Bayesian Dictionary Learning for EEG Source Identification

Trine Nyholm Kragh & Laura Nyrup Mogensen
Mathematical Engineering, MATTEK

Master's Thesis

STUDENT REPORT
AALBORG UNIVERSITY

AALBORG UNIVERSITY
STUDENT REPORT

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

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: 35

Date of Completion:
October 28, 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:

Projektperiode:
Efterårssemestret 2019
Forårssemestret 2020

Projektgruppe:
Mattek9b

Deltager(e):
Trine Nyholm Kragh
Laura Nyrup Mogensen

Vejleder(e):
Jan Østergaard
Rasmus Waagepetersen

Oplagstal: 1

Sidetal: 35

Afleveringsdato:
28. oktober 2019

Abstract:

Her er resuméet

Preface

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

Aalborg University, October 28, 2019

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

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

Danish Summary

Dansk resume ?

Contents

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 localisation from EEG measurements. At first a motivation for the 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

Electroencephalography (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 [19]. This is what makes a kind of neural interaction across different parts of the brain(?).

EEG measurements are provided by a varies number of metal electrodes, referred to as sensors, carefully placed on a human scalp. Each sensor read the present electrical signals, which are then displayed on a computer, as a sum of sinusoidal waves relative to 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 measurements from a single sensor do not correspond to the activity of a single 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 measurments resulting from physical movement of e.g. eyes and jawbone [19]. 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[16,

p. 68][17].

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 highly interest to separate and localize the sources of the neural activities measured on the scalp. Note that a source do not correspond to a single neuron but is typically a collection of synchronized/phase locked active neurons which are generating a constructive interference resulting in a measurable signal on the scalp(?).

The waves resulting from EEG measurements have been classified into 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 rhythm for normal adults, associated with active thinking, active attention or solving concrete problems [16, p. 11]. An example of EEG measurements within the four categories is illustrated by figure 1.2.

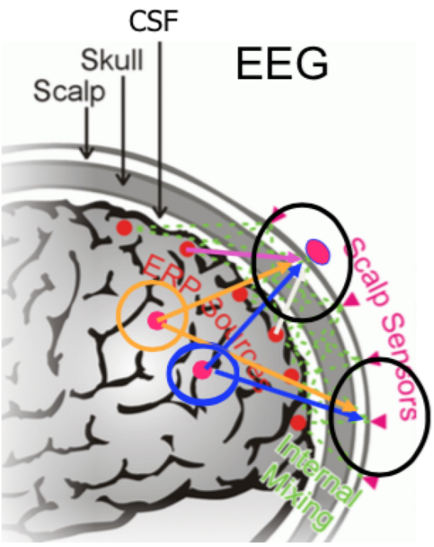

Figure 1.1: Illustration of volume conduction, source [4](we will make our own figure here instead)

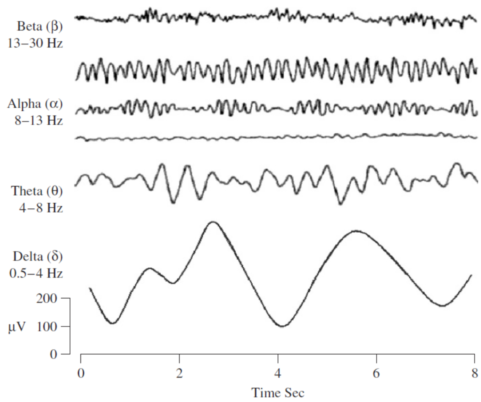

Figure 1.2: Example of time dependent EEG measurements within the four defined categories, source [16]

EEG is widely used within the medical field, especially research of the cognitive processes in the brain. Diagnosis and management of neurological disorders such as epilepsy is one example.

EEG capitalize on the procedure being non-invasive and fast. Neural activity can be measured within fractions of a second after a stimuli has been provided [19, p. 3]. When a person is exposed to a certain stimuli, e.g. visual or audible, the measured activity is said to result from evoked potential.
Over the past two decades, especially functional integration has become an area of interest[9]. Within neurobiology functional integration referrers to the study of the correlation among activities in different regions of the brain. In other words, how do different part of the brain work together to process information and conduct a response[10]. For this purpose separation and localisation of the single sources which contribute to the EEG measurement is of interest. An article from 2016 point 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 do not allow interpretations about the interaction between sources[17].

The hearing aid industry is one example where this research is highly prioritised. At Eriksholm research center which is a part of the hearing aid manufacture Oticon cognitive hearing science is a research area within fast development[18]. One main purpose of Eriksholm is to make it possible for a hearing aid to identify the attended sound source and hereby exclude noise from elsewhere [1], [5]. This is where EEG and occasionally so called in-ear EEG is interesting, especiallaly in conjunction with the technology of beamforing, which allows for receiving only signals from a specific direction. It is essentially the well known but unsolved cocktail problem which is sought improved by use of EEG. However the focus of this research do consider the correlation between EEG measurements and the sound source rather than localisation of the activated source from the EEG[1]. Hence a source localisation approach could potentially be of interest regarding hearing aids in order to improve the results. (Furthermore, a real-time application to provide feedback from EEG measurements would be essential.)?

1.1.1 Modelling

Considering the issue of localising activated sources from EEG measurements, a known option is to model the observed data by the following linear system

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

where $\mathbf{Y} \in \mathbb{R}^{M \times N_d}$ is the EEG measurements from M sensors at N_d data points, $\mathbf{A} \in \mathbb{R}^{M \times N}$ is an unknown mixing matrix and $\mathbf{X} \in \mathbb{R}^{N \times N_d}$ is 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 [4]. 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 the model is referred to as the EEG inverse problem.

To solve the EEG inverse problem the concept of compressive sensing makes a solid foundation including sparse signal recovery and dictionary learning. Independent Component Analysis (ICA) is a common applied method to solve the inverse problem [13], [12], here statistical independence between source activity is assumed.
Application of ICA have shown great results regarding source separation of high-density EEG. Furthermore, an enhanced signal-to-noise ratio of the unmixed independent source time series processes allow essential study of the behaviour and relationships between multiple EEG source processes [7].
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[2].
This means that the EEG inverse problem can not be over-complete(er det correct i forhold til teorien om ICA?). 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 [4]. 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 localisation 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[2]. Instead of identifying one mixing matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ this approach learns N_{model}(number of sources? or datapoints) 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 [15]. 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 [20]. Here the hard orthonormal constraint in ICA is replaced with a soft reconstruction cost.
In 2015 O. Balkan et. al., [2], 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 uncorrelation 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[2], thus it is considered the state of the art within the area of source identification.

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 [3] also by O. Balkan et. al., also targeting the case of more active sources than sensors [3]. 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.

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 overcomplete EEG inverse problem, in order to localise 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 localised from the EEG inverse problem, in the overcomplete 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 concept compressive sensing. Associated theory regarding sparse signal recovery is described along the limitations of the common solution approaches. Finally the state of the art methods regarding non-sparse signal is presented.

3.1 Linear Algebra

A relation between some observed measurements $\mathbf{y}$ can be described as a linear combinations between a coefficient matrix 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 a 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. The linear model consist of M equations and N unknown.

For the case of $\mathbf{A}$ been a square matrix ($M = N$) a solution can be found to the linear model if the $\mathbf{A}$ has full rank – $\mathbf{A}$ consist of linearly independent columns or rows. For $M > N$ the matrix said to have full rank when the columns are linearly independent. For $M < N$ the matrix has full rank when the rows are linearly independent. For linearly systems/model with $M = N$ is called determined, $M > N$ overdetermined and $M < N$ underdetermined.
When full rank do not occur the matrix is then called rank-deficient.

By inverting $\mathbf{A}$ from (3.1) the unknowing vector $\mathbf{x}$ can be achieved. Square matrix is invertible if an only if its has full rank or its determinant $\det(\mathbf{A}) \neq 0$. For rectangular matrices ($M > N$ and $M < N$) left-sided and right-sided inverse exists. With the left-inverse the least norm solution of (3.1) can be found.

For an determined system there will exist a unique solution. For an overdetermined system there do not exist a solution and for underdetermined systems there exist infinitely many solutions.
For rank-deficient matrices there do not exist an inverse and therefore (3.1) can not be solve by inverting the model. But rank-deficient matrices meaning have non-empty null space leading to infinitely solutions to (3.1) [6, p. ix].

As described in chapter 1 the linear model of interest consist of M sensors which is known and N sources which is unknown. Furthermore, we also have that $M < N$ – an underdetermined system. It is therefore of interest to find a solution in the infinitely 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. Assuming linear acquisition of the original information the relation between the measurements and the signal to be recovered is described by the linear model giving in (3.1) [8].

In compressive sensing terminology, $\mathbf{x}$ is the signal of interest which is sought recovered by solving the linear system (3.1). In the typical compressive sensing case where $M < N$ the system becomes underdetermined and there are infinitely many solutions, provided that a solution exist. Such system is also referred to as overcomplete *(as the number of column basis vectors is greater than the dimension of the input)*.
However, by enforcing certain sparsity constraints it is possible to recover the wanted signal [8], hence the term sparse signal recovery.

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

$$\|\mathbf{x}\|_0 \coloneqq \mathrm{card}(\mathrm{supp}(\mathbf{x})).$$

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

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

where $[N]$ a set of integers $\{1, 2, \cdots, N\}$ [8, 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) which is k-sparse with $k < M$ an optimisation problem can be used. An optimisation problem is defined as

$$\min f_0(\mathbf{x}) \quad \text{s.t} \quad f_i(\mathbf{x}) \leq b_i, \quad i \in [n],$$

where $f_0 \ : \ \mathbb{R}^N \mapsto \mathbb{R}$ is an objective function and $f_i \ : \ \mathbb{R}^N \mapsto \mathbb{R}$ are the constraint functions.
To find the k-sparse solution our optimisation problem can be written as

$$\min_{\Omega_k} \|\mathbf{x}\|_0 \quad \text{s.t} \quad \mathbf{y} = \mathbf{A}\mathbf{x},$$

where $\mathbf{x}$ is all possible candidates to a k-sparse signal $\mathbf{x}^*$. The objective function is given by an ℓ_0 norm with the constraint function been the linear model described in (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 a NP-hard problem. Instead by replacing the ℓ_0-norm with the ℓ_1-norm, the optimisation problem can approximated and therefore become computational feasible [6, p. 27]

$$\min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{s.t} \quad \mathbf{y} = \mathbf{A}\mathbf{x}. \tag{3.2}$$

With this optimisation problem we find the best k-term approximation of the signal $\hat{\mathbf{x}}^*$. This method is referred to as Basis Pursuit.
The following theorem justifies that the ℓ_1 optimisation problem finds a sparse solution [8, p. 62-63].

Theorem 3.2.1
A measurement matrix $\mathbf{A}$ is defined as $\mathbf{A} \in \mathbb{R}^{M \times N}$ with columns $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_N]$. By assuming uniqueness of a solution $\mathbf{x}^*$ of

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

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 we need to realise that the size of the set $\{\mathbf{a}_j, j \in S\} \leq M$, that we can not have more than M linearly independence columns. So when $M \ll N$ then we automatically achieve a sparse signal.

Proof

By way of contradiction, lets assume that the set $\{\mathbf{a}_j, j \in S\}$ is linearly dependent with $S = \text{supp}(\mathbf{x}^*)$. This means that there exists a non-zero vector $\mathbf{v} \in \mathbb{R}^N$ supported on S such that $\mathbf{Av} = \mathbf{0}$. Then, for any $t \neq 0$,

$$\|\mathbf{x}^*\|_1 < \|\mathbf{x}^* + t\mathbf{v}\|_1 = \sum_{j \in S} |x_j^* + tv_j| = \sum_{j \in S} \text{sgn}(x_j^* + tv_j)(x_j^* + tv_j).$$

If $|t|$ is small enough, namely, $|t| < \min_{j \in S} \frac{|x_j^*|}{\|\mathbf{v}\|_\infty}$, then

$$\text{sgn}(x_j^* + tv_j) = \text{sgn}(x_j^*), \quad \forall j \in S.$$

It the follows that

$$\|\mathbf{x}^*\|_1 < \sum_{j \in S} \text{sgn}(x_j^*)(x_j^* + tv_j) = \sum_{j \in S} \text{sgn}(x_j^*) x_j^* + t \sum_{j \in S} \text{sgn}(x_j^*) v_j = \|\mathbf{x}^*\|_1 + t \sum_{j \in S} \text{sgn}(x_j^*) v_j.$$

This is a contradiction, because we can always choose a small $t \neq 0$ such that $t \sum_{j \in S} \text{sgn}(x_j^*) v_j \leq 0$ and therefore the set $\{\mathbf{a}_j, j \in S\}$ must be linearly independent.■

The Basis Pursuit makes the foundation of several algorithms solving alternative versions of (3.2) where noise is incorporated. A different type of solution method includes a greedy algorithm such as the Orthogonal Matching Pursuit [8, P. 65].

Algorithm 1 Orthogonal Matching Pursuit (OMP)

1. Give an measurement matrix $\mathbf{A}$ and a measurement vector $\mathbf{y}$
2. Initial $S^0 = \varnothing$ and $x_0 = \mathbf{0}$
3. Iterate until stopping criterion is met:

$$S^{n+1} = S^n \cup \{j_{n+1}\}, \quad j_{n+1} = \arg\max_{j \in [N]} (|(\mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{x}^n))_j|)$$
$$x^{n+1} = \arg\min_{\mathbf{z} \in \mathbb{C}^N} (\|\mathbf{y} - \mathbf{Az}\|_2, \text{supp}(\mathbf{z}) \subset S^{n+1})$$

4. $\mathbf{x}^* = \mathbf{x}^n$

3.2.3 Conditions on the dictionary

The construction of the matrix $\mathbf{A}$ is of course essential for the solution of the optimisation problem. So far no one has manage to construct a matrix which is proved to be optimal for some compressive sensing set up. However some certain constructions have shown sufficient recovery guarantee.

To ensure an exact or an approximate reconstruction of the sparse signal $\mathbf{x}$ some conditions associated to the matrix $\mathbf{A}$ must be satisfied.
This includes at first the null space condition, a property of the A matrix which is hard to check in practise. The Restricted isometry condition is a stronger condition concerning the orthogonality of the matrix. Furthermore the coherence of a matrix is a measure of quality and is used to determine whether the matrix A is a good choice for the optimisation problem.
(Is it necessary describe these conditions on A in details?)

Dictionary learning

In cases where the dictionary is unknown it is possible to learn the dictionary from the observed measurement provided that several observations are available $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_n]$. In dictionary learning framework the inverse problem is defined as

$$\min_{A,X} = \frac{1}{2} \sum_{s=1}^{N_d} \|\mathbf{Y} - \mathbf{A}\mathbf{X}\|_F^2 + \gamma \sum_{s=1}^{N_d} g(\mathbf{x}_s),$$

where the function $g(\cdot)$ promotes sparsity of the source vector at time t The true dictionary $\mathbf{A}$ is recovered if the sources $\mathbf{x}_s$ are sparse $(k_s < M)$.

Include specific dictionary learning algorithm here...

3.2.4 Multiple Measurement Vector Model

The linear model (3.1) is also referred to as a single measurement model. In order to adapt the model to practical use the model is expanded to include multiple measurements and take noise into account.
A multiple measurement vector (MMV) model consist of the observation matrix $\mathbf{Y} \in \mathbb{R}^{M \times L}$, the source matrix $\mathbf{X} \in \mathbb{R}^{N \times L}$ which have $k \leq M$ rows that are non-zero and a dictionary matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$:

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

where L denote the number of observed samples each consisting of M measurements and $\mathbf{E} \in \mathbb{R}^{M \times L}$. From the MMV model the non-zero rows of the source matrix $\mathbf{X}$ are the ones of interest, which one want to recover [4, p. 11].

3.2.5 Limitations of compressive sensing

(we are not yet sure where to put a section explaining the issue of $k > M$.)

- If the source signal is sparse it is enough just to find the non-zero rows of X denoted by the set S, because then the source signal can be obtained by the psudo-inverse solution $\hat{\mathbf{X}} = \mathbf{A}_S^{perp}\mathbf{Y}$ where $\mathbf{A}_S$ is derived from the dictionary matrix $\mathbf{A}$ by deleting the columns associated with the zero rows of X. S is called the support. (We identify the locations of sources)

- in general for $k > M$ it is not possible to recover $\mathbf{x}$ as the system is underdetermined/overcomplete, furthermore we can not find the true dictionary $\mathbf{A}$ by dictionary learning methods because when $k > M$, where M is the dimension of $\mathbf{y}$, any random dictionary can be used create $\mathbf{y}$ from $\geq M$ basis vectors. that is generally the accuracy of recovery a $\mathbf{A}$ increases as $k << M$.

3.3 Covariance-Domain Dictionary Learning

Covariance-domain dictionary learning (Cov-DL) is an algorithm proposed in [2] which claims to be able to identify more sources N than available observations M for the linear model.

Consider the multiple measurement vector model as above

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

Let s be the index of time segments that the observed data $\mathbf{Y} \in \mathbb{R}^{M \times L}$ is divided into and let S_f be the sample frequency. As such the observed data is divided into segments $\mathbf{Y}_s \in \mathbb{R}^{M \times t_s S_f}$, 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

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

Cov-DL takes advantage of the dictionary framework and transformation into another domain – covariance domain – to recover the mixing matrix $\mathbf{A}$ from the observed data $\mathbf{Y}$. An important aspect of this method is the prior assumption that the sources are statistical independent within the defined time segments. This implies the entries in $\mathbf{X}_s$ to be uncorrelated.

Cov-DL works together with another algorithm to find the source matrix $\mathbf{X}$, in this thesis M-SBL is used for the source recovery and is described in section 4.1.
In this section we assume that $\mathbf{X}$ is known but in practice a random sparse matrix will be used to represent the sources.
The section is inspired by chapter 3 in [4] and the article [2].

3.3.1 Covariances domain representation

The observed data $\mathbf{Y}_s$ can be described in the covariance domain by the sample covariance matrix. Considering the observations $\mathbf{Y}_s \in \mathbb{R}^{M\times L}$ the sample covariance is defined to find the covariance among the M variables across the L observations, that is essentially the covariance matrix averaged over all observations, resulting in a $M \times M$ matrix $\mathbf{\Sigma}_{\mathbf{Y}_s} = [\sigma_{jk}]$. Each entry is defined by[wiki?]

$$\sigma_{jk} = \frac{1}{L}\sum_{i=1}^{L}(y_{ji} - \mathbb{E}_s[y_j])(y_{ki} - \mathbb{E}_s[y_k])$$

Let the observations(argument for at vi antager zero mean på vores observationer, pre-normalisering?) be normalised resulting in zeros mean $\mathbb{E}_s[\mathbf{Y}_s] = 0$.
Using matrix notation the sample covariance of $\mathbf{Y}_s$ can be written as

$$\Sigma_{\mathbf{Y}_s} = \frac{1}{L}\mathbf{Y}_s\mathbf{Y}_s^T$$

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

$$\mathbf{\Sigma}_{\mathbf{X}_s} = \frac{1}{L_s}\mathbf{X}_s\mathbf{X}_s^T$$

From the assumption of uncorrelated sources the sample covariance matrix is expected to be nearly diagonal, thus it can be expressed as

$$\mathbf{\Sigma}_{\mathbf{X}_s} = \Lambda + \mathbf{E}_s.$$

where Λ is a diagonal matrix consisting of the diagonal entries of $\Sigma_{\mathbf{X}_s}$ and $\mathbf{E}_s$ contains the remaining entries which is expected to be zeros or nearly zeros[2].

Each segment of observations can be modelled as

skal vi have $\mathbb{E}_s[]$ omkring her eller ej?, wiki covariance. giver det mening at udelade $\frac{1}{L}$ på begge sider her?

$$\begin{aligned}
\mathbf{Y}_s\mathbf{Y}_s^T &= (\mathbf{A}\mathbf{X}_s)(\mathbf{A}\mathbf{X}_s)^T \\
&= \mathbf{A}\mathbf{X}_s\mathbf{X}_s^T\mathbf{A}^T \\
&= \mathbf{A}\Sigma_{X_s}\mathbf{A}^T \\
&= \mathbf{A}\mathbf{\Lambda}\mathbf{A}^T + \mathbf{E}_s = \sum_{i=1}^{N}\Lambda_{ii}\mathbf{a}_i\mathbf{a}_i^T + \mathbf{E}_s \qquad (3.3)
\end{aligned}$$

Remember that N is the dimension of X hence the number of possible sources and k is the number of active sources. It has been shown that by this model it is possible to identify $O(M^2)$ sources given the true dictionary[14](dog er vektoriseringen også includeret i resultatet - så måske flyttes udtalelsen?). The purpose of the Cov-DL algorithm is instead to find the dictionary $\mathbf{A}$ from this expression and then still allow for $O(M^2)$ sources to be identified.

3.3.2 Determination of dictionary

er det muligt at vektoriceringen kun er til for at gøre dictionary learning problemet simplere? og ikke har noget med $O(M^2)$ at gøre, læs Pal2015

In order enable the possibilities of identifying $O(M^2)$ sources and learning the corresponding dictionary A the model in (3.3) is rewritten. At first the both sides of the expression is vectorized. Because the covariance matrix 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 vectorisation of its lower triangular part. Further let $\text{vec}^{-1}(\cdot)$ be the inverse function. Note that $\Lambda_{s_{ii}}$ is a scalar hence not vectorised

$$\text{vec}(\boldsymbol{\Sigma}_{\mathbf{Y}_s}) = \sum_{i=1}^{N} \Lambda_{s_{ii}} \text{vec}(\mathbf{a}_i \mathbf{a}_i^T) + \text{vec}(\mathbf{E}_s) \tag{3.4}$$

$$= \sum_{i=1}^{N} \mathbf{d}_i \Lambda_{s_{ii}} + \text{vec}(\mathbf{E}_s) \tag{3.5}$$

$$= \mathbf{D}\boldsymbol{\delta}_s + \text{vec}(\mathbf{E}_s), \quad \forall s. \tag{3.6}$$

The vector $\boldsymbol{\delta}_s \in \mathbb{R}^N$ contains the diagonal entries of the source sample-covariance matrix $\boldsymbol{\Sigma}_{\mathbf{X}_s}$ and the matrix $\mathbf{D} \in \mathbb{R}^{M(M+1)/2 \times N}$ consists of the columns $\mathbf{d}_i = \text{vech}(\mathbf{a}_i \mathbf{a}_i^T)$. Note that $\mathbf{D}$ and δ_s are unknown while the left hand side is known from the observed data.

include noise in the original problem

From this expression it is now possible to learn $\mathbf{D}$ and then find the associated matrix $\mathbf{A}$. Comparing this expression to the original compressive sensing problem (3.1) it is clear that higher dimensionality is achieved by representing the problem in the covariance domain. Which allows for the number of active sources to exeed the number of observartions(måske for tidligt at konkludere her).

The Cov-DL method consists of two different algorithms for recovery of $\mathbf{D}$ and $\mathbf{A}$ depending on the number of total sources N relative to the number of observations M.

Cov-DL1 – overcomplete D

The case where $N > \frac{M(M+1)}{2}$ results in an overcomplete system similar to the original system being overcomplete when $N > M$. Though, it is again possible to solve the overcomplete system if certain sparsity is withhold. Namely $\boldsymbol{\delta}_s$ being $\frac{M(M+1)}{2}$-sparse. Note that this sparsity constraint is weaker than the original constraint $k < M$, which allows for identification of remarkably more sources than within the original domain as it is not necessarily violated when $k > M$. Assuming a sufficient sparsity on $\boldsymbol{\delta}_s$ it is possible to learn the dictionary matrix of the covariance domain $\mathbf{D}$ by traditional dictionary learning methods, as introduced in section 3.2.3, applied to the observations represented in the covariance domain $\text{vec}(\boldsymbol{\Sigma}_{\mathbf{Y}_s})$ $\forall s$.
When $\mathbf{D}$ is known it is possible to find the original mixing matrix $\mathbf{A}$ through the

relation $\mathbf{d}_i = \text{vec}(\mathbf{a}_i\mathbf{a}_i^T)$.
To do so each column is found by the optimisation problem

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

for which the global minimizer is $\mathbf{a}_i^* = \sqrt{\lambda_i}\mathbf{b}_i$. Here λ_i is the largest eignevalue of $\text{vec}^{-1}(\mathbf{d}_i)$ and $\mathbf{b}_i$ is the corresponding eigenvector.

redegørelse for resultatet her skal laves

Algorithm 2 Cov - DL2 undercomplete D

1. Segmentation of measurements
2. Transformation to covariance domain
3. Vectorising
4. Learn D, by ?
5. Find A

Cov-DL2 – undercomplete D

Chapter 4

Independent Component Analysis

(it is not the intention to keep a whole chapter about ICA within the rapport, but rather a smaller section within the chapter of compressive sensing as a common solution method, and the rest as appendix.)

Independent component analysis (ICA) is a method*(within sparse signal recovery?)* which can separate statistical independent sources $\mathbf{X}$ and identify the mixing matrix $\mathbf{A}$ given the observed measurements $\mathbf{Y}$.
The main aspect of ICA is to assume statistical independent between the wanted components and nongaussiantity of the data. [11, p. 3].
Through this section the mathematical concepts of ICA will be explained and defined in the noise-less case.

To understand what ICA is let us describe a situation where ICA could be used. Two people are having a conversation inside a room full of people which talk simultaneous with each other. The conversation between the two first mentioned people will the be affected by the surrounding conversations and noise. Such environment is often referred to as the cocktail party problem and is a difficult environment to be in as a hearing impaired.
Let us describe the situation with some mathematical notations. The observed conversation which is affected by surrounding noise is denoted $\mathbf{y}$, the original individual conversations in the room is denoted by $\mathbf{x}$. All the original conversations are effected by each other and possibly noise, let us denoted this mixture by a matrix $\mathbf{A}$. We omit the time indexes and the time dependence to view the problem with random

vectors. The problem can then be described as a linear model

$$\mathbf{y} = \mathbf{A}\mathbf{x} = \sum_{i=1}^{n} \mathbf{a}_i x_i, \tag{4.1}$$

$$y_i = a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n, \quad i = 1, \ldots, n. \tag{4.2}$$

The only known variables in this model are the observed conversation $\mathbf{y}$, the rest are unknown.

ICA is a method which can be use to recover the unknown mixture $\mathbf{A}$ and sources $\mathbf{x}$, which in the ICA aspect are the components, from the known $\mathbf{y}$. By use of the statistical properties of the components $\mathbf{x}$ it is then possible to estimate/recover the original conversations and then the mixture $\mathbf{A}$.
One of the properties is the assumption that the components $\mathbf{x}$ are statistical independent. By independence, one means that the joint probability density function (pdf) can be factorised into the marginal pdfs of the components $\mathbf{x}$, that is

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

for the random variables x_i.
Furthermore it is assumed that the independent components $\mathbf{x}$ do not have Gaussian distributions as this will effect the recovery of $\mathbf{x}$ due to ICA using higher-order cumulant method. For Gaussian distribution the cumulant will be zero which is not wanted in the recovery process. Thus the distribution of the independent components are unknown. This will further be described in section 4.0.1.
The last assumption is that the mixing matrix $\mathbf{A}$ must be a square matrix – there must be the same number of independent components as observation [11, p. 152-153].

From the right-hand side of (4.1) being unknown some statistics can not be computed e.g. the variance of $\mathbf{x}$. Instead ICA assume that $\mathbf{x}$ has unit variance (= 1). This of course must also apply to the mixing matrix $\mathbf{A}$ which will be restricted in the recovery method, described in section 4.0.1. By this addition the algorithm for ICA is simplified. To simplify even more, without loss of generality assume that $\mathbb{E}[\mathbf{y}] = 0$ and $\mathbb{E}[\mathbf{x}] = 0$.
To achieve zero mean, the observed data $\mathbf{y}$ is normalised as a preprocessing. This addition do not affect the recover/estimation of $\mathbf{A}$ [11, p. 154].

Another preprocessing step is to whiten the observed data $\mathbf{y}$. This ensures that the independent components $\mathbf{x}$ are uncorrelated and have variance equal 1. Furthermore, this also reduce the complexity of ICA and therefore simplifies the recovering process.
Whiting is a process which uses the eigenvalue decomposition (EVD) to find the

eigenvalues and associated eigenvectors from the covariance of observed data $\mathbf{y}$:

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

where $\mathbf{D}$ is a diagonal matrix of eigenvalues and $\mathbf{E}$ is the associated eigenvectors. With $\mathbf{E}$ and $\mathbf{D}$ a whiting matrix is constructed as

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

By multiplying the observed data $\mathbf{y}$ with a whiting matrix $\mathbf{V}$ the data becomes white:

$$\begin{aligned} \mathbf{y}_{\text{white}} &= \mathbf{V}\mathbf{y}, \\ \mathbf{y}_{\text{white}} &= \mathbf{V}\mathbf{A}\mathbf{x} = \mathbf{A}_{\text{white}}\mathbf{x} \end{aligned} \tag{4.3}$$

By the whiting process the mixing matrix $\mathbf{A}_{\text{white}}$ becomes orthogonal because

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

This mean that ICA can restrict its search for $\mathbf{A}_{\text{white}}$ to the orthogonal matrix space- That is instead of estimateing n^2 parameters ICA now only has to estimate an orthogonal matrix which have $n(n-1)/2$ parameters/degrees of freedom [11, p. 159].
Furthermore, to support the assumption of nongaussiantity for an orthogonal mixing matrix $\mathbf{A}_{\text{white}}$ it is not possible to distinct the pdfs of the $\mathbf{y}_{\text{white}}$ and $\mathbf{x}$ as $\mathbf{A}_{\text{white}}$ is no longer included in the pdf of $\mathbf{y}_{\text{white}}$ and therefore are the two pdfs equal [11, p. 161-163].

can we elaborate this last part

4.0.1 Recovery of the Independent Components

A way to recover the independent components could be to take advantage of the assumption of nongaussiantity. By finding the estimate which maximise the nongaussiantity the real independent component can be found based on the fact that it must have a nongaussian distribution as mention in section 4. But before the estimation a measure of nongaussiantity must be introduce – this could be the kurtosis.

Kurtosis

Kurtosis is a quantitative measure used for nongaussianity of random variables. (Excess) Kurtosis of a random variable y is defined as

Tjek lige op denne definition

$$\begin{aligned} \text{kurt}(y) &= \mathbb{E}\left[\left(\frac{y-\mu}{\sigma}\right)^4\right] - 3 \\ &= \mathbb{E}[y^4] - 3(\mathbb{E}[y^2])^2, \end{aligned}$$

which is the fourth-order cumulant of the random variable y. By assuming that the random variable y has variance $\mathbb{E}[y^2] = 1$, the kurtosis is rewritten as

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

The kurtosis of nongaussian random variables will then almost always be different from zero. For gaussian random variables the fourth moment equals $3(\mathbb{E}[y^2])^2$ thus the kurtosis will then be zero [11, p. 171].
By using the absolute value of the kurtosis gaussian random variables are still zero but the nongaussian random variables will be greater than zero. In this case the random variables are called supergaussian.

One complication with kurtosis as a measure is that kurtosis is sensitive to outliers [11, p. 182].

The Gradient Algorithm To recover the independent components $\mathbf{x}$ the nongaussianity is wish maximised. One way to do this is to use a gradient algorithm to maximise the kurtosis of $\mathbf{y}$.

The idea behind a gradient algorithm is to move in certain directions computed from the gradient of an objective function – in this case it is the kurtosis – until convergence is achieved.
Let $\mathbf{w}$ be an initial vector used to compute the direction and let $y = \mathbf{w}^T\mathbf{y}_{\text{white}}$ given some samples of the observed and preprocessed vector. The direction $\mathbf{w}$ which provide the highest kurtosis is the new direction for the algorithm. This continues until convergence is reach defined by some tolerance.
Regn lige efter
The gradient of $|\operatorname{kurt}(\mathbf{w}^T\mathbf{y}_{\text{white}})|$ is computed as

$$\begin{aligned}\frac{\partial|\operatorname{kurt}(\mathbf{w}^T\mathbf{y}_{\text{white}})|}{\partial\mathbf{w}} &= 4\operatorname{sign}(\operatorname{kurt}(\mathbf{w}^T\mathbf{y}_{\text{white}}))(\mathbb{E}[\mathbf{y}_{\text{white}}(\mathbf{w}^T\mathbf{y}_{\text{white}})^3] - 3\mathbf{w}\mathbb{E}[(\mathbf{w}^T\mathbf{y}_{\text{white}})^2] \\ &= 4\operatorname{sign}(\operatorname{kurt}(\mathbf{w}^T\mathbf{y}_{\text{white}}))(\mathbb{E}[\mathbf{y}_{\text{white}}(\mathbf{w}^T\mathbf{y}_{\text{white}})^3] - 3\mathbf{w}\|\mathbf{w}\|^2).\end{aligned} \tag{4.4}$$

The absolute value of kurtosis is optimised onto the unit sphere, $\|\mathbf{w}\|^2 = 1$, the algorithm must project onto the unit sphere in every step. This can easily be done by dividing $\mathbf{w}$ with its norm.
Furthermore, the last part of (4.4) can be omitted as it do not effect the direction. The expectation operator is omitted to achieve an adaptive algorithm:

$$\frac{\partial|\operatorname{kurt}(\mathbf{w}^T\mathbf{y}_{\text{white}})|}{\partial\mathbf{w}} = 4\operatorname{sign}(\operatorname{kurt}(\mathbf{w}^T\mathbf{y}_{\text{white}}))\mathbf{y}_{\text{white}}(\mathbf{w}^T\mathbf{y}_{\text{white}})^3$$

The expectation operator from the definition of kurtosis can not be omitted and must therefore be estimated. This can be done by a time-average estimate, denoted as γ:

$$\gamma = ((\mathbf{w}^T \mathbf{y}_{\text{white}})^4 - 3) - \gamma$$

This results in the following gradient algorithm.

Algorithm 3 Gradient Algorithm with Kurtosis

1. Center the observed data to achieve zero mean
2. Whiten the centered data
3. Create the initial random vector $\mathbf{w}$ and the initial value for γ
4. Compute $\mathbf{w} = \gamma \mathbf{y}_{\text{white}}(\mathbf{w}^T \mathbf{y}_{\text{white}})^3$
5. Normalise $\mathbf{w} = \frac{\mathbf{w}}{\|\mathbf{w}\|}$
6. Update $\mathbf{w}$
7. Update $\gamma = ((\mathbf{w}^T \mathbf{z})^4 - 3) - \gamma$
8. Repeat until convergence
9. Independent components are found as $\mathbf{x} = \mathbf{w}\mathbf{y}_{\text{white}}$

Fixed-Point Algorithm - FastICA A fixed-point algorithm to maximise the nongaussianity is more efficient than the gradient algorithm as the gradient algorithm converge slow depending on the choice of γ. The fixed-point algorithm is an alternative that could be used. By using the gradient given in (4.4) and then set the equation equal with $\mathbf{w}$:

$$\mathbf{w} \propto (\mathbb{E}[\mathbf{y}_{\text{white}}(\mathbf{w}^T \mathbf{y}_{\text{white}})^3] - 3\|\mathbf{w}\|^2 \mathbf{w})$$

By this new equation, the algorithm find $\mathbf{w}$ by simply calculating the right-hand side:

$$\mathbf{w} = \mathbb{E}[\mathbf{y}_{\text{white}}(\mathbf{w}^T \mathbf{y}_{\text{white}})^3] - 3\mathbf{w}$$

As with the gradient algorithm the fixed-point algorithm do also divide the found $\mathbf{w}$ by its norm. Therefore is $\|\mathbf{w}\|$ omitted from the equation.
Instead of γ the fixed-point algorithm compute $\mathbf{w}$ directly from previous $\mathbf{w}$.

The fixed-point algorithm have been summed in the following algorithm.

Algorithm 4 Fixed-Point Algorithm with Kurtosis

1. Center the observed data to achieve zero mean
2. Whiten the centered data
3. Create the initial random vector $\mathbf{w}$
4. Compute $\mathbf{w} = \mathbb{E}[\mathbf{y}_{\text{white}}(\mathbf{w}^T\mathbf{y}_{\text{white}})^3] - 3\mathbf{w}$
5. Normalise $\mathbf{w} = \frac{\mathbf{w}}{\|\mathbf{w}\|}$
6. Update $\mathbf{w}$
7. Repeat until convergence
8. Independent components are found as $\mathbf{x} = \mathbf{w}\mathbf{y}_{\text{white}}$

The fixed-point algorithm is also called for FastICA as the algorithm has shown to converge fast and reliably, then the current and previous $\mathbf{w}$ laid in the same direction [11, p. 179].

Negentropy

Another measure of nongaussianity is the negentropy which is based on the differential entropy. The differential entropy H of a random variable/vector $\mathbf{y}$ with density $p_y(\boldsymbol{\theta})$ is defined as

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

The entropy describes the information of a random variable. For variables becoming more random the entropy becomes larger, e.g. gaussian random variables have a high entropy, in fact gaussian random variables have the highest entropy among the random variables of the same variance [11, p. 182].

To use the negentropy to define the nongaussianity within random variables, the differential entropy is normalised to obtain a entropy value equal to zero when the random variable is gaussian and non-negative otherwise. The negentropy J 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 $\mathbf{y}$ [11, 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. Instead it could be an idea to use an approximation of the negentropy.

Approximation of Negentropy

The way to approximate the negentropy is to look at the high-order cumulants using polynomial density expansions such that the approximation could be given as

$$J(y) \approx \frac{1}{12}\mathbb{E}[y^3]^2 + \frac{1}{48}\mathrm{kurt}(y)^2. \tag{4.5}$$

The random variable y has zero mean and unit variance and the kurtosis is introduced in the approximation. The approximation suffers from nonrobustness with the kurtosis and therefore a more generalised approximation is presented to avoid the nonrobustness.

For the generalised approximation the use of expectations of nonquadratic functions is introduced. The polynomial functions y^3 and y^4 from (4.5) are replaced by G^i with i being an index and G being some function. The approximation in (4.5) then becomes

$$J(y) \approx (\mathbb{E}[G(y)] - \mathbb{E}[G(\nu)])^2.$$

The choice of G can lead to a better approximation than (4.5) and by choosing one which do not grow to fast more robust estimators can be obtained. The choice of G could be the two following functions

$$\begin{aligned} G_1(y) &= \frac{1}{a_1}\log(\cosh(a_1 y)), \quad 1 \leq a_1 \leq 2 \\ G_2(y) &= -\exp\left(\frac{-y^2}{2}\right) \end{aligned}$$

Gradient Algorithm with Negentropy As described in section 4.0.1 the gradient algorithm is used to maximising negentropy. The gradient of the approximated negentropy is given as

$$\mathbf{w} = \gamma \mathbf{y}_{\mathrm{white}} g(\mathbf{w}^T \mathbf{y}_{\mathrm{white}})$$

with respect to $\mathbf{w}$ and where $\gamma = \mathbb{E}[G(\mathbf{w}^T\mathbf{y}_{\mathrm{white}})] - \mathbb{E}[G(\nu)]$ with ν being the standardised gaussian random variable. g is the derivative of the nonquadratic function G. To omitted the expectation γ as we did with the sign of kurtosis, γ is estimated as

$$\gamma = (G(\mathbf{w}^T\mathbf{y}_{\mathrm{white}}) - \mathbb{E}[G(\nu)]) - \gamma.$$

For the choice of g the derivative of the functions presented in (4.6) could be use to achieve a robust result. Alternative a derivative which correspond to the fourth-power as seen in the kurtosis could be used. The functions g could be

$$
\begin{aligned}
g_1(y) &= \tanh(a_1 y), \quad 1 \leq a_1 \leq 2 \\
g_2(y) &= y \exp\left(\frac{-y^2}{2}\right) \\
g_3(y) &= y^3
\end{aligned}
\tag{4.6}
$$

Algorithm 5 Gradient Algorithm

1. Center the observed data to achieve zero mean
2. Whiten the centered data
3. Create the initial random vector $\mathbf{w}$ and the initial value for γ
4. Update

$$\mathbf{w} = \gamma \mathbf{y}_{\text{white}} g(\mathbf{w}^T \mathbf{y}_{\text{white}})$$

5. Normalise $\mathbf{w}$

$$\mathbf{w} = \frac{\mathbf{w}}{\|\mathbf{w}\|}$$

6. Check sign of γ, if not a known prior, update

$$\gamma = (G(\mathbf{w}^T \mathbf{y}_{\text{white}}) - \mathbb{E}[G(\nu)]) - \gamma$$

7. Repeat until convergence
8. Independent components are found as $\mathbf{x} = \mathbf{w}\mathbf{y}_{\text{white}}$

Fixed-Point Algorithm with Negentropy As described in the section with kurtosis, the fixed-point algorithm removed the learning parameter and compute $\mathbf{w}$ directly:

$$\mathbf{w} = \mathbb{E}[\mathbf{z} g(\mathbf{w}^T \mathbf{z})]$$

.

Write the expression from this equation to the on in the algorithm

Algorithm 6 Fixed-Point Algorithm with Negentropy (FastICA)

1. Center the observed data to achieve zero mean
2. Whiten the centered data
3. Create the initial random vector $\mathbf{w}$
4. Update

$$\mathbf{w} = \mathbb{E}[\mathbf{y}_{\text{white}} g(\mathbf{w}^T \mathbf{y}_{\text{white}})] - \mathbb{E}[g'(\mathbf{w}^T \mathbf{y}_{\text{white}})]\mathbf{w}$$

5. Normalise $\mathbf{w}$

$$\mathbf{w} = \frac{\mathbf{w}}{\|\mathbf{w}\|}$$

6. Repeat from 4. until convergence
7. Independent components are found as $\mathbf{x} = \mathbf{w}\mathbf{y}_{\text{white}}$

4.1 MSB

...

Bibliography

[1] Alickovic, Emina et al. “A Tutorial on Auditory Attention Identification Methods”. In: *Front. Neurosci* 13:153 (2019).

[2] Balkan, Ozgur, Kreutz-Delgado, Kenneth, and Makeig, Scott. “Covariance-Domain Dictionary Learning for Overcomplete EEG Source Identification”. In: *ArXiv* (2015).

[3] 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).

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

[5] 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).

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

[7] Delorme, Arnaud et al. “Blind separation of auditory event-related brain responses intoindependent components”. In: *PLoS ONE* 7(2) (2012).

[8] Foucart, Simon and Rauhut, Hoger. *A Mathematical Introduction to Compressive Sensing.* Birkher Basel, 2013.

[9] Friston, Karl J. “Functional and Effective Connectivity: A Review”. In: *BRAIN CONNECTIVITY* 1 (2011).

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

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

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

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

[14] 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).

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

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

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

[18] *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.

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

[20] 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).

Appendix A

Appendix A