Domain Adaptation for Attention Steering

Johanna Wilroth

LUND UNIVERSITY

Department of Automatic Control

MSc Thesis
TFRT-6110
ISSN 0280-5316

Department of Automatic Control
Lund University
Box 118
SE-221 00 LUND
Sweden

Printed in Sweden by Tryckeriet i E-huset
Lund 2020

Abstract

A major problem in the development of intelligent hearing aids is often referred to as the *cocktail party problem*. It describes the remarkable ability of the human brain of filter out unwanted sounds in a noisy environment, while focusing on a single talker or conversation. Without the ability to select and enhance a specific sound source of choice while suppressing the background, the hearing aids generally amplify the volume of everyone in the environment. The problem of knowing which speaker to enhance is unsolved and most people with hearing aids still experience discomfort in noisy environments. This thesis uses EEG data from real-life scenarios where the subjects for each trial listened to one female voice and one male voice at the same time while giving attention to one of the speech streams. The stories were simulated to come from a distance of 2.4m in a direction of ±60° from the listener. Due to both instrumental and human factors, data from different subjects will differ and it is not possible to create a classifier which works on all data. It is said that the data from each subject lives in different domains, and they need to be transported to the same domain in order to be classified together. The transportation is called domain adaptation, and this thesis have used and compared two domain adaptation methods: *Parallel transport* and *Optimal transport*. Two different classification problems are considered in this thesis: attention to male voice vs female voice and attention to left side vs right side. The classification accuracy differed greatly depending on which data was used. Generally, the results were better for male/female separation which almost always gave successful results, and the highest classification accuracy reached 95%. Transportation of several subjects for the left/right separation problem did not give results above the level of chance, however the best classification accuracy reached above 93% which is considered a successful result.

Acknowledgements

I would like to give a huge thanks to my main supervisor Carolina Bergeling and assistant supervisor Frida Heskebeck at Lund University. Their time, support and guidance with everything from coding and selection of interesting methods to new perspectives of the problem and relevant comments on this report have been invaluable.

Through the collaboration with Eriksholm Research Centre, I would like to thank my supervisor Emina Alickovic who has shared her expertise within the research of attention steering. She gave me directions when I got lost in the problem formulation and helped me to interpretend the results.

My examinator Bo Bernhardsson at Lund University deserves to be thanked for all the valuable discussions about new perspectives and inputs on the written report. Thank you Olle Kjellqvist at Lund University who has been part in my weekly update meetings and has given me both advice and encouragement. I am also very grateful for the insight in the dataset provided by Pex Tufvesson, which completely got me back on track and contributed to an understanding of the results. All my gym discussions with Nils Vreman helped me structure my work and I thank you for your encouragement. I would like to thank Mikael Johansson and Casper Karrén who, although late in the work process, provided valuable insight in the brain functions of auditory processing and EEG analysis.

To my parents Mikael and Helena Wilroth, to my siblings Magdalena and Markus; thank you for all your love and support throughout my whole life. Without you, I would not be who I am today and I am forever grateful for everything you have done for me.

Contents

1

Introduction

Imagine that you are standing in the middle of a room at a party. The mingle has just started and everywhere around you people are grouped together, chattering happily with a drink in their hand. Background music is playing from the speakers and the light is dim. The environment is excited, but noisy, and it is hard to distinguish what the groups are talking about over the buzz. Your friend next to you says something and you snap out of your daydreaming and focus on him. Immediately, the background impressions are damped and you hear him perfectly.

The scenario above is often referred to as the *cocktail party problem*. It describes the remarkable ability of the human brain of filter out unwanted sounds in a noisy environment, while focusing on a single talker or conversation [Alickovic et al., 2019]. For a normal hearing person, this is done several times during the day without any special thought or effort: in the lunch room at work, on the subway station or in the grocery store. Sometimes it is even done consciously, when pretending to listen to your friend next to you while actually eavesdropping on the conversation at the nearby table.

Even though the suppressing and enhancing of different sound sources happens instantly and with impressive accuracy in the brain, the mathematical model would be extremely advanced. This is a major problem in hearing aids, where the aim is to boost the volume for people with hearing loss. Without the ability to select and enhance a specific sound source of choice while suppressing the background, the hearing aids generally amplify the volume of everyone in the environment. Over the last few years, intelligent hearing aids have entered the market which are better at suppressing background noises significantly different from speech. However, the problem of knowing which speaker to enhance is unsolved and most people with hearing aids still experience discomfort in noisy environments [Han et al., 2019]. Scientists hope that the key to the *cocktail party problem* in hearing aids is to understand how the brain distinguishes the difference between attended and unattended sound sources, a process known as *auditory attention decoding (AAD)*. After decades of research, there has recently been a major breakthrough [O'Sullivan et al., 2015].

It was discovered that the cortical activity register the speech amplitude and that there is a significant difference of the speech representation if the sound source was attended or unattended [O'Sullivan et al., 2015]. The cortical activity, commonly known as *brain waves*, are actually electrical pulses from the communication between neurons. These signals can be detected by several different methods, such as Magnetoencephalography, Computed Tomography and Magnetic Resonance Imaging [O'Sullivan et al., 2015; Satheesh Kumar and Bhuvaneswari, 2012]. However, the many advantages of Electroencephalography (EEG) made researchers ask if it could be used for auditory attention decoding. EEG is a cheap and widely available technique where the signals are picked up by several small electrodes placed on the head. Unlike some of the other methods, an EEG scanning is able to capture both the radial and the tangential components of the signal, which makes the method effective and accurate. There are however some disadvantages with EEG, mainly that it has limited spatial resolution, but it is still a widely used technique around the world. In 2015, J. O'Sullivan et al. published their article *Attentional Selection in a Cocktail Party Environment Can Be Decoded from Single-Trial EEG* where they showed for the first time that it is possible to use EEG for auditory attention decoding. Ever since, the use of EEG scanning in this field has exploded and it has become an important aid in the research for the solution to the *cocktail party problem* in hearing aids.

A common approach of distinguishing between attended and unattended sound sources in the EEG data is through machine learning (ML) and classification. Given a training set, the ML algorithm first learns how to classify the data and the algorithm can thereafter be used on new data, as in real life conversations. However, many problems remain before this technique can be fully implemented into hearing aids. One major issue is that the EEG scannings are made for different subjects. This means that the EEG-measurements will differ due to instrument imperfectness or human factors, and a classification made for subject A might not work on subject B. Hence, the classification algorithm would need to be constructed from scratch for each new subject, which is time-consuming and not possible in real-time situations. This has raised the question: is it possible to find a way of constructing one classification algorithm which works for several subjects?

This thesis focuses on *domain adaptation*, a specific field in machine learning where the source data distribution (subject A) is different from the target data distribution (subject B) [Weiss et al., 2016]. The objective is to investigate if domain adaptation can be used on EEG data for the purpose of creating one classification algorithm that works on multiple data sources. Two different domain adaptation methods are used: *parallel transport (PT)* and *optimal transport (OT)*, which both use covariance matrices on the Riemannian manifold. Covariance matrices are powerful tools when working with time series and their properties are preserved through the transportation.

The EEG-dataset used in the thesis, further explained in Chapter 3, reflects complex, real life situations. Through 60 trials, each subject got to listen to two different recorded fictional stories at the same time with one female and one male voice. The stories were simulated to come from a distance of 2.4m in a direction of ±60° from the listener. In each trial, the subject was asked to give attention to one of the speakers while suppressing the other and EEG-data was collected from 64 electrodes placed on the head of the subject. Two different classification problems are considered:

- Attention to male voice vs female voice
- Attention to left side vs right side

The goal of the thesis is to use the two domain adaptation methods *parallel transport* and *optimal transport* to transport data from several subjects and thereafter create a classifier which gets an accuracy above the level of chance.

The outline of the thesis is: Transfer learning and domain adaptation are defined in Chapter 2. The dataset and preprocessing steps are explained in Chapter 3. The brain neural mechanisms for attention steering between different sound sources and location is briefly explained in Chapter 4 and Chapter 5 contains a short pipeline of the structure of the Matlab script. The different classification methods and statistically significant classification performance are explained in Chapter 6 and the visualization technique t-SNE is explained in Chapter 7. The two transportation methods use Riemannian geometry, which is defined in Chapter 8. The two domain adaptation methods Parallel transport and Optimal transport are explained in Chapters 9 and 10 respectively. The results are presented in Chapter 11 and they are discussed in Chapter 12. Lastly, Chapter 13 sums up the thesis with some conclusions.

2

Learning to learn in Machine Learning

2.1 Introduction

The ability of *learning to learn* is vital to survival for all humans and animals. Babies do not have to relearn every possible motor skill in each new environment since they learn how to generalize and adapt the skill in different situations. For each new motor skill learned, the biological cognitive system grows and it gets easier to learn a new one [Patricia and Caputo, 2014].

Recently, the concept of *learning to learn* has been applied in Machine Learning (ML) - a subset of artificial intelligence where the goal is to get computer systems to learn for themselves from provided data [*What is Machine Learning? A definition* 2020]. By observations, experience or instructions, the ML-algorithm is able to discover patterns in the data and the skill is learned if the *training* set is sufficiently large. A different *testing* set is used to evaluate how well the ML-algorithm works on similar data [*How to Build A Data Set For Your Machine Learning Project* 2020]. However, in real-life the learned skill needs to function in many different situations. The testing data may not be similar to the training data and in these cases, the ML-algorithm needs to be generalized. This is when *learning to learn*-methods are used.

One of the most common *learning to learn*-methods is called *transfer* *learning*, and a specific case of transfer learning used in this thesis is called *domain adaptation* [Patricia and Caputo, 2014]. They will both be explained in more detailed after a few definitions.

2.2 Definitions and notations

The objective of classification is to predict a class label $y_i \in \mathscr{Y}$ to a feature vector $\mathrm{x}_i \in \mathrm{X}$. For example; if 'feathers', 'claws' and 'beak' are features of an image *i*, the class label from the ML-algorithm should be 'bird'. X is a particular learning sample with *n* number of feature vectors and $\mathscr{Y}$ is the label space with all possible labels. The feature space $\mathscr{X}$ contains all possible feature vectors [Weiss et al., 2016].

In machine learning, a domain $\mathscr{D}$ is defined as $\mathscr{D} = \{\mathscr{X}, \mathrm{P(X)}\}$ where $\mathscr{X}$ is the feature space and P(X) is the marginal probability distribution with the learning sample $\mathrm{X} = \{\mathrm{x}_1, ..., \mathrm{x}_n\} \in \mathscr{X}$. A task $\mathscr{T}$ in a given domain $\mathscr{D}$ is defined as $\mathscr{T} = \{\mathscr{Y}, \mathrm{P(Y|X)}\}$ where $\mathscr{Y}$ is the label space, Y is the label sample and P(Y|X) is the conditional distribution [Weiss et al., 2016].

The training data D_S comes from a *source domain* $\mathscr{D}_S = \{\mathscr{X}_S, \mathrm{P}(\mathrm{X}_S)\}$ and is defined as $D_S = \{(\mathrm{x}_{S1}, \mathrm{y}_{S1}), ..., (\mathrm{x}_{Sn}, \mathrm{y}_{Sn})\}$ where $\mathrm{x}_{Si} \in \mathscr{X}_S$ and $\mathrm{y}_{Si} \in \mathscr{Y}_S$. The testing data D_T, as well as all other data used after the training, comes from a *target domain* $\mathscr{D}_T = \{\mathscr{X}_T, \mathrm{P}(\mathrm{X}_T)\}$ and is defined as $\mathscr{D}_T = \{(\mathrm{x}_{T1}, \mathrm{y}_{T1}), ..., (\mathrm{x}_{Tn}, \mathrm{y}_{Tn})\}$ where $\mathrm{x}_{Ti} \in \mathscr{X}_T$ and $\mathrm{y}_{Ti} \in \mathscr{Y}_T$ [Ben-David et al., 2010]. The source task and conditional distribution are denoted $\mathscr{T}_S$ and $\mathrm{P}(\mathrm{Y}_S|\mathrm{X}_S)$ meanwhile the target task and conditional distribution are denoted $\mathscr{T}_T$ and $\mathrm{P}(\mathrm{Y}_T|\mathrm{X}_T)$ respectively. All notations are presented in Table 2.1 [Weiss et al., 2016] and Figure 2.1 shows an illustration of the differences between traditional machine learning and transfer learning.

2.3 Transfer Learning and Domain Adaptation

In classical machine learning, the domains and tasks of the source and target are the same, hence $\mathscr{D}_S = \mathscr{D}_T$ and $\mathscr{T}_S = \mathscr{T}_T$. This is the same as fulfilling the conditions:

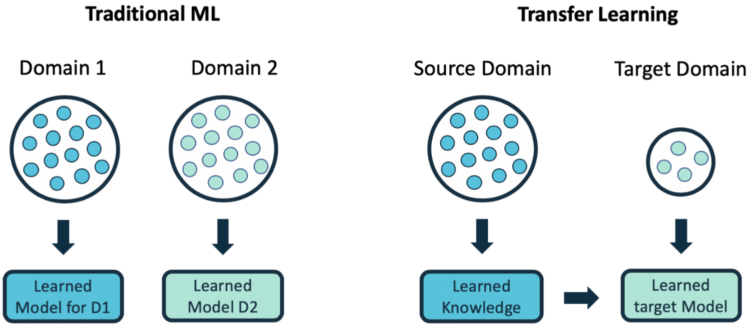


Figure 2.1: The left figure shows an example of traditional machine learning where a classifier is created for each domain. In this thesis, the blue datapoints in domain 1 would represent all trials for subject 1 and the green datapoints in domain 2 would represent all trials for subject 2. The classifier created for domain 1 will probably not work on the datapoints in domain 2. The right figure shows an example of transfer learning where learned knowledge from the source domain (subject 1) is transferred to the classifier of the target domain (subject 2) [Asgarian, 2020].

$$\begin{aligned} \mathscr{X}_S &= \mathscr{X}_T && \text{Same feature space} \\ \mathscr{Y}_S &= \mathscr{Y}_T && \text{Same label space} \\ \mathrm{P}(\mathrm{X}_S) &= \mathrm{P}(\mathrm{X}_T) && \text{Same marginal distribution} \\ \mathrm{P}(\mathrm{Y}_S|\mathrm{X}_S) &= \mathrm{P}(\mathrm{Y}_T|\mathrm{X}_T) && \text{Same conditional distribution} \end{aligned}$$

When one or more of these conditions are not satisfied, generalization methods built on the *learning to learn*-principle need to be used. This is most commonly referred to as *transfer learning*. The exact definition of transfer learning varies between researchers and papers, but the one used in this thesis comes from Weiss et al. in the article *A survey of transfer learning*:

DEFINITION 2.3.1 "Given a source domain $\mathscr{D}_S$ with corresponding source task $\mathscr{T}_S$ and a target domain $\mathscr{D}_T$ with a corresponding task $\mathscr{T}_T$, transfer learning is the process of improving the target conditional distribu-

Table 2.1: Notations used in this chapter [Weiss et al., 2016].

Notation	Description
$\mathscr{X}$	Input feature space
$\mathscr{Y}$	Label space
$\mathscr{T}$	Predictive learning task
Subscript S	Denotes source
Subscript T	Denotes target
$\mathscr{D}_S$	Source domain
$\mathscr{D}_T$	Target domain
D_S	Source domain data
D_T	Target domain data
P(X)	Marginal distribution
P(Y|X)	Conditional distribution
X	Particular learning sample
x_i	Feature vector *i*
y_i	Class label *i*

tion $\mathrm{P}(\mathrm{Y}_T|\mathrm{X}_T)$ by using the related information from $\mathscr{D}_S$ and $\mathscr{T}_S$, where $\mathscr{D}_S \neq \mathscr{D}_T$ and/or $\mathscr{T}_S \neq \mathscr{T}_T$." □

The specific case when $\mathscr{X}_S = \mathscr{X}_T$, $\mathscr{Y}_S = \mathscr{Y}_T$ and the mismatch between the source and target only comes from the probability distributions is called *domain adaptation*, and this is the case studied in this thesis [Kouw and Loog, 2018]. The definition of domain adaptation presented by Weiss et al. is:

DEFINITION 2.3.2 "Given a source feature space $\mathscr{X}_S$ with corresponding source label space $\mathscr{Y}_S$ and a target feature space $\mathscr{X}_T$ with corresponding target label space $\mathscr{Y}_T$, domain adaptation is the specific case of transfer learning when $\mathscr{X}_S$ = $\mathscr{X}_T$, $\mathscr{Y}_S$ = $\mathscr{Y}_T$ and the mismatch between source and target comes from $\mathrm{P}(\mathrm{X}_S) \neq \mathrm{P}(\mathrm{X}_T)$ and/or $\mathrm{P}(\mathrm{Y}_S|\mathrm{X}_S) \neq \mathrm{P}(\mathrm{Y}_T|\mathrm{X}_T)$." □

After the transportation, the marginal and conditional distributions are merged into the joint distribution P(X,Y)=P(Y|X)P(X). In domain adaptation, this joint distribution can be broken down to two different cases [Kouw and Loog, 2019]:

$$\begin{array}{llll} \mathrm{P}(\mathrm{X}_S) = \mathrm{P}(\mathrm{X}_T) & \& & \mathrm{P}(\mathrm{Y}_S|\mathrm{X}_S) \neq \mathrm{P}(\mathrm{Y}_T|\mathrm{X}_T) & \text{Concept shift} \\ \mathrm{P}(\mathrm{X}_S) \neq \mathrm{P}(\mathrm{X}_T) & \& & \mathrm{P}(\mathrm{Y}_S|\mathrm{X}_S) = \mathrm{P}(\mathrm{Y}_T|\mathrm{X}_T) & \text{Covariate shift} \end{array}$$

It also exists a third case, when the joint distribution is presented as P(X,Y)=P(X|Y)P(Y) and the mismatch is in the marginal (prior probability) distribution of Y [Kouw and Loog, 2019]:

$$\mathrm{P}(\mathrm{Y}_S) \neq \mathrm{P}(\mathrm{Y}_T) \quad \& \quad \mathrm{P}(\mathrm{X}_S|\mathrm{Y}_S) = \mathrm{P}(\mathrm{X}_T|\mathrm{Y}_T) \quad \text{Prior shift}$$

Concept shift

A concept shift is sometimes referred to as a data shift where the conditional distribution of the learning sample X is different between the source and target domains. An example could be when developing a flu prognosis of a specific patient based on features such as age, general health, socio-economic status and severity of the flu. Lets assume that the doctor meets and examines the patient four times, where the first three times are used for training a model and the fourth is used for testing the model. Two classes are considered: "remission" and "complications". A concept shift would occur if the aspects defining what would belong to "remission" and "complications" differ between training and testing. For example, in training the only aspects of "severity of the flu" is the fever level, however in testing the doctor realised that the level of nausea also needs to be considered and adds it to the "complications" class [Kouw and Loog, 2019].

Figure 2.2 shows an illustration of a concept shift.

Covariate shift

A covariate shift is sometimes referred to as a data shift where the marginal distribution of the learning sample X is different between the source and

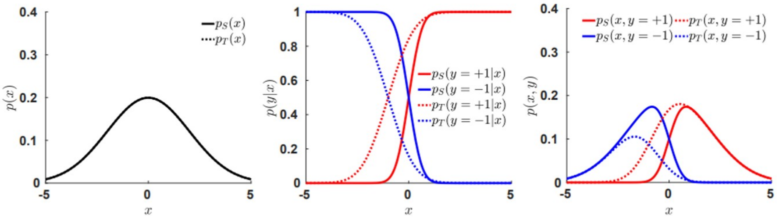


Figure 2.2: An illustration of a concept shift. The left figure shows equal prior probability distributions for source and target domains. In the middle figure, the conditional distributions of the target domain is shifted to the left of the conditional distributions of the source domain. The red and blue colours represent two different classes *y*. This gives a shifted joint distributions (right figure) [Kouw and Loog, 2019].

target domains. This is a very common case and several studies have been made on this scenario. It often occurs when there is a selection bias, hence when the selected data does not represent the whole picture [Tran and Aussem, 2015].

In the medical example above, a covariate shift would occur if the model of the flu prognosis was developed when the patient was a child, and the same model was used when the patient was an adult or elderly. Unlike a concept shift, the same aspects of the two classes and the same features are used, however the age in the features are different.

Covararite shifts are closely related to when the training and testing datasets are produced. The example above with several years between creating the model and using the model is an extreme situation, and covariate shifts do also occur between different trials and sessions even though they are made during the same day. It is also very common with covariate shifts in non-stationary time series [Raza et al., 2016].

Figure 2.3 shows an illustration of a covariate shift.

Prior shift

Prior probability shift is when only the distribution over Y changes and everything else stays the same, which can be seen as shift in the target la-

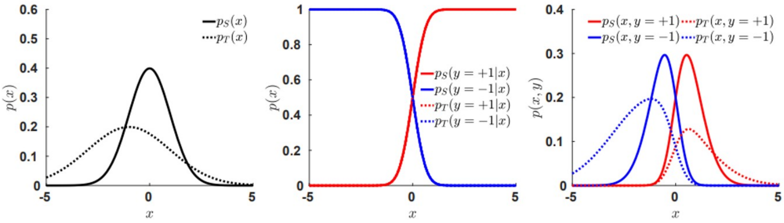


Figure 2.3: An illustration of a covariate shift. The left figure shows that the target prior proabability distribution is shifted compared to the source prior proabability distribution. In the middle figure, the conditional distributions are equal for the source and target domain. The red and blue colours represent two different classes y. This gives a shifted joint distributions (right figure) [Kouw and Loog, 2019].

bels [Kouw and Loog, 2019]. In the medical example, this could be that the same features (age, general health, socio-economic status and severity of the flu) and the aspects of the two class labels "remission" and "complications" are the same. However, how the aspects are weighted are different between training and testing. For example, consider the aspect *level of fever* of the feature *severity of the flu*. In training, a temperature above 37C°was included in "complications" but in testing, this temperature level was changed to 37.2C°.

Figure 2.4 shows an illustration of a prior shift.

2.4 Domain adaptation on EEG data

The electrical pulses in the brain can be picked up by EEG electrodes and are represented as time series. These time series are usually nonstationary due to electrode placements, changes in attention levels, blinking or other motor movements and environmental factors. This almost always gives covariate shifts in the EEG signals when comparing trial-to-trial or session-to-session [Razaa et al., 2019]. However, when comparing subject-to-subject (the procedure in this thesis), anatomic differences between individuals could give differences also in the conditional distribution, resulting in concept shifts [Albuquerque et al., 2019].

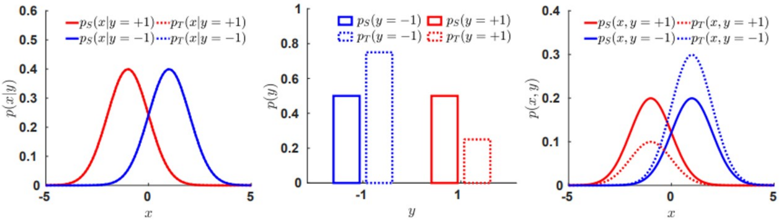


Figure 2.4: An illustration of a prior shift for the two classes "red" and "blue". The left figure shows that the conditional distributions, for each class, is the same in the source and target domain. The conditional distributions are however different between the classes. The middle figure shows that the prior probabilities for both classes in the source domain are 1/2, however they are to 2/3 and 1/3 respectively in the target domain. This gives a shifted joint distribution (right figure) [Kouw and Loog, 2019].

3

DTU Dataset

3.1 Introduction

The dataset used in this thesis was presented in the article *Noise-robust cortical tracking of attended speech in real-world acoustic* in 2017 by Søren Asp Fuglsang *et al*. The speech material for the experiments was recorded at the Technical University of Denmark (DTU), and will be referred to as the DTU-dataset. The dataset was in March 2018 made public and can be downloaded at *zenodo.org* [Fuglsang et al., 2018].

3.2 Procedure

Each subject got to listen to two different speech streams at the same time, one in the left ear and one in the right ear, while being asked to give attention to one of them with minimized motor activity. The subjects listened to the speech streams in a soundproof, electrically-shielded booth with ER-2 insert earphones (Etymotic Research). Data from 64 scalp electrodes and two mastoid electrodes was collected from 60 trials at a sample rate of 512Hz for each subject. Data from the two mastoid electrodes were removed in the script. Figure 3.1 shows the placement of the electrodes on the head of the subject. After each trial, the subject answered multiple-choice questions about the stories to verify that the subject had attention to the correct story.

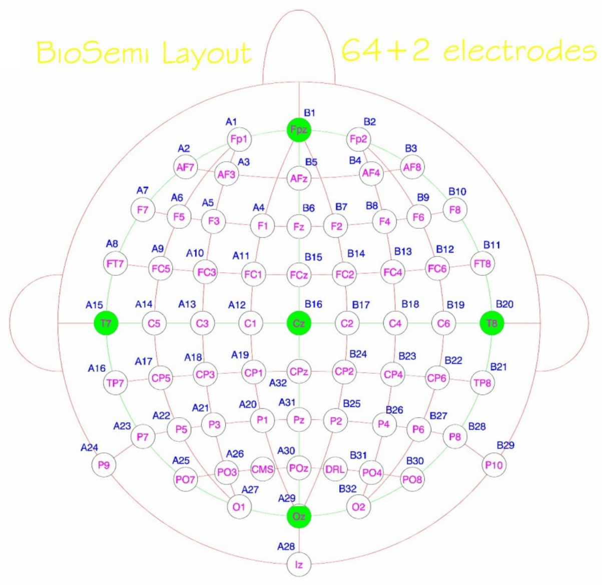


Figure 3.1: The placement of the 64 scalp electrodes and two mastoid electrodes on the head of the user [*BioSemi headcap* 2020].

3.3 Participants

Data was recorded from 19 subjects between 19 and 30 years old without hearing problems and with no reported neurological disorders. One subject was excluded due to missing data in several trials and the remaining data is public and available at *zenodo.org*.

3.4 Speech stream material

The speech streams were recorded in an anechoic chamber at DTU by one female and one male professional storyteller at a sample rate of 44100Hz. The sound pressure level (SPL) for the speech streams were 65dB and they were normalized to have similar root-mean square values. The two hours audio recordings for each storyteller were divided into 50-seconds sections used for the trials.

To reflect real life situations some of these recordings were simulated in a mildly reverberant room and some in a highly reverberant room by the room acoustic modeling software Odeon. Hence, there are three environment scenarios:

1. Anechoic
2. Mildly reverberant
3. Highly reverberant

3.5 Preprocessing of the EEG and audio data

Both the EEG data and audio data were preprocessed through the Fieldtrip and COCOHA toolboxes in Matlab. The script *preproc_data.m* which is used for the preprocessing can be downloaded from *zenodo.org*.

The preprocessing scripts include these steps [Fuglsang et al., 2018]:

1. Filter out harmonics and 50Hz line noise in the EEG data
2. Downsample the EEG data to 64Hz
3. Minimize filter startup artifacts by a 1st order detrend
4. Highpass the EEG data at 0.1Hz by a 4th order forward-pass Butterworth filter
5. Denoising
6. Select events corresponding to attended talker
7. Split continuous data into trials
8. Split data into cells
9. Add attended and unattended audio and extract envelopes

10. Extract the envelopes and downsampling of the audio signal - further described below

11. Remove single-talker trials with no unattended talker

12. Trim trials to be the same length

13. Append data cells as trials

14. Save data

In scenarios with several talkers, researchers have noticed an increase in oscillatory alpha (frequency band 7-14Hz) power due to the brain activity of ignoring the unattended speakers [Paul et al., 2020]. Therefore, an additional Butterworth bandpass filter of order 6 with frequency band $[8-12]$Hz was applied to both the EEG data and the audio data.

Extract the envelopes of the audio signal

It is very common to extract the *envelopes* of oscillating signals, such as speech audio, when working with audio features. The envelope outlines the amplitudes of the oscillating signal into a smooth curve, illustrated in Figure 3.2. The envelope extraction is made with a Hilbert transform through a 31-band gammatone filterbank with a frequency range 80-8000Hz. The absolute value of the signals were computed and thereafter raised to the power of 0.3, to mimic the human auditory system. The filterbank signal outputs were summarised across the channels, giving the final envelopes. These audio envelopes were downsampled to 64Hz and aligned in time with the EEG data by the start-triggers stored in the EEG data struct. Finally, the envelopes were lowpassed at 9Hz, centered and Z-normalized across the time dimension [Wong et al., 2018].

Only the anechoic signals were used for the envelope extractions. For the trials with simulated reverberant, the envelopes were derived from their underlying clean signals.

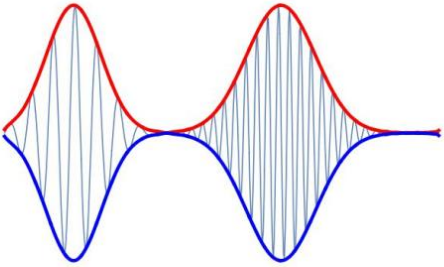


Figure 3.2: The upper and lower envelopes of an oscillating signal [*Waves Packed in Envelopes* 2020].

Visalization of the data

Figure 3.3 shows the t-SNE visualization for subjects 2 and 7 to illustrate that the datapoints from each subject are separated. Each datapoint represents one trial where the colour shows which voice (male vs female) and which direction (left vs right) the attention was to in that particular trial. The two black lines are an illustration of a linear classification model between attention to the male voice and the female voice. The figure clearly shows that the classifier created for subject 7 would not work well for the data from subject 2. The red dashed line shows an illustration of a linear classification model between attention to the male voice and the female voice based on all data. Also in this case, the classification accuracy would not be very good since both sides of the line contain several datapoints with both attention to the male and female voice. This highlights the problem formulation in domain adaptation. It is said that the datapoints from each subject live in their own domain. A classifier created in one domain usually does not work for data in another domain. The solution is to transport the data to the same domain, which in a t-SNE visualization means that the datapoints are merged together, while keeping as much of their structure as possible.

Figures 3.4-3.6 show the t-SNE visualization of all subjects. It seems to be possible to distinguish between attended male (M) and attended female (F) voices easier than if the attended sound source is to the left (L) or right (R) side of the listener. This separation is clearer in specific subjects such as number 2, 3, 7, 10 and 15.

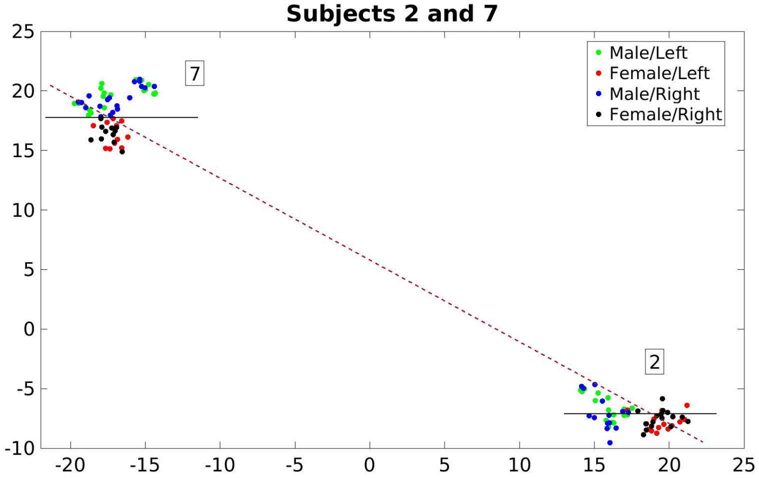


Figure 3.3: t-SNE visualisation of subjects 2 and 7 with attention to: ML = Male/Left, FL = Female/Left, MR = Male/Right and FR = Female/Right. The figure illustrates that a linear classifier between attention to male and female voices created on subject 7 (black line) would not work well on data from subject 2. The red dashed line shows that a linear classifier on data from both subjects would not work well without the use of domain adaptation.

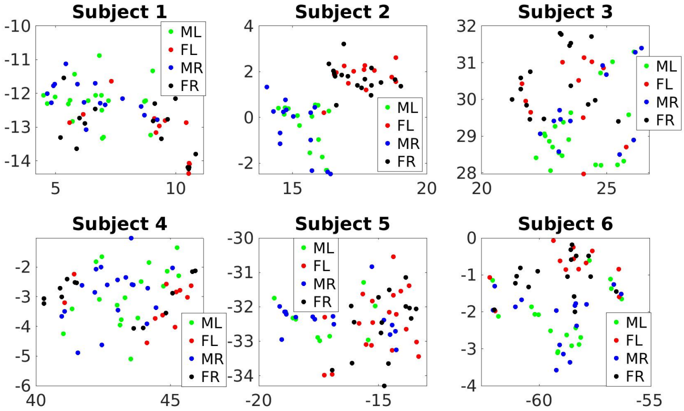


Figure 3.4: t-SNE visualisation of subjects 1-6 with attention to: ML = Male/Left, FL = Female/Left, MR = Male/Right and FR = Female/Right.

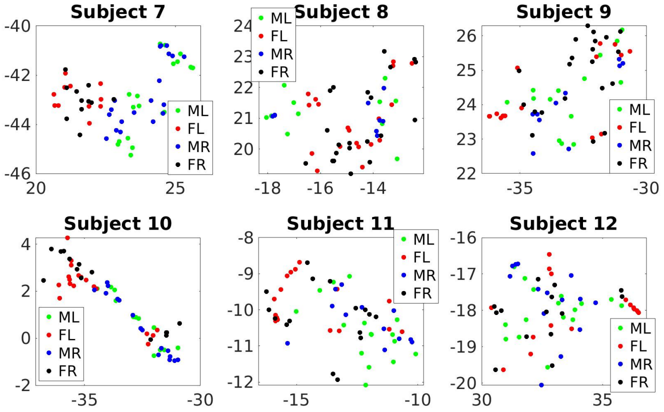


Figure 3.5: t-SNE visualisation of subjects 7-12 with attention to: ML = Male/Left, FL = Female/Left, MR = Male/Right and FR = Female/Right.

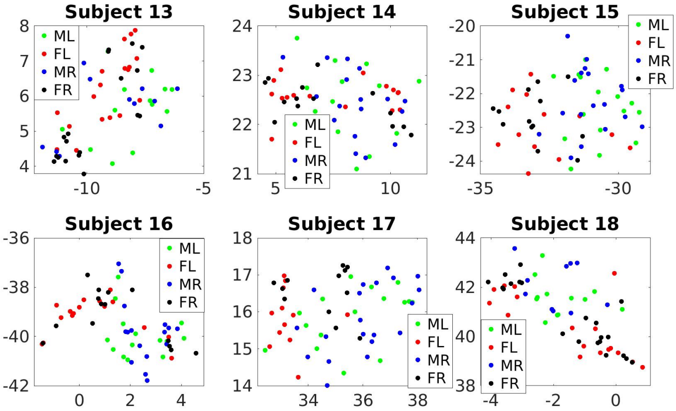


Figure 3.6: t-SNE visualisation of subjects 13-18 with attention to: ML = Male/Left, FL = Female/Left, MR = Male/Right and FR = Female/Right.

4

Attention steering in the brain

4.1 How the brain distinguishes different sound sources

The human brain uses three different characteristics to distinguishing between sound sources: pitch, loudness and quality. In speeches, the vocal cords produce vibrations which are detected by the ears. The number of vibrations during a certain time period is called the *pitch*. These vibrations are visible as oscillations of the speech signals, where few oscillations is a feature of low pitches mostly common in male voices. Correspondingly, speech streams with many oscillations is a feature of high pitches mostly common in female voices. Figure 4.1 shows the signals before any preprocessing of one female and one male speech stream from the DTU-dataset.

The definition of *loudness* is "the attribute of a sound that determines the magnitude of the auditory sensation produced and that primarily depends on the amplitude of the sound wave involved" [Merriam-Webster, 2020] and it is measured in decibels (dB). The loudness is determined by the SPL, frequency content and duration of the sound.

The quality of the sound source is often referred to as the *timbre* and it describes the characteristics of a voice, such as thin, bright, harsh or dark. Timbre is the reason why the same tone on a guitar and a piano sounds different. It is determined by the harmonic content, vibrato and the attack-decay envelope of the sound. The harmonic content gives the waveform shape of the signal in the time-domain. It is easier to distinguish the harmonic content in the frequency domain which is given by the Fourier transform of the time-domain signal, see Figure 4.2. The figure shows that the amplitude shapes of the female and male speech streams seem quite similar, however the female amplitude is slightly greater. The vibrato describes periodic changes in the pitch. The attack-decay envelope, on the other hand, describes the shape of the rise and decay of the amplitude [Risset and Wessel, 1982].

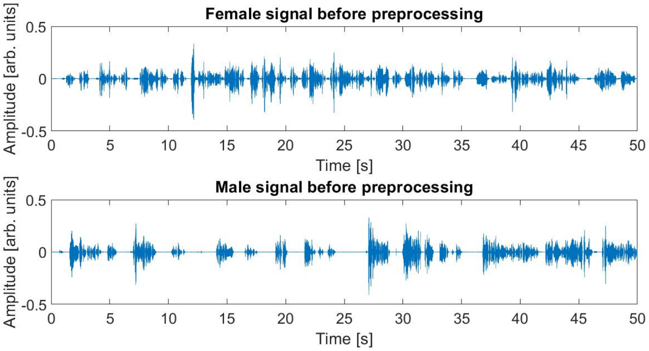


Figure 4.1: The signals before preprocessing of one female speech stream and one male speech stream in the time-domain.

Figures 4.3 and 4.4 show the signals of one female and one male speech stream after the preprocessing steps in time- respectively frequency domain. The different pitches and attack-decay envelopes of the female and male sound is more apparent in Figure 4.3 comparing to the raw data presented in Figure 4.1.

4.2 How the brain distinguishes the sound source location

The subjects listened to the speech streams binaurally through ER-2 insert earphones. The software Odean was also used to simulate that the two talkers are positioned at a distance of 2.4m, at an angle of ±60° along the

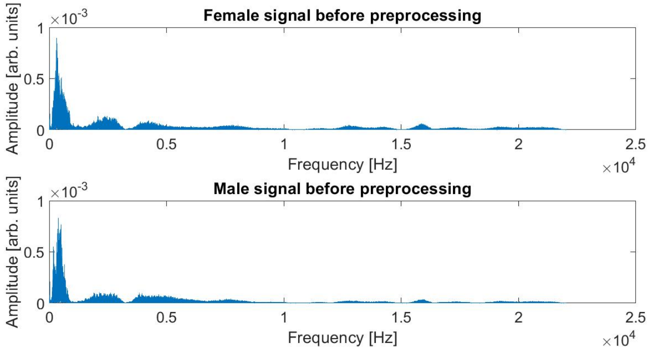


Figure 4.2: The signals before preprocessing of one female speech stream and one male speech stream in the frequency domain which is used to determine the harmonic content of a sound.

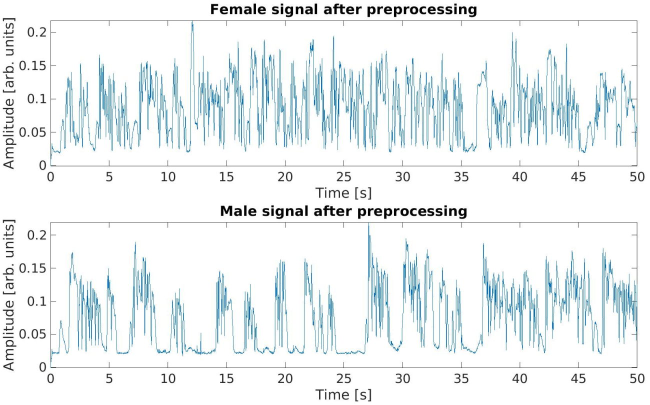


Figure 4.3: The envelopes after preprocessing of one female speech stream and one male speech stream in the time-domain.

azimuth direction from the listener, shown in Figure 4.5. These simulation algorithms ensure that the brain still perceives that each speech stream from each ear plug reaches both ears, just like in real-life situations.

In real life situations, the brain easily recognizes the location of a sound source. For example, hearing a car driving by, you would instinctively

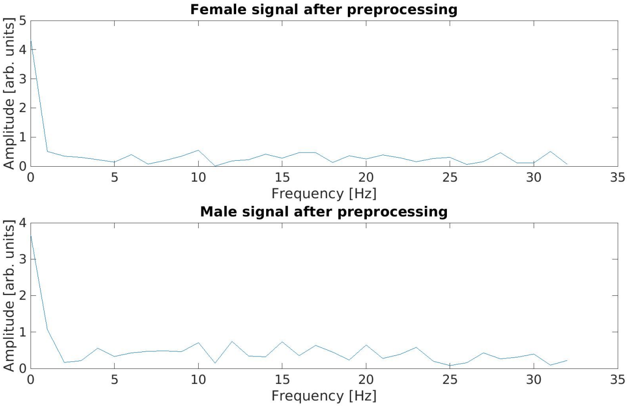


Figure 4.4: The envelopes after preprocessing of one female speech stream and one male speech stream in the frequency domain.

know if it is on your left or on your right side. The localization of a sound source, illustrated in Figure 4.6, gives information about [Risoud et al., 2018]:

- The azimuth angle in the horizontal plane
- The elevation in the vertical plane
- The distance to the sound source

The brain mainly uses three features to solve the problem: time-of-arrival, sound pressure level (SPL) and the spectral shape of the sound source [Risoud et al., 2018].

The azimuth angle in the horizontal plane is evaluated by the difference of the time-of-arrival and the SPL to each ear. Since the two ears are separated by the head, the time of the sound source to the ear further away is longer than to the closest ear. The head is also responsible for an acoustic shadow, which gives a difference between the SPL between the two ears [Risoud et al., 2018]. Even though the subjects used insert earphones, the simulation in the software Odeon makes it possible for the sound to reach both ears.

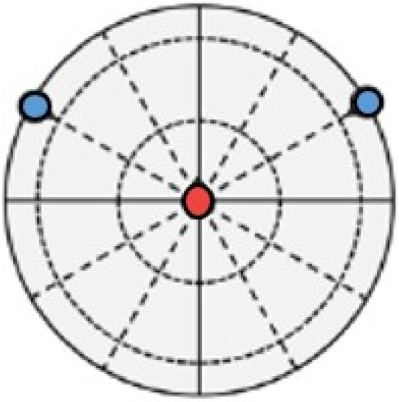


Figure 4.5: The red circle represents the listener in the center with the two talkers (blue circles) positioned at a distance of 2.4m, at an angle of ±60° along the azimuth direction from the listener [Fuglsang et al., 2017].

The other two attributes, elevation and distance, are instead determined by one ear using monaural cues, meaning that different shapes of the sound source reach the ear. Naturally the original sound source first reaches the ear, but due to reflection, diffraction and absorption from the body and environment, different shapes of the spectral also reach the ear. This is used to evaluate both the elevation in the vertical plane and the distance to the sound source [Risoud et al., 2018].

In the DTU-dataset, both the distance and the elevation in the vertical plane are the same for the two sound sources. The attribute of interest is instead the azimuth angle in the horizontal plane, or rather if the attended sound source is on the left or the right side of the listener.

The problem of attention steering towards a specific location is indeed very complex. It is common knowledge that the brain is divided into a right and a left hemisphere. The right hemisphere controls the left side of the body and is responsible for functions such as creativity, imagination, intuition, auditory and non-verbal stimuli processing such as music awareness. The left hemisphere controls the right side of the body and is responsible for functions such as analytic thought, logic, language and reasoning. This means that sounds reaching the right ear is processed in the left hemi-

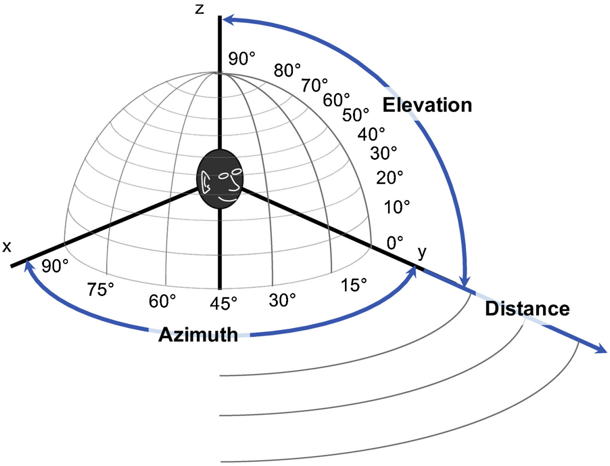


Figure 4.6: The three attributes of sound source localization [Risoud et al., 2018].

sphere and vice versa. These two hemispheres cooperate and exchange information through a communication route called the *corpus callosum* (CC) [*Right-Brain Hemisphere* 2020; *Left-Brain Hemisphere* 2020].

In a single-talker quiet environment scenario, the listener receives the speech stream to both ears, with a small time delay to the ear farthest away. The pathways of the signals are crossed to reach the opposite hemisphere, meaning that the sound stream reaching the right ear is treated in the left hemisphere and vice versa. However, the right hemisphere does not understand language and this signal is re-routed through the CC to the left hemisphere for language processing, which gives a small time delay [Steinberg and Sciarini, 2013]. This time delay of the re-routing step is one reason of the so called *right ear advantage* (REA), meaning that sounds reaching the right ear arrive to the left hemisphere first and are therefore preprocessed before the other signal. It is the same reason why the majority of the population favour using the right hand when writing or right foot when playing football. Often without knowing it, most people are also *right-eared* [Jerger and Martin, 2004].

In a noisy environment, or two talker scenario such as in the DTU-dataset, the problem is even more complex. The listener receives both speech streams to both ears, still with a time delay to the ear farthest away. All signals are crossed to reach the opposite hemisphere and the signals to the right hemisphere are re-routed through the CC to the left hemisphere with a time delay [Steinberg and Sciarini, 2013]. In noisy environments, one important component of the auditory system is the medial olivocochlear bundle (MOC). The purpose of the MOC is to enhance the signal of attention [Smith and Keil, 2015]. In the single-talker quite environment scenario described above, the MOC is inactivated since the attended sound source does not have any other noise to compete with. In this two-talker scenario without any attention steering, the MOC inhibits the left ear signal which favours the REA. This behaviour might have been essential for the human survival where the reaction to the footfalls of predators in noisy environments is the difference between life and death [Poeppel, 2003]. Figure 4.7 shows the signal pathways for a person listening to two speech streams, where the REA is illustrated by a thicker red pathway compared to the blue pathway. This is however a simplified figure and it does not illustrate real-life scenarios perfectly. In the DTU-dataset, both speech streams reach both ears, which would give a more complex illustration.

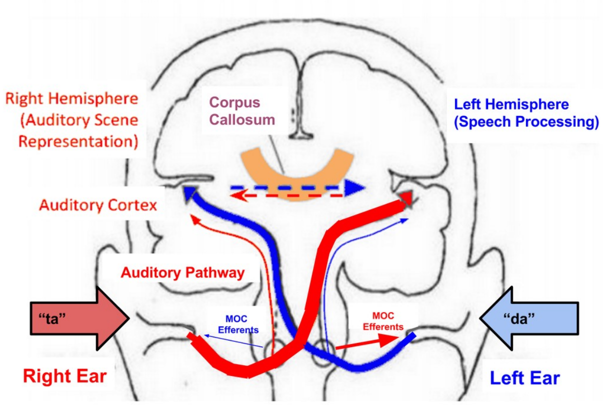


Figure 4.7: A simplified two talker scenario where the listener receives one speech stream in each ear. The signals process from the ears to the opposite hemisphere and are thereafter re-routed through the CC for speech processing and auditory scene representation. The MOC is activated by inhibiting the left ear signal, resulting in the REA represented by the thicker red pathway [*Spatial hearing loss* 2020].

Now add the problem of attention steering. Given the two-talker scenario described above, the listener now pays attention to one of the speech streams. How does this affect the signal pathways? The MOC in this case inhibits the unattended signal [Smith and Keil, 2015]. Figure 4.8 illustrates the case where attention is to the speech stream in the left ear, which is represented by the thicker blue pathway.

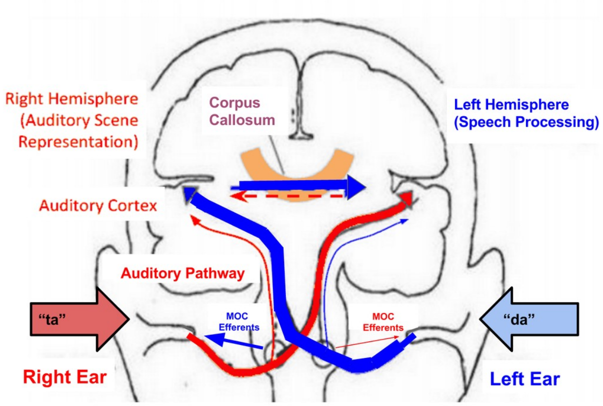


Figure 4.8: A simplified two talker scenario where the listener receives one speech stream in each ear and attention is to the left ear. The signals process from the ears to the opposite hemisphere and are thereafter re-routed through the CC for speech processing and auditory scene representation. The MOC is activated by inhibiting the right ear signal, represented by the thicker blue pathway [*Spatial hearing loss* 2020].

4.3 Attention steering with EEG

Naturally, the description above of the signal pathways is simplified and all the neural underlying mechanisms are not yet understood by researchers. The purpose of all these scenarios and left/right hemisphere comparison is to highlight the complexity of the problem of attention steering. The time delays, both from the separation of the two ears and from the re-routing between the hemispheres, are important keys which help the brain to evaluate the location of the attended sound source. These time-delays, together with all the signal crossings between left and right need be picked up by the EEG electrodes. Even though EEG scannings are quite accurate and outperforms several other techniques, they are not able to catch all the information. The limited spatial resolution means that it is hard to decide which areas of the brain that are activated [Burle et al., 2015], which might be a problem with all the cross-pathways. A Singular Value Decomposition (SVD) analysis was made on the EEG data to investigate if some of the electrodes were much more inactive and could be removed, but the results did not give any significant difference and all electrodes were kept. Several studies show that EEG data can be used for auditory attention decoding, however the problem of deciding which direction the attended sound source is coming from is more difficult.

5

Pipeline

The pipeline of the Matlab script is:

1. Load EEG data and audio data
2. Preprocess the EEG data and audio data with a Butterworth bandpass filter of order 6 between the frequencies [8 12]Hz
3. The preprocessed data includes data from two mastoid electrodes which are irrelevant in this thesis. They are in this step removed.
4. Add the two audio data files to the EEG data
5. Extract the class labels for each trial
6. Compute the covariance matrices for each subject
7. Visualize before transportation with t-SNE
8. Apply parallel transport
    - Visualize after parallel transport with t-SNE
    - Compute the classification accuracy with SVM, *k*-nearest neighbour and decision tree
    - Compute the probability that the classification accuracy is above the level of chance
9. Apply optimal transport
    - Visualize after optimal transport with t-SNE

- Compute the classification accuracy with SVM, *k*-nearest neighbour and decision tree
- Compute the probability that the classification accuracy is above the level of chance

6

Classification methods

6.1 Introduction

Data classification methods are used to categorize data from predefined classes. This thesis uses two classification problems uses a dataset with four classes:

- Attention to male voice vs female voice
- Attention to left side vs right side

Three different classification methods are used: Support Vector Machine (SVM), *k*-Nearest Neighbour (kNN) and Decision Tree.

6.2 Background

Each subject did 60 listening trials, which represent 60 datapoints, where attention was to one of the classes stated above. Cross-validation is used to compute the correct classification rate. It means that a model, also called a classifier, is created to distinguish between the predetermined classes by a training dataset. This classifier is then used to another set of datapoints (testing data), and an accuracy is computed as the correct number of predicted classes over the total number of test datapoints. In cross-validation, this procedure is repeated several times for different training and test datapoints, and a correct classification rate is computed as the mean of all the accuracies from the different classifiers [Combrissona and Jerbi, 2015].

For classification of one single subject before any transportation, one datapoint is used for testing and the other 59 datapoints for training. This procedure is repeated 60 times so that each datapoint is used for testing once. The prediction of each testing datapoint is either 1 (correct predicted class) or 0 (incorrect predicted class). A correct classification rate is at last computed as the mean of these 60 predictions.

Cross-validation is also used on multiple subjects after the transportation. This means that the number of times the classification procedure is repeated equals the total number of datapoints from all subjects. For example, transportation of two subjects (A and B) creates 120 classifiers since each subject has 60 trials.

6.3 Statistically significant classification performance

The performance of the correct classification rate is in this thesis compared to how much it differs from the *level of chance*, i.e. if the classifier would randomly select classes of the testing datapoints. The problem in this thesis contains two classes for each problem (male/female or left/right), and for an infinite number of datapoints the level of chance would be 50%. However, the number of datapoints in real experiments is never infinite and the less datapoints used, the higher risk that chance plays into the results. Therefore, the reliability of the correct classification rate depends on if it is statistically significant greater than the level of chance. Table 6.1 shows the statistically significant classification performance with different number of datapoints n for two classes with a significance level of $p = 0.05$ [Combrissona and Jerbi, 2015].

One way of determine how close the accuracy is to become statistically significant is through the binomial cumulative distribution function:

Table 6.1: Statistically significant classification performance with different number of datapoints n for two classes with a significance level of $p = 0.05$. The values are rounded to the first digit and the binomial cumulative distribution function is used to compute the threshold values [Combrissona and Jerbi, 2015].

n	Acc (%)	n	Acc (%)	n	Acc (%)
20	70,0	80	58,7	300	54,7
40	62,5	100	58,0	400	54,0
60	60,0	200	56,0	500	53,6

$$y = F\left(x|n, p_c\right) = \sum_{i=0}^{x} \binom{n}{i} p_c^i (1-p_c)^{n-i} I_{(0,1,\ldots,n)}(i) \tag{6.1}$$

where x is the number of correct classified samples in n testing trials and p_c is the level of chance. Due to the cross-validation method where all datapoints are used in testing, n in both Table 6.1 and Equation 6.1 is the total number of datapoints, hence the number of subjects times the number of trials. For example, transportation of two subjects would give $n = 120$ and transportation of three subject would give $n = 160$. The level of chance in 2-classification problems is $p_c = 0.50$. $I_{(0,1,\ldots,n)}$ is an indicator function which ensures that x only uses values of $(0, 1, \ldots, n)$. The value $y \in [0, 1]$ is the probability of observing up to x correct classified samples, hence a y close to 0 indicates that the result is far from statistically significant and $y = 1$ shows that it is statistically significant with a 95% confidence interval.

Taking into account the total number of subjects $N = 18$, the probability that two or more of these subjects get a probability above y is:

$$f = 1 - \left(y^N + Ny^{N-1}\left(1-y\right)\right) \tag{6.2}$$

where $f = 1$ indicates that the result were due entirely by chance and $f = 0$ indicates that the result is statistically significant. An f-value in between 0 and 1 lies in a grey zone where it is not possible to know for sure if it is statistically significant or not. However, one can interpretend that, for example, $f = 0.005$ is closer to become statistically significant than $f = 0.5$.

6.4 Support Vector Machine (SVM)

The Matlab function used for SVM is called *fitcsvm.m* and it is used for two-class (binary) classification problems. The SVM classifier finds a hyperplane which separates the datapoints of the different classes. It exists one parallel plane on each side of the hyperplane which are in contact with the closest datapoints. The datapoints on these parallel planes are called *support vectors* and the two parallel planes create a *margin*. The best hyperplane found by the SVM has the maximal margin width, and SVM is therefore sometimes referred to as a *maximum margin separator* [Awad and Khanna, 2015]. These features are illustrated in Figure 6.1 where the hyperplane separates the two classes $"+"$ and $"-"$.

When it is not possible to separate the input data with a hyperplane, the SVM uses predefined kernel functions to map the data into a new, higher-dimensional space. This higher-dimensional space is created in a way such that it is now possible to separate the classes with a hyperplane [Awad and Khanna, 2015].

6.5 *k*-Nearest Neighbour

The Matlab function used for *k*-nearest neighbour is called *fitcknn.m* where both the distance metric and the number of nearest neighbours *k* can be determined by the user. The classifier assigns the datapoints to the class with the most similar characteristics of a specific number of nearest neighbours. An example where potatoes should be assigned to one of the classes "fruits", "vegetables" and "grains" is illustrated in Figure 6.2. The considered characteristics of the classes are "crunchiness" and "sweetness". The number of nearest kinds of food is in this example chosen to be four and since two of them are vegetables, this is the class the potatoes will be assigned to [Zhang, 2016].

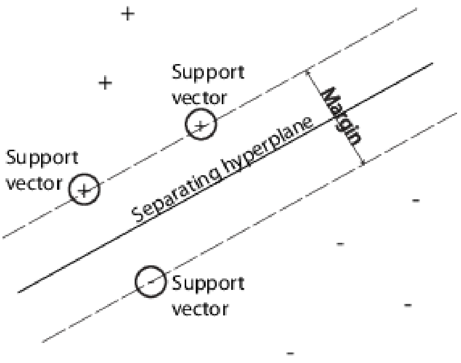


Figure 6.1: Illustration of SVM where the hyperplane separates the two classes "$+$" and "$-$". The closest datapoint on each side of the hyperplane are called support vectors and the width between the two parallel planes created by these support vectors is called the margin. SVM finds the hyperplance with the maximum width of the margin [*Support Vector Machines for Binary Classification* 2020].

Choosing a good value of number of nearest neighbours k is a balance between underfitting and overfitting. A too small value might increase the variance caused by random errors. However, a too large value would reduce the impact of small patterns which could be of importance [Zhang, 2016]. This thesis present the results from $k = 2$ and $k = 4$ since they in most cases gave the highest accuracies.

Naturally, the distance function plays a significant part of finding the nearest neighbour of a specific datapoint. The default distance function of the *fitcknn*-function is Euclidean distance and this is the one which will be used in this thesis. The Euclidean distance D between the two samples p and q is computed by:

$$D(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2} \tag{6.3}$$

where n is the number of characteristics [Zhang, 2016].

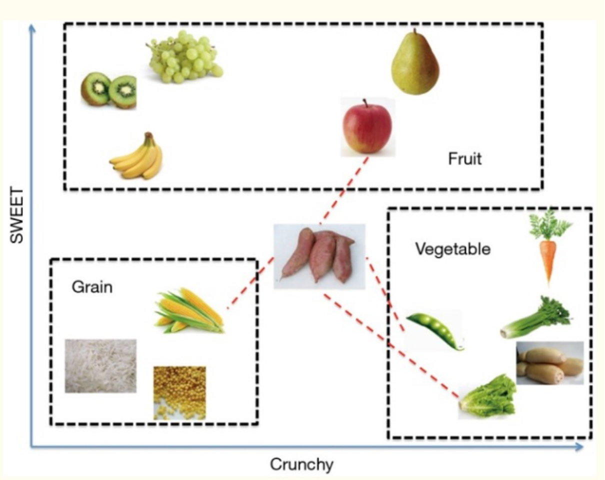


Figure 6.2: An example of a 4-nearest neighbours classification. Since two of the nearest neighbour of potatoes are vegetables, this is the class potatoes will be assigned to [Zhang, 2016].

6.6 Classification tree

The Matlab function used for the classification tree method is called *fitctree.m*. The method starts with a root node and branches out to internal nodes and leaf nodes and the structure reminds of an inverted tree. The root node, sometimes also called the starting node or parent node, represents a feature or attribute. Each branch can be viewed as a decision rule which, if it is fulfilled, leads to the next node. The leaf nodes, sometimes also called end nodes or child nodes, represent the final outcome from the selected decisions. All nodes between the root node and leaf nodes are called the internal nodes [Song and Lu, 2015].

Three of the most important steps in a classification tree are *splitting*, *stopping* and *pruning*. Splitting is the procedure where a node is divided into two or more sub-nodes. The splitting continuous until a pre-determined stopping criteria is met. The stopping criteria considers complexity and robustness of the model, too much splitting often results in overfitting and too few could result in underfitting. One way of dealing with this problem and finding a good size and complexity of the tree is through pruning. This procedure first builds a large and complex tree, and then prunes it down by removing unimportant nodes [Song and Lu, 2015].

Figure 6.3 illustrates a simple classification tree example with a binary root variable Y and two variables $x_1, x_2 \in \{0, 1\}$. Through the decision rules "Yes" or "No", the outcome is one of the leaf nodes $\{R_1, \ldots, R_5\}$ depending on the values of x_1 and x_2 [Song and Lu, 2015].

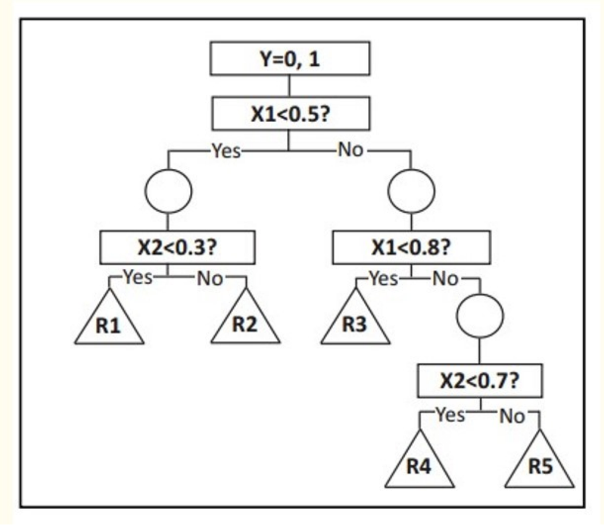


Figure 6.3: A simple classification tree example where the root node is the binary variable Y and the outcome is one of the leaf nodes $\{R_1, \ldots, R_5\}$ [Song and Lu, 2015].

7

t-SNE for visualization

7.1 Introduction

One common way of understanding and gaining knowledge of data is through visualization. However, reducing high dimensional data to two or three dimensions for visualization might result in a loss of important information. This is a challenging problem and various techniques have been developed over the last decades. The article *Visualizing Data using t-SNE* by Laurens van der Maaten and Geoffrey Hinton compares their own developed method *t-distributed Stochastic Neighbor Embedding* (t-SNE) with seven other commonly used techniques: (1) Sammon Mapping, (2) Curvilinear Components Analysis, (3) Stochastic Neighbor Embedding, (4) Isomap, (5) Maximum Variance Unfolding, (6) Locally Linear Embedding and (7) Laplacian Eigenmaps. Their results show that t-SNE is superior at clustering and revealing global structure in a single map [Maaten and Hinton, 2008]. Several articles support the advantages of using t-SNE. For example, the result in *Application of t-SNE to human genetic data* by Wentian Li et al. shows that t-SNE has a more robust way of taking care of outliers compared to the Principal Component Analysis (PCA)-technique [Li et al., 2017]. It has also been shown that t-SNE is exceptionally good at distinguishing important functional states in biomacromolecules simulations [Zhou et al., 2018]. Due to its widely use and superior results, t-SNE is the dimensionality reduction technique used in this thesis.

All notations used in this chapter is collected in Table 7.1

Table 7.1: Notations used in this chapter [Maaten and Hinton, 2008].

Notation	Description
$\mathscr{X}$	High-dimensional data set
$\mathscr{Y}$	Low-dimensional map
x_i	Datapoint in the high-dimensional data set
y_i	Map point on the low-dimensional map
n	Number of datapoints
$p_{j\|i}$	Conditional probability for high-dimensional datapoints
$q_{j\|i}$	Conditional probability for low-dimensional map points
p_{ij}	Joint probability for high-dimensional datapoints
q_{ij}	Joint probability for low-dimensional map points
σ_i	Variance of the Gaussian centered on datapoint x_i
P_i	The conditional probability distribution over all datapoints except x_i
Q_i	The conditional probability distribution over all map points except y_i
$H(P_i)$	Shannon entropy of P_i

7.2 Stochastic Neighbor Embedding (SNE)

t-SNE is developed from the Stochastic Neighbor Embedding (SNE)-technique presented by Geoffrey Hinton and Sam Roweis [Hinton and Roweis, 2002]. Therefore, SNE is first explained before moving on to t-SNE in the next section.

Consider a high-dimensional data set $\mathscr{X} = \{x_1, x_2, ..., x_n\}$ which should be visualized in a two-dimensional map $\mathscr{Y} = \{y_1, y_2, ..., y_n\}$ while preserving as much of the significant structure as possible. This is preferably done by keeping similar datapoints close to each other, which indicates that the distances to the neighbours are relevant.

Each data point x_i has a distance to all other data points which can be computed by various techniques. The default method is usually the Euclidean distance. The first step of the SNE-algorithm is to convert these distances into conditional probabilities which represent the relation between the data point and its neighbours. The interpretation of the conditional probability $p_{j|i}$ is the probability that datapoint x_j is picked as a neighbour to datapoint

x_i due to their probability density. This is under the presumption that x_i is Gaussian [Maaten and Hinton, 2008]. With the variance σ_i of the Gaussian centered on datapoint x_i, the conditional probability is computed using:

$$p_{j|i} = \frac{\exp\left(-\left\|x_i - x_j\right\|^2 / 2\sigma_i^2\right)}{\sum_{k \neq i} \exp\left(-\left\|x_i - x_k\right\|^2 / 2\sigma_i^2\right)} \tag{7.1}$$

The equation shows that nearby datapoints give a small value of the L_2-norm which results in a high value of the conditional probability. It is irrelevant to know the conditional probability $p_{i|i}$, and it is therefore put to zero [Maaten and Hinton, 2008].

In a similar way, the conditional probability $q_{j|i}$ for the datapoints y_i and y_j on the low-dimensional map Y is computed by:

$$q_{j|i} = \frac{\exp\left(-\left\|y_i - y_j\right\|^2\right)}{\sum_{k \neq i} \exp\left(-\left\|y_i - y_k\right\|^2\right)} \tag{7.2}$$

where the variance of the Gaussian is put to $1/\sqrt{2}$ and $q_{i|i}$ is put to zero [Maaten and Hinton, 2008].

The aim of the SNE-algorithm is to minimise the discrepancy between $p_{j|i}$ and $q_{j|i}$. The interpretation of the case where the two conditional probabilities are equal is that no information is lost in the dimensionality reduction and that the map points y_i and y_j model the datapoints x_i and x_j perfectly. However, this is not reasonable in reality and the SNE-algorithm needs a measurement method of how close the conditional probabilities are to each other. This is done by minimizing the sum of the Kullback-Leibler divergences:

$$C = \sum_i KL(P_i||Q_i) = \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}} \tag{7.3}$$

where C is the cost function. Given the datapoint x_i, the conditional probability distribution for all other datapoints is denoted P_i. Correspondingly, given the map point y_i, the conditional probability distribution for all other map points is denoted Q_i. The equation shows that the value of the cost function is zero if $q_{j|i} = p_{j|i}$ and that it is negative if $q_{j|i} > p_{j|i}$. The interpretation of the latter case is that the cost is small if widely separated datapoints in the high-dimensional set (small $p_{j|i}$) are represented by nearby map points (large $p_{j|i}$). On the other hand, the cost is large if nearby datapoints are represented by widely separated map points, hence if $q_{j|i} < p_{j|i}$ [Maaten and Hinton, 2008].

One important parameter in the SNE-algorithm is the *perplexity* which usually is a number in the range [5,50] and describes how many neighbours each datapoint takes into account in the calculations. The perplexity depends on the Shannon entropy $H(P_i)$ of P_i and is defined as:

$$\begin{aligned} Perp(P_i) &= 2^{H(P_i)} \\ H(P_i) &= -\sum_j p_{j|i} \log_2 p_{j|i} \end{aligned} \tag{7.4}$$

With a fixed perplexity set by the user, it is possible to select an effective value of the variance σ_i. Through a binary search, σ_i is set to the value which gives the specified perplexity to the conditional probability distrbution P_i over all datapoints, except for the datapoint x_i [Maaten and Hinton, 2008].

Finally, a gradient descent method is used to minimize the cost function in Equation 7.3 [Maaten and Hinton, 2008]:

$$\frac{\delta C}{\delta y_i} = 2\sum_j \left(p_{j|i} - q_{j|i} + p_{i|j} - q_{i|j}\right)(y_i - y_j) \tag{7.5}$$

The SNE-technique gives reasonable good visualizations, but it has a few disadvantages. Firstly, it is often necessary to redo the optimization a few times to find good enough values of the parameters. This gives a cost function which is difficult to optimize and a long computational time. Secondly, due to the scaling r^m for a datapoint i in a sphere with radius r and dimensional m, the area of the two-dimensional map has to be unreasonably large to be able to capture all the information from the datapoints. This is referred to as the *crowding problem*. The t-SNE technique was developed to solve these problems.

7.3 t-distributed Stochastic Neighbour Embedding (t-SNE)

The main difference between SNE and t-SNE lies in how the cost function C is evaluated:

1. t-SNE uses a symmetric cost function which gives simpler gradients to optimise and reduces the computational time.
2. In the low dimensional space, the similarities between two points is measured by a Student-t distribution with one degree of freedom in t-SNE instead of a Gaussian distribution, which deals with the *crowding problem*.

The symmetric cost function is given by:

$$C = \sum_i KL(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{7.6}$$

where P and Q are the same as in SNE, hence the joint probability distribution in the high- respectively low-dimensional space. Just like before, $p_{ii} = q_{ii} = 0$ and the symmetric property comes from $p_{ij} = p_{ji}$ and $q_{ij} = q_{ji}$ $\forall i, j$. In a similar way as for SNE, the joint probabilities q_{ij} in the low-dimensional space is defined as:

$$q_{ij} = \frac{\exp\left(-\,||y_i - y_j||^2\right)}{\sum_{k \neq l} \exp\left(-\,||y_k - y_l||^2\right)} \tag{7.7}$$

However, defining p_{ij} correspondingly as:

$$p_{ij} = \frac{\exp\left(-\left\|x_i - x_j\right\|^2 / 2\sigma^2\right)}{\sum_{k \neq l} \exp\left(-\left\|x_k - x_l\right\|^2 / 2\sigma^2\right)} \tag{7.8}$$

causes problems for outliers in the high-dimensional space. A huge value of x_i gives a very small value of p_{ij} $\forall j$, which becomes a problem when evaluating the cost function. According to (7.6), unreasonable small values of p_{ij} reduces the impact of q_{ij}, and hence the location of the map point y_i becomes irrelevant. Naturally, this is not desirable and t-SNE has an effective way of dealing with outliers by putting the joint probabilities p_{ij} in the high-dimensional space to:

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n} \tag{7.9}$$

where $p_{j|i}$ and $p_{i|j}$ are estimated by Equation 7.1 and n is the number of datapoints. Thereby, the impact of outliers x_i is reduced.

Just like for the SNE technique, a gradient descent method is used to minimize the cost function in Equation 7.6:

$$\frac{\delta C}{\delta y_i} = 4 \sum_j \left(p_{ij} - q_{ij}\right)\left(y_i - y_j\right) \tag{7.10}$$

Due to the symmetry, the gradients are simpler, less time-consuming and easier to optimize compared to the corresponding gradient descent method for SNE in (7.5).

The *crowding problem* refers to cases where the two-dimensional map area has to become unreasonably large to capture all the important information from the high-dimensional space. Since it is not possible to create the desired map size, the map points end up too close to each other which results in a crowded low-dimensional map. t-SNE deals with this problem by replacing the Gaussian distribution from SNE with a Student t-distribution with one degree of freedom. This changes the evaluation of the joint probabilities in the low-dimensional space in Equation 7.7 to:

$$q_{ij} = \frac{\left(1 + \left\|y_i - y_j\right\|^2\right)^{-1}}{\sum_{k \neq l} \left(1 + \left\|y_k - y_l\right\|^2\right)^{-1}} \tag{7.11}$$

Thereby, q_{ij} does not approach zero just as quickly as in Equation 7.7 for large pairwise distances $\left\|y_i - y_j\right\|$, which in turn gives a lower value of the cost function in Equation 7.6. In conclusion, large pairwise distances are now not as dependent of the map size and may be evaluated in the same way as nearby map points.

The use of the Student t-distribution also changes the gradient descent method in Equation 7.10 to:

$$\frac{\delta C}{\delta y_i} = 4 \sum_j \left(p_{ij} - q_{ij}\right)\left(y_i - y_j\right)\left(1 + \left\|y_i - y_j\right\|^2\right)^{-1} \tag{7.12}$$

The added contribution $\left(1 + \left\|y_i - y_j\right\|^2\right)^{-1}$ might seem to increase the computational time compared to Equation 7.10. However, the evaluation of the joint probability in Equation 7.11 is less time-consuming than the corresponding evaluation in Equation 7.7 since the exponential is removed. All things considered, the gradient descent method in Equation 7.12 is quite time effective.

t-SNE summary

Considering both the symmetric cost function and the Student t-distribution with one degree of freedom, the relevant t-SNE equations are:

$$
\begin{aligned}
C &= \sum_i KL(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}} && \text{Symmetric cost function} \\
p_{ij} &= \frac{p_{j|i} + p_{i|j}}{2n} && \text{Joint probability high-dim} \\
q_{ij} &= \frac{\left(1 + \left\|y_i - y_j\right\|^2\right)^{-1}}{\sum_{k \neq l} \left(1 + \|y_k - y_l\|^2\right)^{-1}} && \text{Joint probability low-dim} \\
\frac{\delta C}{\delta y_i} &= 4 \sum_j (p_{ij} - q_{ij})(y_i - y_j) \left(1 + \left\|y_i - y_j\right\|^2\right)^{-1} && \text{Gradient descent method}
\end{aligned}
\tag{7.13}
$$

8

Riemannian geometry

8.1 Introduction

The geometry taught in elementary and high schools is called *Euclidean geometry* after the Alexandrian Greek mathematician Euclid (300 BC). It is built on five simple axioms for plane and three-dimensional solid geometry, but it could be expanded to high-dimensional manifolds [*Euclidean geometry* 2020]:

1. Given two points, there is a straight line that joins them.
2. A straight line of finite length can be extended continuously without bounds.
3. A circle can be constructed when a point for its center and a distance for its radius are given.
4. All right angles are equal.
5. Through a point not on a given line there is only one line parallel to the given line.

Two examples given by these axioms is that the sum of all angles in a triangle always is 180 degrees (two right angles) and two parallel lines will always have the same distance to each other. Hence, Euclidean geometry is quite obvious, easy to understand and visualize and for 2000 years this was taken for granted as the only true geometry. However, in the early

19th century scientist realized that manifolds could be described using *non-Euclidean geometry*.

One such example is the *Riemannian geometry* developed by the German mathematician Bernhard Riemann in the mid-19th century [*Riemannian geometry* 2020]. While Euclidean geometry studies shapes on a plane, Riemannian geometry studies the shapes on a curved space such as the surface of a cylinder or sphere. This means that a line in Riemannian geometry is a great circle, i.e. the equator on Earth. All lines on a curved space must intersect with each other, which implies that the firth Euclidean axiom stated above is rejected in Riemannian geometry:

5. There are no lines parallel to the given line since all lines must intersect.

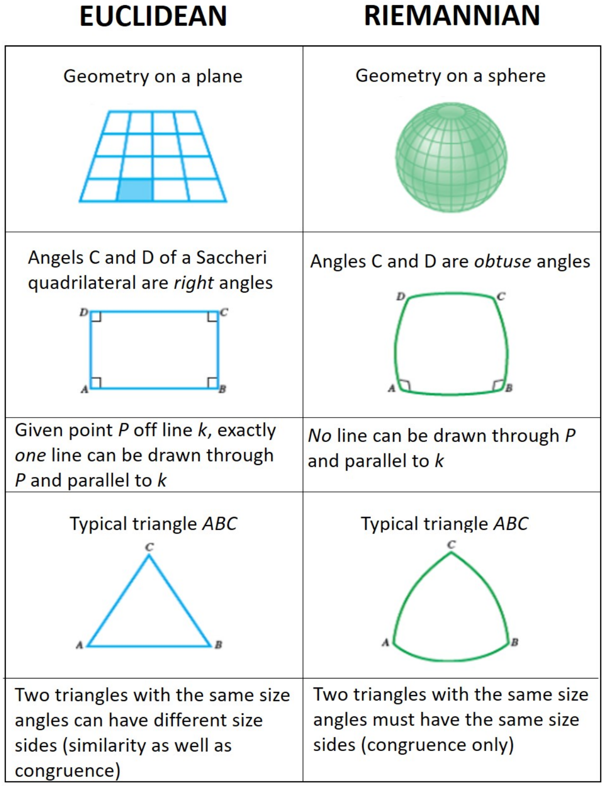
This gives some easily understandable differences compared to the examples above: In Riemannian geometry, the sum of all angles in a (large) triangle is greater than two right angles and it does not exist any parallel lines. Since the Earth is a sphere, Riemannian geometry is used to compute the routes of airplanes. However, small geometries on Earth could be approximated as Euclidean [*Riemannian geometry* 2020] . The differences between Euclidean and Riemannian geometry is illustrated in Figure 8.1.

Due to the complexity of high-dimensional datasets, the use of Euclidean geometry in algorithms often gives unreliable results. Riemannian geometry has been proven successful when working with covariance matrices, and therefore it is used in both *parallel transport* and *optimal transport* [Yair et al., 2019; Yair et al., 2020].

8.2 Definitions

Covariance matrices

By definition, all symmetric positive definit (SPD) matrices $\mathbf{P} \in \mathbb{R}^{d \times d}$ are symmetric with strictly positive eigenvalues, where d is the number of EEG



EUCLIDEAN	RIEMANNIAN
Geometry on a plane	Geometry on a sphere
Angels C and D of a Saccheri quadrilateral are *right* angles	Angles C and D are *obtuse* angles
Given point *P* off line *k*, exactly *one* line can be drawn through *P* and parallel to *k*	*No* line can be drawn through *P* and parallel to *k*
Typical triangle *ABC*	Typical triangle *ABC*
Two triangles with the same size angles can have different size sides (similarity as well as congruence)	Two triangles with the same size angles must have the same size sides (congruence only)

Figure 8.1: Differences between Euclidean and Riemannian geometry. A Saccheri quadrilateral (middle figures) has two equal sides perpendicular to the base [Miller et al., 2014].

channels plus the two audio-files. One of the most commonly used SPD matrix is the covariance matrix:

$$\mathbf{P}_{s,i} = \mathbb{E}\left[\left(\boldsymbol{x}_{s,i}(t) - \mu_{s,i}\right)\left(\boldsymbol{x}_{s,i}(t) - \mu_{s,i}\right)^T\right] \tag{8.1}$$

where $\mu_{s,i} = \mathbb{E}\left[\boldsymbol{x}_{s,i}(t)\right]$ for subject s and trial i. For $d = 66$ (64 EEG channels and two audio-files), each element in the 66×66 covariance matrix $\mathbf{P}_{s,i}$ describes the covariance between the corresponding channels and audio-files. The structure is illustrated in Figure 8.2.

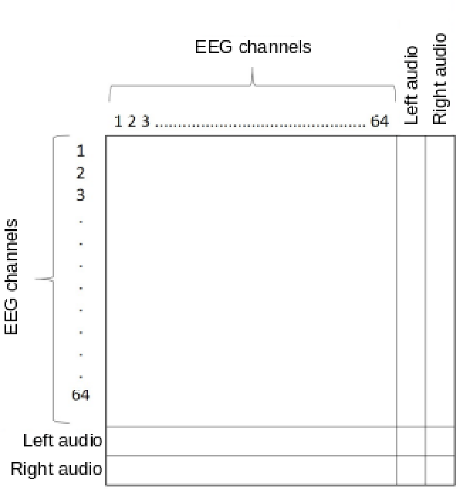


Figure 8.2: The covariance matrix structure with 64 EEG channels and two speech streams.

Some of the advantages of using covariance matrices are that they are usually robust to noise and they work well with Riemannian geometry.

Riemannian distance

The Riemannian manifold $\mathscr{M}$ is simply connected and one way to describe its curvature is through a so called *sectional curvature*. The manifold has a tangent space $\mathscr{T}_{\mathbf{P}}\mathscr{M}$ at the point $\mathbf{P} \in \mathscr{M}$ and the sectional curvature is defined by the point $\mathbf{P}$ and a two-dimensional plane on the tangent space. In this thesis, the point $\mathbf{P}$ is a covariance matrix. There exists a unique curve between two covariance matrices $\mathbf{P}_1, \mathbf{P}_2 \in \mathscr{M}$:

$$\varphi(t) = \mathbf{P}_1^{\frac{1}{2}} \left(\mathbf{P}_1^{-\frac{1}{2}} \mathbf{P}_2 \mathbf{P}_1^{-\frac{1}{2}} \right)^t \mathbf{P}_1^{-\frac{1}{2}}, \quad 0 \leq t \leq 1 \tag{8.2}$$

Thereby, it exists a unique *Riemannian distance* along the curve between the two covariance matrices:

$$d_R^2(\mathbf{P}_1,\mathbf{P}_2) = \left\| log\left(\mathbf{P}_1^{-\frac{1}{2}}\mathbf{P}_2\mathbf{P}_1^{-\frac{1}{2}}\right)\right\|_F^2 = \sum_{i=1}^{n} log^2\left(\lambda_i\left(\mathbf{P}_1^{-\frac{1}{2}}\mathbf{P}_2\mathbf{P}_1^{-\frac{1}{2}}\right)\right) \tag{8.3}$$

where $\|\cdot\|_F$ is the Frobenius norm, $log(\mathbf{P})$ is the matrix logarithm and $\lambda_i(\mathbf{P})$ is the i-th eigenvalue of $\mathbf{P}$.

Riemannian mean

The *Riemannian mean* $\bar{\mathbf{P}}_s$ for subject s is a 66×66 symmetric matrix computed by the Fréchet mean:

$$\bar{\mathbf{P}}_s \triangleq \arg\min_{\mathbf{P}_s \in \mathscr{M}} \sum_i d_R^2(\mathbf{P}_s,\mathbf{P}_{s,i}) \tag{8.4}$$

where $d_R^2(\mathbf{P}_s,\mathbf{P}_{s,i})$ is the Riemannian distance defined above. The interpretation of the Riemannian mean is the same as finding the center of mass in a high-dimensional Riemannian geometric figure. The Riemannian mean of two covariance matrices $\mathbf{P}_1,\mathbf{P}_2 \in \mathscr{M}$ is the midpoint of the curve $\varphi(t)$ in Equation 8.2:

$$\bar{\mathbf{P}} = \varphi\left(\frac{1}{2}\right) = \mathbf{P}_1^{\frac{1}{2}}\left(\mathbf{P}_1^{-\frac{1}{2}}\mathbf{P}_2\mathbf{P}_1^{-\frac{1}{2}}\right)^{\frac{1}{2}}\mathbf{P}_1^{-\frac{1}{2}} \tag{8.5}$$

The Riemannian mean for more than two covariance matrices can be computed by an iterative algorithm (1) developed by Barachant *et al*, where $\mathrm{Log}_{\bar{\mathbf{P}}}(\mathbf{P}_i)$ and $\mathrm{Exp}_{\bar{\mathbf{P}}}\left(\bar{\mathbf{S}}\right)$ are defined in Equations 9.1 and 9.2.

Algorithm 1: The Riemannian mean iterative algorithm for more than two SPD matrices [Barachant et al., 2013]

```
**Input:** a set of SPD matrices $\{\mathbf{P}_i \in \mathscr{M}\}_{i=1}^{n}$ where $n$ is the number of trials
**Output:** the Riemannian mean matrix $\bar{\mathbf{P}}$
  1. Compute the initial term $\bar{\mathbf{P}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{P}_i$
  2. while $\left\|\bar{\mathbf{S}}\right\|_F > \varepsilon$ do
        a) Compute the Euclidean mean in the tangent space
           $\bar{\mathbf{S}} = \frac{1}{n}\sum_{i=1}^{n}\mathrm{Log}_{\bar{\mathbf{P}}}(\mathbf{P}_i)$
        b) Update $\bar{\mathbf{P}} = \mathrm{Exp}_{\bar{\mathbf{P}}}\left(\bar{\mathbf{S}}\right)$
     end
```

Weigthed Riemannian mean

In optimal transport, explained in Chapter 10, a transportation plan matrix Γ is developed by a Sinkhorn OT algorithm. This transportation plan uses the weighted mean:

$$\hat{x}_i = t(x_i) = \arg\min_{x\in\mathbb{R}^n}\sum_j \Gamma[i,j]\,\|x - z_i\|_2^2 \tag{8.6}$$

to define a well-defined transportation map in Equation 10.7. The solution to the weighted mean problem is presented in Algorithm 2.

Algorithm 2: The Weighted Riemannian mean iterative algorithm for more than two SPD matrices [Yair et al., 2020]

```
Input: a set of SPD matrices $\{\mathbf{P}_i \in M\}_{i=1}^{n}$ and non-negative
 weights $\{w_i\}_{i=1}^{n}$ such that $\sum_i w_i = 1$
Output: the weighted Riemannian mean matrix $\bar{\mathbf{P}}$ satisfying
 $\bar{\mathbf{P}} = \arg\min_{\mathbf{P} \in M} \sum_i w_i d_R^2 (\mathbf{P}, \mathbf{P}_i)$
   1. Compute the initial term $\bar{\mathbf{P}} = \frac{1}{n} \sum_{i=1}^{n} w_i \mathbf{P}_i$
   2. while $\left\|\bar{\mathbf{S}}\right\|_F > \varepsilon$ do
          a) Compute the Euclidean mean in the tangent space
             $\bar{\mathbf{S}} = \frac{1}{n} \sum_{i=1}^{n} w_i \mathrm{Log}_{\bar{\mathbf{P}}} (\mathbf{P}_i)$
          b) Update $\bar{\mathbf{P}} = \mathrm{Exp}_{\bar{\mathbf{P}}} \left(\bar{\mathbf{S}}\right)$
      end
```

9

Method 1: Parallel Transport

9.1 Introduction

The first domain adaptation-method used in this thesis is named *parallel transport* (PT) and it was presented in the article *Parallel Transport on the Cone Manifold of SPD Matrices for Domain Adaptation* by Or Yair *et al* in 2019. The article focuses on covariance matrices that do not live in the same region of the manifold, a scenario occurring when data is collected from several subjects and/or sessions. Covariance matrices are commonly used and proven to be effective features when working with time series.

Briefly described, the parallel transport method uses Riemannian geometry to:

1. Compute the Riemannian mean M_s of all covariance matrices for each subject s.
2. Compute the Riemannian mean D of all M_s from step 1
3. Project all the covariance matrices from the Riemannian manifold onto a Riemannian tangent plane at M_s for each subject s.
4. Move all the data to D using parallel transport
5. Project the covariance matrices from the Riemannian tangent plane back to the Riemannian manifold

6. Project the covariance matrices to the Euclidean tangent space for classification and plotting purposes

The reason for the last step is that Euclidean geometry is convenient since the default metric in both the t-SNE and the classification algorithm is Euclidean.

An illustration of parallel transport is shown in Figure 9.3. The Riemannian mean D is a 66×66-matrix, and the dimensional reduction through t-SNE to two dimensions is the reason it does not look like it is the mean in the figure.

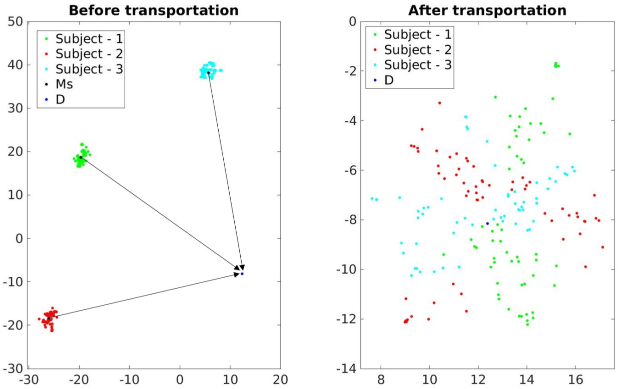


Figure 9.1: Parallel transportation of data from subjects 1-3, where M_s is the Riemannian mean of subject s and D is the target Riemannian mean of of all M_s.

All notations used in this chapter are presented in Table 9.1.

Table 9.1: Notations used in this chapter [Yair et al., 2019].

Notation	Description
n	Number of trials
d	Number of EEG channels and audio-files
N	Number of subjects
$\mathbf{P}_{s,i}$	Covariance matrix for trial i and subject s
$\mathscr{M}$	The Riemannian manifold
$\mathscr{T}_{\mathbf{P}}\mathscr{M}$	The tangent plane of the manifold $\mathscr{M}$ at the symmetric matrix $\mathbf{P}$
$\varphi(t)$	The unique curve between two covariance matrices $0 \leq t \leq 1$
$d_R^2(P_1, P_2)$	The squared Riemannian distance between the covariance matrices $\mathbf{P}_1, \mathbf{P}_2 \in \mathscr{M}$
M_s	Riemannian mean of $\mathbf{P}_{s,i}, \forall i$, for subject s
D	Riemannian mean of $M_s, \forall s$
$\mathbf{S}_{s,i}$	Symmetric matrix $\in \mathscr{T}_{\mathbf{P}}\mathscr{M}$ for trial i and subject s

9.2 Definitions

Riemannian mean

Algorithm 1 in Chapter 8 is used to compute the Riemannian mean in the parallel transport method. Since the number of covariance matrices is the same as the number of trials (which are more than two), Algorithm 1 is used to compute the Riemannian mean M_s for each subject s. For generalization purposes, the Riemannian mean D of the centroids M_s, $s = 1, ..., N$ is also computed by Algorithm 1 (step 2), even though Equation 8.5 could be used in the specific case of $N = 2$. Theoretically, the point D could be chosen arbitrary. However, computing D as the mean of the centroids M_s has the advantage of an overall minimum transportation which avoids unnecessary distortions.

Exponential and Logarithm Maps

As described in the introduction to the chapter, parallel transport first projects the covariance matrices to the tangent plane $\mathscr{T}_{\mathbf{P}}\mathscr{M}$ of the manifold $\mathscr{M}$, moves the data to the Riemannian mean D and then projects the covariance matrices back to the manifold. The projection of $\mathbf{P}_{s,i}$ onto the tangent plane at the point $\mathbf{P}$ (step 3) is done by the *Logarithm map*:

$$\mathbf{S}_{s,i} = \mathrm{Log}_{\mathbf{P}}(\mathbf{P}_{s,i}) = \mathbf{P}^{\frac{1}{2}} \log\left(\mathbf{P}^{-\frac{1}{2}} \mathbf{P}_{s,i} \mathbf{P}^{-\frac{1}{2}}\right) \mathbf{P}^{\frac{1}{2}} \quad \in \mathscr{T}_{\mathbf{P}}\mathscr{M} \tag{9.1}$$

where $\mathbf{S}_{s,i}$ is a symmetric matrix which can be represented as a vector. After the transport, this vector is projected back to the manifold by the *Exponential map* (step 5):

$$\mathbf{P}_{s,i} = \mathrm{Exp}_{\mathbf{P}}(\mathbf{S}_{s,i}) = \mathbf{P}^{\frac{1}{2}} \exp\left(\mathbf{P}^{-\frac{1}{2}} \mathbf{S}_{s,i} \mathbf{P}^{-\frac{1}{2}}\right) \mathbf{P}^{\frac{1}{2}} \quad \in \mathscr{M} \tag{9.2}$$

The exponential and logarithm maps are illustrated in Figure 9.2. The grey/black line between the two points x_0 and x on the Riemannian manifold $\mathscr{M}$ is the minimum length curve. u is a vector on the tangent plane of x_0. The exponential map projects u to a point $x \in \mathscr{M}$ in the direction of u. The logarithmic map is the inverse where the point $x \in \mathscr{M}$ is projected to the tangent plane $u \in \mathscr{T}_{x_0}\mathscr{M}$ [Calinon, 2020].

9.3 Transportation

Step 4 in the introduction contains the transportation of all the covariance matrices $\mathbf{P}_{s,i}$ from M_s to D for subject $s = [1,...,N]$ and trial $i = [1,...,n]$. As described above, M_s is the Riemannian mean of all the covariance matrices for subject s and D is the Riemannian mean of all M_s.

The transportation could be explained through the map $\Gamma : \mathscr{T}_{M_s}\mathscr{M} \rightarrow \mathscr{T}_D\mathscr{M}$, meaning that the covariance matrices at the tangent plane of M_s are moved to the tangent plane of D. Since the transportation is made on the tangent plane, it is actually the symmetric matrices $\mathbf{S}_{s,i} \in \mathscr{T}_{M_s}\mathscr{M}$ from the Logarithm map in Equation 9.1 which are transported using:

$$\mathbf{S}^{D}_{s,i} = \Gamma_{M_s \rightarrow D}\left(\mathbf{S}^{M_s}_{s,i}\right) \triangleq \mathbf{E}\mathbf{S}^{M_s}_{s,i}\mathbf{E}^{T} \tag{9.3}$$

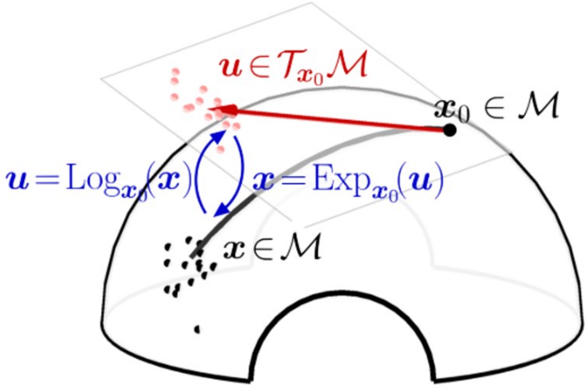

$u \in \mathcal{T}_{x_0}\mathcal{M}$
$x_0 \in \mathcal{M}$
$u = \mathrm{Log}_{x_0}(x)$
$x = \mathrm{Exp}_{x_0}(u)$
$x \in \mathcal{M}$

Figure 9.2: Illustration of the exponential and logarithmic maps between the Riemannian manifold $\mathscr{M}$ and the tangent plane $\mathscr{T}_{x_0}\mathscr{M}$, where $x_0 \in \mathscr{M}$ [Calinon, 2020].

where $\mathbf{E} = (DM_s^{-1})^{\frac{1}{2}}$ [Yair et al., 2019].

Figure 9.3 shows an illustration of the parallel transport method of the vector $u \in \mathscr{T}_g\mathscr{M}$. The goal is to transport u, along infinitesimally close tangent spaces, to the tangent space $\mathscr{T}_h\mathscr{M}$ of the point h on the manifold $\mathscr{M}$. The black vectors show the direction of the transportation in each tangent space. Using infinitesimally close tangent spaces gives a smooth transportation with preserved features of the vector u [Calinon, 2020].

In the last step, the covariance matrices $\mathbf{P}_{s,i}^D$ are projected from the Riemannian manifold to the Euclidean tangent plane to facilitate the classification and plotting. This is done by approximating the Riemannian distances d_R^2 between the covariance matrices $\mathbf{P}_{s,i}^D$ and $\mathbf{P}_{s,j}^D$ as a squared Euclidean distances by:

$$d_R^2\left(\mathbf{P}_{s,i}^D, \mathbf{P}_{s,j}^D\right) \approx \|\tilde{\mathbf{S}}_{s,i}^D - \tilde{\mathbf{S}}_{s,j}^D\|_F^2 \tag{9.4}$$

where $\tilde{\mathbf{S}}_{s,i}^D = D^{-\frac{1}{2}}\mathrm{Log}_D\left(\mathbf{P}_{s,i}^D\right)D^{-\frac{1}{2}} = \log\left(D^{-\frac{1}{2}}\mathbf{P}_{s,i}^D D^{-\frac{1}{2}}\right)$. Since $\tilde{\mathbf{S}}_{s,i}^D$ are symmetric matrices in the Euclidean space, they could be vectorized from only the upper (or lower) triangular elements with a gain factor of $\sqrt{2}$ on

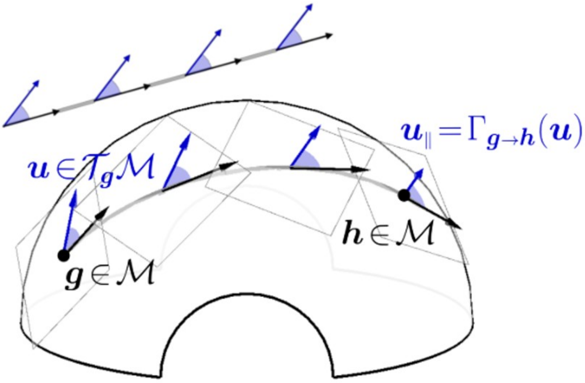

$u_{\parallel} = \Gamma_{g \to h}(u)$
$u \in \mathcal{T}_g\mathcal{M}$
$h \in \mathcal{M}$
$g \in \mathcal{M}$

Figure 9.3: An illustration of the parallel transport method of the vector $u \in \mathscr{T}_g\mathscr{M}$. The goal is to transport u, along infinitesimally close tangent spaces, to the tangent space $\mathscr{T}_h\mathscr{M}$ of the point h on the manifold $\mathscr{M}$. [Calinon, 2020].

all the elements except the diagonal. These feature vectors are used for classification and plotting through t-SNE.

9.4 Matlab

All the steps from the introduction have now been explained and summarized, including the equations, in Table 9.2.

Steps 3-5 can be combined with gives the projection to the tangent plane, transportation along the tangent planes and projection back to the manifold in one equation:

$$\mathbf{P}^{D}_{s,i} = \mathrm{Exp}_D\left(\Gamma_{M_s \to D}\left(\mathrm{Log}_D\left(\mathbf{P}^{M_s}_{s,i}\right)\right)\right) = \mathbf{E}\mathbf{P}^{M_s}_{s,i}\mathbf{E}^T \qquad (9.5)$$

where $\mathbf{E} = (DM_s^{-1})^{\frac{1}{2}}$ [Yair et al., 2019]. Hence, the Matlab pseudo code is given by Algorithm 3.

Table 9.2: The parallel transport steps

Step	Description	Matlab
1.	Compute the Riemannian means $M_s, \forall s$	Algorithm 1, Chapter 8
2.	Compute the Riemannian mean D of all M_s	Algorithm 1, Chapter 8
3.	Project all the covariance matrices $\mathbf{P}_{s,i}$ from the manifold $\mathscr{M}$ to the tangent plane $\mathscr{T}_{M_s}\mathscr{M}$	$\mathbf{S}_{s,i}^{M_s} = \mathrm{Log}_D\left(\mathbf{P}_{s,i}^{M_s}\right), \quad (9.1)$
4.	Move all the data to D	$\mathbf{S}_i^D = \Gamma_{M_s \to D}\left(\mathbf{S}_{s,i}^{M_s}\right), \quad (9.3)$
5.	Project the symmetric matrices $\mathbf{S}_{s,i}$ back to the manifold $\mathscr{M}$	$\mathbf{P}_i^D = \mathrm{Exp}_D\left(\mathbf{S}_i^D\right), \quad (9.2)$
6.	Project the covariance matrices to the Euclidean tangent space	$\tilde{\mathbf{S}}_i^D = \log\left(D^{-\frac{1}{2}}\mathbf{P}_i^D D^{-\frac{1}{2}}\right)$

Algorithm 3: Domain Adaptation Using Parallel Transport for SPD Matrices [Yair et al., 2019]

```
Input: $\{\mathbf{P}_{1,i}\}_{i=1}^n, \dots, \{\mathbf{P}_{s,i}\}_{i=1}^n, \dots, \{\mathbf{P}_{N,i}\}_{i=1}^n$ where $\mathbf{P}_{s,i}$ is the
 covariance matrix for subject $s$ and trial $i$.
Output: $\{\tilde{\mathbf{S}}_{s,i}\}_{i=1}^n, \dots, \{\tilde{\mathbf{S}}_{s,i}\}_{i=1}^n, \dots, \{\tilde{\mathbf{S}}_{N,i}\}_{i=1}^n$ where $\tilde{\mathbf{S}}_{s,i}$ is the
 new representation of $\mathbf{P}_{s,i}$ in a Euclidean space.
    1. For each $i \in \{1,2,\dots,n\}$, compute the Riemannian mean $M_s$ of the
       subset $\{\mathbf{P}_{s,i}\}$
    2. Compute $D$, the Riemannian mean of $\{M_s\}_{s=1}^N$
  3-5. For all $s$ and $i$, apply projection and Parallel Transport using
       Equation 9.5:
           $\mathbf{P}_{s,i}^D = \mathbf{E}\mathbf{P}_{s,i}^{M_s}\mathbf{E}^T, \quad \mathbf{E} = (DM_s^{-1})^{\frac{1}{2}}$
    6  For all $i$, project the transported matrix to the tangent space via:
           $\tilde{\mathbf{S}}_{s,i}^D = \log\left(D^{-\frac{1}{2}}\mathbf{P}_{s,i}^D D^{-\frac{1}{2}}\right)$
```

10

Method 2: Optimal transport

10.1 Introduction

The second method is called *Optimal Transport* (OT) and was presented in the article *Optimal Transport on the Manifold of SPD matrices for domain adaptation* in March 2020 by Or Yair *et al*. Optimal transport is similar to parallel transport since they both uses Riemannian geometry on SPD matrices, in these cases covariance matrices. A difference between the methods is however that PT moves the covariance matrices from all the subject to a Riemannian mean D, meanwhile OT keeps the covariance matrices from one subject and moves all the other covariance matrices to this domain. Another difference is that optimal transport views the covariance matrices as *measures* with densities $\hat{\boldsymbol{f}} \in \mathbb{R}^n$, where n is the number of trials. This is used to create an optimal transportation plan by minimizing a certain cost function with respect to these densities.

Briefly described, the optimal transport method uses Riemannian geometry to:

1. Compute the densities $\hat{\boldsymbol{f}}_s \in \mathbb{R}^n$ for each subject
2. Compute the Riemannian distances between the covariance matrices of the target data from subject A and all the covariance matrices from the other subjects

3. The Riemannian distances from step (2) are viewed as transportation costs. An optimal transportation plan Γ^* is produced by minimizing the transportation cost with the Sinkhorn OT algorithm
4. Apply the transportation plan using the map $t(\mathbf{P}_{s,i})$ which is defined by the *weighted Riemannian mean*
5. Project the covariance matrices to the Euclidean tangent space for classification and plotting purposes

All notations used in this chapter are presented in Table 10.1.

Table 10.1: Notations used in this chapter [Yair et al., 2019].

Notation	Description
n	Number of trials
d	Number of EEG channels and audio-files
N	Number of subjects
$\mathscr{M}$	The Riemannian manifold
$\mathscr{T}_{\mathbf{P}}\mathscr{M}$	The tangent plane of the manifold $\mathscr{M}$ at the symmetric matrix $\mathbf{P}$
$\mathbf{P}_{s,i}$	Covariance matrix $\in \mathscr{M}$ for trial i and subject s
$\mathbf{S}_{s,i}$	Covariance matrix $\in \mathscr{T}_{\mathbf{P}}\mathscr{M}$ for trial i and subject s
$d_R^2(\mathbf{P}_2, \mathbf{P}_1)$	The Riemannian distance between the covariance matrices $\mathbf{P}_1, \mathbf{P}_2 \in \mathscr{M}$
$\mathbf{C}$	Cost function
Γ	Transportation plan
f	Density of a covariance matrix
$t(\mathbf{x})$	Transportation map
M_s	Riemannian mean of $\mathbf{P}_{s,i}, \forall i$, for subject s
D	Riemannian mean of $M_s, \forall s$
$\mathbf{S}_{s,i}$	Symmetric matrix $\in \mathscr{T}_{\mathbf{P}}\mathscr{M}$ for trial i and subject s

10.2 Definitions

Covariance matrices $\mathbf{P}$, the Riemannian manifold $\mathscr{M}$, Riemannian squared distance $d_R^2(\mathbf{P}_s, \mathbf{P}_1), s \neq 1$ and weighted Riemannian mean are all defined in Chapter 8.

Cost function

The cost function $\mathbf{C}(\mathbf{P}_s,\mathbf{P}_1) \in \mathbb{R}^{n_1 \times n_2}$, $s \neq 1$ is a matrix describing the cost of moving all covariance matrices from subject $s \neq 1$ to subject 1, where n_1 is the number of trials in the source set and n_2 is the number of trials in the target set. In the DTU-dataset, $n_1 = n_2 = n$. It makes sense to choose the cost function as the Riemannian distance matrix $d_R^2(\mathbf{P}_s,\mathbf{P}_1)$, $s \neq 1$, where a large distance between two SPD matrices gives a large cost.

Optimal transportation plan

One way of solving the optimal transport problem is by the Kantorovich formulation where the aim is to find the optimum transportation plan γ^* [Yair et al., 2020]. Imagine the transportation of the covariance matrix $\mathbf{P}_2$ to the location of the covariance matrix $\mathbf{P}_1$ on the Riemannian manifold $\mathscr{M}$. This transportation γ can be done in many different ways, but only one route γ^* is optimal in the sense that it occurs at a minimal cost $C(\mathbf{P}_2,\mathbf{P}_1)$, meaning a minimal Riemannian distance between the two covariance matrices. The Kantorovich formulation defines two continuous Borel measurements μ_1 and μ_2 on the Riemannian manifold $\mathscr{M}$ which have densities [Yair et al., 2020]:

$$\begin{aligned} f_1(\mathbf{P}_1) &= \int_{\mathscr{M}} \gamma(\cdot,\mathbf{P}_1)\, d\text{vol} \\ f_2(\mathbf{P}_2) &= \int_{\mathscr{M}} \gamma(\mathbf{P}_2,\cdot)\, d\text{vol} \end{aligned} \tag{10.1}$$

The Kantorovich optimal transportation plan γ^* in the continuous case is given by the solution to:

$$\inf_\gamma \int_{\mathscr{M}\times\mathscr{M}} C(\mathbf{P}_2,\mathbf{P}_1)\, d\gamma(\mathbf{P}_2,\mathbf{P}_1) \tag{10.2}$$

The problem in this thesis is discrete and data from each subject s contains n_s number of trials, where each trial is a covariance matrix. The aim is to move all covariance matrices from subject 2 to the location of subject 1. The densities in Equation 10.1 are in this case sampled with n_1 respectively n_2 points which gives two density vectors $\hat{\mathbf{f}}_1 \in \mathbb{R}^{n_1}$, $\hat{\mathbf{f}}_2 \in \mathbb{R}^{n_2}$ and the discrete optimal transport plan Γ^* is the solution to:

$$\min_{\Gamma \in F} \langle \Gamma, \mathbf{C} \rangle \tag{10.3}$$

where $F = \left\{ \Gamma \in \mathbb{R}^{n_1 \times n_2} | \Gamma 1_{n_2} = \hat{\mathbf{f}}_1, \Gamma^T 1_{n_1} = \hat{\mathbf{f}}_2 \right\}$ and $\mathbf{C} \in \mathbb{R}^{n_1 \times n_2}$ is the cost matrix [Yair et al., 2020].

An extended and more accurate version of the classical optimal transportation plan in Equation 10.3 is preferred for larger values of the number of trials n_1 and n_2. A regularization entropy term $h(\Gamma) = -\sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \Gamma[i,j] \log(\Gamma[i,j])$ is subtracted accordingly:

$$\min_{\Gamma \in F} \langle \Gamma, \mathbf{C} \rangle - \frac{1}{\lambda} h(\Gamma) \tag{10.4}$$

where $\lambda \in [0, \infty)$ is adaptively set to:

$$\lambda = \frac{1}{2m^2} \tag{10.5}$$

for $m = 0.05 \cdot \text{median}\{\mathbf{C}[i,j]\}_{i,j}$ [Yair et al., 2020]. Equation 10.4 is solved by a sinkhorn optimal transport algorithm described in Algorithm 4.

The polar factorization theorem

Step 4 in the introduction is about applying the computed transportation plan Γ. This gives the map:

$$t(\mathbf{x}_i) = \mathbf{T}\mathbf{x}_i + \mathbf{b} \tag{10.6}$$

Algorithm 4: Sinkhorn OT between the covariance matrices of source subject s, $\mathbf{P}_{s,i}$ and the covariance matrices of the target subject $\mathbf{P}_{1,j}$ for the trials $i, j = [1 \ldots n]$ [Cuturi, 2013].

```
Input: The transportation cost matrix $\mathbf{C}(\mathbf{P}_s, \mathbf{P}_1) \in \mathbb{R}^{n\times n}$ and the
 densities $\hat{\boldsymbol{f}} \in \mathbb{R}^n$
Output: The matrix transportation plan $\Gamma \in \mathbb{R}^{n\times n}$
    1. Compute the number of trials $n$
    2. Compute $\lambda = 1/(2 * (0.05 * median(\mathbf{C}(:)))^2)$
    3. $K = \exp(-\lambda \mathbf{C})$
    4. $u = length(n, 1)/n$
    5. $\tilde{K} = K./\hat{\boldsymbol{f}}$
    6. for i=1:1000, or any other stopping criterion
          • $u = 1./(\tilde{K}(\hat{\boldsymbol{f}}./(K^T u)))$
    7. $v = \hat{\boldsymbol{f}}./(K^T u)$
    8. $\Gamma = (K. * u). * v^T$
```

where $\mathbf{x}_i \in \mathbb{R}^d, \forall i \in \{1, 2, ..., n\}$ is the source sample, $\mathbf{S} \in \mathbb{R}^{d\times d}$ and $\mathbf{T} = \left(\mathbf{S}\mathbf{S}^T\right)^{\frac{1}{2}} > 0$. Just like earlier, n is the number of trials and d is the number of channels and audio files. Equation 10.6 is the solution to the OT problem for two discrete distributions μ_1 and μ_2 with N Diracs, if and only if four conditions are fulfilled [Yair et al., 2020]:

1. The source sample $\mathbf{x}_i \in \mathbb{R}^d, \forall i \in \{1, 2, ..., n\}$ fulfills $\mathbf{x}_i \neq \mathbf{x}_j$ for $i \neq j$
2. The source and target distribution weights are $\frac{1}{n}$
3. The target samples are defined as $\mathbf{z}_i = \mathbf{T}\mathbf{x}_i + \mathbf{b}$, where $\mathbf{T} > 0$
4. The cost function is $\mathbf{C}(\mathbf{x}, \mathbf{z}) = ||\mathbf{x} - \mathbf{z}||_2^2$

As explained above, the cost function in this thesis is set to be the squared Riemannian distance matrix, hence:

$$\mathbf{C}(\mathbf{P}_s, \mathbf{P}_1) = d_R^2(\mathbf{P}_s, \mathbf{P}_1) \in \mathbb{R}^{n \times n}, \quad s \neq 1 \tag{10.7}$$

The map $t(\mathbf{x})$ is well-defined if it is computed by the *weighted* Riemannian mean (defined in Section 8.2) which gives a strictly convex optimization problem. Finally, step 4 in the introduction gives the solution to the OT problem by:

$$\tilde{\mathbf{P}}_{s,i}^{D} = t(\mathbf{P}_i) = \arg\min_{\mathbf{P} \in \mathscr{M}} \sum_{j=1}^{n} \Gamma[i,j]\, d_R^2(\mathbf{P}_{s,j}, \mathbf{P}_{1,j}) \quad s \neq 1 \tag{10.8}$$

where $\mathscr{M}$ is the Riemannian manifold.

Step 5 in the introduction is the same as the last step in the parallel transport method, hence projecting the covariance matrices to the Euclidean tangent space for classification and plotting purposes:

$$\tilde{\mathbf{S}}_{s,i}^{D} = D^{-\frac{1}{2}} \operatorname{Log}_D \left(\tilde{\mathbf{P}}_{s,i}^{D} \right) D^{-\frac{1}{2}} = \log \left(D^{-\frac{1}{2}} \tilde{\mathbf{P}}_{s,i}^{D} D^{-\frac{1}{2}} \right) \tag{10.9}$$

where D is the Riemannian mean of all the covariance matrices.

10.3 Matlab

All the steps from the introduction have now been explained and the Matlab pseudo code is given by Algorithm 5.

Algorithm 5: Domain Adaptation Using Optimal Transport for SPD Matrices [Yair et al., 2020]

```
Input: the source set $\{\mathbf{P}_{1,i}\}_{i=1}^{n}$ and the target sets $\left\{\mathbf{P}_{s,j}\right\}_{j=1}^{n}$,
  $s \in \{2,\ldots,N\}$ where $\mathbf{P}_{s,i}$ and $\mathbf{P}_{s,j}$ are the covariance matrices for
  subject $s$ and trial $i$ respectively trial $j$, and $N$ is the total number
  of subjects.
Output: the adapted source sets $\left\{\tilde{\mathbf{S}}_{s,i}\right\}_{i=1}^{n}$, where $\tilde{\mathbf{S}}_{s,i}$ is the new
  representation of $\mathbf{P}_{s,i}, \forall s$ in a Euclidean space.
  1. For each subject $s$, compute the densities$\hat{\boldsymbol{f}}_s = 1/n$
  2. For each trial $[i,j]$ and subject $s \neq 1$, compute the cost function:
     $\mathbf{C}(\mathbf{P}_{s,i}, \mathbf{P}_{1,j}) = d_R^2(\mathbf{P}_{s,i}, \mathbf{P}_{1,j})$
  3. Compute the transportation plan $\Gamma$ using sinkhorn OT in Algorithm
     4.
  4. Apply the transportation plan:
     $$\tilde{\mathbf{P}}_{s,i} = t(\mathbf{P}_{s,i}) = \arg\min_{\mathbf{P}\in\mathscr{M}} \sum_{j=1}^{n} \Gamma[i,j]\, d_R^2(\mathbf{P}_s, \mathbf{P}_{1,j}) \quad s \neq 1$$
  5. For all $s$ and $i$, project the transported matrix to the tangent space
     via:
     $$\tilde{\mathbf{S}}_{s,i}^{D} = \log\left(D^{-\frac{1}{2}} \mathbf{P}_{s,i}^{D} D^{-\frac{1}{2}}\right)$$
```

11

Results

11.1 Classification for each subject before transportation

Male/female attention classification

Figure 11.1 shows that cross-validation classification accuracy for each subject with SVM, *3*-nearest neighbours and decision tree classification methods. The figure shows that the classification accuracy differed greatly between the subjects. SVM gave the best results and decision tree gave, for most subjects, the worst results. All subjects, excepts two with decision tree, were above the statistically significant level of 60% for $n = 60$ testing points. Table 11.1 shows the classification accuracies with SVM where subjects 2, 3, 7, 8, 12, 15 and 18 were over 90%.

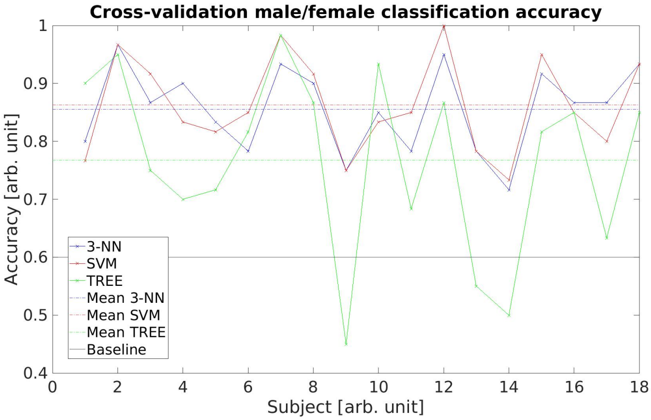


Figure 11.1: Male/female cross-validation classification accuracy with SVM, *3*-nearest neighbour and decision tree for all 18 subjects. Baseline is the level of chance for $n = 60$ testing points.

Table 11.1: Male/female cross-validation classification accuracy with SVM for each subject. All subjects gave statistically significant results (above 60%).

Subject #	Acc (%)	Subject #	Acc (%)	Subject #	Acc (%)
1	76,67	7	98,33	13	78,33
2	96,67	8	91,67	14	73,33
3	91,67	9	75,00	15	95,00
4	83,33	10	83,33	16	85,00
5	81,67	11	85,00	17	80,00
6	85,00	12	100	18	93,33

Left/Right attention classification

Figure 11.2 shows that cross-validation classification accuracy for each subject with SVM, *3*-nearest neighbours and decision tree classification methods. The results differed greatly between subjects and were generally lower than for male/female classification. Only SVM gave a mean accuracy above the statistically significant level of 60%, and the worst results were from decision tree. Table 11.2 shows the classification accuracies with SVM where subjects 2, 9, 10, 11, 12, 14, 17 and 18 gave the best results.

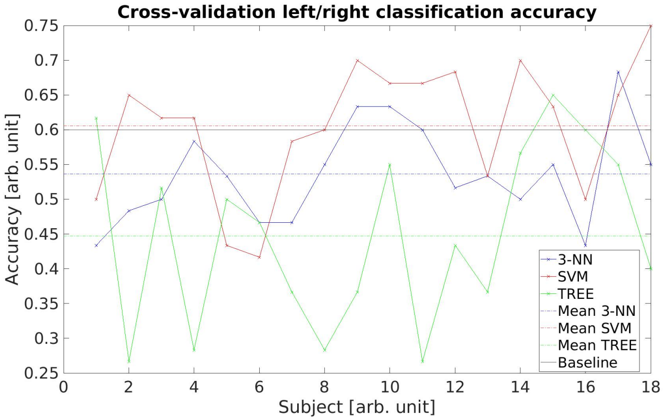


Figure 11.2: Left/right cross-validation classification accuracy with SVM, *3*-nearest neighbour and decision tree for all 18 subjects. Baseline is the level of chance for $n = 60$ testing points.

11.2 Transportation of two subjects

Several subjects were tested and the classification accuracy after the transportation differed greatly depending on which subjects were used. The thesis presents the best results for male/female classification acquired by subjects 2 and 7, and the best results for left/right classification acquired by subjects 4 and 9. Two results are also presented with randomized subjects.

Table 11.2: Left/right cross-validation classification accuracy with SVM for each subject. 13 subjects gave statistically significant results (above 60%).

Subject #	Acc (%)	Subject #	Acc (%)	Subject #	Acc (%)
1	50,00	7	58,33	13	53,33
2	65,00	8	60,00	14	70,00
3	61,67	9	70,00	15	63,33
4	61,67	10	66,67	16	50,00
5	43,33	11	66,67	17	65,00
6	41,67	12	68,33	18	75,00

Transportation of subjects 2 and 7

Table 11.3 shows the cross-validation classification accuracy for subjects 2 and 7 which gave the best results for male/female classification. SVM, decision tree and k-Nearest Neighbour are used for both male/female and left/right classification. Visualization through t-SNE is presented in the Appendix.

Table 11.3: Cross-validation classification accuracy for both male/female and left/right classification with subjects 2 and 7. $f = 0$ means that the result is statistically significant at a 0.05 level and $f = 1$ means that the result could be due to chance. The highest male/female and left/right accuracies acquired are bolded.

Male/Female	PT (%)	f	OT (%)	f
SVM	**95,00**	0	80,83	0
Decision tree	84,17	0	**92,50**	0
2-nearest neighbour	91,67	0	87,50	0
4-nearest neighbour	94,17	0	**92,50**	0
Left/Right	PT (%)	f	OT (%)	f
SVM	**65,83**	4,17e-6	55,83	4,62e-1
Decision tree	47,50	1	**57,50**	1,68e-1
2-nearest neighbour	45,83	1	53,33	9,10e-1
4-nearest neighbour	50,00	1	49,17	1

Transportation of subjects 4 and 9

Table 11.4 shows the cross-validation classification accuracy for subjects 4 and 9 which gave the best results for left/right classification. SVM, decision tree and *k*-Nearest Neighbour are used for both male/female and left/right classification.

Table 11.4: Cross-validation classification accuracy for both male/female and left/right classification with subjects 4 and 9. $f = 0$ means that the result is statistically significant at a 0.05 level and $f = 1$ means that the result could be due to chance. The highest male/female and left/right accuracies acquired are bolded.

Male/Female	PT (%)	*f*	OT (%)	*f*
SVM	42,50	1	55,00	6,42e-1
Decision tree	55,83	4,62e-1	**61,67**	2,26e-3
2-nearest neighbour	**81,67**	0	53,33	9,10e-1
4-nearest neighbour	79,17	0	59,17	3,96e-2
Left/Right	PT (%)	*f*	OT (%)	*f*
SVM	65,00	1,69e-5	52,50	9,69e-1
Decision tree	55,00	6,42e-1	52,50	9,69e-1
2-nearest neighbour	**85,00**	0	**85,83**	0
4-nearest neighbour	75,83	5,55e-16	74,17	5,45e-14

Transportation of subjects 5 and 17

Table 11.5 shows the cross-validation classification accuracy for subjects 5 and 17 which were selected randomly. SVM, decision tree and *k*-Nearest Neighbour were used for both male/female and left/right classification.

Table 11.5: Cross-validation classification accuracy for both male/female and left/right classification with subjects 5 and 17. $f = 0$ means that the result is statistically significant at a 0.05 level and $f = 1$ means that the result could be due to chance. The highest male/female and left/right accuracies acquired are bolded.

Male/Female	PT (%)	*f*	OT (%)	*f*
SVM	73,33	4,79e-13	**77,50**	0
Decision tree	61,67	2,26e-3	69,17	7,39e-9
2-nearest neighbour	75,83	5,55e-16	72,50	3,87e-12
4-nearest neighbour	**83,33**	0	73,33	4,79e-13
Left/Right	PT (%)	*f*	OT (%)	*f*
SVM	55,00	6,42e-1	45,83	1
Decision tree	55,83	4,62e-1	39,17	1
2-nearest neighbour	**80,00**	0	**74,17**	5,45e-14
4-nearest neighbour	75,83	5,55e-16	68,33	4,02e-8

Transportation of subjects 1 and 10

Table 11.6 shows the cross-validation classification accuracy for subjects 1 and 10 which were selected randomly. SVM, decision tree and k-Nearest Neighbour were used for both male/female and left/right classification.

Table 11.6: Cross-validation classification accuracy for both male/female and left/right classification with subjects 1 and 10. $f = 0$ means that the result is statistically significant at a 0.05 level and $f = 1$ means that the result could be due to chance. The highest male/female and left/right accuracies acquired are bolded.

Male/Female	PT (%)	f	OT (%)	f
SVM	78,33	0	62,50	7,41e-4
Decision tree	74,17	5,45e-14	69,17	7,39e-9
2-nearest neighbour	80,00	0	**70,00**	1,26e-9
4-nearest neighbour	**82,50**	0	**70,00**	1,26e-9
Left/Right	PT (%)	f	OT (%)	f
SVM	**88,33**	0	**59,17**	3,96e-2
Decision tree	48,33	1	50,83	9,98e-1
2-nearest neighbour	53,33	9,10e-1	**59,17**	3,96e-2
4-nearest neighbour	50,00	1	56,67	2,96e-1

11.3 Transportation of three subjects

Several subjects were tested and the classification accuracy after the transportation differed greatly depending on which subjects were used. The thesis presents the best results for male/female classification acquired by subjects 2, 7 and 12, and the best results for left/right classification acquired by subjects 4, 9 and 18. Two results are also presented with randomized subjects.

Transportation of subjects 2, 7 and 12

Table 11.7 shows the cross-validation classification accuracy for subjects 2, 7 and 12 which gave the best results for male/female classification. SVM, decision tree and *k*-Nearest Neighbour were used for both male/female and left/right classification. Visualization through t-SNE is presented in the Appendix.

Table 11.7: Cross-validation classification accuracy for both male/female and left/right classification with subjects 2, 7 and 12. $f = 0$ means that the result is statistically significant at a 0.05 level and $f = 1$ means that the result could be due to chance. The highest male/female and left/right accuracies acquired are bolded.

Male/Female	PT (%)	f	OT (%)	f
SVM	**93,89**	0	58,33	1,45e-2
Decision tree	81,67	0	70,56	7,44e-15
2-nearest neighbour	92,78	0	67,22	1,68e-10
4-nearest neighbour	92,22	0	**72,78**	0
Left/Right	**PT (%)**	f	**OT (%)**	f
SVM	62,22	2,15e-5	60,56	4,69e-4
Decision tree	47,22	1	46,67	1
2-nearest neighbour	**77,22**	0	**77,78**	0
4-nearest neighbour	71,67	1,11e-16	70,56	7,44e-15

Transportation of subjects 4, 9 and 18

Table 11.8 shows the cross-validation classification accuracy for subjects 4, 9 and 18 which gave the best results for left/right classification. SVM, decision tree and *k*-Nearest Neighbour were used for both male/female and left/right classification.

Table 11.8: Cross-validation classification accuracy for both male/female and left/right classification with subjects 4, 9 and 18. $f = 0$ means that the result is statistically significant at a 0.05 level and $f = 1$ means that the result could be due to chance. The highest male/female and left/right accuracies acquired are bolded.

Male/Female	PT (%)	f	OT (%)	f
SVM	56,11	1,81e-1	63,89	6,50e-7
Decision tree	52,78	9,11e-1	**73,33**	0
2-nearest neighbour	**84,44**	0	60,56	4,69e-4
4-nearest neighbour	83,89	0	60,00	1,19e-3
Left/Right	PT (%)	f	OT (%)	f
SVM	**71,67**	1,11e-16	**57,22**	5,83e-2
Decision tree	52,22	9,61e-1	47,22	1
2-nearest neighbour	59,44	2,89e-3	53,33	8,26e-1
4-nearest neighbour	58,33	1,45e-2	56,11	1,81e-1

Transportation of subjects 1, 10 and 12

Table 11.9 shows the cross-validation classification accuracy for subjects 1, 10 and 12 which were selected randomly. SVM, decision tree and k-Nearest Neighbour were used for both male/female and left/right classification.

Table 11.9: Cross-validation classification accuracy for both male/female and left/right classification with subjects 1, 10 and 12. $f = 0$ means that the result is statistically significant at a 0.05 level and $f = 1$ means that the result could be due to chance. The highest male/female and left/right accuracies acquired are bolded.

Male/Female	PT (%)	f	OT (%)	f
SVM	**81,11**	0	57,78	3,00e-2
Decision tree	67,22	1,68e-10	**62,22**	2,15e-5
2-nearest neighbour	70,56	7,44e-15	37,78	1
4-nearest neighbour	77,78	0	45,00	1
Left/Right	PT (%)	f	OT (%)	f
SVM	**67,22**	1,68e-10	58,33	1,45e-2
Decision tree	38,89	1	**60,00**	1,19e-3
2-nearest neighbour	27,78	1	31,11	1
4-nearest neighbour	35,56	1	36,67	1

Transportation of subjects 2, 6 and 14

Table 11.10 shows the cross-validation classification accuracy for subjects 2, 6 and 14 which were selected randomly. SVM, decision tree and k-Nearest Neighbour were used for both male/female and left/right classification.

Table 11.10: Cross-validation classification accuracy for both male/female and left/right classification with subjects 2, 6 and 14. $f = 0$ means that the result is statistically significant at a 0.05 level and $f = 1$ means that the result could be due to chance. The highest male/female and left/right accuracies acquired are bolded.

Male/Female	PT (%)	f	OT (%)	f
SVM	**80,56**	0	50,00	1
Decision tree	62,78	7,01e-6	**60,00**	1,19e-3
2-nearest neighbour	66,67	7,52e-10	34,44	1
4-nearest neighbour	75,00	0	42,22	1
Left/Right	PT (%)	f	OT (%)	f
SVM	**93,33**	0	**46,67**	1
Decision tree	52,78	9,11e-1	38,89	1
2-nearest neighbour	52,22	9,61e-1	45,00	1
4-nearest neighbour	47,22	1	40,56	1

Transportation of all 18 subject

Table 11.10 shows the cross-validation classification accuracy for all 18 subjects with SVM, decision tree and k-Nearest Neighbour for both male/female and left/right classification.

Table 11.11: Cross-validation classification accuracy for both male/female and left/right classification with all 18 subjects. $f = 0$ means that the result is statistically significant at a 0.05 level and $f = 1$ means that the result could be due to chance. The highest male/female and left/right accuracies acquired are bolded.

Male/Female	PT (%)	f	OT (%)	f
SVM	67,41	0	52,78	1,20e-1
Decision tree	56,76	2,14e-9	52,41	2,49e-1
2-nearest neighbour	**80,19**	0	**54,26**	8,01e-4
4-nearest neighbour	79,35	0	53,33	2,30e-2
Left/Right	PT (%)	f	OT (%)	f
SVM	61,20	0	53,43	1,71e-2
Decision tree	54,54	2,50e-4	51,94	5,23e-1
2-nearest neighbour	**78,98**	0	**74,44**	0
4-nearest neighbour	71,20	0	69,82	0

12

Discussion

12.1 Classification accuracy for each subject before transportation

Both PT and OT seem to work as they capture and preserve the structure of the data before and after the transportation, shown with subjects 2 and 7 in the Appendix. The problem is that the results differed greatly depending on which subjects were used and if it is not possible to distinguishing the separation before the transportation it wont exist after the transportation.

Male/female attention steering

Figure 11.1 shows the male/female cross-validation classification accuracy for each subject before the transportation. The results differed greatly depending on the subjects, but they were all (except two with decision tree) above the statistically significant level of 60%. It seems that SVM gives the best accuracy and subjects number 2, 7, 12, 15 and 18 reached the highest values. Table 11.1 shows the classification accuracy for each subject with SVM, which all are statistically significant. This indicates that it is possible to detect the differences in pitch and timbre with EEG data and covariance matrices and that the distinguishing is easier for some subjects.

It would have been very interesting to know the gender of listener. Several behavioural and neural activation studies show that there is a difference between male and female listener's perception of voices. The study *Gender differences in the temporal voice areas* by Marle-Marie Ahrens *et. al.*

shows that there exist regions in the brain that are locally activated for female listeners in a classification task between vocal and nonvocal sound sources. These regions were not activated for the male listeners. Generally, the females in the study performed a better classification accuracy compared to the male listeners, however if this is due to the activated regions is at present unknown. The distinguishing between vocal sounds (male and female speeches from infants, children, adults and elderly) and nonvocal sounds (laughs, cries, sighs and coughs) might be an easier classification task than the one between male and female voices. The subjects in the study by Marle-Marie Ahrens *et. al.* listened to one sound at the time, without the attended/unattended problem, which is another difference between this thesis and the results in the study. However, it would still be interesting to investigate if the gender of the listener differed between subjects with really good and not as good classification accuracy.

Studies show that the attention steering problem is easier when the voices are of different gender, as in the DTU-dataset [Treisman and Phil, 1964]. It would have been interesting to investigate how much the results would have differed in this thesis if the two speech streams were from the same gender, which generally decreases the pitch impact on the attention steering. One might also take it one step further and use the same voice but different stories to really study the differences between attended and unattended sound sources. Even though it is not a real-life situation, it might spread some light over the neural functions in the brain.

It would also have been interesting to know the results on the multiple-choice questionnaires the subjects made after each trial. Were there any significant difference between subjects with really good and not as good classification accuracy?

Left/right attention steering

Figure 11.2 shows the left/right cross-validation classification accuracy for each subject before the transportation. The results differed greatly depending on the subjects, and between the classification methods were SVM outperforms the other two. The accuracies are generally lower than for male/female classification and several subjects did not give statistically significant results. This indicates that attention location steering is a harder problem than the male/female classification problem. Table 11.2 shows the classification accuracies for all subjects with SVM where the highest value reached 75% for subject 18.

The results differed greatly between the subjects in both the male/female and the left/right classification problems. One reason for this could be that they all used the same electrode head cap when recording the EEG data. The head cap has a specific size which may not fit all individuals. For example, subjects 2, 12 and 18 gave good results for both classification problems while subjects 1, 5 and 6 have lower classification accuracies in both problems.

One reason why the results are not always good with left/right attention steering could come from the nature of *volume conduction*. Volume conduction is used when the electrodes are not in contact with the actual source generator. This occurs in EEG data since the electrodes are placed outside the head while the neuron firing happens inside the brain. However, the volume conduction does not only transport the electrical pulses from firing neurons to the nearest electrode, but spread it in all directions. This means that the electrical pulses from a single neuron is registered in many, and sometimes all, electrodes on the head [Carvalhaes and Acacio de Barros, 2015]. This attribute of volume conduction complicates the measure of attention steering EEG data, especially in the left/right classification problem where the signals already are crossed several times between the two hemispheres. One way to solve the volume conduction problem is with a spatial filter on the EEG data with makes it possible to extract more information from the data. One of the most common spatial filter is called *Surface Laplacian*. Unfortunately, due to lack of time, a Surface Laplacian spatial filter was not used in this thesis. It would be very interesting to investigate if it would have made a significant difference and is a relevant field for future work.

12.2 Parallel transport vs Optimal transport

Both parallel transport and optimal transport have been proven to work on EEG data with motor cues [Yair et al., 2019; Yair et al., 2020]. They seem to work with the DTU-dataset in the sense that when a structure exists (such as male/female separation in subjects 2 and 7 which is presented in the t-SNE visualization in the Appendix), the transportation preserves this structure. The best male/female classification accuracy were from subjects 2 and 7 in Table 11.3 reached 95,00% for PT with SVM and 92,50% for OT with decision tree and 4-nearest neighbour. Just like the classification for each subject before the transportation, both PT and OT generally gave lower accuracies for left/right classification, which for some subjects are not statistically significant. However, the best results were high: 93,33% with PT and SVM (Table 11.10) and 85,83% with OT and 2-nearest neighbour (Table 11.4). Transportation of three subjects in Tables 11.7-11.10 gave classification accuracies around the same values as for transportation of two values. This indicates that the number of subjects transported is irrelevant, and the results depend more on which subjects are used and how well the datapoints were separated before the transportation. Generally, PT gave slightly better results than OT.

Table 11.11 shows the cross-validation classification accuracies after PT and OT for all 18 subjects. It gave statistically significant results for male/female classification which reached above 80% with PT. Left/right classification gave a bit lower accuracies but were statistically significant for 2-nearest neighbour with 78,98% for PT and 74,44%. This indicates that it is possible to use both PT and OT in attention steering problems with statistically significant results.

12.3 Classification methods

SVM, *k*-nearest neighbour and decision trees are all widely used classification methods. It is not often obvious which method to use since their performance differ between various problems, especially for high-dimensional data such as the DTU-dataset. The results in this thesis show that the best classification methods for attention steering between male/female and left-/right with the DTU-dataset seem to be SVM and *k*-nearest neighbour. Two *k*-values are presented in the results ($k = 2$ and $k = 4$) since they in most cases gave be highest accuracies. $k = 3$ gave similar results, but higher values of k seem to decrease the accuracy.

Decision trees gave, in almost all cases, the lowest accuracy and is not recommended to be used in future work with similar problems. Some reasons for this could be that decision trees tend to overfit, are sensitive to small data disturbances (which would create a completely different tree) and that they usually find the local optima at the current level - not necessarily the global optima [*Decision Trees – Tree Development and Scoring* 2020].

Two other classification methods were also tested: Regression trees and a pattern recognition neural network with different numbers of neurons and hidden layers. However, the regression trees gave poor results and the pattern recognition neural network gave accuracies in the same values as SVM, but the training time was much longer. Therefore, these two methods are not presented in the thesis.

12.4 Pros and cons with EEG data

EEG data has many advantages, such as a very low cost compared to other methods, widely available and easy to use and it is able to capture both the radial and the tangential components of the signal. There are however some disadvantages that cannot be overlooked, especially when considering the development of intelligent hearing aids. When collecting the EEG data, the subjects wear a head cap with a smooth layer of gel between the electrodes and the head. It is not reasonable in real-life situations that the hearing aids are connected to a head cap with gel. However, EEG analysis can be used in the research of understanding how the brain solves the *cocktail party problem*, but different mechanisms have to be considered when developing intelligent hearing aids.

12.5 The use of covariance matrices

This thesis uses covariance matrices of the EEG data and the two speech streams for both classification and plotting. Covariance matrices are powerful tools when working with time series and they have been used in earlier studies for classification problems with EEG data. For example, both parallel transport and optimal transport have used covariance matrices on the BCI Competition IV dataset 2a with good results [Yair et al., 2019; Yair et al., 2020]. However, in these dataset the subjects had four cued motor imageries: left hand, right hand, feet and tongue, and it might be greater differences in the brain wave activity for different motor movements than between attention steering to a left or right sound source.

One problem with covariance matrices is that they do not capture the time dependence between the EEG signals. Especially in the left/right case, where the time difference between the ears and signal crossings in the brain, are crucial mechanisms for the brain to distinguishing the sound location, important information is probably lost in the covariance matrices. Some domain adaptation methods of interest in future attention steering problems which do not use covariance matrices are *Transfer Component Analysis* (TCA), *Subspace Alignment* (SA) and *Information Theoretical Learning* (ITL).

12.6 The goal of the thesis

The goal of the thesis is to use the two domain adaptation methods *parallel transport* and *optimal transport* to transport data from several subjects and thereafter create a classifier which gets an accuracy above the level of chance. Two classification problems were considered:

- Attention to male voice vs female voice
- Attention to left side vs right side

The accuracy for male/female classification were above the level of chance for almost all combinations of subjects used. However, the accuracy for left/right classification were only above the level of chance for transportation of a few subjects. Interestingly, transportation of all 18 subjects gave an accuracy above the level of chance with 2-nearest neighbour for both male/female and left/right classification. The results were not as good as when transporting only the "best" subjects, but it indicates that the "good" subjects outweigh the "bad" subjects. With this in mind, the goal can be considered to be achieved even though more work needs to be done, for example with a Surface Laplacian spatial filter, to improve the results.

12.7 Future work

The two domain adaptation methods used in this thesis are quite similar since they both use covariance matrices on the Riemannian manifold. There might exist domain adaptation methods built which capture the time dependence between the EEG signals which would work better for left-/right attention steering problems. It would be interesting to try PT and OT on a different dataset where the two speakers have the same gender, which would focus more on the left/right attention steering problem. Even though reverberation is added to some trails, which can be seen as background noise, are added there might be essential to add more noise to really match a *cocktail party situation*. Another aspect of the DTU-dataset is that all subjects have normal hearing. The data might not look the same for subjects with hearing loss and it would be interesting to investigate the differences, both with and without the use of hearing aids. Future work could also include how the use of a Surface Laplacian spatial filter would affect the results.

13 Conclusions

Attention steering is a very complex problem which involves several parts of the brain. Different sound sources can be distinguished with pitch, loudness and timbre analysis and the brain uses time-of-arrival, SPL and spectral shape to determine the voice location. The classification accuracy of EEG signals differed greatly depending on which subjects were used. It seems to be easier to distinguishing between male/female voices than between the location of the sound sources, since these classification accuracies generally were higher and almost always above the level of chance. Meanwhile, several subjects did not give statistically significant results for left/right classification. Table 11.11 shows the transportation of all 18 subjects, were PT gave an accuracy above 80% for male/female classification and above 77% for left/right classification for 2-nearest neighbour. This indicates that the number of "good" subjects outweigh the number of "bad" subjects in the DTU-dataset and that it is possible to distinguishing between male/female and left/right attention steering with PT and OT.

EEG data and covariance matrices seem to work for domain adaptation in attention steering problems, but there might exist other methods which are more effective. Future work would also include data with added noise, a Surface Laplacian spatial filter, same gender of the two storytellers and more focus on left/right attention steering. Much work remains in solving the *cocktail party problem* in intelligent hearing aids.

14

Appendix

14.1 Visualization with t-SNE

Transportation of subjects 2 and 7

Male/Female classification Figures 14.1-14.3 show the t-SNE visualization for subjects 2 and 7 before transportation, after parallel transport and after optimal transport.

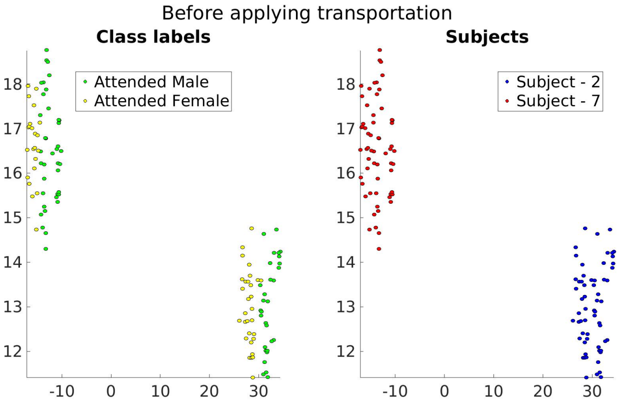


Figure 14.1: Male/female: Before applying transportation of subjects 2 and 7

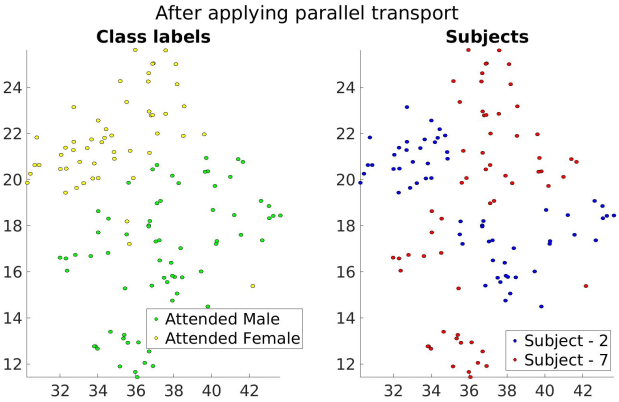


Figure 14.2: Male/female: After parallel transport of subjects 2 and 7.

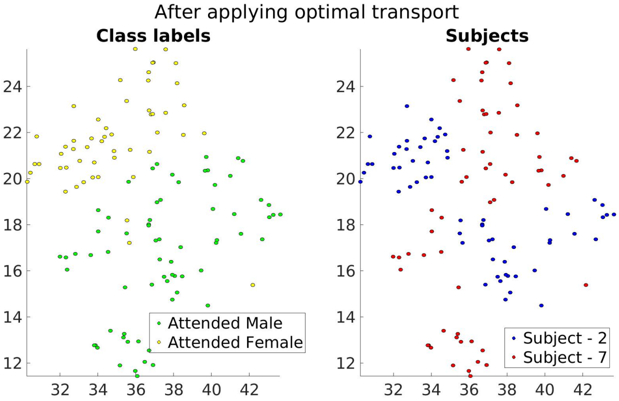


Figure 14.3: Male/female: After optimal transport of subjects 2 and 7.

Left/right classification classification Figures 14.4-14.6 show the t-SNE visualization for subjects 2 and 7 before transportation, after parallel transport and after optimal transport.

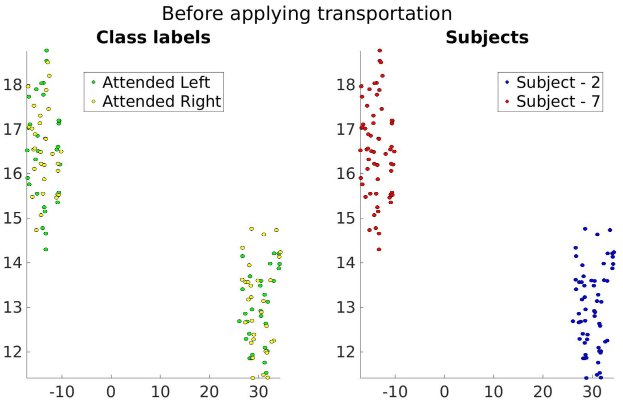


Figure 14.4: Left/right: Before applying transportation of subjects 2 and 7

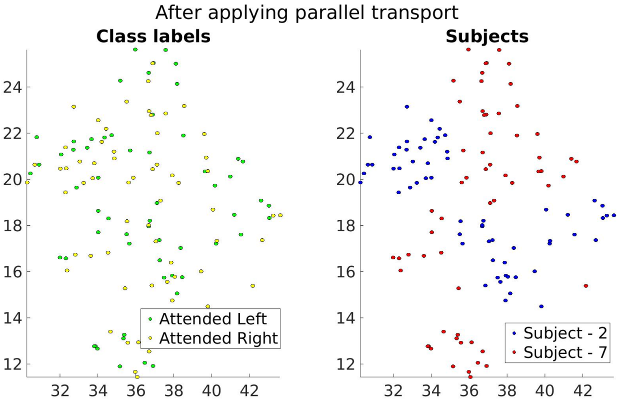


Figure 14.5: Left/right: After parallel transport of subjects 2 and 7.

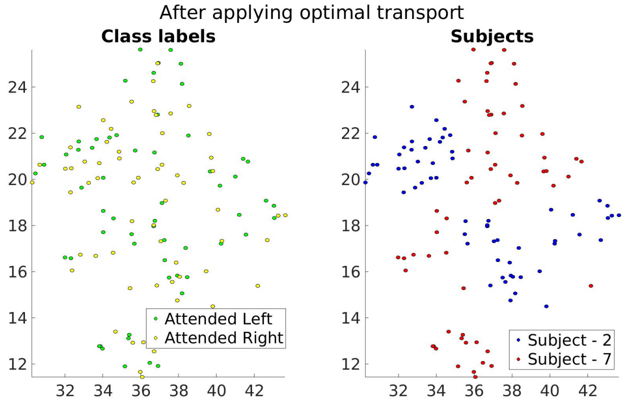


Figure 14.6: Left/right: After optimal transport of subjects 2 and 7.

Transportation of subjects 2, 7 and 12

Male/Female classification Figures 14.7-14.9 show the t-SNE visualization for subjects 2, 7 and 12 before transportation, after parallel transport and after optimal transport.

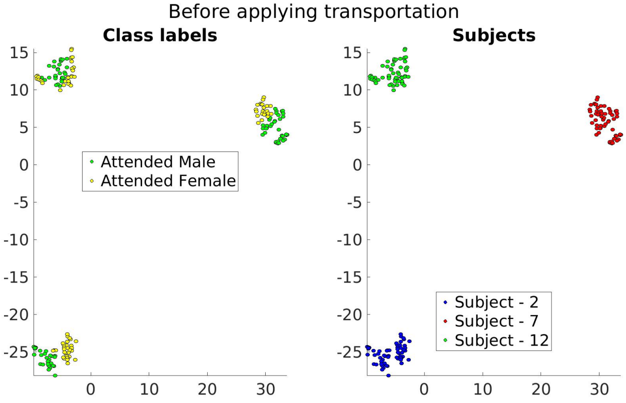


Figure 14.7: Male/female: Before applying transportation of subjects 2, 7 and 12

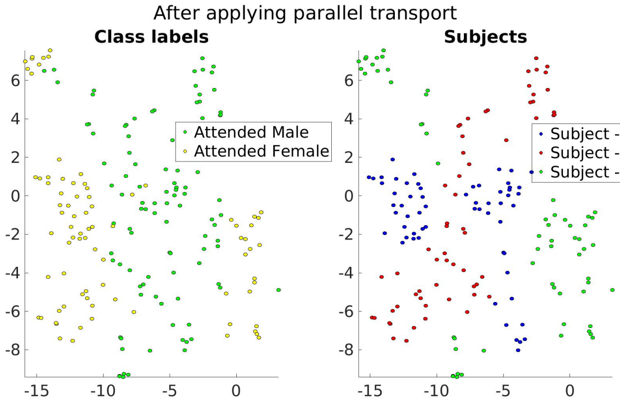


Figure 14.8: Male/female: After parallel transport of subjects 2, 7 and 12.

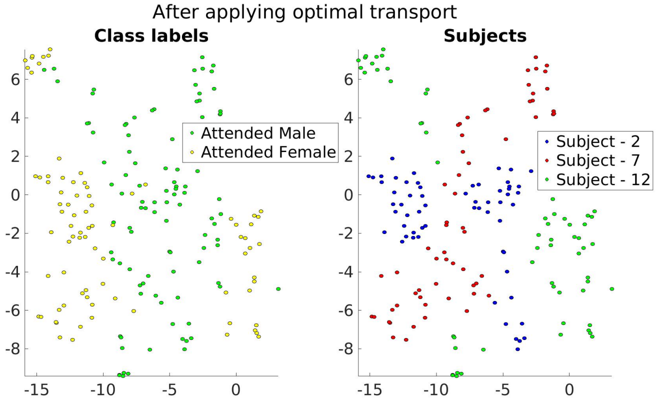


Figure 14.9: Male/female: After optimal transport of subjects 2, 7 and 12.

Left/right classification classification Figures 14.10-14.12 show the t-SNE visualization for subjects 2, 7 and 12 before transportation, after parallel transport and after optimal transport.

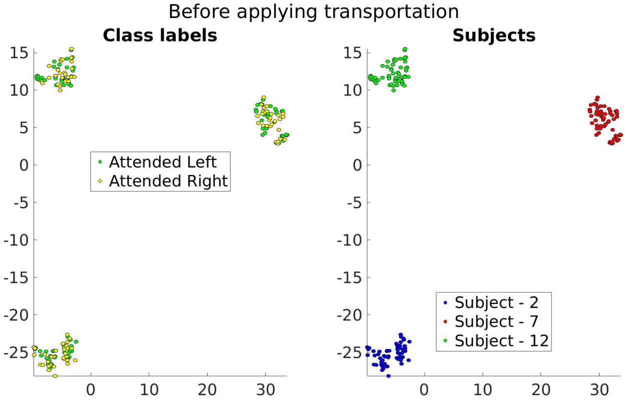


Figure 14.10: Left/right: Before applying transportation of subjects 2, 7 and 12.

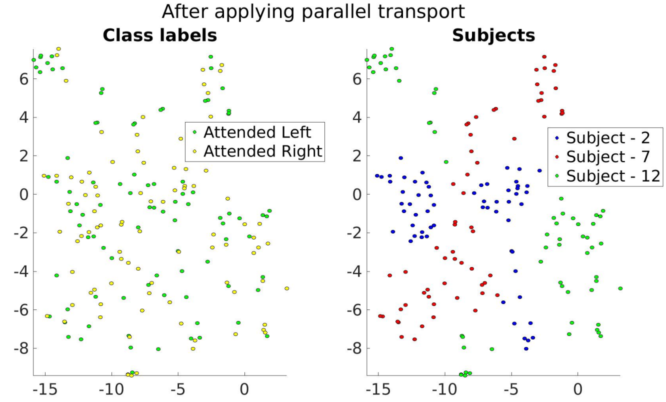


Figure 14.11: Left/right: After parallel transport of subjects 2, 7 and 12.

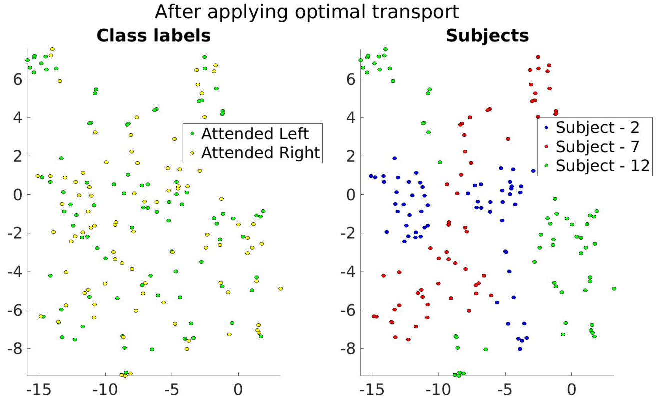


Figure 14.12: Left/right: After optimal transport of subjects 2, 7 and 12.

Bibliography

Albuquerque, I., J. Monteiro, O. Rosanne, A. Tiwari, J.-F. Gagnon, and T. H. Falk (2019). “Cross-subject statistical shift estimation for generalized electroencephalography-based mental workload assessment”. DOI: arXiv:1906.08823.

Alickovic, E., T. Lunner, F. Gustafsson, and L. Ljung (2019). “A tutorial on auditory attention identification methods”. *Frontiers in Neuroscience* **13**:153. DOI: 10.3389/fnins.2019.00153.

Asgarian, A. (2020). *An introduction to transfer learning*. URL: https://medium.com/georgian-impact-blog/transfer-learning-part-1-ed0c174ad6e7 (visited on 2020-05-25).

Awad, M. and R. Khanna (2015). “Support vector machines for classification”. In: pp. 39–66. ISBN: 978-1-4302-5989-3. DOI: 10.1007/978-1-4302-5990-9_3.

Barachant, A., S. Bonnet, M. Congedo, and C. Jutten (2013). “Classification of covariance matrices using a riemannian-based kernel for bci applications”. *Neurocomputing* **112**, pp. 172–178. DOI: https://doi.org/10.1016/j.neucom.2012.12.039.

Ben-David, S., J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan (2010). “A theory of learning from different domains”. *Mach Learn* **79**, pp. 151–175. DOI: 10.1007/s10994-009-5152-4.

BioSemi headcap (2020). URL: https://www.biosemi.com/headcap.htm (visited on 2020-05-24).

Burle, B., L. Spieser, C. Roger, L. Casini, T. Hasbroucq, and F. Vidala (2015). “Spatial and temporal resolutions of eeg: is it really black and white? a scalp current density view”. *Int J Psychophysiol* **97**:3, pp. 210–220. DOI: 10.1016/j.ijpsycho.2015.05.004.

Calinon, S. (2020). “Gaussians on riemannian manifolds for robot learning and adaptive control”. DOI: `arXiv:1909.05946`.

Carvalhaes, C. and J. Acacio de Barros (2015). “The surface laplacian technique in eeg: theory and methods”. *International Journal of Psychophysiology* **97**, pp. 174–188. DOI: `doi.org/10.1016/j.ijpsycho.2015.04.023`.

Combrissona, E. and K. Jerbi (2015). “Exceeding chance level by chance: the caveat of theoretical chance levels in brain signal classification and statistical assessment of decoding accuracy”. *Journal of Neuroscience Methods* **250**, pp. 126–136. DOI: `doi.org/10.1016/j.jneumeth.2015.01.010`.

Cuturi, M. (2013). “Sinkhorn distances: lightspeed computation of optimal transport”. *Advances in Neural Information Processing Systems* **26**, pp. 2292–2300. DOI: `arXiv:1306.0895[stat.ML]`.

Decision Trees – Tree Development and Scoring (2020). URL: `https://www.edupristine.com/blog/decision-trees-development-and-scoring` (visited on 2020-06-16).

Euclidean geometry (2020). URL: `https://www.britannica.com/science/Euclidean-geometry` (visited on 2020-03-25).

Fuglsang, S. A., T. Dau, and J. Hjortkjær (2017). “Noise-robust cortical tracking of attended speech in real-world acoustic”. *NeuroImage* **156**, pp. 435–444. DOI: `https://dx.doi.org/10.1016/j.neuroimage.2017.04.026`.

Fuglsang, S. A., D. D. Wong, and J. Hjortkjær (2018). “Eeg and audio dataset for auditory attention decoding”. *Zenodo*. DOI: `10.5281/zenodo.1199011`.

Han, C., J. O’Sullivan, Y. Luo, J. Herrero, A. D. Mehta, and N. Mesgarani (2019). “Speaker-independent auditory attention decoding without access to clean speech sources”. *Science Advances* **5**:5. DOI: `10.1126/sciadv.aav6134`.

Hinton, G. and S. Roweis (2002). “Stocharstic neighbor embedding”. *Advances in Neural Information Processing Systems* **15**, pp. 833–840.

How to Build A Data Set For Your Machine Learning Project (2020). URL: `https://towardsdatascience.com/how-to-build-a-data-set-for-your-machine-learning-project-5b3b871881ac` (visited on 2020-02-14).

Jerger, J. and J. Martin (2004). “Hemispheric asymmetry of the right ear advantage in dichotic listening”. *Hearing Research* **198**:1-2, pp. 125–136. DOI: doi.org/10.1016/j.heares.2004.07.019.

Kouw, W. M. and M. Loog (2019). “Technical report: an introduction to domain adaptation and transfer learning”. DOI: arXiv:1812.11806.

Kouw, W. M. and M. Loog (2018). “Technical report: an introduction to domain adaptation and transfer learning”.

Left-Brain Hemisphere (2020). URL: https://www.encyclopedia.com/medicine/encyclopedias-almanacs-transcripts-and-maps/left-brain-hemisphere (visited on 2020-05-18).

Li, W., J. E. Cerise, Y. Yang, and H. Han (2017). “Application of t-sne to human genetic data”. *Journal of Bioinformatics and Computational Biology* **15**:4. ISSN: 1757-6334. DOI: 10.1142/S0219720017500172.

Maaten, L. van der and G. Hinton (2008). “Visualizing data using t-sne”. *Journal of Machine Learning Research* **9**, pp. 2579–2605.

Merriam-Webster (2020). “Dictionary”. DOI: https://www.merriam-webster.com/dictionary/loudness.

Miller, C. D., V. E. Heeren, J. Hornsby, and C. Heeren (2014). *Mathematical Ideas*. Pearson.

O’Sullivan, J. A., A. J. Power, N. Mesgarani, S. Rajaram, J. J. Foxe, B. G. Shinn-Cunningham, M. Slaney, S. A. Shamma, and E. C. Lalor (2015). “Attentional selection in a cocktail party environment can be decoded from single-trial eeg”. *Cerebral Cortex* **25**:7, pp. 1697–1706. DOI: 10.1093/cercor/bht355.

Patricia, N. and B. Caputo (2014). “Learning to learn, from transfer learning to domain adaptation: a unifying perspective”. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1442–1449. ISSN: 1063-6919. DOI: 10.1109/CVPR.2014.187.

Paul, B. T., M. Uzelac, E. Chan, and A. Dimitrijevic (2020). “Poor early cortical differentiation of speech predicts perceptual difficulties of severely hearing-impaired listeners in multi-talker environments”. *Scientific reports* **10**:6141. DOI: doi.org/10.1038/s41598-020-63103-7.

Poeppel, D. (2003). “The analysis of speech in different temporal integration windows: cerebral lateralization as ‘asymmetric sampling in time’”. *Speech Communication* **41**:1, pp. 245–255. DOI: doi.org/10.1016/S0167-6393(02)00107-3.

Raza, H., H. Cecotti, Y. Li, and G. Prasad (2016). “Adaptive learning with covariate shift-detection for motor imagery-based brain–computer interface”. *Soft Computing* **20**, pp. 3085–3096. DOI: doi.org/10.1007/s00500-015-1937-5.

Razaa, H., D. Ratheeb, S.-M. Zhouc, H. Cecottid, and G. Prasadb (2019). “Covariate shift estimation based adaptive ensemble learning for handling non-stationarity in motor imagery related eeg-based brain-computer interface”. *Neurocomputing* **343**, pp. 154–166. DOI: doi.org/10.1016/j.neucom.2018.04.087.

Riemannian geometry (2020). URL: https://www.britannica.com/science/Riemannian-geometry (visited on 2020-03-25).

Right-Brain Hemisphere (2020). URL: https://www.encyclopedia.com/medicine/encyclopedias-almanacs-transcripts-and-maps/right-brain-hemisphere (visited on 2020-05-18).

Risoud, M., J. Hanson, F. Gauvrith, C. Renard, P. Lemestre, N. Bonne, and C. Vincent (2018). “Sound source localization”. *European Annals of Otorhinolaryngology, Head and Neck Diseases* **135**:4, pp. 259–264. DOI: doi.org/10.1016/j.anorl.2018.04.009.

Risset, J.-C. and D. L. Wessel (1982). *The Psychology of Music*. Academic Press Series in Cognition and Perception. Chap. 2.

Satheesh Kumar, J. and P. Bhuvaneswari (2012). “Analysis of electroencephalography (eeg) signals and its categorization - a study”. *Procedia Engineering* **38**, pp. 2525–2536. DOI: 10.1016/j.proeng.2012.06.298.

Smith, D. W. and A. Keil (2015). “The biological role of the medial olivocochlear efferents in hearing: separating evolved function from exaptation”. *Frontiers in systems neuroscience* **9**:12. DOI: doi.org/10.3389/fnsys.2015.00012.

Song, Y.-y. and Y. Lu (2015). “Decision tree methods: applications for classification and prediction”. *Shanghai Arch Psychiatry* **27**:2, pp. 130–135. DOI: 10.11919/j.issn.1002-0829.215044.

Spatial hearing loss (2020). URL: https://www.wikiwand.com/en/Spatial_hearing_loss#/Corpus_callosum (visited on 2020-05-17).

Steinberg, D. D. and N. V. Sciarini (2013). *An Introduction to Psycholinguistics*. 2nd ed. Routledge. Chap. 12.

Support Vector Machines for Binary Classification (2020). URL: `https://se.mathworks.com/help/stats/support-vector-machines-for-binary-classification.html` (visited on 2020-06-03).

Tran, V.-T. and A. Aussem (2015). “A practical approach to reduce the learning bias under covariate shift”. *ECML PKDD 2015 - European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, pp. 71–86. DOI: `10.1007/978-3-319-23525-7_5`.

Treisman, A. M. and D. Phil (1964). “Selective attention in man”. *British Medical Bulletin* **20**:1, pp. 12–16. DOI: `doi.org/10.1093/oxfordjournals.bmb.a070274`.

Waves Packed in Envelopes (2020). URL: `https://thatsmaths.com/2018/04/26/waves-packed-in-envelopes/` (visited on 2020-06-03).

Weiss, K., T. M. Khoshgoftaar, and D. Wang (2016). “A survey of transfer learning”. *J Big Data* **3**:9. DOI: `10.1186/s40537-016-0043-6`.

What is Machine Learning? A definition (2020). URL: `https://expertsystem.com/machine-learning-definition/` (visited on 2020-02-14).

Wong, D. D., S. A. Fuglsang, J. Hjortkjær, E. Ceolini, M. Slaney, and A. de Cheveigné (2018). “A comparison of regularization methods in forward and backward models for auditory attention decoding”. *Frontiers in Neuroscience* **12**:531, p. 16. DOI: `doi.org/10.3389/fnins.2018.00531`.

Yair, O., M. Ben-Chen, and R. Talmon (2019). “Parallel transport on the cone manifold of spd matrices for domain adaptation”. *IEEE Transactions on signal processing* **67**:7, pp. 1797–1811. DOI: `10.1109/TSP.2019.2894801`.

Yair, O., F. Dietrich, R. Talmon, and I. G. Kevrekidis (2020). “Optimal transport on the manifold of spd matrices for domain adaptation”. DOI: `arXiv:1906.00616[cs.LG]`.

Zhang, Z. (2016). “Introduction to machine learning: k-nearest neighbors”. *Ann Transl Med* **4**:11. DOI: `10.21037/atm.2016.03.37`.

Zhou, H., F. Wang, and P. Tao (2018). “T-distributed stochastic neighbor embedding (t-sne) method with the least information loss for macromolecular simulations”. *J Chem Theory Comput* **14**:11, pp. 5499–5510. DOI: `10.1021/acs.jctc.8b00652`.

Lund University
Department of Automatic Control
Box 118
SE-221 00 Lund Sweden

Document name
MASTER’S THESIS

Date of issue
July 2020

Document Number
TFRT-6110

Author(s)
Johanna Wilroth

Supervisor
Frida Heskebeck, Dept. of Automatic Control, Lund University, Sweden
Carolina Bergeling, Dept. of Automatic Control, Lund University, Sweden
Bo Bernhardsson, Dept. of Automatic Control, Lund University, Sweden (examiner)

Title and subtitle

Domain Adaptation for Attention Steering

Abstract

A major problem in the development of intelligent hearing aids is often referred to as the cocktail party problem. It describes the remarkable ability of the human brain of filter out unwanted sounds in a noisy environment, while focusing on a single talker or conversation. Without the ability to select and enhance a specific sound source of choice while suppressing the background, the hearing aids generally amplify the volume of everyone in the environment. The problem of knowing which speaker to enhance is unsolved and most people with hearing aids still experience discomfort in noisy environments. This thesis uses EEG data from real-life scenarios where the subjects for each trial listened to one female voice and one male voice at the same time while giving attention to one of the speech streams. The stories were simulated to come from a distance of 2.4m in a direction of ±60° from the listener. Due to both instrumental and human factors, data from different subjects will differ and it is not possible to create a classifier which works on all data. It is said that the data from each subject lives in different domains, and they need to be transported to the same domain in order to be classified together. The transportation is called domain adaptation, and this thesis have used and compared two domain adaptation methods: Parallel transport and Optimal transport. Two different classification problems are considered in this thesis: attention to male voice vs female voice and attention to left side vs right side. The classification accuracy differed greatly depending on which data was used. Generally, the results were better for male/female separation which almost always gave successful results, and the highest classification accuracy reached 95%. Transportation of several subjects for the left/right separation problem did not give results above the level of chance, however the best classification accuracy reached above 93% which is considered a successful result.

Keywords

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title
0280-5316

ISBN

Language
English

Number of pages
1-108

Recipient’s notes

Security classification

http://www.control.lth.se/publications/