Faculty of Engineering and Architecture
Departement ELIS
2015–2016

Recognize Emotion in the brain using EEG Data

by

Andreas DE LILLE

Promotors:	Prof. J. DAMBRE
	Dr. Ir. P. VAN MIERLO
Assistent:	Ir. T. VERHOEVEN

Contents

Nomenclature

BCI Brain Computer Interface

CSP Common Spatial Patterns

DASM Differential Asymmetry

DCAU Differential Caudality

DE Differential Entropy

DEAP Dataset for Emotion Analysis using Physiological Signals

EEG Electroencephalography

ELM Extreme Learning Machine

GELM Graph regularized Extreme Learning Machine

GSR Galvanic Skin Response

KNN k-nearest neightbors

LDA Linear Disciminant Analysis

LDS Linear Dynamic System

LR Logistic Regression

MEG magnetoencephalography

MRMR Minimal Redundancy Maximal Relevance

OCR Optical Character Recognition

OOB Out of Bag

PCA Principal Component Analysis

PSD Power Spectral Density

RASM Rational Asymmetry

RCAU Rational Caudality

RF Random Forests

SAM self-assessment manikins

1

Introduction

This chapter describes the context of the thesis, starting with brain computer interfaces(BCI), before defining some BCI basics. After that, the P300 speller and P300 paradigm are introduced. Before the need for an emotionally aware P300 speller is justified, the basic process of emotion in the brain is explained.

1.1 Brain computer interfaces

A Brain Computer Interface (BCI), creates a direct neural link from the brain to the computer[1], that tries to recognize patterns and based on the extracted information, performs actions. A BCI removes to need for physical actions, i.e. typing or moving a mouse, for the transfer of information. The neural link provided by the BCI is made of two important components. The first component is the extraction component, which extract brain signals from the brain. The second component is the computer that interprets signals and performs actions based on the outcome.

1.1.1 Electroencephalography (EEG)

Different technologies exist to analyze the brain, the most convenient method is via Electroencephalography (EEG), since it is a non-invasive method. Non-invasive methods, in contrast to invasive methods require no surgery; they simply measure electrical activity using electrodes placed on the scalp.

The electrical activity in a brain is caused when an incoming signal arrives in a neuron. This triggers some sodium ions to move inside the cell, which in turn, causes a voltage rise[2]. When this increase in voltage reaches a threshold, an action potential is triggered in the form of a wave of electrical discharge that travels to neighboring neurons. When this reaction occurs simultaneously in a lot of neurons, the change in electrical potential becomes significantly, making it visible to the EEG surface electrodes. EEG can thus only capture synchronized activity of many, many neurons.

Signals originating from the cortex, close to the skull, are most visible, while signals originating deeper in the brain cannot be observed directly. Even for signals originating close to the cortex, EEG is far from precise as the bone between the the cortex and electrodes distorts the signal.

Additionally other artifacts like eye and muscle movement add a lot a noise to the signal, noise removal techniques are therefor advised. Even though the noise is persistent and EEG data has very low spatial resolution, it still can provide significant insight into the electrical activity of the cortex while offering excellent temporal resolution[3].

Note that EEG records electrical activity, other methods like magnetoencephalography (MEG) measure brain activity using magnetic fields. Since MEG is more prone to noise from external magnetic signals, i.e. the earth's magnetic field and electromagnetic communication, a magnetic shielded room is required, making this method very expensive and not mobile.

EEG uses electrodes which are placed on the scalp to measure the electrical activity. To ensure that experiments are replicable, standards for locations of electrodes have been developed. One of these systems is the 10/20 system, an internationally recognized methods to describe location of scalp electrodes[4]. The numbers 10 and 20 refer to the distances between the electrodes, which are either 10% or 20% of the total front-back or left-right distance of the skull. Each site is identified with a letter that determines the lobe and hemisphere location.

- **F:** Frontal
- **T:** Temporal
- **C:** Central
- **P:** Parietal
- **O:** Occipital

Note that no central lobe exists; the C letter is only used for identification purposes. The letter z indicates that the electrode is placed on the central line. Even numbers are use for the right hemisphere, while odd numbers are used for the left hemisphere. A picture of a 23 channel 10/20 system is added below for clarification. Even though some experiment setups may use a different set of channels than shown in figure 1.1, they all follow the same naming convention.

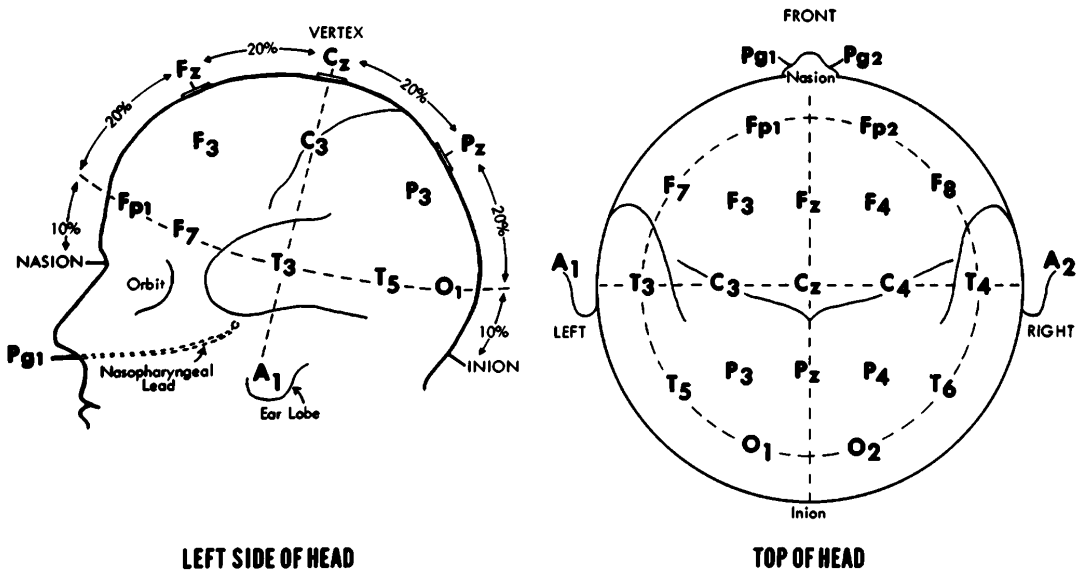

Figure 1.1: The electrode placement of a 23 channel system[5].

Two different types of EEG channels exist, monopolar and dipolar. A monopolar channel records the potential difference of a signal, compared to a neutral electrode, usually connected to an ear lobe of mastoid. A bipolar channel is obtained by subtracting two monopolar EEG signals, which improves SNR by removing shared artifacts[6].

In the frequency domain, brain waves are usually split up into different bands[7, 8], each band has a different medical interpretation. These wavebands are:

1. **Alpha:** 8-13Hz, indicate how relaxed and/or inactive the brain is.
2. **Beta:** 13-30HZ, indicate a more active and focused state of mind.
3. **Gamma:** 30-50Hz, relate to simultaneous processing of information from different brain areas.
4. **Delta:** 0-4hz, these waves are generated during dreamless sleep and meditation.
5. **theta:** 4-8Hz, occur during dreaming.

Most muscle and eye artifacts have a frequency around 1.2Hz. Artificats caused by nearby power lines, have a frequency around 50Hz[2]. To remove most of this noise, a bandpass filter is usually applied to filter out frequencies below 4Hz and above 40-45Hz.

1.1.2 Person specific recognition versus general recognition

BCI applications are used in two settings. The first mode is a person specific setting, where a BCI interface is calibrated on a subject. The second mode is a general BCI interface that works 'cross-subject', meaning that it should be able to works for different persons. It is much harder to achieve good results for the second setting, since EEG data is very personal from nature. Transfer learning has provided good results in imaginary motion recognition in the past. For emotion recognition though simple person specific classifiers are used, as finding person independent EEG features cross person is still an ongoing topic of research.

1.2 Emotion recognition

Psychology makes a clear distinction between physiological behavior and the conscious experience of an emotion, called expression[2]. The expression consists of many parts, including the facial expression, body language and voice concern. Unlike expression, the physiological aspect of an emotion, e.g. heart rate, skin conductance and pupil dilation, is much harder to control. To really know one's emotions, it seems, one has to research the physiological aspect of the emotion. One possibility for this is analysis of brain activity via Electroencephalography[9], which is the main method for this thesis.

1.2.1 Emotion in the brain

Before emotions can be recognized, a classification model is needed. A common model to classify emotions is the bipolar arousal-valence model[2, 10], that places emotions in a two dimensional space. The main advantage of using a multidimensional model, is that all emotions are modelled

in its space, even when no particular discrete label can be used to define the current feeling. Figure 1.2 shows the mapping of different emotions for this model.

Even though arousal and valence describe emotion quite well, a third dimension can also be added. The new model then has three dimensions: arousal, valence and dominance. Arousal indicates how active a person is and ranges from inactive, bored to active, excited. The valence indicates if the emotion is perceived as positive or negative. The third dimension, the dominance, indicates how strong the emotional feeling was and ranges from a weak feeling to an empowered, overwhelming feeling. The dominance component can aid to filter out samples of strong feelings, since feelings with low dominance are less likely to show significant effects.

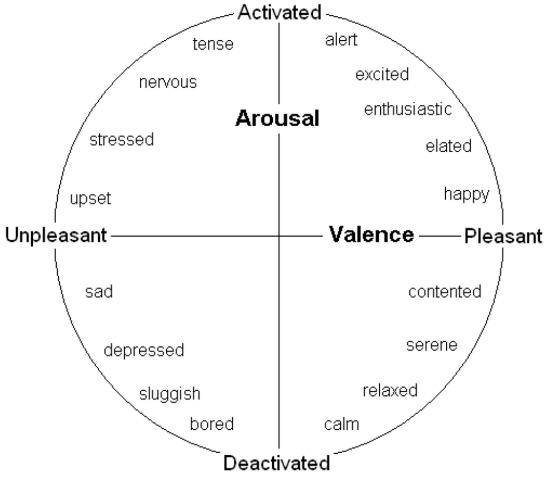

Figure 1.2: The arousal - valence model maps emotions in a two dimensional plane.

1.2.2 Features

In this thesis two categories of features are observed. EEG features and non-EEG features. Non-EEG features are physiological signals like heart rate, skin conductivity, respiration rate, ... They are easy to measure as they do not require an EEG cap. The EEG features are features extracted from the EEG signals of a subjects. Note that the literature does not agree on a specific set of EEG signals nor does it agree on what channels and/or waveband are important. However the literature does agree on certain things; the right hemisphere of a subject is generally speaking, more active during negative emotions than the left hemisphere, which is in turn more active during positive emotions[10, 9, 11]. The features are discussed in more depth in 2.1.

1.3 Machine learning

The next important topic that needs to be covered is machine learning. Machine learning is the missing link between the features and the emotion recognition. As machine learning is a very broad domain, the discussion will be limited to the application of machine learning and machine learning techniques as this is the most relevant part for this thesis. One possible definition for Machine learning is: "the science of getting computers to act without being explicitly programmed". To do so, machine learning uses pattern recognition to find patterns or structure

in the data. A simple example of machine learning is the Optical Character Recognition (OCR), where a computer recognises characters in pictures.

Let's get a look at the following example, to further explain how machine learning works. Suppose one has a price list of houses that are for sale combined with their total area. Logic sense dictates us that a bigger house will have a higher asking price than a smaller house. Therefore the asking price of a house is correlated to the asking price. Suppose you want to predict how much a certain home is worth, based on their area. This is possible with machine learning, first you need to train your machine learning algorithm with a list of asking prices and the corresponding area of the house. This should give you a coefficient, lets say you pay 1000 euro for each square meter. Once this is done you can predict prices of new houses based on the corresponding area.

This will give some reasonable results, but the algorithm will probably have some flaws. This is due to the fact that the area of the house is only one feature that determines the price, there are many other that we haven't taken into consideration. Looking with more detail at the data, i.e. adding additional features will thus improve the performance of our algorithm. For example, a house with 5 bedrooms is more expensive than a house with only 3 bedrooms.

Machine learning algorithms are responsible for finding the relation between features and the predicted value. There exist many machine learning algorithms, one way to group these algorithms, is to look at their produced output. In the asking price examples above, the output is a price, which is a continuous value. The OCR example from above, where characters are recognized in a picture is a classification problem, as there is only a limited set of characters.

Another way to group the algorithms is based on their training data. In the asking price examples above one gets labelled results; the asking price is given for each area, this is referred to as supervised machine learning. The other possibility is unsupervised machine learning, which often results in finding groups of similar data points (clustering), without knowing the actual labels. Note that the combination of supervised and unsupervised data, also known as semi supervised learning, is also possible. Suppose you have a dataset with 5000 webpages and you want to categorise them in 10 distinct categories, e.g. science, nature, cooking, ... , but you only have the labels for 100 of the 5000 pages. Then you could first cluster the pages in similar groups using unsupervised learning. As soon as a group contains a labelled page, you can label all the pages in the group, since clustering returns groups of similar examples. Semi supervised learning has the advantage that one can also use unlabelled data. Unlabelled data is often easy and cheaper to obtain, unlike labelled data which is usually quite rare; if you had a fast and easy way to label the data then you wouldn't be needing machine learning.

In this thesis machine learning is used to find patterns in the aforementioned features that indicate the user's emotional state. The general process of machine learning is shown in Figure 1.3, the process starts with gathering EEG data, from which features are selected. These features are then fed to a machine learning algorithm, which outputs a prediction.

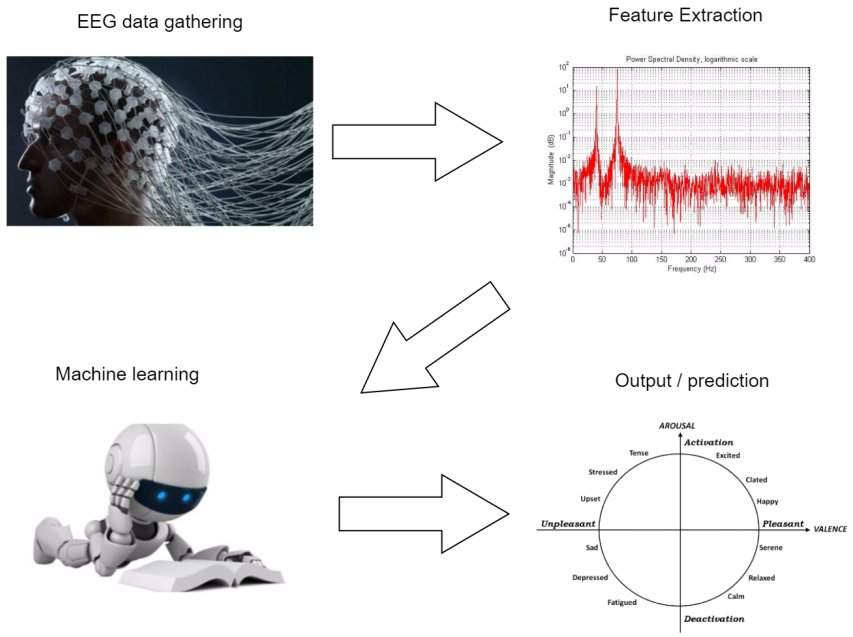

Figure 1.3: Basic steps of machine learning.

1.3.1 Over and underfitting / high bias and high variance

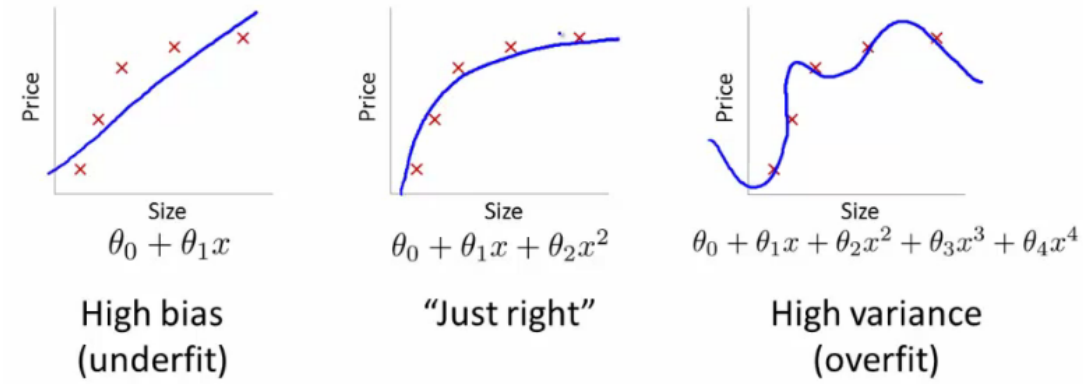

Figure 1.4: Overfitting versus underfitting[12].

Suppose the example in Figure 1.4, where one tries to find a good function to fit the given data points. Looking at the three proposed functions, one can easily see that the middle figure is the most likely generator function of the red points.

The figure on the left corresponds to an underfit, where the proposed function is not able to capture sufficient detail of the points. The function is not complex enough to approach the generator function, which is known as high bias. A high bias problem has a high training error, as the function is not able to fit the points correctly, this is visible in Figure 1.5

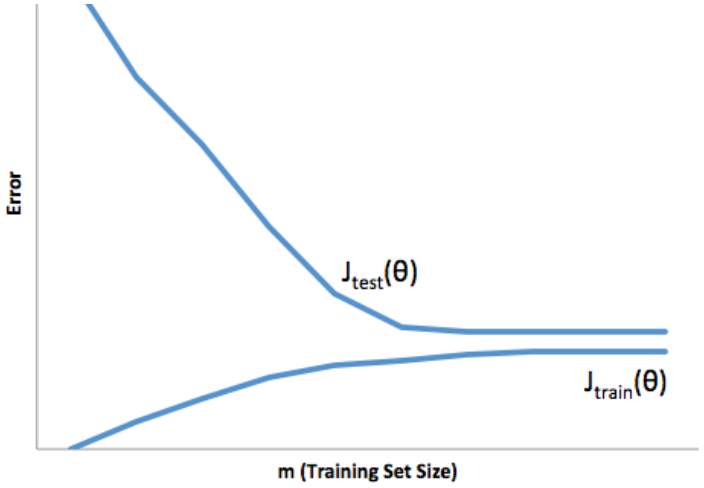

Figure 1.5: A high bias function is not complex enough to approach the generator function closely.

The function on the rights The function on the right corresponds to an overfit; the function 'goes through' each point exactly, but one can see that in between data points the behaviour of the hypothesis function is not logical. This problem is known as a high variance problem, where the train error is close to zero, but the test error is quite dramatic.

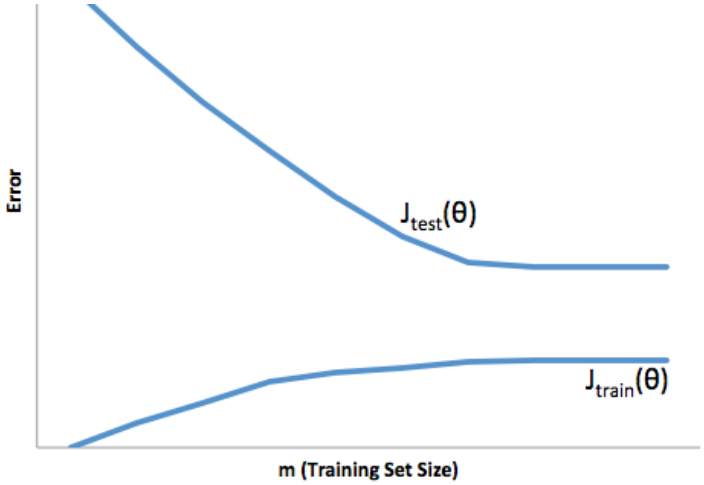

Figure 1.6: A High variance function is too complex and fits the data point too closely.

Another way to explain the bias variance tradeoff is by an example. Suppose you have a dart board, as shown in Figure 1.7. Suppose the situation on the top left corner, this corresponds to a world class player that has perfect aim, and very little variation on his precision. The situation on the left bottom corresponds to a player that has very little variation on his precision, but that is consistently aiming too high, he is biased to hit higher than needed. The pictures on the right side are different, there the person may or may not have a biased aim, but it is clear that he has a lot of variation in the precision of his aim.

In the context of machine learning, the low bias corresponds to having a hypothesis set that is close to the generator function, which allows you to get quite close. However you still have to pick the right function from that set, which is hard to do if you don't have enough data. If you are not able to take the best solution from the hypothesis set, you have high variance.

Figure 1.7: The bias variance explained using the dartboard example found at [13]

1.4 Goal of the thesis

The first goal is finding relevant features for emotion recognition in a person specific setting. This algorithm tries to accurately predict the emotions of one person, using only training data from that person. This is already quite challenging as there are fuzzy boundaries and individual variation of emotion. Emotion is function of context, space, time, language culture and race. [14]

The Second goal is finding features for emotion recognition in a cross-persons setting. In this setting the features should generalise well across different persons, thus the algorithm should be able to recognize emotions from unseen persons. Emotion recognition is harder in a cross-subject setting, since EEG features for emotion recognition are very personal[15].

The main problem is that there are already a lot of features known, but, as is often the case with EEG data, training data is expensive and limited. Using a lot of features will thus quickly result in overfitting. Additionally, a lot of different features are reported in the literature, as you can see in Table 1.1.

Another point to note is that even though a simple limited set of features might work, it is more likely to have less accurate results as optimal features might have been left out. This problem is even more severe when considering that EEG data is person specific[15], features that work good for one person, might not work for another person. Finding a good set of features that

Table 1.1: Six different papers on emotion recognition, six different feature sets

study	**features used**
[16]	Alpha and beta power
[17]	PSD and asymmetry features
[18]	PSD
[19]	discrete wavelet transform of alpha, beta and gamma band
[2]	alpha/beta ratio, Fpz beta and alpha band power
[11]	PSD, RCAU, DCAU, DASM, RASM, DE

works for all persons is a non trivial problem. One solution to this problem could be to use a large pool of possible features from which a limited set of good features are selected. This allows to provide good features to the machine learning algorithm, while still keeping the set of features limited in size. The machine learning algorithm which features to use and which to neglect.

1.4.1 Dataset

One of the most used datasets in the context of emotion recognition is the Dataset for Emotion Analysis using Physiological Signals (DEAP) dataset[15].

This dataset consists of several parts, the first part is a rating of 120 music videos by 14 - 16 persons. Each video is rated for valence, arousal and dominance on a scale ranging from 1 to 9. This part of the dataset is not used during this thesis, because it contains no EEG recordings.

The next part of the dataset is the physiological experiment that contains emotional reactions of 32 subjects. The emotional reactions were triggered using music video excerpts; each subject watched 40 one-minute videos, while several physiological signals were recorded. These physiological signals consist of 32 channel 512Hz EEH and peripheral physiological signals. More concretely, this dataset contains following signals:

There also exists a preprocessed version of the physiological experiment database, which has downsampling to 128Hz, noise removal and EOG artifact removal. This dataset is used during this thesis.

Additionally facial video of 22 out of 32 subjects was recorded, so research in facial expressions is also possible with this dataset. These videos are also rated on 4 scales: arousal, valence, dominance and liking. The liking component indicates how much the person liked the video excerpt and should not be confused with the valence component; it inquires information about the participants' tastes, not their feelings, i.e. a person can like a video that triggers angry or sad emotions. However strong correlations were observed[15]. The liking rates are neglected, since they are not part of the emotion space.

For assessment of these scales, the self-assessment manikins (SAM), were used[15]. SAM visualizes the valence, arousal and dominance scale with pictures, each picture corresponds to a discrete value. The user can click anywhere in between the different figures, which makes the

Table 1.2: The available signals in the DEAP dataset

channel	name	category	channel	name	category
1	Fp1	EEG	**21**	F8	EEG
2	AF3	EEG	**22**	FC6	EEG
3	F3	EEG	**23**	FC2	EEG
4	F7	EEG	**24**	Cz	EEG
5	FC5	EEG	**25**	C4	EEG
6	FC1	EEG	**26**	T8	EEG
7	C3	EEG	**27**	CP6	EEG
8	T7	EEG	**28**	CP2	EEG
9	CP5	EEG	**29**	P4	EEG
10	CP1	EEG	**30**	P8	EEG
11	P3	EEG	**31**	PO4	EEG
12	P7	EEG	**32**	O2	EEG
13	PO3	EEG	**33**	hEOG	non-EEG
14	O1	EEG	**34**	vEOG	non-EEG
15	Oz	EEG	**35**	zEMG	non-EEG
16	Pz	EEG	**36**	tEMG	non-EEG
17	Fp2	EEG	**37**	GSR	non-EEG
18	AF4	EEG	**38**	respiration belt	non-EEG
19	Fz	EEG	**39**	plethysmograph	non-EEG
20	F4	EEG	**40**	temperature	non-EEG

scales continuous. All dimension are given by a float between 1 and 9. In this thesis, a preprocessing step scaled and translated these values to ensure they range between 0 and 1, a more convenient interval.

The used SAM figures are shown in Figure 1.8. The first row gives the valence scale, ranging from sad to happy. The second row shows the arousal scale, ranging from bored to excited. The last row represents the different dominance levels. The left figure represents a submissive emotion, while the right figure corresponds with a dominant feeling.

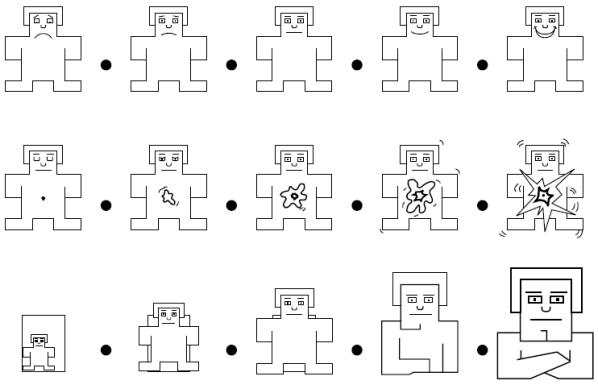

Figure 1.8: The images used for the SAM[15].

2

Methods

This chapter starts by going over the possible features for emotion recognition, before going through some emotion recognition studies found in the literature. The last part will go over different feature selection techniques.

2.1 Features

Usually, good features are needed before one can train a machine learning algorithm. Two categories of features are observed: EEG features and non-EEG features. This section will cover them briefly.

2.1.1 EEG-features

EEG features are extracted from the electroencephalography measurements from the subject's scalp. This section will go through the used EEG features in this thesis.

The power spectral density (PSD) of a signal gives the distribution of the signal's energy in the frequency domain. By calculating the spectral density for different waveband of the signal, one can determine how much alpha, beta, ... power is in the signal.

Differential entropy (DE) is defined as follows [11]

$$-\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} exp(\frac{(x-\mu)^2}{2\sigma^2}) log(\frac{1}{\sqrt{2\pi\sigma^2}}) exp(\frac{(x-\mu)^2}{2\sigma^2}) dx$$

According to [20], the differential entropy of a certain band is equivalent to the logarithmic power spectral density for a fixed length EEG sequence, which simplifies the calculations significantly.

The most known feature for valence recognition is the frontal asymmetry of the alpha power[3]. The right hemisphere is generally speaking, more active during negative emotion than the left hemisphere which is in turn more active during positive emotions[10, 9, 11]. The asymmetry can be calculated in different ways, one of them is the differential asymmetry (DASM) , where the left alpha power is subtracted from the right alpha power.

$$DASM = DE_{left} - DE_{right}$$

Another way to measure the asymmetry if by division. The Rational Asymmetry (RASM) does exactly this and is given by:

$$RASM = \frac{DE_{left}}{DE_{right}}$$

With DE_{left} and DE_{right} being the left and right differential entropy respectively.

Another reported feature in literature is the caudality, or the asymmetry in fronto-posterior direction[21]. This can again be calculated in two ways. The first method is the differential Caudality (DCAU) is defined as:

$$DCAU = DE_{front} - DE_{post}$$

Another way to determine the Caudality is the Rational Caudality (RCAU) , which is defined as:

$$RCAU = \frac{DE_{front}}{DE_{post}}$$

With DE_{front} and DE_{post} being the frontal and posterior power respectively.

One way to determine the arousal is by looking at the different wavebands. Each waveband has their own medical interpretation, see 1.1.1. More alpha power corresponds to a more relaxed brain, while more beta power corresponds to a more active brain. The alpha / beta ratio therefore seems a good indicator for the arousal state of a person.

The Alpha/ beta ratio is limited to comparing two wavebands. Other frequently used features are powerband fractions. Where the fractions of waveband power is determined for a channel, given by:

$$frac_{band} = \frac{power_{band}}{power_{total}}$$

2.1.2 non-EEG features

The aforementioned EEG features are just one class of physiological features. The DEAP dataset contains several physiological measurements, listed below [15]. For each of these measurements the average, standard deviation, variation, median, minimum, maximum and the standard deviation is calculated.

The Galvanic Skin Response uses two electrodes on the middle and index finger of the subjects left hand to measure the skin resistance. It has been reported that the mean value of the GSR is related to the level of arousal[22, 15].

The respiration belt, indices the user's respiration rate. Slow respiration is linked to relaxation (low arousal), while fast and irregular respiration patterns corresponds to anger or fear, both emotions with low valence and high arousal[15].

A plethysmograph is a measurement of the volume of blood in the subject's left thumb. This can be interpreted as the the blood pressure. Blood pressure offers valuable insight into the emotional state of a person as it correlated with emotion; stress is known to increase blood pressure[15].

The heart rate is not explicitly in the DEAP dataset, but can be extracted from the plethysmograph, by looking at local minima and maxima[15]. This is clearly visible when looking at the plethysmograph's output, shown in Figure 2.1.

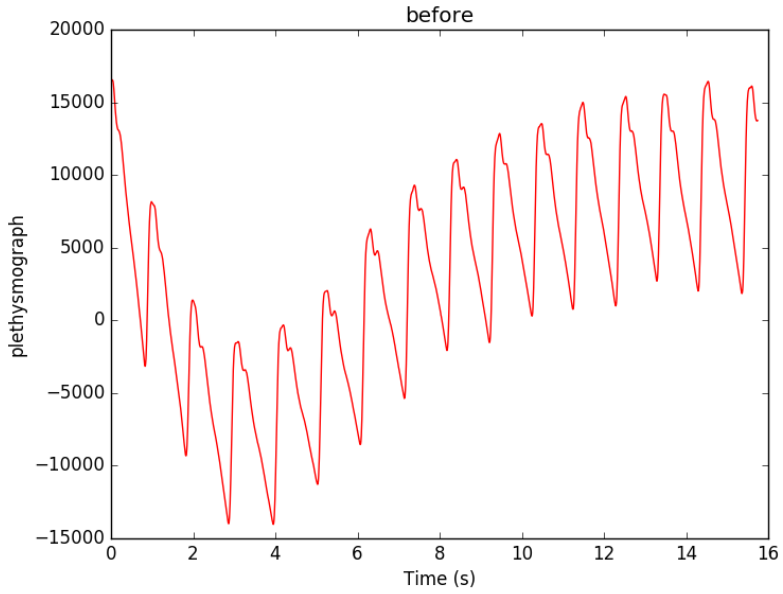

Figure 2.1: The plethysmograph before smoothing.

The heart rate extraction is done in two steps. First the plethysmograph signal is smoothed using a butter filter to avoid noise being selected as local optima. In the second stage the local optima are located, which is shown in Figure 2.2

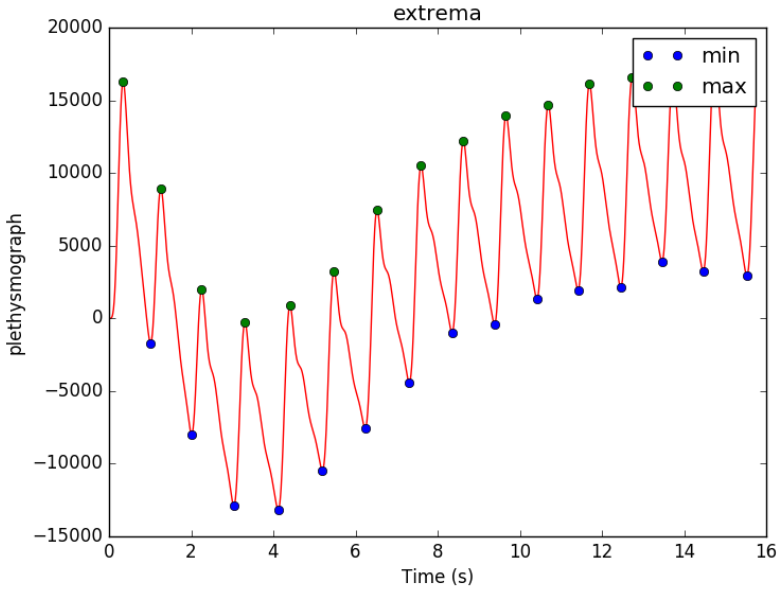

Figure 2.2: The local optima in the plethysmograph.

These optima correspond to a heart beat, therefore the time between two consecutive local minima or maxima corresponds to the time between two heart beats, known as the interbeat interval. Getting the average heart rate from the interbeat interval is straight forward.

The last physiological feature is the skin temperature of the subject.

2.2 Emotion recognition Studies

This section will give a overview of (some) other studies that did similar research and their conclusions.

2.2.1 DEAP method

The first method of emotion recognition is the DEAP method, described in the DEAP paper[15]. The research found that Valence shows the strongest correlations with the EEG signals. Additionally the study found correlations in all frequency bands, with an increase in power for the lower range wavebands for an increase in valence. These effects occur in the occipital regions of the brain, above the visual cortices, which might indicate that the subject is focussing on a pleasurable sound. A central decrease in beta power was observed together with a occipital and right temporal increase in power for positive emotions. The research conclude that these observed correlations concur with other neurological studies, but that the absolute value of the correlations are seldom bigger than 0.1 for a cross person setting. For a person specific setting, the absolute values of correlations were around 0.5. Getting a single classifier to work for all participants, is still an ongoing topic of research.

The DEAP paper also present their own classification method for person specific emotion classification. They start by performing features selection using the Fisher's linear discriminant for feature selection. The Fisher's linear discriminant is defined as

$$J(f) = \frac{|\mu_1 - \mu_2|}{\sigma_1^2 + \sigma_2^2}$$

With μ and σ being the mean and standard deviation of feature f.

The Fisher's discriminant was calculated for each feature, before a threshold of 0.3 was applied. The used classifier was a Naive Bayes classifier, which assumes independence of features. The Naive Bayes classifier is a simple classifier that uses the following equation:

$$G(f_1, ..., f_n) = argmax_c p(C = c) \prod_{i=1}^{n} p(F_i = f_i | C = c)$$

With F being the set of features and C the classes. $p(F_i = f_i | C = c)$ is estimated by assuming Gaussian distributions of features and modelling these from the training set.

2.2.2 Stable Emotion Recognition over Time

In [11], research is done to find EEG patterns for emotion recognition that are stable over time. EEG patterns are not only subject dependent, they are also dependent on the subjects mood and thus might vary in time. The paper starts by researching different EEG features: PSD, DE, DASM, RASM, DCAU, RCAU, these features are explained in 2.1.

Their machine learning set-up is as follows, first they perform feature extraction of the aforementioned features. Then feature smoothing is done using a Linear Dynamic system (LDS) , that can be expressed by:

$$x_t = z_t + w_t$$
$$z_t = Az_{t-1} + v_t$$

x_t denotes the observed variables or features, while z_t denotes the hidden emotion variables. A is a transformation matrix and w_t is Gaussian noise. The need for a linear dynamic system is supported by the assumption that emotion change gradually over time. The LDS filters out components that are not associated with emotional states.

The list of features at this point is too big and may contain uncorrelated features that might lead to performance degradation of the classifier. Two methods for this are compared, principal component analysis (PCA) and minimal redundancy maximal relevance (MRMR).

PCA uses an orthogonal transformation to create a lower dimensional feature space starting from the original higher dimensional feature space. It does so by minimizing the loss of information, i.e. the principal component should have the largest possible variance.

PCA cannot preserve original domain information like channel and frequency, therefore the paper also uses the MRMR method. MRMR uses mutual information in combination with maximal dependency criterion and minimal redundancy. The algorithm starts by searching features satisfying:

$$maxD(S,c), D = \frac{1}{|S|}\sum_{x_d \in S} I(x_d;c)$$

Where S is the feature subset to select. When two features are highly correlated, the maximal dependency is not likely to change when one of the correlated features is removed. This is expressed by the minimal redundancy condition.

$$minR(S), R = \frac{1}{|S|^2}\sum_{x_{di},x_{dj} \in S} I(x_{di}, x_{dj})$$

The two conditions are then combined to from the Maximal Relevance Minimum Redundancy, which can be expressed as:

$$max\varphi(D,R), \varphi = D - R$$

Note that incremental search methods exists and are often used in practice. After performing the dimensionality reduction, the samples from the DEAP data set are classified in high / low valence and high/low arousal, giving a total of four classes. All values close to the separation border are removed from the training data, as they might confuse the classifier.

For the classification, three conventional and one newly developed pattern classifiers were compared. k-nearest neighbors (KNN) , logistic regression (LR), Support Vector Machines (SVM) and Graph regularized Extreme Learning Machine (GELM) .

Extreme Learning Machine (ELM) is a single layer feed forward neural network[23]. GELM is based on the idea that similar shapes should have similar properties and obtains better results for face recognition[**?**] and as the paper concludes, also for emotion classification.

The study found then performed a study on the different features and concluded that DE features are the most suitable EEG features, followed by the asymmetry features (RASM, DASM, DCAU and RCAU). The LDS smoothing was also found to be the better feature smoothing method.

2.3 Feature selection methods

Feature selection is the process of selecting good features from a set of feature. The need for this is twofold: first by reducing the number of features, you can protect yourself against overfitting. This is important when the dataset is limited. Second, reducing the number of features can speedup the learning process of a learning algorithm as fewer parameters need to be optimized. Additionally, in the context of research and this thesis, looking at which features are important gives insight in the problem. In the context of this thesis, knowing what feature are relevant can help neuroscientists understand the working of the brain better.

2.3.1 Independent Metrics

These feature selection methods select features based on statistical tests or another independent metric.

Pearson Correlation

The Pearson correlation coefficient measures the linear relationship between two variables. The output is a value r, that lies between -1 and 1, corresponding to perfect negative correlation and perfect positive correlation respectively. A correlation value of 0 means that there is no correlation.

More formally[24], the Pearson product-moment coefficient of correlation, r between variables X_i and Y_i of datasets X and Y is defined as:

$$r = \frac{SS_{xy}}{\sqrt{SS_{xx}SS_{yy}}}$$

with

$$SS_{xy} = \sum_i (X_i - \tilde{X})(Y_i - \tilde{Y})$$

and

$$SS_{xx} = \sum_i (X_i - \tilde{X})^2$$
$$SS_{yy} = \sum_i (y_i - \tilde{Y})^2$$

The Pearson correlation coefficient is fast and simple to calculate, but has some major shortcomings. First off, it can only see linear relation ships and will not see the correlation between a value x and x^2.

In the context of this thesis, whether the correlation is positive or negative is not important; a learning algorithm needs features that have significant correlation. As a result the absolute value of the r value is reported as this allows for faster comparison of correlations.

Mutual Information

Mutual information is a more robust option for correlation estimation. The mutual information, I, of two variables X and Y is defined as [25]:

$$I(X,Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) log(\frac{p(x,y)}{(x)p(y)}$$

Using the mutual information directly for feature ranking might be inconvenient for two reasons. Firstly, it doesn't lie in a fixed range and it is hard to compute for continuous variables. One solution for this problem is to normalize the mutual information scores, so that the results lies between 0 and 1.

The normalized mutual information, NMI of variables X and Y is given by:

$$NMI(X,Y) = \frac{H(X)+H(Y)}{H(X,Y)}$$

With $H(X)$ and $H(Y)$ being the Shannon entropy of variable X and variable Y, defined as:

$$H(X) = \sum_{i \in X} p_i log(\frac{1}{p_i}) = -\sum_i p_i log(p_i)$$
$$H(Y) = \sum_{i \in Y} p_i log(\frac{1}{p_i}) = -\sum_i p_i log(p_i)$$

Distance Correlation

Distance correlation is a relatively new technique that is designed explicitly to address shortcomings of Pearson correlation. A Pearson correlation coefficient of zero implies that the variables might be independent, but as mentioned before, does not guarantee this.

The distance covariance is defined as[26]:

$$dCov^2(X,Y) = \frac{1}{n^2} \sum_l imits_{k,l=1}^{n} A_{k,l} B_{k,l}$$

With A, B being simple linear functions of the pairwise distances between sample elements. This metric is a covariance metric, which means that it is not normalized. The distance correlation is the normalized version of the distance covariance and is defined as:

$$dCor(X,Y) = \frac{dCov(X,Y)}{\sqrt{dVar(X) dVar(Y)}}$$

With $dCov(X,Y)$ being the aforementioned distance covariance, $dVar(X)$ and $dVar(Y)$ are the distance standard deviations.

The distance correlation has the disadvantage that is much slower than mutual information or Pearson correlation, but in return, the distance correlation is able to detect more complex relationships between two variables.

2.3.2 Machine Learning Methods

These methods select features by applying an arbitrary machine learning technique and looking at the coefficients of the features.

Linear Regression

Another way of finding relevant features is to use model based ranking. In model based ranking an arbitrary machine learning method is used to build a model. Looking at the coefficients of the trained model, the importance is determined by its own coefficient. High coefficients mean that the feature has a lot of influence on the output, while low coefficients correspond to less important features.

The first method to use is simple linear regression. This method tries to find a linear combination of features that produces the output value. Linear regression can achieve good results given that the data doesn't contain a lot of noise and the features are (relatively) independent. When the set

of features contains correlated features, the model becomes unstable. As a result, small changes in input data might lead to huge differences in output coefficients. for example assume the 'real output' is given by $Y = X_1 + X_2$ and the dataset contains output in the form of $Y = X_1 + X_2 + \epsilon$ with ϵ being some random noise. Further more assume that X_1 and X_2 are linearly correlated, meaning that $X_1 \approx X_2$. The suspected output of the model should be $Y = X_1 + X_2$, but since noise is added the algorithm might end up with arbitrary combinations of X_1 and X_2, e.g. $Y = -X_1 + 3X_2$ and rate one feature much higher than another one, while in reality they are of equal importance. This is due to the noise; by maximizing the performance, the algorithm will minimize the influence of noise on the output, which result in unstable behaviour when sufficient correlated features are present.

Lasso Regression

Lasso regression uses L1 regularization, that adds a penalty $\alpha \sum_{i=1}^{n} |w_i|$ to the loss function. the result is that the coefficients of weak features are forced to zero, as each non-zero feature adds to the penalty. This form is regularization is thus quite aggressive, it removes weak features completely. The problem with this is, again, stability; coefficient can vary significantly even for small changes in training data, when there are correlated features.

Ridge Regression

Ridge regression uses L2 regularization, which add a L2 norm penalty to the loss function, given by $\alpha \sum_{i=1}^{n} w_i^2$. Where the L1 norm forces the coefficients to zero, the L2 regularization forces the coefficients to be spread out more equally. The result is that correlated features tend to get similar coefficients, as this minimizes the loss function, which in turn results in a more stable model.

SVM

Just like Regression uses linear regression to get coefficients, it is also possible to use SVM for feature importance estimation.

Random Forests

Random forests (RF) is a efficient learning algorithm based on model bagging and aggregation ideas[27]. The Random forests work by creating different decision trees. On their own, decision trees are very prone to overfitting. Random forests solve this problem by creating an aggregation of trees.

Additionally, some randomness is included, each tree looks at a random subset of the samples and a random subset of the features. This principle is shown in Figure 2.3. This random subset of samples is called the bootstrap sample and is selected out of N samples, by picking N times a sample, with replacement. This results, on average, in 2/3 of the samples being selected (with some doubles). The other 1/3 of the samples are then used as out of bag (oob) set. Averaging the performance of each tree on the out of bag set, offers an indication of the generalisation of the random forest.

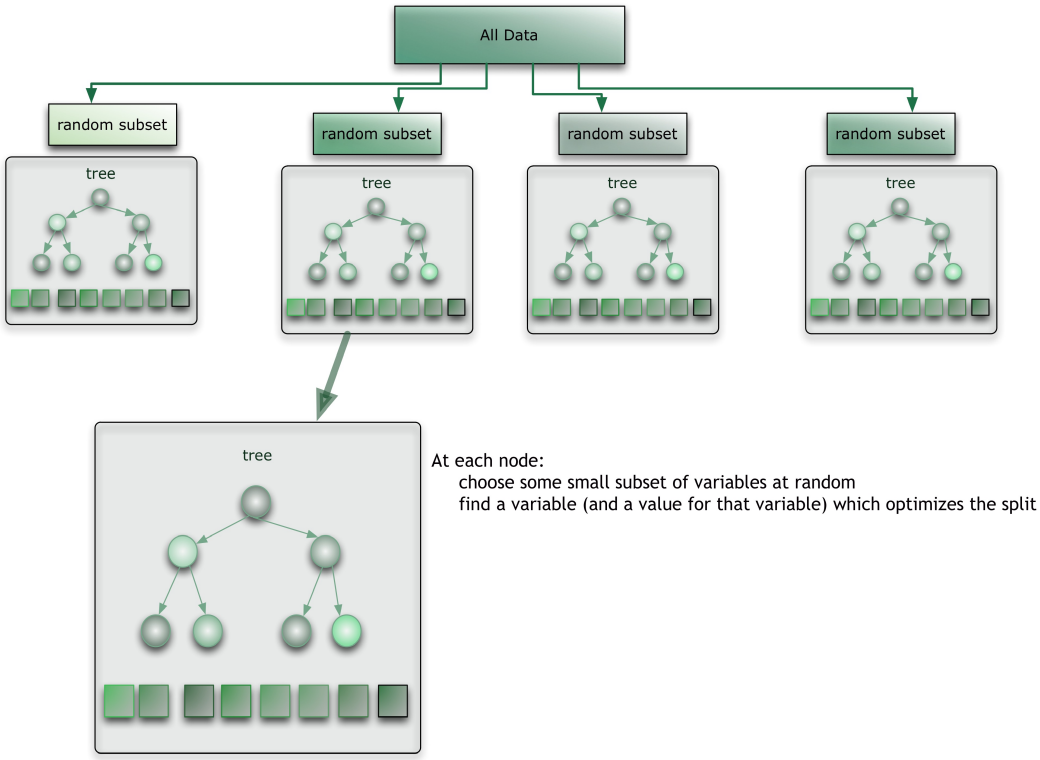

Figure 2.3: The structure of a random forest, found at [28]

To understand which features are good, one needs to understand the internal workings of a decision tree. Suppose the following example[1], where one tries to find an algorithm to predicted whether or not a person will play tennis on a given day. Suppose the training data is given by Table 2.1 and a prediction for the 15^{th} sample needs to be made.

[1]This example is based extensively on this youtube video: https://www.youtube.com/watch?v=eKD5gxPPeY0

Table 2.1: suppose the following training examples for a decision tree.

Day	Outlook	Humidity	Wind	Play tennis
1	sunny	high	weak	no
2	sunny	high	strong	no
3	overcast	high	weak	yes
4	rain	high	weak	yes
5	rain	normal	weak	yes
6	rain	normal	strong	no
7	overcast	normal	strong	yes
8	sunny	high	weak	no
9	sunny	normal	weak	yes
10	rain	normal	weak	yes
11	sunny	normal	strong	yes
12	overcast	high	strong	yes
13	overcast	normal	weak	yes
14	rain	high	strong	no
15	rain	high	weak	?

A decision tree will take a feature and split the data based on the possible outcomes of this feature. In case the features are continuous values, ranges are selected. In some cases the leafs will be pure, like the leaves displayed in green in Figure 2.4. All examples in here have the same output. In case the leave is not pure, then another split is needed. Note that not all random forests split until all leaves are pure; random forest can be limited in depth, in that case the output is chose by a majority voting of the samples.

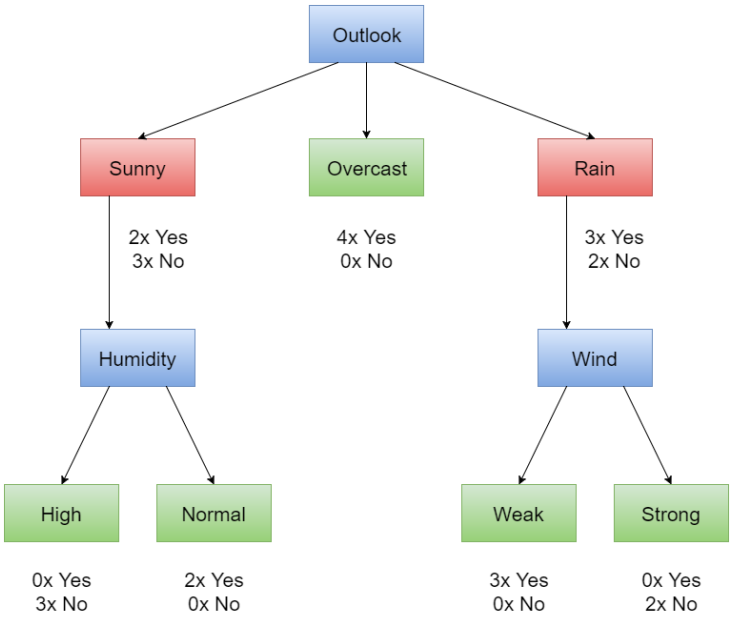

Figure 2.4: A decision tree for the data in Table 2.1

Once the tree is constructed it becomes clear that the predicted output of sample 15 is yes. This is obtained simply by following the tree branches. Even though the features are selected at random, they have influence on the accuracy. Good features will reduce the impurity significantly, thus the impurity reductions are a good indication for how important a feature is.

Since the importance is averaged over different nodes and different trees, it is also capable of detecting combinations of features that work well. One feature may not be important on its own, but might be a very good feature when combined with other features. Suppose the following example in Table **??**:

label	**feature A**	**feature B**
Happy	+	+
Happy	-	-
Sad	-	+
Sad	+	-

Table 2.2: Some feature are not significant on its own, but a might be part of a combination of features.

It is clear that feature A and B are very important when it comes to predicting whether or not a person is happy or sad. When both features have the same sign, the person is happy, otherwise he is not. This problem occurs in many simple selection methods.

This problem does not occur for random forest though, as combinations of features are also 'tested' in the sense that a tree might split on them in different stages. Once the combination of features occurs randomly in a decision tree, the impurity will drop significantly, which will result in higher importance rankings.

2.3.3 Dimensionality Reduction methods

The algorithms described below perform a dimensionality reduction, often by projecting a high dimensional to a lower dimensional features space. Looking at coefficients of these trained models, gives insight in which features are important. Note that some of this methods could also be seem as machine learning algorithms.

Common Spatial Patterns

Common Spatial Patterns (CSP) is a supervised technique that has its origin in the optimization of motor imagery BCIs[29]. It is a common technique in BCI research[30, 31, 32]. CSP creates linear combinations of the original EEG channels that maximize the variance for one class while simultaneously minimizing the variance of the other class [30]. One disadvantage of using CSP is that the default version can only distinguish between 2 classes, though one can easily aggregate multiple CSP models to create one-vs-one and one-vs-all models, similarly to the one-vs-one and one-vs-all SVMs.

The input for a CSP filter is a set of N labelled samples $E_j (j = 1...N)$, with dimension N_{ch} x T_j, with N_{ch} being the number of EEG channels and T_j the number of samples in a single trial[29].

First the train data is split into two classes, before computing the covariance matrices of both classes.

$$\Sigma_1 = \sum_{j \in C_1} X \frac{E_j E_j^T}{trace(E_j E_j^T)}$$
$$\Sigma_2 = \sum_{j \in C_2} X \frac{E_j E_j^T}{trace(E_j E_j^T)}$$

Note that the average of E_j is expected to be zero, because a bandpass filter is applied that make the DC component of the signal zero. The next step is to calculate the composite covariance matrix.

$$\Sigma = \Sigma_1 + \Sigma 2$$

Next the covariance matrix is diagonalised by calculating the eigenvalues and eigenvectors of Σ.

$$V^T \Sigma V = P$$

The eigenvalues are then found on the diagonal of P, each eigenvalue corresponds to an eigenvector found in the columns of V.

The next step is the whitening transformation.

$$U = P^{\frac{1}{2}} V^T$$

Which results in

$$U \Sigma U^T = 1$$

Next the following two matrices are calculated:

$$R_1 = U \Sigma_1 U^T$$
$$R_2 = U \Sigma_2 U^T$$

R_1 is then diagonalised

$$Z^T R_1 Z = D = diag(d1, ..., d_m)$$

The eigenvalues on the diagonal are then sorted, as larger eigenvalues correspond to higher importances. Next the filters are determined by:

$$W = Z^T U$$

The EEG channels can then be filtered as follows:

$$E^{CSP} = W E^{orig}$$

Since CSP filters create simple linear combination of incoming channels, they can also be used as feature selection mechanism. The first and last row of the resulting matrix W shows the coefficients for which the variance is maximized between the two signals. Looking at those coefficients, one can determine which channels are of more importance than other.

Linear Discriminant Analysis

Linear Discriminant Analysis (LDA), is a machine learning technique often used in combination with CSP[30, 31, 32]. LDA looks for a projection of the data where the data is linearly separable, as shown in Figure 2.5. Looking at the coefficients of the LDA model, one can again determine the importance of the different features.

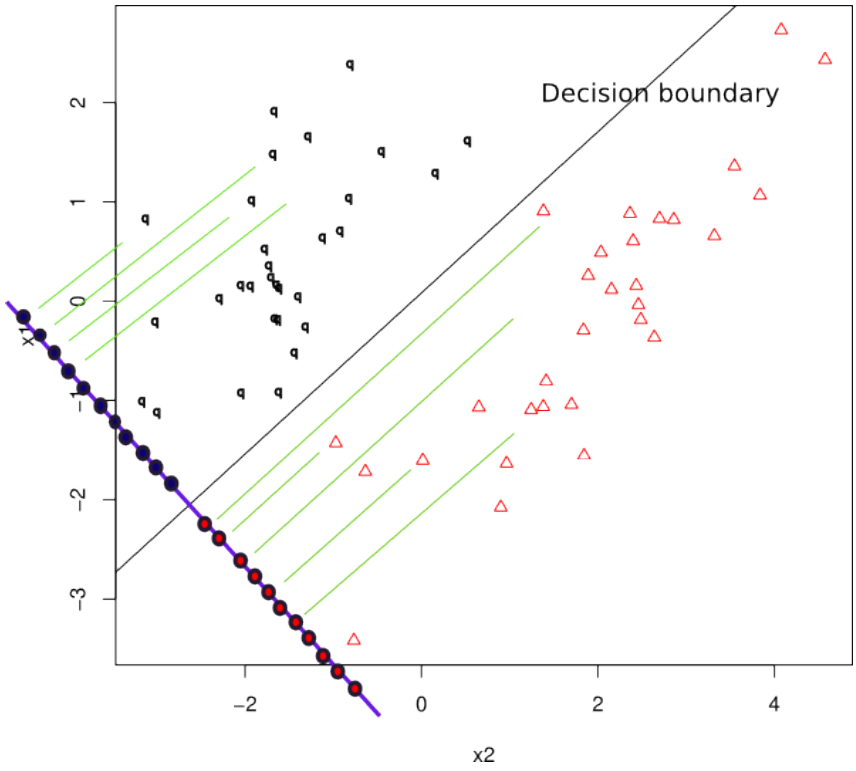

Figure 2.5: LDA finds a projection of the data where the separation of the data is clear.

Principal Component Analysis

Principal Component Analysis (PCA) is a technique to do dimension reduction. Intuitively, PCA can be seen as fitting an n-dimensional ellipsoid to the data. The Principal components are then the axes of the ellipsoid. Less variation in one direction, corresponds to a smaller axis, removing that axis, will only remove a small fraction of the information. This is shown in Figure 2.6, where the ellipsoid covers a three dimensional features space. The ellipsoid has three axes: a,b and c. Intuitively, one can see that there is more variation (information) in the c ans b direction, while the a axis is relatively small.

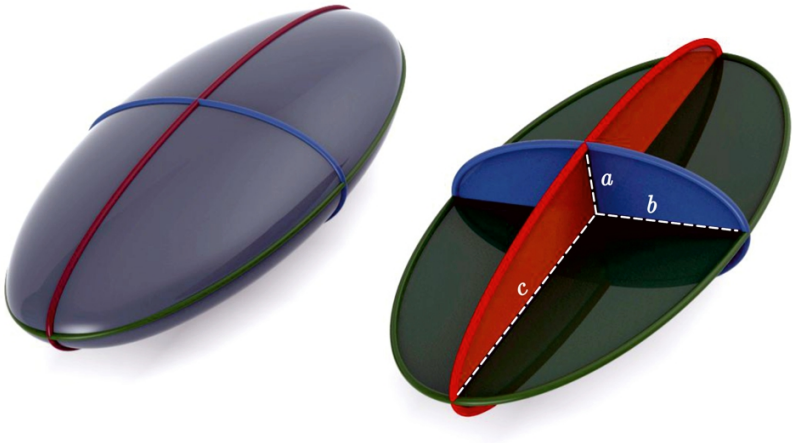

Figure 2.6: Suppose a three-dimensional feature space, where all points lie in the ellipsoid in the left.

Remove the a axis by projecting the data on the plane given by vectors b and c, and one will end up with a two dimensional projection of the data in the form of an ellipse. This would be the black plane in Figure 2.6. This process can be repeated for higher dimensional features spaces. In other words, PCA will thus, without going into too much detail, start with an n-dimensional ellipsoid and iteratively remove the smallest axis in each iteration until the desired number of dimensions is obtained. Note that the ellipsoid should be adjusted in each step.

2.3.4 Advanced methods

These methods are more advanced feature selection methods found in the literature.

RF feature selection

One advanced method for feature selection is the two-step method using random forest, described in [27]. The paper states that there are two possible motivations for feature selection. The first motivation is to do interpretation, find out which features are important and use them for research. In the context of BCI, feature interpretation could help neuroscientist find out which parts of the brain are affected by an emotion, for example. The second motivation is to improve machine learning techniques, having fewer features will not only speed up training and prediction times, it also reduces the complexity, which often has a good influence on the generalisation property of a machine learning algorithm. Additionally in the context of BCI research and EEG data gathering, using fewer electrodes means less preprocessing time; mounting 32 electrodes to the brain of a subject is a time consuming task.

The selection procedure itself consists of two steps, in the first step data is fitted to a random forest and the importance values for each feature are determined, by taking the average and standard deviation of the importances over all trees. All features are then ranked based on their importance ranking, before features with small importance are cancelled.

Then depending on the motivation of feature selection, a second step is performed. For feature interpretation the second steps by fitting a random forest with a single feature. The OOB is then averages over multiple runs. the runs are inserted because a random forest has an element of randomness, fitting the same data twice to a random forest, will not give you the same random forest. The average OOB score and its standard deviation is then used to determine an initial OOB score.

$$OOB_{init} = AVG(OOB) - STD(OOB)$$

The standard deviation is used to avoid noisy results, a result is only regarded as better, when there is statistical prove. Next features are added iteratively, when a larger features set has a better average OOB score (taking the standard deviation into account), the feature set is replaced by the larger feature set.

The other second step is used for prediction, here the algorithm starts similarly, by determining an initial average OOB score and standard deviation. The idea behind the standard deviation is the same as with the interpretation step, noise removal.

$$OOB_{init} = AVG(OOB) - STD(OOB)$$

The next part is different, now a feature is introduced in each iteration. When the average OOB score of the feature is better, the feature is added to the features set, otherwise it is neglected. This is a greedy forward selection algorithm, once a feature is selected it remains selected. The difference between step two interpretation and step two prediction is that here single features are added to the feature set, while step two-interpretation always takes the feature set containing all features with higher importance than the lastly added feature. step two prediction on the other hand is able to select a distinct set of features out of the results from step one.

In the end the paper notes several observations, the step two-prediction method provides better OOB scores using fewer features. Additionally they mention that highly correlated features might confuse the algorithm, as correlated features have lower importances.

Bibliography

[1] H. Verschore, "A brain-computer interface combined with a language model: the requirements and benefits of a p300 speller," afstudeerwerk, Ghent University, June 2012.

[2] D. O. Bos, "Eeg-based emotion recognition," 2007.

[3] M.-K. Kim, M. Kim, E. Oh, and S.-P. Kim, "A review on the computational methods for emotional state estimation from the human eeg," *Computational and Mathematical Methods in Medicine*, vol. 2013, no. 573734, p. 13, 2013.

[4] T. C. Technologies, *10/20 System Positioning manual*. Fortis Tower, 2012.

[5] unknown, "Electrode placement," 2015.

[6] Y. Yang, S. Chevallier, J. Wiart, and I. Bloch, "Time-frequency optimization for discrimination between imagination of right and left hand movements based on two bipolar electroencephalography channels," *EURASIP journal on Advances in Signal Processing*, vol. 2014, no. 38, 2014.

[7] K.-E. Ko, Hyun-Chang, and K.-B. Sim, "Emotion recognition using eeg signals with relative power values and bayesian network," *International Journal of Control, Automation, and Systems*, 2009.

[8] Brainworks, "What are brainwaves?," 2015.

[9] Y. Lio and O. Sourina, "Eeg databases for emotion recognition," *International Conference on Cyberworlds*, 2013.

[10] Y. Lio, O. Sourina, and M. K. Nguyen, "Real-time eeg based human emotion recognition and visualization," 2010.

[11] W. Zheng, J. Zhu, and B. Lu, "Identifying stable patterns over time for emotion recognition from EEG," *CoRR*, vol. abs/1601.02197, 2016.

[12]

[13]

[14] J. Kim and E. Andre, "Emotion recognition based on phys- Â´ iological changes in music listening," *IEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 12, pp. 2067–2083, 2008.

[15] S. Koelstra, C. Muhl, M. Soleymani, J.-S. Lee, A. Yazdani, T. Ebrahimi, T. Pun, A. Nijholt, and I. Patras, "Deap: A database for emotion analysis ;using physiological signals," *Affective Computing, IEEE Transactions on*, vol. 3, pp. 18–31, Jan 2012.

[16] D. O. Bos, "Eeg-based emotion recognition the influence of visual and auditory stimuli."

[17] Y. P. Lin, C. H. Wang, T. P. Jung, T. L. Wu, S. K. Jeng, J. R. Duann, and J. H. Chen, "Eeg-based emotion recognition in music listening," *IEEE Transactions on Biomedical Engineering*, vol. 57, pp. 1798–1806, July 2010.

[18] L. Brown, B. Grundlehner, and J. Penders, "Towards wireless emotional valence detection from eeg," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 2188–2191, Aug 2011.

[19] M. Murugappan, "Human emotion classification using wavelet transform and knn," in *Pattern Analysis and Intelligent Robotics (ICPAIR), 2011 International Conference on*, vol. 1, pp. 148–153, June 2011.

[20] R. N. Duan, J. Y. Zhu, and B. L. Lu, "Differential entropy feature for eeg-based emotion classification," pp. 81–84, Nov 2013.

[21] Y.-P. Lin, Y.-H. Yang, and T.-P. Jung, "Fusion of electroencephalographic dynamics and musical," 2014.

[22] P. LanG, M. Greenwald, M. Bradley, and A. Hamm, "Looking at pictures: affective, facial, visceral, and behavioral reactions.," *Psychophysiology*, vol. 30, pp. 261–273, May 1993.

[23] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1, pp. 489–501, 2006.

[24] S. L. Crawford, "Statistical primer for cariovascular research," 2006.

[25] J. P. Pluim, A. Maintz, and M. A. Viergever, "Mutual-information-based registration of medical images: A survey," *IEEE Transactions of Medical Imaging*, vol. 22, pp. 986–1003, August 2003.

[26] G. J. Szekely and M. L. Rizzo, "Brownian distance covariance," *The annals of Applied Statistics*, vol. 3, no. 4, pp. 1236–1265, 2009.

[27] R. Genuer, J.-M. Poggi, and C. Tuleau-Malot, "Variable selection using random forests," *Pattern recognition letters*, pp. 2225–2236, 2010.

[28] Citizennet, 2016.

[29] S. Vercruysse, "Adaptieve common spatial patterns voor de classificatie van ingebeelde bewegingen," afstudeerwerk, UGent, 2011.

[30] A. Coone, "A study on different preprocessing and machine learning techniques for the detection of error-potentials in brain-computer interfaces," afstudeerwerk, Ghent university, June 2011.

[31] F. Lee, R. Scherer, R. Leeb, C. Neuper, H. Bischof, and G. Pfurtscheller, "A comparative analysis of multi-class eeg classification for brain computer interface," in *Proceedings of the 10th Computer Vision Winter Workshop*, pp. 195–204, 2005.

[32] G. Pfurtscheller, C. Neuper, C. Guger, W. Harkam, H. Ramoser, A. Schlögl, B. Obermaier, and M. Pregenzer, "Current trends in graz brain-computer interface (bci) research," *IEEE Transactions on rehabilitation Engineering*, vol. 8, pp. 216–219, JUNE 2000.