Bayesian Dictionary Learning for EEG Source Identification

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:
Bayesian Dictionary Learning for EEG Source Identification

Theme:

Project Period:
Fall Semester 2019
Spring Semester 2020

Project Group:
Mattek9b

Participant(s):
Trine Nyholm Kragh
Laura Nyrup Mogensen

Supervisor(s):
Jan Østergaard
Rasmus Waagepetersen

Copies: 1

Page Numbers: 53

Date of Completion:
December 19, 2019

Abstract:

Here is the abstract

Matematik-Teknologi
Aalborg Universitet
http://www.aau.dk

AALBORG UNIVERSITET
STUDENTERRAPPORT

Titel:
Bayesian Bibliotek Læring for EEG Kilde Identifikation

Tema:

Abstract:

Her er resuméet

Projektperiode:
Efterårssemestret 2019
Forårssemestret 2020

Projektgruppe:
Mattek9b

Deltager(e):
Trine Nyholm Kragh
Laura Nyrup Mogensen

Vejleder(e):
Jan Østergaard
Rasmus Waagepetersen

Oplagstal: 1

Sidetal: 53

Afleveringsdato:
19. december 2019

Preface

Here is the preface. You should put your signatures at the end of the preface.

Aalborg University, December 19, 2019

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

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

Danish Summary

Dansk resume ?

Contents

Todo list

Introduction

Introduktion til hele projektet, skal kunne læses som en appetitvækker til resten af rapporten, det vi skriver her skal så uddybes senere. Brug dog stadigvæk kilder.

- kort intro a EEG og den brede anveldelse, anvendelse indenfor høreapperat.
- intro af model, problem med overbestemt system
- Seneste forslag til at løse dette
- vi vil efterviser dette og udvide til realtime tracking
- opbygningen af rapporten

Chapter 1

Motivation

This chapter examines existing literature concerning source recovery from Electroencephalography (EEG) measurements. At first a motivation for the source recovery problem is given, considering the application within the hearing aid industry. Further, the state of the art methods are presented followed by a description of the contribution proposed in this thesis.

1.1 EEG Measurements

EEG is a technique used within the medical field. It is an imaging technique measuring electric signals on the scalp, caused by brain activity. The human brain consist of an enormous amounts of cells, called neurons. These neurons are mutually connected in neural nets and when a neuron is activated, for instance by a physical stimuli, local current flows are produced [21]. This is also considered as a neural interaction across different parts of the human brain.
EEG measurements are provided by a number of metal electrodes, referred to as sensors, carefully placed on a human scalp. Each sensor reads the present electrical signals over time.
It takes a large amount of active neurons to generate an electrical signal that is recordable on the scalp as the current have to penetrate the skull, skin and several other thin layers. Hence it is clear that the EEG measurements from a single sensor do not correspond to the activity of one specific neuron in the brain, but rather a collection of many activities within the range of the one sensor. Nor is the range of a single sensor separated from the other sensors. Thus the same activity can easily be measured by two or more sensors. Furthermore, interfering signals can occur in the measurements resulting from physical movement of e.g. eyes and jawbone [21]. Lastly, the transmission of the electric field through the biological tissue to the sensor has an unknown mixing effect on the signal. This process is called volume conduction [18, p. 68] [19].

This clarifies the mixture of electrical signals with noise that form the EEG measurements. The concept is sought illustrated on figure 1.1.
It will be clear later that it is of interest to separate and localize the sources of the neural activities measured on the scalp.
The waves resulting from EEG measurements 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 extensively 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 [18, p. 11]. An example of EEG measurements within the four categories is illustrated by figure 1.2.

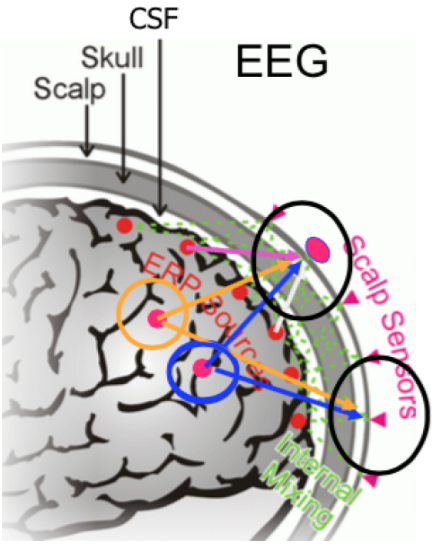

Figure 1.1: Illustration of volume conduction, source: [6](we will make our own figure here instead)

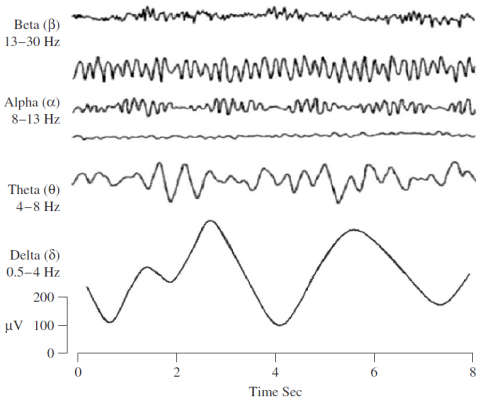

Figure 1.2: Example of time dependent EEG measurements within the four defined categories, source: [18]

EEG is widely used in the medical field, especially within research of the cognitive processes in the brain. Diagnosis and management of neurological disorders such as epilepsy is one example of application.
EEG capitalizes on the procedure being non-invasive and fast. Neural activity can be measured within fractions of a second after a stimuli has been provided [21, p. 3]. When a person is exposed to a certain stimulus, e.g. visual or audible, the measured activity is said to result from evoked potential. Over the past two decades, functional integration has become an area of interest [11]. 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 [12]. For this purpose separation and localization of the single sources which contribute to the EEG measurement is of interest. An article from 2016 [19] points out the importance of performing analysis regarding functional integration at source level rather than at EEG level. It is argued through experiments that analysis at EEG level does not allow interpretations about the interaction between sources.
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 [20]. One main purpose at Eriksholm is to make it possible for a hearing aid to identify the user-inttended sound source and thereby exclude noise from elsewhere [2] [7]. This is where EEG and occasionally so called in-ear EEG is interesting, especially in conjunction with the technology of beamforming. With beamforming it is possible for a hearing aid to receive only signals from a specific direction. It is essentially the well known but unsolved cocktail problem which is sought improved by use of EEG.

1.1.1 Modelling

Consider the issue of localizing activated sources from EEG measurements. A known approach is to model the observed data by the following linear system

$$\mathbf{Y} = \mathbf{A}\mathbf{X}.$$

$\mathbf{Y} \in \mathbb{R}^{M \times L}$ is the EEG measurements of L samples over time each consisting of M measurements, one for each sensor. $\mathbf{A} \in \mathbb{R}^{M \times N}$ is an unknown mixing matrix and $\mathbf{X} \in \mathbb{R}^{N \times N_d}$ is makes the actual activation of sources within the brain. The i-th column of $\mathbf{A}$ represent the relative projection weights from the i-th source to every sensor [6]. This is in general referred to as a multiple measurement vector model. The aim in this case is to identify both $\mathbf{A}$ and $\mathbf{X}$ given the measurements $\mathbf{Y}$. For this specific set up is referred to as the EEG inverse problem.

To solve the EEG inverse problem the concept of sparse signal recovery makes a solid foundation, including compressive sensing and dictionary learning. Independent Component Analysis (ICA) is a commonly applied method to solve the inverse problem [15], [14]. Here statistical independence between source activity is the essential assumption.
Application of ICA have shown great results regarding source separation of high-density EEG.
However, a significant flaw to this method is that the EEG measurements are only separated into a number of sources that are equal or less than the number of sensors [4]. This means that the EEG inverse problem can not be over-complete. That is an assumption which undermines the reliability and usability of ICA, as the number of simultaneous active sources easily exceed the number of sensors [6]. This is especially

a drawback when low-density EEG are considered, that is EEG equipment with less than 32 sensors. Improved capabilities of low-density EEG devices are desirable due to its relative low cost, mobility and ease to use.

This makes a foundation to look at the existing work considering the over-complete inverse EEG problem.

1.2 Related Work and Our Contribution

As mentioned above ICA has been a solid method for source localization in the case where a separation into a number of sources equal to the number of sensors was adequate. To overcome this issue an extension of ICA was suggested, referred to as the ICA mixture model [4]. Instead of identifying one mixing matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ this approach learns N_{model} different mixing matrices $\mathbf{A}_i \in \mathbb{R}^{M \times M}$. The method was further adapted into the Adaptive Mixture ICA (AMICA) which showed successful results regarding identification of more sources than available sensors [17]. However an assumption of no more than M simultaneously active sources has to be made which is still an essential limitation, especially when considering low-density EEG.
Other types of over-complete ICA algorithms have been proposed to overcome the problem of learning over-complete systems. One is the Restricted ICA (RICA), an efficient method used for unsupervised learning in neural networks [22]. Here the hard orthonormal constraint in ICA is replaced with a soft reconstruction cost.
In 2015 O. Balkan et. al., [4], suggested a new approach also targeting the identification of more sources than sensors regarding EEG. The suggested method, referred to as Cov-DL, is a covariance based dictionary learning algorithm. The point is to transfer the forward problem into the covariance domain, which has higher dimensionality than the original EEG sensor domain. This can be done when assuming the scalp mixing is linear and using the assumed natural non-correlation of sources within a certain time-window. The Cov-DL algorithm stands out from the other straight forward dictionary learning methods as it does not relay on the sparsity of active sources, this is an essential advantage when low-density EEG is considered.
Cov-DL was tested on found to outperform both AMICA and RICA [4], thus it is considered the state of the art within the area of source identification.

(number of sources? or datapoints)

(?)

It is essential to note that the Cov-DL algorithm do only learn the mixing matrix $\mathbf{A}$, the projection of sources to the scalp sensors, and not the explicit source activity time series $\mathbf{X}$.
For this purpose a multiple measurement sparse bayesian learning (M-SBL) algorithm was proposed in [5] also by O. Balkan et. al., also targeting the case of more active sources than sensors [5]. Here the mixing matrix which is known should fulfil the exact support recovery conditions. Though, the method was proven to outperform

the recently used algorithm M-CoSaMP even when the defined recovery conditions was not fulfilled.

Find ud af hvad forkortelsen står for

The two state of the art methods for source identification makes the foundation of this thesis. This thesis propose an algorithm with the purpose of solving the EEG inverse problem using the presented methods on EEG measurement. To extent the existing results the algorithm is expanded into a real-time application, in order to provide feedback based on the source activity.
The intention of the feedback is to adjust the direction of the beam within the hearing aid depending on the source activity. For this, the application is tested within a simulation environment where the receiving direction of the test person can be adjusted in real-time. The quality of the final results is measured by the capability of improving the listener experience and the time used to proved useful feedback.

As such our contribution *(hopefully)* consists of tests of existing methods on new real-time measurement and furthermore include a feedback to control the microphone beam on a hearing aid.

note: Evt. kunne vi lave en figur der lidt ala mindmap sætte et system overblik op og så highlighte de "bokse" vi vælger at arbejde med.

Chapter 2

Problem Statement

From the motivation and related work described in chapter 1 it is stated that EEG measurement of the brain activity has great potential to contribute within the hearing aid industry, regarding the development of hearing aids with improved performance in situations as the cocktail party problem. By solving the over-complete EEG inverse problem, in order to localize the sources of the brain activity, the results could be used to guide and adapt the hearing aids performance such as move the microphone beam in the direction of interest. This lead to the following problem statement.

How can sources of activation within the brain be localized from the EEG inverse problem, in the over-complete case of less sensors than sources and how can such algorithm be extended to a real-time application providing feedback to improve the intentional listening experience?

From the problem statement some clarifying sub-questions have been made.

- How can the over-complete EEG inverse problem be solved by use of compressive sensing included domain transformation?
- How can Cov-DL be used to estimate the mixing matrix $\mathbf{A}$ from the over-complete EEG inverse problem?
- How can M-SBL be used to estimate the source matrix $\mathbf{X}$ from the over-complete EEG inverse problem?
- How can an application be formed to constitute this source identification process operating in real-time?
- How can the feedback of the system be used to control the microphone beam of a simulated hearing aid. Especially how to analyse the feedback versus the listening experience in order to improve this.

Chapter 3

Sparse Signal Recovery

This chapter gives an introduction to the sparse signal recovery. Associated theory regarding compressive sensing is described along the common solution approaches and their limitations.

3.1 Linear Algebra

Some measurement vector $\mathbf{y}$ can be described as a linear combinations of a coefficient matrix $\mathbf{A}$ and some vector $\mathbf{x}$ such that

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

where $\mathbf{y} \in \mathbb{R}^M$ is the observed measurement vector consisting of M measurements, $\mathbf{x} \in \mathbb{R}^N$ is an unknown vector of N elements, and $\mathbf{A} \in \mathbb{R}^{M \times N}$ is a coefficient matrix which models the linear measurement process column-wise. 3.1 makes a system of linear equations with M equations and N unknowns and will be referred to as a linear model for the rest of the report.

In the case of $\mathbf{A}$ being a square matrix, $M = N$, a solution can be found to the linear model, provided that a solution exist, if $\mathbf{A}$ has full rank – $\mathbf{A}$ consist of linearly independent columns or rows. A linear model with $M = N$ is called determined, $M > N$ over-determined and $M < N$ under-determined. When full rank does not occur the matrix is called rank-deficient.

By inverting $\mathbf{A}$ from (3.1) the unknown vector $\mathbf{x}$ can be achieved. A square matrix is invertible if an only if it has full rank or equivalent its determinant $\det(\mathbf{A}) \neq 0$. For rectangular matrices, $M > N$ and $M < N$, left-sided and right-sided inverse exists.

For a determined system there will exist a unique solution. For an over-determined system there does not exist a solution and for under-determined systems there exist

infinitely many solutions [8, p. ix].
As described in chapter 1 the linear model of interest consists of M sensors which make the observed measurements $\mathbf{y}$ and N sources which make the unknown vector $\mathbf{x}$. Here it is of interest to find a solution to the case where the system consist of more sources than sensors – hence a solution has to be found within the infinite solution set.

3.2 Compressive Sensing

Compressive sensing is the theory of efficient recovery or reconstruction 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 original measurements, then the relation between the measurements and the signal to be recovered can be described by the linear model (3.1) [10].
In compressive sensing terminology, $\mathbf{x} \in \mathbb{R}^N$ is the signal of interest which is sought recovered from the measurements $\mathbf{y} \in \mathbb{R}^M$ by solving the linear system (3.1). The coefficient matrix $\mathbf{A}$ is in the context of compressive sensing referred to as the mixing matrix or the dictionary matrix. In the typical compressive sensing case the system is under-determined, $M < N$, and there exist infinitely many solutions, provided that a solution exist. However, by enforcing certain sparsity constraints it is possible to recover the wanted signal, hence the term sparse signal recovery [10].

3.2.1 Sparseness

A signal is said to be k-sparse if the signal has at most k non-zero coefficients. For the purpose of counting the non-zero entries of a vector representing a signal the ℓ_0-norm is defined

$$\|\mathbf{x}\|_0 := \text{card}(\text{supp}(\mathbf{x})).$$

The function $\text{card}(\cdot)$ gives the cardinality of the input and the support vector of $\mathbf{x}$ is given as

$$\text{supp}(\mathbf{x}) = \{j \in [N] \ : \ x_j \neq 0\},$$

where $[N]$ is a set of integers $\{1, 2, \cdots, N\}$ [10, p. 41]. The set of all k-sparse signals is denoted as

$$\Omega_k = \{\mathbf{x} \ : \ \|\mathbf{x}\|_0 \leq k\}.$$

3.2.2 Optimisation Problem

To find a solution to the linear model (3.1) assuming the solution is k-sparse, it can be viewed as an optimisation problem. The optimisation problem is defined as

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

$$\mathbf{x}^* = \arg\min_{\mathbf{x}\in\mathbf{\Omega}_k} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_0 \quad \text{subject to} \quad \|\mathbf{x}\|_0 \leq k.$$

The objective function is given by an ℓ_0 norm with the constraint function being the linear model (3.1). Unfortunately, this optimisation problem is non-convex due to the definition of ℓ_0-norm and is therefore difficult to solve – it is an NP-hard problem. Instead, by replacing the ℓ_0-norm with the ℓ_1-norm, the optimisation problem can be approximated and hence becomes computationally feasible [8, p. 27]

3.2: skal vi indfør z som en approximation til x. og så et nyt omega eller? eller kan vi beholde x

$$\mathbf{x}^* = \arg\min_{\mathbf{x}\in\mathbf{\Omega}_k} \|\mathbf{x}\|_1 \quad \text{subject to} \quad \mathbf{y} = \mathbf{A}\mathbf{x}. \tag{3.2}$$

With this optimisation problem we find the best k-sparse solution $\mathbf{x}^*$. This method is referred to as Basis Pursuit.
The following theorem justifies that the ℓ_1 optimisation problem finds a sparse solution [10, p. 62-63].

Theorem 3.2.1
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

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

To prove this theorem one needs to realise that the set $\{\mathbf{a}_j, j \in S\} \leq M$, with $S = \text{supp}(\mathbf{x}^*)$, can not have more than M linearly independence columns. This will be done by a contradiction. So when $M \ll N$ a sparse signal is automatically achieved.

Proof
Assume that the set $\{\mathbf{a}_l, l \in S\}$ 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 $t \neq 0$,

$$\|\mathbf{x}^*\|_1 < \|\mathbf{x}^* + t\mathbf{v}\|_1 = \sum_{l\in S} |x_l^* + tv_l| = \sum_{l\in S} \text{sgn}(x_l^* + tv_l)(x_l^* + tv_l). \tag{3.3}$$

For a small $|t|$

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

then the sign function become

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

By including this result in (3.3) 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. ■

The Basis Pursuit algorithm makes the foundation of several algorithms solving alternative versions of (3.2) where noise is incorporated. An alternative solution method includes greedy algorithms such as the Orthogonal Matching Pursuit (OMP) [10, P. 65]. At each iteration of the OMP algorithm an index set S is updated by adding the index corresponding to a column in $\mathbf{A}$ that best describes the residual, hence greedy. That is the part of $\mathbf{y}$ that is not yet explained by $\mathbf{Ax}$ is included. Then $\mathbf{x}$ is updated as the vector, supported by S, which minimize the residual, that is also the orthogonal projection of $\mathbf{y}$ onto the $\operatorname{span}\{\mathbf{a}_l \mid l \in S\}$. The algorithm for OMP can be found in the appendix 6.

3.2.3 Conditions on the Mixing Matrix

In section 3.2.2 the mixing matrix $\mathbf{A}$ was assumed known, in order to solve the optimisation problem (3.2). However, in practise it is only the measurement vector $\mathbf{y}$ which is known. In this case the mixing matrix $\mathbf{A}$ is considered a estimate of the true mixing matrix.
To ensure exact or approximately reconstruction of the sparse signal $\mathbf{x}$, the mixing matrix must be constructed with certain conditions in mind.

Null Space Condition

The null space property is a necessary and sufficient condition on $\mathbf{A}$ for exact reconstruction of every sparse signal $\mathbf{x}$ that solves the optimisation problem (3.2) [10, p. 77]. The null space of the matrix $\mathbf{A}$ is defined as

$$\mathcal{N}(\mathbf{A}) = \{\mathbf{z} \ : \ \mathbf{Az} = \mathbf{0}\}.$$

The null space property is defined as

Definition 3.1 (Null Space Property)
A matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ is said to satisfy the null space property relative to a set $S \subset [N]$ if

$$\|\mathbf{v}_S\|_1 < \|\mathbf{v}_{\overline{S}}\|_1 \quad \text{for all} \quad \mathbf{v} \in \text{null}(\mathbf{A}) \setminus \{\mathbf{0}\}, \tag{3.4}$$

where the vector $\mathbf{v}_S$ is the restriction of $\mathbf{v}$ to the indices in S, and $\bar{S}$ is the set $[N] \setminus S$.

Kom med en beskrivelse af nulrumsbetingelsen

Theorem 3.2.2
For a matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ with $S \subset [N]$, a vector $\mathbf{x} \in \mathbb{R}^N$ with $\text{supp}(\mathbf{x}) \subset S$ is the unique solution of (3.2) with $\mathbf{y} = \mathbf{A}\mathbf{x}$ if and only if $\mathbf{A}$ satisfies the null space property relative to S.

Proof
$\Rightarrow$:
Let $S \subseteq [N]$ be a fixed index set. Assume that a vector $\mathbf{x} \in \mathbb{R}^N$ with the support $\text{supp}(\mathbf{x}) \subset S$ is the unique minimizer of $\|\mathbf{z}\|_1$ with respect to $\mathbf{A}\mathbf{z} = \mathbf{A}\mathbf{x}$. Thus, for any vector $\mathbf{v} \in \text{null}(\mathbf{A}) \setminus \{\mathbf{0}\}$, the vector $\mathbf{v}_S$ is the unique minimizer of $\|\mathbf{z}\|_1$ with respect to $\mathbf{A}\mathbf{z} = \mathbf{A}\mathbf{v}_S$. But

$$\mathbf{0} = \mathbf{A}(\mathbf{v}_S + \mathbf{v}_{\overline{S}}) \implies \mathbf{A}\mathbf{v}_S = \mathbf{A}(-\mathbf{v}_{\overline{S}}), \quad \text{with} \ -\mathbf{v}_{\overline{S}} \neq \mathbf{v}_S,$$

or else $\mathbf{v} = \mathbf{0}$. It is then concluded that $\|\mathbf{v}_S\|_1 < \|\mathbf{v}_{\overline{S}}\|_1$ which establishes the null space property relative to S.

$\Leftarrow$:
Conversely, given an index set $S \subseteq [N]$ assume that the null space property relative to S holds. Given a vector $\mathbf{x} \in \mathbb{R}^N$ with $\text{supp}(\mathbf{x}) \subset S$ and a vector $\mathbf{z} \in \mathbb{R}^N$ where $\mathbf{z} \neq \mathbf{x}$, such that $\mathbf{A}\mathbf{z} = \mathbf{A}\mathbf{x}$. Consider then a vector $\mathbf{v}$ given by $\mathbf{v} := \mathbf{x} - \mathbf{z} \in \text{null}(\mathbf{A}) \setminus \{\mathbf{0}\}$. From the null space property, the following is obtained:

$$\begin{aligned} \|\mathbf{x}\|_1 \leq \|\mathbf{x} - \mathbf{z}_S\|_1 &= \|\mathbf{v}_S\|_1 + \|\mathbf{z}_S\|_1 \\ &< \|\mathbf{v}_{\overline{S}}\|_1 + \|\mathbf{z}_S\|_1 \\ &= \|-\mathbf{z}_{\overline{S}}\|_1 + \|\mathbf{z}_S\|_1 = \|\mathbf{z}\|_1. \end{aligned}$$

This establishes the required sparseness of $\|\mathbf{x}\|_1$. ■

Unfortunately, this is a condition which is hard to check in practice. Instead coherence can be used as a measure on $\mathbf{A}$ where a small coherence lead to a good choice of $\mathbf{A}$. Another condition which can be used in practice is the restricted isometry property (RIP) [**??**].

3.3 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 furthermore 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 model,

$$\mathbf{y} = \mathbf{A}\mathbf{x} \tag{3.5}$$

where $\mathbf{x} \in \mathbb{R}^N$ contain the N underlying speak signals and $\mathbf{A}$ is a mixing matrix where the coefficients depends (more or less?) on the distance from the source to the microphone. As such each y_i is a weighted sum of all the present sources of speak.

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 .

when we assume independence it is enough to solve system, why?

By independence, one means that changes in one source signal do not affect the other source signals. The theoretically definition of joint probability density function (pdf) of $\mathbf{x}$ is

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

herover er ukommenteret et afsnit jeg ikke forstår

The possibility of separating a signal into independent and non-Gaussian components originates from the central limit theorem [13, 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.

3.3.1 Assumptions and Preprocessing

For simplicity assume $\mathbf{A}$ 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 $\mathbf{x} = \mathbf{A}^{-1}\mathbf{y}$ [13, 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 assume to have unit variance as well. Any scalar multiplier within a source can be cancelled out by dividing the corresponding column in $\mathbf{A}$ with the same scalar [13, p. 154]. For further simplification it is assumed without loss of generality that $\mathbb{E}[\mathbf{y}] = 0$ and $\mathbb{E}[\mathbf{x}] = 0$[13, p. 154]. In case this assumption is not true, the measurements can be centred 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. That is multiplying the measurement vector $\mathbf{y}$ with a whitening matrix $\mathbf{V}$,

whiteining is a linear change of coordinates of the mixed data`http://arnauddelorme.com/ica_for_dummies/` "By rotating the axis and minimizing Gaussianity of the projection in the first scatter plot, ICA is able to recover the original sources which are statistically independent

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

to obtain a new measurement vector $\mathbf{y}_{white}$ that is whited. 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 consists of the associated eigenvectors. From $\mathbf{E}$ and $\mathbf{D}$ a whitening matrix is constructed [13, p.159].

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

where $\mathbf{D}^{-1/2} = \text{diag}(d_1^{-1/2}, \ldots, d_n^{-1/2})$ is a componentwise operation.
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 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 $\mathbb{E}[\mathbf{x}\mathbf{x}^T] = \mathbf{I}$ because of assumed uncorrelation and zero mean. Consequently ICA can restrict its search for the mixing matrix to the orthogonal matrix space – that is instead of estimating N^2 parameters ICA one now only has to estimate an orthogonal matrix which has $N(N-1)/2$ parameters/degrees of freedom [13, p. 159]
.

se udkommentering herunder?

3.3.2 Recovery of the Independent Components

Now the ICA model is established, the next step is the estimation of the mixing coefficients a_{ij} and independent components x_i. The simple and intuitive method is to take advantage of the assumption of non-Gaussian independent components. Consider again the ICA model of a single measurement vector $\mathbf{y} = \mathbf{A}\mathbf{x}$ where the independent components can be estimated by the inverted model $\mathbf{x} = \mathbf{A}^{-1}\mathbf{y}$. Let $\mathbf{A}^{-1} = \mathbf{B}$, now a single independent component can be seen as the linear combination

Herfra og ned til kurtosis skal lige tjekkes i gennem og rettes til

$$x_i = \mathbf{b}_{i\bullet}\mathbf{y} = \sum_k b_k y_k \tag{3.6}$$

where $\mathbf{b}_{i\bullet}$ is the i-th row of $\mathbf{B}$. The issue is now to determine $\mathbf{b}_j$ (the j-th column) such that it equals the i-th row from the inverse $\mathbf{A}$. As $\mathbf{A}$ is unknown it is not possible to determine $\mathbf{b}_j$ exactly, but an estimate can be found to make a good approximation. Rewriting (3.6)

$$x_i = \mathbf{b}_{i\bullet}\mathbf{y} = \mathbf{b}_{i\bullet}\mathbf{A}\mathbf{x} = \mathbf{q}^T\mathbf{x} = \sum_{k=1} q_k x_k$$

it is seen how x_i is a linear combination of all x_k, thus the equality only holds true when $\mathbf{q}$ consist of only one non-zero element that equals 1. Due to 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. Then, since $\mathbf{q}^T\mathbf{x} = \mathbf{b}_{i\bullet}\mathbf{y}$, it is possible to vary the coefficients in $\mathbf{b}$ and look at the distribution of $\mathbf{b}_{i\bullet}\mathbf{y}$. Finding the vector $\mathbf{b}$ that maximizes the non-Gaussianity would then correspond to $\mathbf{q} = \mathbf{A}^T\mathbf{b}$ having only a single non-zero element. Thus maximizing the non-Gaussianity of $\mathbf{b}_{i\bullet}\mathbf{y}$ results in one of the independent components [13, p. 166].
Considering the N-dimensional space of vectors $\mathbf{b}$ there exist $2N$ local maxima, corresponding to x_i and $-x_i$ for all n independent components [13, p. 166].

3.3.3 Kurtosis

To maximize the non-Gaussianity a measure for Gaussianity is needed. Kurtosis is a quantitative measure used for non-Gaussianity of random variables. 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

uddyb? og tilføj kilde til kurtosis

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

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 [13, 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 [13, p. 182].

Consider again the vector $\mathbf{q} = \mathbf{A}^T\mathbf{b}$ such that $\mathbf{b}_{i\bullet}\mathbf{y} = \sum_{k=1} q_k x_k$. By the additive property of kurtosis

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

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 the optimisation problem of maximising the kurtosis of $\mathbf{b}_{i\bullet}\mathbf{y}$ is similar to maximizing $|\text{kurt}(x_i)| = |\sum_{k=1} q_k^4 \text{kurt}(x_k)|$ on the unit sphere.

hvordan kommer dette frem?

Due to the described preprocessing $\mathbf{b}$ is assumed to be white and it can be shown that $\|\mathbf{q}\| = \|\mathbf{b}_j\|$ [13, p. 174]. This shows that constraining $\|\mathbf{q}\|$ to one is similar to constraining $\|\mathbf{b}_j\|$ to one.

3.3.4 The Gradient Algorithm with Kurtosis

In practise, to recover the mixing matrix $\mathbf{A}$ by maximizing the kurtosis of $\mathbf{b}_{i\bullet}\mathbf{y}$, gradient optimisation methods are used.
The general idea behind a gradient algorithm is to determine the direction for which $\text{kurt}(\mathbf{b}_{i\bullet}\mathbf{y})$ is growing the most, based on the gradient.
The gradient of $|\text{kurt}(\mathbf{b}_{i\bullet}\mathbf{y})|$ is computed as

$$\frac{\partial|\text{kurt}(\mathbf{b}_{i\bullet}\mathbf{y})|}{\partial \mathbf{b}_j} = 4\text{sign}(\text{kurt}(\mathbf{b}_{i\bullet}\mathbf{y}))(\mathbb{E}[\mathbf{y}(\mathbf{b}_{i\bullet}\mathbf{y})^3] - 3\mathbf{y}\mathbb{E}[(\mathbf{b}_{i\bullet}\mathbf{y})^2]) \tag{3.7}$$

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

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

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.

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

3.3.5 Basic ICA algorithm

Algorithm 1 combines the above theory, to give an overview of the ICA procedure.

Algorithm 1 Basis ICA

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

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

3.3.6 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.

henvis til appendix

3.4 Limitations of compressive sensing

Through this chapter the concept of sparse signal recovery has been explained. The essential limitation of signal recovery from an under-determined system is that $k \leq M$ is necessary in order to uniquely recover the k-sparse signal $\mathbf{X} \in \mathbb{R}^N$ from the measurements $\mathbf{Y} \in \mathbb{R}^M$. That is the number of measurements must be greater than the number of active sources within the signal to be recovered. Similarly it is not possible to recover the true dictionary $\mathbf{A}$ by dictionary learning methods if $k > M$. Because in that case any random dictionary of full rank can be used to create $\mathbf{y}$ from $\geq M$ basis vectors [6, p. 30].
When considering source recovery from EEG measurements, described in section 1.1, it is not reasonable to assume that $k < M$ and especially not in the case of low density EEG measurements. This motivates the next two chapters where the possibility of source recovery for $k > M$ is explored. The methods, proposed recently by O. Balkan, are taking advantage of the covariance domain and.

skal dette argumenteres yderligere, som værende uafhængig af motivations kapitlet?

Chapter 4

Dictionary Learning

As clarified in section 3.2.3 the estimation of the dictionary matrix $\mathbf{A}$ is essential to achieve the best recovery of the sparse signal $\mathbf{x}$ from the measurements $\mathbf{y}$. Pre-constructed dictionaries do exist which in many cases results in simple and fast algorithms for reconstruction of $\mathbf{x}$ [9]. Pre-constructed dictionaries are 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 [9]. Hence the results of using such dictionaries depend on how well they fit the data of interest, which is creating a certain limitation. An alternative 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 fixed dictionary which can take part in the application. Different dictionary learning algorithms exist. One is the K-SVD which is to be elaborated in this chapter. The K-SVD algorithm was presented in 2006 by Elad et al. and found to outperform pre-constructed dictionaries when computational cost is of secondary interest [1]. Before the K-SVD algorithm can be investigated the linear model (3.1) must be expanded to be considering sets of training data.

4.1 Multiple Measurement Vector Model

The linear model (3.1) is also referred to as a single measurement vector (SMV) model. In order to adapt the model (3.1) to a practical use the model is expanded to include multiple measurement vectors and take noise into account. A multiple measurement vector (MMV) model consists of the observed measurement matrix $\mathbf{Y} \in \mathbb{R}^{M \times L}$, the source matrix $\mathbf{X} \in \mathbb{R}^{N \times L}$, the dictionary matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ and the noise vector $\mathbf{E} \in \mathbb{R}^{M \times L}$:

$$\mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{E}. \tag{4.1}$$

L denotes the number of observed measurement vectors each consisting of M measurements, that is L samples are given. For $L = 1$ the linear model will just be the SMV model (3.1). The matrix $\mathbf{X}$ consists of k-sparse vectors $\mathbf{x}_1, \dots, \mathbf{x}_L$ which have been stacked column-wise such that $\mathbf{X}$ consist of at most k non-zero rows. As for the SMV model (3.1) the MMV model (4.1) is under-determined with $M \ll N$ and $k < M$ [8, p. 42].

The support of $\mathbf{X}$ denotes the index set of non-zero rows of $\mathbf{X}$ and $\mathbf{X}$ is said to be row-sparse. As the columns in $\mathbf{X}$ are k-sparse and as mentioned before, $\mathbf{X}$ has at most k non-zero rows, the non-zero values occur in common location for all columns. By using this joint information it is possible to recover $\mathbf{X}$ from fewer measurements [8, p. 43].

4.2 K-SVD

Consider $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_L]$, $\mathbf{y}_j \in \mathbb{R}^M$ as a training database, created by $\mathbf{y}_j = \mathbf{A}\mathbf{x}_j$ for which one want to learn the best suitable dictionary $\mathbf{A}$ and sparse representation $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_L]$, $\mathbf{x}_j \in \mathbb{R}^N$. For a known sparsity constraint k this can be defined by an optimisation problem similar to the general compressive sensing problem of multiple measurements [9]

$$\min_{\mathbf{A},\mathbf{X}} \sum_{j=1}^{L} \|\mathbf{y}_j - \mathbf{A}\mathbf{x}_j\|_2^2 \quad \text{subject to} \quad \|\mathbf{x}_j\|_0 \le k,\ 1 \le j \le L. \tag{4.2}$$

The learning consists of jointly solving the optimization problem on $\mathbf{X}$ and $\mathbf{A}$. The uniqueness of $\mathbf{A}$ depends on the recovery sparsity condition. As clarified earlier recovery is only possible if $k < M$ [6].

4.2.1 K-SVD Algorithm

The dictionary learning algorithm K-SVD provides an updating rule which is applied to each column of $\mathbf{A}_0 = [\mathbf{a}_0, \dots, \mathbf{a}_N]$ where $\mathbf{A}_0$ being a random initial dictionary matrix. Updating first $\mathbf{a}_j$ and then the corresponding coefficients in $\mathbf{X}$ which it is multiplied with the i-th row in $\mathbf{X}$ denoted by $\mathbf{x}_{i\bullet}$.
Let $\mathbf{a}_{j_0}$ be the column to be updated and let the remaining columns be fixed. By rewriting the objective function in (4.2) using matrix notation it is possible to isolate the contribution from $\mathbf{a}_{j_0}$.

Tjek nedenstående udledning. a og x er ikke lige lange da a_j er M lang mens $x_{i\bullet}$ er L lang

$$\begin{aligned} \|\mathbf{Y} - \mathbf{A}\mathbf{X}\|_F^2 &= \left\| \mathbf{Y} - \sum_{j=1}^{N} \mathbf{a}_j \mathbf{x}_{i\bullet} \right\|_F^2 \\ &= \left\| \left(\mathbf{Y} - \sum_{j \neq j_0}^{N} \mathbf{a}_j \mathbf{x}_{i\bullet} \right) - \mathbf{a}_{j_0} \mathbf{x}_{i_0\bullet} \right\|_F^2, \end{aligned} \tag{4.3}$$

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 (4.3) the term in the parenthesis is denoted by $\mathbf{E}_{j_0}$, an error matrix, and hence by minimising (4.3) with respect to $\mathbf{a}_{j_0}$ and $\mathbf{x}_{i_0\bullet}$ leads to the optimal contribution from j_0

$$\min_{\mathbf{a}_{j_0},\mathbf{x}_{i_0\bullet}} \| \mathbf{E}_{j_0} - \mathbf{a}_{j_0}\mathbf{x}_{i_0\bullet}\|_F^2 . \tag{4.4}$$

The optimal solution to (4.4) is known to be the rank-1 approximation of $\mathbf{E}_{j_0}$. This comes from the Eckart–Young–Mirsky theorem [**?**] saying that 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

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

with $\mathbf{U} \in \mathbb{R}^{M\times M}$ and $\mathbf{V} \in \mathbb{R}^{N\times N}$ being unitary matrices[1] and $\mathbf{\Sigma} = \text{diag}[\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_M] \in \mathbb{R}^{M\times N}$ 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 $k < \text{rank}(\mathbf{E}_{j_0})$ is then given by : kilde

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

Since the outer product always have rank-1 letting $\mathbf{a}_{j_0} = \mathbf{u}_1$ and $\mathbf{x}_{i_0\bullet} = \boldsymbol{\sigma}_j \mathbf{v}_1^T$ solves the optimisation problem (4.4). However in order to preserve the sparsity in $\mathbf{X}$ while optimising, only the non-zero entries in $\mathbf{x}_{i_0\bullet}$ 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 $\mathbf{x}_{i_0\bullet}$. A matrix $\mathbf{P}_{i_0}$ is defined to restrict $\mathbf{x}_{i_0\bullet}$ to only contain the non-zero-rows corresponding to N_{j_0} non-zero rows:

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

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 $\mathbf{x}_{i_0\bullet}^{(R)}$ the rank-1 approximation is found and the original representation vector is updated as $\mathbf{x}_{i_0\bullet} = \mathbf{x}_{i_0\bullet}^{(R)}\mathbf{P}_{i_0}^T$.

The main steps of K-SVD is described in algorithm 2.

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

Algorithm 2 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 $\mathbf{P}_{i_0}$
      $\mathbf{E}_{j_0} = \mathbf{Y} - \sum_{j \neq j_0}^{N} \mathbf{a}_j \mathbf{x}_{i\bullet}$
      $\mathbf{E}_{j_0}^{(R)} = \mathbf{E}_{j_0}\mathbf{P}_{i_0}$
      $\mathbf{E}_{j_0}^{(R)} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$ ▷ perform SVD
      $\mathbf{a}_{j_0} \leftarrow \mathbf{u}_1$ ▷ update the j_0 col. in $\mathbf{A}_{(k)}$
      $(\mathbf{x}_{i_0\bullet})^{(R)} \leftarrow \boldsymbol{\sigma}_1 \mathbf{v}_1$
      $\mathbf{x}_{i_0\bullet} \leftarrow (\mathbf{x}_{i_0\bullet})^{(R)} \mathbf{P}_{i_0}^T$ ▷ 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

The dictionary learning algorithm K-SVD is a generalisation of the well known K-means clustering also referred to as vector quantization. In K-means clustering a set of K vectors is learned referred to as mean vectors. Each signal sample is then represented by its nearest mean vector. That corresponds to the case with sparsity constraint $k = 1$ and the representation reduced to a binary scalar $x = \{1, 0\}$. Further instead of computing the mean of K subsets the K-SVD algorithm computes the SVD factorisation of the K different sub-matrices that correspond to the K columns of $\mathbf{A}$.

Chapter 5

Covariance-Domain Dictionary Learning

The section is inspired by chapter 3 in [6] and the article [4]. INTRO..

5.1 Introduction

Covariance-domain dictionary learning (Cov-DL) is an algorithm proposed by O. Balkan [4], claiming to successfully identify more active sources k than available measurements M from the multiple measurement vector model

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

where $\mathbf{Y} \in \mathbb{R}^{M \times L}$ is the observed measurement matrix, $\mathbf{X} \in \mathbb{R}^{N \times L}$ the the source matrix and $\mathbf{E} \in \mathbb{R}^{M \times L}$ is the additional noise matrix .
Let f be the sample frequency of the observed data $\mathbf{Y}$ and let s denoted a segment index. As such the observed data can be divided into segments $\mathbf{Y}_s \in \mathbb{R}^{M \times t_s f}$, possibly overlapping, where t_s is the length of the segments in seconds. For each segment the linear model still holds and is rewritten into

f er samples pr sek., L er antal sampels i alt, Ls er antal samples pr segment og ts er længen pr segment i sekunder

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

Cov-DL takes advantage of the covariance domain where the dimensionality is increased allowing for an enlarged number of sources to be active while the dictionary remains recoverable. An important 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. From the assumption of uncorrelated sources it can be assumed that the sample covariance of $\mathbf{X}_s$ becomes nearly diagonal. This is of importance when the system is transformed to the covariance domain.

The Cov-DL do only recover the mixing matrix $\mathbf{A}$ given the measurements $\mathbf{Y}$. Given $\mathbf{A}$ the source matrix $\mathbf{X}$ is to be recovered by use of the Multiple Sparse Bayesian Learning algorithm, this is described in section 6

5.2 Covariances domain representation

Consider the covariance of a vector $\mathbf{x}_i$

$$\mathbf{\Sigma} = \mathbb{E}[(\mathbf{x}_i - \mathbb{E}[\mathbf{x}_i])(\mathbf{x}_i - \mathbb{E}[\mathbf{x}_i])].$$

Assume that all samples has zero mean and the same distribution within one segment. The observed measurements $\mathbf{Y}_s \in \mathbb{R}^{M \times L}$ can be described in the covariance domain by the sample covariance $\hat{\mathbf{\Sigma}}$ which is defined as the covariance among the M measurements across the L_s samples. That is a $M \times M$ matrix $\mathbf{\Sigma}_{\mathbf{Y}_s} = [\sigma_{jk}]$ with entries

$$\sigma_{jk} = \frac{1}{L} \sum_{i=1}^{L} y_{ji} y_{ki}^T.$$

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

$$\hat{\mathbf{\Sigma}}_{\mathbf{Y}_s} = \frac{1}{L} \mathbf{Y}_s \mathbf{Y}_s^T.$$

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

$$\hat{\mathbf{\Sigma}}_{\mathbf{X}_s} = \frac{1}{L} \mathbf{X}_s \mathbf{X}_s^T = \mathbf{\Lambda}_s + \varepsilon$$

From the assumption of uncorrelated sources within $\mathbf{X}_s$ the sample covariance matrix is expected to be nearly diagonal, thus it can be written as $\mathbf{\Delta}_s + \varepsilon$ where $\mathbf{\Delta}_s$ is a diagonal matrix consisting of the diagonal entries of $\hat{\mathbf{\Sigma}}_{\mathbf{X}_s}$ and ε is the estimation error[4].
Each segment is then modelled in the covariance domain as

$$\begin{aligned}
\hat{\mathbf{\Sigma}}_{\mathbf{Y}_s} = \frac{1}{L_s} \mathbf{Y}_s \mathbf{Y}_s^T &= \frac{1}{L_s} (\mathbf{A}\mathbf{X}_s + \mathbf{E}_s)(\mathbf{A}\mathbf{X}_s + \mathbf{E}_s)^T \\
\mathbf{Y}_s \mathbf{Y}_s^T &= (\mathbf{A}\mathbf{X}_s)(\mathbf{A}\mathbf{X}_s)^T + \mathbf{E}_s \mathbf{E}_s^T + \mathbf{E}_s (\mathbf{A}\mathbf{X}_s)^T + \mathbf{A}\mathbf{X}_s \mathbf{E}_s^T \\
&= \mathbf{A}\mathbf{X}_s \mathbf{X}_s^T \mathbf{A}^T + \mathbf{E}_s \mathbf{E}_s^T + \mathbf{E}_s \mathbf{X}_s^T \mathbf{A}^T + \mathbf{A}\mathbf{X}_s \mathbf{E}_s^T \\
&= \mathbf{A}(\mathbf{\Delta}_s + \varepsilon)\mathbf{A}^T + \mathbf{E}_s \mathbf{E}_s^T + \mathbf{E}_s \mathbf{X}_s^T \mathbf{A}^T + \mathbf{A}\mathbf{X}_s \mathbf{E}_s^T \\
&= \mathbf{A}\mathbf{\Delta}_s \mathbf{A}^T + \mathbf{A}\varepsilon\mathbf{A}^T + \mathbf{E}_s \mathbf{E}_s^T + \mathbf{E}_s \mathbf{X}_s^T \mathbf{A}^T + \mathbf{A}\mathbf{X}_s \mathbf{E}_s^T \qquad (5.1) \\
&= \mathbf{A}\mathbf{\Delta}_s \mathbf{A}^T + \widetilde{\mathbf{E}} \qquad (5.2) \\
& \qquad (5.3)
\end{aligned}$$

igere argumenta-
lvendig her?

From (5.1) to (5.2) all terms where noise is included are defined as a united noise term $\widetilde{\mathbf{E}}$. . By vector notation (5.2) is rewritten and then vectorised. Because the covariance matrix $\hat{\mathbf{\Sigma}}_{\mathbf{Y}_s}$ is symmetric it is sufficient to vectorize only the lower triangular parts, including the diagonal. For this the function $\text{vec}(\cdot)$ is defined to map a symmetric $M \times M$ matrix into a vector of size $\frac{M(M+1)}{2}$ making a row-wise vectorization of its upper triangular part. Furthermore, let $\text{vec}^{-1}(\cdot)$ be the inverse function for de-vectorisation. This results in the following model

$$\begin{aligned}
\mathbf{\Sigma}_{\mathbf{Y}_s} &= \sum_{i=1}^{N} \mathbf{\Delta}_{s_{ii}} \mathbf{a}_i \mathbf{a}_i^T + \widetilde{\mathbf{E}} \\
\text{vec}(\mathbf{\Sigma}_{\mathbf{Y}_s}) &= \sum_{i=1}^{N} \mathbf{\Lambda}_{s_{ii}} \text{vec}(\mathbf{a}_i \mathbf{a}_i^T) + \text{vec}(\widetilde{\mathbf{E}}) \\
&= \sum_{i=1}^{N} \mathbf{d}_i \mathbf{\Lambda}_{s_{ii}} + \text{vec}(\widetilde{\mathbf{E}}) \\
&= \mathbf{D}\boldsymbol{\delta}_s + \text{vec}(\widetilde{\mathbf{E}}), \quad \forall s.
\end{aligned} \tag{5.4}$$

Here $\boldsymbol{\delta}_s \in \mathbb{R}^N$ contains the diagonal entries of the source sample-covariance matrix $\mathbf{\Lambda}_s$ and the matrix $\mathbf{D} \in \mathbb{R}^{M(M+1)/2 \times N}$ consists of the columns $\mathbf{d}_i = \text{vec}(\mathbf{a}_i \mathbf{a}_i^T)$. Note that $\mathbf{D}$ and $\boldsymbol{\delta}_s$ are unknown while $\text{vec}(\mathbf{\Sigma}_{\mathbf{Y}_s})$ is known from the observed data. By this transformation to the covariance domain one segments is now represented as the single measurement model with $M(M+1)/2$ "measurements". It has been shown that this model allow for identification of $k \leq M(M+1)/2$ active sources [16], which 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 model to find the dictionary $\mathbf{A}$ from $\mathbf{D}$ and then still allow for $k \leq M(M+1)/2$ active sources to be identified. That is the number of active sources are allowed to exceed the number of observations as intended.

5.3 Determination of the Dictionary

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

Under-determined D

In the case of $N > \frac{M(M+1)}{2}$ $\mathbf{D}$ becomes under-determined. This is similar to the original system being under-determined when $N > M$. Thus, it is again possible to solve the under-determined system if certain sparsity is withhold. Namely $\boldsymbol{\delta}_s$ being $\frac{M(M+1)}{2}$-sparse. Assuming the sufficient sparsity on $\boldsymbol{\delta}_s$ is withhold it is possible to learn the dictionary matrix of the covariance domain $\mathbf{D}$ by traditional dictionary learning methods applied to the observations represented in the covariance domain

$\text{vec}(\mathbf{\Sigma}_{\mathbf{Y}_s})$ for all s. For this K-SVD algorithm, described in section 4.2 is used. Note here that the number of samples that are used to learn the dictionary is remarkable reduces as one segment effectively corresponds to one sample in the covariance domain.
When $\mathbf{D}$ is learned it is possible to find the original mixing matrix $\mathbf{A}$ that generated $\mathbf{D}$ through the relation $\mathbf{d}_j = \text{vec}(\mathbf{a}_j\mathbf{a}_j^T)$. Here each column is found by the optimisation problem

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

for which the global minimizer is $\mathbf{a}_j^* = \sqrt{\lambda_j}\mathbf{b}_j$. Here λ_j is the largest eigenvalue of $\text{vec}^{-1}(\mathbf{d}_j)$,

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

redegørelse for resultatet her skal laves

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

Over-determined D

In the case of $N < \frac{M(M+1)}{2}$ an over-determined system is achieved and it is not possible to find $\mathbf{D}$ by dictionary learning methods.

how else can they live on the same space??

By assuming that $\frac{M(M+1)}{2}$ will be close to N, because $N > M$ is given, then the measurements in the covariance domain $\text{vec}(\mathbf{\Sigma}_{\mathbf{Y}_s})$ will live on or near a subspace of dimension N. This subspace is spanned by the columns of $\mathbf{D}$, and is denoted as $(\mathbf{D})$. To learn $\mathcal{R}(\mathbf{D})$ without having to impose any sparsity constraint on $\boldsymbol{\delta}_s$ it is possible

evt. teoretisk beskrivelse af PCA i appendix?

to use Principal Component Analysis(PCA). By use of PCA a set of basis vectors $\mathbf{U}$ is achieved such that $\mathcal{R}(\mathbf{U}) = \mathcal{R}(\mathbf{D})$. This however do not imply that $\mathbf{D} = \mathbf{U}$.
In the case of two sets of basis vectors span the same space, namely $\mathcal{R}(\mathbf{U}) = \mathcal{R}(\mathbf{D})$, the projection operator of the given subset must be unique. Which is true if and

kilde foruden phd p. 51?

only if $\mathbf{D}(\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T = \mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T$. Remember from the above derivation the condition that $\mathbf{d}_i = \text{vec}(\mathbf{a}_i\mathbf{a}_i^T)$. From this it is possible to obtain $\mathbf{A}$ through the optimisation problem

$$\begin{aligned} \min_{\mathbf{a}_i} \|\mathbf{D}(\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T - \mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\|_F^2 \\ \text{s.t. } \mathbf{d}_i = \text{vec}(\mathbf{a}_i\mathbf{a}_i^T) \end{aligned} \tag{5.5}$$

where $\mathbf{U}$ is learned by use of PCA performed on $\text{vec}(\mathbf{\Sigma}_{\mathbf{Y}_s})$.
To solve this optimization problem the cost function is minimized by use of quasi-Newton optimization methods. Several specific quasi-Newton methods exist but the

basic principal will be presented here. The Newton optimization method is a multidimensional gradient method. The method is based on a quadratic approximation of the optimization problem by use of the Taylor series, which is elaborated in [3, p. 29]. Let $f(\mathbf{x})$ be the cost function and $\boldsymbol{\delta}$ be the change in $\mathbf{X}$. By differentiating the Taylor approximation of $f(\mathbf{x}+\boldsymbol{\delta})$ and setting it equal to zero, the optimal change in $\mathbf{x}$ is found to be $\boldsymbol{\delta} = -\mathbf{H}^{-1}\mathbf{g}$. Where $\mathbf{g}$ is the gradient and $\mathbf{H}$ is the Hessian. The quasi-Newton methods deviate from the basic Newton method by letting the direction search be based on a positive semi-definite matrix $\mathbf{S}$ which is generated from available data in order to approximate $\mathbf{H}^{-1}$. Details of the method is found in [3, p. 175]

5.3.1 Pseudo Code of the Cov-DL Algorithm

Algorithm 3 Cov-DL

procedure COV-DL($\mathbf{Y}_s$)
  for $s \leftarrow 1, \cdots,$ n_seg **do**
    compute sample covariance matrix $\widehat{\mathbf{\Sigma}}_{\mathbf{Y}_s}$
    $\mathbf{y}_{\text{cov}_s} = \text{vec}(\widehat{\mathbf{\Sigma}}_{\mathbf{Y}_s})$
  end for
  $\mathbf{Y}_{\text{cov}} = \{\mathbf{y}_{\text{cov}_s}\}_{s=1}^{\text{n\_seg}}$
  if $N > \frac{M(M+1)}{2}$ **then**
    procedure K-SVD($\mathbf{Y}_{\text{cov}}$)
      returns $\mathbf{D} \in \mathbb{R}^{M(M+1)/2} \times N$
    end procedure
    for $j \leftarrow 1, \cdots, N$ **do**
      $\mathbf{T} = \text{vec}^{-1}(d_j)$
      $\lambda_j \leftarrow \max\{eigenvalue(\mathbf{T})\}$
      $\mathbf{b}_j \leftarrow eigenvector(\lambda_j)$
      $\mathbf{a}_j \leftarrow \sqrt{\lambda_j}\mathbf{b}_j$
    end for
    $\mathbf{A} = \{\mathbf{a}_j\}_{j=1}^{N}$
  end if

  if $N < \frac{M(M+1)}{2}$ **then**
    procedure PCA(vec($\mathbf{\Sigma}_{\mathbf{Y}_s}$))
      returns $\mathbf{U} \in \mathbb{R}^{M(M+1)/2 \times N}$
    end procedure
    procedure QUASI-NEWTON(problem (5.5))
      returns $\mathbf{A} = \{\mathbf{a}_j\}_{j=1}^{N}$
    end procedure
  end if
end procedure

5.3.2 Remarks to be considered and investigated in he code

- the number of samples for dictionary learning is reduced remarkably when Cov-DL is used, this is mentioned above. increasing overlap will improved this.
- the effect of the length of segments L_s. it will effect the diagonality of $cov(X)$
- the values of the individual sources must not be constant over time, that is the power of the sourced, it will not be at problem for EEG data.

- for Cov-DL2 the solution tends to be unique when $M < N < M(M+1)/2$, that is the found cost function may have a local minima. for this it is recommended to use several random initial point.

Chapter 6

Multiple Sparse Bayesian Learning

....
INTRODUCTION
....

Consider the multiple measurement vector model (MMV)

$$\mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{E},$$

where the unknown matrices $\mathbf{A} \in \mathbb{R}^{M \times N}$ and $\mathbf{X} \in \mathbb{R}^{N \times L}$ are wished recovered from the known measurement matrix $\mathbf{Y} \in \mathbb{R}^{M \times L}$ in the case of $M < N$. In chapter 5 the mixing matrix $\mathbf{A}$ was found. In this chapter, a method to recover the source matrix $\mathbf{X}$ is sought. The approach is to find the support set S of $\mathbf{X}$ providing the non-zeros rows of $\mathbf{X}$ which corresponds to localisation of the active sources. By ... the active sources are identified in order to fully recover $\mathbf{X}$.
The chapter is inspired by [23] and the articles [25], [5].

6.1 Maximum a Posterior Estimation

With the knowledge of $\mathbf{A}$ and $\mathbf{Y}$ it is possible to find maximum likelihood estimate of $\mathbf{X}$. By maximising the likelihood $p(\mathbf{Y}|\mathbf{X})$ an estimate of $\mathbf{X}$ can be achieved in the case of more sensors than sources, $M > N$. But in the desired case where $M < N$ the estimation becomes complicate as the MMV model becomes under-determined and potentially an infinitely number of solutions exist with equal likelihoods.
As the optimisation problem of the MMV model is NP-hard another estimation method must be used.

Bayesian probability ...

indsæt her hvad tanken bag Bayesian egenligt er

Within this Bayesian framework the source matrix $\mathbf{X}$ is seen as a variable which is drawn from some distribution $p(\mathbf{X})$ such that it is possible to narrow down the infinitely solution space. Assuming a prior belief that $\mathbf{Y}$ is generated from a sparse coefficient matrix, that is the distribution from where $\mathbf{X}$ is drawn has a sharp, possibly infinite, spike at zero surrounded by fat tails.

By applying an exponential function $\exp(-(\cdot))$ transformation onto our optimisation problem a Gaussian likelihood function $p(\mathbf{Y}|\mathbf{X})$ with a λ-dependent variance is achieved:

$$p(\mathbf{Y}|\mathbf{X}) \propto \exp\left(-\frac{1}{\gamma}\|\mathbf{Y}-\mathbf{A}\mathbf{X}\|_2^2\right),$$

with a prior distribution $p(\mathbf{X}) \propto \exp(-\|\mathbf{X}\|_0)$ [23, p. 137]. The optimisation problem is then rewritten by Bayers formula

$$\begin{aligned}\hat{\mathbf{X}} &= \arg\max_{\mathbf{X}} p(\mathbf{Y}|\mathbf{X})p(\mathbf{X}) \\ &= \arg\max_{\mathbf{X}} \frac{p(\mathbf{Y}|\mathbf{X})p(\mathbf{X})}{p(\mathbf{Y})} \quad \text{(Bayers Formular)} \\ &= \arg\max_{\mathbf{X}} p(\mathbf{X}|\mathbf{Y}).\end{aligned}$$

From this it it possible to view the optimization problem as a MAP estimation challenge.

6.2 Empirical Bayesian Estimation

Different MAP estimation approaches exists separated by the choice of sparsity inducing prior and optimization method. Some problems have shown to occur when using a fixed and algorithm-dependent prior as the posterior is not sparse enough if a prior is not as sparse leading to a 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 sparse prior can be avoided. ARD is a method where a prior is introduced to determine the relevance of a parameter. This prior is modulated by a vector of hyperparameters affecting the prior variance of each row in X. This can also be view as a regularisation of the solution space which is narrowed to consist only of relevant information [24].
An empirical prior can be used with ARD as the empirical prior is flexible and depends on the unknown hyperparameter $\boldsymbol{\gamma}$ and therefore more data-dependent – such prior can be controlled to induce sparsity.

Let the likelihood $p(\mathbf{Y}|\mathbf{X})$ be Gaussian, with known noise variance σ^2. Then for each column in $\mathbf{Y}$ and $\mathbf{X}$ the likelihood is written as

$$\begin{aligned} p(\mathbf{y}_j|\mathbf{x}_j) &= \mathcal{N}(\mathbf{A}\mathbf{x}_j, \sigma^2\mathbf{I}) \\ &= (2\pi\sigma^2)^{-N/2} \exp\left(-\frac{1}{2\sigma^2}\|\mathbf{y}_j - \mathbf{A}\mathbf{x}_j\|_2^2\right). \end{aligned}$$

With the use of ARD the i-th row of the sources matrix $\mathbf{X}$, $\mathbf{x}_{i\bullet}$, is assigned an L-dimensional independent Gaussian prior with zero mean and a variance controlled by γ_i which is unknown:

$$p(\mathbf{x}_{i\bullet}; \gamma_i) = \mathcal{N}(0, \gamma_i\mathbf{I}).$$

By combing the row priors

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

a full prior of $\mathbf{X}$ is achieved modulated by the hyperparameter vector $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_M]^T$. By combining the full prior and the likelihood $p(\mathbf{y}_j|\mathbf{x}_j)$ the posterior of the j-th column of the source matrix $\mathbf{X}$ becomes

$$p(\mathbf{x}_j|\mathbf{y}_j; \boldsymbol{\gamma}) = \frac{p(\mathbf{x}_j, \mathbf{y}_j; \boldsymbol{\gamma})}{\int p(\mathbf{x}_j, \mathbf{y}_j; \boldsymbol{\gamma})\, d\mathbf{x}_j} = \mathcal{N}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}),$$

with mean and covariance given as

$$\begin{aligned} \boldsymbol{\Sigma} &= \mathrm{Cov}(\mathbf{x}_j|\mathbf{y}_j; \boldsymbol{\gamma}) = \boldsymbol{\Gamma} - \boldsymbol{\Gamma}\mathbf{A}^T\boldsymbol{\Sigma}_y^{-1}\mathbf{A}\boldsymbol{\Gamma}, \quad \forall j = 1, \ldots, L \\ \mathcal{M} &= [\boldsymbol{\mu}_{1\bullet}, \ldots, \boldsymbol{\mu}_{L\bullet}] = \mathbb{E}[\mathbf{X}|\mathbf{Y}; \boldsymbol{\gamma}] = \boldsymbol{\Gamma}\mathbf{A}^T\boldsymbol{\Sigma}_y^{-1}\mathbf{Y}, \end{aligned} \tag{6.1}$$

where $\boldsymbol{\Gamma} = \mathrm{diag}(\boldsymbol{\gamma})$ and $\boldsymbol{\Sigma}_y = \sigma^2\mathbf{I} + \mathbf{A}\boldsymbol{\Gamma}\mathbf{A}^T$. Let now the posterior mean serve as the point estimate for $\mathbf{X}$ without involving the support set S.
It is clear that row sparsity is achieved whenever $\gamma_i = 0$. From this the posterior must satisfy the following

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

which ensure that the posterior mean $\mathcal{M}$ of the i-th row, $\boldsymbol{\mu}_{i\bullet}$, will be zero. Now, instead of estimating the support set?/sparsity profile of our source matrix $\mathbf{X}$ it is sufficient to estimate the hyperparameters γ_i[23, p. 147].
Each different hyperparameter $\boldsymbol{\gamma}$ correspond to different hypothesis for the prior distribution of the underlying generation of $\mathbf{Y}$. Therefore the determination of $\boldsymbol{\gamma}$ is seen as a model selection for which an empirical Bayesian strategy can be used. By the empirical Bayesian strategy the unknown weights, making the source matrix $\mathbf{X}$, are treated as nuisance parameters and are integrated out.

By integrating the likelihood of $\mathbf{Y}$ with respect to the unknown sources $\mathbf{X}$ the marginal likelihood of the observed mixed data $\mathbf{Y}$, $p(\mathbf{Y};\boldsymbol{\gamma})$ is achieved [23, p. 146]. By applying the $-2\log(\cdot)$ transformation the marginal likelihood function is transformed to a cost function

$$\begin{aligned}\mathcal{L}(\boldsymbol{\gamma}) &= -2\log\left(\int p(\mathbf{Y}|\mathbf{X})p(\mathbf{X};\boldsymbol{\gamma})\, d\mathbf{X}\right)\\ &= -2\log(p(\mathbf{Y};\boldsymbol{\gamma}))\\ &= \log(|\boldsymbol{\Sigma}_y|) + \frac{1}{L}\sum_{j=1}^{L}\mathbf{y}_j^T\boldsymbol{\Sigma}_y^{-1}\mathbf{y}_j\end{aligned}$$

To minimise the marginal log likelihood $\mathcal{L}(\boldsymbol{\gamma})$ with respect to $\boldsymbol{\gamma}$ the evidence maximisation (EM) algorithm can be used. The E-step of the EM algorithm is to compute the posterior moments using (6.1) while the M-step is the following update rule of γ_i:

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

The M-step is very slow on large data. Instead one could use a fixed point update to fasten the convergence on large data, however convergence is no longer ensured. The fixed point updating step is achieved by taking the derivative of the marginal log likelihood $\mathcal{L}(\boldsymbol{\gamma})$ with respect to $\boldsymbol{\gamma}$ and equating it with zero. This lead to the following update equation which can replace the above M-step in the EM-algorithm:

$$\gamma_i^{(k+1)} = \frac{\frac{1}{L}\|\boldsymbol{\mu}_{i\bullet}\|_2^2}{1-\gamma_i^{-1(k)}\Sigma_{ii}}, \quad \forall i = 1,\ldots,M.$$

Empirically this alternative update rule have shown use full in highly under-determined large scale cases by driving many hyper parameters 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 $\mathbf{X}$ can be extracted,

$$S = \{i|\hat{\gamma}_i \neq 0\},$$

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

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

Algorithm 4 M-SBL

1. Given $\mathbf{Y}$ and a dictionary matrix $\mathbf{A}$.
2. Initialise $\boldsymbol{\gamma}$, e.g $\boldsymbol{\gamma} = \mathbf{1}$.
3. Compute the posterior moments $\boldsymbol{\Sigma}$ and $\mathcal{M}$.
4. Update $\boldsymbol{\gamma}$ using EM or fixed-point.
5. Repeat step 3 and 4 until convergence to a fixed point $\boldsymbol{\gamma}^*$.
6. Update the posterior moments $\boldsymbol{\Sigma}^*$ and $\mathcal{M}^*$ with γ^*.
7. Extract support set S
8. Set source matrix estimate $\hat{\mathbf{X}}^* = \mathcal{M}^*_S$

Notes:

- Bayesian replace the troublesome prior with a distribution that, while still encouraging sparsity, is somehow more computationally convenient. Bayesian approaches to the sparse approximation problem that follow this route have typically been divided into two categories: (i) maximum a posteriori (MAP) estimation using a fixed, computationally tractable family of priors and, (ii) empirical Bayesian approaches that employ a flexible, parameterized prior that is 'learned' from the data

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, Emina 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, Ozgur, Kreutz-Delgado, Kenneth, and Makeig, Scott. "Covariance-Domain Dictionary Learning for Overcomplete EEG Source Identification". In: *ArXiv* (2015).

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

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

[7] Bech Christensen, Christian 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).

[8] C. Eldar, Yonina and Kutyniok, Gitta. *Compressed Sensing: Theory and Application*. Cambridge University Presse, New York, 2012.

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

[10] Foucart, Simon and Rauhut, Hoger. *A Mathematical Introduction to Compressive Sensing*. Springer Science+Business Media New York, 2013.

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

[12] Friston, Karl J. "Functional integration and inference in the brain". In: *Progress in Neurobiology 590* 1-31 (2002).

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

[14] Makeig, Scott et al. "Blind separation of auditory event-related brain responses into independent components". In: *Proc. Natl. Acad. Sci. USA* 94 (1997).

[15] Makeig, Scott et al. "Independent Component Analysis of Electroencephalographic Data". In: *Advances in neural information processing systems* 8 (1996).

[16] Pal, Piya 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).

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

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

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

[20] *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.

[21] Teplan, M. "Fundamentals of EEG". In: *Measurement science review* 2 (2002).

[22] V. Le, Quoc et al. "ICA with Reconstruction Cost for Efficient Overcomplete Feature Learning". In: *NIPS'11 International Conference on Neural Information Processing Systems P. 1017-1025* (2011).

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

[24] Wipf, D. P. and Nagarajan, S. S. "A New View of Automatic Relevance Determination". In: *Advances in Neural Information Processing Systems 20, MIT Press* (2008).

[25] 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

Extended ICA Algorithms

This appendix provide an extension to the basic algorithm for ICA regarding the measure of non-Gaussianity and the computation method. This extended algorithm is referred to as fast ICA and is more commonly used for source separation. This is the algorithm used to apply ICA on EEG measurements for comparison within the thesis.

A.1 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 practise can disable convergence. A fixed-point iteration algorithm to maximise the non-Gaussianity is an alternative that could be used.
Consider the gradient step derived in section 3.3.4. 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 $\mathbf{b}_j$, hence be equal to $\mathbf{b}_j$. In this case adding the gradient to $\mathbf{b}_j$ does not change the direction and convergence is achieved.

wiki: The fixed point is stable if the absolute value of the derivative of $\mathbf{w}$ at the point is strictly less than 1?

Letting the gradient given in (3.7) be equal to $\mathbf{w}$ and considering the same simplifications again suggests the new update step as [13, p. 179]

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

After the fixed point iteration $\mathbf{b}_j$ is again divided by its norm to withhold the constraint $\|\mathbf{b}_j\| = 1$. Instead of γ the fixed-point algorithm compute $\mathbf{b}_j$ directly from previous $\mathbf{b}_j$.
The fixed-point algorithm is referred to as FastICA. The algorithm has shown to converge fast and reliably, then the current and previous $\mathbf{w}$ laid in the same direction [13, p. 179].

A.1.1 Negentropy

An alternative measure of non-Gaussianity is the negentropy, which is 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 the highest entropy among the random variables of the same variance [13, 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

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

with $\mathbf{y}_{\text{gaus}}$ being a Gaussian random variable of the same covariance and correlation as $\mathbf{y}$ [13, p. 182].

As the kurtosis is sensitive for outliers the negentropy is instead difficult to compute computationally as the negentropy require a 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 comulants including the kurtosis. The following approximation is stated without further elaboration, the derivation can be found in [13, p. 182].

A.1.2 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 [13, p. 188]. Algorithm 5 show Fast ICA using negentropy, this is the algorithm which is implemented for comparison with the source separation methods which are tested in this thesis.

Algorithm 5 Fast ICA – with negentropy

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

procedure FASTICA($\mathbf{y}$)
    $k = 0$
    Initialise random vector $\mathbf{b}_{j(k)}$ ▷ unit norm
    for $j \leftarrow 1, 2, \cdots, N$ **do**
        while convergance critia not meet **do**
            $k = k + 1$
            $\mathbf{b}_{j(k)} \leftarrow \mathbb{E}[\mathbf{y}(\mathbf{b}_j^T\mathbf{y})] - \mathbb{E}[g'(\mathbf{b}_j^T\mathbf{y})]\mathbf{b}_j$ ▷ g defined in [13, p. 190]
            $\mathbf{b}_{j(k)} \leftarrow \mathbf{b}_j / \|\mathbf{b}_j\|$
        end while
        $x_j = \mathbf{b}_j^T\mathbf{y}$
    end for
end procedure

A.2 OMP

$\mathbf{A}^*$ is the adjoint of a matrix $\mathbf{A}$.

Algorithm 6 Orthogonal Matching Pursuit (OMP)

$k = 0$
Initialize $S_{(0)} = \varnothing$
Initialize $x_{(0)} = \mathbf{0}$
procedure OMP($\mathbf{A}, \mathbf{y}$)
    while stopping criteria not meet **do**
        $k = k + 1$
        $j_{(k)} = \arg\max_{j \in [N]} \{|(\mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{x}_{(k-1)}))_j|\}$
        $S_{(k)} = S_{(k-1)} \cup \{j_{(k)}\}$
        $\mathbf{x}_{(k)} = \arg\min_{\mathbf{z} \in \mathbb{C}^N} \{\|\mathbf{y} - \mathbf{A}\mathbf{z}\|_2 \mid \mathrm{supp}(\mathbf{z}) \subset S_{(k)}\}$
    end while
    $\mathbf{x}^* = \mathbf{x}_{(k)}$
end procedure

Appendix B

Cases

B.1 Toy Test Example, M-SDL

General Setup

Consider the linear multiple measurement vector model

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

with $\mathbf{Y} \in \mathbb{R}^{m \times L}$ being a known measurement matrix, $\mathbf{A} \in \mathbb{R}^{m \times n}$ a known mixing matrix and $\mathbf{X} \in \mathbb{R}^{n \times L}$ being the source matrix we wish to recover in this case.

For this toy example consider a signal $\mathbf{X}$ that is constructed as a merge of k independent signals. As such one column of X is one sample containing k active signals(sources) and n-k zero entries.
A random mixing matrix $\mathbf{A}$ is generated as a random Normal distributed hence it has normalised columns:
`A = np.random.randn(m, n)`
The measurements $\mathbf{Y}$ is generated as the product of $\mathbf{A}$ and $\mathbf{X}$:
`Y = np.dot(A, X)`
The error between all the elements of the true $\mathbf{X}$ and the recovered $\hat{\mathbf{X}}$ by using the mean square error (MSE):

$$\text{MSE} = \frac{1}{L}\sum_{i=1}^{L}(\mathbf{X} - \hat{\mathbf{X}}_i)^2$$

Consider now $\mathbf{Y}$ and $\mathbf{A}$ known then by use of the M-SDL algorithm $\mathbf{X}$ is sought recovered as $hat\mathbf{X}$. The true $\mathbf{X}$ is then to be used for comparison.

Case 1 - $k > m$

The following variables are used:

- $m = 3$ (number of sensors)
- $n = 8$ (number of sources)
- $L = 100$ (number of samples)
- No segmentations – $\mathbf{Y} = \mathbf{AX}$
- Iterations = 1000
- $k = 4$ is active sources (row-wise)

Results

The MSE of case 1 was found to be 0.141 when rounded to the nearest three decimal. For visual comparison each active source are plotted against the reconstructed source in figure **??**.

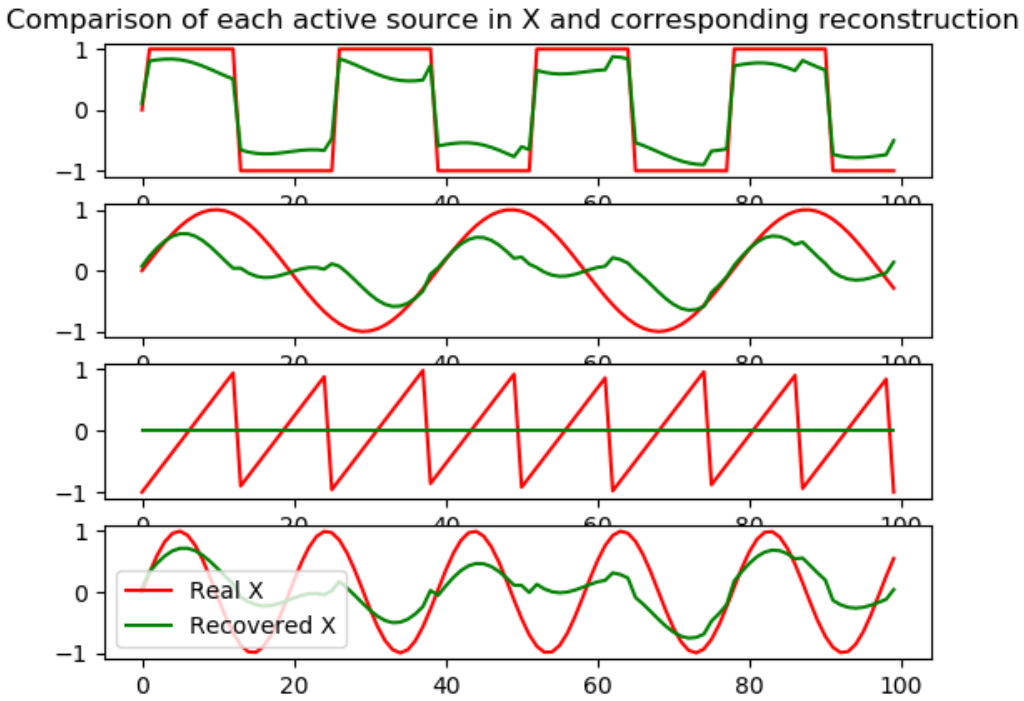

Figure B.1: Comparison of each source for k = 4

It is seen that one source was reconstructed as zero, that is the source was not reconstructed in the right location but in another. figure B.2 show the comparison for

one random chosen sample(row of X),

For the next visualisation we look at the first 4 sources of $\mathbf{X}$ and $\hat{\mathbf{X}}$ to see how well the estimation is.

Figure B.2: comparison of a single sample

Case 2 - $k < m$

The following variables are used:

- $m = 6$ (number of sensors)
- $n = 8$ (number of sources)
- $L = 100$ (number of samples)
- No segmentations – $\mathbf{Y} = \mathbf{AX}$
- Iterations = 1000
- $k = 4$ is active sources (row-wise)

Results

The MSE of case 1 was found to be 0.010 when rounded to the nearest three decimal. For visual comparison each active source are plotted against the reconstructed source in figure **??**.

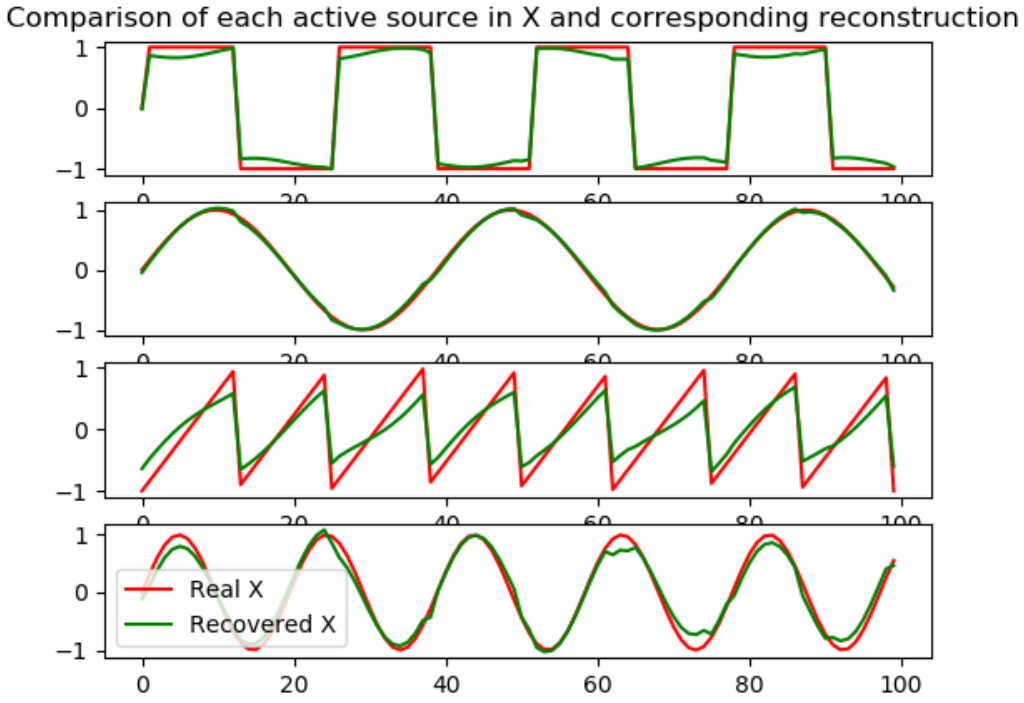

Figure B.3: Comparison of each source for k = 4

Figure B.4 show the comparison for one random chosen sample(row of X),

For the next visualisation we look at the first 4 sources of $\mathbf{X}$ and $\hat{\mathbf{X}}$ to see how well the estimation is.

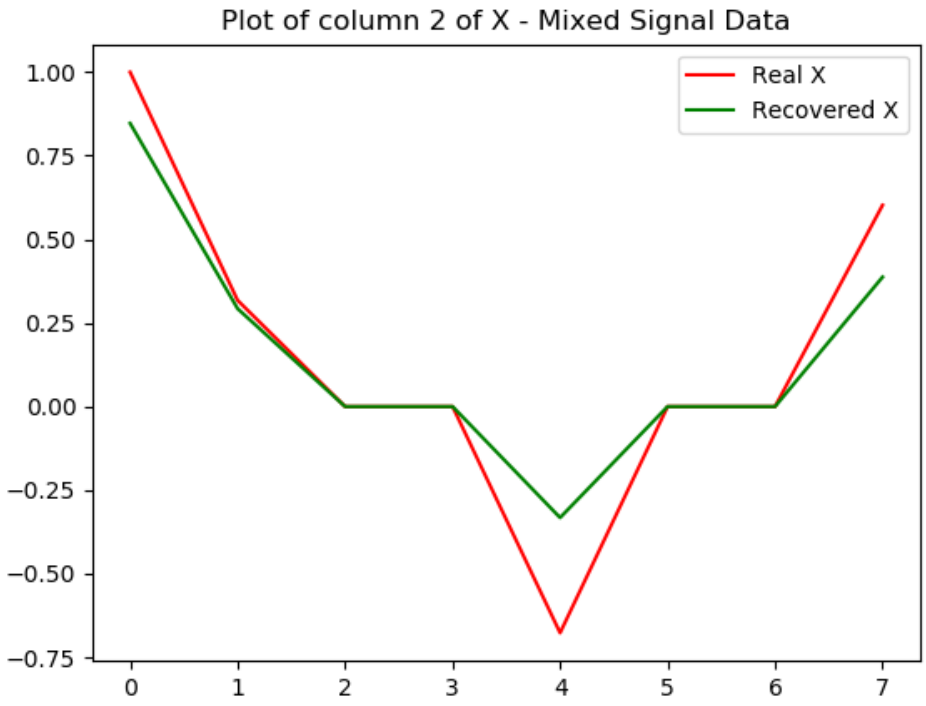

Figure B.4: comparison of a single sample

B.2 Rossler data test, M-SDL

General Setup