A Report on EEG SIGNAL PROCESSING

By

SAI NIKHIL METTUPALLY **2011AAPS136H** **B.E. (Hons.)Electronics and Communication**

Prepared in the partial fulfilment of

Lab Oriented Project (ECE F366)

Under the Guidance of

Dr. BVVSN Prabhakar Rao

Assistant Professor, Department of EEE/ECE/EIE

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

Hyderabad Campus

SEMESTER II
2013 - 2014

CERTIFICATE

This is to certify that the project entitled - 'EEG Signal Processing' submitted by C K Shriram (2011A3PS340H), Kashyap L S J Josyula (2011AAPS074H) and Sai Nikhil Mettupally (2011AAPS136H) in the partial fulfilment of the requirements of Lab Oriented Project (ECE/EEE F366/F367) at Birla Institute of Technology and Science-Pilani, Hyderabad Campus, is an authentic work carried out by them under my supervision and guidance during Semester-II of the academic year 2013-2014.

Date:

Place: Hyderabad

Dr. BVVSN Prabhakar Rao

Assistant Professor, Department of EEE/ECE/EIE

BITS-Pilani, Hyderabad Campus

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

Hyderabad Campus

SEMESTER II

2013 - 2014

ABSTRACT

Electroencephalography (EEG) is the recording of electrical activity along the scalp. EEG measures voltage fluctuations resulting from ionic current flows within the neurons of the brain. In clinical contexts, EEG refers to the recording of the brain's spontaneous electrical activity over a short period of time, usually 20–40 minutes, as recorded from multiple electrodes placed on the scalp. EEG is most often used to diagnose epilepsy, which causes obvious abnormalities in EEG readings. It is also used to diagnose sleep disorders, coma, encephalopathies, and brain death.

In our current work we have studied and applied different neuro-signal processing methods based on EEG. EEG signals are classified into different bands. First we proceed with the artifact removal, then processing of EEG data and classification of the signals. The data set was recorded from the experiment conducted on 9 subjects under different conditions. They were told to move their left hand, right hand, legs and tongue. The data was collected by placing 22 EEG electrodes on the scalp and 3 electrodes for EOG data.

The raw data contains some artifacts (noise). This was removed by applying wavelet transform based adaptive filter. Also using the availability of EOG signal data, we can reduce the artifacts in the EEG data in all channels through estimation of the channel between the EOG and each channel of EEG electrodes. The estimation can be done by many different algorithms. In this project the first attempt made was an adaptive filter. Here the adaptive filter is given the inputs of the 3 EOG channels and is tasked with the estimation of the EEG channel. The adaptive filters applied to each EOG signal try to estimate the EEG signal based on the same common input in three different stages. The raw EEG scalp potentials are known to have a poor spatial resolution owing to volume conduction. Hence it is difficult to capture the signals of interest. We use a technique called Common Spatial Pattern (CSP) to analyze multichannel data based on recordings from two classes (conditions). CSP yields a data-driven supervised decomposition of the signal parameterized by a matrix. CSP filters maximize the variance of the spatially filtered signal under one condition while minimizing it for the other condition. After processing of EEG signals we now classify the signals using Support Machine Vector (SVM) algorithm. We were able to classify the signals with a classification rate of 95%.

ACKNOWLEDGEMENT

Firstly, we would thank the Department of Electrical Engineering, BITS-Pilani Hyderabad Campus for giving us this opportunity of implementing and gaining practical knowledge as a part of the Lab Oriented Project (LOP) course in our curriculum.

We would like to express our sincere thanks to Asst. Prof Dr. BVVSN Prabhakar Rao, Department Of Electrical Engineering, BITS–Pilani Hyderabad Campus for offering us a project in the area of our interest and guiding us. We thank you Sir for giving us the opportunity to carry out a project in this esteemed institution.

Table Of Contents

EEG Signal Processing

1 EEG INTRODUCTION

1.1 EEG GENERATION FROM THE BRAIN

When neurons are activated, they produce synaptic currents which then induce a magnetic field measurable by EMG and a secondary electrical field over the scalp measurable by EEG. The human head consists of different layers including the scalp, skull, brain and many other thin layers in between. The skull attenuates the signals approximately one hundred times more than the soft tissue. On the other hand, most of the noise is generated either within the brain (internal noise) or over the scalp (system noise or external noise).Therefore, only large populations of active neurons can generate enough potential to be recordable using the scalp electrodes. These measurements are then amplified greatly before further processing.

1.2 APPLICATIONS OF EEG SIGNAL PROCESSING

- Monitoring alertness, coma, and brain death;
- Locating areas of damage following head injury, stroke, and tumor;
- Testing afferent pathways (by evoked potentials);
- Monitoring cognitive engagement (alpha rhythm);
- Producing biofeedback situations;
- Controlling anesthesia depth (servo anesthesia);
- Investigating epilepsy and locating seizure origin;
- Testing epilepsy drug effects;
- Assisting in experimental cortical excision of epileptic focus;
- Monitoring the brain development;
- Testing drugs for convulsive effects;
- Investigating sleep disorders and physiology;

1.3 CHARACTERISTICS OF EEG SIGNALS

EEG signals reflect often brain rhythms that reflect the current state of the brain for example sleep or wakefulness. The main characteristics that are used to make inferences about the brain rhythms are the frequency and amplitude of the signal. These characteristics are however not independent of other factors and often change with age and also show variations from person to person.
The main brain waves distinguished by frequency are listed below.

1.3.1 Delta Waves

Delta waves lie within the range of 0.5–4 Hz. These waves are primarily associated with deep sleep and may be present in the waking state. It is very easy to confuse artefact signals caused

by the large muscles of the neck and jaw with the genuine delta response. This is because the muscles are near the surface of the skin and produce large signals, whereas the signal that is of interest originates from deep within the brain and is severely attenuated in passing through the skull. Signal processing methods need to be applied to distinguish genuine delta responses from artefacts.

1.3.2 Theta Waves

Theta waves lie within the range of 4–7.5 Hz. They are generally associated with access to subconscious material and creative inspiration or deep meditation. The theta wave plays an important role in infancy and childhood. Larger contingents of theta wave activity in the waking adult are abnormal and are caused by various pathological problems. The changes in the rhythm of theta waves have been used in maturational and emotional studies.

1.3.3 Alpha Waves

Lie in the range of 8-13 Hz and are mostly recorded over the posterior part of the brain called the occipital region of the brain. Most commonly they are rounded or sinusoidal in shape but in rare cases observed to have sharp negative peaks with rounded positive peaks. Alpha waves have been thought to indicate both a relaxed awareness without any attention or concentration. The alpha wave is the most prominent rhythm in the whole realm of brain activity and possibly covers a greater range than has been previously documented. Peaks have been observed in both the beta and the other ranges while in an alpha setting, showing alpha characteristics. The normal amplitude of the alpha wave is around 50uV. The origin and the significance of the alpha wave is an active research area.

1.3.4 Beta Waves

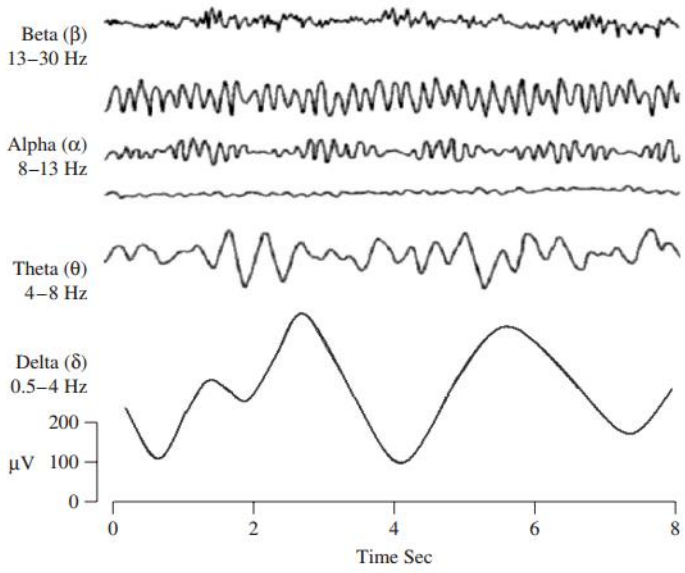
A beta wave is the electrical activity of the brain varying within the range of 14–26 Hz. A beta wave is the usual waking rhythm of the brain associated with active thinking, active attention, focus on the outside world, or solving concrete problems, and is found in normal adults. A high-level beta wave may be acquired when a human is in a panic state. Rhythmical beta activity is encountered chiefly over the frontal and central regions. Importantly, a central beta rhythm can be blocked by motor activity or tactile stimulation. The amplitude of beta rhythm is normally under 30μV.

1.3.5 Gamma Waves

The frequencies above 30 Hz correspond to the gamma range. Although the amplitudes of these rhythms are very low and their occurrence is rare, detection of these rhythms can be used for confirmation of certain brain diseases. The regions of high EEG frequencies and highest levels of cerebral blood flow (as well as oxygen and glucose uptake which are relevant to fMRI measurements) are located in the frontocentral area. The gamma wave band has also been proved to be a good indication of event-related synchronization (ERS) of the brain and can be used to demonstrate the locus for right and left index finger movement, right toes, and the rather broad and bilateral area for tongue movement.

The image shows the various types of waves described above and their frequency bands. Note that here the gamma waves are included in the beta range. This is a somewhat prevalent practice and the gamma waves are sometimes called fast beta waves.

These rhythms are cyclic in nature and correspond to the steady state responses of the brain, in addition to this transients such as an event-related potential (ERP) and containing positive occipital sharp transient (POST) signals (also called rho (ρ) waves) may be observed in EEG signals. Artefacts caused by the eye interference such as fluttering of eyelids may be similar to an alpha rhythm in the posterior part of the brain and need to be filtered out. Other extraneous signals may be caused by any bone defects or brain malfunction.



1.4 EEG RECORDING SYSTEMS

1.4.1 Requirements and Current Standards

More recent EEG systems consist of a number of delicate electrodes, a set of differential amplifiers (one for each channel) followed by filters. The conversion from analogue to digital EEG is performed by means of multichannel analogue-to-digital converters (ADCs) The EEG signals are generally restricted to within 100Hz and hence require only a sampling rate of around 200Hz to be completely captured with no aliasing. Applications exist that can even do with 100Hz of sampling rate. For very fine observation of the EEG signals, rarely 2000 Hz sampling rate is used in very high precision applications. Although a low sampling rate will suffice, we need a high accuracy in quantization and hence 16 bit or higher quantization is often employed in EEG signal recording. Given these system parameters, we can calculate that for a 128 electrode system with 500 Hz sampling, approximately 0.5 Gb of memory is required.
The EEG recording electrodes and their proper function are crucial for acquiring high quality data. Different types of electrodes are often used in the EEG recording systems, such as:

- disposable (gel-less, and pre-gelled types);
- reusable disc electrodes (gold, silver, stainless steel, or tin);
- headbands and electrode caps;
- saline-based electrodes;
- Needle electrodes.

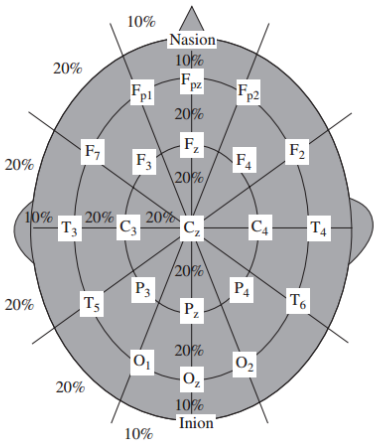
For multichannel recordings with a large number of electrodes, electrode caps are often used. Commonly used scalp electrodes consist of Ag–AgCl disks, less than 3 mm in diameter, with long flexible leads that can be plugged into an amplifier. Needle electrodes are those that have to be implanted under the skull with minimal invasive operations. High impedance between the cortex and the electrodes as well as the electrodes with high impedances can lead to distortion, which can even mask the actual EEG signals. Commercial EEG recording systems are often equipped with impedance monitors. To enable a satisfactory recording the electrode impedances should read less than 5 kOhm and be balanced to within 1 kOhm of each other. For more accurate measurement the impedances are checked after each trial.

1.4.2 Modes of Recording

Two different modes of recordings, namely differential and referential, are used. In the differential mode the two inputs to each differential amplifier are from two electrodes. In the referential mode, on the other hand, one or two reference electrodes are used. Several different reference electrode placements can be found in the literature. Physical references can be used as vertex (Cz), linked-ears, linked-mastoids, ipsilateral ear, contralateral ear, C7, bipolar references, and tip of the nose. There are also reference-free recording techniques, which actually use a common average reference. The choice of reference may produce topographic distortion if the reference is not relatively neutral. In modern instrumentation, however, the choice of a reference does not play an important role in the measurement.

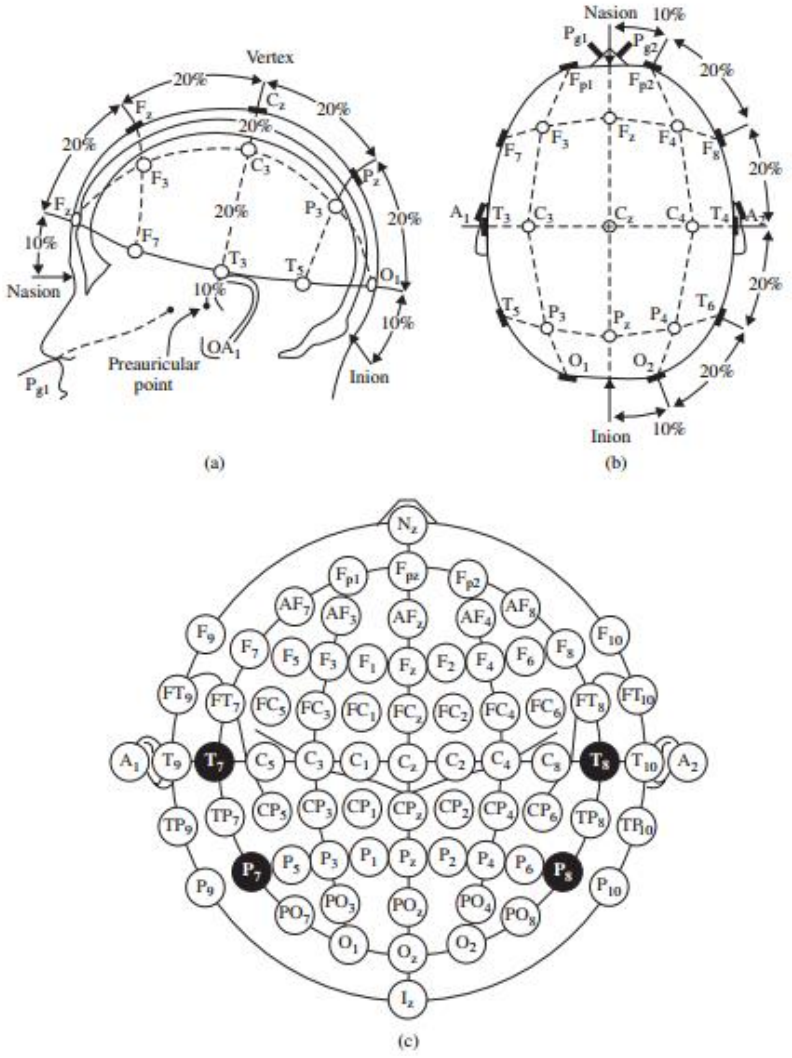
1.4.3 Electrode Placement Standards

The diagram below represents the placement of 21 electrodes in a standard recommended system called the 10-20 system.



Several other recording systems exist for example, in the Maudsley electrode positioning system, the conventional 10–20 system has been modified to capture better the signals from epileptic foci in epileptic seizure recordings. The only difference between this system and the 10–20 conventional system is that the outer electrodes are slightly lowered to enable better capturing of the required signals. The advantage of this system over the conventional one is that it provides a more extensive coverage of the lower part of the cerebral convexity which is important for epilepsy studies. Many systems have been proposed by researchers over the years mostly to cater to various specialized needs.

The diagram in the right shows a more detailed diagram of the 10-20 system for the placement of 75 electrodes around the skull:



2 DATA DESCRIPTION

2.1 EXPERIMENTAL CONDITIONS

This data set consists of EEG data from 9 subjects. The cue-based BCI paradigm consisted of four different motor imagery tasks, namely the imagination of movement of the left hand (class 1), right hand (class 2), both feet (class 3), and tongue (class 4). Two sessions on different days were recorded for each subject. Each session is comprised of 6 runs separated by short breaks. One run consists of 48 trials (12 for each of the four possible classes), yielding a total of 288 trials per session. At the beginning of each session, a recording of approximately 5 minutes was performed to estimate the EOG influence. The recording was divided into 3 blocks: (1) two minutes with eyes open (looking at a fixation cross on the screen), (2) one minute with eyes closed, and (3) one minute with eye movements. The timing scheme of one session is illustrated in Figure 1.

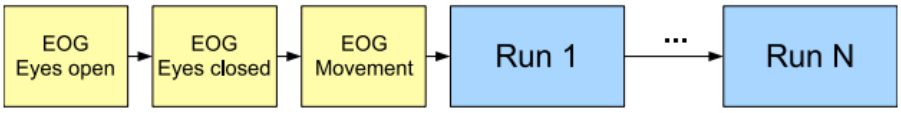
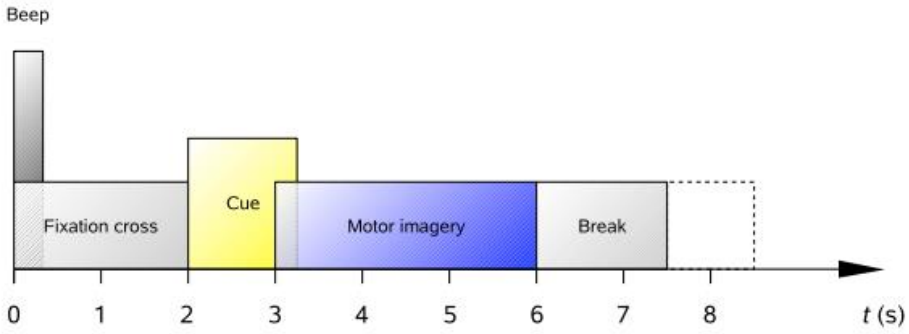


Figure 1: Timing scheme of one session.

The subjects were sitting in a comfortable armchair in front of a computer screen. At the beginning of a trial (t = 0 s), a fixation cross appeared on the black screen. In addition, a short acoustic warning tone was presented. After two seconds (t = 2 s), a cue in the form of an arrow pointing either to the left, right, down or up (corresponding to one of the four classes left hand, right hand, foot or tongue) appeared and stayed on the screen for 1.25 s. This prompted the subjects to perform the desired motor imagery task. No feedback was provided. The subjects were ask to carry out the motor imagery task until the fixation cross disappeared from the screen at t = 6 s. A short break followed where the screen was black again. The paradigm is illustrated in Figure 2



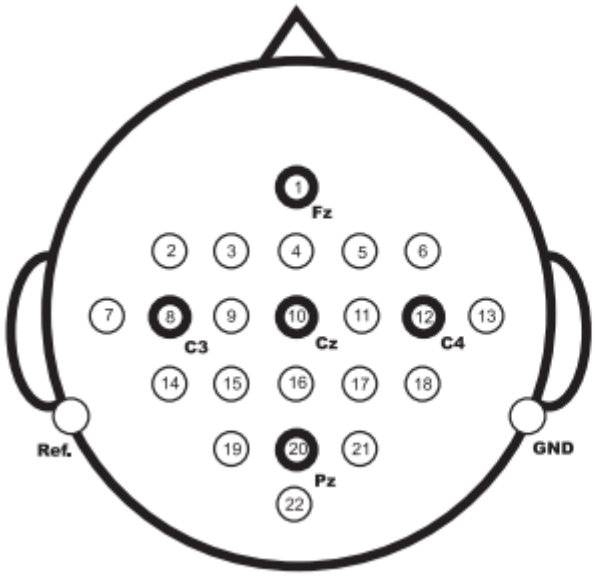
2.2 Data Collection and Properties

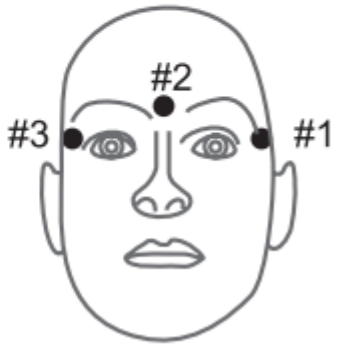
Twenty-two Ag/AgCl electrodes (with inter-electrode distances of 3.5 cm) were used to record the EEG; the montage is shown in Figure 3 left. All signals were recorded monopolarly with the left mastoid serving as reference and the right mastoid as ground. The signals were sampled with 250 Hz and bandpass-filtered between 0.5 Hz and 100 Hz. The sensitivity of the amplifier was set to 100 µV. An additional 50 Hz notch filter was enabled to suppress line noise.

In addition to the 22 EEG channels, 3 monopolar EOG channels were recorded and also sampled with 250 Hz (see Figure 3 right). They were bandpass filtered between 0.5 Hz and 100 Hz (with the 50 Hz notch filter

enabled), and the sensitivity of the amplifier was set to 1 mV. The EOG channels are provided for the subsequent application of artifact processing methods [1] and must not be used for classification. A visual inspection of all data sets was carried out by an expert and trials containing artifacts were marked.

The electrode placement scheme is shown below for reference.





3 EEG Signal Processing Methods

3.1 Pre Processing Methods

3.1.1 Artefacts Commonly Observed

The main artefacts can be divided into patient-related (physiological) and system artefacts. The patient-related or internal artefacts are body movement-related, EMG, ECG (and pulsation), EOG, ballistocardiogram, and sweating. The system artefacts are 50/60 Hz power supply interference, impedance fluctuation, cable defects, electrical noise from the electronic components, and unbalanced impedances of the electrodes.

3.1.2 Some common filters

The raw EEG signals are of the magnitude of few micro volts and have frequency content up to 300Hz in general. These signals need to be amplified, filtered for noise before further interpretation. Amplification can be done before or after the ADC. Some standard filters used are mentioned below.

1. High pass filters with a cut-off frequency of usually less than 0.5 Hz are used to remove the disturbing very low frequency components such as those of breathing.
2. Low pass filters with a cut-off frequency of approximately 50–70 Hz to remove high frequency noise.
3. Notch filters with a null frequency of 50 Hz are often necessary to ensure perfect rejection of the strong 50 Hz power supply

3.2 WAVELET TRANSFORM BASED ADAPTIVE FILTERING

The finite oscillatory nature of the wavelet makes them extremely useful in real life situations in which signals are not stationary. While Fourier transform of a signal offers only frequency resolution, wavelet transform offer "variable time-frequency" resolution which is the hallmark of wavelet transforms.

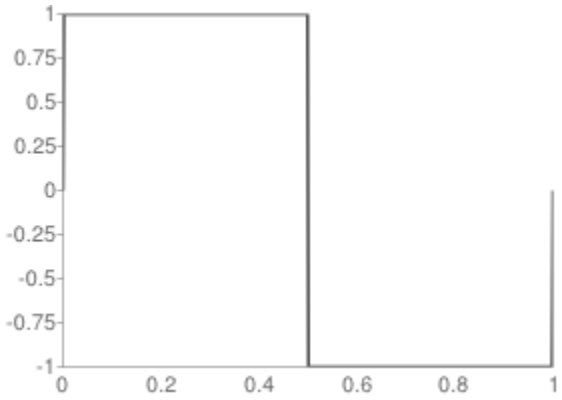
A wavelet transform decomposes the signal into basis functions which are known as wavelets. Wavelet transform is calculated separately for different segments of time-domain signal at different frequencies resulting in Multi-resolution analysis or MRA. It is designed in such a way that the product of the time resolution and frequency resolution is constant. Therefore it gives good time resolution and poor frequency resolution at high frequencies whereas good frequency and poor time resolution at low frequencies. This feature of MRA makes it excellent for signals having high frequency components for short durations and low frequency components for long duration. E.g. noise in signals, images, video frames etc.

3.2.1 What is a Wavelet?

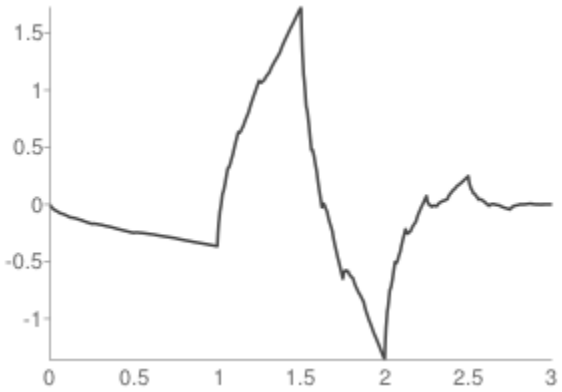
Wavelets are mathematical functions with oscillatory nature similar to sinusoidal waves with the difference being that they are of "finite oscillatory nature". For a function defined over real axis, to be classed as a wavelet, it must satisfy the following three properties:

- The integral of ψ(t) is zero $\int_{-\infty}^{\infty} \psi(t)dt = 0$
- The integral of the square of ψ(t) is unity $\int_{-\infty}^{\infty} \psi^2(t)dt = 1$
- Admissibility Condition $C_\psi = \int_0^{\infty} \frac{(|\psi(t)|)^2}{t} dt$ satisfies $0 < C_\psi < \infty$

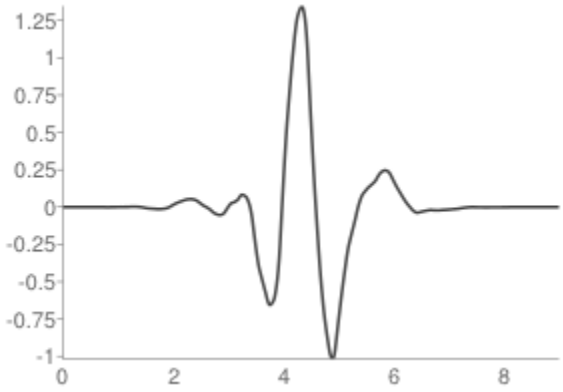
Examples of Wavelets



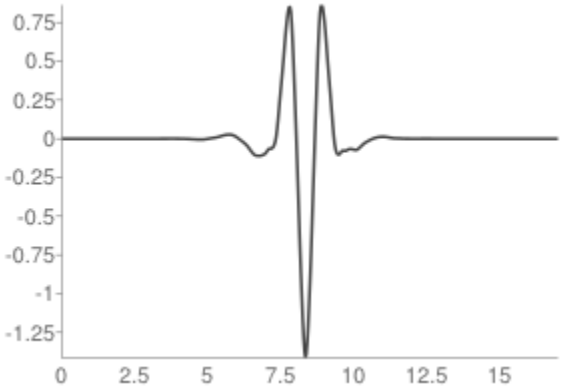
Haar Wavelet



Daubchies 2 Wavelet



Symlet 5 wavelet



Coiflet 3 Wavelet

3.2.2 Continuous Wavelet Transform

A basic wavelet transform of the signal x(t) has the following form:

$$W_x(a,b) = \frac{1}{\sqrt{a}} \int \psi\left(\frac{t-b}{a}\right) x(t)dt$$

where ψ(t) is the mother wavelet and acts as a window function to localize the integration and a, b are dilation and translation factors respectively.

The synthesis of x(t) from $W_x(a,b)$ involves a linear combination of the original wavelets Using the coefficients of $W_x(a,b)$. This synthesis or inversion is given by the formula:

$x(t)=\int_0^\infty \int_{-\infty}^\infty Wx(a, b)\psi_{a,b}(t)\frac{db\,da}{a^2}$, where $\psi a, b(t) = \frac{1}{\sqrt{a}}\psi(\frac{t-b}{a})$

3.2.3 Discrete Wavelet Transform

The signal x(t) is analyzed over infinitely many dilations and translations of the mother wavelet. Clearly there will be a lot of redundancy in the CWT. We can in fact retain the key features of the transform by only considering subsamples of the CWT. This leads us to the discrete wavelet transform (DWT). Generally, the orthogonal wavelets are employed because this method associates the wavelet to orthonormal bases of $L^2(R)$.

The DWT operates on discretely sampled function or time series x (t), usually defining time t=0, 1, 2… N-1 to be finite. It analyses the time series of discrete dilations and translations of the mother wavelet i.e. a and b take only integral values.

Let $a=a_0^j$ and $b=b_0k$ where j, k ϵ*Z*. In this case the discretized wavelet function is

$$\psi jk(t) = \frac{1}{\sqrt{a0j}}\psi\left(\frac{t-ka0jb0}{a0j}\right),$$ and its wavelet transform is given by

$$Wx(j,k) = \frac{1}{\sqrt{a0j}}\int x(t)\psi(\frac{t-kb0a0j}{a0j})$$

Given $d_j(k) = W_x(j, k)$ we hope to recover x(t) from formula like

$$x(t) = \sum_{j=0}^{\infty}\sum_{k=-\infty}^{\infty} dj(k)\Psi jk(t)$$

This formula is called the wavelet series where dj(k) is wavelet coefficient and $\Psi_{jk}(t)$ is the dual wavelet. The dual wavelet can be determined using Daubechies wavelet frame theory.

3.2.3.1 DWT Implementation using Pyramid Algorithm

The actual process involved in DWT calculation is as follows:

- The signal is decomposed into one or more levels of resolution (Octaves).
- The low pass filter produces the average signal, while the high pass filter produces detail signal.
- In Multi Resolution Analysis (MRA), the average signal at one level is sent to another set of filters, which produces the average and detail signals at the next octave.

- The low pass filter applies a scaling function to the signals, while the high pass filter applies the wavelet function.

- Applying the following difference equation with the scaling function's coefficients, h, gives an approximation of the signal. This is also known as the low pass output, where W are the scaling coefficients, and j represents the octave.

$$W(j,n) = \sum_{m=0}^{2n} W(j-1,m)h(2n-m)$$

where W(j, n) is the n^{th} scaling coefficient at the j^{th} stage.

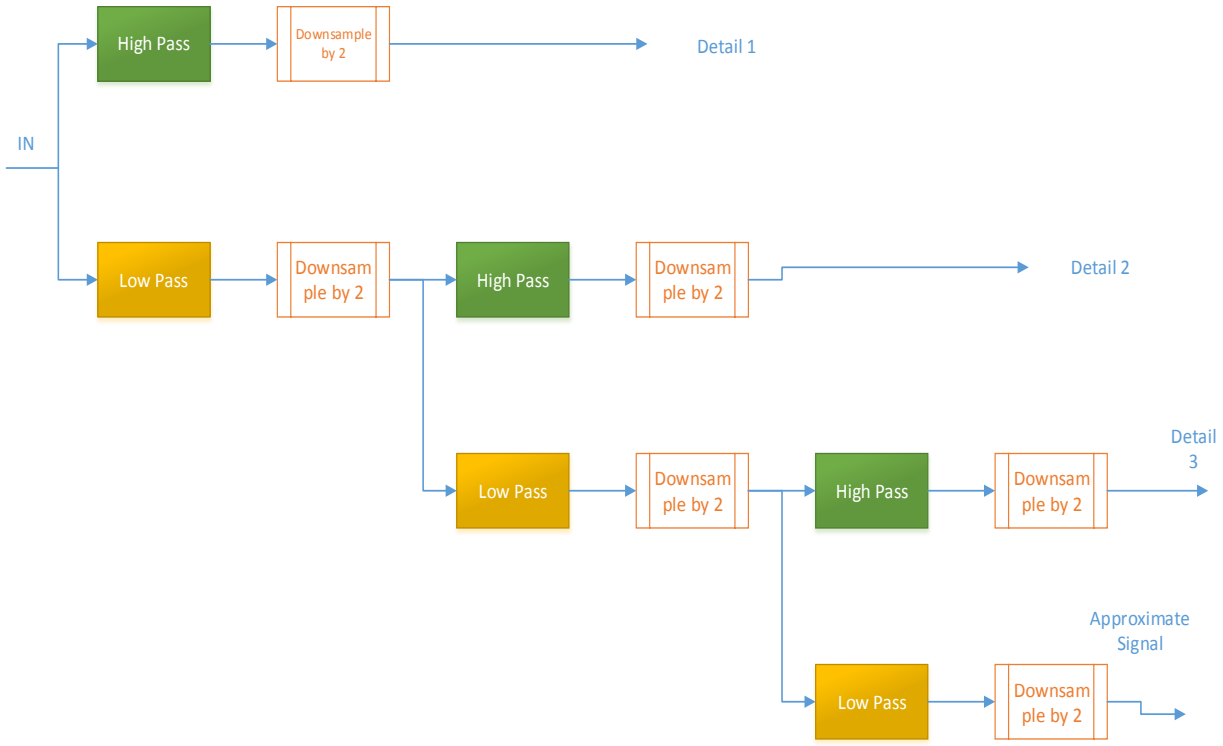
- Convolution with the wavelet function's coefficients, g, produces the detail signal, also called high pass output Wh

$$Wh(j,n) = \sum_{m=0}^{2n} W(j-1,m)g(2n-m)$$

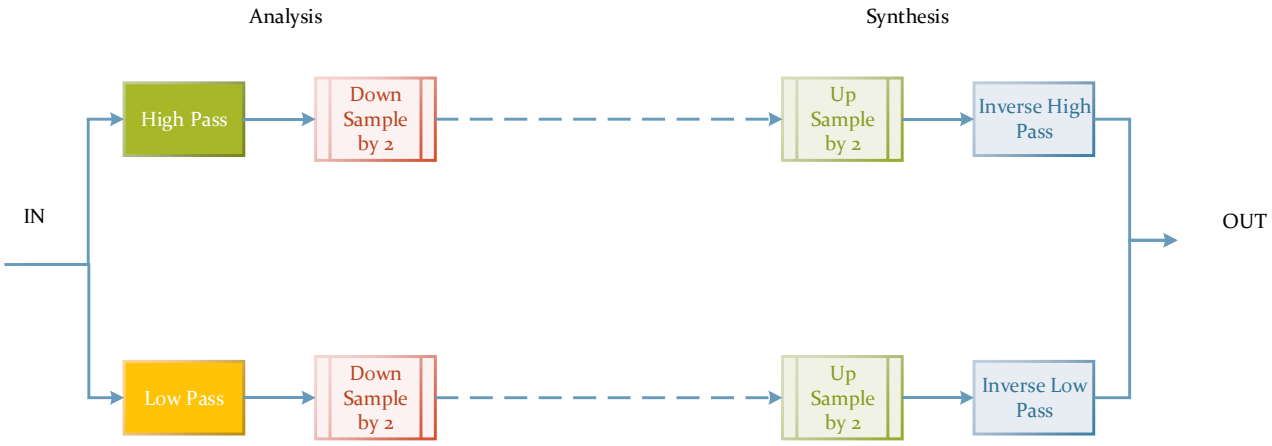
where $W_h(j, n)$ is the nth wavelet coefficient at the jth stage

- The DWT of a signal can be computed recursively using a pair of filter with the fast pyramid algorithm, by Mallat and Meyer.

Three Octave Decomposition of a Signal



1- Dimensional Signal Analysis and Synthesis Using DWT



I-Dimensional, I-Octave DWT and Inverse DWT

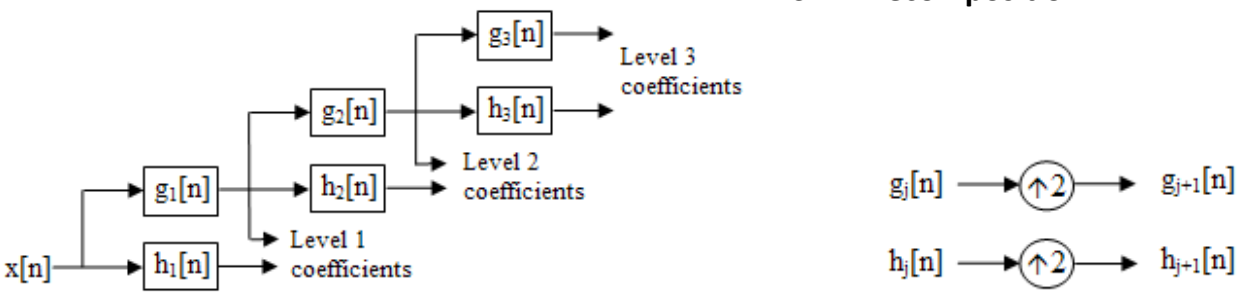
3.2.4 Stationary Wavelet Transform

The Stationary wavelet transform (SWT) is a wavelet transform algorithm designed to overcome the lack of translation-invariance of the discrete wavelet transform (DWT). Translation-invariance is achieved by removing the downsamplers and upsamplers in the DWT and upsampling the filter coefficients by a factor of 2^{j-1} in the jth level of the algorithm.

The SWT is an inherently redundant scheme as the output of each level of SWT contains the same number of samples as the input – so for a decomposition of N levels there is a redundancy of N in the wavelet coefficients.

3.2.4.1 SWT Implementation

SWT Decomposition

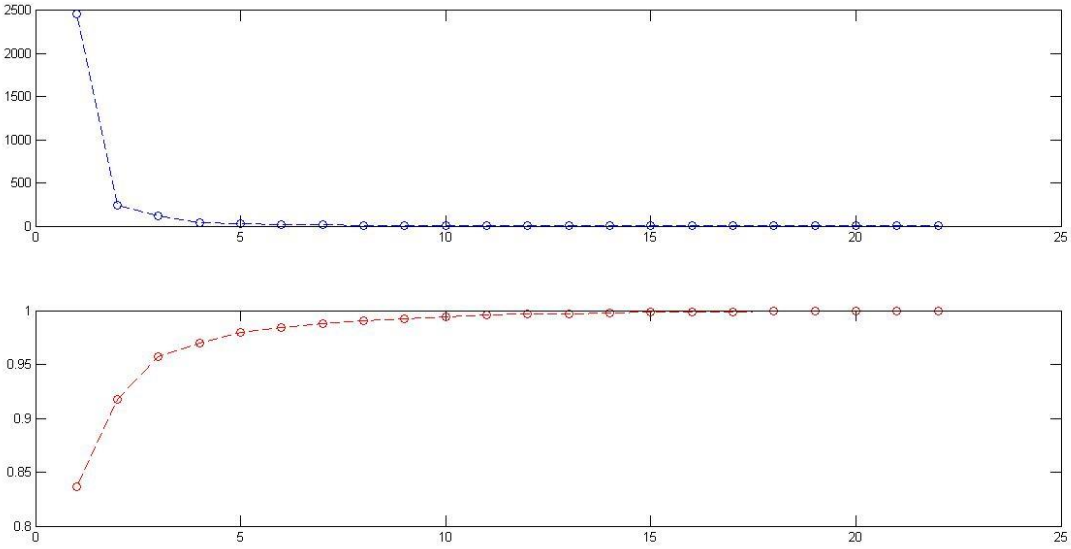


4 PCA REDUCTION OF EEG SIGNALS

The value of the cumulative variance accounted for by principle components.

1. 0.8365
2. 0.9177
3. 0.9570
4. 0.9699
5. 0.9794
6. 0.9842
7. 0.9882
8. 0.9907
9. 0.9927
10. 0.9943
11. 0.9956
12. 0.9965
13. 0.9973
14. 0.9978
15. 0.9982
16. 0.9986
17. 0.9990
18. 0.9993
19. 0.9995
20. 0.9997
21. 0.9999
22. 1.0000

The plot shows to cumulative variance accounted for by the components in the bottom plot and the variance accounted for by each variable in the top plot. We can clearly see that a lot of compression is directly possible without using any auto regressive methods or cross regression methods.



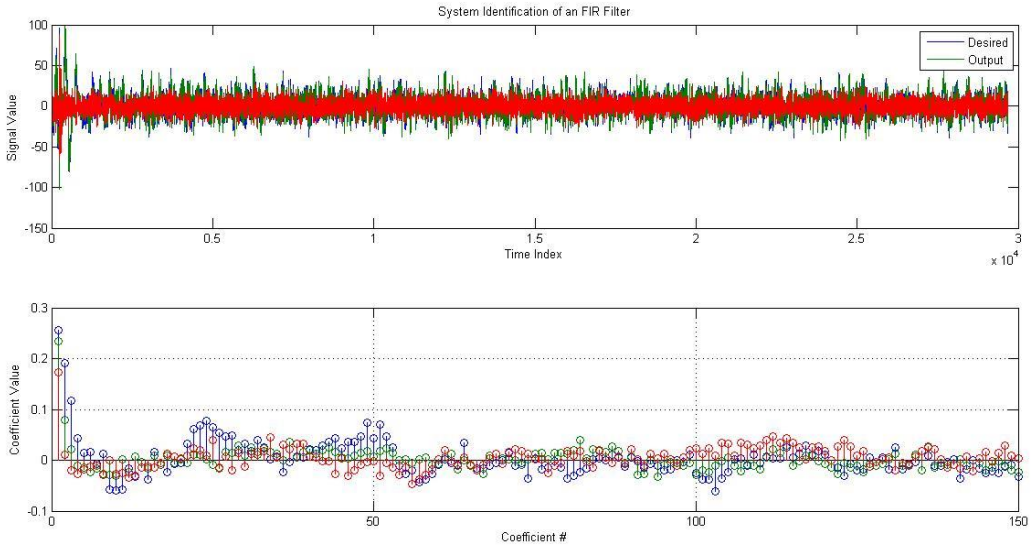
It is observed that the number of channels of independent data that are needed for complete representation are 2-3. This is a vast potential for data compression that has been exploited. Further work can be done in terms of the above indicated autoregressive and cross regressive terms that have the potential to reduce the number of variables even further vastly.

5 ARTEFACT REDUCTION USING EOG SIGNALS

Using the availability of EOG signal data, we can reduce the artefacts in the EEG data in all channels through estimation of the channel between the EOG and each channel of EEG electrodes. The estimation can be done by many different algorithms. In this project the first attempt made was an adaptive filter. Here the adaptive filter is given the inputs of the 3 EOG channels and is tasked with the estimation of the EEG channel.

5.1.1 Basic Adaptive filtering

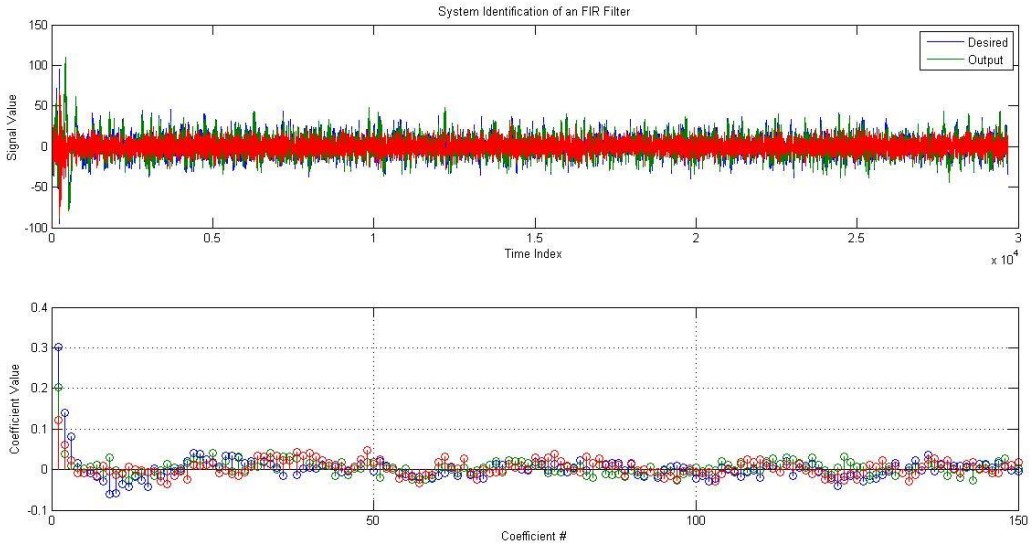
The results from the application of APRU algorithm based adaptive filter to the EEG signal is shown below.



The blue signal is the original EEG signal observed in a rest state with only EOG artifacts affecting it. The green signal is the estimated signal from the adaptive filter and the red signal is the error between the two. We observe that the estimated signal fits moderately well and the error that remains un-estimated is mainly noise.

5.1.2 Conditioning of EOG signals

Considering a vector representation of the EOG signals it is observed that they tend to be non-orthogonal to each other and often have large components in common. This essentially means that the adaptive filters applied to each EOG signal try to estimate the EEG signal based on the

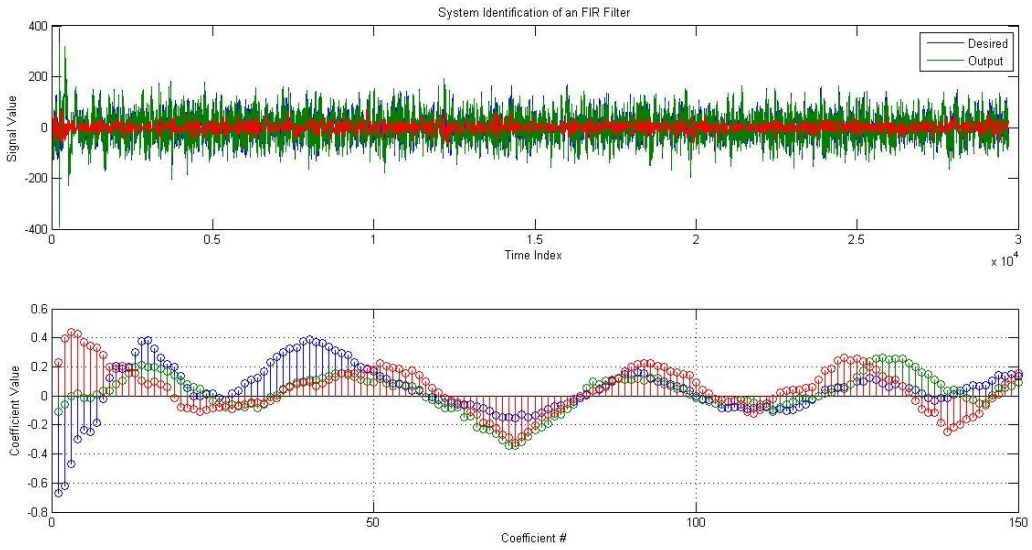


same common input in three different stages. This is generally detrimental to the learning performance of the adaptive filters. To mitigate this effect we use the PCA algorithm to ensure that the channels of EOG are orthogonal to each other. As the filter is essentially a linear algorithm and only first order independence (orthogonality) is required to ensure optimal performance, the PCA algorithm suffices.

The plot after this modification is shown above. The performance of the filter is seen to have improved and the coefficients of the filter shown in the second plot are much less noisy and well-conditioned than the first set of results.

5.1.3 Stationary Wavelet Transform as a preprocessing method

The SWT or Stationary Wavelet Transform can be applied to the data prior to adaptive filtering to increase the resolution and accuracy of the filters. As each channel of the transformed data can be approximately thought of as being approximately one band of frequencies, the adaptive filter when applied to each channel independently have a much more accurate filter response. Moreover the noise in the system that cannot be modeled are generally in different scales from the genuine artifact response and hence do not affect the coefficients in the learning process. The effects of using this method are dramatic as shown below.

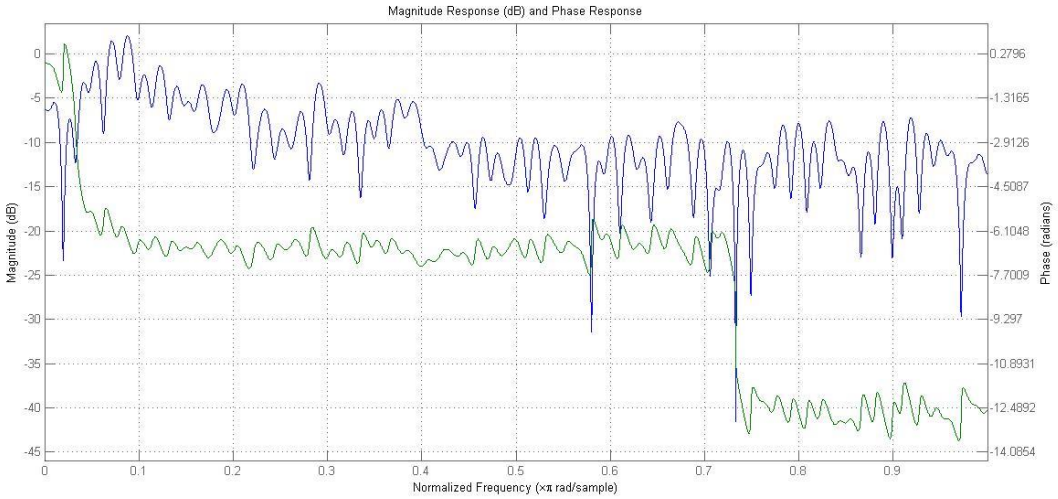


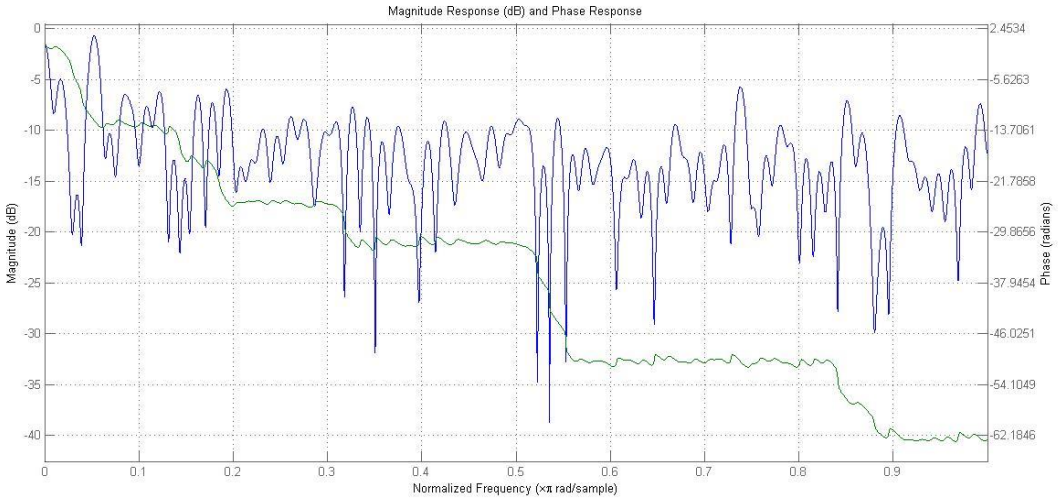
We can see that the un-modeled error is negligible and the filter coefficients are excellently conditioned. Thus the EOG based artifact correction algorithm is chosen as APRU based adaptive filter with SWT based preprocessing.

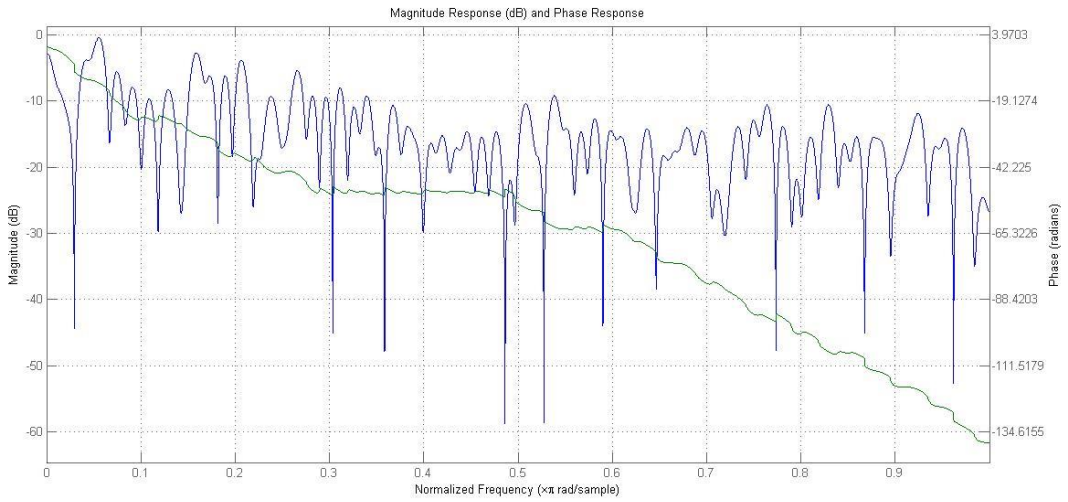
5.2 INTERPRETATION OF THE FILTER COEFFICIENTS

The filter coefficients can be thought of as modelling the channel characteristics between each EOG electrode and the EEG electrode corresponding to that filter. Observing the characteristics of the coefficients and verifying them with the physically observed characteristics of the corresponding areas in the head can be a good measure of the accuracy of the filter. The plots below show the characteristics of the filters developed for each channel of EOG.

The first filter shows that magnitude response remains almost constant and has no significant effect while the phase plot shows that the EEG interference is simply a delayed version of the EOG signal with different delays for different frequency ranges. We also observe large flat zones where the phase response remains constant. Similar observations are made about channel 2. Channel 3 is observed to have an approximately linear phase while showing no significant trend in the magnitude response and we can infer that the third principle component models mainly noise that has no correspondence with the interference to EEG signals.





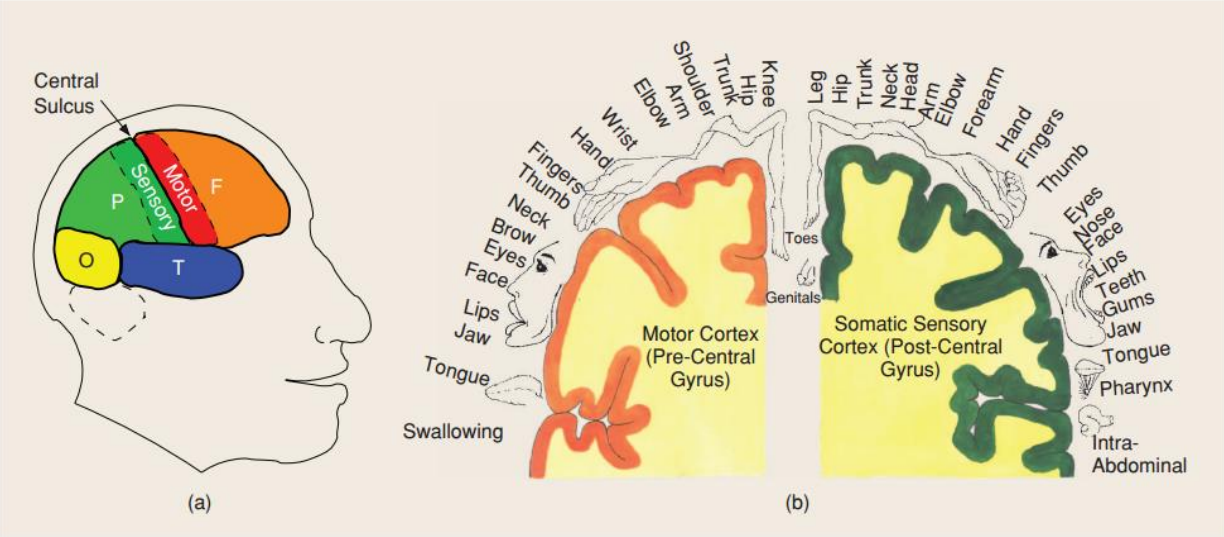


This interpretation can be verified from literature where the EEG interference has been observed to be modeled well by the shifted versions of the EOG signals. Thus we can conclude that the adaptive filter is a good approach to artifact cancellation with EOG data given.

6 SPATIAL FILTERING

6.1 NEUROPHYSIOLOGICAL BACKGROUND

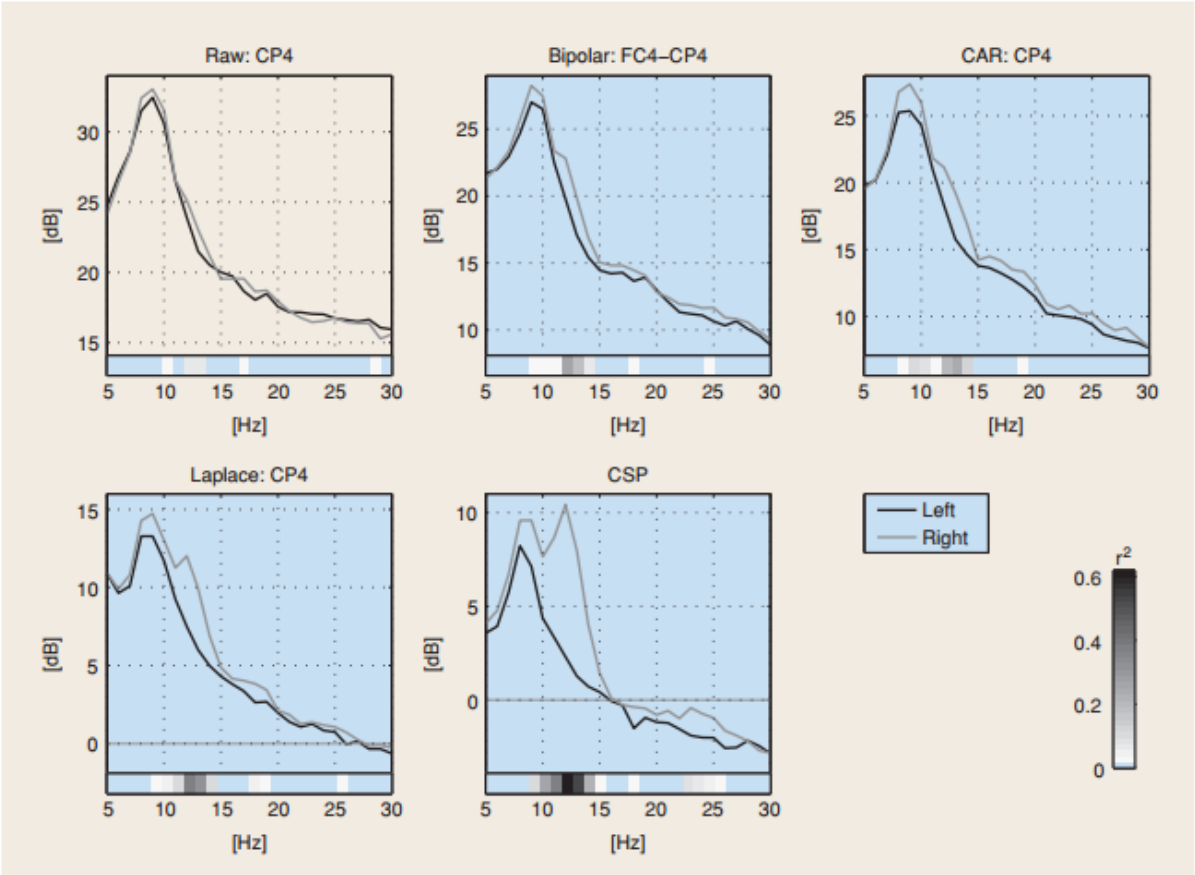
Macroscopic brain activity during resting wakefulness comprises distinct “idle” rhythms located over various cortical areas, e.g. the occipital α-rhythm (8–12 Hz) can be measured over the visual cortex. The perirolandic sensorimotor cortices show rhythmic macroscopic EEG oscillations (μ-rhythm, sensori motor rhythm, SMR) [20], [24] with spectral peak energies of about 9–14 Hz (localized predominantly over the post-central somatosensory cortex) and around 20 Hz (over the pre-central motor cortex). The occipital α-rhythm is quite prominent and can be seen in the raw EEG with the naked eye if the subject closes the eyes (idling of the visual cortex). In contrast the μ-rhythm has a much weaker amplitude and can only be observed after appropriate signal processing. In some subjects no μ-rhythm can be observed in scalp EEG. Our system is based on the modulation of the SMR. In fact, motor activity, both actual and imagined] as well as somatosensory stimulation have been reported to modulate the μ-rhythm. Processing of motor commands or somatosensory stimuli causes an attenuation of the rhythmic activity termed event-related de-synchronization (ERD), while an increase in the rhythmic activity is termed event related synchronization (ERS). For BCIs, the important fact is that the ERD is caused also by imagined movements (healthy users, see Figure 2) and by intended movements in paralyzed patients [2].



[FIG3] (a) Lobes of the brain: frontal, parietal, occipital, and temporal (named after the bones of the skull beneath which they are located). The central sulcus separates the frontal and parietal lobe. (b) Geometric mapping between body parts and motor/somatosensory cortex. The motor cortex and the somatosensory cortex are shown at the left and right part of the figure, respectively. Note that in each hemisphere there is one motor area (frontal to the central sulcus) and one sensori area (posterior to the central sulcus). The part which is not shown can be obtained by mirroring the figure folded at the center.

6.2 NEED FOR SPATIAL FILTERING

Raw EEG scalp potentials are known to have a poor spatial resolution owing to volume conduction. In a simulation study only half the contribution to each scalp electrode came from sources within a 3 cm radius. This is in particular a problem if the signal of interest is weak, e.g. sensorimotor rhythms, while other sources produce strong signals in the same frequency range like the α-rhythm of the visual cortex or movement and muscle artifacts. The demands are carried to the extremes when it comes to single-trial analysis as in BCI. While some approaches try to achieve the required signal strength by training the subjects an alternative is to calibrate the system to the specific characteristics of each user. For the latter, data-driven approaches to calculate subject-specific spatial filters have proven to be useful. As a demonstration of the importance of spatial filters, Figure 4 shows spectra of left versus right hand motor imagery at the right hemispherical sensorimotor cortex. All plots are computed from the same data but using different spatial filters. While the raw channel only shows a peak around 9 Hz that provides almost no discrimination between the two conditions, the bipolar and the common average reference filter can improve the discrimination slightly. However, the Laplace filter and even more the CSP filter reveal a second spectral peak around 12 Hz with strong discriminative power. By further investigations, the spatial origin of the non-discriminative peak could be traced back to the visual cortex, while the discriminative peak originates from sensorimotor rhythms. Note that in many subjects, the frequency ranges of visual and sensorimotor rhythms overlap or completely coincide.



[FIG4] **Spectra of left versus right hand motor imagery. All plots are calculated from the same dataset but using different spatial filters. The discrimination between the two conditions is quantified by the r^2-value. CAR stands for common average reference.**

So we can observe that spatial filtering can bring out strong discriminative characteristics that remain hidden in many users and can drastically improve classification accuracy. We have adopted a CSP based approach to spatial filtering as opposed to the other filters mentioned in this section.

6.3 INTRODUCTION TO CSP ANALYSIS

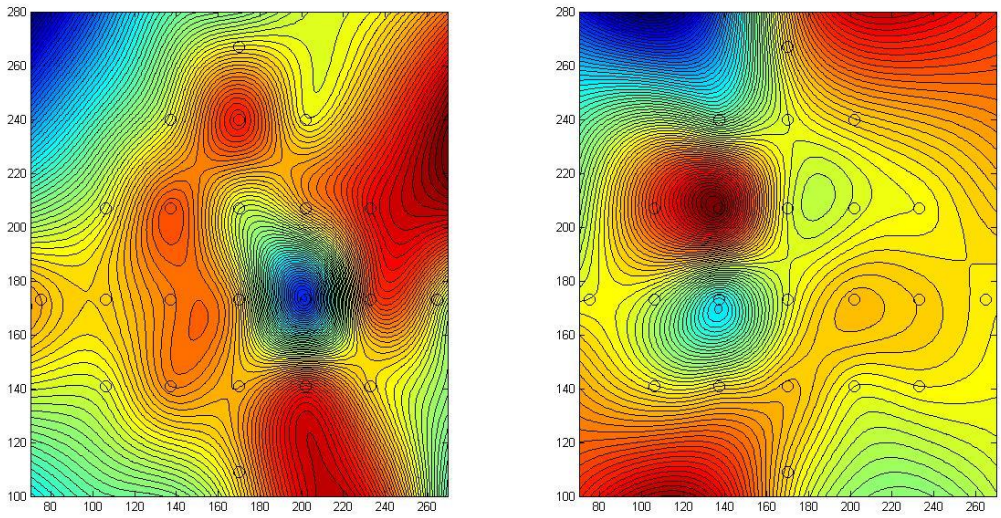
CSP is a technique to analyze multichannel data based on recordings from two classes (conditions). CSP yields a data-driven supervised decomposition of the signal parameterized by a matrix
$W \in R^{C\,X\,C}$ (C being the number of channels) that projects the signal $X \in R^{C\,X\,T}$(T is the number of samples in single trial) in the original sensor space to $X \in R^{C\,X\,T}$, which lives in the surrogate sensor space, as follows:

$$X_{CSP}(t) = W^T X(t)$$

We call each column vector of W a spatial filter or simply a filter.
In a nutshell, CSP filters maximize the variance of the spatially filtered signal under one condition while minimizing it for the other condition. Since variance of band-pass filtered signals is equal to band-power, CSP analysis is applied to approximately band-pass filtered signals in order to obtain an effective discrimination of mental states that are characterized by ERD/ERS effects.

6.4 CSP RESULTS

Some of the results obtained from CSP can be interpreted to show intuitively the working of CSP and conclusively prove its effectiveness in the selection of optimal spatial coefficients for classification purposes.
The first instance considered is the classification of left hand and right hand movement. The first and second class data was fed to the CSP algorithm and an optimal spatial filter for classification was developed. The obtained filters are shown below.



The above plot shows the coefficients of the filter across space as a gradient field. This representation allows us to visualize the effect of the filter on the selection of signals from the scalp. We can clearly see that the filter has shown a strong selectivity to the side of the brain meaning that it has identified the best possible differentiator between the classes i.e. the hemisphere of the brain in which activity occurs. This is just an example of the effectiveness of CSP for this problem.

CSP runs on binary input data but our classification problem has four classes, so we can adopt various strategies to convert the problem to a form amenable to CSP. These are:

1. **One versus rest problem**

Here each class is labelled as 1 first and the rest of the classes are labelled as 0 and a classification is performed. So we need as many classifiers as there are classes in the data

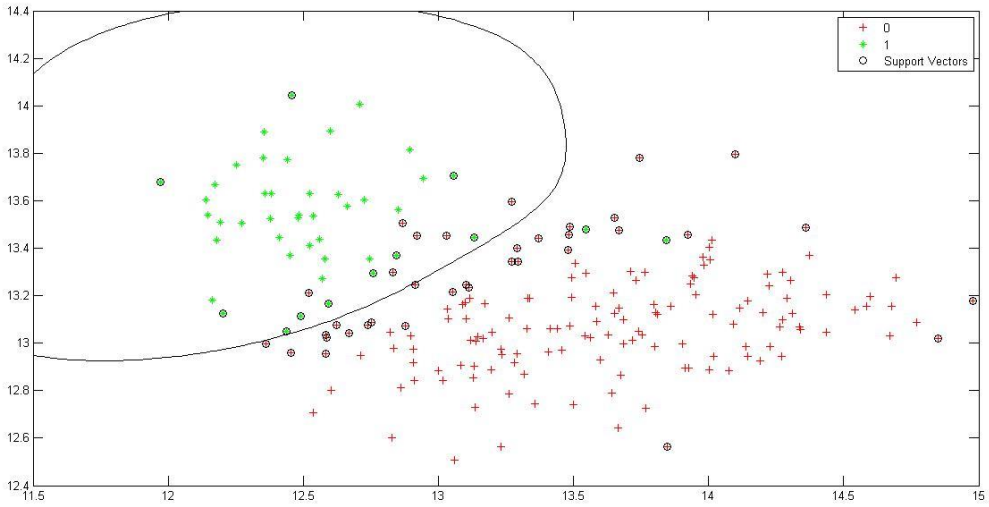
2. Pair Wise Classification

Each pair of classes are considered together and classified. After all the classifiers are developed, their outputs are considered for a polling and the net output class is chosen.

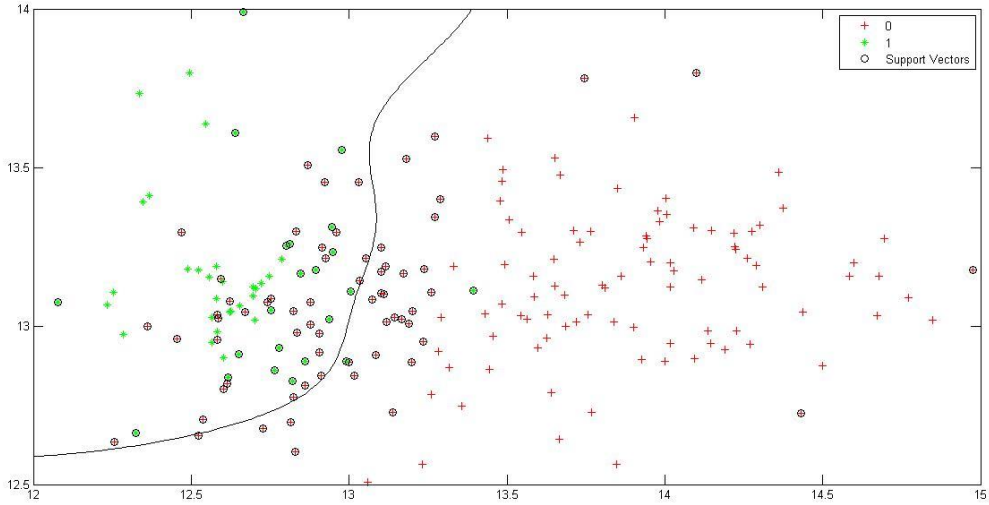
3. Divide and Conquer

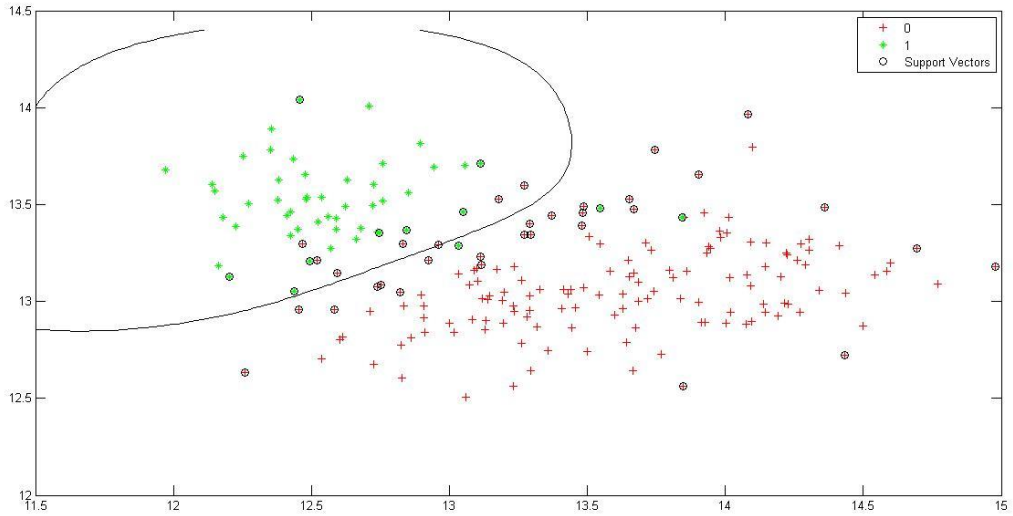
This is similar to the one versus all approach but a tree based system is adopted for the classification meaning that the number of classifiers needed is the total number of classes-1

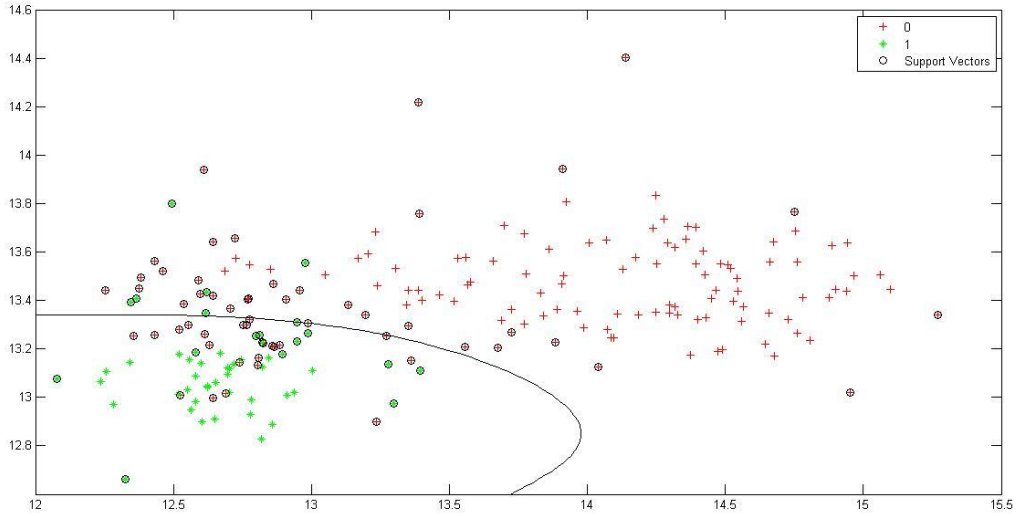
The plots below summarizes the value of the average power in each of the first and last eigenvalue spatial filter for the one versus all problems considered. This gives an approximate



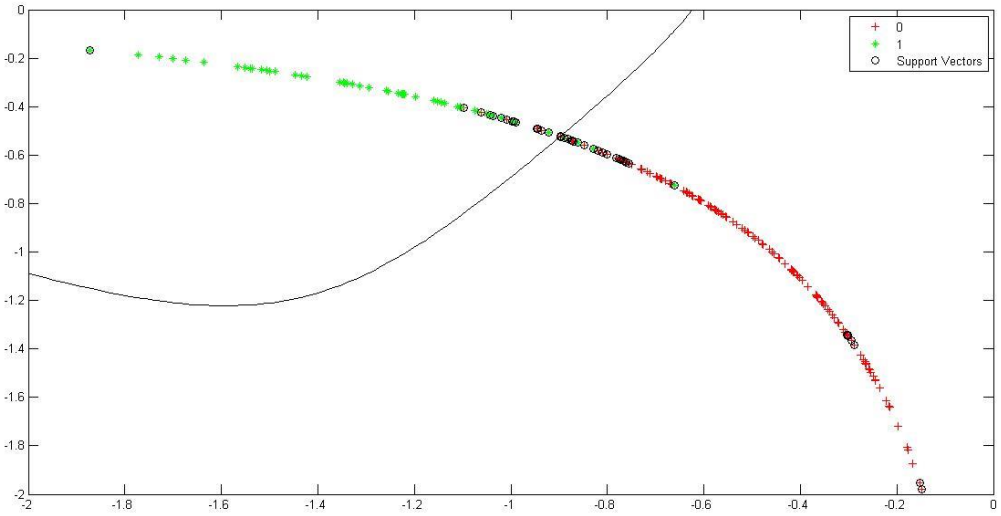
idea of how good the discrimination is after the application of the spatial filter.




0
1
Support Vectors
14.5
14
13.5
13
12.5
12
11.5
12
12.5
13
13.5
14
14.5
15


0
1
Support Vectors
14.6
14.4
14.2
14
13.8
13.6
13.4
13.2
13
12.8
12
12.5
13
13.5
14
14.5
15
15.5

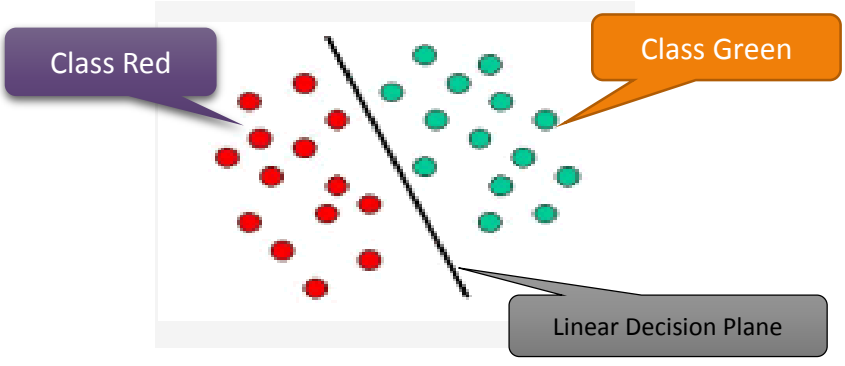
An even better picture can be obtained by plotting after normalization which aligns the point on a circular trajectory.



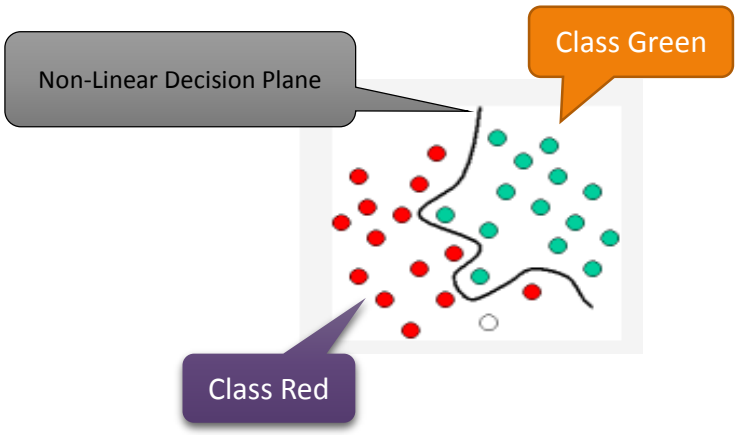
7 CLASSIFICATION USING SUPPORT VECTOR MACHINE (SVM)

7.1 SUPPORT VECTOR MACHINE

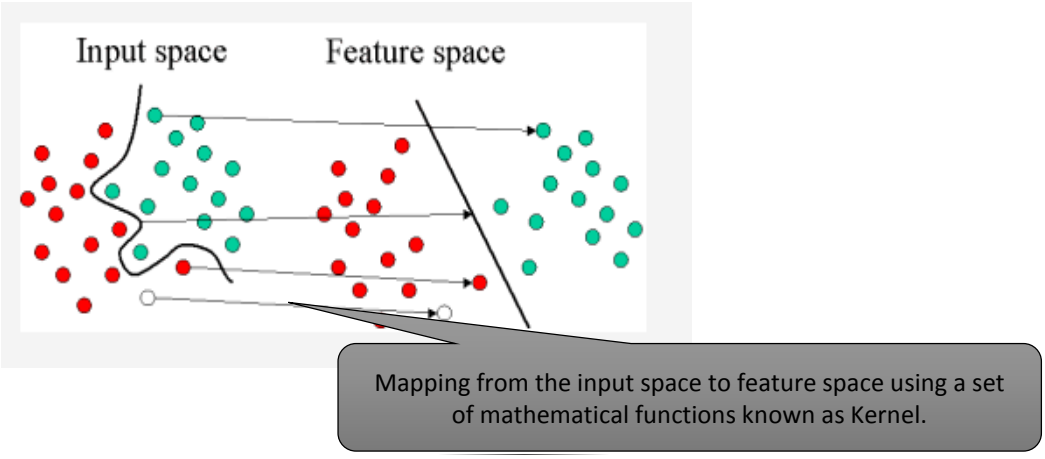
In machine learning, support vector machines are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.



More formally, a support vector machine constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks. There are many hyperplanes that might classify the data. One reasonable choice as the best hyperplane is the one that represents the largest separation, or margin, between the two classes. So we choose the hyperplane so that the distance from it to the nearest data point on each side is maximized. If such a hyperplane exists, it is known as the maximum-margin hyperplane and the linear classifier it defines is known as a maximum margin classifier.



Whereas the original problem may be stated in a finite dimensional space, it often happens that the sets to discriminate are not linearly separable in that space. For this reason, it was proposed that the original finite-dimensional space be mapped into a much higher-dimensional space, presumably making the separation easier in that space. To keep the computational load reasonable, the mappings used by SVM schemes are designed to ensure that dot products may be computed easily in terms of the variables in the original space, by defining them in terms of a kernel function K(x,y) selected to suit the problem.



7.1.1 Mathematical Explanation

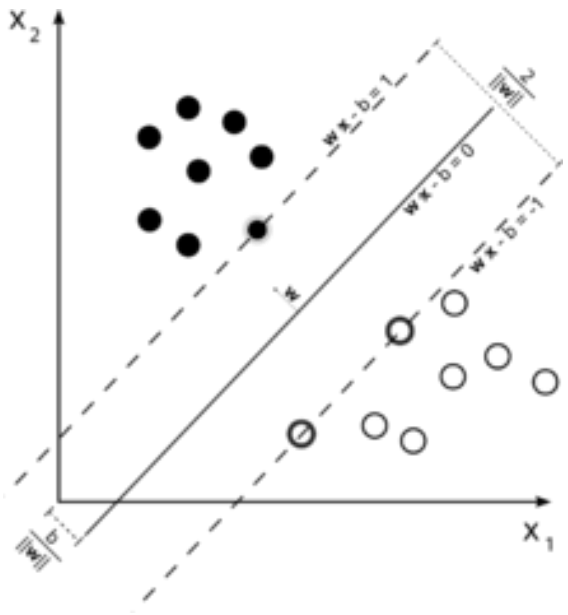
Given some training data $\mathcal{D}$, a set of *n* points of the form:

$$\mathcal{D} = \{(\mathbf{x}_i, y_i) \mid \mathbf{x}_i \in \mathbb{R}^p,\ y_i \in \{-1, 1\}\}_{i=1}^{n}$$

where the y_i is either 1 or −1, indicating the class to which the point $\mathbf{x}_i$ belongs. Each $\mathbf{x}_i$ is a *p*-dimensional real vector. We want to find the maximum-margin hyperplane that divides the points having $y_i = 1$ from those having $y_i = -1$. Any hyperplane can be written as the set of points $\mathbf{x}$ satisfying

$$\mathbf{w} \cdot \mathbf{x} - b = 0,$$

where dot(.) denotes the dot product and $\mathbf{w}$ the (not necessarily normalized) normal vector to the hyperplane.



Maximum-margin hyperplane and margins for an SVM trained with samples from two classes. Samples on the margin are called the support vectors.

The parameter $\frac{b}{\|\mathbf{w}\|}$ determines the offset of the hyperplane from the origin along the normal vector $\mathbf{w}$.

If the training data are linearly separable, we can select two hyperplanes in a way that they separate the data and there are no points between them, and then try to maximize their distance. The region bounded by them is called "the margin". These hyperplanes can be described by the equations

$$\mathbf{w} \cdot \mathbf{x} - b = 1$$

and

$$\mathbf{w} \cdot \mathbf{x} - b = -1.$$

By using geometry, we find the distance between these two hyperplanes is $\frac{2}{\|\mathbf{w}\|}$, so we want to minimize $\|\mathbf{w}\|$. As we also have to prevent data points from falling into the margin, we add the following constraint: for each $i$either

$$\mathbf{w} \cdot \mathbf{x}_i - b \geq 1 \qquad \text{for } \mathbf{x}_i\text{of the first class}$$

or

$$\mathbf{w} \cdot \mathbf{x}_i - b \leq -1 \qquad \text{for } \mathbf{x}_i\text{of the second.}$$

This can be rewritten as:

$$y_i(\mathbf{w} \cdot \mathbf{x}_i - b) \geq 1, \quad \text{for all } 1 \leq i \leq n. \qquad (1)$$

We can put this together to get the optimization problem:

Minimize (in $\mathbf{w}, b$) $\|\mathbf{w}\|$ subject to (for any $i = 1, \ldots, n$)

$$y_i(\mathbf{w} \cdot \mathbf{x_i} - b) \geq 1.$$

8 MATLAB CODE

Included in this section are all the MATLAB codes implemented as part of this project.

8.1 CODE FOR ORGANIZING DATA INTO A READABLE FORMAT

```
clc; clear all;
[s,h]=sload('D:\Acad\Sem 6\EEG Signal processing\Data\EEG +Eog\A01T.gdf');
pos=h.EVENT.POS;
dur=h.EVENT.DUR;
typ=(h.EVENT.TYP==769)+2.*(h.EVENT.TYP==770)+3.*(h.EVENT.TYP==771)...
+4.*(h.EVENT.TYP==772)+100.*(h.EVENT.TYP==768)+101.*(h.EVENT.TYP==1023)...
+1000.*(h.EVENT.TYP==32766)+150.*(h.EVENT.TYP==276)+151.*(h.EVENT.TYP==277)+1
52.*(h.EVENT.TYP==1072);
info=[typ pos pos+dur];
%% EEG with only EOG effects
d=s(50056:91518,1:22);
x1=s(50056:91518,23);
x2=s(50056:91518,24);
x3=s(50056:91518,25);
count=0;
d=d';
temp1=isnan(d);
    if(sum(max(temp1))~=0)
    count=count+1
    [row1,col1]=find(temp1);
    row1=unique(row1);
        for z=1:size(row1,1)
        tdata1=iddata(d(row1(z),:)',[],250);
        tintp1=misdata(tdata1);
        d(row1(z),:)=(tintp1.OutputData)';
        end
    end

%% Event Data
j=1;class=[];
for i=1:size(info,1)
if (info(i,1)==100)
if (info(i+1,1)~=101)
class(j,1)=info(i+1,1);
data{j}=s(info(i,2):info(i,3),:)';
j=j+1;
end
end
end
dat=cell(4,25);
for i=1:size(class,1)
for j=1:25
dat{class(i,1),j}=[dat{class(i,1),j}; data{1,i}(j,:)];
end
end
```

```
% end
%%
j=1;class_c=[];
for i=1:size(info,1)
if (info(i,1)==100)
if (info(i+1,1)==101)
class_c(j,1)=info(i+2,1);
data_c{j}=s(info(i+1,2):info(i+1,3),:)';
j=j+1;
end
end
end
%% Interpolate Incomplete Data
count=0;
for i=1:4
for j=1:25
temp=isnan(dat{i,j});
    if(sum(max(temp))~=0)
    count=count+1
    [row,col]=find(temp);
    row=unique(row);
        for z=1:size(row,1)
        tdata=iddata((dat{i,j}(row(z),:))',[],250);
        tintp=misdata(tdata);
        dat{i,j}(row(z),:)=(tintp.OutputData)';
        end
    end
end
end
%%
count=1;
data_new=cell(273,1);
%y1=sort(class);y=class;
for i=1:4
for k=1:size(dat{i,1},1)
for j=1:25
    data_new{count}=[data_new{count};dat{i,j}(k,:)];
end
count=count+1;
end
end
%save('C:\Users\Shriram\Desktop\Data_Raw.mat','dat','y1');
save('C:\Users\Shriram\Desktop\Data_Raw_Unord.mat','data_new','y');
%%
datcpress=cell(1,22);compress=[];
for i=1:25
p=[dat{1,i};dat{2,i};dat{3,i};dat{4,i}];
[pc,score,latent,tsquare] = princomp(p);
pcomre=pc(:,1:10);
datcpress{i}=p*pcomre;
compress=[compress datcpress{i}];
end
```

8.2 ARTIFACT REMOVAL WITH ADAPTIVE FILTERING

```
clear all;
load('C:\Users\Shriram\Desktop\eeg_eog.mat');
[COEFF,SCORE,latent] = princomp(eeg');
eeg_c=(eeg'*COEFF)';
d  = eeg_c(1,:);
d=padarray(d,[0 13],'symmetric','post');
ds=swt(d,4,'sym3');

%% APRU Algorithm with prefiltered eog & eeg

order=200;
[COEFF,SCORE,latent] = princomp(eog');
eogf=eog'*COEFF;
eogf=padarray(eogf,[13 0],'symmetric','post');
mu = 0.08;                          % Step size
po = 2;                            % Projection order
offset = 0.05; % Offset
dfilt=ds;

for i=5:5
ha1(i) = adaptfilt.apru(order,mu,po,offset);
[y1,e1] = filter(ha1(i),eogf(:,1)',ds(i,:));

ha2(i) = adaptfilt.apru(order,mu,po,offset);
[y2,e2] = filter(ha2(i),eogf(:,2)',e1);

ha3(i) = adaptfilt.apru(order,mu,po,offset);
[y3,e3] = filter(ha3(i),eogf(:,3)',e2);

dfilt(i,:)=y1+y2+y3;
end
%%

y=iswt(dfilt,'sym3');
figure,subplot(2,1,1); plot(1:size(d,2),[d;y;d-y]);
title('System Identification of an FIR Filter');
legend('Desired','Output');
xlabel('Time Index'); ylabel('Signal Value');
subplot(2,1,2); stem(1:size(ha1(1,5).coefficients,2),[ha1(1,5).coefficients'
ha2(1,5).coefficients' ha3(1,5).coefficients']);
xlabel('Coefficient #'); ylabel('Coefficient Value');  grid on;
```

8.3 COMMON SPATIAL PATTERNS PROGRAM

```
function [unmixing] = csp_custom(dat1, dat2)

% CSP Common spatial pattern decomposition
% This implements Ramoser, H., Gerking, M., and Pfurtscheller, G. "Optimal
% spatial filtering of single trial EEG during imagined hand movement."
```

```
% IEEE Trans. Rehab. Eng 8 (2000), 446, 441.

R1=zeros(size(dat1{1},1));
R2=zeros(size(dat1{1},1));

for i=1:size(dat1,1)
R1_temp = dat1{i}*dat1{i}';
R1_temp = R1_temp/trace(R1_temp);
R1=R1+R1_temp;
end
R1=R1./size(dat1,1);

for i=1:size(dat2,1)
R2_temp = dat2{i}*dat2{i}';
R2_temp = R2_temp/trace(R2_temp);
R2=R2+R2_temp;
end
R2=R2./size(dat2,1);

% R2 = dat2*dat2';
% R2 = R2/trace(R2);

% Ramoser equation (2)
Rsum = R1+R2;

% Find Eigenvalues and Eigenvectors of RC
% Sort eigenvalues in descending order
[EVecsum,EValsum] = eig(Rsum);
[EValsum,ind] = sort(diag(EValsum),'descend');
EVecsum = EVecsum(:,ind);

% Find Whitening Transformation Matrix - Ramoser Equation (3)
W = sqrt(pinv(diag(EValsum))) * EVecsum';

% Whiten Data Using Whiting Transform - Ramoser Equation (4)
S1 = W * R1 * W';
S2 = W * R2 * W';

% Ramoser equation (5)
% [U1,Psi1] = eig(S1);
% [U2,Psi2] = eig(S2);

% Generalized eigenvectors/values
[B,D] = eig(S1,S2);

% Simultanous diagonalization
% Should be equivalent to [B,D]=eig(S1);

% Verify algorithim
% Disp('test1:Psi1+Psi2=I')
% Psi1+Psi2

% Sort ascending by default
```

```
% [Psi1,ind] = sort(diag(Psi1)); U1 = U1(:,ind);
% [Psi2,ind] = sort(diag(Psi2)); U2 = U2(:,ind);

[D,ind]=sort(diag(D));
B=B(:,ind);

%Resulting Projection Matrix-these are the spatial filter coefficients
unmixing = B'*W;
```

8.4 SUPPORT VECTOR MACHINES FOR CLASSIFICATION

```
%% Support Vector Machines

opts = statset('MaxIter',30000);
% Train the classifier
svmStruct =
svmtrain(Xtrain,Ytrain,'showplot',true,'kernel_function','rbf','rbf_sigma',1,
'kktviolationlevel',0,'options',opts);

% Make a prediction for the test set
Y_svm = svmclassify(svmStruct,Xtest,'showplot',true);
C_svm = confusionmat(Ytest,Y_svm);
% Examine the confusion matrix for each class as a percentage of the true
class
C_svm = bsxfun(@rdivide,C_svm,sum(C_svm,2)) * 100
figure;
plotconfusion(Ytest',Y_svm')
%error rate
100*(1-sum(abs(Y_svm-Ytest))./109)
```

8.5 COMPLETE PROGRAM EXECUTING ALL COMPONENTS TO MAKE A COMPLETE PREDICTION

```
clc; clear all;
load('tldata1.mat');
[b,a]=butter(5,[8/125,30/125],'bandpass');
%data3=[data1;data2];
y=zeros(size(data3,1),1);
%% Filtering Data
for i=1:size(data3,1)
for j=1:44
%data3{i}=data3{i}(:,625:1500);
data3{i}(j,:)=filter(b,a,data3{i}(j,:));
end
end
%%
% for i=1:size(data1,1)
% for j=1:size(data1{i},2)
%    data1{i}(:,j)=(data1{i}(:,j)'-mean(data1{i}'))';
```

```
% end
% end
%
% for i=1:size(data2,1)
% for j=1:size(data2{i},2)
%    data2{i}(:,j)=(data2{i}(:,j)'-mean(data2{i}'))';
% end
% end


%%
clas=1;
if clas==1
data1=data3(1:69); data2=[data3(70:end)];%data3(207:end)];
k1=1;k2=3;
end
if clas==2
data1=data3(70:138); data2=[data3(1:69);data3(139:end)];
k1=1;k2=2;
end
if clas==3
data1=data3(139:206); data2=[data3(1:138);data3(207:end)];
k1=2;k2=3;
end
if clas==4
data1=data3(1:206); data2=[data3(207:end)];
k1=1;k2=3;
end
y(1:size(data1,1))=1;
%%
W=csp_custom(data1,data2);
for i=1:size(data1,1)
 Z=W*data1{i};
 datacsp_1{i}=[Z(1:1,:);Z(22:22,:)];
end
for i=1:size(data2,1)
 Z=W*data2{i};
 datacsp_2{i}=[Z(1:1,:);Z(22:22,:)];
end
datacsp=[datacsp_1';datacsp_2'];
dat_mat_1=[];dat_mat_2=[];
for i=1:(size(data1,1)+size(data2,1))

dat_mat_1=[dat_mat_1;datacsp{i,1}(1,:)];dat_mat_2=[dat_mat_2;datacsp{i,1}(2,:
)];

dat_classify(i,:)=log(diag(datacsp{i,1}*datacsp{i,1}'));%./trace(datacsp{i,1}
*datacsp{i,1}'));
end

%% Prepare the Data: Response and Predictors
% Response
Y = y;

tabulate(Y)
```

```
% Predictor matrix
X = double(dat_classify(:,1:end));


% we will hold 20% of the data, selected randomly, for
% test phase.
cv = cvpartition(length(dat_classify),'holdout',0.2);

% Training set
Xtrain = X(training(cv),:);
Ytrain = Y(training(cv),:);
% Test set
Xtest = X(test(cv),:);
Ytest = Y(test(cv),:);

disp('Training Set')
tabulate(Ytrain)
disp('Test Set')
tabulate(Ytest)



%% Support Vector Machines

opts = statset('MaxIter',30000);
% Train the classifier
svmStruct =
svmtrain(Xtrain,Ytrain,'showplot',true,'kernel_function','rbf','rbf_sigma',1,
'kktviolationlevel',0,'options',opts);

% Make a prediction for the test set
Y_svm = svmclassify(svmStruct,Xtest,'showplot',true);
C_svm = confusionmat(Ytest,Y_svm);
% Examine the confusion matrix for each class as a percentage of the true
class
C_svm = bsxfun(@rdivide,C_svm,sum(C_svm,2)) * 100
figure;
plotconfusion(Ytest',Y_svm')
%error rate
100*(1-sum(abs(Y_svm-Ytest))./109)

%% Show CSP data
load('Plotcoords.mat');
gx=70:270;
gy=100:280;
zz=gridfit(Xaxis,Yaxis,W(1,:),gx,gy);
figure;
subplot(1,2,1),contourf(gx,gy,zz,70)
hold on
set(gca,'color','none')
subplot(1,2,1),scatter(Xaxis,Yaxis,100,'k')

zz=gridfit(Xaxis,Yaxis,W(22,:),gx,gy);
```

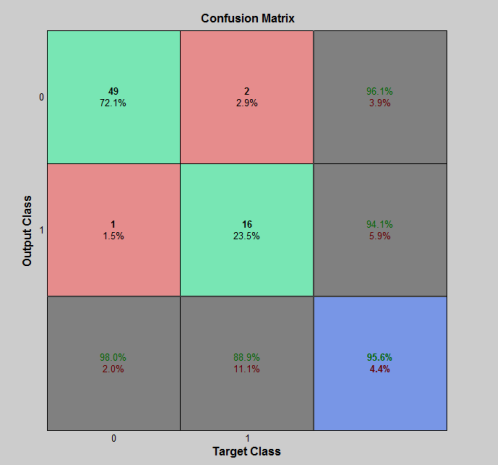
```
subplot(1,2,2),contourf(gx,gy,zz,70)
hold on
set(gca,'color','none')
subplot(1,2,2),scatter(Xaxis,Yaxis,100,'k')

subplot(2,2,3),zz=gridfit(Xaxis,Yaxis,W(2,:),gx,gy);
contourf(gx,gy,zz,70)
hold on
set(gca,'color','none')
subplot(2,2,2),scatter(Xaxis,Yaxis,100,'k')

subplot(2,2,4),zz=gridfit(Xaxis,Yaxis,W(21,:),gx,gy);
contourf(gx,gy,zz,70)
hold on
set(gca,'color','none')
subplot(2,2,2),scatter(Xaxis,Yaxis,100,'k')
```

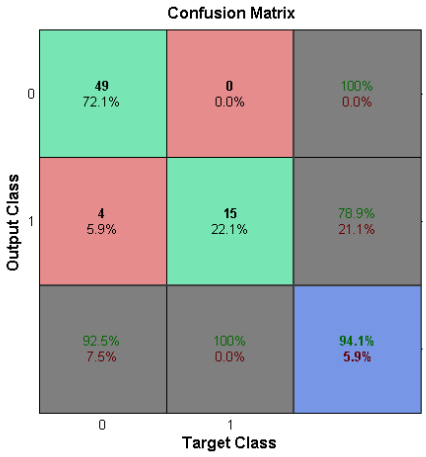
9 RESULTS

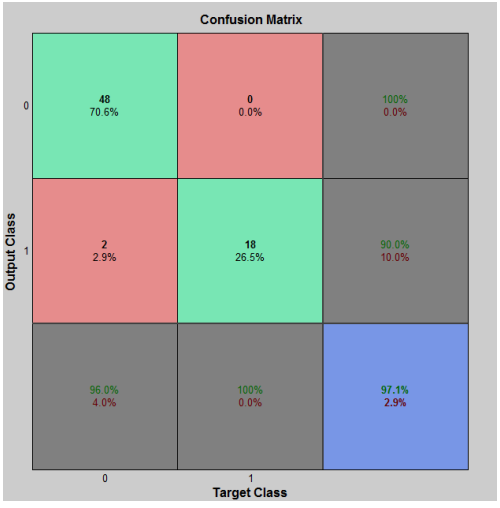
We have successfully classified the movements of left hand, right hand, legs and tongue into four classes respectively. 75 % of the data has been used as the training set to train the classifier and the rest of the data has been used as the testing set. The following confusion matrices have been obtained for respective movements.

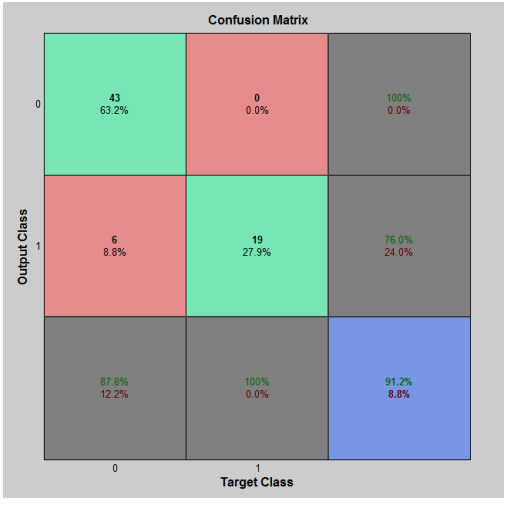
1.Classification rate of class_1=97.8947% 2.Classification rate of class_2=95.7895%

3.Classification rate of class_3=91.2435% 4.Classification rate of class_4=93.6842%









10 REFERENCES

1. BCI Competition 2008 { Graz data set A} C Bruner et al
2. A. Kübler, F. Nijboer, J. Mellinger, T.M. Vaughan, H. Pawelzik, G. Schalk, D.J. McFarland, N. Birbaumer, and J.R. Wolpaw, "Patients with ALS can use sensorimotor rhythms to operate a brain-computer interface," Neurology, vol. 64, no. 10 pp. 1775–1777, 2005.
3. The Wavelet Tutorial,
   Website: http://users.rowan.edu/~polikar/WAVELETS/WTpart1.html
4. Stationary Wavelet Transform
   Website: en.wikipedia.org/wiki/**Stationary_wavelet_transform**
5. Discrete Stationary Wavelet Transform
   Website: http://www.mathworks.in/help/wavelet/ug/discrete-stationary-wavelet-transform-swt.html
6. Support Vector Machines Tutorial
   Website:
   http://www.cs.columbia.edu/~kathy/cs4701/documents/jason_svm_tutorial.pdf
   https://www.statsoft.com/textbook/support-vector-machines
7. Machine Learning with MATLAB
   Website: www.mathworks.in/videos/machine-learning-with-matlab-81984.html
8. Common Spatial Pattern
   Website:
   http://www.slideshare.net/yokotatsuya/introduction-to-common-spatial-pattern-filters-for-eeg-motor-imagery-classification
9. Common Spatial Pattern Method for Channel Selection in Motor Images based BCI.
   Website: sccn.ucsd.edu/~yijun/pdfs/EMBC05.pdf
10. Analysis and simulation of brain signal data by EEG signal processing technique using MATLAB- Guru murthy.
11. Monitoring of micro-sleep and sleep states using EEG signals- Miguel Rivera.