

TECHNICAL UNIVERSITY OF CRETE
ELECTRICAL AND COMPUTER ENGINEERING DEPARTMENT



Tensor-based fMRI Signal Processing

by

Paris Karakassis

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DIPLOMA DEGREE OF

ELECTRICAL AND COMPUTER ENGINEERING

September 2014

THESIS COMMITTEE

Professor Athanasios P. Liavas, *Thesis Supervisor*

Professor Michael Zervakis

Associate Professor George Karystinos

Abstract

Functional magnetic resonance imaging (fMRI) is one of the most popular methods in studying the human brain. fMRI provides a non-invasive way to measure brain activity, detecting local changes of blood oxygen level density (BOLD) in the brain, over time. The purpose of fMRI signal analysis is the localization of brain areas that are related with particular tasks. The problem of fMRI signal analysis can be considered as a blind source separation problem (BSS) , which is the problem of extracting a set of source signals from a set of mixed signals, without using prior information (or with very little prior information) about the source signals or the mixing process.

In this thesis, initially, we study the usage of nonnegative matrix factorization models in BSS problems, assuming the existence of delays in propagation environments with or without echo. Next, we study the usage of tensor factorization models, through PARAFAC and Nonnegative Tensor Factorization models, in BSS problems, as well as how these models can be extended under the assumption of propagation environments without echo. Finally, we use these tensor factorization models in fMRI data analysis. In our analysis, we processed fMRI data from different subjects performing the same tasks (group task related fMRI analysis) and we extracted common activation brain maps as well as common activation time signals.

Acknowledgements

I would like to thank my family and my friends for their support and encouragement. I would also like to thank my supervisor, Professor Athanasios Liavas, for his patient guidance and for his useful critiques on this work.

Table of Contents

Table of Contents	7
List of Figures	8
List of Figures	9
List of Abbreviations	11
1 Introduction	13
1.1 Purpose	15
1.2 Notation	15
1.3 Thesis Outline	16
2 Nonnegative Matrix Factorization Models	17
2.1 Introduction	17
2.2 Problem Formulation	18
2.2.1 Cost Functions	18
2.2.2 Biconvexity	19
2.2.3 Alternating Optimization	20
2.2.4 Derivatives for Matrix functions	21
2.2.5 Multiplicative Method	22
2.3 Shifted NMF	25
2.4 Convolutional NMF	35
2.5 Constrained NMF	38
2.5.1 NMF with Sparsity Constraints	38
2.5.2 Uni-Orthogonal NMF	39
2.5.3 NMF with Temporal Smoothness Constraints	43
3 Tensor Factorization Models	45
3.1 Introduction	45

3.2	Definitions	46
3.3	PARAFAC and Nonnegative Tensor Factorization models	47
3.3.1	PARAFAC with Orthogonality Constraints	50
3.4	Shift Invariant Multilinear Decomposition Model	53
4	Introduction to Functional Magnetic Resonance Imaging and Results	67
4.1	Introduction	67
4.2	MR Physics and BOLD Imaging	67
4.3	The Scanning Session	70
4.4	Modeling the BOLD Response	71
4.5	Data Analysis	73
4.5.1	Preprocessing	74
4.5.2	Task-Related Data Analysis	76
4.6	Experiments & Results	77
5	Discussion & Future Work	85
5.1	Conclusion	85
5.2	Future Work	85
5.2.1	The Convolutional PARAFAC model	85
5.2.2	Tensor Rank Estimation	86
5.2.3	Higher-order tensors	86
5.2.4	The Tucker model	86
5.2.5	Statistical analysis methods on subject variability factor	86

Appendices

A	87
A.1	Derivative of f_F with respect to matrix \mathbf{S}	87
A.2	Splitting of $\frac{\partial f_F}{\partial \mathbf{S}}$ into nonnegative parts	89
A.3	Derivative of f_F with respect to matrix $\boldsymbol{\tau}$	93
A.4	Hessian of f_F with respect to matrix $\boldsymbol{\tau}$	95
A.5	Derivative of $f_{\mathbf{V}}$ with respect to matrix $\boldsymbol{\tau}$	96
A.6	Hessian of $f_{\mathbf{V}}$ with respect to matrix $\boldsymbol{\tau}$	98

Bibliography	101
---------------------	-----------	------------

List of Figures

1.1	An example of fMRI data [1].	13
1.2	Abstract view of the BSS problem.	14
2.1	An example system of the Shifted NMF model.	25
2.2	An example system of the Convolutional NMF model.	35
3.1	An example illustration of the PARAFAC model in the BSS framework. . .	47
4.1	MRI Scanner Cutaway.	68
4.2	Nuclear Magnetic Resonance (NMR).	69
4.3	A hypothetical BOLD response (black curve) to a constant 10sec neural activation (gray curve) [2].	70
4.4	Two popular models of the HRF [2].	73
4.5	An example of fMRI time series preprocessing stages.	78
4.6	Temporal profile of rank-one component that corresponds to left visual discrimination - left hand coordination.	79
4.7	Spatial profile of rank-one component that corresponds to left visual discrimination - left hand coordination brain areas. The map was thresholded such that the 5% of the voxels with the largest voxel absolute score are shown.	80
4.8	Temporal profile of rank-one component that corresponds to right visual discrimination - right hand coordination.	81
4.9	Spatial profile of rank-one component that corresponds to right visual discrimination - right hand coordination brain areas. The map was thresholded such that the 5% of the voxels with the largest voxel absolute score are shown.	81
4.10	Temporal profile of rank-one component that corresponds to both left visual discrimination - left hand coordination and right visual discrimination - right hand coordination hemodynamic responses.	82

- 4.11 Spatial profile of rank-one component that corresponds to both left visual discrimination - left hand coordination and right visual discrimination - right hand coordination brain areas. The map was thresholded such that the 5% of the voxels with the largest voxel score are shown. 83
- 4.12 Spatial delay map of rank-one component that corresponds to both left visual discrimination - left hand coordination and right visual discrimination - right hand coordination brain areas. The map contains the delays that correspond to the 5% of the voxels with the largest voxel score are shown. . 84

List of Abbreviations

BSS	Blind Source Separation
PCA	Principal Components Analysis
ICA	Independent Component Analysis
NMF	Nonnegative Matrix Factorization
NTF	Nonnegative Tensor Factorization
LS	Least Squares
NNLS	Nonnegative Least Squares
PARAFAC or CP	Parallel Factor Analysis
SCP	Shift Invariant Multilinear Decomposition
DFT	Discrete Fourier Transform
IDFT	Inverse Discrete Fourier Transform
EEG	Electroencephalography
MEG	Magnetoencephalography
NIRS	Near-infrared spectroscopy
MR	Magnetic Resonance
NMR	Nuclear Magnetic Resonance
fMRI	Functional Magnetic Resonance Imaging
BOLD	Blood Oxygenation Level Dependent
HRF	Hemodynamic Response Function

TR

Repetition Time

Chapter 1

Introduction

Functional magnetic resonance imaging or functional MRI (fMRI) is a non-invasive functional neuroimaging procedure that measures brain activity by detecting changes associated with blood flow, over time. This technique relies on the fact that cerebral blood flow and neuronal activation are coupled, since when neural activity increases in an brain area, the metabolic demands in this area rise. Thus, the vascular system concentrates oxygen (oxygenated hemoglobin) into the area.

In fMRI data, a brain is represented by a finite set of volume elements (voxels). For each voxel, we have a time series that indicates the concentration of oxygen in this area over time. These time series are known as blood oxygenation level dependent (BOLD) signals. Purpose of fMRI data analysis is to determine which brain areas are activated, when a specific task is performed, based on the BOLD signals analysis. Hence, brain activation maps related to specific tasks can be obtained. This procedure can be shown very useful in understanding how the human brain works. Also, the study of how brain activations maps, as well as how activation time patterns change over different trials, can be used for diagnostic purposes. For example, fMRI could provide an in vivo means to investigate alterations in brain function related to the earliest symptoms of Alzheimer's disease, possibly before development of significant irreversible structural damage.

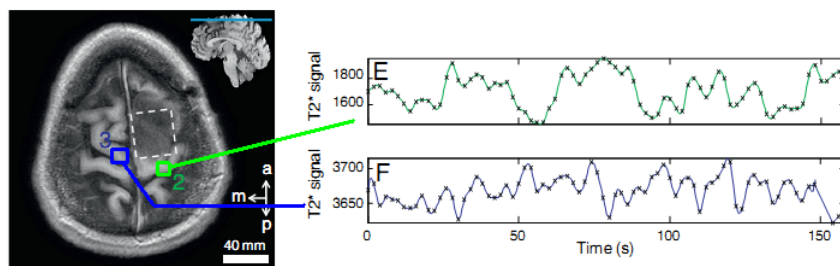


Figure 1.1: An example of fMRI data [1].

The BOLD signal of a voxel could correspond to oxygenation changes that are not only related to the specific task that we study, but also related to irrelevant factors. Thus, in order to isolate the signals that we are interested in, we can consider the problem of fMRI data analysis as a blind source separation problem (BSS). Blind source separation

(BSS) refers to the problem of extracting a set of source signals from a set of mixed signals, without using prior information (or with very little prior information) about the source signals or the mixing process. The classical example of a source separation problem is the cocktail party problem, where a number of people are talking simultaneously in a room (for example, at a cocktail party), and a listener is trying to follow one of the discussions. The human brain can handle this sort of auditory source separation problem, but it is a difficult problem in digital signal processing. Hence, BSS aims in enhancing noisy speech in real world environments and the applications are not just limited to speech/audio processing but also include topics in astronomical, satellite, econometric, and biomedical signal processing.

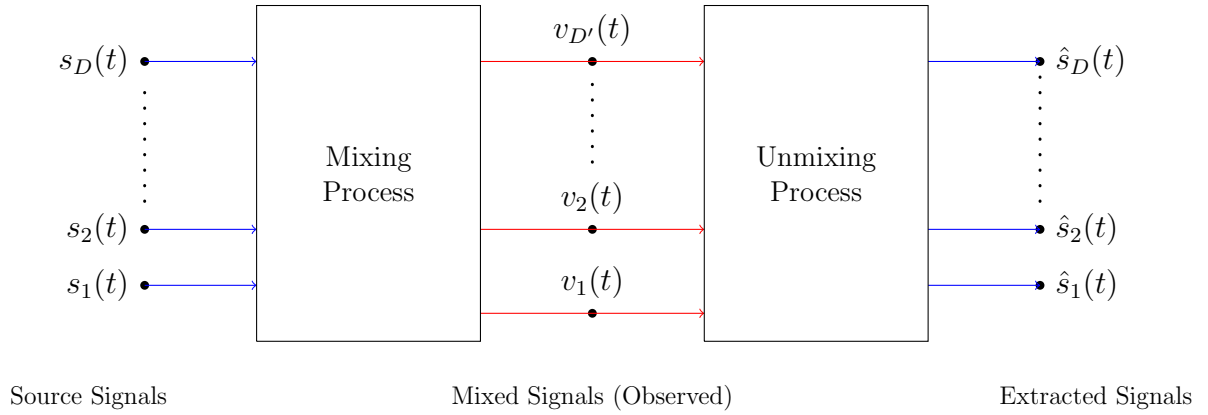


Figure 1.2: Abstract view of the BSS problem.

BSS problems are, in general, highly under-determined, since lack of prior knowledge (number of sources, characteristics of the source signals and the mixing procedure) may lead to a set of multiple solutions for the same problem. A variety of methods in addressing the BSS problem have been proposed in the literature. The most popular of them, among others, are principal component analysis (PCA), independent component analysis (ICA), nonnegative matrix factorization (NMF), as well as tensor factorizations models.

Tensors are mathematical objects that have recently gained great popularity due to their ability to model multi-way (multidimensional) data dependencies [3], [4], [5]. Tensors often offer more natural representations of data, e.g., consider video, which consists of obviously correlated images over time. Tensor factorization (or decomposition) into latent factors is very important for numerous tasks, such as BSS, feature selection, dimensionality reduction, multiway clustering, data visualization and interpretation, and others. The Canonical Decomposition or Canonical Polyadic Decomposition (CANDECOMP or CPD), also known as Parallel Factor Analysis (PARAFAC), and the Tucker Decomposition are

the two most widely used tensor factorization models.

1.1 Purpose

In this work, we focus on how tensor factorization models can be used in BSS problems and particularly in fMRI. We begin by presenting the NMF model and extensions of this model, Shifted NMF and Convolutional NMF, which assume the existence of time propagation delays between the sources and the sensors, in the framework of BSS problems. Next, we extend the NMF and the Shifted NMF models into tensor decomposition models for BSS problems. Finally, we apply tensor factorization models in real fMRI data from different subjects performing the same task, in order to obtain common brain activation maps and common descriptions of activation time patterns, related to the specific task.

1.2 Notation

Scalars are denoted by small letters, vectors, matrices, and tensors are denoted by small, capital, and calligraphic capital bold letters, respectively; for example, x , \mathbf{x} , \mathbf{X} , and \mathcal{X} . $\boldsymbol{\tau}$ exceptionally denotes a matrix. Sets are denoted by blackboard bold capital letter; for example, \mathbb{U} . Specifically, \mathbb{Z} , \mathbb{R} , and \mathbb{C} denote the sets of integer, real, and complex numbers, respectively. \mathbb{R}_+ denotes the set of real nonnegative numbers. $\mathbb{R}^{I \times J}$ denotes the set of $(I \times J)$ real matrices. $\mathbb{R}_+^{I \times J}$ denotes the set of $(I \times J)$ real nonnegative matrices. $\mathbb{C}^{I \times J}$ denotes the set of $(I \times J)$ complex matrices. $\mathbb{R}^{I \times J \times K}$ denotes the set of $(I \times J \times K)$ real tensors. $\mathbb{R}_+^{I \times J \times K}$ denotes the set of $(I \times J \times K)$ real tensors. $\mathbb{C}^{I \times J \times K}$ denotes the set of $(I \times J \times K)$ complex tensors. $\|\cdot\|_F$ denotes the Frobenius norm of the tensor or matrix argument. Inequality $\mathbf{A} \geq \mathbf{0}$ means that matrix \mathbf{A} has nonnegative elements. The transpose, conjugate and hermitian matrices of a matrix \mathbf{A} are denoted by \mathbf{A}^T , \mathbf{A}^* , and \mathbf{A}^H , respectively. The outer product of two vectors \mathbf{a} and \mathbf{b} is denoted as $\mathbf{a} \circ \mathbf{b}$ (see Definition 3.2.1), the Kronecker product of two matrices \mathbf{A} and \mathbf{B} is denoted as $\mathbf{A} \otimes \mathbf{B}$ (see Definition 3.2.2), and the Khatri-Rao product of two matrices \mathbf{A} and \mathbf{B} is denoted as $\mathbf{A} \circledast \mathbf{B}$ (see Definition 3.2.3). Finally, we introduce some Matlab style notations. $\mathbf{A}_{:,l}$ and $\mathbf{A}_{k,:}$ denote the l^{th} column and the k^{th} row of a matrix \mathbf{A} , respectively.

1.3 Thesis Outline

The thesis is organized as follows:

- In Chapter 2, we present the NMF model in the framework of BSS problems, as well as how the NMF model can be extended in order to incorporate source signals propagation time delays in anechoic environments (Shifted NMF) and echoic environments (Convolutional NMF).
- In Chapter 3, we present the PARAFAC and the nonnegative tensor factorization (NTF) models in the framework of BSS problems, as well as how these models can be extended in order to incorporate source signals time delays in anechoic environments (Shift Invariant Tensor Decomposition).
- In Chapter 4, we introduce the functional magnetic resonance imaging (fMRI) and we present results of processing real f-MRI data, using the PARAFAC model with orthogonality constraints and the SCP model as presented in Chapter 3.
- Finally, in Chapter 5, we conclude our work and make suggestions for future work.

Chapter 2

Nonnegative Matrix Factorization Models

2.1 Introduction

The main subject of this chapter is the factorization of a nonnegative matrix $\mathbf{V} \in \mathbb{R}_+^{N \times M}$ into a pair of nonnegative matrices $\mathbf{W} \in \mathbb{R}_+^{N \times D}$ and $\mathbf{S} \in \mathbb{R}_+^{D \times M}$, when $D \leq \min\{N, M\}$, such that

$$\mathbf{V} = \mathbf{W}\mathbf{S}. \quad (2.1)$$

Nonnegative matrix factorization (NMF) can be employed in many applications, as many real-world data are nonnegative and the corresponding hidden components have a physical meaning only when they are nonnegative. Factors \mathbf{W} and \mathbf{S} may have different interpretations in different applications [4]. In a blind source separation problem (BSS), matrix \mathbf{W} plays the role of mixing matrix, while matrix \mathbf{S} expresses the source signals. Specifically, each row of matrix \mathbf{S} expresses the time course of a single source signal and each column of matrix \mathbf{W} expresses the proportion in which each source appears to sensors, since

$$\mathbf{V} = \sum_{d=1}^D \mathbf{W}_{:,d} \mathbf{S}_{d,:} = \sum_{d=1}^D \mathbf{W}_{:,d} \circ \mathbf{S}_{:,d}^T, \quad (2.2)$$

where \circ denotes the outer product of two vectors.

Thus, each column of \mathbf{V} is reconstructed using a linear combination of nonnegative basis elements (columns of \mathbf{W}). Moreover, bases can only be summed up (since \mathbf{S} is nonnegative) in order to approximate the original data matrix \mathbf{V} . So, NMF leads to a part-based representation in which a zero-value represents the absence and a positive number represents the presence of a source.

In general, NMF is more complicated. In many cases of BSS, for example, the number of sources D is unknown. Even if D were known, there exist more than one pairs of \mathbf{W} and \mathbf{S} matrices that fulfil equation (2.1). For every invertible matrix $\mathbf{\Pi} \in \mathbb{R}^{D \times D}$, where

$\mathbf{W}\mathbf{\Pi} = \mathbf{W}' \geq \mathbf{0}$ and $\mathbf{\Pi}^{-1}\mathbf{S} = \mathbf{S}' \geq \mathbf{0}$, we have

$$\mathbf{W}'\mathbf{S}' = \mathbf{W}\mathbf{\Pi}\mathbf{\Pi}^{-1}\mathbf{S} = \mathbf{W}\mathbf{S} = \mathbf{V}. \quad (2.3)$$

From here on, we will refer to this ambiguity as rotation ambiguity¹. Also, for every $\alpha \in \mathbb{R}_{>0}$, let $\alpha\mathbf{W} = \mathbf{W}' \geq \mathbf{0}$ and $\frac{1}{\alpha}\mathbf{S} = \mathbf{S}' \geq \mathbf{0}$. Then

$$\mathbf{W}'\mathbf{S}' = \alpha\mathbf{W}\frac{1}{\alpha}\mathbf{S} = \mathbf{W}\mathbf{S} = \mathbf{V}. \quad (2.4)$$

From here on, we will refer to this ambiguity as scalar scaling ambiguity. Therefore, given a matrix \mathbf{V} and the number of sources D , recovering matrices \mathbf{W} and \mathbf{S} is not trivial and requires more assumptions.

Consider now a BSS problem where columns of \mathbf{V} correspond to measurements in different time instances. In this case, we make three more assumptions when we select the NMF model. The first one is that the mixing process is linear. the second one is that the mixing matrix \mathbf{W} is time invariant. The third one is that all signals arrive at the sensors at the same time, i.e. without relative delays. In this chapter, we study how the NMF model can be extended in order to accommodate propagation delays in anechoic and echoic propagation environments, assuming that the number of sources is known, the mixing process is linear and the mixing matrix \mathbf{W} is time invariant, as well as how rotation ambiguity can be handled.

2.2 Problem Formulation

2.2.1 Cost Functions

To find an approximate factorization $\mathbf{V} \approx \mathbf{W}\mathbf{S}$, we need first to define cost functions that quantify the quality of the approximation. In general, such a cost function can be constructed using some measure of distance between two nonnegative matrices \mathbf{A} and \mathbf{B} . Many functions can be used for this purpose with each one penalizing differently non-zero distances.

One particularly useful measure of the distance between matrices \mathbf{A} and \mathbf{B} is defined

¹A rigorous definition of the rotation ambiguity would require matrix $\mathbf{\Pi}$ to be orthogonal, i.e. $\mathbf{\Pi}\mathbf{\Pi}^T = \mathbf{I}$.

as

$$\|\mathbf{A} - \mathbf{B}\|_F = \sqrt{\sum_{ij} (\mathbf{A}_{i,j} - \mathbf{B}_{i,j})^2}, \quad (2.5)$$

and is usually termed as Frobenius norm. The Frobenius norm generalizes the Euclidean norm from vectors to matrices. It is lower bounded by zero, and vanishes if, and only if, $\mathbf{A} = \mathbf{B}$. Penalizing behaviour of this function gets more intense as absolute error grows, due to the quadratic factors. Another useful measure is the following:

$$D(\mathbf{A}||\mathbf{B}) = \sum_{ij} \left(\mathbf{A}_{i,j} \log \frac{\mathbf{A}_{i,j}}{\mathbf{B}_{i,j}} - \mathbf{A}_{i,j} + \mathbf{B}_{i,j} \right). \quad (2.6)$$

Like the Euclidean distance, this is also lower-bounded by zero, and vanishes if, and only if, $\mathbf{A} = \mathbf{B}$. However, it is not a “distance”, because it is not symmetric in \mathbf{A} and \mathbf{B} , and does not satisfy the triangle inequality. It reduces to Kullback-Leibler divergence, or relative entropy, when $\sum_{ij} \mathbf{A}_{i,j} = \sum_{ij} \mathbf{B}_{i,j} = 1$, so that \mathbf{A} and \mathbf{B} can be regarded as normalized probability distributions.

We now consider two alternative formulations of NMF as constrained optimization problems, using the two cost functions, as declared above:

$$\begin{aligned} & \underset{\mathbf{W}, \mathbf{S}}{\text{minimize}} && f_F(\mathbf{W}, \mathbf{S}) := \|\mathbf{V} - \mathbf{WS}\|_F^2 \\ & \text{subject to} && \mathbf{W} \geq \mathbf{0}, \\ & && \mathbf{S} \geq \mathbf{0}, \end{aligned} \quad (2.7)$$

$$\begin{aligned} & \underset{\mathbf{W}, \mathbf{S}}{\text{minimize}} && f_D(\mathbf{W}, \mathbf{S}) := D(\mathbf{V}||\mathbf{WS}) \\ & \text{subject to} && \mathbf{W} \geq \mathbf{0}, \\ & && \mathbf{S} \geq \mathbf{0}. \end{aligned} \quad (2.8)$$

2.2.2 Biconvexity

Let $\mathbb{X} \subseteq \mathbb{R}^{N \times D}$ and $\mathbb{Y} \subseteq \mathbb{R}^{D \times M}$ be two non-empty convex sets, and let $\mathbb{B} \subseteq \mathbb{X} \times \mathbb{Y}$. Then, we define the \mathbf{X} - and \mathbf{Y} - sections of \mathbb{B} as follows:

$$\begin{aligned} \mathbb{B}_{\mathbf{X}} &\triangleq \{\mathbf{Y} \in \mathbb{Y} : (\mathbf{X}, \mathbf{Y}) \in \mathbb{B}\}, \\ \mathbb{B}_{\mathbf{Y}} &\triangleq \{\mathbf{X} \in \mathbb{X} : (\mathbf{X}, \mathbf{Y}) \in \mathbb{B}\}. \end{aligned} \quad (2.9)$$

Definition 2.2.2.1 [6] *The set $\mathbb{B} \subseteq \mathbb{X} \times \mathbb{Y}$ is called a **biconvex set** on $\mathbb{X} \times \mathbb{Y}$ or **biconvex** for short, if $\mathbb{B}_{\mathbf{X}}$ is convex for every $\mathbf{X} \in \mathbb{X}$ and $\mathbb{B}_{\mathbf{Y}}$ is convex for every $\mathbf{Y} \in \mathbb{Y}$.*

Definition 2.2.2.2 [6] *A function $f : \mathbb{B} \rightarrow \mathbb{R}$ on a biconvex set $\mathbb{B} \subseteq \mathbb{X} \times \mathbb{Y}$ is called a **biconvex function** on \mathbb{B} or **biconvex**,, for short, if*

$$f_{\mathbf{X}}(\bullet) \triangleq f(\mathbf{X}, \bullet) : \mathbb{B}_{\mathbf{X}} \rightarrow \mathbb{R} \quad (2.10)$$

is a convex function on $\mathbb{B}_{\mathbf{X}}$ for every fixed $\mathbf{X} \in \mathbb{X}$ and

$$f_{\mathbf{Y}}(\bullet) \triangleq f(\bullet, \mathbf{Y}) : \mathbb{B}_{\mathbf{Y}} \rightarrow \mathbb{R} \quad (2.11)$$

is a convex function on $\mathbb{B}_{\mathbf{Y}}$ for every fixed $\mathbf{Y} \in \mathbb{Y}$.

Definition 2.2.2.3 [6] *An optimization problem of the form*

$$\underset{(\mathbf{X}, \mathbf{Y}) \in \mathbb{B}}{\text{minimize}} \quad f(\mathbf{X}, \mathbf{Y}) \quad (2.12)$$

*is said to be a **biconvex optimization problem** or **biconvex** for short, if the feasible set \mathbb{B} is biconvex on $\mathbb{X} \times \mathbb{Y}$, and the objective function f is biconvex on \mathbb{B} .*

Optimization problems (2.7)-(2.8) follow definition 2.2.2.3, so these problems are biconvex. Different from convex optimization problems, biconvex problems may have a large number of local minima. Thus, the question arises whether the convex substructures of a biconvex optimization problem can be utilized more efficiently for the solution of such problems than in the case of general non-convex optimization problems. For this purpose, next, we discuss the alternating optimization technique for biconvex minimization problems which exploits the convex substructures of the problem.

2.2.3 Alternating Optimization

Alternating optimization (AO) is an iterative algorithmic scheme for the solution of optimization problems. The idea underlying AO is to replace the joint optimization of an objective function f over all variables with a sequence of easier optimizations involving grouped subsets of the variables. This process is applied iteratively, until convergence is met.

Let us now consider the problems in expressions (2.7)–(2.8). Recalling the convex substructures of these problems, AO exploits biconvexity by breaking down the initial

problem into a sequence of convex subproblems, until optimum points are found for both variables. Therefore, problems (2.7)–(2.8) can be solved through AO by the following algorithmic scheme.

Algorithm: Alternating Optimization in NMF

Input: $\mathbf{V} \in \mathbb{R}_+^{N \times M}$, $\mathbf{W}^0 \in \mathbb{R}_+^{N \times D}$, $\mathbf{S}^0 \in \mathbb{R}_+^{D \times M}$

Output: $\mathbf{W} \in \mathbb{R}_+^{N \times D}$, $\mathbf{S} \in \mathbb{R}_+^{D \times M}$

while *convergence criterion is not met* **do**

$\mathbf{S}^{k+1} = \underset{\mathbf{S} \geq 0}{\operatorname{argmin}} f(\mathbf{W}^k, \mathbf{S})$

$\mathbf{W}^{k+1} = \underset{\mathbf{W} \geq 0}{\operatorname{argmin}} f(\mathbf{W}, \mathbf{S}^{k+1})$

end

Now, the only that remains abstract is to define how a function can be minimized. To do so, definitions of first and second derivative are needed.

2.2.4 Derivatives for Matrix functions

Concept of differentiation concerns approximation of functions by linear (specifically, affine) functions in case of first order approximation and by quadratic functions in case of second order approximation. Next, we quote some basic definitions.

Definition 2.2.4.1 [7] *Let f be a differentiable real valued function of the real $(m \times n)$ matrix $\mathbf{X} = [\mathbf{X}_{i,j}]$. Then,*

$$\frac{\partial f}{\partial \mathbf{X}} = \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} = \left[\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}_{i,j}} \right] = \begin{bmatrix} \frac{\partial f}{\partial \mathbf{X}_{1,1}}(\mathbf{X}) & \cdots & \frac{\partial f}{\partial \mathbf{X}_{1,n}}(\mathbf{X}) \\ \vdots & \ddots & \vdots \\ \frac{\partial f}{\partial \mathbf{X}_{m,1}}(\mathbf{X}) & \cdots & \frac{\partial f}{\partial \mathbf{X}_{m,n}}(\mathbf{X}) \end{bmatrix} \quad (m \times n) \quad (2.13)$$

is the matrix of first order partial derivatives of $f(\mathbf{X})$ and

$$\left. \frac{\partial f}{\partial \mathbf{X}} \right|_{\mathbf{X}=\mathbf{X}_0} = \left. \frac{\partial f}{\partial \mathbf{X}} \right|_{\mathbf{X}_0} = \frac{\partial f(\mathbf{X}_0)}{\partial \mathbf{X}} \equiv \left[\left. \frac{\partial f(\mathbf{X})}{\partial \mathbf{X}_{i,j}} \right|_{\mathbf{X}=\mathbf{X}_0} \right] \quad (m \times n) \quad (2.14)$$

is the matrix of first order partial derivatives of $f(\mathbf{X})$ evaluated at the $(m \times n)$ matrix \mathbf{X}_0 .

In order to proceed to the second order derivative, we employ the vectorization operator. Vectorization of a matrix is a linear transformation which converts the matrix into a column

vector, i.e.

$$\text{vec}(\mathbf{X}) = [\mathbf{X}_{1,1}, \dots, \mathbf{X}_{m,1}, \mathbf{X}_{1,2}, \dots, \mathbf{X}_{m,2}, \dots, \mathbf{X}_{1,n}, \dots, \mathbf{X}_{m,n}]^T. \quad (2.15)$$

Definition 2.2.4.2 [7] *Let f be a doubly differentiable real valued function of the real $(m \times n)$ matrix $\mathbf{X} = [\mathbf{X}_{i,j}]$. Then,*

$$\frac{\partial^2 f}{\partial \text{vec}(\mathbf{X}) \partial \text{vec}(\mathbf{X})^T} = \frac{\partial^2 f(\mathbf{X})}{\partial \text{vec}(\mathbf{X}) \partial \text{vec}(\mathbf{X})^T} \quad (mn \times mn) \quad (2.16)$$

is the Hessian matrix of the second order partial derivatives of $f(\mathbf{X})$. Here, it is important to note the order in which the partial derivatives are arranged. They have the same order as for the function $f(\text{vec}(\mathbf{X}))$. Accordingly,

$$\left. \frac{\partial^2 f}{\partial \text{vec}(\mathbf{X}) \partial \text{vec}(\mathbf{X})^T} \right|_{\mathbf{X}=\mathbf{X}_0} = \frac{\partial^2 f(\mathbf{X}_0)}{\partial \text{vec}(\mathbf{X}) \partial \text{vec}(\mathbf{X})^T} \quad (2.17)$$

is the Hessian matrix evaluated at $\mathbf{X} = \mathbf{X}_0$.

2.2.5 Multiplicative Method

One of the most popular approaches of solving the NMF problem is the one proposed in [8], called Multiplicative Method. This method is an iterative procedure, like AO, and can be characterized as a variation of the gradient descent method. The main advantages of this method are easiness of implementation, the property of preserving nonnegativity constraints in each optimization step without additional operations, and the existence of theoretical results that prove convergence to a local minimum.

The pseudocode of this method for the solution of NMF problems follows, where MR_S^f and MR_W^f denote the multiplicative update rules for matrices \mathbf{S} and \mathbf{W} , respectively, given a cost function f .

Algorithm: Multiplicative Method in NMF

Input: $\mathbf{V} \in \mathbb{R}_+^{N \times M}$, $\mathbf{W}^0 \in \mathbb{R}_+^{N \times D}$, $\mathbf{S}^0 \in \mathbb{R}_+^{D \times M}$

Output: $\mathbf{W} \in \mathbb{R}_+^{N \times D}$, $\mathbf{S} \in \mathbb{R}_+^{D \times M}$

while *convergence criterion is not met* **do**

$\mathbf{S}^{k+1} = \text{MR}_S^f(\mathbf{W}^k, \mathbf{S}^k)$

$\mathbf{W}^{k+1} = \text{MR}_W^f(\mathbf{W}^k, \mathbf{S}^{k+1})$

end

Multiplicative Update Rules for NMF

Let $f(\mathbf{X})$ be a cost function of the nonnegative matrix \mathbf{X} . If the derivative of f with respect to \mathbf{X} can be written for all $d \in \{1, \dots, D\}$, $m \in \{1, \dots, M\}$ in the form

$$\left[\frac{\partial f(\mathbf{X})}{\partial \mathbf{X}} \right]_{d,m} = \left[\frac{\partial f(\mathbf{X})^+}{\partial \mathbf{X}} \right]_{d,m} - \left[\frac{\partial f(\mathbf{X})^-}{\partial \mathbf{X}} \right]_{d,m}, \quad (2.18)$$

where $\left[\frac{\partial f(\mathbf{X})^+}{\partial \mathbf{X}} \right]_{d,m} \geq 0$ and $\left[\frac{\partial f(\mathbf{X})^-}{\partial \mathbf{X}} \right]_{d,m} \geq 0$, then we define

$$\left[\frac{\partial f(\mathbf{X})^+}{\partial \mathbf{X}} \right]_{d,m} \quad \text{and} \quad \left[\frac{\partial f(\mathbf{X})^-}{\partial \mathbf{X}} \right]_{d,m} \quad (2.19)$$

as the positive and the negative part, respectively, of the derivative of f with respect to \mathbf{X} .

When the derivative is separable in the above way, the multiplicative update step, in the $k+1^{\text{th}}$ iteration, for all $d \in \{1, \dots, D\}$ and $m \in \{1, \dots, M\}$, is

$$\mathbf{X}_{d,m}^{k+1} = \mathbf{X}_{d,m}^k \frac{\left[\frac{\partial f(\mathbf{X})^-}{\partial \mathbf{X}} \right]_{d,m}}{\left[\frac{\partial f(\mathbf{X})^+}{\partial \mathbf{X}} \right]_{d,m}}. \quad (2.20)$$

A small constant ϵ is added to the numerator and the denominator to avoid division by zero or forcing $\mathbf{X}_{d,m}$ to zero. If the partial derivative is positive (i.e. $\left[\frac{\partial f(\mathbf{X})^+}{\partial \mathbf{X}} \right]_{d,m} > \left[\frac{\partial f(\mathbf{X})^-}{\partial \mathbf{X}} \right]_{d,m}$), $\mathbf{X}_{d,m}$ will decrease and vice versa if the partial derivative is negative. The property of automatically ensuring nonnegativity holds, since the updates are based on multiplication and division of purely nonnegative variables.

In case that square Euclidean distance or Kullback-Leibler divergence are selected as cost functions, we have

$$\begin{aligned} f_F(\mathbf{W}, \mathbf{S}) &= \|\mathbf{V} - \mathbf{WS}\|_F^2 & f_D(\mathbf{W}, \mathbf{S}) &= D(\mathbf{V} \parallel \mathbf{WS}) \\ \frac{\partial f_F(\mathbf{W}, \mathbf{S})}{\partial \mathbf{W}} &= \mathbf{WSS}^T - \mathbf{VS}^T & \frac{\partial f_D(\mathbf{W}, \mathbf{S})}{\partial \mathbf{W}} &= \mathbf{JS}^T - \left[\frac{\mathbf{V}}{\mathbf{WS}} \right] \mathbf{S}^T \\ \frac{\partial f_F(\mathbf{W}, \mathbf{S})}{\partial \mathbf{S}} &= \mathbf{W}^T \mathbf{WS} - \mathbf{W}^T \mathbf{V} & \frac{\partial f_D(\mathbf{W}, \mathbf{S})}{\partial \mathbf{S}} &= \mathbf{W}^T \mathbf{J} - \mathbf{W}^T \left[\frac{\mathbf{V}}{\mathbf{WS}} \right], \end{aligned}$$

where \mathbf{J} is the matrix with ones (has the same dimensions as matrix \mathbf{V}) and division operator denotes the Hadamard division. As can be seen, in both cases, the derivative satisfies

the above conditions, since all involving matrices are nonnegative in NMF problems. Theorems 1 and 2 state that square Euclidean distance and Kullback-Leibler divergence are non-increasing under multiplicative update steps. Proofs of these theorems can be found in [8] and are based on the usage of an auxiliary function similar to that used in Expectation Maximization Algorithm.

Theorem 1 [8] *The Euclidean distance $\|\mathbf{V} - \mathbf{WS}\|_F^2$ is non-increasing under the update rules*

$$\mathbf{S}_{d,m} = \mathbf{S}_{d,m} \frac{[\mathbf{W}^T \mathbf{V}]_{d,m}}{[\mathbf{W}^T \mathbf{WS}]_{d,m}} \quad \text{and} \quad \mathbf{W}_{n,d} = \mathbf{W}_{n,d} \frac{[\mathbf{VS}^T]_{n,d}}{[\mathbf{WSS}^T]_{n,d}}. \quad (2.21)$$

The Euclidean distance is invariant under these updates if, and only if, \mathbf{W} and \mathbf{S} are at a stationary point of the distance.

Theorem 2 [8] *The Kullback-Leibler divergence $D(\mathbf{V} \parallel \mathbf{WS})$ is non-increasing under the update rules*

$$\mathbf{S}_{d,m} = \mathbf{S}_{d,m} \frac{\sum_n \frac{\mathbf{W}_{n,d} \mathbf{V}_{n,m}}{[\mathbf{WS}]_{n,m}}}{\sum_n \mathbf{W}_{n,d}} \quad \text{and} \quad \mathbf{W}_{n,d} = \mathbf{W}_{n,d} \frac{\sum_m \frac{\mathbf{S}_{d,m} \mathbf{V}_{n,m}}{[\mathbf{WS}]_{n,m}}}{\sum_m \mathbf{S}_{d,m}}. \quad (2.22)$$

The Kullback-Leibler divergence is invariant under these updates if, and only if, \mathbf{W} and \mathbf{S} are at a stationary point of the Kullback-Leibler divergence.

Multiplicative Versus Additive Update Rules

It is useful to contrast these multiplicative updates with those arising from gradient descent. In particular, a simple additive update for \mathbf{S} , that reduces the squared distance based on gradient descent method, can be written as

$$\begin{aligned} \mathbf{S}_{d,m} &= \mathbf{S}_{d,m} - \boldsymbol{\eta}_{d,m} \left[\frac{\partial f_F(\mathbf{W}, \mathbf{S})}{\partial \mathbf{S}} \right]_{d,m} \\ &= \mathbf{S}_{d,m} + \boldsymbol{\eta}_{d,m} \left([\mathbf{W}^T \mathbf{V}]_{d,m} - [\mathbf{W}^T \mathbf{WS}]_{d,m} \right), \end{aligned} \quad (2.23)$$

where $\boldsymbol{\eta}$ is the gradient step size matrix. If all $\boldsymbol{\eta}_{d,m}$ are set equal to some small positive number, this is equivalent to conventional gradient. As long as this number is sufficiently small, the update should reduce $f_F(\mathbf{W}, \mathbf{S}) = \|\mathbf{V} - \mathbf{WS}\|_F^2$. Now, if we set

$$\boldsymbol{\eta}_{d,m} = \frac{\mathbf{S}_{d,m}}{[\mathbf{W}^T \mathbf{WS}]_{d,m}}, \quad (2.24)$$

then we obtain the update rule for \mathbf{S} that is given in **Theorem 1**. Note that this rescaling results in a multiplicative factor with the positive component of the gradient in the denominator and the absolute value of the negative component in the numerator of the factor.

For the Kullback-Leibler divergence, scaled gradient descent takes the form

$$\mathbf{S}_{d,m} = \mathbf{S}_{d,m} + \boldsymbol{\eta}_{d,m} \left(\sum_n \mathbf{W}_{n,d} \frac{\mathbf{V}_{n,m}}{[\mathbf{WS}]_{n,m}} - \sum_n \mathbf{W}_{n,d} \right). \quad (2.25)$$

Again, if the $\boldsymbol{\eta}_{d,m}$ are small and positive, this update reduces $D(\mathbf{V} \parallel \mathbf{WS})$. If we now set

$$\boldsymbol{\eta}_{d,m} = \frac{\mathbf{S}_{d,m}}{\sum_n \mathbf{W}_{n,d}}, \quad (2.26)$$

then we obtain the update rule for \mathbf{S} that is given in **Theorem 2**. This scaling can also be interpreted as a multiplicative rule with positive component of the gradient in the denominator and negative component at the numerator of the multiplicative factor.

2.3 Shifted NMF

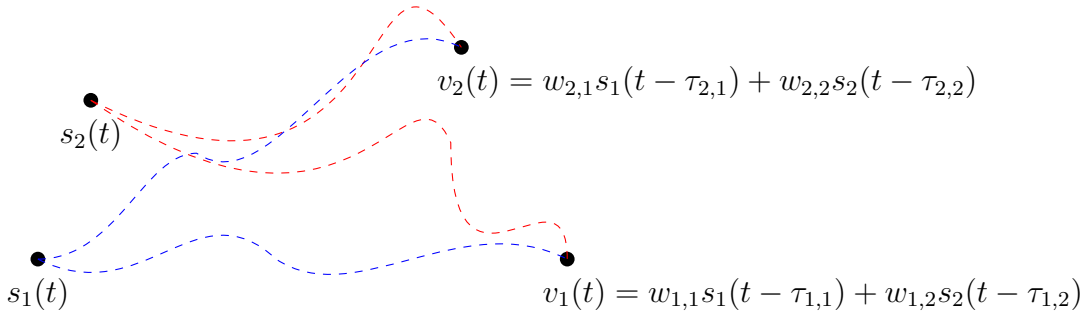


Figure 2.1: An example system of the Shifted NMF model.

In this section, we study how the NMF model can be transformed in order to incorporate signal propagation delays in anechoic environments. We call anechoic a propagation environment when it is echo and reflection free. Hence, in an anechoic environment each source signal reaches each sensor via the direct propagation path only (Figure 2.1), which may be modelled by two parameters: an attenuation factor w and a propagation time delay τ [9]. Assuming that the attenuation factors are nonnegative, Shifted NMF model can be

employed. Shifted NMF model was proposed in [10] and follows the form

$$\mathbf{V}_{n,m} = \sum_{d=1}^D \mathbf{W}_{n,d} \mathbf{S}_{d,m-\tau_{n,d}}, \quad (2.27)$$

for all $n \in \{1, \dots, N\}$, $m \in \{1, \dots, M\}$, where D denotes the number of sources, N denotes the number of sensors, M denotes the number of time instances, $\mathbf{S} \in \mathbb{R}_+^{D \times M}$ denotes the matrix of D source signals' time courses over M time instances, $\mathbf{W} \in \mathbb{R}_+^{N \times D}$ denotes the mixing matrix of D sources into N sensors, $\mathbf{V} \in \mathbb{R}_+^{N \times M}$ denotes the matrix of measurements from N sensors across M time instances, and $\tau_{n,d}$ denotes the propagation delay of the d^{th} source signal to the n^{th} sensor.

In order to analyse the Shifted NMF model, the transformation of the model in the frequency domain is convenient. In the following, we quote some useful relations that concern the transformations between time and frequency domains.

Definition 2.3.1 *Given a sequence of N samples x_n , where $n \in \{1, \dots, N\}$, the Discrete Fourier Transform (**DFT**) is defined as*

$$\tilde{x}_k = \frac{1}{\sqrt{N}} \sum_{n=1}^N x_n e^{-j2\pi(k-1)(n-1)/N}, \quad k \in \{1, \dots, N\}. \quad (2.28)$$

*The Inverse Discrete Fourier Transform (**IDFT**) is defined as*

$$x_n = \frac{1}{\sqrt{N}} \sum_{k=1}^N \tilde{x}_k e^{j2\pi(k-1)(n-1)/N}, \quad n \in \{1, \dots, N\}. \quad (2.29)$$

Parseval's Theorem for DFT *The total energy of a discrete time signal can be calculated by summing power-per-sample across time or spectral power across frequency. As a result,*

$$\sum_{n=1}^N |x_n|^2 = \sum_{k=1}^N |\tilde{x}_k|^2, \quad (2.30)$$

where sequence $\{\tilde{x}_k\}$ is the DFT of sequence $\{x_n\}$, both of length N .

Definition 2.3.2 We define as DFT matrix the Vandermonde matrix for the roots of unity, up to a normalization factor

$$\mathbf{Q} = \frac{1}{\sqrt{M}} \begin{bmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ 1 & \omega & \omega^2 & \omega^3 & \dots & \omega^{M-1} \\ 1 & \omega^2 & \omega^4 & \omega^6 & \dots & \omega^{2(M-1)} \\ 1 & \omega^3 & \omega^6 & \omega^9 & \dots & \omega^{3(M-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{M-1} & \omega^{2(M-1)} & \omega^{3(M-1)} & \dots & \omega^{(M-1)(M-1)} \end{bmatrix}, \quad (2.31)$$

where $\omega = e^{-j2\pi/M}$ and $\mathbf{Q}\mathbf{Q}^H = \mathbf{Q}^H\mathbf{Q} = \mathbf{I}_M$, where \mathbf{I}_M is the M -dimensional identity matrix.

Lemma 2.3.1 The DFT of a vector $\mathbf{a} \in \mathbb{R}^M$ is $\tilde{\mathbf{a}} = \mathbf{Q}\mathbf{a}$, where \mathbf{Q} is the $M \times M$ square DFT matrix. The IDFT of a vector $\tilde{\mathbf{a}} \in \mathbb{C}^N$ is $\mathbf{a} = \mathbf{Q}^H\tilde{\mathbf{a}}$.

Lemma 2.3.2 The row-wise DFT of a matrix $\mathbf{A} \in \mathbb{R}^{N \times M}$ is $\tilde{\mathbf{A}} = \mathbf{A}\mathbf{Q}$, where \mathbf{Q} is the $M \times M$ square DFT matrix. The row-wise IDFT of matrix $\tilde{\mathbf{A}} \in \mathbb{C}^{N \times M}$ is $\mathbf{A} = \tilde{\mathbf{A}}\mathbf{Q}^H$.

Lemma 2.3.3 The column-wise DFT of a matrix $\mathbf{A} \in \mathbb{R}^{N \times M}$ is $\tilde{\mathbf{A}} = \mathbf{Q}\mathbf{A}$, where \mathbf{Q} is the $N \times N$ square DFT matrix. The column-wise IDFT of matrix $\tilde{\mathbf{A}} \in \mathbb{C}^{N \times M}$ is $\mathbf{A} = \mathbf{Q}^H\tilde{\mathbf{A}}$.

We consider the transformation of the Shifted NMF model in the frequency domain as the row-wise Discrete Fourier Transform of matrix \mathbf{V} , that is,

$$\begin{aligned}
\tilde{\mathbf{V}}_{n,f} &= \frac{1}{\sqrt{M}} \sum_{m=1}^M \mathbf{V}_{n,m} e^{-j2\pi \frac{(f-1)(m-1)}{M}} \quad \forall n \in \{1, \dots, N\}, f \in \{1, \dots, M\} \\
&= \frac{1}{\sqrt{M}} \sum_{m=1}^M \left(\sum_{d=1}^D \mathbf{W}_{n,d} \mathbf{S}_{d,m-\tau_{n,d}} \right) e^{-j2\pi \frac{(f-1)(m-1)}{M}} \\
&= \frac{1}{\sqrt{M}} \sum_{d=1}^D \mathbf{W}_{n,d} \sum_{m=1}^M \mathbf{S}_{d,m-\tau_{n,d}} e^{-j2\pi \frac{(f-1)(m-1)}{M}} \\
&\stackrel{m'=m-\tau_{n,d}}{=} \frac{1}{\sqrt{M}} \sum_{d=1}^D \mathbf{W}_{n,d} \sum_{m'=1-\tau_{n,d}}^{M-\tau_{n,d}} \mathbf{S}_{d,m'} e^{-j2\pi \frac{(f-1)(m'+\tau_{n,d}-1)}{M}} \\
&= \frac{1}{\sqrt{M}} \sum_{d=1}^D \mathbf{W}_{n,d} \left(\sum_{m'=1-\tau_{n,d}}^{M-\tau_{n,d}} \mathbf{S}_{d,m'} e^{-j2\pi \frac{(f-1)(m'-1)}{M}} \right) e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \\
&= \sum_{d=1}^D \mathbf{W}_{n,d} \left(\frac{1}{\sqrt{M}} \sum_{m'=1}^M \mathbf{S}_{d,m'} e^{-j2\pi \frac{(f-1)(m'-1)}{M}} \right) e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \quad (\text{due to circular shift}) \\
&= \sum_{d=1}^D \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{d,f}.
\end{aligned} \tag{2.32}$$

In matrix notation, the Shifted NMF model in the frequency domain can be expressed as

$$\tilde{\mathbf{V}}_{:,f} = \left[\mathbf{W} \odot e^{-j2\pi \frac{(f-1)\boldsymbol{\tau}}{M}} \right] \tilde{\mathbf{S}}_{:,f} \quad \forall f \in \{1, \dots, M\}, \tag{2.33}$$

where $\tilde{\mathbf{S}}$ denotes the row-wise DFT of matrix \mathbf{S} , \odot denotes the Hadamard product, and $\left[e^{-j2\pi \frac{(f-1)\boldsymbol{\tau}}{M}} \right]_{n,d} = e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}}$.

In order to recover matrices \mathbf{W} , \mathbf{S} , and $\boldsymbol{\tau}$, given a matrix \mathbf{V} and the number of sources D , we define the least squares cost function

$$f_F(\mathbf{W}, \mathbf{S}, \boldsymbol{\tau}) := \frac{1}{2} \sum_{n=1}^N \sum_{m=1}^M \left(\mathbf{V}_{n,m} - \sum_{d=1}^D \mathbf{W}_{n,d} \mathbf{S}_{d,m-\tau_{n,d}} \right)^2. \tag{2.34}$$

Then, estimates of the unknown quantities can be obtained by minimizing f_F . Due to Parseval's Theorem, the following equality is valid.

$$\begin{aligned} \sum_{n=1}^N \sum_{m=1}^M \left(\mathbf{v}_{n,m} - \sum_{d=1}^D \mathbf{w}_{n,d} \mathbf{s}_{d,m-\tau_{n,d}} \right)^2 &= \sum_{f=1}^M \left\| \tilde{\mathbf{v}}_{:,f} - \left[\mathbf{W} \odot e^{-j2\pi \frac{(f-1)\tau}{M}} \right] \tilde{\mathbf{S}}_{:,f} \right\|_2^2 \\ &\stackrel{(a)}{=} \sum_{f=1}^M \left\| \tilde{\mathbf{v}}_{:,f} - \left[\mathbf{W} \odot e^{-j2\pi \frac{(f-1)\tau}{M}} \right] \mathbf{S} \mathbf{Q}_{:,f} \right\|_2^2, \end{aligned} \quad (2.35)$$

where at (a) we use Lemma 2.3.2. Hence,

$$\begin{aligned} f_F(\mathbf{W}, \mathbf{S}, \boldsymbol{\tau}) &= \frac{1}{2} \sum_{n=1}^N \sum_{m=1}^M \left(\mathbf{v}_{n,m} - \sum_{d=1}^D \mathbf{w}_{n,d} \mathbf{s}_{d,m-\tau_{n,d}} \right)^2 \\ &= \frac{1}{2} \sum_{f=1}^M \left\| \tilde{\mathbf{v}}_{:,f} - \left[\mathbf{W} \odot e^{-j2\pi \frac{(f-1)\tau}{M}} \right] \mathbf{S} \mathbf{Q}_{:,f} \right\|_2^2 \\ &= \frac{1}{2} \sum_{f=1}^M \left\| \tilde{\mathbf{v}}_{:,f} - \left[\mathbf{W} \odot e^{-j2\pi \frac{(f-1)\tau}{M}} \right] \tilde{\mathbf{S}}_{:,f} \right\|_2^2 =: g_F(\mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau}). \end{aligned} \quad (2.36)$$

Therefore, solving the Shifted NMF model can be expressed as an optimization problem with the following equivalent forms

$$\underset{\mathbf{W} \geq \mathbf{0}, \mathbf{S} \geq \mathbf{0}, \boldsymbol{\tau}}{\text{minimize}} \quad f_F(\mathbf{W}, \mathbf{S}, \boldsymbol{\tau}) \quad \equiv \quad \underset{\mathbf{W} \geq \mathbf{0}, \tilde{\mathbf{S}} \mathbf{Q}^* \geq \mathbf{0}, \boldsymbol{\tau}}{\text{minimize}} \quad g_F(\mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau}). \quad (2.37)$$

The procedure of solving the above optimization problems, given a matrix \mathbf{V} and D , as described in [10], is based on alternately updating \mathbf{W} , \mathbf{S} , and $\boldsymbol{\tau}$, until a convergence criterion is met. Next, we analyse the update steps for matrices \mathbf{W} , \mathbf{S} , and $\boldsymbol{\tau}$, using the multiplicative update approach.

Algorithm: Shifted NMF with Multiplicative Update Method

Input: $\mathbf{V} \in \mathbb{R}_+^{N \times M}$, $\mathbf{W}^0 \in \mathbb{R}_+^{N \times D}$, $\mathbf{S}^0 \in \mathbb{R}_+^{D \times M}$, $\boldsymbol{\tau}_0 \in \mathbb{R}^{N \times D}$

Output: $\mathbf{W} \in \mathbb{R}_+^{N \times D}$, $\mathbf{S} \in \mathbb{R}_+^{D \times M}$, $\boldsymbol{\tau} \in \mathbb{R}^{N \times D}$

while *convergence criterion is not met* **do**

 update \mathbf{W} , for fixed \mathbf{S} and $\boldsymbol{\tau}$

 update \mathbf{S} , for fixed \mathbf{W} and $\boldsymbol{\tau}$

 update $\boldsymbol{\tau}$, for fixed \mathbf{S} and \mathbf{W}

end

W Update

We recall the transformation of the Shifted NMF model in the frequency domain (2.32)

$$\tilde{\mathbf{V}}_{n,f} = \sum_{d=1}^D \mathbf{W}_{n,d} \underbrace{e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{d,f}}_{\tilde{\mathbf{S}}_{d,f}^n} = \sum_{d=1}^D \mathbf{W}_{n,d} \tilde{\mathbf{S}}_{d,f}^n, \quad (2.38)$$

for all $n \in \{1, \dots, N\}$, $f \in \{1, \dots, M\}$. Then, the model in the time domain can be expressed as

$$\begin{aligned} \mathbf{V}_{n,m} &= \frac{1}{\sqrt{M}} \sum_{f=1}^M \tilde{\mathbf{V}}_{n,f} e^{-j2\pi \frac{(f-1)(m-1)}{M}} \\ &= \frac{1}{\sqrt{M}} \sum_{f=1}^M \left(\sum_{d=1}^D \mathbf{W}_{n,d} \tilde{\mathbf{S}}_{d,f}^n \right) e^{-j2\pi \frac{(f-1)(m-1)}{M}} \\ &= \sum_{d=1}^D \mathbf{W}_{n,d} \left(\frac{1}{\sqrt{M}} \sum_{f=1}^M \tilde{\mathbf{S}}_{d,f}^n e^{-j2\pi \frac{(f-1)(m-1)}{M}} \right) \\ &= \sum_{d=1}^D \mathbf{W}_{n,d} \mathbf{S}_{d,m}^n, \end{aligned} \quad (2.39)$$

for all $n \in \{1, \dots, N\}$, $m \in \{1, \dots, M\}$, where \mathbf{S}^n is the row-wise IDFT of matrix $\tilde{\mathbf{S}}^n$ ($\tilde{\mathbf{S}}_{d,f}^n = e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{d,f}$). In matrix notation, the final equation of (2.39), for all $n \in \{1, \dots, N\}$, can be written as

$$\mathbf{V}_{n,:} = \mathbf{W}_{n,:} \mathbf{S}^n. \quad (2.40)$$

Thus, matrix \mathbf{W} can be updated by performing the least-squares NMF multiplicative update, as presented in the subsection Multiplicative Update Rules for NMF, i.e.

$$\mathbf{W}_{n,d} \leftarrow \mathbf{W}_{n,d} \frac{\mathbf{V}_{n,:} [\mathbf{S}_{d,:}^n]^T}{\mathbf{W}_{n,:} \mathbf{S}^n [\mathbf{S}_{d,:}^n]^T}, \quad (2.41)$$

for all $n \in \{1, \dots, N\}$, $d \in \{1, \dots, D\}$.

S Update

Consider the least-squares cost function in the frequency domain, as defined in relation (2.37),

$$f_F(\mathbf{W}, \mathbf{S}, \boldsymbol{\tau}) = \frac{1}{2} \sum_{f=1}^M \left\| \tilde{\mathbf{V}}_{:,f} - \underbrace{\left[\mathbf{W} \odot e^{-j2\pi \frac{(f-1)\boldsymbol{\tau}}{M}} \right]}_{\mathbf{W}^f} \mathbf{S} \mathbf{Q}_{:,f} \right\|_2^2 = \frac{1}{2} \sum_{f=1}^M \left\| \tilde{\mathbf{V}}_{:,f} - \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \right\|_2^2. \quad (2.42)$$

Then, the derivative of f_F with respect to \mathbf{S} is equal to [A.1]

$$\frac{\partial f_F(\mathbf{W}, \mathbf{S}, \boldsymbol{\tau})}{\partial \mathbf{S}} = \Re \left\{ \sum_{f=1}^M \mathbf{W}^{fH} \left[\mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} - \tilde{\mathbf{V}}_{:,f} \right] \mathbf{Q}_{f,:}^* \right\}, \quad (2.43)$$

where $\mathbf{W}^{fH} = [\mathbf{W}^f]^H$. Since $\mathbf{Q}_{f,:}^* = \mathbf{e}_f^T \mathbf{Q}^*$, where \mathbf{e}_f is the f^{th} column of the M -dimensional identity matrix, and the matrix \mathbf{Q} is symmetric, the derivative of f_F with respect to \mathbf{S} is equal to

$$\begin{aligned} \frac{\partial f_F(\mathbf{W}, \mathbf{S}, \boldsymbol{\tau})}{\partial \mathbf{S}} &= \Re \left\{ \sum_{f=1}^M \mathbf{W}^{fH} \left[\mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} - \tilde{\mathbf{V}}_{:,f} \right] \mathbf{e}_f^T \mathbf{Q}^H \right\} \\ &= \Re \left\{ \left[\sum_{f=1}^M \mathbf{W}^{fH} \left[\mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} - \tilde{\mathbf{V}}_{:,f} \right] \mathbf{e}_f^T \right] \mathbf{Q}^H \right\}. \end{aligned} \quad (2.44)$$

Notice that

$$\begin{aligned} \sum_{f=1}^M \mathbf{W}^{fH} \left[\mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} - \tilde{\mathbf{V}}_{:,f} \right] \mathbf{e}_f^T &= \sum_{f=1}^M \mathbf{W}^{fH} \left[\mathbf{W}^f \tilde{\mathbf{S}}_{:,f} - \tilde{\mathbf{V}}_{:,f} \right] \mathbf{e}_f^T \\ &= \frac{\partial g_F(\mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau})}{\partial \tilde{\mathbf{S}}}. \end{aligned} \quad (2.45)$$

Hence,

$$\frac{\partial f_F(\mathbf{W}, \mathbf{S}, \boldsymbol{\tau})}{\partial \mathbf{S}} = \Re \left\{ \frac{\partial g_F(\mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau})}{\partial \tilde{\mathbf{S}}} \mathbf{Q}^* \right\} = \Re \left\{ \text{IDFT} \left[\frac{\partial g_F(\mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau})}{\partial \tilde{\mathbf{S}}} \right] \right\}. \quad (2.46)$$

Therefore, by taking the real part of the row-wise IDFT of the derivative in the frequency domain $\left(\frac{\partial g_F(\mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau})}{\partial \tilde{\mathbf{S}}}\right)$, the corresponding derivative in the time domain $\left(\frac{\partial f_F(\mathbf{W}, \mathbf{S}, \boldsymbol{\tau})}{\partial \mathbf{S}}\right)$ is obtained. Splitting the derivative in the frequency domain into what constitutes the positive and the negative part of the corresponding derivative in the time domain, gives [A.2]

$$\tilde{\mathbf{G}}^+ = \sum_{f=1}^M \mathbf{W}^{fH} \mathbf{W}^f \tilde{\mathbf{S}}_{:,f} \mathbf{e}_f^T, \quad \tilde{\mathbf{G}}^- = \sum_{f=1}^M \mathbf{W}^{fH} \tilde{\mathbf{V}}_{:,f} \mathbf{e}_f^T, \quad (2.47)$$

$$\mathbf{G}^+ = \Re \left\{ \tilde{\mathbf{G}}^+ \mathbf{Q}^H \right\}, \quad \mathbf{G}^- = \Re \left\{ \tilde{\mathbf{G}}^- \mathbf{Q}^H \right\}. \quad (2.48)$$

Consequently, by taking the real part of the row-wise IDFT of $\tilde{\mathbf{G}}^+$ and $\tilde{\mathbf{G}}^-$, the corresponding positive and negative part of the derivative of f_F with respect to \mathbf{S} can be computed. As a result, matrix \mathbf{S} can be updated using the multiplicative update as

$$\mathbf{S} \leftarrow \mathbf{S} \odot \frac{\mathbf{G}^-}{\mathbf{G}^+}, \quad (2.49)$$

where \odot denotes the Hadamard product and $\frac{\mathbf{A}}{\mathbf{B}}$ denotes the Hadamard division.

$\boldsymbol{\tau}$ Update

The delays $\boldsymbol{\tau}$ are unconstrained. Consider the least-squares cost function of Shifted NMF in the frequency domain

$$g_F(\mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau}) = \frac{1}{2} \sum_{f=1}^M \left\| \tilde{\mathbf{V}}_{:,f} - \left[\mathbf{W} \odot e^{-j2\pi \frac{(f-1)\boldsymbol{\tau}}{M}} \right] \tilde{\mathbf{S}}_{:,f} \right\|_2^2. \quad (2.50)$$

Define $\mathbf{T} \in \mathbb{R}^{ND \times 1} := \text{vec}(\boldsymbol{\tau})$, i.e. the vectorized form of matrix $\boldsymbol{\tau}$ such that $\mathbf{T}_{n+(d-1)N} = \tau_{n,d}$. Then, the derivative of g_F with respect to $\boldsymbol{\tau}_{n,d}$ is [A.3]

$$\mathbf{g}_{n+(d-1)N} = \left[\frac{\partial g_F}{\partial \mathbf{T}} \right]_{n+(d-1)N} = \left[\frac{\partial g_F}{\partial \boldsymbol{\tau}} \right]_{n,d} = 2\pi \frac{\mathbf{W}_{n,d}}{M} \sum_{f=1}^M (f-1) \Im \left\{ \tilde{\mathbf{S}}_{d,f}^* e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{E}}_{n,f} \right\}, \quad (2.51)$$

where $\tilde{\mathbf{E}}_{n,f} = \tilde{\mathbf{V}}_{n,f} - \sum_{d'=1}^D \mathbf{W}_{n,d'} \tilde{\mathbf{S}}_{d',f} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}}$. The Hessian of g_F has the following structure [A.4]:

$$\mathbf{H}_{t,t'} = \begin{cases} 0, & \text{if } n' \neq n, \\ \left(\frac{2\pi}{M}\right)^2 \mathbf{W}_{n,d} \mathbf{W}_{n,d'} \sum_{f=1}^M (f-1)^2 \Re \left\{ \tilde{\mathbf{S}}_{d,f} \tilde{\mathbf{S}}_{d',f}^* e^{j2\pi \frac{(f-1)(\tau_{n,d'} - \tau_{n,d})}{M}} \right\}, & \text{if } d' \neq d \text{ \& } n' = n, \\ \left(\frac{2\pi}{M}\right)^2 \mathbf{W}_{n,d} \sum_{f=1}^M (f-1)^2 \Re \left\{ \tilde{\mathbf{S}}_{d,f} \tilde{\mathbf{E}}_{n,f}^* e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} + \mathbf{W}_{n,d} \left| \tilde{\mathbf{S}}_{d,f} \right|^2 \right\}, & \text{if } d' = d \text{ \& } n' = n, \end{cases} \quad (2.52)$$

where $t := n + (d-1)N$ and $t' := n' + (d'-1)N$. As a result, $\boldsymbol{\tau}$ can be estimated by the Newton-Raphson method as

$$\mathbf{T} \leftarrow \mathbf{T} - \eta \mathbf{H}^{-1} \mathbf{g}, \quad (2.53)$$

where η is a step size parameter that is tuned to keep decreasing the cost function. Function g_F for fixed matrices \mathbf{W} and \mathbf{S} is not convex. Thus, the descent property of the Newton-Raphson method is not guaranteed. In that case, we use the Newton-Raphson method when Hessian is positive definite, otherwise, we use the gradient descent method.

However, the above method is still sensitive to local minima. Based on empirical observations, estimating the delays by the following cross-correlation procedure reduces the effect of local minima [10]. Let

$$\tilde{\mathbf{R}}_{n,f}^d = \tilde{\mathbf{V}}_{n,f} - \sum_{d' \neq d} \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{d',f}, \quad (2.54)$$

which is the signal at the n^{th} sensor at frequency $(f-1)$, when subtracting all but the d^{th} source out of $\tilde{\mathbf{V}}_{n,f}$. Then, the cross-correlation in the frequency domain between the d^{th} source signal and $\tilde{\mathbf{R}}_{n,f}^d$ is given by

$$\tilde{\mathbf{c}}_f = \tilde{\mathbf{R}}_{n,f}^{d*} \tilde{\mathbf{S}}_{d,f}. \quad (2.55)$$

Assuming that true matrices \mathbf{W} , \mathbf{S} , and $\boldsymbol{\tau}$ are available, relation (2.54) can be written as

$$\tilde{\mathbf{R}}_{n,f}^d = \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{d,f}. \quad (2.56)$$

Then, the cross-correlation terms take the following form

$$\tilde{\mathbf{c}}_f = \tilde{\mathbf{R}}_{n,f}^d \tilde{\mathbf{S}}_{d,f}^* = \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{d,f} \tilde{\mathbf{S}}_{d,f}^* = \mathbf{W}_{n,d} \left| \tilde{\mathbf{S}}_{d,f} \right|^2 e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}}, \quad (2.57)$$

and estimates of $\tau_{n,d}$ and $\mathbf{W}_{n,d}$ can be obtained, for all $n \in \{1, \dots, N\}$, $d \in \{1, \dots, D\}$, from

$$t = \underset{m}{\operatorname{argmax}} \mathbf{c}_m, \quad \tau_{n,d} = \operatorname{mod}(M + 1 - t, M), \quad \mathbf{W}_{n,d} = \sqrt{M} \frac{\mathbf{c}_t}{\|\tilde{\mathbf{S}}_{d,:}\|_2^2}, \quad (2.58)$$

where

$$\begin{aligned} \mathbf{c}_m &= \Re \left\{ (DFT[\tilde{\mathbf{c}}_f])_m \right\} \\ &= \Re \left\{ \frac{1}{\sqrt{M}} \sum_{f=1}^M \tilde{\mathbf{c}}_f e^{-j2\pi \frac{m(f-1)}{M}} \right\} \\ &= \Re \left\{ \frac{1}{\sqrt{M}} \sum_{f=1}^M \mathbf{W}_{n,d} |\tilde{\mathbf{S}}_{d,f}|^2 e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} e^{-j2\pi \frac{m(f-1)}{M}} \right\} \\ &= \frac{1}{\sqrt{M}} \mathbf{W}_{n,d} \sum_{f=1}^M |\tilde{\mathbf{S}}_{d,f}|^2 \Re \left\{ e^{j2\pi \frac{(f-1)(\tau_{n,d}-m)}{M}} \right\} \\ &\leq \frac{1}{\sqrt{M}} \mathbf{W}_{n,d} \|\tilde{\mathbf{S}}_{d,:}\|_2^2. \end{aligned} \quad (2.59)$$

Quantity $\frac{1}{\sqrt{M}} \mathbf{W}_{n,d} \sum_{f=1}^M |\tilde{\mathbf{S}}_{d,f}|^2 \Re \left\{ e^{j2\pi \frac{(f-1)(\tau_{n,d}-m)}{M}} \right\}$, for nonnegative $\mathbf{W}_{n,d}$, becomes maximum for $\tau_{n,d} = m$. In our experiments, we apply the cross-correlation procedure every 20 iterations.

Remarks

In general, the Shifted NMF model is not unique, since scaling and rotation ambiguities remain. Thus, additional constraints, such as sparsity, have proven useful [11]. Furthermore, prior information, such as smoothness, has also been proposed to improve the component identification capability [12]. The present algorithm for Shifted NMF can straightforwardly be extended to incorporate these constraints in order to improve the identification capability where the model in general is no guaranteed to find a unique decomposition. However, the component identification is in general difficult when the components are not sparse or the problem ill-conditioned.

The DFT is based on the assumption that the signals are periodic with period equal to the number of temporal samples (M). In general, this is not the case. However, by zero padding the ends or introducing a window function, periodicity can be enforced. Both

windowing and zero padding will favor small delays and particularly windowing is also computationally expensive.

At last, correspondence between the cost function in the time and the frequency domain does not exist for cost functions such as the Kullback-Leibler divergence. So, multiplicative update rules formation needs investigation.

2.4 Convolutional NMF

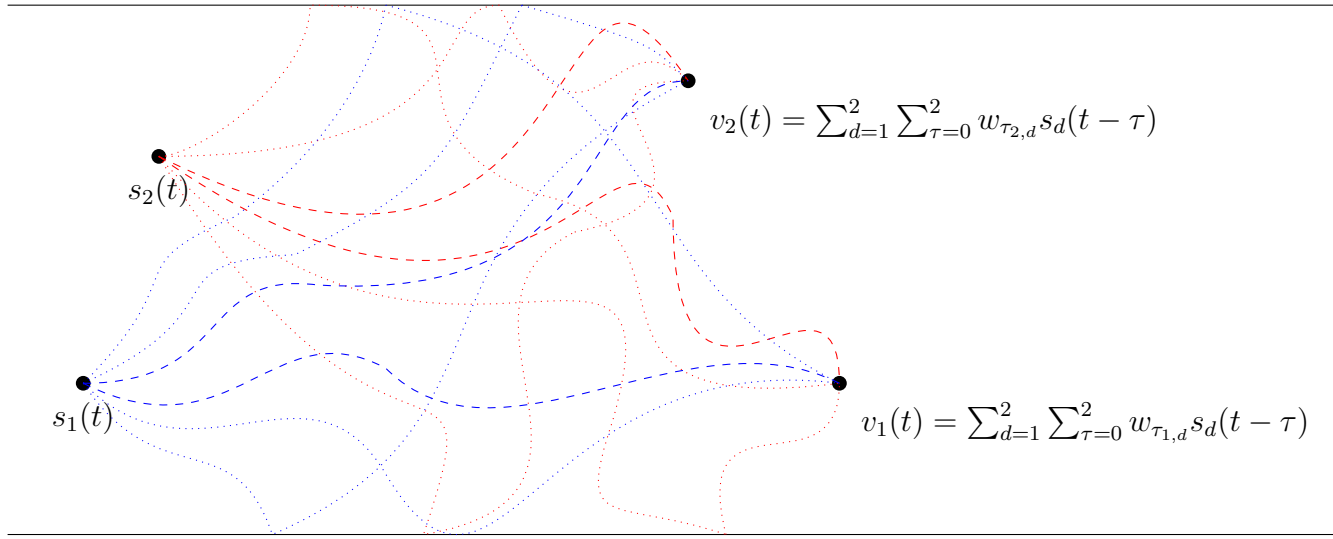


Figure 2.2: An example system of the Convolutional NMF model.

An extension of the Shifted NMF model is the Convolutional NMF model. The Convolutional NMF model is useful when the propagation environment is modelled as echoic, i.e. the model allows signal propagation via a number of reflection paths in addition to the direct propagation path (Figure 2.2). This model was presented in [13], where the Kullback-Leibler divergence was selected as distance function. Later, in [14], the same model was investigated for the square of Euclidean distance as distance function.

The Convolutional NMF model assuming T propagation paths, as proposed in [13], follows the form

$$\mathbf{V} = \sum_{\tau=0}^{T-1} \mathbf{W}_{\tau} \overset{\tau \rightarrow}{\mathbf{S}}, \quad (2.60)$$

where $\mathbf{W}_{\tau} \in \mathbb{R}_+^{N \times D}$ for all $\tau \in \{0, \dots, T-1\}$, is a set of T unknown matrices, $\mathbf{S} = \overset{0 \rightarrow}{\mathbf{S}} \overset{\tau \rightarrow}{\in} \mathbb{R}_+^{D \times M}$ represents the matrix of D source signals' time courses over M time instances, \mathbf{S}

is a τ column shifted version of \mathbf{S} . In other words, $\mathbf{S}^{\tau \rightarrow}$ denotes the τ positions (columns) shifting operator to the right, with the columns shifted in from outside the matrix set to zero. Analogously, $\mathbf{S}^{\tau \leftarrow}$ means that the columns of \mathbf{S} are shifted τ columns to the left.

Estimating unknown parameters of Convolutional NMF model, as proposed in [13] and [14], can be achieved by using the multiplicative updates approach, as

Algorithm: Convolutional NMF with Multiplicative Updates

Input: $\mathbf{V} \in \mathbb{R}_+^{N \times M}$, $\mathbf{W}_0^0 \in \mathbb{R}_+^{N \times D}$, $\mathbf{W}_1^0 \in \mathbb{R}_+^{N \times D}$, \dots , $\mathbf{W}_{(T-1)}^0 \in \mathbb{R}_+^{N \times D}$, $\mathbf{S}^0 \in \mathbb{R}_+^{D \times M}$
Output: $\mathbf{W}_0 \in \mathbb{R}_+^{N \times D}$, $\mathbf{W}_1 \in \mathbb{R}_+^{N \times D}$, \dots , $\mathbf{W}_{(T-1)} \in \mathbb{R}_+^{N \times D}$, $\mathbf{S} \in \mathbb{R}_+^{D \times M}$
while *convergence criterion is not met* **do**
 for $\tau = 0 : T - 1$ **do**
 update \mathbf{W}_τ , for fixed \mathbf{S} and all $\mathbf{W}_{\tau' \neq \tau}$
 update \mathbf{S} , for fixed $\mathbf{W}_0, \dots, \mathbf{W}_{T-1}$
 end
end

Next, we analyse the update steps for the Kullback-Leibler divergence and the squared Euclidean distance functions.

Kullback-Leibler Divergence

In case that the Kullback-Leibler divergence be chose as distance function, then solving the Convolutional NMF problem can be expressed as the optimization problem

$$\begin{aligned}
 &\text{minimize} \quad f_D(\mathbf{W}_0, \mathbf{W}_1, \dots, \mathbf{W}_{T-1}, \mathbf{S}) := \sum_{n=1}^N \sum_{m=1}^M \left(\mathbf{v}_{n,m} \log \frac{\mathbf{v}_{n,m}}{\Lambda_{n,m}} - \mathbf{v}_{n,m} + \Lambda_{n,m} \right) \\
 &\text{subject to} \quad \mathbf{W}_\tau \geq \mathbf{0}, \quad \tau = 0, \dots, T-1, \\
 &\quad \quad \quad \mathbf{S} \geq \mathbf{0},
 \end{aligned} \tag{2.61}$$

where $\Lambda = \sum_{\tau=0}^{T-1} \mathbf{W}_\tau^{\tau \rightarrow} \mathbf{S}$.

Then, the problem (2.61) can be solved using the multiplicative update rules. The update rules for this case will be the same as when performing NMF for each iteration of

τ , plus some shifting to appropriately line up the arguments, i.e.

$$\mathbf{S}^{k+1} \leftarrow \mathbf{S}^k \odot \frac{[\mathbf{W}_\tau^k]^T \begin{bmatrix} \mathbf{V} \\ \mathbf{\Lambda} \end{bmatrix}^{\tau \leftarrow}}{[\mathbf{W}_\tau^k]^T \mathbf{J}} \quad \text{and} \quad \mathbf{W}_\tau^{k+1} \leftarrow \mathbf{W}_\tau^k \odot \frac{\frac{\mathbf{V}}{\mathbf{\Lambda}} \begin{bmatrix} \mathbf{S} \\ \mathbf{S} \end{bmatrix}^{\tau \rightarrow k+1}}{\mathbf{J} \begin{bmatrix} \mathbf{S} \\ \mathbf{S} \end{bmatrix}^{\tau \rightarrow k+1}} \quad \forall \tau \in \{0, \dots, T-1\}, \quad (2.62)$$

where \mathbf{J} is the matrix with ones (has the same dimensions with matrix \mathbf{V}) and $\mathbf{\Lambda}$ is updated after any update of \mathbf{S} or \mathbf{W}_τ .

Squared Euclidean Distance

In case that the squared Euclidean distance is chosen as distance function, solving the Convolutional NMF problem can be expressed as the optimization problem

$$\begin{aligned} \text{minimize} \quad & f_F(\mathbf{W}_0, \mathbf{W}_1, \dots, \mathbf{W}_{T-1}, \mathbf{S}) := \frac{1}{2} \left\| \mathbf{V} - \sum_{\tau=0}^{T-1} \mathbf{W}_\tau^{\tau \rightarrow} \mathbf{S} \right\|_F^2 \\ \text{subject to} \quad & \mathbf{W}_\tau \geq \mathbf{0}, \quad \tau = 0, \dots, T-1, \\ & \mathbf{S} \geq \mathbf{0}, \end{aligned} \quad (2.63)$$

Then, problem (2.63) can be solved using the multiplicative update rules. The update rules for this case will be the same as when performing NMF for each iteration of τ , plus some shifting to appropriately line up the arguments, i.e. in the $k+1^{th}$ iteration

$$\mathbf{S}^{k+1} \leftarrow \mathbf{S}^k \odot \frac{\mathbf{W}_\tau^{kT} \mathbf{V}}{\mathbf{W}_\tau^{kT} \mathbf{\Lambda}} \quad \text{and} \quad \mathbf{W}_\tau^{k+1} \leftarrow \mathbf{W}_\tau^k \odot \frac{\mathbf{V} \mathbf{S}^{\tau \rightarrow k+1T}}{\mathbf{\Lambda} \mathbf{S}^{\tau \rightarrow k+1T}} \quad \forall \tau \in \{0, \dots, T-1\}, \quad (2.64)$$

where $\mathbf{\Lambda} = \sum_{\tau=0}^{T-1} \mathbf{W}_\tau^{\tau \rightarrow} \mathbf{S}$ and it is updated after any update of \mathbf{S} or \mathbf{W}_τ .

Practical Improvements

In both cases of distance function selection, there is the danger that \mathbf{S} has been more influenced by the last \mathbf{W}_τ used for its update, rather than the entire ensemble of \mathbf{W}_τ 's. Therefore, it is often useful to obtain different updates for matrix \mathbf{S} over τ 's (\mathbf{S}^τ) and then

set as the final update, the average of \mathbf{S}^τ 's, i.e. in the $k + 1^{th}$ iteration

$$\mathbf{S}^{k+1} \leftarrow \frac{1}{T} \sum_{\tau=0}^{T-1} \mathbf{S}^{\tau^{k+1}}. \quad (2.65)$$

Note here that operation in equation (2.65) is an intuitive operation which is obtained empirically, rather than justified theoretically.

From above, it is clear that the updates of \mathbf{S} and \mathbf{W}_τ 's rely on the most recent update of $\mathbf{\Lambda}$. This means that $\mathbf{\Lambda}$ should be updated correspondingly once for each \mathbf{W}_τ in each iteration. Nevertheless, updating $\mathbf{\Lambda}$ is computationally demanding if only an individual \mathbf{W}_τ has new values. Therefore, instead of direct calculation of $\mathbf{\Lambda}$, the following simpler formulation is used (supposing we are in the $k + 1^{th}$ iteration

$$\mathbf{\Lambda}_{new} \leftarrow \mathbf{\Lambda}_{old} - \mathbf{W}_\tau^{k\tau\rightarrow} \mathbf{S} + \mathbf{W}_\tau^{k+1\tau\rightarrow} \mathbf{S}. \quad (2.66)$$

In practice, the nonnegativity of $\mathbf{\Lambda}_{new}$ can not be guaranteed, due to subtraction operation and small numerical errors. The negative values can be treated as small positive constants (i.e $\epsilon = 10^{-6}$).

Remarks

In general, the Convolutional NMF model is not unique, since it can be shown that rotation and scaling ambiguities there are exists.

2.5 Constrained NMF

For some tasks, it may be advantageous to perform NMF with additional constraints placed on either \mathbf{W} or \mathbf{S} in order to resolve the rotation ambiguity. Next, we study three constrained NMF models, the NMF with sparsity constraints, the NMF model orthogonal constraints, and the NMF model with smoothness constraints.

2.5.1 NMF with Sparsity Constraints

One increasingly popular and powerful constraint is that the rows of \mathbf{S} have a parsimonious activation pattern for each basis contained in the columns of \mathbf{W} . These constraints are called ‘‘Sparsity Constraints’’. A signal is said to be sparse when it is zero or nearly zero more than might be expected from its variance. The advantage of a sparse signal

representation is that the probability of two or more activation patterns being active simultaneously is low. Thus, sparse representations lend themselves to good separability. The addition of the sparseness constraint on \mathbf{S} provides a means of trading-off the sparsity of the representation against accurate reconstruction.

The most widely used method in optimization with sparsity constraints is the weighted sum method [15]. This method creates an aggregate objective function by multiplying each constituent cost function by a weighting factor and summing the weighted costs. Combining our reconstruction cost function f for a NMF problem with sparsity constraints on \mathbf{S} matrix results in the following optimization problem

$$\begin{aligned} & \text{minimize} && f_{SP}(\mathbf{W}, \mathbf{S}) = f(\mathbf{W}, \mathbf{S}) + \lambda \|\mathbf{S}\|_1. \\ & \text{subject to} && \mathbf{W} \geq \mathbf{0}, \\ & && \mathbf{S} \geq \mathbf{0}. \end{aligned} \tag{2.67}$$

The left term of the objective function corresponds to the NMF problem, while the right term is an additional constraint on \mathbf{S} that enforces sparsity by minimising the l_1 norm of its columns. The parameter λ controls the trade-off between sparsity and accurate reconstruction.

This objective function creates a new problem: the right term is a strictly increasing function of the absolute value of its argument, so it is possible that the objective can be decreased by scaling up \mathbf{W} and scaling down \mathbf{S} ($\mathbf{W} \rightarrow \alpha \mathbf{W}$ and $\mathbf{S} \rightarrow \frac{1}{\alpha} \mathbf{S}$, with $\alpha > 0$). This situation does not alter the left term in the objective function, but will cause the right term to decrease, resulting in the elements of \mathbf{W} growing without bound and \mathbf{S} tending toward zero. Consequently, the solution arrived at by the optimization algorithm is not influenced by the right term of the objective function and the resultant \mathbf{S} matrix is not sparse. Therefore another constraint needs to be introduced in order to make the cost function well-defined. This is achieved by fixing the norm of the i^{th} column of \mathbf{W} to unity which constrains the scale of the elements in \mathbf{W} and \mathbf{S} .

Notice here that function f was not specified in problem (2.67) on purpose. The reason is that this problem formulation is general, regardless selection of f and NMF model extension.

2.5.2 Uni-Orthogonal NMF

In many applications we need to impose additional orthogonality constraints which automatically provide very sparse representation of the estimated factor matrices. Now,

we consider the optimization problem in relation (2.7), with additional orthogonality constraints on matrix \mathbf{W} , i.e.

$$\begin{aligned} & \underset{\mathbf{W}, \mathbf{S}}{\text{minimize}} && f_F(\mathbf{W}, \mathbf{S}) = \|\mathbf{V} - \mathbf{WS}\|_F^2 \\ & \text{subject to} && \mathbf{W}^T \mathbf{W} = \mathbf{I}, \\ & && \mathbf{W} \geq \mathbf{0}, \\ & && \mathbf{S} \geq \mathbf{0}, \end{aligned} \tag{2.68}$$

The following analysis of these problems is based on [16]. The main advantages are (1) uniqueness of the solution and (2) clustering interpretations. We will show it is equivalent to K-means clustering.

Uniqueness of Uni-Orthogonal NMF

As we saw in the introduction of this chapter, NMF model is not unique, since the rotation and scaling ambiguities exist. However, when we impose additional orthogonality constraints on matrix \mathbf{W} , it can be shown that this degree of freedom is eliminated.

Proposition 1. [16] With the orthogonality constraint $\mathbf{W}^T \mathbf{W} = \mathbf{I}$ and the nonnegativity constraint $\mathbf{W} \geq \mathbf{0}$, there exist no matrices (\mathbf{A}, \mathbf{B}) that satisfies $\mathbf{AB} = \mathbf{I}$, $\mathbf{WA} \geq \mathbf{0}$, and the orthogonality constraint $(\mathbf{WA})^T (\mathbf{WA}) = \mathbf{I}$, except when \mathbf{A} and \mathbf{B} are permutation matrices, i.e. $\mathbf{A} = \mathbf{P}$, $\mathbf{B} = \mathbf{P}^T$, $\mathbf{PP}^T = \mathbf{P}^T \mathbf{P} = \mathbf{I}$, $\mathbf{P}_{ij} = 0$ or 1.

Proof. [16] $(\mathbf{WA})^T (\mathbf{WA}) = \mathbf{I}$ implies that $\mathbf{A}^T \mathbf{A} = \mathbf{I}$. Except $\mathbf{A} = \mathbf{I}$ or permutation matrix, at least one off-diagonal element of \mathbf{A} must be negative. Also, because of orthogonality, each row of \mathbf{W} has exactly one nonzero element. Suppose that, for the l^{th} row of \mathbf{W} , the k_l^{th} element is nonzero. Then, $[\mathbf{WA}]_{l,:} = \sum_k \mathbf{W}_{l,k} \mathbf{A}_{k,:} = \mathbf{W}_{l,k_l} \mathbf{A}_{k_l,:}$. Since we wish $[\mathbf{WA}]_{l,:} \geq \mathbf{0} \forall l$, there can be no negative elements in \mathbf{A} . ■

One can notice that there are not any assumptions about nonnegativity for the matrix \mathbf{S} in the above proof. Thus, the Uni-Orthogonal Semi-NMF model, i.e.

$$\mathbf{V} = \mathbf{WS}, \tag{2.69}$$

where $\mathbf{V} \in \mathbb{R}^{N \times M}$, $\mathbf{W} \in \mathbb{R}_+^{N \times D}$ with $\mathbf{W}^T \mathbf{W} = \mathbf{I}$, and $\mathbf{S} \in \mathbb{R}^{D \times M}$, when $D \leq \min\{N, M\}$, is also unique.

Uniqueness of the above models states that there is an unique optimal solution for each model (Uni-Orthogonal NMF & Uni-Orthogonal Semi-NMF). However, calculating the unique optimal solution, for these models, may be difficult because of the existence of local minima.

Uni-Orthogonal NMF and Clustering

Next, we prove the equivalence of the Uni-Orthogonal NMF model to the K-means clustering based on [16].

Theorem 1. [16] Uni-Orthogonal NMF problem

$$\begin{aligned} & \underset{\mathbf{W}, \mathbf{S}}{\text{minimize}} && f_F(\mathbf{W}, \mathbf{S}) = \frac{1}{2} \|\mathbf{V} - \mathbf{WS}\|_F^2 \\ & \text{subject to} && \mathbf{W}^T \mathbf{W} = \mathbf{I}, \\ & && \mathbf{W} \geq \mathbf{0}, \\ & && \mathbf{S} \geq \mathbf{0}, \end{aligned} \tag{2.70}$$

is equivalent to K-means clustering.

Proof. [16] We write

$$\begin{aligned} J &= \|\mathbf{V} - \mathbf{WS}\|_F^2 \\ &= \text{Tr}(\mathbf{V}^T \mathbf{V} - 2\mathbf{V}^T \mathbf{WS} + \mathbf{S}^T \mathbf{W}^T \mathbf{WS}) \\ &\stackrel{\mathbf{W}^T \mathbf{W} = \mathbf{I}}{=} \text{Tr}(\mathbf{V}^T \mathbf{V} - 2\mathbf{WSV}^T + \mathbf{S}^T \mathbf{S}). \end{aligned} \tag{2.71}$$

The zero gradient condition $\frac{\partial \mathcal{J}}{\partial \mathbf{S}} = \frac{\partial f_F}{\partial \mathbf{S}} = \mathbf{W}^T \mathbf{V} - \mathbf{S} = \mathbf{0}$, when $\mathbf{W}^T \mathbf{W} = \mathbf{I}$, gives $\mathbf{S} = \mathbf{W}^T \mathbf{V}$. Thus, for $\mathbf{S} = \mathbf{W}^T \mathbf{V}$,

$$\begin{aligned} J &= \text{Tr}(\mathbf{V}^T \mathbf{V} - 2\mathbf{WW}^T \mathbf{VV}^T + (\mathbf{W}^T \mathbf{V})^T \mathbf{W}^T \mathbf{V}) \\ &= \text{Tr}(\mathbf{V}^T \mathbf{V} - 2\mathbf{WW}^T \mathbf{VV}^T + \mathbf{V}^T \mathbf{WW}^T \mathbf{V}) \\ &= \text{Tr}(\mathbf{V}^T \mathbf{V} - \mathbf{W}^T \mathbf{VV}^T \mathbf{W}). \end{aligned} \tag{2.72}$$

Since $\text{Tr}(\mathbf{V}^T \mathbf{V})$ is a constant, the optimization problem in relation (2.71) becomes

$$\begin{aligned} & \underset{\mathbf{W}}{\text{maximize}} && \text{Tr}(\mathbf{W}^T \mathbf{V} \mathbf{V}^T \mathbf{W}) \\ & \text{subject to} && \mathbf{W}^T \mathbf{W} = \mathbf{I}, \\ & && \mathbf{W} \geq \mathbf{0}. \end{aligned} \quad (2.73)$$

According to Theorem 2 below, this is identical to K-means clustering. ■

One can see that Theorem 1 holds even if \mathbf{V} and \mathbf{S} are not nonnegative, i.e., \mathbf{V} and \mathbf{S} have mixed-sign elements.

Theorem 2. [17, 18] The K-means clustering minimizes

$$J = \sum_{k=1}^K \sum_{i \in C_k} \|\mathbf{s}_i - \mathbf{c}_k\|_2^2 = \sum_{k=1}^K \sum_{i=1}^N \mathbf{G}_{i,k} \|\mathbf{s}_i - \mathbf{c}_k\|_2^2, \quad (2.74)$$

where N is the number of points we want to group in K clusters, \mathbf{c}_k denotes the cluster centroid of the k^{th} cluster, and \mathbf{G} is a cluster indicator matrix: $\mathbf{G}_{i,k} = 1$ if $\mathbf{s}_i \in C_k$ and $\mathbf{G}_{i,k} = 0$ if $\mathbf{s}_i \notin C_k$. More generally, the Kernel K-means with mapping $\mathbf{s}_i \rightarrow \phi(\mathbf{s}_i)$ minimizes

$$J_\phi = \sum_{k=1}^K \sum_{i \in C_k} \|\phi(\mathbf{s}_i) - \bar{\phi}_k\|_2^2 = \sum_{k=1}^K \sum_{i=1}^N \mathbf{G}_{i,k} \|\phi(\mathbf{s}_i) - \bar{\phi}_k\|_2^2, \quad (2.75)$$

where N is the number of points we want to group in K clusters, $\bar{\phi}_k$ denotes the cluster centroid of the k^{th} cluster in the feature space, and \mathbf{G} is a cluster indicator matrix: $\mathbf{G}_{i,k} = 1$ if $\phi(\mathbf{s}_i) \in C_k$ and $\mathbf{G}_{i,k} = 0$ if $\phi(\mathbf{s}_i) \notin C_k$. Let the normalized $\mathbf{G}' = \mathbf{G}(\mathbf{G}^T \mathbf{G})^{-1/2}$. Both clusterings can be solved via the optimization problem

$$\begin{aligned} & \underset{\mathbf{G}'}{\text{maximize}} && \text{Tr}(\mathbf{G}'^T \mathbf{H} \mathbf{G}') \\ & \text{subject to} && \mathbf{G}'^T \mathbf{G}' = \mathbf{I}, \\ & && \mathbf{G}' \geq \mathbf{0}, \end{aligned} \quad (2.76)$$

where $\mathbf{H}_{i,j} = \phi(\mathbf{s}_i)^T \phi(\mathbf{s}_j)$ is the kernel. For K-means, $\phi(\mathbf{s}_i) = \mathbf{s}_i$ and $\mathbf{H}_{i,j} = \mathbf{s}_i^T \mathbf{s}_j$. ■

2.5.3 NMF with Temporal Smoothness Constrains

In some applications prior knowledge about source signals properties allow us to suppose that source signals are correlated over time, i.e. source signals change over time smoothly. Next, we present how temporal smoothness constraints can be applied in an NMF problem, based on [19].

Suppose that each factorized temporal source (each row vector of the matrix \mathbf{S}) represents a temporal signal with length T . Then, we write the i^{th} source as $s_i(t) \equiv \mathbf{s}_{i,t}$ and the i^{th} row vector of \mathbf{S} as \mathbf{s}_i . When $\mathbf{s}_i(t)$ is temporally locally smooth, its short-term variance is relatively small compared to a larger long-term variance. Let $\mathbf{m}_i = \frac{1}{T} \sum_{t=0}^{T-1} \mathbf{s}_i(t)$ denote the mean value of $\{\mathbf{s}_i(t)\}$ given T observations, and let $\bar{\mathbf{s}}_i$ denote the short-term exponentially weighted average of temporal signal $\mathbf{s}_i(t)$, namely

$$\bar{\mathbf{s}}_i(t) = \alpha \bar{\mathbf{s}}_i(t-1) + (1-\alpha) \mathbf{s}_i(t) \equiv \alpha \bar{\mathbf{s}}_i(t-1) + \beta \mathbf{s}_i(t), \quad (2.77)$$

where $0 < \alpha < 1$ is a forgetting factor that determines the local smoothness range, and $\beta = 1 - \alpha$. In particular, the temporal smoothness under consideration is measured by the ratio of short-term variance against long-term (i.e. the complete data) variance in the temporal domain

$$R = \log \frac{\sum_{t=1}^T (\mathbf{s}_i(t) - \bar{\mathbf{s}}_i(t))^2}{\sum_{t=1}^T (\mathbf{s}_i(t) - \mathbf{m}_i)^2} = \log \frac{\sum_{t=1}^T \alpha^2 (\mathbf{s}_i(t) - \bar{\mathbf{s}}_i(t-1))^2}{\sum_{t=1}^T (\mathbf{s}_i(t) - \mathbf{m}_i)^2}. \quad (2.78)$$

Hence, the smaller the ratio value R , the smoother is the temporal signal $\bar{\mathbf{s}}_i(t)$. In vector notation, let $\bar{\mathbf{s}}_i$ denote the short-term average vector corresponding to the row vector $\mathbf{s}_i = [\mathbf{s}_{i,1}, \dots, \mathbf{s}_{i,T}]$. If we further constrain the variance of the row vector \mathbf{s}_i to 1, then the optimization problem may be equivalently rewritten as

$$R = \frac{1}{T} \|\mathbf{s}_i - \bar{\mathbf{s}}_i\|^2 = \frac{1}{T} (\mathbf{s}_i - \bar{\mathbf{s}}_i) (\mathbf{s}_i - \bar{\mathbf{s}}_i)^T, \quad s.t. \quad \text{var}[\mathbf{s}_i] = 1. \quad (2.79)$$

Now, we wish to represent $\bar{\mathbf{s}}_i$ in terms of \mathbf{s}_i . Note that the exponentially weighted average $\bar{\mathbf{s}}_i(t)$ is indeed the convolution product between $\mathbf{s}_i(t)$ and a template operator. Suppose $\bar{\mathbf{s}}_i(0) = 0$ and the template vector has an exponentially decreasing property with length L (here we use $L = 5$) namely, template $= [\beta, \alpha\beta, \alpha^2\beta, \alpha^3\beta, \alpha^4\beta]$ (if $\alpha = 0.5$, then $\text{sum}(\text{template}) = 0.9688$). The convolution operation can also be conveniently expressed as a matrix product operation

$$\bar{\mathbf{s}}_i^T = \mathbf{T} \mathbf{s}_i, \quad (2.80)$$

where

$$\mathbf{T} = \begin{bmatrix} \beta & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ \alpha\beta & \beta & 0 & 0 & 0 & 0 & 0 & \dots & 0 \\ \alpha^2\beta & \alpha\beta & \beta & 0 & 0 & 0 & 0 & \dots & 0 \\ \alpha^3\beta & \alpha^2\beta & \alpha\beta & \beta & 0 & 0 & 0 & \dots & 0 \\ \alpha^4\beta & \alpha^3\beta & \alpha^2\beta & \alpha\beta & \beta & 0 & 0 & \dots & 0 \\ 0 & \alpha^4\beta & \alpha^3\beta & \alpha^2\beta & \alpha\beta & \beta & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & 0 & \alpha^4\beta & \alpha^3\beta & \alpha^2\beta & \alpha\beta & \beta \end{bmatrix}, \quad (2.81)$$

is a $T \times T$ Toeplitz matrix with the right-shifted template appearing at each row. R can be written as

$$R = \frac{1}{T} \|\mathbf{s}_i^T - \mathbf{T}\mathbf{s}_i^T\|^2 = \frac{1}{T} \|(\mathbf{I} - \mathbf{T}) \mathbf{s}_i^T\|^2. \quad (2.82)$$

Combining our reconstruction cost function f for a NMF problem, with temporal smoothness constraints on matrix \mathbf{S} results in the following optimization problem

$$\begin{aligned} & \text{minimize} && f_{SM}(\mathbf{W}, \mathbf{S}) = f(\mathbf{W}, \mathbf{S}) + \frac{\lambda}{T} \|(\mathbf{I} - \mathbf{T}) \mathbf{S}^T\|_F^2 \\ & \text{subject to} && \mathbf{W} \geq \mathbf{0}, \\ & && \mathbf{S} \geq \mathbf{0}. \end{aligned} \quad (2.83)$$

Notice here that function f was not specified in relation (2.84) on purpose. The reason is that this problem formulation is general, regardless selection of f and NMF model extension.

Chapter 3

Tensor Factorization Models

3.1 Introduction

In many scientific areas, such as signal processing, neuroimaging, chemometrics, and others, data appear as multidimensional arrays. In this chapter, we study how multidimensional arrays (or tensors) can be processed in order to reveal their hidden structure or patterns across different dimensions. The term *way* is used to express each dimension of data. In case of data in vector form, we have *one-way* data, in case of data in matrix form, we have *two-way* data, e.t.c. In this thesis we focus on *three-way* tensors, but the models we discuss can be extended to higher-*way* tensors. An important advantage that comes with multi-*way* factorization models is that the rotation ambiguity we mentioned in the previous chapter does not appear.

The two most popular factorization models for *N-way* tensors are the Tucker model and the more restricted PARAFAC model. In this thesis, we focus on the PARAFAC model. The reason for this choice is that the PARAFAC model comes with theoretical background that guarantees essentially unique factorizations of multi-linear data when some mild conditions are satisfied. Also, complicated multidimensional patterns can be interpreted easily with PARAFAC.

Two-way factorization methods, as NMF, can be applied to multi-way data processing, after reshaping data in a two-way form. However, in this case, data-internal relations become corrupted, as the multidimensional structure is flattened. Another way to apply a two-way factorization model is averaging along trials. This choice makes sense only under the assumption that noise in data is uncorrelated, otherwise information would be lost in the addition of correlated noise. However, two-way factorization methods, in general, suffer from the rotation ambiguity.

In this chapter we introduce the PARAFAC model and we show how it can be used in BSS problems, when a set of realizations of the same BSS problem is available. As in the previous chapter, we assume that the mixing process is linear and the mixing matrix is time invariant. Additionally, we assume that the source signals and the mixing matrix are realization invariant. At last, we begin assuming that there are no propagation delays

between sources and sensors and in the sequel, we show how propagation delays can be incorporated into the PARAFAC model, when the propagation environment is modelled as an anechoic one.

3.2 Definitions

Definition 3.2.1 Let $\mathbf{a} \in \mathbb{R}^N$, $\mathbf{b} \in \mathbb{R}^P$, and $\mathbf{c} \in \mathbb{R}^J$. The **outer product** of \mathbf{a} and \mathbf{b} is defined as the rank-one matrix with elements

$$[\mathbf{a} \circ \mathbf{b}]_{n,p} = \mathbf{a}_n \mathbf{b}_p, \quad (3.1)$$

for all $n \in \{1, \dots, N\}$, $p \in \{1, \dots, P\}$, and the outer product of \mathbf{a} , \mathbf{b} and \mathbf{c} is defined as the rank-one tensor with elements

$$[\mathbf{a} \circ \mathbf{b} \circ \mathbf{c}]_{n,p,j} = \mathbf{a}_n \mathbf{b}_p \mathbf{c}_j, \quad (3.2)$$

for all $n \in \{1, \dots, N\}$, $p \in \{1, \dots, P\}$, and $j \in \{1, \dots, J\}$.

Definition 3.2.2 Let $\mathbf{A} \in \mathbb{R}^{N \times M}$ and $\mathbf{B} \in \mathbb{R}^{P \times K}$. The **Kronecker product** (or tensor product) of \mathbf{A} and \mathbf{B} is defined as the matrix

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} \mathbf{A}_{1,1}\mathbf{B} & \cdots & \mathbf{A}_{1,M}\mathbf{B} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{N,1}\mathbf{B} & \cdots & \mathbf{A}_{N,M}\mathbf{B} \end{bmatrix} \in \mathbb{R}^{NP \times MK}. \quad (3.3)$$

Definition 3.2.3 Let $\mathbf{A} \in \mathbb{R}^{N \times M}$ and $\mathbf{B} \in \mathbb{R}^{P \times M}$. The **Khatri-Rao product** of \mathbf{A} and \mathbf{B} is defined as the matrix

$$\mathbf{A} \circledast \mathbf{B} = \begin{bmatrix} \mathbf{A}_{:,1} \otimes \mathbf{B}_{:,1} & \mathbf{A}_{:,2} \otimes \mathbf{B}_{:,2} & \cdots & \mathbf{A}_{:,M-1} \otimes \mathbf{B}_{:,M-1} & \mathbf{A}_{:,M} \otimes \mathbf{B}_{:,M} \end{bmatrix}. \quad (3.4)$$

Definition 3.2.4 Let $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$, $\mathbf{A} \in \mathbb{R}^{I \times F}$, $\mathbf{B} \in \mathbb{R}^{J \times F}$, and $\mathbf{C} \in \mathbb{R}^{K \times F}$. We define the rank of \mathcal{X} as the minimum integer positive number F such that

$$\mathcal{X} = \sum_{f=1}^F \mathbf{A}_{:,f} \circ \mathbf{B}_{:,f} \circ \mathbf{C}_{:,f}. \quad (3.5)$$

3.3 PARAFAC and Nonnegative Tensor Factorization models

Consider a set of K realizations, $\mathbf{V} \in \mathbb{R}^{K \times N \times M}$, of a BSS problem. Then, the PARAFAC or CP decomposition of \mathbf{V} , when $D \leq \min(KN, NM, MK)$, is

$$\mathbf{v}_{k,n,m} = \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} \mathbf{S}_{m,d}, \quad (3.6)$$

for all $k \in \{1, \dots, K\}$, $n \in \{1, \dots, N\}$, and $m \in \{1, \dots, M\}$, where N is the number of sensors, D is the number of sources, $\mathbf{A} \in \mathbb{R}^{K \times D}$ expresses the realization variability, $\mathbf{W} \in \mathbb{R}^{N \times D}$ expresses the mixing matrix, and $\mathbf{S} \in \mathbb{R}^{M \times D}$ expresses the source signals over M time instances. Relation (3.6) can be also expressed as

$$\mathbf{V} = [\![\mathbf{A}, \mathbf{W}, \mathbf{S}]\!] = \sum_{d=1}^D \mathbf{A}_{:,d} \circ \mathbf{W}_{:,d} \circ \mathbf{S}_{:,d}. \quad (3.7)$$

As can be seen, using the PARAFAC model in BSS problems, we assume that the mixing matrix \mathbf{W} and the source matrix \mathbf{S} are invariant in each realization. Then, $\mathbf{A}_{k,d}$ expresses the contribution of rank one matrix $\mathbf{W}_{:,d} \circ \mathbf{S}_{:,d}$ in the k^{th} realization. Physical interpretation of matrix \mathbf{A} is relevant to framework that realizations happen. For instance, in biomedical BSS applications, each realization may correspond to different subjects or different trials.

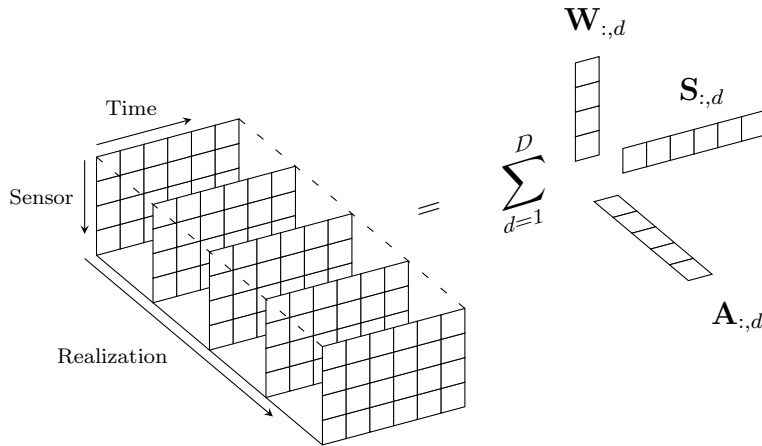


Figure 3.1: An example illustration of the PARAFAC model in the BSS framework.

The nonnegative factorization of a nonnegative tensor $\mathbf{V} \in \mathbb{R}_+^{K \times N \times M}$ can be obtained using the PARAFAC model with nonnegative constraints (NTF), i.e.

$$\mathbf{V} = \llbracket \mathbf{A}, \mathbf{W}, \mathbf{S} \rrbracket = \sum_{d=1}^D \mathbf{A}_{:,d} \circ \mathbf{W}_{:,d} \circ \mathbf{S}_{:,d}, \quad (3.8)$$

where now $\mathbf{A} \in \mathbb{R}_+^{K \times D}$, $\mathbf{W} \in \mathbb{R}_+^{N \times D}$ and $\mathbf{S} \in \mathbb{R}_+^{M \times D}$.

Degeneracy and Uniqueness

The problem of finding a best rank- D approximation for tensors of order 3 (three-way), in the unconstrained case has no solution, in general. There exists $\mathcal{A} \in \mathbb{R}^{d_1 \times d_2 \times d_3}$ such that

$$\inf \left\| \mathcal{A} - \sum_{d=1}^D \mathbf{U}_{:,d}^{(1)} \circ \mathbf{U}_{:,d}^{(2)} \circ \mathbf{U}_{:,d}^{(3)} \right\| \quad (3.9)$$

is not attained by any choice of matrices $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$, $\mathbf{U}^{(3)}$. It is also, in general, not possible to determine a priori if a given $\mathcal{A} \in \mathbb{R}^{d_1 \times d_2 \times d_3}$ will fail to have a best rank- D approximation [20]. Moreover, such failures can occur with positive probability and in some cases with certainty, i.e. where the infimum in (3.9) is never attained. This phenomenon is called PARAFAC degeneracy. Roughly speaking, this refers to solutions in which some component loadings are highly correlated in all modes and the elements of these components become arbitrarily large [21]. PARAFAC degeneracy makes the estimation unstable, the algorithm slow to converge (or even diverge), and the result difficult to interpret, largely because the model is plagued by strong inter-component cancellations.

The PARAFAC model has solution, which is unique (up to scaling and permutation), when [22]

$$\sum_{r=1}^F k_{\mathbf{U}^{(r)}} \geq 2D' + (F - 1), \quad (3.10)$$

where F is the number of *ways* (3 in our case), D' is the rank of the tensor and $k_{\mathbf{X}}$ is the Kruskal rank, denoting the largest number of columns of matrix \mathbf{X} that is guaranteed to be linearly independent. Thus, $k_{\mathbf{X}} \leq \text{rank}(\mathbf{X})$. When one models the data using a low rank approximation ($D < D'$), the above criterion guarantees that the residuals are uniquely defined [23], which leads to essential uniqueness. Essential uniqueness means that if $\mathcal{A} = \llbracket \mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \mathbf{U}^{(3)} \rrbracket$ then $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$, and $\mathbf{U}^{(3)}$, are unique up to a common permutation and scaling/counter-scaling of columns, i.e. there exists a permutation matrix

Π and diagonal scaling matrices Λ_1 , Λ_2 , and Λ_3 such that

$$\begin{aligned}\hat{\mathbf{U}}^{(1)} &= \mathbf{U}^{(1)}\Pi\Lambda_1, & \hat{\mathbf{U}}^{(2)} &= \mathbf{U}^{(2)}\Pi\Lambda_2, & \hat{\mathbf{U}}^{(3)} &= \mathbf{U}^{(3)}\Pi\Lambda_3, \\ \Lambda_1\Lambda_2\Lambda_3 &= \mathbf{I}, & \mathcal{A} &= \llbracket \hat{\mathbf{U}}^{(1)}, \hat{\mathbf{U}}^{(2)}, \hat{\mathbf{U}}^{(3)} \rrbracket.\end{aligned}$$

The condition (3.10) is sufficient but not necessary for essential uniqueness. This condition does not hold when $D' = 1$. It is also necessary for $D' = 2$ and $D' = 3$ but not for $D' > 3$ [24, 22]. In the presence of noise, if tensor \mathcal{A} for a rank- D does not belong to the degeneration class, all matrices $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$, and $\mathbf{U}^{(3)}$ will have full rank and uniqueness is guaranteed by proofs given in [25] and [26].

In the case of nonnegative tensor factorization model (NTF), for any nonnegative tensor $\mathcal{A} \in \mathbb{R}_+^{d_1 \times d_2 \times d_3}$ and any given $D \in \mathbb{N}$, a best nonnegative rank- D approximation always exists (up to scaling and permutation) in the sense that the infimum in relation (3.9) is attained by some nonnegative tensors $\sum_{d=1}^D \mathbf{U}_{:,d}^{(1)} \circ \mathbf{U}_{:,d}^{(2)} \circ \mathbf{U}_{:,d}^{(3)}$ [20].

All the above conclusions can be extended, without loss of generality, to higher order tensors (multi-way tensors) with the appropriate extension of all above relations.

PARAFAC Factors Estimation

Let a tensor $\mathcal{V}^o \in \mathbb{R}^{K \times N \times M}$ admit the PARAFAC factorization form

$$\mathcal{V}^o = \llbracket \mathbf{A}^o, \mathbf{W}^o, \mathbf{S}^o \rrbracket = \sum_{d=1}^D \mathbf{A}_{:,d}^o \circ \mathbf{W}_{:,d}^o \circ \mathbf{S}_{:,d}^o, \quad (3.11)$$

where matrices $\mathbf{A}^o \in \mathbb{R}^{K \times D}$, $\mathbf{W}^o \in \mathbb{R}^{N \times D}$, and $\mathbf{S}^o \in \mathbb{R}^{M \times D}$. We observe the noisy tensor $\mathcal{V} = \mathcal{V}^o + \mathcal{E}$, where \mathcal{E} is the additive noise. Estimates of \mathbf{A}^o , \mathbf{W}^o , and \mathbf{S}^o can be obtained by computing matrices $\mathbf{A} \in \mathbb{R}^{K \times D}$, $\mathbf{W} \in \mathbb{R}^{N \times D}$, $\mathbf{S} \in \mathbb{R}^{M \times D}$ that solve the optimization problem

$$\underset{\mathbf{A}, \mathbf{W}, \mathbf{S}}{\text{minimize}} \quad f_{\mathcal{V}}(\mathbf{A}, \mathbf{W}, \mathbf{S}), \quad (3.12)$$

where $f_{\mathcal{V}}$ is a function measuring the quality of the factorization. A common choice for $f_{\mathcal{V}}$ is

$$f_{\mathcal{V}}(\mathbf{A}, \mathbf{W}, \mathbf{S}) = \frac{1}{2} \|\mathcal{V} - \llbracket \mathbf{A}, \mathbf{W}, \mathbf{S} \rrbracket\|_F^2. \quad (3.13)$$

If $\mathcal{V} = \llbracket \mathbf{A}, \mathbf{W}, \mathbf{S} \rrbracket$, then its matrix unfoldings, with respect to the first, second, and third dimension, are given by [3]

$$\mathbf{V}_{\mathbf{A}} = \mathbf{A}(\mathbf{S} \circledast \mathbf{W})^T, \quad \mathbf{V}_{\mathbf{W}} = \mathbf{W}(\mathbf{S} \circledast \mathbf{A})^T, \quad \mathbf{V}_{\mathbf{S}} = \mathbf{S}(\mathbf{W} \circledast \mathbf{A})^T. \quad (3.14)$$

Thus, $f_{\mathbf{V}}$ can be expressed as

$$\begin{aligned} f_{\mathbf{V}}(\mathbf{A}, \mathbf{W}, \mathbf{S}) &= \frac{1}{2} \left\| \mathbf{V}_{\mathbf{A}} - \mathbf{A}(\mathbf{S} \circledast \mathbf{W})^T \right\|_F^2 \\ &= \frac{1}{2} \left\| \mathbf{V}_{\mathbf{W}} - \mathbf{W}(\mathbf{S} \circledast \mathbf{A})^T \right\|_F^2 \\ &= \frac{1}{2} \left\| \mathbf{V}_{\mathbf{S}} - \mathbf{S}(\mathbf{W} \circledast \mathbf{A})^T \right\|_F^2. \end{aligned} \quad (3.15)$$

These expressions form the basis for the AO tensor factorization in the sense that, if we fix two matrix factors, then we can update the third by solving a least-squares problem of the forms

$$\underset{\mathbf{A}}{\text{minimize}} \quad \frac{1}{2} \left\| \mathbf{V}_{\mathbf{A}} - \mathbf{A}(\mathbf{S} \circledast \mathbf{W})^T \right\|_F^2, \quad (3.16)$$

$$\underset{\mathbf{W}}{\text{minimize}} \quad \frac{1}{2} \left\| \mathbf{V}_{\mathbf{W}} - \mathbf{W}(\mathbf{S} \circledast \mathbf{A})^T \right\|_F^2, \quad (3.17)$$

$$\underset{\mathbf{S}}{\text{minimize}} \quad \frac{1}{2} \left\| \mathbf{V}_{\mathbf{S}} - \mathbf{S}(\mathbf{W} \circledast \mathbf{A})^T \right\|_F^2, \quad (3.18)$$

respectively.

For the case of NTF, the nonnegative factors can be estimated by solving the constrained optimization problem

$$\underset{\mathbf{A} \geq 0, \mathbf{W} \geq 0, \mathbf{S} \geq 0}{\text{minimize}} \quad f_{\mathbf{V}}(\mathbf{A}, \mathbf{W}, \mathbf{S}). \quad (3.19)$$

Based on the unfolding equations (3.14) and the expressions of $f_{\mathbf{V}}$ in (3.15), an AO procedure can be employed, in which a sequence of nonnegative least-squares (NNLS) problems

$$\underset{\mathbf{A} \geq 0}{\text{minimize}} \quad \frac{1}{2} \left\| \mathbf{V}_{\mathbf{A}} - \mathbf{A}(\mathbf{S} \circledast \mathbf{W})^T \right\|_F^2, \quad (3.20)$$

$$\underset{\mathbf{W} \geq 0}{\text{minimize}} \quad \frac{1}{2} \left\| \mathbf{V}_{\mathbf{W}} - \mathbf{W}(\mathbf{S} \circledast \mathbf{A})^T \right\|_F^2, \quad (3.21)$$

$$\underset{\mathbf{S} \geq 0}{\text{minimize}} \quad \frac{1}{2} \left\| \mathbf{V}_{\mathbf{S}} - \mathbf{S}(\mathbf{W} \circledast \mathbf{A})^T \right\|_F^2, \quad (3.22)$$

will be solved.

3.3.1 PARAFAC with Orthogonality Constraints

As we discussed in the previous section, the PARAFAC model has always an optimal solution if the parameter matrices are constrained to have nonnegative elements [20]. In [27], Theorem 2 states that if one of the parameter matrices $\mathbf{A}, \mathbf{W}, \mathbf{S}$ is constrained to be

column-wise orthonormal, then the PARAFAC model has also always an optimal solution. Also, the orthogonal constrained PARAFAC model can be unique under more relaxed conditions than unconstrained PARAFAC model [24, 28].

Let us consider the PARAFAC model for a tensor $\mathbf{V} \in \mathbb{R}^{K \times N \times M}$,

$$\mathbf{V} = [\mathbf{A}, \mathbf{W}, \mathbf{S}] = \sum_{d=1}^D \mathbf{A}_{:,d} \circ \mathbf{W}_{:,d} \circ \mathbf{S}_{:,d}, \quad (3.23)$$

where matrices $\mathbf{A} \in \mathbb{R}^{K \times D}$, $\mathbf{W} \in \mathbb{R}^{N \times D}$ with $\mathbf{W}^T \mathbf{W} = \mathbf{I}_D$, and $\mathbf{S} \in \mathbb{R}^{M \times D}$. Then, estimates of the parameter matrices \mathbf{A} , \mathbf{W} and \mathbf{S} can be obtained by solving the constrained optimization problem

$$\begin{aligned} & \underset{\mathbf{A}, \mathbf{W}, \mathbf{S}}{\text{minimize}} && f_{\mathbf{V}}(\mathbf{A}, \mathbf{W}, \mathbf{S}) \\ & \text{subject to} && \mathbf{W}^T \mathbf{W} = \mathbf{I}_D. \end{aligned} \quad (3.24)$$

Based on the unfolding equations (3.14) and the expressions of $f_{\mathbf{V}}$ in (3.15), an AO procedure can be employed for solving a sequence of least-squares problems

$$\underset{\mathbf{A}}{\text{minimize}} \quad \frac{1}{2} \left\| \mathbf{V}_{\mathbf{A}} - \mathbf{A} (\mathbf{S} \circledast \mathbf{W})^T \right\|_F^2, \quad (3.25)$$

$$\begin{aligned} & \underset{\mathbf{W}}{\text{minimize}} \quad \frac{1}{2} \left\| \mathbf{V}_{\mathbf{W}} - \mathbf{W} (\mathbf{S} \circledast \mathbf{A})^T \right\|_F^2 \\ & \text{subject to} \quad \mathbf{W}^T \mathbf{W} = \mathbf{I}_D, \end{aligned} \quad (3.26)$$

$$\underset{\mathbf{S}}{\text{minimize}} \quad \frac{1}{2} \left\| \mathbf{V}_{\mathbf{S}} - \mathbf{S} (\mathbf{W} \circledast \mathbf{A})^T \right\|_F^2, \quad (3.27)$$

respectively. The constrained least-squares problem in (3.26) is known as the orthogonal Procrustes problem and it has a closed-form solution as we show in the sequel.

The Orthogonal Procrustes problem

Let $\mathbf{Y} \in \mathbb{R}^{N \times M}$, $\mathbf{A} \in \mathbb{R}^{N \times D}$ and $\mathbf{X} \in \mathbb{R}^{D \times M}$. We consider the optimization problem

$$\begin{aligned} & \underset{\mathbf{A}}{\text{minimize}} \quad f_{\text{OP}}(\mathbf{A}) := \|\mathbf{Y} - \mathbf{A}\mathbf{X}\|_F^2 \\ & \text{subject to} \quad \mathbf{A}^T \mathbf{A} = \mathbf{I}_D. \end{aligned} \quad (3.28)$$

Then, the optimal solution $\hat{\mathbf{A}}$ for this problem is given by setting

$$\hat{\mathbf{A}} = \mathbf{U}\mathbf{V}^T, \quad (3.29)$$

where matrices $\mathbf{U} \in \mathbb{R}^{N \times D}$ and $\mathbf{V} \in \mathbb{R}^{D \times D}$ are given by the singular value decomposition of matrix $\mathbf{M} = \mathbf{YX}^T = \mathbf{U}\Sigma\mathbf{V}^T$.

Proof :

$$\begin{aligned}
\hat{\mathbf{A}} &= \underset{\mathbf{A}}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{AX}\|_F^2 \\
&= \underset{\mathbf{A}}{\operatorname{argmin}} \operatorname{Tr} \left((\mathbf{Y} - \mathbf{AX})^T (\mathbf{Y} - \mathbf{AX}) \right) \\
&\stackrel{\mathbf{A}^T \mathbf{A} = \mathbf{I}}{=} \underset{\mathbf{A}}{\operatorname{argmin}} \|\mathbf{X}\|_F^2 + \|\mathbf{Y}\|_F^2 - 2\operatorname{Tr}(\mathbf{Y}^T \mathbf{AX}) \\
&= \underset{\mathbf{A}}{\operatorname{argmax}} \operatorname{Tr}(\mathbf{Y}^T \mathbf{AX}) \\
&= \underset{\mathbf{A}}{\operatorname{argmax}} \operatorname{Tr}(\mathbf{XY}^T \mathbf{A}) \\
&\stackrel{\mathbf{YX}^T = \mathbf{U}\Sigma\mathbf{V}^T}{=} \underset{\mathbf{A}}{\operatorname{argmax}} \operatorname{Tr}(\mathbf{V}\Sigma\mathbf{U}^T \mathbf{A}) \\
&= \underset{\mathbf{A}}{\operatorname{argmax}} \operatorname{Tr}(\Sigma\mathbf{U}^T \mathbf{A}\mathbf{V}) \\
&\stackrel{\mathbf{A} = \mathbf{U}\mathbf{L}\mathbf{V}^T}{=} \mathbf{U} \left(\underset{\mathbf{L}}{\operatorname{argmax}} \operatorname{Tr}(\Sigma\mathbf{L}) \right) \mathbf{V}^T \\
&= \mathbf{U} \left(\underset{\mathbf{L}}{\operatorname{argmax}} \sum_{i=1}^D \Sigma_{i,i} \mathbf{L}_{i,i} \right) \mathbf{V}^T
\end{aligned} \tag{3.30}$$

Since we require $\mathbf{A}^T \mathbf{A} = \mathbf{I}_D$, this implies that $\mathbf{L}^T \mathbf{L} = \mathbf{I}_D \Rightarrow \sum_{k=1}^D \mathbf{L}_{k,i}^2 = 1 \ \forall i \in \{1, \dots, D\} \Rightarrow \mathbf{L}_{i,i} \leq 1 \ \forall i \in \{1, \dots, D\}$. Also, we know that $\Sigma_{i,i} \geq 0 \ \forall i \in \{1, \dots, D\}$. Thus, quantity

$$\sum_{i=1}^D \Sigma_{i,i} \mathbf{L}_{i,i} \tag{3.31}$$

can achieve maximum for $\mathbf{L} = \mathbf{I}_D$ and we conclude that the optimal solution $\hat{\mathbf{A}} = \mathbf{U}\mathbf{V}^T$.

3.4 Shift Invariant Multilinear Decomposition Model

Degeneration of the PARAFAC (CP) model could indicate inappropriateness of the model for the data [29]. Data generated from the PARAFAC model with shifted factors are no longer multilinear and therefore the PARAFAC model is no longer a valid model for the data. Extending the PARAFAC model to incorporate delays forms the Shifted CP (SCP) model. The SCP model was proposed in [30] and an algorithm devised based on exhaustive integer searches over all possible shifts. This is however very expensive, making the estimation infeasible when including many shifts. In [23], it was proposed to solve the model in the frequency domain rather than the time-domain. The attractive property being that each integer delay has a closed form solution while keeping the remaining delays fixed, given by calculating cross-correlations which is inexpensive in the frequency domain. Furthermore, in a frequency representation, non-integer delays can be estimated using gradient-based searches. The SCP model is

$$\mathbf{V}_{k,n,m} = \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} \mathbf{S}_{m-\tau_{n,d},d}, \quad (3.32)$$

for all $k \in \{1, \dots, K\}$, $n \in \{1, \dots, N\}$, and $m \in \{1, \dots, M\}$, where N is the number of sensors, D is the number of sources, $\mathbf{A} \in \mathbb{R}^{K \times D}$ expresses the realization variability, $\mathbf{W} \in \mathbb{R}^{N \times D}$ expresses the mixing matrix, and $\mathbf{S} \in \mathbb{R}^{M \times D}$ expresses the source signals over M time instances. Here, each source signal $\mathbf{S}_{:,d}$ is shifted according to $\tau_{n,d}$ time-samples, depending on the index n of the second mode. Hence, the shifts will be along the index m .

Uniqueness

The rigorous proof of uniqueness by Kruskal using Kruskal rank given in equation (3.10) is involved. However, the uniqueness assuming that \mathbf{A} , \mathbf{W} , and \mathbf{S} , have full rank can be proven by considering the CP model in a slab representation [25], [26]. For the n^{th} slab, the CP model reads

$$\begin{aligned} \mathbf{V}_{:,n,:} &= \mathbf{A} \text{diag}(\mathbf{W}_{n,:}) \mathbf{S} \\ &= \mathbf{A} \mathbf{\Pi} [\mathbf{\Pi}^{-1} \text{diag}(\mathbf{W}_{n,:}) \mathbf{\Lambda}] \mathbf{\Lambda}^{-1} \mathbf{S} \\ &= \hat{\mathbf{A}} \text{diag}(\hat{\mathbf{W}}_{n,:}) \hat{\mathbf{S}}. \end{aligned} \quad (3.33)$$

Thus, if two solutions \mathbf{A} , \mathbf{W} , \mathbf{S} and $\hat{\mathbf{A}}$, $\hat{\mathbf{W}}$, $\hat{\mathbf{S}}$ exist, there must be a mapping from one solution to the other given by $\mathbf{\Pi}$ and $\mathbf{\Lambda}$. However, for this mapping the term $\mathbf{\Pi}^{-1} \text{diag}(\mathbf{W}_{n,:}) \mathbf{\Lambda}$

has to be diagonal for all k 's which, when \mathbf{A} , \mathbf{W} , \mathbf{S} have full rank, restricts $\mathbf{\Pi}$ and $\mathbf{\Lambda}$ to be simple scaling and permutation matrices [25], [26].

For the SCP model, we instead have

$$\begin{aligned}\mathbf{V}_{:,n,:} &= \mathbf{A} \text{diag}(\mathbf{W}_{n,:}) \mathbf{S}^n \\ &= \mathbf{A} \mathbf{\Pi} [\mathbf{\Pi}^{-1} \text{diag}(\mathbf{W}_{n,:}) \mathbf{\Lambda}] \mathbf{\Lambda}^{-1} \mathbf{S}^n \\ &= \hat{\mathbf{A}} \text{diag}(\hat{\mathbf{W}}_{n,:}) \hat{\mathbf{S}}^n,\end{aligned}\tag{3.34}$$

where $\mathbf{S}_{m',d}^n = \mathbf{S}_{m-\tau_{n,d},d}$. Although the CP model is extended such that \mathbf{S} is shifted, still has to remain diagonal for all values of n . This again strongly restricts $\mathbf{\Pi}$ and $\mathbf{\Lambda}$. The obvious ambiguities are scaling, permutation, relative shift, and onset as well as period of the time-series [23].

SCP Factors Estimation

In order to analyse the SCP model, the transformation of the model in the frequency domain is useful. We assume that the source signals are periodic with period equal to the number of temporal samples (M). Then, based on the relations that define the transformations between time and frequency domains, we consider the transformation of the SCP model in the frequency domain, as the tube-wise DFT of tensor \mathbf{V} , that is

$$\begin{aligned}\tilde{\mathbf{V}}_{k,n,f} &= \frac{1}{\sqrt{M}} \sum_{m=1}^M \mathbf{V}_{k,n,m} e^{-j2\pi \frac{(f-1)(m-1)}{M}} \quad \forall k \in \{1, \dots, K\}, n \in \{1, \dots, N\}, f \in \{1, \dots, M\} \\ &= \frac{1}{\sqrt{M}} \sum_{m=1}^M \left(\sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} \mathbf{S}_{m-\tau_{n,d},d} \right) e^{-j2\pi \frac{(f-1)(m-1)}{M}} \\ &= \frac{1}{\sqrt{M}} \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} \sum_{m=1}^M \mathbf{S}_{m-\tau_{n,d},d} e^{-j2\pi \frac{(f-1)(m-1)}{M}} \\ &\stackrel{m'=m-\tau_{n,d}}{=} \frac{1}{\sqrt{M}} \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} \sum_{m'=1-\tau_{n,d}}^{M-\tau_{n,d}} \mathbf{S}_{m',d} e^{-j2\pi \frac{(f-1)(m'+\tau_{n,d}-1)}{M}} \\ &= \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} \left(\frac{1}{\sqrt{M}} \sum_{m'=1}^M \mathbf{S}_{m',d} e^{-j2\pi \frac{(f-1)(m'-1)}{M}} \right) e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \quad (\text{due to circular shift}) \\ &= \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}.\end{aligned}\tag{3.35}$$

In matrix notation, the SCP model in the frequency domain can be expressed, for all $f \in \{1, \dots, M\}$, as

$$\tilde{\mathbf{V}}_{:,f} = \sum_{d=1}^D \mathbf{A}_{:,d} \left[\mathbf{W}_{:,d} \odot e^{-j2\pi \frac{(f-1)\tau_{:,d}}{M}} \right]^T \tilde{\mathbf{S}}_{f,d} = \mathbf{A} \text{diag} \left(\tilde{\mathbf{S}}_{f,:} \right) \left[\mathbf{W} \odot e^{-j2\pi \frac{(f-1)\tau}{M}} \right]^T. \quad (3.36)$$

Let a tensor $\mathbf{V}^o \in \mathbb{R}^{K \times N \times M}$ admit the SCP factorization form

$$\mathbf{V}_{k,n,m}^o = \sum_{d=1}^D \mathbf{A}_{k,d}^o \mathbf{W}_{n,d}^o \mathbf{S}_{m-\tau_{n,d},d}^o, \quad (3.37)$$

for all $k \in \{1, \dots, K\}$, $n \in \{1, \dots, N\}$, and $m \in \{1, \dots, M\}$, where matrices $\mathbf{A}^o \in \mathbb{R}^{K \times D}$, $\mathbf{W}^o \in \mathbb{R}^{N \times D}$, $\boldsymbol{\tau}^o \in \mathbb{R}^{N \times D}$ and $\mathbf{S}^o \in \mathbb{R}^{M \times D}$. We observe the noisy tensor $\mathbf{V} = \mathbf{V}^o + \boldsymbol{\mathcal{E}}$, where $\boldsymbol{\mathcal{E}}$ is the additive noise. Estimates of \mathbf{A}^o , \mathbf{W}^o , $\boldsymbol{\tau}^o$ and \mathbf{S}^o can be obtained by computing matrices $\mathbf{A} \in \mathbb{R}^{K \times D}$, $\mathbf{W} \in \mathbb{R}^{N \times D}$, $\boldsymbol{\tau} \in \mathbb{R}^{N \times D}$, and $\mathbf{S} \in \mathbb{R}^{M \times D}$ that solve the optimization problem

$$\underset{\mathbf{A}, \mathbf{W}, \mathbf{S}, \boldsymbol{\tau}}{\text{minimize}} \quad f_{\mathbf{V}}(\mathbf{A}, \mathbf{W}, \mathbf{S}, \boldsymbol{\tau}), \quad (3.38)$$

where $f_{\mathbf{V}}$ is a function measuring the quality of the factorization. A convenient choice for $f_{\mathbf{V}}$ is the least-squares function

$$f_{\mathbf{V}}(\mathbf{A}, \mathbf{W}, \mathbf{S}, \boldsymbol{\tau}) := \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N \sum_{m=1}^M \left| \mathbf{V}_{k,n,m} - \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} \mathbf{S}_{m-\tau_{n,d},d} \right|^2, \quad (3.39)$$

since we want to operate both in time and frequency domains. Parseval's Theorem offers a one-to-one correspondence between the least-squares errors in the time and the frequency domain, so that the minimization can be performed arbitrarily between the two domains. Parseval's Theorem states that

$$\begin{aligned} & \sum_{k=1}^K \sum_{n=1}^N \sum_{m=1}^M \left| \mathbf{V}_{k,n,m} - \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} \mathbf{S}_{m-\tau_{n,d},d} \right|^2 \\ &= \sum_{k=1}^K \sum_{n=1}^N \sum_{f=1}^M \left| \mathbf{V}_{k,n,m} - \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right|^2. \end{aligned} \quad (3.40)$$

The RHS of equation (3.40) is equal to

$$\begin{aligned}
& \sum_{f=1}^M \left\| \tilde{\mathbf{v}}_{:, :, f} - \mathbf{A} \text{diag} \left(\tilde{\mathbf{S}}_{f, :} \right) \left[\mathbf{W} \odot e^{-j2\pi \frac{(f-1)\tau}{M}} \right]^T \right\|_F^2 \\
&= \sum_{f=1}^M \left\| \text{vec} \left(\tilde{\mathbf{v}}_{:, :, f} \right) - \text{vec} \left(\mathbf{A} \text{diag} \left(\tilde{\mathbf{S}}_{f, :} \right) \left[\mathbf{W} \odot e^{-j2\pi \frac{(f-1)\tau}{M}} \right]^T \right) \right\|_2^2 \\
&= \sum_{f=1}^M \left\| \text{vec} \left(\tilde{\mathbf{v}}_{:, :, f} \right) - \left(\left[\mathbf{W} \odot e^{-j2\pi \frac{(f-1)\tau}{M}} \right] \circledast \mathbf{A} \right) \tilde{\mathbf{S}}_{f, :}^T \right\|_2^2.
\end{aligned} \tag{3.41}$$

Using Lemma 2.3.3, the symmetry of DFT matrix \mathbf{Q} , and (3.41), (3.40) can be rewritten as

$$\begin{aligned}
& \sum_{k=1}^K \sum_{n=1}^N \sum_{m=1}^M \left| \mathbf{v}_{k,n,m} - \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} \mathbf{S}_{m-\tau_{n,d},d} \right|^2 \\
&= \sum_{f=1}^M \left\| \text{vec} \left(\tilde{\mathbf{v}}_{:, :, f} \right) - \left(\left[\mathbf{W} \odot e^{-j2\pi \frac{(f-1)\tau}{M}} \right] \circledast \mathbf{A} \right) \mathbf{S}^T \mathbf{Q}_{:, f} \right\|_2^2.
\end{aligned} \tag{3.42}$$

Hence,

$$\begin{aligned}
f_{\mathbf{v}}(\mathbf{A}, \mathbf{W}, \mathbf{S}, \tau) &= \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N \sum_{m=1}^M \left| \mathbf{v}_{k,n,m} - \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} \mathbf{S}_{m-\tau_{n,d},d} \right|^2 \\
&= \frac{1}{2} \sum_{f=1}^M \left\| \text{vec} \left(\tilde{\mathbf{v}}_{:, :, f} \right) - \left(\left[\mathbf{W} \odot e^{-j2\pi \frac{(f-1)\tau}{M}} \right] \circledast \mathbf{A} \right) \mathbf{S}^T \mathbf{Q}_{:, f} \right\|_2^2 \\
&= \frac{1}{2} \sum_{f=1}^M \left\| \text{vec} \left(\tilde{\mathbf{v}}_{:, :, f} \right) - \left(\left[\mathbf{W} \odot e^{-j2\pi \frac{(f-1)\tau}{M}} \right] \circledast \mathbf{A} \right) \tilde{\mathbf{S}}_{f, :}^T \right\|_2^2 \\
&=: f_{\tilde{\mathbf{v}}}(\mathbf{A}, \mathbf{W}, \tilde{\mathbf{S}}, \tau).
\end{aligned} \tag{3.43}$$

Therefore, solving the SCP model can be expressed as an optimization problem with the following equivalent forms

$$\underset{\mathbf{A}, \mathbf{W}, \mathbf{S}, \tau}{\text{minimize}} \quad f_{\mathbf{v}}(\mathbf{A}, \mathbf{W}, \mathbf{S}, \tau) \equiv \underset{\mathbf{A}, \mathbf{W}, \tilde{\mathbf{S}}, \tau}{\text{minimize}} \quad f_{\tilde{\mathbf{v}}}(\mathbf{A}, \mathbf{W}, \tilde{\mathbf{S}}, \tau), \tag{3.44}$$

where $\tilde{\mathbf{S}} = \mathbf{Q}\mathbf{S}$.

The procedure of solving the above optimization problems given a tensor \mathbf{V} and D , as presented in [23], is based on alternatingly updating matrices \mathbf{S} , $\boldsymbol{\tau}$, \mathbf{A} , and \mathbf{W} . In each step, all matrices except one are fixed and the free matrix is tuned in order to minimize the cost functions $f_{\mathbf{V}}$ and $f_{\tilde{\mathbf{V}}}$. The parameters in the SCP model are updated in the sequence \mathbf{S} , $\boldsymbol{\tau}$, \mathbf{A} , \mathbf{W} . Next, we present the update steps for each matrix separately for the unconstrained case and for the case that nonnegative constraints are applied.

S Update

For the unconstrained case, an estimate of matrix \mathbf{S} can be obtained in the frequency domain. Specifically, for all $f \in \{1, \dots, M\}$, we have

$$\frac{\partial f_{\tilde{\mathbf{V}}}(\mathbf{A}, \mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau})}{\partial \tilde{\mathbf{S}}_{f,:}} = \left[\tilde{\mathbf{S}}_{f,:} (\mathbf{W}^f \otimes \mathbf{A})^T - \text{vec}(\tilde{\mathbf{V}}_{:,:,f})^T \right] (\mathbf{W}^f \otimes \mathbf{A})^*. \quad (3.45)$$

By setting $\frac{\partial f_{\tilde{\mathbf{V}}}(\mathbf{A}, \mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau})}{\partial \tilde{\mathbf{S}}_{f,:}}$ equal to the zero vector, we get

$$\begin{aligned} \tilde{\mathbf{S}}_{f,:} &= \text{vec}(\tilde{\mathbf{V}}_{:,:,f})^T (\mathbf{W}^f \otimes \mathbf{A})^* \left[(\mathbf{W}^f \otimes \mathbf{A})^T (\mathbf{W}^f \otimes \mathbf{A})^* \right]^{-1} \\ &= \text{vec}(\tilde{\mathbf{V}}_{:,:,f})^T (\mathbf{W}^f \otimes \mathbf{A})^* \left[(\mathbf{W}^{fT} \mathbf{W}^{f*}) \odot (\mathbf{A}^T \mathbf{A}) \right]^{-1}, \end{aligned} \quad (3.46)$$

for all $f \in \{1, \dots, M\}$. Matrix \mathbf{S} is real-valued if the following relation is satisfied in the frequency domain

$$\tilde{\mathbf{S}}_{M-f+1,d} = \tilde{\mathbf{S}}_{f,d}^*, \quad (3.47)$$

that is, $\tilde{\mathbf{S}}$ is conjugate symmetric. This constraint is enforced by updating the first $\lfloor M/2 \rfloor + 1$ elements, i.e. up to the Nyquist frequency, while setting the remaining elements according to relation (3.31). Then, the estimate of matrix $\tilde{\mathbf{S}}$ is obtained and the estimate of matrix \mathbf{S} is given by using the column-wise IDFT (Lemma 2.3)

$$\mathbf{S} = \mathbf{Q}^H \tilde{\mathbf{S}}. \quad (3.48)$$

For the case of nonnegative constraints, consider the least-squares cost function in the frequency domain as defined in relation (3.27)

$$\begin{aligned}
 f_{\mathbf{V}}(\mathbf{A}, \mathbf{W}, \mathbf{S}, \boldsymbol{\tau}) &= \frac{1}{2} \sum_{f=1}^M \left\| \text{vec}(\tilde{\mathbf{V}}_{:, :, f}) - \left(\underbrace{\left[\mathbf{W} \odot e^{-j2\pi \frac{(f-1)\boldsymbol{\tau}}{M}} \right]}_{\mathbf{W}^f} \circledast \mathbf{A} \right) \mathbf{S}^T \mathbf{Q}_{:, f} \right\|_2^2 \\
 &= \frac{1}{2} \sum_{f=1}^M \left\| \text{vec}(\tilde{\mathbf{V}}_{:, :, f}) - (\mathbf{W}^f \circledast \mathbf{A}) \mathbf{S}^T \mathbf{Q}_{:, f} \right\|_2^2.
 \end{aligned} \tag{3.49}$$

Then, the derivative of $f_{\mathbf{V}}$, with respect to \mathbf{S} , is equal to

$$\frac{\partial f_{\mathbf{V}}(\mathbf{A}, \mathbf{W}, \mathbf{S}, \boldsymbol{\tau})}{\partial \mathbf{S}} = \Re \left\{ \sum_{f=1}^M \mathbf{Q}_{:, f}^* \left[\mathbf{Q}_{f, :} \mathbf{S} (\mathbf{W}^f \circledast \mathbf{A})^T - \text{vec}(\tilde{\mathbf{V}}_{:, :, f})^T \right] (\mathbf{W}^f \circledast \mathbf{A})^* \right\}. \tag{3.50}$$

Since $\mathbf{Q}_{:, f}^* = \mathbf{Q}^* \mathbf{e}_f$, where \mathbf{e}_f is the f^{th} column of the M -dimensional identity matrix, and the matrix \mathbf{Q} is symmetric, the derivative of f_F with respect to \mathbf{S} is equal to

$$\begin{aligned}
 \frac{\partial f_{\mathbf{V}}(\mathbf{A}, \mathbf{W}, \mathbf{S}, \boldsymbol{\tau})}{\partial \mathbf{S}} &= \Re \left\{ \sum_{f=1}^M \mathbf{Q}^* \mathbf{e}_f \left[\mathbf{Q}_{f, :} \mathbf{S} (\mathbf{W}^f \circledast \mathbf{A})^T - \text{vec}(\tilde{\mathbf{V}}_{:, :, f})^T \right] (\mathbf{W}^f \circledast \mathbf{A})^* \right\} \\
 &= \Re \left\{ \mathbf{Q}^H \sum_{f=1}^M \mathbf{e}_f \left[\mathbf{Q}_{f, :} \mathbf{S} (\mathbf{W}^f \circledast \mathbf{A})^T - \text{vec}(\tilde{\mathbf{V}}_{:, :, f})^T \right] (\mathbf{W}^f \circledast \mathbf{A})^* \right\}.
 \end{aligned} \tag{3.51}$$

Notice that

$$\begin{aligned}
 &\sum_{f=1}^M \mathbf{e}_f \left[\mathbf{Q}_{f, :} \mathbf{S} (\mathbf{W}^f \circledast \mathbf{A})^T - \text{vec}(\tilde{\mathbf{V}}_{:, :, f})^T \right] (\mathbf{W}^f \circledast \mathbf{A})^* \\
 &= \sum_{f=1}^M \mathbf{e}_f \left[\tilde{\mathbf{S}}_{f, :} (\mathbf{W}^f \circledast \mathbf{A})^T - \text{vec}(\tilde{\mathbf{V}}_{:, :, f})^T \right] (\mathbf{W}^f \circledast \mathbf{A})^* \\
 &= \sum_{f=1}^M \mathbf{e}_f \frac{\partial f_{\tilde{\mathbf{V}}}(\mathbf{A}, \mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau})}{\partial \tilde{\mathbf{S}}_{f, :}} \\
 &= \frac{\partial f_{\tilde{\mathbf{V}}}(\mathbf{A}, \mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau})}{\partial \tilde{\mathbf{S}}}.
 \end{aligned} \tag{3.52}$$

Hence,

$$\begin{aligned} \frac{\partial f_{\mathbf{v}}(\mathbf{A}, \mathbf{W}, \mathbf{S}, \tau)}{\partial \mathbf{S}} &= \Re \left\{ \mathbf{Q}^H \frac{\partial f_{\mathbf{v}}(\mathbf{A}, \mathbf{W}, \tilde{\mathbf{S}}, \tau)}{\partial \tilde{\mathbf{S}}} \right\} \\ &= \Re \left\{ IDFT \left[\frac{\partial f_{\mathbf{v}}(\mathbf{A}, \mathbf{W}, \tilde{\mathbf{S}}, \tau)}{\partial \tilde{\mathbf{S}}} \right] \right\}. \end{aligned} \quad (3.53)$$

Therefore, by taking the real part of the IDFT of the derivative in the frequency domain, the corresponding derivative in the time domain is obtained. Nonnegative estimate of matrix \mathbf{S} can be obtained using a projected gradient scheme or the multiplicative updates scheme when all factors are constrained to be nonnegative.

Multiplicative updates can be used by splitting the derivative in the frequency domain into what constitutes the positive and the negative part of the corresponding gradient in the time domain. This gives

$$\begin{aligned} \tilde{\mathbf{G}}^+ &= \sum_{f=1}^M \mathbf{e}_f \tilde{\mathbf{S}}_{f,:} (\mathbf{W}^f \circledast \mathbf{A})^T (\mathbf{W}^f \circledast \mathbf{A})^* \\ &= \sum_{f=1}^M \left\{ \mathbf{e}_f \tilde{\mathbf{S}}_{f,:} (\mathbf{W}^{fT} \mathbf{W}^{f*}) \right\} \odot (\mathbf{A}^T \mathbf{A}^*), \\ \tilde{\mathbf{G}}^- &= \sum_{f=1}^M \mathbf{e}_f \text{vec}(\tilde{\mathbf{v}}_{:,f})^T (\mathbf{W}^f \circledast \mathbf{A})^*, \\ \mathbf{G}^+ &= \Re \left\{ \mathbf{Q}^H \tilde{\mathbf{G}}^+ \right\}, \\ \mathbf{G}^- &= \Re \left\{ \mathbf{Q}^H \tilde{\mathbf{G}}^- \right\}. \end{aligned}$$

Consequently, by taking the real part of the IDFT of $\tilde{\mathbf{G}}^+$ and $\tilde{\mathbf{G}}^-$, the corresponding positive and negative part of the derivative of $f_{\mathbf{v}}$ with respect to \mathbf{S} can be calculated. As a result, matrix \mathbf{S} can be updated using the multiplicative update approach as

$$\mathbf{S} \leftarrow \mathbf{S} \odot \frac{\mathbf{G}^-}{\mathbf{G}^+}, \quad (3.54)$$

where \odot denotes the Hadamard product and $\frac{\mathbf{A}}{\mathbf{B}}$ denotes the Hadamard division.

A Update

The SCP model in the frequency domain can be written, for all $k \in \{1, \dots, K\}$, $n \in \{1, \dots, N\}$, and $f \in \{1, \dots, M\}$, as

$$\begin{aligned}
 \tilde{\mathbf{v}}_{k,n,f} &= \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} \underbrace{e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}}_{\tilde{\mathbf{S}}_{f,d}^n} \\
 &= \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} \tilde{\mathbf{S}}_{f,d}^n \\
 &= \mathbf{W}_{n,:} \text{diag}(\mathbf{A}_{k,:}) \tilde{\mathbf{S}}_{f,:}^{nT}.
 \end{aligned} \tag{3.55}$$

Then, for all $k \in \{1, \dots, K\}$, $n \in \{1, \dots, N\}$,

$$\begin{aligned}
 \tilde{\mathbf{v}}_{k,n,:} &= \mathbf{W}_{n,:} \text{diag}(\mathbf{A}_{k,:}) \tilde{\mathbf{S}}^{nT} \\
 \Leftrightarrow \tilde{\mathbf{v}}_{k,n,:} \mathbf{Q}^H &= \mathbf{W}_{n,:} \text{diag}(\mathbf{A}_{k,:}) \tilde{\mathbf{S}}^{nT} \mathbf{Q}^H \\
 \Leftrightarrow \mathbf{v}_{k,n,:} &= \mathbf{W}_{n,:} \text{diag}(\mathbf{A}_{k,:}) \mathbf{S}^{nT} \\
 \Leftrightarrow \text{vec}(\mathbf{v}_{k,n,:}) &= \text{vec}(\mathbf{W}_{n,:} \text{diag}(\mathbf{A}_{k,:}) \mathbf{S}^{nT}) \\
 &= (\mathbf{S}^n \circledast \mathbf{W}_{n,:}) \mathbf{A}_{k,:}^T.
 \end{aligned} \tag{3.56}$$

Since $\text{vec}(\mathbf{v}_{k,n,:}) = \mathbf{v}_{k,n,:}^T$, we obtain, for all $k \in \{1, \dots, K\}$, $n \in \{1, \dots, N\}$,

$$\begin{aligned}
 \mathbf{v}_{k,n,:} &= \mathbf{A}_{k,:} (\mathbf{S}^n \circledast \mathbf{W}_{n,:})^T \\
 \Leftrightarrow \mathbf{v}_{:,n,:} &= \mathbf{A} (\mathbf{S}^n \circledast \mathbf{W}_{n,:})^T.
 \end{aligned} \tag{3.57}$$

Then, the least squares cost function of SCP model in the time domain can be written as

$$f_{\mathbf{v}}(\mathbf{A}, \mathbf{W}, \mathbf{S}, \boldsymbol{\tau}) = \frac{1}{2} \sum_{n=1}^N \left\| \mathbf{v}_{:,n,:} - \mathbf{A} (\mathbf{S}^n \circledast \mathbf{W}_{n,:})^T \right\|_F^2. \tag{3.58}$$

The partial derivative of $f_{\mathbf{v}}$, with respect to matrix \mathbf{A} , is equal to

$$\frac{\partial f_{\mathbf{v}}(\mathbf{A}, \mathbf{W}, \mathbf{S}, \boldsymbol{\tau})}{\partial \mathbf{A}} = \sum_{n=1}^N \mathbf{A} (\mathbf{S}^n \circledast \mathbf{W}_{n,:})^T (\mathbf{S}^n \circledast \mathbf{W}_{n,:}) - \mathbf{v}_{:,n,:} (\mathbf{S}^n \circledast \mathbf{W}_{n,:}). \tag{3.59}$$

If there are no constraints on matrix \mathbf{A} , an estimate of matrix \mathbf{A} is given by setting $\frac{\partial f_{\mathbf{V}}(\mathbf{A}, \mathbf{W}, \mathbf{S}, \tau)}{\partial \mathbf{A}}$ equal to the zero matrix. Then,

$$\begin{aligned}
& \sum_{n=1}^N \mathbf{A} (\mathbf{S}^n \circledast \mathbf{W}_{n,:})^T (\mathbf{S}^n \circledast \mathbf{W}_{n,:}) = \sum_{n=1}^N \mathbf{V}_{:,n,:} (\mathbf{S}^n \circledast \mathbf{W}_{n,:}) \\
& \Leftrightarrow \mathbf{A} \sum_{n=1}^N (\mathbf{S}^n \circledast \mathbf{W}_{n,:})^T (\mathbf{S}^n \circledast \mathbf{W}_{n,:}) = \sum_{n=1}^N \mathbf{V}_{:,n,:} (\mathbf{S}^n \circledast \mathbf{W}_{n,:}) \\
& \Leftrightarrow \mathbf{A} \sum_{n=1}^N (\mathbf{S}^{nT} \mathbf{S}^n) \odot (\mathbf{W}_{n,:}^T \mathbf{W}_{n,:}) = \sum_{n=1}^N \mathbf{V}_{:,n,:} (\mathbf{S}^n \circledast \mathbf{W}_{n,:}) \\
& \Leftrightarrow \mathbf{A} = \sum_{n=1}^N \mathbf{V}_{:,n,:} (\mathbf{S}^n \circledast \mathbf{W}_{n,:}) \left[\sum_{n=1}^N (\mathbf{S}^{nT} \mathbf{S}^n) \odot (\mathbf{W}_{n,:}^T \mathbf{W}_{n,:}) \right]^{-1}.
\end{aligned} \tag{3.60}$$

For the case of nonnegative constraints, an estimate of matrix \mathbf{A} can be obtained using a projected gradient scheme or the multiplicative updates scheme when all factors are constrained to be nonnegative.

Multiplicative updates can be used by splitting the derivative in the frequency domain into what constitutes the positive and the negative part of the corresponding gradient in the time domain. This gives

$$\begin{aligned}
\mathbf{G}^+ &= \mathbf{A} \sum_{n=1}^N (\mathbf{S}^{nT} \mathbf{S}^n) \odot (\mathbf{W}_{n,:}^T \mathbf{W}_{n,:}), \\
\mathbf{G}^- &= \sum_{n=1}^N \mathbf{V}_{:,n,:} (\mathbf{S}^n \circledast \mathbf{W}_{n,:}).
\end{aligned}$$

As a result, matrix \mathbf{A} can be updated using the multiplicative update approach as

$$\mathbf{A} \leftarrow \mathbf{A} \odot \frac{\mathbf{G}^-}{\mathbf{G}^+}, \tag{3.61}$$

where \odot denotes the Hadamard product and $\frac{\mathbf{A}}{\mathbf{B}}$ denotes the Hadamard division.

W Update

The SCP model in the frequency domain, for all $k \in \{1, \dots, K\}$, $n \in \{1, \dots, N\}$, and $f \in \{1, \dots, M\}$, can be written as

$$\begin{aligned}
 \tilde{\mathbf{v}}_{k,n,f} &= \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} \underbrace{e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}}_{\tilde{\mathbf{S}}_{f,d}^n} \\
 &= \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} \tilde{\mathbf{S}}_{f,d}^n \\
 &= \mathbf{A}_{k,:} \text{diag}(\mathbf{W}_{n,:}) \tilde{\mathbf{S}}_{f,:}^n.
 \end{aligned} \tag{3.62}$$

Then, for each $n \in \{1, \dots, K\}$,

$$\begin{aligned}
 \tilde{\mathbf{v}}_{:,n,:} &= \mathbf{A} \text{diag}(\mathbf{W}_{n,:}) \tilde{\mathbf{S}}^n{}^T \\
 \Leftrightarrow \tilde{\mathbf{v}}_{:,n,:} \mathbf{Q}^H &= \mathbf{A} \text{diag}(\mathbf{W}_{n,:}) \tilde{\mathbf{S}}^n{}^T \mathbf{Q}^H \\
 \Leftrightarrow \mathbf{v}_{:,n,:} &= \mathbf{A} \text{diag}(\mathbf{W}_{n,:}) \mathbf{S}^n{}^T \\
 \Leftrightarrow \text{vec}(\mathbf{v}_{:,n,:}) &= \text{vec}(\mathbf{A} \text{diag}(\mathbf{W}_{n,:}) \mathbf{S}^n{}^T) = (\mathbf{S}^n \circledast \mathbf{A}) \mathbf{W}_{n,:}^T \\
 \Leftrightarrow \text{vec}(\mathbf{v}_{:,n,:})^T &= \mathbf{W}_{n,:} (\mathbf{S}^n \circledast \mathbf{A})^T,
 \end{aligned} \tag{3.63}$$

and the least squares cost function of SCP model in the time domain can be rewritten as

$$f_{\mathbf{v}}(\mathbf{A}, \mathbf{W}, \mathbf{S}, \boldsymbol{\tau}) = \frac{1}{2} \sum_{n=1}^N \left\| \text{vec}(\mathbf{v}_{:,n,:})^T - \mathbf{W}_{n,:} (\mathbf{S}^n \circledast \mathbf{A})^T \right\|_2^2. \tag{3.64}$$

The partial derivative of $f_{\mathbf{v}}$, with respect to each vector $\mathbf{W}_{n,:}$, for all $n \in \{1, \dots, N\}$, is equal to

$$\begin{aligned}
 \frac{\partial f_{\mathbf{v}}(\mathbf{A}, \mathbf{W}, \mathbf{S}, \boldsymbol{\tau})}{\partial \mathbf{W}_{n,:}} &= \mathbf{W}_{n,:} (\mathbf{S}^n \circledast \mathbf{A})^T (\mathbf{S}^n \circledast \mathbf{A}) - \text{vec}(\mathbf{v}_{:,n,:})^T (\mathbf{S}^n \circledast \mathbf{A}) \\
 &= \mathbf{W}_{n,:} \left[\left(\mathbf{S}^n{}^T \mathbf{S}^n \right) \odot (\mathbf{A}^T \mathbf{A}) \right] - \text{vec}(\mathbf{v}_{:,n,:})^T (\mathbf{S}^n \circledast \mathbf{A}).
 \end{aligned} \tag{3.65}$$

If there are no constraints on matrix \mathbf{W} , an estimate of matrix \mathbf{W} is given by setting $\frac{\partial f_{\mathbf{V}}(\mathbf{A}, \mathbf{W}, \mathbf{S}, \boldsymbol{\tau})}{\partial \mathbf{W}_{n,:}}$ equal to the zero vector, for all $n \in \{1, \dots, N\}$. Then,

$$\begin{aligned} \mathbf{W}_{n,:} \left(\mathbf{S}^{nT} \mathbf{S}^n \right) \odot (\mathbf{A}^T \mathbf{A}) &= \text{vec}(\mathbf{V}_{:,n,:})^T (\mathbf{S}^n \circledast \mathbf{A}) \\ \Leftrightarrow \mathbf{W}_{n,:} &= \text{vec}(\mathbf{V}_{:,n,:})^T (\mathbf{S}^n \circledast \mathbf{A}) \left[\left(\mathbf{S}^{nT} \mathbf{S}^n \right) \odot (\mathbf{A}^T \mathbf{A}) \right]^{-1}. \end{aligned} \quad (3.66)$$

For the case of nonnegative constraints, an estimate of matrix \mathbf{W} can be obtained using a projected gradient scheme or the multiplicative updates scheme when all factors are constrained to be nonnegative.

Multiplicative updates can be used by splitting the derivative in the frequency domain into what constitutes the positive and the negative part of the corresponding gradient in the time domain, which gives

$$\begin{aligned} \mathbf{G}^{n+} &= \mathbf{W}_{n,:} \left(\mathbf{S}^{nT} \mathbf{S}^n \right) \odot (\mathbf{A}^T \mathbf{A}), \\ \mathbf{G}^{n-} &= \text{vec}(\mathbf{V}_{:,n,:})^T (\mathbf{S}^n \circledast \mathbf{A}). \end{aligned}$$

As a result, matrix \mathbf{W} can be updated using the multiplicative update approach, for all $n \in \{1, \dots, N\}$, as

$$\mathbf{W}_{n,:} \leftarrow \mathbf{W}_{n,:} \odot \frac{\mathbf{G}^{n-}}{\mathbf{G}^{n+}}, \quad (3.67)$$

where \odot denotes the Hadamard product and $\frac{\mathbf{A}}{\mathbf{B}}$ denotes the Hadamard division.

$\boldsymbol{\tau}$ Update

The delays $\boldsymbol{\tau}$ are unconstrained. Consider the least squares cost function of SCP model in the frequency domain

$$f_{\mathbf{V}}(\mathbf{A}, \mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau}) = \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N \sum_{f=1}^M \left| \tilde{\mathbf{v}}_{k,n,f} - \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right|^2. \quad (3.68)$$

Define $\mathbf{T} \in \mathbb{R}^{ND \times 1} = \text{vec}(\boldsymbol{\tau})$, i.e. the vectorized form of matrix $\boldsymbol{\tau}$ such that $\mathbf{T}_{n+(d-1)N} = \tau_{n,d}$. Then, the partial derivative of $f_{\mathbf{V}}$, with respect to $\tau_{n,d}$, for all $n \in \{1, \dots, N\}$, $d \in \{1, \dots, D\}$, is

$$\begin{aligned}
\mathbf{g}_{n+(d-1)N} &= \left[\frac{\partial f_{\tilde{\mathbf{V}}}}{\partial \mathbf{T}} \right]_{n+(d-1)N} \\
&= \left[\frac{\partial f_{\tilde{\mathbf{V}}}}{\partial \boldsymbol{\tau}} \right]_{n,d} \\
&= 2\pi \frac{\mathbf{W}_{n,d}}{M} \sum_{f=1}^M (f-1) \Im \left\{ e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \left(\mathbf{A}_{:,d}^T \tilde{\boldsymbol{\epsilon}}_{:,n,f} \right) \right\},
\end{aligned} \tag{3.69}$$

where $\tilde{\boldsymbol{\epsilon}}_{k,n,f} = \tilde{\mathbf{V}}_{k,n,f} - \sum_{d'=1}^D \mathbf{A}_{k,d'} \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'}$. The Hessian of $f_{\tilde{\mathbf{V}}}$ has the following structure:

$$\mathbf{H}_{t,t'} = \begin{cases} 0, & \text{if } n' \neq n, \\ \mathbf{W}_{n,d} \mathbf{W}_{n,d'} \left(\mathbf{A}_{:,d'}^T \mathbf{A}_{:,d} \right) \left(\frac{2\pi}{M} \right)^2 \sum_{f=1}^M (f-1)^2 \Re \left\{ e^{j2\pi \frac{(f-1)(\tau_{n,d'} - \tau_{n,d})}{M}} \tilde{\mathbf{S}}_{f,d'}^* \tilde{\mathbf{S}}_{f,d} \right\}, & \text{if } d' \neq d \text{ \& } n' = n, \\ \mathbf{W}_{n,d} \left(\frac{2\pi}{M} \right)^2 \sum_{f=1}^M (f-1)^2 \left(\Re \left\{ e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \left(\mathbf{A}_{:,d}^T \tilde{\boldsymbol{\epsilon}}_{:,n,f} \right) \right\} + \mathbf{W}_{n,d} \left| \tilde{\mathbf{S}}_{f,d} \right|^2 \left\| \mathbf{A}_{:,d} \right\|^2 \right), & \text{if } d' = d \text{ \& } n' = n, \end{cases} \tag{3.70}$$

where $t := n + (d-1)N$ and $t' := n' + (d'-1)N$. As a result, $\boldsymbol{\tau}$ can be estimated by the Newton-Raphson method as

$$\mathbf{T} \leftarrow \mathbf{T} - \eta \mathbf{H}^{-1} \mathbf{g}, \tag{3.71}$$

where η is a step size parameter that is tuned to keep decreasing the cost function. Function $f_{\tilde{\mathbf{V}}}$, for fixed matrices \mathbf{A} , \mathbf{W} , and \mathbf{S} , is not convex. Thus, the descent property of the Newton-Raphson method is not guaranteed. In that case, we use the Newton-Raphson method when the Hessian is positive definite, otherwise we use the gradient descent method.

However, the above method is still sensitive to local minima. Estimating the delays by the following cross-correlation procedure reduces the effect of local minima [23]. Consider the SCP model in the frequency domain, for all $k \in \{1, \dots, K\}$, $n \in \{1, \dots, N\}$, and $f \in \{1, \dots, M\}$,

$$\tilde{\mathbf{V}}_{k,n,f} = \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \Leftrightarrow \tilde{\mathbf{V}}_{:,n,f} = \sum_{d=1}^D \mathbf{A}_{:,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}. \tag{3.72}$$

Let, for all $n \in \{1, \dots, N\}$ and $f \in \{1, \dots, M\}$,

$$\tilde{\mathbf{R}}_{:,n,f}^d = \tilde{\mathbf{V}}_{:,n,f} - \sum_{d'=1, d' \neq d}^D \mathbf{A}_{:,d'} \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'}, \quad (3.73)$$

which is the remaining signal in the frequency domain, over all realizations at the n^{th} sensor in a vectorized form, when subtracting all but the d^{th} source out of $\tilde{\mathbf{V}}_{:,n,f}$. Assuming that the true matrices \mathbf{A} , \mathbf{W} , \mathbf{S} , and $\boldsymbol{\tau}$ are available, relation (3.57) can be written as

$$\tilde{\mathbf{R}}_{:,n,f}^d = \mathbf{A}_{:,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}. \quad (3.74)$$

Next, let

$$\tilde{\mathbf{G}}_{n,f}^d = \mathbf{A}_{:,d}^T \tilde{\mathbf{R}}_{:,n,f}^d = \|\mathbf{A}_{:,d}\|_2^2 \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}. \quad (3.75)$$

Then, the cross-correlation terms become, for all $f \in \{1, \dots, M\}$,

$$\tilde{\mathbf{c}}_f^{n,d} = \tilde{\mathbf{G}}_{n,f}^d \tilde{\mathbf{S}}_{f,d}^* = \|\mathbf{A}_{:,d}\|_2^2 \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \tilde{\mathbf{S}}_{f,d}^* = \|\mathbf{A}_{:,d}\|_2^2 \mathbf{W}_{n,d} \left| \tilde{\mathbf{S}}_{f,d} \right|^2 e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}}, \quad (3.76)$$

and estimates of $\tau_{n,d}$ and $\mathbf{W}_{n,d}$ can be obtained, for all $n \in \{1, \dots, N\}$, $d \in \{1, \dots, D\}$, from

$$t = \underset{m}{\operatorname{argmax}} |\mathbf{c}_m^{n,d}|, \quad \tau_{n,d} = \operatorname{mod}(M + 1 - t, M), \quad \mathbf{W}_{n,d} = \sqrt{M} \frac{\mathbf{c}_t^{n,d}}{\|\mathbf{A}_{:,d}\|_2^2 \left\| \tilde{\mathbf{S}}_{:,d} \right\|_2^2}, \quad (3.77)$$

where

$$\begin{aligned} \mathbf{c}_m^{n,d} &= \Re \left\{ DFT \left[\tilde{\mathbf{c}}_f^{n,d} \right]_m \right\} \\ &= \Re \left\{ \frac{1}{\sqrt{M}} \sum_{f=1}^M \tilde{\mathbf{c}}_f^{n,d} e^{-j2\pi \frac{m(f-1)}{M}} \right\} \\ &= \Re \left\{ \frac{1}{\sqrt{M}} \sum_{f=1}^M \|\mathbf{A}_{:,d}\|_2^2 \mathbf{W}_{n,d} \left| \tilde{\mathbf{S}}_{f,d} \right|^2 e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} e^{-j2\pi \frac{m(f-1)}{M}} \right\} \\ &= \frac{1}{\sqrt{M}} \|\mathbf{A}_{:,d}\|_2^2 \mathbf{W}_{n,d} \sum_{f=1}^M \left| \tilde{\mathbf{S}}_{f,d} \right|^2 \Re \left\{ e^{j2\pi \frac{(f-1)(\tau_{n,d}-m)}{M}} \right\} \\ &\leq \frac{1}{\sqrt{M}} \|\mathbf{A}_{:,d}\|_2^2 \|\mathbf{W}_{n,d}\| \left\| \tilde{\mathbf{S}}_{:,d} \right\|_2^2. \end{aligned} \quad (3.78)$$

Quantity $\frac{1}{\sqrt{M}} \|\mathbf{A}_{:,d}\|_2^2 |\mathbf{W}_{n,d}| \sum_{f=1}^M \left| \tilde{\mathbf{S}}_{f,d} \right|^2 \Re \left\{ e^{j2\pi \frac{(f-1)(\boldsymbol{\tau}_{n,d}-m)}{M}} \right\}$ becomes maximum for $\boldsymbol{\tau}_{n,d} = m$. If \mathbf{W} is constrained positive, only positive values of $\mathbf{c}^{n,d}$ are considered. In our experiments, we apply the cross-correlation procedure every 20 iterations.

Chapter 4

Introduction to Functional Magnetic Resonance Imaging and Results

4.1 Introduction

Functional magnetic resonance imaging (fMRI) has, in less than two decades, become the most commonly used method for the study of human brain function. fMRI is a technique that uses magnetic resonance imaging to measure brain activity by measuring changes in the local oxygenation of blood, which in turn reflects the amount of local brain activity. The analysis of fMRI data is exceedingly complex, requiring the use of sophisticated techniques from signal and image processing and statistics in order to go from the raw data to the finished product, which is generally a statistical map showing which brain regions responded to a particular manipulation of mental or perceptual functions.

The most common method of fMRI takes advantage of the fact that, when neurons in the brain become active, the amount of blood flowing through that area is increased. This phenomenon has been known for more than 100 years, though the mechanisms that cause it remain only partly understood. What is particularly interesting is that the amount of blood that is sent to the area is more than is needed to replenish the oxygen that is used by the activity of the cells [31]. Thus, the activity related increase in blood flow caused by neuronal activity leads to a relative surplus in local blood oxygen. The signal measured in fMRI depends on this change in oxygenation and is referred to as the blood oxygenation level dependent (BOLD) signal.

4.2 MR Physics and BOLD Imaging

The magnetic resonance (MR) scanner uses superconducting electromagnets to produce a static, uniform magnetic field of high strength. The strength of the static magnetic field created by an MRI scanner is expressed in units of Tesla (one Tesla is equal to 10,000 Gauss). Ten years ago, the standard field strength used in fMRI research was 1.5 Tesla (T), whereas the standard today is 3 T. For comparison, the Earth's magnetic

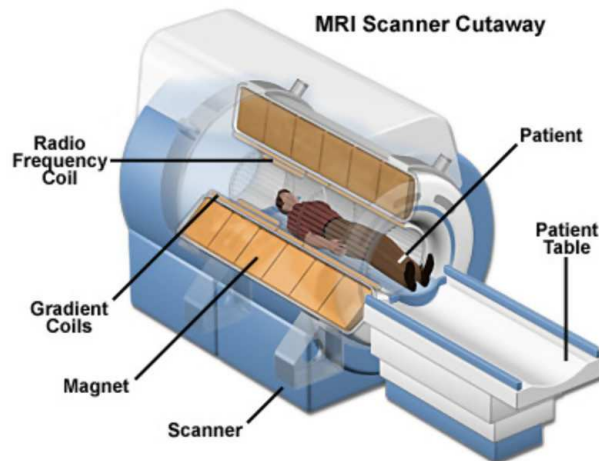


Figure 4.1: MRI Scanner Cutaway.

field is approximately $5 \cdot 10^{-5}$ Tesla. Even so, a number of research centers have scanners considerably stronger than this (e.g., above 10 T).

The static magnetic field aligns the nuclear spins of all the atoms in patient's body. Then, in order to create the MR signal (images), the scanner uses a series of changing magnetic gradients and oscillating electromagnetic fields (RF signals), produced by radio frequency coils, known as a pulse sequence. For MRI, scanners are tuned to the frequency of hydrogen nuclei (known as Larmor frequency, calculated for the hydrogen), which are the most common in the human body due to their prevalence in water molecules. When a pulse is turned on, the RF signal is absorbed by the hydrogen nuclei. As a result, the magnetization alignment of hydrogen nuclear spins within the static magnetic field changes. When the pulse is turned off, the hydrogen nuclei relax to their original equilibrium alignment, which releases electromagnetic energy. This procedure is called Nuclear Magnetic Resonance (NMR)(Figure 4.2). The frequency of the emitted signal (NMR signal) is equal to the Larmor frequency of the excited nuclei. The receiver coils are tuned to the frequency of hydrogen nuclei. Thus, irrelevant noise is excluded and the extremely weak NMR signal can be detected, producing the raw MR signal. Spatial resolution is provided by additional magnetic fields known as gradients. The strength of each gradient changes linearly along a single spatial dimension. Thus, three mutually orthogonal gradients are used to localize a signal in three spatial dimensions [2, 32] .

The pulse sequence is defined on the main computer that controls the scanner. In most fMRI experiments, a second computer creates the stimuli that are presented to the subject and records the subject's behavioural responses. This second computer is synchronized with the first, so that the onset of each stimulus presentation occurs at a precisely controlled

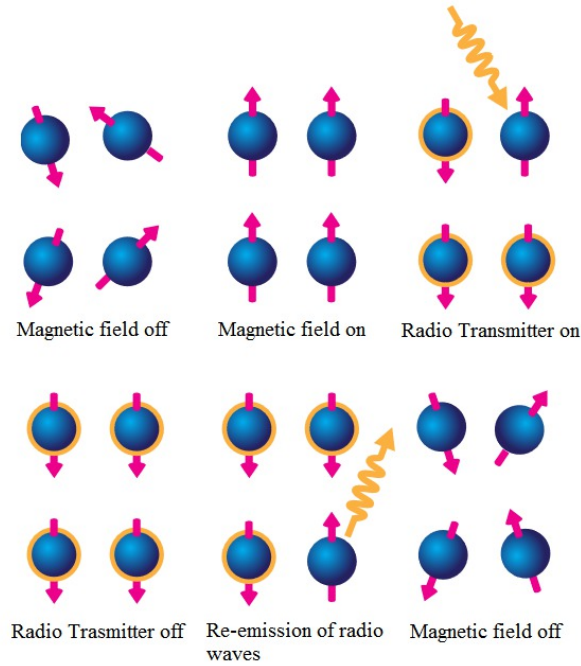


Figure 4.2: Nuclear Magnetic Resonance (NMR).

moment during image acquisition.

Two general types of pulse sequences are common, depending on whether the goal is structural or functional imaging. The goal of structural MR is usually to measure the density of water molecules, which differs, for example, in bone, gray matter, cerebrospinal fluid, and tumors. The vast majority of fMRI experiments measure the blood oxygen-level dependent (BOLD) signal. The physics of this process is complex and far beyond the scope of this thesis. For our purposes, it suffices to know that the BOLD signal is a measure of the ratio of oxygenated to deoxygenated hemoglobin. Hemoglobin is a molecule in the blood that carries oxygen from the lungs to all parts of the body. It has sites to bind up to four oxygen molecules. A key discovery that led eventually to BOLD fMRI was that hemoglobin molecules fully loaded with oxygen have different magnetic properties than hemoglobin molecules with empty binding sites [33].

The theory, which is not yet fully worked out, is that active brain areas consume more oxygen than inactive areas. When neural activity increases in an area, metabolic demands rise and, as a result, the vascular system concentrates oxygenated hemoglobin into the area. An idealized example of this process is shown in Figure 4.3. The concentration of oxygenated hemoglobin into the area causes the ratio of oxygenated to deoxygenated hemoglobin (i.e., the BOLD signal) to rise quickly. As this happens, the vascular system

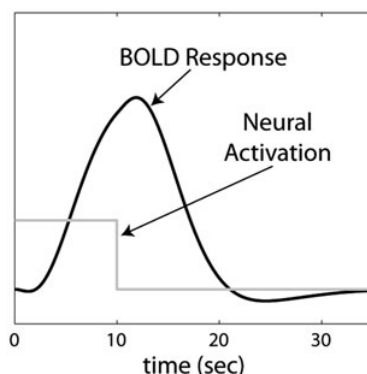


Figure 4.3: A hypothetical BOLD response (black curve) to a constant 10sec neural activation (gray curve) [2].

over compensates, in the sense that the BOLD signal actually rises well above baseline to a peak at around 6sec after the end of the neural activity that elicited these responses. Following this peak, the BOLD signal gradually decays back to baseline over a period of 20–25 sec.

4.3 The Scanning Session

An experimental session that collects fMRI data also commonly includes a variety of other types of scans. At least four different types of scans are commonly acquired.

Typically, the first scan completed in each session is the localizer. This is a quick structural scan (1–2 min) of low spatial resolution and is used only to locate the subject’s brain in three-dimensional space. This knowledge is needed to optimize the location of the slices that will be taken through the brain in the high-resolution structural scan and in the functional scans that follow.

Frequently, the second type of scan completed is the high-resolution structural scan. The structural scan plays a key role in the analysis of the functional data. Because speed is a high priority in fMRI (i.e., to maximize temporal resolution), spatial resolution is sacrificed when collecting functional data. The high-resolution structural scan can compensate somewhat for this loss of spatial information. This is done during preprocessing when the functional data are aligned with the structural image. After completion of this mapping, the spatial coordinates of activation observed during fMRI can be determined by examining the aligned coordinates in the structural image.

The third step is often to collect the functional data. This can be done in one long run that might take 20–30 min to complete, or it can be broken down into 2 or 3 shorter runs,

with brief rests in between. There are many parameter choices to make here, but two are especially important for the subsequent fMRI data analysis. One choice is the time between successive whole brain scans, which is called the repetition time (TR). If the whole brain is scanned, typical TRs range from 2–3 sec. The other important choice is the voxel size, which determines the spatial resolution of the functional data. The brain is a continuous medium, in the sense that neurons exist at (almost) every set of coordinate values inside the brain. fMRI data, however, are discrete. The analog-to-digital conversion is performed by dividing the brain into a set of cubes. These cubes are called voxels (volume-pixels).

A typical voxel size in functional imaging might be $3\text{ mm} \times 3\text{ mm} \times 3.5\text{ mm}$. In this case, in a typical human brain, 33 separate slices might be acquired each containing an 64×64 array of voxels for a whole brain total of 135,168 voxels. In each fMRI run, a BOLD response is recorded every TR seconds in each voxel. Thus, for example, in a 30 min run with a TR of 2 sec, 135,168 BOLD responses could be recorded 900 separate times (i.e., 30 times per minute \times 30 min), for a total of 121,651,200 BOLD values. This is an immense amount of data, and its sheer volume greatly contributes to the difficulties in data analysis.

Many studies stop when the functional data acquisition is complete, but some other types of scans are also common. A fourth common type of scan is the field map. The ideal scanner has a completely uniform magnetic field across its entire bore. Even if this were true, placing a human subject inside the bore distorts this field to some extent. After the placement of the subject inside the scanner, all inhomogeneities in the magnetic field are corrected via a process known as shimming. If shimming is successful, the magnetic field will be uniform at the start of scanning. Sometimes, however, especially in less reliable machines, distortions in the magnetic field will appear in the middle of the session. The field map, which takes only a minute or two to collect, measures the homogeneity of the magnetic field. Thus, the field map can be used during later data analysis to correct for possible nonlinear distortions in the strength of the magnetic field that develop during the course of the scanning session.

4.4 Modeling the BOLD Response

The goal of almost all fMRI experiments is to obtain information about neural activity. However, the BOLD response measured in most fMRI experiments provides only an indirect measure of neural activation [34, 35]. Although it is commonly assumed that the BOLD signal increases with neural activation, it is known that the BOLD response is much more sluggish than the neural activation that is presumed to drive it. As a result, the peak of

the BOLD signal lags considerably behind the peak neural activation (e.g., see Figure 4.3).

Almost all current applications of fMRI assume that the transformation from neural activation to BOLD response can be modeled as a linear, time-invariant system. Although it is becoming increasingly clear that the transformation is, in fact, nonlinear (e.g., [36, 37, 38]). It also appears that these departures from linearity are not severe so long as events are well separated in time (e.g., at least a few seconds apart) and brief exposure durations are avoided [38].

In the linear systems approach, one can conceive of the vascular system that responds to a sudden oxygen debt as a black box. The input is neural activation and the output is the BOLD response. Let $N_i(t)$ denote the neural activation induced by this event at time t and let $B_i(t)$ denote the corresponding BOLD response. Then, from the systems theory perspective, the box represents the set of all mathematical transformations that convert the neural activation $N_i(t)$ into the BOLD response $B_i(t)$. For convenience, we will express this mathematical relationship as

$$B_i(t) = f(N_i(t)), \quad (4.1)$$

where the operator f symbolizes the workings of the black box.

A system of this type is said to be linear and time-invariant if and only if it satisfies the superposition principle, which is stated as follows:

$$\begin{aligned} \text{If } f(N_1(t)) = B_1(t) \quad \text{and} \quad f(N_2(t)) = B_2(t), \\ \text{then } f(\alpha_1 N_1(t) + \alpha_2 N_2(t)) = \alpha_1 B_1(t) + \alpha_2 B_2(t), \end{aligned} \quad (4.2)$$

for any constants α_1, α_2 .

In other words, if we know what the BOLD response is to neural activation $N_1(t)$ and to neural activation $N_2(t)$, then we can determine exactly what the BOLD response will be to any weighted sum of these two neural activations by computing the same weighted sum of the component BOLD responses.

If the superposition principle holds, then there is a straightforward way to determine the BOLD response to any neural activation from the results of one simple experiment. All we need to do is to measure the BOLD response that occurs when the neural activation is an impulse. Denote the BOLD response in this idealized experiment by $h(t)$. In linear systems theory, the function $h(t)$ is called the impulse response function and is the response of the system to an impulse. In the fMRI literature, $h(t)$ is known as the hemodynamic response function (HRF).

If the relationship between neural activation and the BOLD response satisfies superposition, then, once we know the HRF, the BOLD response to any neural activation $N(t)$ can be computed exactly from the convolution integral:

$$B(t) = \int_0^t N(\tau) h(t - \tau) d\tau. \quad (4.3)$$

The convolution integral massively simplifies the analysis of fMRI data and, as a result, it forms the basis for the most popular methods of fMRI data analysis.

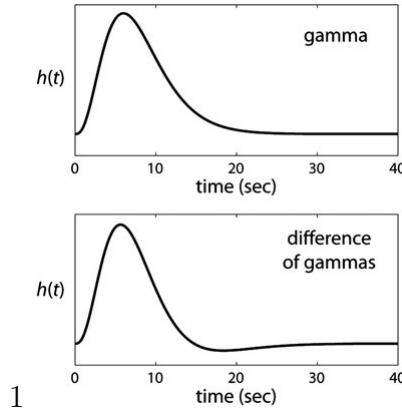


Figure 4.4: Two popular models of the HRF [2].

Given that the HRF plays such a critical role in analysing fMRI data, the natural next question to ask is: how can we determine numerical values of the HRF? A popular method is to select a specific mathematical function for the HRF based on our knowledge of what we think this function should look like. For example, we know the HRF should peak at roughly 6 s and then slowly decay back to baseline. So, we could select a mathematical function with these properties and then just assume that this is a good model of the HRF. In fact, this is, by far, the most popular method for determining the HRF in fMRI data analysis. The most popular choices are a gamma function or the difference of two gamma functions. Examples of both of these models are shown in Figure 4.4.

4.5 Data Analysis

The size of acquired fMRI data greatly complicates its analysis. First, as mentioned above, a typical scanning session generates a huge amount of data. Second, fMRI data is characterized by substantial spatial and temporal correlations. For example, the sluggish nature of the BOLD response means that, if the BOLD response in some voxel is greater

than the average on one TR, then it is also likely to be greater than the average on the ensuing TR. Similarly, because brain tissue in neighbouring voxels will be supplied by a similar vasculature, a large response in one voxel increases the likelihood that a large response will also be observed at neighboring voxels. A third significant challenge to fMRI data analysis is the noisy nature of fMRI data. Typically the signal that the data analysis techniques are trying to find is less than 2 or 3 % of the total BOLD response.

The analysis of fMRI BOLD data is broken down into two general stages - preprocessing and task-related analysis. Preprocessing includes a number of steps that are required to prepare the data for task-related analysis. This includes, for example, aligning the functional and structural scans, correcting for any possible head movements that might have occurred during the functional run, and various types of smoothing (to reduce noise). Typically, the same preprocessing steps are always completed, regardless of the particular research questions that the study was designed to address. In contrast, the task-related analyses include all analyses that are directed at these questions.

4.5.1 Preprocessing

The most common goal of fMRI research is to identify brain areas activated by the task under study. The data that come directly out of the scanner, however, are poorly suited to this goal. The preprocessing of fMRI data includes all transformations that are needed to prepare the data for the more interesting task-related analyses. Preprocessing steps typically are the same for all experiments, so any analysis that does not depend on the specific hypotheses that the experiment was designed to test is typically called preprocessing.

The variability in raw fMRI data is so great that it can easily swamp out the small changes in the BOLD response induced by most cognitive tasks. Some of this variability is unavoidable in the sense that it is due to factors that we cannot control or even measure (e.g., thermal and system noise). But other sources of variability are systematic. For example, when a subject moves his or her head, the BOLD response sampled from each spatial position within the scanner suddenly changes in a predictable manner. The analyses done during preprocessing remove as many of these systematic non-task-related sources of variability as possible.

Typically, the first preprocessing step is slice-time correction. Almost all fMRI data are collected in slices. If the TR is 2.5 s, then the time between the acquisition of the first and last slice will be almost this long. Slice-time correction corrects for these differences in the time when the slices are acquired.

The second step is to correct for variability due to head movement. Arguably, this is probably the most important preprocessing step. Even small, almost imperceptible head movements can badly corrupt fMRI data [32]. When a subject moves his or her head, brain regions will move to new spatial locations within the scanner, and as a result, activation in those regions will be recorded in different voxels than they were before the movement occurred. Mathematical methods for correcting for head movements depend heavily on the assumption that when a subject moves his or her head, the brain does not change shape or size and therefore can be treated as a rigid body. Head movement correction then becomes a problem of rigid body registration (e.g., [39]). The BOLD responses from one TR are taken as the standard and then rigid body movements are performed separately on the data from every other TR until each of these data sets agrees as closely as possible with the data from the standard.

The third step, called co-registration, is to align the structural and functional data. This is critical because the spatial resolution of the functional data is poor. For example, with functional data a voxel size of $3 \times 3 \times 3.5$ mm is common. With structural images, however, the voxel size might be $0.86 \times 0.86 \times 0.89$ mm, which is an improvement in resolution by a factor of almost 50.

The fourth step, normalization, warps the subjects structural image to a standard brain atlas. There are huge individual differences in the sizes and shapes of individual brains, and these differences extend to virtually every identifiable brain region. These differences make it difficult to assign a task-related activation observed in some cluster of voxels to a specific neuroanatomical brain structure. An alternative is to register the structural scan of each subject separately to some standard brain where the coordinates of all major brain structures have already been identified and published in an atlas. Then, we could determine the coordinates of a significant cluster within this standard brain, look these coordinates up in the atlas, and thereby determine which brain region the cluster is in. The process of registering a structural scan to the structural scan from some standard brain is called normalization [22].

Step five spatially smooths the data with the goal of reducing non systematic high frequency spatial noise. In this step, the BOLD value in each voxel is replaced by a weighted average of the BOLD responses in neighbouring voxels. The weight is greatest at the voxel being smoothed and decreases with distance. There are a number of advantages to spatially smoothing fMRI data. Most of these are due to the effects of the smoothing process on noise in the data. First, because smoothing is essentially an averaging operation, it makes the distribution of the BOLD responses more normal (i.e., because of the central limit theorem). Because the statistical models that dominate fMRI data analysis assume

normally distributed noise, smoothing therefore transforms the data in a way that makes it more likely to satisfy the assumptions of our statistical models. A second benefit is that smoothing is required by a number of popular methods for solving the multiple comparisons problem (i.e., those that depend on Gaussian random field theory). A third benefit of smoothing, which is the most important of all, is that it can reduce noise and therefore increase signal-to-noise ratio.

Finally, in step six, temporal filtering (e.g. detrending) is done primarily to reduce the effects of slow fluctuations in the local magnetic field properties of the scanner.

4.5.2 Task-Related Data Analysis

At the pre-processing stages, the quality of the fMRI data is improved. After that, statistical analysis is attempted to determine which voxels are activated by the stimulation. Most fMRI studies are established upon the correlation of hemodynamic response function with stimulation. Activation defines the local intensity changes in the voxels. These methods can be grouped into two broad categories: the univariate methods (hypothesis testing methods), and the multivariate methods (exploratory methods) [40].

The univariate methods attempt to define which voxels can be characterized as activated given one signal model. This allows the parametrization of the response and then the estimation of the model parameters. The univariate methods are widely used to analyse brain data obtained from fMRI. In these methods, signal estimation and the presence or the absence of activation are defined by the statistical test. One of the most popular methods is the generalized linear model (GLM), which is based upon the hypothesis of linear correlation between neuro-activities and the tasks [41].

Multivariate methods are also applied to fMRI data analysis, which extract information from dataset, often with any prior knowledge of the experimental conditions. They use some structural properties, such as decorrelation, independence, similarity measures, that can discriminate between features of interest present in the data. Unlike the univariate methods which carry out voxel-wise statistical analysis, multivariate methods provide statistical inference about the whole brain so as to describe brain responses in terms of spatial patterns [42]. A wide range of multivariate statistical methods is being increasingly employed to analyse the fMRI time series. fMRI data are essentially multivariate in nature, since information about thousands of measured locations (voxels) are being impacted in each scan [43]. Those methods aim at summarizing the spatial and temporal structures of the data. As the distribution of brain regions are involved in a task, it seems to be desirable to use the spatially distributed information from different areas to understand a brain

function. So the multivariate approaches seem to be interesting in this case to consider the spatially distributed information. The most common multivariate methods, among others, are Principal Component Analysis (PCA), Independent Component Analysis (ICA), and Multi-Voxel Pattern Analysis (MVPA).

Tensor factorization models have been also proposed in processing fMRI data. In contrast to PCA and ICA, tensor factorization models are unique under mild conditions, without enforcing any constraints to source signals. In Andersen and Rayens (2004) it was demonstrated how the PARAFAC (CP) model is useful in the analysis of neuroimaging data such as fMRI [44]. However, degeneration of the unconstrained PARAFAC model is a frequent problem in the analysis of fMRI data. Additional applications of multilinear (multiway) modeling in fMRI include the PICA model [45], where the ICA model is extended for tensors. Also, the Shifted PARAFAC model (SCP) has been proposed in processing fMRI data, since time shifts occur naturally in fMRI data. For instance, these could be due to hemodynamic delay [46] or they could arise in stimuli studies [47], where delays play a particularly important role. Finally, the PARAFAC model with orthogonality constraints over the spatial mode (matrix \mathbf{W}) has been proposed in processing fMRI data in [48], in order to reduce cross-talk between spatial components and eliminate degeneration phenomena.

4.6 Experiments & Results

The dataset that we processed is a visuomotor task [49] data available from fMRI GIFT toolbox¹. The dataset consists of three subjects scanned with repetition time $TR=1$ sec. Here, we briefly describe the experiment. The experiment contained two identical but spatially offset, periodic, visual stimuli, shifted by 20 sec from one another. The visual stimuli were projected via an LCD projector onto a rear-projection screen subtending approximately 25 degrees of visual field, visible via a mirror attached to the MRI head coil. The stimuli consisted of an 8 Hz reversing checkerboard pattern presented for 15 sec in the right visual hemifield, followed by 5 sec of an asterisk fixation, followed by 15 sec of checkerboard presented to the left visual hemifield, followed by 20 sec of an asterisk fixation. The 55 sec set of events was repeated four times for a total of 220 sec. The motor stimuli consisted of participants touching their right thumb to each of their four fingers sequentially, back and forth, at a self-paced rate using the hand on the same side on which the visual stimulus is presented.

¹<http://mialab.mrn.org/software/gift/>

As we present in the sequel, fMRI data from this experiment, when analysed with the PARAFAC (with orthogonal constraints) and the SCP models, separates into two different task-related components (one in left visual and motor cortex, one in right visual and motor cortex). Thus, the extracted hemodynamic responses represent visuomotor tasks corresponding to left visual discrimination - left hand coordination and right visual discrimination - right hand coordination.

Data preprocessing

In our experiments, we preprocessed voxel time series by applying an 10^{th} order moving average filter and removing quadratic trends (Fig. 4.5).

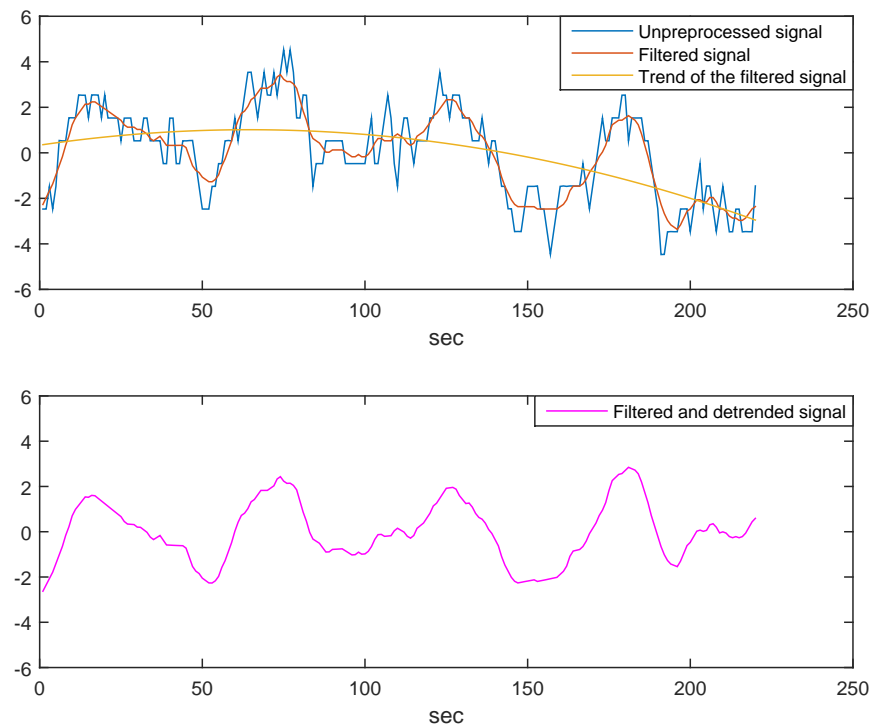


Figure 4.5: An example of fMRI time series preprocessing stages.

Results using the PARAFAC model using orthogonality constraints

Unconstrained PARAFAC tensor decomposition may introduce components that are hard to interpret from a biological perspective (degeneration of the PARAFAC model). For example, one common empirical hypothesis is that different regions in the brain will

be responsible for different tasks. Hence, in a healthy brain, the cross-talk between spatial components should be as small as possible. We can satisfy this by using $\mathbf{W}^T \mathbf{W} = \mathbf{I}$ [48]. Also, for interpretability reasons, we can assume the weighting of spatio-temporal maps in each subject to be nonnegative ($\mathbf{A} \geq \mathbf{0}$).

By fitting a rank-3 model, two common to all subjects components were identified, corresponding to left visual discrimination - left hand coordination and right visual discrimination - right hand coordination hemodynamic responses.

1st Component (Left visual discrimination - left hand coordination)

Strength of 1 st component over subjects		
Subject 1	Subject 2	Subject 3
0.7114	0.6726	0.2037

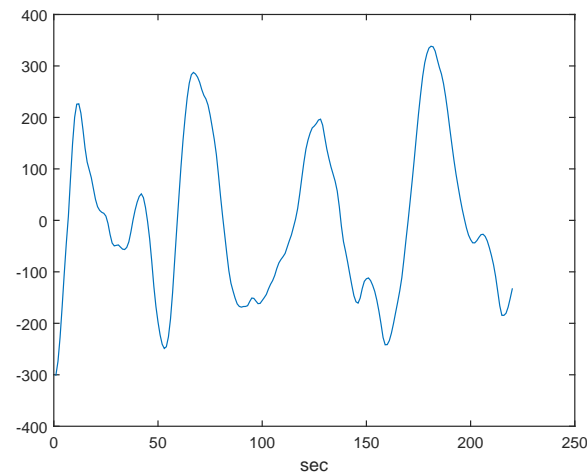


Figure 4.6: Temporal profile of rank-one component that corresponds to left visual discrimination - left hand coordination.

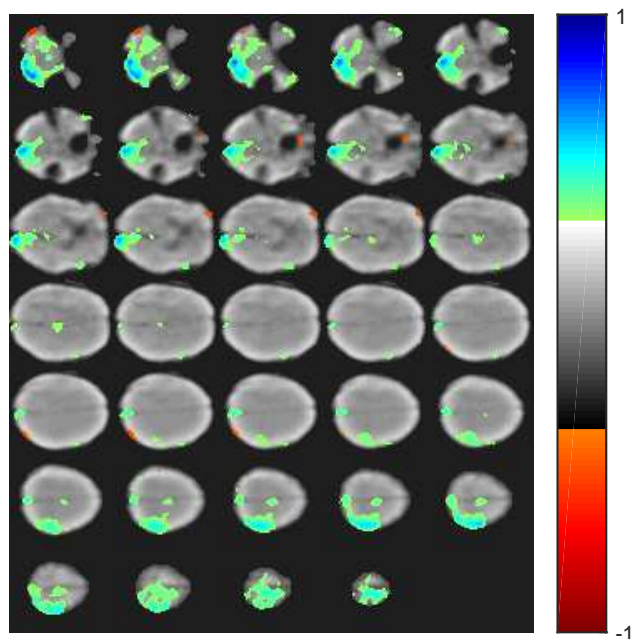


Figure 4.7: Spatial profile of rank-one component that corresponds to left visual discrimination - left hand coordination brain areas. The map was thresholded such that the 5% of the voxels with the largest voxel absolute score are shown.

2^{nd} Component (Right visual discrimination - right hand coordination)

Strength of 2^{nd} component over subjects		
Subject 1	Subject 2	Subject 3
0.5228	0.1436	0.8403

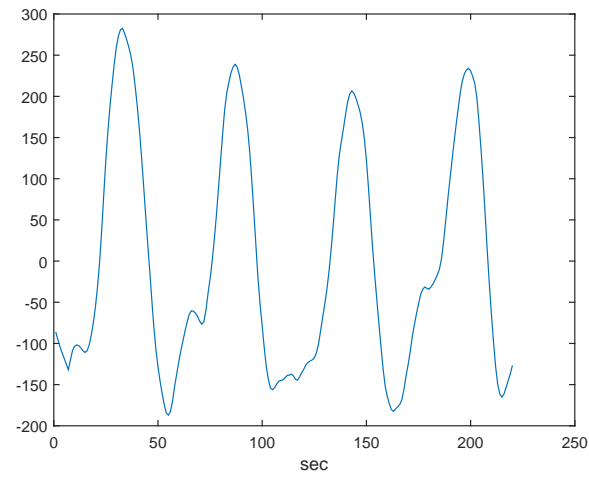


Figure 4.8: Temporal profile of rank-one component that corresponds to right visual discrimination - right hand coordination.

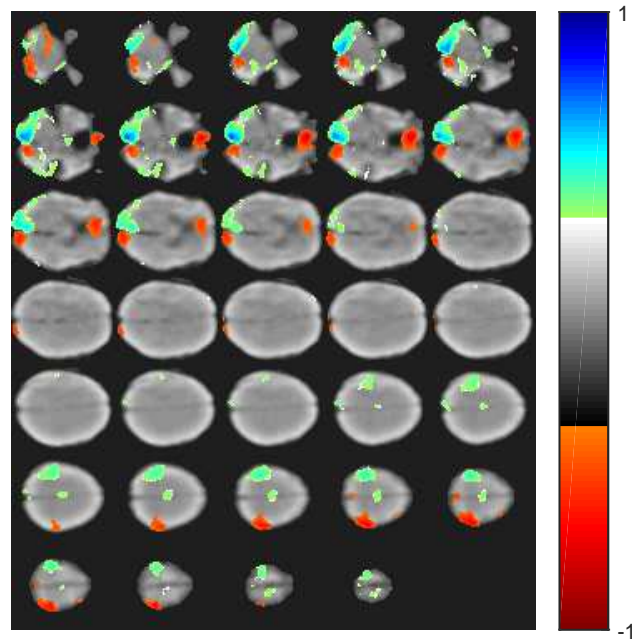


Figure 4.9: Spatial profile of rank-one component that corresponds to right visual discrimination - right hand coordination brain areas. The map was thresholded such that the 5% of the voxels with the largest voxel absolute score are shown.

Results using the SCP model

In the analysis of the fMRI data using the SCP model, matrix \mathbf{A} pertained to subject strengths, matrix \mathbf{W} to the spatial activities (i.e. voxel strengths), matrix \mathbf{S} to the temporal profiles, while delays occur over the spatial mode. Matrix \mathbf{A} was constrained to be nonnegative such that only activities that are similar across subjects are estimated. Matrix \mathbf{W} was also constrained to be nonnegative since the estimated BOLD response in \mathbf{S} is assumed to have a similar temporal profile across the voxels.

By fitting a rank-2 model, one common to all subjects component was identified, corresponding to both left visual discrimination - left hand coordination and right visual discrimination - right hand coordination hemodynamic responses.

Common component

Strength of common component over subjects		
Subject 1	Subject 2	Subject 3
0.8063	0.1932	0.5591

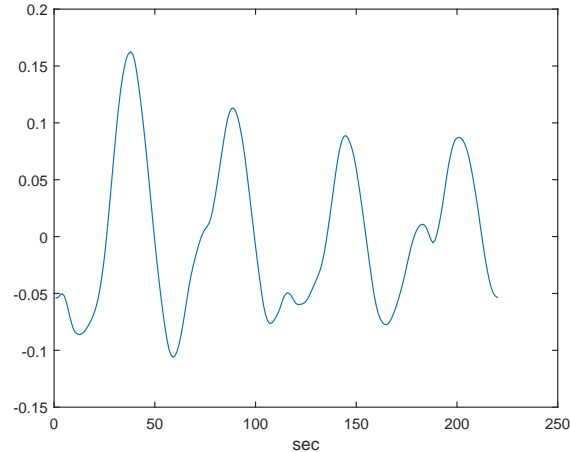


Figure 4.10: Temporal profile of rank-one component that corresponds to both left visual discrimination - left hand coordination and right visual discrimination - right hand coordination hemodynamic responses.

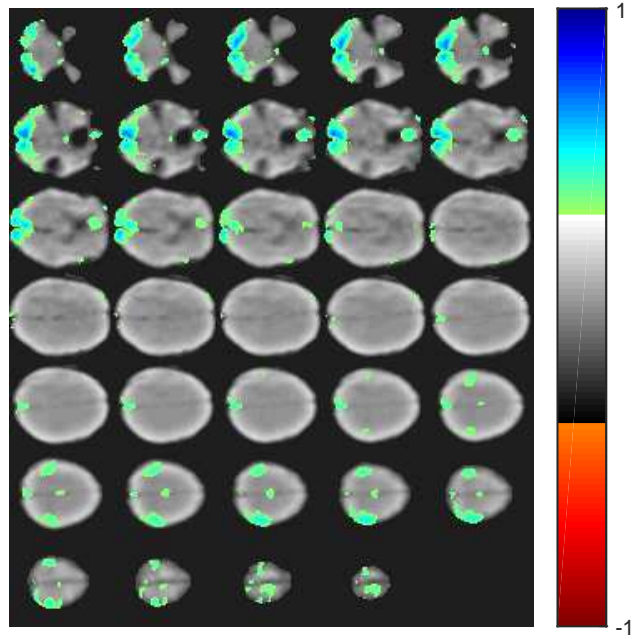


Figure 4.11: Spatial profile of rank-one component that corresponds to both left visual discrimination - left hand coordination and right visual discrimination - right hand coordination brain areas. The map was thresholded such that the 5% of the voxels with the largest voxel score are shown.

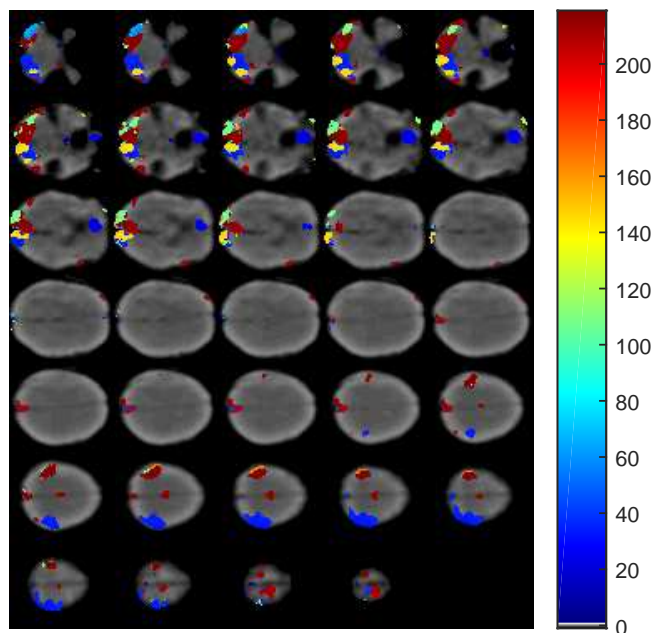


Figure 4.12: Spatial delay map of rank-one component that corresponds to both left visual discrimination - left hand coordination and right visual discrimination - right hand coordination brain areas. The map contains the delays that correspond to the 5% of the voxels with the largest voxel score are shown.

Conclusions on results

As we see, both the models were able to identify the corresponding visual and motor areas. However, in SCP model, relative shift and offset ambiguities make the delay interpretation complex.

Chapter 5

Discussion & Future Work

5.1 Conclusion

We considered the BSS problem. We studied how the NMF model can be used in BSS problems and how it can be extended into more complicated models, when we make assumptions about the signals propagation environment (anechoic & echoic environments). Also, we studied how assumptions about sparsity and smoothness can be incorporated into these models and when the NMF model and K-means clustering are related. Next, we studied how the PARAFAC model can be used in BSS problems, when more than one realizations, of the same BSS problem, are available and how we can extend it under the assumption of anechoic environments. Finally, we introduced fMRI and we present how the PARAFAC model and the Shifted PARAFAC (SCP) model can be used in exploratory analysis of fMRI data. We conclude that the PARAFAC model with orthogonality constraints seems to be the most suitable model in fMRI data analysis.

5.2 Future Work

In this section, we refer possible directions on how this work could be continued.

5.2.1 The Convulsive PARAFAC model

For the case of fMRI data analysis, the Shifted CP model is constrained such that each time profile only has one specific delay value for each voxel. Assuming that a set of simple time signals appear at different time instances among voxels, it is possible to model the propagation environment as an echoic one. Thus, Shifted CP model can be extended in order to allow an arbitrary number of possible component delays at each voxel within the length T of the convulsive filter [50].

5.2.2 Tensor Rank Estimation

Model selection in tensor decomposition is important for real applications if the rank of the original data tensor is unknown and the observed tensor is noisy. Various techniques, among them CORCONDIA (Core Consistency Diagnostics) and ARD (Automatic Relevance Determination), have been proposed for estimating the rank of a tensor, although finding an efficient method for estimating the tensor rank is still an open problem.

5.2.3 Higher-order tensors

In this thesis, we studied how *3-way* tensors can be factorized using the PARAFAC and the Shifted CP models. These models can be extended for higher-*way* tensors in order to model higher dimensional phenomena and datasets.

5.2.4 The Tucker model

The Tucker model is an extension of the PARAFAC model, since allows the interaction of each signature in a latent factor with signatures of other latent factors. The Tucker model has been proposed for a wide spectrum of applications. However, the Tucker model is, in general, not unique. Thus, an constrained framework, designed on specific application, is needed.

5.2.5 Statistical analysis methods on subject variability factor

Analysing fMRI data with the PARAFAC and the SCP model provides factors that indicate the presence level of spatio-temporal components over subjects. Statistical methods could be developed, in order to exploit the presence variability of common spatio-temporal across subject, for diagnostic purposes.

Appendix A

A.1 Derivative of f_F with respect to matrix \mathbf{S}

We have the cost function

$$\begin{aligned}
 f_F(\mathbf{W}, \mathbf{S}, \boldsymbol{\tau}) &= \frac{1}{2} \sum_{f=1}^M \left\| \tilde{\mathbf{V}}_{:,f} - \underbrace{\left[\mathbf{W} \circ e^{-j2\pi \frac{(f-1)\boldsymbol{\tau}}{M}} \right]}_{\mathbf{W}^f} \mathbf{S} \mathbf{Q}_{:,f} \right\|_2^2 \\
 &= \frac{1}{2} \sum_{f=1}^M \left\| \tilde{\mathbf{V}}_{:,f} - \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \right\|_2^2 \\
 &= \frac{1}{2} \sum_{f=1}^M \left\| \tilde{\mathbf{V}}_{:,f} \right\|_2^2 - 2\Re \left\{ \tilde{\mathbf{V}}_{:,f}^H \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \right\} + \mathbf{Q}_{:,f}^H \mathbf{S}^T \mathbf{W}^{fH} \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f}.
 \end{aligned}$$

Then, the derivative of f_F with respect to \mathbf{S} is equal to

$$\frac{\partial f_F}{\partial \mathbf{S}} = \frac{1}{2} \sum_{f=1}^M \frac{\partial \left(\mathbf{Q}_{:,f}^H \mathbf{S}^T \mathbf{W}^{fH} \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \right)}{\partial \mathbf{S}} - \frac{\partial \left(2\Re \left\{ \tilde{\mathbf{V}}_{:,f}^H \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \right\} \right)}{\partial \mathbf{S}},$$

where

$$\begin{aligned}
 \frac{\partial \left(2\Re \left\{ \tilde{\mathbf{V}}_{:,f}^H \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \right\} \right)}{\partial \mathbf{S}} &= \frac{\partial \left(\tilde{\mathbf{V}}_{:,f}^H \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \right)}{\partial \mathbf{S}} + \frac{\partial \left(\tilde{\mathbf{V}}_{:,f}^T \mathbf{W}^{f*} \mathbf{S} \mathbf{Q}_{:,f}^* \right)}{\partial \mathbf{S}} \\
 &\stackrel{(1)}{=} \left(\mathbf{W}^{fH} \tilde{\mathbf{V}}_{:,f} \mathbf{Q}_{f,:}^* \right)^* + \mathbf{W}^{fH} \tilde{\mathbf{V}}_{:,f} \mathbf{Q}_{f,:}^* \\
 &= 2\Re \left\{ \mathbf{W}^{fH} \tilde{\mathbf{V}}_{:,f} \mathbf{Q}_{f,:}^* \right\}
 \end{aligned}$$

and

$$\begin{aligned}
 \frac{\partial \left(\mathbf{Q}_{:,f}^H \mathbf{S}^T \mathbf{W}^{fH} \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \right)}{\partial \mathbf{S}} &= \frac{\partial \left(\text{Tr} \left(\mathbf{Q}_{:,f}^H \mathbf{S}^T \mathbf{W}^{fH} \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \right) \right)}{\partial \mathbf{S}} \\
 &\stackrel{(2)}{=} \mathbf{W}^{fH} \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \mathbf{Q}_{f,:}^* + \left(\mathbf{W}^{fH} \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \mathbf{Q}_{f,:}^* \right)^* \\
 &= 2\Re \left\{ \mathbf{W}^{fH} \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \mathbf{Q}_{f,:}^* \right\},
 \end{aligned}$$

where in points (1) and (2) were used relations (100) and (118) from [51], respectively.

So, the first derivative of f_F with respect to \mathbf{S} is

$$\begin{aligned}
\frac{\partial f_F}{\partial \mathbf{S}} &= \frac{1}{2} \sum_{f=1}^M 2\Re \left\{ \mathbf{W}^{fH} \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \mathbf{Q}_{f,:}^* \right\} - 2\Re \left\{ \mathbf{W}^{fH} \tilde{\mathbf{V}}_{:,f} \mathbf{Q}_{f,:}^* \right\} \\
&= \sum_{f=1}^M \Re \left\{ \mathbf{W}^{fH} \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \mathbf{Q}_{f,:}^* - \mathbf{W}^{fH} \tilde{\mathbf{V}}_{:,f} \mathbf{Q}_{f,:}^* \right\} \\
&= \sum_{f=1}^M \Re \left\{ \mathbf{W}^{fH} \left[\mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} - \tilde{\mathbf{V}}_{:,f} \right] \mathbf{Q}_{f,:}^* \right\} \\
&= \Re \left\{ \sum_{f=1}^M \mathbf{W}^{fH} \left[\mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} - \tilde{\mathbf{V}}_{:,f} \right] \mathbf{Q}_{f,:}^* \right\}.
\end{aligned}$$

Since $\mathbf{Q}_{f,:}^* = \mathbf{e}_f^T \mathbf{Q}^*$, the first derivative of f_F with respect to \mathbf{S} equals to

$$\begin{aligned}
\frac{\partial f_F}{\partial \mathbf{S}} &= \Re \left\{ \sum_{f=1}^M \mathbf{W}^{fH} \left[\mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} - \tilde{\mathbf{V}}_{:,f} \right] \mathbf{e}_f^T \mathbf{Q}^* \right\} \\
&= \Re \left\{ \left[\sum_{f=1}^M \mathbf{W}^{fH} \left[\mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} - \tilde{\mathbf{V}}_{:,f} \right] \mathbf{e}_f^T \right] \mathbf{Q}^* \right\}.
\end{aligned}$$

A.2 Splitting of $\frac{\partial f_F}{\partial \mathbf{S}}$ into nonnegative parts

In this part, we prove that $\frac{\partial f_F}{\partial \mathbf{S}}$, in Shifted NMF model, is equal to the subtraction of two nonnegative matrices. As we showed in the previous section, $\frac{\partial f_F}{\partial \mathbf{S}}$ can be written as

$$\begin{aligned} \frac{\partial f_F}{\partial \mathbf{S}} &= \sum_{f=1}^M \Re \left\{ \mathbf{W}^{fH} \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \mathbf{Q}_{f,:}^* \right\} - \Re \left\{ \mathbf{W}^{fH} \tilde{\mathbf{V}}_{:,f} \mathbf{Q}_{f,:}^* \right\} \\ &= \underbrace{\left[\sum_{f=1}^M \Re \left\{ \mathbf{W}^{fH} \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \mathbf{Q}_{f,:}^* \right\} \right]}_{\mathbf{G}^+} - \underbrace{\left[\sum_{f=1}^M \Re \left\{ \mathbf{W}^{fH} \tilde{\mathbf{V}}_{:,f} \mathbf{Q}_{f,:}^* \right\} \right]}_{\mathbf{G}^-} \\ &= \mathbf{G}^+ - \mathbf{G}^-. \end{aligned} \quad (\text{A.1})$$

So we have to prove that matrices \mathbf{G}^+ and \mathbf{G}^- are nonnegative.

We begin with matrix \mathbf{G}^+ . For all $d' \in \{1, \dots, D\}$ and $\{m \in \{1, \dots, M\}$

$$\begin{aligned} [\mathbf{G}^+]_{d',m} &= \left[\sum_{f=1}^M \Re \left\{ \mathbf{W}^{fH} \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \mathbf{Q}_{f,:}^* \right\} \right]_{d',m} \\ &= \sum_{f=1}^M \Re \left\{ \left[\underbrace{\mathbf{W}^{fH} \mathbf{W}^f \mathbf{S} \mathbf{Q}_{:,f} \mathbf{Q}_{f,:}^*}_{\mathbf{G}^{+f}} \right]_{d',m} \right\} \\ &= \sum_{f=1}^M \Re \left\{ [\mathbf{G}^{+f}]_{d',m} \right\}. \end{aligned} \quad (\text{A.2})$$

Each element of matrix \mathbf{G}^{+f} , for all $d' \in \{1, \dots, D\}$, $\{m \in \{1, \dots, M\}$ is equal to

$$\begin{aligned} [\mathbf{G}^{+f}]_{d',m} &= \sum_{n=1}^N \mathbf{W}_{d',n}^{fH} \sum_{d=1}^D \mathbf{W}_{n,d}^f \sum_{u=1}^M \mathbf{S}_{d,u} \mathbf{Q}_{u,f} \mathbf{Q}_{f,m}^* \\ &= \sum_{n=1}^N \sum_{d=1}^D \sum_{u=1}^M \mathbf{W}_{d',n}^{fH} \mathbf{W}_{n,d}^f \mathbf{S}_{d,u} \mathbf{Q}_{u,f} \mathbf{Q}_{f,m}^*. \end{aligned} \quad (\text{A.3})$$

Thus,

$$\begin{aligned} [\mathbf{G}^+]_{d',m} &= \sum_{f=1}^M \Re \left\{ [\mathbf{G}^{+f}]_{d',m} \right\} \\ &= \frac{1}{M} \sum_{n=1}^N \mathbf{W}_{n,d'} \sum_{d=1}^D \mathbf{W}_{n,d} \sum_{u=1}^M \mathbf{S}_{d,u} \sum_{f=1}^M \cos \left(2\pi (f-1) \frac{\tau_{n,d'} - \tau_{n,d} + m - u}{M} \right) \end{aligned}$$

We focus on terms $\sum_{u=1}^M \mathbf{S}_{d,u} \sum_{f=1}^M \cos \left(2\pi (f-1) \frac{\tau_{n,d'} - \tau_{n,d} + m - u}{M} \right)$ and we set $b = \tau_{n,d'} - \tau_{n,d} + m - u$. Then,

$$\begin{aligned} \sum_{u=1}^M \mathbf{S}_{d,u} \sum_{f=1}^M \cos \left(2\pi (f-1) \frac{\tau_{n,d'} - \tau_{n,d} + m - u}{M} \right) &= \sum_{u=1}^M \mathbf{S}_{d,u} \sum_{f=0}^{M-1} \cos \left(f 2\pi \frac{b}{M} \right) \\ &= \frac{1}{2} \sum_{u=1}^M \mathbf{S}_{d,u} \left(1 + \frac{\sin \left((2M-1) \pi \frac{b}{M} \right)}{\sin \left(\pi \frac{b}{M} \right)} \right). \end{aligned}$$

In this point, we study all possible values of $1 + \frac{\sin \left((2M-1) \pi \frac{b}{M} \right)}{\sin \left(\pi \frac{b}{M} \right)}$ for all possible values of b .

If $b \rightarrow 0$, then quantity $\frac{\sin \left((2M-1) \pi \frac{b}{M} \right)}{\sin \left(\pi \frac{b}{M} \right)}$ is indeterminate. Thus, its value can be calculated by the L'Hospital's rule, as

$$\begin{aligned} \lim_{b \rightarrow 0} 1 + \frac{\sin \left((2M-1) \pi \frac{b}{M} \right)}{\sin \left(\pi \frac{b}{M} \right)} &= 1 + \lim_{b \rightarrow 0} \frac{\cos \left((2M-1) \pi \frac{b}{M} \right) \frac{\pi(2M-1)}{M}}{\cos \left(\pi \frac{b}{M} \right) \left(\frac{\pi}{M} \right)} \\ &= 1 + (2M-1) \\ &= 2M. \end{aligned} \tag{A.4}$$

If $b \neq 0$, then let's assume that $\tau_{n,d'}, \tau_{n,d} \in \mathbb{Z}$. Then $b \in \mathbb{Z}^*$, and

$$\begin{aligned} 1 + \frac{\sin \left((2M-1) \pi \frac{b}{M} \right)}{\sin \left(\pi \frac{b}{M} \right)} &= 1 + \frac{\sin \left(2\pi b - \pi \frac{b}{M} \right)}{\sin \left(\pi \frac{b}{M} \right)} \\ &= 1 + \frac{-\sin \left(\pi \frac{b}{M} \right)}{\sin \left(\pi \frac{b}{M} \right)} \\ &= 0. \end{aligned} \tag{A.5}$$

If $b \neq 0$ and $\tau_{n,d'}, \tau_{n,d} \in \mathbb{R}$, then quantity $1 + \frac{\sin \left((2M-1) \pi \frac{b}{M} \right)}{\sin \left(\pi \frac{b}{M} \right)}$ has not a fixed sign for all $b \in \mathbb{R}^*$.

Therefore, if $\tau_{n,d'}, \tau_{n,d} \in \mathbb{Z}$, then

$$\sum_{u=1}^M \mathbf{S}_{d,u} \sum_{f=1}^M \cos \left(2\pi (f-1) \frac{\tau_{n,d'} - \tau_{n,d} + m - u}{M} \right) = M \mathbf{S}_{d, \tau_{n,d'} - \tau_{n,d} + m} \geq 0. \quad (\text{A.6})$$

If $\tau_{n,d'}, \tau_{n,d} \in \mathbb{R}$, then $\sum_{u=1}^M \mathbf{S}_{d,u} \sum_{f=1}^M \cos \left(2\pi (f-1) \frac{\tau_{n,d'} - \tau_{n,d} + m - u}{M} \right)$ is not always nonnegative.

We now consider \mathbf{G}^- . For all $d' \in \{1, \dots, D\}$ and $\{m \in \{1, \dots, M\}$

$$\begin{aligned} [\mathbf{G}^-]_{d',m} &= \left[\sum_{f=1}^M \Re \left\{ \mathbf{W}^{fH} \tilde{\mathbf{V}}_{:,f} \mathbf{Q}_{f,:}^* \right\} \right]_{d',m} \\ &= \left[\sum_{f=1}^M \Re \left\{ \mathbf{W}^{fH} \mathbf{V} \mathbf{Q}_{:,f} \mathbf{Q}_{f,:}^* \right\} \right]_{d',m} \\ &= \sum_{f=1}^M \Re \left\{ [\mathbf{G}^{-f}]_{d',m} \right\}. \end{aligned} \quad (\text{A.7})$$

Each element of matrix \mathbf{G}^{-f} , for all $d' \in \{1, \dots, D\}$, $\{m \in \{1, \dots, M\}$ is equal to

$$\begin{aligned} [\mathbf{G}^{-f}]_{d',m} &= \sum_{n=1}^N \mathbf{W}_{d',n}^{fH} \sum_{u=1}^M \mathbf{V}_{n,u} \mathbf{Q}_{u,f} \mathbf{Q}_{f,m}^* \\ &= \sum_{n=1}^N \sum_{u=1}^M \mathbf{W}_{d',n}^{fH} \mathbf{V}_{n,u} \mathbf{Q}_{u,f} \mathbf{Q}_{f,m}^*. \end{aligned} \quad (\text{A.8})$$

Thus, for all $d' \in \{1, \dots, D\}$ and $\{m \in \{1, \dots, M\}$,

$$\begin{aligned} [\mathbf{G}^-]_{d',m} &= \sum_{f=1}^M \left[\Re \left\{ \mathbf{G}^{-f} \right\} \right]_{d',m} \\ &= \sum_{n=1}^N \sum_{u=1}^M \mathbf{W}_{n,d'} \mathbf{V}_{n,u} \sum_{f=1}^M \cos \left(2\pi (f-1) \frac{(\tau_{n,d'} + m - u)}{M} \right) \\ &= \frac{1}{2} \sum_{n=1}^N \mathbf{W}_{n,d'} \sum_{u=1}^M \mathbf{V}_{n,u} \left(1 + \frac{\sin \left((2M-1) \pi \frac{\tau_{n,d'} + m - u}{M} \right)}{\sin \left(\pi \frac{\tau_{n,d'} + m - u}{M} \right)} \right). \end{aligned} \quad (\text{A.9})$$

We focus on terms $\sum_{u=1}^M \mathbf{V}_{n,u} \left(1 + \frac{\sin\left((2M-1)\pi \frac{\tau_{n,d'}+m-u}{M}\right)}{\sin\left(\pi \frac{\tau_{n,d'}+m-u}{M}\right)} \right)$ and we set

$b = \tau_{n,d'} + m - u$. Then,

$$\sum_{u=1}^M \mathbf{V}_{n,u} \left(1 + \frac{\sin\left((2M-1)\pi \frac{\tau_{n,d'}+m-u}{M}\right)}{\sin\left(\pi \frac{\tau_{n,d'}+m-u}{M}\right)} \right) = \sum_{u=1}^M \mathbf{V}_{n,u} \left(1 + \frac{\sin\left((2M-1)\pi \frac{b}{M}\right)}{\sin\left(\pi \frac{b}{M}\right)} \right).$$

In this point, we study all possible values of $1 + \frac{\sin\left((2M-1)\pi \frac{b}{M}\right)}{\sin\left(\pi \frac{b}{M}\right)}$ for all possible values of b .

If $b \rightarrow 0$, then quantity $\frac{\sin\left((2M-1)\pi \frac{b}{M}\right)}{\sin\left(\pi \frac{b}{M}\right)}$ is indeterminate. Thus, its value can be calculated by the L'Hospital's rule, as

$$\begin{aligned} \lim_{b \rightarrow 0} 1 + \frac{\sin\left((2M-1)\pi \frac{b}{M}\right)}{\sin\left(\pi \frac{b}{M}\right)} &= 1 + \lim_{b \rightarrow 0} \frac{\cos\left((2M-1)\pi \frac{b}{M}\right) \frac{\pi(2M-1)}{M}}{\cos\left(\pi \frac{b}{M}\right) \left(\frac{\pi}{M}\right)} \\ &= 1 + (2M-1) \\ &= 2M. \end{aligned} \tag{A.10}$$

If $b \neq 0$, then let's assume that $\tau_{n,d'} \in \mathbb{Z}$. Then $b \in \mathbb{Z}^*$, and

$$\begin{aligned} 1 + \frac{\sin\left((2M-1)\pi \frac{b}{M}\right)}{\sin\left(\pi \frac{b}{M}\right)} &= 1 + \frac{\sin\left(2\pi b - \pi \frac{b}{M}\right)}{\sin\left(\pi \frac{b}{M}\right)} \\ &= 1 + \frac{-\sin\left(\pi \frac{b}{M}\right)}{\sin\left(\pi \frac{b}{M}\right)} \\ &= 0. \end{aligned} \tag{A.11}$$

If $b \neq 0$ and $\tau_{n,d'} \in \mathbb{R}$, then quantity $1 + \frac{\sin\left((2M-1)\pi \frac{b}{M}\right)}{\sin\left(\pi \frac{b}{M}\right)}$ has not a fixed sign for all $b \in \mathbb{R}^*$.

Therefore, if $\tau_{n,d'} \in \mathbb{Z}$, then

$$\sum_{u=1}^M \mathbf{V}_{n,u} \left(1 + \frac{\sin\left((2M-1)\pi \frac{\tau_{n,d'}+m-u}{M}\right)}{\sin\left(\pi \frac{\tau_{n,d'}+m-u}{M}\right)} \right) = M \mathbf{V}_{n,\tau_{n,d'}+m} \geq 0. \tag{A.12}$$

If $\tau_{n,d'} \in \mathbb{R}$, then $\sum_{u=1}^M \mathbf{V}_{n,u} \left(1 + \frac{\sin\left((2M-1)\pi \frac{\tau_{n,d'}+m-u}{M}\right)}{\sin\left(\pi \frac{\tau_{n,d'}+m-u}{M}\right)} \right)$ is not always nonnegative.

A.3 Derivative of f_F with respect to matrix τ

First, we analyse function f_F as

$$\begin{aligned}
f_F(\mathbf{W}, \tilde{\mathbf{S}}, \tau) &= \frac{1}{2} \sum_{n=1}^N \sum_{f=1}^M \left\| \tilde{\mathbf{v}}_{n,f} - \sum_{d=1}^D \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right\|^2 \\
&= \frac{1}{2} \sum_{n=1}^N \sum_{f=1}^M \left(\tilde{\mathbf{v}}_{n,f} - \sum_{d=1}^D \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right)^* \left(\tilde{\mathbf{v}}_{n,f} - \sum_{d=1}^D \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right) \\
&= \frac{1}{2} \sum_{n=1}^N \sum_{f=1}^M \left(\tilde{\mathbf{v}}_{n,f}^* - \sum_{d=1}^D \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \right) \left(\tilde{\mathbf{v}}_{n,f} - \sum_{d=1}^D \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right) \\
&= \frac{1}{2} \sum_{n=1}^N \sum_{f=1}^M \tilde{\mathbf{v}}_{n,f}^* \tilde{\mathbf{v}}_{n,f} - \tilde{\mathbf{v}}_{n,f}^* \sum_{d=1}^D \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} - \tilde{\mathbf{v}}_{n,f} \sum_{d=1}^D \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \\
&\quad + \left(\sum_{d=1}^D \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \right) \left(\sum_{d=1}^D \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right).
\end{aligned}$$

Then, the first derivative of f_F with respect to $\boldsymbol{\tau}$ is equal to

$$\begin{aligned}
& \frac{\partial f_F(\mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau})}{\partial \boldsymbol{\tau}_{n,d}} = \\
& \frac{1}{2} \sum_{f=1}^M - \left(-j2\pi \frac{(f-1)}{M} \right) \tilde{\mathbf{V}}_{n,f}^* \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} - \left(j2\pi \frac{(f-1)}{M} \right) \tilde{\mathbf{V}}_{n,f} \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \\
& \quad + \left(\sum_{d'=1}^D \mathbf{W}_{n,d'} e^{j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'}^* \right) \left(\left(-j2\pi \frac{(f-1)}{M} \right) \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right) \\
& \quad + \left(\left(j2\pi \frac{(f-1)}{M} \right) \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \right) \left(\sum_{d'=1}^D \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \\
& = \frac{1}{2} \sum_{f=1}^M \left(-j2\pi \frac{(f-1)}{M} \right) \mathbf{W}_{n,d} \left(\tilde{\mathbf{V}}_{n,f} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* - \tilde{\mathbf{V}}_{n,f}^* e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right) \\
& \quad + \left(-j2\pi \frac{(f-1)}{M} \right) \mathbf{W}_{n,d} \left(\sum_{d'=1}^D \mathbf{W}_{n,d'} e^{j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'}^* \right) \left(e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right) \\
& \quad - \left(-j2\pi \frac{(f-1)}{M} \right) \mathbf{W}_{n,d} \left(\sum_{d'=1}^D \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \left(e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \right) \\
& = \frac{1}{2} \sum_{f=1}^M \left(-j2\pi \frac{(f-1)}{M} \right) \mathbf{W}_{n,d} 2j \\
& \quad \left(\Im \left\{ \tilde{\mathbf{V}}_{n,f} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \right\} - \Im \left\{ \left(\sum_{d'=1}^D \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \left(e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \right) \right\} \right) \\
& = \sum_{f=1}^M \left(2\pi \frac{(f-1)}{M} \right) \mathbf{W}_{n,d} \left(\Im \left\{ \tilde{\mathbf{V}}_{n,f} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* - \left(\sum_{d'=1}^D \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \left(e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \right) \right\} \right) \\
& = \sum_{f=1}^M \left(2\pi \frac{(f-1)}{M} \right) \mathbf{W}_{n,d} \Im \left\{ \left(\tilde{\mathbf{V}}_{n,f} - \sum_{d'=1}^D \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \left(e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \right) \right\} \\
& = 2\pi \frac{\mathbf{W}_{n,d}}{M} \sum_{f=1}^M (f-1) \Im \left\{ e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \left(\tilde{\mathbf{V}}_{n,f} - \sum_{d'=1}^D \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \right\}.
\end{aligned}$$

A.4 Hessian of f_F with respect to matrix τ

The Hessian of function f_F with respect to τ is calculated for three cases.

- Case 1: $n' = n$ and $d' = d$. Then the Hessian takes the following form

$$\begin{aligned}
& \frac{\partial^2 f_F(\mathbf{W}, \tilde{\mathbf{S}}, \tau)}{\partial^2 \tau_{n,d}} = \\
& \frac{1}{2} \sum_{f=1}^M \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{W}_{n,d} \tilde{\mathbf{V}}_{n,f} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* + \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{W}_{n,d} \tilde{\mathbf{V}}_{k,n,f}^* e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \\
& - \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \left(\sum_{d'=1}^D \mathbf{W}_{n,d'} e^{j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'}^* \right) \\
& + \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{W}_{n,d}^2 |\tilde{\mathbf{S}}_{f,d}|^2 \\
& - \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \left(\sum_{d'=1}^D \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \\
& + \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{W}_{n,d}^2 |\tilde{\mathbf{S}}_{f,d}|^2 \\
& = \frac{1}{2} \sum_{f=1}^M \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \left(\tilde{\mathbf{V}}_{n,f} - \sum_{d'=1}^D \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \\
& + \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \left(\tilde{\mathbf{V}}_{n,f}^* - \sum_{d'=1}^D \mathbf{W}_{n,d'} e^{j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'}^* \right) \\
& + 2 \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{W}_{n,d}^2 |\tilde{\mathbf{S}}_{f,d}|^2 \\
& = \sum_{f=1}^M \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{W}_{n,d} \\
& \quad \left(\Re \left\{ e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \left(\tilde{\mathbf{V}}_{n,f} - \sum_{d'=1}^D \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \right\} + \mathbf{W}_{n,d} |\tilde{\mathbf{S}}_{f,d}|^2 \right) \\
& = \mathbf{W}_{n,d} \left(\frac{2\pi}{M} \right)^2 \sum_{f=1}^M (f-1)^2 \\
& \quad \left(\Re \left\{ e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \sum_{k=1}^K \left(\tilde{\mathbf{V}}_{n,f} - \sum_{d'=1}^D \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \right\} + \mathbf{W}_{n,d} |\tilde{\mathbf{S}}_{f,d}|^2 \right).
\end{aligned}$$

- Case 2: $n' = n$ and $d' \neq d$. Then the Hessian takes the following form

$$\begin{aligned}
\frac{\partial^2 f_F(\mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau})}{\partial \boldsymbol{\tau}_{n,d} \partial \boldsymbol{\tau}_{n,d'}} &= \frac{1}{2} \sum_{f=1}^M \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{W}_{n,d} \mathbf{W}_{n,d'} e^{j2\pi \frac{(f-1)(\tau_{n,d'} - \tau_{n,d})}{M}} \tilde{\mathbf{S}}_{f,d'}^* \tilde{\mathbf{S}}_{f,d} \\
&\quad + \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{W}_{n,d'} \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)(\tau_{n,d} - \tau_{n,d'})}{M}} \tilde{\mathbf{S}}_{f,d}^* \tilde{\mathbf{S}}_{f,d'} \\
&= \sum_{f=1}^M \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{W}_{n,d} \mathbf{W}_{n,d'} \Re \left\{ e^{j2\pi \frac{(f-1)(\tau_{n,d'} - \tau_{n,d})}{M}} \tilde{\mathbf{S}}_{f,d'}^* \tilde{\mathbf{S}}_{f,d} \right\} \\
&= \mathbf{W}_{n,d} \mathbf{W}_{n,d'} \left(\frac{2\pi}{M} \right)^2 \sum_{f=1}^M (f-1)^2 \Re \left\{ e^{j2\pi \frac{(f-1)(\tau_{n,d'} - \tau_{n,d})}{M}} \tilde{\mathbf{S}}_{f,d'}^* \tilde{\mathbf{S}}_{f,d} \right\}.
\end{aligned}$$

- Case 3: $n' \neq n$. Then the Hessian is equal to 0.

A.5 Derivative of $f_{\mathbf{V}}$ with respect to matrix $\boldsymbol{\tau}$

First, we analyse function $f_{\mathbf{V}}$ as

$$\begin{aligned}
f_{\mathbf{V}}(\mathbf{A}, \mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau}) &= \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N \sum_{f=1}^M \left\| \tilde{\mathbf{v}}_{k,n,f} - \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right\|^2 \\
&= \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N \sum_{f=1}^M \left(\tilde{\mathbf{v}}_{k,n,f} - \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right)^* \left(\tilde{\mathbf{v}}_{k,n,f} - \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right) \\
&= \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N \sum_{f=1}^M \left(\tilde{\mathbf{v}}_{k,n,f}^* - \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \right) \left(\tilde{\mathbf{v}}_{k,n,f} - \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right) \\
&= \frac{1}{2} \sum_{k=1}^K \sum_{n=1}^N \sum_{f=1}^M \tilde{\mathbf{v}}_{k,n,f}^* \tilde{\mathbf{v}}_{k,n,f} - \tilde{\mathbf{v}}_{k,n,f}^* \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} - \tilde{\mathbf{v}}_{k,n,f} \sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \\
&\quad + \left(\sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \right) \left(\sum_{d=1}^D \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right).
\end{aligned}$$

Then, the first derivative of $f_{\mathbf{V}}$ with respect to $\boldsymbol{\tau}$ is equal to

$$\begin{aligned}
\frac{\partial f_{\mathbf{V}}(\mathbf{A}, \mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau})}{\partial \boldsymbol{\tau}_{n,d}} &= \frac{1}{2} \sum_{k=1}^K \sum_{f=1}^M - \left(-j2\pi \frac{(f-1)}{M} \right) \tilde{\mathbf{v}}_{k,n,f}^* \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \\
&\quad - \left(j2\pi \frac{(f-1)}{M} \right) \tilde{\mathbf{v}}_{k,n,f} \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \\
&\quad + \left(\sum_{d'=1}^D \mathbf{A}_{k,d'} \mathbf{W}_{n,d'} e^{j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'}^* \right) \left(\left(-j2\pi \frac{(f-1)}{M} \right) \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right) \\
&\quad + \left(\left(j2\pi \frac{(f-1)}{M} \right) \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \right) \left(\sum_{d'=1}^D \mathbf{A}_{k,d'} \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \\
&= \frac{1}{2} \sum_{k=1}^K \sum_{f=1}^M \left(-j2\pi \frac{(f-1)}{M} \right) \mathbf{A}_{k,d} \mathbf{W}_{n,d} \left(\tilde{\mathbf{v}}_{k,n,f} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* - \tilde{\mathbf{v}}_{k,n,f}^* e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right) \\
&\quad + \left(-j2\pi \frac{(f-1)}{M} \right) \mathbf{A}_{k,d} \mathbf{W}_{n,d} \left(\sum_{d'=1}^D \mathbf{A}_{k,d'} \mathbf{W}_{n,d'} e^{j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'}^* \right) \left(e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \right) \\
&\quad - \left(-j2\pi \frac{(f-1)}{M} \right) \mathbf{A}_{k,d} \mathbf{W}_{n,d} \left(e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \right) \left(\sum_{d'=1}^D \mathbf{A}_{k,d'} \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \\
&= \sum_{k=1}^K \sum_{f=1}^M \left(2\pi \frac{(f-1)}{M} \right) \mathbf{A}_{k,d} \mathbf{W}_{n,d} \\
&\quad \Im \left\{ \tilde{\mathbf{v}}_{k,n,f} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* - \left(\sum_{d'=1}^D \mathbf{A}_{k,d'} \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \left(e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \right) \right\} \\
&= \sum_{k=1}^K \sum_{f=1}^M \left(2\pi \frac{(f-1)}{M} \right) \mathbf{A}_{k,d} \mathbf{W}_{n,d} \Im \left\{ \left(\tilde{\mathbf{v}}_{k,n,f} - \sum_{d'=1}^D \mathbf{A}_{k,d'} \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \left(e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \right) \right\} \\
&= 2\pi \frac{\mathbf{W}_{n,d}}{M} \sum_{f=1}^M (f-1) \Im \left\{ e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \sum_{k=1}^K \mathbf{A}_{k,d} \left(\tilde{\mathbf{v}}_{k,n,f} - \sum_{d'=1}^D \mathbf{A}_{k,d'} \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \right\}.
\end{aligned}$$

A.6 Hessian of $f_{\mathbf{V}}$ with respect to matrix $\boldsymbol{\tau}$

The Hessian of function $f_{\mathbf{V}}$ with respect to $\boldsymbol{\tau}$ is calculated for three cases.

- Case 1: $n' = n$ and $d' = d$. Then the Hessian takes the following form

$$\begin{aligned}
& \frac{\partial^2 f_{\mathbf{V}}(\mathbf{A}, \mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau})}{\partial^2 \boldsymbol{\tau}_{n,d}} = \\
& \frac{1}{2} \sum_{k=1}^K \sum_{f=1}^M \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{A}_{k,d} \mathbf{W}_{n,d} \tilde{\mathbf{V}}_{k,n,f} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \\
& + \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{A}_{k,d} \mathbf{W}_{n,d} \tilde{\mathbf{V}}_{k,n,f}^* e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \\
& - \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \left(\sum_{d'=1}^D \mathbf{A}_{k,d'} \mathbf{W}_{n,d'} e^{j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'}^* \right) \\
& + \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{A}_{k,d}^2 \mathbf{W}_{n,d}^2 \left| \tilde{\mathbf{S}}_{f,d} \right|^2 \\
& - \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \left(\sum_{d'=1}^D \mathbf{A}_{k,d'} \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \\
& + \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{A}_{k,d}^2 \mathbf{W}_{n,d}^2 \left| \tilde{\mathbf{S}}_{f,d} \right|^2 \\
& = \frac{1}{2} \sum_{k=1}^K \sum_{f=1}^M \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \left(\tilde{\mathbf{V}}_{k,n,f} - \sum_{d'=1}^D \mathbf{A}_{k,d'} \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \\
& + \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{A}_{k,d} \mathbf{W}_{n,d} e^{-j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d} \left(\tilde{\mathbf{V}}_{k,n,f}^* - \sum_{d'=1}^D \mathbf{A}_{k,d'} \mathbf{W}_{n,d'} e^{j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'}^* \right) \\
& + 2 \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{A}_{k,d}^2 \mathbf{W}_{n,d}^2 \left| \tilde{\mathbf{S}}_{f,d} \right|^2 \\
& = \sum_{k=1}^K \sum_{f=1}^M \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{A}_{k,d} \mathbf{W}_{n,d} \\
& \quad \left(\Re \left\{ e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \left(\tilde{\mathbf{V}}_{k,n,f} - \sum_{d'=1}^D \mathbf{A}_{k,d'} \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \right\} + \mathbf{A}_{k,d} \mathbf{W}_{n,d} \left| \tilde{\mathbf{S}}_{f,d} \right|^2 \right) \\
& = \mathbf{W}_{n,d} \left(\frac{2\pi}{M} \right)^2 \sum_{f=1}^M (f-1)^2 \\
& \quad \left(\Re \left\{ e^{j2\pi \frac{(f-1)\tau_{n,d}}{M}} \tilde{\mathbf{S}}_{f,d}^* \sum_{k=1}^K \mathbf{A}_{k,d} \left(\tilde{\mathbf{V}}_{k,n,f} - \sum_{d'=1}^D \mathbf{A}_{k,d'} \mathbf{W}_{n,d'} e^{-j2\pi \frac{(f-1)\tau_{n,d'}}{M}} \tilde{\mathbf{S}}_{f,d'} \right) \right\} + \mathbf{W}_{n,d} \left| \tilde{\mathbf{S}}_{f,d} \right|^2 \left\| \mathbf{A}_{:,d} \right\|_2^2 \right).
\end{aligned}$$

- Case 2: $n' = n$ and $d' \neq d$. Then the Hessian takes the following form

$$\begin{aligned}
\frac{\partial^2 f_{\mathbf{V}}(\mathbf{A}, \mathbf{W}, \tilde{\mathbf{S}}, \boldsymbol{\tau})}{\partial \boldsymbol{\tau}_{n,d} \partial \boldsymbol{\tau}_{n,d'}} &= \frac{1}{2} \sum_{k=1}^K \sum_{f=1}^M \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{A}_{k,d'} \mathbf{A}_{k,d} \mathbf{W}_{n,d} \mathbf{W}_{n,d'} e^{j2\pi \frac{(f-1)(\boldsymbol{\tau}_{n,d'} - \boldsymbol{\tau}_{n,d})}{M}} \tilde{\mathbf{S}}_{f,d'}^* \tilde{\mathbf{S}}_{f,d} \\
&\quad + \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{A}_{k,d} \mathbf{A}_{k,d'} \mathbf{W}_{n,d'} \mathbf{W}_{n,d} e^{j2\pi \frac{(f-1)(\boldsymbol{\tau}_{n,d} - \boldsymbol{\tau}_{n,d'})}{M}} \tilde{\mathbf{S}}_{f,d}^* \tilde{\mathbf{S}}_{f,d'} \\
&= \sum_{k=1}^K \sum_{f=1}^M \left(2\pi \frac{(f-1)}{M} \right)^2 \mathbf{A}_{k,d'} \mathbf{A}_{k,d} \mathbf{W}_{n,d} \mathbf{W}_{n,d'} \Re \left\{ e^{j2\pi \frac{(f-1)(\boldsymbol{\tau}_{n,d'} - \boldsymbol{\tau}_{n,d})}{M}} \tilde{\mathbf{S}}_{f,d'}^* \tilde{\mathbf{S}}_{f,d} \right\} \\
&= \mathbf{W}_{n,d} \mathbf{W}_{n,d'} \left(\frac{2\pi}{M} \right)^2 \sum_{f=1}^M (f-1)^2 \Re \left\{ e^{j2\pi \frac{(f-1)(\boldsymbol{\tau}_{n,d'} - \boldsymbol{\tau}_{n,d})}{M}} \tilde{\mathbf{S}}_{f,d'}^* \tilde{\mathbf{S}}_{f,d} \right\} \sum_{k=1}^K \mathbf{A}_{k,d'} \mathbf{A}_{k,d} \\
&= \mathbf{W}_{n,d} \mathbf{W}_{n,d'} \left(\frac{2\pi}{M} \right)^2 \sum_{f=1}^M (f-1)^2 \Re \left\{ e^{j2\pi \frac{(f-1)(\boldsymbol{\tau}_{n,d'} - \boldsymbol{\tau}_{n,d})}{M}} \tilde{\mathbf{S}}_{f,d'}^* \tilde{\mathbf{S}}_{f,d} \right\} (\mathbf{A}_{:,d'}^T \mathbf{A}_{:,d}).
\end{aligned}$$

- Case 3: $n' \neq n$. Then the Hessian is equal to 0.

Bibliography

- [1] A. Rayshubskiy, T. J. Wojtasiewicz, C. B. Mikell, M. B. Bouchard, D. Timerman, B. E. Youngerman, R. A. McGovern, M. L. Otten, P. Canoll, G. M. McKhann, *et al.*, “Direct, intraoperative observation of ~ 0.1 hz hemodynamic oscillations in awake human cortex: implications for fmri,” *Neuroimage*, vol. 87, pp. 323–331, 2014.
- [2] F. G. Ashby, *An introduction to fMRI*. Springer, 2015.
- [3] P. M. Kroonenberg, *Applied multiway data analysis*, vol. 702. John Wiley & Sons, 2008.
- [4] A. Cichocki, R. Zdunek, A. H. Phan, and S.-i. Amari, *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*. John Wiley & Sons, 2009.
- [5] T. G. Kolda and B. W. Bader, “Tensor decompositions and applications,” *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.
- [6] J. Gorski, F. Pfeuffer, and K. Klamroth, “Biconvex sets and optimization with biconvex functions: a survey and extensions,” *Mathematical Methods of Operations Research*, vol. 66, no. 3, pp. 373–407, 2007.
- [7] L. Helmut, *Handbook of matrices*. New York: John Wiley and Sons, 1996.
- [8] D. D. Lee and H. S. Seung, “Algorithms for non-negative matrix factorization,” in *Advances in neural information processing systems*, pp. 556–562, 2001.
- [9] S. V. Vaseghi, *Advanced signal processing and digital noise reduction*. Springer-Verlag, 2013.
- [10] M. Morup, K. H. Madsen, and L. K. Hansen, “Shifted non-negative matrix factorization,” in *2007 IEEE Workshop on Machine Learning for Signal Processing*, pp. 139–144, IEEE, 2007.

-
- [11] P. O. Hoyer, “Non-negative matrix factorization with sparseness constraints,” *Journal of machine learning research*, vol. 5, no. Nov, pp. 1457–1469, 2004.
 - [12] V. P. Pauca, J. Piper, and R. J. Plemmons, “Nonnegative matrix factorization for spectral data analysis,” *Linear algebra and its applications*, vol. 416, no. 1, pp. 29–47, 2006.
 - [13] P. Smaragdis, “Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs,” in *International Conference on Independent Component Analysis and Signal Separation*, pp. 494–499, Springer, 2004.
 - [14] W. Wang, A. Cichocki, and J. Chambers, “A multiplicative algorithm for convolutive non-negative matrix factorization based on squared euclidean distance,” 2009.
 - [15] P. D. O’grady and B. A. Pearlmutter, “Convolutive non-negative matrix factorisation with a sparseness constraint,” in *2006 16th IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing*, pp. 427–432, IEEE, 2006.
 - [16] C. Ding, T. Li, W. Peng, and H. Park, “Orthogonal nonnegative matrix t-factorizations for clustering,” in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 126–135, ACM, 2006.
 - [17] C. Ding and X. He, “K-means clustering via principal component analysis,” in *Proceedings of the twenty-first international conference on Machine learning*, p. 29, ACM, 2004.
 - [18] H. Zha, X. He, C. Ding, M. Gu, and H. D. Simon, “Spectral relaxation for k-means clustering,” in *Advances in neural information processing systems*, pp. 1057–1064, 2002.
 - [19] Z. Chen and A. Cichocki, “Nonnegative matrix factorization with temporal smoothness and/or spatial decorrelation constraints,” *Laboratory for Advanced Brain Signal Processing, RIKEN, Tech. Rep*, vol. 68, 2005.
 - [20] L.-H. Lim and P. Comon, “Nonnegative approximations of nonnegative tensors,” *Journal of chemometrics*, vol. 23, no. 7-8, pp. 432–441, 2009.
 - [21] A. Stegeman, “Degeneracy in candecomp/parafac and indscal explained for several three-sliced arrays with a two-valued typical rank,” *Psychometrika*, vol. 72, no. 4, pp. 601–619, 2007.

-
- [22] N. D. Sidiropoulos and R. Bro, “On the uniqueness of multilinear decomposition of n-way arrays,” *Journal of chemometrics*, vol. 14, no. 3, pp. 229–239, 2000.
 - [23] M. Mørup, L. K. Hansen, S. M. Arnfred, L.-H. Lim, and K. H. Madsen, “Shift-invariant multilinear decomposition of neuroimaging data,” *NeuroImage*, vol. 42, no. 4, pp. 1439–1450, 2008.
 - [24] G. Favier and A. L. de Almeida, “Overview of constrained parafac models,” *EURASIP Journal on Advances in Signal Processing*, vol. 2014, no. 1, p. 142, 2014.
 - [25] R. A. Harshman, “Determination and proof of minimum uniqueness conditions for parafac1,” *UCLA Working Papers in phonetics*, vol. 22, no. 111-117, p. 3, 1972.
 - [26] J. Mocks, “Topographic components model for event-related potentials and some biophysical considerations,” *IEEE transactions on biomedical engineering*, vol. 6, no. 35, pp. 482–484, 1988.
 - [27] W. P. Krijnen, T. K. Dijkstra, and A. Stegeman, “On the non-existence of optimal solutions and the occurrence of degeneracy in the candecomp/parafac model,” *Psychometrika*, vol. 73, no. 3, pp. 431–439, 2008.
 - [28] M. Sørensen, L. D. Lathauwer, P. Comon, S. Icart, and L. Deneire, “Canonical polyadic decomposition with a columnwise orthonormal factor matrix,” *SIAM Journal on Matrix Analysis and Applications*, vol. 33, no. 4, pp. 1190–1213, 2012.
 - [29] R. Bro, “Parafac. tutorial and applications,” *Chemometrics and intelligent laboratory systems*, vol. 38, no. 2, pp. 149–171, 1997.
 - [30] S. Hong and R. A. Harshman, “Shifted factor analysis part iii: N-way generalization and application,” *Journal of chemometrics*, vol. 17, no. 7, pp. 389–399, 2003.
 - [31] R. A. Poldrack, J. A. Mumford, and T. E. Nichols, *Handbook of functional MRI data analysis*. Cambridge University Press, 2011.
 - [32] S. A. Huettel, A. W. Song, and G. McCarthy, *Functional magnetic resonance imaging*, vol. 1. Sinauer Associates Sunderland, 2004.
 - [33] L. Pauling and C. D. Coryell, “The magnetic properties and structure of hemoglobin, oxyhemoglobin and carbonmonoxyhemoglobin,” *Proceedings of the National Academy of Sciences*, vol. 22, no. 4, pp. 210–216, 1936.

- [34] S. Ogawa, T.-M. Lee, A. R. Kay, and D. W. Tank, "Brain magnetic resonance imaging with contrast dependent on blood oxygenation," *Proceedings of the National Academy of Sciences*, vol. 87, no. 24, pp. 9868–9872, 1990.
- [35] S. Ogawa, T.-M. Lee, A. S. Nayak, and P. Glynn, "Oxygenation-sensitive contrast in magnetic resonance image of rodent brain at high magnetic fields," *Magnetic resonance in medicine*, vol. 14, no. 1, pp. 68–78, 1990.
- [36] G. M. Boynton, S. A. Engel, G. H. Glover, and D. J. Heeger, "Linear systems analysis of functional magnetic resonance imaging in human v1," *Journal of Neuroscience*, vol. 16, no. 13, pp. 4207–4221, 1996.
- [37] R. B. Buxton and L. R. Frank, "A model for the coupling between cerebral blood flow and oxygen metabolism during neural stimulation," *Journal of Cerebral Blood Flow & Metabolism*, vol. 17, no. 1, pp. 64–72, 1997.
- [38] A. L. Vazquez and D. C. Noll, "Nonlinear aspects of the bold response in functional mri," *NeuroImage*, vol. 7, no. 2, pp. 108–118, 1998.
- [39] W. D. Penny, K. J. Friston, J. T. Ashburner, S. J. Kiebel, and T. E. Nichols, *Statistical parametric mapping: the analysis of functional brain images*. Academic press, 2011.
- [40] M. Behroozi, M. R. Daliri, and H. Boyaci, "Statistical analysis methods for the fmri data," *Basic and Clinical Neuroscience*, vol. 2, no. 4, pp. 67–74, 2011.
- [41] K. J. Friston, A. P. Holmes, K. J. Worsley, J.-P. Poline, C. D. Frith, and R. S. Frackowiak, "Statistical parametric maps in functional imaging: a general linear approach," *Human brain mapping*, vol. 2, no. 4, pp. 189–210, 1994.
- [42] V. Calhoun, T. Adali, G. Pearlson, and J. Pekar, "Spatial and temporal independent component analysis of functional mri data containing a pair of task-related waveforms," *Human brain mapping*, vol. 13, no. 1, pp. 43–53, 2001.
- [43] E. Formisano, F. De Martino, and G. Valente, "Multivariate analysis of fmri time series: classification and regression of brain responses using machine learning," *Magnetic resonance imaging*, vol. 26, no. 7, pp. 921–934, 2008.
- [44] A. H. Andersen and W. S. Rayens, "Structure-seeking multilinear methods for the analysis of fmri data," *NeuroImage*, vol. 22, no. 2, pp. 728–739, 2004.

-
- [45] C. F. Beckmann and S. M. Smith, "Tensorial extensions of independent component analysis for multisubject fmri analysis," *Neuroimage*, vol. 25, no. 1, pp. 294–311, 2005.
- [46] R. B. Buxton, E. C. Wong, and L. R. Frank, "Dynamics of blood flow and oxygenation changes during brain activation: the balloon model," *Magnetic resonance in medicine*, vol. 39, no. 6, pp. 855–864, 1998.
- [47] M. I. Sereno, A. Dale, J. Reppas, K. Kwong, *et al.*, "Borders of multiple visual areas in humans revealed by functional magnetic resonance imaging," *Science*, vol. 268, no. 5212, p. 889, 1995.
- [48] B. Sen and K. K. Parhi, "Extraction of common task signals and spatial maps from group fmri using a parafac-based tensor decomposition technique," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pp. 1113–1117, IEEE, 2017.
- [49] V. D. Calhoun, T. Adali, V. McGinty, J. J. Pekar, T. Watson, and G. Pearlson, "fmri activation in a visual-perception task: network of areas detected using the general linear model and independent components analysis," *NeuroImage*, vol. 14, no. 5, pp. 1080–1088, 2001.
- [50] M. Mørup, L. K. Hansen, and K. H. 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)*, pp. 439–443, IEEE, 2011.
- [51] K. B. Petersen, M. S. Pedersen, *et al.*, "The matrix cookbook," *Technical University of Denmark*, vol. 7, p. 15, 2008.