KTH Computer Science
and Communication

Brain Pattern Recognition

An evaluation of how the choice of training data affect classification accuracy for inexperienced BCI-users

Ragnhild Karlsson
Mikael Eriksson

Degree Project in Computer Science, DD143X
Supervisor: Pawel Andrzej Herman
Examiner: Örjan Ekeberg

CSC, KTH 2014-04-29

Abstract

Brain Computer Interfaces (BCIs) is a new emerging technique where the user is able to control a computer by thought. One challenge in the development of BCIs is to classify the thoughts of the user. The algorithm used for this purpose needs training data. This study examined how the choice of training data affected the classification accuracy. A previous study (Herman et.al 2008) showed that in sessions where the user have little expericence of BCI there was a clear positive effect of choosing the latest session as training data compared to choosing earlier sesisons. However in general there is a positive effect on performance of having more training data. The objective of this study thus were to determine which of these two strategies produced the best classification accuracy (CA) generalizing between sessions.

The method used was to create an ensemble of classifiers, one for each time sample of a single trial and thereafter using majority vote to decide the class of the trial. Classifiers used were support vector machines (SVMs) and linear discriminant analysis (LDA). For each subject data for 3 sessions were used, labeled A, B and C in chronological order. Session A and B were used as training sets, and session C as the test data. The result in this study could not confirm the results of Herman et. al (2008) instead a slight positive effect of session A (average CA on session C 62%) compared to session B (average CA on session C 58%) could be seen, but in general there was no big difference in CA based on the choice of training data (average CA on session C using training sets: A=62%,B=58%, A&B=61%).

Our results show that it is not always the case that training data recorded closer in time to the test data generate higher CA. Therefore we suggest that it could be a safer choice to use more than the latest session as training data. Still more studies are needed to confirm that using more sessions for training really is better also on data where there is a bigger gap in performance between the latest and earlier sessions.

Table of contents

1. Introduction

Today the standard way for humans to interact with computers is by their hands. However, during the last two decades a growing field that combines neuro and computer science has started to explore the possibilities for humans to interact directly with a computer by their thoughts (Nicolas-Alonso & Gomez-Gil, 2012). Right from the beginning the greatest goal for research in this field has been to develop Brain Computer Interfaces (BCIs) that could be used by people suffering from severe motor disorders. Therefore the development of BCIs has the potential to give persons that are locked into their own bodies a way to interact with their environment by thought-controlled devices such as communication applications and neuro prostheses (Nicolas-Alonso & Gomez-Gil, 2012). Beside medical applications, there is an increasing interest in non medical applications such as computer games and market research. Today in 2014 there are already two pioneer companies offering portable devices for brain activity registration together with APIs to a more general public (Emotiv, 2014; Neurosky 2014). The number of scientific publications about game applications using BCIs has also clearly increased in the last 7 years (Hwang et al., 2013). Altogether this shows that BCIs could evolve to be a tool more commonly used by application developers.

During this rapid development a range of different methods for recording brain activity patterns has been tested but the most commonly used method in BCIs today is electroencephalography (EEG) (Hwang et al., 2013).

To record and also successfully interpret brain activity, BCIs today are limited to use a restricted number of predefined thoughts that the system is trained to recognize, i.e. the systems control signals. One of the most commonly used control signals have been imaginary movements (motor imagery), an example is BCIs where the user could make a binary choice by imagining moving her right or left hand (Hwang et al.,2013).

To enable recognition of these control signals all BCI systems need to contain the following stages: signal acquisition, preprocessing or signal enhancement, feature extraction, and classification (Nicolas-Alonso & Gomez-Gil, 2012). In BCI applications used in a real world environment all these stages must be performed together in real time to enable direct feedback to the user of how her thoughts are interpreted. This is called an online session and in these there are little or no time for tuning the methods that are used. Online sessions have therefore been combined with so called offline sessions, i.e. performing tests of for example different classifiers on already recorded sets of EEG data without the requirement of giving real time feedback (Nicolas-Alonso & Gomez-Gil, 2012).

This paper will have an offline approach and focus on the stage of classification in an EEG based BCI system that uses hand motor imagery as a control signal.

One goal in the construction of classification algorithms for BCIs is that they should have good generalization capacity i.e. that a classification algorithm that is trained on data from one session should perform well when tested on data from a later session. This is a challenge since EEG data from one and the same person performing the exact same mental task is known to differ over time. However if the user gets the chance to train on producing the control signal and gets feedback on how her ongoing thought task is classified, then the patterns often become more stable after some training sessions (Wolpaw, McFarland & Vaughan 2000; Grosse-Wentrup & Schölkopf, 2013).
Clearly this assumes that the classifier used to give feedback generalizes reasonably well also over these early and more unstable sessions. This leads to a choice that must be addressed in the construction of the classifier used. Generally, a classifier performs better the more training data it uses. From this perspective it is best to train the classifier on all available data. But as the patterns become more stable over time it could be the case that training only on the latest session gives better performance. In this paper we want to examine which of these effects that is stronger, in other words:
Which gives the best classification performance, using more training data or train only on data from the latest available session?

2. Objective

In this paper we want to look closer on the problem of choosing training data for classification algorithms used in BCIs, i.e. if it is better to give the classifier all available recorded data from the subject or only the latest session data.

In a previous study (Herman et. al 2008), EEG data from the sensorimotor cortex (C3 and C4 according to the 10-20 EEG lead system) was recorded for 8 subjects, who each performed a common BCI task, i.e. imaginary movement of the right or left hand, during sessions where the subject got direct feedback. In the study the generalization capacity between early training sessions was tested with different classification algorithms. Generalization capacity was evaluated by training and testing on sessions with varying length of time in between. In the evaluation the classifiers was divided into two classes, A and B. The difference between the classes was that the classifiers in class A was trained only on data from the session directly before the test session, and the classifiers in class B were trained only on data from the second last session before the test session. The results showed a clear trend, with the classifiers in class A having a considerable higher classification accuracy (CA) than the classifiers in class B. We got the possibility to use

data from 4 of these subjects in the study of Herman et. al (2008), more specifically data from three successive early training sessions for each of the four subjects.

We intend to use this EEG data to see how the choice of training data affects the performance of some classifier algorithms that is commonly used in BCI applications today. We plan to compare the CA on the last session if the classifier is either trained on the second session or trained on both earlier sessions. This will show if more training data (training on both earlier sessions) or only a session when the user has more BCI experience (second session) leads to better classification performance.

2.1 Hypothesis

We have no prior hypothesis regarding if more training data or training on data from the latest session would give better classification performance. However we expect an observable effect where classification performance in general is higher when testing on data closer in time to the training data.

3. Background

This background first explains the basics of EEG to record brain activity (section 3.1), followed by theory on how to perform feature extraction on these recorded patterns of imaginary movements (section 3.2). Then follows a section discussing factors that create possible variability in BCI performance (section 3.3), and especially the training effect (section 3.3.1). This background then ends with theory of classification algorithms (section 3.4) and two sections introducing the two most common classification algorithms used for BCIs, Support Vector Machines (SVM, section 3.41) and Linear Discriminant Analysis (LDA, section 3.42) (Hwang, Kim, Choi & Im, 2013).

3.1 Electroencephalography and signal acquisition

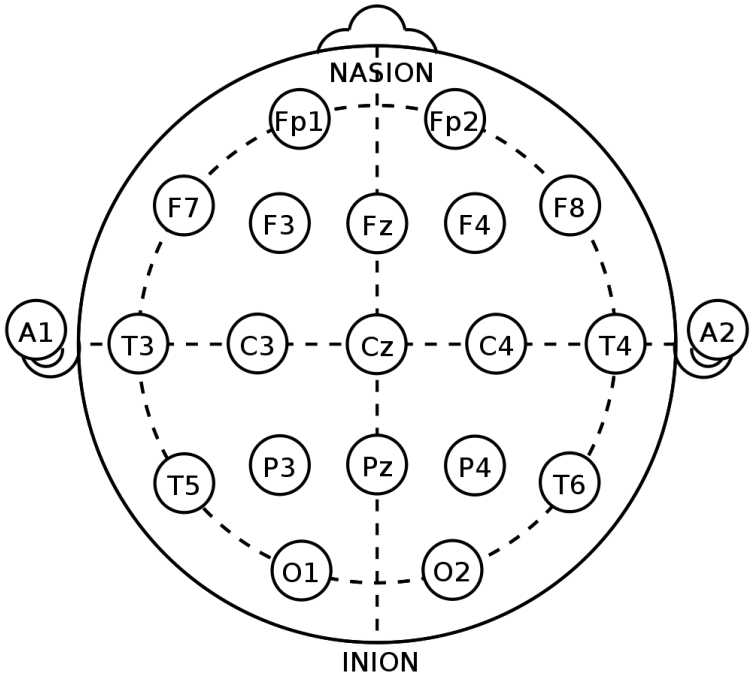

Figure 1: Illustration of the international standard for electrode placement (10-20) on the scalp of the subject.

EEG is recording electrical currents from the scalp with electrodes placed at certain chosen locations, usually according to the international standard for electrode placement for EEG, called 10-20 (Niedermeyer & da Silva, 2005, p.140). The signal that is recorded is the potential difference between a reference electrode and the active electrode, extracted with the help of a third electrode referred to as the ground electrode. The third electrode is used to measure the differential voltage between the active electrode and the reference electrode (Nicolas-Alonso & Gomez-Gil, 2012). As there is several layers between the electrodes and the actual brain tissue, EEG is sensitive to other disturbing currents, for example muscle activity and exterior objects such as power lines, creating possible unwanted signals called artifacts in the recorded pattern. This leads to a situation where parts of recorded data often need to be rejected in order to obtain an acceptable quality (Nicolas-Alonso & Gomez-Gil, 2012). The general resolution of the data is also affected by the material of the electrodes being used and if conductive gel is applied to the subject's skin (Nicolas-Alonso & Gomez-Gil, 2012).

3.2 Brain activity patterns during motor imagery - Feature extraction

The point of interest when recording brain activity through EEG is patterns extracted from different electrodes in different frequency ranges. Well known EEG rhythms are therefore

labeled after location and frequency range (Niedermeyer & da Silva, 2005, p.167). When studying motor imagery, the most common choice of EEG-rhythms to study are so called beta-waves and mu-waves recorded from sensory motor cortex (Nicolas-Alonso et al.,2012). Beta rhythm is defined as EEG-rhythms recorded in the frequency range 14-30 Hz. Mu rhythms are rhythms recorded from the sensorimotor cortex in the frequency range 8-13 Hz (Niedermeyer & da Silva, 2005, p.167,175). In awake relaxed persons are one or both of these rhythms often (but not always) present. What makes Beta and Mu rhythms interesting for BCI control is that it has been shown that motor imagery tasks can produce so called event related desynchronization (ERD) and event related synchronization (ERS) in these rhythms. ERD is defined as a decrease in the amplitude of the rhythm and ERS is the opposite i.e., an increase in the amplitude of the rhythm (Niedermeyer & da Silva, 2005, chapter 51). Typically ERDs caused by hand motor imagery is observed at the contralateral side of the imagined movement while ERS (which are more uncommon in the context of hand motor imagery) is observed at the ipsilateral side. These two patterns have been interpreted as ERD standing for a readiness in the involved cortical areas to actually perform the movement while ERS stand for the opposite i.e. that the area is idle or inhibited (Graimann, Allison & Pfurtscheller 2010, p. 52)

In the international standard for electrode placement for EEG (called the 10-20 system) the electrodes for channels C3 and C4 are placed right over the primary sensorimotor cortex, making them suitable targets for recording sensorimotor rhythms (beta and mu rhythms) (Millán, Franzé, Mouriño, Cincotti & Babiloni, 2002). In another study made by Millán et.al (2002) it was shown that the optimal electrode placement for obtaining ERS and ERD in sensory motor cortex varied between subjects. For some individuals it could therefore be hard to categorize motor imagery control signals when the EEG features are extracted only from electrodes placed on C3 and C4 (Millán, Franzé, Mouriño, Cincotti & Babiloni, 2002).

3.3 Performance variation of BCIs

EEG data from one subject performing the same action is known to vary strongly over time, also within the same session. (Shenoy, Krauledat, Blankertz, Rao & Müller, 2006; Grosse-Wentrup & Schölkopf, 2013). In addition to the effect that training have on the control signal (that is discussed in detail in section 3.3.1) there are also other factors contributing to a session-to-session variance which is discussed below (section 3.3.2).

3.3.1 The training effect - Adaption of man to machine

Within a BCI setup there are two parts which can adopt to each other to a higher or lesser degree, the user and the BCI methods (Graimann, 2010, p. 331-332). For the user to

control a BCI she must able to voluntarily modulate the electrical signals so it can be successfully differentiated into separate classes.

When the control signal generated by the user is created by imaginary movement Neuper et al. (2005) describe that the user can adopt different techniques to cope with this task. They bring up examples where the user can for example try to visualize movement of their own hand, or visualize someone else performing the required hand movement, compared to creating a kinesthetic feeling of movement. From Neuper et al.'s (2005) results it could be concluded that only kinesthetic motor imagery created ERD/ERS patterns which could be detected using EEG recordings from the sensory motor cortex. Hwang et al. (2009) continued this research by training subjects to perform kinesthetic motor imagery and could successfully increase classification rates for subjects that initially had trouble performing kinesthetic motor imagery. Thus showing one example of how learning with feedback could increase the performance of the BCI.

Another training effect reported by Millan et. al (2002) shows that features extracted from electrodes placed on C3 and C4 became more relevant the more the subject trained on performing the motor imagery task. The study made by Herman et al. (2008) further shows that a training effect affects classification i.e., CA is higher when training data and test data is close to each other in time.

3.3.2 More factors of variance in BCI performance over time

Below follows a description of factors that have been shown to generate possible session to session variability in EEG task related data except the training effect, which is discussed in detail above (section 3.3.1).

- Changes in recording conditions: As the recording method can be varied there is an apparent risk that changes can occur over time, creating a variability in the recorded brain activity patterns (Graimann, 2010, p. 331-332). One example of this phenomenon is the placement of the EEG electrodes on the scalp of the subject, creating a possible variation over session to session (Guger, Ramoser, & Pfurtscheller, 2000). Park et al. (2013) investigated the effect slight changes in placement of EEG electrodes had on CA when using different feature extraction methods. They found that the CA was strongly affected by slight changes in electrode placement when some of the commonly used methods for feature extraction was used.

- Mood and motivation: Nijboer et al. (2010) argues that it is often hypothesized that the mood and motivation of the subject is correlated with BCI performance, and that

this could explain the interindividual differences. Since mood and motivation also can vary in the same individual over time, this is applicable to intra individual differences, such as between sessions with the same subject. Nijboer et al. (2010) also performed tests on 6 subjects with Amyotrophic Lateral Sclerosis to see if any intra subject changes could be detected over sessions based on psychological factors such as mood and motivation. They found that the psychological factors played a bigger role when the amount of sessions increased, especially for sensorimotor rhythms sessions. Further they found that the challenge and confidence the subjects felt towards the task was positively related to BCI performance for two of the subjects. Fear of failing the task was negatively related to BCI performance for one of the subject.

- Concentration and fatigue - During recording of brain activity there is a natural risk of the subject getting tired or losing concentration, perhaps even due to factors which are outside the control of the study. These factors can create changes over time in the recorded brain activity patterns (Graimann, 2010; Sun, S., & Zhang, C. 2006).

3.4 BCI classification

Classification algorithms serve the central part in BCIs by classifying the user input into computer commands. In this section the most common algorithms for BCI applications will be introduced and their ability to generalize briefly mentioned. Lastly different problems that BCI classifiers must handle are presented and discussed.

Hwang et al. (2013) performed a thorough literature survey on BCI research articles from 2007 to 2011 and investigated which classification algorithms that were most commonly used. They found that a range of different algorithms were used such as LDA,SVMs, linear regression and neural networks. The most common over the period of 2007 to 2011 was LDA followed by SVM, together being used in more than 50% of the articles reviewed.

Both LDA and SVM are linear methods that are known to have good abilities to generalize, i.e., when the system has been trained it can also successfully handle new data (Lotte,Congedo, Lécuyer, Lamarche & Arnaldi, 2007). But since both LDA and SVM are sensitive to noisy data and strong outliers, both methods should make use of regularization methods to increase performance and lessen the impact from outliers and noise (Lotte,Congedo, Lécuyer, Lamarche & Arnaldi, 2007; Nicolas-Alonso et al.,2012). A more in detail review of SVM and LDA can be found in section 3.4.2 and 3.4.3 respectively.

In general the more data that is available to the classifier, the better the performance of the classifier will be. This is based on the premise that training data comes from the same distribution as the test data, as then more training data will likely to enhance the generalization capacity (Marsland, 2009).

3.4.1 Common challenges in Classification for BCIs

This section explains many challenges that come with classification of EEG data due to properties of its features (see section 3.1 & 3.3 for more details). First, details of the bias-variance dilemma is explained, followed by noise, outliers and the "curse of dimensionality". Last the problem of handling the time dimension in EEG data is discussed.

3.4.1.1 Bias-Variance Dilema

Tightly connected to a classification algorithms ability to generalize is how the bias-variance dilemma is handled, a dilemma which affects most classification algorithms used in BCIs (Marsland, 2009, p. 177-178; Lotte,Congedo, Lécuyer, Lamarche & Arnaldi, 2007). Simply put, bias problem is when the complexity of the function that the classifier uses for classification is too simple to meet the complexity of data and when the classifier in general have adapted too little to the available training data. This gives a situation with stable but biased performance. Problem with variance on the other hand is when the classification algorithm adapts too strongly to the available data, and as a consequence has a highly varying performance, typically with good results on training data but poor results on data which it has not seen before. The more the data varies, the harder it is to find an optimal balance between these two (Marsland, 2009, p. 177-178). One approach used to balancing the variance and bias tradeoff is cross-validation, which splits data into three sets: the training set, the validation set and the test set. While training the classifier on the training set the validation set is used to validate the current learning, and thereby test the classifiers' current generalization ability. The test set is saved to be used for the finals sets (Marsland, 2009, p. 67-69).

3.4.1.2 Curse of dimensionality

Data extracted from EEG is gathered from a set of electrodes, one channel for each electrode, creating a large possible of set of features to base the classification on (Lotte,Congedo, Lécuyer, Lamarche & Arnaldi, 2007). With a large set of features a classifier must be able to handle what is known as the "curse of dimensionality" in machine learning. It can be summarised as that with the dimension of indata to the classifier increasing, more training data will be needed to accurately fit the classification (Marsland, 2009, s. 106-108).

3.4.1.3 Handling time information in EEG data

Another problem that needs to be considered in the context of BCI classifiers is how to handle the time information in the EEG signal, i.e., to handle that different patterns in the EEG signal (obtained during one mental task) is relevant during different time segments from the recording of one task. Lotte et al. (2007) lists a range of different approaches that have been used to address this problem; two of them being (a) concatenation of features from different time segments and (b) combination of classifications at different time segments.

The first tactic (a) creates a feature vector based on concatenation of different time samples, creating a possibly high dimensional vector depending on the number of samples included. Haselsteiner et al. (2000) remark that the strength of combining time samples comes in its simplicity and that you can apply standard classification algorithms to the feature vector. But in turn they point out that weaknesses are that there is no standard on which time samples to chose nor how many of them.

The second tactic (b) classifies the separate time samples one by one or in small groups with different classifiers, and then the result on the whole task is the aggregated result of all individual classifiers. This method of splitting up the data and letting different classifiers combined effort be the result is referred to as ensemble learning in machine learning. (Marsland, 2009, p. 155-164). An open question when using machine learning is how to aggregate the result of the different classifiers.

One basic approach is majority voting, where the most common result of the classifiers are chosen as the final ensemble result. Marsland (2009) writes the following about ensembles using majority voting:

> If the number of classifiers is odd and the classifiers are each independent of each other, then majority voting will return the correct label if more than half of the classifiers agree. Assuming that each individual classifier has a success rate of *p*, the probability of the ensemble getting the correct answer is a **binomial distribution** of the form:
>
> $$\sum_{k=T/2+1}^{T} \binom{T}{k} p^k (1-p)^{T-k}, \qquad (7.5)$$
>
> where T is the number of classifiers. If p > 0.5 then this sum approaches 1 as $T \to \infty$. (Marsland, 2009, p. 162)

This means that the success rate of the ensemble classification will increase as the number of classifiers increases, as long as each individual classifier has a slightly better success rate than random (50% in case of two classes with about the same amount of data

points). This shows the inherent strength of the ensemble method (Marsland, 2009, p. 162-164).

Sun et al. (2007) describes that it is important that the individual classifiers of an ensemble are different, for example due to the fact that they have trained on different parts of data or are using different parameters, otherwise a singular classifier could perform the same job as the ensemble. Further with an increasing number of classifiers the computational burden increases, especially during training where tuning of each of the classifier's parameters can be necessary (Sun, Zhang, & Zhang 2007).

3.4.2 Support Vector Machines (SVMs)

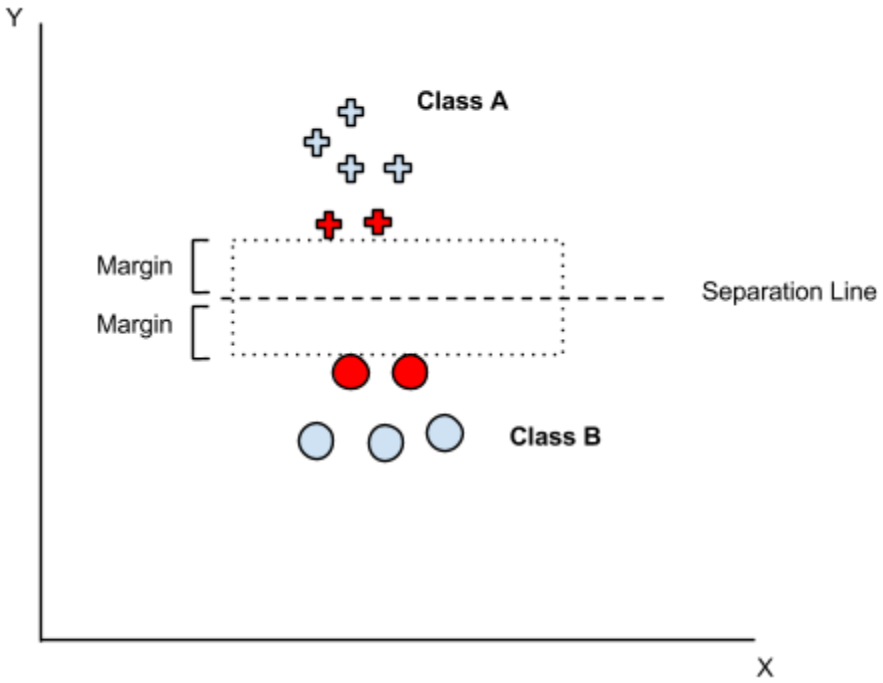

Figure 2: Illustration showing two classes A and B in 2-dimensional space being separated so that the margin gets maximized. Points marked in red color function as support vectors.

SVMs are based on the natural classification concept that two linearly separable classes in 2-dimensional space should be separated by a line equidistant from the two classes (Marsland, 2009, p. 120-124). This means that the separation line in the 2-dimensional case will be placed so that the distance (referred to as margin) between the two classes is as wide as possible, see figure 2. The margin is measured using a select set of points from each class (marked red in figure 2), referred to as the support vectors, these are chosen to maximize the margin (Marsland, 2009, p. 120-124).

As the margin lies symmetric around the line, it will create a cylinder around the separation line in 3-dimensional space and a hypercylinder in higher dimensional spaces. In the same way the line or lines used to separate data will become hyperplanes in higher dimensions (Marsland, 2009, p. 120-124). Thereby allowing the algorithm to solve classification problems in higher dimensions. Due to SVM creating hyperplanes to separate classes it constrains data to be linearly separable to effectively be classified and therefore SVMs can be regarded as a linear classifier (Nicolas-Alonso & Gomez-Gil, 2012).

To solve nonlinear classification problems the SVM algorithm builds on Cover's theorem which states that a non-linearly separable problem transformed into a high dimensional non-linear space increases the likelihood of the data becoming linearly separable (Cover, 1965 according to Nicolas-Alonso & Gomez-Gil, 2012). Thus making it possible for SVM to classify also non-linearly separable data by transforming the given data. This is usually done by introducing a kernel function which maps indata into a higher dimension. In BCI applications one common kernel function to use is the radial basis function (Nicolas-Alonso & Gomez-Gil, 2012). With the use of a kernel function the SVM can act as a non-linear classifier, allowing a non-linear boundary to separate the classes (Nicolas-Alonso & Gomez-Gil, 2012).

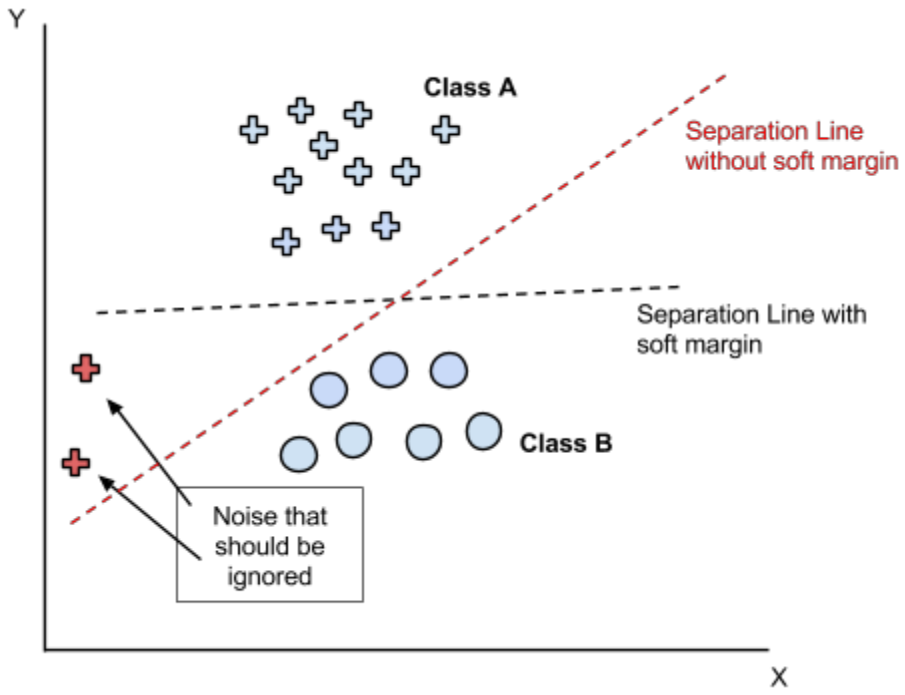

Figure 3: Illustration showing an example of how SVMs with a soft margin parameter can ignore noise in data to possibly increase generalization capacity.

Another issue that classifiers must be able to handle is noise in data. Noise can if left unchecked negatively affect the classification performance, see for example how noise klan affect the placement of the red separation line in figure 3 (Steinwart & Christmann, 2008, p. 14-15). One way to handle noise with SVMs is to introduce a so called soft margin parameter, which allows the separating hyperplane to ignore certain data points, thereby possibly lessening the risk of noise having effect on classification performance for new data, see for example the black separation line in figure 3 which ignores data assumed to be outliers (Steinwart & Christmann, 2008, p. 14-15).

Below are strengths and weaknesses of SVM listed:

Strengths

- Its ability to cope with high dimensional problems, reducing the effect of "the curse of dimensionality" (Rakotomamonjy & Guigue, 2008; Lotte, Congedo, Lécuyer, Lamarche, Arnaldi, 2007)
- Can be used both on linear and nonlinear problems (Nicolas-Alonso & Gomez-Gil, 2012; Marsland, 2009).
- Due to the benefit of a maximum margin possibly combined with regularization, a SVMs has good generalization abilities (Lotte, Congedo, Lécuyer, Lamarche, Arnaldi, 2007)

Weaknesses

- Does not work well on large data sets due to the algorithm being computationally heavy (Marsland, 2009; Lotte, Congedo, Lécuyer, Lamarche, Arnaldi, 2007).

3.4.3 Linear discriminant analysis (LDA)

Classifiers based on LDA are the most commonly used classifiers in BCI applications today (Hwang et.al, 2013). As the name indicates LDA is a classification method that tries to linearly separate two (or more) classes of data. The basic idea in LDA is that multidimensional data could be easier discriminated when it is projected down to an vector w where the two classes is as far as possible from each other (see figure 4 below).

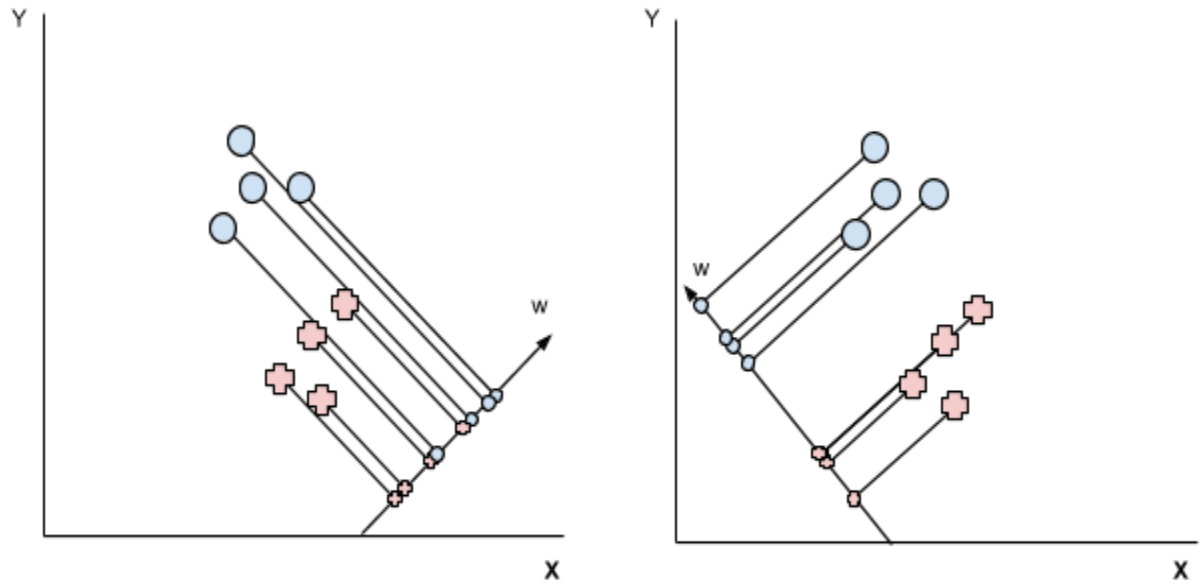

Figure 4: Illustration showing two classes A and B in 2-dimensional space projected onto the one dimensional vector w in two cases, the first case (to the left) not being able to differentiate between the classes, while the second case (to the right) successfully differentiates the two classes.

In the case of two different classes of data the classification is done by introducing a hyperplane that intersects the projection vector, and by that divide the data points into two sets, one on each side of the plane (Lotte et.al, 2007; Nicolas-Alonso & Gomez-Gil, 2012).

Figure 4 shows how the discrimination ability of the two classes is totally dependent on the choice of the projection vector *w*. To find the optimal projection vector w it is therefore essential to define a measure of separation as a function of *w* (Marsland, 2009). In the simplest version of LDA, this separation function is only based on the means of the two classes.

$J(w) = |w(\mu_1 - \mu_2)|$ *where* μ_1 *and* μ_2 *are the mean vector for each class*

and w is chosen to maximize J(w). This simple separation function unfortunately cannot handle cases when variance within the classes is high in comparison to the difference in mean (see figure 5):

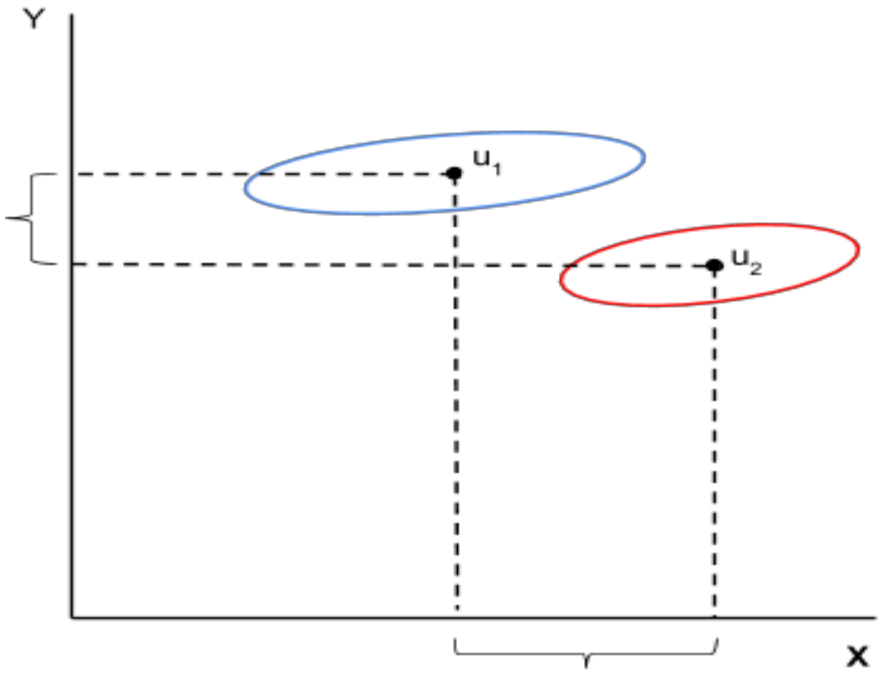

Figure 5: Example of data where one axis (the x-axis) has greater distance between means (μ_1 and μ_2) but the other axis (the y-axis) has better separation of the two classes if projected to this axis.

The standard solution to this is using Fisher LDA (Nicolas-Alonso & Gomez-Gil, 2012) where the separation function has an additional term S, representing the sum of the variances of the two classes, when projected to the vector w:

$J(w) = |w(\boldsymbol{\mu_1}-\boldsymbol{\mu_2})| /S$

The maximal value of J is obtained when data points from the same classes are projected close to each other and at the same time, the projected means are as far apart as possible (Marsland, 2009).

As both Lotte et.al (2007) and Nicolas-Alonso & Gomez-Gil (2012) point out, that LDA could perform better when combined with different regularization methods. This because regularization is a method to handle problems related to noise in data, extreme outliers and small training data sets, which are all common in the BCI context.

4. Method

4.1 Data

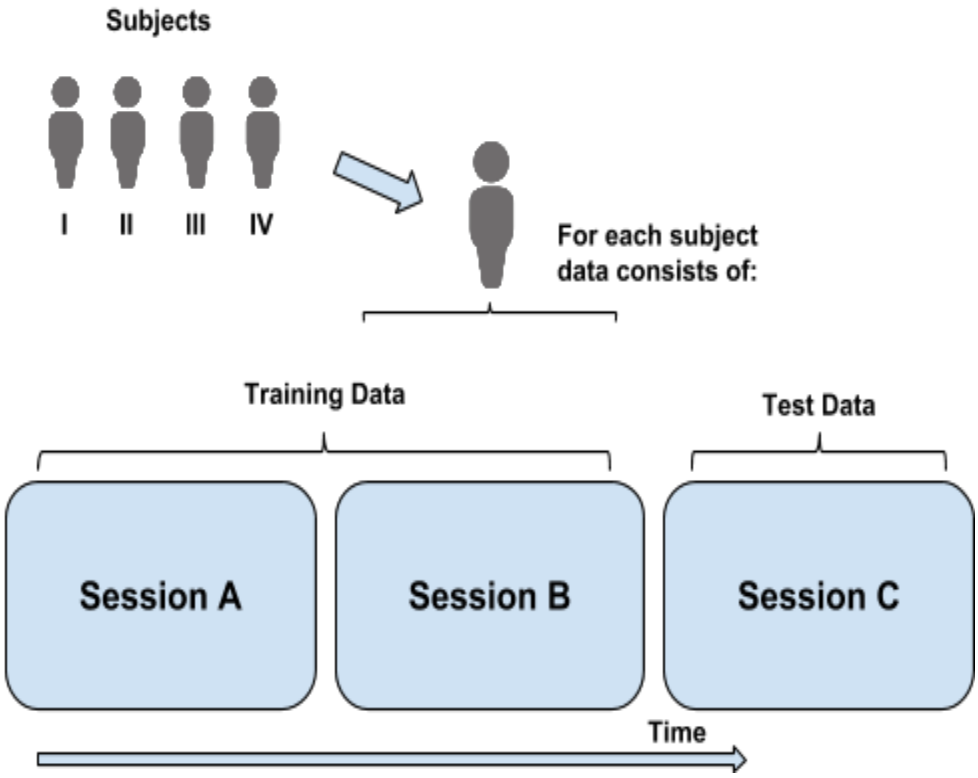

Figure 6: Illustration showing available data from the four subjects, for each subject there was 3 sessions: A, B and C, where A was the earliest session and C the latest session.

The data used in this study was given to us by our supervisor, and originated from the study Herman et al. (2008). The data consisted of 4 subjects performing imaginary movement of either left or right hand recorded through electrodes placed over the sensorimotor cortex (C3 and C4 in the international standard system 10-20). The data consisted of 3 sessions for each subject, each session was conducted with direct feedback to the subject, see figure 6 for an illustration of the data. The data had gone through a feature extraction using spatio-temporal analysis and also the least relevant parts of the signal had been discarded.

This means that the data we were given consisted of a total of 12 session, and each session contained a number of trials, ranging from 100 to 180 trials. Each trial in turn contained 85 time samples in the form of 4-dimensional vectors (2 signals from each of the 2 electrodes). These four numbers represented the signal power in the beta and mu

frequency bands for the two electrodes. Coupled to each trial were also the correct class of the trial, represented as either -1 or 1.

4.2 Construction of ensembles and normalization

Due to the problem of handling time information in EEG recordings (see section 3.4.1.3) we chose to create one unique classifier for each of the 85 time samples of one trial, thus making an ensemble of classifiers to handle one whole trial at the time. The results of each individual classifier for one trial was gathered and a majority vote decided the class of the tested trial. This method allowed us to avoid problems regarding concatenation of features as there is no standard method on how to concatenate features (Haselsteiner & Pfurtscheller, 2000). Concatenation of features also runs the risk of creating high dimensional data, leading to poorer performance due to *the curse of dimensionality* described in section 3.4.1.2. Early test of our ensemble method also showed to give reasonable results, further motivating us to use this method.

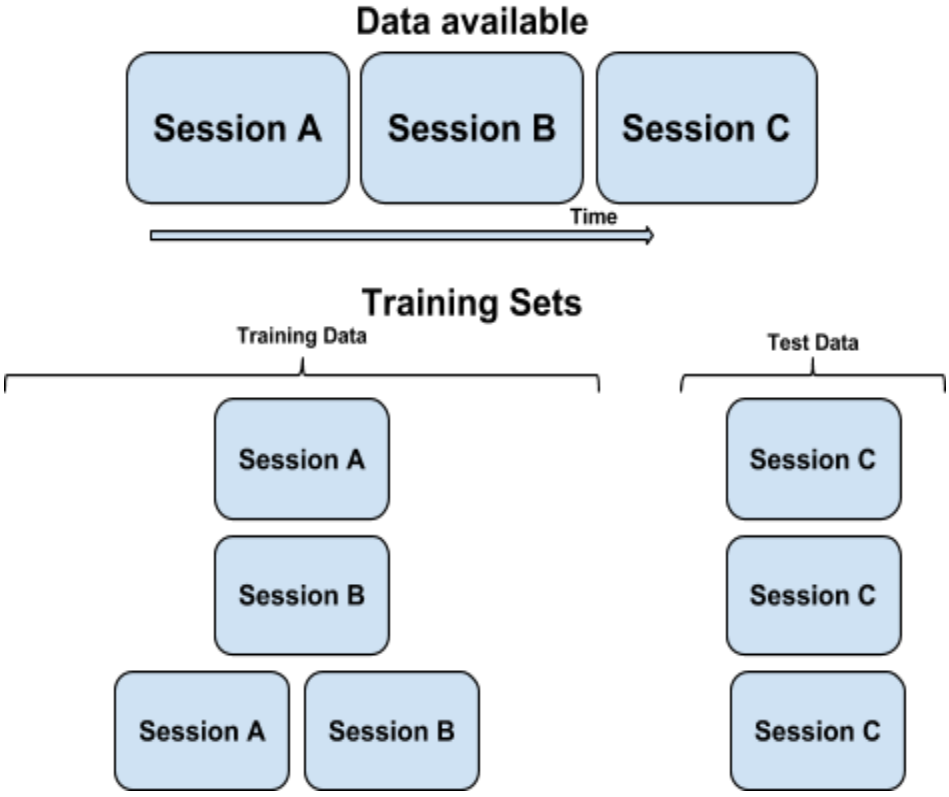

Figure 7: Illustration showing available data and the training sets used to test generalization ability on the test data.

As training data for the ensembles we used the two earlier sessions (A and B in figure 7) of each subject, and saved the last session C as test data to test the generalization ability.

Training was performed either by training on one separate session or training on both training sessions combined into one data set (when training on a single session the other training session was also used to test generalization). This created three possible training sets, the two individual training session alone and these two sessions combined into one set (see figure 6).

For each training set all of the 85 classifiers in the ensemble trained on N patterns of their respective time sample, where N was the number of trials in the current training set. Lastly the generalization ability was evaluated on the final test session for each possible training set. CA was measured by dividing the number of correctly classified trials with the total amount of trials tested on, rounded to the nearest percent (see formulae below).

$$Classification\ Accuracy\ (CA) = \frac{Correctly\ classified\ trials}{total\ amount\ of\ trials}$$

Scaling of data (normalization) while training the classifiers was performed separately before the data was presented to the classifiers to allow more control of the normalization process. Each features mean and standard deviation was calculated, and then the respective data was scaled by subtracting the mean and dividing by the standard deviation, once for each classifier and their respective part of the data set.

Test data was normalized before presented to the classifiers according to its own mean and standard deviation, a process which cannot be adopted in an online BCI context as the full test set would be unknown.

4.21 SVM ensemble

The 85 SVMs used as an ensemble were trained with a linear kernel function, no auto scaling of data and the use of quadratic programming to place the decision hyperplane. All other parameters used default values, including the soft margin parameter (see section 3.4.2 for details regarding this parameter). The use of the default value for the soft margin parameter is due to that our investigation of different possible values of the parameter values gave little to no effect on the CA when testing on the other training session (when available) or when using cross validation on the training set.

4.22 LDA ensemble

The 85 LDA classifiers used as an ensemble were trained using a linear discriminant type. Similar to the case of SVMs, our investigation of tuning available regularization parameters had small or no effect on the CA of the ensemble, and was thereby kept at default values.

4.3 Analysis of results

Due to the limited scope of this paper both in respect to data resources and time, the analysis of the results does not consist of statistical hypothesis testing, which naturally limits the conclusions that can be drawn. The analyses of results are therefore based on trends in the results, in the meaning that differences of a few percentages in two results will be rather irrelevant, and instead a big picture must be used by observing trends between and within the different subjects. As stated in our objective (section 2) we are interested in two different trends: the training effect on the human user and that more data generally means better generalization in classification.

The training effect should create a trend in our results where training on the middle session (session B) should generally get better results while testing on the last session (session C) compared to training on the earliest session (session A). This would reflect that the users gets better at creating thoughts that are easy for the classifier to correctly classify as time goes by.

The effect of more data increasing the generalization ability for classification should create a trend in our results so that when training on session A and B together spawns better results than training on just one of these sessions alone, when testing on session C.

We are interested to see if there is any general trend showing that any of these effects are stronger and therefore have implications on how training data should be chosen.

5. Results

In this section results are displayed as comparisons of CA between the three different choices of training data (session A, session B, or both session A and B). results are also displayed separately for the two types of classifiers used: SVMs and LDA. First the average results for each classification method is presented (see section 5.1) and thereafter the results for each one of the four subjects (see section 5.2 and 5.3).

5.1 Average classification accuracy (CA) for LDA and SVM ensembles

Table 1: Comparison of average CA for the four subject on session C, displayed separately for the two classifiers: LDA and SVM, when either training on session A, B or both of these sessions together.

LDA			SVM	
Training data	**Average CA on session C**		**Training data**	**Average CA on session C**
A	62%		A	62%
B	58%		B	58%
A&B	61%		A&B	61%

The aggregated results for LDA and SVM in table 1 show no big difference in CA for the different choices of training data. The results thereby do not support the hypothesis i.e., that choosing the session closest in time to the the test session (session B), would give better performance than choosing session A as training data.

Further the two methods used for classification, LDA and SVM, achieved equal average results, as can be seen in table 1.

5.2 Classification Accuracy (CA) for each subject using ensemble of LDA classifiers

This section shows the resulting CA using an ensemble of LDA classifiers, separated into one table for each subject (I - IV), using training data from session A and B. For each set of training data CA is shown when testing on already seen data (the training data itself) and on the test session C. With session A as training data CA is also shown when testing on session B.

Table 2: CA for subject I using an ensemble of LDA classifiers trained on data from session A, B and both A and B together. Tests on already seen data is colored light blue and test on never before seen data from the other sessions are colored dark blue.

Trained on session	***Accuracy on session A***	***Accuracy on session B***	***Accuracy on session A & B***	***Accuracy on session C***
A	*69%*	*57%*	*-*	*63%*
B	*-*	*62%*	*-*	*57%*
A & B	*-*	*-*	*67%*	*64%*

Table 3: CA for subject II using an ensemble of LDA classifiers trained on data from session A, B and both A and B together. Tests on already seen data is colored light blue and test on never before seen data from the other sessions are colored dark blue.

Trained on session	***Accuracy on session A***	***Accuracy on session B***	***Accuracy on session A & B***	***Accuracy on session C***
A	*78%*	*76%*	*-*	*53%*
B	*-*	*84%*	*-*	*53%*
A & B	*-*	*-*	*79%*	*53%*

Table 4: CA for subject III using an ensemble of LDA classifiers trained on data from session A, B and both A and B together. Tests on already seen data is colored light blue and test on never before seen data from the other sessions are colored dark blue.

Trained on session	**Accuracy on session A**	**Accuracy on session B**	**Accuracy on session A & B**	**Accuracy on session C**
A	68%	55%	-	64%
B	-	69%	-	63%
A & B	-	-	64%	63%

Table 5: CA for subject IV using an ensemble of LDA classifiers trained on data from session A, B and both A and B together. Tests on already seen data is colored light blue and test on never before seen data from the other sessions are colored dark blue.

Trained on session	**Accuracy on session A**	**Accuracy on session B**	**Accuracy on session A & B**	**Accuracy on session C**
A	70%	67%	-	68%
B	-	74%	-	60%
A & B	-	-	71%	65%

5.2 Classification Accuracy (CA) for each subject using ensemble of SVM classifiers

This section shows the resulting CA using an ensemble of SVM classifiers, separated into one table for each subject (I - IV), using training data from session A and B. For each set of training data CA is shown when testing on already seen data (the training data itself) and

on the test session C. With session A as training data CA is also shown when testing on session B.

Table 6: CA for subject I using an ensemble of SVM classifiers trained on data from session A, B and both A and B together. Tests on already seen data is colored light blue and test on never before seen data from the other sessions are colored dark blue.

Trained on session	***Accuracy on session A***	***Accuracy on session B***	***Accuracy on session A & B***	***Accuracy on session C***
A	*69%*	*57%*	*-*	*62%*
B	*-*	*63%*	*-*	*58%*
A & B	*-*	*-*	*66%*	*65%*

Table 7: CA for subject II using an ensemble of SVM classifiers trained on data from session A, B and both A and B together. Tests on already seen data is colored light blue and test on never before seen data from the other sessions are colored dark blue.

Trained on session	***Accuracy on session A***	***Accuracy on session B***	***Accuracy on session A & B***	***Accuracy on session C***
A	*79%*	*77%*	*-*	*54%*
B	*-*	*84%*	*-*	*51%*
A & B	*-*	*-*	*80%*	*51%*

Table 8: CA for subject III using an ensemble of SVM classifiers trained on data from session A, B and both A and B together. Tests on already seen data is colored light blue and test on never before seen data from the other sessions are colored dark blue.

Trained on session	***Accuracy on session A***	***Accuracy on session B***	***Accuracy on session A & B***	***Accuracy on session C***
A	*67%*	*54%*	*-*	*63%*
B	*-*	*69%*	*-*	*63%*
A & B	*-*	*-*	*65%*	*63%*

Table 9: CA for subject IV using an ensemble of SVM classifiers trained on data from session A, B and both A and B together. Tests on already seen data is colored light blue and test on never before seen data from the other sessions are colored dark blue.

Trained on session	***Accuracy on session A***	***Accuracy on session B***	***Accuracy on session A & B***	***Accuracy on session C***
A	*70%*	*68%*	*-*	*68%*
B	*-*	*74%*	*-*	*60%*
A & B	*-*	*-*	*71%*	*64%*

6. Discussion

Classifying the users thoughts is a central and hard task in all BCI systems. This is known to be even more difficult on early training sessions where the user have little experience of performing the task to generate the control signal, i.e the thoughts the BCI system is trained to respond to (Wolpaw, McFarland & Vaughan 2000;Grosse-Wentrup & Schölkopf, 2013). In this study we wanted to use data from early training sessions to examine how the CA on data from an unseen session was affected by the choice of training data for the classification algorithm. In a previous study by Herman et.al (2008) a clear trend was observable where training data recorded closer in time to the test data gave better generalization capacity than training data from earlier sessions. Herman et.al (2008) results showed a clear gap in CA between using the latest session as training data or using earlier sessions. However, in their study they only compared the results of using a single session as training data but in general classification algorithms perform better the more training data they get, as long as data can be considered to come from the same distribution as the test data (see section 3.4 above).

We intended to investigate which of these effect that was stronger, to see if using the closest in time session or using more sessions as training data gave the best CA on a unseen session. As data we had a small sample of the data from the study by Herman et. al (2008), our hypothesis therefore was that we would see the same effect of using training data closest in time to test data i.e. using the latest session as training data would give better CA than using a single earlier session.

The unexpected result we got was that we could not see any positive effect of using the latest session as training data. In fact for the four subject we had there was no one for which using the closest session was better than using a single earlier session. Using the earlier one could for one subject even improve the CA on the last session with 8% compared to using the last session (see table 5). It is hard to give any reasons based in theory, for the conclusion that choosing a single earlier session as training data would give systematically better results than choosing one that is closer in time when the user has more experience. A more reasonable conclusion is that the sample of data we had was too small to see the training effect that was clearly observable in Herman et. al (2008). In their study they used data from 8 subjects and a larger number of sessions for each person. In our study we had just one observation of interest for each of the subjects and for subject II which initially had the highest generalization capacity proved in the testing phase to achieve results equal to that of chance (close to 50%, see table 3 and 7). This led to a situation with practically only three observations of interest. As we discussed in section 3.32 there are numerous factors creating session to session variance in brain activity

patterns and it is plausible that these factors in our samples overpowered the effect of user training.

Regarding the effect of using more sessions as training data we could observe that there was a slightly positive effect for one subject when compared to using a single session (see table 2 and 6). When there was a bigger gap in performance between the two classifiers using different single sessions as training data, the classifier that used both sessions produced a result in between. From this it is seems to be a reasonably good choice to use the two latest session as training data especially as our result have shown that it is not always the case that the latest session performs better than the earlier ones. Still more studies are needed to validate this conclusion.

6.1 Method discussion

The method of using an ensemble of classifiers, one for each time sample can be seen as a rather unconventional choice of how to classify BCI data. Therefor it could be speculated that this choice could be the reason to our rather unexpected results, but there are reasons to argue that this is not the case.

The strongest argument for using an ensemble was that we got results that is similar to earlier studies of this data, which points out that the expected CA lies around 60% to 70%, which also is achieved by our ensembles of classifiers. This indicates that our method is stable compared to other classification process on this data set, and not results created by chance.

Further both the ensemble of linear SVMs and LDA have matching results. That these independent but both linear methods produce equal results could be seen as a quality assurance i.e. if they instead had produced contradictory results it would have given reason to expect errors in the implementation. The equality in performance between the two methods was seen already in the initial tests which motivated us to utilize ensembles to investigate our objectives.

However there are also objections that could be raised against using an ensemble of classifiers working on each time sample. One objection is that the ensemble uses the advantage of having the whole trial available when classifying as this is not realistic in an online situation. Further this offline method made it possible to use low dimensional data which is quite uncommon in a BCI context (Lotte,Congedo, Lécuyer, Lamarche & Arnaldi, 2007).This is a problem because our results are therefore less applicable for situations with high dimensionality on indata. It could for example been the case that there would be a

stronger positive effect of using both earlier sessions as training data. This is based on the reasoning that the higher dimensionality you have the more training data you need (see section 3.4.1.2 about the curse of dimensionality).

Another objection that could be made is how normalization is done. Using the mean and standard derivation from the whole session for normalizing before classification is of course not possible in an online situation but there are similar approaches that are. For example some data could be collected in start of each session to decide which mean and which standard derivation that should be used for normalization in this specific session.

The methods of choice also has the limitation that we used only linear classifiers which further restrains the scope of this study, as the use of nonlinear classifiers could have given different results. But as linear methods stands for a majority of the classification algorithms used in BCI today (Hwang et al., 2013) we argued that this limitation is reasonable. As mentioned above the comparison of the two independent linear methods achieved results could also be seen as a quality assurance.

6.2 Conclusion

In summary it is hard to draw any clear conclusion from such a small data sample, but from our results we can see that is not always the case that training data recorded closer in time to the test data generate higher CA. Therefore from our result it seems to be a safer choice to use more than the latest session as training data. Still more studies are needed to confirm that using more session for training really is better also on data where the training effect is strong.

References

Graimann, B., Allison, B., & Pfurtscheller, G. (Eds.). (2010). *Brain-computer interfaces: Revolutionizing human-computer interaction*. Springer.

Grosse-Wentrup, M., & Schölkopf, B. (2013). A Review of Performance Variations in SMR-Based Brain− Computer Interfaces (BCIs). In *Brain-Computer Interface Research* (pp. 39-51). Springer Berlin Heidelberg.

Guger, C., Ramoser, H., & Pfurtscheller, G. (2000). Real-time EEG analysis with subject-specific spatial patterns for a brain-computer interface (BCI). *Rehabilitation Engineering, IEEE Transactions on*, *8*(4), 447-456.

Haselsteiner, E., & Pfurtscheller, G. (2000). Using time-dependent neural networks for EEG classification. *Rehabilitation Engineering, IEEE Transactions on*, *8*(4), 457-463.

Herman, P., Prasad, G., & McGinnity, T. M. (2008, August). Design and on-line evaluation of type-2 fuzzy logic system-based framework for handling uncertainties in BCI classification. In *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE* (pp. 4242-4245). IEEE.

Hoffmann, U., Vesin, J. M., Ebrahimi, T., & Diserens, K. (2008). An efficient P300-based brain–computer interface for disabled subjects. *Journal of Neuroscience methods*, *167*(1), 115-125.

Hwang, H. J., Kim, S., Choi, S., & Im, C. H. (2013). EEG-Based Brain-Computer Interfaces: A Thorough Literature Survey. *International Journal of Human-Computer Interaction*, *29*(12), 814-826.

Hwang, H. J., Kwon, K., & Im, C. H. (2009). Neurofeedback-based motor imagery training for brain–computer interface (BCI). *Journal of neuroscience methods*, *179*(1), 150-156.

Lotte, F., Congedo, M., Lécuyer, A., Lamarche, F., & Arnaldi, B. (2007). A review of classification algorithms for EEG-based brain–computer interfaces. *Journal of neural engineering*, *4*.

Marsland, S. (2009). *Machine Learning: An Algorithmic Perspective.* Boca Raton: Chapman & Hall/CRC.

Millán, J. D. R., Franzé, M., Mouriño, J., Cincotti, F., & Babiloni, F. (2002). Relevant EEG features for the classification of spontaneous motor-related tasks. *Biological cybernetics*, *86*(2), 89-95.

Müller, K. R., Krauledat, M., Dornhege, G., Curio, G., & Blankertz, B. (2004). Machine learning techniques for brain-computer interfaces.

Neuper, C., Scherer, R., Reiner, M., & Pfurtscheller, G. (2005). Imagery of motor actions: Differential effects of kinesthetic and visual–motor mode of imagery in single-trial EEG. *Cognitive Brain Research*, *25*(3), 668-677.

.

Nicolas-Alonso, L. F., & Gomez-Gil, J. (2012). Brain computer interfaces, a review. *Sensors*, *12*(2), 1211-1279.

Niedermeyer, E., & da Silva, F. L. (Eds.). (2005). *Electroencephalography: basic principles, clinical applications, and related fields*. Lippincott Williams & Wilkins.

Nijboer, F., Birbaumer, N., & Kübler, A. (2010). The influence of psychological state and motivation on brain–computer interface performance in patients with amyotrophic lateral sclerosis–a longitudinal study. *Frontiers in neuroscience*, *4*.

Park, S. A., Hwang, H. J., Lim, J. H., Choi, J. H., Jung, H. K., & Im, C. H. (2013). Evaluation of feature extraction methods for EEG-based brain–computer interfaces in terms of robustness to slight changes in electrode locations. *Medical & biological engineering & computing*, *51*(5), 571-579.

Rakotomamonjy, A., & Guigue, V. (2008). BCI competition III: dataset II-ensemble of SVMs for BCI P300 speller. *Biomedical Engineering, IEEE Transactions on*, *55*(3), 1147-1154.

Shenoy, P., Krauledat, M., Blankertz, B., Rao, R. P., & Müller, K. R. (2006). Towards adaptive classification for BCI. *Journal of neural engineering*, *3*(1), R13.

Steinwart, I., & Christmann, A. (2008). *Support vector machines*. Springer.

Sun, S., Zhang, C., & Zhang, D. (2007). An experimental evaluation of ensemble methods for EEG signal classification. *Pattern Recognition Letters*, *28*(15), 2157-2163.

Sun, S., & Zhang, C. (2006). Adaptive feature extraction for EEG signal classification. *Medical and Biological Engineering and Computing*, *44*(10), 931-935.

Wolpaw, J. R., McFarland, D. J., & Vaughan, T. M. (2000). Brain-computer interface research at the Wadsworth Center. *Rehabilitation Engineering, IEEE Transactions on*, *8*(2), 222-226.

Figure references

Figure 1:
トマトン124 (2010). Electrodes of International 10-20 system for EEG. [Illustration]. http://commons.wikimedia.org/wiki/File%3A21_electrodes_of_International_10-20_system_for_EEG.svg