Universiteit Utrecht

fUS data analysis and interpretation for stimuli decoding

A thesis presented for the degree of Master of Science
in Human-Computer Interaction Information Science,
Utrecht University

Author: Marina Batlló Rius (7273002)
External Supervisors: Théo Lambert, Alan Urban, Urban Lab, NERF
First Supervisor: Egon van den Broek
Second Supervisor: Ad Feelders

July 2023

Contents

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 everywhere nowadays, the interaction between it and humans plays a vital role. Although this interaction can usually be 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 and 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 and Kaçar, 2021, Glavas et al., 2022, Nijholt et al., 2008], education through or of technology [Gezgez and Kaçar, 2021, Glavas et al., 2022, Hernandez-Cuevas et al., 2020], or user experience evaluation [Cano et al., 2022, Diya et al., 2019].

However, one of the main characteristics differentiating 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-aided communication [Kundu and Ari, 2022], environmental control [Jeyakumar et al., 2022, Leeb et al., 2015], wheelchair locomotion control [Pawuś and Paszkiel, 2022], and even for helping in neuro-rehabilitation [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 and Adeli, 2013]. For each of these elements, different technologies and methods are and have been developing, as each of them presents various advantages and drawbacks.

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 leading technologies used in BCIs are electroencephalogram (EEG), magnetoencephalography (MEG), functional Magnetic Resonance Imaging (fMRI), and electrocorticography (ECoG) [Min et al., 2010, Ortiz-Rosario and Adeli, 2013].

EEG is a non-invasive method that uses metal electrodes on the scalp to measure the electric potentials resulting from neuronal activity. It is the most widely used technology for BCI applications.

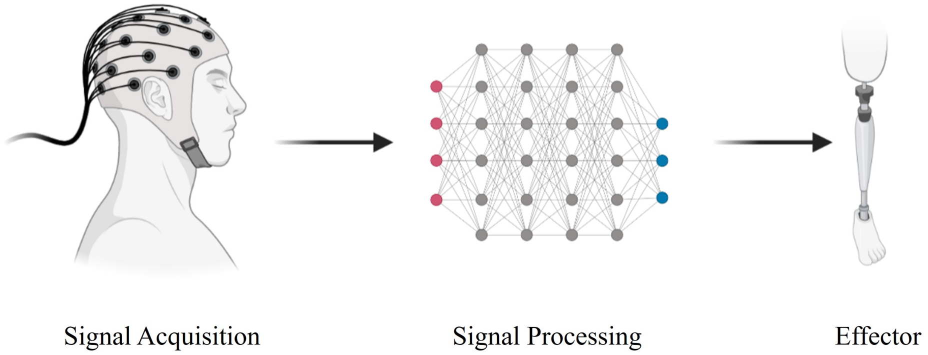


Figure 1: Basic elements of BCI systems. Created in BioRender.

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 significant area of the brain, giving low accuracy in spatial information [Casson et al., 2018].

MEG is similar to EEG but measures magnetic fields instead of electric potentials. 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, making them irrelevant 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 it is possible to obtain brain activity information by measuring these changes. 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 costly and require special facilities and personnel, making it difficult to make them scalable. Apart, subjects must remain very still, resulting in an approach that is difficult to directly implement 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].

Method	**Measurement**	**Temporal resolution**	**Spatial resolution**	**Limitations**	**Examples**
EEG	Electric potentials of cortical activity	Very high (1 ms)	Coarse (1 cm)	Spatial resolution and superficial	[Abiri et al., 2019, Torres et al., 2020]
MEG	Magnetic fields of cortical activity	Very high (1 ms)	Coarse (1 - 2 cm)	Spatial resolution and superficial	[Dash et al., 2020, Rathee et al., 2021]
fMRI	BOLD changes in susceptibility-weighted MR signal	Low (1 – 2 s); limited by hemodynamic delays	Very high (3 - 6 mm).	Temporal resolution, expensive, needs still subjects	[Du et al., 2022, Sorger and 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, 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	Spatial resolution	[Khalaf et al., 2019a, Khalaf et al., 2019b]

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

2.2 Task definition and signal processing

2.2.1 Task definition and understanding

The first step in making BCI work is not only to use the right technology, but also to have a deep understanding of the task to be performed. This is necessary to ensure the best possible performance of the devices used. Firstly, some devices do not cover the whole brain, so positioning is key to their acquisition. Once the task is known and understood, it is possible to define where the device should be positioned to obtain the most relevant information. In addition, in order to better control the devices, it is also essential to understand where and what patterns of activity are taking place when the tasks are performed, in order to relate them to specific actions.

To better describe this brain activity, cognitive models have been introduced. This understanding of the underlying processes 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 studied by many psychologists before, who began to develop cognitive models to explain these processes and to try to reproduce and predict the human mind.

Today, these models have evolved into computational cognitive models, which deal with the simulation of human problem solving and mental processes in a computerised model. These models are widely used in HCI research because they can create more efficient and accurate user models [Dupret and Piwowarski, 2008], and specifically in BCI, they can help predict user intentions. Furthermore, cognitive models can be used not only to understand the task at hand, but also to improve the BCI training of the user and to decode the information obtained from brain activity measurements. However, current computational cognitive models are still limited by technology and by the current limited understanding of the human brain, which means that current models are usually unable to represent the full complexity of a process. A better understanding and deeper research into this issue is fundamental to understanding how and which brain regions are important for specific tasks in BCI.

Once the task has been defined and prior knowledge of how the task is processed in the brain has been acquired, the instruments and post-processing can be carried out with more information, leading to better results.

2.2.2 Methods for signal processing

There is a wide range of methods for processing and translating the signals. In general, signal processing involves feature extraction and feature classification or translation. Feature extraction techniques are usually divided into time-frequency and spatio-temporal techniques. Within these groups, many different approaches can be used and new ones are emerging frequently [Pawar and Dhage, 2020]. Depending on the imaging technique used and the type of application, one method or another may be more appropriate. Some common examples of signal processing techniques are linear discriminant analysis [Dodia et al., 2019, Fu et al., 2019, Lo et al., 2022], support vector machine algorithms [Maher et al., 2023, Singh and Singh, 2020, Xu et al., 2019] or CNNs [Fahimi et al., 2019, Roy, 2022, Zhu et al., 2019].

Once the features have been extracted, it is necessary to classify the signals in order to predict the user's intentions. Again, there are many ways to perform these classifications, the most widely used being machine learning or deep learning techniques [Aggarwal and Chugh, 2019]. Although many different methods have been used, each individual technique, or even the use of different combinations of them, still has several problems, including: not being suitable for the data analysed, difficult interpretation, problems with overlapping data, achieving a high enough accuracy of classification, overfitting the data, needing ground truth to evaluate the method, among many others. Each of these tools also has individual issues that need to be considered when using them, ranging from computational requirements to the need for predefined parameters.

2.3 Effector devices

Once the signal has been processed and translated, the command extracted from it must be sent to the effector device. Effector devices can take a variety of forms depending on their purpose. Many different effector devices that use BCI to receive information have been studied, particularly from an HCI perspective where more effective devices are being researched.

Effector devices are not only computers directly controlled by BCI systems [Kumar and Sharma, 2012], but also include a wide range of technologies. In Vasiljevic et al. [Vasiljevic and de Miranda, 2020], a review of different games using consumer devices is conducted. The devices used as effectors in this review include game consoles, PCs, mobile devices and websites. The review also includes applications of BCI information in technologies such as adaptive systems involving cognitive state recognition [Cutrell and Tan, 2008] or emotion recognition [Li et al., 2009]. BCI signals have also been used to evaluate user experience in interactive systems [Cano et al., 2022], where the effector device is not directly applied, but by evaluating the brain signals when using a particular application, to control prosthetics [Gannouni et al., 2020], to enhance the VR/AR experience where the effector device is the VR/AR glasses [Zhang, 2021], or even for music composition and performance where the effector device includes a loudspeaker [Williams, 2019].

As reviewed, many different technologies and methods can be used as BCIs. However, existing neuroimaging techniques are still limited and present many problems for more accurate and functional BCIs, such as low temporal resolution, low spatial resolution, invasiveness, low brain coverage, long training periods, or failure to function between subjects [Mridha et al., 2021]. Therefore, there is a need to develop new technologies that can provide brain imaging with high temporal and spatial resolution. There is also a need to develop a validated, reliable and robust way of processing the information extracted from these new techniques. One promising technology currently under investigation is functional ultrasound imaging, where the Urban Lab at NERF is conducting significant research to improve brain imaging and decoding.

3 Functional Ultrasound Imaging

3.1 Principle of Functional Ultrasound Imaging

Functional Ultrasound Imaging (fUS) is a hemodynamic neuroimaging technique that measures blood volume using power Doppler imaging. Unlike the beam scanning commonly used in ultrasound techniques, fUS uses planar illumination, which increases the number of samples per pixel and therefore sensitivity [Émilie Mace 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), which is also determined by the computer hardware optimisation software used to process the raw data. The spatial resolution depends directly on the pitch of the ultrasound transducer used, which also determines the depth of field (typically in the range of 100-300μm^3 and up to several cm in depth). Recording repeated images over time allows us to track relative changes in blood volume per voxel, which in turn reflects changes in neuronal activity [Nunez-Elizalde et al., 2022]. To avoid interference of the skull with the ultrasound, all data is obtained through a cranial window, making fUS a minimally invasive technique.

There are many features that can be considered when comparing brain imaging techniques. The most important are spatio-temporal resolution, brain coverage and restraint method. Although other techniques can surpass the spatio-temporal resolution of fUS, it is important to note that this technology offers full brain coverage, which differentiates it from other techniques. To better compare fUS with other technologies, we need to look at the whole picture (see **Figure 2**).

As can be seen, fUS covers a wide range of spatio-temporal resolution comparable to many other techniques, achieving temporal resolutions that, for example, fMRI cannot achieve, or spatial resolutions that neither fMRI nor EEG can achieve. Furthermore, although the temporal resolution of fUS is lower than that of EEG or neuropixel, the latter measure electrical signals that are much faster than those measured by fUS, which are haemodynamic signals. Apart from these advantages, fUS has also become a tool applicable to behaving animals [Urban et al., 2015] and has even been scaled up to humans [Demene et al., 2016, Soloukey et al., 2020].

3.2 Linear fUS vs volumetric fUS

When referring to fUS, we can now distinguish between two different techniques: linear and volumetric fUS. Linear fUS (lfUS) provides 2D images down to a depth of several centimetres, with a spatial resolution of ∼100x300x100μm^3 and a temporal resolution of up to 10Hz. The higher quality images of brain haemodynamics obtained with linear fUS have already proven useful for single-trial decoding of the timing and targets 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 already has the potential to be used in broader BCI applications.

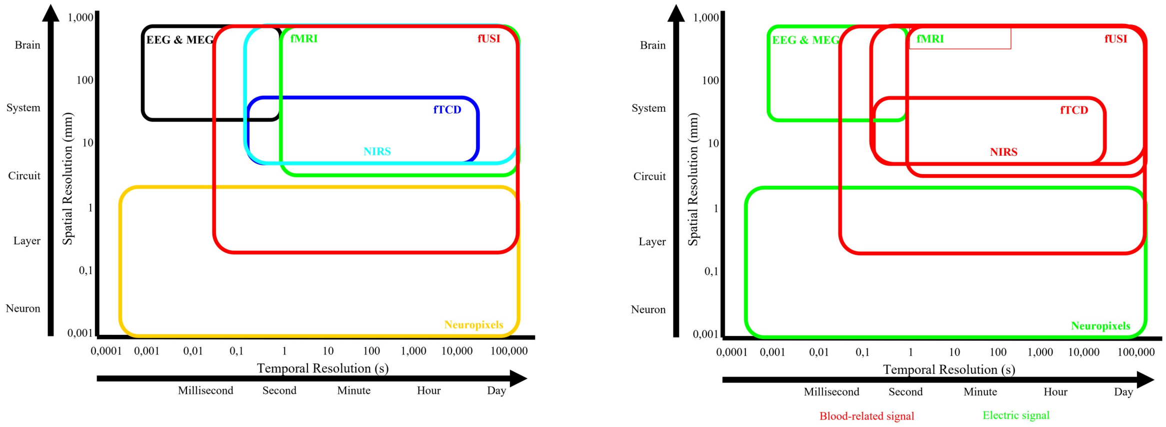


Figure 2: Comparison of brain imaging techniques in terms of spatiotemporal resolution, applicability conditions and type of signal.

Although linear fUS has proven useful for these applications, it is limited to cross-sectional imaging, which hinders direct visualisation of the whole brain. In order to perform whole-brain imaging with linear fUS, it is necessary to move the probe to multiple positions [Émilie Mace et al., 2018]. The problem with this is the need to repeatedly present the stimulus in different positions, which requires more acquisition time and can lead to habituation of the animal [eun Kang Miller et al., 2018]. In addition, the need to move the probe results in the loss of a significant amount of brain activity due to its distribution throughout the brain [Montaldo et al., 2022].

To overcome these limitations, a 2D array transducer was used by Rabut et al. [Rabut et al., 2019] to obtain 3D images directly in anaesthetised rats. However, this first attempt had several limitations, such as the decrease in frame rate, the fact that they used acute conditions, and the anaesthetics, which have been shown to alter the haemodynamic response [Schlegel et al., 2015]. For this reason, further investigations to improve volumetric fUS (vfUS) were carried out by Brunner et al. [Brunner et al., 2020]. In their work, they developed a vfUS system suitable for real-time 3D imaging of awake subjects with high spatial ($\sim$220x280x175μm^3) and temporal (up to 6 Hz) resolution, allowing scanning of almost the entire rodent brain.

3.3 General pipeline for data analysis

With the images obtained by fUS, known as power Doppler images (Figure 3 B), the next step is to register the data. Registration relies on a geometric transformation to fit the data to a reference map of the brain, such as the Paxinos rat brain atlas or the Allen CFF (**Figure 3 C**) [Paxinos and Watson, 2006, Wang et al., 2020]. This procedure facilitates the comparison of different animals and provides a basis for region-averaging procedures. The different trials in a fUS 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 atlas used, maximising the signal-to-noise ratio and providing a temporal map for interpretation (Figure 3 E). It is then possible to proceed with signal processing and interpretation. In this work, we will explore alternatives to facilitate interpretability and visualisation in this step.

The problem with obtaining whole-brain images, particularly when using vfUS, is that the amount of information obtained is very large, on the order of 150 full-brain images per experimental trial at a resolution of $\sim$150μm^3. Although region averaging can provide useful results, it is important to note

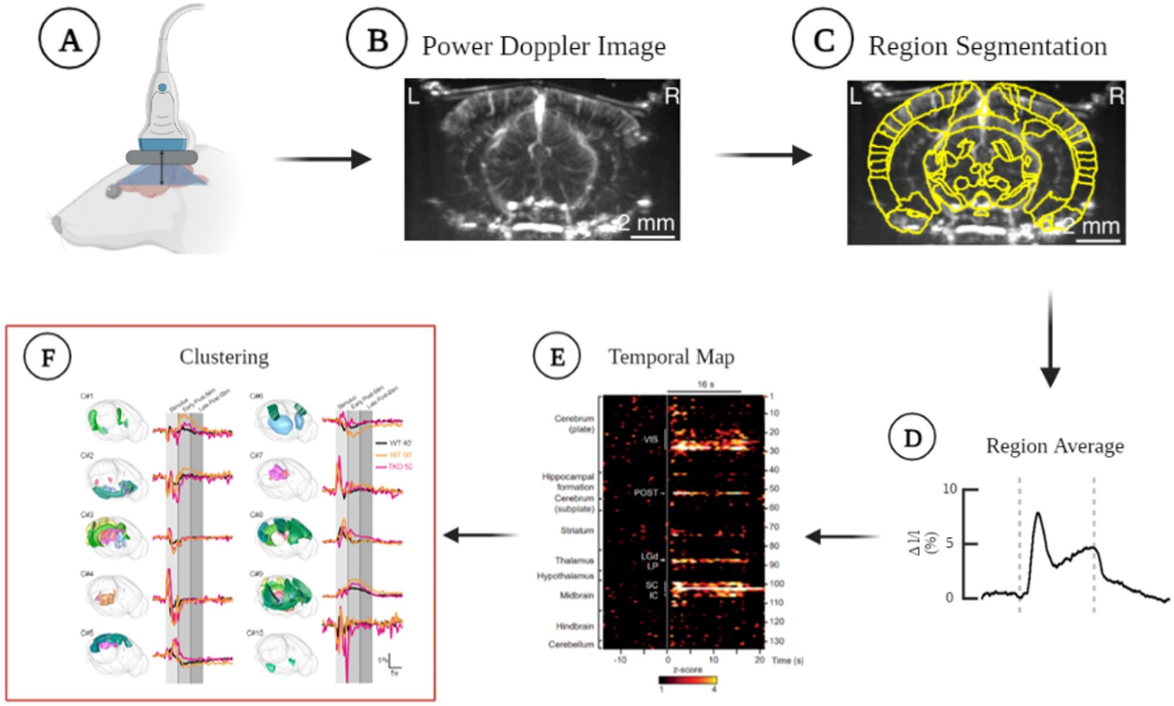


Figure 3: fUS process to interpret brain activity. (**A**) Record fUS images for a specific experimental setting. (**B**) Obtain the multiple power-doppler images recorded through the probe. (**C**) Register each of the images on the decided Atlas to facilitate intra-animal comparison and further steps. (**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.

that it can lead to misinterpretation of data [Poldrack, 2007] or loss of important cues [Constable, 2006]. However, this technology opens the door to new paradigms that combine traditional analysis methods with whole-brain imaging for brain decoding purposes.

4 Animal models in BCI research

Although fUS is a promising technology for BCI, it is still in its developmental stage, with studies being conducted primarily in animal models. Although the ultimate goal of BCI is to be applied to humans to open up new ways of communicating between them and different technologies, some studies are still not possible directly in humans. When doing basic research to improve the results of any technology, especially when exploring new areas, studies with humans become very limited. This is where the importance of animal models comes in, as various studies can be carried out in vivo for future human application, taking into account ethical issues. There are many advantages to using animals in research. Firstly, it is possible to use invasive techniques in animals that would be ethically questionable in humans. In addition, more advanced recording methods are available for smaller animal models such as rodents. There is also the possibility and affordability of using larger numbers of subjects, as animal testing is more readily available than human testing. In addition to the use of rodent animal models, for example, there is an increase in possibilities and experimental settings. These include genetic tools to create and study different defects, physical tools such as circuit manipulation, or more invasive histology.

It is well known that mammalian species have conserved many characteristics and advanced brain functions. As a result, animal models have already contributed to the translation of the first BCI technologies to humans. The use of animals allows us to understand more specific and potentially relevant issues in decoding brain activity and to apply this knowledge directly to humans. Although fUS technology has been used in larger mammals, whole brain imaging is not possible in larger brains. Therefore, the use of rodents opens up new opportunities to study and better understand how processing takes place throughout the brain.

5 Proposal

In this thesis I propose a pipeline to facilitate the implementation of vfUS data in BCI systems. As mentioned above, vfUS data allows us to obtain whole brain images. In this case, it is possible to obtain images with a frame rate of 6Hz and a resolution of $\sim 200 \mu m^3$. This allows us to combine and compare information from different regions to understand how specific processes in the brain work, which can have a major impact on understanding brain processes for BCI. The main aims of this research are twofold.

First, I pretend to build a pipeline to visualise high-dimensional data in a way that allows insight into a given stimulus. To achieve this, an initial stimulus, denoted A, is administered to rodents and the resulting circuit activity is visualised by region averaging. Subsequently, a second data set, denoted B, will be acquired and analysed in relation to the first data set. The aim of this analysis is to compare the brain's response to both stimuli, allowing researchers to identify regions of interest that show notable differences between the activity evoked by stimuli A and B. These differences may manifest as increased or decreased activity within specific regions, or changes in the type of activation. Various clustering techniques, such as k-means or hierarchical clustering, are used to classify atlas-based brain regions according to different activity patterns. The different cluster changes in each brain region between stimuli A and B will then be visualised. This process aims to enable the tracking of specific brain regions and activity changes between different and specific stimuli.

The second objective is to propose a region selection method to determine ROIs for further study. This will allow the direct use of fUS data to classify two related stimuli. To evaluate the effectiveness of the proposed region selection method, its classification performance will be compared with alternative approaches. The aim of this study is not to introduce a novel classification methodology, but rather to explore how a robust analysis pipeline can enhance our understanding of brain processes and influence the selection of relevant regions for stimulus classification using pre-existing, explicable models. To validate the usefulness of the proposed pipeline in identifying relevant regions, we will compare the performance of cluster-based selected brain regions with those selected based on prior knowledge or data-driven methods. In addition, the study will investigate the significance of whole-brain imaging data compared to cortex-only data for stimulus classification. This comparative analysis will provide insight into the performance of simpler imaging techniques relative to more comprehensive data acquisition methods. The whole study proposal can be seen in **Figure 4**.

For this research, the stimuli used to explore new ways of interpreting and classifying fUS data will consist of two different thermal stimuli applied directly to rodents. The first stimulus will be a 40^{o}C thermal stimulus, which is considered to be non-painful. The corresponding stimulus that we will be trying to decode will be a 50^{o}C stimulus, which is considered painful when applied to the subjects. Painful stimulation is known to cause changes, not only in brain activity, but also in the animal's behaviour. These changes and movements will mimic real life conditions where brain waves are not perfect and many different disturbances can be found.

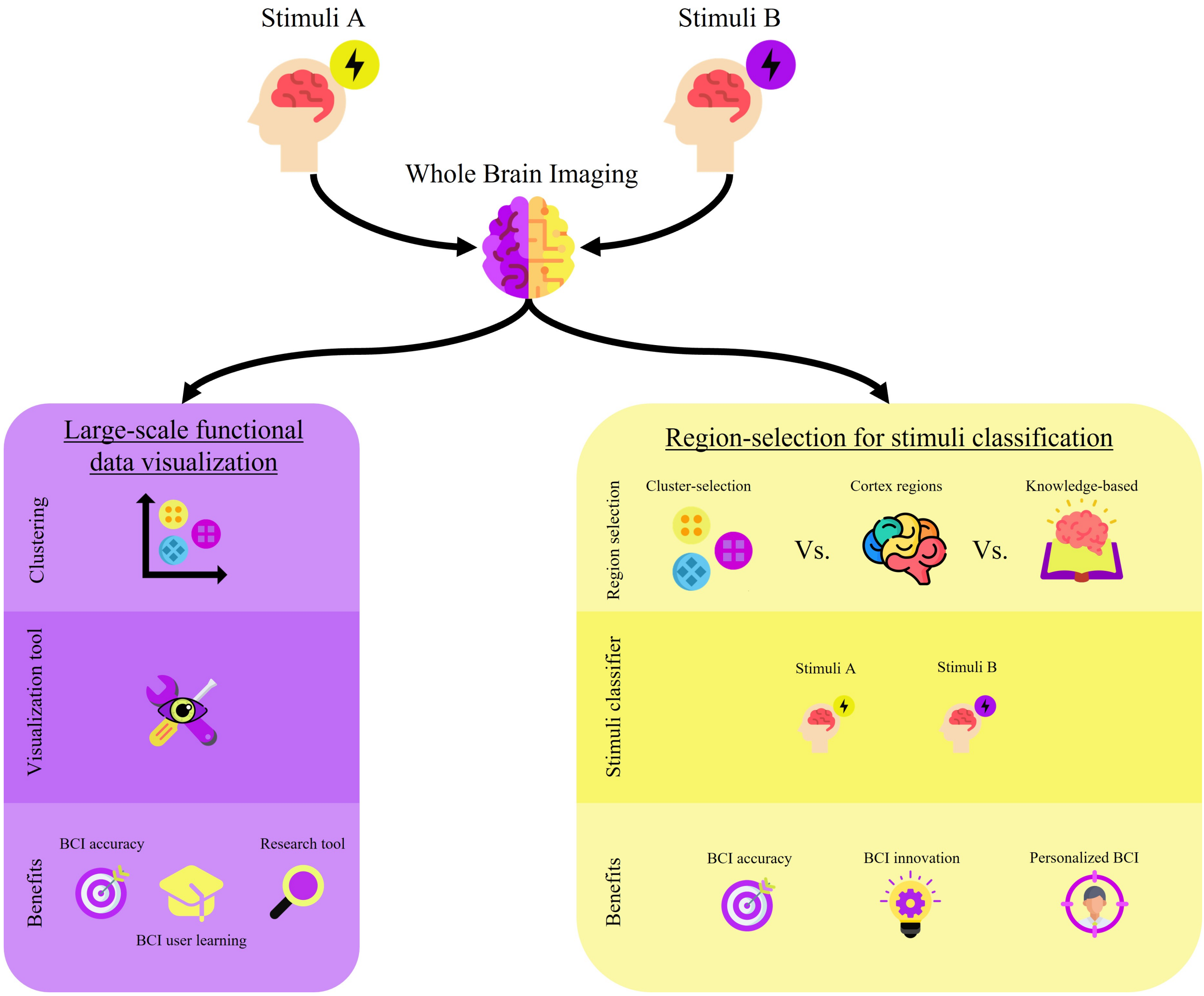


Figure 4: Research proposal including process, obtained results and potential benefits of every part.

Research Methodology

For this study, the first step consists of using region-averaged data (see **section 1. Data set**) to cluster the different activity patterns and visualize them in a comprehensive way (see section **2. Large-scale functional data visualization**). Then, the process will be used to select ROIs for stimuli classification and compare its performance to other region-selection methods (see section **3. Region-selection for stimuli classification**). The process and methodology will be further explained. An overview of the steps can be seen in **Figure 5**.

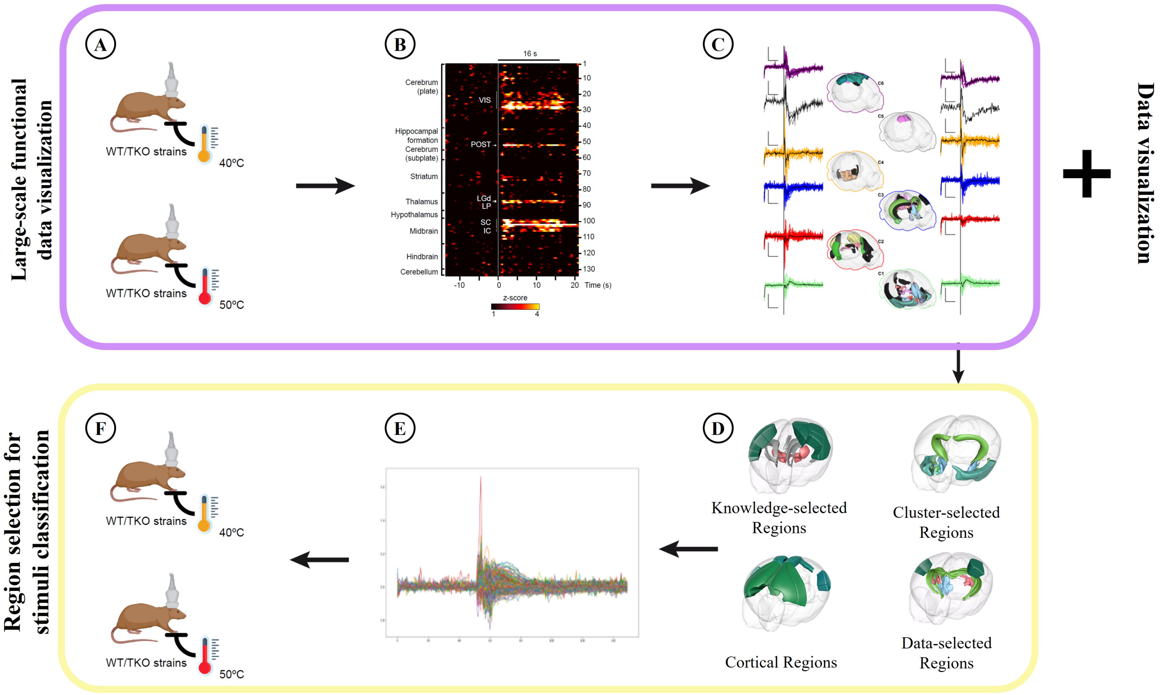


Figure 5: Process of this Master Thesis. (**A**) Get data from two different stimuli. (**B**) Region-averaging and single-trial averaging will be performed in the data after registration to obtain the temporal map. (**C**) The region-averaged data will be clustered using the best method in order to find activity patterns. (**D**) From the clustering, an analysis will be conducted to find the regions that are of most interest when differentiating both conditions. Together region selection methods will be used to compare results. (**E**) Single trial will be used to compare how different region selection methods perform when classifying the stimulus (**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

The first step to obtain or data set will be to register the data obtained through vfUS, following the typical fUS pipeline, which includes fitting the data in a brain map. In this case, we will use the Allen Mouse Brain Reference Atlas [Wang et al., 2020]. 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.

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 30^{o}C stimuli, three times a 40^{o}C stimuli, and one time a 50^{o}C stimuli (which induces pain in the subject), with randomized appearance per round. The 30^{o}C stimuli is just used as control condition. On the other hand, the 50^{o}C 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 6A**).

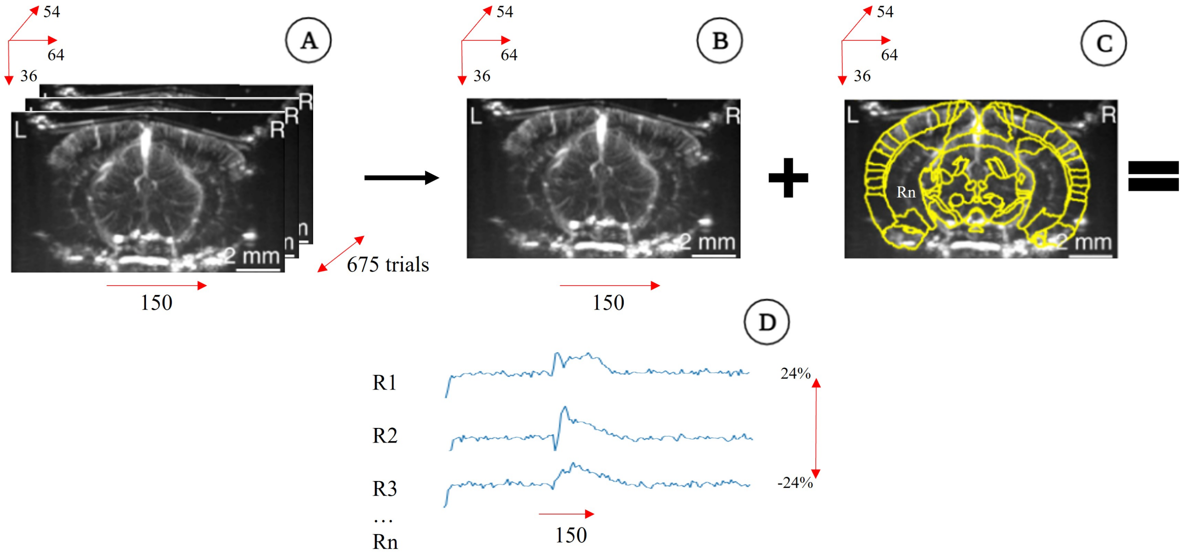


Figure 6: Dataset used in the work. The numbers represent the number of frames. (**A**) Original dataset, with all the trials. (**B**) Averaged dataset. (**C**) Registration in the Allen brain Atlas. (**D**) Region Averaged signals.

In each session, volumetric images of the brain were acquired using a fUS probe composed of a 2D matrix of $\sim$1,024 piezoelectric elements of $300\mu m^3$ at 15MHz. The probe covered the full volume of the brain ($\sim$1cm^3 in rodents). A direct computation of the fUS images was performed, resulting in images obtained at a frame rate of 2Hz and with a resolution of $\sim$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**). The used regions in this work can be seen in **Annex A**.

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 30^{o}C stimuli, three for the 40^{o}C, that were then averaged, and one for the 50^{o}C. 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 $\sim$ -24 and $\sim$24 (ΔI/I) (see **Figure 7**), where ΔI/I represents the normalization given a baseline of the doppler signal obtained.

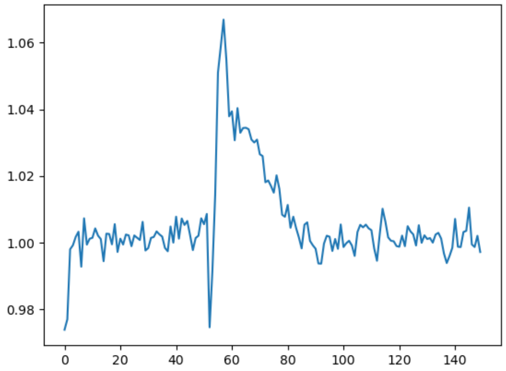


Figure 7: Example signal of neural activity extracted from a fUS recording.

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 40ºC and 50ºC stimuli. The data will be used as a standard case of decoding an evoked activity by looking for the difference between the two temperatures. The 50ºC pain evoked data set will also be used as a replication of real-world applications where there are no perfect signals, as the behavioural changes in the animal are expected to be inferred from the signals. Studying this data will provide insight into the contribution of perturbations to regular activity and help identify which regions can be used for fine decoding.

2 Large scale functional data visualization

Due to the high dimensionality of the data obtained (∼120,000 voxels), the analysis of fUS signals is not an easy process. Not only the volume, but also the fact that they are time series can make their comparison and interpretation particularly difficult. In this paper, we propose a cluster-based approach based on grouping the data according to their activity pattern to facilitate their interpretation. This method makes it possible to find differences in 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. **Evaluation of the results:** different measures are used to quantify how relevant is the grouping of the data.

The choice of methodology can have a major impact on the results of clustering. For this reason, different approaches are reviewed in this paper. However, as the aim is to facilitate the implementation of fUS data into BCIs, we have chosen well-established, intuitive and computationally efficient methods to make the pipeline more generalisable and easily applicable.

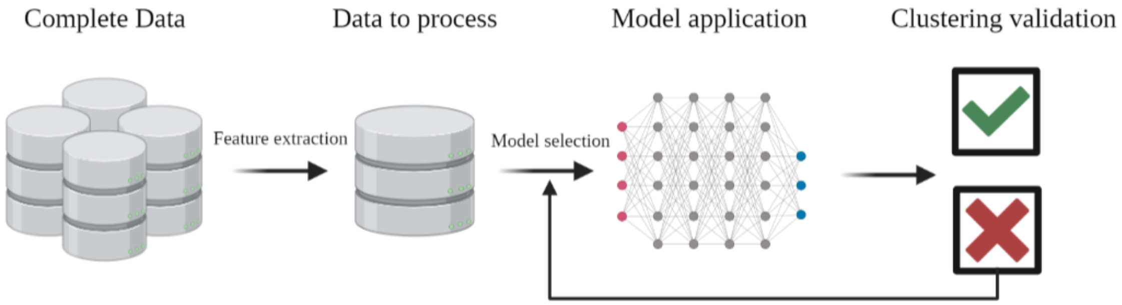


Figure 8: Generalized clustering process for a dataset. The model selection needs to be validated and changed if results are not relevant. Created in BioRender.

2.1 Feature selection

We call each input variable that is given to the clustering model a feature. Although it is possible to use the raw data, this can sometimes be problematic as the model may use unimportant features to group the data and learn from noise. Therefore, the important parameters of the data must be selected. A common way of doing this in time series analysis is to rely on certain signal parameters, such as amplitude or time-to-peak [Krishnan and Athavale, 2018]. However, these types of parameters are difficult to determine in fUS signals because of the variability/noise in the signal (see **Figure 9**).

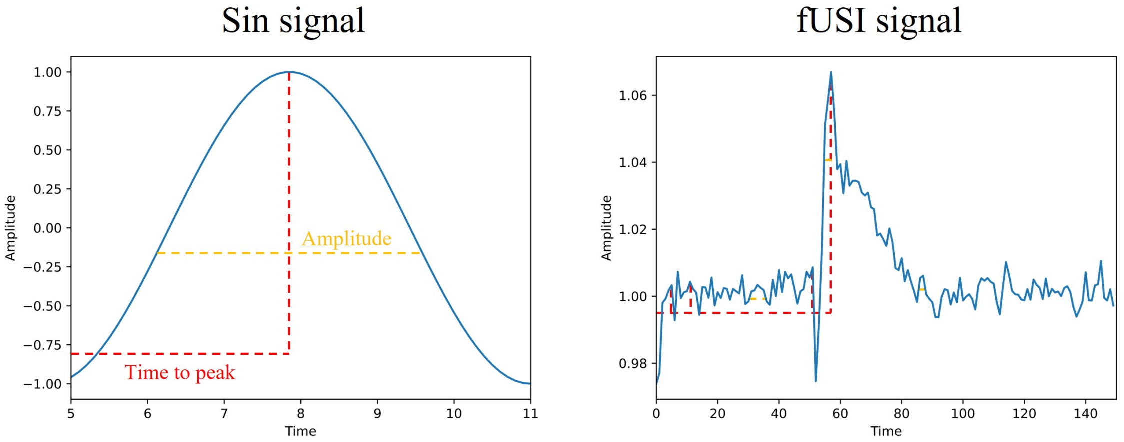


Figure 9: Comparison of measuring signal features in a sin signal vs a fUS signal. It can be observed that in a fUS signal is very complex to choose which or where to measure possible features.

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 common practice to use dimensionality reduction techniques to improve the computational time and the amount of data required to sample the input space properly. It is well known that increasing the dimensionality of the data leads to an exponential increase in the amount of data needed to sample the input space properly [Shashmi, 2021]. In addition, reducing the dimensionality of our data can help eliminate the irrelevant features, resulting in better outputs [Maaten et al., 2009]. It should be noted that dimensionality reduction is not useful in all cases of data analysis. In this work, we will try two different dimensionality reduction techniques on the raw data to compare their results. These two methods were chosen because they are widely used methods that have shown good performance and because their specific characteristics are relevant for our data.

Principal Component Analysis (PCA)

PCA is a dimensionality reduction technique that transforms the data into a set of orthogonal components that maximise the variance explained in the data set. PCA applies an orthogonal transformation to the data, which consists of a linear transformation represented by orthogonal matrices composed of orthogonal vectors of unit norm. This results in the projection of our original features onto the space defined by the principal components [Kurita, 2019]. We will use PCA because it is a well-established and efficient technique for dimensionality reduction and is able to detect multicollinearity, thus avoiding the use of data that does not provide additional information. Furthermore, PCA focuses on retaining the maximum amount of variance with the minimum number of components, making it relevant to our specific needs.

When using PCA, we need to decide how many principal components to use for further investigation. This is usually done by observing how much variability within the data is represented by each component. Plotting this information results in a curve with the number of components on the X-axis and the variance explained on the Y-axis. Generally, the inflection point of this curve indicates the appropriate number of components to choose.

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. We decided to include ICA into the analysis as it is specially fitted for non-gaussian data and can help disentangle underlying variation factors in the data.

2.2 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. [Émilie Mace et al., 2018] a K-means algorithm was successfully used to identify activity patterns, demonstrating that FUS data can be analysed without using too complicated methods. Therefore, 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 and 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].

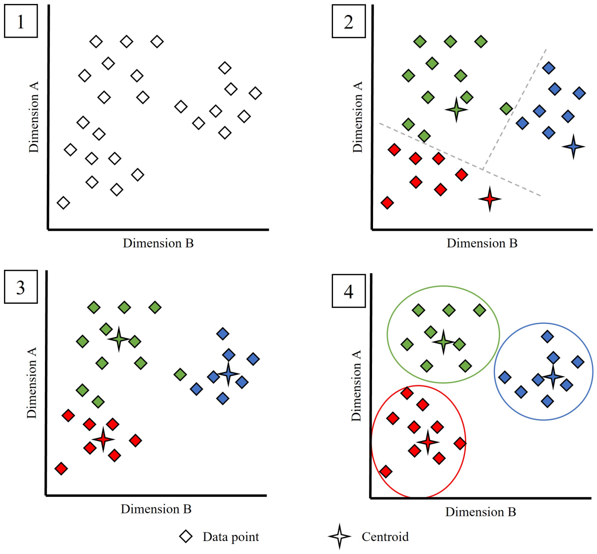


Figure 10: K-means process. **(1)** The data to be processed in the n-dimensional space. **(2)** The centroids are randomly decided, and each datapoint is assigned to the closest centroid. **(3)** The centroid is corrected so it is in the centre of the datapoints assigned. **(4)** The datapoints are 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 use several methods that will be further explained in 2.3.

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 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 extracting more information from the analysis. For our specific data, this can help us understand how the different activity patterns relate to each other.

2.3 K-shape Clustering

K-shape clustering is a clustering algorithm based on the K-means algorithm. However, K-shape is specifically designed to cluster time series based on their shape. It does this by using as its distance measure a normalised measure from the cross-correlation method, which 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 with K-means and agglomerative clustering, K-shape requires the number of clusters to be specified before grouping [Paparrizos and Gravano, 2015]. We

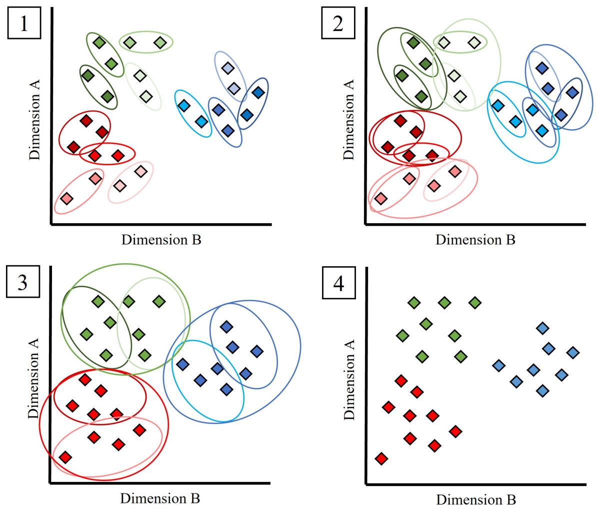


Figure 11: Hierarchical clustering process. **(1)** The nearest points on the data are clustered together. **(2-3)** The nearest clusters are merged in an iterative process until meeting the number of clusters specified. **(4)** Final result.

decided to use this method because is especially suited to cluster depending on the time series shape rather than deciding by the data distribution.

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.

3.1 Feature Selection evaluation

Before determining the most appropriate combination, it is necessary to examine the performance of the dimensionality reduction techniques. To do this, 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 they are distributed in space and whether the partitioning of the components makes sense [Hastie et al., 2009].

It is important to note that if the minimum number of components that appear to explain a high level of variance does not have a clear partition or does not appear to be significant, it is not useful to increase these components.

3.2 Clustering Algorithm evaluation

Once the data to be input has been evaluated and selected, the different clustering algorithms can be applied to it to evaluate the output. Various metrics can be used to evaluate the quality of a clustering by examining the similarity within a cluster and with the other clusters. Some of these metrics can and will also be used to determine the optimal number of clusters within a clustering algorithm. Since in this case we are using data without known labels, also known as ground truth, the evaluation must be done using the model itself.

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:

$$s = \frac{b-a}{\max(a,b)} \tag{1}$$

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].

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:

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

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].

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:

$$R_{ij} = \frac{s_i + s_j}{d_{ij}} \tag{3}$$

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:

$$DB = \frac{1}{k} \sum_{i=1}^{k} \max(R_{ij}) \text{ (with } i \neq j\text{)} \tag{4}$$

With k being the number of clusters [Pedregosa et al., 2011].

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, making it appropriate for high-complexity datasets. 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:

$$s_c = \frac{\text{cov}(R(X), R(Y))}{\sigma_{R(X)}\sigma_{R(Y)}} \tag{5}$$

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].

3.2.5 Euclidean Distance

The Euclidean distance measures the difference between two signals point by point. It requires the signals to be of equal length, as it is the vector norm between two time-series. Other distance measures can also be used in this type of data. Some of them, such as Dynamic time wrap, are specially used to correct time shifts, which we want to avoid, as time shifts are relevant in our specific case.

In this case, and because of the complexity of the data, there is no numerical process to address the quality of a distance measure. Therefore, we decided to use Euclidean distance because of its ease of implementation and its direct availability in Python packages. However, as with any distance metric, its results should be treated with caution due to potential limitations. The distance is defined by:

$$d = 2\sqrt{a^2 + b^2} \tag{6}$$

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. In case none of the techniques result satisfactory, other possibilities will be further studied.

3.3 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 the 40ºC and 50ºC conditions are clustered together to ensure that there is a common basis for interpreting the data. This means that each cluster will contain a set of regions from both the 40ºC response and the 50ºC response, which will later be split for visualisation purposes. If we clustered the conditions separately, we would get different patterns for each, which would be very difficult to compare. The aim of this section is to find those activity patterns that are exclusive or stronger to the 50ºC condition, as they will be composed of those regions that encode the pain response. The analysis will be based on observation and quantification through similarity measures between the time series. We will use different visualisation methods for this, which will be chosen at a later stage depending on the grouping and complexity of the results. In order to quantify how similar or dissimilar the different regions are, we will use different parameters to characterise specific clusters. Although we have mentioned that these parameters are difficult to apply to fUS signals, by using the average of the signals included in a cluster, the results may remain significant. The parameters have been selected according to relevant features for fUS data found in [Lambert et al., 2022]. The features that we will include are:

- Time to peak: time point corresponding to the first inflexion point after the 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.
- Timepoint at half maximum: the point when the signal exceeds half the maximum value after the stimulus has begun.

4 Region selection for stimuli classification

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.

4.1 Region selection methods

Three different methods will be used to compare and decide which regions perform better to classify the different stimuli.

4.1.1 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.

4.1.2 Cluster-based 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.

4.1.3 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.

4.1.4 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 pain is processed in the brain is not completely understood, a common biological pathway related to it is the Somatosensory pathway [Tan and Kuner, 2021]. Therefore, we will use regions that are a fundamental part of this pathway.

4.2 Classification algorithms

To approach the classification, we will use supervised learning, which means that in this case the input data will have a label: painful or not painful. However, due to the high dimensionality of our data, using the raw signals for training can be problematic, as overfitting can occur and the models would not perform well. In order to study this, we will select a set of parameters from the signals to be used as features for the input to the classification algorithm, which will be selected after the clustering, where we will see which features better distinguish the two conditions. Then we will compare the performance of the raw signals with the extracted features. We will also use the individual trials of each mouse for the classification, as we cannot use the average, as this would leave us with a single input.

Many different algorithms can be used for binary classification problems. These algorithms are first trained on the labelled data to predict the label of new data not used for training. New algorithms are being developed all the time. However, in this work we are not looking for the highest possible accuracy, but rather for a better understanding of brain signals and how stimulus processing works. Therefore, we will not explore complex algorithms or approaches, but will use the most common explicable models to better understand the results. The algorithms we will explore in this thesis are the following.

4.2.1 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].

4.2.2 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 $w \in R^p$ and a label vector $y \in \{1, -1\}^n$ the algorithm searches for $w \in R^p$ and b $\in R^p$ where the prediction given by $\text{sign}(w^T(x) + b)$ is correct for the largest number of samples. The optimization problem that SV for classification solve can be described as

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

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

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

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:

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

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

4.2.3 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:

$$E = \sum_k (p_k(1 - p_k))$$

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 [Zhang, 2004], ensemble methods [Breiman, 1996], or supervised neural networks [Sperduti and Starita, 1997].

4.3 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 this, we need to split the data into training data, validation data and test data. This must be done before any data is input to ensure that the splitting does not affect the model. The training data will be used to train the model, while the validation data will be used to tune the hyperparameters. Test data can then be used to evaluate the classification. Of our data, 70% is used for training (80%) and validation (20%). To avoid bias in the model and to ensure that the results do not depend on a simple partition, we will use 5-fold cross-validation, which means that different groups will be randomly created from this data stack 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 on data that has 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). Then, the different region selection methods will be used to get different groups of regions from the single-trial dataset. This will leave us with various datasets composed of a limited number of features and characteristics of the signals corresponding to the selected regions of interest. For each dataset, we will extract the features chosen in **2.4. Region-wise analysis** Both the raw data and the features will be used separately to find the most optimal way for good classification.

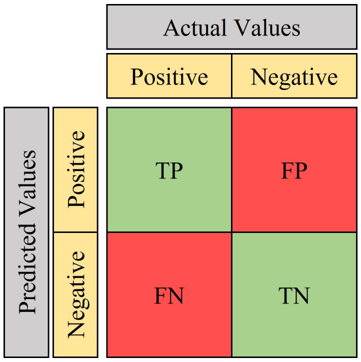
4.4 Classification evaluation

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.

4.4.1 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**.



		Actual Values	
		Positive	Negative
Predicted Values	Positive	TP	FP
	Negative	FN	TN

Figure 12: Classification matrix to evaluate Classification performance

Using these values, it is possible to compute the accuracy of the model, which is the most widely measure to evaluate the performance of a model. With this measure, we will be able to compare the different models and determine how good are different region selection methods to help classify different stimuli.

The accuracy of a model is determined through the following equation:

$$\text{accuracy} = \frac{\text{True positives} + \text{True negatives}}{\text{Number of predictions}}$$

After obtaining the accuracy values for the different region-selection methods we will be able to define if there are significant different through statistical tests such as paired t-test. With this, it will be possible to conclude which region selection method performs better.

Results

5 Large-scale functional data visualization

5.1 Feature Selection

In this section, we wanted to select wether to reduce the dimensionality of our data or use the raw data. Both PCA and ICA were used to approach this. 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 threshold was set at 70% in order to retain significant variability while maximising our dimensionality reduction, leading to better interpretability of the data. The outcomes can be seen on **Figure 13**.

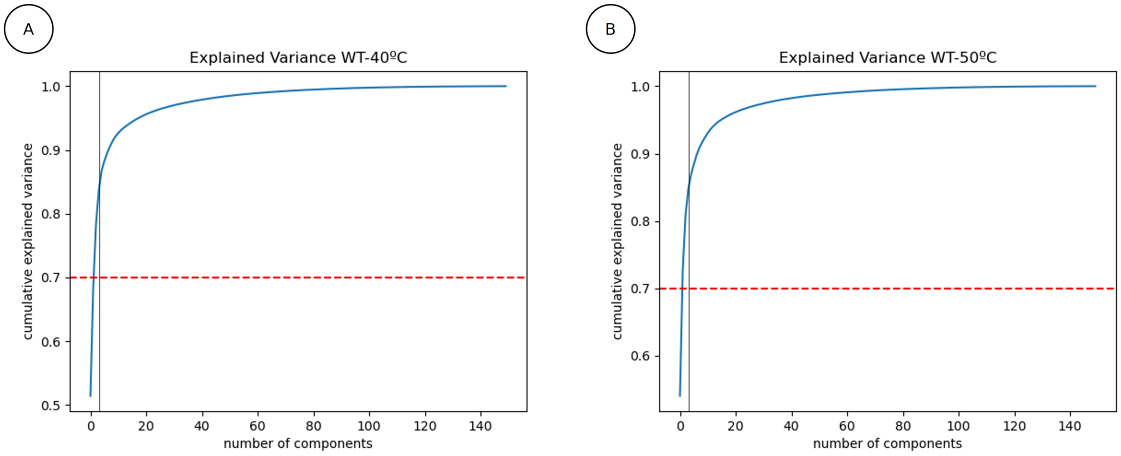


Figure 13: Explained variances for all the experimental conditions. **A)**WT-40ºC; **B)**WT-50ºC.

From these plots we could conclude that a minimum of three principal components is sufficient to achieve the desired threshold of 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 determine the relevance of the identified principal components to the clustering algorithm, a plot of the three principal components was generated for each experimental condition. The purpose of this plot was to assess whether these components could be partitioned in a meaningful way to ensure that the information from each of the components was relevant to the clustering process. The results can be seen on **Figure 14**.

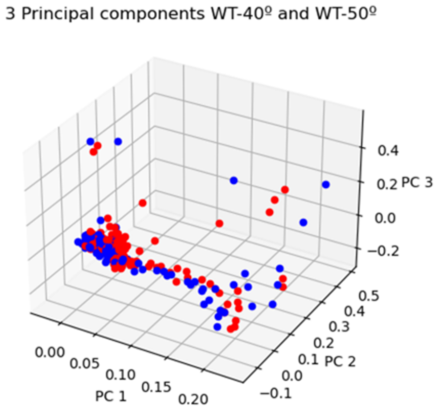


Figure 14: First 3 Principal components for each region and the different experimental conditions. WT-40º in blue and WT50º in red.

The plots generated from the application of PCA to the data indicate that, despite achieving over 70% explained variance, there is no clear distinction between the data points of the different conditions. Other thresholds were also applied given the specific clustering task, to see if increasing the explained variance could have some effect on the applicability of PCA. However, results did not show better results with PCA than raw data. 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 15**.

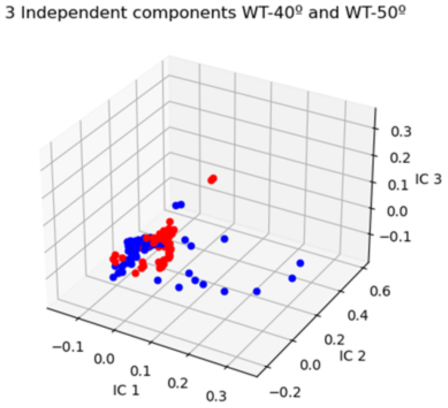


Figure 15: First 3 Independent components for each region and the different experimental conditions. WT-40º in blue and WT50º in red

However, visual inspection of the distributions of the first three independent components again failed to reveal a clear separation of the data. Similar to ICA, different numbers of independent components were applied to evaluate the clustering task, providing no clear advantage on its application. With this in mind, the clustering process was performed on the original raw data in an attempt to optimise performance.

5.2 Clustering techniques

To group our regions and find regions with similar activity patterns, we used a clustering approach. For this project, 3 main clustering techniques were evaluated: K-means clustering, K-shape clustering and hierarchical clustering. It is worth noting that all these techniques require the determination of the optimal number of clusters prior to their application. Therefore, in order to determine the optimal number of clusters, a comprehensive evaluation was performed using three different tests for each of the clustering techniques. The range evaluated for each of the techniques was between 2 and 12 clusters.

The first test used was the silhouette score, which compares the distance between clusters to the distance within clusters. The higher the score, the better the clusters are discriminated. The second test used was the Calinski-Harabasz index. Similar to the silhouette score, the higher the value, the better the clustering should be. Finally, the Davies-Bouldin was calculated. Contrary to the previous tests, a lower index indicates better clustering. A detailed presentation of the results obtained for the different clustering techniques used in this study is depicted in **Figure 16**.

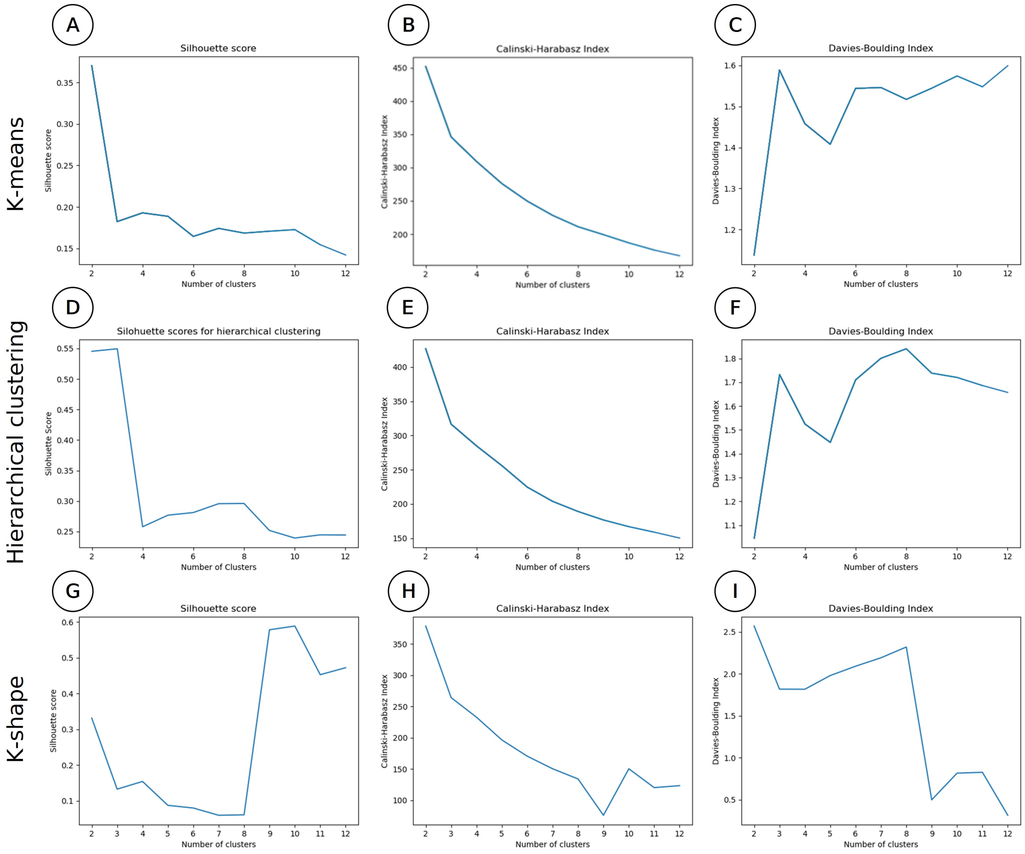


Figure 16: Tests to find the optimal number of clusters. For each of the Clustering techniques, from left to right: Silhouette score, Calinski-Harabasz index and Davies-Boulding Index. The specific values are not relevant for the decision. To decide, is important to look at the 'elbows' or big changes on the values between the number of clusters. The visualization has been made to make these changes more obvious.

The analysis of the clustering techniques shows that the three algorithms propose different numbers of clusters. Furthermore, there are discrepancies between the algorithms and the scoring methods. In particular, 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. To determine the best number of clusters, a common method is to find the 'elbows' of the different plots. This refers to the abrupt changes in score between clusters. The basic idea behind this method is to plot the variance against the number of clusters and identify when adding more clusters gives worse results. This point is often referred to as the 'elbow' of the plot, suggesting that the number of clusters is a reasonable choice.

For instance, **Figure 16, A** indicates that K = 4 would be the 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 K-means and hierarchical clustering give similar results, the K-shape algorithm gives significantly different results in comparison, and the optimal value for K seems to be around 9 or 10. (**Figure 16, G-I**). These variations between clustering techniques make it challenging to identify the optimal value of K based solely on the evaluation scores.

Therefore, in order to determine the optimal number of clusters, we decided to compare and evaluate different K values on a single technique. We chose agglomerative clustering for this purpose, as it is easy to follow and compare the divisions due to its mode of operation. We decided to study the different clustering outputs from K = 2 applied to all our data together in order to have a baseline for comparison. 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)**.

It should be noted that due to the specific data, which are complex time series, the decision for the specific number of Ks is very complex. Although in general a validation would be done with specific benchmarks for the data, in this case this was not possible. Therefore, the optimal number of clusters was decided in discussion with experts on fUS data, who are able to determine an optimal number of clusters based on whether hemodynamic responses shapes were likely to originate from the same activity type. To support this decision, we computed the average of the inter and intra-cluster correlation for each K. Results can be seen in **Table 3**.

K	12	11	10	9	8	7	6	5	4	3	2
Intra-Cluster correlation	0.40	0.39	0.37	0.33	0.32	0.29	0.28	0.30	0.31	0.25	0.16
Inter-Cluster correlation	0.28	0.28	0.28	0.24	0.29	0.31	0.30	0.29	0.31	0.37	0.28

Table 3: Intra-cluster and Inter-cluster Spearman correlation depending on the applied number of clusters.

As results show, the intra-cluster correlation increases with the increase in the number of clusters, while the inter-cluster correlation does not significantly increase, meaning that the clustering is still differentiated between groups, while becoming more homogeneous inside each group.

Furthermore, validation is mainly used for further generalisation. Although we try to propose a pipeline, the specific number of clusters needs to be evaluated for each experimental setup. It therefore cannot be replicated for new types of data. In this case, it was concluded that using less than 8 clusters would still leave some clusters with significant outliers from the respective group. To validate that no more clusters should be added, we also computed K = 13 and K = 14. With both values, we obtain a cluster with a single datapoint. As we were searching to group different time traces, having a single region in a cluster is not significant for our goal. Therefore, and taking into account the expert input together with the correlations computed, the final number of chosen clusters was K = 12.

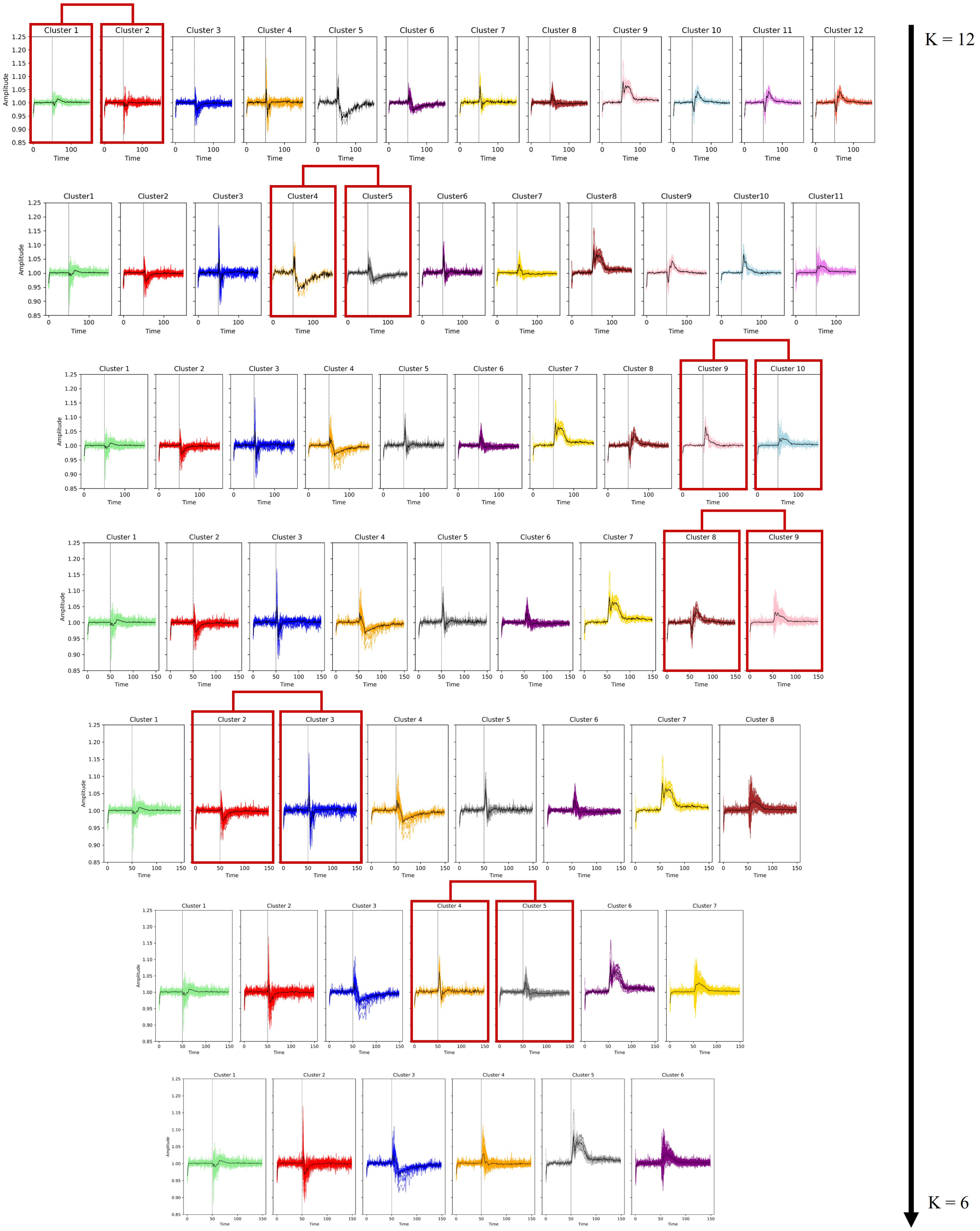


Figure 17: Representation of the temporal traces clustered using K-Means with different Values of K (from top to bottom, in a range of 6:12). The red squares indicate which clusters are merged in each step.

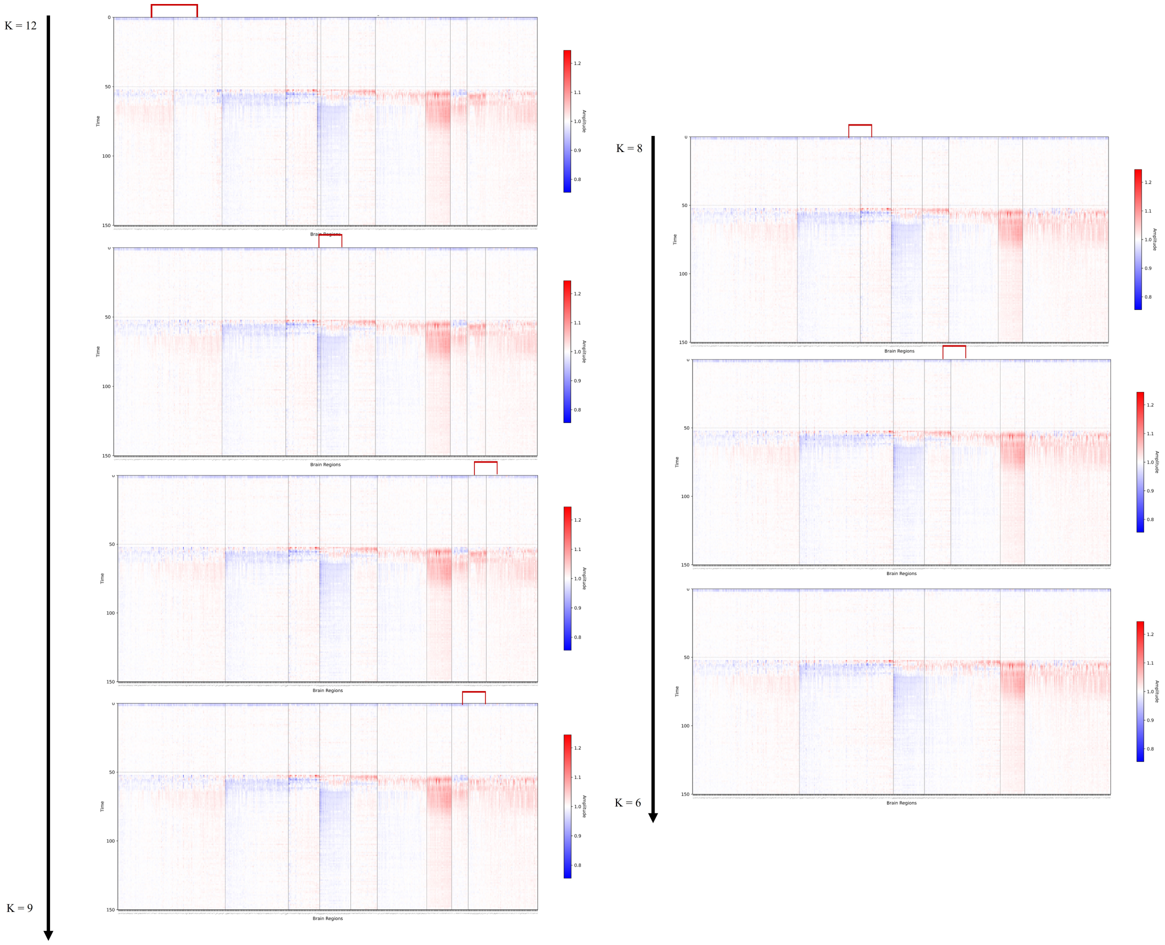


Figure 18: Colormesh representation of the temporal traces clustered using K-Means and different Values of K (range of 6:12). The red lines indicate which clusters are merged in each step. Blue represents deactivation of the brain region, while red refers to the activation of the same.

The next step was to select the optimal clustering technique. We used 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 worse than the other two techniques. In particular, some clusters had relatively small sizes, while others were less homogeneous compared to the other cases. In addition, the K-shape algorithm was slower than the other cases. Therefore, it was decided to continue the evaluation process only for the K-Means and Agglomerative clustering techniques.

5.3 Performance Evaluation

To determine the optimal clustering result, multiple scores were used 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 4. 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 20**.

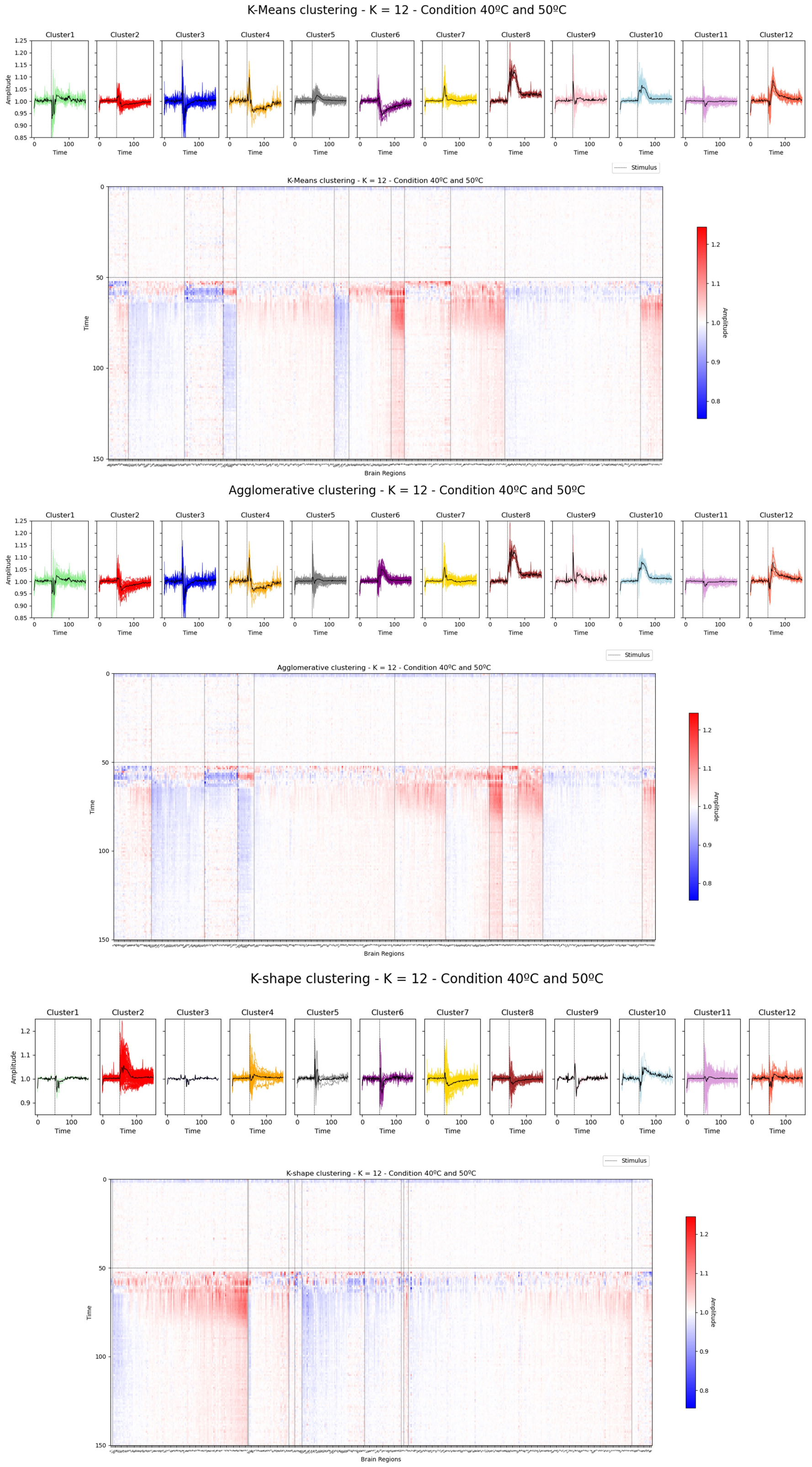


Figure 19: Clustering of the data using different clustering algorithms.

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

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

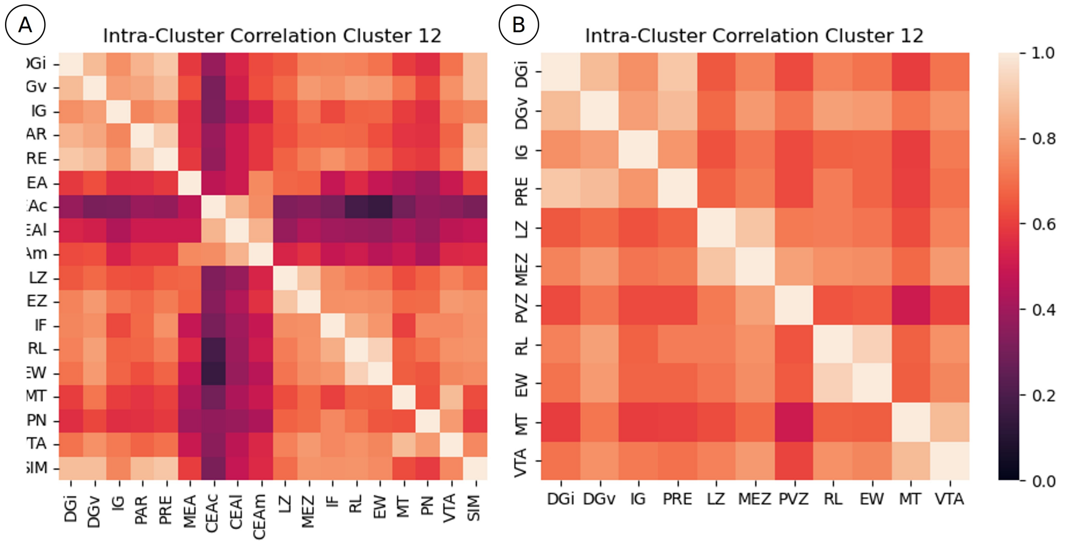


Figure 20: Intra-Cluster heatmap for Cluster 12. (**A**) K-means clustering (**B**) Hierarchical clustering.

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 21**.

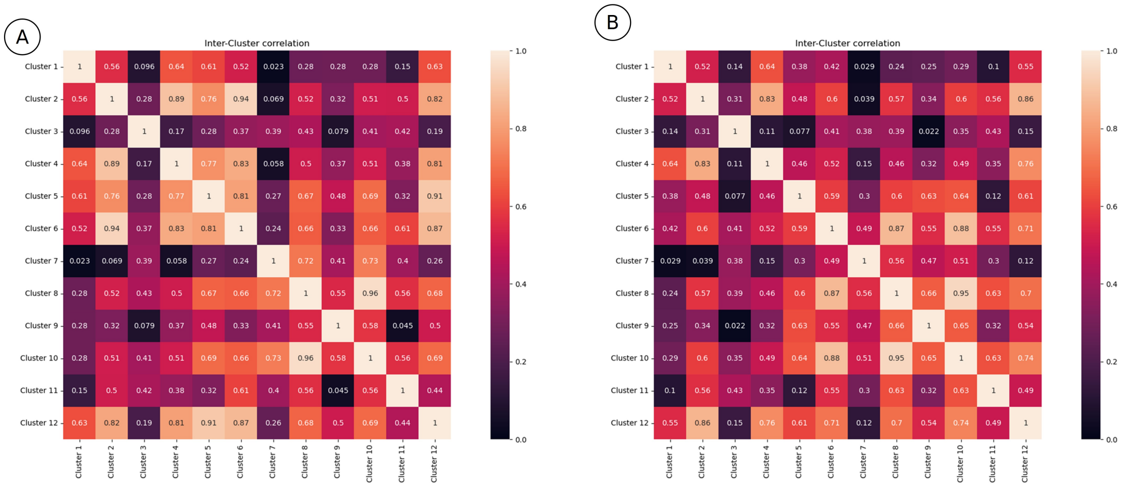


Figure 21: Inter-Cluster comparison for each of the clustering techniques. (**A**) K-means clustering (**B**) Hierarchical clustering.

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

However, although it can sometimes provide useful insights, the Spearman correlation only considers the monotonic relationship between the samples, ignoring the temporal aspect. In order to overcome this limitation and to provide a comprehensive comparison of the results, we also calculated the 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 5**.

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

Table 5: Intra-cluster Spearman correlation of the different clusters using different clustering techniques.

The Euclidean distance between clusters can be seen in **Figure 22**.

The results of both clustering techniques were consistent with Spearman correlation. Given the positive results of both approaches and the complexity of efficiently comparing time series, hierarchical clustering was ultimately chosen due to its implementation. When interpreting the data, this technique provides additional information about the merging clusters through the dendrogram, allowing a better understanding of the relationships between them. This insight allows the identification of brain regions with stronger connections and a greater degree of interdependence.

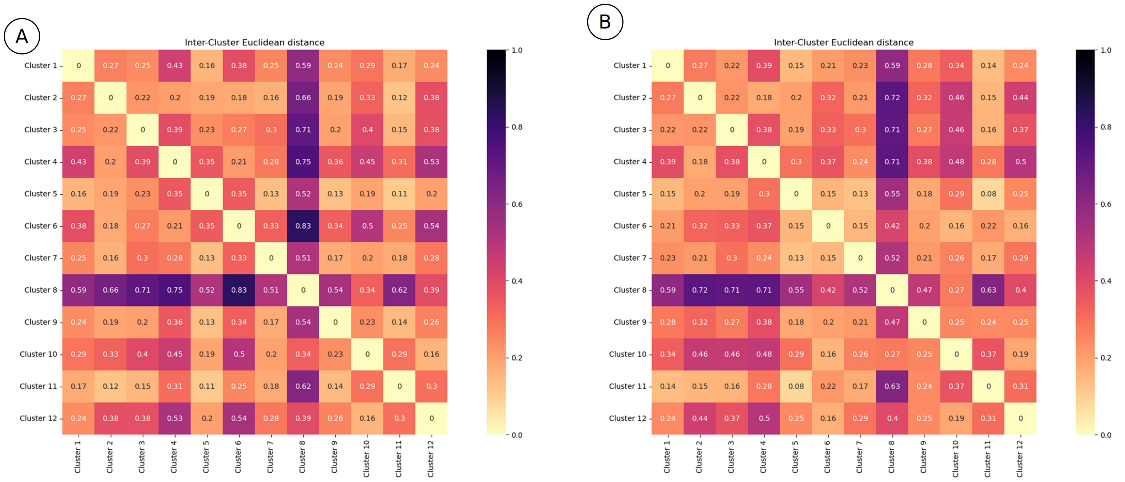


Figure 22: Inter-Cluster Euclidean distance for each of the clustering techniques. (**A**) K-means clustering (**B**) Hierarchical clustering.

5.4 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 23**. It is important to mention that the data presented here were also clustered with data from TKO mice that is not presented in this work. Therefore, some clusters may not be present in this study.

The subsequent step in the analysis process was to determine the regions of interest for classifying our specific stimulus. 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 fUS data. For each average, we visualized these features. An example of this can be seen in **Figure 24**.

Once these characteristics had been calculated for the average of each group, the cluster with the maximum value for a particular characteristic and the cluster with the minimum value were taken. An assessment was then made of which regions corresponding to the 40ºC stimulus changed from lowest to 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 appeared in cluster 8, which only contained regions that moved from other clusters, as the specific pattern only appears in the 50ºC condition. However, when we looked at which clusters the regions came from, we found that the movements were from clusters 6, 9 or 10, which had a similar pattern to cluster 8. Therefore, we used the features that did not involve cluster 8, which resulted in cluster 11 and cluster 12 having the greatest difference in activity.

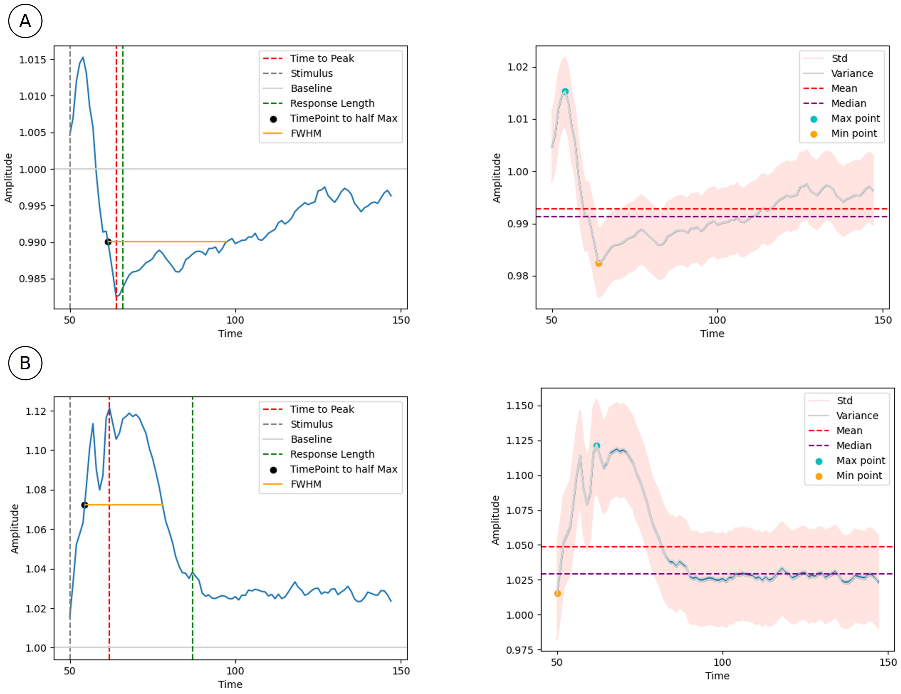


Figure 24: : Feature visualization in two different cluster averages. (A) Features in Cluster 1. (B) Features in Cluster 8.

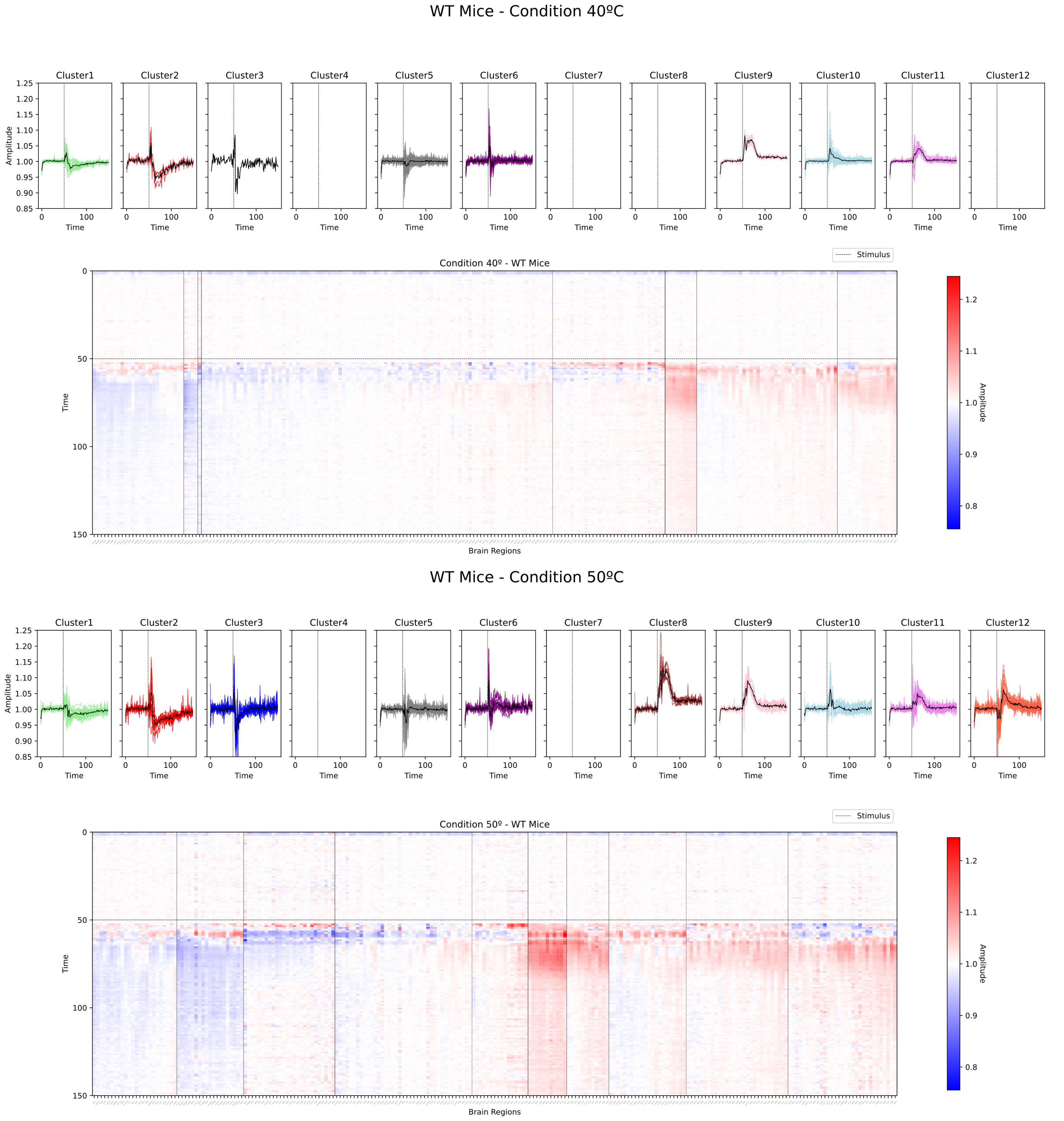


Figure 23: Hierarchical Clustering with K = 12 applied to all the datasets at the same time. For visualization purposes, each stimulus has been visualized separately using the labels obtained in the joined clustering.

5.5 Quantitative cluster description

Although time series are difficult to reflect upon and compare, it is possible to use basic statistical measures to better understand the distribution of the datapoints within the clusters. Therefore, the mean, median, mode and std of each of the time traces of the clusters was computed, and then averaged per cluster. Results can be seen in **Table 6**.

Cluster#	**Number of regions**	**Mean**	**Median**	**Mode**	**Std**
1	50	0.9952	0.9958	0.9615	0.0108
2	23	0.9881	0.9894	0.9397	0.0207
3	27	0.9981	1.0007	0.9068	0.0191
5	139	0.9993	1.0002	0.9507	0.0084
6	48	1.0042	1.0038	0.9558	0.0116
8	11	1.0321	1.0252	0.9691	0.0369
9	21	1.0165	1.0108	0.9604	0.0231
10	62	1.0042	1.0025	0.9710	0.0110
11	46	1.0078	1.0048	0.9604	0.0147
12	31	1.0089	1.0066	0.9296	0.0191

Table 6: Average of basic statistics per cluster obtained. Cluster 4 and 7 are not presented as they do not have data from WT mice, which is the one used on this work.

To see the distribution of each of the statistics inside each cluster see **Annex B**. Although descriptive statistics can help understand the homogeneity of the clusters, complex time series are difficultly represented by them. Therefore, it has to be noticed that these statistics do not provide a clear representation of the clusters.

5.6 Qualitative cluster description

In order to unambiguously refer to each cluster, compare them and define the different patterns of activity in each cluster, a brief qualitative description was made for each cluster, taking into account its main visible characteristics. These qualitative descriptions were made with the involvement of neurobiology experts and data analysts to ensure a correct and relevant description for each group. The descriptions can be found in **Table 7**.

Cluster#	**Qualitative description**
1	Weak short-term activation
2	Strong peak with mid-term activation
3	Strong peak with rapid deactivation
4	Only in TKO mice
5	Short term deactivation
6	Strong short-term activation with re-activation
7	Only in TKO mice
8	Very strong long-term activation
9	Strong long-term activation
10	Small mid-term activation
11	Medium long-term activation
12	Weak deactivation with long-term strong activation

Table 7: Qualitative descriptions of the final clusters.

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

5.7 Clustering visualization

Due to the large amount of data managed in clustering, its interpretation is not easy. Although the clustering quantification techniques can be used to select which regions to use, it is also important to be able to 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 visualisation tool has been created to show the movement of the different brain regions within the clusters as a function of their state. This makes it possible to see how each region behaves when different stimuli are applied, and can help to draw further conclusions on the subject. To this end, an interactive Sankey diagram tool has been created.

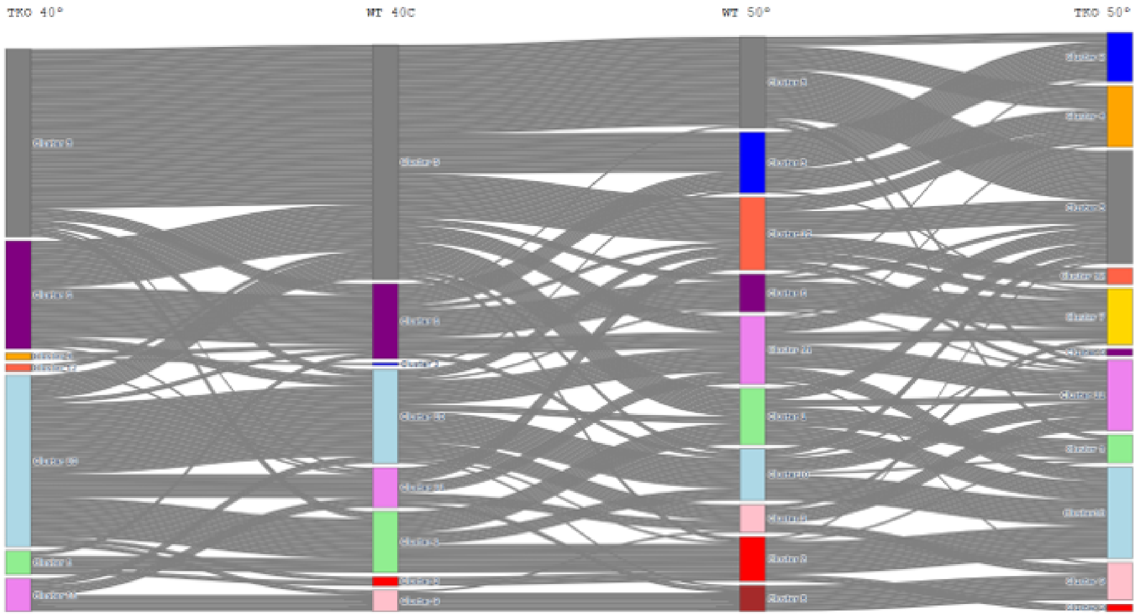


Figure 25: Sankey diagram for the visualization of brain region movement within clusters. From left to right: TKO mice at 40ºC, WT mice at 40ºC, WT mice at 50ºC and TKO mice at 50ºC.

In this case, we visualized the data for Wild Type mice and Triple Knock-Out (TKO) mice in both 40ºC and 50ºC experimental conditions. Although the TKO mice data has not been used in this work, it was added to illustrate the complexity of the visualization. The diagram can be seen in **Figure 25**.

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 as a function of its condition. This gives new insights into the behaviour of each region in each condition. For example, if a region is in cluster 5 for the 40ºC condition, and then we find the same region in cluster 12 for the 50ºC condition, we know that when pain is applied to the animal, this particular region responds with a stronger and longer activation.

To facilitate the implementation of this pipeline in other studies, a tool has been developed to automatically generate this type of graph from other data than the one used here. As well as allowing researchers to upload new data, it also facilitates the visualisation of specific regions rather than all at once. Finally, it supports the visualisation of individual time traces and the use of basic classification algorithms using specific regions for training.

5.8 Pipeline validation and insights

Having developed the entire pipeline and a visualisation tool, we wanted to see if the results we obtained had any real biological significance. In addition, we wanted to explore some of the insights it might provide given our experimental settings. To do this, we looked at whether regions that were spatially close to each other behaved similarly, and whether the somatosensory pathway, which was also used in the research, showed similarities in clusters, as would be expected.

First, we looked at cortical regions which are situated next to each other and have no direct implications for pain processing. For this, we chose the auditory areas of the brain which can be seen in **Figure 26**.

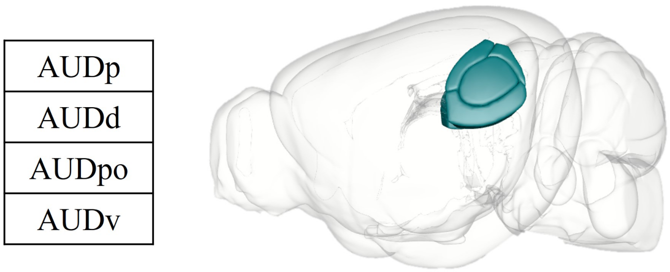


Figure 26: Auditory areas of the brain. Visualization of the areas in the left hemisphere.

We then investigated in which clusters each of these regions was for both conditions. Results can be seen in **Table 8**.

Region	Cluster for 40º stimulus	Cluster for 50º stimulus
AUDp	5	1
AUDd	5	1
AUDpo	5	1
AUDv	10	10

Table 8: Cluster belonging for each of the auditory regions given each stimulus.

We can see that three of the four regions have the same behaviour. Apart, we see that the three first regions do change cluster, showing a slight activation. This is due to the fact that during the experiment some noise artifacts that can be perceived by mice occur. However, we can see that there is an expected relationship between these areas, which validates the idea that the pipeline allows the study of brain region relationships and behaviours.

Next, we picked the somatosensory regions, which are expected to behave similarly and to give a response to a painful stimulus. The used regions were not those related to the somatosensory pathway, but exclusively the somatosensory regions, which can be seen in **Figure 27**. The clustering results for these regions can be seen in **Table 9**.

Region	Cluster for 40º stimulus	Cluster for 50º stimulus
SSp-bf	5	1
SSp-n	5	2
SSp-tr	10	10
SSp-ll	10	10
SSp-ul	10	1
SSp-un	10	1
SSs	10	10

Table 9: Cluster belonging for each of the somatosensory regions given each stimulus.

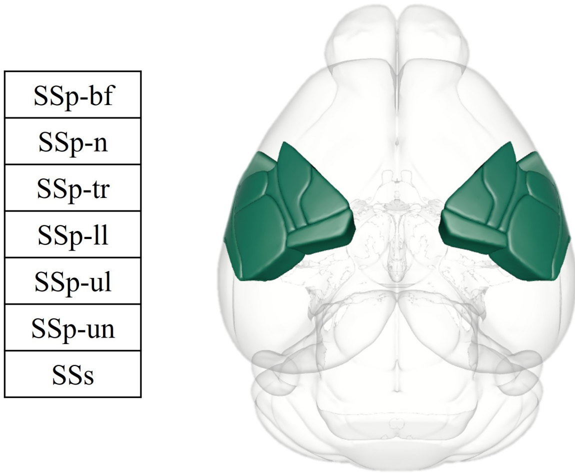


Figure 27: Somatosensory regions. The visualization is done for both hemispheres.

Here we can see that all the somatosensory regions are in closely related clusters, but in this case they activate in different ways. For example, we can see that the region associated with the nose (SSp-n) is highly activated, which could be due to behavioural changes when the animal is in pain. Both the upper limb and the unassigned (SSp-ul, SSp-un) also change clusters, showing a change in activity, probably linked to the fact that the mice receive the stimuli through the upper paw, while the lower limb and trunk (SSp-ll, SSp-tr) remain stable. All of these behaviours can provide interesting insights into what is happening in the brain, opening up the possibility of further studies on a given stimulus.

Finally, and worth mentioning, is that we found that there were some clusters that solely occurred in condition 50ºC, meaning that all the regions belonging to those clusters have a change in activation, potentially giving interesting information on the processing of the stimuli.

6 Region selection for classification

6.1 Dataset split and feature extraction

Before actually classifying our data, we cleaned the original dataset as there were some trials that had not been properly recorded during the experiments. Of the original 675 trials, 521 were used, with 396 trials for the 40ºC condition and 125 trials for the 50ºC condition. Apart from that, some trials had some regions that showed clear signals. However, as not all regions were used for classification, and to avoid eliminating all trials, we only discarded specific trials if the regions with clear signals were used for classification. We also eliminated the regions that ended with a low number of trials ($N_{50Ccondition}$ ¡ 100) from all the procedures, as they could not be compared to the other regions. The non-used regions can be found in **Annex C**. We then split our dataset into training (70%) and test (30%) groups. To approach the data in different ways, we not only used the raw data to classify, but 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 10**.

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

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

Table 10: Features extracted from each trial and Python functions used.

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.

6.2 Pre-processing and classifier selection

Given our data set, we performed data augmentation to balance the trials in both conditions. To do this, we calculated the standard deviation of the noise for each signal and added a random noise value in the range of the standard deviation. This takes into account the characteristics of the signal and only uses baseline fluctuations that are already present in our data. This is done to ensure that the added noise does not significantly distort or alter the underlying information present in the original signal. This allowed us to obtain significant augmented trials. We were then able to begin the process of classifier selection.

To choose the best classifier and data (raw or features) to use, we trained different classifiers using 8 regions 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. It is important to mention that the partition for the validation set on each fold was done before data augmentation to avoid contamination of the partition. 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. the process was repeated in 10 different randomized groups of regions for robustness. The specifications for the classifiers used are as follows:

- Logistic Regression: C = 0.1, class weight = None, fit intercept = True, random state = 0 , penalty = ’l2’, max iter = 10000, solver = ’sag’.
- SVM: 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.
- Decision Trees: criterion = ’gini’, max depth = 2, max features = ’sqrt’, min samples leaf = 1, min samples split = 8, splitter = ’best’.

The averaged results can be seen in **Table 11**.

Data	**Logistic Regression**	**SVM**	**Decision Trees**
Raw data	79.1%	**87.9%**	77.1%
Features	72.6%	**74.3%**	72.8%

Table 11: Accuracy results for the different classifiers with each type of data trained in the randomized group of regions.

Given these results, we decided to proceed with the study using SVM in the raw data and the previously mentioned parameters.

6.3 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.

6.3.1 Data-based selection

We started by calculating the individual accuracy for each region. We then selected the region with the highest accuracy and paired it with all the regions that had an individual accuracy above 70%. Each time we trained new classifiers using 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 that can be seen in **Figure 28**.

POL
MG
PP
SSp-bf
LGv
DGv
SF
CA2v

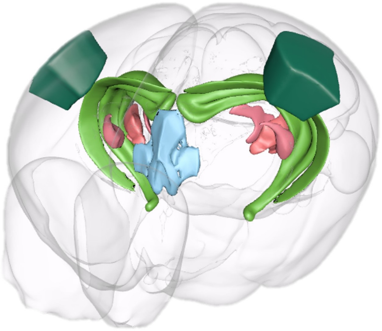


Figure 28: Data-based selected regions. In the table, the abbreviation of the selected regions. The image contains the regions coloured for both hemispheres.

6.3.2 Cluster-based selection

As previously stated in **4.4** we used those regions that moved from Cluster 5 to Cluster 12. This resulted in using the regions presented in **Figure 29**.

MEA
COAa
PIR
IA
BLAv
BMAa
EPv
CA3i
CA3v

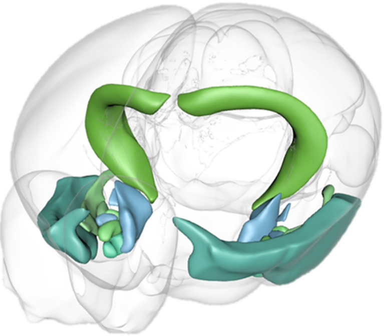


Figure 29: Cluster-based selected regions. In the table, the abbreviation of the selected regions. The image contains the regions coloured for both hemispheres.

6.3.3 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 regions on **Figure 30**.

ACAd
ACAv
AUDd
AUDpo
AUDv
Mop
Mos
RSPagl
RSPd

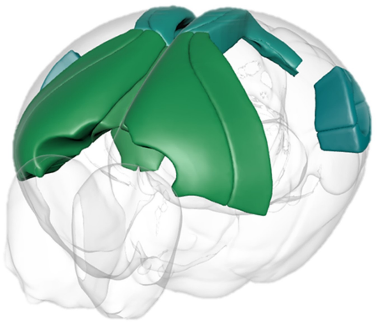


Figure 30: Cortical regions. In the table, the abbreviation of the selected regions. The image contains the regions colored for both hemispheres.

6.3.4 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 regions presented in **Figure 31**.

SSp-bf
SSp-n
SSp-tr
SSp-ll
SSp-ul
SSp-un
SSs
VM
VP

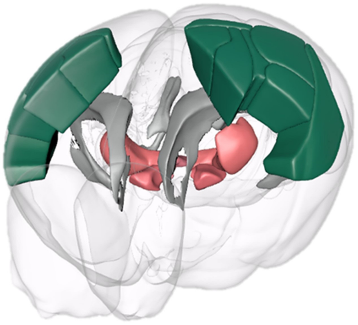


Figure 31: Somatosensory pathway regions. In the table, the abbreviation of the selected regions. The image contains the regions colored for both hemispheres.

6.3.5 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. To make sure the results were not dependant on the test dataset, we performed 10 different train/test splits and trained 10 classifiers per group of regions. Then we averaged the results from the 10 iterations. Final results can be seen in **Figure 32**.

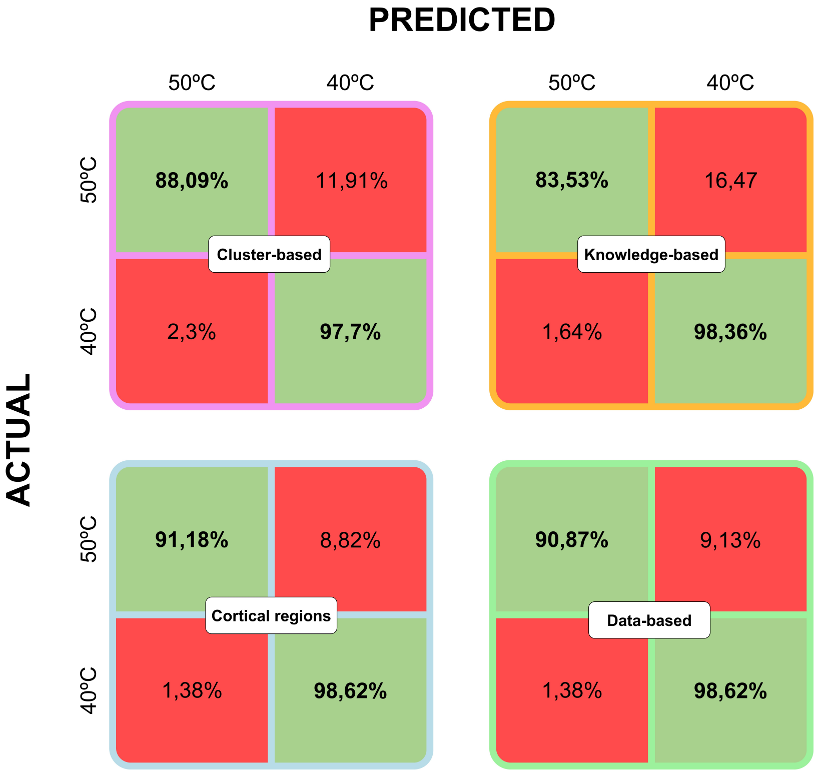
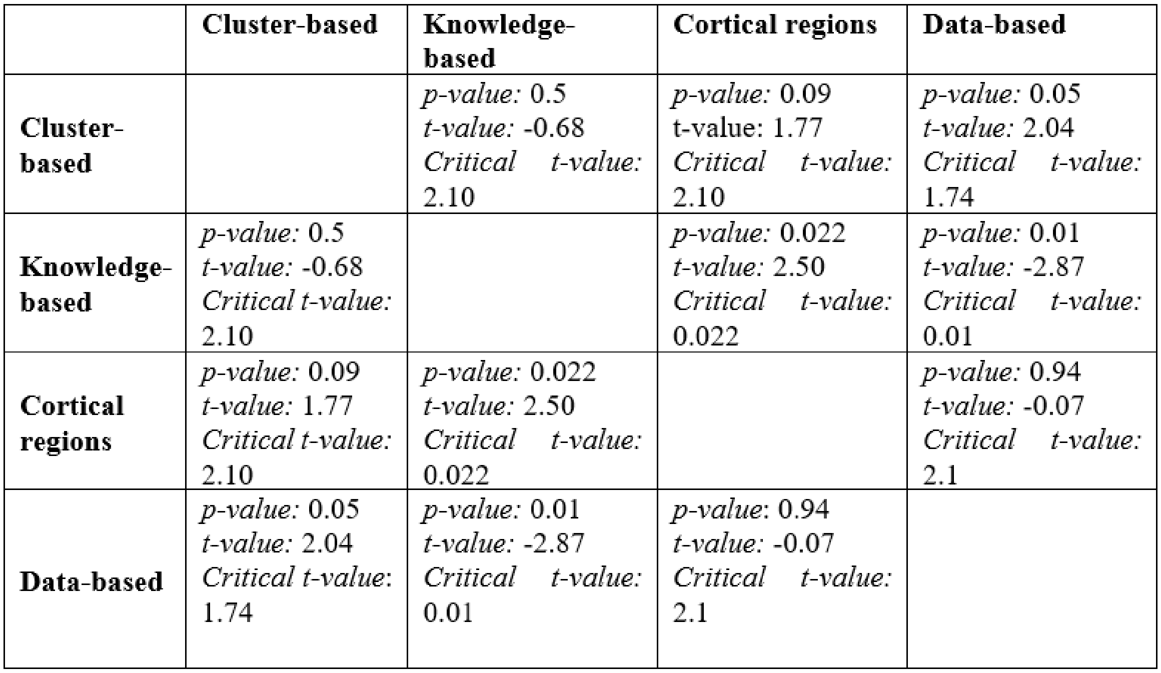


Figure 32: Confusion matrix of the classification values obtained in the test set for each of the region selection methods.

All the region selection methods performed higher than when randomized selection was made. All of the 4 methods performed with an accuracy higher than 90%. To compare the performance of each of the region groups, we applied an ANOVA test. The results indicated an F critical value of 2.87 and an F-value of 3.65, meaning that the null hypothesis of equal means is rejected. Then we performed a paired t-test to compare the classifiers between them. The effect size was evaluated through Cohen's D with a correction for small samples. As we were performing multiple t-tests, a Bonferroni correction was applied, lowering the α level to 0.083.

Based on the results shown in Figure 33, we can see that no significant differences were found between the region selection methods. However, we can see that the results from the Knowledge-based and Cluster-based approach are slightly lower than in the other two classifiers, without reaching a significant difference. This suggests that all classification methods were valid for classifying the different conditions.

Not only we compared the classifiers between them, but we also compared the performance between conditions. For all the classifiers, the 40ºC condition was significantly better classified than the 50ºC ($p - value < 0.05$ for all the methods). However, this could be due to the difference on the number of trials between both conditions, although the effect was reduced through data augmentation.



	Cluster-based	**Knowledge-based**	**Cortical regions**	**Data-based**
Cluster-based		*p-value:* 0.5 *t-value:* -0.68 *Critical t-value:* 2.10	*p-value:* 0.09 t-value: 1.77 *Critical t-value:* 2.10	*p-value:* 0.05 *t-value:* 2.04 *Critical t-value:* 1.74
Knowledge-based	*p-value:* 0.5 *t-value:* -0.68 *Critical t-value:* 2.10		*p-value:* 0.022 *t-value:* 2.50 *Critical t-value:* 0.022	*p-value:* 0.01 *t-value:* -2.87 *Critical t-value:* 0.01
Cortical regions	*p-value:* 0.09 *t-value:* 1.77 *Critical t-value:* 2.10	*p-value:* 0.022 *t-value:* 2.50 *Critical t-value:* 0.022		*p-value:* 0.94 *t-value:* -0.07 *Critical t-value:* 2.1
Data-based	*p-value:* 0.05 *t-value:* 2.04 *Critical t-value:* 1.74	*p-value:* 0.01 *t-value:* -2.87 *Critical t-value:* 0.01	*p-value*: 0.94 *t-value:* -0.07 *Critical t-value:* 2.1	

Figure 33: paired t-test results to compare each of the region selection methods between them. A p¡0.083 indicates statistical differences between the accuracy results of the compared methods. In green, the comparisons that showed significant differences. All the tests have a df = 18.

Discussion

This study introduces a pipeline to facilitate the implementation of vFUs data into BCI systems and support its interpretation. This kind of data presents inherent challenges due to its high-dimensional and complex nature.

This proposal had two main objectives. The first was to develop a pipeline for large-scale functional data visualisation. The developed work incorporates hierarchical clustering and visualisation techniques to complement the high spatio-temporal resolution of fUS imaging. By applying this pipeline, specific brain regions with consistent and significant changes in response to stimuli can be identified and analysed, providing a more comprehensive understanding of the behaviour of individual regions of brain activity. The results show that by clustering the data into different activity patterns and visualising the inter-cluster changes of individual regions between conditions, the interpretation of the data can be aided. For our specific case, and after several trials and expert evaluation, we concluded that hierarchical clustering with K = 12 provided satisfactory grouping into differentiated activity patterns. These parameters will need to be re-evaluated for implementation on other data, as this will largely depend on the data we are trying to cluster. However, the results obtained can be used as a first approach in similar experimental conditions.

Furthermore, through a qualitative description of the clusters and with the support of expertise, it is possible to understand how a particular brain region behaves in two separate but related experimental conditions. This was illustrated in the work by studying different groups of regions, where we could understand the behaviour of different ROIs by comparing the cluster they were assigned to in each condition. However, the insights gained through this method were not validated with further experiments in this paper. It is important to recognise that this method cannot be used alone to draw conclusions about brain behaviour, but rather as a tool to guide further research.

The second objective was to propose a region selection method to achieve high performance in stimulus decoding. To evaluate the effectiveness of the proposed approach, we compared different region selection methods by using the selected brain regions to classify the stimuli using algorithms already applied in neuroscience research [Maher et al., 2023, Singh and Singh, 2020, Xu et al., 2019]. Results show that general methods for region selection, such as knowledge-based methods, that, are specific for the stimulus but not to the studied data yield to inferior results (90,95%).

The proposed method following the pipeline showed results comparable to data-based selection (92.9% for cluster-based selection vs. 94.7% for data-based selection), suggesting that the approach is able to assist in identifying the best regions to use for decoding the stimuli and given specific data. Finally, we examined the performance of cortical regions compared to deeper regions. The results show that their performance is also comparable (94.9%). This finding is particularly interesting as it suggests that, under certain experimental conditions, imaging deeper in the brain does not necessarily improve stimulus decoding. Furthermore, the classification results show that fUS data are able to provide a high accuracy for the decoding of the stimuli, as an accuracy of more than 90% was achieved in all region selection methods. However, it is important to emphasise that this accuracy is highly dependent on the selection of relevant regions, as randomised region selection led to lower classification results (less than 90% in all cases). This underlines the importance of using regions that are specifically relevant to the stimulus studied, and increases the relevance of the study for the successful application of fUS in BCI.

While this study provides interesting insights, it is important to acknowledge its limitations. Firstly, the research was conducted in a single experimental condition. Although this is common practice in neuroscience, investigating the effects of the proposed pipeline in other experimental conditions would help to determine the relevance and robustness of the work. Secondly, no ground truth was available to validate the proposed interpretation and visualisation. This means that specific expertise or further studies for the specific stimuli are needed to support the information extracted from the clustering and visualisation processes. Finally, the dataset used had a limited number of trials, which complicates the spatial selection of parameters for stimulus prediction.

Considering the results, the developed pipeline shows potential to contribute to the improvement of task understanding and decoding accuracy in BCI systems. The approach opens the door to a better understanding of the specific tasks developed with BCIs and suggests alternative means of selecting relevant data for the desired outcomes of researchers. By clustering brain regions and visualising cluster changes, the approach supports the identification of the most informative brain regions for decoding specific stimuli. The pipeline's insights into the performance of different brain regions in classifying stimuli could inform the design of BCI systems. By understanding which types of brain activity are most relevant to a stimulus, it may be possible to guide the development of signal processing algorithms and feature extraction methods tailored to specific decoding tasks. Further comparative studies between the developed pipeline and other machine learning-based approaches, such as deep learning models, could provide a comprehensive evaluation of the pipeline's performance and its potential for scalability and generalisation across different experimental conditions and populations.

Finally, although this study is based on vFUs data, the proposed approach is not restricted to this type of imaging. The pipeline could potentially be used with other types of brain imaging techniques to aid region selection, which may be relevant for techniques that require specific electrode positioning, such as EEG. In conclusion, the work presented here can help in the interpretation of high-dimensional brain data. This can aid in the understanding of specific tasks, which in turn can help in the design of more accurate and reliable BCIs. Further investigations can help to determine the relevance of the pipeline and its effects in different experimental settings.

Annexes

Annex A: Brain regions used in this work with their abbreviations. As seen in [Wang et al., 2020].

Brain Region	Abbr.	Brain Region	Abbr.
Anterior cingulate area, dorsal part	ACAd	Subparafascicular nucleus	SPF
Anterior cingulate area, ventral part	ACAv	Posterior thalamic nucleus	PoT
Infralimbic area	ILA	Ventral anterior nucleus of the thalamus	VAL
Prelimbic area	PL	Ventromedial nucleus of the hypothalamus	VM
Dorsal auditory area	AUDd	Ventral pallidum	VP
Primary auditory area	AUDp	Lateral zone of the cerebellum	LZ
Posterior auditory area	AUDpo	Mesencephalic reticular formation	MEZ
Ventral auditory area	AUDv	Periventricular nucleus of the hypothalamus	PVR
Primary motor area	MOp	Paraventricular nucleus of the hypothalamus	PVZ
Secondary motor area	MOs	Central linear nucleus	CLI
Retrosplenial area, lateral agranular part	RSPagl	Dorsal raphe nucleus	DR
Retrosplenial area, dorsal part	RSPd	Interfascicular nucleus	IF
Retrosplenial area, ventral part	RSPv	Rostral linear nucleus	RL
Primary somatosensory area, barrel field	SSp-bf	Pedunculopontine nucleus	PPN
Primary somatosensory area, nose	SSp-n	Substantia nigra pars compacta	SNc
Primary somatosensory area, trunk	SSp-tr	Anterior thalamic nucleus	AT
Primary somatosensory area, lower limb	SSp-ll	Cuneate nucleus	CUN
Primary somatosensory area, upper limb	SSp-ul	Dorsal tegmental nucleus	DT
Primary somatosensory area, unassigned	SSp-un	Edinger-Westphal nucleus	EW
Supplemental somatosensory area	SSs	Lateral tegmental field	LT
Temporal association areas	TEa	Medial terminal nucleus	MT
Visceral area	VISC	Median raphe nucleus	MRN
Agranular insular area	AI	Reticular nucleus	RR
Laterointermediate area	VISli	Oculomotor nucleus	III
Anterior visual area	VISa	Pontine nuclei	PN
Anterolateral visual area	VISal	Paramedian tract nucleus	Pa4
Anteromedial visual area	VISam	Periaqueductal gray	PAG
Lateral Visual area	VISl	Pretectal nucleus	PRT
Primary visual area (posterolateral)	VISp(pl)	Red nucleus	RN
Primary visual area (anterolateral)	VISp(al)	Substantia nigra pars reticulata	SNr
Primary visual area (anteromedial)	VISp(am)	Superior colliculus, deep layers (anterior)	SCd(a)
Primary visual area (posteromedial)	VISp(pm)	Superior colliculus, deep layers (posterior)	SCd(p)

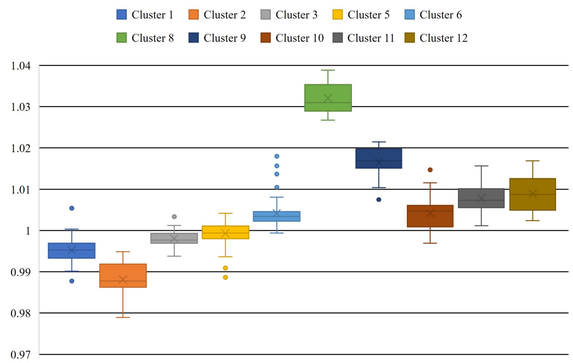
Brain Region	Abbr.	Brain Region	Abbr.
Posteromedial visual area	VISpm	Superior colliculus, intermediate layers (anterolateral)	SCi(al)
Rostrolateral area	VISrl	Superior colliculus, intermediate layers (posterolateral)	SCi(pl)
Posterolateral visual area	VISpl	Superior colliculus, superficial layers (anterolateral)	SCs(al)
Postrhinal area	VISpor	Superior colliculus, superficial layers (posterolateral)	SCs(pl)
Cortical amygdalar area, anterior part	COAa	Superior colliculus, intermediate layers (Anteromedial)	SCi(am)
Cortical amygdalar area, posterolateral part	COApl	Superior colliculus, superficial layers (Anteromedial)	SCs(am)
Cortical amygdalar area, posteromedial part	COApm	Superior colliculus, superficial layers (posteromedial)	SCs(pm)
Piriform area	PIR	Ventral tegmental area	VTA
Thalamic reticular nucleus	TR	Ventral tegmental nucleus	VTN
Primary auditory cortex	PA	Inferior colliculus, central nucleus	ICc
Lateral amygdala	LA	Inferior colliculus, dorsal nucleus	ICd
Basolateral amygdala, anterior part	BLAa	Inferior colliculus, external nucleus	ICe
Basolateral amygdala, posterior part	BLAp	Medial vestibular nucleus	MEV
Basolateral amygdala, ventral part	BLAv	Nucleus basalis	NB
Basomedial amygdala, anterior part	BMAa	Subparaventricular zone of the hypothalamus	SAG
Basomedial amygdala, posterior part	BMAp	Parabrachial nucleus, lateral division	PBG
Endopiriform nucleus, dorsal part	EPd	Superior colliculus	SCO
Endopiriform nucleus, ventral part	EPv	Gigantocellular reticular nucleus	GRN
Claustrum	CLA	Intermediate reticular nucleus	IRN
Anterior pretectal nucleus	APr	Cochlear nucleus	CN
Hypothalamic area, tuberal part	HATA	Ventral nucleus of the lateral lemniscus	VNC
CA1 subfield of the hippocampus, dorsal part	CA1d	Nucleus of the trapezoid body	NTB
CA1 subfield of the hippocampus, stratum oriens (inhibitory interneurons)	CA1i	Nucleus of the solitary tract	NTS
CA1 subfield of the hippocampus, stratum pyramidale (pyramidal cells)	CA1v	Subparaventricular zone	SPVO
CA2 subfield of the hippocampus, dorsal part	CA2d	Primary visual cortex	VI
CA2 subfield of the hippocampus, inhibitory interneurons	CA2i	Paraventricular nucleus	PARN
CA2 subfield of the hippocampus, ventral part	CA2v	Nucleus x	x
CA3 subfield of the hippocampus, dorsal part	CA3d	Nucleus y	y
CA3 subfield of the hippocampus, inhibitory interneurons	CA3i	Phylogenetically conserved brain regions	PHY
CA3 subfield of the hippocampus, ventral part	CA3v	Nucleus raphe	NRaphe
Dentate gyrus, dorsal part	DGd	Anterior commissure, level VII	ACVII
Dentate gyrus, inhibitory interneurons	DGi	Laterodorsal tegmental nucleus	LDT

Brain Region	Abbr.	Brain Region	Abbr.
Dentate gyrus, ventral part	DGv	Locus coeruleus	LC
Induseum griseum	IG	Nucleus of the inferior colliculus	NI
Entorhinal cortex, lateral division	ENTl	Raphe pallidus nucleus	RPO
Entorhinal cortex, medial division	ENTm	Bed nucleus of the stria terminalis	B
Parasubiculum	PAR	Dorsal tegmental nucleus	DTN
Postsubiculum	POST	Interfascicular nucleus at level 5 of the medulla	I5
Pre-subiculum	PRE	Principal sensory nucleus of the trigeminal nerve at level 5 of the medulla	P5
Prosubiculum	ProS	Vagus nerve nucleus	V
Subiculum	SUB	Parabrachial nucleus	PB
Medial amygdala	MEA	Nucleus laminaris, lateral part	NLL
Intercalated amygdala	IA	Pontine central gray	PCG
Anterior amygdaloid area	AAA	Paragigantocellular nucleus	PG
Central amygdala, capsular part	CEAc	Solitary nucleus, gelatinous part	SG
Central amygdala, lateral part	CEAl	Supratrigeminal nucleus	SUT
Central amygdala, medial part	CEAm	Tegmental reticular nucleus	TRN
Caudoputamen, dorsolateral part	CPcdl	Superior olivary complex	SOC
Caudoputamen, dorsomedial part	CPcdm	Principal sensory nucleus of the trigeminal nerve	PSV
Caudoputamen, central part, lateral zone	CPcml	Pararubral nucleus, caudal part	PRNc
Caudoputamen, central part, medial zone	CPcmm	Pontine reticular nucleus	PRNr
Caudoputamen, ventral part	CPcv	Superior lateral cortex	SLC
Pallidum, dorsocentral part	CPmdc	Anterior nucleus of the cortex, rostral part 1	ANcr1
Pallidum, dorsolateral part	CPmdl	Anterior nucleus of the cortex, rostral part 2	ANcr2
Pallidum, dorsomedial part	CPmdm	Copula pyramidis	COPY
Pallidum, central part, medial zone	CPmmc	Paraflocculus	PFL
Pallidum, medial part, lateral zone	CPmml	Paramedian lobule	PRM
Pallidum, ventral part, central zone	CPmvc	Crus II of the cerebellar cortex	SIM
Pallidum, ventral part, lateral zone	CPmvl	Crus II of the cerebellar cortex	CENT2
Lateral septal nucleus	LS	Crus III of the cerebellar cortex	CENT3
Septofimbrial nucleus	SF	Culmen of the cerebellar cortex	CUL4
External segment of the globus pallidus	GPe	Lingula of the cerebellar cortex	LING
Internal segment of the globus pallidus	GPi	Dentate nucleus	DN
Substantia innominata	SI	Interpositus nucleus	IP
Triangular nucleus of septum	TRS	Bed nucleus of the stria terminalis	BST
Anterodorsal nucleus of thalamus	AD	Mediodorsal nucleus of thalamus	MD
Anteromedial nucleus of thalamus	AM	Pretectal nucleus	PR
Anteroventral nucleus of thalamus	AV	Superficial mesencephalic nucleus	SMT
Interanterodorsal nucleus of thalamus	IAD	Nucleus of the reuniens	RE
Interanteromedial nucleus of thalamus	IAM	Parataenial nucleus	PT
Laterodorsal nucleus of thalamus	LD	Paraventricular nucleus of thalamus	PVT
Epithalamus	EPI	Intercalated nucleus of the amygdala	Xi
Intergeniculate leaflet	IGL	Reticular nucleus of thalamus	RT
Ventral lateral geniculate nucleus	LGv	Dorsal lateral geniculate nucleus	LGd

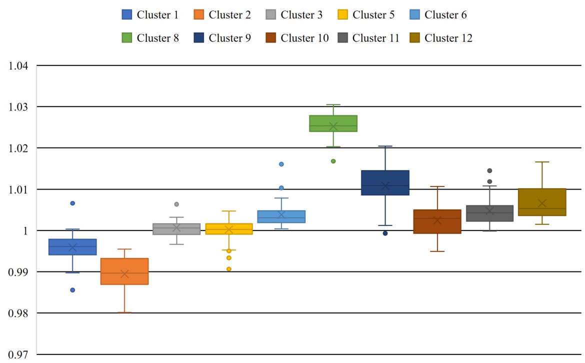
Brain Region	Abbr.	Brain Region	Abbr.
Intergeniculate nucleus	IntG	Medial geniculate nucleus	MG
Supragenual nucleus	SubG	Peripeduncular nucleus	PP
Central lateral nucleus of thalamus	CL	Subparafascicular nucleus	SPA
Central medial nucleus of thalamus	CM	Posterior nucleus of thalamus	PO
Paracentral nucleus	PCN	Lateral posterior nucleus of thalamus	LP
Rhomboid nucleus	RH	Subgeniculate nucleus	SGN
Parafascicular nucleus	PF	Polysensory cortex	POL
Posterior intralaminar nucleus	PIL	Intermediodorsal nucleus of thalamus	IMD
Ethmoid nucleus	Eth		

Annex B: Distribution of basic statistics inside a cluster

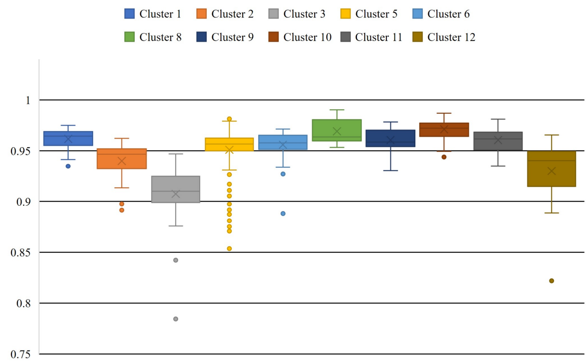
Mean values of the clusters



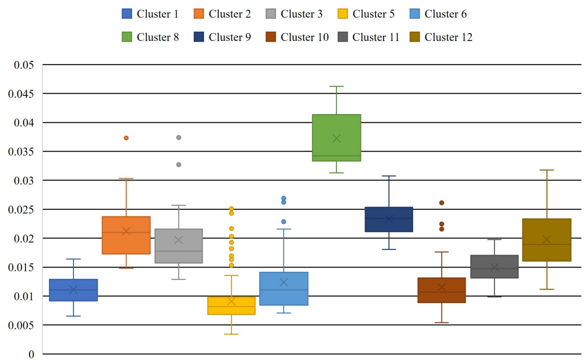
Medians values of the clusters



Mode values of the clusters



Std values of the clusters



.

Annex C: 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

References

[Abiri et al., 2019] Abiri, R., Borhani, S., Sellers, E. W., Jiang, Y., and Zhao, X. (2019). A comprehensive review of eeg-based brain–computer interface paradigms. *Journal of neural engineering*, 16:011001.

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

[Bamdad et al., 2015] Bamdad, M., Zarshenas, H., and Auais, M. A. (2015). Application of bci systems in neurorehabilitation: a scoping review. *Disability and Rehabilitation: Assistive Technology*, 10:355–364.

[Breiman, 1996] Breiman, L. (1996). Bagging predictors. *Machine learning*, 24:123–140.

[Brunner et al., 2020] Brunner, C., Grillet, M., Sans-Dublanc, A., Farrow, K., Lambert, T., Mace, E., Montaldo, G., and Urban, A. (2020). A platform for brain-wide volumetric functional ultrasound imaging and analysis of circuit dynamics in awake mice. *Neuron*, 108:861–875.e7.

[Cano et al., 2022] Cano, S., Soto, J., Acosta, L., Peñeñory, V. M., and Moreira, F. (2022). Using brain-computer interface to evaluate the user experience in interactive systems. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging Visualization*, pages 1–9.

[Casson et al., 2018] Casson, A. J., Abdulaal, M., Dulabh, M., Kohli, S., Krachunov, S., and Trimble, E. (2018). *Electroencephalogram*. Springer.

[Claron et al., 2021] Claron, J., Royo, J., Arcizet, F., Deffieux, T., Tanter, M., and Pouget, P. (2021). The supplementary eye field tracks cognitive efforts. *bioRxiv*.

[Constable, 2006] Constable, R. T. (2006). *Challenges in fMRI and its limitations*. Springer.

[Cutrell and Tan, 2008] Cutrell, E. and Tan, D. (2008). Bci for passive input in hci. volume 8, pages 1–3. Citeseer.

[Dash et al., 2020] Dash, D., Wisler, A., Ferrari, P., Davenport, E. M., Maldjian, J., and Wang, J. (2020). Meg sensor selection for neural speech decoding. *IEEE Access*, 8:182320–182337.

[Demene et al., 2016] Demene, C., Bernal, M., Delanoe, C., Auvin, S., Biran, V., Alison, M., Mairesse, J., Harribaud, E., Pernot, M., and Tanter, M. (2016). Functional ultrasound imaging of the brain activity in human neonates. pages 1–3. IEEE.

[Dhanabalan et al., 2022] Dhanabalan, T., Jose, N. C., and Chukwuka, O. (2022). Non-invasive brain-computer interface for prosthetic limbs control in military.

[Diya et al., 2019] Diya, S. Z., Prorna, R. A., Rahman, I. I., Islam, A. B., and Islam, M. N. (2019). Applying brain-computer interface technology for evaluation of user experience in playing games. pages 1–6. IEEE.

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

[Dodia et al., 2019] Dodia, S., Edla, D. R., Bablani, A., Ramesh, D., and 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 et al., 2022] Du, B., Cheng, X., Duan, Y., and Ning, H. (2022). fmri brain decoding and its applications in brain–computer interface: A survey. *Brain Sciences*, 12:228.

[Dupret and Piwowarski, 2008] Dupret, G. E. and Piwowarski, B. (2008). A user browsing model to predict search engine click data from past observations. pages 331–338.

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

[eun Kang Miller et al., 2018] eun Kang Miller, J., Miller, B. R., and Yuste, R. (2018). An increase in spontaneous activity mediates visual habituation. *bioRxiv*, page 507814.

[Fahimi et al., 2019] Fahimi, F., Zhang, Z., Goh, W. B., Lee, T.-S., Ang, K. K., and Guan, C. (2019). Inter-subject transfer learning with an end-to-end deep convolutional neural network for eeg-based bci. *Journal of neural engineering*, 16:026007.

[Fu et al., 2019] Fu, R., Tian, Y., Bao, T., Meng, Z., and Shi, P. (2019). Improvement motor imagery eeg classification based on regularized linear discriminant analysis. *Journal of medical systems*, 43:1–13.

[Gannouni et al., 2020] Gannouni, S., Belwafi, K., Aboalsamh, H., AlSamhan, Z., Alebdi, B., Almassad, Y., and Alobaedallah, H. (2020). Eeg-based bci system to detect fingers movements. *Brain Sciences*, 10:965.

[Gezgez and Kaçar, 2021] Gezgez, C. and Kaçar, E. (2021). Virtual character control by brain-computer interface and comparison of performance metrics. pages 1–7. IEEE.

[Glavas et al., 2022] Glavas, K., Prapas, G., Tzimourta, K. D., Tzallas, A. T., Giannakeas, N., and Tsipouras, M. G. (2022). Intra-user analysis based on brain-computer interface controlled game. pages 386–390. IEEE.

[Guyon et al., 2009] Guyon, I., Luxburg, U. V., and Williamson, R. C. (2009). Clustering: Science or art. pages 1–11. NIPS Vancouver, BC, Canada.

[Han et al., 2020] Han, C.-H., Müller, K.-R., and 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*, 28:2102–2112.

[Hastie et al., 2009] Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer.

[Hernandez-Cuevas et al., 2020] Hernandez-Cuevas, B., Egbert, W., Denham, A., Mehul, A., and Crawford, C. S. (2020). Changing minds: Exploring brain-computer interface experiences with high school students. pages 1–10.

[Jeyakumar et al., 2022] Jeyakumar, V., Krishnan, P. T., Sundaram, P., and Raj, A. N. J. (2022). *Brain-computer interface in Internet of Things environment*. Elsevier.

[Kamrani, 2014] Kamrani, E. (2014). Title: On-chip integrated functional near infra-red spectroscopy (fnirs) photoreceiver for portable brain imaging référence.

[Khalaf et al., 2019a] Khalaf, A., Sejdic, E., and 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 et al., 2019b] Khalaf, A., Sejdic, E., and Akcakaya, M. (2019b). Eeg-ftcd hybrid brain–computer interface using template matching and wavelet decomposition. *Journal of Neural Engineering*, 16:036014.

[Krishnan and Athavale, 2018] Krishnan, S. and Athavale, Y. (2018). Trends in biomedical signal feature extraction. *Biomedical Signal Processing and Control*, 43:41–63.

[Kumar and Sharma, 2012] Kumar, S. and Sharma, M. (2012). Bci: Next generation for hci. *International Journal of Advanced Research in Computer Science and Software Engineering.*

[Kundu and Ari, 2022] Kundu, S. and Ari, S. (2022). Brain-computer interface speller system for alternative communication: a review. *IRBM*, 43:317–324.

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

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

[Lambert et al., 2022] Lambert, T., Niknejad, H. R., Kil, D., Brunner, C., Montaldo, G., Nuttin, B., and Urban, A. (2022). Single voxel clustering for prior-free profound spatial analysis of functional ultrasound signals.

[Leeb et al., 2015] Leeb, R., Tonin, L., Rohm, M., Desideri, L., Carlson, T., and Millán, J. D. R. (2015). Towards independence: A bci telepresence robot for people with severe motor disabilities. *Proceedings of the IEEE*, 103:969–982.

[Li et al., 2009] Li, M., Chai, Q., Kaixiang, T., Wahab, A., and Abut, H. (2009). Eeg emotion recognition system. *In-vehicle corpus and signal processing for driver behavior*, pages 125–135.

[Lo et al., 2022] Lo, Y. T., Premchand, B., and Libedinsky, C. (2022). Neural correlates of learning in a linear discriminant analysis brain-computer interface paradigm. *Journal of Neural Engineering*, 19:056041.

[Maaten et al., 2009] Maaten, L. V. D., Postma, E., and den Herik, J. V. (2009). Dimensionality reduction: a comparative. *J Mach Learn Res*, 10:13.

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

[Maher et al., 2023] Maher, O. N., Haikal, A. Y., Elhosseini, M. A., and Saafan, M. (2023). An optimized quadratic support vector machine for eeg based brain computer interface. *International journal of electrical and computer engineering systems*, 14:83–91.

[Min et al., 2010] Min, B.-K., Marzelli, M. J., and Yoo, S.-S. (2010). Neuroimaging-based approaches in the brain–computer interface. *Trends in biotechnology*, 28:552–560.

[Montaldo et al., 2022] Montaldo, G., Urban, A., and Mace, E. (2022). Annual review of neuroscience functional ultrasound neuroimaging.

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

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

[Nicolas-Alonso and Gomez-Gil, 2012] Nicolas-Alonso, L. F. and Gomez-Gil, J. (2012). Brain computer interfaces, a review. *sensors*, 12:1211–1279.

[Nijholt et al., 2008] Nijholt, A., Tan, D., Allison, B., del R. Milan, J., and Graimann, B. (2008). *Brain-computer interfaces for HCI and games.*

[Norman et al., 2021] Norman, S. L., Maresca, D., Christopoulos, V. N., Griggs, W. S., Demene, C., Tanter, M., Shapiro, M. G., and Andersen, R. A. (2021). Single-trial decoding of movement intentions using functional ultrasound neuroimaging. *Neuron*, 109:1554–1566.e4.

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

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

[Paparrizos and Gravano, 2015] Paparrizos, J. and Gravano, L. (2015). k-shape: Efficient and accurate clustering of time series. pages 1855–1870.

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

[Pawar and Dhage, 2020] Pawar, D. and Dhage, S. (2020). Feature extraction methods for electroencephalography based brain-computer interface: A review. *IAENG International Journal of Computer Science*, 47.

[Pawuś and Paszkiel, 2022] Pawuś, D. and Paszkiel, S. (2022). Bci wheelchair control using expert system classifying eeg signals based on power spectrum estimation and nervous tics detection. *Applied Sciences*, 12:10385.

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

[Pedregosa et al., 2011] 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., and Duchesnay, E. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830.

[Poldrack, 2007] Poldrack, R. A. (2007). Region of interest analysis for fmri. *Social cognitive and affective neuroscience*, 2:67–70.

[Rabut et al., 2019] Rabut, C., Correia, M., Finel, V., Pezet, S., Pernot, M., Deffieux, T., and Tanter, M. (2019). 4d functional ultrasound imaging of whole-brain activity in rodents. *Nature Methods*, 16:994–997.

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

[Roy, 2022] 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 et al., 2015] Schlegel, F., Schroeter, A., and 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.

[Shashmi, 2021] Shashmi, K. (2021). Curse of dimensionality — a “curse” to machine learning.

[Sinaga and Yang, 2020] Sinaga, K. P. and Yang, M.-S. (2020). Unsupervised k-means clustering algorithm. *IEEE access*, 8:80716–80727.

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

[Sinha et al., 2010] Sinha, G., Shahi, R., and Shankar, M. (2010). Human computer interaction. *Proceedings - 3rd International Conference on Emerging Trends in Engineering and Technology, ICETET 2010*, pages 1–4.

[Soloukey et al., 2020] 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., and Zeeuw, C. I. D. (2020). Using functional ultrasound (fus) to map brain functionality and tumor vasculature with micrometer-millisecond precision. *Neuro-Oncology*, 22.

[Sorger and Goebel, 2020] Sorger, B. and Goebel, R. (2020). Real-time fmri for brain-computer interfacing. *Handbook of clinical neurology*, 168:289–302.

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

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

[Torres et al., 2020] Torres, E. P., Torres, E. A., Hernández-Álvarez, M., and Yoo, S. G. (2020). Eeg-based bci emotion recognition: A survey. *Sensors*, 20:5083.

[Urban et al., 2015] Urban, A., Dussaux, C., Martel, G., Brunner, C., Mace, E., and Montaldo, G. (2015). Real-time imaging of brain activity in freely moving rats using functional ultrasound. *Nature methods*, 12:873–878.

[Vasiljevic and de Miranda, 2020] Vasiljevic, G. A. M. and 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*, 36:105–142.

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

[Wang et al., 2021] Wang, Z., Cao, C., Zhou, Y., Chen, L., Gu, B., Liu, S., Xu, M., He, F., and Ming, D. (2021). Integrating eeg and nirs improves bci performance during motor imagery. pages 511–514. IEEE.

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

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

[Wolpaw et al., 2000] Wolpaw, J. R., McFarland, D. J., and Vaughan, T. M. (2000). Brain-computer interface research at the wadsworth center. *IEEE transactions on rehabilitation engineering*, 8:222–226.

[Xu et al., 2019] Xu, Y., Hua, J., Zhang, H., Hu, R., Huang, X., Liu, J., and 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 et al., 2014] 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., and Nurmikko, A. V. (2014). Wireless neurosensor for full-spectrum electrophysiology recordings during free behavior. *Neuron*, 84:1170–1182.

[Zhang, 2004] Zhang, H. (2004). The optimality of naive bayes. *Aa*, 1:3.

[Zhang, 2021] Zhang, Y. (2021). Invasive bci and noninvasive bci with vr/ar technology. volume 12153, pages 186–192. SPIE.

[Zhu et al., 2019] Zhu, X., Li, P., Li, C., Yao, D., Zhang, R., and 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 et al., 2009] Zickler, C., Donna, V. D., Kaiser, V., Al-Khodairy, A., Kleih, S., Kübler, A., Malavasi, M., Mattia, D., Mongardi, S., and 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.

[Émilie Mace et al., 2011] Émilie Mace, Montaldo, G., Cohen, I., Baulac, M., Fink, M., and Tanter, M. (2011). Functional ultrasound imaging of the brain. *Nature Methods*, 8:662–664.

[Émilie Mace et al., 2018] Émilie Mace, Montaldo, G., Trenholm, S., Cowan, C., Brignall, A., Urban, A., and Roska, B. (2018). Whole-brain functional ultrasound imaging reveals brain modules for visuomotor integration. *Neuron*, 100:1241–1251.e7.