ABSTRACT

Emotion or sentiment is a strong feeling derived from one's circumstances, mood, or relationships with others. It plays an important role in the interaction between humans. There are a number of techniques for the detection of human sentiments using channels such as voice and facial expressions. Many a times these channels can be faked with or without the knowledge of the person. Therefore, making use of a reliable system such as Electroencephalography to record physiological signals becomes vital. The data is collected from a publicly available platform, which is a multichannel (32 Channel) data taken from 14 subjects (7 males/7 females) acquired using the NeuroScan software. The subjects undergo an experiment which gives the EEG signals. The data necessary is derived from these signals. The project is divided into two stages. The first stage of the project is to infer all the necessary information from EEG signals, reduce them to our requirements and identify the sentiment of a subject. Two models are built in this stage. PC-RoDx (Principal Component-Reduction of Dimensions) uses Principal Component Analysis and DEAn (Detection of Emotions and its Analysis) uses Sentimental Analysis. The second stage of the project involves optimizing the models to reach at the best solution. Another model called HOPe (Highly Optimized Performance) is built. This uses High Performance Computing for arriving at the most accurate and optimal model for sentiment detection. The main objective of this project is to find the sentiment of the subject after applying the models PC-RoDx, DEAn and HOPe on the obtained data.

**1. INTRODUCTION**

**1.1. Introduction to Sentiments**

Emotion is an affective state of consciousness in which joy, sorrow, fear, hate, or the like, is experienced. Emotion can also be defined as a mental and physiological state which is associated with a wide variety of feelings, thoughts, and behaviour. Emotions are complex. According to some theories, they are states of feeling that result in physical and psychological changes that influence our behavior. The physiology of emotion is closely linked to arousal of the nervous system with various states and strengths of arousal relating, apparently, to particular emotions. Emotion is also linked to behavioral tendency. Extroverted people are more likely to be social and express their emotions, while introverted people are more likely to be more socially withdrawn and conceal their emotions.

An emotion is a subjective experience which makes studying emotions one of the most confused and still open fields of research in psychology.The reason why studying emotions becomes important is the fact that emotion is an important aspect in the interaction between humans. It is fundamental to human experience, influencing cognition, perception, and everyday tasks such as learning, communication, and rational decision-making. Therefore there is a great interest for detecting emotions automatically.

There are a number of techniques that have been used to detect the emotions of human over the years using channels such as voice and facial expressions. Even though they are a powerful means of interaction between humans, these channels are not reliable and can easily be faked. Hence we are trying to use EEG signals as a new channel for emotion detection.



Figure 1: Examples of basic emotions

**1.2. Introduction to EEG**

The state of mind of a person is supported by the brain activity, and hence features of the state of mind appear in the scalp potentials, as seen on an electroencephalogram (EEG). **Electroencephalography** (**EEG**) is an electrophysiological monitoring method to record electrical activity of the brain. It is typically noninvasive, with the electrodes placed along the scalp, although invasive electrodes are sometimes used in specific applications. EEG measures voltage fluctuations resulting from ionic current within the neurons of the brain. In clinical contexts, EEG refers to the recording of the brain's spontaneous electrical activity over a period of time, as recorded from multiple electrodes placed on the scalp. Diagnostic applications generally focus on the spectral content of EEG, that is, the type of neural oscillations (popularly called "brain waves") that can be observed in EEG signals.

EEG is most often used to diagnose epilepsy, which causes abnormalities in EEG readings. It is also used to diagnose sleep disorders, coma, encephalopathy, and brain death. EEG used to be a first-line method of diagnosis for tumors, stroke and other focal brain disorders, but this use has decreased with the advent of high-resolution anatomical imaging techniques such as magnetic resonance imaging (MRI) and computed tomography (CT). Despite limited spatial resolution, EEG continues to be a valuable tool for research and diagnosis, especially when millisecond-range temporal resolution (not possible with CT or MRI) is required.

The EEG is used in the evaluation of brain disorders. Most commonly it is used to show the type and location of the activity in the brain during a seizure. It also is used to evaluate people who are having problems associated with brain function. These problems might include confusion, coma, tumors and long-term difficulties with thinking or memory, or weakening of specific parts of the body (such as weakness associated with a stroke).

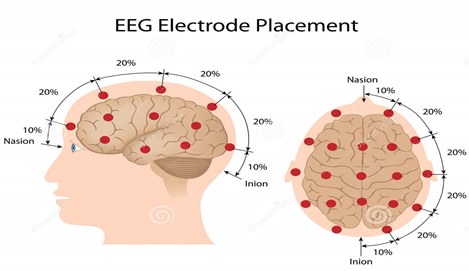


Figure 2: Placement of electrodes according to 10-20 system

**Advantages of EEG**

Several other methods to study brain function exist, including functional magnetic resonance imaging (fMRI), positron emission tomography, magneto encephalography (MEG), [Nuclear magnetic resonance spectroscopy](https://en.wikipedia.org/wiki/Nuclear_magnetic_resonance_spectroscopy), [Electrocorticography](https://en.wikipedia.org/wiki/Electrocorticography), [Single-photon emission computed tomography](https://en.wikipedia.org/wiki/Single-photon_emission_computed_tomography), [Near-infrared spectroscopy](https://en.wikipedia.org/wiki/Near-infrared_spectroscopy) (NIRS), and [Event-related optical signal](https://en.wikipedia.org/wiki/Event-related_optical_signal) (EROS). Despite the relatively poor spatial sensitivity of EEG, it possesses multiple advantages over some of these techniques:

* Hardware costs are significantly lower than those of most other techniques.
* EEG has very high temporal resolution, on the order of milliseconds rather than seconds.
* EEG has a high time resolution, is rather simple in use, cheap and almost does not disturb a subject.

**1.3. 10-20 system or International 10-20 system**

The **10–20 system** or **International 10–20 system** is an internationally recognized method to describe and apply the location of scalp electrodes in the context of an EEG test or experiment. This method was developed to ensure standardized reproducibility so that a subject's studies could be compared over time and subjects could be compared to each other. This system is based on the relationship between the location of an electrode and the underlying area of cerebral cortex. The "10" and "20" refer to the fact that the actual distances between adjacent electrodes are either 10% or 20% of the total front–back or right–left distance of the skull.

Each site has a letter to identify the lobe and a number to identify the hemisphere location. The letters F, T, C, P and O stand for frontal, temporal, central, parietal, and occipital lobes, respectively. (Note: There exists no central lobe; the "C" letter is used only for identification purposes.)

**1.4. Parts of a Human Brain**

The human brain is divided into 4 regions namely- Frontal, Parietal, Temporal and Occipital.

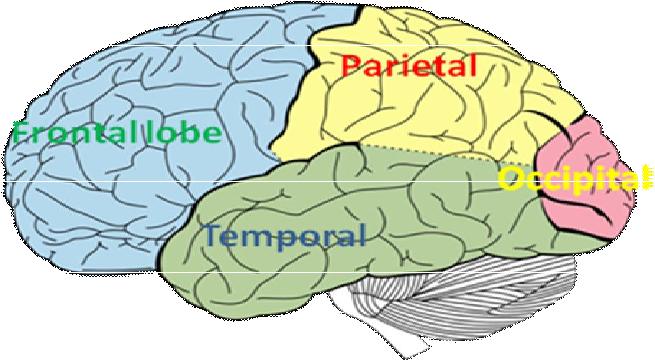


Figure 3: Different parts of the human brain.

The semantic of the EEG signal depends mainly on the places where the signals are captured. Each part of the brain has its own function. The brain has four main areas as shown in Figure 3. The frontal lobe is responsible for body limb movements and facial muscle movements. The parietal region is responsible for sensory information such as taste, pressure, sound and temperature. The occipital region is the center of visual processing. Finally, the temporal region is the center of auditory processing.

**1.5. Electrode Placement**

There are totally 32 electrodes placed on the human brain as shown.

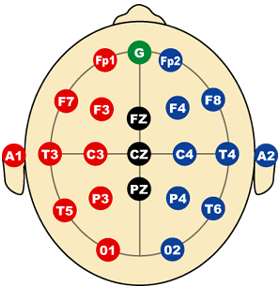


Figure 4: A top view of the brain that shows the placement of EEG electrodes

Even numbers (2, 4, 6, 8) refer to electrode positions on the right hemisphere, whereas odd numbers (1, 3, 5, 7) refer to those on the left hemisphere. A "z" (zero) refers to an electrode placed on the midline.

Only 12 out of 32 electrodes are responsible for finding the sentiments of the subjects. They are FP1, FP2, F3, F4, C3, C4, P3, P4, O1, O2, T3 and T4. Each of these electrodes is present in one of the two hemispheres-right hemisphere and left hemisphere, of the human brain. Negative emotions are captured by the right hemisphere whereas positive emotions are captured by the left hemisphere of the human brain.

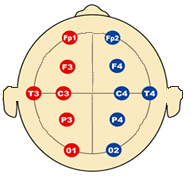


Figure 5: Placement of Sentiment Electrodes

An electrode value is said to be a positive sentiment when some text or speech indicates emotions like happiness, kindness, enthusiasm etc. In contrast, an electrode value is said to be a negative sentiment when it indicates emotions like sadness, violence, hatred etc. In addition to these two, there is a third type of sentiment known as neutral sentiment. In cases where a person has no emotions we say that he/she possesses a neutral sentiment.

**2. Literature Survey**

**2.1. Evolution of Emotions**

The study of the evolution of emotions dates back to the 19th century. Evolution and natural selection has been applied to the study of human communication, mainly by Charles Darwin in his 1872 work, *The Expression of the Emotions in Man and Animals*. Darwin researched the expression of emotions in an effort to support his theory of evolution. He proposed that much like other traits found in animals, emotions also evolved and were adapted over time. His work looked at not only facial expressions in animals and specifically humans, but attempted to point out parallels between behaviors in humans and other animals.

According to modern evolutionary theory, different emotions evolved at different times. Primal emotions, such as fear, are associated with ancient parts of the brain and presumably evolved among our pre-mammal ancestors. Filial emotions, such as a human mother's love for her offspring, seem to have evolved among early mammals. Social emotions, such as guilt and pride, evolved among social primates. Sometimes, a more recently evolved part of the brain moderates an older part of the brain, such as when the cortex moderates the fear response. Evolutionary psychologists consider human emotions to be best adapted to the life our ancestors led in nomadic foraging bands.

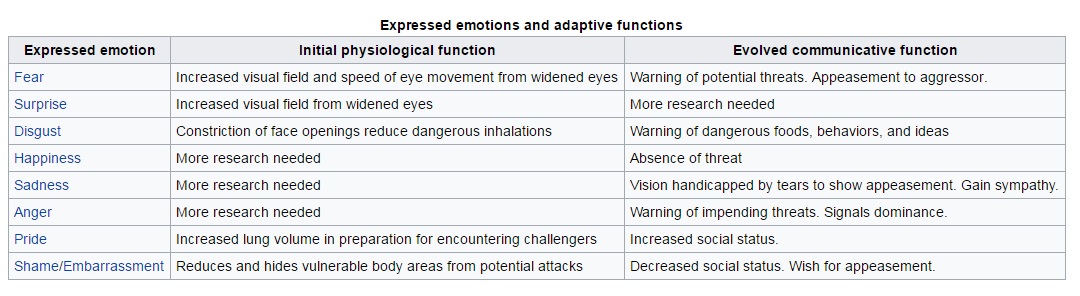


Figure 6: Emotions and their functions

**2.2. Existing Systems**

There has been a lot of work done in the field of emotion and mental state detection by analyzing facial expressions or/and speech. Some of these systems showed a lot of success such as those discussed in “*Mind reading machines: automated inference of cognitive mental states from video*” and “*Real-time emotion detection system using speech: Multi-modal fusion of different timescale features*”.

The system proposed in Mind reading machine uses an automated inference of cognitive mental states from observed facial expressions and head gestures in video. The system is based on a multilevel dynamic Bayesian network classifier which models cognitive mental states as a number of interacting facial and head displays. Here they combines machine vision and supervised statistical machine learning to model hidden mental states of a person based upon the observable facial and head displays of that person. The principle contribution is a multi-level DBN classifier for inferring cognitive mental states from videos of facial expressions and head gestures in real time. The strengths of the system include being fully automated, user-independent, and supporting purposeful head displays while de-coupling that from facial display recognition.

The system in Real-time emotion detection system using speech, proposed in makes use of multimodal fusion of different timescale features of the speech. They also, make use of the meaning of the words to infer both the angry and neutral emotions. The system consists of four major parts; speech acquisition, feature extraction at each timescale level, machine learning for each feature set, and information fusion to merge the information.

**2.3. Drawback of Existing Systems**

Although facial expressions are considered to be a very powerful means for humans to communicate their emotions, the main drawback of using facial expressions or speech is the fact that they are not reliable indicators of emotion because they can either be faked by the user or may not be produced as a result of the emotion.

The other alternative for emotion and cognitive state detection is analyzing physiological signals because they are not experiencing the same drawback of video and speech. These types of signals cannot be faked due to the fact that they are produced from some involuntary secretion glands as a result of specific stimulus.

Some of the systems that rely on detecting physiological signals make use of the signals generated from the peripheral nervous system such as skin temperature variation, heart rate, blood pressure and skin conductance.

One of the systems that was able to classify four different emotions, anger, sadness, stress and surprise is proposed by Kim. In this system, he made use of ECG and body temperature to recognize the four emotions. They tested their hypothesis on large dataset generated from 50 subjects and were able to reach an accuracy of 78.4% and 61.8% for three and four emotion categories respectively. Based on the cognitive theory of emotion, the brain is the center of every human action.

Consequently, emotions and cognitive states can be detected through analyzing physiological signals that are generated from the central nervous system such as EEG signals. However, there is little work done in this area of research. Thanks to the success of brain computer interface systems, a few new studies have been done to find the correlation between different emotions and EEG signals. Most of these studies combine both EEG signals with other physiological signals generated from the peripheral nervous system. However, in this research, we will focus on inferring emotion from EEG signals.

**3. Proposed System**

**3.1. Objectives**

* The main purpose of the proposed system is to carry out analysis to investigate which type of sentiment an individual possesses while performing tasks such as categorization and recognition.
* The information is collected using a channel called Electroencephalography (EEG). The first step is to infer all the necessary information from EEG signals and reduce them to our requirement by building a model known as PC-RoDx (Principal Component-Reduction of Dimention).
* The second step is to detect the sentiments of a person using a model named DEAn (Detection of Emotions and its Analysis).
* Finally we optimize these two models to arrive at the most accurate and optimal solution. This is done using the third model HOPe (Highly Optimized Performance).

**3.2. Methodology**

After the EEG signals were obtained by performing certain experiment, it was preprocessed to make it understandable by the user, so that it must serve as an input to the proposed system, PC-RoDx. The data is converted to a CSV file format in the Microsoft Excel. CSV stands for “Comma-Separated Values”. It is then loaded to the R environment in a Rstudio platform.

The main objective of PC-RoDx (Principal Component –Reduction of Dimensions) is Dimensionality Reduction. This means that any data that has dimensions (i.e columns) greater than 10, have to be reduced to dimensions lower than their initial dimensions. Correlation between the channels for a single subject is observed after selecting a particular subject’s EEG signal data. We then eliminate the correlation and we obtain the reduced principal component (PCs). These PCs are extracted as loadings and serve as an input to Sentimental Analysis.

Sentiment Analysis is a technique that helps us in analyzing the sentiments of a subject which can either be a positive or a negative sentiment. In essence, it is the process of determining the emotional tone behind a series of words, used to gain an understanding of the attitudes, opinions and emotions expressed. Mathematical operations on the dataset were performed to obtain the polarity of the electrodes. Finally the data was sorted subject-wise and then categorized into positive or negative values. The sum of the components was calculated for all the electrodes and the result was plotted. The plot above the point 0 corresponded to the sentiment of the subject being positive which indicated that he/she was happy with the experiment whereas the plot below 0 represented a negative sentiment which shows the subject was unhappy with the experiment.

The sentiments of a subject are captured as EEG signals by the process of Electroencephalography and how we reduced the number of electrodes to just 12 by concentrating on only those which were within our requirements. Further, with the help of this data, models were built for the analysis and detection of sentiments using two prominent techniques called the Sentiment Analysis and Principal Component Analysis. Finally these models were then optimized in terms of accuracy and the time and space it consumed by using High Performance Computing (HPC) or simply the HOPe model. The HOPe model included profiling of the R code, finding the time each piece of code and finally memory profiling to examine how much space in memory each piece of code took, thereby increasing its efficiency.

**3.3. Data Acquisition Process**

In our analysis we are using EEG data. In an experiment, 14 subjects comprising of 7 male and 7 female, were seated in a room in front of a computer screen. Images were flashed in front of the subject in an interval of 20ms and he/she was asked to identify i.e. categorize and recognize the images. Participants gave their responses following a go/nogo paradigm. The result of this could either be a positive sentiment or a negative sentiment. A positive sentiment denoted that the subject was successfully able to identify the image and a negative sentiment denoted that he/she was not able to identify the image.

In the animal categorization task, participants had to respond whenever there was an animal in the picture whereas the recognition task started with a learning phase. Participants were instructed to carefully examine and learn the probe image in order to recognize it in the following series.

The EEG signals were recorded from 32 electrodes mounted on the scalp of the subject. 25 groups of file (13 from the first day and 12 from the second day), each group of file corresponding to 100 were recorded for each subject.

**3.4. Data Pre-processing**

Figure 7: Steps involved in Pre-processing stage

As observed in the figure 7, after the EEG signals have been recorded, they are filtered and merged together. It was found that out of 32 channels only 12 channels (or electrodes) were responsible for recording the sentiments of a subject during the experiment which helps us in cleaning the data.

The EEG data is converted to a CSV file format in Microsoft Excel. CSV stands for "**C**omma-**S**eparated **V**alues". Its data fields are most often separated, or delimited, by a comma. Files in this format can be imported to and exported from programs that store data in tables. Finally the CSV file needs to be loaded to the R studio workspace or R environment. The 12 sentiment electrodes can be then extracted from the loaded data. These are referred to as processed data which serves as input data to PC-RoDx.

**4. Software Requirements**

**4.1. Introduction to RStudio**

The program was written and compiled in RStudio. RStudio is a free and open-source integrated development environment (IDE) for R. It was founded by JJ Allaire. RStudio makes R easier to use. It includes a console, syntax-highlighting editor that supports direct code execution, as well as tools for plotting, history, debugging and workspace management.

RStudio is mainly available in two editions

1. RStudio Desktop- Here the program is run locally as a regular desktop application

2. RStudio Server- It allows accessing RStudio using a web browser while it is running on a remote Linux server.

Prepackaged distributions of RStudio Desktop are available for Windows, macOS, and Linux. RStudio is available in open source and commercial editions and runs on the desktop (Windows, macOS, and Linux) or in a browser connected to RStudio Server or RStudio Server (Debian/Ubuntu, RedHat/CentOS, and SUSE Linux). RStudio is written in the C++ programming language. RStudio version v1.0.136 was released on December 21st, 2016.

**4.2. Introduction to R Language**

The R language was created by Ross Ihaka and Robert Gentleman at the University of Auckland, New Zealand. It is an open source programming language and software environment for statistical computing and graphics that is supported by the R Foundation for Statistical Computing. R is a GNU package. The source code for the R software environment is written primarily in C, Fortran, and R. It is freely available under the GNU General Public License.

R has stronger object-oriented programming facilities than most statistical computing languages. It is an interpreted language; users typically access it through a command-line interpreter. R supports procedural programming with functions and, for some functions, object-oriented programming with generic functions.

The capabilities of R are extended through user-created packages, which allow specialized statistical techniques, graphical devices, import/export capabilities, reporting tools etc. These packages are developed primarily in R, and sometimes in Java, C, C++, and Fortran.

A core set of packages is included with the installation of R, with more than 7,801 additional packages (as of January 2016) available at the Comprehensive R Archive Network (CRAN), Bioconductor, Omegahat, GitHub, and other repositories.

**Why R?**

There are many programming languages available these days. The following table shows the comparison of R with other programming languages.

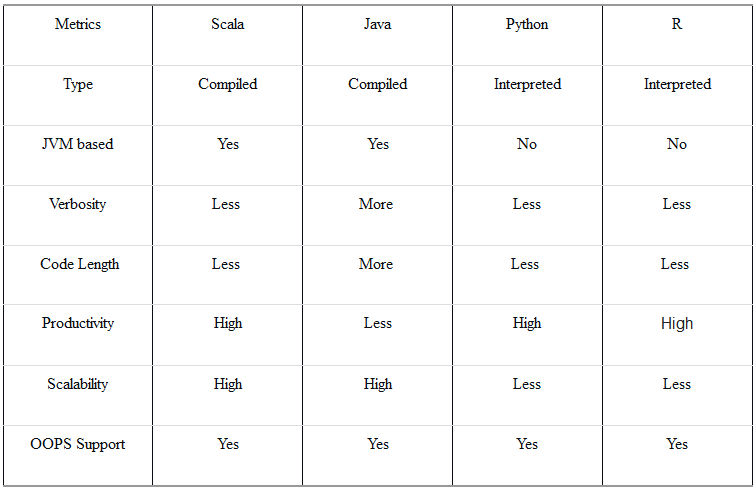
****

Figure 8: Comparison of R with other languages

## 4.3. Microsoft Excel

## It is the world’s most widely used spreadsheet program. For the project we have used 2010 version of excel.

## Microsoft Excel has the basic features of all spreadsheets, using a grid of cells arranged in numbered rows and letter-named columns to organize data manipulations like arithmetic operations. It can display data as line graphs, histograms and charts, and with a very limited three-dimensional graphical display. It has a programming aspect, Visual Basic for Applications, allowing the user to employ a wide variety of numerical methods, for example, for solving differential equations of mathematical physics, and then reporting the results back to the spreadsheet.

**4.3. System Specification**

The specifications of the four systems used during the project are:

1. LENOVO-PC

Rating: 4.9

Processor: Intel(R) Core(TM) i3-3110M CPU @ 2.40GHz 2.40 GHz

Installed memory (RAM): 2.00 GB

System type: 64-bit Operating System

Edition: Windows 7 Professional

2. DELL INSPIRON

Rating: 7.1

Processor: Intel(R) Core(TM) i3-5005U CPU @ 2.00GHz 2.00 GHz

Installed memory (RAM): 4.00 GB

System type: 64-bit Operating System

Edition: Windows 10 Home Single Language

3. DELL INSPIRON N4110

Rating: 4.7

Processor: Intel(R) Core(TM) i5-2430M CPU @ 2.40GHz 2.40 GHz

Installed memory (RAM): 4.00 GB

System type: 64-bit Operating System

Edition: Windows 7 Home Basic

4. DELL-PC

Rating: 3.4

Processor: Intel(R) Core(TM)2 Duo CPU T6600 @ 2.20GHz 2.20 GHz

Installed memory (RAM): 3.00 GB

System type: 32-bit Operating System

Edition: Windows 7 Ultimate

**5. Software Requirement Specification**

**5.1. Extendibility**

The system developed can be quite easily extended to add new functionality. It is flexible enough to adapt to new changes that would be added in future.

**5.2. Portability**

The system was developed on RStudio which can be installed on Windows, Linux and Mac OS. Therefore the system is portable in a way wherein it can easily adapt to the RStudio platform of choice.

**5.3. Maintainability**

The system is easily maintainable since the code written is easy and simple to understand even for the beginner with basic knowledge of RStudio.

**5.4. Modifiability**

At any point of time this system can be updated or re-modified to suit the required needs.

**5.5. Scalability**

The number of subjects can be increased according to the scenario (all the subjects in case of the categorization and recognition experiment) since the proposed system works very well on large datasets.

**6. System Design Specification**

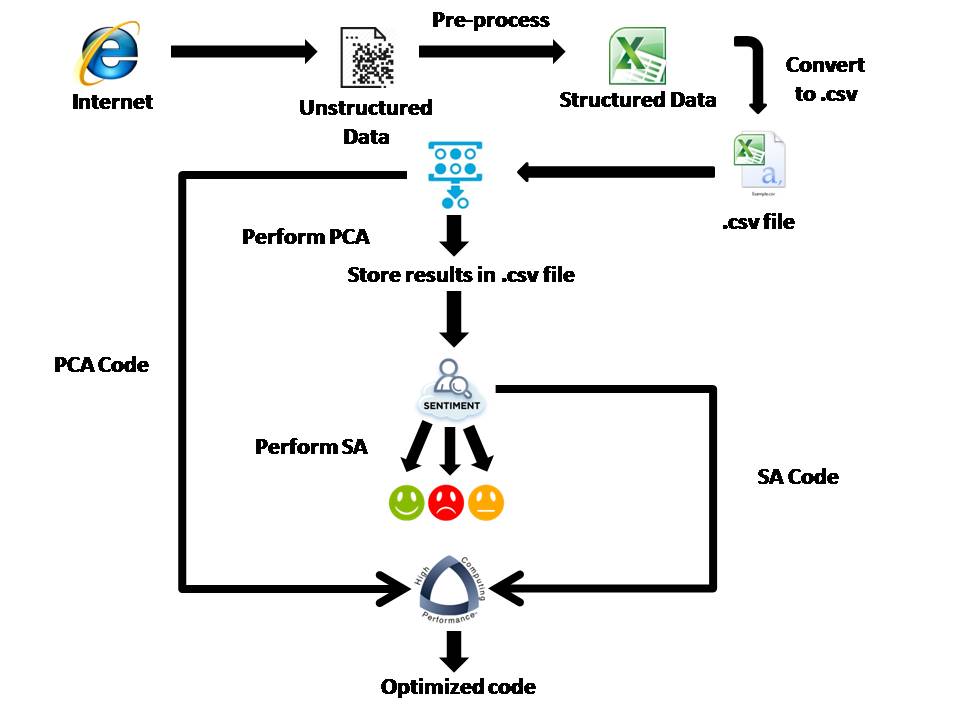
**6.1. High-Level Design**

**PC-RoDx**

**HOPe**

**DEAn**

Figure 9: High-Level Design

**6.2. System Architecture**Figure 10: System Architecture

**6.3. Data Flow Diagram**

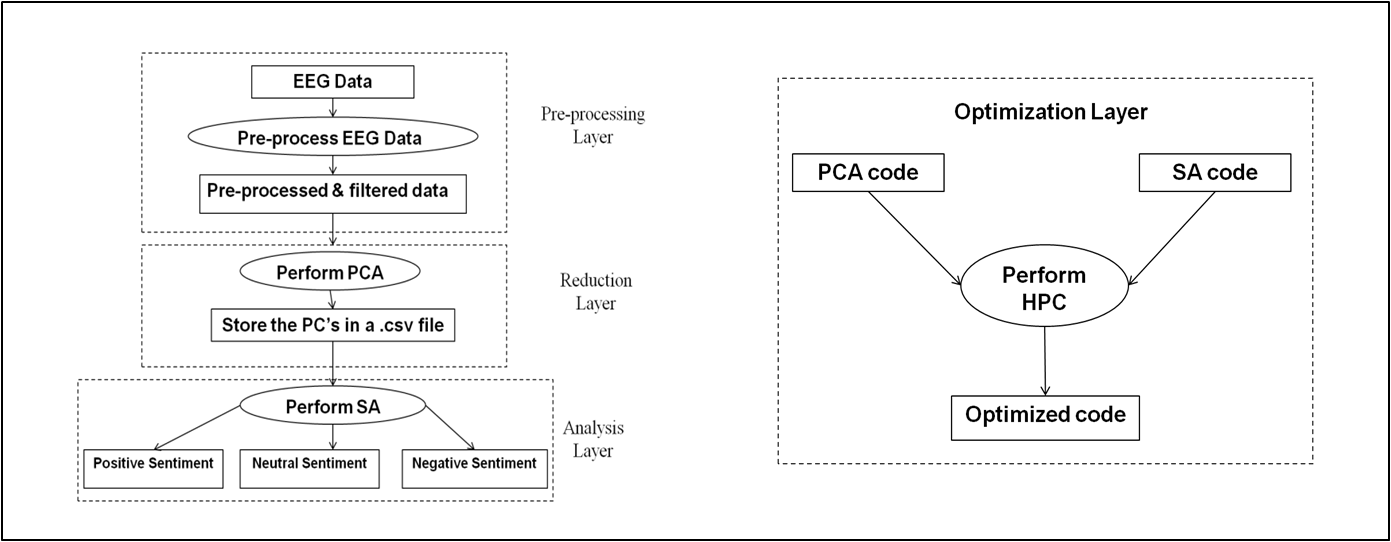
****

Figure 11: Data Flow Diagram

**6.4. Sequence Diagram**

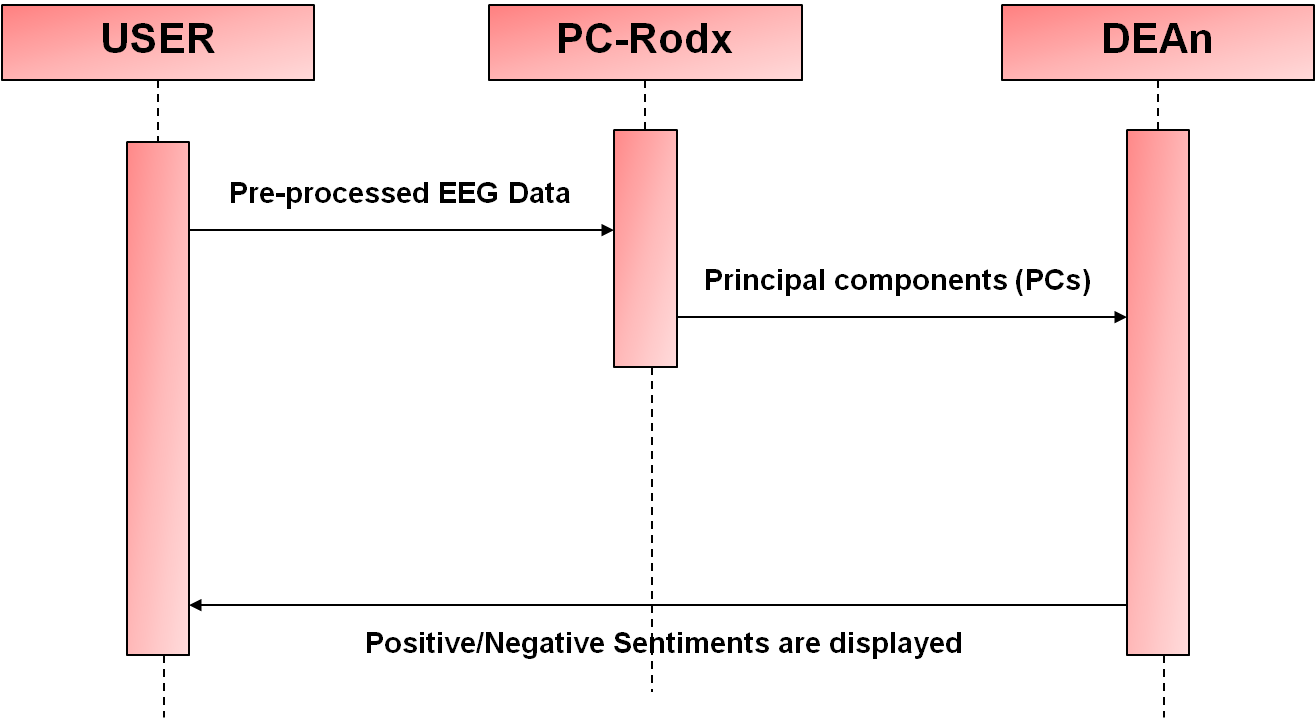
****

Figure 11: Sequence Diagram

**6.5. Detailed Design**

**6.5.1. PC-RodX: Detailed Design**

An R code is developed to perform the programming part for PC-RoDx. The steps in the program flow have been illustrated in figure 12.

After the preprocessed state, it has been found that out of 32 channels only 12 channels (or electrodes) are responsible for recording the sentiment of a subject during the experiment. These are then extracted to a separate variable. As there are 14 subjects, the user is given a chance to choose a particular subject to check his/her sentiment.

For the particular data obtained, the different electrodes must be then checked for correlation. Correlation is how much one electrode is dependent of another electrode.

Then Principal Component Analysis is performed on the data by using an R function called as - prcomp() command.

By using the Kaiser’s rule, we choose the PCs that have maximum variance. It is known that the first few PCs have greater variance.

Finally the loadings or rotation of each PCs can be extracted and written onto an external file such as a CSV file. This data serves as an input to the next stage, which is Sentimental Analysis.

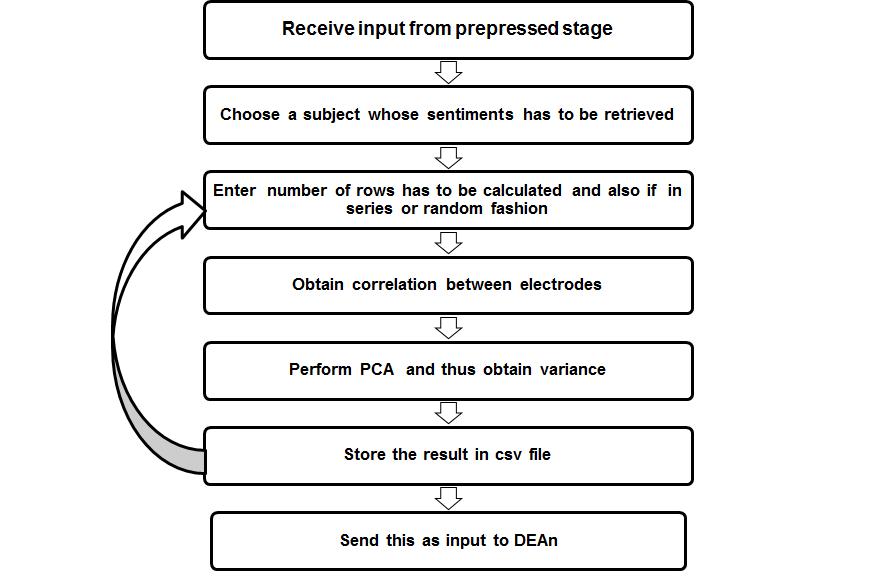


Figure 12: PC-RodX Program Flow

**6.5.1.1 Aspects and their functionalities**

**6.5.1.1.1 Correlation**

Correlation between the channels for a single subject is observed after selecting a particular subject’s EEG signal data.  Correlation is a statistical method which enables the researcher to find whether two variables are related and to what extent they are related.

It is considered as the sympathetic movement of two or more variables. We can observe this when a change in one particular variable is accompanied by changes in other variables as well, and this happens either in the same or opposite direction, then the resultant variables are said to be correlated.

There are three types of Correlation:

1. Positive Correlation:  when values of one variable increase with the increase in another variable
2. Weak Correlation:  if the values of one variable decrease with the decrease in another variable
3. No Correlation: no change in a variable with any change in another variable

**6.5.1.1.2 Analysis Technique**

The underlying technique that was studied to perform PC-RoDx is Principal Component Analysis. The origin of principal component analysis (PCA) has been traced by Jolliffe to Pearson and Hotelling.

The following table illustrates as to why use PCA,

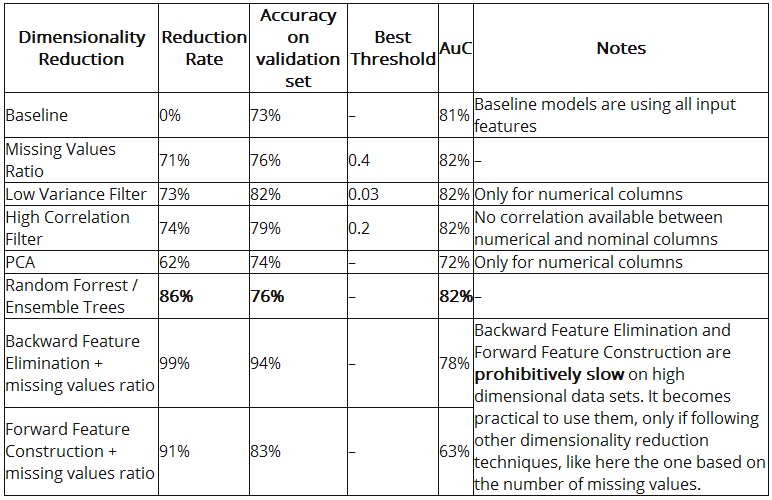


Figure 13: Features of PCA

Two applications of PCA are data reduction and interpretation. PCA reduces the dimensionality of a data set involving a large number of variables. The original sets of variables are transformed into a new set of variables called the principal components (PCs). The PCs are ordered so that the first variables represent most of the variations in the original variables.

If a multivariate dataset is visualized as a set of coordinates in a high-dimensional data space (1 axis per variable), PCA can supply the user with a lower-dimensional picture, a projection of this object when viewed from its most informative viewpoint. This is done by using only the first few principal components so that the dimensionality of the transformed data is reduced.

This technique is especially useful when the variables within the data set are highly correlated and when there is a higher than normal ratio of explanatory variables to the number of observation. Principal components seeks to transform the original variable to a new set of variables that are

Linear combinations of the variables in the data set,

1. Uncorrelated with each other, and
2. Ordered according to the amount of variation of the original variables.

These new set of variables are called as Principal Components. The number of principal components is less than or equal to the smaller of the number of original variables or the number of observations.

The results of a PCA are discussed in terms of component scores or loadings. These are the weight by which each standardized original variable should be multiplied.

**6.5.1.1.3 Dimensionality Reduction**

The main objective of PC-RoDx (Principal Component – Reduction of Dimensions) is Dimensionality Reduction. This means that any data that has dimensions (i.e columns) greater than 10, have to be reduced to dimensions lower than their initial dimensions.

The transformation, **T** = **X** **W**, Where **X** is a data vector, **W** is weight or loadings are being transformed from an original space of *p* variables to a new space of *p* variables which are uncorrelated over the dataset.

The loadings are locations along each component (or eigenvector) are then associated with values across all variables. This association between the components and the original variables is called the component’s eigenvalue.

If there are large differences between the variances of the elements of X, then those variables whose variances are largest will tend to dominate the first few PCs.

If the standard deviations (and thus the variances) are quite similar it is possible to use the covariance matrix to carry out the PCA. If the variances are very different, it is advisable to use the correlation matrix to carry out the PCA. This is equivalent to perform the PCA on the covariance matrix of the standardized data.

**6.5.1.1.4 Criterion to select Principal Components**

However, not all the principal components need to be kept. According to the cumulative proportion of explained variance criterion, that suggests to retain as many PCs as are needed in order to explain approximately 80 to 90% of the total variance.

There are two prominent techniques to identify the principal components.

1. **The Kaiser criterion**: This rule was proposed by Kaiser (1960), and is widely used. It suggests to retain as many principal components as are the eigenvalues of R larger than 1. The motivation underlying this rule is that one could retain all the principal components that have a variance larger than the one related to the original variables (that is equal to 1 for standardized data).
2. **The scree test**: This is a graphical method first proposed by Cattell (1966). Here eigenvalues must be plotted as a simple line plot. In general, it is observed that smooth decrease of eigenvalues appears to level off to the right of the plot. The flat portion corresponds to noise components, unable to capture the leading variability; Hence only those PCs must be selected till where the elbow of the scree plot occurs.

There often exists a dilemma of which of the two techniques seem plausible.

It appears as if the first method (Kaiser Criterion) sometimes retains too many factors, while the second technique (scree test) sometimes retains too few.

Both do quite well under normal conditions, that is, when there are relatively few factors and many cases.

In practice, an additional important aspect is the extent to which a solution is interpretable. Therefore, one usually examines several solutions with more or fewer factors, and chooses the one that makes the best "sense."

**6.5.1.1.5 Writing all relevant values to a Text File**

After the output has been generated, it is converted into the CSV file internally using an R function. This output can be visualized as a set of PCs. This output, however contributes only to the selected subject.

All the 12 electrodes placed contribute to the variance. As emphasized, the results may vary if a different subject chosen or if the amount of data selected is different.

**6.5.1.1.6 Graph of Results**

**6.5.1.1.6.1 Output as Biplot**

Another way to visualize this output is through Biplot.

In general it assumes that two components explain sufficient amount of the variance to provide a meaningful visual representation of the structure of cases and variables.

The plot is showing:

1. The score of each case (i.e., time each signal was recorded) on the first two principal components; numbers

2. The loading of each variable (i.e., each sporting event) on the first two principal components; arrows

The left and bottom axes are showing [normalized] principal component scores and the top and right axes are showing the loadings.

**6.5.1.1.6.2 Output as a dot plot**

This plot shows that the data is independent from each other. There exists no correlation and hence the data is called uncorrelated data.

As we can visualize, the plot shows each component’s relation with every other component. No values in any components overlap with each other.

**6.5.2. DEAn: Detailed Design**

An R code is developed in order to perform the programming part for DEAn. The steps involved have been illustrated as shown in figure 14.

The first model PC-RoDx gives PCs as the output which serves as an input to the second model DEAn(Detection of Emotions and its Analysis). The next step is to calculate the sum of the PCs. Finally an analysis technique known as Sentiment Analysis is performed on this.

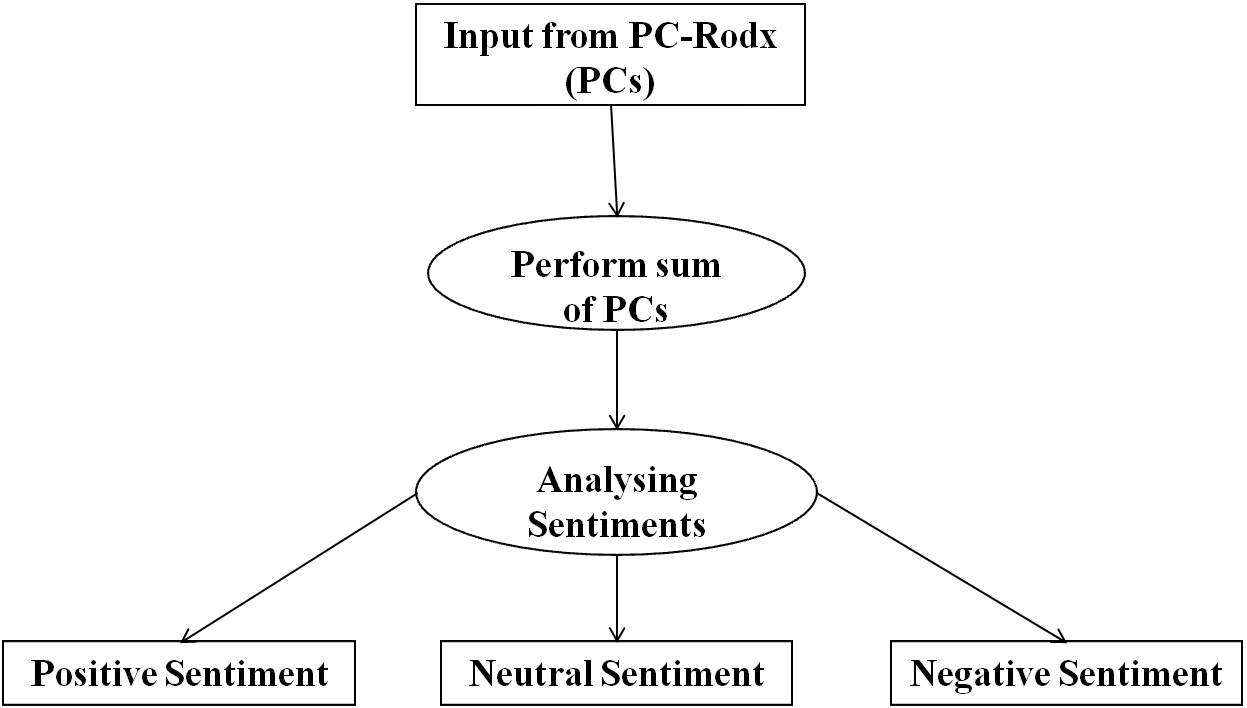


Figure 14: DEAn Program Flow

**6.5.2.1. Sentiment Analysis**

Emotions accompany everyone in daily life playing a key role in non-verbal communication, and they are essential to the understanding of human behavior. Emotion recognition could be done from text, speech, facial expression or gesture. Here, we concentrate on the recognition of inner emotions from electroencephalogram (EEG) signals as humans could control their facial expressions or vocal intonations.

Sentiment Analysis is a technique that helps us analyze the sentiments of a subject which can be positive or negative. If the sentiment was a positive one, it indicates that the subject could successfully categorize and recognize the image that was displayed in front of them. A negative sentiment represents a failure of the experiment.

**Applications of Sentiment Analysis**

There are many applications of Sentiment Analysis. A common use case for this technology is to discover how people feel about a particular topic. For example, twitter, with over 319 million active users and over 500 million tweets per day, has now become a goldmine for organizations and individuals. Sentiment analysis offers these organizations the ability to monitor different social media sites in real time. Twitter sentiment classification aims to classify the sentiment polarity of a tweet as positive, negative or neutral. This helps us in knowing views of different people all over the world about a particular topic.

The applications of sentiment analysis are broad and powerful. The ability to extract insights from social data is a practice that is being widely adopted by organizations across the world. SA has been useful in many other areas like Politics/political science, Law/policy making, Sociology and Psychology.

A basic task in sentiment analysis is classifying the polarity of a given text at the document, sentence, or feature/aspect level—whether the expressed opinion in a document, a sentence or an entity feature/aspect is positive, negative, or neutral.



Figure 15: Polarity of Sentiment

**6.5.2.2. Chi-Squared Test**

While performing mathematical operations like mean, median, minima and maxima of a specified column(s) i.e. the electrode values, which helps us in finding the polarity of the electrodes, it was found that three of the 12 electrodes did not match the theoretical aspects. These three electrodes were FP2, P4 and T4. In order to overcome this problem, a popular test known as chi-squared test (short for Pearson's chi-squared test) was performed for only these three electrodes.

The chi-squared test, also written as *χ*2 test**,** is applicable to many situations in which experimental frequencies are compared to theoretical frequencies based on a hypothesis. 'chi-squared test' often is used as short for Pearson's chi-squared test. The formula for *χ*2 is as follows



Under the null hypothesis, it has approximately a chi-squared distribution whose number of degrees of freedom is given by

(Number of rows-1)(Number of columns-1)

The chi-square test is used in testing the null hypothesis, which states that there is no significant difference between the expected and observed result. If the obtained value is very high, we reject the null hypothesis and conclude that there is a significant difference among the expected and observed frequencies. The p-value helps us in finding this. It is the probability of observing a sample statistic as extreme as test statistic. We reject the null hypothesis when the p-value is less than the significance level which is any value between 0 and 1. . In our case after performing the function it was found that the p-value was less than the significance level. Therefore, we rejected the null hypothesis.

Finally we categorize the sentiment of a person into either a positive value or a negative value with respect to selected subjects. This is done using a graph which is plotted in order to visualize the output.

**6.5.3. HOPe: Detailed Design**

Programmers waste enormous amounts of time thinking about, or worrying about, the speed of noncritical parts of their programs, and these attempts at efficiency actually have a strong negative impact when debugging and maintenance are considered. Thus, we make use of a technique to optimize the code and make it run faster and in a much more efficient way. We call this technique as High Performance Computing (HPC).

High Performance Computing most generally refers to the practice of aggregating the computing power in a way that delivers much higher performance than one could get out of typical desktop computer or a workstation in order to solve large problems in science, engineering or even business. In other words HPC can also be defined as the use of parallel processing for running advanced application programs efficiently, reliably and quickly.

In order to implement High Performance Computing (HPC), we build a new model called HOPe which is an acronym for Highly Optimized Performance. In HOPe, we use a set of techniques that is used to optimize the other 2 models i.e. DEAn and PC-RoDx.

Optimizing a code or a model is not as easy as it sounds. Optimizing the code in order to make it run faster is an iterative process. It includes the following steps:

1. Find the biggest bottleneck i.e. the slowest part of your code.
2. Try to eliminate it.
3. Repeat it until your code is fast enough.

HPC includes the following:

* Measuring performance describes how to find the bottlenecks in your code using line profiling. To understand performance, you use a profiler. There are a number of different types of profilers. R uses a fairly simple type called a sampling or statistical profiler. A sampling profiler stops the execution of code every few milliseconds and records which function is currently executing (along with which function called that function, and so on).
* Improving performance outlines seven general strategies for improving the performance of your code.
* Code organization teaches you how to organize your code to make optimization as easy, and bug free, as possible.
* Parallelize teaches you how to use parallelization to spread computation across all the cores in your computer.

To Implement HPC in HOPe, we use the following techniques:

1. Profiling R code
2. Using Time\_sys function

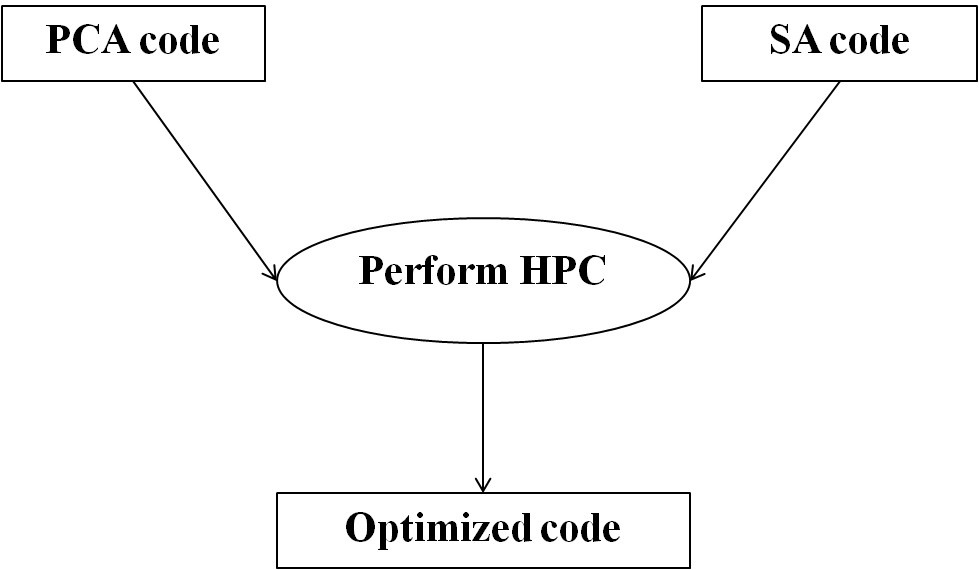


Figure 16: HOPe Program Flow

**6.5.3.1. HPC Techniques**

**6.5.3.1.1. Profiling R code**

Profiling R code gives you the chance to identify bottlenecks and pieces of code that needs to be more efficiently implemented. We can reduce the amount of time necessary to run an R routine by as much as 90% with very simple changes to our code. Profiling is the analysis of a computer program performed by measuring the time spent on each line of code, code coverage or memory usage during its execution. Profiling is the first step towards efficient programming. The development of efficient software depends on identification of key bottlenecks. Focusing the optimization only on the bottlenecks is known to maximize efficiency in both development time and program runtime.

R includes a sampling based profiling mechanism that records information about calls on the stack at specified time intervals. The traditional way to collect profiling data in R is to call Rprof to start profiling, run the code to be profiled, and then call Rprof again to end profiling.

In previous versions of R (prior to 3.0), Rprof only worked at the function level (i.e., the profiler only provided information on the functions in the stack). Since version 3.0, it is possible to perform line profiling. If the line.profiling option is selected, the file generated by Rprof also includes information on the specific line of code in the stack (not only the function). This extension made it possible to identify the specific lines that slow down the code.

The output of Rprof is quite simple: a file that shows the name and some other characteristics of a function every time it is found to be in the stack. The most basic summary function is summaryRprof for summarizing profile results. Rprof works by recording at fixed intervals (by default every 20 msecs) which R function is being used, and recording the results in a file. summaryRprof will give you a list with four elements:

* by.self: the time spent in function alone.
* by.total: the time spent in function and callees.
* by.line: the time spent in that line of code.
* sample.interval: the sampling interval, by default every 20 msecs.
* sampling.time: the total time of profiling run.

The syntax of Rprof is as shown below:

**Rprof(filename = "Rprof.out", append = FALSE, interval = 0.02, memory.profiling = FALSE, gc.profiling = FALSE, line.profiling = FALSE, numfiles = 100L, bufsize = 10000L)**

The syntax of summaryRprof is as shown below:

**summaryRprof(filename = "Rprof.out", chunksize = 5000 memory = c("none", "both", "tseries", "stats"), lines = c("hide", "show", "both"),index = 2, diff = TRUE, exclude = NULL, basenames = 1)**

**6.5.3.1.2. Using Time\_sys function**

The time\_sys() is a user defined function which comprises of a standard R function called system.time(). The system.time() is used to find the time taken for each function to process or in other words, it is used to time how fast R processes an expression or a function. This function is especially useful for comparing the speed of 2 coding options. However, the system.time command takes a single R expression as its argument. The implementation of this function remains very simple. Its syntax is as follows:

**system.time(expr)**

* expr – An R expression to evaluate.

**7. Testing**

Testing is the stage of implementing which is aimed at earning system running accurately and efficiently. An error or anomaly in program code can remain undetected indefinitely. To prevent this from happening, the code is tested at each level. Testing is performed to ensure that the system as a whole is bug free. At each stage or phase a different technique for eliminating the errors that exists is used. However, some requirement error and design errors are likely to remain undetected. Ultimately, these errors will be reflected in the code. Testing is usually associated with the code and is used to detect the errors remaining from the earlier phase. The performance factors like turnaround time, back up, file protection and human factors are some of the performance criteria for system testing. Hence testing performs a critical role for quality assurance and for ensuring the reliability of software.

**7.1. Unit testing**

The first level of testing is unit testing. In this, the smallest units of software design and the modules are tested against the specifications processed during the design. It consists of a number of test runs such as valid path through the code, and the exception and error handling paths. Unit testing is essentially for verification of the code produced during coding phase and hence the goal is to test the internal logic of the modules. Unit testing involves checking all pages for errors and omissions. Each module of this project was found to be working according to the expected output from the module.

**7.2. System testing**

System testing ensures that the entire integrated software system meets its requirements. It tests a configuration to ensure known and predictable results. System Testing involves the integration of the individual units of the system that were tested during unit testing. We make a note of how our integrated model works on four different systems. Then we perform High Performance Computation on the obtained result, and find out which system is faster. Hence we can successfully conclude that our project “Multichannel EEG based Principal Component Sentiment Disorder Detection Model” works well as per the specified requirements.

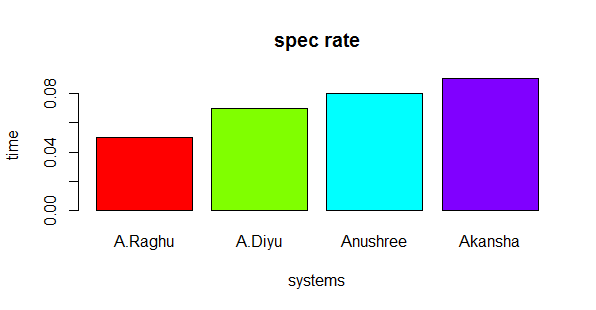
****

Figure 17: Observed SPEC rate for all the four systems

**8. Implementation**

**8.1. PC-RoDx: Implementation**

**1. Main.PCA()**

This is the main function that corresponds to finding the principal components

Step 1: Start

Step 2: [Read the file from a csv file]

my.data <- call ReadFun()

Step 3: [Store only the relevant sentiment electrodes]

12 Relevant sentiment electrodes with their subject name

Preprocessed.data <- my.data[c(1,3,4,5,6,7,8,9,10,11,12,15,16)]

Step 4: [Prompt user to select name of user from the list]

print("Choose one SUBJ in the Program. 1: CBA 2: CLM 3: EGA 4: FSA 5: GMI 6: GRO 7: HTH 8: MBA 9: MMA 10: MTA 11: PLA 12: SCE 13: SPH 14: WPA ")

ANS <- (name\_of\_subj)

Step 5: [Choose EEG data for only selected subject]

Switch ANS do

Case CBA

name <- EEG data for this subject

Case CLM

Name <- EEG data for this subject

This repeates for the remaining 12 subjects

Step 6: [Prompt user to select random or series data]

If series

Set <- call Series()

Else

Set <- call PCADecisionBoundary()

Step 7: [Perform PCA]

The principal Component Analysis is done here with the help of a built-in R function

My.pr <- call PrincompFunc()

Step 8: [Find and print Eigen Values]

Std.dev <- call ChoosePCA

Print Eigen values

Step 9: [Find relevant eigen values or squared standard deviation]

X1 <- call Comp.used()

Print X1

Step 10: [Print max in X1 and store in X2]

X2 <- max(X1)

Step 11: [Write output to csv file]

loadings <- as.data.frame(my.pr$loadings[,1:x])

write loadings into CSV file

Step 12: Plot uncorrelated graph

Step 13: Plot Biplot

Step 14: Plot Dot plot

Step 15: Stop

**2. Series()**

To select series data from the EEG data of each subject

Step 1: Start

Step 2: Input X and N

X is the data under each subject and N is the number of data to be selected

Step 3: Return only selected rows

Step 4: Stop

**3. PCADecisionBoundary()**

To select random data from the EEG data of each subject

Step 1: Start

Step 2: [Take any random data from the data]

rand <- x[sample(nrow(x), 100), ]

Step 3: [Return only selected rows]

return(see)

Step 4: Stop

**4. comp.used (X)**

Chose only relevant PCs using Kaiser’s criterion

Step 1: Start

Step 2: [Choose those components that are greater than 1 ]

chosen <- {

While x >1

}

Step 3: [Print the components and return them]

print("The component to be used are :")

return(chosen)

Step 4: Stop

**5. ChoosePCA(X)**

Chose all PCs

Step 1: Start

Step 2: [Obtain all Eigen values]

EV <- x$sdev^2

Step 3: [Return the values]

Step 4: Stop

**8.2. DEAn: Implementation**

**1. compute()**

Given the data for all the electrodes, this algorithm finds the mean, median, minima and maxima of the electrodes

STEP 1: START

STEP 2: [Read the values of electrodes]  
    INPUT FP1, FP2, F3, F4, C3, C4, P3, P4, O1, O2, T3, T4

STEP 3: Make computations:

* [Calculate the mean]  
      x̅ (FP1, FP2, F3, F4, C3, C4, P3, P4, O1, O2, T3, T4)
* [Calculate the median]  
      median symbol( FP1, FP2, F3, F4, C3, C4, P3, P4, O1, O2, T3, T4)
* [Calculate the minima]  
       ∧ (FP1, FP2, F3, F4, C3, C4, P3, P4, O1, O2, T3, T4)
* [Calculate the maxima]  
      ∨ (FP1, FP2, F3, F4, C3, C4, P3, P4, O1, O2, T3, T4)

STEP 4: Display the results

* [Display the mean]  
      OUTPUT mean
* [Display the median]  
      OUTPUT median
* [Display the minima]  
      OUTPUT minima
* [Display the maxima]  
      OUTPUT maxima

STEP 5: Go to STEP 3 till the end of the column

STEP 6: EXIT

**2. Chisq()**

This algorithm is used to compute x-squared, degree of freedom and p-value for the observed data.

STEP 1: START

STEP 2: [Read the values of electrodes]  
    INPUT FP2, P4, T4

STEP 3: [Calculate x-squared, degree of freedom and p-value]

*χ*2  , df, p-value

STEP 4: [Display the results]

OUTPUT x-squared, degree of freedom and p-value

STEP 5: EXIT

**3. Sort\_data()**

This algorithm is used to sort the data according to the subject and categorize it into either positive or negative values.

STEP 1: START

STEP 2: [Read the value of one of the electrodes]  
    INPUT FP1, FP2, F3, F4, C3, C4, P3, P4, O1, O2, T3 or T4

STEP 3: [Sort the data according to the subject]

Sort subject-wise

STEP 4: [Categorize the sorted data into either positive or negative value]

Categorize the sorted data

STEP 5: [Display the sorted data]

OUTPUT the negative or positive value for one of the subjects

STEP 5: Go to STEP 3 till the sorted data is obtained

STEP 6: EXIT

**8.3. HOPe: Implementation**

**1. Rprof()**

This is used to enable and disable the profiling of execution of R expressions.

Step 1: Start

Step 2: [Takes options from the user through its parameters]

It takes as parameters as the name of to file to be created, the memory.profiling and line.profiling enable/disable options.

Step 3: [Takes a function]

Input is main\_func()

Step 4: [Outputs the profiling information]

Outputs the profiling information of its input into a file specified in its parameter.

Step 5: Stop.

**2. summaryRprof()**

It is used to summarize the output of Rprof() to show the amount of time used by the R function.

Step 1: Start

Step 2: [Takes options from the user through its parameters]

It takes as parameters the name of the file created by Rprof() and “lines” to indicate whether the output should be shown or not.

Step 3: [Displays the results]

It displays the output of Rprof() which includes by.self, by.total, sample.interval and sampling.time.

Step 4: Stop.

**3. Time\_sys()**

Step 1: Start

Step 2: [Takes a function as its input]

Input is main\_func()

Step 3: [Displays the time taken]

Outputs the amount of time taken in seconds to execute main\_func() in terms of user system and elapsed time into a csv file called yes.csv

Step 4: Stop.

**9. Results**

**9.1. PC-RoDx : Results**

**9.1.1. Program Output**

[1] " Choose one SUBJ from the following :

1: CBA 2: CLM 3: EGA 4: FSA 5: GMI 6: GRO 7: HTH 8: MBA 9: MMA 10: MTA

11: PLA 12: SCE 13: SPH 14: WPA”

Enter the name of the subject in capital:- CBA

Choose 1 for series data and 2 for random data -> 1

Call:

princomp(x = x, cor = TRUE, scores = TRUE)

Standard deviations:

Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6

2.37011237 1.90360426 1.53477619 0.50538313 0.24920890 0.18827085

Comp.7 Comp.8 Comp.9 Comp.10 Comp.11 Comp.12

0.13870265 0.11839742 0.09105778 0.07773800 0.04013444 0.03398931

12 variables and 100 observations.

[1] "The eigen values according to Kaiser's criterion are:"

Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6

5.617432641 3.623709175 2.355537951 0.255412112 0.062105074 0.035445912

Comp.7 Comp.8 Comp.9 Comp.10 Comp.11 Comp.12

0.019238424 0.014017949 0.008291519 0.006043196 0.001610773 0.001155273

[1] "The component to be used are :"

Comp.1 Comp.2 Comp.3

Enter the name of the subject in capital:- CBA

Choose 1 for series data and 2 for random data -> 2

Call:

princomp(x = x, cor = TRUE, scores = TRUE)

Standard deviations:

Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7

1.8504992 1.5195394 1.3472885 1.2362911 0.9574071 0.8306295 0.6322942

Comp.8 Comp.9 Comp.10 Comp.11 Comp.12

0.5947774 0.5096736 0.3938435 0.3320542 0.1943729

12 variables and 100 observations.

[1] "The eigen values according to Kaiser's criterion are:"

Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6

3.42434746 2.30900005 1.81518622 1.52841567 0.91662830 0.68994543

Comp.7 Comp.8 Comp.9 Comp.10 Comp.11 Comp.12

0.39979602 0.35376019 0.25976719 0.15511267 0.11025997 0.03778084

[1] "The component to be used are :"

Comp.1 Comp.2 Comp.3 Comp.4

**9.1.2. Explanation of Results**

The PC-RoDx experiment may be carried out on a larger scale, with multiple trials, for longer time periods to measure the stability and validity of the predicted results.

The result of analysis of EEG data of the 1st subject is displayed in the figure. The first line asks the user to select the subject from the list of subject names. As you select the first subject, another line prompts the user to enter if he/she wants to choose series data or a random data. According to the option chosen by the user, the program computes the Eigen values. Eigen values are is square of standard deviation.

According to Kaiser’s criterion, only those principal components are chosen which has Eigen Values greater than 1. In the case of the subject CBA series data, only first 3 components are chosen that corresponds to the concentration of variability in the dataset. Similarly in the case of subject CBA random data, only first 4 components are chosen that corresponds to the concentration of variability in the dataset.

After this, the loadings also called Principal Components are extracted in the form of a CSV file. This is given as an input to the next model, DEAn for further analysis.

**9.1.3 Graphs**

**9.1.3.1 Scatter Plot**

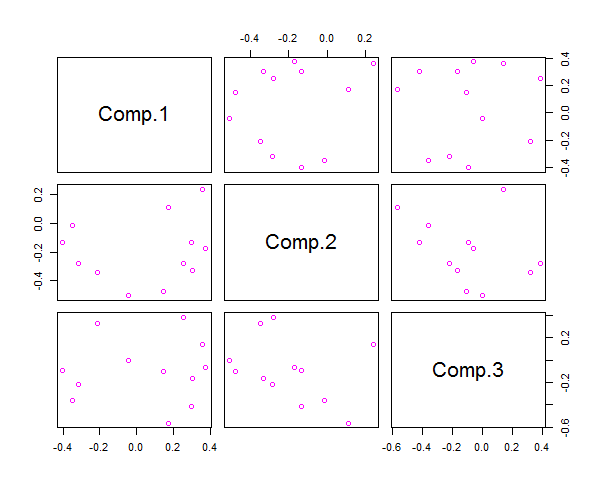
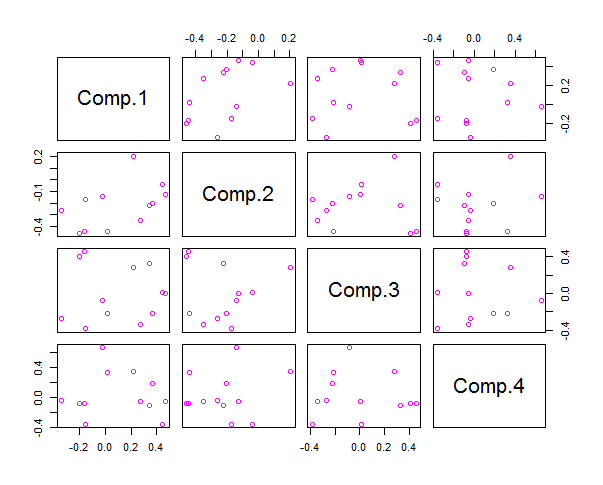


Figure 18: CBA Series data Figure 19: CBA Random data

**Explanation of Scatter Plot:-**

Scatterplots are useful for interpreting trends in statistical data. Each observation (or point) in a scatterplot has two coordinates;

* The first corresponds to the first piece of data in the pair (that’s the*X*coordinate; the amount that you go left or right).
* The second coordinate corresponds to the second piece of data in the pair (that’s the *Y*-coordinate; the amount that you go up or down).

The point representing that observation is placed at the intersection of the two coordinates.

Three things can happen while analyzing a Scatter Plot.

* If the data show an uphill pattern as you move from left to right, this indicates a *positive relationship between X and Y.* As the*X-*values increase (move right), the *Y*-values tend to increase (move up).
* If the data show a downhill pattern as you move from left to right, this indicates a *negative relationship between X and Y.*As the *X*-valuesincrease (move right) the *Y*-values tend to decrease (move down).
* If the data don’t seem to resemble any kind of pattern (even a vague one), then no relationship exists between*X*and *Y*.

It is observed in this graph that there is visible pattern in the graphs. Hence there is no relationship between X and Y. Therefore, we can say that the relationship or correlation between the variables is completely removed. Also it is observed that the principal components obtained are not dependent on each other.

**9.1.3.2 Biplot**

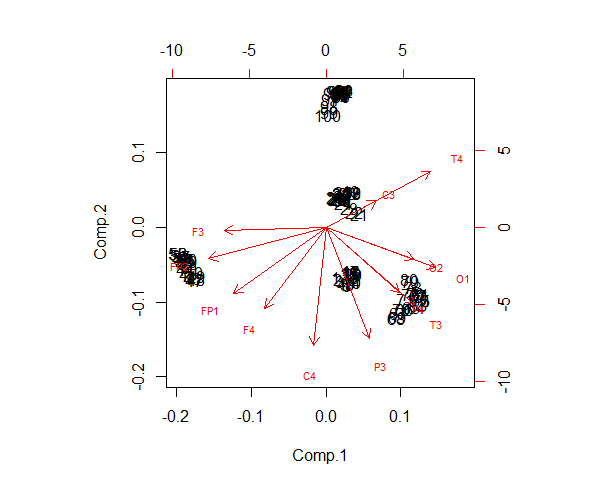
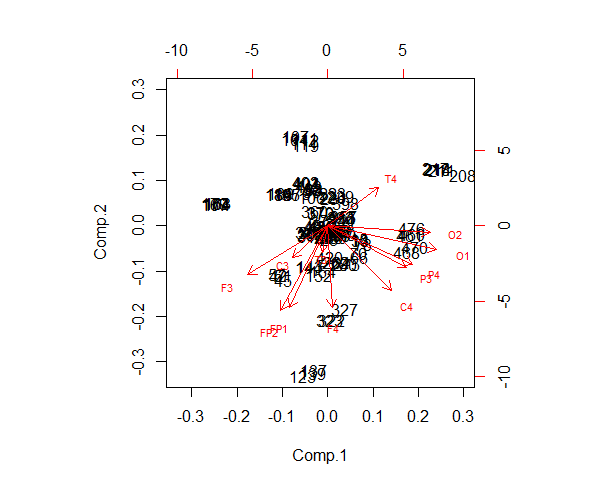


Figure 20: CBA Series data Figure 21: CBA Series data

**Explanation of Biplot:-**

A Biplot is an enhanced scatterplot that uses both points and vectors to represent structure.

As used in Principal Component Analysis, the axes of a biplot are a pair of principal components. These axes are drawn in black and are labled PC1, PC2, etc.

A biplot uses points to represent the scores of the observations on the principal components, and it uses vectors to represent the coefficients of the variables on the principal components.

**Interpreting Points:** The relative location of the points can be interpreted. Points that are close together correspond to observations that have similar scores on the components displayed in the plot. To the extent that these components fit the data well, the points also correspond to observations that have similar values on the variables.

**Interpreting Vectors:** Both the direction and length of the vectors can be interpreted. Vectors point away from the origin in some direction. A vector points in the direction which is most like the variable represented by the vector. This is the direction which has the highest squared multiple correlation with the principal components. The length of the vector is proportional to the squared multiple correlation between the fitted values for the variable and the variable itself.

The fitted values for a variable are the result of projecting the points in the space orthogonally onto the variable's vector (to do this, you must imagine extending the vector in both directions). The observations whose points project farthest in the direction where vector points are the observations that have the most of whatever the variable measures. Those points that project at the other end have the least. Those projecting in the middle have an average amount.

Thus, vectors that point in the same direction correspond to variables that have similar response profiles, and can be interpreted as having similar meaning in the context set by the data.

**9.1.4. Conclusion**

The basis of this provides an insight about what is Principal Component Analysis and how it can be used in combination with R programming to make up the system called as PC-RoDx. PC-RoDx abbreviates to Principal Component – Reduction of Dimensions. This model provides an efficient way of reducing the dimensionality of a multichannel dataset derived from EEG signal. It is observed that the dataset contains the electrodes which are highly correlated. PC-RoDx it helps to removes correlation among the different channels in the data set. The loadings or rotation, also called as Principal Components are extracted in the forms of CSV file. With the use of the loadings, we plot the biplot graph, which agrees with the output of the proposed system and visualizes the relationship between the 12 sentiment electrodes. The dot plot graph obtained implies that the data is uncorrelated with each other.

**9.2. DEAn: Results**

**9.2.1. Program Output**

**1. Subject- CBA**

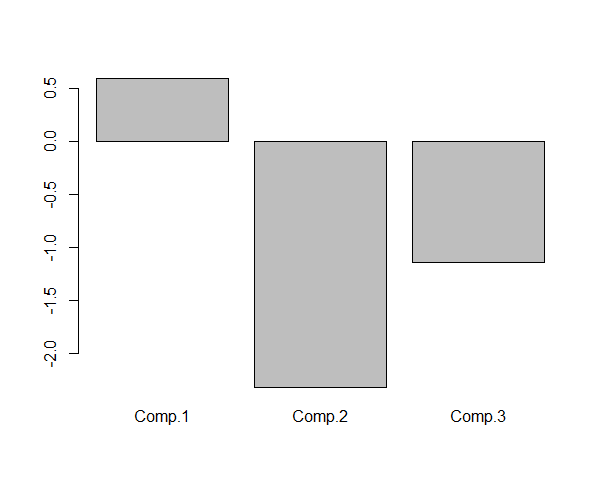
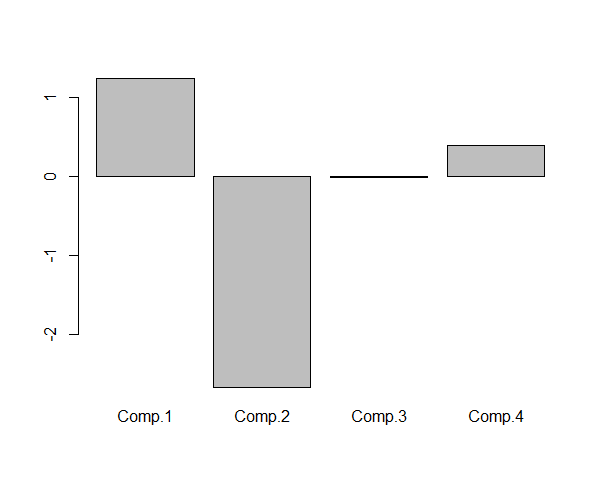
** **

Figure 22: Series graph for CBA Figure 23: Random graph for CBA

**2. Subject- CLM**

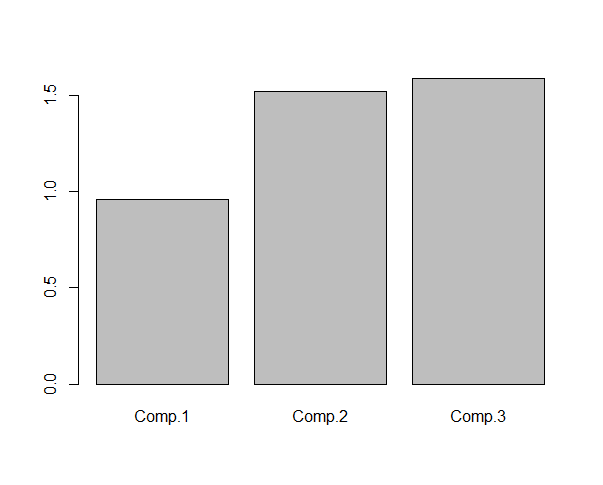
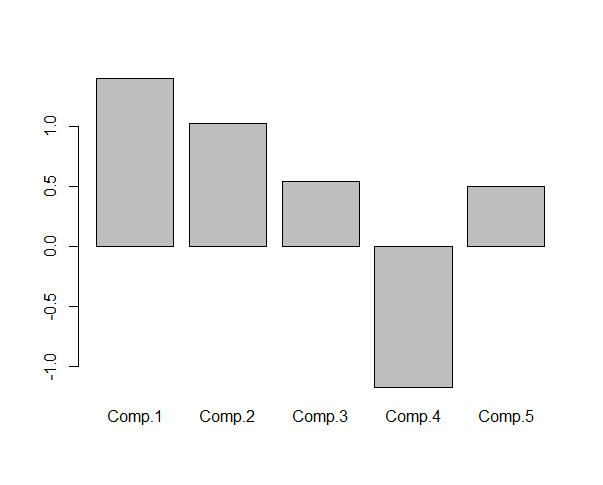
** **

Figure 24: Series graph for CLM Figure 25: Random graph for CLM

**3. Subject- EGA**

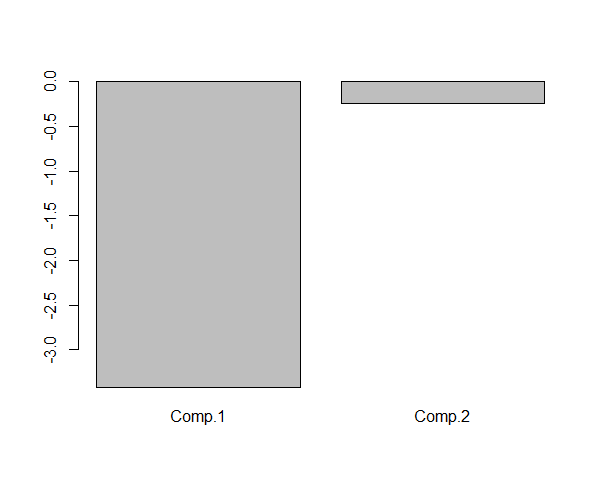
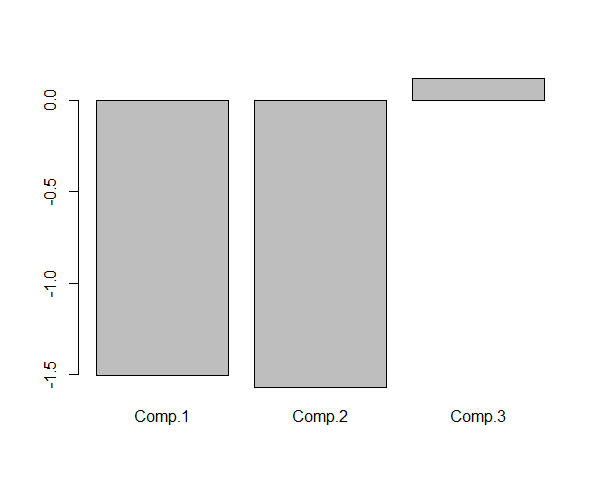
****

Figure 26: Series graph for EGA Figure 27: Random graph for EGA

**4. Subject- GMI**

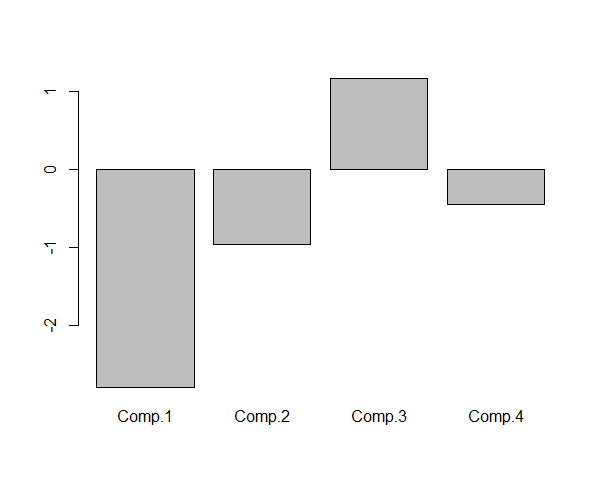
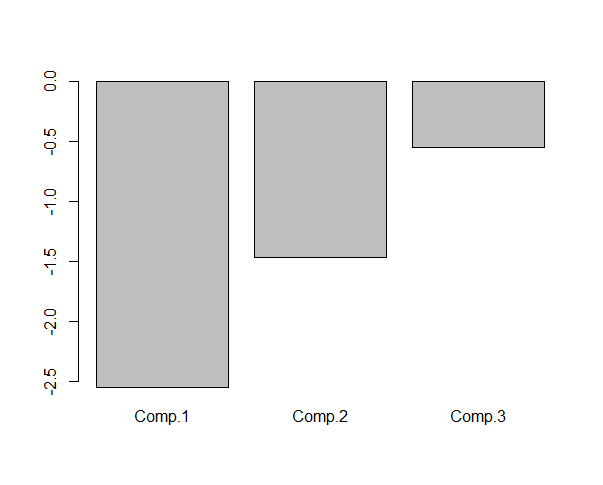
****

Figure 28: Series graph for GMI Figure 29: Random graph for GMI

**5. Subject- GRO**

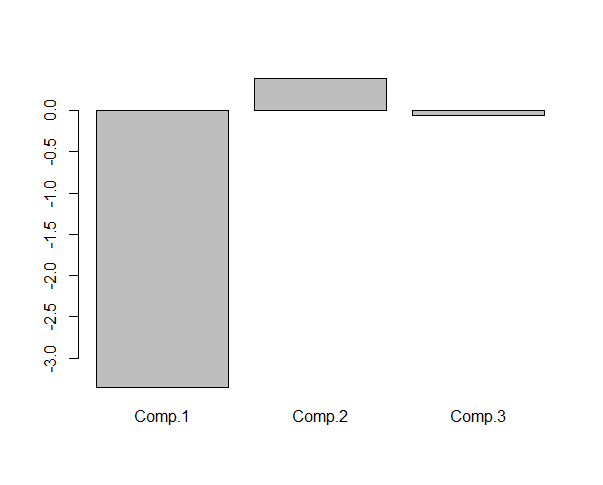
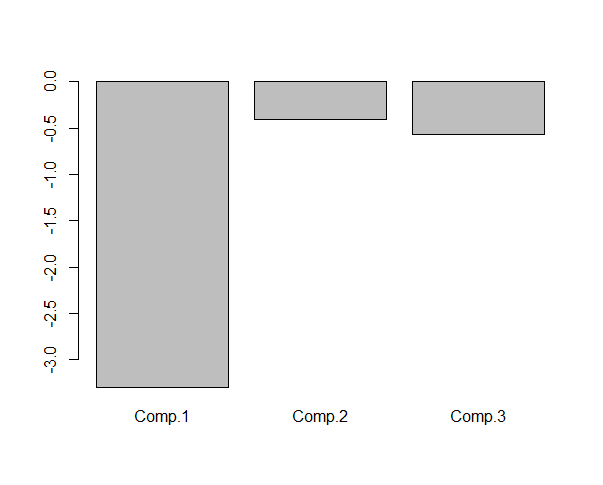
****

Figure 30: Series graph for GRO Figure 31: Random graph for GRO

**6. Subject- HTH**

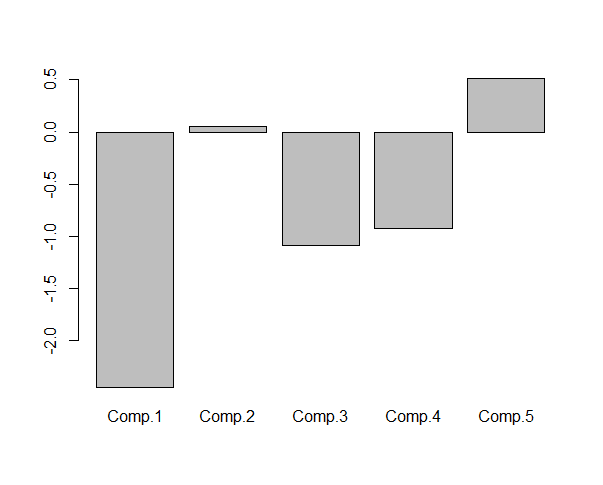
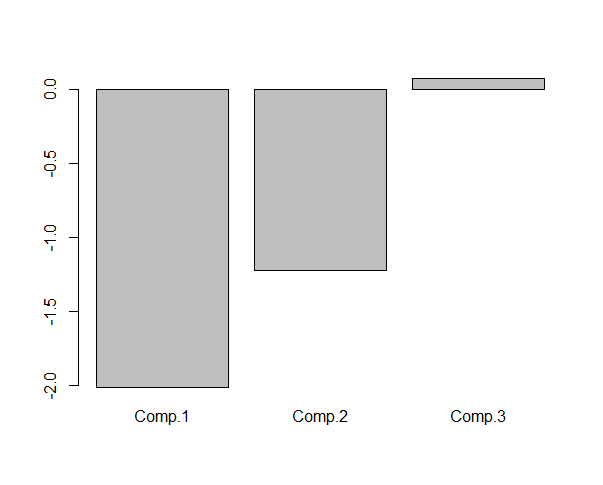
****

Figure 32: Series graph for HTH Figure 33: Random graph for HTH

**9.2.2. Explanation of Results**

The above graphs represent the results obtained for some of the subjects who underwent the experiment. The bar plot helps us in knowing whether the person was happy or sad during the experiment.

The plot above the point 0 shows that the sentiment of the subject is positive whereas the plot below 0 represents a negative sentiment. The results can either be in series or randomly generated. These results may vary as the model PC-RoDx generates random set of principal components every time it is calculated.

**9.2.3. Conclusion**

The proposed system DEAn(Detection of Emotion and its Analysis) uses an analysis technique known as Sentiment analysis. Mathematical operations on the dataset were performed to obtain the polarity of the electrodes. In order to satisfy the theoretical aspects of the electrodes, a chi square test was performed. Finally the data was sorted subject-wise and then categorized into positive or negative values. The sum of the components was calculated for all the electrodes and the result was plotted using a bar plot. The plot above the point 0 corresponded to the sentiment of the subject being positive which indicated that he/she was happy with the experiment whereas the plot below 0 represented a negative sentiment which shows the subject was unhappy with the experiment.

**9.3. HOPe : Results**

**9.3.1. Program Output**

1. Time\_sys()

user system elapsed

* 1. 0.16 10.06

1. Profiling of the code

$by.self

self.time self.pct total.time total.pct

#29 2.34 51.54 2.34 51.54

#95 1.82 40.09 1.82 40.09

#3 0.28 6.17 0.28 6.17

#128 0.04 0.88 0.04 0.88

#8 0.04 0.88 0.04 0.88

#130 0.02 0.44 0.02 0.44

$by.total

total.time total.pct self.time self.pct

#4 4.54 100.00 0.00 0.00

#29 2.34 51.54 2.34 51.54

#95 1.82 40.09 1.82 40.09

#3 0.28 6.17 0.28 6.17

#128 0.04 0.88 0.04 0.88

#8 0.04 0.88 0.04 0.88

#139 0.04 0.88 0.00 0.00

#130 0.02 0.44 0.02 0.44

$by.line

self.time self.pct total.time total.pct

#3 0.28 6.17 0.28 6.17

#4 0.00 0.00 4.54 100.00

#8 0.04 0.88 0.04 0.88

#29 2.34 51.54 2.34 51.54

#95 1.82 40.09 1.82 40.09

#128 0.04 0.88 0.04 0.88

#130 0.02 0.44 0.02 0.44

#139 0.00 0.00 0.04 0.88

$sample.interval

[1] 0.02

$sampling.time

[1] 4.54

**9.3.2. Explanation of Results**

**1. Time\_sys()**

The output of this function consists of three different parameters, namely-user, system and elapsed. The first two entries are the total user and system CPU times of the current R process and any child processes on which it has waited, and the third entry is the real elapsed time since the process was started. In other words, user CPU time gives the CPU time spent by the current process (i.e., the current R session) and system CPU time gives the CPU time spent by the kernel (the operating system) on behalf of the current process.

**2. Profiling of R code**

The Rprof() is a sampling based profiling mechanism that records information about calls on the stack at specified time intervals. It results in a set of profiling information. The summaryRprof() is used to summarize the profiling information. Its output consists of 5 parameters.

1. by.self: the time spent in function alone.
2. by.total: the time spent in function and callees.
3. by.line: the time spent in that line of code.
4. sample.interval: the sampling interval, by default every 20 msecs.
5. sampling.time: the total time of profiling run.

**9.3.3. Conclusion**

With the help of the EEG data, models were built for the analysis and detection of sentiments using two prominent techniques called the Sentiment Analysis and Principal Component Analysis. These models were then optimized in terms of accuracy and the time and space it consumed by using High Performance Computing (HPC) or simply the HOPe model. The HOPe model included profiling of the R code, and time\_sys to find the time each piece of code takes to execute, which helps us in increasing its efficiency.

**10. Conclusion and Future work**

This system was mainly developed with a motive to identify the sentiment of different individuals undergoing any experiment.

* Emotion assessment can be integrated in human computer interaction systems that will lead to improve these systems by making them get close to human-human interaction. This will enhance the usability of the systems and improve the quality of life of disabled people who find difficulty in using the interfaces provided to healthy people.
* Capturing users’ emotions can be helpful in quantifying customers’ experience. It requires an automated system to detect the customers’ emotions without having them state it explicitly. The problem with predicting customers’ experience is that the current evaluation methods such as relying on customers’ self reports are very subjective. Due to the inaccuracy of self reports, market researchers can use this system which captures the users’ affective states without asking them for their direct opinion.
* Also this system can be helpful for people who suffer from psychological problems to interact and communicate easily with computers and humans by capturing the persons’ emotion and make the system self adapt based on the users’ current emotion.

The system also provides scope for future work. The system can be enhanced to obtain different sentiment apart from classifying them as just positive or negative. They can be added with more features to identify the sentiments such as anger, disgust, sadness or calmness. The proposed system is scalable wherein multiple datasets can be added in future for experimentation. Furthermore, the proposed system is a major step towards making a big difference in the fields of healthcare and education and we hope to achieve this milestone that would be a boon to the society.

**References**