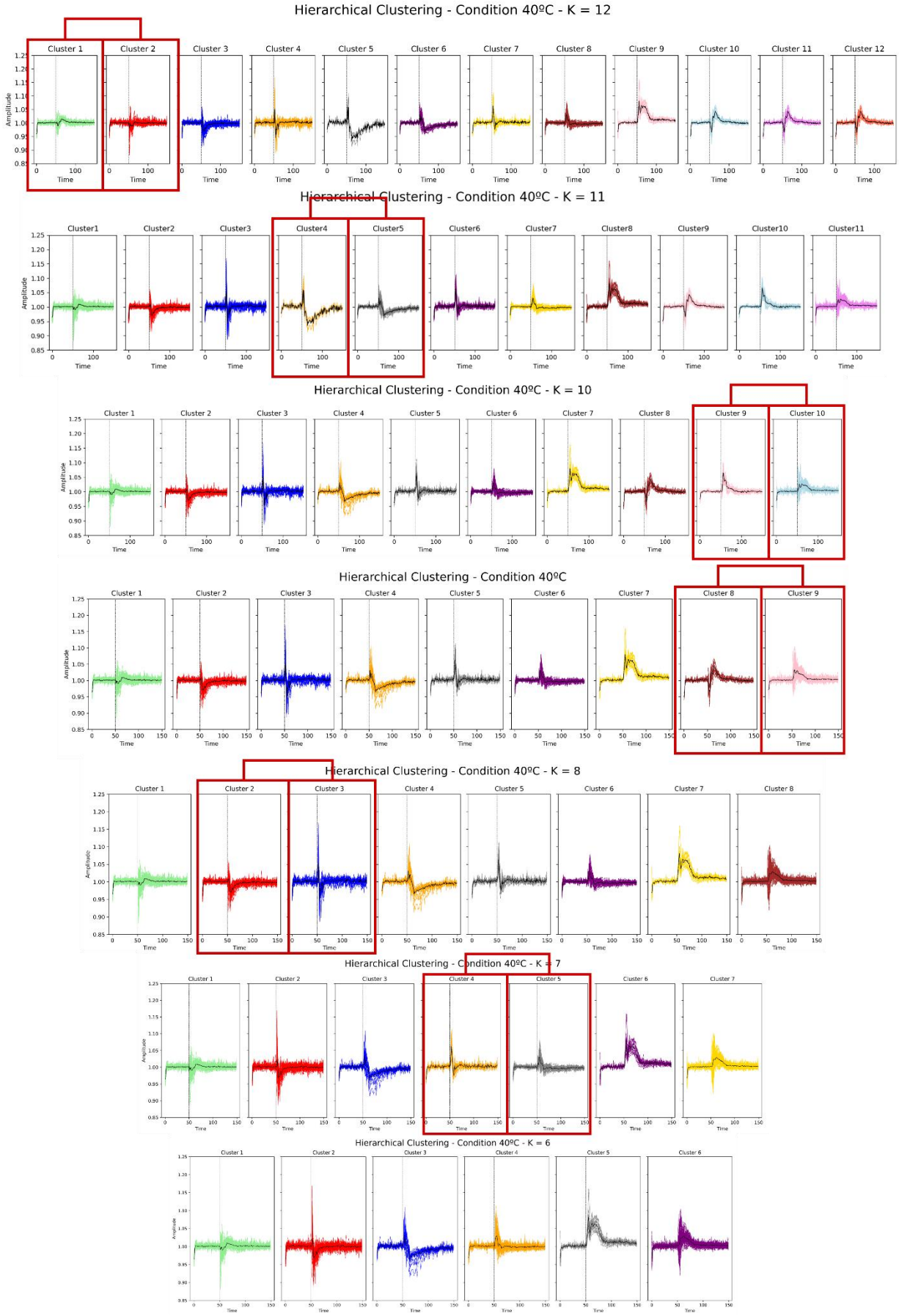

Figure 17:*Representation of the temporal traces clustered using K-Means and different Values of K (from top to bottom, in a range of 2:12). The red squares indicate which clusters are merged in each step.*

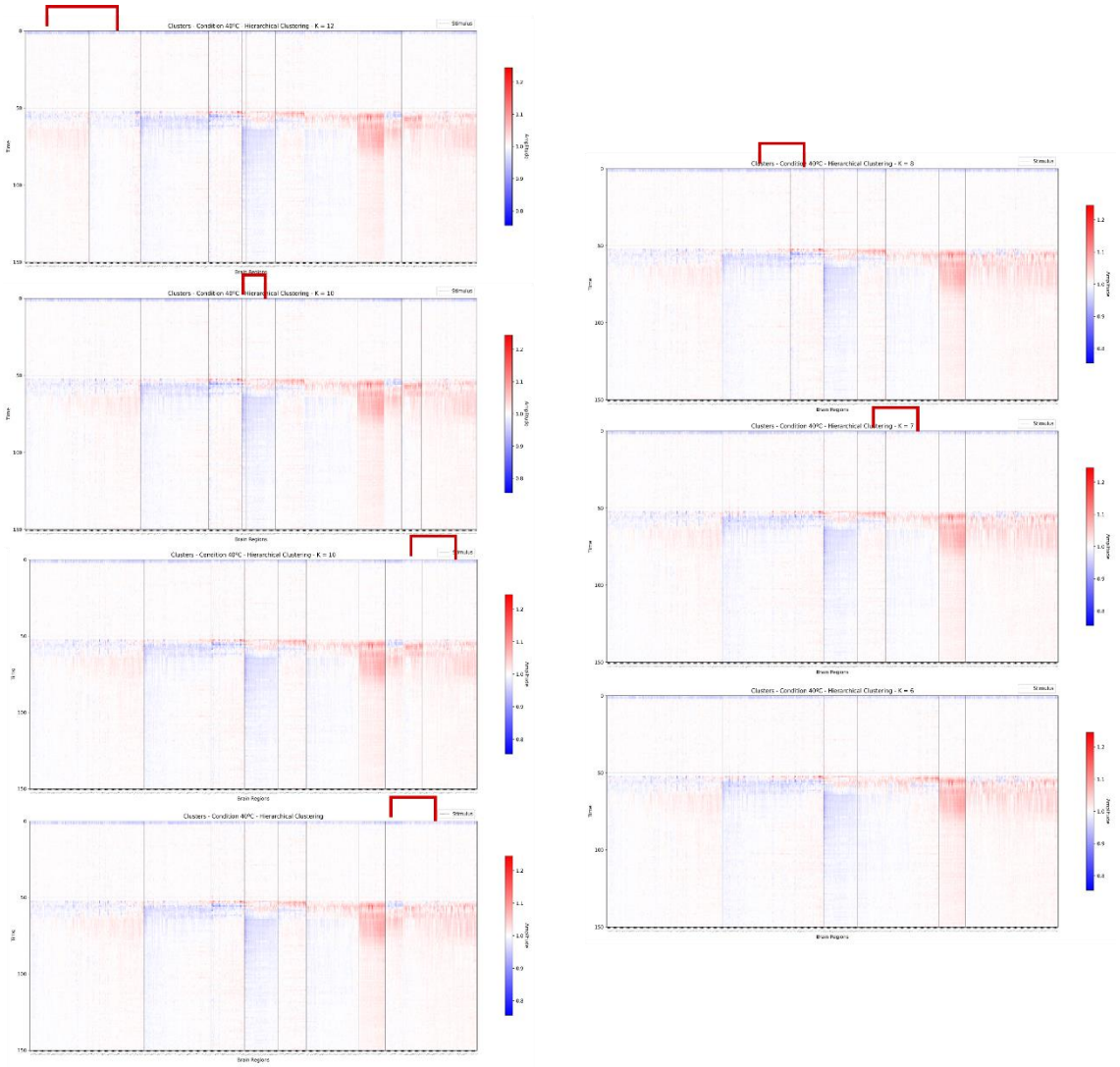

***Figure 18**: Colormesh representation of the temporal traces clustered using K-Means and different Values of K (from top to bottom, in a range of 2:12). The red lines indicate which clusters are merged in each step.*

Based on the presented visualizations, it was concluded that using less than 8 clusters would still leave some clusters with significant outliers from the respective group. Conversely, reducing the number of clusters from 12 to 10 would result in divisions being made in regions that were already relatively similar. Hence, subsequent steps were executed utilizing a clustering approach with a total of 9 clusters.

Next step was to choose the optimal clustering technique. We clustered condition 40ºC using the different clustering techniques and then predicted the labels for condition 50ºC (see **Figure 19**). However, the label prediction for condition 50ºC showed new activity patterns mixed within the existing clusters, leaving space for optimization. Not only this, but the fact of obtaining different activity patterns in each conditions made its comparison difficult, as the change in activity for each region could not be easily followed.

As a result, the approach was changed, and both conditions were clustered together. Furthermore, to ensure adequate space for the incorporation of new activity patterns into clusters, the K value was increased to 12 clusters. This decision was made by applying the previous comparison and concluding that the use of a K value of 12 appeared to be significant.

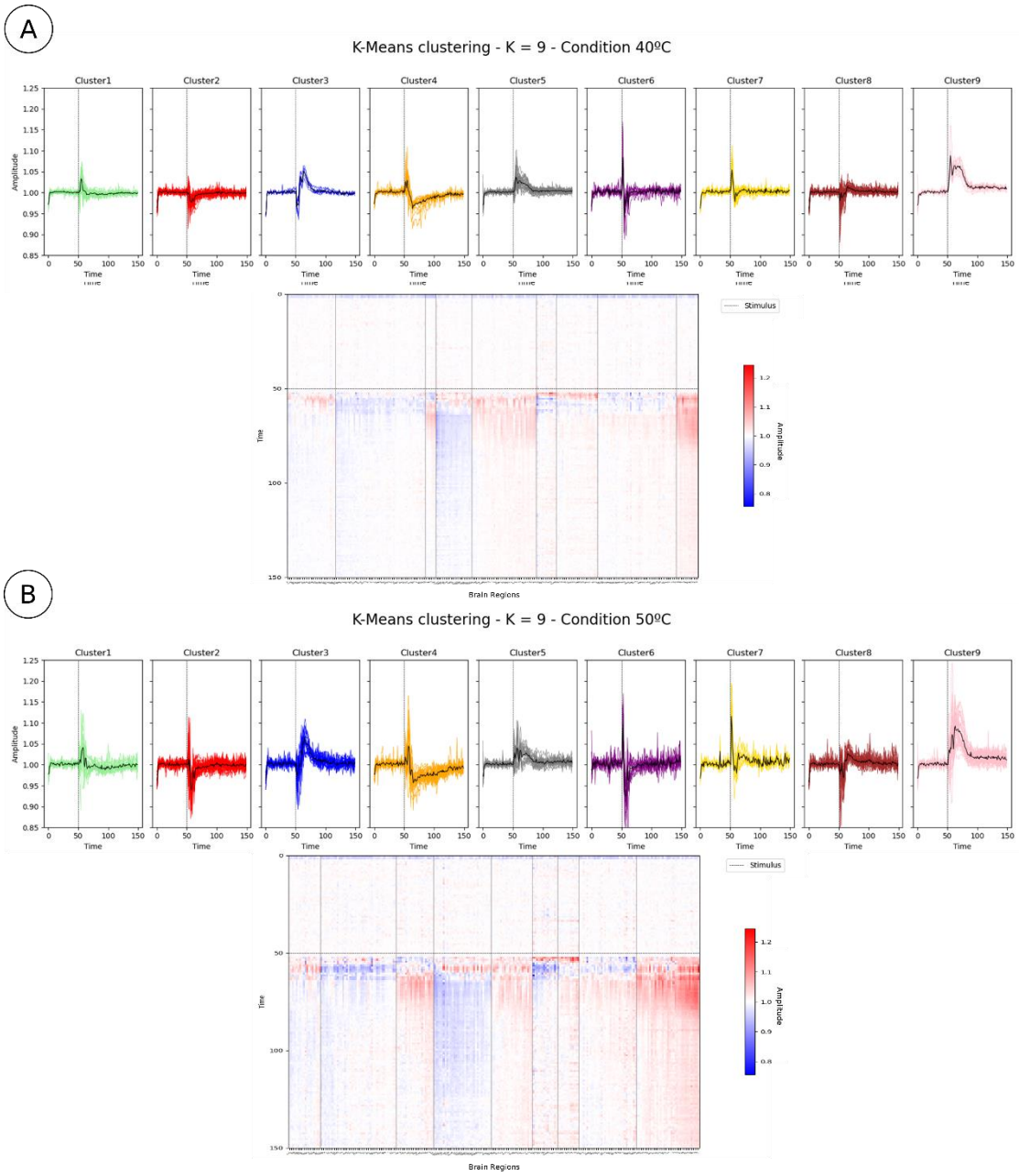

Figure 19: *(**A**) Clustering of the data for condition 40ºC using K-Means and K = 9. On top, the temporal traces by cluster. Under it, a colour mesh of the temporal traces. (**B**) Data from condition 50ºC clustered by predicting the labels for each region depending on the previous cluster applied to condition 40ª. On top, temporal traces by cluster. Under it, a colour mesh of the temporal traces.*

With this new approach, we used the different clustering techniques. First, K-Means, then Hierarchical clustering, and finally, K-shape, all with K = 12. The evaluation of the results showed that the K-shape algorithm performed inferiorly to the other two techniques. Specifically, certain clusters displayed relatively small sizes, while others were less homogeneous compared to the other cases. Moreover, the K-shape algorithm performed slower than the other cases. Thus, it was decided to continue the evaluation process solely for the K-Means and Agglomerative clustering techniques.

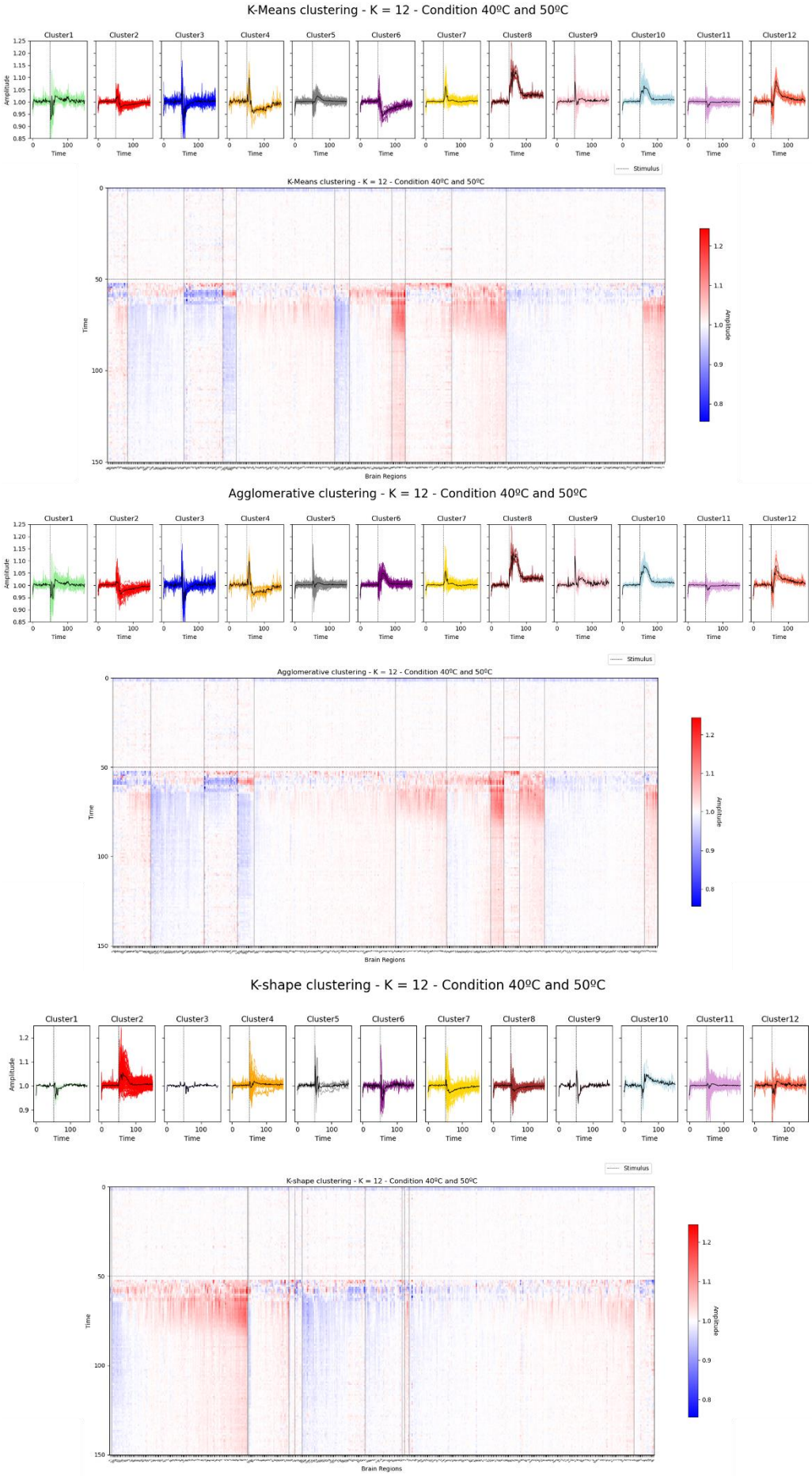

Figure 20: *Clustering of the data using different clustering algorithms.*

4.3. Performance evaluation

To determine the optimal clustering outcome, multiple scores were employed to quantify the results. First, Spearman's correlation was calculated between the samples present in each cluster, to assess their similarity. The intra-cluster average can be seen on **Table 3**. Additionally, a heatmap was generated for each cluster to visually represent the samples present within it. An illustrative example of this approach can be observed in **Figure 21**.

Table 3: *Intra-cluster Spearman correlation of the different clusters using different clustering techniques.*

Cluster #	1	2	3	4	5	6	7	8	9	10	11	12
K-Means	0.38	0.57	0.38	0.83	0.35	0.79	0.24	0.91	0.31	0.64	0.19	0.65
Hierarchical	0.31	0.63	0.43	0.77	0.17	0.46	0.24	0.91	0.63	0.73	0.23	0.74

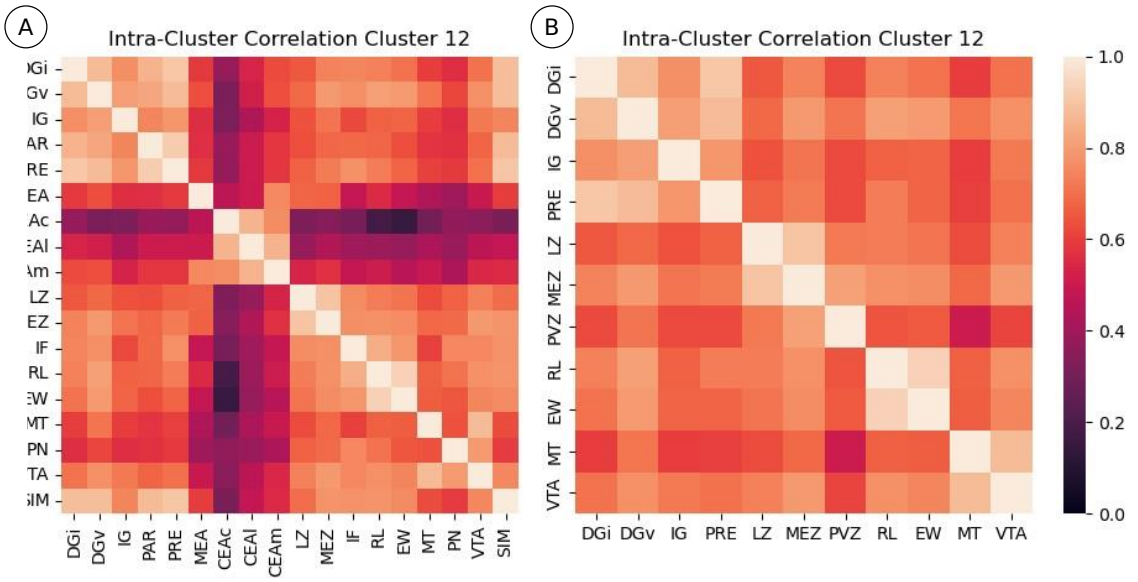

Figure 21: *Intra-Cluster heatmap for Cluster 12. (A) K-means clustering (B) Hierarchical clustering.*

Then we computed the correlation between the different clusters to enable the comparison of inter-cluster correlation with intra-cluster correlation. Results for each clustering technique are presented in **Figure 22**.

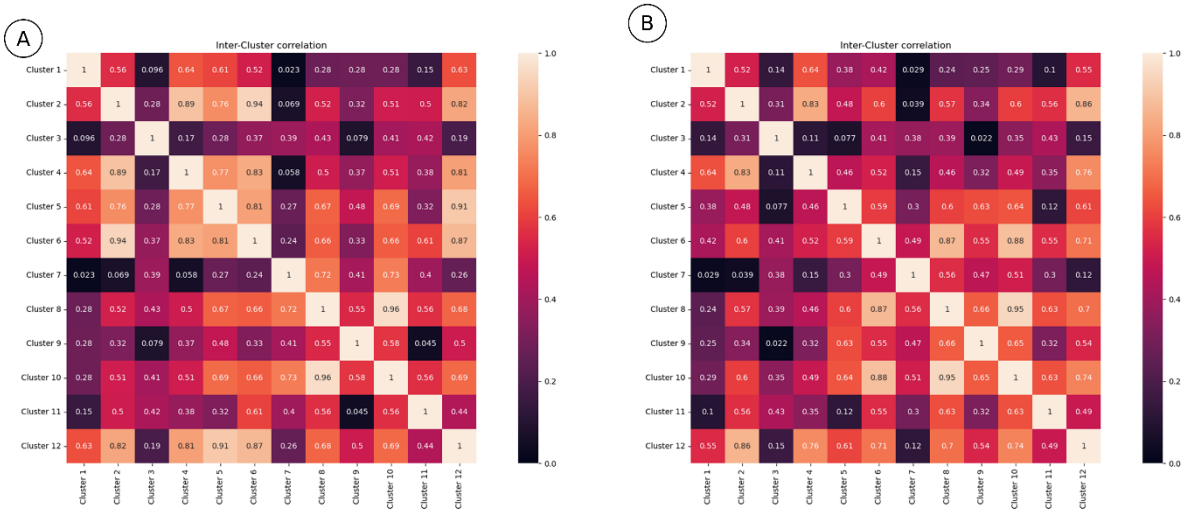

Figure 22: *Inter-Cluster comparison for each of the clustering techniques. (A) K-means clustering (B) Hierarchical clustering.*

Upon analysis, it is evident that the intra-cluster correlation varies considerably between clusters for both clustering techniques, with some demonstrating high correlation values between the samples, while others exhibiting lower values. This trend is also observed in the inter-cluster comparison, with some clusters displaying significant differentiation from others, while others remain quite similar.

However, although it can sometimes give useful insights, Spearman correlation only considers the monotonic relationship between the samples, ignoring the temporal aspect. To overcome this limitation and provide a comprehensive comparison of the results, we also computed cross-correlation between the time series.

The same procedure was done using the Euclidean distance as a measure Notably, a lower value on this measure implies greater similarity between samples. Results can be seen in **Table 4**.

Table 4: *Intra-cluster Spearman correlation of the different clusters using different clustering techniques.*

Cluster #	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
K-Means	0.22	0.13	0.21	0.16	0.12	0.17	0.15	0.17	0.17	0.17	0.11	0.20
Hierarchical	0.22	0.17	0.20	0.18	0.13	0.16	0.16	0.17	0.17	0.16	0.10	0.19

The Euclidean distance between clusters can be seen in **Figure 23**.

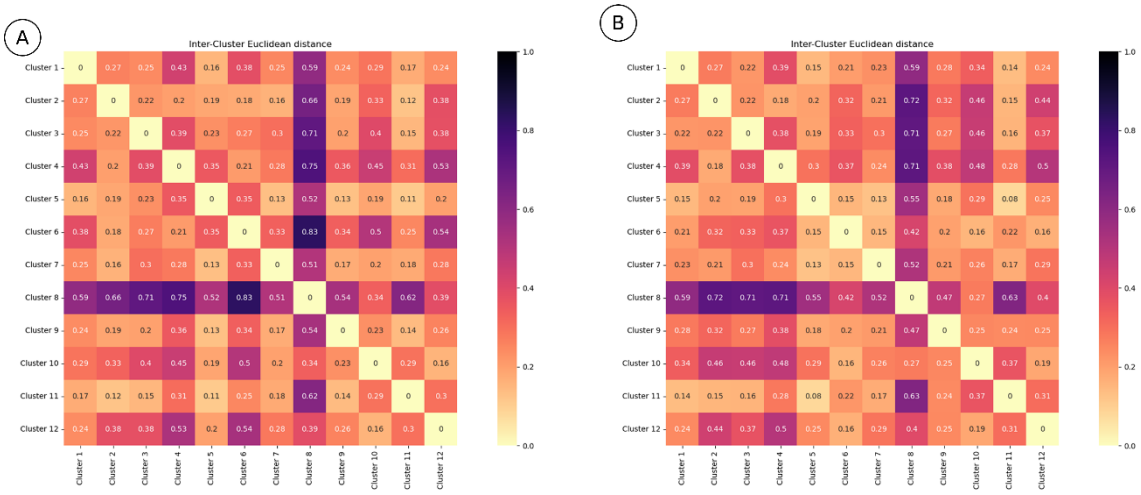

Figure 23: *Inter-Cluster Euclidean distance for each of the clustering techniques. (A) K-means clustering (B) Hierarchical clustering.*

The results obtained from both clustering techniques were consistent with Spearman correlation. Given the positive outcomes of both approaches and the complexity of comparing time series efficiently, Hierarchical clustering was ultimately selected due to its implementation. When interpreting the data, this technique offers additional information through the Dendrogram regarding the merging clusters, providing a better understanding of the relationships between them. This insight enables the identification of brain regions with stronger connections and a greater degree of interdependence.

4.4. Region-wise analysis

Once the clustering methodology was decided, the dataset was clustered using Agglomerative clustering with K=12. The final result with the stimulus separated can be seen in **Figure 24**.

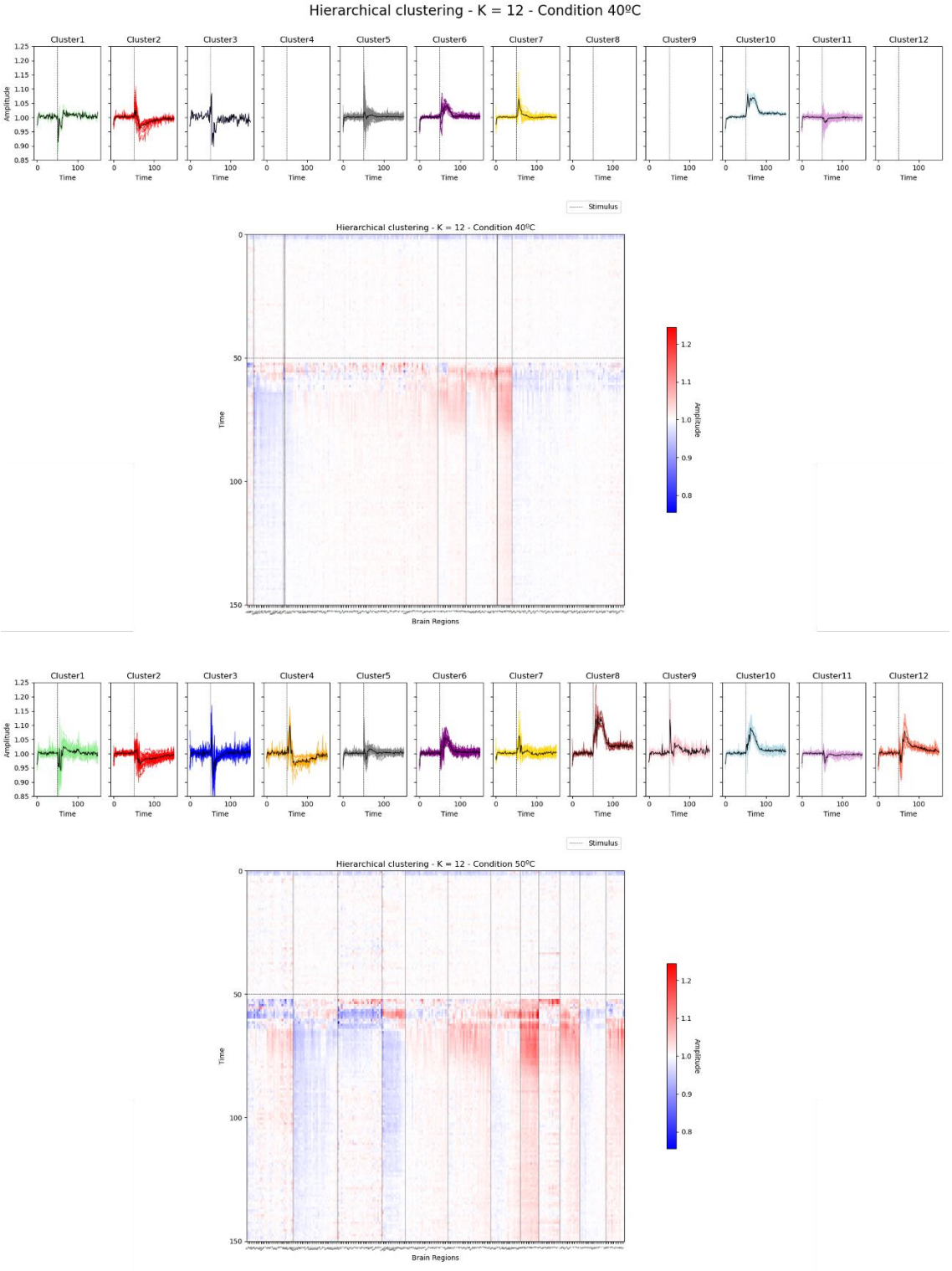

Figure 24: *Hierarchical Clustering with K = 12 applied to all the dataset at the same time. For visualization purposes, each stimulus has been visualized separately using the labels obtained in the joined clustering.*

The subsequent step in the analysis process was to determine the regions of interest for the subsequent procedures. Utilizing the clustering results, we were able to identify four clusters in which no regions were present for condition 40ºC, indicating that these activity patterns were specific to condition 50ºC. Consequently, these clusters likely represent the most intriguing regions in terms of identifying each stimulus. To further assess this and minimize the number of regions employed, we characterized the average of each cluster using a set of common signal features. For each average, we visualized these features. An example of this can be seen in **Figure 25**.

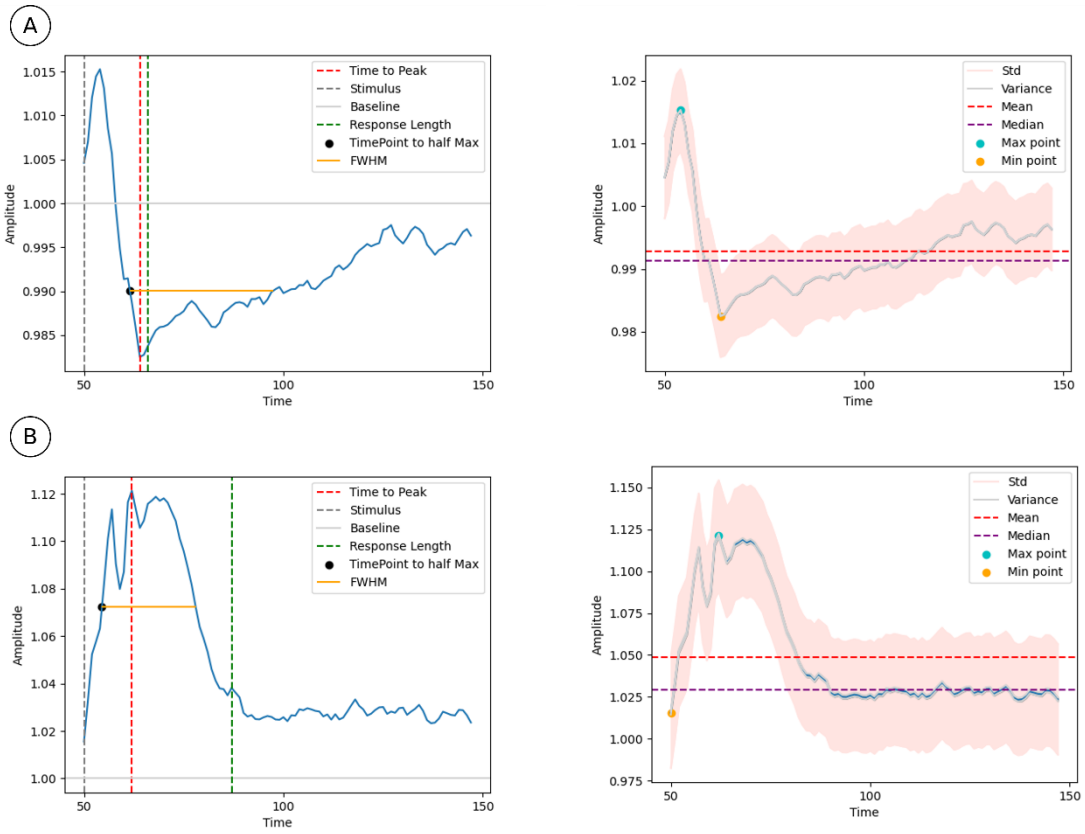

Figure 25: *Feature visualization in two different cluster averages. (A) Features in Cluster 4. (B) Features in Cluster 8.*

Once we computed these features for the average of each group, we took the cluster with the maximum value on a specific feature and the cluster with the minimum value. Following this, an assessment was carried out to determine which regions corresponding to the 40ºC stimulus changed from the lowest to the highest or vice versa when the 50ºC stimulus was applied, as this would indicate the greatest change. This was done for each of the selected features. However, for most of the features, the minimum or maximum value resulted to appear in cluster 8, which had only regions moving from other clusters to it, as the specific pattern only appears in condition 50ºC. Nevertheless, when observing from which clusters the regions came, we observed that the movements were from Cluster 6,9 or 10, which were similar in pattern to Cluster 8. Therefore, we used the features that did not involve cluster 8, resulting in Cluster 11 and Cluster 12 having the largest activity difference.

4.5. Qualitative cluster description

To be able to clearly refer to each cluster, compare them, and define the different activity patterns in each, a short qualitative description was made for each of them taking into account their main visible characteristics. These qualitative descriptions were made involving neurobiology experts and data analysts to ensure a correct and relevant description for each group.

Nº Cluster	Main characteristics	Qualitative description
1	- Deactivation after stimulus - Minimal activation after	Short term deactivation
2	- Minimal activation - Small deactivation after	Weak activation before weak decay
3	- Strong activation - Strong rapid deactivation - Rapid return to baseline	Strong peak with rapid deactivation
4	- Strong activation - Small deactivation after	Strong peak with mid-term deactivation
5	- Very small deactivation - Small, maintained activation	Weak deactivation with mid-term activation
6	- Medium, maintained activation	Medium long-term activation
7	- Small, rapid activation	Weak short-term activation
8	- Very strong, maintained activation	Very strong long-term activation
9	- Strong activation - Rapid return to baseline - Small, maintained activation	Strong short-term activation with re-activation
10	- Strong, maintained activation	Strong long-term activation
11	- Very small deactivation	Small mid-term deactivation
12	- Small, rapid deactivation - Strong maintained activation	Weak deactivation with long-term strong activation

By using these qualitative descriptions, it is possible to easily compare clusters while keeping in mind the different activity patterns for further discussion.

4.6. Clustering Visualization

Due to the large volume of data managed in the clustering, its interpretation is not easy. Although the clustering quantification techniques can be used to choose which regions to use it is important to also relate this quantification to the background data, in this case, to understand how the different regions are behaving, and how the changes are happening across the brain. For this reason, a visualization tool was created to unlock the visualization of the movement of the different brain regions within the clusters depending on their condition. Through this, it is possible to see how each region behaves when applied different stimuli, and it can help make further conclusions on the topic. With this purpose, an interactive Sankey diagram was created. It is important to mention that this tool also includes data from genetically modified mice (TKO mice) that is not used in this project. However, the visualization of the Wild type mice clustering can be done using it and matches the results presented in this work. The diagram can be seen in **Figure 26**.

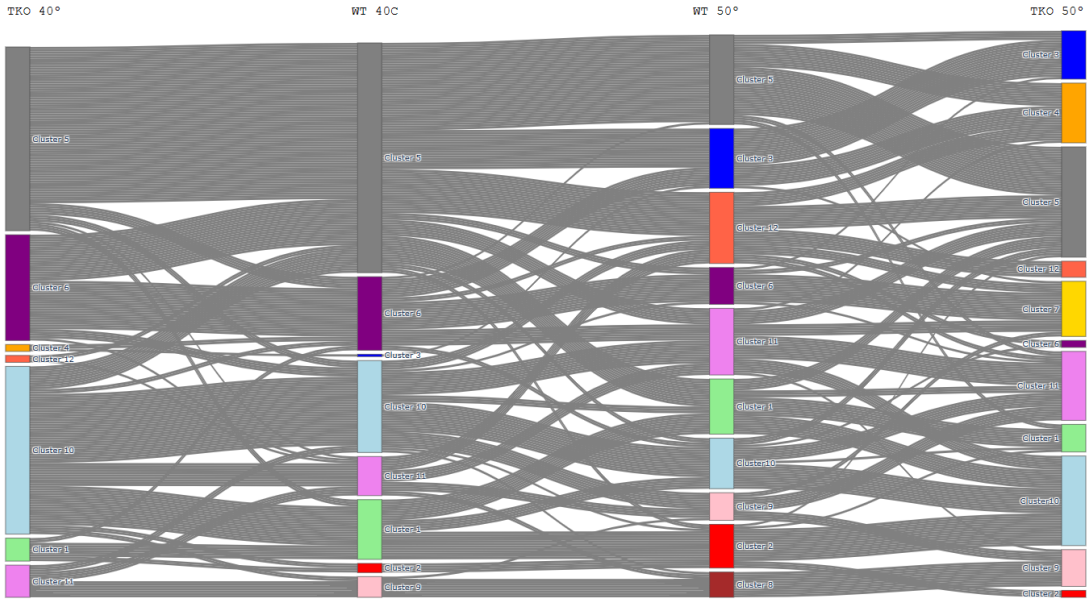

***Figure 26**: Sankey diagram for the visualization of brain region movement within clusters.*

From this visualization, it is possible to take specific regions and see the changes in their activity pattern. Through the qualitative descriptions made in section 1.5, it is possible to understand the changes in activity for each region depending on their condition. This gives new insights in the behaviour of each region in each condition. For example, if a region is in cluster 5 for condition 40ºC and then we find the same region in cluster 12 for condition 50º, we know that, when applying pain to the animal, this specific region reacts with a stronger, longer activation. Maybe add examples of specific regions that have been found interesting.

5. Classification

5.1. Dataset Split and feature extraction

Before actually classifying our data, we cleaned the original dataset as there were some trials that were found not to have been properly registered during the experiments through the observation of video-recorded data. From the original 675 trials, 521 were finally used, with 396 trials for the 40ºC condition and 125 trials for the 50ºC condition. Apart, some trials had some regions which showed plain signals. However, as not all the regions were used for classification, and to avoid eliminating all the trial, we only discarded specific trials if the regions with plain signals were being used for classification. We also eliminated those regions that ended with a low number of trials ($N_{50ºC\ condition} < 100$) from all the procedure, as they could not be compared to the other regions. The non-used regions can be found on **Annex 1**. We then split our dataset into train (70%) and test (30%) groups. To approach the data in different ways, we used not only the raw data to classify, but we also extracted specific signal features to compare the results of both approaches. The specific features extracted, and the specific packages used to compute them can be found in

Table 5: Features extracted from each trial and Python functions used.

Feature	Function
Skewness	*Skew* function from *stats* module from *scipy* package version 1.10.0
Kurtosis	*Kurtosis* function from *stats* module from *scipy* package version 1.10.0
Maximum peak	*Max* function from Python 3.10.9
Minimum peak	*Min* function from Python 3.10.9

Difference between maximum and minimum peaks	Max-Min peak value
Time to maximum peak	Time to max peak
Response length	1. Find signal derivatives with *gradient* function from *numpy* package version 1.23.5 2. Find longest sequence of negative derivatives 3. Find last time point of the longest sequence of negative derivatives
Full width at half maximum (FWHM)	1. Find absolute maximum timepoint 2. Find width of the absolute maximum peak at a 0.5 relative height using *peak_width* function from *signal* module from *scipy* package version 1.10.0
Time between maximum and minimum peak	Absolute time difference between max and min peaks.
Permutation entropy	*Perm_entropy* function from *antropy* package version 0.1.5
Singular Value Decomposition(Svd) entropy	*Svd_entropy* function from *antropy* package version 0.1.5
Spectral Entropy	*Spectral_entropy* function from *antropy* package version 0.1.5
Approximate entropy	*App_entropy* function from *antropy* package version 0.1.5
Sample entropy	*Sample_entropy* function from *antropy* package version 0.1.5

These features were chosen based on the cluster characterization made in **4.4**, looking to explain our time traces as good as possible. Apart, common spectral features were added to avoid limiting our feature extraction to temporal analysis. To select the best features to train the classifier, we performed an ANOVA test to discard half of the features given its relevance to explain the dataset. The test was performed using the *SelectKBest* function from the *feature_selection* module of the *scipy* package, aiming to choose the best 7 features. The resulting selected features were skewness, kurtosis, maximum peak, minimum peak, FWHM, Spectral Entropy and Svd Entropy.

5.2. Pre-processing and classifier selection

Given our dataset, we performed data-augmentation to balance the trials in both conditions. To do so, we computed the standard deviation of the noise for each signal and added a random noise value in the range of the standard deviation, to obtain significant trials. After this, we could begin the classifier selection process.

To choose the best classifier and data (raw or features) to use, we trained different classifiers using a 8 regions set selected randomly. Then, using 5-fold cross-validation (with a train-validation partition of 80% and 20%) and grid selection, we chose the best hyperparameters for each of the three pre-selected classifiers. Finally, we trained each classifier using the same regions and the optimized hyperparameters and compared their accuracy to decide which one to use. To train the classifiers and predict results we used models from the *Sklearn* package version 1.2.1. Results can be seen in

	Logistic Regression	SVM	Decision Trees
Raw Data	78%	89%	77%
Selected Features	77%	76%	73%

Given these results, we decided to proceed with the study using SVM in the raw data using the following hyperparameters: random_state = 1, C = 2, max_iter = -1, coef0 = 0, degree=2, gamma = 'scale', tol = 1, kernel = 'rbf', shrinking= True, decision_function_shape= 'ovo', probability= True.

5.3. Region selection methods

Once the data, classifier and parameters had been established, we proceeded to compare the different region selection methods to define how each process affected the classifier accuracy.

5.3.1. Data-based selection

We started calculating the individual accuracy for each region. Then, we selected the region with highest accuracy, and paired it with all the regions that had an individual accuracy above 70%. Each time we trained new classifiers and used the same train/test set. We repeated the process, taking the best pair and creating groups of three with all the regions with a higher individual accuracy than 70%. This process was repeated until the classifier started to get confused when adding regions, resulting in lower accuracy. This resulted in a group of 8 regions: ['POL' 'MG' 'PP' 'SSp-bf' 'LGv' 'DGv' 'SF' 'CA2v']

5.3.2. Cluster-based region selection

As previously stated in **4.4** we used those regions that moved from Cluster 11 to Cluster 12. This resulted in using the following regions: 'MEA', 'COAa', 'PIR', 'IA', 'BLAv', 'BMAa','EPv', 'CA3i','CA3v'

5.3.3. Cortical regions

Here we selected all those regions from the cortical space of the brain. However, many regions fall into this category, so to avoid confusing the classifier we followed the same process than in 5.3.1 but only applied to the cortical regions, to select the best cortical group to use. This resulted in the following regions: ['ACAd', 'ACAv','AUDd','AUDpo','AUDv','MOp','MOs','RSPagl','RSPd']

5.3.4. Knowledge-based selection

Finally, we chose regions from a biological system commonly related to pain processing. In this case we used regions belonging to the high-level of the somatosensory pathway. This means that, although other brain regions might also be related to it, we only took the most important ones that perform the main processing activities. This resulted in the following regions: ['SSp-bf','SSp-n','SSp-tr','SSp-ll','SSp-ul','SSp-un','SSs','CM','VM','VP']

5.3.5. Comparison of region selection methods

Once the groups of regions were selected, we compared their performance measurement values to define which region selection method performed better for our data. Results can be seen in

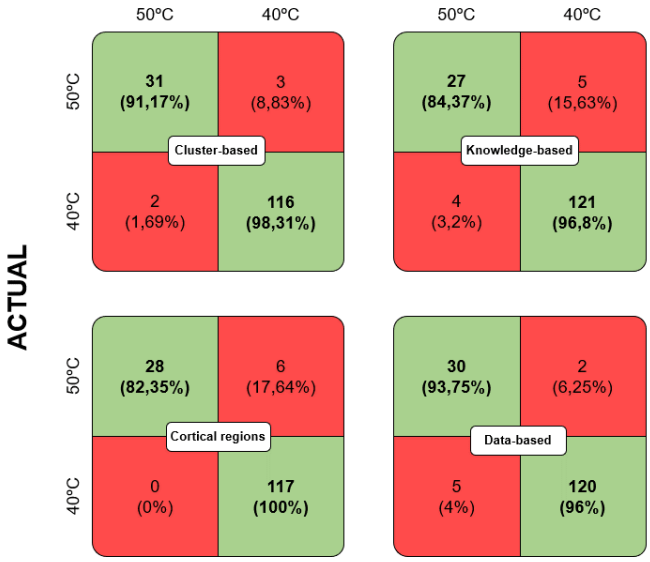

Figure 27: *Confusion matrix of the classification values obtained in the test set for each of the region selection methods.*

Annex 1: Regions not used for classification due to having multiple trials with incorrect registration.

LING	SPVO	B	NTS	ACVII	VI
LC	ANcrl	COPY	VNC	IP	SG
DN	ANcr2	PRM	PHY	y	x

Bibliography:

Abiri, R., Borhani, S., Sellers, E. W., Jiang, Y., & Zhao, X. (2019). A comprehensive review of EEG-based brain–computer interface paradigms. *Journal of Neural Engineering*, *16*(1), 011001.

Aggarwal, S., & Chugh, N. (2019). Signal processing techniques for motor imagery brain computer interface: A review. *Array*, *1*, 100003.

Bamdad, M., Zarshenas, H., & Auais, M. A. (2015). Application of BCI systems in neurorehabilitation: a scoping review. *Disability and Rehabilitation: Assistive Technology*, *10*(5), 355–364.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, *3*(Jan), 993–1022.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*, 123–140.

Brunner, C., Grillet, M., Sans-Dublanc, A., Farrow, K., Lambert, T., Macé, E., Montaldo, G., & Urban, A. (2020). A Platform for Brain-wide Volumetric Functional Ultrasound Imaging and Analysis of Circuit Dynamics in Awake Mice. *Neuron*, *108*(5), 861-875.e7. https://doi.org/10.1016/j.neuron.2020.09.020

Cano, S., Soto, J., Acosta, L., Peñeñory, V. M., & Moreira, F. (2022a). Using Brain-Computer Interface to evaluate the User eXperience in interactive systems. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 1–9.

Cano, S., Soto, J., Acosta, L., Peñeñory, V. M., & Moreira, F. (2022b). Using Brain-Computer Interface to evaluate the User eXperience in interactive systems. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 1–9.

Casson, A. J., Abdulaal, M., Dulabh, M., Kohli, S., Krachunov, S., & Trimble, E. (2018). Electroencephalogram. In *Seamless healthcare monitoring* (pp. 45–81). Springer.

Claron, J., Royo, J., Arcizet, F., Deffieux, T., Tanter, M., & Pouget, P. (2021). The Supplementary Eye Field Tracks Cognitive Efforts. *BioRxiv*.

Constable, R. T. (2006). Challenges in fMRI and its limitations. In *Functional MRI* (pp. 75–98). Springer.

Cutrell, E., & Tan, D. (2008). BCI for passive input in HCI. *Proceedings of CHI*, *8*, 1–3.

Dash, D., Wisler, A., Ferrari, P., Davenport, E. M., Maldjian, J., & Wang, J. (2020). MEG sensor selection for neural speech decoding. *IEEE Access*, *8*, 182320–182337.

Demene, C., Bernal, M., Delanoe, C., Auvin, S., Biran, V., Alison, M., Mairesse, J., Harribaud, E., Pernot, M., & Tanter, M. (2016). Functional ultrasound imaging of the brain activity in human neonates. *2016 IEEE International Ultrasonics Symposium (IUS)*, 1–3.

Dhanabalan, T., Jose, N. C., & Chukwuka, O. (2022). *Non-invasive brain-computer interface for prosthetic limbs control in Military*.

Diya, S. Z., Prorna, R. A., Rahman, I. I., Islam, A. B., & Islam, M. N. (2019). Applying brain-computer interface technology for evaluation of user experience in playing games. *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, 1–6.

Dizeux, A., Gesnik, M., Ahnine, H., Blaize, K., Arcizet, F., Picaud, S., Sahel, J.-A., Deffieux, T., Pouget, P., & Tanter, M. (2019). Functional ultrasound imaging of the brain reveals propagation of task-related brain activity in behaving primates. *Nature Communications*, *10*(1), 1–9.

Dodia, S., Edla, D. R., Bablani, A., Ramesh, D., & Kuppili, V. (2019). An efficient EEG based deceit identification test using wavelet packet transform and linear discriminant analysis. *Journal of Neuroscience Methods*, *314*, 31–40.

Du, B., Cheng, X., Duan, Y., & Ning, H. (2022). fMRI Brain Decoding and Its Applications in Brain–Computer Interface: A Survey. *Brain Sciences*, *12*(2), 228.

Dupret, G. E., & Piwowarski, B. (2008). A user browsing model to predict search engine click data from past observations. *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 331–338.

Durand, S., Heller, G. R., Ramirez, T. K., Luviano, J. A., Williford, A., Sullivan, D. T., Cahoon, A. J., Farrell, C., Groblewski, P. A., & Bennett, C. (2022). Acute head-fixed recordings in awake mice with multiple Neuropixels probes. *Nature Protocols*, 1–34.

Fahimi, F., Zhang, Z., Goh, W. B., Lee, T.-S., Ang, K. K., & Guan, C. (2019). Inter-subject transfer learning with an end-to-end deep convolutional neural network for EEG-based BCI. *Journal of Neural Engineering*, *16*(2), 026007.

Fu, R., Tian, Y., Bao, T., Meng, Z., & Shi, P. (2019). Improvement motor imagery EEG classification based on regularized linear discriminant analysis. *Journal of Medical Systems*, *43*, 1–13.

Gannouni, S., Belwafi, K., Aboalsamh, H., AlSamhan, Z., Alebdi, B., Almassad, Y., & Alobaedallah, H. (2020). EEG-based BCI system to detect fingers movements. *Brain Sciences*, *10*(12), 965.

Gezgez, C., & Kaçar, E. (2021). Virtual Character Control by Brain-Computer Interface and Comparison of Performance Metrics. *2021 International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*, 1–7.

Glavas, K., Prapas, G., Tzimourta, K. D., Tzallas, A. T., Giannakeas, N., & Tsipouras, M. G. (2022). Intra-User Analysis Based on Brain-Computer Interface Controlled Game. *2022 45th International Conference on Telecommunications and Signal Processing (TSP)*, 386–390.

Guyon, I., von Luxburg, U., & Williamson, R. C. (2009). Clustering: Science or art. *NIPS 2009 Workshop on Clustering Theory*, 1–11.

Han, C.-H., Müller, K.-R., & Hwang, H.-J. (2020). Enhanced performance of a brain switch by simultaneous use of EEG and NIRS data for asynchronous brain-computer interface. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *28*(10), 2102–2112.

Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction* (Vol. 2). Springer.

Hernandez-Cuevas, B., Egbert, W., Denham, A., Mehul, A., & Crawford, C. S. (2020). Changing Minds: Exploring Brain-Computer Interface Experiences with High School Students. *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, 1–10.

Hoo, Z. H., Candlish, J., & Teare, D. (2017). What is an ROC curve? In *Emergency Medicine Journal* (Vol. 34, Issue 6, pp. 357–359). BMJ Publishing Group Ltd and the British Association for Accident ….

Hoyer, P. O. (2004). Non-negative matrix factorization with sparseness constraints. *Journal of Machine Learning Research*, *5*(9).

Jansen, A., Flöel, A., Deppe, M., van Randenborgh, J., Dräger, B., Kanowski, M., & Knecht, S. (2004). Determining the hemispheric dominance of spatial attention: a comparison between fTCD and fMRI. *Human Brain Mapping*, *23*(3), 168–180.

Jeyakumar, V., Krishnan, P. T., Sundaram, P., & Raj, A. N. J. (2022). Brain-computer interface in Internet of Things environment. In *5G IoT and Edge Computing for Smart Healthcare* (pp. 231–255). Elsevier.

Kamrani, E. (2014). *Title: On-Chip Integrated Functional Near Infra-Red Spectroscopy (fNIRS) Photoreceiver for Portable Brain Imaging Référence*. https://publications.polymtl.ca

Karanam Shashmi. (2021, August 11). *Curse of Dimensionality — A "Curse" to Machine Learning*. Towards Data Science.

Khalaf, A., Sejdic, E., & Akcakaya, M. (2019a). Common spatial pattern and wavelet decomposition for motor imagery EEG-fTCD brain-computer interface. *Journal of Neuroscience Methods*, *320*, 98–106.

Khalaf, A., Sejdic, E., & Akcakaya, M. (2019b). EEG-fTCD hybrid brain–computer interface using template matching and wavelet decomposition. *Journal of Neural Engineering*, *16*(3), 036014.

Khosla, N. (2004). Dimensionality reduction using factor analysis. *Griffith University, Australia*.

Krishnan, S., & Athavale, Y. (2018). Trends in biomedical signal feature extraction. In *Biomedical Signal Processing and Control* (Vol. 43, pp. 41–63). Elsevier Ltd. https://doi.org/10.1016/j.bspc.2018.02.008

Kumar, S., & Sharma, M. (2012). BCI: Next generation for HCI. *International Journal of Advanced Research in Computer Science and Software Engineering*.

Kundu, S., & Ari, S. (2022). Brain-Computer interface speller system for alternative communication: a review. *IRBM*, *43*(4), 317–324.

Kurita, T. (2019). Principal component analysis (PCA). *Computer Vision: A Reference Guide*, 1–4.

Ladecola, C. (2017). The neurovascular unit coming of age: a journey through neurovascular coupling in health and disease. *Neuron*, *96*(1), 17–42.

Leeb, R., Tonin, L., Rohm, M., Desideri, L., Carlson, T., & Millán, J. D. R. (2015). Towards independence: A BCI telepresence robot for people with severe motor disabilities. *Proceedings of the IEEE*, *103*(6), 969–982. https://doi.org/10.1109/JPROC.2015.2419736

Li, M., Chai, Q., Kaixiang, T., Wahab, A., & Abut, H. (2009). EEG emotion recognition system. *In-Vehicle Corpus and Signal Processing for Driver Behavior*, 125–135.

Lo, Y. T., Premchand, B., & Libedinsky, C. (2022). Neural correlates of learning in a linear discriminant analysis brain-computer interface paradigm. *Journal of Neural Engineering*, *19*(5), 056041.

Macé, É., Montaldo, G., Cohen, I., Baulac, M., Fink, M., & Tanter, M. (2011). Functional ultrasound imaging of the brain. *Nature Methods*, *8*(8), 662–664. https://doi.org/10.1038/nmeth.1641

Macé, É., Montaldo, G., Trenholm, S., Cowan, C., Brignall, A., Urban, A., & Roska, B. (2018). Whole-Brain Functional Ultrasound Imaging Reveals Brain Modules for Visuomotor Integration. *Neuron*, *100*(5), 1241-1251.e7. https://doi.org/10.1016/j.neuron.2018.11.031

Mahajan, S., Hermann, J. K., Bedell, H. W., Sharkins, J. A., Chen, L., Chen, K., Meade, S. M., Smith, C. S., Rayyan, J., & Feng, H. (2020). Toward standardization of electrophysiology and computational tissue strain in rodent intracortical microelectrode models. *Frontiers in Bioengineering and Biotechnology*, *8*, 416.

Maher, O. N., Haikal, A. Y., Elhosseini, M. A., & Saafan, M. (2023). An Optimized Quadratic Support Vector Machine for EEG Based Brain Computer Interface. *International Journal of Electrical and Computer Engineering Systems*, *14*(1), 83–91.

Miller, J. K., Miller, B. R., & Yuste, R. (2018). An increase in spontaneous activity mediates visual habituation. *BioRxiv*, 507814.

Min, B.-K., Marzelli, M. J., & Yoo, S.-S. (2010). Neuroimaging-based approaches in the brain–computer interface. *Trends in Biotechnology*, *28*(11), 552–560.

Montaldo, G., Urban, A., & Macé, E. (2022). *Annual Review of Neuroscience Functional Ultrasound Neuroimaging*. https://doi.org/10.1146/annurev-neuro-111020

Mridha, M. F., Das, S. C., Kabir, M. M., Lima, A. A., Islam, M. R., & Watanobe, Y. (2021). Brain-computer interface: Advancement and challenges. *Sensors*, *21*(17), 5746.

Neisser, U. (1997). *Cognitive psychology: Classic edition*. Psychology press.

Nicolas-Alonso, L. F., & Gomez-Gil, J. (2012). Brain computer interfaces, a review. *Sensors*, *12*(2), 1211–1279.

Nijholt, A., Tan, D., Allison, B., del R. Milan, J., & Graimann, B. (2008). Brain-computer interfaces for HCI and games. In *CHI'08 extended abstracts on Human factors in computing systems* (pp. 3925–3928).

Norman, S. L., Maresca, D., Christopoulos, V. N., Griggs, W. S., Demene, C., Tanter, M., Shapiro, M. G., & Andersen, R. A. (2021). Single-trial decoding of movement intentions using functional ultrasound neuroimaging. *Neuron*, *109*(9), 1554-1566.e4. https://doi.org/10.1016/j.neuron.2021.03.003

Nunez-Elizalde, A. O., Krumin, M., Reddy, C. B., Montaldo, G., Urban, A., Harris, K. D., & Carandini, M. (2022). Neural correlates of blood flow measured by ultrasound. *Neuron*, *110*(10), 1631–1640.

Ortiz-Rosario, A., & Adeli, H. (2013). Brain-computer interface technologies: from signal to action. *Reviews in the Neurosciences*, *24*(5), 537–552.

Paparrizos, J., & Gravano, L. (2015). k-shape: Efficient and accurate clustering of time series. *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, 1855–1870.

Paulk, A. C., Kfir, Y., Khanna, A. R., Mustroph, M. L., Trautmann, E. M., Soper, D. J., Stavisky, S. D., Welkenhuysen, M., Dutta, B., & Shenoy, K. v. (2022). Large-scale neural recordings with single neuron resolution using Neuropixels probes in human cortex. *Nature Neuroscience*, *25*(2), 252–263.

Pawar, D., & Dhage, S. (2020). Feature Extraction Methods for Electroencephalography based Brain-Computer Interface: A Review. *IAENG International Journal of Computer Science*, *47*(3).

Pawuś, D., & Paszkiel, S. (2022). BCI Wheelchair Control Using Expert System Classifying EEG Signals Based on Power Spectrum Estimation and Nervous Tics Detection. *Applied Sciences*, *12*(20), 10385.

Paxinos, G., & Watson, C. (2006). *The rat brain in stereotaxic coordinates: hard cover edition*. Elsevier.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Ping Shung, K. (2018, March 15). *Accuracy, Precision, Recall or F1?* Towards Data Science.

Poldrack, R. A. (2007). Region of interest analysis for fMRI. *Social Cognitive and Affective Neuroscience*, *2*(1), 67–70.

Rabut, C., Correia, M., Finel, V., Pezet, S., Pernot, M., Deffieux, T., & Tanter, M. (2019). 4D functional ultrasound imaging of whole-brain activity in rodents. *Nature Methods*, *16*(10), 994–997. https://doi.org/10.1038/s41592-019-0572-y

Rathee, D., Raza, H., Roy, S., & Prasad, G. (2021). A magnetoencephalography dataset for motor and cognitive imagery-based brain-computer interface. *Scientific Data*, *8*(1), 1–10.

Roy, A. M. (2022). Adaptive transfer learning-based multiscale feature fused deep convolutional neural network for EEG MI multiclassification in brain–computer interface. *Engineering Applications of Artificial Intelligence*, *116*, 105347.

Schlegel, F., Schroeter, A., & Rudin, M. (2015). The hemodynamic response to somatosensory stimulation in mice depends on the anesthetic used: Implications on analysis of mouse fMRI data. *NeuroImage*, *116*, 40–49. https://doi.org/https://doi.org/10.1016/j.neuroimage.2015.05.013

Sinaga, K. P., & Yang, M.-S. (2020). Unsupervised K-means clustering algorithm. *IEEE Access*, *8*, 80716–80727.

Singh, D., & Singh, S. (2020). Realising transfer learning through convolutional neural network and support vector machine for mental task classification. *Electronics Letters*, *56*(25), 1375–1378.

Sinha, G., Shahi, R., & Shankar, M. (2010). Human Computer Interaction. *Proceedings - 3rd International Conference on Emerging Trends in Engineering and Technology, ICETET 2010*, 1–4. https://doi.org/10.1109/ICETET.2010.85

Soloukey, S., Vincent, A. J. P. E., Satoer, D. D., Mastik, F., Smits, M., Dirven, C. M. F., Strydis, C., van der Steen, A. F. W., Bosch, J. G., & de Zeeuw, C. I. (2020). Using functional ultrasound (fUS) to map brain functionality and tumor vasculature with micrometer-millisecond precision. *Neuro-Oncology*, *22*.

Sorger, B., & Goebel, R. (2020). Real-time fMRI for brain-computer interfacing. *Handbook of Clinical Neurology*, *168*, 289–302.

Sperduti, A., & Starita, A. (1997). Supervised neural networks for the classification of structures. *IEEE Transactions on Neural Networks*, *8*(3), 714–735.

Tan, L. L., & Kuner, R. (2021). Neocortical circuits in pain and pain relief. *Nature Reviews Neuroscience*, *22*(8), 458–471.

Torres, E. P., Torres, E. A., Hernández-Álvarez, M., & Yoo, S. G. (2020). EEG-based BCI emotion recognition: A survey. *Sensors*, *20*(18), 5083.

Urban, A., Dussaux, C., Martel, G., Brunner, C., Mace, E., & Montaldo, G. (2015). Real-time imaging of brain activity in freely moving rats using functional ultrasound. *Nature Methods*, *12*(9), 873–878.

van der Maaten, L., Postma, E., & van den Herik, J. (2009). Dimensionality reduction: a comparative. *J Mach Learn Res*, *10*(66–71), 13.

Vasiljevic, G. A. M., & de Miranda, L. C. (2020). Brain–computer interface games based on consumer-grade EEG Devices: A systematic literature review. *International Journal of Human–Computer Interaction*, *36*(2), 105–142.

Wang, Q., Ding, S.-L., Li, Y., Royall, J., Feng, D., Lesnar, P., Graddis, N., Naeemi, M., Facer, B., & Ho, A. (2020a). The Allen mouse brain common coordinate framework: a 3D reference atlas. *Cell*, *181*(4), 936–953.

Wang, Q., Ding, S.-L., Li, Y., Royall, J., Feng, D., Lesnar, P., Graddis, N., Naeemi, M., Facer, B., & Ho, A. (2020b). The Allen mouse brain common coordinate framework: a 3D reference atlas. *Cell*, *181*(4), 936–953.

Wang, Z., Cao, C., Zhou, Y., Chen, L., Gu, B., Liu, S., Xu, M., He, F., & Ming, D. (2021). Integrating EEG and NIRS improves BCI performance during motor imagery. *2021 10th International IEEE/EMBS Conference on Neural Engineering (NER)*, 511–514.

Williams, D. A. H. (2019). Evaluating BCI for musical expression: Historical approaches, challenges and benefits. *Brain Art: Brain-Computer Interfaces for Artistic Expression*, 145–158.

Winn, H. R. (2022). *Youmans and Winn neurological surgery*. Elsevier Health Sciences.

Wolpaw, J. R., McFarland, D. J., & Vaughan, T. M. (2000). Brain-computer interface research at the Wadsworth Center. *IEEE Transactions on Rehabilitation Engineering*, *8*(2), 222–226.

Xu, Y., Hua, J., Zhang, H., Hu, R., Huang, X., Liu, J., & Guo, F. (2019). Improved transductive support vector machine for a small labelled set in motor imagery-based brain-computer interface. *Computational Intelligence and Neuroscience*, *2019*.

Yin, M., Borton, D. A., Komar, J., Agha, N., Lu, Y., Li, H., Laurens, J., Lang, Y., Li, Q., Bull, C., Larson, L., Rosler, D., Bezard, E., Courtine, G., & Nurmikko, A. v. (2014). Wireless neurosensor for full-spectrum electrophysiology recordings during free behavior. *Neuron*, *84*(6), 1170–1182. https://doi.org/10.1016/j.neuron.2014.11.010

Zhang, H. (2004). The optimality of naive Bayes. *Aa*, *1*(2), 3.

Zhang, Y. (2021). Invasive BCI and noninvasive BCI with VR/AR technology. *International Conference on Artificial Intelligence, Virtual Reality, and Visualization (AIVRV 2021), 12153*, 186–192.

Zhu, X., Li, P., Li, C., Yao, D., Zhang, R., & Xu, P. (2019). Separated channel convolutional neural network to realize the training free motor imagery BCI systems. *Biomedical Signal Processing and Control*, *49*, 396–403.

Zickler, C., di Donna, V., Kaiser, V., Al-Khodairy, A., Kleih, S., Kübler, A., Malavasi, M., Mattia, D., Mongardi, S., & Neuper, C. (2009). BCI applications for people with disabilities: defining user needs and user requirements. *Assistive Technology from Adapted Equipment to Inclusive Environments, AAATE, 25*, 185–189.