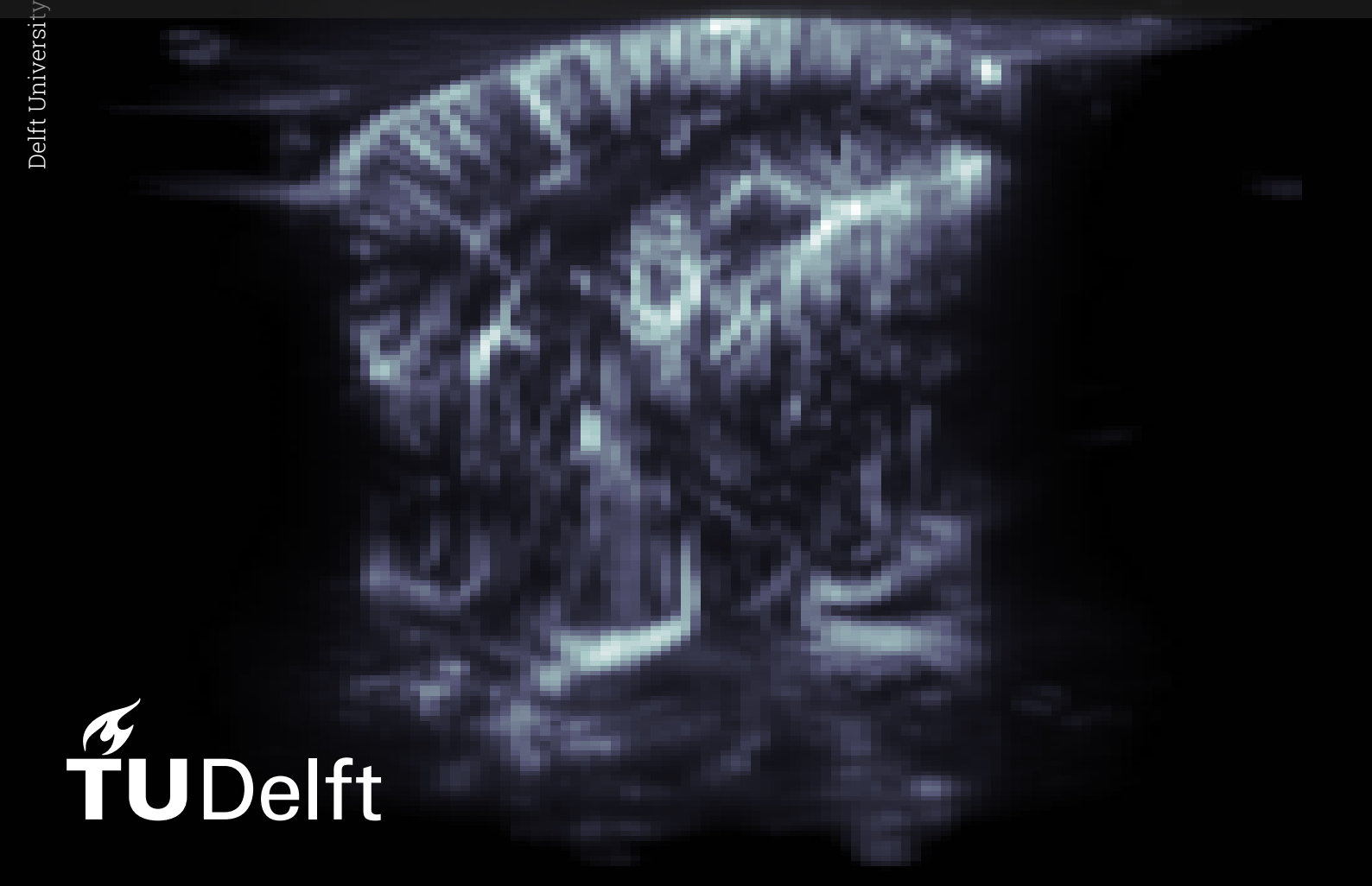
Identifying Neural Signatures of Learning in Functional Ultrasound via Tensor Decomposition

Assessing Robustness to Data Quality Issues in Functional Ultrasound

Kai Grotepass

Delft University of Technology



TU Delft

Identifying Neural Signatures of Learning in Functional Ultrasound via Tensor Decomposition

Thesis

submitted in partial fulfillment of the
requirements for the degree of

Master of Science

in

Electrical Engineering

by

Kai Grotepass

Supervisor: dr. B. Hunyadi
Faculty: Faculty of Electrical Engineering, Mathematics and Computer Science

TU Delft

Delft University of Technology
Department of
Microelectronics

The undersigned hereby certify that they have read and recommend to the Faculty of Electrical Engineering, Mathematics and Computer Science for acceptance a thesis entitled **“Identifying Neural Signatures of Learning in Functional Ultrasound via Tensor Decomposition”** by **Kai Grotepass** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: 20 August 2025

Chairman: _____
dr. B. Hunyadi

Advisor: _____
dr.ir. P. Kruizinga

Committee Members: _____
dr.ir. M. Jafarian

Abstract

Understanding the mechanisms that underpin learning effects remains a central goal in neuroscience, with functional neuroimaging offering a powerful avenue for observing brain activity. This work is situated within the domain of haemodynamic functional neuroimaging, with a specific focus on functional ultrasound (fUS)—a relatively novel modality that combines high spatio-temporal resolution and portability. The goal of this project is to investigate the capacity of tensor-decomposition-based techniques to extract meaningful, interpretable representations of stimulus-evoked learning effects from a multi-subject fUS dataset while being robust to the non-idealities present in said data. While numerous approaches exist for analysing neuroimaging data, many are limited by scalability, interpretability, or an inability to capture changing neurological patterns. This work is thus motivated by the need for methods capable of handling large, dynamic datasets while extracting interpretable results. Tensor decompositions offer such a framework but their ability to capture subject-specific time-varying effects and their application to functional ultrasound is still under-explored. Several tensor decomposition algorithms are examined, assessed in terms of their learning effect extraction capabilities and robustness to non-idealities, and a novel shifted canonical polyadic decomposition variant is developed to address time-varying effects. Results from synthetic data analysis demonstrate that tensor decompositions are robust to non-idealities in functional ultrasound data (to varying degrees) and can recover interpretable latent components. Due to the data's partial scan coverage, no novel signatures of learning were identified in the real data; however, the extracted components further substantiate the potential of these algorithms and highlight a promising direction for analysing learning effects in functional neuroimaging data.

Acknowledgements

This thesis has been a journey hallmarked by unexpected twists and turns, often poor visibility, and a great deal of learning (for both myself and the mice). In many ways I owe this experience – and the culmination of this work – to my supervisor, Bori Hunyadi, who gave me the freedom to explore in whatever direction I chose from our very first meeting, unfaltering support of my decisions, and the guidance needed to overcome the obstacles encountered along the path. Bori, for the paper suggestions, the advice, the meetings that ran over lunch, the time taken during the holiday, the faith you placed in me, and more, thank you, I am deeply grateful.

To Pieter Kruizinga, you may not know this, but your array processing guest lecture is what inspired me to choose to work on functional ultrasound in the first place – in a way you were part of this project before it even began. Thank you for inspiring me, for your counsel, for agreeing to be a part of my committee, and for always having time to discuss any conundrums I encountered despite your own obligations. Without you I would likely still be waving my hands around in the proverbial dark forest of functional ultrasound.

I would like to thank the third committee member, Matin Jafarian, for her interest in my project and her willingness to be present at my defence during the holiday. I have not yet had the honour of discussing my project with you directly, so I hope you find the fruit of my metaphorical travels interesting if nothing else.

Thank you to the bio-detail subgroup, the presentations were always thought-provoking, and specifically to Sofia, our last discussion is what sparked me to take the leap and walk the path that led me here; looking back over my shoulder I can still see the fork in the road and am grateful for where I stand.

To all my friends in SPS: you had your own journeys, but oft we walked the same roads, and my journey was better for it. For the company, the counsel, and the conversation, you have my heartfelt gratitude. I shall not name you lest I forget someone and shame myself, but I hope that all my fellow travellers from the past two years know this: you will not be soon forgotten, and my best wishes travel with you wherever your roads lead.

For Çağla, who stood beside me, got lost and found with me, travelled high and low with me, thank you for choosing to travel with me, it would have been a far harder path had I taken it alone.

From lands far and wide, the support and faith of my family over the last 25 years are what has made me who I am and allowed me to get here. Looking back there has never been a time where you did not offer a helping hand, a piece of advice or a word of encouragement when I needed it. You are the giant's shoulders upon which I stand, the wind in my sails, and the light that banishes the darkness.

Most of all to my parents – you are the ones who taught me to explore, to question, to trust myself. For all that I have learned from you, all that I will learn, and for always having time for me – thank you.

Kai Grotepass
Delft, The Netherlands
August 2025

Contents

1

Introduction

The vehicle of consciousness – the brain – has been the subject of interest and study for centuries[1]. Modern technology has brought this study forward in leaps and bounds in the form of sophisticated neuroimaging techniques, but just as these novel imaging modalities offer new opportunities, they also present new challenges. Although these techniques can illuminate in great detail **what** is happening in the brains of living subjects, there is still a great need to process and decipher this information to shed light on the underlying mechanisms.

There are a variety of neuroimaging techniques that provide researchers, neurologists, and medical practitioners invaluable insights into the brain, each technique with its own advantages and disadvantages; many novel neuroimaging techniques are still being researched to this day [34]. Within the realm of neuroimaging there are several general types, including structural, connectivity, and, of particular interest, functional. Functional neuroimaging refers to techniques that measure the neuronal activity within the brain, often non-invasively on awake patients [14]. These characteristics offer a unique lens through which to study the actual mechanisms of thought and behaviour directly through recordings as they happen. Common challenges in processing functional neuroimaging data include large data volumes, temporal and spatial alignment between recordings or patients, and the general signal processing problem of distinguishing which signals are noise/artefacts or of actual interest.

Functional neuroimaging can be broadly divided into two main categories, based on whether the techniques measure the electrical or haemodynamic signals [14]. Electric functional neuroimaging techniques such as electroencephalography (EEG) generally offer very high temporal resolutions (hundreds to thousands of Hz [18]), but have much poorer spatial resolution. The other type is haemodynamic functional neuroimaging techniques, like magnetic resonance imaging (fMRI), which are conversely characterised by good spatial resolution but relatively poor temporal resolution (0.5-1 Hz). Based on their characteristics, these two functional imaging types naturally lend themselves to different problems, with haemodynamic techniques used to identify regions associated with specific brain functions, while electrical methods are more oriented towards the characterisation of temporal dynamics related to sensory, motor, and cognitive function.

[1] Leonardo da Vinci's dissections of the human brain were performed more than 500 years ago [29].

This work is directed toward the former of these two problem types, and unsurprisingly is thus also focused on a functional haemodynamic neuroimaging modality. The data in question involves an experiment where mice are subjected to sequences of stimuli, and are expected to learn a specific response over time. To capture their neurological responses the mice are recorded with an ultrasound probe which (after some processing) produces functional ultrasound (fUS) data. This functional modality has a few advantages over the ubiquitous fMRI. By using high-frequency acoustic signals fUS is able to map changes in brain blood volume at speeds up to 4 Hz [20], a modest improvement over conventional fMRI, while retaining good spatial resolution. Beyond this, it is an inexpensive and highly portable technology, which makes it well suited to task-based experimental setups, as mice do not need to be sedated during scanning. The specific problem of interest is more complex than simple region activation though, as the mice are expected to learn over time and the hope is to identify the neurological signatures associated with these learning effects.

This experimental setup adds some additional challenges beyond just the standard neuroimaging data processing issues. These challenges include: the dynamic nature of the effects of interest (meaning that the analysis needs to capture temporal changes), the multi-subject nature of the data, and fUS motion artefacts introduced by awake subject activity and stimulus responses. The data analysis methods used to process the data need to account for these various issues.

There are a variety of different algorithms to isolate effects in neuroimaging data, but many of them have limitations or assumptions which make them ill-suited to this task. Many machine learning algorithms suffer from lack of interpretability and dependence on training data, while more general techniques which are sometimes employed to extract regions in haemodynamic neuroimaging data such as generalised linear models, graph theory analysis or manifold learning either struggle to capture the time-varying effects of interest or cannot handle the large data volumes well. The most promising family of analysis tools is matrix and tensor based decompositions. They are able to extract meaningful latent components from high order and large volume data. There are however a variety of different options within this family of analysis approaches which rely on different assumptions.

With this context the research question of this work is thus:

How well can tensor decomposition based algorithms extract meaningful stimulus-evoked learning signatures from multi-subject functional ultrasound recordings in terms of component interpretability?

With sub questions

1. *Can tensor decompositions capture dynamic learning effects in a meaningful way?*
2. *How robust are tensor decompositions to non-idealities in fUS data?*
3. *Which stimulus evoked learning effects can tensor decompositions effectively identify?*

The structure is as follows. A discussion regarding processing techniques for neuroimaging from literature elaborates on the pros and cons of various methods, and why tensor decompositions are particularly promising for this task. Necessary background follows and covers the underlying mechanisms of fUS as well as the model

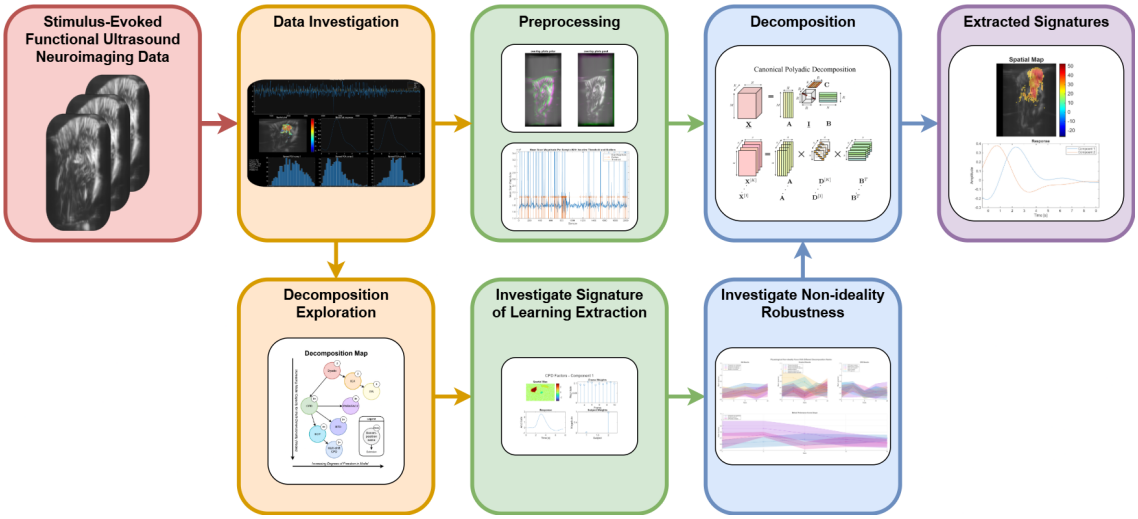


Figure 1.1: Project pipeline overview depicting the broad steps taken to extract meaningful components from functional ultrasound data using tensor decompositions.

definitions of selected tensor decompositions to provide a sufficient foundation to build the methodology thereafter. The body of the work starts with a description of the experimental data, the non-idealities present and possible learning effect structures. Several algorithms are then defined and the shifted CPD algorithm is developed, followed by dynamic effect representation tests. Based on the identified non-idealities a synthetic data generation model mimicking fUS is defined and used to analyse the non-ideality robustness of different algorithms. Finally the best performing algorithms are applied to the real data, the results are discussed and conclusions are drawn with some final notes on limitations and future work to be done. The broad steps of this structure are shown in figure 1.1.

2

Related Work

Advances in neuroimaging techniques such as functional MRI (fMRI) and functional ultrasound imaging (fUS) offer unique opportunities to gain insight into the underlying mechanisms of learning, but there are several challenges associated with processing neuroimaging data from these modalities, as well as isolating these effects. The focus of this work is the analysis of fUS data to identify learning signatures evoked by stimuli. However, as learning progresses the functional associations are expected to change, making them more challenging to isolate. Furthermore, meaningful responses can be masked by unrelated neural activations or other measurement artefacts, potentially obfuscating instances of learning effects [26]. To make matters worse, fUS data generally has both high data order (multi-dimensional data with, e.g. space $\times$ time $\times$ recording $\times$ subject) and volume, making data analysis far more challenging, as the algorithms need to be sufficiently performant to make analysis worthwhile.

While fUS is a comparatively new modality, a variety of different analysis techniques have been employed to analyse other neuroimaging data, but they all have their own trade-offs, and many are not applicable to this task. The available techniques can be loosely categorized based on whether they are data-driven or hypothesis-driven and data-driven further grouped on if they are based on a static data model or more complex dynamic model of neurological states [11].

Hypothesis-Based Algorithms

In this setting little to no a-priori assumptions regarding the learning signatures are made, which means that almost all hypothesis-based algorithms are not applicable. To elaborate, an example of such an algorithm would be atlas-based region-of-interest extraction [27] which, in this application, would isolate specific regions where learning would be expected to take place. It would then extract estimates of the signals in these regions from the data that could then be inspected. This would require knowledge of the specific regions where learning is expected to occur and would also rely extremely heavily on effective atlas registration, which is a particular challenge in fUS due to lack of anatomical landmarks (see appendix D for more details on registration of the dataset of interest).

Deep Learning Algorithms

Moving to data-driven algorithms, deep neural networks have become very popular in recent years thanks to their ability to learn complex non-linear mappings and have been applied to great effect in the neuroimaging setting [30], but there are challenges and issues with both supervised and unsupervised deep learning options for this application. Firstly, these algorithms typically require large amounts of data and supervised options need annotated data. Having annotated data would defeat the point of identifying signatures of learning, as then these signatures could already be characterised and hypothesis-driven algorithms applied. For unsupervised options the challenge is to frame the problem, one option would be to apply an unsupervised clustering algorithm to identify subjects where learning does and does not occur, but as there are several other differences one could expect between subjects in such a dataset this would likely not identify the desired clusters. Additionally, these models often produce results which are not easily interpreted, which in combination with the other limitations means that these algorithms are not well suited to the task of identifying unknown learning signatures.

Data-driven Static Methods

On the topic of unsupervised clustering, more conventional clustering algorithms which do not rely on deep learning could be an option. Methods like k-means can be used to extract various types of groups within the fUS data. By clustering voxels one could identify unique regions (parcellation), or by clustering on a subject level one could separate the subjects into those exhibiting learning and those not. Aside from the aforementioned issue of subject level data likely including other differentiating qualities and leading to unhelpful clusters these clusters would also lack the structure to capture patterns that exhibit the dynamic behaviour expected from the task-evoked learning setting and additional processing would be needed. Manifold learning techniques like UMAP [22] and t-SNE [5] can be used on a subject level to visualize similarity between subjects in a lower dimension and thus identify which subjects exhibit learning effects, but have the same issues mentioned and further require some degree of consistent structure in feature vectors. Parcellation is an option, but comes with the issue of having a discrete region associated with an entire time-course, making interpretation challenging as the time courses likely have various other artefacts and unwanted signals, and also makes comparisons between subjects challenging as the regions identified may differ.

Data-driven Dynamic Methods

Moving to the dynamic setting it is possible to use the unsupervised data driven clustering techniques on different images in temporal windows to identify unique brain states and then model the state changes statistically (though this is often done after parcellation). Hidden Markov Models (HMMs) [15] or other techniques [11] can then be used to model the temporal evolution of these neural states. This relies on the state identification step, the possible parcellation, on the identified states being comparable between subjects, and generally struggles to manage the multi-dimensional nature of the data in a meaningful way. It is also challenging to capture the signatures of learning in this manner as brain states will change as learning occurs.

Decomposition Based methods

Finally, there are a variety of matrix and tensor decomposition based approaches which identify the underlying components comprising the data. Matrix techniques like ICA [23] are very effective at dealing with large data volumes, and identify easily interpretable latent components. Matrix decompositions are very similar to the parcellation methods described prior, with the added benefits of non-binary region allocation and inherent signal separation into different components, both greatly enhancing interpretability. The one limit of these algorithms is that they operate only on matrices so are not ideally suited to the multi-dimensional nature of the fUS dataset in question. Conveniently these benefits are retained when moving to higher order tensor decompositions like IVA [2], CPD, coupled CPD (CCPD) [17], or convCP [24], while allowing for additional dimensions in the data and still extracting meaningful and interpretable latent components.

Summary

In summary, analysing stimulus-evoked learning through neuroimaging requires techniques that can handle multi-dimensional, large-volume, temporally evolving data without relying heavily on prior information. While many techniques have not been applied to fUS due to its relative recency, approaches developed in the context of fMRI and other neuroimaging settings still have significant promise thanks to the similarities to fUS. Many machine learning based approaches lack interpretability and require large amounts of training data and time, making them unsuited to this task. The only techniques discussed here which offer interpretability, efficiency on large data volumes, and the ability to meaningfully incorporate multi-dimensional data are tensor decompositions. This work explores and expands on several tensor decompositions with the aim of identifying stimulus-evoked learning signatures from fUS data.

3

Background

3.1. Conventions

This section elaborates on several notational, nomenclatural, and variable meanings used throughout this work in an attempt to reduce ambiguity and help orient the reader in the context. Starting with notation, several mathematical operations are either specific to the tensor context, or have symbols which have other meanings in other contexts, so these are all described in table 3.1.

Table 3.1: Symbols in tensor and matrix operations.

Symbol	Operation Description
$\cdot$	Scalar multiplication
$*$	Convolution
$\circ$	Outer product of vectors or tensors
$\otimes$	Kronecker product
$\odot$	Column-wise Khatri-Rao product
$\circledast$	Hadamard product
$\times_n$	Mode-n product of a tensor with a matrix

Some terms used to describe parts of a three-way tensor are shown in figure 3.1. The term ”slices” is shared with the neuroimaging context where slices of the brain are imaged, so there is some ambiguity. To clarify at least one point on this topic: in this work the term **frontal slices** always refers to tensor anatomy not biological anatomy or scan location.

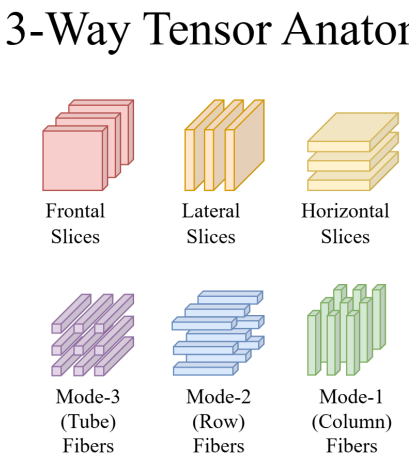


Figure 3.1: Terminology for anatomy of a three-way Tensor.

In this context the terms ”way”, ”order”, and ”dimension” are largely interchangeable, referring to the number of dimensions in a tensor (for example a three-way tensor, third-order tensor and a three-dimensional tensor are the same thing). Unfolding refers to collapsing one dimension into another (as depicted visually in figure 3.2) thereby reducing the overall order of the tensor.

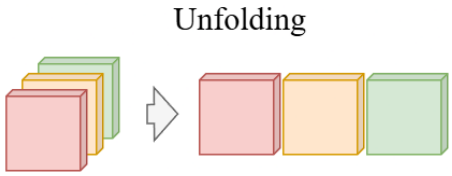


Figure 3.2: Depiction of a third-order tensor unfolded into a second order tensor (matrix).

For tensor decompositions a ”component” refers to a single (typically rank one) outer product of factors, a ”factor” refers to one vector in such a component, ”factor matrix” refers to all factors associated with one dimension and ”rank” refers to the number of components extracted. These terms are also described for a mode-two rank-two decomposition in figure 3.3, and in this setting a factor is generally a time-course, a spatial map, or a vector of weights, for a specific component.

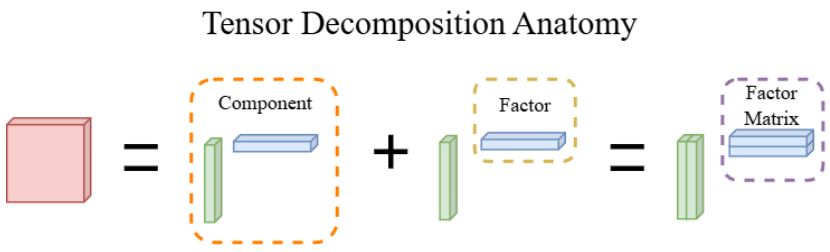


Figure 3.3: Tensor decomposition terms.

Table 3.2 shows the common variables used throughout this work. When developing tensor decompositions M, N, and K are used as general dimensional placeholders. These generally align with the use of these variables for the data, however there are

instances where transposes or permutations are made to explicitly enforce certain structures on the data. These values are scalar and refer to total dimensional sizes, and in most cases the lowercase letter is used to refer to a specific value in that dimension (i.e. $1 \leq n \leq N$)

Table 3.2: Notation for Variables Used in this work.

Symbol	Description
X	First spatial dimension (Dorsal–Ventral)
Y	Second spatial dimension (Rostral–Caudal)
Z	Slice index (Medial–Lateral)
M	Voxels in a recording (typically $X \times Y$)
N	Time points in a recording
N_f	Time points per frame
F	Frames
K	Recordings
P	Subjects

Tensor, matrix, and vector notation is described in table 3.3 while tensor element indexing notation is described in table 3.4. When indexing, the selected convention is that the dimension order will be retained with all collapsed dimensions (size-one dimensions) removed, i.e. a vector extracted with indexing will always be a column vector unless explicitly permuted or transposed.

Table 3.3: Notation for scalars, vectors, matrices and tensors.

Notation	Description
x	Scalar
$\mathbf{x}$	Vector
$\mathbf{X}$	Matrix
$\underline{\mathbf{X}}$	Tensor

Table 3.4: Tensor indexing notation.

Notation	Description
$\underline{\mathbf{X}}_{(i,j,k)}$	Scalar element at position (i, j, k)
$\underline{\mathbf{X}}_{(:,j,k)}$	Column fibre at position (j, k)
$\underline{\mathbf{X}}_{(:,:,k)}$	Frontal slice at position (k)

3.2. Functional Ultrasound

Functional ultrasound is a neuroimaging technique which has gained traction over the last decade since its first appearance in literature in 2011 [20]. This technology uses ultrasonic plane-wave illumination to measure blood volume in neurovasculature at higher spatio-temporal resolution than many other neuroimaging techniques. The scanning depth of fUS means that it can be used to get full depth planar scans of human (and by implication rodent) brains. Ultrasound probes are very portable, and the primary drawbacks of this technology are that the ultrasound waves generally cannot penetrate through bone [8] and that it has an indirect measure for neurological activation in the form of blood volume.

There are three main steps for fUS data collection beyond the planar scanning itself: compound image creation, filtering, and Doppler intensity extraction. First a number

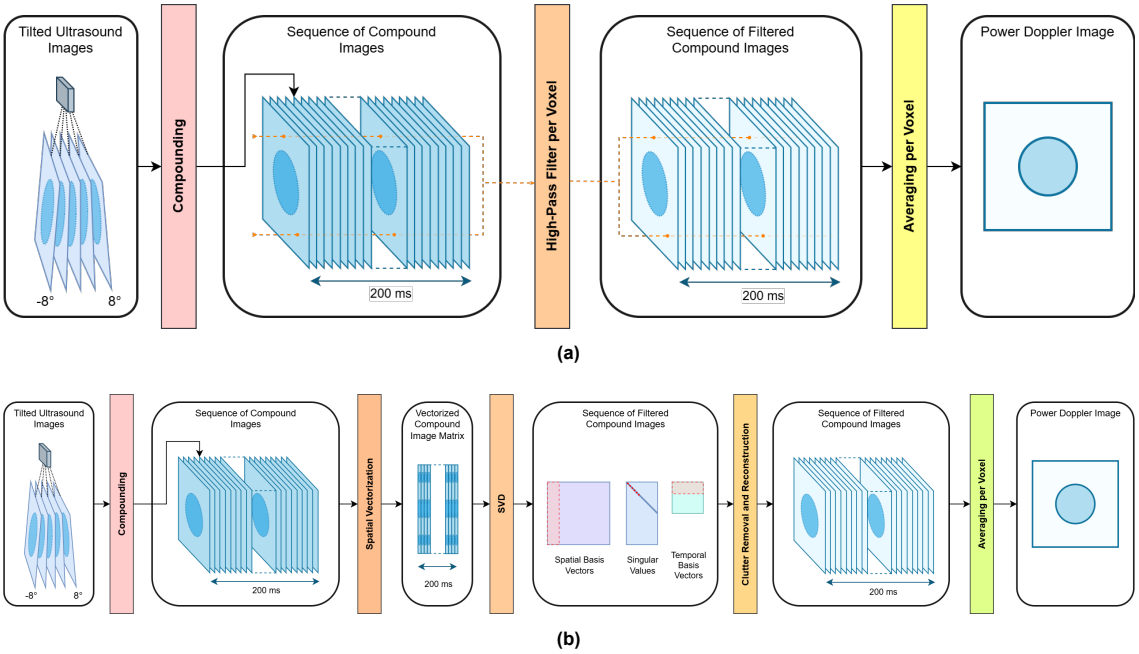


Figure 3.4: A diagrammatic representation of the basic steps of the fUS algorithm with two different clutter removal pipelines. Figure a) shows the originally proposed high-pass filter based clutter removal, while b) shows the singular value decomposition based method which showcases improved preservation of smaller blood vessels.

of images are obtained from several plane wave emissions tilted at slightly different angles (in the original paper, [20], angles of $-8°$ to $8°$ with a $1°$ increment were used). These 17 images are then added together to produce a compound image once every millisecond (effectively 1kHz). Ultrasound probes measure the reflections of the emitted waves created by scatterers meaning that these measurements are affected by Doppler effects if a scatterer is moving. This phenomenon is what allows fUS to measure blood volume as red blood cells move, but also introduces undesirable clutter effects as a result of tissue motion from respiration, cardiac cycles, subject movement, and probe motion. Tissue motion is slower than the motion of most blood cells and can be removed by applying a high-pass filter to each voxel within a fUS sequence of typically 200 compound images, as shown in figure 3.4a. Unfortunately the low-pass filtering method also removes the slow-moving capillary flow measurements, so alternatively a singular value decomposition (SVD) based clutter filter [10] can be used. The SVD is applied to a matrix of spatially vectorised and temporally stacked compound images to extract orthogonal constituent components. The high spatial coherence of tissue based clutter effects means that their decomposed components have high energy compared to the much more localised vasculature effects, and in turn that these components can be isolated in the SVD based on the magnitude of the diagonal singular values and removed from the data. The outline of this clutter removal is depicted with the other steps in figure 3.4b. A frequency spectrum per voxel can be obtained from the filtered fUS sequence and analysed to identify the blood cell velocity in that voxel, but generally in fUS each voxel is instead averaged over the 200 filtered compound images to create an intensity map. This intensity map is known as the power Doppler image (PDI) and is proportional to the blood volume.

It is worth noting that the blood volume associated with the PDI intensity is an indirect measure of neurological activity caused by neurovascular coupling. As such it is im-

portant to recognise that a PDI response is both delayed from underlying neurological activity, and may be influenced by any number of factors affecting the cardiovascular system which are unrelated to neurological activity. An approximation of the stages of a stimulus-response model from [7] are shown in figure 3.5. This model highlights the challenge of identifying the underlying neurological response from a blood volume signal when neither the perfusion response nor the haemodynamic response function (HRF) are known, as estimating both responses from data is not unique thanks to uncertainties such as time delays. In practice, unless one has access to neurological activity measurements like firing rate or the true perfusion response or HRF, it is common to simply estimate a single response function by viewing the stimulus signal as an approximation of the underlying neural activity. For the purposes of simplicity the term haemodynamic response function or HRF is often used to refer to a combined response from stimulus to the final haemodynamic response; this practice is adopted henceforth.

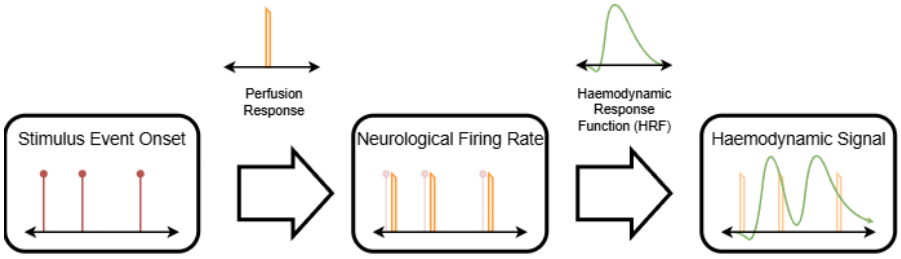


Figure 3.5: Haemodynamic response model based on a decaying perfusion response and a convolution with a haemodynamic response function (HRF) based on [7].

3.3. Decompositions

Matrix and tensor decompositions underpin almost all the algorithms discussed in this work and are built up in this section. The first step is describing the outline of the family of dyadic (or matrix) decompositions, which are not used directly, but lay a foundation to be extended with additional constraints and higher dimensions. Independent component analysis (ICA) is a specific dyadic decomposition which forms the basis of a three-dimensional independent vector analysis (IVA). Unlike IVA, which is a three-dimensional decomposition but interprets data as a series of linked dyadic decompositions, the canonical polyadic decomposition (CPD) is a more general extension of the dyadic formulation to higher order tensors. The CPD is also known as parallel factorization (PARAFAC), and an extension thereof called parallel factorization 2 (PARAFAC2) allows for more freedom. Finally there are some notes on the block term decomposition (BTD) and a variant of CPD which allows for shifts in one mode called, unsurprisingly, the shifted canonical decomposition (SCP). A visual map of these decompositions (as well as the multi-shift CPD developed later in this work) and how they relate is included in figure 3.6.

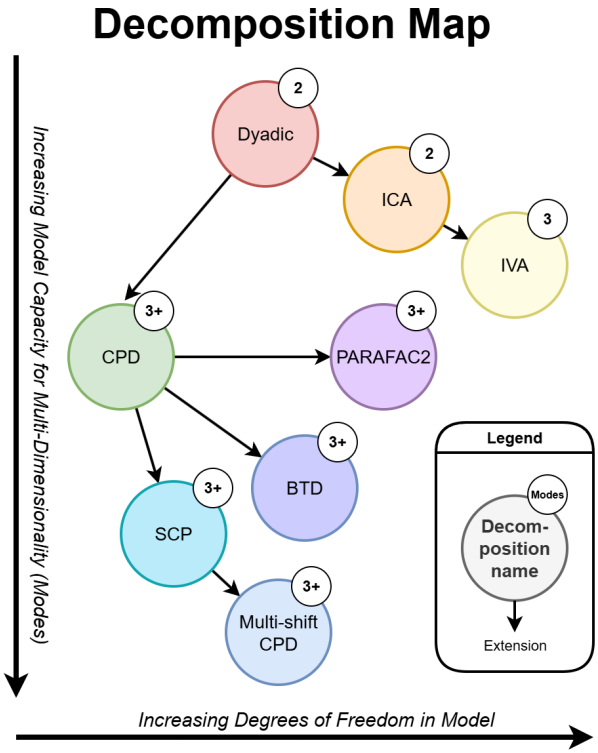


Figure 3.6: Decomposition map showing relationships, number of modes and relative degrees of freedom.

3.3.1. Dyadic Decompositions

The dyadic decomposition is based on a simple premise of breaking a matrix into constituent parts to facilitate interpretation or operations thereon. It can be simply framed for a matrix $\mathbf{X} \in \mathbb{R}^{M \times N}$ as a matrix product or summation of R rank-one vector products

$$\begin{aligned} \mathbf{X} &= \mathbf{A}\mathbf{B}^T \\ \mathbf{X} &= \sum_{r=1}^{R} \mathbf{A}_{(:,r)} \circ \mathbf{B}_{(:,r)}. \end{aligned} \tag{3.1}$$

A simple graphical representation of this is depicted in figure 3.7. This decomposition often also includes a diagonal weight matrix between $\mathbf{A} \in \mathbb{R}^{M \times R}$ and $\mathbf{B} \in \mathbb{R}^{N \times R}$ (for example in the SVD), but this is neglected in this work with the weights instead being absorbed into the respective component vectors. Looking at equation (3.1) it is clear that the decomposition is entirely non-unique, the natural remedy to which is additional constraints. These constraints could be something like orthogonality, as in the case of SVD, or statistical independence between components as in ICA.

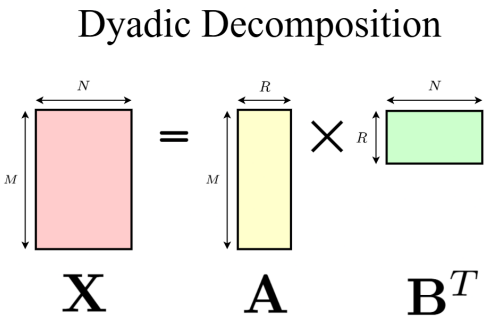


Figure 3.7: Graphical depiction of the dyadic decomposition.

3.3.2. ICA

ICA [33] is a dyadic decomposition which imposes statistical independence assumptions on the second factor matrix. By treating each component vector in the second factor matrix $\mathbf{B}$ as several realisations of a random variable, ICA can maximize the statistical independence between the components in $\mathbf{B}$. Each row of $\mathbf{X}$ is thus composed of a linear mix of multiple independent sources, and when framed in this way the decomposition is also known as an unmixing problem. This model is depicted graphically in figure 3.8.

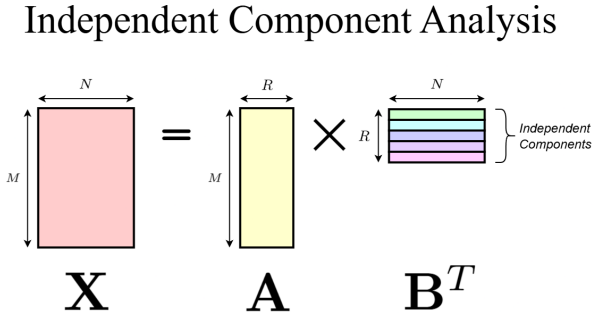


Figure 3.8: Graphical depiction of ICA's data model.

Since neither $\mathbf{A}$ nor $\mathbf{B}$ are known, ICA solves for a demixing matrix $\mathbf{W} \in \mathbb{R}^{R \times M}$ to extract independent sources. The demixed sources $\hat{\mathbf{B}} \in \mathbb{R}^{N \times R}$ are obtained from

$$\hat{\mathbf{B}}^T = \mathbf{W}\mathbf{X}. \tag{3.2}$$

By maximizing the independence of the sources (columns) of $\hat{\mathbf{B}}$ ICA aims for

$$\mathbf{W} = \mathbf{A}^{-1} \text{ (or equivalently } \mathbf{W}^{-1} = \mathbf{A}), \tag{3.3}$$

so that ideally when substituted into equation (3.2) with equation (3.1)

$$\hat{\mathbf{B}}^T = \mathbf{W}\mathbf{X} = \mathbf{W}\mathbf{A}\mathbf{B}^T = \mathbf{A}^{-1}\mathbf{A}\mathbf{B}^T = \mathbf{B}^T, \tag{3.4}$$

assuming the independence assumption matches the underlying data structure of $\mathbf{X}$. There are various approaches to maximize the independence of sources and estimate

$\mathbf{W}$, such as minimizing mutual information, maximizing non-Gaussianity or using maximum likelihood estimates, some details of which may be found in [33].

There are a few nuances to be addressed regarding the assumptions and ambiguities of ICA. Firstly, thanks to both $\mathbf{A}$ and $\mathbf{B}$ being unknown there are obvious sign and permutation ambiguities in the extracted demixing matrix $\mathbf{W}$. More importantly equation (3.3) is only possible if $\mathbf{W}^{-1}$ exists, which is only possible if $\mathbf{W}$ is square ($R = M$). This is of course not a reasonable assumption, so to apply ICA, dimension reduction in the form of principal component analysis (PCA) is first applied to the measurement matrix $\mathbf{X}$ such that M is reduced to the number of desired sources R. This is for the over-defined case since the under-defined case is not relevant to this work. As a result the number of independent components to extract needs to be known a-priori or selected by the user (though, approaches do exist to estimate the number of components from the data).

3.3.3. IVA

As is, ICA can only be applied to matrices, not tensors, unless the tensors are first restructured. For example, in the case of three-dimensional data ($\underline{\mathbf{X}} \in \mathbb{R}^{M \times N \times K}$) this limitation can be circumvented by unfolding the third mode along one of the first two modes. However, if unfolding along the second mode this implicitly assumes that all tensor frontal slices share the same mode-one mixing matrices and that the same random variables (RVs) are responsible for the components in each tensor frontal slice. If the data tensor is of the shape $(\mathbb{R}^{\text{space} \times \text{time} \times \text{recording}})$, then this unfolding assumes that all recordings have the same spatial maps, and the same RVs generating the time-courses. Conversely, if unfolded along the first mode it implies that all frontal slices are a mix of exactly the same components. Alternatively, without unfolding, one could frame it as separate dyadic decompositions for each frontal slice k where $1 \leq k \leq K$:

$$\mathbf{X}^{[k]} = \mathbf{A}^{[k]}(\mathbf{B}^{[k]})^T. \tag{3.5}$$

This is effectively just K separate ICA decompositions, and is a viable approach, with some caveats. Unfortunately the permutation ambiguity between sources discussed prior makes the comparison between recordings challenging as an additional step to group similar sources between recordings is needed. An added challenge to this grouping is that there is no guarantee that the different applications of ICA will identify sources that are at all related.

Independent vector analysis (IVA), first introduced in [16], addresses these issues by extending ICA to a three-dimensional setting by jointly solving for sources in multiple frontal slices of a tensor simultaneously. By treating the sources as being generated by multivariate RVs, instead of univariate RVs as in ICA, this model is able to capture both independence between components in a frontal slice, and dependences in components across frontal slices. These multivariate sources are each represented as a random vector known as the source component vector (SCV). The r'th SCV is defined as:

$$\mathbf{b}_r = \left[b_r^{[1]}, ..., b_r^{[K]}\right]^T. \tag{3.6}$$

To be clear, $b_r^{[k]}$ is a random variable for the r'th source of the k'th frontal slice while $b_r^{[k]}[n]$ would be a single sample instance of this random variable and $\mathbf{b}_r$ is a random

vector of the r'th source for all K recordings. All the instances of a single SCV may also be captured in a matrix as

$$\mathbf{B}_r = \begin{bmatrix} b_r^{[1]}[1] & \cdots & b_r^{[1]}[N] \\ \vdots & \ddots & \vdots \\ b_r^{[K]}[1] & \cdots & b_r^{[K]}[N] \end{bmatrix} \tag{3.7}$$

Independent vector analysis estimates a demixing matrix $\mathbf{W}^{[k]}$ for each recording by enforcing independence between every SCV, while maximizing the dependence between the random variables that compose the SCV. This approach automatically solves the problem of permutation ambiguity as sources are already grouped together in an SCV. It is also able to exploit information between different recordings to improve source estimates while still having no hard restriction on the differences in demixing matrices and extracted sources between recordings.

The data model is shown graphically in figure 3.9, where one can see that it is similar to multiple ICA's, but that there is an added dependence across frontal slices, as depicted by the component colours.

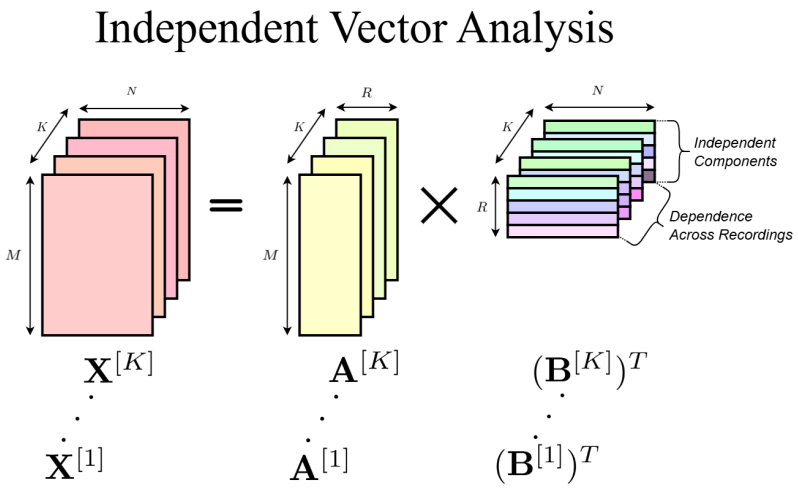


Figure 3.9: Graphical depiction of IVA's data model.

3.3.4. CPD

The canonical polyadic decomposition (CPD) [3] is the natural extension of the dyadic decomposition to higher dimensions. Much like a dyadic decomposition, CPD decomposes a tensor into R rank-one tensors. While it can be trivially extended to higher orders, the third-order version of this model using outer products is

$$\underline{\mathbf{X}} = \sum_{r=1}^{R} \mathbf{A}_{(:,r)} \circ \mathbf{B}_{(:,r)} \circ \mathbf{C}_{(:,r)}, \tag{3.8}$$

and the similarity to equation (3.1) is immediately clear. As before $\underline{\mathbf{X}} \in \mathbb{R}^{M \times N \times K}$ and the factor matrices are $\mathbf{A} \in \mathbb{R}^{M \times R}$ and $\mathbf{B} \in \mathbb{R}^{N \times R}$ with the added $\mathbf{C} \in \mathbb{R}^{K \times R}$. Two equivalent depictions of this model are shown in figure 3.10 for the third-order case, as higher orders are challenging to visualise.

For the third-order case this formulation can be rewritten in a frontal slice form as

$$\mathbf{X}^{[k]} = \mathbf{A}\mathbf{D}^{[k]}\mathbf{B}^T, \tag{3.9}$$

where $\mathbf{D}^{[k]} \in \mathbb{R}^{R\times R}$ is a matrix whose diagonal elements are the k'th row of the third-mode factor matrix $\mathbf{C}$. In this form the relation to IVA is clear and shows that even when applied to a three-dimensional tensor and formulated in the same way this decomposition has some stark differences to the IVA algorithm. There are far fewer degrees of freedom in the CPD model than in IVA; using the same terminology as in IVA, there is only a single mixing matrix $\mathbf{A}$ and a single source matrix $\mathbf{B}$ for all frontal slices, with these frontal slices able to vary only in terms of the magnitude of each component per frontal slice. This is in contrast to IVA, which can have an entirely arbitrary mixing matrix for each frontal slice, and while there are dependence constraints on the source matrices, they can in theory also vary significantly between different frontal slices. As an important note, the uniqueness issues that plague the dyadic decomposition are not an issue for the CPD (under mild conditions), and the imposed structure allows for more interpretable and informative components to be extracted, assuming that the data actually exhibits the assumed structure.

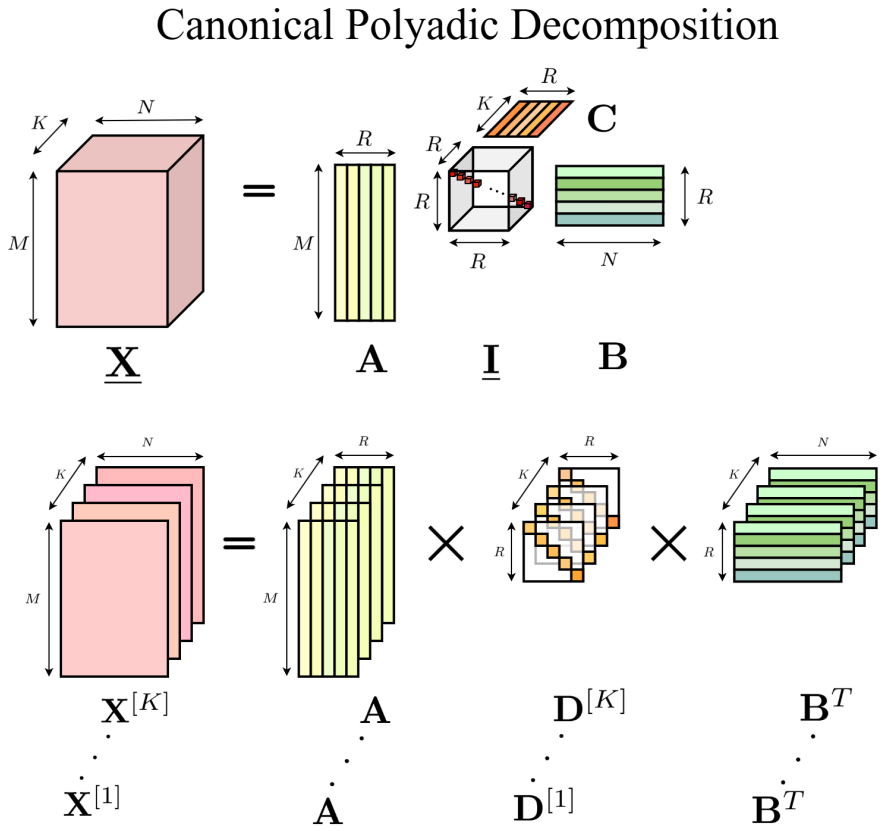


Figure 3.10: Two graphical depictions of CPD's data model in both a standard three-way tensor format (top) and in a flattened matrix multiplication format (bottom) illustrating the relation to the PARAFAC2 and IVA decompositions.

3.3.5. PARAFAC2

Parallel factorization 2 (PARAFAC2) [13] is an extension of CPD which allows for the factor matrix in one of the modes to vary for the different values in another mode, offering an intermediate option with more freedom than CPD, but with more structure

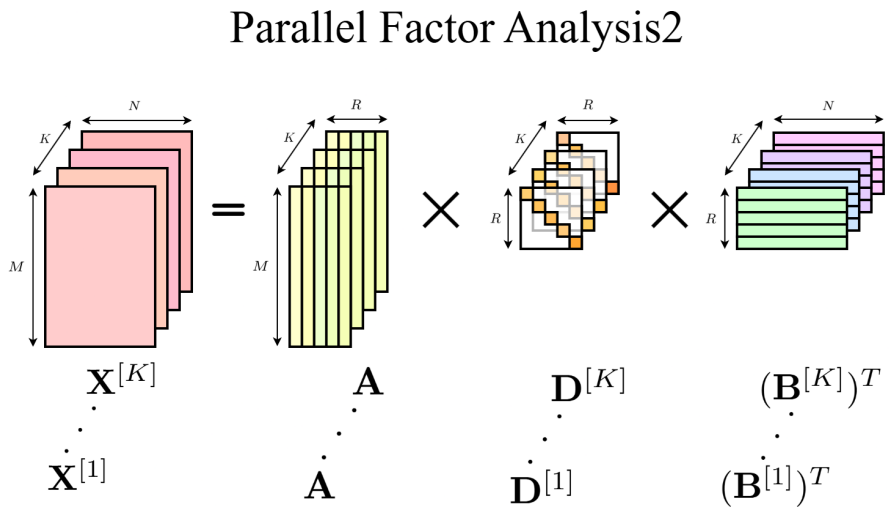


Figure 3.11: Graphical depiction of PARAFAC2's data model.

than IVA. Only the mode-three tensor version of PARAFAC2 is used in this work, and in this setting it is typical to have the second mode factor matrix vary along the third mode [35]. This structure is reminiscent of the model in equation (3.5) employed by IVA and the CPD model in equation (3.9), as a data tensor $\underline{\mathbf{X}} \in \mathbb{R}^{M \times N \times K}$ is again split into frontal slices along the third axis as

$$\mathbf{X}^{[k]} = \mathbf{A}\mathbf{D}^{[k]}(\mathbf{B}^{[k]})^T, 1 \leq k \leq K, \tag{3.10}$$

with an additional constraint that the cross product of the second mode loadings is constant:

$$(\mathbf{B}^{[k]})^T\mathbf{B}^{[k]} = \mathbf{H}. \tag{3.11}$$

Comparing equation (3.10) to the CPD model in equation (3.9) one can directly observe the added freedom in the second mode factor matrix $\mathbf{B}^{[k]}$ which varies per frontal slice for the PARAFAC2 model. Similarly $\mathbf{D}^{[k]} \in \mathbb{R}^{R \times R}$ is a matrix whose diagonal elements are the k'th row of the third mode factor matrix $\mathbf{C} \in \mathbb{R}^{K \times R}$ as in equation (3.9).

The decomposition is shown visually in figure 3.11 and by comparing this structure to equation (3.5) and figure 3.9 there are also some similarities. While in the IVA case the mixing matrix $\mathbf{A}^{[k]}$ is unconstrained and can vary along the third mode, in the PARAFAC2 case $\mathbf{A}$ is fixed. Conversely IVA imposes an independence structure onto the mode-two matrix $\mathbf{B}^{[k]}$ and dependence for different k's while PARAFAC2 imposes only the constraint of equation (3.11).

3.3.6. BTD

Another decomposition which is used in the neuroimaging context is the block term decomposition (BTD) [9]. The block term decomposition is an extension of CPD, where each term is no longer the outer product of single vectors, but is the outer product of an arbitrary number of vectors in each component [28]. This decomposition does not make any statistical assumptions regarding the structure of the data as is the case for ICA and IVA, and as such has the potential to capture patterns in the data that may be

suppressed by such assumptions. BTD is a general framework, but specific variants of it add structure which often has advantages for decomposing neuroimaging data such as fUS.

Despite these potential advantages, after an investigation, it was not used in this work and is included here only for completeness. The first challenge with BTD is rank estimation, since there are now far more parameters to select than just the number of components thanks to the variable per-component rank. While this rank estimation is challenging, the main reason for its exclusion is based on uniqueness issues which deteriorate the interpretability and quality of results and further increase the challenge of rank estimation.

3.3.7. SCP

The last CPD extension discussed is the shifted canonical decomposition (SCP) [25]. This model is very similar to a normal three-dimensional CPD, except that components in one mode (the second by convention) can have a different shift per element in another mode (the third by convention). The data model can be viewed per element q in the frequency domain as

$$\underline{\tilde{\mathbf{X}}}_{(m,q,k)} = \sum_{r=1}^{R} \mathbf{A}_{(m,r)} \tilde{\mathbf{B}}_{(q,r)} \mathbf{C}_{(k,r)} e^{-j2\pi \frac{q}{N} \tau_{(k,r)}} \tag{3.12}$$

Where $\underline{\tilde{\mathbf{X}}} \in \mathbb{C}^{M \times Q \times K}$ is the DFT of $\underline{\mathbf{X}}$ along the second mode, $\mathbf{A} \in \mathbb{R}^{M \times R}$, $\mathbf{C} \in \mathbb{R}^{K \times R}$ are the mode-one and mode-three factor matrices, and $\tilde{\mathbf{B}} \in \mathbb{C}^{Q \times R}$ is the DFT of the second factor matrix along the first mode. $\tau_{(k,r)}$ is the time shift applied to component r at the mode-three index k. The exponential component is responsible for the phase shift associated with the time shift $\tau_{(k,r)}$.

By exploiting the shift-invariance properties in the frequency domain this algorithm's underlying model is still an entirely linear model, and can be implemented efficiently with an alternating algorithm. In terms of flexibility it is somewhere between CPD and PARAFAC2, as the second mode has added freedom to shift, but not as much freedom as in PARAFAC2, which can model shifts as well as more arbitrary vector rotations.

3.4. Tools and Resources

Data analysis, algorithm implementation and visualisation was performed in MATLAB (R2025a) [21] using the Image Processing Toolbox (25.1), Medical Imaging Toolbox (25.1), Signal Processing Toolbox (25.1), Statistics and Machine Learning Toolbox (25.1), Matlab Tensor Tools [31] and Group ICA/IVA software (4.0.6.8) [6]. ChatGPT (OpenAI, 2025) versions GPT-4o and GPT-5 were used to support grammatical checking and formatting. The experimental design, the experiment itself and data acquisition for the functional ultrasound dataset used was done by Laurens Bosman, Stephanie Dijkhuizen, Bas Koekkoek, Bas Generowicz and Pieter Kruizinga from the Center of Ultrasound Brain imaging at Erasmus MC (CUBE).

4

Data Investigation

The underlying fUS data plays a significant role in the simulated investigation setup, the selection and definition of decompositions, as well as the structural assumptions, due to dependencies on tensor orders and lengths. Therefore it is prudent to first provide context by discussing the dataset and experimental setup in more detail. The assumptions implicit in a decomposition algorithm dictate which data patterns are identified and which are suppressed, so, to inform the decomposition selection, the first section describes the experimental learning paradigm itself, and speculates on the possible effects that may be present. The following section covers practical details, including the data structure, number of subjects, scan orientations and locations, number of recordings, and how the data is represented in tensor format. The final part of this section highlights initial findings related to data non-idealities and errors, which also influence algorithm selection.

4.1. Learning Paradigm - Classical Eye-Blink Conditioning

The experiment producing the data is based on classical eye-blink conditioning (EBC) [32], specifically the delay eye-blink conditioning (d-EBC) paradigm. In this setting mice are subjected to a conditioned stimulus (CS) and an unconditioned stimulus (US) which evokes an eye-blink. The eye-blink response is also referred to as the conditioned response (CR) in this setting. In the d-EBC case the CS precedes the US and then both stimuli terminate simultaneously as shown in figure 4.1. Over time the mice learn to associate the US with the CS, and blink before the US occurs. The CS in this case is a LED light shone into one eye, and the US is a puff of air.

Without going into the underlying neuroscience underpinning this learning phenomenon, we can hypothesize three general learning effects which could conceivably occur and be of interest:

- A response magnitude change in a region (figure 4.2a)
- Changes in the regions activated (figure 4.2b)
- Changing time-to-activation in a region after stimulus (figure 4.2c)

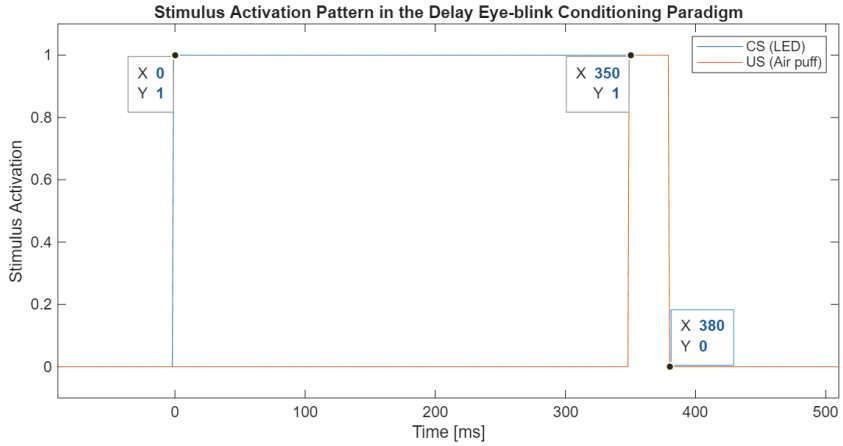
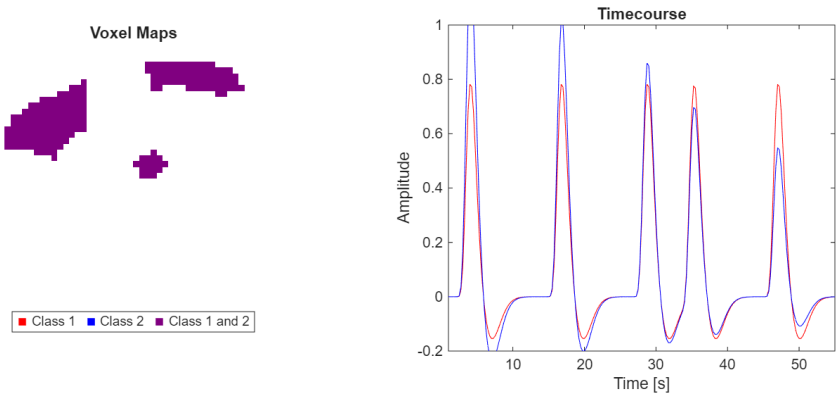
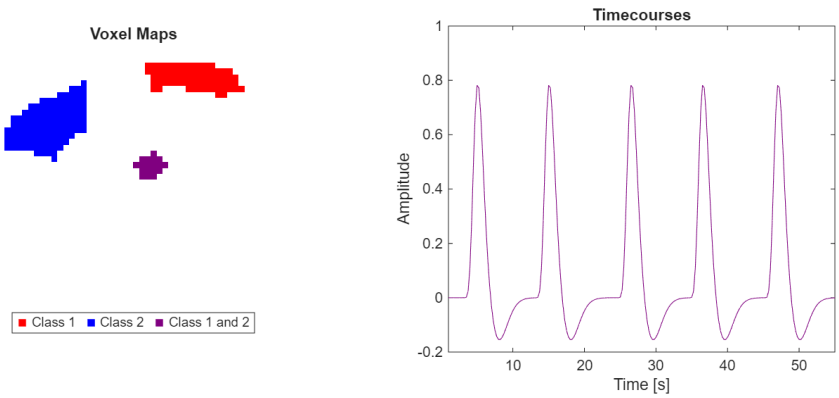


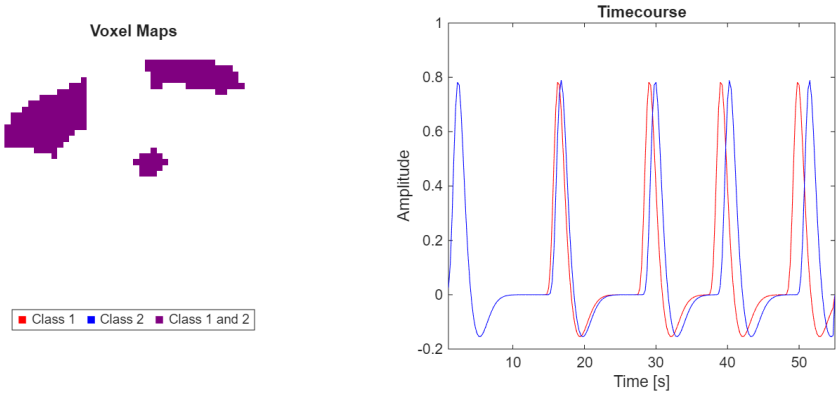
Figure 4.1: Example of a single stimulus event showing CS and US activation over time in ms relative to CS activation onset.



(a) Spatial maps and time-courses for two classes showing differences in response magnitude.



(b) Spatial maps and time-courses for two classes showing differences in activated regions.



(c) Spatial maps and time-courses for two classes showing differences in time-to-activation.

Figure 4.2: Spatial maps and time-courses illustrating different hypothetical learning effects comparing a reference (class 1) to a learning effect (class 2).

The first possible effect is fairly obvious, as the degree to which a region is activated could increase as the subject learns to react to the CS and a responsible region has a stronger response. Instead of an increase in magnitude in a specific region an entirely new region responsible for activating the learned response or retrieving a memory relating to the response is also conceivable. Finally, it could be that a specific region reacts faster over time as the CR becomes associated with the CS and occurs earlier. Combinations of these effects are of course also possible, so ideally the selected algorithms should be able to account for all three occurring.

4.2. Data Format

On a more pragmatic note this subsection goes into more depth than the experimental paradigm, and delves into the data setup and tensor definitions. The data consists of 4 sagittal planar scans of 12 habituated murine subjects being exposed to the aforementioned stimuli. There are on average 5 recordings per scan location (slice), each having been recorded separately for a total of 20 recordings per subject. Each recording consists of a series of fUS PDI images (156 by 128 voxels after registration and cropping) taken at a rate of 4 Hz over a period of $\sim$500 s for a typical size of 2000 PDIs per recording. Due to truncated recordings and absent slices 1 subject and numerous recordings were discarded, leaving 11 subjects and at least 3 recordings for all slices excluding one slice of one subject which only has 2 viable recordings. Since extreme magnitude artefacts were observed at the start and end of many recordings a central subset of 1000 PDIs in each recording were selected to ensure the removal of these effects. These experimental parameters are recorded in table 4.1.

Table 4.1: Data parameters before and after processing for the delay EBC experiment.

Parameter	Original	Processed
Subjects	12	11
Slices	4	4
Recordings/Slice	5	3
PDIs/Recording	$\sim$2000	1000
Voxels	256×128	156×128
Total Voxels	32768	19968

With these parameters established and the dimension lengths of all recordings aligned the data can now be represented as tensors in preparation for the tensor decompositions. The k'th recording of slice index $z \in \{1, 2, 3, 4\}$ of subject $p \in \{1, 2, \ldots, 11\}$ with $X = 156$, $Y = 128$, $N = 1000$ can be represented as

$$\underline{\mathbf{X}}_{rec}^{[k,z,p]} \in \mathbb{R}^{X \times Y \times N} \tag{4.1}$$

or by vectorising the spatial dimensions into a unified dimension of size $M = 19968$ as

$$\mathbf{X}_{rec}^{[k,z,p]} \in \mathbb{K}^{M \times N}. \tag{4.2}$$

The vectorised recordings for all subjects can be added to form a grouped tensor per slice location:

$$\underline{\mathbf{X}}_{slice}^{[z]} \in \mathbb{R}^{M \times N \times K \times P}, \tag{4.3}$$

with $K = 3$ and $P = 11$. And by concatenating the slices on a fifth dimension as well one can form a single tensor

$$\underline{\mathbf{X}} \in \mathbb{R}^{M \times N \times K \times P \times Z} \tag{4.4}$$

of the entire dataset.

4.3. Non-idealities

In preparation for the model selection, pre-processing setup, and synthetic data construction, an initial investigation of the data using several simple techniques is performed. These methods include correlations, direct inspection, spatial averaging, temporal averaging, spatial clustering, and spatial gradients, the details of which can be found in appendix A. In an ideal case the data is composed of only the signal of interest: the stimulus responses and learning effects in regions aligned over all subjects and recordings. Anything other than this signal which deteriorates the spatial maps or time courses is considered a non-ideality. This is a list of the identified non-idealities in the data as well as some typical fUS non-idealities:

- Response fluctuations - A region's response fluctuates both in magnitude and in time-to-peak after stimulus events.
- Tissue deformation - Between different subjects and recordings there are changes in region location and overlap due to tissue deformation.
- Extraneous signals - There are of course other neurological signals measured in the fUS data aside from the stimulus-evoked responses.
- Measurement noise - The measurement process itself introduces noise on each voxel.
- Movement artefacts - When the subject or the ultrasound probe moves it introduces large movement artefacts over the whole image.
- Reference misalignment - It was found that the stimulus reference was misaligned between different recordings.
- Scan plane consistency - Ideally a scan of a specific slice is in the same location along the off-plane axis so that the same regions are captured in different recordings and subjects, which is not always the case in this data.
- In-plane misalignment - In-plane misalignment hinders cross-recording spatial comparison. This can be addressed with registration techniques, however even then there are often still small misalignments.
- Ultrasound-specific effects - There are a number of effects specific to ultrasound such as reverberations, shadows and mirrors, though these are often addressed in preprocessing stages, or cannot be removed as they are a feature of the material properties. Due to the complexity of modelling such effects they are not explored further in this work

These non-idealities are mapped in figure 4.3 in terms of their cause – if it is an undesirable physiological reality being measured, or if it is some artefact introduced by the measurement or processing steps – and in terms of their primary effect, namely temporal or spatial.

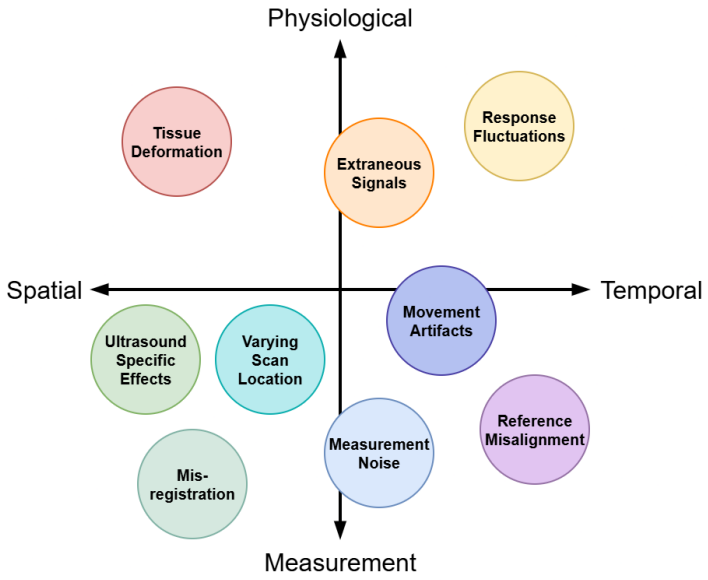


Figure 4.3: Map of fUS non-idealities in terms of underlying cause and dimension of primary impact.

Several of these non-idealities are removed or minimized with pre-processing steps. With perfect spatial pre-processing the tissue deformation and spatial misalignment can be entirely removed, however this is a challenging task, especially for fUS as there is not a lot of structural information contained in the PDIs to rely upon. The registration process used for this data is discussed and outlined in appendix D. Movement artefacts create very significant increases in magnitude in entire PDIs and do not react well to typical standard deviation based outlier detection algorithms, but can be handled with a slightly modified algorithm described in appendix C. Finally, reference misalignments can be very challenging to fix without additional information. In this instance it is presumed to be the result of the experimental setup and thus expected to be a fixed offset, so it was tackled by selecting a group of aligned recordings as a reference, identifying the optimal shift for the remaining recordings based on their stimuli response times, and adjusting the reference signal accordingly for the affected recordings. These steps, the standard pre-processing and some optional artefact removal steps which will be discussed later are shown in figure 4.4.

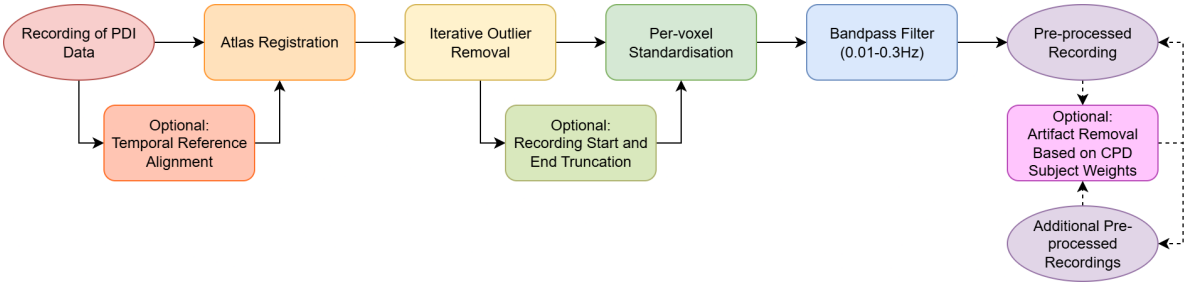


Figure 4.4: Pre-processing pipeline for the fUS dataset, including various optional steps required for either specific recordings or specific decomposition algorithms.

5

Decomposition Development & Modelling Learning Effects

5.1. Algorithms

When analysing neuroimaging data with decompositions both the arrangement of the data and the decomposition chosen change the implicit structural assumptions significantly. This section motivates algorithms based on three promising tensor decompositions (IVA, CPD, and PARAFAC2) using the theorised characteristics of the signals of interest and the inherent structure of the data. Thereafter an additional decomposition algorithm based on SCP is developed.

5.1.1. IVA

Starting with the most unconstrained model of the data, IVA lets one extract components without relying on any alignment other than in a single mode. This is incredibly powerful, as the investigation of the data showed that there are various effects that can detrimentally impact the comparability of data in different recordings and subjects, and this model can accommodate for many of these without issue. IVA generally performs well on diverse sets of tasks and is regularly used in the neuroimaging setting, with two common approaches: temporal IVA (tIVA) and spatial IVA (sIVA), each operating on the temporal or spatial modes respectively.

By assuming independence between components in the temporal dimension (and dependence across related components between recordings) tIVA is able to identify similar patterns of response in different recordings – something which is very reasonable to expect in a stimulus-evoked setting if the stimuli are applied similarly to all recordings. This allows tIVA to exploit information in multiple recordings for source estimates with no limitations on the spatial maps at all. Misalignments or warping in the spatial maps may introduce difficulties when interpreting the extracted components of tIVA, but will have little to no detrimental effects on the algorithm's ability to identify these components. This approach can identify learning effects which cause variations in the response magnitudes of regions by comparing sources in the same SCV to one another as well as region activation differences between subjects by inspecting the associated spatial maps.

The alternative structure (namely, sIVA) assumes that the spatial maps can be sepa-

rated into independent components with dependence across recordings. In practice this translates to components with similar spatial maps between recordings/subjects which generally do not overlap with the spatial maps of other components. As such it is well suited to identifying learning effects which trigger specific region activations, however it is dependent on spatial alignment between subjects to ensure the extracted spatial components are consistent. It is also able to capture both magnitude and shifted structures in the time courses, but since these are entirely unconstrained this means that any undesired artefacts or signals are also included, which can make interpretation challenging.

Applying these algorithms to the data starts with the data tensor $\underline{\mathbf{X}}$ described in equation (4.4). The immediate issue is of course that IVA is only suited to decompose three-way tensors, and $\underline{\mathbf{X}}$ is a five-way tensor. But, because tIVA relies exclusively on statistical independence between components in the second dimension (time courses) and dependence across recordings – and **all** recordings are exposed to the same stimulus – this issue can be simply remedied by flattening the recording, subject and slice dimensions as

$$\underline{\mathbf{X}}_{\text{temporal}} \in \mathbb{R}^{M \times N \times KPZ}, \tag{5.1}$$

with the tensor subscript referring to the data along the second mode, since in many of these decompositions the second mode factors are the ones with unique characteristics. For sIVA the data structure is more challenging as there are four different brain slices in the data capturing different regions. Since these slices cannot be compared spatially the only real recourse is to split the dataset into four (one per slice location), assess each slice's decomposition independently and then compare the components manually afterwards. For this we start with the per-slice tensor $\underline{\mathbf{X}}_{slice}^{[z]}$ from equation (4.3), similarly flatten the recordings and subjects, and permute the first two dimensions to form

$$\underline{\mathbf{X}}_{\text{spatial}}^{[z]} \in \mathbb{R}^{N \times M \times KP}. \tag{5.2}$$

The steps that follow this data structuring include dimensionality reduction to the desired number of sources R, applying the core IVA algorithm to get the de-mixing matrix, and extracting the sources. Since these steps are identical for both tIVA and sIVA, the data tensor distinction is omitted for the rest of this section and a general tensor notation $\underline{\mathbf{X}} \in \mathbb{R}^{M \times N \times K}$ with frontal slices denoted $\mathbf{X}^{[k]}$ is used for illustration. As discussed prior, the data model for IVA is the same as ICA – a simple linear model,

$$\mathbf{X}^{[k]} = \mathbf{A}^{[k]}(\mathbf{B}^{[k]})^T + \mathbf{N}^{[k]}, 1 \leq k \leq K, \tag{5.3}$$

with the noise captured in $\mathbf{N}^{[k]} \in \mathbb{R}^{M \times N}$. The dimensionality reduction step projects all the recordings to a common column subspace using PCA. For PCA the data is first concatenated horizontally and then broken down using the singular value decomposition (SVD) as follows

$$\left[\mathbf{X}^{[1]}, \mathbf{X}^{[2]}, ..., \mathbf{X}^{[K]}\right] = \mathbf{USV}. \tag{5.4}$$

Then the columns of $\mathbf{U}$ associated with the R largest diagonal values in $\mathbf{S}$ are selected to form $\mathbf{U}_s$ which defines the common subspace. The data matrices are then all projected onto this subspace:

$$\hat{\mathbf{X}}^{[k]} = \mathbf{U}_s^T \mathbf{X}^{[k]} \in \mathbb{R}^{R \times N}. \tag{5.5}$$

IVA is now applied to all the data to get de-mixing matrices $\mathbf{W}^{[k]}$ and thus the sources,

$$(\hat{\mathbf{B}}^{[k]})^T = \mathbf{W}^{[k]}\hat{\mathbf{X}}^{[k]}. \tag{5.6}$$

Based on equations (3.3) and (5.5) the mixing matrices $\mathbf{A}^{[k]}$ can also be extracted using $\mathbf{W}^{[k]}$ and $\mathbf{U}_s$ by inverting the de-mixing matrix and then projecting back to the original space:

$$\hat{\mathbf{A}}^{[k]} = \mathbf{U}_s(\mathbf{W}^{[k]})^{-1} \in \mathbb{R}^{M\times R}. \tag{5.7}$$

To inspect the spatial maps of each component it is of course necessary to reshape them back into 2D maps from their vector form in $\hat{\mathbf{A}}^{[k]}$ or $\hat{\mathbf{B}}^{[k]}$ for tIVA and sIVA respectively.

Both tIVA and sIVA are very flexible and make very few assumptions regarding the data, only really relying on temporal and spatial alignment respectively. In the case of tIVA this means that it may struggle to deal with component recognition if there are temporal shifts, and may not capture meaningful shifts in an interpretable manner. For sIVA this means that it relies rather heavily on the spatial registration of the recordings. The flexibility of these algorithms also has some downsides in that they have no way to prioritise the identification of signals of interest, and will capture other unimportant signals and artefacts, potentially hindering interpretation.

5.1.2. PARAFAC2

Instead of leaving some modes entirely unconstrained, the next data model explicitly exploits the shared information across recordings and subjects. All the recordings of a specific slice should theoretically have the same structures in the spatial dimension, assuming good co-registration, which is information that can be exploited. It is also conceivable that since the same stimulus sequences are used for all recordings the time-courses for stimulus related components will also have the same or similar profiles. IVA exploits one or the other of these effects while leaving the other mode free, but it could be beneficial to place more stringent constraints on a dimension than just statistical dependence across recordings/subjects, as this could help in suppressing unwanted artefacts. The parallel factor analysis 2 (PARAFAC2) model enforces shared factors in the first mode for all recordings/subjects while still allowing for (structured) variation per recording/subject in the second mode. Similar to IVA, this decomposition can be applied assuming either shared spatial maps (referred to as temporal PARAFAC2 since the temporal dimension has the latent space which distinguishes PARAFAC2 from the other decompositions) or shared time-courses (referred to as spatial PARAFAC2 for the same reason).

While the assumptions differ, the underlying data model for PARAFAC2 has a lot of commonality with IVA, and the data is structured in similar fashion to sIVA. Here, due to the increased structural enforcement on both the first and second dimensions, decomposing different slices simultaneously is not advisable as they contain different regions. Starting with equation (5.2), one can easily restructure it by separating out the (physical) slices in the data to create

$$\underline{\mathbf{X}}^{[z]}_{\text{temporal}} \in \mathbb{R}^{M\times N\times KP}, \tag{5.8}$$

for temporal PARAFAC2 and the permuted version with switched space and time dimensions

$$\underline{\mathbf{X}}^{[z]}_{\text{spatial}} \in \mathbb{R}^{N\times M\times KP} \tag{5.9}$$

for spatial PARAFAC2. Here again the algorithmic steps that follow are the same regardless of the data structure so we move to a general $\underline{\mathbf{X}} \in \mathbb{R}^{M \times N \times K}$ with frontal slices denoted $\mathbf{X}^{[k]}$. With this generalized notation the data model is

$$\mathbf{X}^{[k]} = \mathbf{A}\mathbf{D}^{[k]}(\mathbf{B}^{[k]})^T + \mathbf{N}^{[k]}, 1 \leq k \leq K. \tag{5.10}$$

The matrices are as described in equation (3.10) and $\mathbf{N}^{[k]}$ captures any unmodelled noise.

Instead of the independence between sources in recordings that IVA uses, PARAFAC2 imposes only a fixed covariance structure on the second mode, allowing the factors in this mode to vary. This relatively simple constraint has some more complex implications, as it creates low rank encodings of structures in the second mode which are then arbitrarily rotated to form $\mathbf{B}^{[k]}$ and scaled by the diagonal values of $\mathbf{D}^{[k]}$ to form the components of recording k. This means that components which share a latent structure can be compared across recordings by looking at the rows of $\mathbf{D}^{[k]}(\mathbf{B}^{[k]})^T$ for different k's while the "mixing matrix" $\mathbf{A}$ is shared between all recordings, and the mixing vectors associated with specific components are in the columns of $\mathbf{A}$.

Making more assumptions than IVA, this approach is better suited to identify some effects than others, but also relies on certain characteristics of the data. The most obvious benefit of temporal PARAFAC2 structure is that responses can be shifted in time and still captured easily by the model, in theory other vector rotations and scalings could also be captured, so as long as stimulus-evoked responses can be well captured in a low rank latent form this approach has potential. A downside as well as potential advantage is that the spatial map is fixed for a given component. This means that if the spatial maps do not align, then the results will be deteriorated, or if in different recordings a different region is activated when a stimulus event occurs then it would either be stored in a different component or not stored at all, and if multiple regions have a similar response then comparison and interpretation would require one to inspect and compare all components. On the other hand, if the spatial maps are aligned, and the learning effects do not manifest as spatially different region activations, then this model leverages the commonality of spatial maps to decompose the data. The converse is true for spatial PARAFAC2 with spatial maps in the second mode. This model can capture some degree of spatial variation such as translations, but it requires that the temporal signals are the same for all recordings. This temporal consistency is not guaranteed in fUS data considering response variations, but since only the stimulus related effects are likely to be correlated across recordings this approach could potentially suppress undesired signals.

5.1.3. CPD

Continuing the trend of increasing the amount of restrictions on the decomposed data the natural next step is to have both the spatial and the temporal factors shared across recordings and subjects to maximally suppress unwanted artefacts using the canonical polyadic decomposition. An added benefit of using the CPD is that it is easy to extend to higher order tensors, and given that there is no latent representation on the second mode like for PARAFAC2 there is no reason not to make full use of this. By noting that the time courses are similar because they share stimulus responses at similar locations, and that the stimulus locations are known, one can directly encode the stimulus similarity into the decomposition by restructuring the data tensor to extract responses using a fourth order CPD instead of extracting time-courses with a

third-order[1] CPD. With the shared factors in the spatial dimension this naturally also needs to be applied to each set of slice data independently.

To achieve this the slice-wise data matrix from equation (5.1),

$$\underline{\mathbf{X}}_{\text{temporal}}^{[z]} \in \mathbb{R}^{M \times N \times KP}, \tag{5.11}$$

is restructured into a four-way tensor

$$\underline{\mathbf{X}}_{\text{f}}^{[z]} \in \mathbb{R}^{M \times N_{\text{f}} \times F \times KP}. \tag{5.12}$$

This restructuring is achieved by first splitting the time dimension of $\underline{\mathbf{X}}_{\text{temporal}}^{[z]}$ into F different frames of length N_{f}, each starting when a stimulus occurs (or optionally a few samples before in cases where the stimulus indices are suspect). These frames are then concatenated along a new dimension which is permuted with the third dimension to form $\underline{\mathbf{X}}_{\text{f}}^{[z]}$. Most of the information from $\underline{\mathbf{X}}_{\text{temporal}}^{[z]}$ is retained in $\underline{\mathbf{X}}_{\text{f}}^{[z]}$, but some loss of temporal data is possible if the frequency of the stimuli combined with the length of the frames leaves gaps in the temporal coverage. Though given the focus on stimulus-evoked learning this is unlikely to contain valuable information.

This structure can now be decomposed using a fourth-order CPD with data model

$$\underline{\mathbf{X}}_{\text{f}}^{[z]} = \sum_{r=1}^{R} \mathbf{A}_{(:,r)} \circ \mathbf{B}_{(:,r)} \circ \mathbf{C}_{(:,r)} \circ \mathbf{D}_{(:,r)} + \underline{\mathbf{N}}, \tag{5.13}$$

Here $\mathbf{A} \in \mathbb{R}^{M \times R}$, $\mathbf{B} \in \mathbb{R}^{N_f \times R}$, $\mathbf{C} \in \mathbb{R}^{F \times R}$, and $\mathbf{D} \in \mathbb{R}^{KP \times R}$ refer to the spatial, temporal, frame and recording/subject factor matrices respectively and $\underline{\mathbf{N}} \in \mathbb{R}^{V \times N_f \times F \times KP}$ accounts for any unmodelled noise in the data. A small implementation detail which is worth mentioning is that the component magnitudes are stored in the recording factor matrix $\mathbf{D}$, with all the component vectors in the other factor matrices scaled to contain no magnitude information.

This model is in many ways more restrictive than the aforementioned IVA and PARAFAC2 options, but this downside comes with some advantages when interpreting data. The IVA and PARAFAC2 models each allow for a lot of variation in a single component between recordings in at least one dimension. For the CPD model this is not the case, and recordings can only vary in the magnitude of a certain component's contribution – which can be directly extracted from the values of the recording factor matrix $\mathbf{D}$. This limitation has the benefit of making inter-recording comparisons exceedingly simple, unlike IVA and temporal PARAFAC2, where one needs to compare entire time courses, the differences are all encoded directly in the values of $\mathbf{D}$. By inspecting the frame factor matrix $\mathbf{C}$ it is possible to directly identify response magnitude trends over time for a specific component (which combined with the recording factor matrix magnitudes can give an indication of how strong this effect is in that recording/subject).

Beyond the improved interpretability, this algorithm is specifically well suited to identify stimuli-evoked response effects. By not allowing component variation between

[1] While it is entirely possible to apply a third order CPD as is, it results in weaker artefact suppression and less interpretable results, though if one expects the response profile to change consistently over time in the same way for all recordings and subjects (unlikely but not entirely inconceivable) then this would be a better model than the fourth order option.

recordings and aligning the frames around stimuli it suppresses noise and other undesired neurological signals in the extracted components. Spatial effects such as specific region activation in only some recordings would also be easy to identify by simply inspecting the spatial maps in $\mathbf{A}$. Unfortunately the rigidity of the structure means that this model cannot effectively capture changes such as temporal shifts or changes in response profile. Furthermore, as the spatial map for a given component is fixed between recordings as in temporal PARAFAC2, it means that this data model is also very reliant on accurate spatial alignment.

5.1.4. Multi-shift CPD

The CPD model offers very interpretable components and owing to its structure has good noise and unwanted signal suppressions properties, but it lacks the ability to address temporal shifts like some of the other models. Since it is known that the time-to-responses after stimuli vary per stimulus, and a change in response time is one of the hypothetical learning effects, the ability to account for and model such effects is very desirable. It would be further beneficial if it were possible to identify the same response profile in different regions with different time shifts, as this could indicate functional relationships between different regions,

The SCP model is a promising candidate for capturing these time shifts while preserving the same structure as the CPD model, but it has a number of drawbacks. The first is a simple matter of volume: a single recording has nearly 20000 voxels and ideally one would have a shift associated with every voxel for every component, and while SCP can do this it takes over 5 minutes to do a single update iteration on one recording, leading to prohibitive runtimes[2]. The second issue is that the model only allows for a different shift in the second mode for each element in the third, but to capture both functional region response shifts and the stimulus instance specific response delay it requires a unique shift to be stored for every voxel (mode 1) **and** every stimulus event (mode 3). Finally it would also be desirable to be able to constrain the range of possible shifts, as the response time variability is only so much. To address all these issues, instead of using the SCP model, we develop and implement a new decomposition model which still shares some of the qualities of SCP.

[2] A runtime comparison between SCP and the full multi-shift CPD is included in appendix E

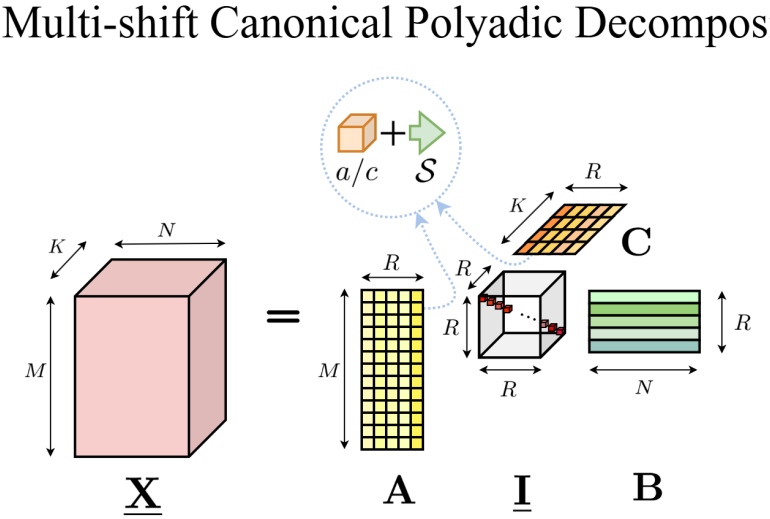


Figure 5.1: Graphical depiction of the multi-shift CPD data model. The model is the same as that of CPD except that each element in the first and third mode factor matrices is composed of both a magnitude and a shift affecting the factors in the third mode.

This model does not operate in the frequency domain like the original SCP and for the three way case the data model is

$$\underline{\mathbf{X}}_{(m,n,k)} = \sum_{r=1}^{R} \mathbf{A}_{(m,r)} \mathbf{B}_{(n+\tau_{(m,r)}+\psi_{(k,r)},r)} \mathbf{C}_{(k,r)} \tag{5.14}$$

or perhaps more intuitively for mode-two fibres

$$\underline{\mathbf{X}}^{T}_{(m,:,k)} = \sum_{r=1}^{R} \mathbf{A}_{(m,r)} \mathbf{C}_{(k,r)} \mathbf{B}^{T}_{(:,r)} \mathcal{S}_N^{(\tau_{(m,r)}+\psi_{(n,r)})}. \tag{5.15}$$

With:

- $\tau_{(m,r)}$ the integer shift associated with the r'th component and m'th entry along the first mode (voxels in this case) and constrained to $|\tau_{(n,r)}| \leq \tau_{\text{max}}$.
- $\psi_{(n,r)}$ the integer shift associated with the r'th component and n'th entry along the third mode (stimulus instance) and constrained to $|\psi_{(n,r)}| \leq \psi_{\text{max}}$.
- $\mathcal{S}_N^p$ is a shifting matrix of size $N \times N$ defined as:

$$\mathcal{S}_N^p = \begin{cases} \mathcal{U}_n^p & \text{if } p > 0 \\ \mathcal{I}_n & \text{if } p = 0 \\ (\mathcal{U}_n^T)^p & \text{if } p < 0 \end{cases} \tag{5.16}$$

where $\mathcal{U}_n$ is the upper shift matrix of size $N \times N$ with ones only on the upper diagonal and $\mathcal{I}_n$ is an $N \times N$ identity matrix. This model also explicitly constrains the number of shifts to a range in each mode.

B Update Development

Much like an alternating least squares algorithm used for CPD this model can be optimised using an alternating approach where all mode factor weights and shifts are

fixed while the remaining factor weights (and shifts if present) are updated. This process then repeats over all modes until a convergence criterion is met. Starting with the factor containing the shifted components $\mathbf{B}$ the main challenge is of course to also account for the shifts. To develop this update step it is easiest to start with a model of order two, then to solve it by including the shifted structures explicitly before extending the model to higher orders.

The mode-two shift model,

$$\mathbf{X}_{(m,:)} = \sum_{r=1}^{R} \mathbf{A}_{(m,r)} (\mathcal{S}_N^{\tau_{(m,r)}})^T \mathbf{B}_{(:,r)}, \tag{5.17}$$

can be written as a single matrix operation

$$\mathbf{X}_{(m,:)} = \mathbf{A}'_m \mathbf{b}', \tag{5.18}$$

by stacking the coefficients and shift matrices horizontally in

$$\mathbf{A}'_m = \begin{bmatrix} \mathbf{A}_{(m,1)} (\mathcal{S}_N^{\tau_{(m,1)}})^T & \mathbf{A}_{(m,2)} (\mathcal{S}_N^{\tau_{(m,2)}})^T & \dots & \mathbf{A}_{(m,R)} (\mathcal{S}_N^{\tau_{(m,R)}})^T \end{bmatrix} \in \mathbb{R}^{N \times NR} \tag{5.19}$$

and vectorising $\mathbf{B}$ into a single column vector

$$\mathbf{b}' = \begin{bmatrix} \mathbf{B}_{(:,1)}^T & \mathbf{B}_{(:,2)}^T & \dots & \mathbf{B}_{(:,R)}^T \end{bmatrix}^T \in \mathbb{R}^{NR}. \tag{5.20}$$

Since the vector $\mathbf{b}'$ here does not depend on the mode-one index m, equation (5.18) can be further expanded to a standard dyadic model including the whole of $\mathbf{X}$ in vectorised form by vertically concatenating for all m's

$$\mathbf{X}' = \mathbf{A}'' \mathbf{b}', \tag{5.21}$$

with $\mathbf{X}' \in \mathbb{R}^{NM}$ transposed rows vertically stacked

$$\mathbf{X}' = \begin{bmatrix} \mathbf{X}_{(1,:)}^T & \mathbf{X}_{(2,:)}^T & \dots & \mathbf{X}_{(M,:)}^T \end{bmatrix}^T \tag{5.22}$$

and $\mathbf{A}'' \in \mathbb{R}^{NM \times NR}$ similarly stacked and having a block structure

$$\mathbf{A}'' = \begin{bmatrix} \mathbf{A}_{(1,1)} (\mathcal{S}_N^{\tau_{(1,1)}})^T & \dots & \mathbf{A}_{(1,R)} (\mathcal{S}_N^{\tau_{(1,R)}})^T \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{(M,1)} (\mathcal{S}_N^{\tau_{(M,1)}})^T & \dots & \mathbf{A}_{(M,R)} (\mathcal{S}_N^{\tau_{(M,R)}})^T \end{bmatrix} \tag{5.23}$$

Now theoretically, despite all the shifts, all the elements of $\mathbf{B}$ can be estimated in closed form using the pseudo-inverse of $\mathbf{A}''$:

$$\mathbf{b}' = (\mathbf{A}''^T \mathbf{A}'')^{-1} \mathbf{A}''^T \mathbf{X}'. \tag{5.24}$$

$\mathbf{B}$ Update Implementation

But there is a catch, even in this mode-two case, $\mathbf{A}''$ is colossal[3], and may not even fit in memory, let alone be performant during operation. On the other hand there is a lot of structure in $\mathbf{A}''$ – there are many repeated entries and it has a sparsity which is at least

[3]For the dataset in question M is in the order of 20000, so even with a modest number of samples per voxel (say 100) and 10 components this matrix takes up nearly 15Gb. For higher order extensions this problem just grows larger.

$\frac{N-1}{N}$. The goal now becomes to avoid explicitly constructing $\mathbf{A}''$ at any point, and since $\mathbf{A}''$ is a very tall matrix we separate the left side of equation (5.24) into the Gram matrix and the cross-product matrix to avoid the large first dimension of $\mathbf{A}''$. Exploiting the block structures we can compute $\mathbf{A}''^T\mathbf{X}' \in \mathbb{R}^{NR}$ very efficiently as stacked weighted sums of shifted versions of the rows of $\mathbf{X}$

$$\mathbf{A}''^T\mathbf{X}' = \begin{bmatrix} \sum_{m=1}^{M} \mathbf{A}_{(m,1)} \mathcal{S}_N^{\tau_{(m,1)}} \mathbf{X}_{(m,:)} \\ \vdots \\ \sum_{m=1}^{M} \mathbf{A}_{(m,R)} \mathcal{S}_N^{\tau_{(m,R)}} \mathbf{X}_{(m,:)} \end{bmatrix}. \tag{5.25}$$

In practice the shift matrices in this step are also not constructed, and simply implemented using matrix indexing (in blocks that share the shift) to avoid redundant computation.

Now for $\mathbf{A}''^T\mathbf{A}'' \in \mathbb{R}^{NR \times NR}$ we exploit the same block structure to compute

$$\mathbf{A}''^T\mathbf{A}'' = \begin{bmatrix} A'''_{1,1} & \dots & A'''_{1,R} \\ \vdots & \ddots & \vdots \\ A'''_{R,1} & \dots & A'''_{R,R} \end{bmatrix} \tag{5.26}$$

with each block A'''_{r_1,r_2} the products of the coefficients and appropriate shifts

$$A'''_{r_1,r_2} = \sum_{m=1}^{M} \mathbf{A}_{(m,r_1)} \mathbf{A}_{(m,r_2)} \mathcal{S}_N^{\tau_{(m,r_1)}} (\mathcal{S}_N^{\tau_{(m,r_2)}})^T \tag{5.27}$$

Since M is very large we can further optimise this step by defining the set of unique delay pairs:

$$\mathcal{V}_{r_1,r_2} = \left\{ \left(\tau_{(m,r_1)}, \tau_{(m,r_2)}\right) \;\middle|\; m = 1, \dots, M \right\} \tag{5.28}$$

and each unique pair $(\alpha, \beta) \in \mathcal{V}_{r_1,r_2}$, define the index set:

$$\mathcal{M}_{\alpha,\beta} = \left\{ m \in \{1, \dots, M\} \;\middle|\; \left(\tau_{(m,r_1)}, \tau_{(m,r_2)}\right) = (\alpha, \beta) \right\} \tag{5.29}$$

Then the expression for A'''_{r_1,r_2} becomes:

$$A'''_{r_1,r_2} = \sum_{(\alpha,\beta) \in \mathcal{V}_{r_1,r_2}} \left(\sum_{m \in \mathcal{M}_{\alpha,\beta}} \mathbf{A}_{(m,r_1)} \mathbf{A}_{(m,r_2)} \right) \mathcal{S}_N^{\alpha} \left(\mathcal{S}_N^{\beta}\right)^T \tag{5.30}$$

This speeds up the process by reducing the number of repeated $\mathcal{S}_N^{\alpha} \left(\mathcal{S}_N^{\beta}\right)^T$ matrix products. A few other small improvements also include computing only the upper triangle of $\mathbf{A}''^T\mathbf{A}''$ since it is symmetric, and using sparse matrix data structures. Computing the inverse of this matrix is still not ideal, but with efficient linear solvers the problem is entirely tractable.

Extending this update step to a higher order setting is surprisingly simple, much like with CPD's alternating least squares algorithm it can be directly formulated in the mode two case by computing the flattened version of the other modes using the column-wise Khatri-Rao product

$$(\mathbf{C} \odot \mathbf{A}) \in \mathbb{R}^{mk \times r} \tag{5.31}$$

and a new shift matrix using a column-wise outer sum formulation

$$\xi = \psi \odot \mathbf{1}_{m\times r} + \mathbf{1}_{k\times r} \odot \tau \in \mathbb{R}^{mk\times r} \tag{5.32}$$

and then simply applying the same update steps with a flattened $\mathbf{X}$ matrix.

τ Update

Unlike updating $\mathbf{B}$, the challenge of estimating the time shifts is not in data volume, but in traversing a combinatorially exploding tree of possible options. Starting with the same mode two shift model in equation (5.17) we instead absorb the shifts into the $\mathbf{B}$ matrix to form a different dyadic model

$$\mathbf{X} = \mathbf{A}'\mathbf{B}'^{T}. \tag{5.33}$$

Here $\mathbf{B}'$ is composed of R sub-matrices

$$\mathbf{B}' = \begin{bmatrix}\mathbf{B}''_1 & \mathbf{B}''_2 & \dots & \mathbf{B}''_R\end{bmatrix} \in \mathbb{R}^{N\times R(2\tau_{\max}+1)} \tag{5.34}$$

each containing all possible shifts of one of R components

$$\mathbf{B}''_r = \begin{bmatrix}\mathcal{S}_N^{\tau_{\max}}\mathbf{B}_{(:,r)} & \mathcal{S}_N^{(\tau_{\max}-1)}\mathbf{B}_{(:,r)} & \dots & \mathcal{S}_N^{-\tau_{\max}}\mathbf{B}_{(:,r)}\end{bmatrix} \in \mathbb{R}^{N\times(2\tau_{\max}+1)}. \tag{5.35}$$

The mixing matrix $\mathbf{A}' \in \mathbb{R}^{M\times R(2\tau_{\max}+1)}$ logically also has R sub-matrices,

$$\mathbf{A}' = \begin{bmatrix}\mathbf{A}''_1 & \mathbf{A}''_2 & \cdots & \mathbf{A}''_R\end{bmatrix}, \tag{5.36}$$

where each $\mathbf{A}''_r \in \mathbb{R}^{M\times(2\tau_{\max}+1)}$ encodes time-shift selection weights for the r-th component. We denote by $\mathbf{A}''_{r(m,:)} \in \mathbb{R}^{1\times(2\tau_{\max}+1)}$ the m-th row of the r-th sub-matrix with the following constraint to ensure selection sparsity such that only a single shifted version of a component is included:

$$\left\|\mathbf{A}''_{r(m,:)}\right\|_0 \leq 1, \quad \forall r \in \{1,\dots,R\},\ \forall m \in \{1,\dots,M\}. \tag{5.37}$$

This new formulation has the shifts explicit, but introduces a new problem – if one were to estimate a new $\mathbf{A}'$ there is no easy way to ensure that the structure of equation (5.37) is preserved. An exhaustive search of all possible shift combinations for each row of $\mathbf{X}$ is not feasible and a convex gradient based method with relaxed constraint is still far too slow. We are thus forced to estimate the optimal values per component.

Let us define the residual matrix $\mathbf{X}\neg r$ by excluding the r-th component from the decomposition:

$$\mathbf{X}_{\neg r} = \mathbf{X} - \mathbf{A}''_{\neg r}\mathbf{B}''^{\mathbf{T}}_{\neg \mathbf{r}}, \tag{5.38}$$

where $\mathbf{A}''_{\neg r}$ and $\mathbf{B}''_{\neg r}$ are formed by removing the r-th sub-matrices from $\mathbf{A}'$ and $\mathbf{B}'$, respectively. Each row of the residual $\mathbf{X}_{\neg r}$ is projected onto the columns of $\mathbf{B}''_r$ to get

$$\mathbf{P}_r = \mathbf{X}_{\neg r}\mathbf{B}''_r \operatorname{diag}\left(\left\|\mathbf{B}''_{m(:,1)}\right\|_2^{-1}, \dots, \left\|\mathbf{B}''_{m(:,2\tau_{\max}+1)}\right\|_2^{-1}\right) \tag{5.39}$$

For each row m of $\mathbf{P}_r$, identify the index of the largest entry

$$j^*_m = \arg\max_j \mathbf{P}_{r(m,j)} \tag{5.40}$$

Then update the shift and coefficient matrices with the optimal values

$$\tau_{(m,r)} = j_m^* - 1 - \tau_{\mathsf{max}}, \tag{5.41}$$

$$\mathbf{A}''_{(m,r)} = \frac{\mathbf{P}_{r(m,j_m^*)}}{\left\|\mathbf{B}''_{m(:,j_m^*)}\right\|_2}. \tag{5.42}$$

and repeat the process for all components r.

The implementation of this process is fairly straightforward, and the main optimisation comes from updating the residual sequentially in each step instead of recomputing it for every component:

$$\mathbf{X}_{\neg r_2} = \mathbf{X}_{\neg r_1} + \mathbf{A}''_{r_2}\mathbf{B}''^{\mathbf{T}}_{\mathbf{r_2}} - \mathbf{A}''_{r_1}\mathbf{B}''^{\mathbf{T}}_{\mathbf{r_1}}. \tag{5.43}$$

Extending the solution to higher orders is also straightforward, as one simply frames it as a matrix operation by computing the Khatri-Rao product of the other factor matrices as usual while also shifting the $\mathbf{B}$ with the relevant shift. Due to the nature of sample shifts though it is essential that the shifted component $\mathbf{B}$ be the last in the Khatri-Rao product, and of course that $\mathbf{X}$ is permuted to have aligned dimensions. One other small nuance is that to ensure consecutive components in the flattened form do not cross-contaminate one another in the projection step one needs to zero pad between the components by $2\tau_{\mathsf{max}}$, and then accordingly also add zeros at the same locations in the flattened form of $\mathbf{X}$.

Application to data

To exploit this model's ability to model multiple signal shifts one would want to have a unique shift for each instance of a stimulus response, and each voxel. This can be simply achieved by unfolding the third mode of equation (5.12) into the fourth to form a tensor

$$\underline{\mathbf{X}}_{\mathsf{f}}^{[z]} \in \mathbb{R}^{M \times N_{\mathsf{f}} \times FKP}. \tag{5.44}$$

and applying the model in equation (5.15). This will then capture a unique magnitude and shift for each response instance of every recording, while keeping the spatial maps and response envelopes fixed.

5.1.5. Decomposition Assumptions

A summary of the assumptions each algorithm places on the data is described in table 5.1 along with the resulting implications of these assumptions.

Table 5.1: Assumptions and structural implications of various tensor decomposition algorithms.

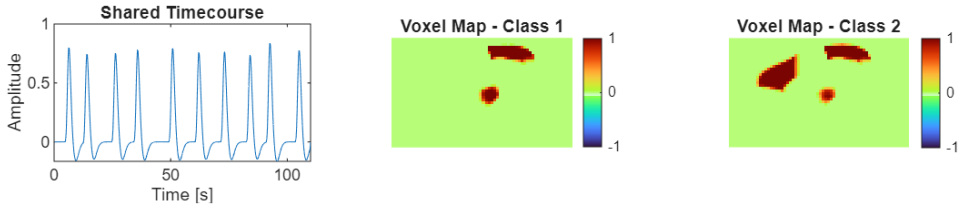
Algorithm	Assumptions	Structural Implications
tIVA	Assumes statistical independence between temporal components within recordings and statistical dependence within the same components across recordings (SCVs).	Time courses of interest will be statistically dependent across recordings, but differ between components.
sIVA	Assumes statistical independence between component spatial factors within recordings and statistical dependence of the same components across recordings (SCVs).	Spatial maps of interest will be statistically dependent across recordings but with minimal spatial overlap between components.
Temporal PARAFAC2	Allows variations across the temporal mode while assuming a common factor structure in the spatial dimension; assumes a shared latent representation of the temporal factors in all recordings.	Spatial maps will be the same for all recordings while time-courses can vary up to a rotation of a latent representation.
Spatial PARAFAC2	Allows variations across the spatial mode while assuming a common factor structure in the temporal dimension; assumes a shared latent representation of the spatial factors in all recordings.	Time-courses will be the same for all recordings while spatial maps can vary up to a rotation of a latent representation.
CPD (4D)	Assumes shared factor matrices across all modes.	Spatial maps, response envelopes and response magnitude changes over time are the same for all recordings and can vary only in component magnitude per recording.
Multi-shift CPD	Assumes shared factor matrices across all modes, but the second mode factors can be shifted by an amount for each element in the other factors.	Spatial maps, response envelopes, and shift per voxel are shared between all recordings but response magnitude and shift per stimulus instance can vary.

5.2. Experimental Setup - Dynamic Effect Representation

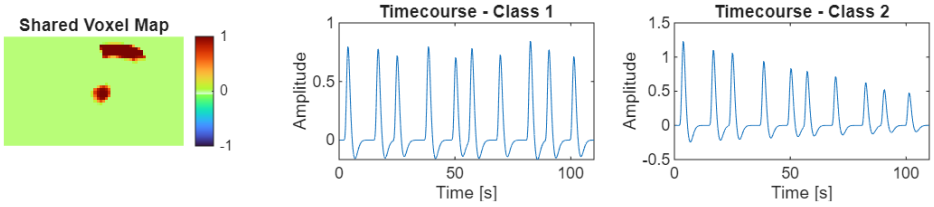
All the algorithms described are in part motivated by their ability to capture the prospective learning effects in an interpretable manner, and their use depends on this ability. To validate this, this section assesses the decompositions' abilities to capture the theorised learning signatures, and if one can observe distinctions between recordings based on these signatures.

For this experiment a synthetic recording for two different subjects is constructed. Each subject belongs to a different class, where the first class is some baseline response, and the second class implies a recording with a learning effect injected into it. There are also shared responses which do not differ between classes alongside the injected effects. The decompositions are applied to both subjects simultaneously, and the decomposed components are inspected directly. There are four possible learning effects which are tested for. The first format is a simple spatial variation, where the same stimulus response occurs in both recordings with an additional region being ac-

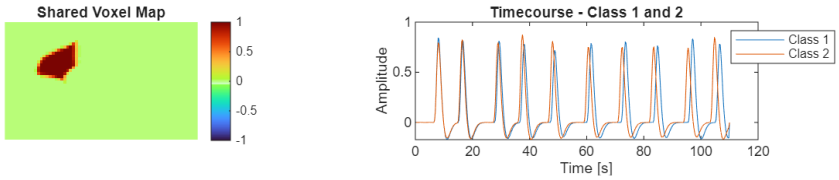
tivated in the second recording, shown in figure 5.2a. In the second case (figure 5.2b) the spatial maps are the same for both recordings, but the temporal response is adjusted to have a varying response magnitude. As learning takes place a certain region could respond faster over time, so a temporal shift effect is modelled in figure 5.2c where one region has a progressively faster response in one of the recordings. Finally learning effects could include localized magnitude changes where a certain region has increasing or decreasing response magnitudes or conceivably that a region does not respond initially, and then starts to respond at a later stage once learning has occurred as depicted in figure 5.2d.



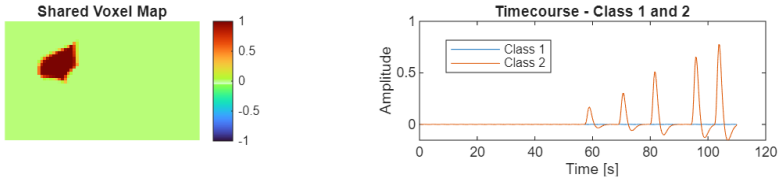
(a) The time-course and spatial maps for both classes in the differing region based synthetic learning effect test.



(b) The spatial map and time-courses for both classes in the differing response magnitude based synthetic learning effect test. One can see the decreasing magnitudes for the second class.



(c) The spatial map and overlaid time-courses for both classes in the temporally shifted learning effect test, showing the gradual speed-up of responses for class 2.



(d) The spatial map and time course for the localized magnitude change which only occurs in class two (one can see the associated time course has zero magnitude for class 1).

Figure 5.2: Figures of the spatial maps and time courses characterising the learning effects in all the different experiments for dynamic effect representation.

5.3. Results - Dynamic Effect Representation

The results of the interpretation of extracted components for all experiments and all decompositions are displayed in table 5.2, and the extracted components themselves can be found in appendix G. For the tabular results in table 5.2 each setup is given an integer score of 0,1 or 2. A score of 0 means that the decomposition was not able

to effectively use the injected learning effect to distinguish between the recordings. A score of 1 or 2 means that the decomposition can distinguish between the recordings using the learning effect, but in the case of a 1 this difference is difficult to interpret from the results themselves, and a 2 means that the differences are easily identified in the extracted components. To illustrate some interesting effects a few relevant components are also shown in figures 5.3 to 5.5.

Table 5.2: Learning effect identification scores for different learning effect types and decompositions. A score of 0 is a failure to extract the learning effect, a score of 1 denotes that there is a class distinction but the components are not interpretable, and a 2 is successful extraction and easily interpretable components.

	Learning Effect Capturing Capacity of Model Variants					
Experimental Differences	tIVA	sIVA	Temporal PARAFAC2	Spatial PARAFAC2	CPD	Multi-shift CPD
Differing Region	2	2	2	2	2	2
Response Magnitude	2	1	1	0	2	2
Response Shift	1	0	0	1	0	2
Isolated Region	2	2	1	2	2	2

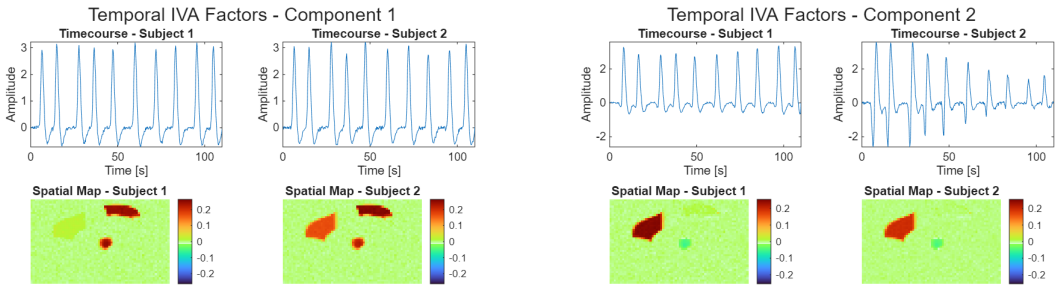


Figure 5.3: This figure contains the plots for components 1 and 2 of the temporal IVA decomposition in the shifted response experimental setup. This model captures some level of distinction between classes, interpreting the effect as a time shift is not obvious, and requires that one look at all the components simultaneously.

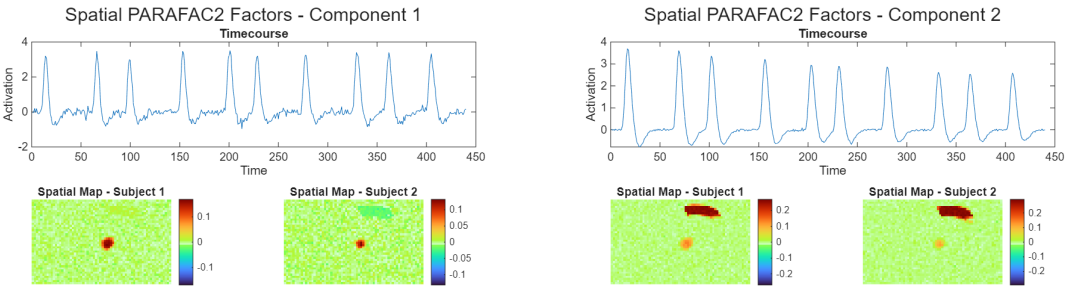


Figure 5.4: This figure contains the plots for components 1 and 2 of the spatial PARAFAC2 decomposition in the changing response magnitude experimental setup. There is no easily distinguishable difference between the subjects in the extracted components.

5.4. Discussion - Dynamic Effect Representation

From table 5.2 there are several interesting observations to be made regarding both the decompositions and the hypothesised learning effects. Looking at the rows it is

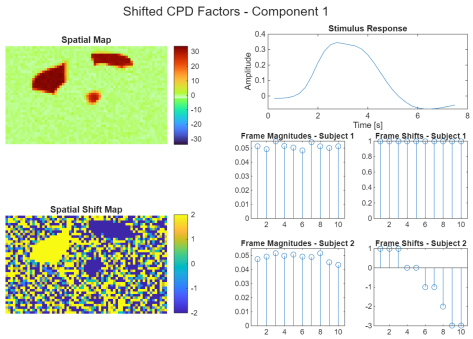


Figure 5.5: The first and only component of the multi-shift CPD decomposition in the time shifted experimental setup. Here one can clearly observe the time shifts in subject 2, and between regions, however there is some level of response warping occuring.

clear to see that the most challenging effect to model is the temporal shifts. This is unsurprising considering that most of these algorithms are based on instantaneous linear models, and have no way to explicitly capture time shifts – which was one of the two motivations behind the development of the multi-shift CPD algorithm.[4] The high levels of flexibility which IVA affords still allows it to capture shifted effects, however not in a particularly interpretable manner, as one can see in figure 5.3. In contrast, the multi-shift CPD component shown in figure 5.5 trivializes the identification of time shifts. Looking at the columns instead of the rows the worst performing decomposition is certainly PARAFAC2. This may not be as conclusive as it seems, as with sufficient components the PARAFAC2 based algorithms will certainly be able to capture the effects in questions, though this would deteriorate interpretability due to increased components needing to be cross examined. The spatial PARAFAC2 model is the only one which failed to capture the magnitude changes entirely, as shown in figure 5.4. This effect is fairly simple, and with the flexibility of PARAFAC2 it should be able to capture this effect in a meaningful manner. It is worth noting that in this case it is likely an issue of experimental setup in the form of decomposition rank selection, and the addition of one or two more components for this algorithm would likely lead to very interpretable results. One last point worth mentioning is that while the effects of interest are very clear in the multi-shift CPD component shown in figure 5.5, one can also see some response warping that occurs. This effect could conceivably be introduced either due to time shifts between regions, or more likely by the fact that the response shifts are not multiples of the sampling frequency, which leads to the stepped structure of the shift graph which could lead to warping during the response update step.

[4]The second motivation was to be able to estimate large numbers of shifts more efficiently than SCP. Evidence of its success may be found in the runtime speed-up comparison between SCP and multi-shift CPD included in appendix E.

6

Robustness to Non-idealities

Beyond just the ability to capture learning effects, it is also important for the decompositions to be robust to various non-idealities that occur in real data. This chapter performs algorithm performance analysis using synthetic data generated from a variety of different scenarios and assesses performance in terms of subgroup identification using the extracted components. The topics covered include discussion of the synthetic data generation, how subgroup identification is performed per decomposition, and the overall results obtained. The synthetic data is generated using a model which is designed to match real fUS data as far as is reasonable.

6.1. Synthetic Data Generation

The generation of synthetic neurological fUS data can be broken down into three steps: generation of the temporal response, generation of the voxel grid, and the combination of the two. The addition of errors, noise, and other non-idealities takes place in one of these steps. This model will be used to investigate the effects of different non-idealities, and is based on both physiological and measurement-induced non-idealities as identified in real fUS data (described in appendix B) using the algorithms described in appendix A.

6.1.1. Ideal Model

This subsection elaborates on the basic underlying model used to generate synthetic fUS data. The steps that follow include the response generation, the spatial map (voxel grid) generation, and the combination of the two.

Response Generation

To start it is assumed that a discrete stimulus timing function $a[n]$, of the form

$$a[n] = \sum_{k=1}^{K} \delta[n - n_k] = \sum_{k=1}^{K} \delta_{n_k}[n], \tag{6.1}$$

is known. Here n_k denotes the sample at which the k'th stimulus occurs and $\delta[n]$ is the discrete unit sample function defined as

$$\delta[n] \;= \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases} \text{, or } \; \delta_d[n] \;= \begin{cases} 1, & n = d \\ 0, & n \neq d \end{cases} \tag{6.2}$$

Using this stimulus signal one can construct a synthetic haemodynamic response,

$$\mathbf{s} = \big[s[1], s[2], ..., s[N]\big]^T, \tag{6.3}$$

of length N, where

$$s[n] = (a * h)[n]. \tag{6.4}$$

Here h refers to a discrete canonical hemodynamic response function (CHRF) [19] described as

$$h[n] = A_1 \cdot \frac{n^{\alpha_1 - 1} e^{-n/\beta_1}}{\beta_1^{\alpha_1} \Gamma(\alpha_1)} - A_2 \cdot \frac{n^{\alpha_2 - 1} e^{-n/\beta_2}}{\beta_2^{\alpha_2} \Gamma(\alpha_2)}. \tag{6.5}$$

where $\Gamma(\alpha)$ is the gamma function and the α, β and A parameters have been tuned to match murine fUS responses.

Voxel Grid Generation

The stimulus response s is modelled as occurring in a number of predefined regions. A binary 3D voxel grid $\underline{\mathbf{W}} \in \mathbb{R}^{X \times Y \times Z}$ of the locations associated with the stimulus response is generated using

$$\underline{\mathbf{W}} = \sum_{j \in \mathcal{J}} \underline{\mathbf{V}}_j. \tag{6.6}$$

In this equation $\underline{\mathbf{V}}_j \in \mathbb{R}^{X \times Y \times Z}$ is the binary voxel grid of the j'th of J non-overlapping regions of interest chosen and constructed from the Allen Mouse Brain Atlas [1], and $\mathcal{J} \subseteq \{1, 2, ..., J\}$ is the index set of the regions associated with the response. Assuming the fUS plane is in the first two spatial dimensions of $\underline{\mathbf{W}}$ and the slice is located at a location n_z, then the fUS measurement spatial map $\mathbf{R} \in \mathbb{R}^{X \times Y}$ is generated as

$$\mathbf{R} = \underline{\mathbf{W}} \times_3 \mathbf{f}, \tag{6.7}$$

where $\mathbf{f} \in \mathbb{R}^{1 \times Z}$ is the off-plane measurement profile vector. This is a vector which aims to model the off-plane resolution typical of fus. While more complex options are available, this function uses a simple truncated Gaussian shape, effectively performing a weighted average of all the off-plane voxels around the scan-plane. Thus equation (6.7) is a contraction over the third dimension of $\underline{\mathbf{W}}$ weighted by the vector $\mathbf{f}$ which is defined as

$$\mathbf{f} = \big[f[1], f[2], ..., f[Z]\big] \tag{6.8}$$

with

$$f[n] = (y * p)[n]. \tag{6.9}$$

Here $y[n] = \delta_{n_z}[n]$, while $p[n]$ refers to the Gaussian spatial window around the slice location to model the effect of the poor out-of-plane resolution and slice compounding found in fUS.

Optionally at this stage, if lower in-plane spatial resolution measurements than those of $\mathbf{R}$ are desired, then the two remaining spatial dimensions of $\underline{\mathbf{T}}$ can simply be sub-sampled (or interpolated for resolutions which are not multiples of the original spatial resolution). This distinction is neglected in the notation as it has no mathematical bearing beyond changing a pair of spatial dimensions (X and Y in this case).

Spatio-temporal Combination

Finally, the full 3D spatio-temporal data tensor $\underline{\mathbf{X}} \in \mathbb{R}^{X \times Y \times N}$ can be easily constructed using the outer product as

$$\underline{\mathbf{X}} = \mathbf{R} \circ \mathbf{s}. \tag{6.10}$$

This equation can be trivially extended to a multiple stimulus case,

$$\underline{\mathbf{X}} = \sum_{i=1}^{S} \mathbf{R}_i \circ \mathbf{s}_i, \tag{6.11}$$

where i simply refers to the i'th of S stimuli with its own response vector $\mathbf{s}_i$ and measurement spatial map $\mathbf{R}_i$. In combination with equations (6.1), (6.6) and (6.7) this implies that each source has its own stimulus timing vector $\mathbf{a}_i$, region index set $\mathcal{J}_i$ and voxel grid $\mathbf{W}_i$. In the rest of this work these subscripts are treated as notational clutter and neglected unless the distinction is significant.

6.1.2. Modelling Non-Idealities

This synthetic data model described so far is the ideal case, laying the groundwork for data generation, but with a number of explicit assumptions worth noting which may differ from real data. Since the spatial maps are fixed, it assumes that region locations and tissues do not change over time. The stimulus response function $h[n]$ does not vary, regardless of time, voxel, region and subject, which is known not to be the case [12]. Furthermore, all data is perfectly aligned with the atlas, since the voxel grids are generated from the atlas itself, and the measurements are entirely noiseless. These idealities and others are addressed in the following subsections.

Sensor Noise

Sensor or measurement noise is easily added to the data by adding a Gaussian white noise tensor $\underline{\mathbf{N}} \in \mathbb{R}^{X \times Y \times N}$ (scaled by c_{SNR} to match a desired SNR) to the measurement

$$\underline{\mathbf{X}} = (\mathbf{R} \circ \mathbf{s}) + \underline{\mathbf{N}}. \tag{6.12}$$

Here $\underline{\mathbf{N}}$ is constructed using

$$\underline{\mathbf{N}}_{(x,y,n)} = c_{SNR} * \mathcal{N}(0, 1), \tag{6.13}$$

where $\underline{\mathbf{N}}_{(x,y,n)}$ refers to the element in the tensor $\underline{\mathbf{N}}$ indexed at position (x, y, n).

Random Response Variation

Since the speed and magnitude of a haemodynamic response to a stimulus is not in general fixed, one can adjust the A's and β's of the CHRF randomly at each stimulus event to create stochastic stimulus response using equation (6.3) and

$$s[n] = \sum_{k=1}^{K} (\delta_{n_k} * h_k)[n], \tag{6.14}$$

instead of equation (6.4). Here h_k is as in equation (6.5) but with small random variations in the A's and β's to create changes in the magnitude and peak time after stimulus.

Region Based Response Variation

Different regions generally have different haemodynamic responses to a stimulus. This variability can be included in the model by introducing a region specific response $\mathbf{s}_j$ and combining equations (6.6), (6.7) and (6.10) to produce

$$\underline{\mathbf{X}} = \sum_{j \in \mathcal{J}} \left[(\underline{\mathbf{V}}_j \times_3 \mathbf{f}) \circ \mathbf{s}_j \right]. \tag{6.15}$$

In this form $\mathbf{s}_j$ can have a scaled, time shifted, and/or otherwise altered haemodynamic response $h_j[n]$ as the vector $\mathbf{s}_j$ is generated from

$$s_j[n] = (a * h_j)[n]. \tag{6.16}$$

It is worth noting that as the contraction with $\mathbf{f}$ in equation (6.15) does not depend on j it could be moved out of the summation, however, since it reduces the size of $\underline{\mathbf{V}}_j$ considerably while the outer product with $\mathbf{s}_j$ increases it, it is worth doing the contraction first for both computation and memory reasons.

Source Based Response Variation

In cases where there are multiple sources and the haemodynamic response to a specific stimulus is expected to be different from the CHRF one can easily add this to the model by having a source specific $h_i[n]$ to generate $s_i[n]$ as

$$s_i[n] = (a_i * h_i)[n]. \tag{6.17}$$

Rigid Misregistration

To model a case where the data is spatially not exactly aligned with the atlas (and other scans) one starts by spatially warping the voxel grid $\underline{\mathbf{W}}$ using a discrete vector field $\underline{\mathbf{T}}$. To remain consistent and since all values are discrete and dimensions are defined, conventional vector notation is not used and $\underline{\mathbf{T}}$ is simply treated as a tensor $\underline{\mathbf{T}} \in \mathbb{R}^{X \times Y \times Z \times 3}$. The voxel grid is then transformed into $\underline{\mathbf{U}} \in \mathbb{R}^{X \times Y \times Z}$ in scalar form as

$$\underline{\mathbf{U}}_{(x,y,z)} = \mathcal{I}\left(\underline{\mathbf{W}}, \underline{\mathbf{T}}_{(x,y,z)}\right), \tag{6.18}$$

or in tensor form as

$$\underline{\mathbf{U}} = \mathcal{I}\left(\underline{\mathbf{W}}, \underline{\mathbf{T}}\right), \tag{6.19}$$

where $\mathcal{I}(\underline{\mathbf{W}}, \cdot)$ is the trilinear interpolation operation, $\underline{\mathbf{U}}_{(x,y,z)}$ is the element at position (x, y, z) in the tensor $\underline{\mathbf{U}}$, and $\underline{\mathbf{T}}_{(x,y,z)} \in \mathbb{R}^3$ refers to the three-element vector at position (x, y, z) of the vector field $\underline{\mathbf{T}}$. In this case $\underline{\mathbf{T}}$ is the vector field produced by a rigid transformation with only translation and rotation. The new voxel field $\underline{\mathbf{U}}$ can be used with equations (6.7) and (6.10) to produce

$$\underline{\mathbf{X}} = (\underline{\mathbf{U}} \times_3 \mathbf{f}) \circ \mathbf{s}. \tag{6.20}$$

Tissue Deformation

Tissue deformation can be modelled as in equations (6.15) and (6.16) with the exception that $\underline{\mathbf{T}}$ is now a flow field generated by applying broad spatial Gaussian smoothing to a random vector field. This is an approximation of the elastic deformation that tissues would experience. While they seem similar, this and the rigid deformation aim at modelling very different types of non-idealities. The non-rigid deformation preserves the isotropic structure of the voxel map and models errors introduced when different scans are not registered correctly (which happens after a measurement). The tissue deformation transformation instead aims at modelling physical changes (which happen before a measurement) and mathematically does not preserve the isotropic structure of $\mathbf{W}$. However, since it is targeted at modelling tissue deformation not measurement issues the resulting measurement is still an isotropic reflection of what is being modelled physically.

Relative Recording Errors

There are issues that have not been mentioned which can occur in data when multiple measurements are taken on different subjects and would not be a problem on their own, but cause inconsistency between data tensors. These errors include truncation, different out-of-plane scan locations and temporal reference misalignment. These three issues are easy to model by: truncating the synthetic data tensor, using a different or random slice location n_z, and shifting the stimulus timings n_k by a fixed amount respectively.

Summary

While they have been presented in isolation, all these non-idealities as well as multiple stimuli can be included simultaneously in one data model as shown in

$$\underline{\mathbf{X}} = \sum_{i=1}^{S} \left[\sum_{j \in \mathcal{J}_i} \left[\left(\mathcal{I}\left(\underline{\mathbf{V}}_j, \underline{\mathbf{T}} \right) \times_3 \mathbf{f} \right) \circ \mathbf{s}_{i,j} \right] \right] + \underline{\mathbf{N}}, \tag{6.21}$$

with the vector $\mathbf{s}_{i,j}$ computed in the manner of equation (6.3), but with

$$s_{i,j}[n] = \sum_{k=1}^{K} (\delta_{n_k} * h_{i,j,k})[n]. \tag{6.22}$$

In equation (6.21) $\underline{\mathbf{T}}$ now refers to both the rigid and tissue deformation vector fields. This can be achieved by simply adding the two vector field tensors together. Additionally this transformation is now applied to each region voxel map $\underline{\mathbf{V}}_j$ separately since each region could have its own response $\mathbf{s}_{i,j}$. Finally $h_{i,j,k}[n]$ now refers to a response which can differ for any combination of region, source, and stimulus instance. A high-level illustration of the synthetic data generation process is shown in figure 6.1.

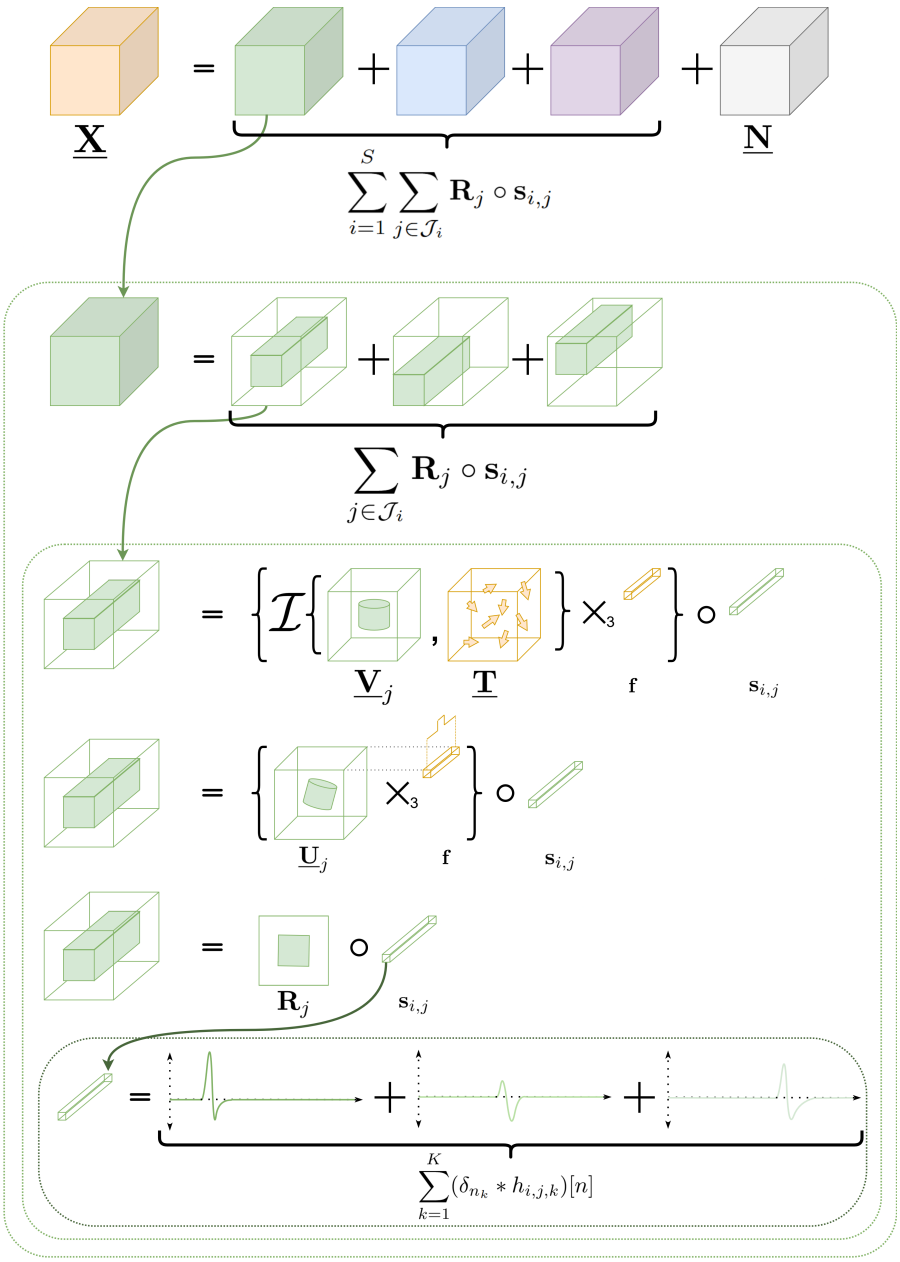


Figure 6.1: This diagram is a graphical representation of the operations shown in equations (6.21) and (6.22) used to generate synthetic fUS data.

There are still a number of issues that are present in real data which are not modelled here, including but not limited to non-white noise, motion artefacts, data corruption, undesired non-stimulus signals and physiological effects.

6.2. Subgroup Identification

Non-idealities are highly stochastic in nature, and so require extensive testing to ensure good estimates of their influence on decompositions, but looking at large numbers of realisations and judging the performance of different decompositions is not realistic. Thus this investigation is framed as a subgroup identification problem with classification accuracy estimates used as indicators for decomposition robustness. Subgroup identification can be used to separate the data into different clusters, but the large

amounts of raw PDI data and potential for misalignment make clustering the data directly ill advised. The decomposition results however are much more compact, and inherently comparable, so clustering algorithms can be applied to the components after decomposition. The components extracted by the different decompositions do not share the same structure so this section elaborates on how the components are processed and clustered for each decomposition.

6.2.1. IVA

From the IVA extracted sources, $\hat{\mathbf{B}}^{[k]}$, the SCV matrices $\hat{\mathbf{B}}_r$ can be obtained using equation (3.7). These are used to get covariance matrix estimates for each SCV

$$\hat{\mathbf{C}}_r = \hat{\mathbf{B}}_r \hat{\mathbf{B}}_r^T \in \mathbb{R}^{K \times K} \tag{6.23}$$

Two clustering algorithms, k-means and spectral clustering, are applied to the data to identify two clusters. The k-means approach is applied to a matrix which is the mean of the SCV covariance matrices, while spectral clustering operates on the sources $\hat{\mathbf{B}}^{[k]}$ directly as features for each recording k. Since the spectral clustering operates on the covariance matrices of the features it is actually also using a matrix which is similar to the SCV covariance matrix estimates, and performing k-means on extracted eigenvectors of this matrix.

6.2.2. PARAFAC2

The structure of the sources produced by PARAFAC2 are similar to those of IVA in that there are temporal or spatial sources which vary between recordings, so a similar clustering approach is applied. Before making use of the source matrices it is first worthwhile to do some intelligent component masking. Since the components are estimated using an optimization process with noise, the elements of $\mathbf{D}$ (the third mode weights) are unlikely to be exactly zero, however they may be negligibly small. This causes some issues in some clustering algorithms as they normalise the extracted components, destroying the weight information, so to preserve some of this information all values of $\mathbf{D}^{[k]}$ below some threshold τ are set to 0 to form $\mathbf{D}'^{[k]}$:

$$\mathbf{D}'^{[k]}_{(i,j)} = \begin{cases} \mathbf{D}^{[k]}_{(i,j)} & , \tau \leq \mathbf{D}^{[k]}_{(i,j)} \\ 0 & , \text{ otherwise} \end{cases} \tag{6.24}$$

with $\tau = 10^{-6}$ used. Secondly a mode-two factor matrix associated with each recording can be extracted as

$$\mathbf{F}^{[k]} = \mathbf{D}'^{[k]} (\mathbf{B}^{[k]})^T. \tag{6.25}$$

Spectral clustering directly uses these source matrices as features for each recording. The other clustering approach uses k-means and a matrix storing all the recording varying component vectors which are associated with a given component, constructed by selecting the r'th component from all recordings:

$$\mathbf{F}_r = \left[(\mathbf{e}_r \mathbf{F}^{[1]})^T, ..., (\mathbf{e}_r \mathbf{F}^{[K]})^T \right]^T, \tag{6.26}$$

where $\mathbf{e}_r = [0, ..., 0, 1, 0, ..., 0]$ is a one-hot column selection vector with a one at position r. These matrices, $\mathbf{F}_r$, can now be used in a manner similar to the SCV matrices to construct covariance matrices for each component.

$$\mathbf{C}_r = \mathbf{F}_r \mathbf{F}_r^T \in \mathbb{R}^{K \times K}. \tag{6.27}$$

K-means is then applied to the mean of these covariance matrices, while spectral clustering is applied to the original sources.

6.2.3. CPD

In contrast to IVA and PARAFAC2 the CPD structure is far simpler to apply clustering algorithms to since differences between recordings are all stored in the forth mode factor matrix $\mathbf{D}$. Thus the first clustering algorithm applies k-means directly to $\mathbf{D}$. The second clustering algorithm only uses components which have some level of spread as determined by the following metric

$$m_r = \frac{\frac{1}{K^2}\sum_{i=1}^{K}\sum_{j=1}^{K}\left|\mathbf{D}_{(i,r)} - \mathbf{D}_{(j,r)}\right|}{\max(\mathbf{D}) - \min(\mathbf{D})}. \tag{6.28}$$

This metric can range from 0 to 0.5, so a threshold of 0.3 is used, and components with a spread lower than this are ignored. K-means is also applied to the covariance matrix of the fourth mode factor matrix

$$\mathbf{C}_D = \mathbf{D}\mathbf{D}^T \in \mathbb{R}^{K\times K}. \tag{6.29}$$

Finally spectral clustering can also be applied to $\mathbf{D}$ as is.

6.2.4. Multi-shift CPD

The clustering of the multi-shift CPD is just a matter of taking the mode three weights $\mathbf{C}$ from the decomposition (equation (5.15)) and restructuring them such that all the weights for a given recording are stored in the same row, and then the same clustering algorithms used earlier (k-means and spectral clustering) can be directly applied to this matrix.

6.3. Experimental Setup

This experiment aims to ascertain how well the different clustering algorithms (and by implication the decompositions) can deal with non-idealities that naturally occur in fUS data as a result of both underlying physiological effects and the measurement paradigm itself. For this experiment the synthetic data configuration is fixed, while the impact of a single non-ideality is tested by varying the strength of its effect and applying all the different decompositions and clustering algorithms.

The shared parameters for the data generation itself are described in table 6.1 while the non-ideality parameters tested are described in table 6.2. Each run includes all the decomposition and clustering algorithms applied to synthetic data consisting of 8 recordings which are randomly allocated to one of two classes (with a minimum of 2 per class). For each non-ideality parameter at least 20 runs are performed, and the accuracy for a given clustering algorithm is averaged. All recordings in a run are generated using the same stimulus signal and the same spatial map generated from the same 6 selected regions. The difference between classes is in the response magnitudes, where class 1 has a fixed response magnitude, while class 2 has a variable response magnitude scale starting at 1.2 and decreases by 0.1 every response. The class difference is selected based on the following considerations. Having both temporal and spatial differences between classes, for instance including or excluding regions from

Table 6.1: Table describing the fixed parameters used in the isolated non-ideality experimental setup to generate synthetic fUS data.

Parameter	Value
Number of samples	220
Sampling rate	4 Hz
Off-plane resolution	850 µm
In-plane resolution	200 µm
Number of stimulus events	5

Table 6.2: Non-ideality effect strengths tested for.

Non-ideality	Effect Strengths
Physiological	
Response magnitude deviation	[0.01, 0.05, 0.1, 0.25]
Response peak variation range	[0.01,0.05,0.1,0.25,0.5]
Non-rigid voxel shift	[1, 2, 3, 4, 5, 10, 15, 20, 40]
Measurement	
Measurement noise SNR	[-20, -10, 0, 5, 10, 15, 20]
Reference shift (temporal misalignment) max	[1 2 3 4 5]
Off-plane index range	[1 5 10 20 50]
Spatial rotation variation	[1 2 3 4 5 10 15 20]
Spatial Translation variation	[1 2 3 4 5 10 20 40]

one class, would be a reasonable way for learning effects to manifest, however this simplifies the subgroup identification significantly as now there are spatial and temporal distinctions, and there is no guarantee that this will indeed be the case, leading to over-optimistic results. Significant spatial differences are also easily captured by all the different decompositions as illustrated in the previous section. Secondly, simply having entirely different signals between classes is both unrealistic, as it is expected that any learning effects would be correlated with the stimulus, and an easier problem than a variable magnitude change of the same signal. In essence the chosen type of class distinction is expected to be realistic and simultaneously challenging enough to get good indications of relative decomposition robustness.

Once the average accuracies are obtained a robustness score is calculated per non-ideality for each decomposition type. Since the topic of interest is the decomposition, not the clustering algorithms used, the best performing accuracy is selected from all the results for a single algorithm. Another important note is that the absolute minimum accuracy one can achieve in this experiment thanks to the nature of unsupervised bi-clustering is 0.5. Furthermore, the accuracy of a completely uninformed cluster allocation in a setting with random cluster allocation and a minimum allocation of 2 per cluster for an even number of recordings, n, can be calculated as follows in equation (6.30):

$$\mathbb{E}(\text{accuracy}) = \frac{\sum_{k=\frac{n}{2}+1}^{n-2} \left[\binom{n}{k} \cdot 2\frac{k}{n}\right] \; + \; \left(\binom{n}{1} + \binom{n}{0}\right) \cdot 2\frac{(n-2)}{n} \; + \; \binom{n}{\frac{n}{2}} \cdot 0.5}{2^n}, \tag{6.30}$$

which equals approximately 0.627 in the case of 8 recordings as in this experiment. This means that any results around this value are no better than random guessing. With this information a robustness score can be calculated using an area under the curve based metric. For a given non-ideality Let $\mathbf{e} = [e_1, e_2, \ldots, e_n]^\top$ be the vector of

ordered (decreasing) non-ideality effect strengths[1], and $\mathbf{s} = [s_1, s_2, \ldots, s_n]^\top$ the vector of associated accuracies for a decomposition. First adjust the accuracy vector by subtracting the uninformed accuracy to obtain:

$$\tilde{s}_i = \max(s_i - 0.627, 0), \quad \text{for } i = 1, 2, \ldots, n \tag{6.31}$$

which ranges from 0 to 0.373. Compute the area under the curve using the trapezoidal rule:

$$\text{Area}_{\text{actual}} = \sum_{i=1}^{n-1} \frac{1}{2}(e_{i+1} - e_i)(\tilde{s}_i + \tilde{s}_{i+1}) \tag{6.32}$$

For perfect robustness the maximum possible area is

$$\text{Area}_{\text{max}} = (e_n - e_1) \cdot (1 - 0.627). \tag{6.33}$$

Finally, the robustness is defined as

$$\text{Robustness} = \frac{\text{Area}_{\text{actual}}}{\text{Area}_{\text{max}}} \tag{6.34}$$

and ranges from 0 to 1, where 1 implies complete robustness to error effects and 0 the opposite.

6.4. Non-ideality Results

The full accuracy plots for all clustering algorithms may be found in appendix H and the robustness scores for all the decompositions are shown in figure 6.2. To aid in interpretation the non-idealities are grouped by type not cause, so the shifting effects are in the 2nd and 3rd columns, the response affecting non-idealities are in columns 3 and 4, and all spatial effects are in the right half of the figure (columns 5-8).

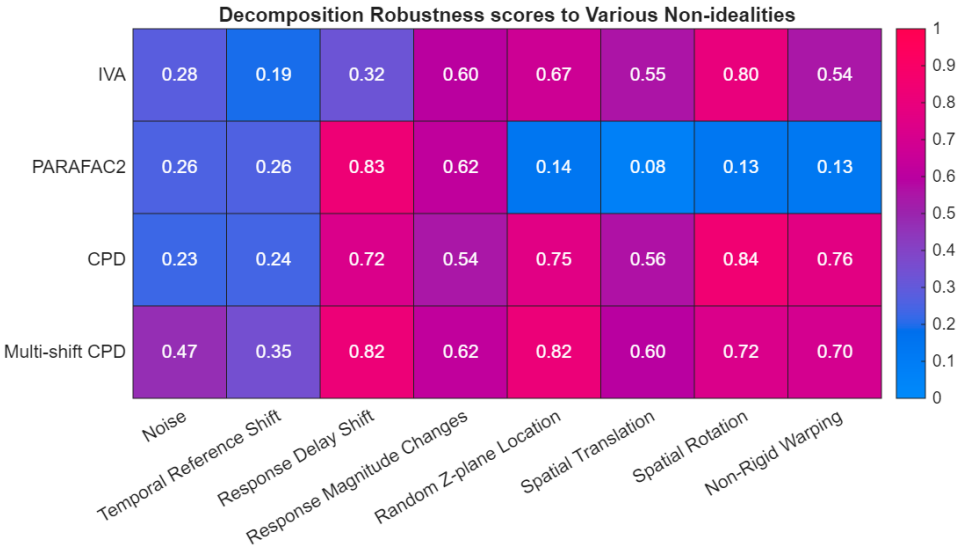


Figure 6.2: Depiction of robustness scores for different decompositions and non-idealities.

The average runtime until convergence for each decomposition algorithm is also included in plotted format for different ranks in figure 6.3.

[1] For the measurement noise case this refers to the noise magnitude not the SNR.

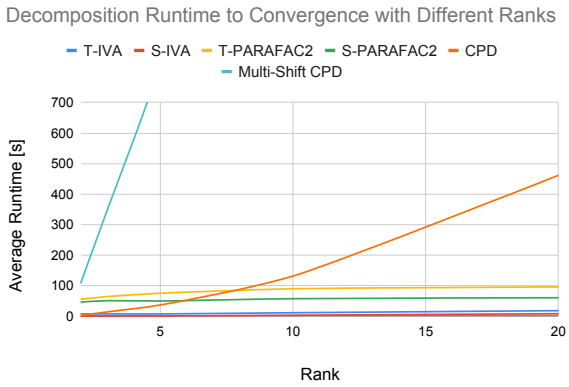


Figure 6.3: Runtime for different decomposition algorithms with different ranks. One can see that with a fixed data tensor size the IVA and PARAFAC2 algorithms' runtime has reasonably slow growth relative to the rank, while CPD continues to scale exponentially with rank increases and multi-shift CPD even more so.

6.5. Discussion

The robustness scores in figure 6.2 convey how susceptible different algorithms are to different types of issues in the data. Firstly, when interpreting this data it is important to take into account that the robustness scores are normalized by the best possible score in the range of tested effect strengths. That means that the score is inherently specific to the range over which experiments are being performed[2] and this needs to either be taken into account when looking at the score, or one must only look at the relative score between different decompositions.

There are three interesting patterns to note in these scores. To start, IVA performs fairly well in general, unless it is exposed to time shifts, as can be seen from the very poor scores showcased relative to the other decompositions in the second and third columns of figure 4.3. While better than IVA, all decompositions seem to perform exceedingly poorly in the reference shift cases, this is of course affected by the large range of shifts (0-5), but even so, looking at figure H.5 one can see that just a single sample shift nearly incapacitates all the algorithms. Lastly, the PARAFAC2 models do not seem to tolerate spatial effects in the slightest, with CPD outperforming it by a very significant margin despite also relying on shared factor matrices across components. This could be the result of the innately different clustering techniques used for CPD and PARAFAC2, but if this is the case it still implies that the results of the CPD based algorithm are easier to interpret in the presence of errors. That being said, CPD works on the best rank 1 estimate per component, meaning that it simply suppresses variations while IVA captures the variations. By forcing the variation into a single mode PARAFAC2 potentially allows for non-idealities to cross contaminate data in other modes (for example a spatial misalignment, instead of being captured in the first mode as in IVA or simply suppressed as in CPD, results in noise from other regions warping the response for a specific subject, which in turn deteriorates accuracy and interpretability).

[2]For example, even though the noise robustness score seems exceedingly poor, it is worth noting that the "distance" being integrated over from SNR -10 to SNR -20 is 68% of the entire range, a range which is sure to be have poor accuracy and as a result will severely impact the area under the curve based robustness score.

7

Application to Real Data

Finally, the decomposition algorithms are applied to a real dataset of murine fUS recordings. First, the dataset restructuring and the recording parameters are described, followed by additional preprocessing steps applied to the data. Two of the most promising decomposition algorithms are then applied to extract components: tIVA and multi-shift CPD. The other algorithms were not used owing either to poor learning effect identification or to bad non-ideality robustness.

7.1. Experimental Setup

The fUS data is processed before the algorithms are applied. Outlier detection and removal is performed on each recording, and aims to identify PDIs which have abnormal magnitudes. These outliers are typically caused by subject movement, and the regularity of occurrences in a given recording or for a given subject can vary greatly. As such an iterative algorithm based on second order magnitude statistics described in appendix C is used for outlier removal. Standardisation, which is heavily impacted by outliers, can now be performed on each voxel making them comparable, otherwise voxels containing large vasculature (which naturally has higher blood volume) dominate the PDIs and results. At this point it is also helpful but not necessary to mask regions which are not in the cerebrum itself, or are not of interest. To prevent accidental loss of information due to misaligned scans, and to aid in the identification and removal of artefacts that occur in and outside of the cerebrum on the PDIs it is chosen not to perform masking.

As seen earlier, it is important to be aware of the data structure when applying different decompositions. In the case of tIVA the model has a lot of freedom, so it can be applied to the real data with minimal restructuring as in equation (5.1):

$$\underline{\mathbf{X}}_{\text{temporal}} \in \mathbb{R}^{M \times N \times KPZ}. \tag{7.1}$$

For the multi-shift CPD algorithm though this is not advised, since it would imply that the factors are the same between subjects/recordings/slices which is not the case. There is another issue with the multi-shift CPD that has not yet been discussed: because of the shift structure alignment, the model introduces additional local minima into the loss landscape (described in more depth and illustrated in appendix F), and since this algorithm does not exhaustively search all shift alignment options like some

other algorithms, nor does it operate in the cyclic and unconstrained frequency space like SCP, it is important to ensure that good factor initialisations are chosen to avoid these local minima. To face these obstacles we first apply a CPD to the full dataset, then use the extracted components to initialise the multi-shift CPD decompositions (which are all coupled along the temporal dimension) for each slice and recording. To achieve this the full dataset tensor

$$\underline{\mathbf{X}} \in \mathbb{R}^{M \times N \times K \times P \times Z} \tag{7.2}$$

is first restructured as in equation (5.12) to have the responses on one dimension and the frame weights on a second, but without separating out the slices to form

$$\underline{\mathbf{X}}_{\mathsf{f}} \in \mathbb{R}^{M \times N_{\mathsf{f}} \times F \times K \times P \times Z}. \tag{7.3}$$

This is further restructured and flattened to contain the slices in the first mode with the voxels, and the recordings with the frames to form

$$\underline{\mathbf{X}}'_{\mathsf{f}} \in \mathbb{R}^{MZ \times N_{\mathsf{f}} \times FK \times P}, \tag{7.4}$$

since it is expected to have different frame weights for different recordings. A CPD is applied to this full tensor to extract components

$$\underline{\mathbf{X}}'_{\mathsf{f}} \approx \sum_{r=1}^{R} \mathbf{A}_{(:,r)} \circ \mathbf{B}_{(:,r)} \circ \mathbf{C}_{(:,r)} \circ \mathbf{D}_{(:,r)} \tag{7.5}$$

with the extracted factor matrices

$$\mathbf{A} \in \mathbb{R}^{MZ \times R}, \mathbf{B} \in \mathbb{R}^{N_{\mathsf{f}} \times R}, \mathbf{C} \in \mathbb{R}^{FK \times R}, \mathbf{D} \in \mathbb{R}^{P \times R}. \tag{7.6}$$

These can be restructured

$$\underline{\mathbf{A}} \in \mathbb{R}^{M \times Z \times R}, \underline{\mathbf{C}} \in \mathbb{R}^{F \times K \times R} \tag{7.7}$$

and then slices/recordings selected to form

$$\mathbf{A}^{[z]} \in \mathbb{R}^{M \times R}, \mathbf{C}^{[k]} \in \mathbb{R}^{F \times R}. \tag{7.8}$$

and

$$\mathbf{E}^{[k]} = \mathbf{C}^{[k]} \odot \mathbf{D} \in \mathbb{R}^{FP \times R} \tag{7.9}$$

Constructing a new data tensor per slice and recording as

$$\underline{\mathbf{Y}}^{[k,z]} = \underline{\mathbf{X}}_{\mathsf{f}(:,:,:,k,:,z)} \in \mathbb{R}^{M \times N_{\mathsf{f}} \times F \times P} \tag{7.10}$$

A final step is applied to $\underline{\mathbf{Y}}^{[k,z]}$ to remove artefacts, where a CPD is performed on it, and any components which have a single weight in the fourth mode larger than approximately 0.4 are removed. This means any component which is primarily contributed by a single subject is considered an artefact, iterating until there are 2 components which do not meet the artefact criteria. This artefact-removed tensor is then flattened to

$$\underline{\mathbf{Y}}'^{[k,z]} \in \mathbb{R}^{M \times N_{\mathsf{f}} \times FP}. \tag{7.11}$$

Now the multi-shift CPD is applied to $\underline{\mathbf{Y}}'^{[k,z]}$ and the three factor matrices are initialised with $\mathbf{A}^{[z]}$, $\mathbf{B}$, and $\mathbf{E}^{[k]}$ respectively. Furthermore, the second mode factors $\mathbf{B}$ are shared between all multi-shift CPDs regardless of slice or recording, effectively a joint multi-shift CPD.

7.2. Results

This section applies the decompositions to real fUS data and discusses the components of the decompositions for analysis purposes. Only the components that had some significance are included herein, and components capturing isolated noise are not shown.

7.2.1. tIVA

Several different rank IVA decompositions were run and tested, the plots are included for a few of the more illustrative components from the rank two decomposition. All the components for this particular decomposition are included in the supplementary material document under section 1.

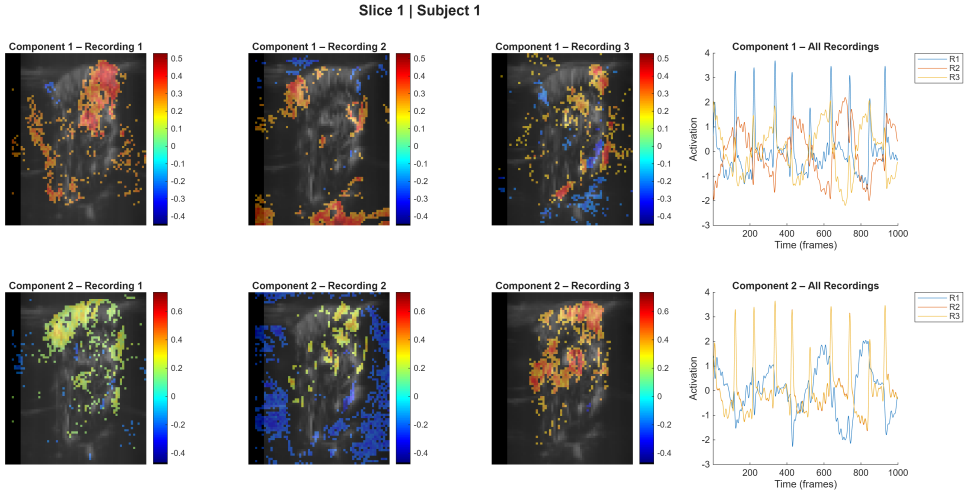


Figure 7.1: tIVA spatial maps and time-courses for Slice 1, Subject 1

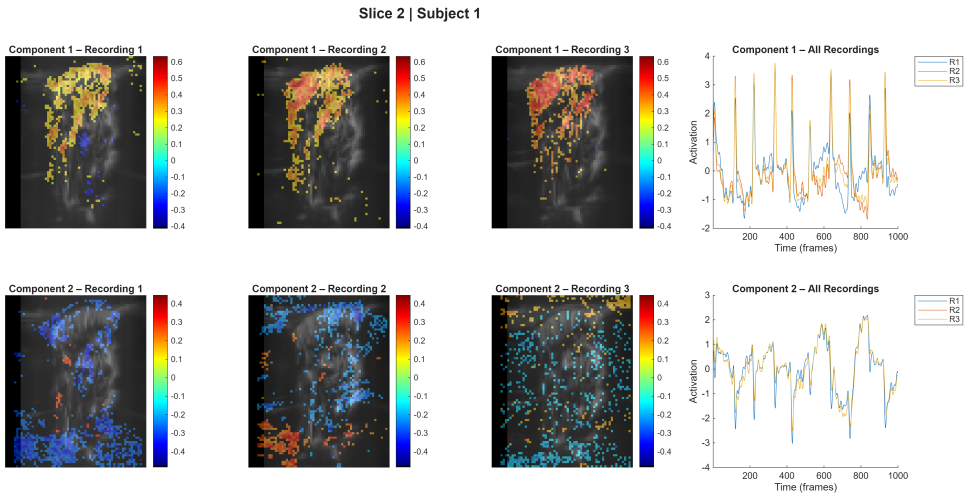


Figure 7.2: tIVA spatial maps and time-courses for Slice 2, Subject 1

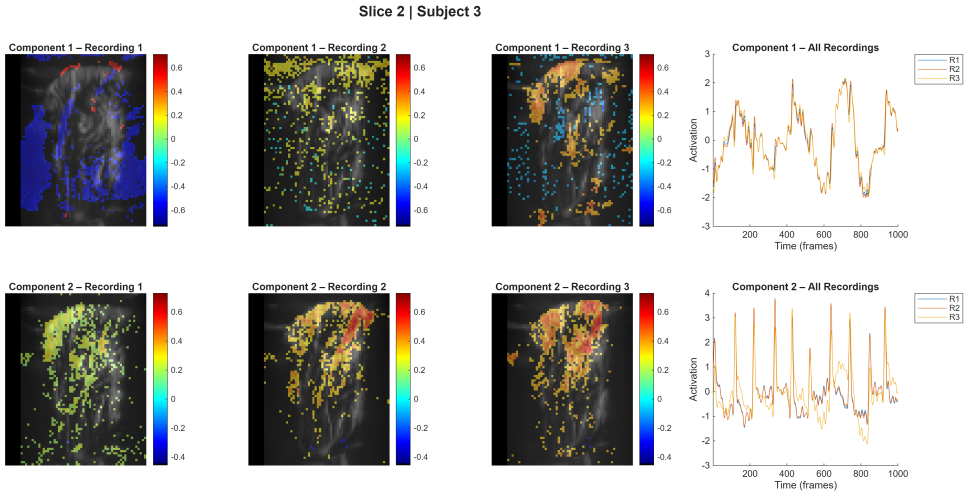


Figure 7.3: tIVA spatial maps and time-courses for Slice 2, Subject 3

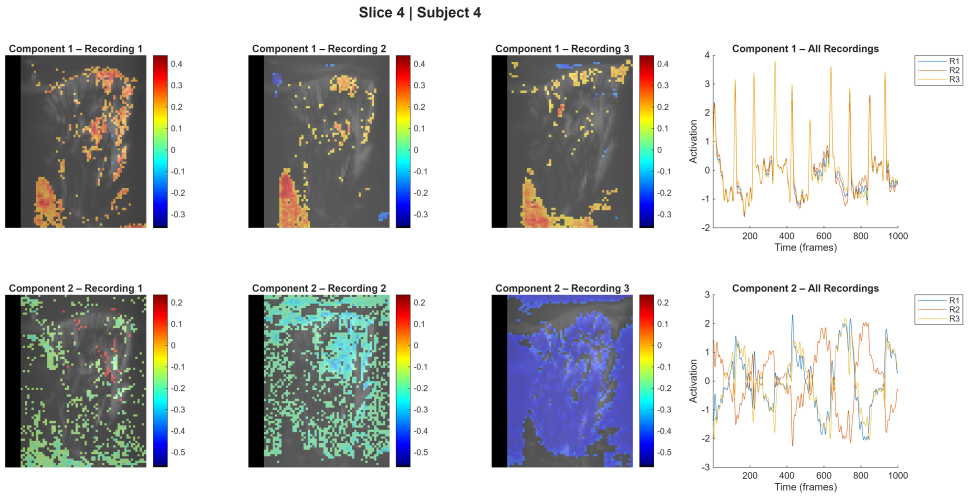


Figure 7.4: tIVA spatial maps and time-courses for Slice 4, Subject 4

7.2.2. Multi-shift CPD

This subsection details some relevant results from the multi-shift CPD algorithm. The first interesting result comes from the CPD components used to initialise a run of the multi-shift algorithm for a single recording, as it illustrates how the responses tend to have a general downward trend in magnitude within a recording session. These results are not from the same decomposition setup as the remaining results, as they specifically look at a single recording to observe trends.

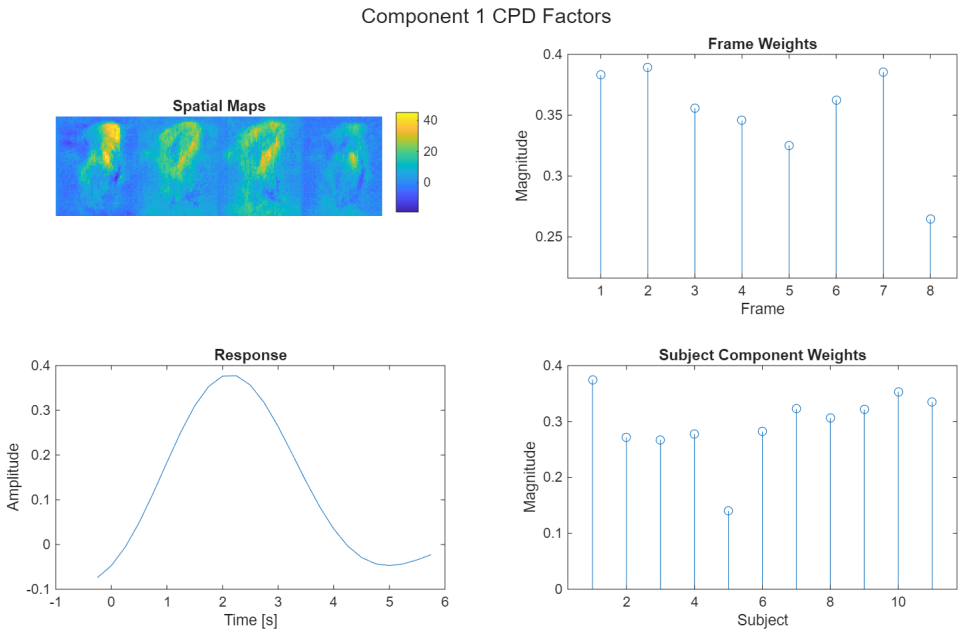


Figure 7.5: CPD components extracted for a single recording prior to applying the joint multi-shift CPD algorithm. The top right frame weights graph illustrates the typical downward trend of responses within a recording.

The following results all pertain to the same rank two multi-shift CPD based decomposition of the data, starting with the joint temporal response functions in figure 7.6. The factors associated with component one for all four slices and component two for slices three and four are shown in figures 7.7 to 7.10. The full results for this decomposition can be found in appendix I.

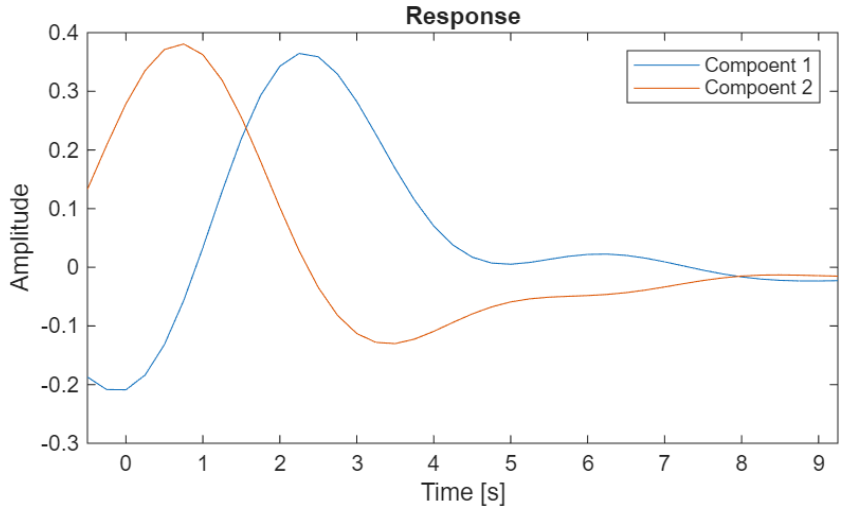


Figure 7.6: The two coupled responses for component one and two for the multi-shift CPD algorithm of rank two.

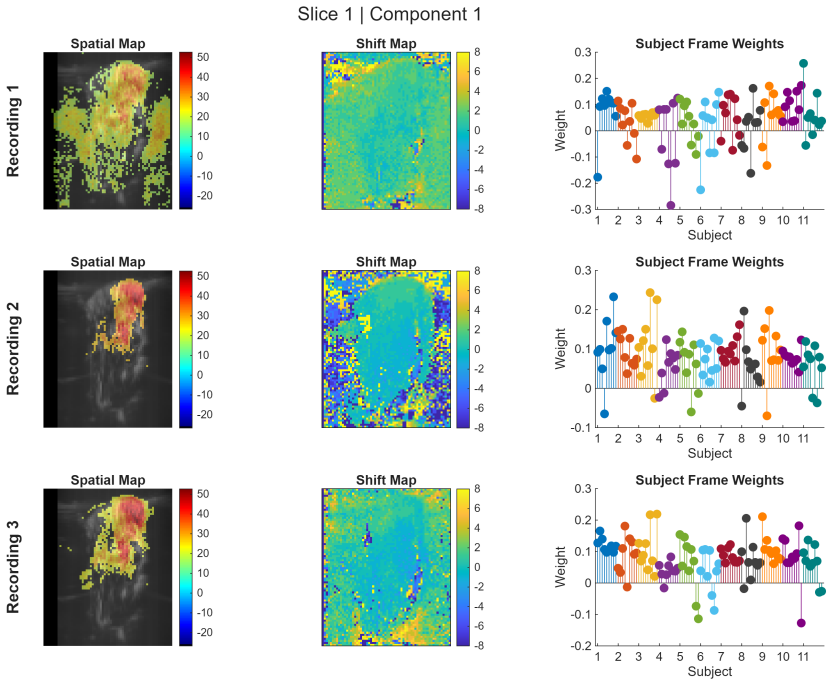


Figure 7.7: Coupled multi-shift CPD results for Slice 1, Component 1

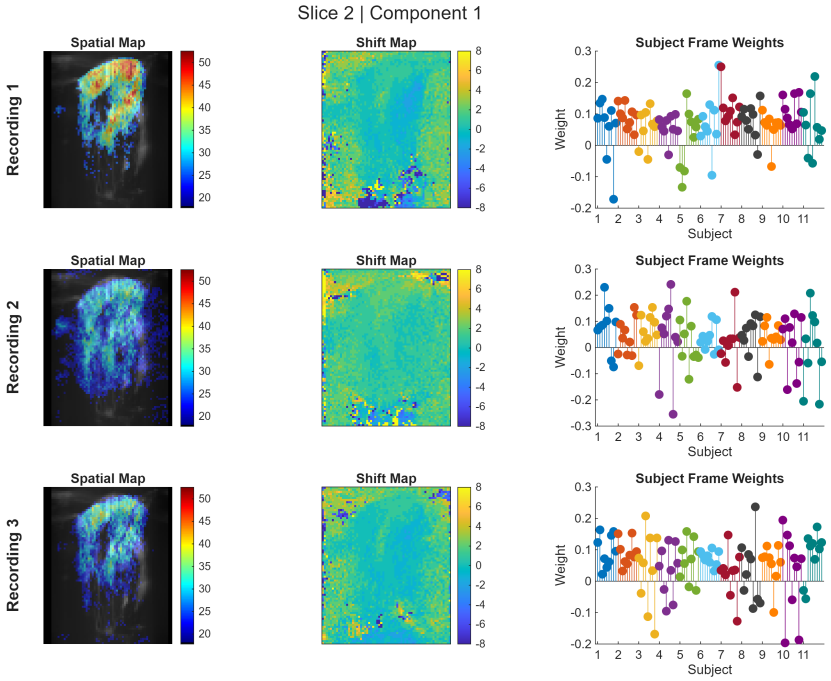


Figure 7.8: Coupled multi-shift CPD results for Slice 2, Component 1

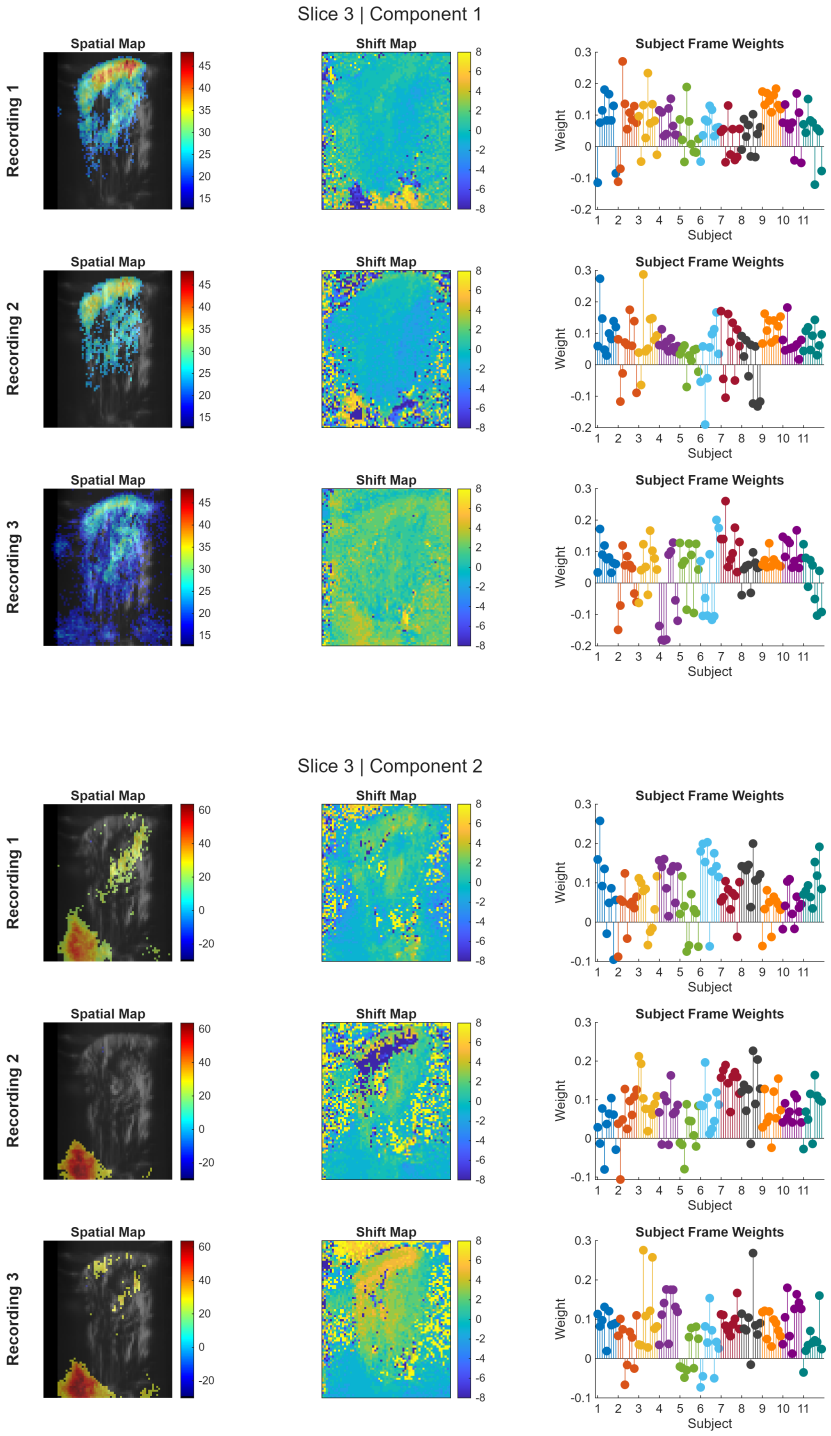


Figure 7.9: Coupled multi-shift CPD results for Slice 3, Components 1 (top) and 2 (bottom).

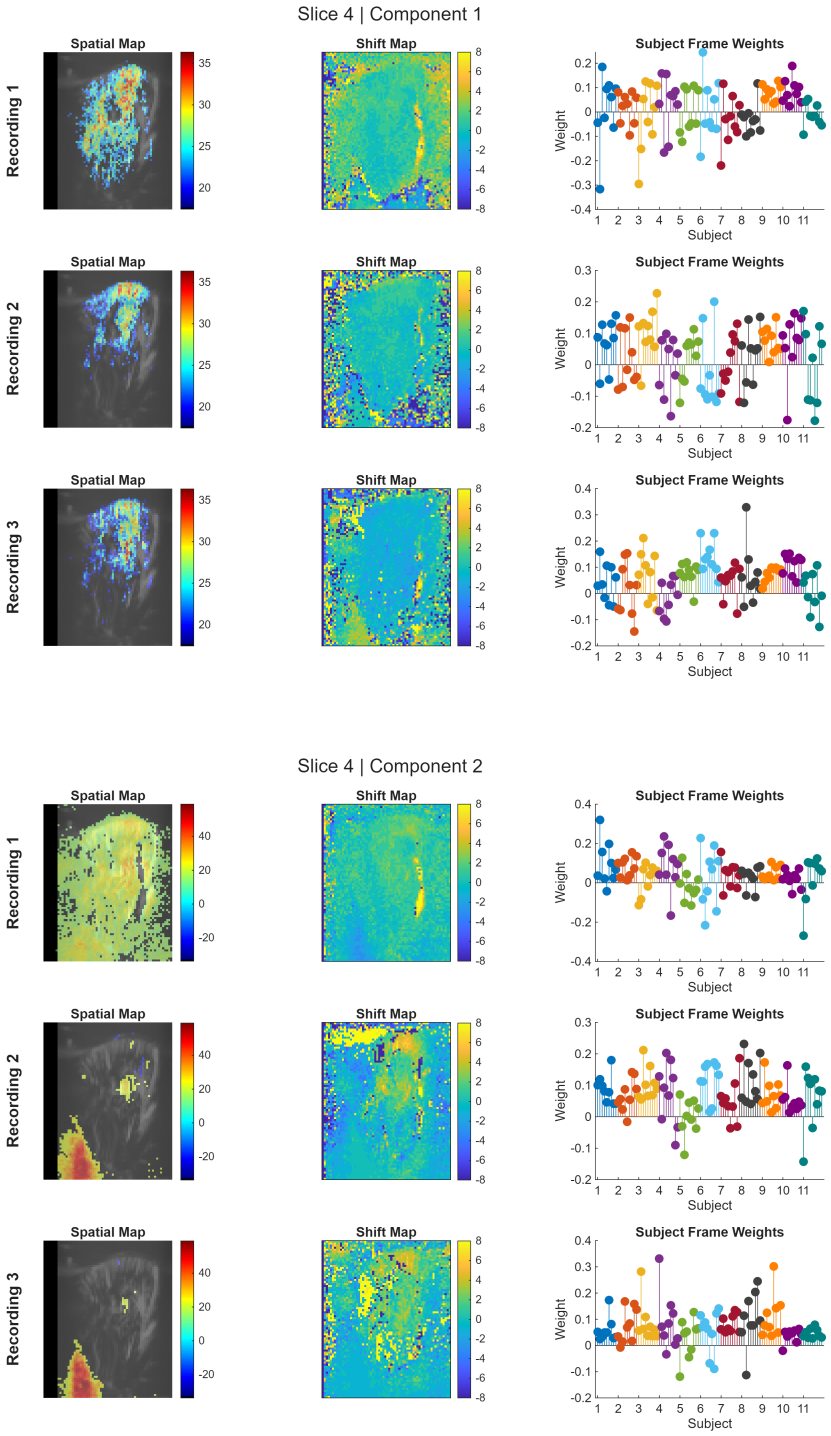


Figure 7.10: Coupled multi-shift CPD results for Slice 4, Components 1 (top) and 2 (bottom).

7.3. Discussion

This section discusses the real fUS data decomposition results, addressing first tIVA, then Multi-shift CPD, and ending with a summary comparing the outcomes.

7.3.1. tIVA

The tIVA results were rather inconclusive, identifying some of the expected responses in some of the recordings, but the extracted responses were generally very noisy, and not consistently restricted to the same component. In this instance the results depicted are from a rank-two tIVA decomposition, however additional higher rank tIVA decompositions were also performed, and did not appear to extract any additional components or identify other learning effects, and were significantly more challenging to interpret due to factors of interest not being consistently located in the same components. From the slice one figure (figure 7.1) we can see the lateral geniculate nucleus (LGN) and visual cortex response is contained in component one for the first and second recording, but in both component one and two for the third recording, so even for a single subject the stimuli responses are not restricted to a single component. The slice two components for subject one and three in figures 7.2 and 7.3 have their responses consistently in the same component when looked at in isolation, however when comparing them one can see that the stimulus response is stored in separate components. There is one interesting effect seen in the artefact shown in the bottom left of all the spatial maps in figure 7.4 which also occurs in several other subjects. This artefact is captured much more clearly by the multi-shift CPD algorithm and will be discussed further in the next subsection.

The fact that the same stimulus responses occur in multiple components even though the IVA algorithm attempts to enforce statistical independence between components can be attributed to a few different things. It could imply that there are no other meaningful signals in the data that are statistically dependent across recordings, hence the decomposition is using the components at its disposal to capture minor distinctions in the typical response. Alternatively it could be that the other signals have very small magnitudes compared to these responses and are simply masked by them. By far the most likely cause though is that due to the response shifts, which are known to be present in the data, the algorithm struggles to capture these signals in the same component as they are not necessarily dependent without some level of temporal shifting. The previous synthetic analysis also illustrated IVA's poor robustness to shift effects, and this issue is likely further exacerbated by the presence of other arbitrary signals in the data.

7.3.2. Multi-shift CPD

Compared to IVA the multi-shift CPD results are far more interpretable and several effects can be identified. Starting with the single recording CPD initialisation components shown in figure 7.5 one can see in the frame weights the effect of stimulus response habituation. This effect is a general trend though, and response magnitudes naturally vary fairly randomly, not to mention the interfering effects of other signals and artefacts further obfuscating such trends. We can see in slice-one's components in figure 7.7 the distinct isolation of the LGN and the visual cortex activation, and see that the overall magnitude of this response seems to increase with consecutive recordings, however it is worth noting that this increase could be affected by other factors such as artefacts influencing the standardisation. A benefit of the multi-shift CPD algorithm is that it directly extracts shifts per voxel, so the sequence of region activation is explicitly modelled in the shift maps. From these shift maps in figure 7.7 one can clearly see the order, from the main blood vessels, to the LGN and then to the cortex as described in literature [32]. It is of course important to recognise that due to the HRF, NRF ambigu-

ity some of this apparent sequence of activation is caused by the movement of blood through the brain, and not purely by neurological activation. Moving on, figure 7.8 shows a cortex activation which dies down over consecutive recordings, and since this is the same component as in slice one, it has a coupled response. Slice three (figure 7.9) and four (figure 7.10) is on the stimulus side, and the central slice shows the same decreasing pattern of activation as in slice two, while slice four shows that the visual cortex activation is greatly reduced compared to slice 1 as one would expect from a single sided visual stimulus. There is another very interesting artefact, also observed in the IVA components, that occurs in component two of both slices three and four – figures 7.9 and 7.10. Looking at the spatial maps, this component is not actually in the cerebrum, and appears to increase over time. Furthermore it is correlated with the stimulus, and has an exceedingly fast response, typically within 0-1 sample of the stimulus event (as can be seen from the response function for component two in figure 7.6 peaking at approximately 0.75 seconds and the shift map having a shift of -2 to -3 samples which is -0.5 to -0.75 seconds). Given the speed of the response, the location, and the strong correlation with the stimulus we theorise that this is an artefact induced by an eye-blink or some motion caused by the stimulus as this would indeed occur much faster than the haemodynamic response being measured. If this is in fact the case, then while not a neurological learning effect, it could still be useful to identify when blink based CRs occur in this experimental paradigm. It is also interesting to see that the shift maps are fairly cohesive even in voxels where there is very little response, an effect which could possibly be attributed to response shape correlations or to general haemodynamic movement.

7.3.3. Summary

In summary, while tIVA has a lot of flexibility, it failed to identify meaningful effects and the interpretation of its extracted components is challenging even for small rank decompositions. The multi-shift CPD algorithm was able to extract interpretable components from the real data and identify several effects, however aside from an intriguing artefact and two well established neurological mechanisms relating to habituation and the visual pathway no significant learning effects were identified. This shows that this algorithm at least has the potential to identify effects relating to changes in response magnitude in fUS data and can capture relative differences in region activation time while being robust to the artefacts present in such data. We attribute the lack of identified learning effects primarily to the fact that the fUS scans in this dataset do not capture the hindbrain – specifically the brainstem and cerebellum – and according to literature almost all the regions associated with the mechanism of delay eye-blink conditioned learning takes place in these two regions [32], with a small region in the midbrain also playing a role.[1]

[1] This hypothesis also appears to match the findings of the shifted stimulus correlations averaged over subjects, included in section 2 of the supplementary material.

8

Conclusion

Functional ultrasound is a neuroimaging modality with good spatio-temporal resolution that has continued to gain traction in the last decade; one of the challenges in neuroscience, and in this new modality, is identifying signatures of learning in noisy data. This work explores the ability of decomposition techniques to tackle this challenge for a specific stimulus-evoked learning fUS dataset. By first analysing the data, the non-idealities present in fUS and the possible characteristics of learning effects are described. These findings motivate and inform the development of an extension to the SCP algorithm called multi-shift CPD. This new algorithm along with several other decompositions regularly used in the neuroimaging setting are assessed using simulated data to ascertain their capacity to capture interpretable learning effects and their robustness to fUS data non-idealities. The most promising decompositions – tIVA and the developed multi-shift CPD – are applied to the data and the extracted components discussed.

8.1. Findings

The synthetic data analyses showed that different tensor decompositions differ in the effects they can extract and in their robustness to different non-idealities. Multi-shift CPD both extracted the widest variety of possible learning effects in an interpretable manner, and was the most robust to fUS non-idealities, though some decompositions were more robust to specific error types. In general, while decomposition methods can overcome non-idealities in data, there are limits to this, so good pre-processing is still vital. The computation time results showed that the effectiveness of multi-shift CPD comes at a cost, and compared to the likes of IVA and PARAFAC2 this algorithm scales very poorly as decomposition rank increases – though this result should be somewhat tempered by the fact that the multi-shift CPD model can capture effects in fewer components than would be needed by the other algorithms if the shifted structure matches the underlying data. Broadly speaking this means that in this setting it is important to consider the type of learning effects of interest, the types of errors and noise in the data, and the available computational resources when selecting the most appropriate decomposition to use.

While no signatures of learning were identified in the real data using tensor decompositions, the components that were extracted and the simulated results indicate that

these tensor decompositions have the capacity to capture such signatures, and their absence in this work is attributed to the data itself not containing such effects. In the real data IVA's flexibility led to capturing unwanted signals and reduced the interpretability of the extracted components. Multi-shift CPD, in contrast, was able to consistently identify meaningful components, including effects related to habituation, the visual pathway, and relative region activations. If the suspected reason for signatures of learning absence in the extracted components is accurate, then the developed multi-shift CPD algorithm shows potential in capturing diverse effects (signatures of learning and functional connectivity maps) in fUS data as well as other neuroimaging modalities.

8.2. Limitations

One of the main challenges with decompositions is in estimating the rank of the underlying data of interest for a specific model. This is made more challenging for models like multi-shift CPD which have multiple parameters impacting the rank, like the range of possible shifts in different modes. While various different ranks were tested for all decompositions for each experiment, it is unfeasible to test all possible combinations, which potentially influences the results for some algorithms if optimal ranks were not found.

Both the synthetic data based investigations have unavoidably limited scope, testing only selected learning effects and non-idealities respectively. For the learning effect capturing analysis it is possible that learning may exhibit in other forms or patterns than those tested, for example changing response envelopes. One of the largest issues in real fUS data is the presence of other physiological responses such as neurological activity, movement or cardiovascular changes, which do not relate to the stimulus or the learning effect. These signals can detrimentally affect the extracted components, but are challenging to model and so are not included in the non-ideality assessment. The assessment of robustness to non-idealities relies on clustering algorithms, and due to the different decompositions capturing components in different forms the clustering steps following each decomposition necessarily differ slightly, which could possibly play a role in the results.

Lastly there are factors relating to the dataset and pre-processing steps which affect the results. The most obvious limitation is that the data did not appear to contain any signatures of learning. Secondly, the performance of the decompositions is heavily contingent on the efficacy of the pre-processing steps. One illustration of this is PARAFAC2's susceptibility to spatial alignment which likely led to its poor performance and exclusion given that the dataset in question contained both tissue deformation and relied partially on manual registration.

8.3. Future Work

This work shows that tensor decomposition based techniques can extract meaningful and interpretable components, but more work is needed to verify these results, and expand the capabilities of these algorithms to accommodate more diverse datasets. Various other tensor decompositions like IVA variants and convCP should be investigated as they have the potential to exploit structures in fUS and capture other patterns, potentially leading to improved learning signature identification. Beyond additional de-

compositions the next step would be to investigate datasets which are known to contain learning effects, either by performing the delay eye-blink conditioning experiment on subjects while ensuring that the hindbrain and brainstem are captured in the data, or by changing to a paradigm where learning effects are present in the forebrain, for example trace eye-blink conditioning. Regarding the synthetic investigation: if other more complex neurological effects can be identified, modelled, and used to investigate different decomposition algorithms it could provide deeper insights into which assumptions are superior for learning effect extraction. Although there are few reliable approaches to estimating the number of components in a decomposition, work is definitely needed to investigate decomposition rank, its effect on different algorithms when used in this setting, and rigorous rank estimation techniques. On a different note, the multi-shift CPD algorithm currently only handles integer shifts, but this model could be extended to allow for non-integer signal shifts, potentially allowing for a better representation of the underlying data. Further extending multi-shift CPD to a framework that can handle more complex coupled structures (for example shared magnitudes with distinct shifts in some modes) in a programmatic way would be the next step to applying it to a wider variety of datasets.

References

[1] Allen Institute for Brain Science. *Allen Reference Atlas – Developing Mouse Brain [brain atlas]*. http://atlas.brain-map.org. RRID:SCR_002990. 2011.

[2] Matthew Anderson, Xi-Lin Li, and Tülay Adalı. "Nonorthogonal Independent Vector Analysis Using Multivariate Gaussian Model". en. In: *Latent Variable Analysis and Signal Separation*. ISSN: 1611-3349. Springer, Berlin, Heidelberg, 2010, pp. 354–361. isbn: 978-3-642-15995-4. doi: 10.1007/978-3-642-15995-4_44. url: https://link-springer-com.tudelft.idm.oclc.org/chapter/10.1007/978-3-642-15995-4_44 (visited on 06/24/2025).

[3] Rasmus Bro. "PARAFAC. Tutorial and applications". In: *Chemometrics and Intelligent Laboratory Systems* 38.2 (Oct. 1997), pp. 149–171. issn: 0169-7439. doi: 10.1016/S0169-7439(97)00032-4. url: https://www.sciencedirect.com/science/article/pii/S0169743997000324 (visited on 06/25/2025).

[4] Clément Brunner et al. "Whole-brain functional ultrasound imaging in awake head-fixed mice". en. In: *Nature Protocols* 16.7 (July 2021). Publisher: Nature Publishing Group, pp. 3547–3571. issn: 1750-2799. doi: 10.1038/s41596-021-00548-8. url: https://www.nature.com/articles/s41596-021-00548-8 (visited on 08/11/2025).

[5] T. Tony Cai and Rong Ma. *Theoretical Foundations of t-SNE for Visualizing High-Dimensional Clustered Data*. arXiv:2105.07536 [stat]. Nov. 2022. doi: 10.48550/arXiv.2105.07536. url: http://arxiv.org/abs/2105.07536 (visited on 07/27/2025).

[6] Vince D. Calhoun et al. "A method for making group inferences using independent component analysis of functional MRI data: Exploring the visual system". In: *NeuroImage* (2001), p. 88.

[7] Huafu Chen, Dezhong Yao, and Zuxiang Liu. "A comparison of Gamma and Gaussian dynamic convolution models of the fMRI BOLD response". In: *Magnetic resonance imaging* 23 (Feb. 2005), pp. 83–88. doi: 10.1016/j.mri.2004.11.002.

[8] G. Cittadini and C. Martinoli. "[Ultrasound and the bone: a difficult relationship]". ita. In: *La Radiologia Medica* 89.1-2 (1995), pp. 12–17. issn: 0033-8362.

[9] Lieven De Lathauwer. "Decompositions of a Higher-Order Tensor in Block Terms—Part II: Definitions and Uniqueness". In: *SIAM Journal on Matrix Analysis and Applications* 30.3 (2008), pp. 1033–1066. doi: 10.1137/070690729. eprint: https://doi.org/10.1137/070690729. url: https://doi.org/10.1137/070690729.

[10] Charlie Demené et al. "Spatiotemporal Clutter Filtering of Ultrafast Ultrasound Data Highly Increases Doppler and fUltrasound Sensitivity". In: *IEEE Transactions on Medical Imaging* 34.11 (Nov. 2015), pp. 2271–2285. issn: 1558-254X. doi: 10.1109/TMI.2015.2428634. url: https://ieeexplore.ieee.org/document/7098422 (visited on 07/21/2025).

[11] Yuhui Du et al. "A survey of brain functional network extraction methods using fMRI data". In: *Trends in Neurosciences* 47.8 (Aug. 2024), pp. 608–621. issn: 0166-2236. doi: 10.1016/j.tins.2024.05.011. url: https://www.sciencedirect.com/science/article/pii/S0166223624000912 (visited on 08/04/2025).

[12] Daniel A. Handwerker, John M. Ollinger, and Mark D'Esposito. "Variation of BOLD hemodynamic responses across subjects and brain regions and their effects on statistical analyses". eng. In: *NeuroImage* 21.4 (Apr. 2004), pp. 1639–1651. issn: 1053-8119. doi: 10.1016/j.neuroimage.2003.11.029.

[13] R. A. Harshman. "PARAFAC2: Mathematical and technical notes". In: *UCLA Working Papers in Phonetics* 22 (1972), pp. 30–44.

[14] B Horwitz, K. J Friston, and J. G Taylor. “Neural modeling and functional brain imaging: an overview”. In: *Neural Networks* 13.8 (Nov. 2000), pp. 829–846. issn: 0893-6080. doi: 10.1016/S0893-6080(00)00062-9. url: https://www.sciencedirect.com/science/article/pii/S0893608000000629 (visited on 07/27/2025).

[15] Sana Hussain et al. “A Novel Hidden Markov Approach to Studying Dynamic Functional Connectivity States in Human Neuroimaging”. In: *Brain Connectivity* 13.3 (Apr. 2023), pp. 154–163. issn: 2158-0014. doi: 10.1089/brain.2022.0031. url: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10079241/ (visited on 07/27/2025).

[16] Taesu Kim, Torbjørn Eltoft, and Te-Won Lee. “Independent Vector Analysis: An Extension of ICA to Multivariate Components”. en. In: *Independent Component Analysis and Blind Signal Separation*. ISSN: 1611-3349. Springer, Berlin, Heidelberg, 2006, pp. 165–172. isbn: 978-3-540-32631-1. doi: 10.1007/11679363_21. url: https://link-springer-com.tudelft.idm.oclc.org/chapter/10.1007/11679363_21 (visited on 04/22/2025).

[17] Li-Dan Kuang et al. “Coupled canonical polyadic decomposition of multi-group fMRI data with spatial reference and orthonormality constraints”. In: *Biomedical Signal Processing and Control* 80 (Feb. 2023), p. 104232. issn: 1746-8094. doi: 10.1016/j.bspc.2022.104232. url: https://www.sciencedirect.com/science/article/pii/S1746809422006863 (visited on 07/27/2025).

[18] Ruikai Li et al. “Design and implementation of high sampling rate and multichannel wireless recorder for EEG monitoring and SSVEP response detection”. English. In: *Frontiers in Neuroscience* 17 (June 2023). Publisher: Frontiers. issn: 1662-453X. doi: 10.3389/fnins.2023.1193950. url: https://www.frontiersin.org/journals/neuroscience/articles/10.3389/fnins.2023.1193950/full (visited on 07/27/2025).

[19] Martin A. Lindquist et al. “Modeling the Hemodynamic Response Function in fMRI: Efficiency, Bias and Mis-modeling”. In: *Neuroimage* 45.1 Suppl (Mar. 2009), S187–S198. issn: 1053-8119. doi: 10.1016/j.neuroimage.2008.10.065. url: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3318970/ (visited on 05/19/2025).

[20] Emilie Macé et al. “Functional ultrasound imaging of the brain”. In: *Nature methods* 8 (July 2011), pp. 662–4. doi: 10.1038/nmeth.1641.

[21] *MATLAB version 25.1.0.2943329 (R2025a)*. The Mathworks, Inc. Natick, Massachusetts, 2024.

[22] Leland McInnes, John Healy, and James Melville. *UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction*. arXiv:1802.03426 [stat]. Sept. 2020. doi: 10.48550/arXiv.1802.03426. url: http://arxiv.org/abs/1802.03426 (visited on 07/27/2025).

[23] Martin J McKeown, Lars Kai Hansen, and Terrence J Sejnowski. “Independent component analysis of functional MRI: what is signal and what is noise?” In: *Current opinion in neurobiology* 13.5 (Oct. 2003), pp. 620–629. issn: 0959-4388. doi: 10.1016/j.conb.2003.09.012. url: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2925426/ (visited on 07/27/2025).

[24] Morten Mørup, Lars Kai Hansen, and Kristoffer Hougaard Madsen. “Modeling latency and shape changes in trial based neuroimaging data”. In: *2011 Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*. ISSN: 1058-6393. Nov. 2011, pp. 439–443. doi: 10.1109/ACSSC.2011.6190037. url: https://ieeexplore.ieee.org/document/6190037 (visited on 07/27/2025).

[25] Morten Mørup et al. “Shift-invariant multilinear decomposition of neuroimaging data”. In: *NeuroImage* 42.4 (Oct. 2008), pp. 1439–1450. issn: 1053-8119. doi: 10.1016/j.neuroimage.2008.05.062. url: https://www.sciencedirect.com/science/article/pii/S1053811908007167 (visited on 07/15/2025).

[26] K. M. Petersson, C. Elfgren, and M. Ingvar. “Learning-related effects and functional neuroimaging”. eng. In: *Human Brain Mapping* 7.4 (1999), pp. 234–243. issn: 1065-9471. doi: 10.1002/(sici)1097-0193(1999)7:4<234::aid-hbm2>3.0.co;2-o.

[27] Russell A. Poldrack. “Region of interest analysis for fMRI”. In: *Social Cognitive and Affective Neuroscience* 2.1 (Mar. 2007), pp. 67–70. issn: 1749-5016. doi: 10.1093/scan/nsm006. url: https://doi.org/10.1093/scan/nsm006 (visited on 08/04/2025).

[28] Athanasios A. Rontogiannis, Eleftherios Kofidis, and Paris V. Giampouras. “Block-Term Tensor Decomposition: Model Selection and Computation”. In: *IEEE Journal of Selected Topics in Signal Processing* 15.3 (Apr. 2021). arXiv:2002.09759 [math], pp. 464–475. issn: 1932-4553, 1941-0484. doi: 10.1109/JSTSP.2021.3051488. url: http://arxiv.org/abs/2002.09759 (visited on 02/25/2025).

[29] R. Saban. “[The beginnings of physiology of the human brain, from antiquity to the Renaissance]”. fre. In: *Vesalius: Acta Internationales Historiae Medicinae* 5.1 (June 1999), pp. 41–47. issn: 1373-4857.

[30] Iván Sánchez Fernández and Jurriaan M. Peters. “Machine learning and deep learning in medicine and neuroimaging”. In: *Annals of the Child Neurology Society* 1.2 (2023), pp. 102–122. doi: https://doi.org/10.1002/cns3.5. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/cns3.5. url: https://onlinelibrary.wiley.com/doi/abs/10.1002/cns3.5.

[31] Andrews Sobral. *Matlab Tensor Tools*. 2014. url: https://github.com/andrewssobral/mtt/.

[32] Kaori Takehara-Nishiuchi. “The Anatomy and Physiology of Eyeblink Classical Conditioning”. en. In: *Behavioral Neuroscience of Learning and Memory*. ISSN: 1866-3389. Springer, Cham, 2016, pp. 297–323. isbn: 978-3-319-78757-2. doi: 10.1007/7854_2016_455. url: https://link-springer-com.tudelft.idm.oclc.org/chapter/10.1007/7854_2016_455 (visited on 02/18/2025).

[33] Alaa Tharwat. “Independent component analysis: An introduction”. In: *Applied Computing and Informatics* 17.2 (Jan. 2020). Publisher: Emerald Publishing Limited, pp. 222–249. issn: 2210-8327. doi: 10.1016/j.aci.2018.08.006. url: https://doi.org/10.1016/j.aci.2018.08.006 (visited on 06/24/2025).

[34] Eelco F. M. Wijdicks. “The First CT Scan of the Brain: Entering the Neurologic Information Age”. en. In: *Neurocritical Care* 28.3 (June 2018). Company: Springer Distributor: Springer Institution: Springer Label: Springer Number: 3 Publisher: Springer New York, pp. 273–275. issn: 1556-0961. doi: 10.1007/s12028-017-0495-3. url: https://link-springer-com.tudelft.idm.oclc.org/article/10.1007/s12028-017-0495-3 (visited on 02/12/2025).

[35] Huiwen Yu and Rasmus Bro. “PARAFAC2 and local minima”. In: *Chemometrics and Intelligent Laboratory Systems* 219 (Dec. 2021), p. 104446. issn: 0169-7439. doi: 10.1016/j.chemolab.2021.104446. url: https://www.sciencedirect.com/science/article/pii/S0169743921002148 (visited on 06/25/2025).

A

Functional Ultrasound Analysis Methods

Inspecting the large volumes of fUS data directly is not practical, so a variety of processing and visualisation methods are used. This is not an exhaustive list, but includes some of the most useful or atypical methods which would not be defined elsewhere.

A.1. Correlation Analysis

The first and simplest option is a per-voxel correlation over the data. Several configurations of data and signals were used with different extracted results. Separate data setups included averages over subjects, over recordings, and without any averaging. Different signals included the binary stimulus signal, and the binary stimulus signal convolved with a generated canonical haemodynamic response function, both correlated with the data at different shifts.

A.2. Mean PDI Magnitude Inspection

A simple step which was used primarily to investigate the magnitude outlier artifacts discussed in appendix C was to average all voxels in a PDI across all PDIs in a recording, and then inspect the resulting ”time-course” of average magnitudes. These magnitude signals were also cross-correlated with the stimulus signal, and it was found that, especially in some subjects, there appeared to be a link between stimulus events and magnitude spikes, speculated to be caused by a fear-induced movement response.

A.3. Spatial Gradient Averaging

Computing the spatial gradients of the per-voxel standardized PDIs and averaging these gradient maps for whole recordings can give some insight into the region boundaries and spatial artifacts in the data. These gradient maps are computed using simple central difference gradient kernels $[-\frac{1}{2}, 0, \frac{1}{2}]$ and $[-\frac{1}{2}, 0, \frac{1}{2}]^T$.

A.4. Independent Component Analysis

The ICA algorithm discussed in chapter 3 is used to extract and inspect the different spatial and temporal components in each region. The fairly weak assumptions this

algorithm imposes on the data and the increased degrees of freedom that it affords makes it very appropriate for data inspection during exploratory stages when little is known about the underlying structure.

A.5. Parcellation - Vector Clustering

To identify unified regions and the associated time courses we developed a fairly simple clustering algorithm referred to here as vector clustering. Within the context of literature it would perhaps be better referred to as an adapted fuzzy c-means with hard allocation and optional spatial smoothing, but since this is very wordy, we opt to use vector clustering in this work. This algorithm solves a constrained minimization problem to decompose the data into two factor matrices. This minimization is formulated as

$$\begin{aligned} &\min_{\mathbf{A},\mathbf{S}} \|\mathbf{X} - \mathbf{A}\mathbf{S}\| \\ &\text{s.t.} \quad \forall i \in \{1, \ldots, m\}, \quad \|\mathbf{A}_{(i,:)}\|_0 \leq 1, \end{aligned}$$

and may be solved using an alternating approach of correlation, projection, and linear least squares – repeating until convergence. The steps are described in the algorithm block, algorithm 1.

Algorithm 1 Vector Clustering

Input: $\mathbf{X} \in \mathbb{R}^{m \times n}$, r
Output: $\mathbf{A} \in \mathbb{R}^{m \times r}$, $\mathbf{S} \in \mathbb{R}^{r \times n}$
Initialize $\mathbf{S}_{(i,:)} \in \mathbb{R}^n$, $\|\mathbf{S}_{(i,:)}\|_2 = 1$ for $i = 1, \ldots, r$
while not converged **do**
  $\mathbf{A} \leftarrow \mathbf{S}\mathbf{X}^T$
  $\mathbf{A}_{(i,:)} \leftarrow \mathbf{A}_{(i,:)} \circledast \mathbf{1}_{j=\arg\max_j |\mathbf{A}_{(i,j)}|}$ for $i = 1, \ldots, m$
  $\mathbf{S} \leftarrow (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{X}$
  $\mathbf{S}_{(i,:)} \leftarrow \mathbf{S}_{(i,:)} / \|\mathbf{S}_{(i,:)}\|_2$ for $i = 1, \ldots, r$
end while

This version does not include spatial smoothing to promote region cohesion, but this may be simply added as in algorithm 2.

Algorithm 2 Vector Clustering with Spatial Smoothing

Input: $\mathbf{X} \in \mathbb{R}^{m \times n}$, r, $x, y \in \mathbb{N}$ s.t. $m = xy$, $\sigma \in \mathbb{R}_+$
Output: $\mathbf{A} \in \mathbb{R}^{m \times r}$, $\mathbf{S} \in \mathbb{R}^{r \times n}$
Define G_σ: Gaussian smoothing filter with standard deviation σ
Initialize $\mathbf{S}_{(i,:)} \in \mathbb{R}^n$, $\|\mathbf{S}_{(i,:)}\|_2 = 1$ for $i = 1, \ldots, r$
while not converged **do**
  $\mathbf{A} \leftarrow \mathbf{S}\mathbf{X}^T$
  $\mathbf{A}^{\text{smooth}}_{(:,j)} \leftarrow \text{vec}\left(G_\sigma * \text{reshape}(\mathbf{A}_{(:,j)}, x, y)\right)$ for $j = 1, \ldots, r$
  $\mathbf{A}_{(i,:)} \leftarrow \mathbf{A}_{(i,:)} \circledast \mathbf{1}_{j=\arg\max_j |\mathbf{A}^{\text{smooth}}_{(i,j)}|}$ for $i = 1, \ldots, m$
  $\mathbf{S} \leftarrow (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{X}$
  $\mathbf{S}_{(i,:)} \leftarrow \mathbf{S}_{(i,:)} / \|\mathbf{S}_{(i,:)}\|_2$ for $i = 1, \ldots, r$
end while

B

Non-Idealities in Functional Ultrasound

This appendix describes the various non-idealities that were identified in the fUS dataset. These effects are categorised into physiological or measurement based on their cause, and several illustrative examples of specific effects are included in figures B.1 to B.6.

B.1. Physiological

- Response fluctuations - Figure B.1 shows that even when averaged with a sliding window, the overall response profile varies both in terms of magnitude and in terms of delay.
- Tissue deformation - Figure B.2 shows the mean log PDIs of two different subjects and how, even with the same spatial resolution in scanning, they can exhibit rather significant differences due to the deformation of brain tissue, making comparison challenging.
- Extraneous signals - Looking at the average visual cortex response in figure B.3 shows that it is clearly not only the stimulus response present in the signal.

B.2. Measurement

- Measurement Noise - The measurement noise can be seen in the form of the speckled effects present in the regions of lower intensity shown in figure B.4.
- Movement artefacts - The movement artefacts are discussed in more depth in appendix C.
- Reference misalignment - It was found that a group of the recordings had a consistent average offset in response time when compared to the other recordings. This was corrected upon discovery by simply aligning the stimulus signal to be consistent across the whole dataset.
- Scan plane consistency - The exact slice being scanned is not always the same between recordings and subjects, meaning that there are potentially differences in the neurological structures captured in different slices. This issue is illustrated

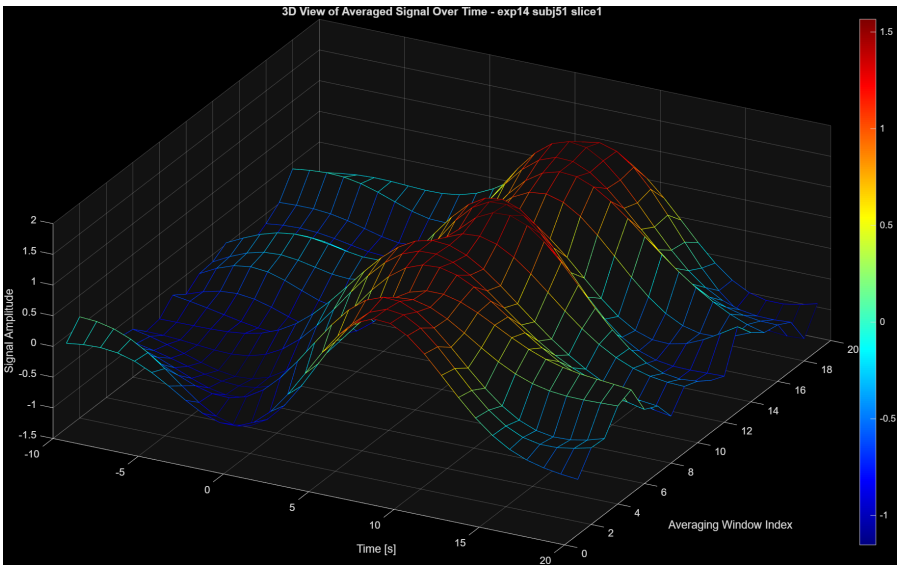


Figure B.1: Responses to the stimulus from a parcellated region averaged over several responses with a sliding window to reduce the impact of extreme responses introduced by artefacts or other signals and identify possible trends.

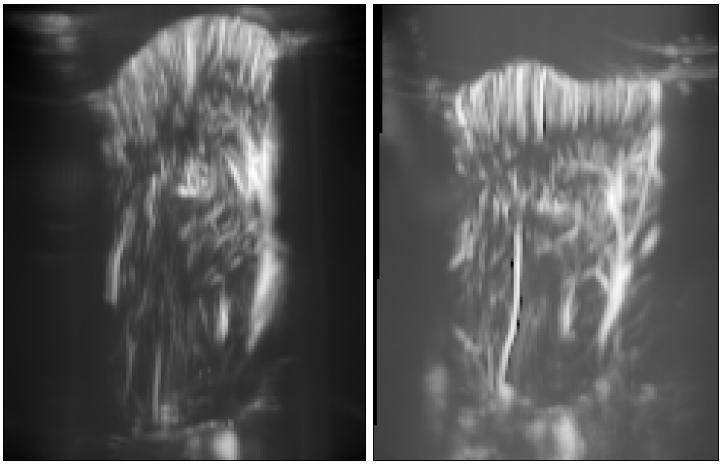


Figure B.2: Mean log PDI images for two different subjects.

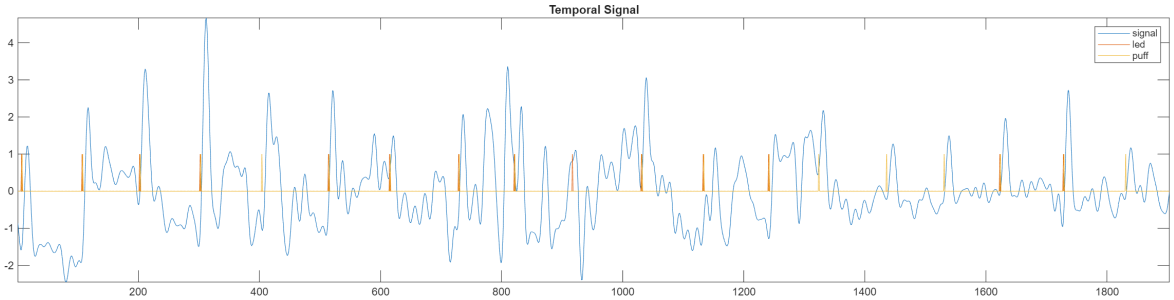


Figure B.3: Average signal from the visual cortex after parcellation illustrating that the data contains various other signals other than just the stimulus responses.

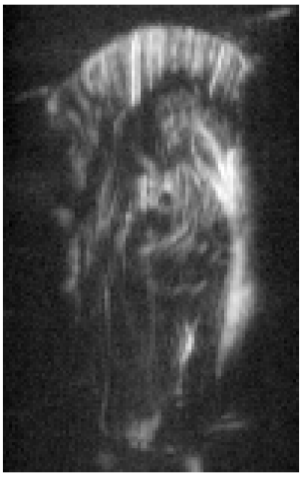


Figure B.4: Log of a single PDI illustrating the presence of sensor noise throughout scans as can be seen by the speckled effect in the dark regions of the image.

in the two images in figure B.5, and while this cannot be remedied through processing the high off-plane resolution of fUS should compensate somewhat.

- In-plane misalignment - Misaligned scans are a common problem in neuroimaging, and the registration issues and steps are discussed more in appendix D.
- Ultrasound specific effects - There are a variety of artefacts that can be present in fUS caused by the properties of the medium as the ultrasound waves propagate through them. A suspected artefact of this nature, identified by the vector clustering algorithm described in appendix A, is shown in figure B.6.

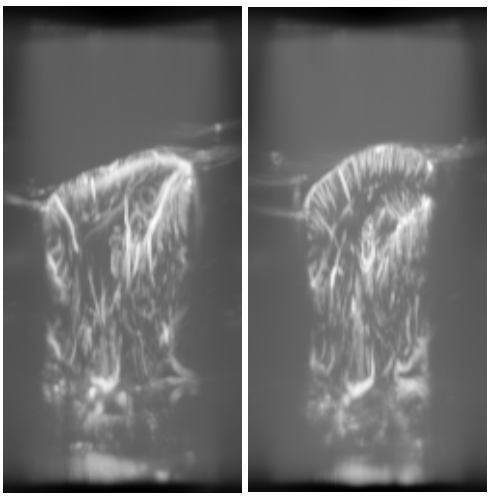


Figure B.5: Two log mean PDI images of different subjects for approximately the same slice location in the off-plane axis showing that there are differences in the vascular structure beyond just location, despite the slice locations being in the same approximate area.

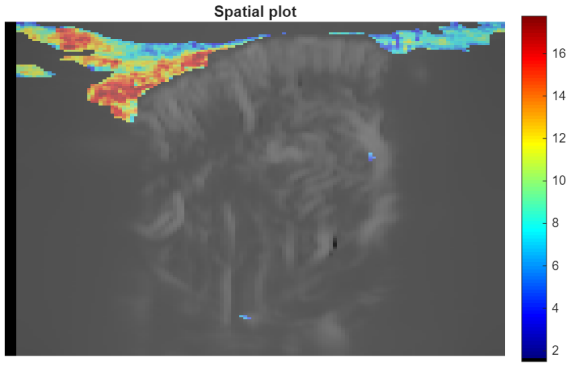


Figure B.6: Spatial mask of a parcellated region which is located outside the cranium; it is suspected to be the result of a fUS-specific artefact.

C

Outlier Detection Algorithm

The (suspected) movement artefacting is a regular issue that affects many of the recordings in this dataset. The focus on Doppler effects in fUS means that movement induces massive increases in signal magnitudes over entire PDIs (potentially in the order of 10^3 times larger than at rest). These artefacts can be exceedingly detrimental as they destroy all other signals in a recording if not removed before the standardisation step. Furthermore, these artefacts can vary wildly in magnitude, and differ in regularity between subjects. Some subjects exhibit no movement artefacts at all, and some exhibit movement artefacts in as much as 5-10% of all PDIs in a recording. To account for these characteristics, an iterative standard deviation-based mean PDI threshold is applied to each recording. This algorithm is described in algorithm 3.

Algorithm 3 Iterative Outlier Thresholding

Input: Sequence of PDIs $\underline{\mathbf{P}} \in \mathbb{R}^{X \times Y \times N}$, parameter b
$\mathcal{O}_{\text{new}}, \mathcal{O} \leftarrow \emptyset$
$\mathcal{N} \leftarrow \{1, 2, \ldots, N\}$
For each $n \in \mathcal{N}$, let $\bar{\mathbf{p}}_{(n)} \leftarrow$ mean of $\underline{\mathbf{P}}_{(:,:,n)}$
repeat
  $\mathcal{O} \leftarrow \mathcal{O}_{\text{new}}$
  $\mathcal{V} \leftarrow \mathcal{N} \setminus \mathcal{O}$
  $\mu \leftarrow \frac{1}{|\mathcal{V}|} \sum_{n \in \mathcal{V}} \bar{\mathbf{p}}_{(n)}$
  $\sigma \leftarrow \sqrt{\frac{1}{|\mathcal{V}|} \sum_{n \in \mathcal{V}} (\bar{\mathbf{p}}_{(n)} - \mu)^2}$
  $\delta \leftarrow \mu + b \cdot \sigma$
  $\mathcal{O}_{\text{new}} \leftarrow \{n \in \mathcal{N} \mid \bar{\mathbf{p}}_{(n)} > \delta\}$
until $\mathcal{O}_{\text{new}} = \mathcal{O}$
Output: Outlier index set $\mathcal{O}$

Testing on recordings without movement artefacts showed that all signals fell within 3 to 4 σ of the mean, so the higher end of this, $b = 4$, was chosen to ensure that relevant signals are not removed in recordings where signal drift is significant. The outlier PDIs were removed and data was interpolated on a pixel-by-pixel level from the surrounding non-outlier PDIs. Figure C.1 illustrates two depictions of the mean spatial magnitude of PDIs in a recording at different magnitude scales with the identified threshold and detected outliers.

There are alternative approaches that could be used for both the outlier detection and

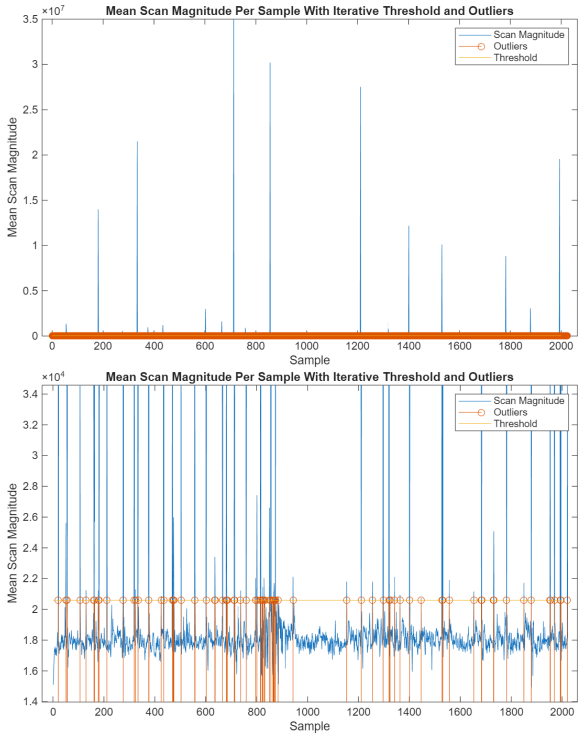


Figure C.1: Illustrative plot of mean scan magnitude and outliers with significant movement artefacts at two different y-ranges showing that the artefacts vary significantly in magnitude.

the interpolation steps. Using an interquartile range based outlier detection would likely work to identify these outliers, and would deal better with the asymmetry of the distribution around the mean caused by the power-based nature of the signal. In terms of what to do with the outliers, an alternative to interpolation would be to simply scale the outlying PDI to an appropriate interpolated magnitude, since these PDIs still partially contain relevant information, simply scaled to a much higher degree (shown in the log PDIs in figure C.2). However, there are generally also other artefacts aside from the scale change in the specific outlier PDI (second image in figure C.2), so the advantages and disadvantages of such a choice would need to be considered based on the application, the regularity of the outliers, and the information contained in these PDIs. Finally, to improve outlier detection results, one could either first perform more extensive drift removal and decrease b, or perform outlier detection with a smaller b on smaller sliding windows of the full recording.

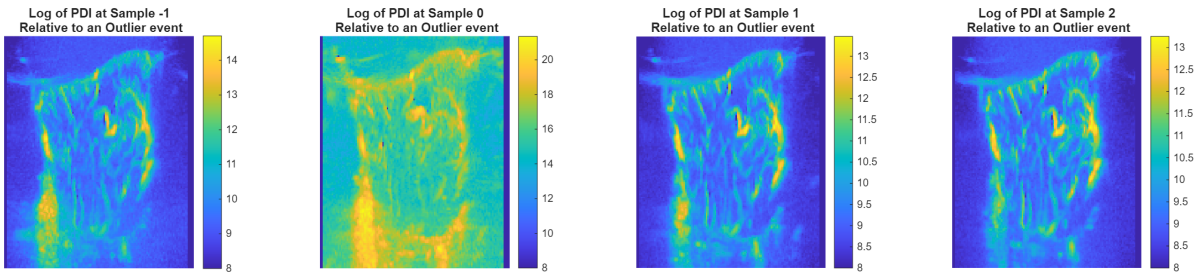


Figure C.2: Illustration of log of scans surrounding a magnitude outlier event.

D

Registration Procedure

To be able to compare and decompose the fUS data it is important to ensure that PDIs between recordings and subjects are as spatially aligned as possible. However, in cases where there are no pre-scan references, this task comes with several challenges. Firstly, fUS PDIs contain primarily signals relating to blood volume, which means that the only real structural elements which can be isolated are blood vessels, and while the overall structure of the vasculature is the same between subjects, the specific locations are highly variable. This means that there are no reliable references that can be used for control point registration. On the other hand, the intensities of voxels vary dramatically, with significant peaks located on large blood vessels, which, when combined with the variations in vascular locations and the proportionally small area of the image occupied by said blood vessels, means that intensity-based registration techniques tend to arbitrarily align the largest blood vessels with no regard for the rest of the structure. Manually aligning all the recordings is both error prone and, with over 240 recordings in the full dataset, extremely time consuming. There is no good solution to these issues without getting the raw data instead of the PDIs, so the final solution is a compromise.

Since a single subject's vasculature is indeed fairly consistent, one can at least make some use of intensity-based registration techniques to co-register the recordings of one slice for a given subject, though this does not always succeed. Registering each slice of each subject now becomes a somewhat more manageable task of just manually aligning around 48 different recordings to an atlas (specifically aligning to the Allen brain atlas [1] using pre-aligned fUS scans from [4]). The overall algorithm for this process is described in pseudocode format in algorithm 4, and an example of the automated registration for a single subject and slice is shown in figure D.1.

Algorithm 4 Affine Registration Pipeline for PDIs

Input: Sequence of PDIs $\underline{\mathbf{X}}^{[k]} \in \mathbb{R}^{X \times Y \times N},\ k = 1, \ldots, K$; reference image I_{atlas}
Let $\mathcal{I}(\underline{\mathbf{X}}, T_1 \circ T_2) := \mathcal{I}(\mathcal{I}(\underline{\mathbf{X}}, T_2), T_1)$
Let $\Delta_t(T)$ and $\Delta_r(T)$ denote translation magnitude (pixels) and rotation (degrees) of transform T
Output: Aligned PDIs $\{\underline{\mathbf{X}}^{[k]}_{\text{aligned}}\}_{k=1}^{K}$
Compute $I^{[k]}_{\text{MLPDI}(x,y)} = \frac{1}{N}\sum_{n=1}^{N} \log \underline{\mathbf{X}}^{[k]}_{(x,y,n)}$ {Mean log-PDI}
Initialize $T_1 := \text{identity}$ {no transform needed for $k = 1$}
for $k = 2$ to K **do**
  Compute affine transform T_k to align $I^{[k]}_{\text{MLPDI}} \to I^{[1]}_{\text{MLPDI}}$
  if $\Delta_t(T_k) > \tau_t$ **or** $\Delta_r(T_k) > \tau_r$ **then**
    for $j = 2$ to K where $j \neq k$ **do**
      Compute $T_{k \to j}$: align $I^{[k]}_{\text{MLPDI}} \to I^{[j]}_{\text{MLPDI}}$
      Store $\|T_{k \to j}\|$
    end for
    $j^* = \arg\min_j \|T_{k \to j}\|$
    $T_k = T_{j^*} \circ T_{k \to j^*}$
    Mark k as *indirect*
  end if
end for
Compute manual transform T_{atlas}: align $I^{[1]}_{\text{MLPDI}} \to I_{\text{atlas}}$
for $k = 1$ to K **do**
  $\underline{\mathbf{X}}^{[k]}_{\text{aligned}} = \mathcal{I}(\underline{\mathbf{X}}^{[k]}, T_{\text{atlas}} \circ T_k)$
end for

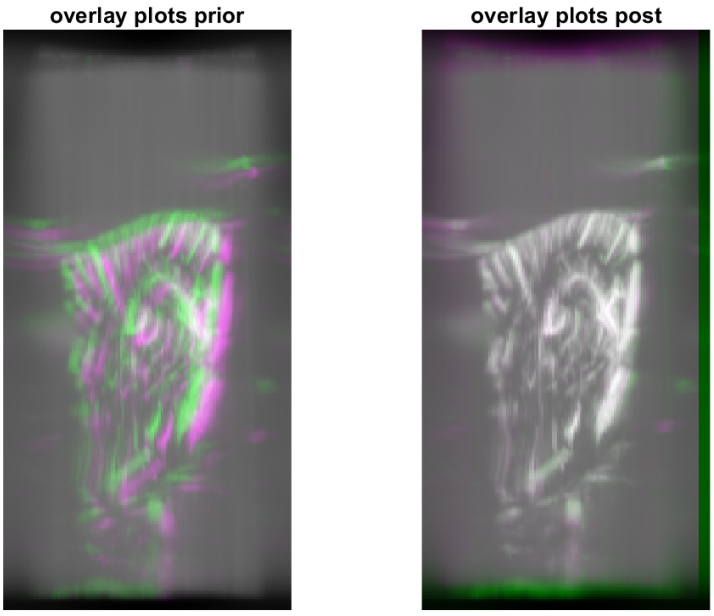


Figure D.1: Difference images of the mean log PDIs of two different recordings for a specific subject and slice, before (left) and after (right) intensity-based registration.

E

SCP and multi-shift CPD Performance Comparison

The SCP algorithm and the proposed multi-shift CPD algorithm have very similar underlying data models, but one of their main differences is that multi-shift CPD was specifically developed to handle large numbers of voxels with unique shifts. To illustrate the difference in focus and the performance differences that result, tests were performed where both algorithms were run on several tensors of different sizes and the computation time per iteration compared. To ensure a fair comparison, the ranks were selected to be equal ($R = 10$), and the multi-shift CPD algorithm was configured to allow only shifts in the first mode, as is done by SCP. These results are shown in figure E.1, and one can see that while SCP is two to three times faster than the multi-shift CPD algorithm when the spatial dimension has a length of 100, the performance gains increase significantly as the spatial dimension increases (over 20 times faster in the 10,000 voxel, 1024 sample case).

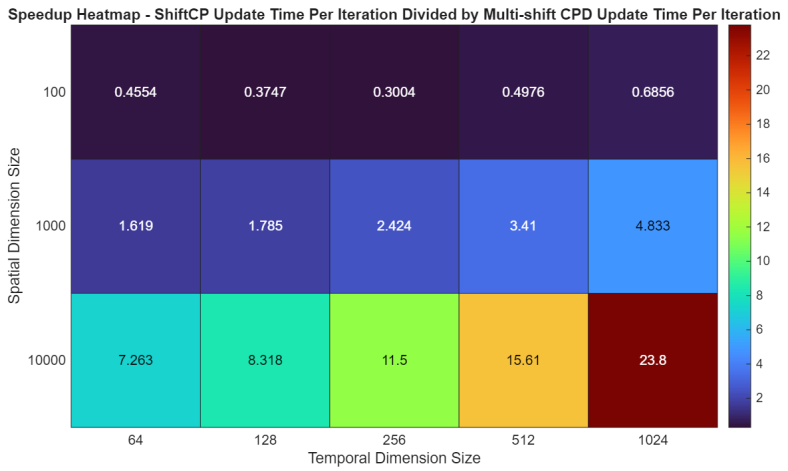


Figure E.1: A heatmap of the time per iteration for SCP divided by the time per iteration of multi-shift CPD to give a speedup for various spatial and temporal dimensions of the data.

F

Multi-shift CPD and Local Minima

There is one significant issue with the multi-shift CPD algorithm which must be addressed. The minimisation problem it aims to solve is already non-convex. However, by allowing for explicit time shifts it indirectly introduces additional local minima into the overall loss landscape, as these shifts can also be captured in the signal's alignment within the component. This is illustrated in a simple case where the true data is known and composed of five shifts of a canonical HRF, depicted in figure F.1. One can see the issue introduced when the underlying signal is misaligned from its optimal position, as the integer shifts account for the shift, but the overall error is higher as the shifts are bounded and the component is a clipped version of the signal. To overcome this issue we initialise the components of the multi-shift CPD using an unshifted CPD, as we can see from the rightmost plot of figure F.1 that the CPD does not suffer from these local minima. This initialisation relies on an assumption about the data: that the shifts of a signal are evenly spread around the optimal alignment of the signal in the component. If this assumption is false, then the loss landscapes instead appear as in figure F.2, showing how the CPD error minima are no longer in the vicinity of the global minima of the multi-shift CPD. As an interesting aside, these local minima are also much steeper, and the CPD no longer has its minima at the optimal location in the case where the signal of interest is uncorrelated with shifts of itself. An example of this is illustrated in the loss landscapes of figure F.3. In the case of haemodynamic responses in fUS, the correlation between shifted versions of a signal is a fairly reasonable assumption as responses occur over multiple samples.

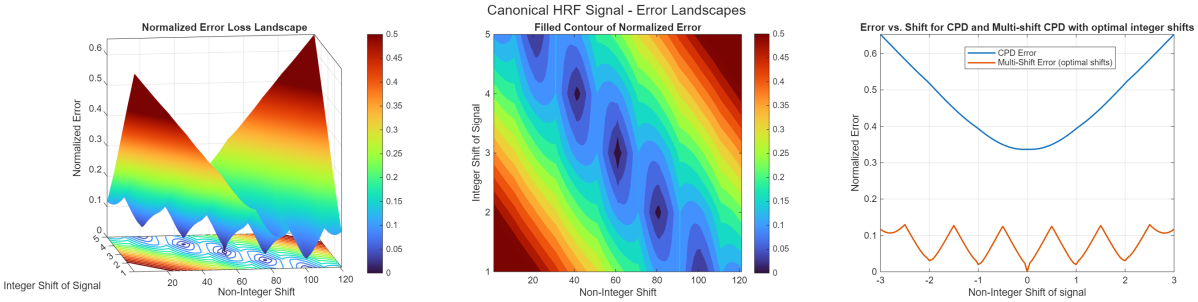


Figure F.1: Loss landscape in terms of normalised error for the multi-shift CPD algorithm in a trivial case where the data is composed of 5 shifts of a known canonical HRF signal for different values of the explicit multi-shift CPD shifts and non-integer shifts of the underlying signal captured in the temporal component (left). The loss landscape from above depicting the optimal ”valley” produced and the local minima therein (centre) and the loss curve along the centre of this valley, effectively assuming optimal integer shifts, plotted with the loss curve of a CPD with the same non-integer shifts in the temporal component.

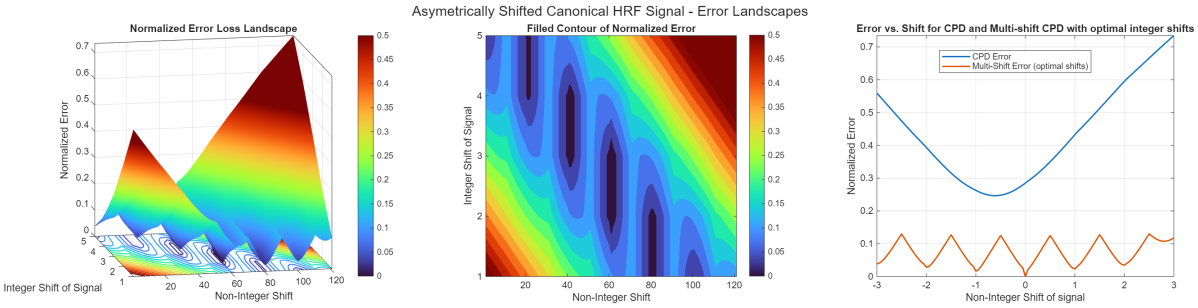


Figure F.2: Loss landscape in terms of normalised error for the multi-shift CPD algorithm in a case where the data is composed of 5 (unevenly spread) shifts of a known canonical HRF signal for different values of the explicit multi-shift CPD shifts and non-integer shifts of the underlying signal captured in the temporal component (left). The loss landscape from above depicting the optimal ”valley” produced and the local minima therein (centre) and the loss curve along the centre of this valley, effectively assuming optimal integer shifts, plotted with the loss curve of a CPD with the same non-integer shifts in the temporal component.

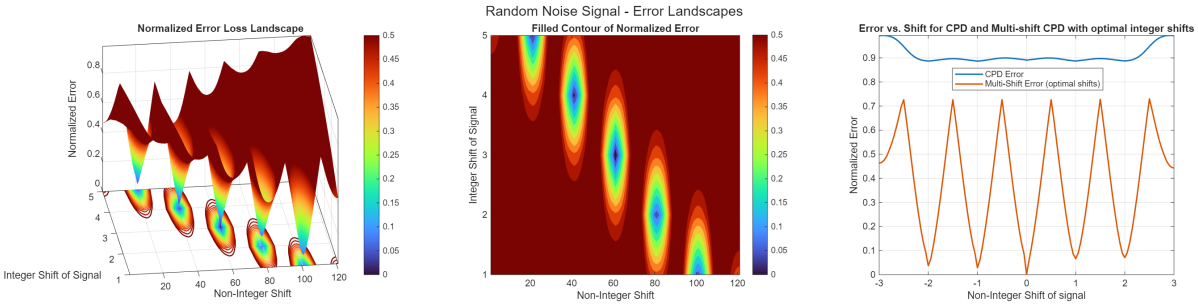


Figure F.3: Loss landscape in terms of normalised error for the multi-shift CPD algorithm in a case where the data is composed of 5 shifts of a known random noise signal for different values of the explicit multi-shift CPD shifts and non-integer shifts of the underlying signal captured in the temporal component (left). The loss landscape from above depicting the optimal ”valley” produced and the local minima therein (centre) and the loss curve along the centre of this valley, effectively assuming optimal integer shifts, plotted with the loss curve of a CPD with the same non-integer shifts in the temporal component.

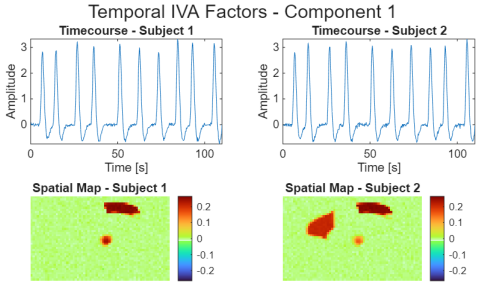
G

Learning Effect Identifiability Results

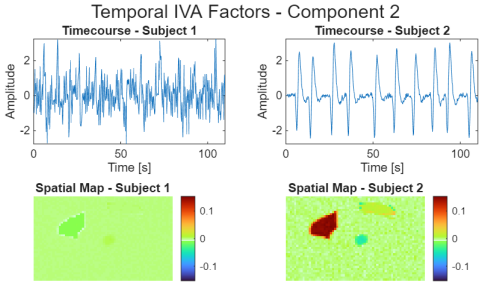
This appendix contains all the extracted components for the synthetic learning effect identifiability experiment. This includes multiple results for all four decompositions tested: IVA, PARAFAC2, CPD, and multi-shift CPD.

Class Differences in Spatial Maps

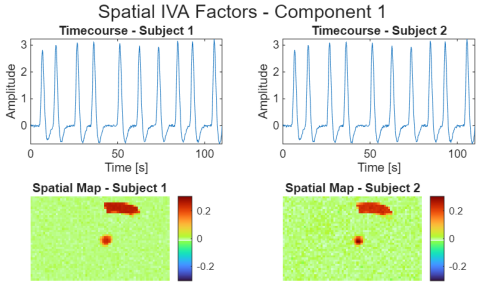
Temporal IVA Factors - Component 1



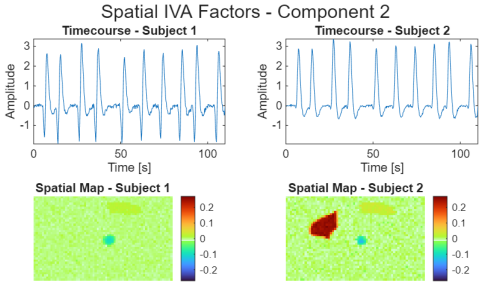
Temporal IVA Factors - Component 2



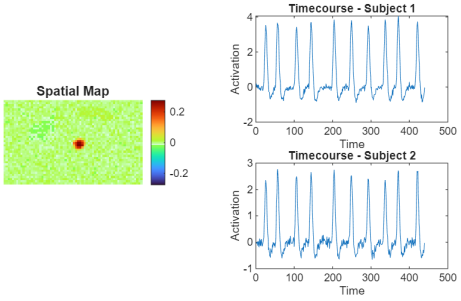
Spatial IVA Factors - Component 1



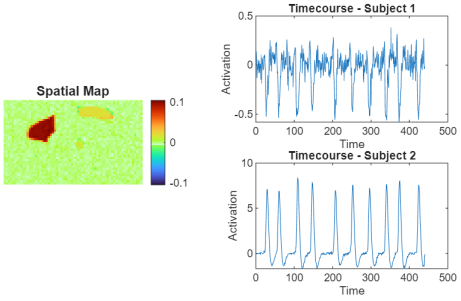
Spatial IVA Factors - Component 2



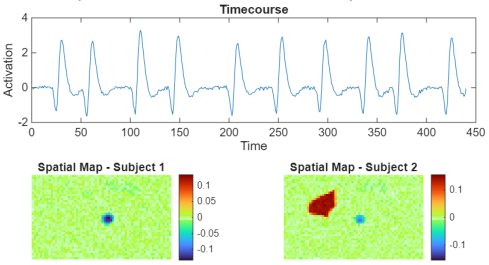
Temporal PARAFAC2 Factors - Component 1



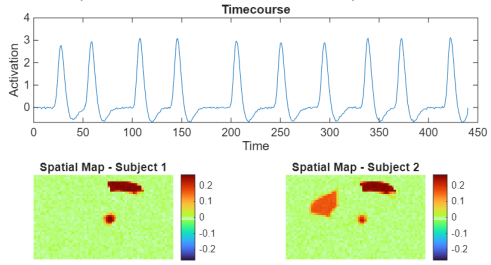
Temporal PARAFAC2 Factors - Component 2



Spatial PARAFAC2 Factors - Component 1

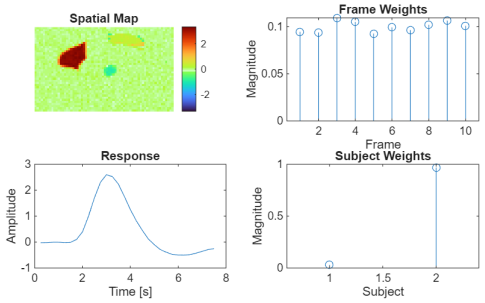


Spatial PARAFAC2 Factors - Component 2

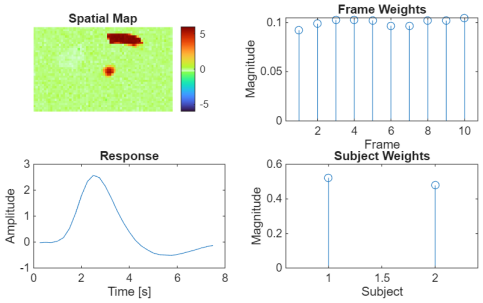


Class Differences in Spatial Maps (continued)

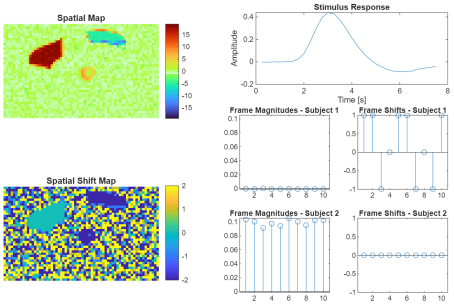
CPD Factors - Component 1



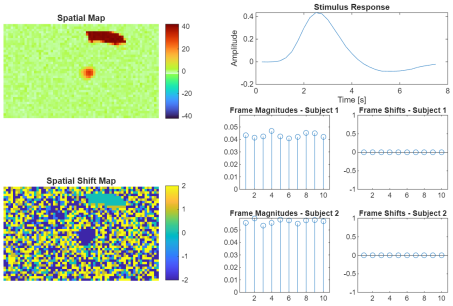
CPD Factors - Component 2



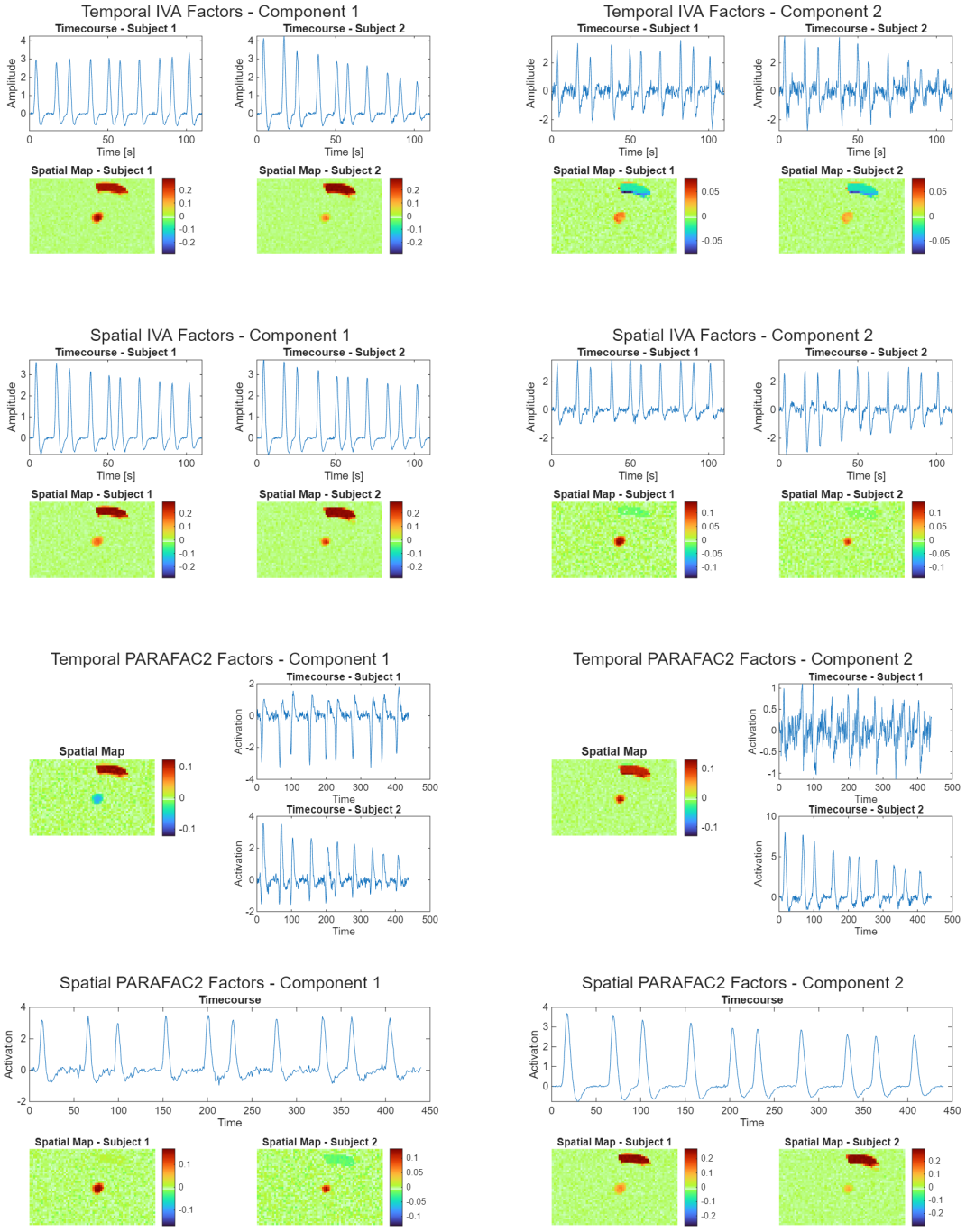
Shifted CPD Factors - Component 1



Shifted CPD Factors - Component 2

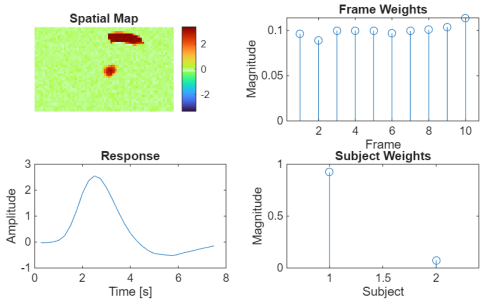


Class Differences in Temporal Magnitudes over Time

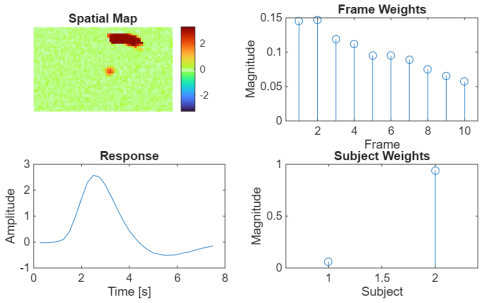


Class Differences in Temporal Magnitudes over Time (continued)

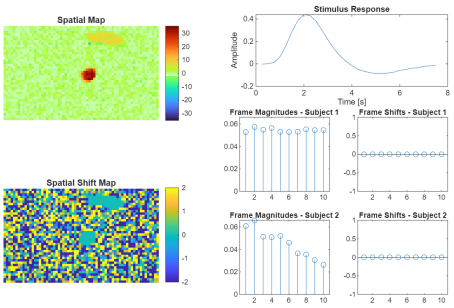
CPD Factors - Component 1



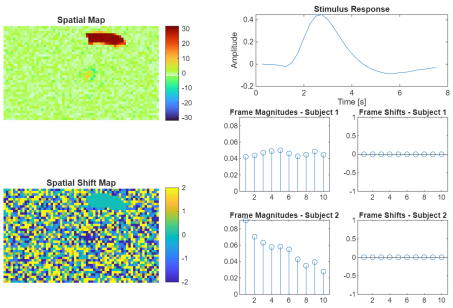
CPD Factors - Component 2



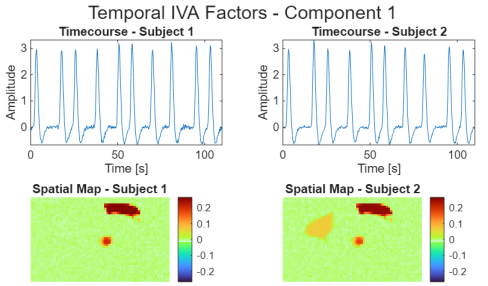
Shifted CPD Factors - Component 1

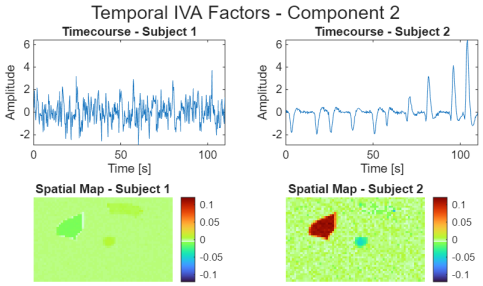


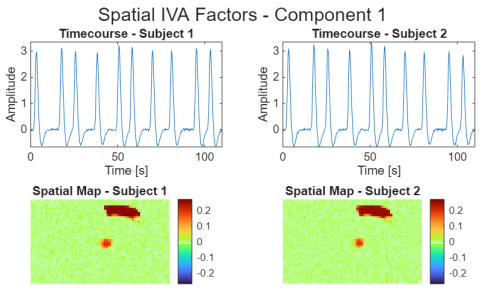
Shifted CPD Factors - Component 2

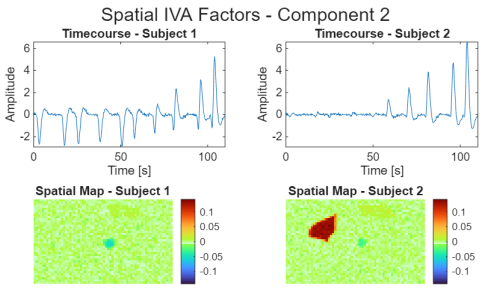


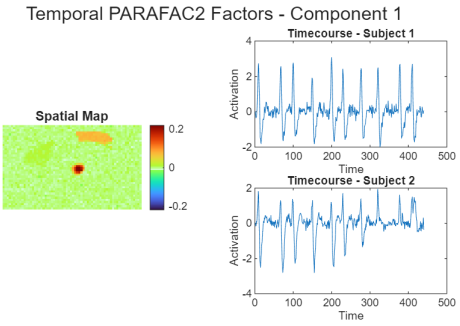
Class Differences in Local Region Delayed Response Occurrence

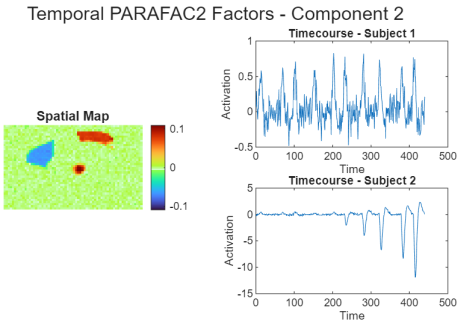


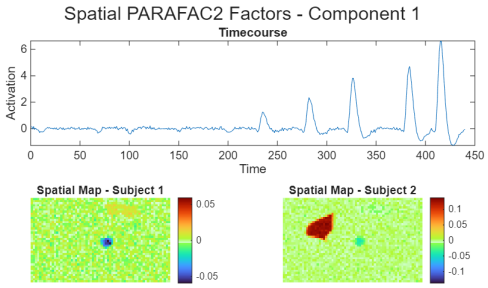


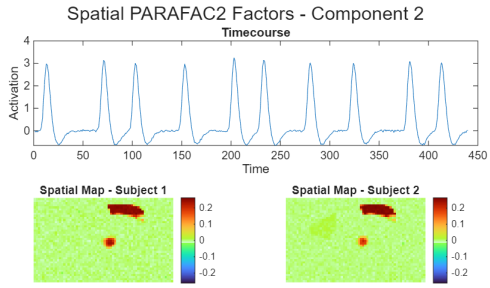




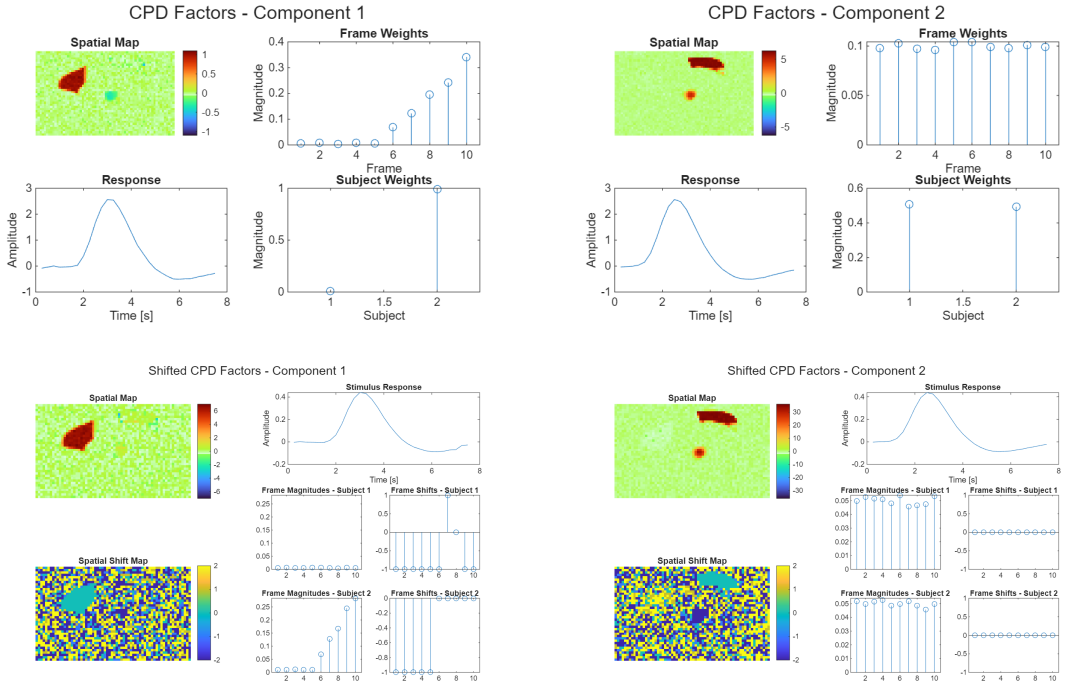




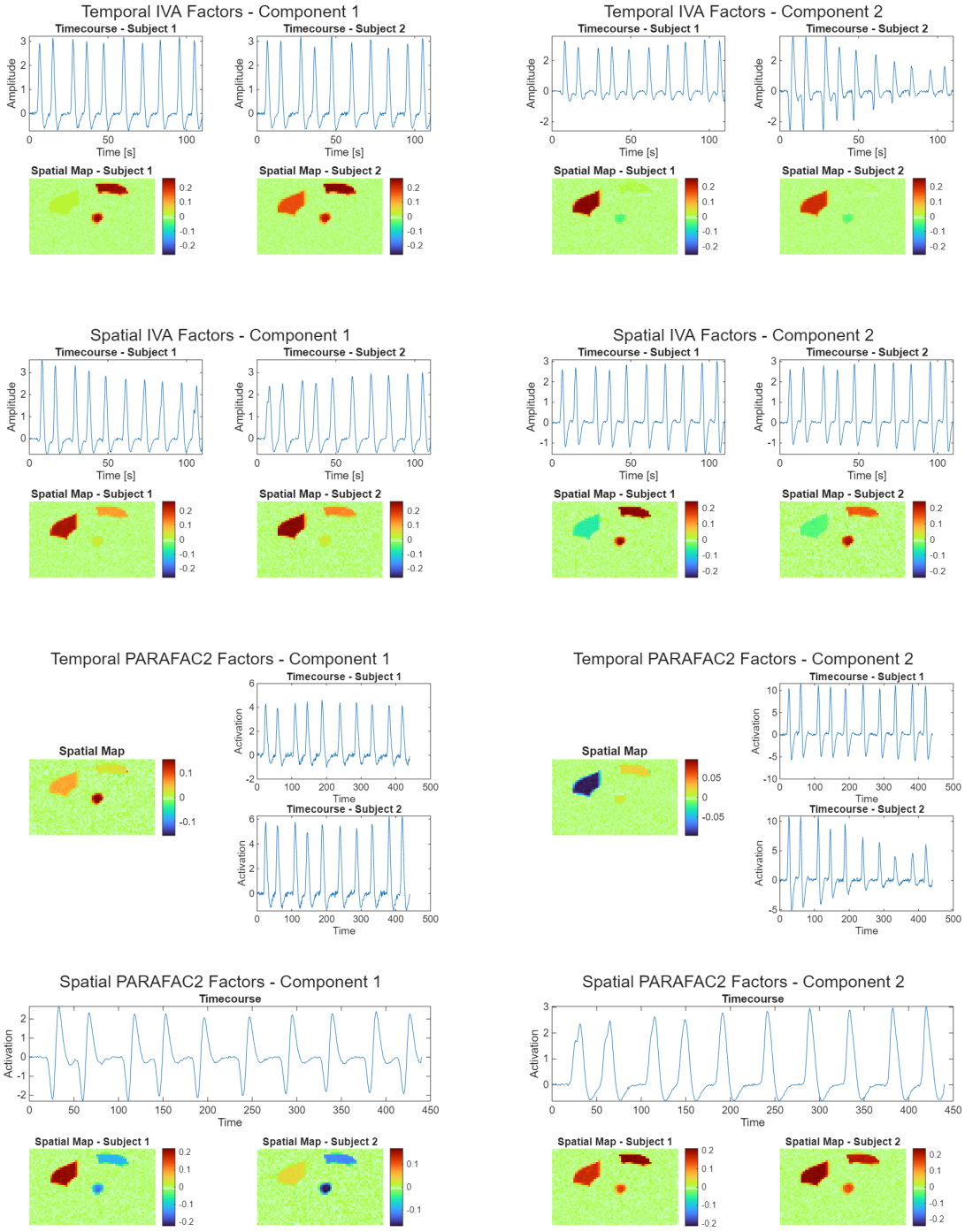




Class Differences in Local Region Delayed Response Occurrence (continued)

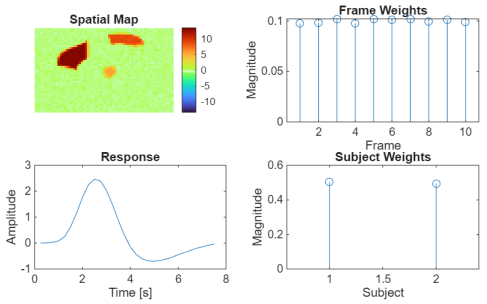


Class Differences in Temporal Shifts over Time

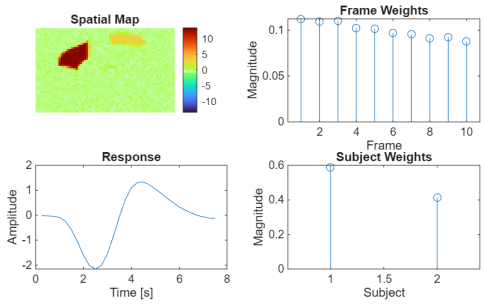


Class Differences in Temporal Shifts over Time (continued)

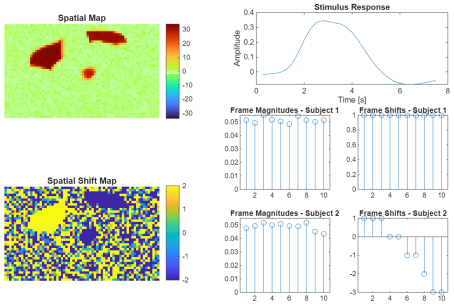
CPD Factors - Component 1



CPD Factors - Component 2



Shifted CPD Factors - Component 1



H

Non-ideality Robustness Results and Discussion

H.1. Results

H.1.1. Isolated Physiological Non-idealities

The following figures (figures H.1 to H.3) indicate the results of the decomposition based subgroup identification algorithms when applied to data with non-idealities caused by physiological effects. There are three main non-idealities assessed here. The first two are response magnitude variation and response peak time variation, which are both caused by either underlying neurological responses varying or by differences in cardiovascular conditions over time, changing the haemodynamic response. The third is non-rigid warping, which models tissue deformation and also serves to capture the possibility of cerebrum differences between subjects.

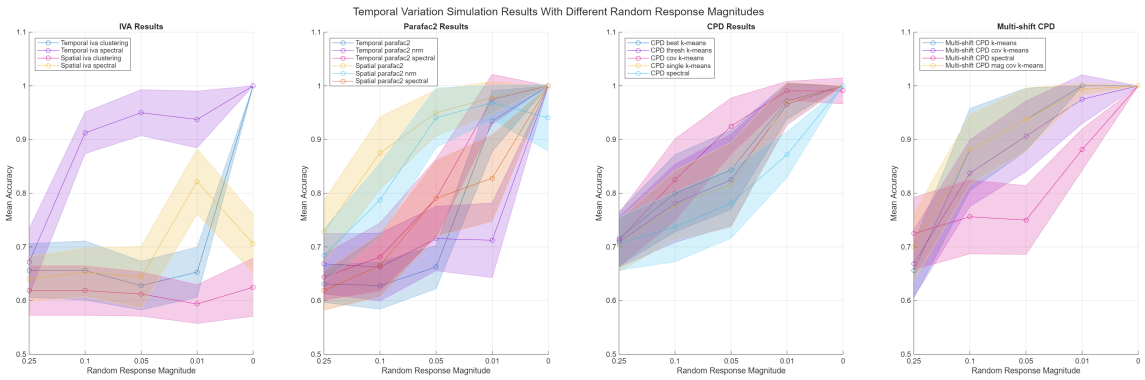


Figure H.1: These graphs show the accuracy of subgroup identification algorithms using IVA, PARAFAC2, CPD, and multi-shift CPD when the simulated response magnitude varies randomly. Results for different maximum magnitude variation are shown with the maxima indicated on the x-axis.

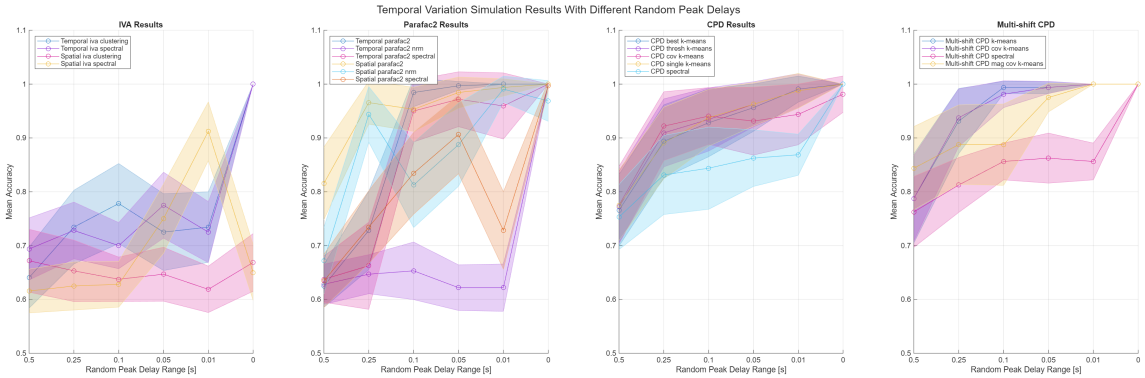


Figure H.2: These graphs show the accuracy of subgroup identification algorithms using IVA, PARAFAC2, CPD, and multi-shift CPD when the simulated response has a random delay added to the response peak. Results for different peak delay ranges are shown with the range indicated on the x-axis. The range value indicates the maximum that a peak can vary in seconds from the default delay (this may be positive or negative).

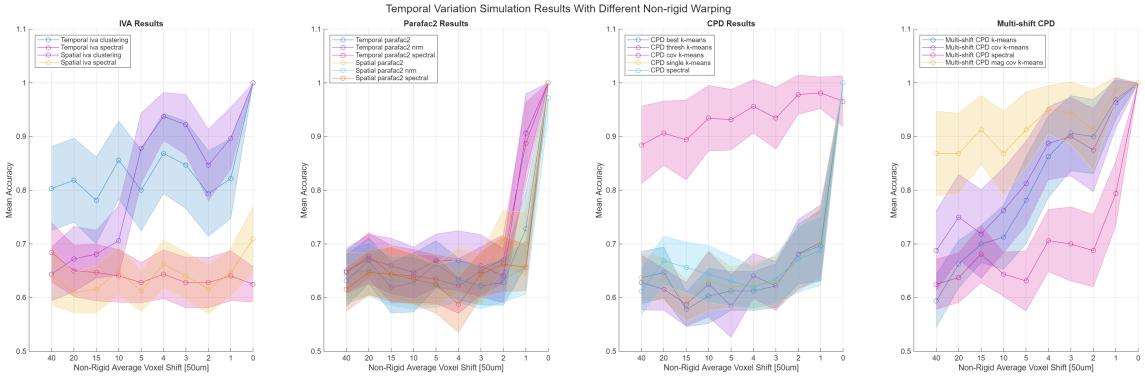


Figure H.3: These graphs show the accuracy of subgroup identification algorithms using IVA (left), PARAFAC2 (centre), and CPD (right) when the 3D spatial map is warped by a non-rigid field before the plane is sampled. Results for different amounts of warping are shown, where the average amount that each voxel is shifted is shown on the x-axis.

H.1.2. Isolated Measurement Non-idealities

There are a number of errors added to the data by the measurement and data processing steps of fUS itself, and the results shown in figures H.4 to H.8 reflect the decomposition-based subgroup identification algorithms' performance when applied to data with these types of errors. These errors can be roughly characterised into three groups based on whether they affect the temporal dimension, spatial dimension, or both dimensions. The only error type included here which affects both dimensions is measurement noise, which is included on every data point in both space and time.

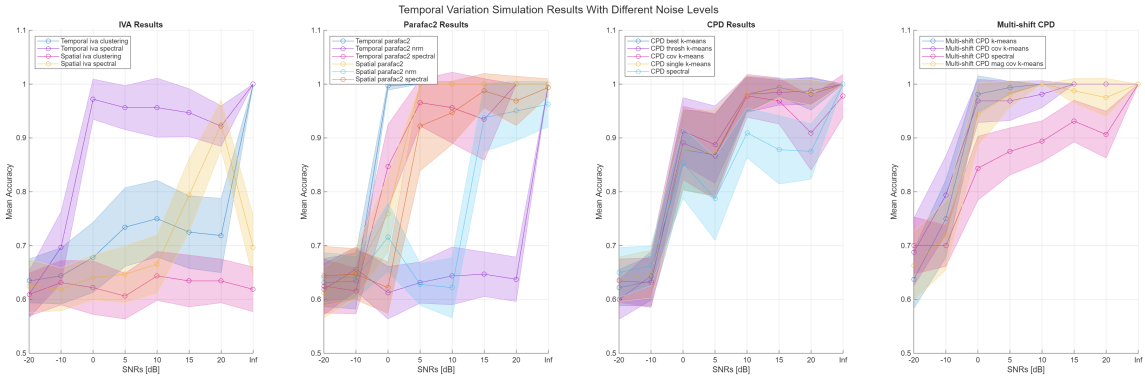


Figure H.4: These graphs show the accuracy of subgroup identification algorithms using IVA, PARAFAC2, CPD, and multi-shift CPD when the simulated data has different amounts of noise added to the signal. The x-axis indicates the magnitude of the noise relative to the signal as an SNR.

Reference misalignments refer to incorrect alignment of different data recordings in the temporal dimension, and the effects thereof are shown in figure H.5.

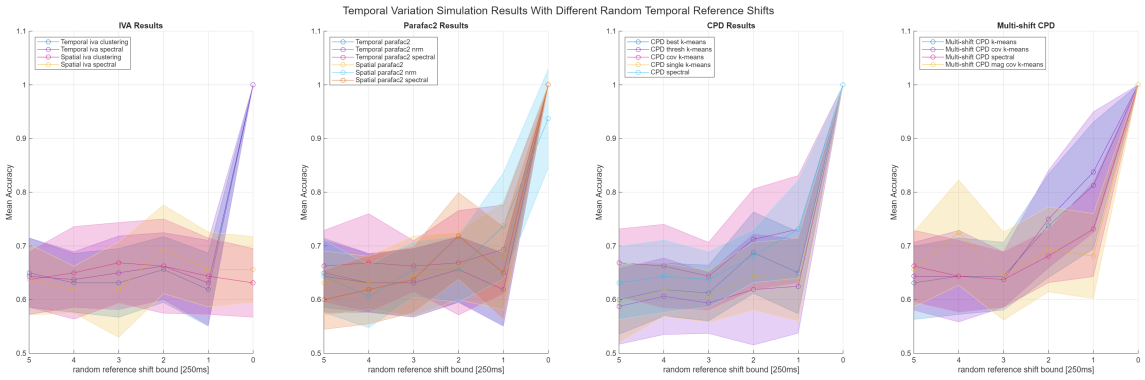


Figure H.5: These graphs show the accuracy of subgroup identification algorithms using IVA, PARAFAC2, CPD, and multi-shift CPD when the simulated data has misalignments in the temporal reference between subjects, effectively misalignments between subjects in the temporal dimension. The bound on the x-axis indicates how many samples of a recording may vary from the true value (what the maximum possible temporal shift of any subject may be).

The spatial dimension of the data is generally affected during measurement either when the fUS planer scan is taken at a different position than other recordings, as in figure H.6, or when the scans of different recordings are not co-registered, resulting in either recordings which are spatially translated relative to one another as in figure H.8, at a different angle as in figure H.7 (rotated relative to one another), or both.

Figure H.6: These graphs show the accuracy of subgroup identification algorithms using IVA, PARAFAC2, CPD, and multi-shift CPD when the measured plane has a random offset (along the off-plane z-axis), effectively an off-plane spatial misalignment. The maximum possible z-axis shift of a recording is indicated on the x-axis of the graphs.

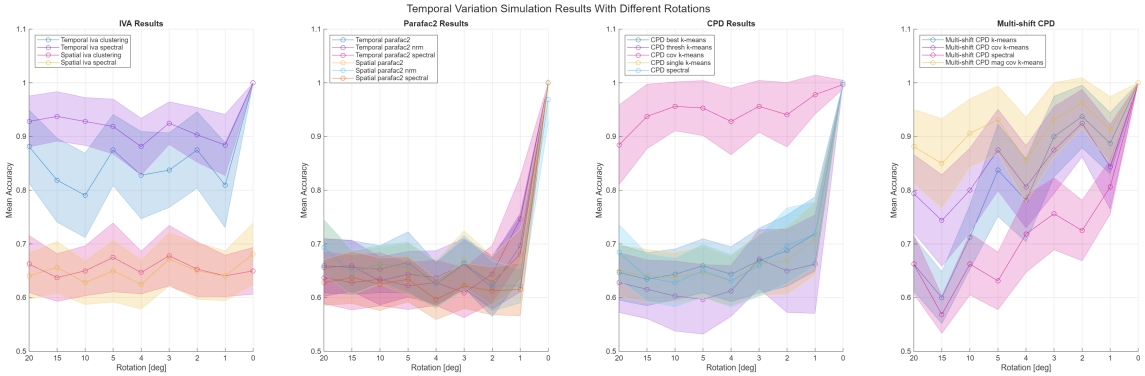


Figure H.7: These graphs show the accuracy of subgroup identification algorithms using IVA, PARAFAC2, CPD, and multi-shift CPD when there is a random spatial rotation around an axis orthogonal to the plane. The maximum possible random rotation of a subject is indicated in degrees on the x-axis.

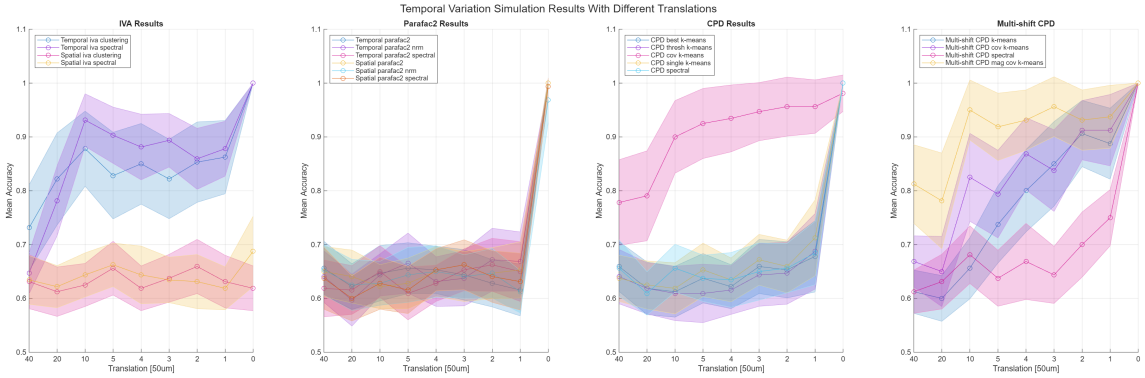


Figure H.8: These graphs show the accuracy of subgroup identification algorithms using IVA (left), PARAFAC2 (centre), and CPD (right) when the simulated data has a random linear in-plane spatial translation. The maximum possible translation distance is shown on the x-axis of the graph.

H.2. Discussion

There are a variety of different non-idealities which exist in fUS and were simulated. This section will first address the overall performance of each algorithm type in the different non-ideality settings, and then discusses and explains several specific phenomena. Firstly, the sIVA clustering algorithms only look at the spatial components, and since the class difference is encoded temporally this is a rather severe disadvantage, and why this algorithm typically performs so poorly in these results. The results are still included for illustrative purposes, to highlight this shortcoming, but also because in some cases it still performs better than expected. This issue could, of course, be overcome fairly simply by including the unconstrained temporal signals of each recording in the clustering algorithm, however, this has its own challenges.

Starting with the random peak variation shown in figure H.1 it is first worth mentioning that in this specific experiment the cluster separation is explicitly encoded into response magnitudes, which means a significant disturbance to this parameter will inherently deteriorate results regardless of the algorithm used, however the tolerance of different algorithms still varies and is worth discussing. Of the options tIVA with spectral clustering has the best performance overall while all other IVA based algorithms perform rather poorly as the random magnitudes deteriorate the distinction between clusters in SCV covariances (which were already rather close thanks to the signals used being very similar). The PARAFAC2 options have average performance overall with the spatial version performing somewhat better. This may seem odd, as these clustering algorithms operate only on the spatial dimensions like the sIVA clustering, but it works thanks to the fact that the mode-three components are still used in PARAFAC2 and cause differences in the spatial maps overall magnitude for each recording. These magnitudes vary slightly between subjects due to the signal differences between clusters, and since the random response magnitude change is zero mean, it is less affected by the randomness than some other algorithms. This effect is however not at all useful in practical settings where there is lots of noise and spatial effects as well, as they will mask this relatively small distinction. Overall CPD performs rather well as it distinguishes using component magnitudes instead of the time course itself, averaging the effects over all the recordings to help suppress the randomness.

While it still affects the response, varying the delay as in figure H.2 has very different effects. In general this very detrimentally effects IVA, even with very small delays, and highlights one of the issues with IVA, in that because SCVs are forced to be independent, it struggles to capture similar but translated structures thanks to the independence assumption either averaging all the shifted structures if they are close enough together, or forcing them to be maximally dissimilar if in different SCVs, and making them far less interpretable as a result. Interestingly the spatial IVA performs reasonably well if the shift is small (<0.05s). This is similar to PARAFAC2's case, where because the component magnitude is split between both components and the delay shifts have very little effect on the overall magnitude of the signal when small the signal differences translate to magnitude differences in the spatial maps which are otherwise identical and allow for reasonable clustering. Again though, this is practically useless in real data settings. The PARAFAC2 algorithms have rather diverse results, but by forcing the same temporal response and exploiting the component magnitudes the spatial PARAFAC2 algorithms do rather well with all delays tested, while the temporal PARAFAC2 manages to capture the differences when the delays are smaller than one

sample (250ms). Thanks to the more rigid structure and averaging effects the CPD captures the differences well through most settings.

The non-rigid 3D spatial warping of the tissue causes almost all algorithms, bar 3, to perform very poorly. The accuracy of the PARAFAC2 algorithms and most of the CPD algorithms plummets with even small amounts of spatial warping as these algorithms rely heavily on spatial alignment for component extraction, the exceptions this norm being the temporal IVA algorithms which do not rely on spatial alignment at all, and the covariance based CPD, as it does not rely directly on the absolute component magnitudes but on structure between recordings.

Moving into the realm of measurement errors with simple measurement noise shown in figure H.4. Here spectral-based tIVA is the only IVA algorithm that really performs well (with the exception of sVIA at high SNR when magnitudes are more preserved). All CPD and PARAFAC2 algorithms also fare fairly well with noise, with the exception of the normalized versions of PARAFAC2, as the destruction of magnitude information allows the noise signals to influence the clustering more heavily.

Temporally misaligning different recordings entirely destroys the performance of all algorithms as can be seen in figure H.5. IVA's struggle with temporal shifting has already been mentioned, and in the case of PARAFAC2, since it uses signal covariances to cluster, the implication is that the temporal shifting creates a more significant change to the covariance scores than the signal differences themselves. Of the three CPD has the best performance, but as it exploits magnitude changes and cannot account for temporal shifts the shifting deteriorates the magnitudes enough for the underlying pattern to be masked and thus the clustering fails in most cases.

Shifting the scan to different locations along the off-plane axis is similar to spatial warping in that it changes the spatial maps more the larger the effect, and the results (figure H.6) are similar to those of spatial warping. Temporal IVAs lack of dependence on the spatial maps is an advantage in this setting, and the covariance based CPD clustering performs well as before due to its lack of reliance on exact values but instead relying on relation between recordings. Unsurprisingly PARAFAC2 performs very poorly thanks to its spatial dependence.

Both spatial translation, figure H.8, and rotation, figure H.7, show the same impact on the algorithms as the other spatial warping effects. Again only tIVA and covariance based CPD have any reasonable results while PARAFAC2 is hamstrung. This indicates that regardless of the particular type of spatial warping the effects are similar. This may change if the cluster differences are instead spatial, as the sIVA and sPARAFAC2 based algorithms allow for more structured variation in these dimensions.

While these results illustrate the advantages and drawbacks of different algorithms, it also shows how significant the contributions of different non-idealities are. Spatial misalignment has a significant effect on all the different algorithms, and illustrates how important co-registration pre-processing is. Temporal misalignment is even more impactful, severely deteriorating the results of all the different algorithms. Finally, since temporal shifts are expected physiological responses they are very challenging to remove, and applying algorithms which can account for these types of changes may be beneficial.

I

Multi-shift CPD Results for Real Data

The figures in figure I.1 show the shared unshifted CPD results with the joint response obtained from the first step of the multi-shift CPD algorithm which is used to initialise the next decomposition steps. The figures thereafter are the shifted components extracted by multi-shift CPD including the per-response shifts.

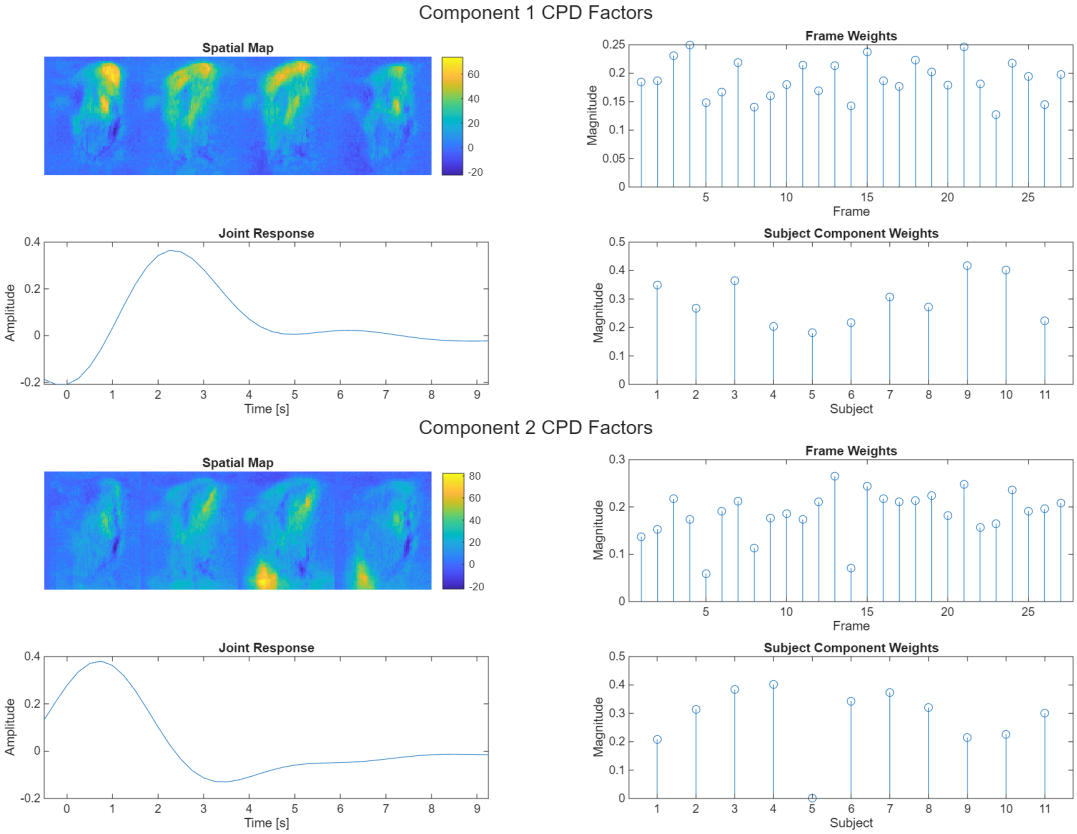
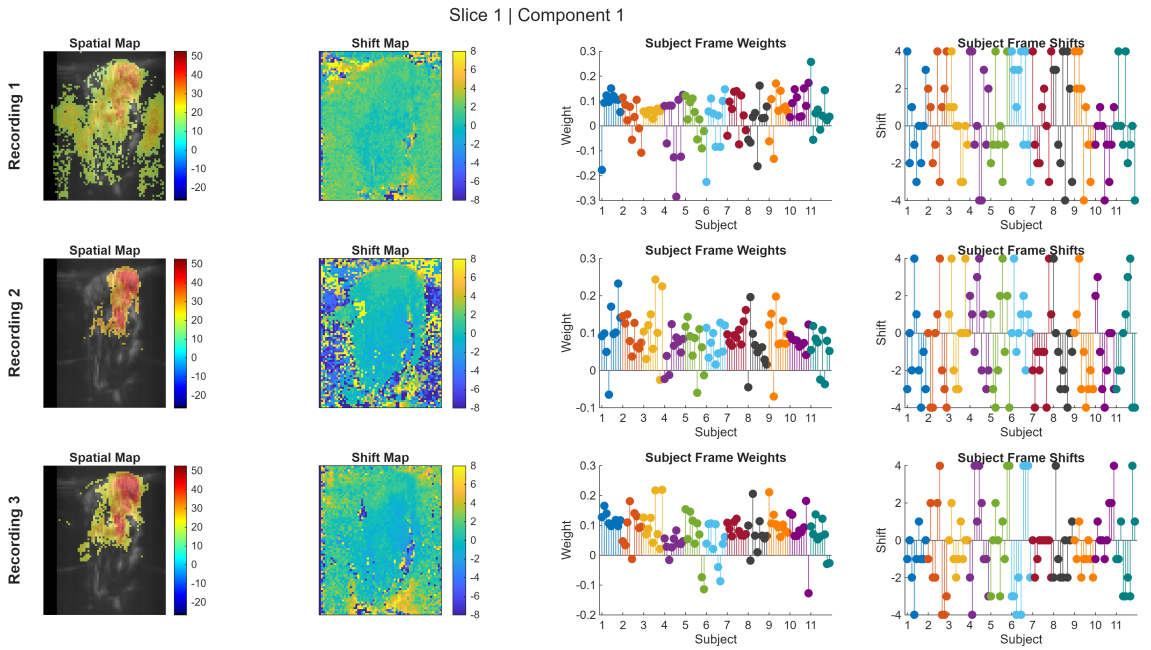
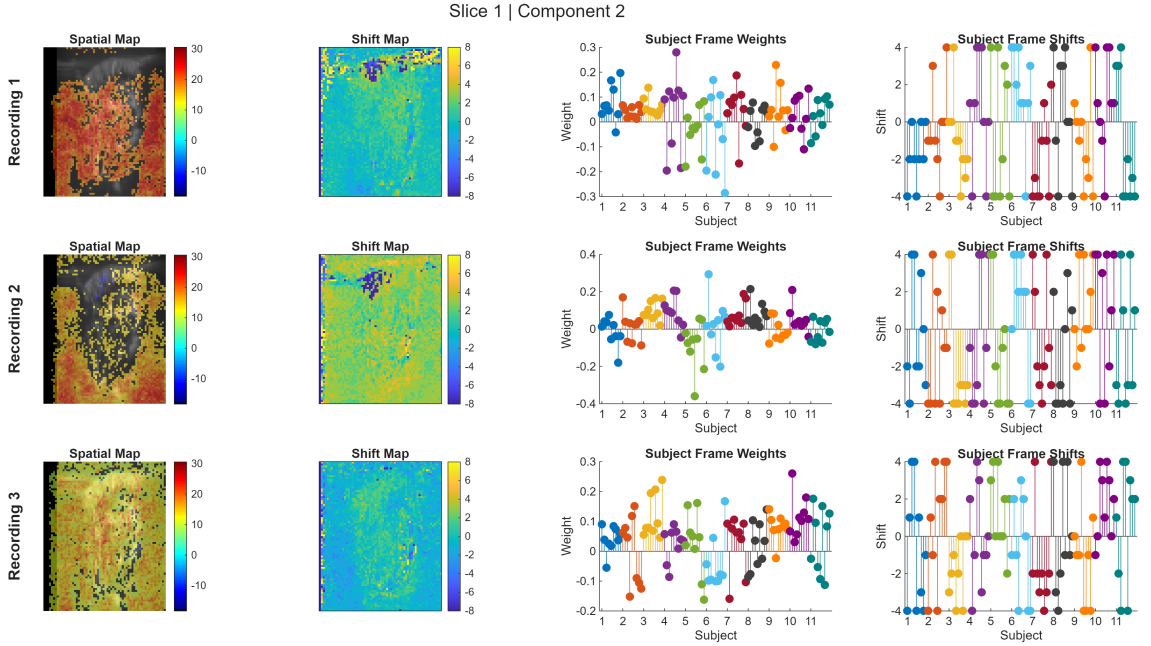
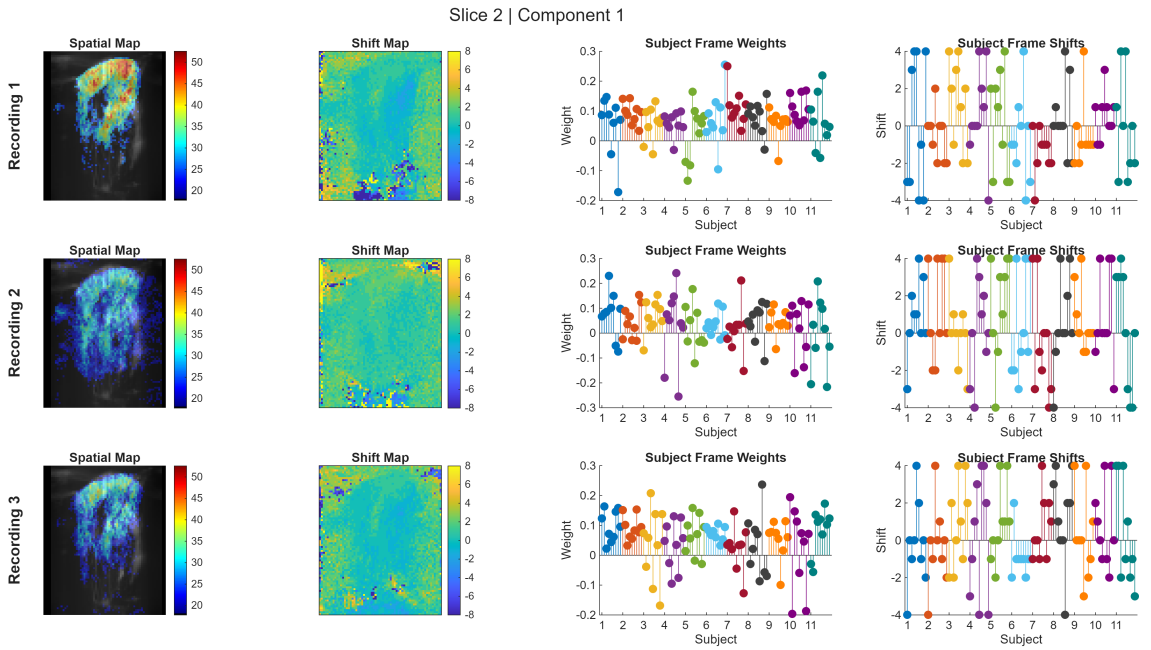
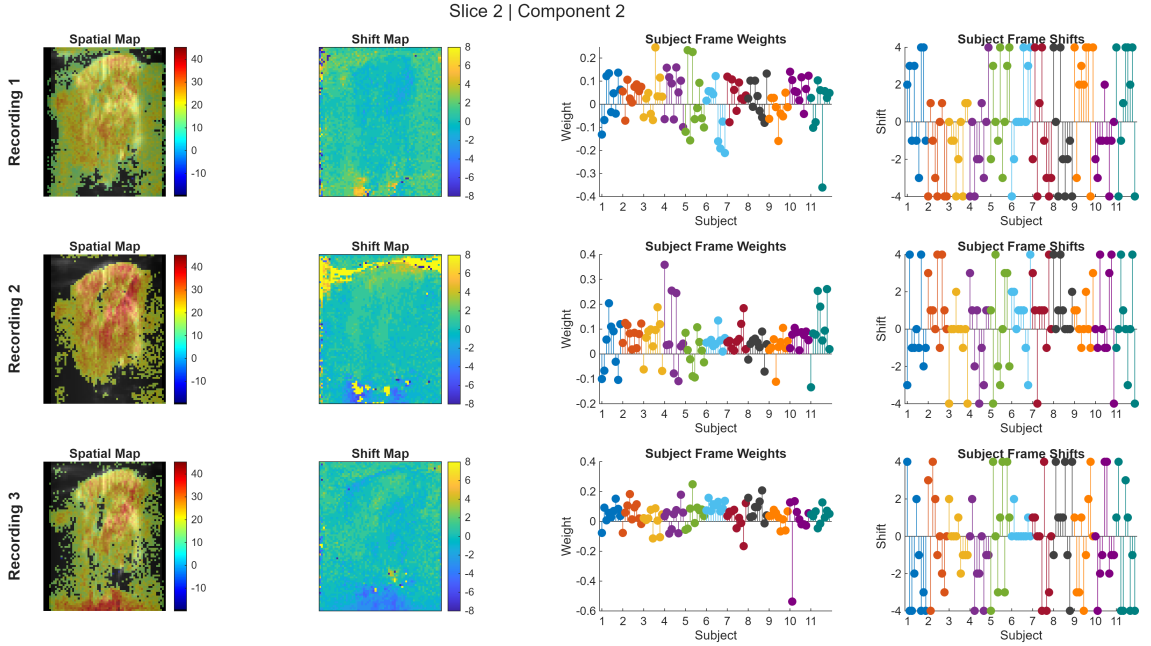


Figure I.1: Extracted CPD component factors used to initialise the multi-shift CPD algorithm. The spatial maps for all four slices, frame weights (including frames for all three recordings in the same plot), component weight per subject, and the final joint multi-shift CPD response functions for components one and two.

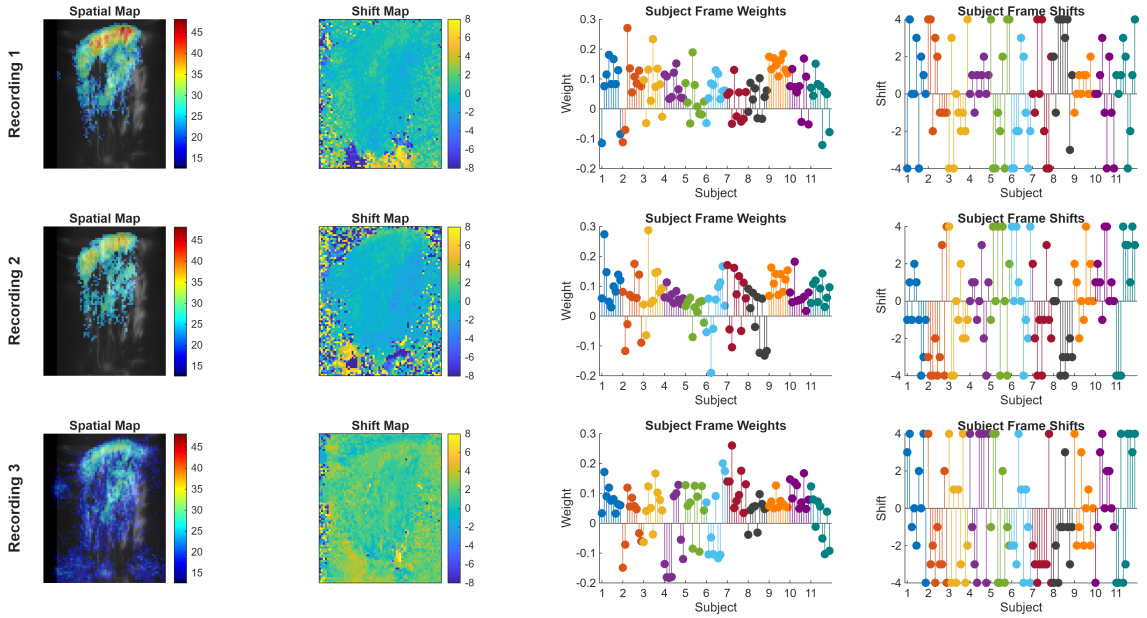

Slice 1 | Component 1
Recording 1
Recording 2
Recording 3
Spatial Map
Shift Map
Subject Frame Weights
Subject Frame Shifts
Weight
Shift
Subject


Slice 1 | Component 2
Recording 1
Recording 2
Recording 3
Spatial Map
Shift Map
Subject Frame Weights
Subject Frame Shifts
Weight
Shift
Subject

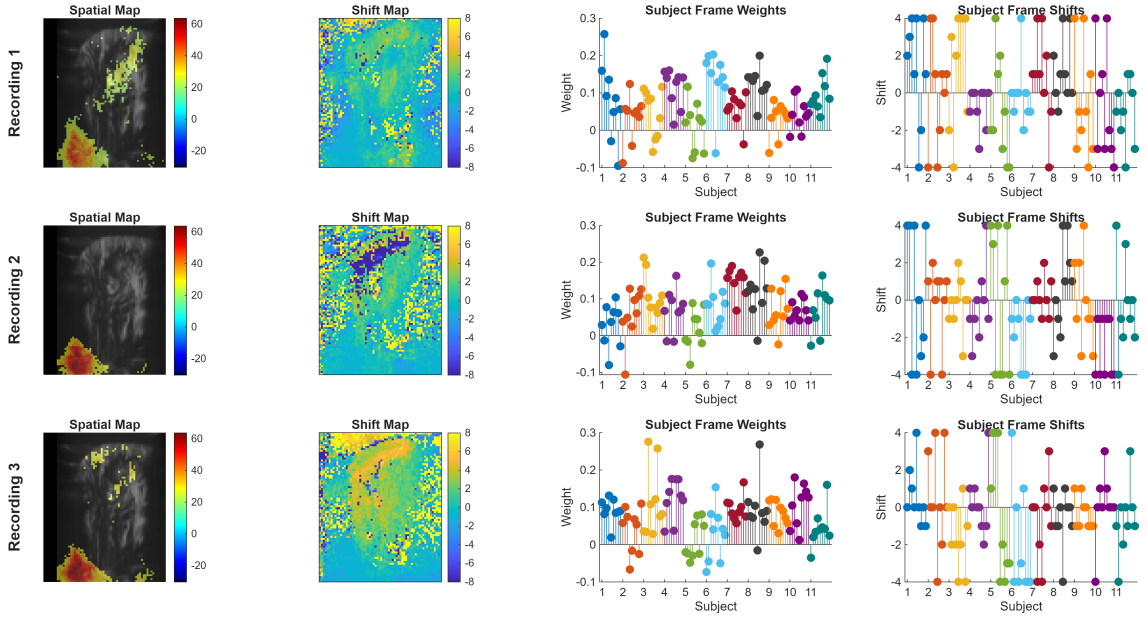

Slice 2 | Component 1
Recording 1
Recording 2
Recording 3
Spatial Map
Shift Map
Subject Frame Weights
Subject Frame Shifts
Weight
Shift
Subject

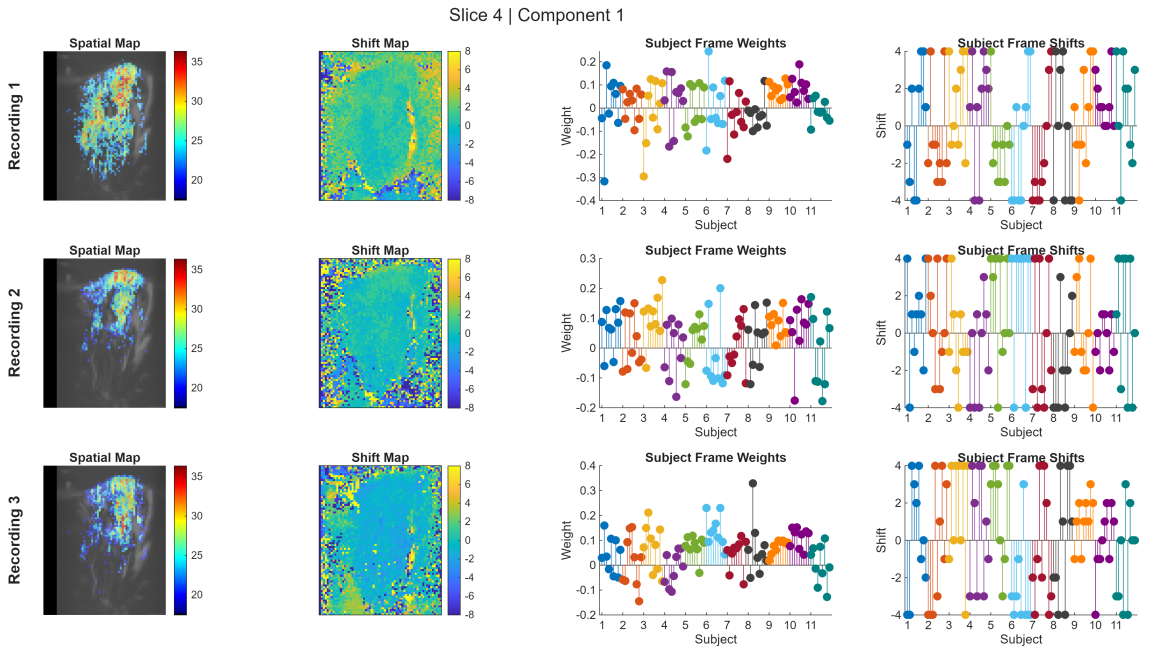

Slice 2 | Component 2
Recording 1
Recording 2
Recording 3
Spatial Map
Shift Map
Subject Frame Weights
Subject Frame Shifts
Weight
Shift
Subject

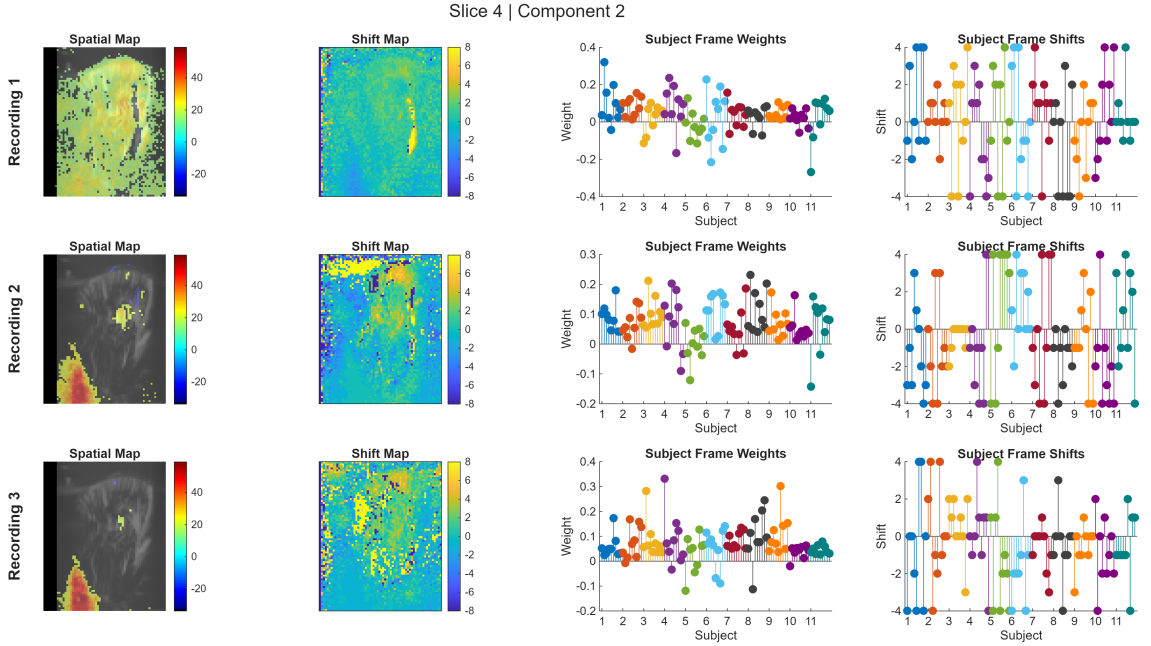
Slice 3 | Component 1



Slice 3 | Component 2




Slice 4 | Component 1
Recording 1
Recording 2
Recording 3
Spatial Map
Shift Map
Subject Frame Weights
Subject Frame Shifts
Weight
Shift
Subject


Slice 4 | Component 2
Recording 1
Recording 2
Recording 3
Spatial Map
Shift Map
Subject Frame Weights
Subject Frame Shifts
Weight
Shift
Subject