Sparse Bayesian Learning for EEG Source Recovery

Trine Nyholm Kragh & Laura Nyrup Mogensen
Mathematical Engineering, MATTEK

Master's Thesis

Mathematical Engineering
Aalborg University
http://www.aau.dk

AALBORG UNIVERSITY
STUDENT REPORT

Title:
Sparse Bayesian Learning for EEG Source Recovery

Theme:
Mathematical modeling of EEG measurements for blind source recovery based on existing methods

Project Period:
Fall Semester 2019
Spring Semester 2020

Project Group:
Mattek10b

Participant(s):
Trine Nyholm Kragh
Laura Nyrup Mogensen

Supervisor(s):
Jan Østergaard
Rasmus Waagepetersen

Copies: 5

Page Numbers: 109

Date of Completion:
June 2, 2020

Abstract:

This thesis treats the problem of recovering original brain source signals from low density EEG scalp measurements. Based on state of the art methods, an algorithm is proposed to reproduce the current results. The algorithm leverage a covariance-domain dictionary learning (Cov-DL) method and a multiple sparse Bayesian learning (M-SBL) method. The proposed application of Cov-DL did not succeed. Thus, an alternative solution was proposed. The final algorithm was tested on EEG and compared to solutions obtained by independent component analysis of high density EEG. A frequency analysis was performed comparing raw EEG and the recovered sources. With respect to practical use, an estimation of the unknown number of active source signals was proposed. It is concluded that the proposed algorithm is unable to reproduce the state of the art results. However, the M-SBL method alone is successful and a potential is seen for an estimation of the number of active sources, from M-SBL.

Preface

This master thesis has been written by Trine Nyholm Kragh and Laura Nyrup Mogensen, 4th semester students of the Master programme in Mathematical Engineering at Aalborg University. The students want to thank their supervisors Jan Østergaard from Department of Electronical Systems and Rasmus Waagepetersen from Department of Mathematical Sciences for their guidance and help to conduct this study. Furthermore, a thank you goes to PhD student Payam Shahsavari Baboukani for sharing the real EEG scalp data base and corresponding guidance.

The theme of this master thesis is mathematical modeling of EEG measurements for source recovery based on existing method, aiming to support existing the results. As for prerequisites, the reader is expected to be familiar with linear algebra, optimization theory and probability theory. Citations are provided in the form [citation number] or [citation number, page number] especially for books, with every source having a unique citation number to be found the bibliography. In appendix A, B and C supplementary theory for the presented methods can be found. In appendix D a list and outline of the scripts developed during this thesis can be found. The scripts are written in Python 3.6 and can be downloaded through GitHub link XX.

Aalborg University, June 2, 2020

Trine Nyholm Kragh
<trijen15@student.aau.dk>

Laura Nyrup Mogensen
<lmogen15@student.aau.dk>

Danish Summary

I dette kandidatspeciale undersøges det hvordan man kan genskabe de originale kilder til den hjerne aktivitet, der måles via Elektroencefalografi (EEG). Med udgangspunkt i lineær algebra kan EEG-målinger modelleres som et lineært system $\mathbf{Y} = \mathbf{AX}$. Her udgør $\mathbf{X}$ de original kilder, $\mathbf{A}$ udgør transformationen af $\mathbf{X}$ til de observerede EEG-målinger, som $\mathbf{Y}$ udgør. At løse systemet med hensyn til de original kilder $\mathbf{X}$ omtales som det inverse EEG-problem. Dette omfatter en estimering af både $\mathbf{A}$ og $\mathbf{X}$. I dette speciale ønskes det at løse det inverse EEG-problem i det specifikke tilfælde hvor der er færre sensorer end der er kilder. Der tages udgangspunkt i to "state of the art" metoder, covariance-domain dictionary learning (Cov-DL) og multiple sparse Bayesian learning (M-SBL), til at løse dette problem. Her er det primære formål er at eftervise de resultater som tidligere er opnået ved anvendelse af disse metoder. Herigennem sættes der fokus på reproducerbarhed af de udvalgte videnskabelige artikler. De to metoder implementeres i én algoritme, som omtales "The main algorithm". Metoden Cov-DL anvendes til at finde transformationsmatricen og M-SBL benytter sig af den fundne transformationsmatrix til at finde matricen med kilder. Sekundært sættes algoritmen i et praktisk perspektiv, hvor det undersøges hvorvidt det er muligt at estimere antallet af aktive kilder i hjernen. Under implementeringen af algoritmen bliver de enkelte metoder testet og analyseret på simuleret data. Testene evalueres ved at sammenligne de fundne estimater med de sande værdier, hertil anvendes mean-square-error (MSE). For Cov-DL lykkes det ikke at estimere transformationsmatricen på samme vis som angivet i den anvendte kilde. Den anden metode, M-SBL, ses at være succesfuld når den sande transformationsmatrice benyttes. Herfra kan det konkluderes, at den tilsvarende videnskabelige artikel tilvejebragte en tilstrækkelig grad af reproducerbarhed.

Med henblik på at anvende den samlede algoritmen på rigtige EEG-målinger vælges det at erstatte Cov-DL med en fast transformationsmatrix, som bestemmes ud fra empiriske tests. Med udgangspunkt i testresultater og det faktum at transformationsmatricen er valgt med en vis grad af tilfældighed, så forventes det ikke at der opnås en tilstrækkelig genskabelse af de original kilder fra EEG-målinger. Med det formål at evaluere algoritmens performance på rigtige EEG-målinger introduceres independent component analysis (ICA) og dens estimater. Dette er en allerede ek-

sisterende og succesfuld metode for det inverse EEG-problem – for systemer med et lige antal sensorer og kilder. Estimaterne fra ICA sammenlignes med de tilsvarende estimater fra den samlede algoritme, hvor et varierende antal at sensorer er fjernet, for at tilnærme det ønskede system. Når ingen sensorer er fjernet, ses et snævert potentiale for et tilstrækkeligt estimat, men for systemer med flere kilder end sensorer fejler metoden, som det kunne forventes ved fastsat transformationsmatrix. Som en alternativ test udføres en frekvensanalyse der sammenligner de rå EEG-målinger med de fundne kilder. Resultatet af dette ændrer ikke på de tidligere konklusioner. Endeligt undersøges muligheden for at estimere antallet af aktive kilder i hjernen. Her ses et potentiale ved test på simuleret data, dog forringes denne performance når antallet af sensorer reduceres.

Contents

Todo list

Introduction

The topic of this thesis arises from the increasing use of electroencephalographic measurements for a wide range of scientific purposes, especially within the medical field. By sensors placed on the head, an electroencephalography captures a mixture of electric signals caused by activity within the brain. One essential issue concerning an electroencephalography is to recover the original source signals which were released inside the brain.

The need for source recovery is confirmed by studies showing how analysis performed on electroencephalographic measurements differs significantly from similar analysis performed directly on the original sources [16]. One area of application, where the use of the recovered source signals has shown potential, is the hearing aid industry. Here it is of special interest to recover the source signals from only few sensors, which potentially can be placed within a hearing aid.

Consider the issue of source recovery from a mathematical perspective. Here the electroencephalographic measurements can be modeled by a linear system of equations. From such model it is possible to recover a limited number of source signals under certain conditions. However, it is a general acknowledged issue that the true number of source signals inside the human brain is unknown. The task complexity of recovering the source signals from the linear system is increased in cases where the number of sources exceeds the number of sensors providing the measurements.

This thesis explores a state of the art mathematical method for source recovery, embracing the case of more sources than sensors. Overall this method, published in 2015, consist of two steps. That is receptively to recover the mixing process that the source signals have undergone and then recover the source signals. The two steps originate from two different approaches considering the mathematical orientation. The main goal of the thesis is to study the two methods with respect to proposing a united algorithm, to be applied on electroencephalographic measurements. The purpose is to support the current results of source recovery from electroencephalographic measurements of few sensors. Furthermore, the issue of the unknown number of active source signals is considered from a perspective of practical application.

The thesis consists of a motivational part, introducing electroencephalography

and the potential use within research. Existing literatures are examined, with respect to identification of state of the art approaches within source signal recovery. The motivational part is concluded by the problem statement specifying the objective of the thesis. Next is the theoretical part. The system model is specified and the solution approach is presented based on existing methods. This is followed by an extensive study of the required theory. The practical aspect of the thesis includes an implementation of the proposed solution to be tested on both synthetic data and new electroencephalographic measurements. Finally, discussion and conclusion upon the achieved results are presented followed by a consideration upon further studies.

Chapter 1

Motivation

This chapter accounts for the motivation behind source signal recovery from Electroencephalography (EEG) measurements. The concept of EEG is introduced along with current applications. The potential and importance of source recovery are considered and is related to the hearing aid industry. A commonly applied mathematical model for EEG measurements is presented. Currently applied methods for source recovery are considered leading to a presentation of the current state of the art methods which succeed to overcome the limitations of previous methods. Lastly, the objective of this thesis is specified.

1.1 Introduction to EEG Measurements

EEG is an imaging technique used within the medical field. EEG measures electric signals on the scalp, caused by brain activity. The human central nerve system consists of various nerve cells connecting the neurons within the brain. Nerve cells respond to certain stimuli, for instance a physical stimulus, and transmit informations between neurons. Generally speaking these activities induce local currents that are transferred throughout the nerve system. Several nearby simultaneous activations result in local potential fields, each referred to as one source signal [26]. EEG measurements are provided by a number of metal electrodes, referred to as sensors, carefully placed on the human scalp. Each sensor captures the present electrical signals over time. For the source signal to reach a sensor it has to penetrate the skull, skin and several other thin layers of biological tissue. This causes an unknown distortion and reduction of a signal. It is most likely that the measurements of one sensor are sums of multiple source signals from different areas of the brain. Furthermore, the range of a single sensor is not separated from the other sensors. Thus the same source signal can easily be measured by two or more sensors. The process of distortion and mixing of signals is called volume conduction [26][28]. The concept of volume conduction is sought illustrated on figure 1.1. From this it is clarified that

EEG measurements are a mixture of fluctuating electrical signals originating from brain activities. Due to this mixing and the nature of the source signals, the true number of sources are in general considered unknown [26]. Furthermore, EEG is a subject for interfering noise. Noise signals can occur in the measurements resulting from physical movement of e.g. eyes and jawbone [30].

The source signals are classified within four groups according to the dominant frequency. The delta wave ($0.5 - 4$ Hz) is observed from infants and sleeping adults, the theta wave ($4 - 8$ Hz) is observed from children and sleeping adults, the alpha wave ($8 - 13$ Hz) is the most extensive studied brain rhythm, which is induced by an adult laying down with closed eyes. Lastly, the beta wave ($13 - 30$ Hz) is considered the normal brain wave for adults, associated with active thinking, active attention or solving concrete problems [26]. An example of signals within the four categories is illustrated by figure 1.2.

Generally, the distribution of EEG measurements of multiple sensors is considered multivariant Gaussian [26, p. 50]. Though the mean and covariance properties generally changes over time. Therefore, EEG measurements are considered quasi-stationary i.e. stationary only within small intervals. This motivates the need for segmentation of the EEG measurements to achieve signals with similar characteristics.

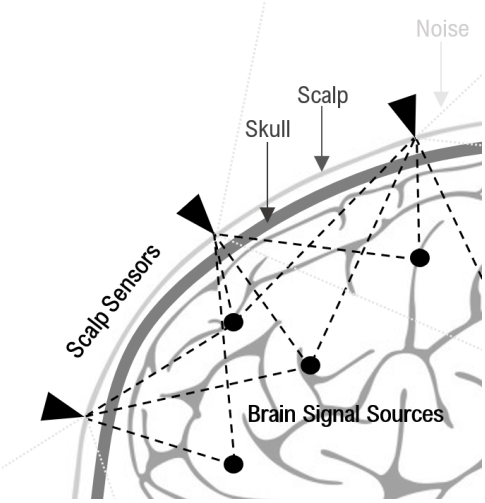

Figure 1.1: Illustration of volume conduction.

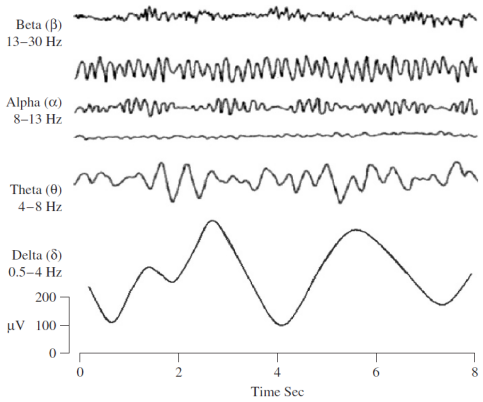

Figure 1.2: Example of time dependent signals within the four defined categories [26].

1.1.1 Application

EEG performed on humans and animals have a great number of applications within both clinical and research purposes. Examples of clinical applications covers diagnosis and management of neurological disorders like epilepsy, and monitor alertness

regarding coma or brain death. EEG capitalizes on the procedure being non-invasive and fast. Neural activity can be measured within fractions of a second after a stimulus has been provided. These advantages contribute to the wide range of applications within research of the neural processes involved in or resulting from actions, emotions or cognition. Today such neural research are used in many different fields [30, p. 4]. The hearing aid industry is one example where this research is highly prioritized. At Eriksholm research center, which is a part of the hearing aid manufacturer Oticon, cognitive hearing science is a research area within fast development [29]. One main purpose at Eriksholm is to make it possible for a hearing aid to identify the user-intended sound source from real time EEG measurements and thereby exclude noise from elsewhere [2][9]. It is essentially the well-known but unsolved cocktail problem which is sought improved by use of EEG. This is where EEG and occasionally so called in-ear EEG is interesting. In conjunction with the technology of beamforming, it is possible for a hearing aid to receive only signals from a specific direction.

Over the past two decades functional integration has become an area of interest regarding EEG research [15]. Within neurobiology functional integration refers to the study of the correlation among activities in different regions of the brain. In other words, how do different parts of the brain work together to process information and conduct a response [16]. For this purpose recovery of the original source signals, from EEG measurements, is of interest. An article from 2016 [28] points out the importance of performing analysis regarding functional integration on the sources, rather than directly on the EEG measurements. This relation is referred to source level versus EEG level. It is argued through experiments that analysis at EEG level does not allow interpretations about the interaction between sources. This emphasizes a potential for improving results within a wide range of EEG research, if the original source signals can be recovered from an EEG measurement.

1.1.2 Modeling

Consider the issue of recovering the source signals from EEG measurements. A known approach is to model the observed data by a linear system

$$\mathbf{y} = \mathbf{A}\mathbf{x}.$$

Let the vector $\mathbf{y} \in \mathbb{R}^M$ be the EEG measurements of one time sample containing M sensor measurements. Let $\mathbf{x} \in \mathbb{R}^N$ be the corresponding N sources within the brain. Non-zero entries of $\mathbf{x}$ represent the active sources at the time of the measurement. Then the matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ represents the linear transformation from $\mathbb{R}^N$ to $\mathbb{R}^M$. $\mathbf{A}$ will be referred to as the mixing matrix as it resembles the volume conduction. The i-th column of $\mathbf{A}$ represents the relative weights from the i-th source to every sensor [6]. Representing one time sample the linear system is in general referred to as a single measurement vector model. Only the measurement vector $\mathbf{y}$ is known, hence

it is impossible to solve the linear system with respect to $\mathbf{x}$ using basic linear algebra. The task, in this case, is to recover first $\mathbf{A}$ and then $\mathbf{x}$, given the measurement vector $\mathbf{y}$. This problem is referred to as the inverse EEG problem. Recovering $\mathbf{x}$ by some estimate $\hat{\mathbf{x}}$ is referred to as source identification and localization. Identification is to estimate the signal of each active source. Localization is to place each active source signal at the right position within the source vector of dimension N, where N is the maximum number of sources.

1.1.3 Solution Method

Independent Component Analysis (ICA) is one commonly applied method to solve the inverse EEG problem [21][20]. ICA is a technique to find the matrix $\mathbf{A}$ such that the rows of $\mathbf{x}$ is statistically independent. Thus statistical independence between the sources is a necessary assumption. With respect to the nature of EEG measurements, this assumption is considered valid [20, p. 3]. Application of ICA has shown great results regarding source recovery from EEG measurements. However, a significant flaw to this method is that the EEG measurements are only separated into the number of sources equal to or less than the number of sensors [4]. Meaning that the inverse EEG problem can not be solved in the case where the maximum number of sources N exceeds the number of sensors M – the model forms an under-determined system. Such limitation undermines the reliability and usability of ICA, as the number of active sources easily exceeds the number of sensors [6]. This is especially a drawback when low-density EEG is considered. Low-density EEG measurements are collected from equipment with less than 32 sensors, increasing the changes for the number of sources to exceed the number of sensors. However, the improved capabilities of low-density EEG devices are desirable due to their relative low cost, mobility and ease to use. Especially within the hearing aid industry, as mentioned earlier, where low-density EEG equipment can be combined with a hearing aid.

This argues the importance of considering the inverse problem of EEG in the under-determined case where $M < N$. In the following section existing work considering the under-determined inverse EEG problem is investigated further.

1.2 Related Work and Our Objective

As mentioned above ICA is a solid method for source identification in the case where separation into a number of sources equal to the number of sensors is adequate. The issue occurs in cases where the number of sources N exceeds the number of sensors M. To overcome this issue, an extension of ICA was suggested, referred to as the ICA mixture model [4]. Instead of identifying one mixing matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ from an under-determined system this approach learns N_{model} different mixing matrices $\mathbf{A}_i \in \mathbb{R}^{M \times M}$, to make computations more tractable. This method was further adapted

into the Adaptive Mixture ICA (AMICA) which has showed successful results regarding identification of more sources than sensors [24]. However, these successful results rely on the assumption that no more than M out of N possible sources is simultaneously active. That is explicit that the source vector of dimension N has at most M non-zero entries. This assumption is still an essential limitation to the framework, especially when considering low-density EEG. Other types of ICA algorithms for under-determined systems were proposed, without overcoming the limitation of jointly active sources exceeding the number of sensors.

In 2015 O. Balkan et. al. suggested a new approach targeting the identification of more active sources than sensors regarding EEG measurements. One method was proposed for learning $\mathbf{A}$ from $\mathbf{y}$ [4] and a different method was proposed for finding $\mathbf{x}$ given $\mathbf{y}$ and $\mathbf{A}$ [5].

To learn $\mathbf{A}$ the suggested method, referred to as Cov-DL, is a covariance-domain based dictionary learning algorithm. The method is based upon theory of dictionary learning and compressive sensing. Which dictates a framework for solving an under-determined system when $\mathbf{x}$ contains a sufficiently amount of zeros. This is similar to the constraint of the presented ICA methods. However, to overcome this, the point of Cov-DL is to transfer the EEG measurements into the covariance-domain. In the covariance-domain a higher dimensionality can be achieved compared to the original EEG sensor domain with dimension M. The transformation can be done under already discussed assumptions of linear mixing and uncorrelated sources which follows from the assumption of independence. As a result the theory of compressive sensing is found to apply to the covariance-domain, allowing to learn $\mathbf{A}$ by dictionary learning. Even in the case where the active sources exceed the number of sensors.

Thus, the Cov-DL method stands out from other straight forward dictionary learning methods as it does not relay on the sparsity of active sources. Where sparseness refers to the number of non-zero elements. This is an essential advantage when low-density EEG is considered. Cov-DL was tested and found to outperform AMICA [4]. As mentioned, the Cov-DL method only learns the mixing matrix $\mathbf{A}$, resembling the volume conduction.

For the purpose of recovering $\mathbf{x}$, from $\mathbf{y}$ and $\mathbf{A}$, a multiple measurement sparse Bayesian learning (M-SBL) method is proposed [5]. M-SBL is based on the concept of finding a set of non-zero indices of the source vector $\mathbf{x}$ which corresponds to finding the localization of sources. Followed by an identification of each source signal. The method builds upon the Bayesian statistic framework and it is targeting the case of more active sources than sensors. The method was proven to outperform the previously used algorithms, even when the defined recovery conditions regarding the found mixing matrix $\mathbf{A}$ was not fulfilled [5].

One drawback, which is not fully covered in the referred literature, is the two methods rely on the number of active sources being known. In practise this is not the case. Hence, an estimation of the number of active sources has to be considered

for the methods to be useful in practice.

The two state of the art methods resulting in source recovery will make the foundation of this thesis. Our aim is to investigate and fully understand the two methods in order to implement and test a joint algorithm. The recovering the original source signals $\mathbf{x}$ from the measurements $\mathbf{y}$, when the number of active sources exceeds the number of measurements. More specifically the goal is to support the current results, by applying the methods on different EEG measurements, through our own implementation of the methods into one algorithm. Secondary it is of interest to consider the practical application of the proposed algorithm, for instance within a hearing aid as described in section 1.1. As mentioned, the number of active sources is in general unknown in practice. Thus, an estimation of the number of active sources is of interest for practical use of the algorithm. For this we want to investigate whether it is possible to estimate the number of active sources based on the recovered source signals.

By figure 1.3 the presented problem of source signal recovery from EEG measurements is considered within the bigger context which has been discussed in this chapter. Leading from the possibilities of EEG measurements to the specific application within hearing aids, as mentions above.

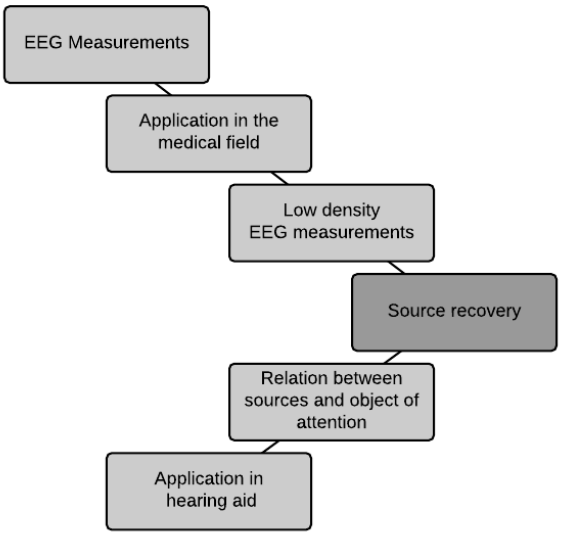

Figure 1.3: Visualization of the specified issue of source signal recovery relative to a bigger context.

Chapter 2

Problem Statement

EEG scalp measurements, a mixture of electrical signals originating from brain activities and noise, can be described as a linear system

$$\mathbf{y} = \mathbf{A}\mathbf{x}.$$

$\mathbf{y}$ is the EEG measurements from sensors placed on the scalp, $\mathbf{A}$ represent the mixing of the electrical signals, denoted as the mixing matrix. And $\mathbf{x}$ is the original electrical signals, denoted as sources. Only the EEG measurements $\mathbf{y}$ is known and it is of interest to recover first the mixing matrix $\mathbf{A}$ and hereby recover the original sources $\mathbf{x}$. The original sources have been shown significantly for practical use compared to the raw EEG measurements. Especially the case where the number of sources exceeds the number of sensors is of interest, resulting from the use low-density EEG equipment which is beneficial due to low cost and easy application. In the linear algebraic sense, this case creates an under-determined linear system which is difficult to solve. Two state of the art methods, targeting this specific case, are seen to successfully recover the sources. The covariance-domain dictionary learning (Cov-DL) method and the multiple sparse Bayesian learning (M-SBL) method. The Cov-DL method recovers the mixing matrix from the given measurements while the M-SBL method localizes and identifies the sources given the recovered mixing matrix and the measurements. However, one drawback of the methods is the required knowledges of the number of active sources as this is an unknown variable in practice.

This motivates the following problem statement.

Can state of the art results within source recovery of EEG measurements targeting the under-determined case be reproduced by an implementation of the methods tested on different data, and how can the potential of practical use be increased, with respect to the unknown number of active sources?

From the problem statement the following sub-questions are established for clarification.

- How is the Cov-DL method recreated to estimate a mixing matrix from the inverse EEG problem, in the under-determined case?
- How is the M-SBL method recreated to estimate a source signal matrix from the inverse EEG problem, in the under-determined case?
- How are the two methods combined into one algorithm, and does the results support the current state of the art results when applied to both synthetic and real EEG scalp measurements.
- How can the number of active sources be estimated, based only on the EEG scalp measurements?

Chapter 3

System Model

Through this chapter the model representing the EEG measurements is specified in details. Along with the model different terminologies are introduced and described for further use in this thesis. At last the solution approach for estimating the model parameters is described, setting the outline of the remaining chapters of the thesis.

3.1 System of Linear Equations

Let $\mathbf{y} \in \mathbb{R}^M$ be some vector. By basic linear algebra $\mathbf{y}$ can always be described as a linear combination of a coefficient matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ and some scalar vector $\mathbf{x} \in \mathbb{R}^N$ such that

$$\mathbf{y} = \mathbf{A}\mathbf{x}. \tag{3.1}$$

Let $\mathbf{y}$ and $\mathbf{A}$ be known. Then (3.1) makes a system of M linear equations with N unknowns, referred to as a linear system.

To solve the linear system (3.1) with respect to $\mathbf{x}$ one must look at the three different cases which can occur. The cases depend on the relation between the number of linear equations M and the number of unknowns N. For $M = N$, the system has one unique solution, provided that a solution exist. If the square coefficient matrix $\mathbf{A}$ has full rank the solution can be found simply by inverting $\mathbf{A}$.

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{y}.$$

For $M > N$ the system is over-determined, having more equations than unknown. There is not always a solution to an over-determined system. For $M < N$ the system is under-determined, having fewer equations than unknowns. There exists infinitely many solutions to an under-determined system, provided that one solution exist [13, p. ix].

Consider now $\mathbf{y} \in \mathbb{R}^M$ as the M observed EEG measurements provided by M sensors at time t. The linear system (3.1) is then considered as a single measurement vector (SMV) model. Modeling the EEG measurements by the SMV model embody the following interpretations, based on chapter 1. Remember that EEG measurements basically are a mixture of the original source signals, resulting from brain activity, affected by volume conduction and noise. The vector $\mathbf{x}$ is seen as the original source signals, with each entry representing the signal of one source. Thus, $\mathbf{x} \in \mathbb{R}^N$ is referred to as the source vector. N is considered the maximum number of sources, however zero entries may occur. The non-zero entries in $\mathbf{x}$ is referred to as the active sources at time t, while a zero entry corresponds to a non-active source. The coefficient matrix $\mathbf{A}$, referred to as the mixing matrix, models the volume conduction and noise by mapping the source vector from $\mathbb{R}^N$ to $\mathbb{R}^M$.

3.2 Multiple Measurement Vector Model of EEG

In practice EEG measurements are sampled over time by a certain sample frequency. Thus multiple EEG measurement vectors are achieved. Let L represent the total number of samples. The SMV model is now expanded to include L measurement vectors and external noise:

$$\mathbf{Y} = \mathbf{AX} + \mathbf{E}. \tag{3.2}$$

$\mathbf{Y} \in \mathbb{R}^{M \times L}$ is the observed measurement matrix, $\mathbf{X} \in \mathbb{R}^{N \times L}$ is the source matrix, and $\mathbf{A} \in \mathbb{R}^{M \times N}$ is the mixing matrix. Furthermore, $\mathbf{E} \in \mathbb{R}^{M \times L}$ is an additional noise matrix, to be expected from physical measurement equipment. The model is now referred to as a multiple measurement vector (MMV) model. As for (3.1) the solution set of the linear system (3.2) depends on the relation between N and M [13, p. 42].

As specified in chapter 1 it is the case where the number of sources exceeds the number of sensors, $M < N$, which is of interest in this thesis.

3.2.1 Segmentation

In chapter 1 it is argued that EEG measurements are only stationary within small segments. Hence, the following segmentation is considered.

Let f be the sample frequency of the observed EEG measurements $\mathbf{Y}$. And let t be the length of an interval in seconds determining the duration of one segment. Choose t sufficiently small such that the assumption of stationarity can be justified. Finally let s be the segment index. As such the observed EEG measurement matrix $\mathbf{Y}$ can be divided into stationary segments $\mathbf{Y}_s \in \mathbb{R}^{M \times L_s}$, possibly overlapping, where $L_s = tf$ is the number of samples within one segment. For each segment the MMV model (3.2) holds and is rewritten into.

$$\mathbf{Y}_s = \mathbf{A}_s\mathbf{X}_s + \mathbf{E}_s, \quad \forall s. \tag{3.3}$$

Note that the mixing matrix $\mathbf{A}$ is not segmented in the same manner as $\mathbf{Y}$ and $\mathbf{X}$, as the size of $\mathbf{A}$ do not change relative to the number of segments. The matrix $\mathbf{A}_s \in \mathbb{R}^{M \times N}$ is merely the mixing matrix that corresponds to $\mathbf{X}_s$ and $\mathbf{Y}_s$.

Based on the assumption that each segment is stationary, it is assumed that each source signal remains either active or non-active throughout the segment. This implies specifically that each row in $\mathbf{X}_s$ is either non-zero or zero respectively.

In order to characterize the source matrix with respect to the number of non-zero rows, the term row sparseness is considered. Let the support $\text{supp}(\mathbf{X})$ denote the index set of the non-zero rows of $\mathbf{X}$. To count the non-zero rows of a matrix the ℓ_0-norm is defined

$$\|\mathbf{X}\|_0 := \text{card}(\text{supp}(\mathbf{X})),$$

where the function $\text{card}(\cdot)$ gives the cardinality of the input set. The segmented source matrix $\mathbf{X}_s$ is said to be p-sparse if it contains at most p non-zero rows:

$$\|\mathbf{X}_s\|_0 \leq p.$$

Now denote the number of active sources by k, then k is defined by the number of non-zero rows of the sources matrix

$$k := \|\mathbf{X}_s\|_0$$

where $k \leq N$.

3.3 Solution Method

A MMV model for EEG measurements is now established. From the model the aim is, for all segments s, to recover the source matrix $\mathbf{X}_s$, by an estimate $\hat{\mathbf{X}}_s$ given only $\mathbf{Y}_s$ As such the original source signals from the brain are recovered as intended by the problem statement. In this section the solution method is presented and discussed, based on the state of the art methods which were lightly presented in chapter 1. This will outline the remaining chapters of the thesis.

Due to the problem statement, the case of interest is when $M < N$, typically resulting from low-density EEG measurements. Thus, $\mathbf{X}_s$ has to be recovered from an under-determined linear system. Hence, the solution must be found in the infinite solution space provided that one solution exists, thus simple linear algebra can not be used. Alternatively, numerical methods can be considered. By mathematical optimization it is possible to restrict the solution by some constraint. Then find a unique optimal solution with respect to some cost function and the corresponding constraint. The theory of compressive sensing provides a framework for solving an under-determined system when $\mathbf{X}_s$ is known to have zero rows, thus being row sparse. More specifically a unique solution $\mathbf{X}_s$ can be found when $\mathbf{X}_s$ is M-sparse,

cf. theorem A.1.1 in appendix A.1. When the mixing matrix $\mathbf{A}_s$ is unknown, as in this current case, the concept of dictionary learning can be used to determine $\mathbf{A}_s$. Still under the assumption that $\mathbf{X}_s$ is M-sparse.

The assumption of $\mathbf{X}_s$ being M-sparse corresponds to the number of the active sources $k \leq M$. However, from chapter 1 it can not be justified to apply this assumption on low density EEG measurements. Hence, the theory of compressive sensing can not be applied directly on the established model, when $M < N$.

A method to overcome this limitation of compressive sensing, is the covariance domain dictionary learning (Cov-DL) method [4], introduced in chapter 1. The method leverages the increased dimensionality of the covariance domain in order to allow the theory of compressive sensing to apply to an under-determined system. Note that this method only applies to the process of learning $\mathbf{A}_s$, in the case where $\mathbf{X}_s$ is not M-sparse. Hence, a different approach is necessary to recover $\mathbf{X}_s$.

For recovering $\mathbf{X}_s$, given both $\mathbf{Y}_s$ and $\mathbf{A}_s$ where $M < N$ and $k \leq N$, the method multiple sparse Bayesian learning (M-SBL) [5], introduced in chapter 1, is considered. This method takes advantage of the Bayesian statistic framework. Here, an empirical Bayesian estimation of $\mathbf{X}_s$ is performed, based on a prior distribution of $\mathbf{X}_s$ being defined by a data-dependent hyperparameter.

Combining the two methods allows recovery of $\mathbf{A}_s$ and $\mathbf{X}_s$ given low-density EEG measurements $\mathbf{Y}_s$ [6]. In the following two chapters each method is studied extensively, with the purpose of proposing the main algorithm in chapter 6.

Chapter 4

Covariance-Domain Dictionary Learning

Through this chapter the method covariance-domain dictionary learning (Cov-DL) is presented in details. Along the presentation of the general method, necessary computational details are derived. The purpose is to recover the mixing matrix $\mathbf{A}_s$ from the segmented MMV model, derived in chapter 3. Especially for the under-determined case. In the context of compressive sensing, the mixing matrix $\mathbf{A}_s$ is referred to a the dictionary matrix. This corresponds to the mixing matrix being estimated as a dictionary matrix, through the process of dictionary learning. This will be elaborated further in the section concerning dictionary learning.

Cov-DL is a method proposed by O. Balkan [4], leveraging the increased dimensionality of the covariance-domain. The method has shown successful recovery of the mixing matrix $\mathbf{A}_s$. Even in the non-sparse, under-determined case with more active sources k than observed measurements M, $k \geq M$. In short the algorithm consists of three steps. First the EEG measurements are transformed onto the covariance-domain. Then, by the increased dimensionality of the covariance-domain, it is possible to learn the transformed mixing matrix of the covariance-domain. The transformed mixing matrix is denoted by $\mathbf{D}$, based on the theory of compressive sensing. Here two different cases will appear depending on the relation between the number of sources N and the found dimension of the covariance-domain, the number of measurements M. Lastly, an inverse transformation is performed on the learned matrix $\mathbf{D}$, in order to obtain the wanted estimate of the mixing matrix $\mathbf{A}_s$. An essential aspect of this method is the prior assumption that the sources within one segment are uncorrelated, that is the rows of $\mathbf{X}_s$ being mutually uncorrelated.

The section is inspired by the article [4] and chapter 3 in [6]. Selected general theory supporting essential parts of the method is elaborated in appendix A.

4.1 Covariance-Domain Representation

Consider a single sample vector $\mathbf{y}_j \in \mathbb{R}^M$, containing EEG measurements. The covariance of $\mathbf{y}_j$ is defined as

$$\mathbf{\Sigma}_{\mathbf{y}_j} = \mathbb{E}[(\mathbf{y}_j - \mathbb{E}[\mathbf{y}_j])(\mathbf{y}_j - \mathbb{E}[\mathbf{y}_j])^T],$$

where $\mathbb{E}[\cdot]$ is the expected value operator. Let $\mathbf{Y}_s = [\mathbf{y}_1, \ldots, \mathbf{y}_{L_s}]$ be the observed measurement matrix containing all samples of segment s. Furthermore, assume that all sample vectors $\mathbf{y}_j$ within one segment have zero mean and the same distribution. Then $\mathbf{Y}_s \in \mathbb{R}^{M \times L_s}$ is described in the covariance-domain by the sample covariance $\widehat{\mathbf{\Sigma}}$. The sample covariance is defined as the empirical covariance among the M measurements across the L_s samples. That is a $M \times M$ matrix $\widehat{\mathbf{\Sigma}}_{\mathbf{Y}_s} = [\sigma_{jk}]$ with entries

$$\sigma_{kj} = \frac{1}{L_s} \sum_{i=1}^{L_s} y_{ij} y_{ij}.$$

Using matrix notation the sample covariance of $\mathbf{Y}_s$ can be written as

$$\widehat{\mathbf{\Sigma}}_{\mathbf{Y}_s} = \frac{1}{L_s} \mathbf{Y}_s \mathbf{Y}_s^T.$$

Similar, the source matrix $\mathbf{X}_s$ can be described in the covariance-domain by the sample covariance matrix:

$$\widehat{\mathbf{\Sigma}}_{\mathbf{X}_s} = \frac{1}{L_s} \mathbf{X}_s \mathbf{X}_s^T = \mathbf{\Lambda}_s + \boldsymbol{\varepsilon}.$$

The second equality comes from the assumption of the sources within $\mathbf{X}_s$ being uncorrelated. By uncorrelated sources $\mathbf{X}_s$ the sample covariance matrix is assumed to be nearly diagonal. Thus it can be written as $\mathbf{\Lambda}_s + \boldsymbol{\varepsilon}$ where $\mathbf{\Lambda}_s$ is a diagonal matrix consisting of the diagonal entries of $\widehat{\mathbf{\Sigma}}_{\mathbf{X}_s}$ and $\boldsymbol{\varepsilon}_s$ is a non-diagonal matrix with entries close to zero representing the estimation error [4].

Each segment is now modeled in the covariance-domain:

$$\begin{aligned}
\widehat{\mathbf{\Sigma}}_{\mathbf{Y}_s} = \frac{1}{L_s} \mathbf{Y}_s \mathbf{Y}_s^T &= \frac{1}{L_s} (\mathbf{A}_s \mathbf{X}_s + \mathbf{E}_s)(\mathbf{A}_s \mathbf{X}_s + \mathbf{E}_s)^T \\
&= \frac{1}{L_s} (\mathbf{A}_s \mathbf{X}_s)(\mathbf{A}_s \mathbf{X}_s)^T + \frac{1}{L_s} \mathbf{E}_s \mathbf{E}_s^T + \frac{1}{L_s} \mathbf{E}_s (\mathbf{A}_s \mathbf{X}_s)^T + \frac{1}{L_s} \mathbf{A}_s \mathbf{X}_s \mathbf{E}_s^T \\
&= \frac{1}{L_s} \mathbf{A}_s \mathbf{X}_s \mathbf{X}_s^T \mathbf{A}_s^T + \frac{1}{L_s} \mathbf{E}_s \mathbf{E}_s^T + \frac{1}{L_s} \mathbf{E}_s \mathbf{X}_s^T \mathbf{A}_s^T + \frac{1}{L_s} \mathbf{A}_s \mathbf{X}_s \mathbf{E}_s^T \\
&= \mathbf{A}_s (\mathbf{\Lambda}_s + \boldsymbol{\varepsilon}_s) \mathbf{A}_s^T + \frac{1}{L_s} \mathbf{E}_s \mathbf{E}_s^T + \frac{1}{L_s} \mathbf{E}_s \mathbf{X}_s^T \mathbf{A}_s^T + \frac{1}{L_s} \mathbf{A}_s \mathbf{X}_s \mathbf{E}_s^T \\
&= \mathbf{A}_s \mathbf{\Lambda}_s \mathbf{A}_s^T + \mathbf{A}_s \boldsymbol{\varepsilon}_s \mathbf{A}_s^T + \frac{1}{L_s} \mathbf{E}_s \mathbf{E}_s^T + \frac{1}{L_s} \mathbf{E}_s \mathbf{X}_s^T \mathbf{A}_s^T + \frac{1}{L_s} \mathbf{A}_s \mathbf{X}_s \mathbf{E}_s^T \qquad (4.1) \\
&= \mathbf{A}_s \mathbf{\Lambda}_s \mathbf{A}_s^T + \widetilde{\mathbf{E}}_s \qquad (4.2)
\end{aligned}$$

From (4.1) to (4.2) all terms where noise, $\boldsymbol{\varepsilon}_s$ and $\mathbf{E}_s$, is included, are aggregated in a joint noise term $\widetilde{\mathbf{E}}_s$. Next, the expression (4.2) is rewritten through a vectorization. Because the covariance matrix $\widehat{\boldsymbol{\Sigma}}_{\mathbf{Y}_s}$ is symmetric it is sufficient to vectorize only the lower triangular part, including the diagonal. For this purpose the function $\mathrm{vec}(\cdot)$ is defined. $\mathrm{vec}(\cdot)$ map a symmetric $M \times M$ matrix into a vector of size $\widetilde{M}$ by row-wise vectorization of the lower triangular part. The increased dimension $\widetilde{M}$ becomes

$$\widetilde{M} := \frac{M(M+1)}{2}. \tag{4.3}$$

Furthermore, let $\mathrm{vec}^{-1} : \mathbb{R}^{\widetilde{M}} \to \mathbb{R}^{M \times M}$ be the inverse function for devectorization.

Let $\mathbf{a}_j$ be the j-th column of $\mathbf{A}_s$, then the matrix product in (4.2) can be written in sum form where $\boldsymbol{\Lambda}_{s_{jj}}$ is the jj-th entry of $\boldsymbol{\Lambda}_s$.

$$\widehat{\boldsymbol{\Sigma}}_{\mathbf{Y}_s} = \sum_{j=1}^{N} \mathbf{a}_j \boldsymbol{\Lambda}_{s_{jj}} \mathbf{a}_j^T + \widetilde{\mathbf{E}}_s, \quad \boldsymbol{\Lambda}_{s_{jj}} \tag{4.4}$$

Applying $\mathrm{vec}(\cdot)$ to (4.4) results in the following expression, which concludes the transformation of model (3.3) into the covariance-domain:

$$\begin{aligned} \mathrm{vec}\left(\widehat{\boldsymbol{\Sigma}}_{\mathbf{Y}_s}\right) &= \sum_{j=1}^{N} \mathrm{vec}\left(\mathbf{a}_j \mathbf{a}_j^T\right) \boldsymbol{\Lambda}_{s_{jj}} + \mathrm{vec}\left(\widetilde{\mathbf{E}}_s\right) \\ &= \sum_{j=1}^{N} \mathbf{d}_j \boldsymbol{\Lambda}_{s_{jj}} + \mathrm{vec}\left(\widetilde{\mathbf{E}}_s\right) \\ &= \mathbf{D}_s \boldsymbol{\delta}_s + \mathrm{vec}\left(\widetilde{\mathbf{E}}_s\right), \quad \forall s. \end{aligned} \tag{4.5}$$

Here $\boldsymbol{\delta}_s \in \mathbb{R}^N$ contains the diagonal entries of the source sample covariance matrix $\boldsymbol{\Lambda}_s$ and the matrix $\mathbf{D}_s \in \mathbb{R}^{\widetilde{M} \times N}$ consists of the columns $\mathbf{d}_j = \mathrm{vec}\left(\mathbf{a}_j \mathbf{a}_j^T\right)$. Note that $\mathbf{D}$ and $\boldsymbol{\delta}_s$ are unknown while $\mathrm{vec}\left(\widehat{\boldsymbol{\Sigma}}_{\mathbf{Y}_s}\right)$ is known from the observed measurements. By this transformation to the covariance-domain, one segment is now represented by s single measurement model with $\widetilde{M}$ "measurements".

It has been shown that this transformed model allows for identification of $k \leq \widetilde{M}$ active sources [23]. This is a much weaker sparsity constraint than the original sparsity constraint $k \leq M$. The purpose of the Cov-DL algorithm is to leverage this transformed model to find the dictionary $\mathbf{A}_s$ from $\mathbf{D}_s$. Still allowing for $k \leq \widetilde{M}$ active sources to be recovered. That is the number of active sources are allowed to exceed the number of sensors as intended.

4.2 Recovery of the Mixing Matrix

The goal is now to learn first $\mathbf{D}_s$ and then the associated mixing matrix $\mathbf{A}_s$. Two methods are considered relying on the relation between M and N. For now the noise vector is ignored.

4.2.1 Under-determined System

When $\widetilde{M} < N$ the transformed model (4.5) makes an under-determined system. This is similar to the original MMV model (3.2) being under-determined when $M < N$. Thus, from the theory of compressive sensing, it is again possible to solve the under-determined system if a certain sparsity is withheld, namely $\boldsymbol{\delta}_s$ being $\widetilde{M}$-sparse. Assuming the sufficient sparsity on $\boldsymbol{\delta}_s$ is withheld it is possible to learn the dictionary matrix of the covariance domain $\mathbf{D}_s$. This can be done by traditional dictionary learning methods applied to the measurements represented in the covariance-domain $\text{vec}\left(\widehat{\boldsymbol{\Sigma}}_{\mathbf{Y}_s}\right)$ for all segments s.

Dictionary Learning

As mentioned, within the theory of compressive sensing the matrix $\mathbf{A}$ is referred to as a dictionary matrix. When the dictionary matrix is not known a priori it is essential how to choose the dictionary matrix in order to achieve the best recovery, of a sparse vector $\mathbf{x}$ from the observed measurements $\mathbf{y}$. This is clarified from the proof of theorem A.1.1 in appendix A.1. One choice is a pre-constructed dictionary. In many cases the use of a pre-constructed dictionary results in simple and fast algorithms for reconstruction of $\mathbf{x}$ [12]. However, a pre-constructed dictionary is typically fitted to a specific kind of data. For instance the discrete Fourier transform or the discrete wavelet transform are used especially for sparse representation of images [12]. Hence the results of using such dictionaries depend on how well they fit the data of interest, which is establishing a certain limitation.

The alternative option is to consider an adaptive dictionary based on a set of training data that resembles the data of interest. For this purpose learning methods are considered to empirically construct a dictionary. There exist several dictionary learning algorithms. One is the K-SVD algorithm which was presented in 2006 by Elad et al. and found to outperform pre-constructed dictionaries, when computational cost is of secondary interest [1]. The concept of the K-SVD algorithm is introduced here, and the more detailed algorithm is to be found in appendix A.2.

Consider, from the general MMV model (3.2), the measurement matrix $\mathbf{Y} \in \mathbb{R}^{M\times L}$ consisting of measurement vectors $\{\mathbf{y}_j\}_{j=1}^{L}$. Let the set of measurement vectors make a set of L training examples each forming a linear system

$$\mathbf{y}_j = \mathbf{A}\mathbf{x}_j.$$

From the linear system one can learn a suitable dictionary $\hat{\mathbf{A}}$, and the sparse representation of the source matrix $\hat{\mathbf{X}} \in \mathbb{R}^N$ with the source vectors $\{\hat{\mathbf{x}}_j\}_{j=1}^{L}$. For a known sparsity constraint k dictionary learning can be defined by the following optimization problem.

$$\min_{\mathbf{A},\mathbf{X}} \sum_{j=1}^{L} \|\mathbf{y}_j - \mathbf{A}\mathbf{x}_j\|_2^2 \quad \text{subject to} \quad \|\mathbf{x}_j\|_0 \le k,\ 1 \le j \le L, \tag{4.6}$$

where both $\mathbf{A}$ and $\mathbf{x}_j$ are quantities to be determined. Learning the dictionary by the K-SVD algorithm consists of joint solving of the optimization problem with respect to $\mathbf{A}$ and $\mathbf{X}$. An initial $\mathbf{A}_0 = [\mathbf{a}_1, \ldots, \mathbf{a}_N]$ is chosen and the corresponding $\mathbf{X}_0 = [\mathbf{x}_1, \ldots, \mathbf{x}_L]$ is determined, where $\mathbf{x}_j = [x_{1j}, \ldots, x_{Nj}]^T$. Then, for each iteration an update rule is applied to every column of $\mathbf{A}_0$. That is updating first $\mathbf{a}_j$ for $j = 1, \ldots, N$ and then the corresponding row $\mathbf{x}_{i\cdot}$ where $i = j$. More details on the K-SVD algorithm are found in appendix A.2. The uniqueness of the dictionary $\hat{\mathbf{A}}$ depends on the recovery sparsity condition. As clarified earlier in section 3.3 the recovery of a unique solution $\mathbf{X}^*$ is only possible if $k < M$ [6].

Application of Dictionary Learning

By the establishment of a dictionary learning algorithm, the transformed mixing matrix $\mathbf{D}_s$ from (4.5) can be learned. Remember that (4.5) is a single vector model, thus in order to make training samples for learning $\mathbf{D}_s$ a further segmentation is needed. This is segmentation of $\mathbf{Y}_s$ indexed by s'. For convenience segment index s will be omitted through out this chapter, as the same theory applies to all segments s. Hence, $\mathbf{Y}_{s'}$ referrers to one segment within the outer segment of measurements $\mathbf{Y}_s$.

The transformed and vectorized measurements $\text{vec}\left(\widehat{\mathbf{\Sigma}}_{\mathbf{Y}_{s'}}\right), \forall s'$ now makes the training dataset for learning $\mathbf{D}$. As such each segment s' provides one training sample. Thus, the number of available training samples, denoted $L_{s'}$, depends on the chosen length of the segments. In practice this will vary with respect to the total amount of available data.

K-SVD is applied to the transformed model (4.5) and $\hat{\mathbf{D}}$ is found. Then it is possible to estimate the mixing matrix $\mathbf{A}$ that generated $\mathbf{D}$ through the known relation

$$\mathbf{d}_j = \text{vec}(\mathbf{a}_j \mathbf{a}_j^T).$$

For each column $\mathbf{d}_j$ for $j = 1, \ldots, N$ the following optimization problem is solved with respect to the corresponding column $\mathbf{a}_j$ of the mixing matrix.

$$\min_{\mathbf{a}_j} \|\mathbf{d}_j - \text{vec}\left(\mathbf{a}_j \mathbf{a}_j^T\right)\|_2^2,$$

equivalent to

$$\min_{\mathbf{a}_j} \|\text{vec}^{-1}(\mathbf{d}_j) - \mathbf{a}_j \mathbf{a}_j^T\|_2^2. \tag{4.7}$$

From [4] the global minimizer to (4.7) is given as $\mathbf{a}_j^* = \sqrt{\lambda_j}\mathbf{b}_j$, without further details

or a source. Here λ_j is the largest eigenvalue of $\text{vec}^{-1}(\mathbf{d}_j)$, where

$$\text{vec}^{-1}(\mathbf{d}_j) = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1N} \\ d_{21} & d_{22} & \cdots & d_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ d_{N1} & d_{N2} & \cdots & d_{NN} \end{bmatrix}, \quad j = 1, \ldots, N$$

and $\mathbf{b}_j$ is the corresponding eigenvector.

From this result each column of the mixing matrix $\mathbf{A}$ can be estimated. Hence, it is possible to determine the mixing matrix in the case where the measurements transformed into the covariance-domain makes an under-determined system. Provided however that the necessary sparsity constraint of $\boldsymbol{\delta}$ being $\widetilde{M}$-sparse is withheld. Remember $\widetilde{M} := \frac{M(M+1)}{2}$ thus $M < k$ is allowed and the original sparsity constraint, $\mathbf{X}$ being M-sparse, is relaxed.

4.2.2 Over-determined System

Consider again the measurements represented in the covariance-domain (4.5). In the case of $\widetilde{M} > N$ an over-determined system is achieved where $\mathbf{D}$ is high and thin. In general such a system is inconsistent. Thus, it is not possible to find $\mathbf{D}$ by traditional dictionary learning methods and different methods must be considered. Let the set for transformed measurements be denoted by

$$\mathbf{Y}_{\text{cov}} := \left\{ \text{vec}\left(\widehat{\boldsymbol{\Sigma}}_{\mathbf{Y}_{s'}} \right) \right\}_{s'=1}^{L_{s'}}.$$

When $\widetilde{M} > N$ it is expected from model (4.5) that the transformed measurements $\mathbf{Y}_{\text{cov}}$ live on or near a subspace of dimension N. This subspace is spanned by the columns of $\mathbf{D} \in \mathbb{R}^{\widetilde{M} \times N}$, and is denoted as $\mathcal{R}(\mathbf{D})$. To learn $\mathcal{R}(\mathbf{D})$ without having to impose any sparsity constraint on $\boldsymbol{\delta}$ it is possible to use principal component analysis (PCA). The basic theory of PCA in found in appendix A.3.

PCA is applied to the set of transformed measurements $\mathbf{Y}_{\text{cov}}$ and the N first principal components are determined. The principal components form a set of basis vectors $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_N]$. That is a new basis which spans the subspace on which $\mathbf{Y}_{\text{cov}}$ lives. Thus the equality $\mathcal{R}(\mathbf{U}) = \mathcal{R}(\mathbf{D})$ can be justified [4]. However, this equality does not imply that $\mathbf{D} = \mathbf{U}$. In the case of two bases spanning the same vector space, namely $\mathcal{R}(\mathbf{U}) = \mathcal{R}(\mathbf{D})$, the projection operator of the given subsets must be the same. Consider the projection operator P projecting onto the space $\mathcal{R}(\mathbf{D})$ spanned by the columns of $\mathbf{D}$, $\text{P} : \mathbb{R}^{\widetilde{M}} \to \mathcal{R}(\mathbf{D})$. Due to $\mathbf{D}$ having full rank it is a well-known result that $\text{P} = \mathbf{D}(\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T$. Thus $\mathcal{R}(\mathbf{U})$ and $\mathcal{R}(\mathbf{D})$ having the same projection operator is true if and only if

$$\mathbf{D}(\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T = \mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T.$$

Now, remember from the relation between $\mathbf{A}$ and $\mathbf{D}$ that $\mathbf{d}_j = \text{vec}(\mathbf{a}_j\mathbf{a}_j^T)$. From this it is possible to obtain $\mathbf{D}$ and then $\mathbf{A}$, such that $\mathbf{D}$ spans $\mathcal{R}(\mathbf{U})$ and $\mathbf{d}_j = \text{vec}(\mathbf{a}_j\mathbf{a}_j^T)$. This is specified by the following optimization problem [4]

$$\min_{\{\mathbf{a}_j\}_{j=1}^N} \|\mathbf{D}(\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T - \mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\|_F^2$$
$$\text{s.t. } \mathbf{d}_j = \text{vec}(\mathbf{a}_j\mathbf{a}_j^T), \tag{4.8}$$

where $\mathbf{U}$ results from PCA performed on $\mathbf{Y}_{\text{cov}}$. From the source proposing the method [4], it is only notified that the optimization problem (4.8) is minimized by quasi-Newton optimization methods. Hence, the exact minimization approach can not be recreated. In the following section the optimization problem is analyzed and processed in order to determine a suitable solution method.

4.2.3 Solution to Optimization Problem

The optimization problem (4.8) consists of an objective function forming a least-squares problem with respect to the Frobenius norm. It is given that the squared norm, both the Euclidean and the Frobenius norm, are strictly convex [10, p.173]. Thus the objective function of (4.8) is assumed to be convex. The constraint in (4.8) is a set of quadratic equality constraints. This categorizes the optimization problem as a quadratically constraint quadratic program. However, the constraints are not necessarily convex. By the constraints not being considered convex the optimization problem does not meet the requirements of a convex optimization problem. Hence, the numerical solution methods for convex optimization problems, for which convergence is ensured, does not apply directly. In fact a non-convex quadratically constraint quadratic program ins know to be a NP-hard problem[7]. Thus, some sort of relaxation is preferred.

Due to the nature of the constraints, it should be possible to reformulate the objective function to include the constraints into the objective function. That is constructing an unconstrained least-squares problem, which is a special subclass of convex optimization [8].

Let $\mathbf{D} = f(\mathbf{a}_1, \ldots, \mathbf{a}_N)$ where $f(\mathbf{a}_1, \ldots, \mathbf{a}_N) = \left\{\mathbf{d}_j = \text{vec}\left(\mathbf{a}_j\mathbf{a}_j^T\right)\right\}_{j=1}^N$. Then an optimization problem without constraints is achieved, and it can be solved by use of basic gradient methods, for instance the Newton method. In order to avoid an explicit expression of the inverse Hessian, used in the Newton method, quasi-Newton methods can be considered[3]. The general idea of quasi-Newton methods is to let the direction of search be based on a positive definite matrix generated from available data as an alternative to the Hessian.

Rendering of general optimization theory and the theory of quasi-Newton methods is omitted in this thesis and the reader is referred to source [3]. For the implementation of Cov-DL in chapter 6 a predefined optimization module, using a quasi-Newton method, will be applied.

4.3 Pseudo Code of the Cov-DL Algorithm

Algorithm 1 Cov-DL

procedure COV-DL($\mathbf{Y}_s$)
  for $s' \leftarrow 1, \cdots, L_{s'}$ **do**
    $\mathbf{y}_{\mathrm{cov}_{s'}} = \mathrm{vec}\left(\widehat{\mathbf{\Sigma}}_{\mathbf{Y}_{s'}}\right)$
  end for
  $\mathbf{Y}_{\mathrm{cov}} = \{\mathbf{y}_{\mathrm{cov}_{s'}}\}_{s'=1}^{L_{s'}}$

  if $N \geq \widetilde{M}$ **then**
    procedure K-SVD($\mathbf{Y}_{\mathrm{cov}}$)
      returns $\mathbf{D} \in \mathbb{R}^{\widetilde{M} \times N}$
    end procedure
    for $j \leftarrow 1, \cdots, N$ **do**
      $\mathbf{T} = \mathrm{vec}^{-1}(\mathbf{d}_j)$
      $\lambda_j \leftarrow \max\{\mathrm{eigenvalue}(\mathbf{T})\}$
      $\mathbf{b}_j \leftarrow$ eigenvector(λ_j)
      $\mathbf{a}_j \leftarrow \sqrt{\lambda_j}\mathbf{b}_j$
    end for
    $\mathbf{A} = \{\mathbf{a}_j\}_{j=1}^{N}$
  end if

  if $N < \widetilde{M}$ **then**
    procedure PCA($\mathbf{Y}_{\mathrm{cov}}$)
      returns $\mathbf{U} \in \mathbb{R}^{\widetilde{M} \times N}$
    end procedure
    procedure MIN. $\mathbf{A}$ IN $(\|\mathbf{D}(\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T - \mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\|_F^2)$
      returns $\mathbf{A} = \{\mathbf{a}_j\}_{j=1}^{N}$
    end procedure
  end if
end procedure

4.4 Remarks

Through this chapter the theoretical aspects of the Cov-DL method proposed by [4] have been investigated in order to create algorithm 1 from which the implementation of Cov-DL will be based. Furthermore, the following remarks are considered with respect to the implementation.

The length of each time segment s has to be defined with respect to the assumption of the signals being stationary. However, it can not be assured that the

assumption is withheld for every segment and this will introduce a source of error. This must be taken into account in the preprocessing part for the implementation of Cov-DL when the EEG measurements are divided into segments.

For each segment a further segmentation is conducted into segments s', each serving as one sample in the covariance-domain. Here the number of samples $L_{s'}$, depending on the chosen length, is most likely to influence the estimated dictionary. This is assuming that more training data will provide better results. Here a certain trade off may be considered. Longer segments s' lead to better sample covariance representation but also a less number of training samples. Opposite, too short segments s' might compromise the sample covariance-domain representation, thus the number of training sample will increase but the training samples might not be as representative. This trade off must be taken into account during the implementation of Cov-DL. Furthermore, overlapping segments might be an option for potential improvement of the Cov-DL method.

Chapter 5

Multiple Sparse Bayesian Learning

In this chapter the multiple sparse Bayesian learning (M-SBL) method is described in details, leading to an algorithm specifying the method. As the method leverage a Bayesian framework the general concept of Bayesian inference is briefly introduced prior to the M-SBL method. The chapter is generally based upon [5] where the method is applied to the MMV model, which is of interest in this thesis. More detailed theory is found in [31] and [32].

Consider again the MMV model (3.3) of EEG measurements

$$\mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{E}. \tag{5.1}$$

For convenience the segment index s is omitted as the same theory applies to every segment. Note that $\mathbf{A}$ is known throughout the chapter, as it is estimated by Cov-DL in chapter 4. The aim is to recover the source matrix $\mathbf{X}$ by an estimate $\hat{\mathbf{X}}$, in the case of fewer measurements than active sources, $M < k$, where $k \leq N$.

In [5] it is proven that exact localization of the active sources can be achieved with M-SBL for $M < k$, when two sufficient conditions are satisfied. The basic approach of M-SBL is to apply Bayesian statistics to find a support set S specifying the non-zero rows of the source matrix $\mathbf{X}$ which corresponds to localization of the active sources. Finally, the values of the localized active sources can be estimated.

5.1 Bayesian Inference

The formal framework of Bayesian statistics is Bayes' theorem [19, p. 86]. The objective of Bayes' theorem is to leverage of both data and some specified prior. This is where the distinguishes from the likelihood of classical frequentist statistics lies.

Consider now the current MMV model (5.1) within the Bayesian framework. The model parameter – the source matrix $\mathbf{X}$ – is wished estimated given the measurement matrix $\mathbf{Y}$. By Bayes' theorem the distribution of $\mathbf{X}$ given $\mathbf{Y}$ is established, that is the posterior distribution

$$p(\mathbf{X}|\mathbf{Y}) = \frac{p(\mathbf{Y}|\mathbf{X})p(\mathbf{X})}{p(\mathbf{Y})}.$$

Here $p(\mathbf{Y}|\mathbf{X})$ is the probability density function of $\mathbf{Y}$ given $\mathbf{X}$, also referred to as the likelihood function. $p(\mathbf{X})$ is a prior distribution of $\mathbf{X}$ and $p(\mathbf{Y})$ is the distribution of $\mathbf{Y}$ serving as a normalizing parameter. By maximizing the posterior distribution $p(\mathbf{X}|\mathbf{Y})$ with respect to $\mathbf{X}$, the maximum a posteriori (MAP) estimate for the source matrix is established

$$\hat{\mathbf{X}}_{\text{MAP}} = \arg\max_{\mathbf{X}} \frac{p(\mathbf{Y}|\mathbf{X})p(\mathbf{X})}{p(\mathbf{Y})}.$$

That is the estimate of $\hat{\mathbf{X}}$ with the highest posterior probability given the measurements $\mathbf{Y}$. In the desired case where $M < N$ the MMV model (5.1) makes an underdetermined system. Hence, an infinite number of solutions of equal likelihoods does potentially exist.

Let the source matrix $\mathbf{X}$ be seen as a variable drawn from some distribution $p(\mathbf{X})$, as such it is possible to narrow down the solution space. Assuming a prior belief that $\mathbf{Y}$ is generated from a sparse source matrix, gives a so-called sparsity inducing prior. That is the entries of $\mathbf{X}$ is drawn from some distribution which has a sharp, possibly infinite, spike at zero surrounded by fat tails. Here the fat tails make room for the non-zero values, which are here seen as outliers.

For simplicity a Gaussian prior is however preferred. The use of a Gaussian distribution can almost be justified if a mixture of two Gaussian distributions are considered such that the variable is drawn from one of the two with equal likelihood. One where the variance of the distribution is close to zero, resembling the narrow spike around the mean at zero. And one with high variance resembling the fat tails.

Different MAP estimation approaches exist separated by the choice of sparsity inducing prior and optimization method. However, regardless of the approach some problems have shown to occur when using a fixed algorithm-dependent prior. One issue occurs if the chosen prior does not assign sufficient probability to the sparse solution, leading to non-recovery. Another issue is that a combinatorial number of suboptimal local solutions can occur. By use of automatic relevance determination (ARD) the problems related to the fixed sparsity inducing prior can be avoided [31, p. 20]. The main asset of this alternative approach is the use of an empirical prior. That is a flexible prior distribution depending on an unknown set of hyperparameters, which is to be learned from the data.

5.1.1 Empirical Bayesian Estimation

First assume that the likelihood function $p(\mathbf{Y}|\mathbf{X})$ is Gaussian, with noise variance $\sigma^2\mathbf{I}$. In general it is assumed that σ^2 is known. Furthermore, the noise-free case where $\sigma^2 \to 0$ will be discussed. Due to $\mathbf{y}_{\cdot j}$ consisting of measurement of individual EEG sensors, it is reasonable to assume independence. Furthermore, it is clear from the MMV model that one sample of measurements $\mathbf{y}_{\cdot j}$ only depends of one source sample $\mathbf{x}_{\cdot j}$. Hence, every entry in $\mathbf{Y}$ are assumed independently and identically distributed with likelihood

$$\begin{aligned} p(y_{ij}|x_{ij}) &\sim \mathcal{N}(\mathbf{A}_{i\cdot}\mathbf{x}_{\cdot j}, \sigma^2) \\ &= \frac{1}{\sigma^2\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{y_{ij} - \mathbf{A}_{i\cdot}\mathbf{x}_{\cdot j}}{\sigma}\right)^2\right). \end{aligned}$$

Now the empirical prior is defined due to the application of ARD. A L-dimensional Gaussian prior is assigned to each row in $\mathbf{X}$. Note that, similar to $\mathbf{Y}$, the parameters x_{ij} are assumed to be independent and identically distributed. The empirical prior for each x_{ij} is then defined by a Gaussian distribution with zero mean and a variance controlled by an unknown hyperparameter γ_i:

$$p(x_{ij};\gamma_i) \sim \mathcal{N}(0, \gamma_i).$$

Note that every entry of the i-th row is controlled by the same hyperparameter γ_i. That is one source signal over time is controlled by one hyperparameter. By combining the prior of each parameter, the prior of $\mathbf{X}$ is fully specified by

$$p(\mathbf{X};\boldsymbol{\gamma}) = \prod_{i=1}^{N} p(\mathbf{x}_{i\cdot};\gamma_i),$$

with the hyperparameter vector $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_N]^T$. Note that the prior can be factorized over columns, resulting in

$$p\left(\mathbf{x}_{\cdot j};\boldsymbol{\gamma}\right) = \prod_{i=1}^{N} p\left(x_{ij};\gamma_i\right).$$

Combining the prior $p\left(\mathbf{x}_{\cdot j};\boldsymbol{\gamma}\right)$ and the likelihood $p(\mathbf{y}_{\cdot j}|\mathbf{x}_{\cdot j})$ the posterior of the j-th column of the source matrix $\mathbf{X}$ becomes

$$\begin{aligned} p(\mathbf{x}_{\cdot j}|\mathbf{y}_{\cdot j};\boldsymbol{\gamma}) &= \frac{p(\mathbf{y}_{\cdot j}|\mathbf{x}_{\cdot j};\boldsymbol{\gamma})p(\mathbf{x}_{\cdot j};\boldsymbol{\gamma})}{p(\mathbf{y}_{\cdot j}|\boldsymbol{\gamma})} \\ &= \frac{p(\mathbf{y}_{\cdot j}|\mathbf{x}_{\cdot j};\boldsymbol{\gamma})p(\mathbf{x}_{\cdot j};\boldsymbol{\gamma})}{\int p(\mathbf{y}_{\cdot j}|\mathbf{x}_{\cdot j})p(\mathbf{x}_{\cdot j};\boldsymbol{\gamma})\ d\mathbf{x}_{\cdot j}} \\ &\propto p(\mathbf{y}_{\cdot j}|\mathbf{x}_{\cdot j};\boldsymbol{\gamma})p(\mathbf{x}_{\cdot j};\boldsymbol{\gamma}) \\ &\sim \mathcal{N}(\boldsymbol{\mu}_{\cdot j}, \boldsymbol{\Sigma}), \end{aligned} \tag{5.2}$$

where the denominator is the marginal likelihood of $\mathbf{y}_{\cdot j}$ also referred to as the evidence. The marginalization is elaborated in the following section. Mean and covariance of (5.2) for every $j = 1, \ldots, L$ is given as

$$\Sigma = \mathrm{Cov}(\mathbf{x}_{\cdot j}|\mathbf{y}_{\cdot j}; \boldsymbol{\gamma}) = \boldsymbol{\Gamma} - \boldsymbol{\Gamma}\mathbf{A}^T\boldsymbol{\Sigma}_y^{-1}\mathbf{A}\boldsymbol{\Gamma} \tag{5.3}$$

$$\mathcal{M} = [\boldsymbol{\mu}_{\cdot 1}, \ldots, \boldsymbol{\mu}_{\cdot L}] = \mathbb{E}[\mathbf{X}|\mathbf{Y}; \boldsymbol{\gamma}] = \boldsymbol{\Gamma}\mathbf{A}^T\boldsymbol{\Sigma}_y^{-1}\mathbf{Y}, \tag{5.4}$$

where $\boldsymbol{\Gamma} = \mathrm{diag}(\boldsymbol{\gamma})$ and $\boldsymbol{\Sigma}_y = \sigma^2\mathbf{I} + \mathbf{A}\boldsymbol{\Gamma}\mathbf{A}^T$. The derivation of the posterior mean and covariance if found in appendix B.1.

Now let the posterior mean $\mathcal{M}$ serve as the estimate for the source matrix $\mathbf{X}$ [31, p. 147]. It is clear that whenever $\gamma_i = 0$ the corresponding $\mathbf{x}_{i\cdot}$ is equal to zero with probability 1:

$$\mathbb{P}(\mathbf{x}_{i\cdot} = \mathbf{0}|\mathbf{Y}; \gamma_i = 0) = 1.$$

This ensures the posterior mean $\mathcal{M}$ of the i-th row, $\boldsymbol{\mu}_{i\cdot}$, becomes zero, whenever $\gamma_i = 0$ as desired.

From this it is evident that for estimating the support set of $\mathbf{X}$ it is sufficient to estimate the hyperparameter $\boldsymbol{\gamma}$, from which the support set S can be extracted. This leads to the actual M-SBL algorithm for which the aim is to estimate $\boldsymbol{\gamma}$ and the corresponding $\mathcal{M}$.

5.2 M-SBL for estimation of X

The M-SBL algorithm is now specified in order to estimate the hyperparameter $\boldsymbol{\gamma}$ and then the corresponding unknown sources $\mathbf{X}$. Due to the empirical Bayesian strategy the unknown source matrix $\mathbf{X}$ is integrated out, also referred to as marginalization. By integrating the posterior with respect to the unknown sources $\mathbf{X}$ the marginal likelihood of the observed data $\mathbf{Y}$ is achieved [31, p. 146]

$$\begin{aligned}\mathcal{L}(\boldsymbol{\gamma}; \mathbf{Y}) &= \int p(\mathbf{Y}|\mathbf{X})p(\mathbf{X}; \boldsymbol{\gamma})\, d\mathbf{X} \\ &= p(\mathbf{Y}|\boldsymbol{\gamma}).\end{aligned}$$

The resulting marginal likelihood of $\boldsymbol{\gamma}$ is to be maximized with respect to $\boldsymbol{\gamma}$, that is the maximum likelihood estimate (MLE). From the ARD approach the MLE is considered the cost function. The $-2\log(\cdot)$ transformation is applied in order for the cost function to be minimized, and factors not depending on $\mathbf{Y}$ is removed. This

result in the following log likelihood:

$$\begin{aligned}\ell(\boldsymbol{\gamma};\mathbf{Y}) &= -2\log(p(\mathbf{Y};\boldsymbol{\gamma})) \\ &= -2\log\left(2\pi^{\frac{M}{2}}|\boldsymbol{\Sigma}_y|^{\frac{1}{2}}\exp\left(-\frac{1}{2}\sum_{j=1}^{L}\mathbf{y}_{\cdot j}^T\boldsymbol{\Sigma}_y^{-1}\mathbf{y}_{\cdot j}\right)\right) \\ &= L\log(|\boldsymbol{\Sigma}_y|) + \sum_{j=1}^{L}\mathbf{y}_{\cdot j}^T\boldsymbol{\Sigma}_y^{-1}\mathbf{y}_{\cdot j}. \end{aligned} \tag{5.5}$$

It is not expected that an explicit solution to the minimization problem can be found by differentiating and letting the expression equal to zero. Hence, the problem has to be solved iteratively based on an initial parameter guess $\boldsymbol{\gamma}_{(0)}$.

One iterative method is the expectation maximization (EM) algorithm. In general each iteration consists of an expectation (E) step, where a function determines the expectation of the likelihood function given the currently estimated parameters. The E-step is followed by an maximization (M) step which computes the parameters by maximizing the expected likelihood found in the E-step. In this case the E-step is to compute the posterior moments using (5.3) and (5.4) while the M-step is the following update rule of γ_i [31, p.147]

$$\gamma_i^{(k+1)} = \frac{1}{L}\|\boldsymbol{\mu}_{i\cdot}\|_2^2 + \boldsymbol{\Sigma}_{ii}, \quad \forall i = 1,\ldots,N.$$

The M-step is, in general, very slow on large data. An alternative is to use a fixed-point update rule to fasten convergence on large data. However, the resulting convergence has been found to sometimes be inferior compared to the convergence obtained by the above update rule [31, p.147]. The general point of a fixed-point update is to define the new value from the previous value. The fixed-point updating step is here achieved by taking the derivative of the marginal log likelihood $\ell(\boldsymbol{\gamma})$ with respect to $\boldsymbol{\gamma}$ and equating it with zero. This leads to the following update rule which can replace the above M-step in the EM-algorithm [31, p.147]

$$\gamma_i^{(k+1)} = \frac{\frac{1}{L}\|\boldsymbol{\mu}_{i\cdot}\|_2^2}{1-\gamma_i^{-1(k)}\boldsymbol{\Sigma}_{ii}}, \quad \forall i = 1,\ldots,N. \tag{5.6}$$

Empirically this alternative update rule has shown to be useful in highly underdetermined large-scale cases. Based on many hyperparameters being driving toward zero, allowing for the corresponding weight in the source matrix to be discarded. For simultaneous sparse approximation problems, this is the process referred to as multiple sparse Bayesian learning, M-SBL.

From the resulting $\boldsymbol{\gamma}^*$ the support set S of the source matrix $\mathbf{X}$ is extracted,

$$S = \{i|\gamma_i^* \neq 0\},$$

concluding the localization of active sources within $\mathbf{X}$. In practice some arbitrary small threshold can be used such that any sufficiently small hyperparameter is discarded [31, p.149]. For identification of the active sources the estimate of the source matrix $\mathbf{X}$ is given as $\hat{\mathbf{X}} = \mathcal{M}$, with $\mathcal{M} = \mathbb{E}[\mathbf{X}|\mathbf{Y}; \boldsymbol{\gamma}^*]$. This leads to the following estimate

$$\hat{\mathbf{X}} = \begin{cases} \mathbf{x}_{i\cdot} = \boldsymbol{\mu}_{i\cdot}, & i \in S \\ \mathbf{x}_{i\cdot} = \mathbf{0}, & i \notin S \end{cases}$$

As mentioned, the case of a the noise free sparse representations should be considered. That is the limit when $\sigma^2 \to 0$. Here the M-SBL steps can be adapted easily by using a modified version the moments, given by [31, p.148] as

$$\Sigma = \left[\mathbf{I} - \boldsymbol{\Gamma}^{1/2}\left(\mathbf{A}\boldsymbol{\Gamma}^{1/2}\right)^{\dagger}\mathbf{A}\right]\boldsymbol{\Gamma}$$
$$\mathcal{M} = \boldsymbol{\Gamma}^{1/2}\left(\mathbf{A}\boldsymbol{\Gamma}^{1/2}\right)^{\dagger}\mathbf{Y}$$

where $(\cdot)^{\dagger}$ is the pseudo-inverse.

5.2.1 When k is Known

From M-SBL the number of active sources k is estimated as the number of non-zero entries in the hyperparameter $\boldsymbol{\gamma}^*$. However, in the current scenario $\mathbf{A}$ is estimated by Cov-DL, prior to the application of M-SBL, where k is provided as input to Cov-DL, cf. chapter 4. Thus, k is known in prior to M-SBL and can hereby be used as a known parameter to the M-SBL method. With k being known the estimation of the support set S from the non-zero rows of $\boldsymbol{\gamma}^*$, cf. section 5.1.1, is overruled. Instead, when generating the support set S^k one choose the k largest entries of $\boldsymbol{\gamma}^*$ [5, p. 3]. The estimate of the source matrix is then found by

$$\hat{\mathbf{X}} = \begin{cases} \mathbf{x}_{i\cdot} = \boldsymbol{\mu}_{i\cdot}, & i \in S^k \\ \mathbf{x}_{i\cdot} = \mathbf{0}, & i \notin S^k \end{cases}$$

5.2.2 Pseudo Code for the M-SBL Algorithm

Algorithm 2 M-SBL

procedure M-SBL($\mathbf{Y}, \mathbf{A}$)
  $\boldsymbol{\gamma}_{(0)} = \mathbf{1} \in \mathbb{R}^N$
  tol = 0.0001
  while p < 3 **or** any($\boldsymbol{\gamma}_{(p)} - \boldsymbol{\gamma}_{(p-1)}$) ≥ tol **do**
    $\boldsymbol{\Gamma} = \text{diag}(\boldsymbol{\gamma}_{(p)})$
    $\boldsymbol{\Sigma} = \boldsymbol{\Gamma} - \boldsymbol{\Gamma}\mathbf{A}^T\boldsymbol{\Sigma}_y^{-1}\mathbf{A}\boldsymbol{\Gamma}$
    $\mathcal{M} = \boldsymbol{\Gamma}\mathbf{A}^T\boldsymbol{\Sigma}_y^{-1}\mathbf{Y}$
    for $i = 1, \ldots, N$ **do**
      $\gamma_{i(p+1)} = \frac{\frac{1}{L}\|\boldsymbol{\mu}_{i\cdot}\|_2^2}{1 - \gamma_{i(p)}^{-1}\Sigma_{ii}}$
    end for
    p + = 1
  end while
  Return $\mathcal{M}, \boldsymbol{\gamma}^*$
end procedure
procedure SUPPORT($\mathcal{M}, \boldsymbol{\gamma}^*$, k)
  Support = $\mathbf{0} \in \mathbb{R}^k$
  for $j = 1, \ldots, k$ **do**
    if $\boldsymbol{\gamma}^*[\arg\max(\boldsymbol{\gamma}^*)]$!= 0 **then**
      Support(j) = $\arg\max(\boldsymbol{\gamma}^*)$
      $\boldsymbol{\gamma}^*[\arg\max(\boldsymbol{\gamma}^*)] = 0$
    end if
  end for
  $\hat{\mathbf{X}} = \mathbf{0} \in \mathbb{R}^{N \times L}$
  for i in Support **do**
    $\hat{\mathbf{X}}_{i\cdot} = \mathcal{M}_{i\cdot}$
  end for
  Return $\hat{\mathbf{X}}$
end procedure

5.2.3 Sufficient Conditions for Exact Source Localization

In [5] it is proven that exact source localization is guaranteed in the under-determined case, $k > M$ when the conditions in the following theorem are fulfilled. The theorem is based on a theoretical analysis of the minima where noise-free conditions are considered, that is letting $\sigma^2 \to 0$. Thus, it is essential that the following theorem applies to the noise-free case.

First, define a function $f : \mathbb{R}^{M \times N} \to \mathbb{R}^{\frac{M(M+1)}{2} \times N}$, such that for $B = f(\mathbf{A})$ the

j-th column is given as $\mathbf{b}_{\cdot j} = \text{vec}(\mathbf{a}_{\cdot j}\mathbf{a}_{\cdot j}^T)$. Here the function $\text{vec}(\cdot)$ corresponds to the function defined in section 4.1, being a vectorization of the lower triangular part of a matrix. Furthermore, $\mathbf{X}_S \in \mathbb{R}^{k \times L}$ denote only the non-zero rows of $\mathbf{X}$ – the active sources.

Theorem 5.2.1
Given a dictionary matrix $\mathbf{A}$ and a set of observed measurement $\mathbf{Y}$, M-SBL recovers the support set of any size k exactly in the noise-free case, if the following conditions are satisfied.

1. The active sources $\mathbf{X}_S$ are orthogonal. That is, $\mathbf{X}_S\mathbf{X}_S^T = \mathbf{\Lambda}$, where $\mathbf{\Lambda}$ is a diagonal matrix.
2. $\text{Rank}(f(\mathbf{A})) = N$.

The proof can be found in [5, p. 16].

Chapter 6

Implementation and Verification

In this chapter the implementation process of the main algorithm is described. The main algorithm is the two methods Cov-DL and M-SBL from respectively chapter 4 and 5 combined into one algorithm.

The implementation of each method is initially tested on a simple deterministic simulated data to verify the implementation. Next, both methods are tested on stochastic simulated data which aim to resemble real EEG measurements. By simulating a data set the true model parameters are known which allows for measuring the precision of the implemented methods. In addition different model variables are investigated in order to improve the model. Finally, the main algorithm is tested on the simulated data sets, and conclusions are drawn based on the results.

6.1 Implementation

In this section the implementation of the main algorithm is described. A flowchart is constructed to illustrate the flow through the code. The main algorithm consists of three main stage: an initialization, application of Cov-DL for recovery of $\mathbf{A}$ and lastly application of M-SBL for recovery of $\mathbf{X}$. At the flowchart, figure 6.1, each stage of the algorithm is illustrated within one horizontal row. Furthermore, the input and output are placed in their own row.

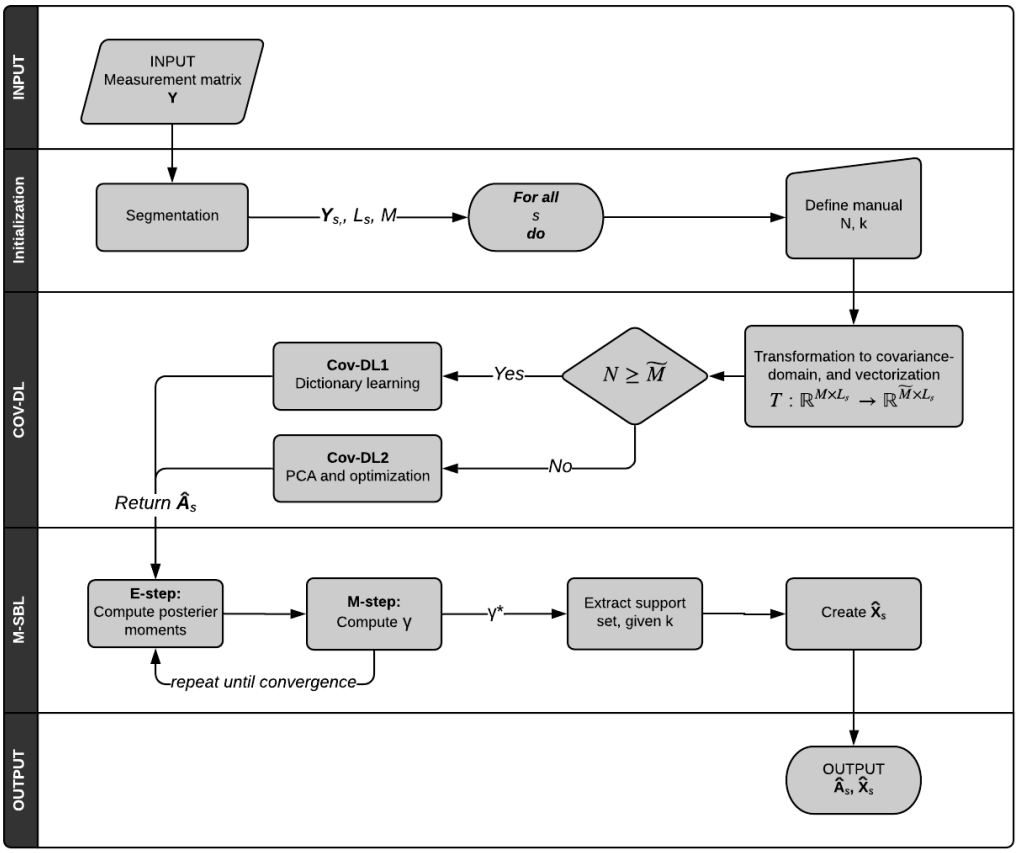

Figure 6.1: Flowchart illustrating the implementation of the main algorithm.

The input of the main algorithm consists of the measurement matrix $\mathbf{Y} \in \mathbb{R}^{M \times L}$, along with the corresponding sample frequency f. Within the initialization stage the measurement matrix $\mathbf{Y}$ goes through a segmentation as described in section 3.2.1. Resulting in non-overlapping segments. The length of the segments is predefined by a time interval of t seconds such that $L_s = tf$. Each segment s is now specified by the measurement matrix $\mathbf{Y}_s \in \mathbb{R}^{M \times L_s}$. After the segmentation a loop is constructed such that the remaining two stages of the main algorithm, are performed for every segment s. First N and k are manually defined. This definition is either known in advance from the data or in the case of real EEG measurements they are unknown and a qualified guess must be made.

With the specifications of one segment, the second stage of the algorithm is initialized, recovery of $\mathbf{A}_s$. The implementation of the Cov-DL stage follows algorithm 1 from section 4.3 closely, thus only the main steps are illustrated on the flow diagram 6.1. First the measurement matrix $\mathbf{Y}_s$ is transformed to the covariance-domain and vectorized. This results in the extension of the dimensionality from M to $\widetilde{M} = \frac{M(M+1)}{2}$. Next, the estimation of $\mathbf{A}_s$ is performed from either Cov-DL1 or

Cov-DL2 depending on the relation between $\widetilde{M}$ and N, as described respectively in section 4.2.1 and 4.2.2. The estimate $\hat{\mathbf{A}}_s$ and the measurement matrix $\mathbf{Y}_S$ serve as the input to the following stage, M-SBL for recovering of $\mathbf{X}_s$.

The finishing stage of the main algorithm consists of the iterative EM algorithm for maximizing the marginal likelihood (5.5) with respect to $\boldsymbol{\gamma}$. The resulting $\boldsymbol{\gamma}^*$ is the hyperparameter from which $\mathbf{X}_s$ is determined as described in section 5.2. Lastly, the output of the main algorithm $\hat{\mathbf{X}}_s$ and $\hat{\mathbf{A}}_s$ is illustrated on the flowchart 6.1.

6.1.1 Coding Practice

The implementation of the main algorithm is performed in Python 3.6. The software and guide to run the scripts are available through appendix D.

The practical implementation process is based on module development. The established model and the three stages of the main algorithm make the system design. For each stage the necessary tasks are identified and divided into smaller modules. For each module the task is specified, and an algorithm is established and implemented. This is followed by a test of the module and possible modifications until the task is performed without error. Due to the time limitation of this thesis, the software was developed along side the dynamic research process. Hence, the specifications to some modules have been redefined and the modification process are repeated. Finally, the modules are united into one stage for which tests are performed, and lastly all the stages are united to the resulting main algorithm.

The software is based on functions, for example one module is specified by one function, for which docstrings is used, following NumPy docstring format[1] allowing insight into the structure and thoughts behind the different software elements.

For each of the stages Cov-DL and M-SBL, the verification and performance tests are described later in this chapter, followed by the testing phase of the main algorithm.

6.2 Data Simulation

To evaluate the performance of the main algorithm as well as the individual stages, synthetic data are simulated with respect to the model $\mathbf{Y} = \mathbf{AX}$. All data sets are simulated based on the following approach, satisfying the sufficient conditions for recovery, displayed in theorem 5.2.1.

A source matrix $\mathbf{X} \in \mathbb{R}^{N \times L}$ is constructed, such that every non-zero row is sampled individually by some function restricted by having zero mean. By this approach the non-zero rows of $\mathbf{X}$ become close to orthogonal [5], which approximates the first conditions of theorem 5.2.1. Then a mixing matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ is constructed with identically distributed and independent entries. As such the source signals are

[1] `https://numpydoc.readthedocs.io/en/latest/`

randomly mixed and the mixing matrix fulfils the second condition of theorem 5.2.1. With known $\mathbf{A}$ and $\mathbf{X}$, the measurement matrix $\mathbf{Y} \in \mathbb{R}^{M \times L}$ is simulated according to the model, by the matrix product $\mathbf{Y} = \mathbf{AX}$. Note that the error matrix $\mathbf{E}$ is omitted in this chapter, as noise is not included in the synthetic data.

Two different kinds of data sets are simulated. Deterministic data having simple and predictable source signals to ensure a solution and easy visualization. And stochastic data having randomized and fluctuating source signals to resemble realistic EEG measurements.

Note that each simulated data set fulfils the sufficient conditions for recovery, thus segmentation of the synthetic data is not necessary.

6.2.1 Deterministic Data

Two different deterministic data sets are simulated, with a different number of zero rows. The first is specified by $N = 5$, $k = 4$, $M = 3$ and $L = 1000$. That is a source matrix $\mathbf{X}$ with 4 rows individually generated and 1 zero row. By the specifications the source matrix $\mathbf{X}$ is mixed into a measurement matrix $\mathbf{Y}$ with 3 measurements per sample. The second deterministic data set is specified by $N = 8$, $k = 4$, $M = 3$ and $L = 1000$. That is 3 additional zero rows. From the specifications the first data set comply to $N \leq \frac{M(M+1)}{2}$ which imply the use of Cov-DL2. The second data set comply to $N > \frac{M(M+1)}{2}$ and $k \leq \frac{M(M+1)}{2}$ implying the use of Cov-DL1. As such it is possible to test both branches of the Cov-DL method.

The four non-zero source signals of $\mathbf{X}$ are defined by the following individual functions, causing the rows to be approximately orthogonal

1. a sinus signal $\sin(2t)$
2. a sawtooth signal with period $2\pi t$
3. a sinus signal $\sin(4t)$
4. a sign function of a sinus signal $\sin(3t)$

with t being a time index defined in the interval $[0, 4]$ with L samples. Each of the four signals are randomly drawn and used to construct a source matrix $\mathbf{X}$ of size $k \times L$, then zero rows are inserted randomly, such that $\mathbf{X} \in \mathbb{R}^{N \times L}$. The mixing matrix $\mathbf{A}$ of size $M \times N$ is randomly generated from a Gaussian distribution. By multiplying the source matrix and the mixing matrix a measurement matrix $\mathbf{Y}$ is simulated. The resulting deterministic data set then consist of $\{\mathbf{Y}, \mathbf{X}, \mathbf{A}\}$.

In figure 6.2 the first deterministic data set, triggering Cov-DL2, is illustrated by the source signals plotted in the top and the measurement signals plotted in the bottom. This illustrates how the source signals are transformed by the mixing matrix $\mathbf{A}$.

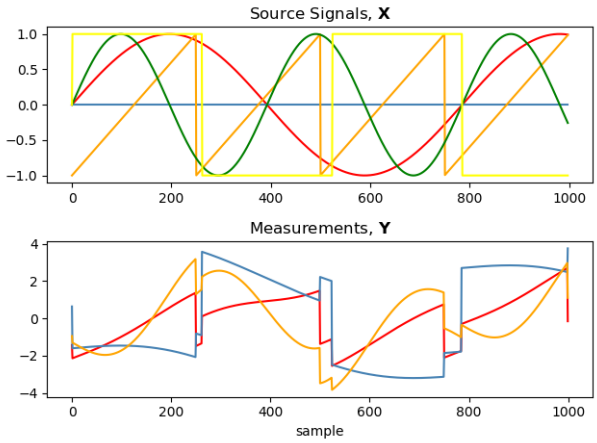

Figure 6.2: Visualization of the source signals $\mathbf{X}$ in comparison to the measurement signals $\mathbf{Y}$ from the deterministic data set specified by $N = 5, M = 3, k = 4$ and $L = 1000$.

6.2.2 Stochastic Data

The purpose of this second kind of data is to resemble EEG measurements for which the main algorithm is intended. Here different data sets are simulated depending on the chosen specifications of N, k, M and L. Every data set is constructed based on four different linear autoregressive processes of various orders, each process representing one source signal

$$x_t^1 = \sum_{i=1}^{2} \phi_i x_{t-i}^1 + w_t^1 \qquad x_t^2 = \sum_{i=1}^{2} \zeta_i x_{t-i}^2 + w_t^2$$

$$x_t^3 = \sum_{i=1}^{3} \eta_i x_{t-i}^3 + w_t^3 \qquad x_t^4 = \sum_{i=1}^{4} \xi_i x_{t-i}^4 + w_t^4$$

where $\boldsymbol{\phi}, \boldsymbol{\zeta}, \boldsymbol{\eta}$ and $\boldsymbol{\xi}$ are different model parameters and w_t^j for $j = 1, \ldots, 4$ are mutually independent Gaussian distributed white noise coefficients. The source matrix $\mathbf{X}$ is constructed by drawing k autoregressive processes, randomly drawn among the four, each of length L. If $k < N$ zero rows are inserted randomly such that $\mathbf{X} \in \mathbb{R}^{N \times L}$. The mixing matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ is, like previously, generated randomly from a Gaussian distribution. By multiplying the source matrix and the mixing matrix, the measurement matrix $\mathbf{Y}$ is simulated. The stochastic data set then consist of $\{\mathbf{Y}, \mathbf{X}, \mathbf{A}\}$.

One simulation of a stochastic data set is illustrated in figure 6.3. The illustrated data set is specified by $N = 5$, $M = 3$, $k = 4$ and $L = 1000$.

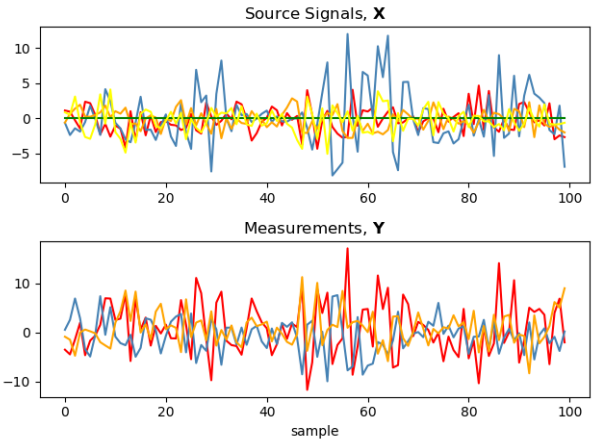

Figure 6.3: Visualization of the source signals $\mathbf{X}$ in comparison to the measurement signals $\mathbf{Y}$ from a stochastic data set specified by $N = 5$, $M = 3$, $k = 4$ and $L = 1000$. For simplicity only samples from $L = 0, \ldots, 100$ are visualized.

6.2.3 Error Measurement

To evaluate the estimates, and hereby the performance of each stage of the main algorithm, it is evident to look at the differences between the true and estimated matrices, mixing matrix $\mathbf{A}$ and source matrix $\mathbf{X}$ – which is possible due to the input data being simulated.

For this task the mean squared error (MSE) has been chosen. The MSE measures the average squared difference between some estimated value and the true value. For $\hat{\mathbf{g}}$ being the estimate of the vector $\mathbf{g}$ the MSE can be written as

$$\mathrm{MSE}(\mathbf{g}, \hat{\mathbf{g}}) = \frac{1}{T} \sum_{i=1}^{T} (g_i - \hat{g}_i)^2,$$

with T being the number of elements in the vector $\mathbf{g}$.

For this thesis the estimates form a matrix. Here the MSE is computed for each row, which for $\mathbf{X}$ is the estimate of one source signal. Then the resulting MSE is the average over all rows. For $\mathbf{X}, \hat{\mathbf{X}} \in \mathbb{R}^{N \times L}$ the MSE is written as

$$\mathrm{MSE}(\mathbf{X}, \hat{\mathbf{X}}) = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{1}{L} \sum_{j=1}^{L} (\mathbf{X}_{ij} - \hat{\mathbf{X}}_{ij})^2 \right).$$

Similarly, the MSE can be written for $\mathbf{A}, \hat{\mathbf{A}} \in \mathbb{R}^{M \times N}$.

The MSE is viewed as a measure of the quality of an estimator, in this case of how M-SBL and Cov-DL perform. The MSE considers both the variance among the estimated samples and the bias which is how far the average estimated value is from the true value [11, p.305]. Thus the larger MSE the more widely dispersed is

the estimate around the true parameter. In particular, $\hat{\mathbf{X}}_1$ is considered a better estimated than $\hat{\mathbf{X}}_2$ if $\text{MSE}(\mathbf{X}, \hat{\mathbf{X}}_1) < \text{MSE}(\mathbf{X}, \hat{\mathbf{X}}_2)$.

One drawback of applying the MSE as the performance measure has to be considered. The general MMV model, when both $\mathbf{A}$ and $\mathbf{X}$ are unknown, is in fact invariant toward mutual interchange of rows. Conditioned on similar interchange of rows in both $\mathbf{A}$ and $\mathbf{X}$. With respect to MSE this introduces a possibility of comparing the wrong parameters leading to a misleading MSE. However, the extent of this issue is assumed to be limited when the estimates of $\mathbf{A}$ and $\mathbf{X}$ are conducted and evaluated separately. Though this will be kept in mind when analysing the results.

6.3 Verification of Algorithms

In this section the implementations of Cov-DL and M-SBL are verified separately, based on the MSE between the true and the estimated model parameters. Remember that the segmentation stage is ignored as the simulated data form one single segment.

6.3.1 Test of Cov-DL

As seen from the flowchart 6.1 Cov-DL takes a measurement matrix $\mathbf{Y}$, N and k as input and returns an estimate $\hat{\mathbf{A}}$ of the mixing matrix $\mathbf{A}$. The Cov-DL algorithm is tested on the two simulations of the deterministic data, specified in section 6.2.1.

Cov-DL1

For measurement matrix $\mathbf{Y}$ specified by $N > \widetilde{M}$ and $k \leq \widetilde{M}$, implying Cov-DL1, the true and estimated values of the mixing matrix $\mathbf{A}$ are plotted in figure 6.4 ~~for visual comparison.~~ Note that each matrix is vectorized such that the corresponding entries are compared. The resulting $\text{MSE}(\mathbf{A}, \hat{\mathbf{A}})$ is seen below. As a reference the MSE is measured between $\mathbf{A}$ and a corresponding estimate being a zero matrix.

$$\text{MSE}(\mathbf{A}, \hat{\mathbf{A}}) = 1.74$$
$$\text{MSE}(\mathbf{A}, \mathbf{0}) = 1.40.$$

From figure 6.4 it is seen that the error of the estimate varies significantly for each entry. Though, the estimated values are seen to fall within the same range as the true values. Furthermore, the $\text{MSE}(\mathbf{A}, \hat{\mathbf{A}})$ is fairly small suggesting that the estimate is acceptable. However, a smaller MSE is obtained from the estimate being a zero matrix, which argues against $\hat{\mathbf{A}}$ being an acceptable estimate.

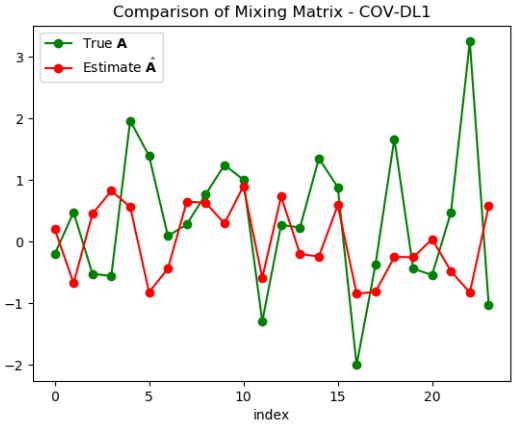

Figure 6.4: Estimated values of $\hat{\mathbf{A}}$ compared to the true values $\mathbf{A}$.

Cov-DL2

For the measurement matrix $\mathbf{Y}$ specified by $N \leq \widetilde{M}$, implying Cov-DL2, the true and estimated values of $\mathbf{A}$ are plotted in figure 6.5 ~~for visual comparison~~. Additionally, $\mathbf{A}_{\text{init}}$ is plotted in the same figure. The matrix $\mathbf{A}_{\text{init}}$ is the initial matrix provide to the optimization solver – a realization of a Gaussian matrix with zero mean and unit variance. The resulting MSE values is seen below, again the zero matrix estimate is used as a reference.

$$\text{MSE}(\mathbf{A}, \hat{\mathbf{A}}) = 3.00$$
$$\text{MSE}(\mathbf{A}, \mathbf{0}) = 0.90.$$

From figure 6.5 the estimate $\hat{\mathbf{A}}$ shows visual tendencies from the true $\mathbf{A}$. However, when it is compared to the initial guess of $\mathbf{A}$, $\mathbf{A}_{\text{init}}$, it is observed that the estimate $\hat{\mathbf{A}}$ have moved further away from the true $\mathbf{A}$ compared to $\mathbf{A}_{\text{init}}$. This suggests some flaw within the optimization process. By printing the convergence message from the used optimization solver, it is confirmed that the optimization process was found to be terminated successfully. With a current cost function value at 0.0 after 26 iterations. This suggests that a global minimum has been found, but the minimum, $\hat{\mathbf{A}}$, does not correspond to the true $\mathbf{A}$. To confirm this the following evaluations of the cost function was conducted.

$$\text{cost}(\hat{\mathbf{A}}) = 0.0$$
$$\text{cost}(\mathbf{A}_{\text{init}}) = 1.64$$
$$\text{cost}(\mathbf{A}) = 1.65$$

These evaluations ensure that the optimization solver did manage to find the solution that minimizes the cost function. By evaluating the cost function with respect to

the true $\mathbf{A}$ it is seen that the true mixing matrix is not a global minimizer to the optimization problem. This suggests that the optimization problem, derived in section 4.2.2, do not fulfil the purpose. However, it has to be mentioned that this is merely an interesting observation rather than a concluding result, as $\text{cost}(\mathbf{A}) = \text{cost}(\hat{\mathbf{A}})$ is not guaranteed.

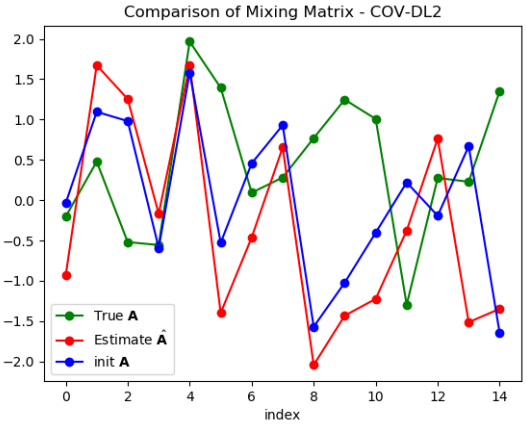

Figure 6.5: The initial $\mathbf{A}_{\text{init}}$ and the estimate $\hat{\mathbf{A}}$ compared to the true values $\mathbf{A}$.

Finally, an additional observation can be gained from both figure 6.4 and 6.5. It appears visually to that the issue of the model being invariant towards the order of the rows in the estimate $\hat{\mathbf{A}}$ is not an issue in this example. For instance neither of the three most negative values in $\mathbf{A}$ is found to be estimated at a different index. As such this potential flaw within the error measurement is not found to contribute to the insufficient results.

Summary with respect to verification of Cov-DL

From the above results it is found that the estimate $\hat{\mathbf{A}}$, especially within the Cov-DL2 branch, can not be considered as a valid estimate of the mixing matrix $\mathbf{A}$. The results suggest immediately that the flaw lies within the derivation of the cost function to the optimization problem. More specifically within the assumptions made throughout the derivation concerning the relation between $\mathbf{A}, \mathbf{D}$ and $\mathbf{U}$. However, in general three scenarios can be considered. The occurrence of a mistake with respect to the implementation, a misinterpretation of the source [4] leading to wrong implementation or lastly the method do not work as claimed by the source. In order to investigate the source of the insufficient results, the main attribute would be a thorough step by step evaluation of the implementation. Elements of special interest could be the found $\mathbf{D}$ relative to $\mathbf{A}$ and the amount of noise resulting from the rows of $\mathbf{X}$ being close to orthogonal. A different aspect could be the assumption of the

optimization problem (4.8) being convex without further investigation regarding the truthfulness of this assumption. The inconsistent results might suggest the optimization problem might not be convex. However, the fact that the optimization is terminated successfully with a cost equal to zero supports the existence of a global minimum.

Due to the time limitation of the thesis, the described investigation towards the source of the error is not conducted. It is concluded that the estimate of $\mathbf{A}$ is not valid. Hence, it will not be used as an input to the next stage of the main algorithm, M-SBL. This conclusion suggests that some alternative to the estimate must be considered. This is discussed further in section 6.4.

6.3.2 Test of M-SBL

From the flowchart 6.1 it seen that the M-SBL algorithm takes the estimated mixing matrix $\hat{\mathbf{A}}$ and measurement matrix $\mathbf{Y}$ as input. however, in order to not, let the performance of Cov-DL affect the result of M-SBL the true mixing matrix $\mathbf{A}$ is used as an input throughout this section, along with the corresponding $\mathbf{Y}$. The implementation is first tested on a deterministic data set specified by $M = N = k = 4$ and $L = 1000$. This result will serve as a reference, showing the best possible performance due to the system having an equal number of equations and unknowns, where a unique solution exists. The resulting estimate is seen in figure 6.6. It is seen that the source signals are estimated exact, with $\text{MSE}(\mathbf{X}, \hat{\mathbf{X}}) = 0$.

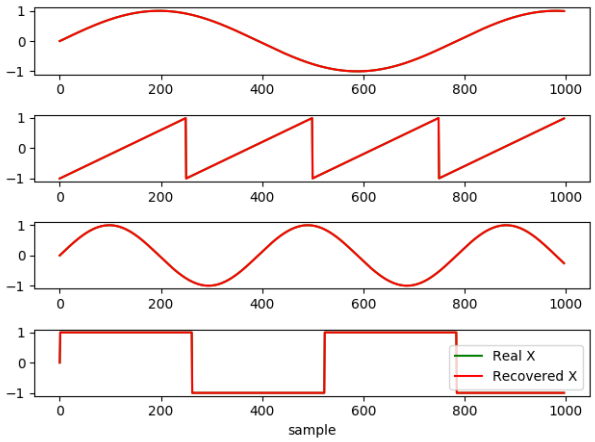

Figure 6.6: Estimated values of $\hat{\mathbf{X}}$ compared to the true values $\mathbf{X}$. From deterministic data set specified by $M = N = k = 4$ and $L = 1000$ given the true mixing matrix $\mathbf{A}$.

Now the desired case of $M < N$ is considered. Two tests are performed on the same two deterministic data sets, as used in the previous section, specified by $M = 3$, $k = 4$, $L = 1000$ and respectively $N = 5$ and $N = 8$.

The estimate $\hat{\mathbf{X}}$ is visualized in figure 6.7 and 6.8. The zero rows of the estimate $\hat{\mathbf{X}}$ is not visualized and therefore the figures do not visualize the exact localization of the source signals.

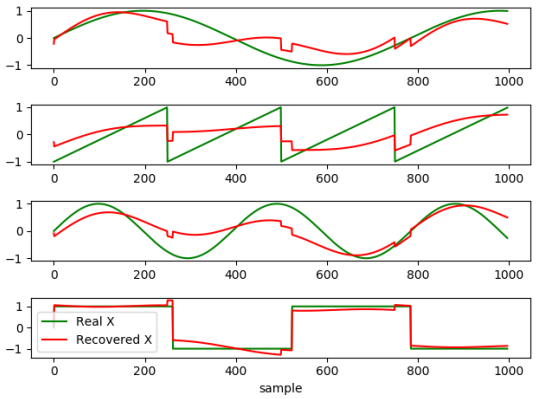

Figure 6.7: Estimated values of $\hat{\mathbf{X}}$ compared to the true values $\mathbf{X}$. From deterministic data set specified by $N = 5$, $M = 3$, $k = 4$ and $L = 1000$ and given the true mixing matrix $\mathbf{A}$.

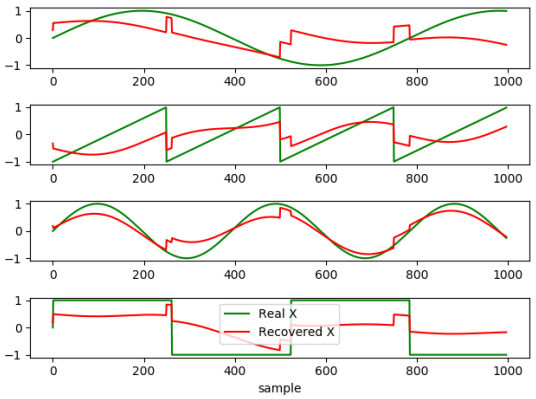

Figure 6.8: Estimated values of $\hat{\mathbf{X}}$ compared to the true values $\mathbf{X}$. From deterministic data set specified by $N = 8$, $M = 3$, $k = 4$ and $L = 1000$ and given the true mixing matrix $\mathbf{A}$.

The resulting MSE between the true $\mathbf{X}$ and the estimate $\hat{\mathbf{X}}$ from figure 6.7 with $N = 5$, becomes

$$\text{MSE}(\mathbf{X}, \hat{\mathbf{X}}) = 0.13.$$

From figure 6.7 it is seen that all four source signals are recovered at the right locations relative to the removal of the zero rows from $\mathbf{X}$ and $\hat{\mathbf{X}}$. As suggested by the achieved MSE the estimate is not exact, but it is clear that the estimates, by looking at figure 6.7, manage to follow the right pattern of the true signals.

The resulting MSE between the true $\mathbf{X}$ and the estimated $\hat{\mathbf{X}}$ from figure 6.8 with $N = 8$ thus more sparse, becomes

$$\text{MSE}(\mathbf{X}, \hat{\mathbf{X}}) = 0.162.$$

From figure 6.8 it is again seen that the source signals are recovered at the right locations. However visually the estimates appear slightly more imprecise. This indicates that the implementation of M-SBL manages to locate and estimate the source signals, however the increased zero rows improve the chance of dislocation and it decrease the accuracy of the estimate.

Possibilities of $N = k$

Due to the problem statement in chapter 2 it is an issue that k has to be known a priori, in order to estimate $\mathbf{A}$ and $\mathbf{X}$. A short discussion in section 5.2.1, describes

how k can be estimated within the M-SBL algorithm. However, one still needs to provide k in order to estimate $\mathbf{A}$, thus a qualified estimate of k can not be avoided.

Similar to k, the maximum number of active sources N is unknown in practice as it is described in chapter 1. The difference between k and N defines the number of zero rows in $\mathbf{X}$. During the estimation of $\mathbf{X}$ the localization of the non-zero rows are, in general, significant in order to minimize the MSE. However, the fact that the true N can not be known for EEG measurements weakens the argument for focusing on the localization rather than only focusing on the value estimation of the source signals. When considering the linear system, $\mathbf{Y} = \mathbf{AX}$, which the model is built upon, $\mathbf{Y}$ does not change by removing the zero rows of $\mathbf{X}$ and the corresponding columns in $\mathbf{A}$.

From this it can be argued that $N = k$ is a sufficient estimate of N. However, remember from chapter 5 that the existence of a solution is limited to $N = k \leq \widetilde{M}$.

Now consider the effect of letting $N = k$ within the M-SBL algorithm. Here it is only the estimation of the support set which is eliminated, as non zero rows will occur. Figure 6.9 shows the estimated source signals for a simulation of the deterministic data set now specified by $N = k = 4$, $M = 3$ and $L = 1000$.

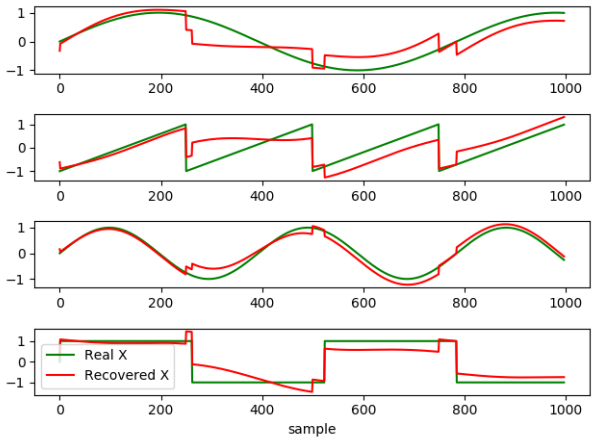

Figure 6.9: Estimated values of $\hat{\mathbf{X}}$ compared to the true values $\mathbf{X}$. From deterministic data $\mathbf{Y}$ specified by $N = k = 4$, $M = 3$ and $L = 1000$ and the true mixing matrix $\mathbf{A}$.

The resulting MSE becomes

$$\text{MSE}(\mathbf{X}, \hat{\mathbf{X}}) = 0.124$$

From the above discussion and the results in figure 6.9 it is confirmed that letting $N = k$ has no disadvantage when a correct localization of the source signal is not a priority. It is chosen that $N = k$ will be used throughout the thesis.

6.3.3 Test on Stochastic Data

The M-SBL algorithm is now tested on two stochastic data sets which resembles real EEG measurements. The first stochastic data set is simulated with specifications $N = k = 8$, $M = 6$, $L = 1000$. The resulting estimate is visualized in figure 6.10 and the MSE becomes

$$\text{MSE}(\mathbf{X}, \hat{\mathbf{X}}) = 1.1.$$

The second stochastic data set is simulated with specifications $N = k = 16$, $M = 6$ and $L = 1000$. This tests the capabilities of the implementation of M-SBL when the distance between M and N is enlarged. The performance relative to the relation between N and M is further investigated for the main algorithm in section 6.4. The resulting estimate is plotted in figure 6.11 and the MSE becomes

$$\text{MSE}(\mathbf{X}, \hat{\mathbf{X}}) = 3.652.$$

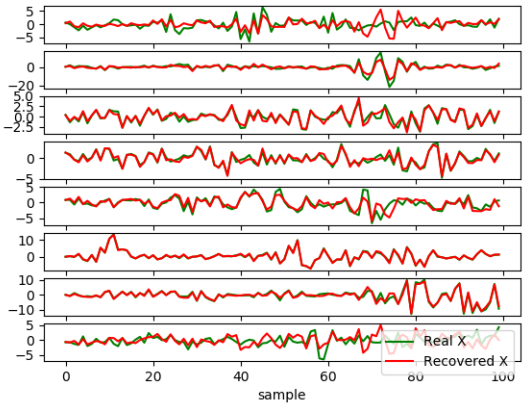

Figure 6.10: Estimated values of $\hat{\mathbf{X}}$ compared to the true values $\mathbf{X}$. From a stochastic data set specified by $N = k = 8$, $M = 6$ and $L = 1000$ and given the true mixing matrix $\mathbf{A}$.

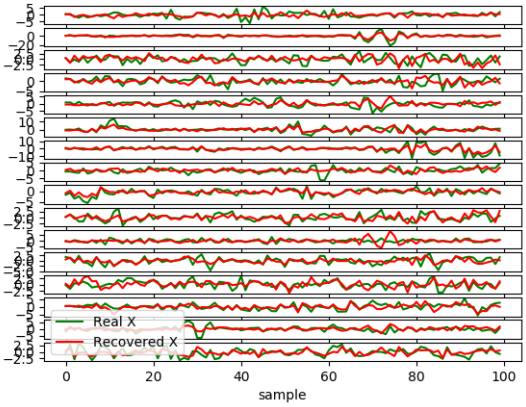

Figure 6.11: Estimated values of $\hat{\mathbf{X}}$ compared to the true values $\mathbf{X}$. From a stochastic data set specified by $N = k = 16$, $M = 6$ and $L = 1000$ and given the true mixing matrix $\mathbf{A}$.

From figure 6.10 it is visually confirmed that the implementation of M-SBL manages to sufficiently recover the stochastic source signals $\mathbf{X}$. Some source signals are nearly perfectly estimated while other are having minor differences. From figure 6.11 the same tendency is seen, though more visual flaws appears compared to figure 6.10. This result suggests that a bigger distance between M and N results in a worse performance from the M-SBL algorithm.

6.4 Test of the Main Algorithm

In this section the performance of the main algorithm is evaluated. That is the algorithm visualized in the flowchart 6.1 where the Cov-DL algorithm and the M-SBL algorithm are combined. However, as discussed due to the negative conclusion on the verification of implementation of Cov-DL an alternative to estimate of the mixing matrix has to be considered before the main algorithm can be tested.

6.4.1 Alternative to Estimate $\hat{\mathbf{A}}$

As concluded ~~does~~ the implementation of Cov-DL not provide a sufficient estimate of the mixing matrix $\mathbf{A}$. Therefore a different approach is necessary.

Replacing the insufficient estimate by a fixed estimate $\hat{\mathbf{A}}_{\text{fix}}$ is one immediately solution. By the term fixed one referrers to the estimate being manual chosen rather than being data dependent. This choice is supported by the observations from Cov-DL2 where $\mathbf{A}_{\text{init}}$ matrix provides an estimate which happens to be at least as good as the one provided by Cov-DL. Thus, the challenge is now to choose a fixed matrix for which its characteristics resemble those of the true mixing matrix. However, from chapter 1 it is clear that no specific characteristics of the mixing matrix are known, which supports the choice of a random matrix of Gaussian distribution or similar, as it was chosen for the initial guess $\mathbf{A}_{\text{init}}$. By randomly generating the fixed estimate~~d~~, an estimate is drawn from the specific distribution. Thus different realizations occur for every data set. From this perspective three fixed estimates of the mixing matrix are defined, by drawing each entry from a specified distribution:

$$\hat{\mathbf{A}}_{\text{uni}} \sim \mathcal{U}(-1, 1)$$

$$\hat{\mathbf{A}}_{\text{norm1}} \sim \mathcal{N}(0, 1)$$

$$\hat{\mathbf{A}}_{\text{norm2}} \sim \mathcal{N}(0, 2)$$

Note that the second matrix $\hat{\mathbf{A}}_{\text{norm1}}$ is generated the same way as the true mixing matrix of the stochastic data set, being a different realization. Thus, $\hat{\mathbf{A}}_{\text{norm1}}$ is expected to have the lowest MSE when compared to the true mixing matrix $\mathbf{A}$. However, it is of interest to investigate whether it is the best estimate of $\mathbf{A}$ which provide the best estimate of $\mathbf{X}$.

A different option regarding a choice for a fixed estimate $\hat{\mathbf{A}}_{\text{fix}}$ is to utilize the ICA algorithm, described in appendix C. By the ICA algorithm it is possible to solve the EEG inverse problem for both $\mathbf{A}$ and $\mathbf{X}$, in the case where $k \leq M$. Consider a simulation of a stochastic data set specified by $N = k = M$. Solving the system by ICA yields an estimate of $\mathbf{A}$. Now reduce the data set $\mathbf{Y}$ such that $M \leq k$. Similar the estimate of $\mathbf{A}$ is reduced by removing the same rows as in $\mathbf{Y}$. This yields an estimate $\hat{\mathbf{A}}_{\text{ICA}}$ which can be used as a fixed input to M-SBL along with the corresponding reduced $\mathbf{Y}$.

The four different fixed estimates $\hat{\mathbf{A}}_{\text{fix}}$ are tested on the stochastic data set specified by $M = 10$, $N = k = 16$ and $L = 1000$. As a reference the true mixing matrix $\mathbf{A}$ is included in the plot, to see the best possible MSE$(\mathbf{X}, \hat{\mathbf{X}})$. To get an average performance 50 different simulations are conducted with the same specifications. For each system $\mathbf{X}$ is estimated from each of the four fixed estimates[2] of $\mathbf{A}$, and the corresponding MSE is computed. The resulting average MSE$(\mathbf{A}, \hat{\mathbf{A}}_{\text{fix}})$ and MSE$(\mathbf{X}, \hat{\mathbf{X}})$ are visualized in figure 6.12, for each of the four $\hat{\mathbf{A}}_{\text{fix}}$.

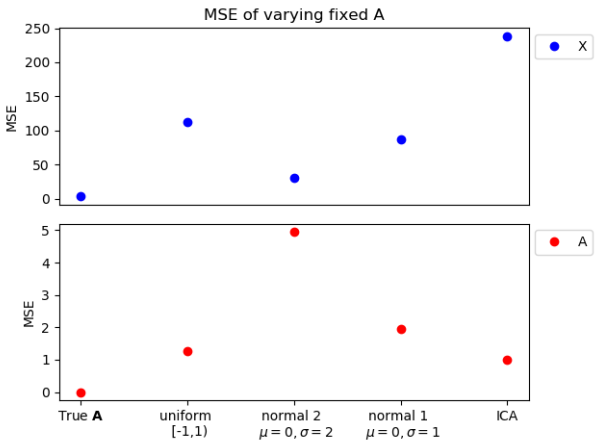

Figure 6.12: Average MSE$(\mathbf{A}, \hat{\mathbf{A}})$ value for each of the four fixed estimates $\hat{\mathbf{A}}_{\text{fix}}$ and the corresponding MSE$(\mathbf{X}, \hat{\mathbf{X}})$. From a stochastic data set specified by $M = 10$, $N = k = 16$ and $L = 1000$.

From figure 6.12 it is first of all seen that relation between the MSE of $\mathbf{A}$ and $\mathbf{X}$ do not behave as expected. The lowest MSE$(\mathbf{A}, \hat{\mathbf{A}}_{\text{fix}})$ results in the highest MSE$(\mathbf{X}, \hat{\mathbf{X}})$ and so forth. The lowest MSE$(\mathbf{A}, \hat{\mathbf{A}}_{\text{fix}})$ is achieved by using $\hat{\mathbf{A}}_{\text{ICA}}$, which confirms that the ICA algorithm manages to estimate $\mathbf{A}$ when $k \leq M$. However, as this do not result in the best estimate of $\mathbf{X}$ a different choice of $\hat{\mathbf{A}}_{\text{fix}}$ is still considered. The lowest MSE$(\mathbf{X}, \hat{\mathbf{X}})$ is achieved by use of $\hat{\mathbf{A}}_{\text{norm2}}$, which resulted in the largest MSE$(\mathbf{A}, \hat{\mathbf{A}}_{\text{fix}})$.

As the main interest in this thesis is to identify the active sources of EEG measurements, a low MSE$(\mathbf{X}, \hat{\mathbf{X}})$ is more desirable than a low MSE$(\mathbf{A}, \hat{\mathbf{A}}_{\text{fix}})$. Furthermore, a disadvantage of using $\hat{\mathbf{A}}_{\text{ICA}}$ is the limitations in practice when $k = M$ is not possible. From these observations a fixed estimate of the mixing matrix drawn from a normal distribution with mean 0 and variance 2, is chosen as the alternative estimate of $\mathbf{A}$. Thus is it concluded that $\hat{\mathbf{A}}_{\text{fix}} := \hat{\mathbf{A}}_{\text{norm2}}$ will replace the Cov-DL stage in the main algorithm, cf. figure 6.1, throughout the remaining parts of this thesis.

Due to the unexpected relation between MSE$(\mathbf{A}, \hat{\mathbf{A}}_{\text{fix}})$ and MSE$(\mathbf{X}, \hat{\mathbf{X}})$ an additional investigation is conducted. Figure 6.13 shows the MSE$(\mathbf{X}, \hat{\mathbf{X}})$ as a function

[2] Note that for each of the 50 repetitions four different realizations of $\hat{\mathbf{A}}_{\text{fix}}$ are fixed.

of the mixing matrix $\mathbf{A}$ with varying SNR. Specifically white noise $\mathbf{W}(\text{SNR})$ is added to the true mixing matrix depending on the desired SNR, such that

$$\hat{\mathbf{A}} = \mathbf{A} + \mathbf{W}(\text{SNR}).$$

The SNR is considered in the interval $[0.01, 2]$. For each SNR 100 simulations are performed an the average MSE$(\mathbf{X}, \hat{\mathbf{X}})$ is computed. Additionally figure 6.14 shows the corresponding average MSE$(\mathbf{A}, \hat{\mathbf{A}})$.

From figure 6.13 it is seen that MSE decreases as the SNR increases. This indicates as first expected that the better estimate of the true mixing matrix $\mathbf{A}$ the better estimate of source matrix $\mathbf{X}$. However, this is still a contradiction to the result seen in figure 6.12. This could possible correspond to true mixing matrix $\mathbf{A}$ being a Gaussian matrix with mean 0 and variance 1 for which Gaussian noise is added. The average MSE$(\mathbf{A}, \hat{\mathbf{A}})$ seen in figure 6.14 is seen to be far below 1 even for a large amount of noise, which is remarkably lower than the MSE values previously seen in figure 6.12.

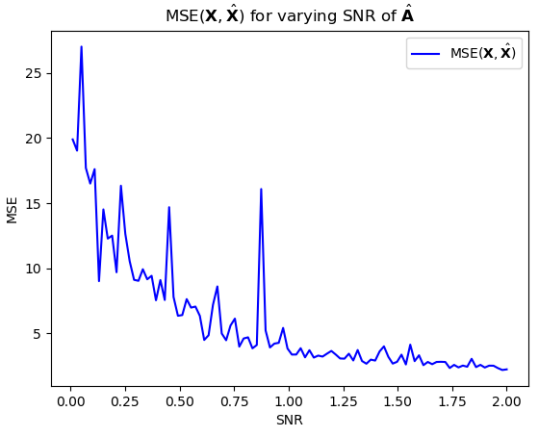

Figure 6.13: MSE$(\mathbf{X}, \hat{\mathbf{X}})$ estimated from stochastic data set specified by $M = 6$, $N = k = 8$ and $L = 1000$, as a function of SNR of given $\hat{\mathbf{A}}$.

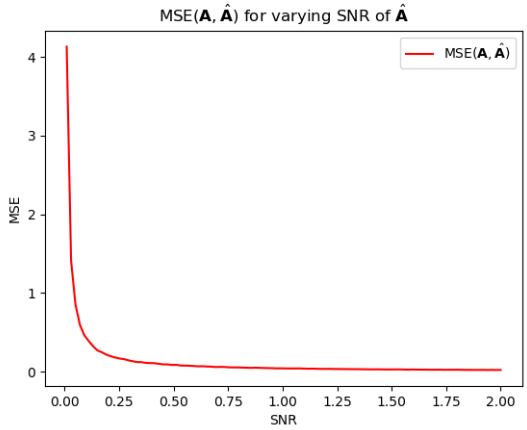

Figure 6.14: MSE$(\mathbf{A}, \hat{\mathbf{A}})$ where $\hat{\mathbf{A}}$ is a function of the SNR. $\hat{\mathbf{A}}$ correspond to $\hat{\mathbf{A}}$ used in figure 6.13.

6.4.2 Performance Test of Main Algorithm

In order to evaluate the performance of the main algorithm, tests are conducted on several simulated stochastic data sets with different specifications. The aim is to see how the relationship between N and M affect the performance, in other words how robust the algorithm is towards low-density measurements, $M < N$. The main algorithm is tested on simulated stochastic data sets specified by $M = 8$, $L = 1000$,

$k = N$ with N in the range $N = M+1, \ldots, 36$, as such $k < \widetilde{M}$ is withheld ensuring a solution. For each value of N ten different data sets are simulated and solved, and the average $\text{MSE}(\mathbf{X}, \hat{\mathbf{X}})$ are used as the result. The average $\text{MSE}(\mathbf{X}, \hat{\mathbf{X}})$ as a function of N are visualized in figure 6.15.

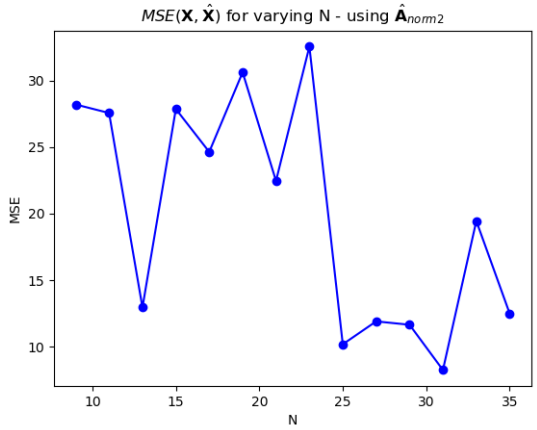

Figure 6.15: Visualization of average $\text{MSE}(\mathbf{X}, \hat{\mathbf{X}})$ of the main algorithm with simulated stochastic data sets specified by $M = 8$, $L = 1000$ and $k = N$ for $N = M+1, \ldots, 36$. The average is computed over ten repetitions for each N.

From figure 6.15 it is seen that the $\text{MSE}(\mathbf{X}, \hat{\mathbf{X}})$ lies in the interval $[5, 40]$ and no clear trend appears in the figure. This suggests that it is not a representative average behaviour which has been visualized. Thus, the test is repeated with 500 repetitions for each value of N. The new result of the $\text{MSE}(\mathbf{X}, \hat{\mathbf{X}})$ is seen in figure 6.16.

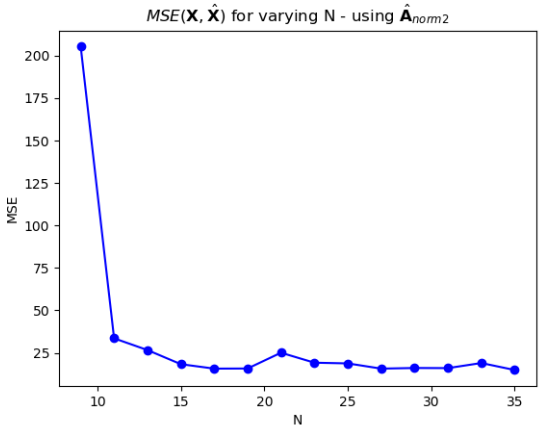

Figure 6.16: Average $\text{MSE}(\mathbf{X}, \hat{\mathbf{X}})$ of the main algorithm with simulated stochastic data sets specified by $M = 8$, $L = 1000$ and $k = N$ for $N = M+1, \ldots, 36$. The average is computed over 500 repetitions for each N.

Figure 6.16 confirms the result of the first test. However, the average MSE for $N = M + 1 = 9$ has increased significantly. To investigate this behaviour the corresponding box-plot presenting the 500 repetitions for $N = 9$ is visualized in figure 6.17, respectively with and without outliers. Here it is clear that the significant increase in average correspond to a few significant outliers.

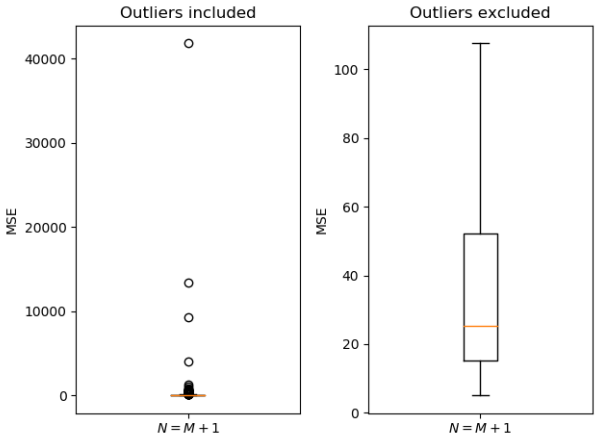

Figure 6.17: Left plot visualize the outliers from the box-plot of 500 repetitions for $k = N = 9$, $M = 8$ and $L = 1000$. Right plot visualize the values of the box-plot without outliers.

Over all, this suggests that the performance of the main algorithm is not affected by the relation between M and N. However, this assumption is counter intuitive and it is a contradiction to the results seen in figure 6.10 and 6.11, where the true $\mathbf{A}$ was utilised. Thus, the choice of the alternative estimate $\hat{\mathbf{A}}_{\text{fix}}$ might have influenced the results negatively. Furthermore, it is worth to notice the wide range of $\text{MSE}(\mathbf{X}, \hat{\mathbf{X}})$ suggesting a very high variance within the results, which add a certain unreliability to the results.

6.5 Summary

Through this chapter the implementation process has been described, followed by verification tests of the two main stages of the main algorithm, respectively the Cov-DL algorithm and the M-SBL algorithm.

From the test of M-SBL on stochastic data sets it was verified that the implementation provide the expected output, and from $\text{MSE}(\mathbf{X}, \hat{\mathbf{X}})$ and the corresponding visual comparison, the estimate was found to be sufficient. The verification of M-SBL was conditioned on the true mixing matrix $\mathbf{A}$ as input, to not let the precision of the estimate $\hat{\mathbf{A}}$ from Cov-DL affect the results. Furthermore, the possibilities of letting $k = N$ was discussed. Neither N nor k is known in practice, but one has to provide the best guess for both N and k to the algorithm in order to provide correspond-

ing number of source signals. By letting $k = N$ one only has to guess the maximal number of active sources and not the relation between active and non-active sources, which is considered easier. With respect to M-SBL, $k = N$ will reduce the chance of dislocation among the rows, which is seen as an advantage. Furthermore, tests on the deterministic data sets confirmed that the estimated active sources were not degraded. Thus, it is confirmed that letting $k = N$ is sufficient, and this will be used when testing the main algorithm on real EEG measurements.

From the verification tests of Cov-DL, providing the estimate $\hat{\mathbf{A}}$, it was found that the implementation of Cov-DL did not manage to provide a sufficient estimate. For the over-determined case it was confirmed that the optimization problem was terminated successfully, but the output did not comply with the theoretically expected result. Besides possible implementation errors this suggests that the theory provided by [6] was misinterpreted. This questions whether the degree of reproducibility of the paper has been sufficient. Due to the time scope of the thesis, this issue is not investigated further. However, as the estimate of $\mathbf{A}$ resulting from Cov-DL is crucial in order to estimate the source signals from real EEG measurements, it was chosen that the best possible alternative to the original estimate must be used, in order to pursue the remaining elements of the thesis. Then, the missing estimate must be taking into account when evaluating the final results.

Different suggestions for an alternative estimate of $\mathbf{A}$ was proposed and evaluated by the resulting $\text{MSE}(\mathbf{X}, \hat{\mathbf{X}})$. Here it was found that the fixed estimate $\hat{\mathbf{A}}_{\text{norm2}}$ generated from a normal distribution with mean 0 and variance 2 provided the best result, when tested on stochastic data sets resembling real EEG measurements.

Lastly, the performance of the main algorithm was tested on stochastic data sets. Here tests were performed on varying N in order to investigate performance relative to the relation between M and N. For each value of N, several repetitions were conducted and the average $\text{MSE}(\mathbf{X}, \hat{\mathbf{X}})$ was evaluated. The $\text{MSE}(\mathbf{X}, \hat{\mathbf{X}})$ was found to lie within an interval from 5 to 40, without any characteristic trend relative to the increasing N. From this is it concluded that the performance does not rely on the relation between N and M. Despite that this was indicated by the tests where the true $\mathbf{A}$ was utilized. Thus, the lack of a precise estimate of $\mathbf{A}$ do influence the final results.

Overall, the implementation of the resulting main algorithm is approved. Thus, the main algorithm is ready to be tested on real EEG measurements in order to evaluate the performance with respect to the problem statement of this thesis. These tests are specified and conducted in the next chapter.

However, by using a fixed estimate of the mixing matrix $\mathbf{A}$ which might be far from the true mixing matrix, the estimated source signals $\hat{\mathbf{X}}$ can in general not be considered reliable. Under different circumstances it would be preferable to investigate the issues of Cov-DL until a sufficient estimate was verified before testing the main algorithm on EEG measurements. Hence, the results to be obtained by

applying the main algorithm to EEG measurements will serve as investigation of the possible extent of the performance of the main algorithm, when $\hat{\mathbf{A}}$ is randomly generated rather than estimated from the given measurements.

Chapter 7

Test on EEG measurements

The main algorithm was implemented and tested on simulated data in chapter 6. In this chapter the main algorithm is tested on EEG measurements, for which it is intended. Two different approaches are considered with respect to evaluating the resulting estimates of the source signals, test by ICA comparison and an alpha wave analysis, respectively.

At first the provided data sets of EEG measurements are described. Followed by a test description and an analysis of the results for both of the evaluation approaches. Finally, a summary is provided to highlight the conclusions.

7.1 Data Description

For this thesis a data base of real EEG scalp measurements has been provided, from the department of electronic systems at Aalborg University. The data base consists of data sets of EEG measurements resulting from three test subjects. For each of three test subjects, two data set is provided. One where the test subject sits still with open eyes and one similar but with closed eyes, resulting in a data base with 6 data sets. For the measurements an EEG cap with 32 sensors measuring the scalp EEG signal with sample frequency at 512 Hz over a varying time period. Before the data base was provided each raw data set had undergone the following preprocessing. The data were bandpass filtered between 1 and 40 Hz. Then decomposed by ICA where the independent components related to eye activity or movement was removed. Thus, for every data set 27 sensors remains. That is 27 channels with names and position available in `EEG.chanlocs` structure. One data set then consist solely of the measurement matrix $\mathbf{Y} \in \mathbb{R}^{27 \times L}$. The data sets are specified in table 7.1.

EEG measurements		$M\_$	L	f_s	n_{seg}	L_s
1.	`S1_Cclean`	27	74161	512	144	516
2.	`S1_Oclean`	27	63245	512	123	514
3.	`S2_Cclean`	27	94918	512	185	513
4.	`S2_Oclean`	27	117900	512	230	512
5.	`S3_Cclean`	27	110060	512	214	514
6.	`S3_Oclean`	27	114065	512	222	513

Table 7.1: Specifications of the available data sets of EEG measurements, including specification of the segments resulting from segmentation into segments of length $t = 1$ seconds.

7.2 Test by ICA Comparison

The test procedure is now described through specification of the evaluation criteria and the practical implementation of the test. Remember the aim of the implemented main algorithm is to estimate the source matrix in the case where the number of active sources exceeds the number of sensors – $M < k \leq N$.

7.2.1 Performance Evaluation

From the description of ICA used on EEG measurements, cf. section 1.1.3, ICA is considered unreliable when using low-density EEG equipment where $M < 32$. For $M \geq 32$ ICA is currently considered the most reliable method for source signal recovery. However, note that the true number of sources is unknown thus there is always some unreliability to the result.

From the view that the sources found by ICA is the best estimate, it is possible to let that estimate serve as a reference for comparison of estimates recovered from $M < N$. In practice that is to perform ICA on a data set $\mathbf{Y} \in \mathbb{R}^{M \times L}$ resulting in $\hat{\mathbf{X}}_{\text{ICA}} \in \mathbb{R}^{N \times L}$ where $M = N$. Then a specific number of sensors are removed from the data set $\mathbf{Y}$ such that $M < N$, the sources signals are now estimated by the main algorithm resulting in $\hat{\mathbf{X}}_{\text{main}} \in \mathbb{R}^{N \times L}$. The performance of the main algorithm is then to be measured by comparison to the $\hat{\mathbf{X}}_{\text{ICA}}$. The question is here whether the main algorithm manage to find the same active sources as ICA, but for $M < N$.

In appendix C ICA is described theoretically and the applied algorithm is verified on simulated data without noise. It was found that ICA manages to estimate $\mathbf{X}$ almost exact, when $M = N = k$. Furthermore, it is seen that for $k < N$ ICA manages to estimate the zero rows ~~as~~ correctly. This supports that the estimate by ICA can serve as a reference.

To compare the two estimates the MSE, cf. section 6.2.3, is used. However, an issue arises due to the fact that ICA do not manage to localize each of the found sources. That is the order of the rows of $\hat{\mathbf{X}}_{\text{ICA}}$ does not necessarily correspond to

the true $\mathbf{X}$. Furthermore, the ICA algorithm is invariant towards the phase and the amplitude. This must necessarily worsen the resulting MSE.

This issue is covered in appendix C.3. Here a function is considered, which manage to pair and fit the rows with the lowest mutual MSE and then arrange the rows of $\hat{\mathbf{X}}_{\text{ICA}}$ such that $\text{MSE}(\mathbf{X}, \hat{\mathbf{X}}_{\text{ICA}})$ is minimised. The fitting consists of a possible phase shift and scaling of the amplitude. The right optimal fit was found through a brute-force search, however this is impossible as the possible number of combinations increases as k increases. This suggests the definition of an optimization problem minimizing the resulting MSE with respect to the combination of row indexes, possible phase and corresponding amplitude scaling. Unfortunately, a successful optimization was not achieved within the time scope of this thesis. Thus, the fitting process is not applied to the results achieved from the EEG measurements in this chapter. This factor must be taken into account when evaluating the results.

Consider again the resulting $\text{MSE}(\hat{\mathbf{X}}_{\text{ICA}}, \hat{\mathbf{X}}_{\text{main}})$. To evaluate further on the question whether the same sources have been found a tolerance for the MSE is introduced. With $\text{MSE}(\hat{\mathbf{X}}_{\text{ICA}}, \hat{\mathbf{X}}_{\text{main}})$ being an average over the MSE of each row within one segment a low value indicate that main part of the rows makes an estimate similar the estimate from ICA. From this perspective a tolerance for $\text{MSE}(\hat{\mathbf{X}}_{\text{ICA}}, \hat{\mathbf{X}}_{\text{main}})$ decides whether the same sources are achieved with success. The tolerance is set to 5 due to previous observations with respect to the simulated data. Especially figure 6.11 indicate that an MSE below 5 is achievable for a system where $M << N$ with use of true $\mathbf{A}$. It could be argued that the tolerance should be increased as the estimate of $\mathbf{A}$ is not expected to be nearly as good. However, this could give a distorted image of the results.

7.2.2 Test Setup

The test setup is visualized in figure 7.1 by a flow diagram, showing the essential steps of the test.

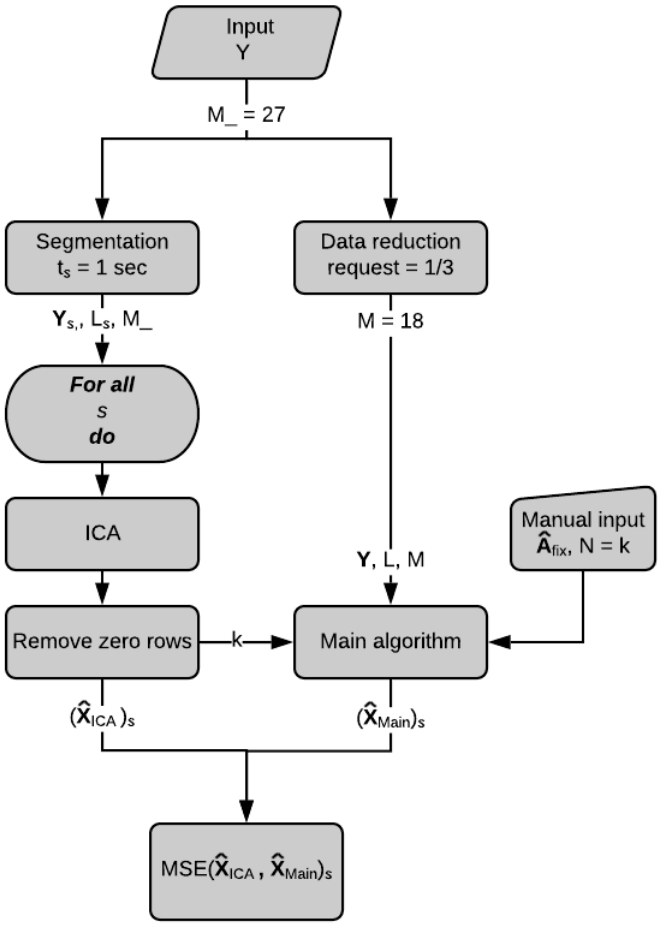

Figure 7.1: Flow diagram for visualization of the test procedure for one data set. Example given for $M < N$ where `request = 1/3` result in $M = 18$.

In the flow diagram the two estimation processes are seen to run parallel but taking the same input. Prior to the application of ICA, the input is divided into segments. That is the same segmentation as inside the main algorithm, cf. section 6.1. The size of the segments is defined due to the expected stationarity of the sources. As described in the motivation chapter 1 sources are stationary if you look at sufficiently small intervals. Segments at $t = 1$ second is chosen from the assumption that the brain activity can be assumed stationary within the short time interval. Furthermore, one must take in mind that shorter time interval lead to more segments and therefore a higher computational complexity. After the segmentation the ICA is applied to every segment s, returning $\hat{\mathbf{X}}_{\mathrm{ICA}s} \in \mathbb{R}^{M \times L_s}$. From appendix C.3 it is seen that ICA manage to estimate the non-active sources by zero rows, when no noise is present. When ICA is applied to the EEG measurements noise is expected. Thus the non-active sources is defined by the average amplitude being within a tolerance

interval around zero, defined by tol = $[10E-03, -10E-03]$. When the non-active sources are identified, they are removed and the resulting estimate is reduced to $\hat{\mathbf{X}}_{\text{ICA}s} \in \mathbb{R}^{k \times L_s}$. The found number of active sources k is then given as input to the main algorithm where $k = N$. In parallel to the ICA process the input data is reduced as specified in the previous section. Then the main algorithm is applied to the reduced data set. Within the main algorithm the data are like wise divided in segments and an estimate $\hat{\mathbf{X}}_{\text{main}s} \in \mathbb{R}^{k \times L_s}$ is returned. Note that $\hat{\mathbf{A}}_{\text{fix}}$ is given as a manual input, replacing the Cov-DL algorithm as concluded in chapter 6. At the end the resulting two estimates have the same dimensions which allow for $\hat{\mathbf{X}}_{\text{main}s}$ to be evaluated with respect to $\hat{\mathbf{X}}_{\text{ICA}s}$ by the MSE.

The described test is performed on the following three cases,

- **Case 0**: $M = N$ to see the best possible result achieved by the main algorithm.
- **Case 1**: $M < N$ every third sensor is removed.
- **Case 2**: $M << N$ every second sensor is removed.

7.3 Results

For each case the test is performed on all the data sets specified in table 7.1. The results are visualized for one data set to get an visual understanding. Lastly, the results of all three data sets are compared in a table.

The results are plotted for data set S1_Cclean. The data set consist of 144 time segments with $L_s = 516$ samples and $M\_ = 27$ sensors.

7.3.1 Case 0, $M = N$

ICA is applied on $\mathbf{Y}_s$ specified by $M\_ = 27$ and $L_s = 516$. The main algorithm is applied on $\mathbf{Y}_s$ without any reduction hence specified by $M = 27$ and $L_s = 516$, given $\hat{\mathbf{A}}_{\text{fix}}$ and $N = k$ provided from ICA. Figure 7.2 show $\text{MSE}\left(\hat{\mathbf{X}}_{\text{main}}, \hat{\mathbf{X}}_{\text{ICA}}\right)$ for all segments s. Figure 7.3 show the same plot but the y-axis is specified to the interval $[-10, 50]$ for better visualization. Furthermore, the MSE tolerance = 5 is plotted, indicting for each segment whether the estimate $\hat{\mathbf{X}}_{\text{main}}$ is sufficiency close to $\hat{\mathbf{X}}_{\text{main}}$. For a majority of the segments the MSE lies under the tolerance, but single outliers appears for which the MSE of the segment is significantly increased. Taking the average over all segments the average achieved MSE is 5.17.

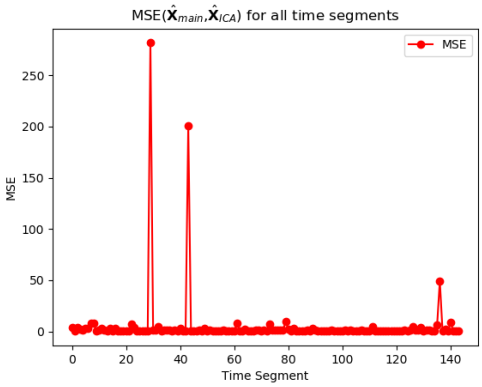

Figure 7.2: MSE$\left(\hat{\mathbf{X}}_{\text{main}}, \hat{\mathbf{X}}_{\text{ICA}}\right)$ for all $n_{\text{seg}} = 144$ segments.

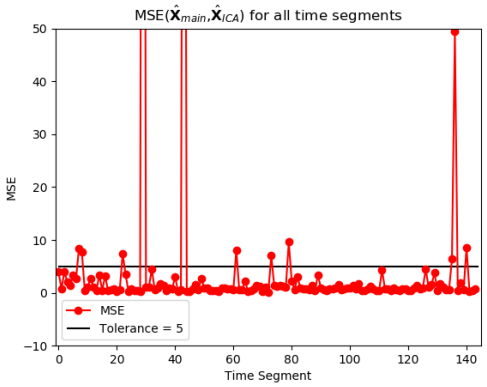

Figure 7.3: MSE$\left(\hat{\mathbf{X}}_{\text{main}}, \hat{\mathbf{X}}_{\text{ICA}}\right)$ for all $n_{\text{seg}} = 144$ segments. Plotted only for the y-axis interval $[-10, 50]$ for better visualization.

To investigate the behavior of a single segment figure 7.4 show the MSE value computed for each row of the two estimates of a specific segment. That is MSE($\hat{\mathbf{X}}_{\text{main}_i}$, $\hat{\mathbf{X}}_{\text{ICA}_i}$) for every row $i = 1, \ldots, k$ in time segment $s = 5$. Additionally figure 7.5 show and compare the corresponding estimates for four random chosen sources. This allows for visual comparison of the estimates relative to the corresponding MSE value seen in figure 7.4. Note that for better visual comparison each plotted row of $\hat{\mathbf{X}}_{\text{ICA}}$ is scaled with respect to the max value of the corresponding row in $\hat{\mathbf{X}}_{\text{main}}$. From figure 7.4 it is seen that the estimate of each source result in a relative low MSE. This indicates that the main algorithm has managed to estimate the same source as the ICA algorithm. In contradiction to this, figure 7.5 do not confirm that the estimates are close, as generally the two signals ~~in one plot~~ does not follow the same trend.

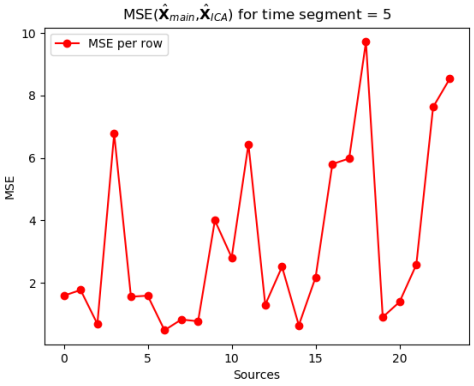

Figure 7.4: $\text{MSE}\left(\hat{\mathbf{X}}_{\text{main}_i}, \hat{\mathbf{X}}_{\text{ICA}_i}\right)$ for every row $i = 1, \ldots, k$ in time segment $s = 5$.

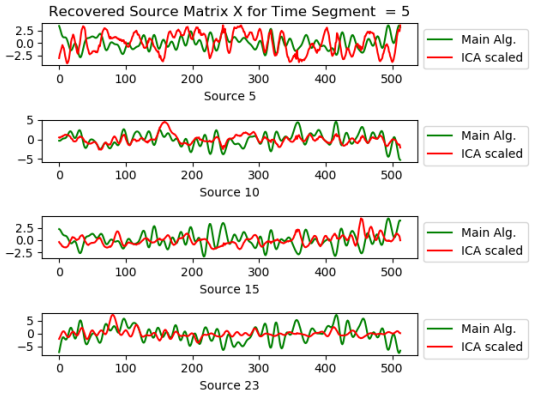

Figure 7.5: Plot comparing four random chosen rows from $\hat{\mathbf{X}}_{\text{main}}$ and $\hat{\mathbf{X}}_{\text{ICA}}$ from time segment $s = 5$ with $M = N$ and $k = 23$. Note $\hat{\mathbf{X}}_{\text{ICA}}$ is scaled for better visualization.

The test is repeated for every data set, and the results are summarized in table 7.2. In general a low MSE is achieved in average over all segments of one data set relative to the tolerance. And the corresponding percentage is likewise relative high, with an average at 83%. A single result is seen to deviate from the tendency, which is the data set of test subject 3 with closed eyes. Here a significant high average MSE value is found, indicating a majority of the segment has resulted in a significantly high MSE, while a percentage of 63% was below the tolerance. In chapter 6 it is found that the main algorithm was capable of providing an almost exact estimate for $M = N$ when the true $\mathbf{A}$ is provided. Thus, it is expected that the general performance is decreased in this case where the true $\mathbf{A}$ is unknown and $\hat{\mathbf{A}}_{\text{fix}}$ is given.

The achieved results will serve as reference when analyzing the results of the following cases where the main algorithm is applied on data set with reduced number of sensors compared to the original data set.

Case 0 $M = N$	Test subject 1		Test subject 2		Test subject 3	
	Open	Close	Open	Close	Open	Close
Average MSE()	2.913	5.172	1.572	15.06	4.753	19.44
Segments below tolerance in %	91	92	98	61	87	63

Table 7.2: Summarized results for case 0. Test is performed on the every data set.

7.3.2 Case 1, $M < N$

The main algorithm is applied on $\mathbf{Y}_s$, where the number of sensors is reduced by one-third. As such the main algorithm is applied on $\mathbf{Y}_s$ specified by $M = 18$ and $L_s = 516$, given $\hat{\mathbf{A}}_{\text{fix}}$ and $N = k$ provided from ICA. ICA is applied on the original data set with segments $\mathbf{Y}_s$ specified by $M\_ = 27$ and $L_s = 516$. The viewed plots correspond to those of case 0, but for the reduced number of sources $M < N$, hence detailed plot description is omitted here.

From figure 7.6 and 7.7 it is seen that a majority of the segments have MSE value close to the tolerance, but the number of outliers have increased compared to case 0. This indicates that for an increased number of segments the main algorithm do not manage to estimate enough sources sufficiently in order to stay below the tolerance. Taking the average over all segments the average achieved MSE is 5.35.

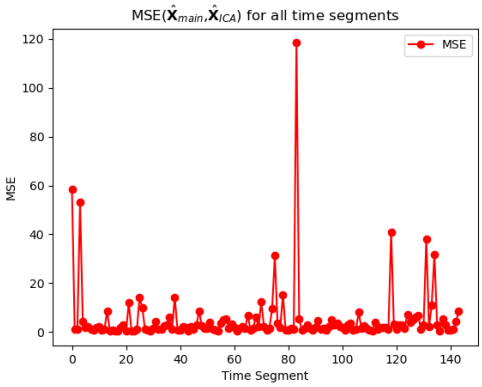

Figure 7.6: $\text{MSE}\big(\hat{\mathbf{X}}_{\text{main}}, \hat{\mathbf{X}}_{\text{ICA}}\big)$ for all $n_{\text{seg}} = 144$ segments.

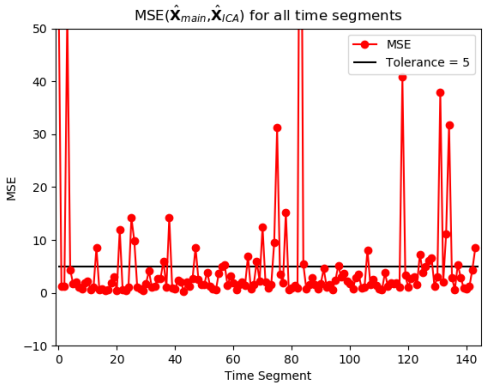

Figure 7.7: $\text{MSE}\big(\hat{\mathbf{X}}_{\text{main}}, \hat{\mathbf{X}}_{\text{ICA}}\big)$ for all $n_{\text{seg}} = 144$ segments. Plotted only for the y-axis interval $[-10, 50]$ for better visualization.

From figure 7.8 and 7.9 showing the results of segment 5, it is seen that the MSE for each source has increased slightly compared to case 0. This supports the observation from figure 7.7.

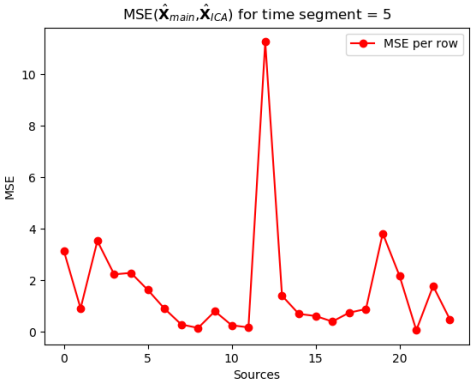

Figure 7.8: $\text{MSE}(\hat{\mathbf{X}}_{\text{main}_i}, \hat{\mathbf{X}}_{\text{ICA}_i})$ for every row $i = 1, \ldots, k$ in time segment $s = 5$.

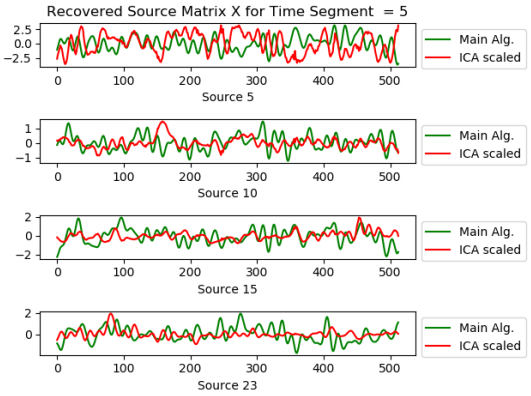

Figure 7.9: Plot comparing four random chosen rows from $\hat{\mathbf{X}}_{\text{main}}$ and $\hat{\mathbf{X}}_{\text{ICA}}$ from time segment $s = 5$ with $M = N$ and $k = 23$. Note $\hat{\mathbf{X}}_{\text{ICA}}$ is scaled for better visualization.

The test is repeated for every data set, and the results are summarized in table 7.3. Comparing table 7.3 to table 7.2, it is seen that the percentage of segments below the tolerance are decreasing, with the majority being close to 50%. This is roughly indicating that half of the time the main algorithm do not manage to provide a sufficient estimate when $M = 2/3N$. Furthermore, both the average MSE and the corresponding percentages appear fluctuating relative to case 0 indicating some unreliability in the results.

Case 1 $M < N$	Test subject 1		Test subject 2		Test subject 3	
	Open	Close	Open	Close	Open	Close
Average MSE()	9.79	5.351	13.89	15.13	6.25	18.21
Segments below tolerance in %	53	80	66	46	77	48

Table 7.3: Summarized results for case 1. Test is performed on the every data set.

7.3.3 Case 2, $M << N$

The main algorithm is applied on $\mathbf{Y}_s$, where the number of sensors is reduced to half. As such the main algorithm is applied on $\mathbf{Y}_s$ specified by $M = 13$ and $L_s = 516$, given $\hat{\mathbf{A}}_{\text{fix}}$ and $N = k$ provided from ICA. ICA is applied on the original data set with segments $\mathbf{Y}_s$ specified by $M\_ = 27$ and $L_s = 516$. The viewed plots correspond to those of case 0 and case 1, but for further reduce number of sources $M << N$, hence detailed plot description is omitted.

From figure 7.10 and 7.11 it is seen that the MSE value for each segment is more

widely scatted around the tolerance, compared to case 0 and 1. Outliers, where the MSE value has increased significantly, do also occur similar to case 1. This indicates that the performance of the main algorithm has decreased further, compared to case 1. Taking the average over all segments the average achieved MSE is 11.36.

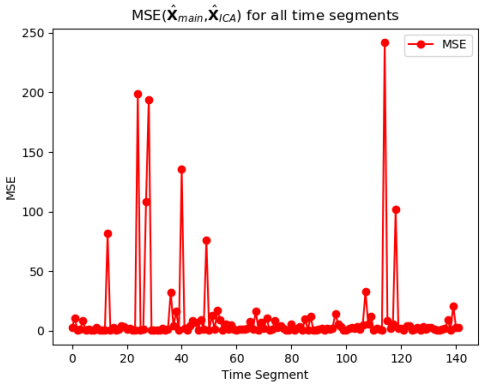

Figure 7.10: $\text{MSE}\left(\hat{\mathbf{X}}_{\text{main}}, \hat{\mathbf{X}}_{\text{ICA}}\right)$ for all $n_{\text{seg}} = 144$ segments.

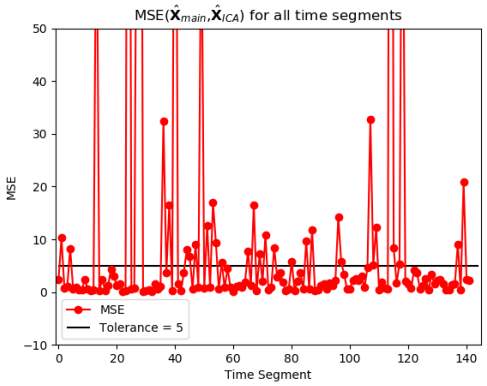

Figure 7.11: $\text{MSE}\left(\hat{\mathbf{X}}_{\text{main}}, \hat{\mathbf{X}}_{\text{ICA}}\right)$ for all $n_{\text{seg}} = 144$ segments. Plotted only for the y-axis interval $[-10, 50]$ for better visualization.

The above indication is supported by figure 7.12 and 7.13 showing an general increase in MSE. However, segment 5 makes a fairly good example as the majority of the sources have achieves a MSE below the tolerance of 5. From figure 7.13 the increased MSE do not appear visually compared to either case 1 or case 0.

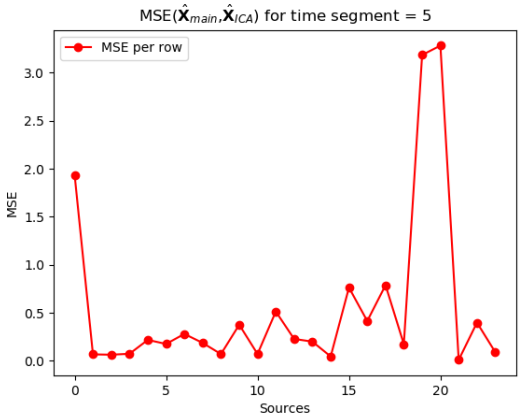

Figure 7.12: $\text{MSE}\left(\hat{\mathbf{X}}_{\text{main}_i}, \hat{\mathbf{X}}_{\text{ICA}_i}\right)$ for every row $i = 1, \ldots, k$ in time segment $s = 5$.

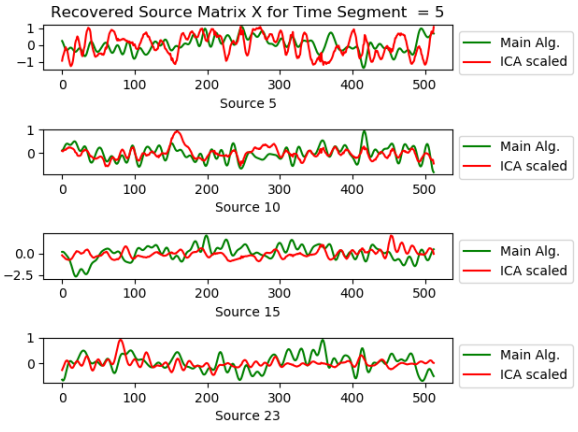

Figure 7.13: Plot comparing four random chosen rows from $\hat{\mathbf{X}}_{\text{main}}$ and $\hat{\mathbf{X}}_{\text{ICA}}$ from time segment $s = 5$ with $M << N$ and $k = 23$. Note $\hat{\mathbf{X}}_{\text{ICA}}$ is scaled for better visualization.

The test is repeated for every data set, and the results are summarized in table 7.4. Comparing table 7.2 to table 7.3 it is generally seen that the percentage of segments below the tolerance are not decreased but improved. Though, without getting close to the tendency from case 0. Furthermore, the average MSE have not increased remarkably compared to case 1. As such the performance of the main algorithm in case 2 is in general not found to be worse than for case 1. However, a clear improvement is not seen either.

Case 2 $M << N$	Test subject 1		Test subject 2		Test subject 3	
	Open	Close	Open	Close	Open	Close
Average MSE()	8.378	11.36	19.58	13.11	13.99	11.96
Segments below tolerance in %	75	74	42	72	69	69

Table 7.4: Summarized results for case 2. Test is performed on the every data set.

7.3.4 Summary of Results

The main algorithm has been tested on six data sets of EEG measurement, for a varying relation between the number of sensors and sources, case 0, 1 and 2 respectively. When the number of sensors is reduced with respect to the number of sources to be found, a significant decrease in performance was found. When comparing case 0 and 1. However, a corresponding decrease of performance was not found when further sensors were removed when comparing case 1 and 2.

From the conclusions made in chapter 6 it was not expected that the main algorithm would provide successful results, without estimating $\mathbf{A}$ form the data. The results of case 0 do however indicate a solid estimate provided by the main algorithm, with an average percentage of successfully estimated segments at 83%.

Furthermore, it is worth to note that the resulting MSE values have potential for improvement when considering optimization of the source localization of the ICA estimate, cf. appendix C.3.

7.4 Alpha Wave Analysis

As mentioned in chapter 1 brain signals can be classified into four groups according to the dominant frequency [26]. It is known that when a person closes the eyes, when relaxing, the amount of alpha frequency raises and become the dominant frequency. The provided EEG measurements consist of measurements from a test subject with both open and closed eyes. Hence, it would be interesting to investigate this relation between the alpha frequency for open and closed eyes. The interesting part is then to

compare the relation achieve from the provided EEG measurements and the sources signals estimated by the main algorithm.

With a test of this kind, it is possible to evaluate the recovered source signals from a different perspective. Here the objective is first of all to see the behavior with respect to the frequency, expected by the theory. Next it is interesting to investigate the aspect of analysis performed on EEG level versus analysis performed on source level, as discussed in chapter 1.

7.4.1 Test Setup

For this comparison the data sets of test subject 1, `S1_OClean` and `S1_CClean`, EEG measurements of open and closed eyes respectively, will be used. It is expected that the power within the alpha frequency band is highest for the closed eyes data set, `S1_CClean`. To compare the amount of alpha frequency in the two data sets, a bandpass filter is used to isolate the alpha frequencies. To perform the filtering a bandpass Butterworth filter of order 5 with cut-off frequencies 8 Hz and 13 Hz will be applied. The filtering is performed in the time domain to both the EEG signals and the source signals recovered from the main algorithm. The filtering process is illustrated in figure 7.14. In the illustrated example only one source signal is investigated in both time and frequency domain, where the fast Fourier transformation (FFT) is applied [22, Chapter 9]. The source signal of interest was recovered from the closed eyes data set `S1_CClean` from time segment 15. The system specification used to recover the source signal was $M = 27$ and $k = 14$.

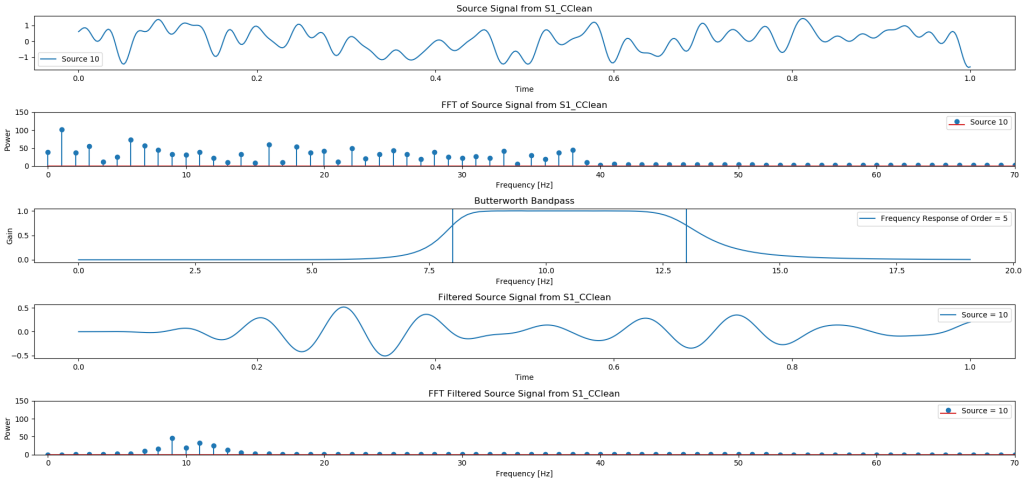

Figure 7.14: Time domain and frequency plot of a recovered source signal, filtered and non-filtered, from the time segment 15.

The first plot in figure 7.14 is the recovered source signal in the time domain. The next plot is the same source signal but transformed to the frequency domain by

the FFT. The plot has been scaled to only show the frequencies from 0-70 Hz and the power from 0 to 150. The third plot illustrates the frequency response of the bandpass Butterworth filter with order 5. The vertical blue lines illustrate the cut-off frequencies at 8 Hz and 13 Hz. Plot number four is the recovered source signal filtered by the bandpass Butterworth filter, plotted in the time domain. The last plot is the filtered source signal plotted in the frequency domain. This verifies that the signal of interest has been filtered according to the alpha band. From the filtered source signal in the time domain, the signal resemble the alpha wave as seen in figure 1.2.

The filtering process is applied to 100 time segments of both the closed eyes and open eyes for respectively the EEG measurements and the recovered source signals. Note that for each time segment all present source signals or sensor measurements have been summed such that only one signal resembles each time segment. Then for each time segment the relation between closed and open eyes is computed, with respect to power within the alpha band. The relation is defined as

$$\text{Relation} = \frac{C}{O},$$

where C is the average power from closed eyes, and O is the average power from the open eyes segment. This is done for both the EEG measurements and the recovered source signals. By this it is possible to compare the relation found on source level and the relation seen on EEG level.

7.4.2 Results

Figure 7.15 shows an example of one time segment. To the left is the power spectrum of the filtered EEG measurements plotted for open and closed eyes respectively. The resulting relation between the two is 1.15. To the right is the power spectrum of the filtered source signals. Likewise for open and closed eyes respectively. The resulting relation between open and closed eyes is here 1.41.

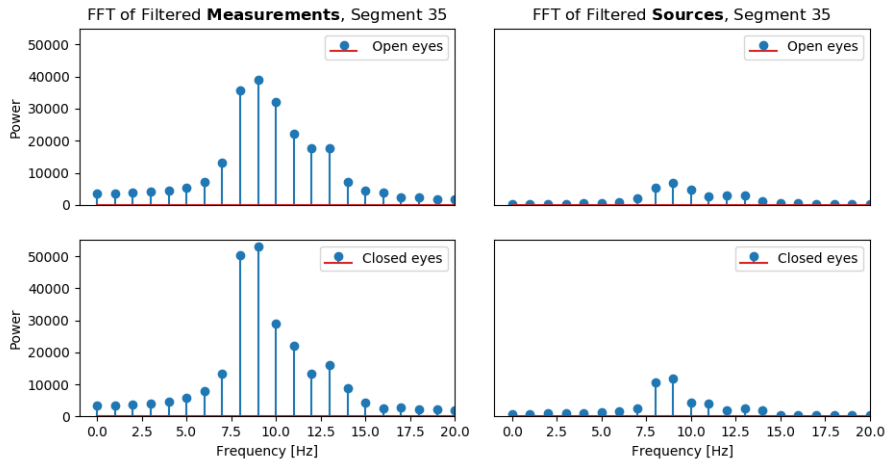

Figure 7.15: power spectrum of the filtered EEG measurements and source signals of time segment 35 for open and closed eyes data set of test subject 1.

By observing figure 7.15 it is seen, for the specific segment, that the power within the alpha band is significantly larger within the EEG measurements compared to the sources. Furthermore, is it seen for both the EEG measurements and the source signals that the power has increased from open to closed eyes. Considering the calculated relations is it seen the that biggest increase in power is found on source level. This behavior does support the theory, however the result of a single segment is not sufficient to draw any conclusion.

Figure 7.16 and 7.17 illustrate the C/O relation computed for the 100 time segments, of the EEG measurements and source signals respectively. The horizontal line in the plots marks the 1/1 relation. As such the segments where the highest power was found for closed eyes lies above the line - supporting the theory. Opposite the segments with least power found for closed eyes lies below the line.

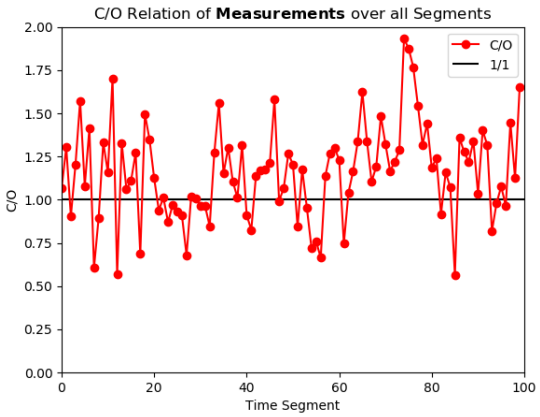

Figure 7.16: The C/O relation for EEG measurements for 100 time segments. The average C/O over all segments is 1.16.

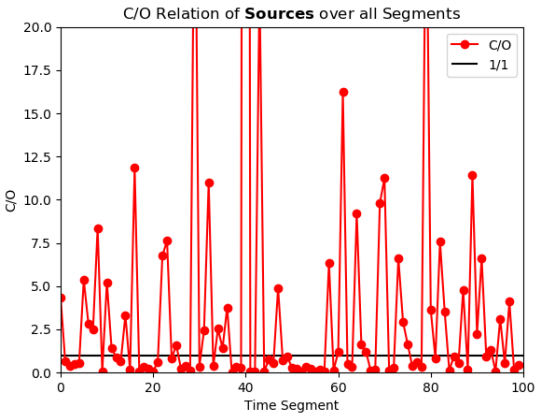

Figure 7.17: The C/O relation of source signals for 100 time segments. The average C/O over all segments is 2.01.

From the figures it is clear the behavior seen from the example of segment 35, is not a continuous behavior. It is seen both on EEG level and source level that relation scatters around the horizontal line, indicating that the relation is not stationary over time. On figure 7.17 is it seen that the C/O relation range from near zero to beyond 20 for a few segments, indicating a significant change in power compared to figure 7.16. With 57 out of 100 segments lying below the horizontal line the behavior is considered more or less random. From these observations the expected behavior was not found. This does support the earlier findings with respect to the main algorithm, indicting a significant unreliability to the result.

With respect to the method for computing the C/O relation it could be considered whether computing the relation for every segment is the right choice. One could argue that summing the power over all segments for respectively open and closed eyes and then compute the C/O relation would yield a different result.

Chapter 8

Estimation of the Number of Active Sources

In this chapter the issue of unknown number of active sources k is considered. The aim is to investigate the possibility of identifying an estimation of a non-active source signal from $\hat{\mathbf{X}}_{\text{main}}$ when the true k is not provided to the main algorithm. Instead of providing the true k one let $k = N$. As such one ask the main algorithm for N active source signals, but there are only $k < N$ active source signal. At first the possibilities are investigated on synthetic data set, cf. section 6.2 and afterwards on a real EEG data set.

8.1 Empirical Test on Synthetic Data

Figure 8.1 visualizes the estimate $\hat{\mathbf{X}}_{\text{main}}$ given $k = N$ and the true $\mathbf{A}$, resulting from a stochastic data set specified by $M = N = 8$, $k = 4$ and $L = 1000$. As seen in section 6.3.3 the case of $M = N$ should be solved almost exact by the M-SBL algorithm with true $\mathbf{A}$ given. From the figure it is seen that the estimates of the zero rows have amplitudes close to zero. This distinguishes them from the remaining source estimates which are seen to be almost exact. Due to the estimates of the zero rows being this close to zero they do not affect the MSE. Thus, the MSE do not indicate flaws within the estimate. Furthermore, it is seen that the estimates of the zero rows form a scaled copy of one of the exact estimates. These observations indicate a potential for distinguishing the estimates of zero rows and hence determine k.

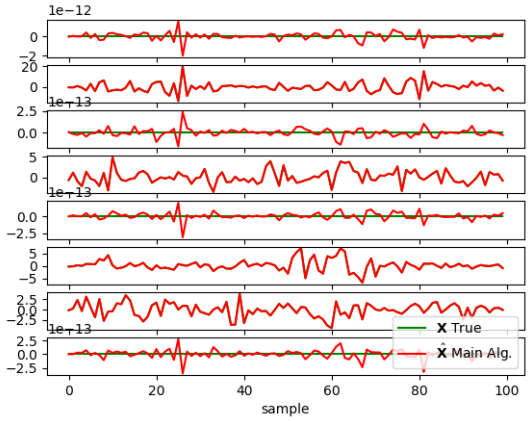

Figure 8.1: Each plot shows one row of the estimate $\hat{\mathbf{X}}_{\text{main}}$ where $k = N$ and true $\mathbf{A}$ is given, compared to the corresponding true row in $\mathbf{X}$. The MSE is $1.196E-29$. Only samples in the interval $[0, 100]$ are plotted.

Consider now the desired case where $M < N$. Figure 8.2 visualizes the estimate $\hat{\mathbf{X}}_{\text{main}}$ given $k = N$ and the true $\mathbf{A}$, resulting from a stochastic data set specified by $M = 6$, $N = 8$, $k = 4$ and $L = 1000$. From figure 8.2 it is seen that the estimates of the zero rows is not as close to zero as in figure 8.1. Thus, this can not be used as the indicator. However, the estimates of the zero rows still appears as a scaled replica of an estimate of a non-zero row.

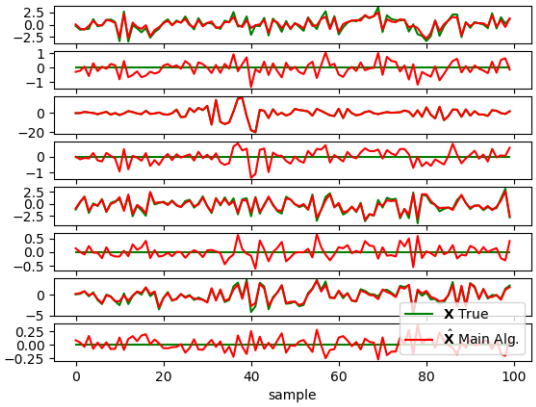

Figure 8.2: Each plot shows one row of the estimate $\hat{\mathbf{X}}_{\text{main}}$ using true $\mathbf{A}$, compared to the corresponding true row in $\mathbf{X}$. The MSE is 0.344. Only samples in the interval $[0, 100]$ are plotted.

A replica in this case not is considered an exact copy but a signal with similar trends over time. One attempt to locate the zero rows is to compare each row of $\hat{\mathbf{X}}_{\text{main}}$ to every other row by the MSE, in order to check if it appears more than one time.

Two rows are considered replicas if their mutual MSE is below a tolerance equal to 1. This operation is performed on the estimated source matrix $\hat{\mathbf{X}}_{\text{main}}$ plotted in figure 8.2. The operation gives the results displayed in table 8.1. From table 8.1 it is seen that source signals of row 2, 4, 6 and 8 are found to appear more than one time. These row indexes correspond to the zero rows of $\mathbf{X}$ as intended. This indicates the possibility of locating the zero rows from the estimate $\hat{\mathbf{X}}_{\text{main}}$ without providing the true k as an input.

Row index	1	2	3	4	5	6	7	8
# replicas	1	3	1	2	1	4	1	3

Table 8.1: Number of replicas for each row in $\hat{\mathbf{X}}_{\text{main}}$ of figure 8.2 based on the tolerance MSE < 1.

It is expected that the precision must depend on the chosen tolerance for the mutual MSE. For comparison table 8.2, 8.3 and 8.4 show the result from a tolerance of 0.5, 1.5 and 2 respectively. It is observed that a tolerance of 0.5 and 2 results in a different conclusion with respect to the number of zero rows – being respectively 2 and 6. From this it is clear that the tolerance is difficult to define and will affect the conclusion of the results.

Row index	1	2	3	4	5	6	7	8
# replicas	1	1	1	1	1	2	1	2

Table 8.2: Number of replicas for each row in $\hat{\mathbf{X}}_{\text{main}}$ of figure 8.2 based on the tolerance MSE < 0.5.

Row index	1	2	3	4	5	6	7	8
# replicas	1	4	1	3	1	4	1	3

Table 8.3: Number of replicas for each row in $\hat{\mathbf{X}}_{\text{main}}$ of figure 8.2 based on the tolerance MSE < 1.5.

Row index	1	2	3	4	5	6	7	8
# replicas	2	6	1	4	2	4	1	4

Table 8.4: Number of replicas for each row in $\hat{\mathbf{X}}_{\text{main}}$ of figure 8.2 based on the tolerance MSE < 2.

The results so far have relied on the true mixing matrix $\mathbf{A}$ given as an input to the main algorithm, due to the conclusion of chapter 6 where the estimate of $\mathbf{A}$ is abandoned. Thus, the results is conditioned on an exact estimate of $\mathbf{A}$ which this thesis does not manage to provide.

Now the investigations are repeated but with use of the main algorithm utilizing $\hat{\mathbf{A}}_{\text{fix}}$, cf. section 6.4. Figure 8.3 shows the estimate $\hat{\mathbf{X}}_{\text{main}}$ given $k = N$ and $\hat{\mathbf{A}}_{\text{fix}}$,

resulting from a stochastic data set specified by $M = 6$, $N = 8$, $k = 4$ and $L = 1000$. As expected, according to the results from section 6.4.2, it is generally seen from figure 8.3 that every row of the estimate is less accurate as the result is based on $\hat{\mathbf{A}}_{\text{fix}}$ instead of the true $\mathbf{A}$.

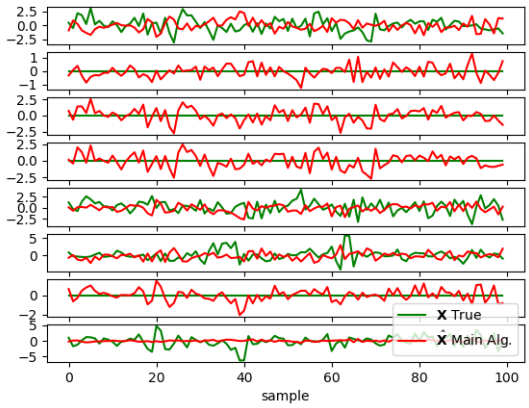

Figure 8.3: Each plot shows one row of the estimate $\hat{\mathbf{X}}_{\text{main}}$ using $\mathbf{A}_{\text{norm2}}$, compared to the corresponding true row in $\mathbf{X}$. The MSE is 128.7. Only samples in the interval $[0, 100]$ are plotted.

Table 8.5 shows the corresponding replica count with an MSE tolerance at 1. From the table it is seen that 7 out of the 8 rows are zero rows, while the true number is 4 rows. This could indicate that the tolerance is set to high. Table 8.6 show the replica count for an MSE tolerance at 0.5. From table 8.6 it is seen that the number of replicas is reduced. However, it still does not results in the right number of zero rows.

Row index	1	2	3	4	5	6	7	8
# replicas	3	5	2	2	4	1	4	5

Table 8.5: Number of replicas for each row in $\hat{\mathbf{X}}_{\text{main}}$ of figure 8.3 based on the tolerance MSE < 1.

Row index	1	2	3	4	5	6	7	8
# replicas	2	2	2	2	1	1	1	3

Table 8.6: Number of replicas for each row in $\hat{\mathbf{X}}_{\text{main}}$ of figure 8.3 based on the tolerance MSE < 0.5.

From the observations made through this investigation, based on synthetic data, the following conclusions are made. From figure 8.2 and table 8.1 a potential is found with respect to identifying the zero rows within the estimate, implying the desired estimate of k – conditioned by an exact estimate of $\mathbf{A}$. Here the zero rows are

identified as the rows of the estimate for which similar signals appear in other rows indicating that no new estimate has been computed. From figure 8.3 and tables 8.5 and 8.6 $\hat{\mathbf{A}}_{\text{fix}}$ is utilized in the main algorithm, as it will be when applied to real EEG data. Here it has not been possible to identify the zero rows correctly, based on the replica count. Thus, it must be concluded that the method is not reliable when the estimate is computed by the main algorithm. However, it is essential that a potential was found under ideal conditions.

To finish the investigation, the replica count method has been applied to the estimation of real EEG data. This is done due to the possibility of seeing a different behavior from the real EEG data compared to the synthetic data.

8.2 Empirical Test on EEG Measurements

Consider now the same estimation of the number of active sources, k, but from the real EEG measurements. For this estimation one can not compare the estimation to the real sources as in the previous section. Hence, the replica count method is just applied an estimation $\hat{\mathbf{X}}_{\text{main}}$ of real EEG measurements and a conclusion is made from the observed result.

The source signal estimation is performed on segment 10 from the `S1_Cclean` EEG data set, where every second sensor is removed to achieve the case where $M < N$. Specifically $\hat{\mathbf{X}}_{\text{main}}$ is computed given $k = N$ and $\hat{\mathbf{A}}_{\text{fix}}$ resulting from segment of EEG measurements specified by $M = 1/2N = 13$.

Figure 8.4 visualize the recovered source matrix $\hat{\mathbf{X}}_{\text{main}}$ from time segment $s = 10$ for $M = 13$ and $k = N = 27$. For visual simplicity only 8 out of 27 sources are visualized.

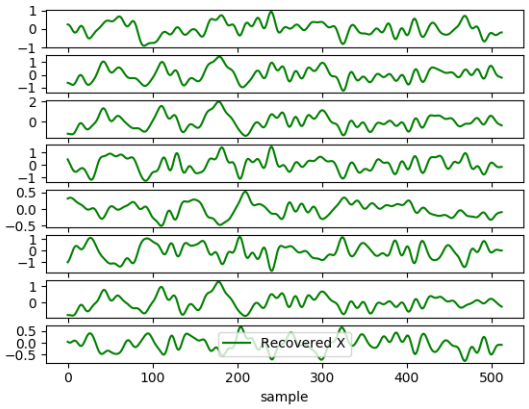

Figure 8.4: The recovered source matrix $\hat{\mathbf{X}}_{\text{main}}$ from time segment $s = 10$ from `S1_Cclean` for $M = 13$ and $k = N = 27$. Note only the first 8 rows of $\hat{\mathbf{X}}_{\text{main}}$ is plotted for simplicity.

From figure 8.4 it is seen that all 10 source signals appears to be active – the non visualized source signals do also appears to be active. Furthermore, there seem not to be any visible indication of an active source being non-active, with respect to being a direct replica.

By applying the replica count method to $\hat{\mathbf{X}}_{\text{main}}$ from figure 8.4 only one source signal is not considered replicas. This potentially active source signal is found in row 13, the last row plot on figure 8.4. This leads to the conclusion that for time segment $s = 10$ with system specification $M = 13$ for the `S1_Cclean` EEG data set only have $k = 1$ active sources.

Chapter 9

Discussion

The purpose of this thesis was to investigate the possibility of reproducing state of the art methods and results for recovery of source signals, from low-density scalp EEG measurements inducing an under-determined system. The considered state of the art methods is multiple sparse Bayesian learning (M-SBL)[5] and covariance-domain dictionary learning (Cov-DL)[4], both published by O. Balkan, et al. in the year of 2014 and 2015 respectively. It was found that this task was not easy as they did not provide any software to reproduce their results. The resulting combination of Cov-DL and M-SBL, the main algorithm, did not manage to fully solve the inverse EEG problem of recovering the mixing matrix $\mathbf{A}$ and a source matrix $\mathbf{X}$ successfully, from the EEG measurements. From the verification of the Cov-DL method, it was concluded that it failed to provide a sufficient estimate $\hat{\mathbf{A}}$, when applied on simulated data. Due to not having a successful estimate of $\mathbf{A}$ the recovery of $\mathbf{X}$ was compromised when using the M-SBL method. However, when using the true $\mathbf{A}$, M-SBL provides an estimate $\hat{\mathbf{X}}$ sufficiently close to the real source matrix, in the under-determined case $M < N$. In the main algorithm the estimate $\hat{\mathbf{A}}$ from Cov-DL was replaced by a fixed mixing matrix $\hat{\mathbf{A}}_{\text{fix}}$. The final performance of the main algorithm was found to vary significantly, hence a sufficient performance was not confirmed. As expected a similar conclusion was to be drawn from tests on real EEG measurement.

In chapter 6 it was concluded that the implementation of the Cov-DL method is unsuccessful. The issue was located to the definition of the optimization problem determining the columns of $\hat{\mathbf{A}}$. This does question the reproducibility of the scientific article [4] which has been used as the main source. The article [4] did not provide any code or implementation specifications. Likewise, it was not possible to recreate or access the exact data, which was used to provide the results presented in the article. Thus, the intention was never to recreate the exact results from the article but rather to demonstrate the conclusion. That the method manages to provide results of a certain success rate. One could argue that testing the implementation

on the same data would lead to a different conclusion. However, this was sought approached by the stochastic simulated data, cf. section 6.2, which was created with inspiration from the article.

In chapter 6 it was concluded that the M-SBL method manage to successfully estimate $\mathbf{X}$ when applied to the simulated stochastic data sets and given the true $\mathbf{A}$. Though, the performance was found to decrease slightly as the number of sources increases relative to the number of sensors. Regarding the reproducibility of the article [5] the results indicate that the provided information about M-SBL has been sufficient. However, this article did, as [4], not provide any code or data disabling the possibility of recreating the exact results. In [5] tests were conducted on random simulations of $\mathbf{A}$ and $\mathbf{X}$ with various noise level added. Thus, due to no counter arguments, the tests of M-SBL in this thesis were conducted on the simulated data sets already created for the tests of Cov-DL, with inspiration from [4]. With respect to the main algorithm, uniting Cov-DL and M-SBL, an alternative to the estimate of $\mathbf{A}$ was necessary. Through empirical tests, which is discussed later, a $\hat{\mathbf{A}}_{\text{fix}}$ was chosen to replace the estimate from Cov-DL. With $\hat{\mathbf{A}}_{\text{fix}}$ the main algorithm manage to estimate $\mathbf{X}$ but with a significantly higher error compared to the use of the true $\mathbf{A}$, which is not considered to be successful.

The performance test of the main algorithm was first conducted on simulated stochastic data resembling real EEG measurements. Inspired by [4] the sources was synthesized by independent auto-regressive processes. The true $\mathbf{A}$ however, was simply generated by a Gaussian distribution. This choice was based on a lack of information to point in different directions. Instead of the true $\mathbf{A}$ being chosen as a stochastic matrix a deterministic matrix could have been chosen instead. The choice of the stochastic true $\mathbf{A}$ for the simulated data sets could affect the results when testing the fixed alternatives for the mixing matrix estimated by Cov-DL. From the test, cf. section 6.4, it was found that a fixed Gaussian estimate, generated with same specifications as the true $\mathbf{A}$, did not lead to the best recovery of the $\mathbf{X}$, which went against the natural expectation – the better estimate of $\mathbf{A}$ the better estimate of $\mathbf{X}$. Here a fixed estimate with Gaussian distribution of higher variance provided the best estimate of $\mathbf{X}$. It is here one could argue that the stochastic true $\mathbf{A}$ had an influence to the results. Furthermore, the choice of error measurement might also be reflected in the results. Not being sufficient towards the purpose.

The common choice of error measurement throughout the thesis was the mean-square error (MSE). The MSE measures the performance of an estimate with respect to the true value, by comparing each element and summing the squared error. Hence, the MSE is providing a measure of how far the estimate is from the true value. This was considered a sufficient error measurement for evaluation of Cov-DL and M-SBL. However, one challenge when using MSE is to set a tolerance defining when an estimate is considered successful. It could be argued that set tolerance should vary with respect to the data of interest. Though, it was not chosen to evaluate to the

estimates with respect to success rate. By this the above challenge was avoided, and the method is replaced by a more soft evaluation based on the MSE. A different choice of error measurement could have been the use of correlation between variables of the estimate and the true values. With respect to the comparison of the main algorithm to ICA as the ground truth, to be discussed next, using the correlations as the evaluation might have overcome the scaling issue with respect to ICA.

Consider now the performance test of the main algorithm on real EEG measurements. The choice of evaluating the performance by considering the ICA solution as the ground truth has been crucial. First of all the foundation for the evaluation was not ideal. It is an issue that the performance of the main algorithm on the simulated data was not as good expected, presumably due to the estimate of $\mathbf{A}$ being replaced by a fixed estimate. Thus, it is not reasonable to trust the results when the algorithm is applied to data for which the true results are not known at all. However, it can be argued that when comparing the obtained result to the best-known solution, in this case provided by ICA, a small error will indicate an acceptable performance from the main algorithm. The arguments accounting for the use of ICA were discussed in section 7.2. However, an unreliability will be present as the ICA algorithm is limited to $M = N$. The true N is unknown, thus ICA is never guaranteed to find all the active sources. Furthermore, the comparison of the main algorithm to ICA was found to be compromised. The localization and phase of the active sources are not necessary the same for the two estimates, which distorted the MSE between the two matrices to an unknown degree. In despite of this issue, the comparison was still performed by MSE, which suggest a potential to improve the found performance. Against the prior expectation, an acceptable performance was found for the case $M = N$, though for the cases of interest where $M < N$ the performance was decreased significantly.

Due to the possible unreliability of the performance evaluation with respect to ICA, an alternative test was considered – the alpha wave analysis. However, from the analysis no new conclusion was made. The expected behavior was not observed, with respect to an increased amount of alpha frequency for the test subject having closed eyes. The behavior was more or less fluctuating over time. However, exceptions from the expected behavior were also found for the raw measurement. This indicates that different approaches, with respect to measuring the power within the alpha band, should be considered. The advantage of this test approach, in general, is that one see past the challenge of recovering the exact source signal, but rather consider the practical usage of the source separation. For instance, when considering the usage of the source signals within a hearing aid, cf. section 1.1.1, the amount of active source signals might be more significant than an exact recovery.

The last issue addressed in this thesis was the estimation of the number of active sources k relative to the maximum number of sources N. Through out the implementation of the main algorithm in chapter 6 it was argued that setting $k = N$, would not compromise the results. In fact perhaps lead to better estimates as fewer

sources must be recovered and therefore reduces the amount of possible errors. This is supported by [5] where the same assumption is used. Likewise for M-SBL, providing k would only reduce possible errors within localization of the sources. However, by setting $k = N$ one must assume that k sources are active, thus no less that k sources are estimated. Hence, to justify this definition of k one must have a qualified guess with respect to the true k. However, this is the issue addressed in the problem statement, as this is not possible in practice.

To address this exact issue an investigation with respect to estimation of k was conducted in chapter 8. Due to interesting empirical observation, it was chosen to analyse the source signals resulting from the main algorithm when $k = N = \widetilde{M}$. That is estimating the maximum number of active sources under the hypothesis that the false estimates where to be identified among the true estimates. By false estimate there is referred to a non-zero estimate of a zero row. From visual observation of results from the simulated stochastic data sets, a potential was seen as the false estimates appear as replicas of the true estimates. These false estimates was identified by searching for the replicates of the true estimates. However, this was not found successful in the desired cases where $M < k$. Here the false estimates manage to diverge more from the true estimates. Furthermore, the identification method was found to have trouble separating the true estimate from the corresponding replicates. Hence alternative methods could have been considered with respect to estimation of k. One obvious approach is to consider the estimate of k which could be provided from M-SBL, if a k was not given as an input to the algorithm. However, the success rate of such estimation of k was not provided in the article [5] of which the article was based. As the performance presented in the article was obtained by providing k to the algorithm, cf. 5.2.1.

Throughout this chapter the choices which ware found essential toward the obtained conclusions have been discussed and alternative choices have been considered. None of these alternatives are sought investigated in the thesis but they will serve as essential aspects to be considered if further work on the main algorithm where to be conducted.

Chapter 10

Conclusion

The main purpose of this thesis is to recreate a state of the art result for solving the EEG inverse problem, with respect to the original brain source signals, in the under-determined case. Secondary the method is sought improved from a practical perspective.

A main algorithm was proposed, based on the state of the art methods covariance-domain dictionary learning (Cov-DL) [4] and multiple sparse Bayesian learning (M-SBL) [5], for recovery of respectively the mixing matrix and the source signals, from the EEG inverse problem. From the initial verification tests it was found that the implementation of the Cov-DL method did not succeed in providing a sufficient recovery of the mixing matrix. Based on brief analysis of the error, it may be concluded that the scientific article proposing the method did not provide a sufficient degree of reproducibility. However, a more extensive error analysis has to be conducted in order to locate the issue. The implementation of the M-SBL method is found successful when conditioned on the true mixing matrix. From this it is concluded that the corresponding scientific article did provide a sufficient degree of reproducibility.

To replace the estimate of the mixing matrix from Cov-DL within the main algorithm, a fixed estimate was chosen, based on empirical tests. By this modification and the corresponding tests, it is not expected that the main algorithm manages to successfully recoverer the source signals. As expected the performance of the resulting main algorithm, applied to synthetic data, was found to be both insufficient and fluctuating for the under-determined case of interest. This indicates an unreliability which complicates any useful conclusions. Though, with respect to investigating the extent of the resulting main algorithm, its performance was tested on real EEG data. Here the performance was evaluated relative to the corresponding solutions provided by independent component analysis (ICA) on the complete system.

From the analysis of the results, a potentially reliable recovery of the source signals was seen for the complete system, while for the case of interest, the under-determined case, the performance was not evaluated as sufficient. Thus it is con-

cluded that only when the proposed main algorithm is conditioned on an exact estimate of the mixing matrix does it provide sufficiently recovered source signals. An alternative test was conducted as an attempt to analyse the recovered source signals from a different perspective. A frequency analysis relating the results from EEG level to similar result on source level. From this analysis no significant founding was discovered in favor of the recovered sources, thus the previous conclusion is preserved.

With respect to the practical perspective adressed in the problem statement, the issue of the unknown number of active source was investigate through empirical tests. A method was proposed with respect to identification of non-active sources recovered without the true k being know. The method showed potential when applied to recovered source signals conditioned on the true mixing matrix. However, when applied to the under-determined case difficulties did arise. Thus, it is concluded that the method does not provide a sufficient estimate of the number of active sources. But the found potential suggests that further work on the method may allow the possibility of finding a reliable estimate.

Overall, it is concluded that a recreation of the specified state of the art methods for source recovery was not successfully provided by the proposed main algorithm - utilizing the proposed alternative to the estimation of the mixing matrix. However when conditioned on an exact estimate of the mixing matrix, the main algorithm showed grate potential for source signal recovery, even for the under-determined case which has been of interest within this thesis. It is furthermore concluded that a potential is found with respect to using the main algorithm in practice, considering an estimation of the unknown number of active sources.

Chapter 11

Further Studies

Based on the accomplished conclusions, further studies regarding the proposed main algorithm and the general issue of source signal recovery will be discussed.

One essential finding in this thesis was the negative result with respect to the proposed implementation of the Cov-DL method. It is concluded that the reproducibility of the corresponding article was not sufficient. However, it is not excluded that further studies would enable a successful implementation. This could be further investigations in form of rewriting the optimization problem or choosing a different optimization process.

A different aspect could be to dismiss the Cov-DL method and do some research with respect to alternative methods for finding the mixing matrix. Such method could be the low resolution electrical tomograph (LORETA), which localizes electrical activity inside the brain. Or, the minimum norm estimates (MNE), which reconstructs the activity on the cortical surface [25].

From the overall perspective of the topic, it could be of interest to alter the view on the EEG measurements. In this thesis the purpose was to recover active source signals from the EEG measurements, by the main algorithm. A news article from April 2020 [17] presents the newest research from Eriksholm Research Center. Research with respect to0 application of EEG measurements within a hearing aid, as mentioned in the chapter 1. They suggest that, by focusing only on recovery of EEG signals resulting from eye movements, the direction of the sight can be measured. As this thesis focus on finding all sources from the provided EEG measurements, this could be a change of focus. The advantage of targeting the specific signals, which was before considered as noise on the EEG measurement, is that fewer sensors is needed and fewer signals are sought recovered. This result in fewer computations, and a potential of avoiding the difficult under-determined case could be present. For this case a different EEG measurements data base must be provided or created since the data used in this thesis do not contain the signals created by the eye movement and surrounding muscle movements.

Bibliography

[1] Aharon, M., Elad, M., and Bruckstein, A. "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation". In: *IEEE Transactions on signal processing* Vol. 54, No. 11 (2006).

[2] Alickovic, E. et al. "A Tutorial on Auditory Attention Identification Methods". In: *Front. Neurosci* 13:153 (2019).

[3] Antoniou, A. and Lu, W-S. *Practical Optimization, Algorithms and Engineering Applications*. Springer, 2007.

[4] Balkan, O., Kreutz-Delgado, K., and Makeig, S. "Covariance-Domain Dictionary Learning for Overcomplete EEG Source Identification". In: *ArXiv* (2015).

[5] Balkan, O., Kreutz-Delgado, K., and Makeig, S. "Localization of More Sources Than Sensors via Jointly-Sparse Bayesian Learning". In: *IEEE Signal Processing Letters* (2014).

[6] Balkan, O. Y. "Support Recovery and Dictionary Learning for Uncorrelated EEG Sources". Master thesis. University of California, San Diego, 2015.

[7] Boyd, S. and d'Aspremont, A. "Relaxations and Randomized Methods for Nonconvex QCQPs". In: *EE392o, Stanford University* (2003).

[8] Boyd, S. and Vandenberhe, L. *Convex Optimization*. Cambridge University Press, 2004.

[9] Christensen, C. B. et al. "Toward EEG-Assisted Hearing Aids: Objective Threshold Estimation Based on Ear-EEG in Subjects With Sensorineural Hearing Loss". In: *Trends Hear, SAGE* 22 (2018).

[10] Dattorro, J. *Convex Optimization and Euclidean Distance Geometry*. Meboo Publising USA, 2005.

[11] Dekking, F.M. et al. *A Modern Introduction to Probability and Statistics*. Springer, 2005.

[12] Elad, M. *Sparse and Redundant Representations*. Springer, 2010.

[13] Eldar, Y. C. and Kutyniok, G. *Compressed Sensing: Theory and Application*. Cambridge University Presse, New York, 2012.

[14] Foucart, S. and Rauhut, H. *A Mathematical Introduction to Compressive Sensing*. Springer Science+Business Media New York, 2013.

[15] Friston, K. J. "Functional and Effective Connectivity: A Review". In: *Brain Connectivity* 1 (2011).

[16] Friston, K. J. "Functional Integration and Inference in the Brain". In: *Progress in Neurobiology 590* 1-31 (2002).

[17] Hovgaard, L. "Hjerneboelger styrer haereapparatet". In: *Ingenioeren* nr. 13, 1. section (2020).

[18] Hyvarinen, A., Karhunen, J., and Oja, E. *Independent Component Analysis*. Ed. by Haykin, Simon. John Wiley and Sons, Inc., 2001.

[19] Kay, S. M. *Intuitive Probability and Random Processes using MATLAB*. 4-th corrected version of the 5-th printing (2012). Springer Science+Business Media, LLC, 2006.

[20] Makeig, S. et al. "Blind Separation of Auditory Event-Related Brain Responses into Independent Components". In: *Proc. Natl. Acad. Sci. USA* 94 (1997).

[21] Makeig, S. et al. "Independent Component Analysis of Electroencephalographic Data". In: *Advances in Neural Information Processing Systems* 8 (1996).

[22] Oppenheim, A. V. and Schafer, R. W. *Discrete-Time Signal Processing*. 3th. Pearson Education Limited, 2014.

[23] Pal, P. and Vaidyanathan, P. P. "Pushing the Limits of Sparse Support Recovery Using Correlation Information". In: *IEEE Transactions on Signal Processing* VOL. 63, NO. 3, Feb. (2015).

[24] Palmer, J. A. et al. "Newton Method for the ICA Mixture Model". In: *ICASSP 2008* (2008).

[25] Rincon, A. L. and Shimoda, S. "The Inverse Problem in Electroencephalography using the Bidomainmodel of Electrical Activity". In: *Journal of Neuroscience Methods* (2016).

[26] Sanei, S. and Chambers, J.A. *EEG Signal Processing*. Johm Wiley and sons, Ltd, 2007.

[27] Spence, L. E., Insel, A. J., and Friedberg, S. H. *Elementary Linear Algebra 2015 - A Customised Edition of Elementary Linear Algebra: A Matrix Approach*. Ed. by Geil, Olav. Pearson Education Limited, 2015.

[28] Steen, F. Van de et al. "Critical Comments on EEG Sensor Space Dynamical Connectivity Analysis". In: *Brain Topography* 32 p. 643-654 (2019).

[29] *Studies within Steering of Hearing Devices using EEG and Ear-EEG*. `https://www.eriksholm.com/research/cognitive-hearing-science/eeg-steering`. Accessed: 2019-10-03.

[30] Teplan, M. “Fundamentals of EEG Measurement”. In: *Measurement science review* 2 (2002).

[31] Wipf, D. P. “Bayesian Methods for Finding Sparse Representations”. PhD thesis. University of California, San Diego, 2006.

[32] Wipf, D. P. and Rao, B. D. “An Empirical Bayesian Strategy for Solving the Simultaneous Sparse Approximation Problem”. In: *IEEE Transactions on Signal Processing* Vol. 55.No. 7 (2007).

Appendix A

Supplementary Theory for Chapter 4

Throughout this chapter supplemental theory for understanding the covariance-domain dictionary learning (Cov-DL) is described. First an introduction to compressive sensing which is the framework behind Cov-DL. Then an dictionary learning algorithm used for finding the dictionary matrix $\mathbf{D}$ in section 4.2.1 will be described. And at last principal component analysis (PCA) is introduced as the method behind finding $\mathbf{D}$ in section 4.2.2.

A.1 Introduction to Compressive Sensing

Compressive sensing is the theory of efficient recovery of a signal from a minimal number of observed measurements. It is build upon empirical observations assuring that many signals can be approximated by remarkably sparser signals. Assume linear acquisition of the observed measurements. Then the relation between the measurements and the signal to be recovered can be modeled by the multiple measurement vector (MMV) model (3.2) [14].

Through this section the introduction of the theory behind compressive sensing will be presented for one measurement vector of (3.2), $\mathbf{y}$, such that the theory is based on the linear system (3.1). This will be done for simplicity, but the theory will still apply for the extended linear system (3.2).

In compressive sensing terminology, $\mathbf{x} \in \mathbb{R}^N$ is the signal of interest sought recovered from the EEG measurement $\mathbf{y} \in \mathbb{R}^M$ by solving the linear system (3.1). In the typical compressive sensing case, the system is under-determined, $M < N$, and there will therefore exist infinitely many solutions, provided that one solution exist. However, by enforcing certain sparsity constraints it is possible to recover the wanted signal, hence the term sparse signal recovery [14]. The sparsity constraints are the ones presented in 3.1 where the ℓ_0 is introduced to count the non-zeros of the signal

of interest, the source vector $\mathbf{x}$. The number of non-zeros (active sources) k describe how sparse the source vector is.

To find a k-sparse solution to the linear system (3.1) it can be viewed as the following optimization problem.

$$\mathbf{x}^* = \arg\min_{\mathbf{x}\in\mathbb{C}} \|\mathbf{x}\|_0 \quad \text{subject to} \quad \mathbf{A}\mathbf{x} = \mathbf{y}.$$

Unfortunately, this optimization problem is non-convex due to the definition of the ℓ_0-norm and is therefore difficult to solve – it is a NP-hard problem. Instead, by replacing the ℓ_0-norm with the ℓ_1-norm, the optimization problem can be approximated and hence becomes computationally feasible [13, p. 27]

$$\mathbf{x}^* = \arg\min_{\mathbf{x}\in\mathbb{C}} \|\mathbf{x}\|_1 \quad \text{subject to} \quad \mathbf{A}\mathbf{x} = \mathbf{y}. \tag{A.1}$$

With this optimization problem the best k-sparse solution $\mathbf{x}^*$ can be found. The optimization problem is referred to as ℓ_1 optimization problem or Basis Pursuit. The following theorem justifies that the ℓ_1 optimization problem finds a sparse solution [14, p. 62-63].

Theorem A.1.1
A mixing matrix $\mathbf{A} \in \mathbb{R}^{M\times N}$ is defined with columns $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_N]$. By assuming uniqueness of a solution $\mathbf{x}^*$ to

$$\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{x}\|_1 \quad \text{subject to} \quad \mathbf{A}\mathbf{x} = \mathbf{y},$$

the system $\{\mathbf{a}_j, j \in \text{supp}(\mathbf{x}^*)\}$ is linearly independent, and in particular

$$\|\mathbf{x}^*\|_0 = \text{card}(\text{supp}(\mathbf{x}^*)) \leq M.$$

Proof
Assume that the set $\{\mathbf{a}_l, l \in S\}$ of l columns from matrix $\mathbf{A} \in \mathbb{R}^{M\times N}$ is linearly dependent with the support $S = \text{supp}(\mathbf{x}^*)$. Thus a non-zero vector $\mathbf{v} \in \mathbb{R}^N$ supported on S exists such that $\mathbf{A}\mathbf{v} = \mathbf{0}$ – the system is linear dependent. The unique solution $\mathbf{x}^*$ can then be written as, for any $t \neq 0$,

$$\|\mathbf{x}^*\|_1 < \|\mathbf{x}^* + t\mathbf{v}\|_1 = \sum_{l\in S} |x_l^* + tv_l| = \sum_{l\in S} \text{sgn}(x_l^* + tv_l)(x_l^* + tv_l). \tag{A.2}$$

For a small $|t|$

$$|t| < \min_{l\in S} \frac{|x_l^*|}{\|\mathbf{v}\|_\infty},$$

then the sign function becomes

$$\operatorname{sgn}(x_l^* + tv_l) = \operatorname{sgn}(x_l^*), \quad \forall l \in S.$$

By including this result in (A.2) and remembering $t \neq 0$:

$$\|\mathbf{x}^*\|_1 < \sum_{l\in S} \operatorname{sgn}(x_l^*)(x_l^* + tv_l) = \sum_{l\in S} \operatorname{sgn}(x_l^*)x_l^* + t\sum_{l\in S} \operatorname{sgn}(x_l^*)v_l = \|\mathbf{x}^*\|_1 + t\sum_{l\in S} \operatorname{sgn}(x_l^*)v_l.$$

From this it can be seen that it is always possible to choose $t \neq 0$ small enough such that

$$t\sum_{l\in S} \operatorname{sgn}(x_l^*)v_l \leq 0,$$

which contradicts that $\mathbf{v}$ make the columns of $\mathbf{A}$ linear dependent. Therefore, the set $\{\mathbf{a}_l, l \in S\}$ must be linearly independent. ■

From the theorem it must be concluded that the choice of the mixing matrix $\mathbf{A}$ has a significant impact on whenever a unique solution $\mathbf{x}^*$ exist for the ℓ_1 optimization problem (A.1). Therefore, when recovering $\mathbf{A}$, some considerations regarding the recovering process of $\mathbf{A}$ must be taken into account. A method for the recovering of $\mathbf{A}$ could be to use a dictionary. This will be explained in the following section 4.2.1.

An alternative solution method to the ℓ_1 optimization includes greedy algorithms like the Orthogonal Matching Pursuit (OMP) [14, P. 65]. The OMP algorithm is an iteration process where an index set S is updated – at each iteration – by adding indices corresponding to the columns of $\mathbf{A}$ which describe the residual best possible, hence greedy. The vector $\mathbf{x}$ is then updated by a vector supported on S which minimize the residual. That is the orthogonal projection of $\mathbf{y}$ onto the $\operatorname{span}\{\mathbf{a}_l \mid l \in S\}$.

A.2 K-SVD Algorithm

The dictionary learning algorithm K-SVD provides an updating rule which is applied to each column of $\mathbf{A}_0 = [\mathbf{a}_1, \ldots, \mathbf{a}_N]$ where $\mathbf{A}_0$ being a random initial dictionary matrix. Updating first $\mathbf{a}_j$ for $j = 1, \ldots, N$ and then the corresponding row of $\mathbf{X}$, $\mathbf{x}_{i\cdot}$ for $j = i$. Let $\mathbf{a}_{j_0}$ be the column to be updated and let the remaining columns be fixed. By rewriting the objective function in (4.6) using matrix notation it is possible to isolate the contribution from $\mathbf{a}_{j_0}$.

$$\begin{aligned}\|\mathbf{Y} - \mathbf{AX}\|_F^2 &= \left\| \mathbf{Y} - \sum_{\substack{j=1 \\ j=i}}^{N} \mathbf{a}_j \mathbf{x}_{i\cdot} \right\|_F^2 \\ &= \left\| \left(\mathbf{Y} - \sum_{\substack{j\neq j_0 \\ j=i}}^{N} \mathbf{a}_j \mathbf{x}_{i\cdot} \right) - \mathbf{a}_{j_0}\mathbf{x}_{i_0\cdot} \right\|_F^2, \end{aligned} \tag{A.3}$$

where $i = j$, $i_0 = j_0$ and where F is the Frobenius norm that works on matrices

$$\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}|a_{ij}|^2}.$$

In (A.3) the term in the parenthesis is denoted by $\mathbf{E}_{j_0}$, an error matrix, and hence by minimizing (A.3) with respect to $\mathbf{a}_{j_0}$ and $\mathbf{x}_{i_0\cdot}$ leads to the optimal contribution from j_0

$$\min_{\mathbf{a}_{j_0},\mathbf{x}_{i_0\cdot}} \|\mathbf{E}_{j_0} - \mathbf{a}_{j_0}\mathbf{x}_{i_0\cdot}\|_F^2 . \tag{A.4}$$

The optimal solution to (A.4) is known to be the rank-1 approximation of $\mathbf{E}_{j_0}$ [12, p. 232]. That is a partial single value decomposition (SVD) makes the best low-rank approximation of a matrix such as $\mathbf{E}_{j_0}$. The SVD is given as

$$\mathbf{E}_{j_0} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T \in \mathbb{R}^{M\times N},$$

with $\mathbf{U} \in \mathbb{R}^{M\times M}$ and $\mathbf{V} \in \mathbb{R}^{N\times N}$ being unitary matrices[1] and $\boldsymbol{\Sigma} = \operatorname{diag}(\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_M) \in \mathbb{R}^{M\times N}$ a diagonal matrix. $\boldsymbol{\sigma}_j$ are the non-negative singular values of $\mathbf{E}_{j_0}$. The best k-rank approximation to $\mathbf{E}_{j_0}$, with $k < \operatorname{rank}(\mathbf{E}_{j_0})$ is then given by [12, p. 232]:

$$\mathbf{E}_{j_0}^{(k)} = \sum_{j=1}^{k} \boldsymbol{\sigma}_j \mathbf{u}_j \mathbf{v}_j^T .$$

Since the outer product always has rank-1 letting $\mathbf{a}_{j_0} = \mathbf{u}_1$ and $\mathbf{x}_{i_0\cdot} = \boldsymbol{\sigma}_j \mathbf{v}_1^T$ solves the optimization problem (A.4). However, in order to preserve the sparsity in $\mathbf{X}$ while optimizing, only the non-zero entries in $\mathbf{x}_{i_0\cdot}$ are allowed to vary. For this purpose only a subset of columns in $\mathbf{E}_{j_0}$ is considered. Those which correspond to the non-zero entries of $\mathbf{x}_{i_0\cdot}$. A matrix $\mathbf{P}_{i_0}$ is defined to restrict $\mathbf{x}_{i_0\cdot}$ to only contain the non-zero-rows corresponding to N_{j_0} non-zero rows:

$$\mathbf{x}_{i_0\cdot}^{(R)} = \mathbf{x}_{i_0\cdot}\mathbf{P}_{i_0}$$

where R denoted the restriction. By applying the SVD to the error matrix which has been restricted $\mathbf{E}_{j_0}^{(R)} = \mathbf{E}_{j_0}\mathbf{P}_{i_0}$ and updating $\mathbf{a}_{j_0}$ and $\mathbf{x}_{i_0\cdot}^{(R)}$ the rank-1 approximation is found and the original representation vector is updated as $\mathbf{x}_{i_0\cdot} = \mathbf{x}_{i_0\cdot}^{(R)}\mathbf{P}_{i_0}^T$.

The main steps of K-SVD is described in algorithm 3.

[1] Unitary matrix: $\mathbf{U}^T\mathbf{U} = \mathbf{U}\mathbf{U}^T = \mathbf{I}$

Algorithm 3 K-SVD

$k = 0$
Initialize random $\mathbf{A}_{(0)}$
Initialize $\mathbf{X}_{(0)} = \mathbf{0}$

procedure K-SVD($\mathbf{A}_{(0)}$)
  Normalize columns of $\mathbf{A}_{(0)}$
  while error $\geq$ limit **do**
    $j = j + 1$
    for $j \leftarrow 1, 2, \ldots, L$ **do** ▷ updating each col. in $\mathbf{X}_{(k)}$
      $\hat{\mathbf{x}}_j = \min_{\mathbf{x}} \|\mathbf{y}_j - \mathbf{A}_{(k-1)}\mathbf{x}_j\|$ subject to $\|\mathbf{x}_j\| \leq k$ ▷ use Basis Pursuit
    end for
    $\mathbf{X}_{(k)} = \{\hat{\mathbf{x}}_j\}_{j=1}^{L}$
    for $j_0 \leftarrow 1, 2, \cdots, N$ **do**
      $\Omega_{j_0} = \{j \mid 1 \leq j \leq L, \mathbf{X}_{(k)}[j_0, j] \neq 0\}$
      From Ω_{j_0} define $\mathbf{P}_{i_0}$
      $\mathbf{E}_{j_0} = \mathbf{Y} - \sum_{j \neq j_0}^{N} \mathbf{a}_j \mathbf{x}_{i\cdot}$
      $\mathbf{E}_{j_0}^{(R)} = \mathbf{E}_{j_0}\mathbf{P}_{i_0}$
      $\mathbf{E}_{j_0}^{(R)} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$ ▷ perform SVD
      $\mathbf{a}_{j_0} \leftarrow \mathbf{u}_1$ ▷ update the j_0 col. in $\mathbf{A}_{(k)}$
      $(\mathbf{x}_{i_0\cdot})^{(R)} \leftarrow \boldsymbol{\sigma}_1 \mathbf{v}_1$
      $\mathbf{x}_{i_0\cdot} \leftarrow (\mathbf{x}_{i_0\cdot})^{(R)} \mathbf{P}_{i_0}^T$ ▷ update the i_0 row in $\mathbf{X}_{(k)}$
    end for
    error = $\|\mathbf{Y} - \mathbf{A}_{(k)}\mathbf{X}_{(k)}\|_F^2$
  end while
end procedure

A.3 Principal Component Analysis

In this section the method behind principal component analysis (PCA) used for the Cov-DL described in section 4.2.2.

PCA is dimensionality reduction method used for reduction of dimensions of large data sets. In short, PCA used the statistical information of mean, variance and correlation between the data to transform the large data set into smaller datasets while maintaining most of the original information. These smaller data sets are known as the principal components and contain most of the information of the data set but with fewer dimensions. For some datasets, before PCA is applied, the data must undergo a standardization/scaling to remove any difference in the data. This is essential as large differences between the data would dominate. The standardization

of a dataset $\mathbf{Z}$ is performed by

$$\tilde{\mathbf{z}}_i = \frac{\mathbf{z}_i - \bar{\mathbf{z}}_i}{s_{\mathbf{z}_i}}, \quad \forall i = 1, \ldots, m$$

where $\mathbf{z}_i$ is a row of a matrix $\mathbf{Z}$, $\bar{\mathbf{z}}_i$ is the sample mean of $\mathbf{z}_i$ and $s_{\mathbf{z}_i}$ is the standard deviation of $\mathbf{z}_i$. The standardized data set is now giving by $\tilde{\mathbf{Z}}$. The standardization step is unnecessary in the case of real EEG scalp measurements as no large difference between the data is present.

With $\mathbf{Z}$ or the scaled data $\tilde{\mathbf{Z}}$ a correlation matrix is computed as

$$\mathbf{\Sigma}_{\mathbf{Z}} = \mathrm{Corr}(\mathbf{Z}) = \frac{1}{m}\mathbf{Z}^T\mathbf{Z}.$$

From the correlation matrix a orthonormal basis of eigenvectors $\mathbf{p}_1, \ldots, \mathbf{p}_m$ with corresponding eigenvalues $\lambda_1, \ldots, \lambda_m$ exists, cf. theorem 6.15 in [27, p. 375]. Furthermore, one assumes that $\lambda_1 \geq \cdots \geq \lambda_m$. The eigenvectors and eigenvalues can be computed from the correlation matrix by e.g. using a singular value decomposition (SVD). With SVD an orthogonal matrix $\mathbf{P}$ with the eigenvectors $\mathbf{p}_1, \ldots, \mathbf{p}_m$ as columns is obtained with the associated eigenvalues as a diagonal matrix denoted as $\mathbf{P}_{\mathrm{diag}}$. The principal components are then defined by

$$\mathbf{u}_i = \mathbf{Z}\mathbf{p}_i,$$

where $\mathbf{p}_i$ is the i-th eigenvector of the correlation matrix $\mathbf{\Sigma}_{\mathbf{Z}}$. Thus, each principal component is a linear combination of the dataset $\mathbf{Z}$ [27, p. 460]. With the principal components the first N components forms a set of basis vectors $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_N]$.

Appendix B

Derivations for Multiple Sparse Bayesian Learning

B.1 Derivation of Posterior Mean $\mathcal{M}$ and Covariance $\mathbf{\Sigma}$

The purpose of this section is to derive the mean $\mathcal{M}$ and covariance $\mathbf{\Sigma}$ of the posterior distribution

$$p\left(\mathbf{x}_{\cdot j}|\mathbf{y}_{\cdot j};\boldsymbol{\gamma}\right) \sim \mathcal{N}(\boldsymbol{\mu}_{\cdot j}, \mathbf{\Sigma}),$$

from (5.2) in section 5.1.1.

Let $\mathbf{x}_{\cdot j} = \mathbf{x}$ and $\mathbf{y}_{\cdot j} = \mathbf{y}$ to ease the notation throughout the derivation. The prior and likelihood is then defined as

$$\begin{aligned} p(\mathbf{x};\boldsymbol{\gamma}) &\sim \mathcal{N}(\mathbf{0}, \boldsymbol{\gamma}\mathbf{I}), \\ p(\mathbf{y}|\mathbf{x}) &\sim \mathcal{N}(\mathbf{A}\mathbf{x}, \sigma^2\mathbf{I}). \end{aligned}$$

From the known SMV model of $\mathbf{y}$ the above implies

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}, \quad \mathbf{e} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$$

From [**conditional_cov**] the conditional covariance is given by

$$\begin{aligned} \mathbf{\Sigma} &= \operatorname{cov}(\mathbf{x}, \mathbf{x}|\mathbf{y}) = \mathbf{\Sigma}_{\mathbf{xx}} - \mathbf{\Sigma}_{\mathbf{xy}}\mathbf{\Sigma}_{\mathbf{yy}}^{-1}\mathbf{\Sigma}_{\mathbf{yx}} \\ \boldsymbol{\mu} &= \boldsymbol{\mu}_{\mathbf{x}} + \mathbf{\Sigma}_{\mathbf{xy}}\mathbf{\Sigma}_{\mathbf{yy}}^{-1}(\mathbf{y} - \boldsymbol{\mu}_{\mathbf{y}}) \end{aligned}$$

Each of the covariances within the expressing will now be found.

Covariance $\mathbf{\Sigma}_{\mathbf{xx}}$ The covariance of $\mathbf{x}$ comes directly from the distribution

$$\mathbf{\Sigma}_{\mathbf{xx}} = \boldsymbol{\gamma}\mathbf{I}$$

Covariance $\boldsymbol{\Sigma}_{\mathbf{xy}}$ The covariance between $\mathbf{x}$ and $\mathbf{y}$ is found by using the linearity of covariance and $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$:

$$\begin{aligned}\boldsymbol{\Sigma}_{\mathbf{yx}} &= \operatorname{cov}(\mathbf{x}, \mathbf{A}\mathbf{x} + \mathbf{e}) \\ &= \operatorname{cov}(\mathbf{x}, \mathbf{A}\mathbf{x}) + \operatorname{cov}(\mathbf{x}, \mathbf{e}) \\ &= \boldsymbol{\Sigma}_{\mathbf{xx}}\mathbf{A}^T \\ &= \boldsymbol{\gamma}\mathbf{I}\mathbf{A}^T\end{aligned}$$

where $\operatorname{cov}(\mathbf{x}, \mathbf{e}) = 0$ because $\mathbf{x}$ and $\mathbf{e}$ are uncorrelated.

Covariance $\boldsymbol{\Sigma}_{\mathbf{yx}}$ The covariance between $\mathbf{y}$ and $\mathbf{x}$ is defined by the transpose of $\boldsymbol{\Sigma}_{\mathbf{yx}}$

$$\begin{aligned}\boldsymbol{\Sigma}_{\mathbf{xy}} &= \left(\boldsymbol{\gamma}\mathbf{I}\mathbf{A}^T\right) \\ &= \mathbf{A}\boldsymbol{\gamma}\mathbf{I}\end{aligned}$$

Covariance $\boldsymbol{\Sigma}_{\mathbf{yy}}$ Lastly the covariance of $\mathbf{y}|\mathbf{x}$ is similarly found using again the linearity of covariance:

$$\begin{aligned}\boldsymbol{\Sigma}_{\mathbf{yy}} &= \operatorname{cov}(\mathbf{A}\mathbf{x} + \mathbf{e}, \mathbf{A}\mathbf{x} + \mathbf{e}) \\ &= \operatorname{cov}(\mathbf{A}\mathbf{x}, \mathbf{A}\mathbf{x}) + \operatorname{cov}(\mathbf{A}\mathbf{x}, \mathbf{e}) + \operatorname{cov}(\mathbf{e}, \mathbf{A}\mathbf{x}) + \operatorname{cov}(\mathbf{e}, \mathbf{e}) \\ &= \mathbf{A}\boldsymbol{\Sigma}_{\mathbf{xx}}\mathbf{A}^T + \boldsymbol{\Sigma}_{\mathbf{ee}} \\ &= \mathbf{A}\boldsymbol{\gamma}\mathbf{I}\mathbf{A}^T + \sigma^2\mathbf{I}\end{aligned}$$

By combining all the found conditional covariances the resulting covariance becomes

$$\boldsymbol{\Sigma} = \boldsymbol{\gamma}\mathbf{I} - \boldsymbol{\gamma}\mathbf{I}\mathbf{A}^T(\mathbf{A}\boldsymbol{\gamma}\mathbf{I}\mathbf{A}^T + \sigma^2\mathbf{I})^{-1}\mathbf{A}\boldsymbol{\gamma}\mathbf{I}.$$

The resulting mean under the assumption that $\boldsymbol{\mu}_{\mathbf{y}} = \mathbf{0}$ becomes

$$\begin{aligned}\boldsymbol{\mu}_{\cdot j} &= \mathbf{0} + \boldsymbol{\gamma}\mathbf{I}\mathbf{A}^T\left(\boldsymbol{\Sigma}_{\mathbf{yy}}\right)^{-1}(\mathbf{y} - \mathbf{0}) \\ &= \boldsymbol{\gamma}\mathbf{I}\mathbf{A}^T\left(\boldsymbol{\Sigma}_{\mathbf{yy}}\right)^{-1}\mathbf{y}.\end{aligned}$$

Appendix C

Independent Component Analysis

This appendix provides the basic theory of independent component analysis (ICA). The theory is necessary if one wants a deeper understanding towards the justification of using the result from ICA as a reference for evaluation of the main algorithm proposed in this thesis. The appendix concludes with an algorithm specifying ICA method applied in the thesis. Additionally, a verification test is conducted to evaluate the applied ICA method on the synthetic data, cf. section 6.2.

C.1 Basic Theory of Independent Component Analysis

Independent component analysis (ICA) is a method that applies to the general problem of decomposition of a measurement vector into a source vector and a mixing matrix. The intention of ICA is to separate a multivariate signal into statistical independent and non-Gaussian signals. And identify the mixing matrix $\mathbf{A}$, given only the observed measurements $\mathbf{Y}$. A well-known application example of source separation is the cocktail party problem, where it is sought to listen to one specific person speaking in a room full of people having interfering conversations. Let $\mathbf{y} \in \mathbb{R}^M$ be a single measurement from M microphones containing a linear mixture of all the speak signals that are present in the room. When additional noise is not considered, the problem can be described as the familiar linear system

$$\mathbf{y} = \mathbf{A}\mathbf{x}, \tag{C.1}$$

where $\mathbf{x} \in \mathbb{R}^N$ contain the N underlying speak signals. $\mathbf{A}$ is a mixing matrix where the coefficients may depend on the distances from a source to the microphone. As such each y_i is a weighted sum of all the sources of speak present to the i-th microphone. By ICA both the mixing matrix $\mathbf{A}$ and the source signals $\mathbf{x}$ are sought estimated from the observed measurements $\mathbf{y}$. The main attribute of ICA is the assumption

that the sources in $\mathbf{x}$ are statistically independent and non-Gaussian distributed, hence the name independent components.
By independence, one means in general that changes in one source signal do not affect the other source signals. In theory N variables $x_1, \ldots, x_N$ is independent if the joint probability density function (pdf) of $\mathbf{x}$ satisfies

$$p(x_1, x_2, \ldots, x_N) = p_1(x_1)p_2(x_2)\cdots p_n(x_N).$$

The possibility of separating a signal into independent and non-Gaussian components originates from the central limit theorem [18, p. 34]. The theorem states that the distribution of any linear mixture of two or more independent random variables tends toward a Gaussian distribution, under certain conditions. For instance the distribution of a mixture of two independent random variables is always closer to a Gaussian distribution than the original variables. In other word the original variables is most non-Gaussian. The application of the central limit theorem within ICA will be elaborated later in this appendix.

C.1.1 Assumptions and Preprocessing

For simplicity assumes that $\mathbf{A}$ is square i.e. $M = N$ and invertible. As such when $\mathbf{A}$ has been estimated the inverse is computed and the components can simply be estimated as $\mathbf{x} = \mathbf{A}^{-1}\mathbf{y}$ [18, p. 152-153]. As both $\mathbf{A}$ and $\mathbf{x}$ are unknown the variances of the independent components can not be determined. However, it is reasonable to assume that $\mathbf{x}$ has unit variance – $\mathbf{A}$ is assumed to have unit variance as well. Any scalar multiplier within a source can be canceled out by dividing the corresponding column in $\mathbf{A}$ with the same scalar [18, p. 154]. For further simplification it is assumed without loss of generality that $\mathbb{E}[\mathbf{y}] = 0$ and $\mathbb{E}[\mathbf{x}] = 0$[18, p. 154]. In case this assumption is not true, the measurements can be centered by subtracting the mean as preprocessing before doing ICA.

A preprocessing step central to ICA is to whiten the measurements $\mathbf{y}$. By the whitening process any correlation in the measurements are removed and unit variance is ensured – the independent components $\mathbf{x}$ becomes uncorrelated and have unit variance. Furthermore, this reduces the complexity of ICA and therefore simplifies the recovering process. Whitening is a linear transformation of the observed data. This is multiplying the measurement vector $\mathbf{y}$ with a whitening matrix $\mathbf{V}$

$$\mathbf{y}_{\text{white}} = \mathbf{V}\mathbf{y},$$

to obtain a new measurement vector $\mathbf{y}_{\text{white}}$ which is considered white. To obtain a whitening matrix, the eigenvalue decomposition (EVD) of the covariance matrix can be used:

$$\mathbb{E}[\mathbf{y}\mathbf{y}^T] = \mathbf{E}\mathbf{D}\mathbf{E}^T,$$

where $\mathbf{D}$ is a diagonal matrix of eigenvalues and $\mathbf{E}$ is a matrix consisting of the associated eigenvectors. From $\mathbf{E}$ and $\mathbf{D}$ a whitening matrix $\mathbf{V}$ is constructed as

$$\mathbf{V} = \mathbf{E}\mathbf{D}^{-1/2}\mathbf{E}^T,$$

where $\mathbf{D}^{-1/2} = \text{diag}(d_1^{-1/2}, \ldots, d_n^{-1/2})$ is a component-wise operation [18, p. 159]. By multiplying the measurement vector $\mathbf{y}$ with a whitening matrix $\mathbf{V}$ the data becomes white

$$\mathbf{y}_{\text{white}} = \mathbf{V}\mathbf{y} = \mathbf{V}\mathbf{A}\mathbf{x} = \mathbf{A}_{\text{white}}\mathbf{x}.$$

Furthermore, the corresponding mixing matrix $\mathbf{A}_{\text{white}}$ becomes orthogonal

$$\mathbb{E}[\mathbf{y}_{\text{white}}\mathbf{y}_{\text{white}}^T] = \mathbf{A}_{\text{white}}\mathbb{E}[\mathbf{x}\mathbf{x}^T]\mathbf{A}_{\text{white}}^T = \mathbf{A}_{\text{white}}\mathbf{A}_{\text{white}}^T = \mathbf{I},$$

where $\mathbb{E}[\mathbf{x}\mathbf{x}^T] = \mathbf{I}$ due to $\mathbf{x}$ having zero mean and uncorrelated entries.

As a consequence ICA can restrict its search for the mixing matrix to the orthogonal matrix space. That is instead of estimating N^2 parameters ICA has only to estimate an orthogonal matrix which has $N(N-1)/2$ parameters [18, p. 159].

C.1.2 Recovery of the Independent Components

The estimation of the mixing coefficients a_{ij} and the independent components x_i by ICA is now elaborated, based on [18, p. 166].

The simple and intuitive method is to take advantage of the assumption of non-Gaussian independent components. Consider again the model of a single measurement vector $\mathbf{y} = \mathbf{A}\mathbf{x}$, where the data vector complies to the assumption of being mixture of independent components. Here the independent components can be estimated by the inverted model

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{y}.$$

Consider first the estimation of a single independent component x_i. Here a linear combination of y_i is considered. Denote for now a single independent component by z such that

$$z = \mathbf{b}^T\mathbf{y} = \sum_k b_k y_k, \tag{C.2}$$

where one want to determine the vector $\mathbf{b}$. This can be rewritten to

$$z = \mathbf{b}^T\mathbf{A}\mathbf{x}$$

From this it is seen that $\mathbf{y}$ is a linear combination of the x_i with coefficients given by the vector $\mathbf{b}^T\mathbf{A}$. Let now $\mathbf{b}^T\mathbf{A}$ be denoted by the $\mathbf{q}$. As such

$$z = \mathbf{b}^T\mathbf{y} = \mathbf{b}^T\mathbf{A}\mathbf{x} = \mathbf{q}^T\mathbf{x} = \sum_k q_k x_k. \tag{C.3}$$

By this expression, consider the thought of $\mathbf{b}$ being one of the rows in $\mathbf{A}^{-1}$, then the linear combination $\mathbf{b}^T\mathbf{y}$ is equal to one of the independent components.

The objective is now to apply the central limit theorem to determine $\mathbf{b}$ such that it equals one of the rows of $\mathbf{A}^{-1}$. As $\mathbf{A}$ is unknown it is not possible to determine $\mathbf{b}$ exactly, but an estimate can be found to make a good approximation.

Due to z denoting some x_i it is clear that the equality in (C.3) only holds true when $\mathbf{q}$ consist of only one non-zero element equal to 1. Thus, from the central limit theorem the distribution of $\mathbf{q}^T\mathbf{x}$ is most non-Gaussian when it equals one of the independent components which was assumed non-Gaussian. As such, since $\mathbf{q}^T\mathbf{x} = \mathbf{b}^T\mathbf{y}$, it is possible to vary the coefficients in $\mathbf{b}$ and look at the distribution of $\mathbf{b}_i\cdot\mathbf{y}$. Finding the vector $\mathbf{b}^T$ that maximizes the non-Gaussianity would then correspond to $\mathbf{q} = \mathbf{A}^T\mathbf{b}^T$ having only a single non-zero element. Thus maximizing the non-Gaussianity of $\mathbf{b}_i\cdot\mathbf{y}$ results in one of the independent components [18, p. 166]. Considering the N-dimensional space of vectors $\mathbf{b}^T$ there exist $2N$ local maxima, corresponding to x_i and $-x_i$ for all N independent components [18, p. 166].

C.1.3 Kurtosis

To maximize the non-Gaussianity, a measure for Gaussianity is needed. Kurtosis is a quantitative measure used for non-Gaussianity of random variables [18, p. 171]. Kurtosis of a random variable y is the fourth-order cumulant denoted by $\text{kurt}(y)$. For y with zero mean and unit variance, kurtosis reduces to

$$\text{kurt}(y) = \mathbb{E}[y^4] - 3.$$

It is seen that the kurtosis is a normalized version of the fourth-order moment defined as $\mathbb{E}[y^4]$. For a Gaussian random variable the fourth-order moment equals $3(\mathbb{E}[y^2])^2$ hence the corresponding kurtosis will be zero [18, p. 171]. Consequently, the kurtosis of non-Gaussian random variables will almost always be different from zero.
The kurtosis is a common measure for non-Gaussianity due to its simplicity both theoretical and computational. The kurtosis can be estimated computationally by the fourth-order moment of sample data when the variance is constant. Furthermore, for two independent random variables x_1, x_2 the following linear properties applies to the kurtosis of the sum

$$\text{kurt}(x_1 + x_2) = \text{kurt}(x_1) + \text{kurt}(x_2) \quad \text{and} \quad \text{kurt}(\alpha x_1) = \alpha^4\text{kurt}(x_1)$$

However, one complication concerning kurtosis as a measure is that kurtosis is sensitive to outliers [18, p. 182].

Consider from (C.3) the vector $\mathbf{q} = \mathbf{A}^T\mathbf{b}$ such that $\mathbf{b}^T\mathbf{y} = \sum_{k=1} q_k x_k$. By the additive property of kurtosis

$$\text{kurt}\left(\mathbf{b}^T\mathbf{y}\right) = \sum_{k=1} q_k^4 \text{kurt}(x_k).$$

Then the assumption of the independent components having unit variance results in $\mathbb{E}[x_i^2] = \sum_{k=1} q_k^2 = 1$. That is geometrically that $\mathbf{q}$ is constrained to the unit sphere, $\|\mathbf{q}\|^2 = 1$.

By this an optimization problem maximizing the kurtosis of $\mathbf{b}^T\mathbf{y}$ is similar to maximizing $|\text{kurt}(x_i)| = |\sum_{k=1} q_k^4 \text{kurt}(x_k)|$ on the unit sphere. Due to the described preprocessing $\mathbf{b}^T$ is assumed to be white and it can be shown that $\|\mathbf{q}\| = \|\mathbf{b}^T\|$ [18, p. 174]. This shows that constraining $\|\mathbf{q}\|$ to one is similar to constraining $\|\mathbf{b}^T\|$ to one.

C.1.4 Basic ICA algorithm

Now a basic ICA algorithm is specified, this algorithm is based on the gradient optimization method with kurtosis.

The general idea behind a gradient algorithm is to determine the direction for which $\text{kurt}(\mathbf{b}^T\mathbf{y})$ is growing the most, based on the gradient. The gradient of $|\text{kurt}(\mathbf{b}^T\mathbf{y})|$ is computed as

$$\frac{\partial|\text{kurt}(\mathbf{b}^T\mathbf{y})|}{\partial\mathbf{b}} = 4\text{sign}(\text{kurt}(\mathbf{b}^T\mathbf{y}))(\mathbb{E}[\mathbf{y}(\mathbf{b}^T\mathbf{y})^3] - 3\mathbf{y}\mathbb{E}[(\mathbf{b}^T\mathbf{y})^2]) \tag{C.4}$$

As $\mathbb{E}[(\mathbf{b}^T\mathbf{y})^2] = \|\mathbf{y}\|^2$ for whitened data the corresponding term does only affect the norm of $\mathbf{b}$ within the gradient algorithm. Thus, as it is only the direction that is of interest, this term can be omitted. Because the optimization is restricted to the unit sphere a projection of $\mathbf{b}$ onto the unit sphere must be performed in every step of the gradient method. This is done by dividing $\mathbf{b}$ by its norm. This gives update step [18, p. 178]

$$\begin{aligned}\Delta\mathbf{b} &\propto \text{sign}\left(\text{kurt}(\mathbf{b}^T\mathbf{y})\right)\mathbb{E}[\mathbf{y}(\mathbf{b}^T\mathbf{y})^3] \\ \mathbf{b} &\leftarrow \mathbf{b}/\|\mathbf{b}\|\end{aligned}$$

The expectation operator can be omitted in order to achieve an adaptive version of the algorithm, now using every measurement $\mathbf{y}$. However, the expectation operator from the definition of kurtosis can not be omitted and must therefore be estimated. This can be done by γ by serving it as the learning rate of the gradient method.

$$\Delta\gamma \propto ((\mathbf{b}^T\mathbf{y})^4 - 3) - \gamma$$

Algorithm 4 combines the above theory, to give an overview of the basic ICA procedure.

Algorithm 4 Basis ICA

procedure PRE-PROCESSING($\mathbf{y}$)
    Center measurements $\mathbf{y} \leftarrow \mathbf{y} - \bar{\mathbf{y}}$
    Whitening $\mathbf{y} \leftarrow \mathbf{y}_{\text{white}}$
end procedure

procedure ICA($\mathbf{y}$)
    $k = 0$
    Initialize random vector $\mathbf{b}_{(k)}$ ▷ unit norm
    Initialize random value $\gamma_{(k)}$
    for $j \leftarrow 1, 2, \cdots, N$ **do**
        while convergence critia not meet **do**
            $k = k + 1$
            $\mathbf{b}_{(k)} \leftarrow \text{sign}\gamma_{(k-1)}\mathbf{y}(\mathbf{b}_{(k-1)}\mathbf{y})^3$
            $\mathbf{b}_{(k)} \leftarrow \mathbf{b}_{(k)}/\|\mathbf{b}_{(k)}\|$
            $\gamma_{(k)} \leftarrow ((\mathbf{b}_{(k)}\mathbf{y})^4 - 3) - \gamma_{(k-1)}$
        end while
        $x_j = \mathbf{b}^T\mathbf{y}$
    end for
end procedure

C.1.5 ICA for sparse signal recovery

ICA is widely used within sparse signal recovery. When ICA is applied to a measurement vector $\mathbf{y} \in \mathbb{R}^M$ it is possible to separate the mixed signal into M or less independent components. However, by assuming that the independent components make a k-sparse signal it is possible to apply ICA within sparse signal recovery of cases where $M < N$ and $k \leq M$.
To apply ICA to such cases, the independent components are obtained by the pseudo-inverse solution

$$\hat{\mathbf{x}} = \mathbf{A}_S^{\dagger}\mathbf{y}$$

where $\mathbf{A}_S$ is derived from the dictionary matrix $\mathbf{A}$ by containing only the columns associated with the non-zero entries of $\mathbf{x}$, specified by the support set S, cf. appendix A.1.

C.2 Fixed-Point Algorithm - FastICA

An advantage of gradient algorithms is the possibility of fast adoption in non-stationary environments due the use of all input, $\mathbf{y}$, at once. A disadvantage of the gradient algorithm is the resulting slow convergence, depending on the choice of

γ for which a bad choice in practice can disable convergence. A fixed-point iteration algorithm to maximize the non-Gaussianity is an alternative that could be used.

Consider the gradient step derived in section C.1.4. In the fixed-point iteration the sequence of γ is omitted and replaced by a constant. This builds upon the fact that for a stable point of the gradient algorithm the gradient must point in the direction of $\mathbf{b}$, hence be equal to $\mathbf{b}$. In this case adding the gradient to $\mathbf{b}$ does not change the direction and convergence is achieved. Letting the gradient given in (C.4) be equal to $\mathbf{b}$ and considering the same simplifications again suggest the new update step as [18, p. 179]

$$\mathbf{b} \leftarrow \mathbb{E}[\mathbf{y}(\mathbf{b}^T\mathbf{y})^3] - 3\mathbf{b}.$$

After the fixed-point iteration $\mathbf{b}$ is again divided by its norm to withhold the constraint $\|\mathbf{b}\| = 1$. Instead of γ the fixed-point algorithm compute $\mathbf{b}$ directly from previous $\mathbf{b}$.

The fixed-point algorithm is referred to as FastICA. The algorithm has shown to converge fast and reliably, when the current and previous $\mathbf{b}$ point in the same direction [18, p. 179].

C.2.1 Negentropy

An alternative measure of non-Gaussianity is the negentropy, based on the differential entropy. The differential entropy H of a random vector $\mathbf{y}$ with density $p_y(\boldsymbol{\eta})$ is defined as

$$H(\mathbf{y}) = -\int p_y(\boldsymbol{\eta}) \log(p_y(\boldsymbol{\eta}))\, d\boldsymbol{\eta}.$$

The entropy describes the information that a random variable gives. The more unpredictable and unstructured a random variable is higher is the entropy, e.g. Gaussian random variables have a high entropy, in fact they have the highest entropy among the random variables of the same variance [18, p. 182].

Negentropy is a normalised version of the differential entropy such that the measure of non-Gaussianity is zero when the random variable is Gaussian and non-negative otherwise. The negentropy J of a random vector $\mathbf{y}$ is defined as

$$J(\mathbf{y}) = H(\mathbf{y}_{\text{gauss}}) - H(\mathbf{y}),$$

with $\mathbf{y}_{\text{gauss}}$ being a Gaussian random variable of the same covariance and correlation as $\mathbf{y}$ [18, p. 182]. As the kurtosis is sensitive for outliers, the negentropy is instead difficult to compute computationally as the negentropy requires an estimate of the pdf. As such an approximation of the negentropy is needed. To approximate the negentropy, it is common to use the higher order cumulants including the kurtosis.

The following approximation of the scalar case is stated without further elaboration, and the derivation can be found in [18, p. 183].

$$J(\mathbf{y}) \approx \frac{1}{12}\mathbb{E}[y^3]^2 \frac{1}{48}\text{kurt}(y)^2.$$

C.2.2 Fixed-Point Algorithm with Negentropy

Maximization of negentropy by use of the fixed-point algorithm is now presented, for derivation of the fixed-point iteration see [18, p. 188]. Algorithm 5 show FastICA using negentropy. This is the algorithm applied for comparison with the source recovery methods tested in this thesis.

Algorithm 5 FastICA – with negentropy

procedure PRE-PROCESSING($\mathbf{y}$)
    Center measurements $\mathbf{y} \leftarrow \mathbf{y} - \bar{\mathbf{y}}$
    Whitening $\mathbf{y} \leftarrow \mathbf{y}_{\text{white}}$
end procedure

procedure FASTICA($\mathbf{y}$)
    $k = 0$
    Initialize random vector $\mathbf{b}_{(k)}$ ▷ unit norm
    for $j \leftarrow 1, 2, \cdots, N$ **do**
        while convergence critia not meet **do**
            $k = k + 1$
            $\mathbf{b}_{(k)} \leftarrow \mathbb{E}[\mathbf{y}(\mathbf{b}_{(k-1)}^T \mathbf{y})] - \mathbb{E}[g'(\mathbf{b}_{(k-1)}^T \mathbf{y})]\mathbf{b}_{(k-1)}$ ▷ g cf. [18, p. 190]
            $\mathbf{b}_{(k)} \leftarrow \mathbf{b}_{(k-1)} / \|\mathbf{b}_{(k-1)}\|$
        end while
        $x_j = \mathbf{b}^T \mathbf{y}$
    end for
end procedure

C.3 Verification of FastICA on Synthetic Data

The purpose of this section is to verify the FastICA algorithm used in this thesis. By this verification the purpose is to justify the FastICA algorithm as a reference point with respect to performance of the developed main algorithm.

The FastICA algorithm is tested on synthetic data simulated as described in section 6.2. Consider the following linear system, which makes a model of EEG measurements

$$\mathbf{Y} = \mathbf{AX},$$

where $\mathbf{Y} \in \mathbb{R}^{M \times L}$, $\mathbf{A} \in \mathbb{R}^{M \times N}$ and $\mathbf{X} \in \mathbb{R}^{N \times L}$. It is expected that the FastICA algorithm manages to solve the linear system for $\mathbf{X}$ and $\mathbf{A}$ given only the measurements $\mathbf{Y}$, in the case where $M = N$.

The FastICA algorithm is applied to $\mathbf{Y}$ and returns the estimates $\hat{\mathbf{X}}_{\text{ICA}}$ and $\hat{\mathbf{A}}_{\text{ICA}}$. When using the FastICA algorithm the output $\hat{\mathbf{X}}_{\text{ICA}}$ do not correspond one to one with the true source signals, which become an issue when the estimation error is measured by the mean squared error (MSE) cf. section 6.2.3. The FastICA algorithm is invariant towards the amplitude and phase of the source signal. Furthermore, the rows are not necessarily place the exact location. In order to get a valid MSE measure of the estimate, a function is defined to fit the estimate to the true source signal $\mathbf{X}$. The function manages to pair the rows and change the phase, such that the total MSE is minimized. Furthermore, each row of the estimate is scaled by the relationship between the maximum value of the true row and the estimated row. From empirical observations only the phase shift performed by multiplying with (-1) has shown necessarily, hence it is easily applied to the fitting function. When the fitting function is applied two the estimate, the full potential of the FastICA algorithm is considered reached.

Figure C.1 illustrates $\hat{\mathbf{X}}_{\text{ICA}}$, without use of the fitting function, resulting from the FastICA algorithm applied to a simulated deterministic data set $\mathbf{Y}$ specified by $M = N = k = 4$ and $L = 1000$. In the figure $\mathbf{Y}$, $\mathbf{X}$ and $\hat{\mathbf{X}}_{\text{ICA}}$ are plotted separately and it is clear to see the invariance towards amplitude and phase. The MSE from the original $\hat{\mathbf{X}}_{\text{ICA}}$ becomes

$$\text{MSE}(\mathbf{X}, \hat{\mathbf{X}}_{\text{ICA}}) = 0.608.$$

In figure C.2 the fitting function has been applied to $\hat{\mathbf{X}}_{\text{ICA}}$. Each row of the fitted estimate is now plotted with the corresponding row of the true source signals. The resulting MSE becomes

$$\text{MSE}(\mathbf{X}, \hat{\mathbf{X}}_{\text{ICA}}) = 0.046.$$

This is an essential change from the first measured MSE, and it is considered to provide a more valid measure of the estimate. From the visualization and the corresponding MSE, it is found that the FastICA algorithm manages to estimate the source signals of the deterministic data set with a sufficiently small error.

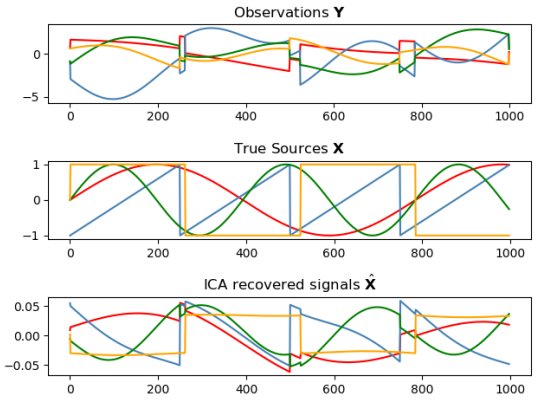

Figure C.1: Plot of simulated deterministic data set $\mathbf{Y}$, specified by $M = N = k = 4$ and $L = 1000$. Corresponding plot of the true $\mathbf{X}$ and the estimated $\hat{\mathbf{X}}$ by ICA.

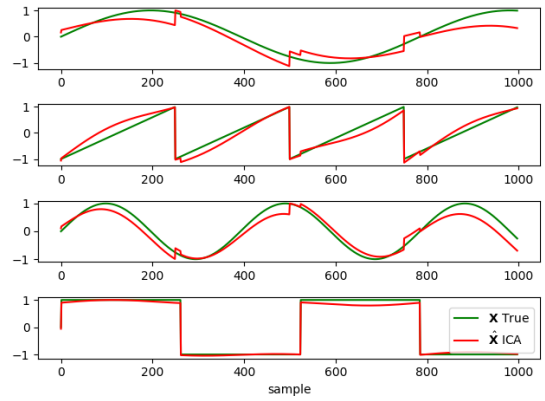

Figure C.2: Direct comparison of the true $\mathbf{X}$ and $\hat{\mathbf{X}}_{\text{ICA}}$ after applying the fitting function.

A similar test is now performed on a stochastic data set $\mathbf{Y}$, cf. section 6.2.2, again specified by $M = N = k = 4$ and $L = 1000$. Figure C.3 show the comparison of the fitted $\hat{\mathbf{X}}_{\text{ICA}}$ and the true source signals $\mathbf{X}$. Note that only the first 100 samples are plotted for easier visualization. The resulting MSE becomes:

$$\text{MSE}(\mathbf{X}, \hat{\mathbf{X}}_{\text{ICA}}) = 0.037.$$

Again the MSE is considered sufficiently small and by that the FastICA is considered verified with respect to solving a linear system with $M = N$.

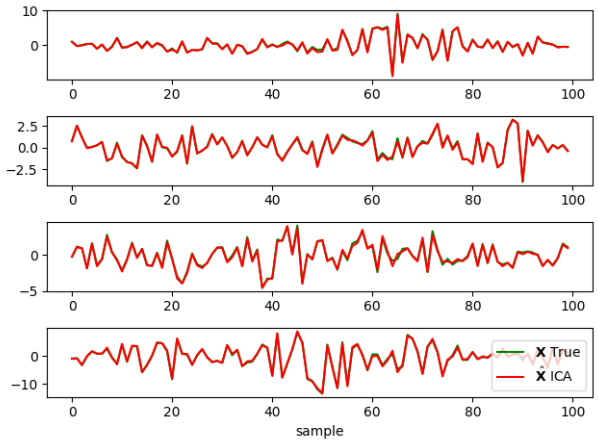

Figure C.3: ICA applied to simulated stochastic data set $\mathbf{Y}$, specified by $M = N = k = 4$ and $L = 1000$ with direct comparison of the true $\mathbf{X}$ and $\hat{\mathbf{X}}_{\text{ICA}}$ after applying the fitting function.

Consider now the case where $k \leq N = M$, that is the sources signal matrix has k

non-zeros rows. The FastICA algorithm is now applied to a stochastic data set $\mathbf{Y}$ specified by $N = M = 6$, $k = 4$ and $L = 1000$. Figure C.4 and C.5 show the comparison of the resulting $\hat{\mathbf{X}}_{\text{ICA}}$ and the true $\mathbf{X}$ before and after the application of the fitting function, respectively. The resulting MSE becomes:

$$\text{MSE}(\mathbf{X}, \hat{\mathbf{X}}_{\text{ICA}}) = 1.784.$$

It is seen from figure C.5 that the FastICA algorithm manages to detect the zero rows of $\mathbf{X}$. Without further test, this indicates the possibility of estimating k from the FastICA algorithm.

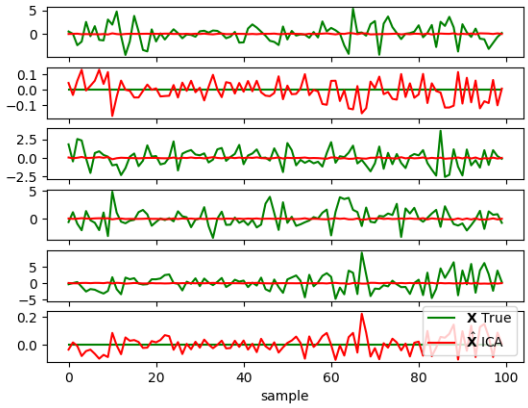

Figure C.4: Plot of simulated stochastic data set $\mathbf{Y}$, specified by $M = N = 6$, $k = 4$ and $L = 1000$. Corresponding plot of the true $\mathbf{X}$ and the estimated $\hat{\mathbf{X}}$ by ICA.

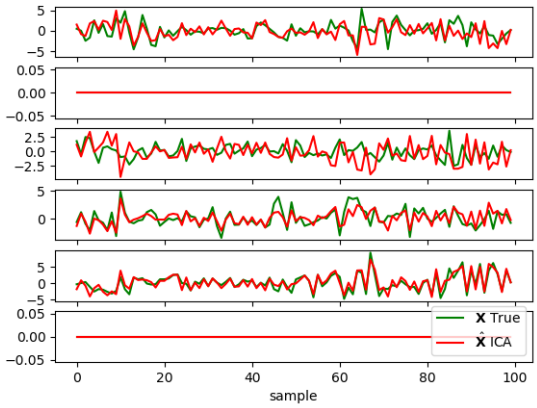

Figure C.5: Direct comparison of the true $\mathbf{X}$ and $\hat{\mathbf{X}}_{\text{ICA}}$ after applying the fitting function.

With these tests the quality of the FastICA algorithm has been verified. As such the FastICA algorithm can be used as a reference, when applied to real EEG data. It is further established that $k \leq M$ can be estimated by FastICA. Remember though that the ICA estimate is conditioned under $k \leq N = M$. However, this condition is not necessarily withheld for real EEG measurements as the true N is always unknown.

Appendix D

Python Scripts

The following list contains descriptions of the essential Python scripts which have been used through this thesis.

1. `Data_Simulation.py`: Contain the functions for simulation of deterministic and stochastic synthetic data, based on manual specification of each system.
2. `Data_EEG.py`: Contain the functions for import of EEG measurements, possible reduction and segmentation.
3. `ICA_Fast.py`: Contain the function to perform ICA.
4. `Main_Algorithm.py`: In this script the main algorithm composed by a compilation of the necessary modules. Remark, by letting `fix = True` Cov-DL is overwritten by $\hat{\mathbf{A}}_{\text{fix}}$.
    - `Cov_DL.py`: Contain the function to perform Cov-DL.
    - `M_SBL.py`: Contain the function to perform M-SBL.
5. `Test_Synthetic_data.py`: Compilation of necessary modules to apply the main algorithm to on synthetic data and generate output, cf. chapter 6.
6. `Test_EEG.py`: Compilation of necessary modules to apply ICA and the main algorithm on EEG measurements, for the three different cases, cf. section 7.2.
7. `Test_AlphaFrequency.py`: Contain the functions and corresponding compilation to perform alpha wave analysis, cf. section 7.4.
8. `Test_k_Estimation.py`: Contain the functions and corresponding compilation generate estimation of k, cf. chapter 8

The Python scripts are all available directly at `https://inset_link.git`. Besides the script, the folder contains the EEG measurements and a folder for which generated plots are allocated, is present.