EEG Based Emotional Functional Classification and Analysis for BCI Applications

A PROJECT REPORT

Submitted by

Shantanu Shrivastav (15BCE1160**)**
Raghhuveer J. (15BCE1295**)**
Utkarsh Shukla (15BCE1323**)**

in partial fulfillment for the award of the degree of

Bachelor of Technology
in
Computer Science and Engineering

School of Computing Science and Engineering
Vandalur - Kelambakkam Road, Chennai - 600 127

April - 2019

School of Computing Science and Engineering

DECLARATION

We hereby declare that the project entitled "EEG Based emotional function classification and analysis for BCI applications submitted by me to the School of Computing Science and Engineering, VIT Chennai, 600127 in partial fulfillment of the requirements of the award of the degree of B.Tech CSE is a bona-fide record of the work carried out by me under the supervision of Prof. Ilakiyaselvan N. We further declare that the work reported in this project, has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma of this institute or of any other institute or University.

Place: Chennai
Date:

Shantanu Shrivastav

Raghhuveer J

Utkarsh Shukla

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Computing Science and Engineering

CERTIFICATE

This is to certify that the report entitled “EEG Based emotional function classification and analysis for BCI applications” is prepared and sub-mitted by Shantanu Shrivastav (Reg. No. 15BCE1160), Raghhuveer J (Reg. No. 15BCE1295), and Utkarsh Shukla (Reg. No. 15BCE1323), to VIT Chennai, in partial fulfillment of the requirement for the award of the degree of B.Tech CSE programme is a bona-fide record carried out under my guidance. The project fulfills the requirements as per the regulations of this University and in my opinion meets the necessary standards for submission. The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma and the same is certified.

Prof. Ilakiyaselvan N. Date:	Dr. B Rajesh Kanna Date:

Internal Examiner Name: Date:	Internal Examiner Name: Date:	External Examiner Name: Date:

Acknowledgement

We wish to express our sincere thanks to a number of people without whom we could not have completed the thesis successfully.

We would like to place on record our deep sense of gratitude and thanks to our guide Prof. Ilakiyaselvan, School of Computer Science and Engineering (SCSE), VIT, Chennai campus whose esteemed support and immense guidance encouraged us to complete the project successfully.

We would like to thank our Program Chair Dr.B Rajesh Kanna & Co-Chair, Dr. C Sweetlin Hemalatha, B. Tech. Computer Science and Engineering and Project Coordinator Dr. B V A N S S Prabhakar Rao, VIT Chennai campus, for their valuable support and encouragement to take up and complete the thesis.

Special mention to Dr. A Nayeemulla Khan, of School of Computing Science and Engineering (SCSE), VIT Chennai campus, for spending hisvaluable time and efforts in sharing their knowledge and for helping us in every minute aspect of software engineering.

We thank our management of VIT Chennai campus for permitting us to use the library resources. We also thank all the faculty members and lab assistants for giving us the courage and the strength that we needed to complete our goal. Also we would like to thank all the participants without them we would have not been able to achieve this milestone. This acknowledgment would be incomplete without expressing the whole hearted thanks to our family and friends who motivated us during the course of our work.

Shantanu Shrivastav
Raghhuveer J
Utkarsh Shukla

Executive Summary

EEG Based Emotional Function Classification and analysis (EBEFCA) for BCI applications is a medical endeavour with a twofold objective, first to identify the biomarkers in EEG signals which can then be mapped to different brain locations to identify the involvement or indifference of various parts of the brain to a particular emotion and secondly to create a complete pipeline for identification and classification of EEG signals in to four basic emotions of happy, sad, disgust and calm. The first aspect of EBEFCA is aimed towards getting a deeper understanding of the neural activities and synaptic interactions in brain corresponding to emotions which can then be used to better treatment of disease like Dementia, Clinical depression, Schizophrenia. The second part has a clear application in BCI where it can be used in Consumer Behaviour Analysis (CBA) and Neural Marketing where the aim is to gauge the effectiveness of media on consumers and understand the experience of a consumer on introduction to a new product.

Table of Content

S. No.	Title	Page No.
	Cover Page	1
	Declaration	2
	Certificate	3
	Acknowledgement	4
	Executive Summary	5
	Abstract	12
1	Introduction	13
1.1	Objectives	13
1.2	Background	13
1.3	Motivation	14
2	Project Description and Goals	15
2.1	Project Description	15
2.2	Goals	16
3	Literature Survey	17
3.1	EEG	17
3.2	Machine Learning and BCI	18
3.3	Classifiers	19
3.3.1	Logistic Regression	19
3.3.2	Support Vector Machines (SVMs)	20
3.3.3	Decision Trees	22
3.3.4	Sequential Ensemble Models (Adaboost)	23
3.3.5	Sequential Ensemble Models (Gradient Boost)	24
3.3.6	Parallel Ensemble Models (Random Forest)	24
3.3.7	Parallel Ensemble Models (Extra Trees)	25

3.3.8	EEGNET	26
3.4	Quantitative analysis	27
4	Technical Specification	28
4.1	Hardware Specification	28
4.2	Software Specification	30
5	Design Approach and Details	32
5.1	Design Approach	32
5.2	Use Case Diagram	34
5.2.1	Modular Description	35
5.3	Constraints and Alternatives	35
5.3.1	Design Constraints	35
5.3.2	Components Constraints	36
5.3.3	Budget Constraints	36
6	Project Modules	37
6.1	Data Acquisition	37
6.1.1	Test Subjects	37
6.1.2	Stimuli and Design	37
6.2	Pre-processing Pipeline	38
6.2.1	Channel artifacts	39
6.2.2	Epoch artifacts	40
6.2.3	IC artifacts	41
6.2.4	Single-channel, single-epoch artifacts	43
6.2.5	Contaminated datasets	44
6.3	Feature Extraction	45
6.3.1	Correlation dimension:	46
6.3.2	Power Spectral Intensity	47

6.3.3	Hjorth Parameters	47
6.3.4	Hurst Exponent	48
6.3.5	Discreet Wavelet Transform	48
6.3.6	Standard Deviation	48
6.3.7	Variance	48
6.3.8	Skewness	49
6.3.9	Kurtosis	49
6.3.10	Fast Fourier Transform	49
7	Results	50
8	Conclusion	56
9	Codes and Standards	57
10	Schedules, Tasks and Milestones	60
11	Future Work	61
12	References	62
	APPENDIX 1 -CODE SNIPPETS	64
	APPENDIX 2- LOG BOOK	67

List of Figures

Figure No.	Figure	Page No.
1	Various EEG Bands	17
2	Mapping of EEG in Human Brain	18
3	Example of Logistic classifier	20
4	Example of selecting a hyperplane for splitting in SVM	22
5	Example of decision tree classifier	23
6	Explanation of AdaBoost Classifier	24
7	Explanation of Gradient Boost Classifier	24
8	Explanation of Random Forest Classifier	25
9	Explanation of Extra Tree Classifier	26
10	EEGNET	27
11	Standard Sensor Location	28
12	Top View and Lateral View	30
13	Architecture of the project	32
14	Emotiv Epoc device with 100% contact with human brain	33
15	Usecase Diagram of the modules	34
16	Channel Artifact Pipeline	39
17	Epoch Artifact Pipeline	41
18	IC Artifact Pipeline	43
19	Single-Channel, Single-Epoch Artifact Pipeline	44
20	Contaminated Dataset Correction Pipeline	45
21	Positive vs Negative Emotion KNN	53

22	Area Under Curve for KNN	53
23	Kernel SVM Positive vs Negative Emotion	54
24	AUC for Kernel SVM	54
25	KNN Emotion Classification	55
26	AUC Emotion Classification	55
27	Gantt Chart for the project timeline	60

List of Tables

Table No.	Tables	Page No.
1	Software Specifications	31
2	Modular Description based on Main Usecase diagram	35
3	Component Constraints	36
4	Hardware Components Price	36
5	Software Components Price	36
6	Disgust vs happy vs clam vs sad	51
7	Positive vs Negative	51
8	Emotions VS Baseline	52
9	Codes and Standards for Emotiv Epoc Plus	57

ABSTRACT

Emotions play an essential role in our daily lives, including decision making, perception, learning, rational thinking and actions. Emotions are the key to understand a human being. Hence, emotion classification proves to be a vital step towards aiding peoples. In recent years, study has been done to discover relationship among brain signals such as electroencephalogram (EEG) signals with emotions. Like various mental and physiological states, emotions is related with a variety of feelings, thoughts, and behaviors. In neuroscience and psychology, event related potential (ERP) is popularly used to research the brain rapid processing of affective emotional stimuli.

Researches are focused on detecting human emotions from physiological signals display, emotion recognition through EEG has a wide practical application. Chief among these are the use in medicine and scientific research, and the field of affective computing. Incorporation of emotions in human-computer interaction giving machines a degree of emotional intelligence.

Proposed use of machine learning systems include multimedia environments that recognize the emotions of the users, such as recommendation and tagging systems, films and games that respond to the user's emotions, and biofeedback devices that can be used to gain control over emotional states. Emotion analysis is a classification problem since the goal of the system is to predict the correct label of emotion. It is thus often a supervised learning task since labels are already assigned to the data by humans

In this endeavor we dive deep into understanding the factors contributing to emotions and identifying the changes in the EEG signals due to introduction of emotional stimulus, the correlation of the different biological features concerned with emotions and creating an Emotion Recognitions Framework.

1. INTRODUCTION

1.1 Objective

One motivation is to provide solutions to the problem of determining emotions in unresponsive patience having mental disabilities. We aim towards extension of the research to a real-time scenario where emotion classification can be directly used in BCI for Consumer Behavior Analysis or in Neural Marketing w here effects of new products or media is measured over a target population who are prospective customers.

The project is broadly classified into three objectives:

- Determining a controlled atmosphere comprised of different tests, audio-visual stimulus and environmental conditions that bring out different emotions within the test subjects in order to extract EEG corresponding to a particular emotion.
- Identifying biomarkers and different attributes of the EEG data that distinctly define the properties and changes in EEG caused by different emotional states. This involves closely analyzing signals from different parts of brain and observing changes in the signals due to change in emotions.
- Creating an emotion classification framework. This is achieved by using different mathematical transforms on recorded EEG data and using signal processing techniques for feature extractions from different time, frequency and energy domains and then employing different Machine Learning and Deep Learning Algorithms for classification the captured signals.

1.2 Background

Brain Computer Interface or BCI is the latest technology being researched upon by scientists all over the world. It is the ongoing uphill battle to create an artificial brain. The human brain is the most complex organ in our body. It has hundreds of tasks to perform like breathing, eating talking, reacting etc. In this work we attempt to understand a part of the brain that recognizes human emotions and reacts via visual stimuli. We will be reading brain waves and co-relate these with the emotions felt by the user to create an interface where the machine reads brain waves and recognizes the emotions. The biggest challenge to this project is data pre-processing and noise removal from the raw data so as to augment the information contained in the EEG recordings.

Current Challenges:

- Noise Removal
- Noise Detection
- Low power signals
- Delicate instruments available for regular use

1.3 Motivation

Our brain is the most important organ of our body and if we are able to understand the language of the brain the possibilities in the future are countless. The motivation behind this project is to try and get a primitive understanding of our brain's working by recognizing emotions felt by a user's brain. Scientists all over the world are now working on a new technology called BCI- Brain Computer Interface which attempts to create a fully functional artificial brain. This project is a part of the ongoing BCI research and will be able to imitate the part of the brain which detects emotions. If created successfully, BCI may be the key to an immortal conscious. This project can also be applied to the departments of civil and military investigations to extract information without having to harm a human being.

2 Project Description and Goals

2.1 Project Description

The project provides emotional classification system for its application in different field of BCI by employing learning gained from analysis data gathered from different test subjects. The project aims at understanding the mapping between the EEG signal generated due to electric interactions between the neurons in the brains with different emotion states. It aims at identifying variations in in the EEG signals when different subjects are introduced to different emotional stimulus, the correlation of the different biological features with concerned emotion and creating an Emotion Recognitions Framework.

a) Data Acquisition

- Determining the environmental conditions in which the EEG data shall be recorded
- Determining the hardware specifications which shall be used for data acquisition

b) Data Cleaning

- **Signal Centric**, cleaning data for any loss of data or introduction of noise due to types of equipment used and other physical factors such as noise due to cellphone signals or loose electrode.
- **Biological Centric**, removing of the data that is noise free but not relevant to the use-case such as data due to heartbeats, eye-blinks etc.

c) Data Analysis and Signal Processing

Different mathematical transforms and signal processing techniques to convert the time series data into an information-rich format, which points us towards the uniqueness of the EEG signal corresponding to different cognitive states.

d) Signal Classification

Different machine learning and deep learning techniques for signal classification and developing a model that understands the EEG signals and the cognitive state to which they correspond.

e) Result's Validation and Comparison

This phase shall include the process of validating the data over standard data and comparing the results from the work that have been done previously. This shall include comparison of the results from different model and ranking of models over the data.

2.2 Goals

- Identifying tests and environment conditions for recording EEG data to bring about specific emotions in subjects
- Identifying hardware specifications like electrodes, number of channels.
- Determining the contamination in the Data Source and creating a pipeline for cleaning the data
- Mapping channels to specific emotions to create a mapping
- Determining the different features that need to be extracted from the clean data
- Developing different models based on features extracted for recognition and classifications

3.Literature Survey

3.1 EEG

An EEG is a test that detects abnormalities in brain waves, or in the electrical activity of a subject's brain. During an EEG, electrodes are in contact with scalp. These are metal disks with thin wires running through the device. Electrical charges are detected by these electrodes that result from the activity of brain cells. The charges are amplified, and they appear as an oscillating graph denoting the varied electric potential. These studies measure electrical activity in brain that result in response to the stimulation of sight, sound, or touch. EEG signals are usually classified by their frequency, amplitude, shape, or electrode position. The EEG bands are (lower than 4Hz), (between 4Hz and 7Hz), (8-15Hz), (16-31Hz), (higher than 31Hz), and (between 8 and 12 Hz). These bands describe various emotional states

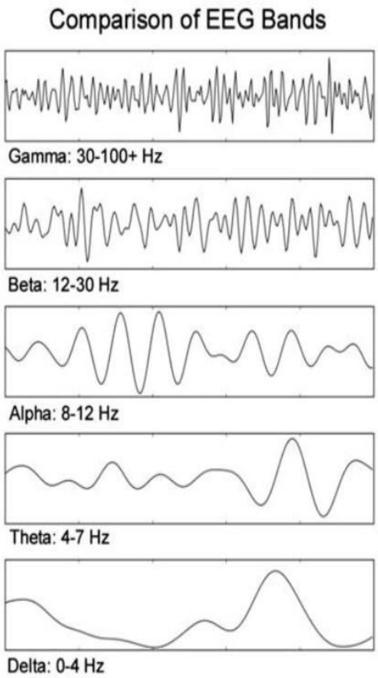


Figure 1: Various EEG Frequency Bands

EEG recently has spread in the research of brain functions associated with emotional and cognitive processes. One of the most commonly used techniques is event-related potentials (ERP) that allows the repeated measurement of ongoing brain activity segments immediately after the presentation of a stimulus. Through averaging the segments it is possible to measure the cerebral voltage formed due to stimuli presented; i.e., by analysis of time amplitude, it is possible to associate components to the stimuli. It is also possible to analyze the oscillations formed due to frequency domain.

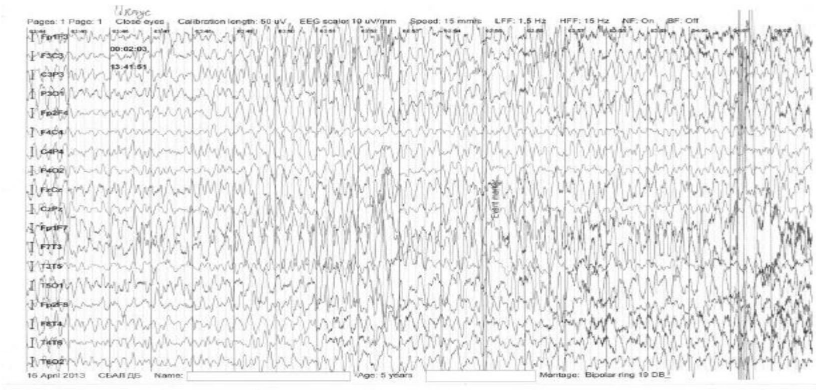


Figure 2: Mapping of EEG in Human Brain

This analysis can be performed in the frequency domain with spectral decomposition represented in power spectral density of each trial through the Fourier transform. However, the time variable with a Fourier transform applied to a series of consecutive time windows or with a discrete wavelet transform analysis can be included. Particular suitability for the investigation of emotions, this study focused on frequency analysis.

3.2 Machine Learning and BCI

Machine Learning is the practice of using algorithms to parse data, learn from it, and then make a determination or prediction about something in the world. Brain–computer interfaces are systems that use signals recorded from the brain to enable communication and control applications for individuals. Brain-computer interface (BCI) is a collaboration between a brain and a device that maps signals from the brain to direct some external activity.

The interface enables a direct communications pathway between the brain and devices that record brain signals which can then be used for controlling different physical objects or analysis of cognition or emotion. Automatic prediction of emotion and classification of the EEG signal has vastly improved primarily due to the development of EEG signal recording technology and exploiting newer machine learning algorithms. The analysis of EEG signals is performed for various research purposes like psychiatric studies, brain-machine interfaces, seizure classification, seizure prediction etc. In this section brief overview of some previous work done in the qualitative analysis of EEG signal data and the corresponding prediction and classification inferences drawn. Brinkmann et al. [1] proposed classification of intracranial EEG data, using Power in Band and inter-electrode synchrony intracranial EEG features. They conclude that optimization of feature selection and best fitting algorithms are subject specific. Luigi et al. [2] proposes a technique for categorization of EEG signals in real time by means of Support Vector Machines, they also give information about feature extraction that requires low computational power which felicitates the use of the algorithm in real time. Lin and Chen et al [3] proposed classification of EEG signals using artefact free signals characterised by 216 global feature descriptors.

A parameter selective computer-aided diagnosis system for EEG classification was proposed by Sood et al. [4] This technique performs the work by identifying the appropriate features from the data for identifying seizures. Tawfik et al. [5] proposed an epileptic seizure detection technique which uses Support Vector Machines. The idea proposed takes into consideration the fact that weighted permutation entropy. Wang and Lyu et al. [6] proposed a new approach where they use elimination-based feature selection method to increase the efficiency of the existing algorithms and diminish the redundant points in the EEG signal. An evolutionary harmony search based algorithm for feature selection on EEG signals was presented by Zainuddin et al. [7]. Zhang et al [8] propose using Linear Support Vector Machine classifier for epileptic seizure prediction wherein spectral power and ratios of spectral power are extracted from intercranial EEG signals and processed by a second-order Kalman filter which are then fed as input to the SVM classifier.

3.3 Classifiers

After testing a range of machine learning algorithms suited for classification of the signal into different emotions, we use three algorithms for this task, in addition to these four ensemble models are explored. The three shortlisted algorithms are listed below:

3.3.1 Logistic Regression

Logistic Regression is primarily used when the response variable is categorical, and we would like to predict the probability of the particular output given our input **x**.

$$\sigma(t) = \frac{1}{1 + e^{-t}} \tag{1}$$

Then this is the logistic function and denotes the success of the **x** and **α** is the intercept while **β** denotes the regression coefficient.

$$T = \alpha + \beta x \tag{2}$$

Adding more independent variables to the logistic regression model will increase the variance explained by the model. However, his can result in overfitting and reduce generalization, hence the regularization terms are added. L1 regularization takes into account the absolute difference of the predicted and computed value as a penalty whereas L2 takes into account the squared difference as penalty. The main aspect where they differ is that L1 reduces the causal effect of the less important features to zero and removes them completely. Using L1 increases the variance which corresponds to overfitting.

$$f(x) = \frac{1}{1 + e^{-(\alpha+\beta x)}} \tag{3}$$

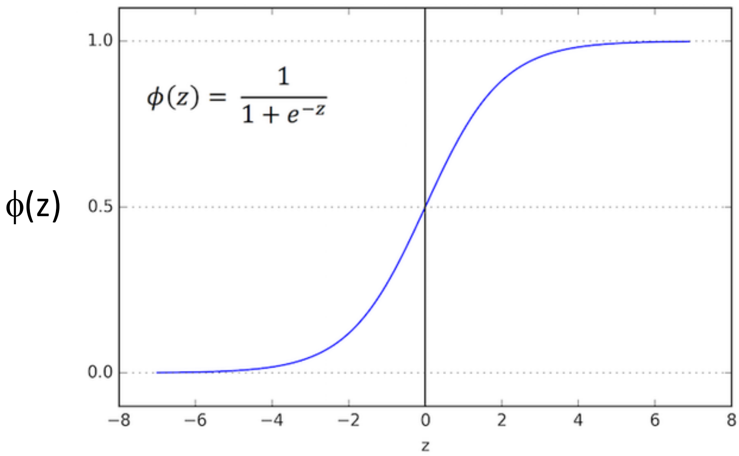


Figure 3: Example of Logistic classifier

3.3.2 Support Vector Machines (SVMs)

SVM determines non-linear class demarcations boundaries by cleverly using linear models, it does so by projecting the points in the input space into a higher using a non-linear mapping. The linear classification model build in the new space serves as a non-linear decision boundary for the input space. Below we explain briefly how SVMs works. Assuming the training data format to be of the type (x_1,y_1) (x_n,y_n) where in each case the value of y_i are either 1 or -1 denoting the class of x_i. x_i represents a vector of size n. The aim is to compute a maximum-margin hyperplane that classifies the vector x_i into two groups of $y_i = 1$ and $y_i = -1$, also making sure that the distance of the nearest point x_i is maximum from the plane. A hyperplane is defined by as set of points x_i satisfying.

$$W.X - b = 0 \tag{4}$$

The bounds p(w,b) of the calculated hyperplane H (w, b) is the distance from the hyperplane to the support vectors, i.e,

$$p(w,b) = min_i \frac{\|\mathbf{w}u_i + b\|}{\|w\|} \tag{5}$$

SVMs are successful when the kernels are used generally referred to as the kernel Trick. In this we define some kernels which enables us to work in higher dimensions (with respect to the input vector) without computing the coordinates of the data in the higher dimension. This approach is generally computationally cheaper than computing the coordinates of the data in the newly defined space. Commonly used kernels are:

SVM task is to find a margin that separates all examples labelled as positive and negative examples. However, this can lead to poorly fit models if any examples are mislabeled or extremely unusual. To account for this, in 1995, Cortes and Vapnik [10]. This lead to better fit. A small χ means a Gaussian with a large variance so the influence of x is more, i.e. if y is a support vector, a small gamma implies the class of this support vector will have influence on deciding the class of the vector x even if the distance between them is large. If χ is large, then variance is small implying the support vector does not have wide-spread influence. Technically speaking, large χ leads to high bias and low variance models, and vice-versa. C is the parameter for the soft margin cost function, which controls the influence of each individual support vector; this process involves trading error penalty for stability. The regularization factor ***C*** is a measure of generalizing the model, it is used to create a tradeoff between bias and variance. For large values of ***C***, the optimization will lead to a smaller-margin hyperplane if that hyperplane does a good job of getting all points classified correctly.

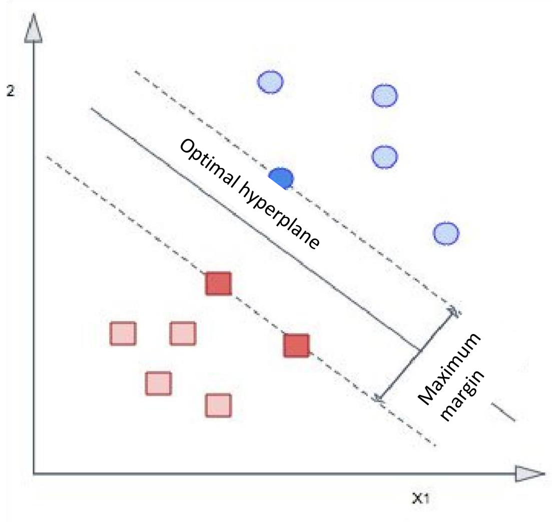
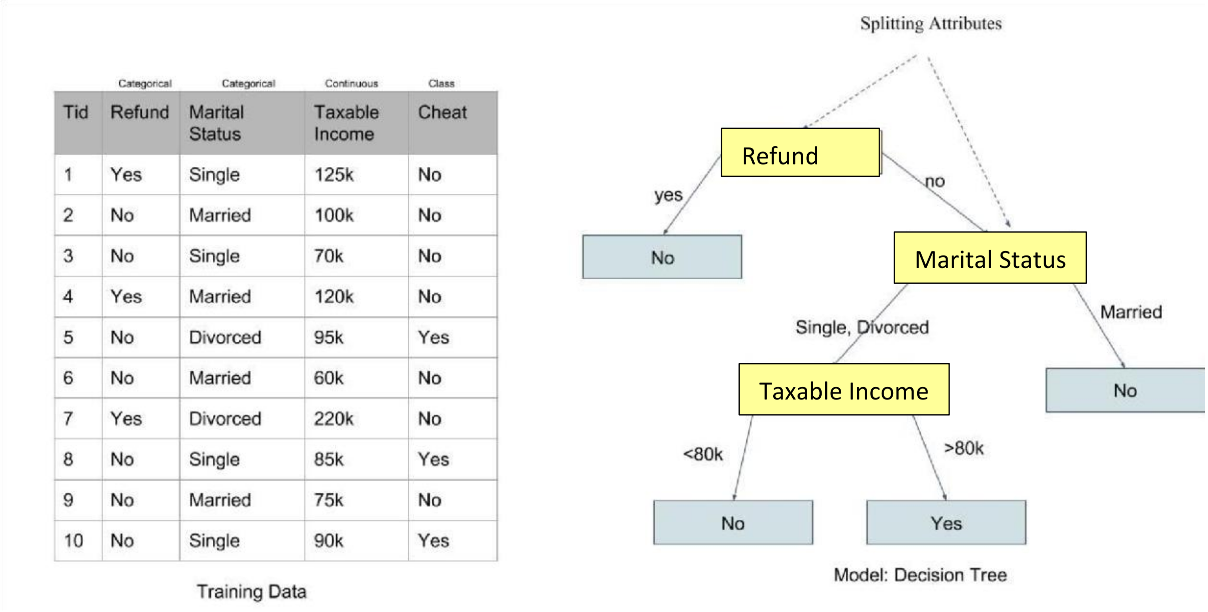


Figure 4: Example of selecting a hyperplane for splitting in SVM

3.3.3 Decision Trees

Decision Tree is a supervised learning algorithm which revolves around the idea of formulating rules and performing decision based splitting based on different attributes to construct a tree structure. The splitting and decision rules learned by the algorithm is based on information gain, defined as the effective change in entropy after a decision rule has been extracted based on an artifact ***a***.

Decision tree is very sensitive to the depth that is chosen. Greater the depth of the tree more it tends to overfit which can be defined as high variance, on the other hand if the tree is too short the results are very generalized, and lot of false positives are encountered.



Tid	Refund (Categorical)	Marital Status (Categorical)	Taxable Income (Continuous)	Cheat (Class)
1	Yes	Single	125k	No
2	No	Married	100k	No
3	No	Single	70k	No
4	Yes	Married	120k	No
5	No	Divorced	95k	Yes
6	No	Married	60k	No
7	Yes	Divorced	220k	No
8	No	Single	85k	Yes
9	No	Married	75k	No
10	No	Single	90k	Yes

Figure 5: Example of decision tree classifier

3.3.4 Sequential Ensemble Models (Adaboost)

AdaBoost is an ensemble supervised learning algorithm that combines a set of weak classifiers to boost their performance. Each sample in the dataset is weighted and the weak learner is trained on a subset of the data. The misclassified samples of the weak learner is assigned higher weights. This enhances the probability of the sample being used in the training of the next classifier. The algorithms aims to creating a classifier that focuses on examples misclassified in the previous steps. Each classifier is also assigned a weight which depends on the accuracy achieved. The final equation for classification can be represented as where f_m stands for the weak classifier and is the corresponding weight. It is exactly the weighted combination of weak classifiers. Below we describe briefly how an ensemble model works For a dataset of points where and in where denotes the two classes, initial weights of the points are set as. Each of the classifiers are trained on the dataset, we select the one with the lowest weighted classification error.

The modified weights depend on the error rate of the classifiers Σm which is defined as the ratio of the number of misclassification over the training set size.

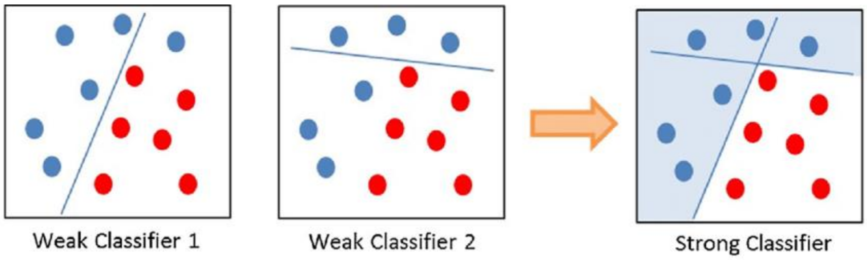


Figure 6: Explanation of AdaBoost Classifier

3.3.5 Sequential Ensemble Models (Gradient Boost)

Gradient boosting algorithm is a supervised model which consists of an ensemble of weak prediction classifiers predominantly decision trees. The classifiers are trained sequentially over the training set where the weak models learn from the misclassification of the previous models. Each model contributes towards reducing the loss function and minimizing the error rate to provide a more accurate estimate of the response variable. The ensemble model due to the use of the boosting exhibits high bias and low variance.

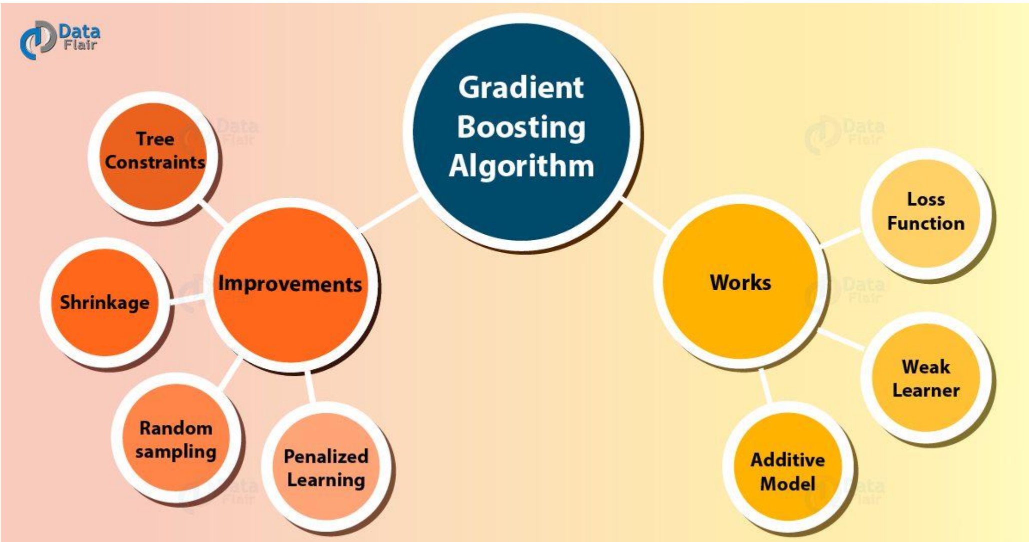


Figure 7: Explanation of Gradient Boost Classifier

3.3.6 Parallel Ensemble Models (Random Forest)

Random forest is a supervised learning algorithm. Random Forests consist of multiple decision trees initialized with different hyper-parameters ensemble using bagging technique. Bagging is a mechanism in which the predictions from multiple base models are used together for training. The main advantage of Random forest over the decision trees is its ability to prevent overfitting as it randomizes the feature subset and builds smaller trees for classification as opposed to a single deep tree. The prediction from the model is achieved by a voting process wherein votes are received from each tree. The important hyper-parameters are the number of trees that the random forest will use for the final prediction and the number of features in the subset that each tree uses.

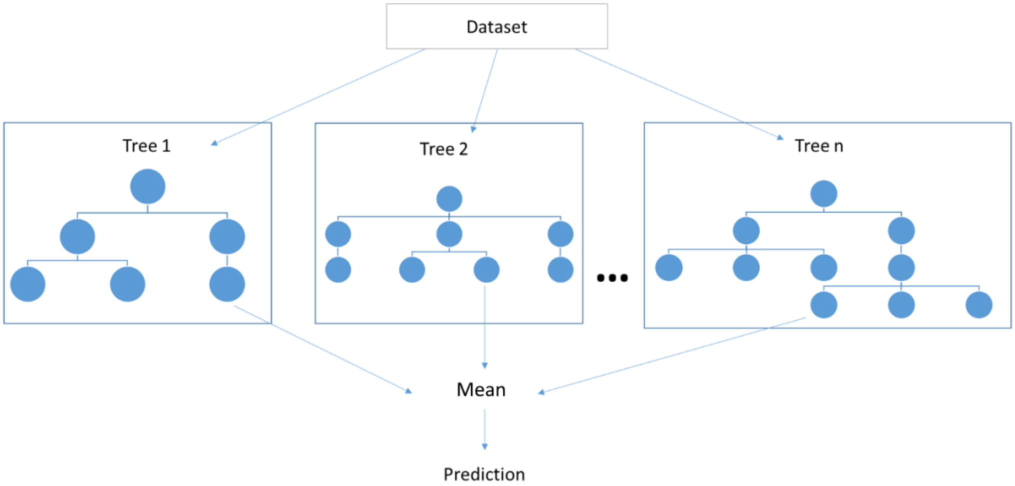


Figure 8: Explanation of Random Forest Classifier

3.3.7 Parallel Ensemble Models (Extra Trees)

Extra Trees is a supervised learning algorithm and is a modification of Random forest. In Extra Trees, the process of feature selection is completely randomized for different trees as opposed to the random forest where feature selection is based on specific rules, the splitting threshold of the nodes in extra tress is also randomized whereas it is fixed in a random forest. The reason that extra trees perform better than the Random forest in some cases is that it makes the decision boundaries smooth and does not use the bagging mechanism which is computationally expensive when the volume of the data is considerably high. Extra Trees is more generalized and tolerant against overfitting since the hyper-parameters of each tree is different and hence the performance and prediction of each tree has minimal correlation.

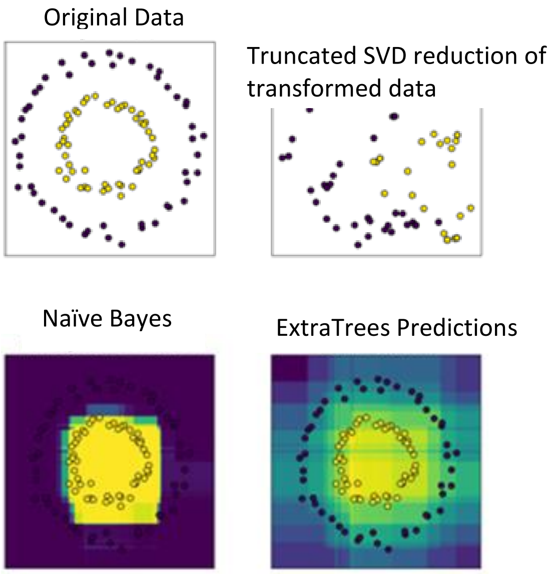


Figure 9: Explanation of Extra Tree Classifier

3.3.8 EEGNET

EEG Net is a CNN architecture used for classification of EEG signals obtained from different BCI paradigms, while simultaneously being as compact as possible, it is convolutional network for EEG-based BCIs. A Depth wise and separable convolution are used to construct an EEG-specific model which encapsulates well-known EEG feature extraction concepts for BCI. EEG Net generalises across paradigms better than the reference algorithms when only limited training data is available. EEG Net is robust enough to learn a wide variety of interpretable features over a range of BCI tasks, suggesting that the observed performances were not due to artefact or noise sources in the data.

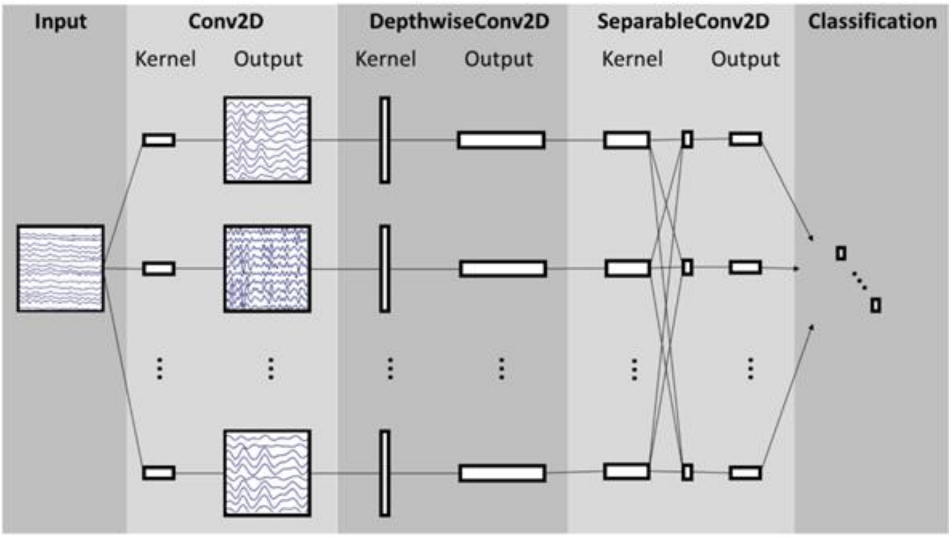


Figure 10: EEG NET

3.4 Quantitative analysis

Many different metrics can be used for the quantitative analysis of the results obtained by the classifier. for understanding these metrics, it is required to understand a few concepts related to them. Let us take an example of an emotion being classified as a Happy or not.

- True positive and in accordance with the above example, it would mean all the emotions actually classified as Happy.
- True negative which means the number of emotions that are not Happy actually classified as not Happy.
- False positive which would mean the number of not Happy classified as Happy.
- False negative which would mean the number of Happy classified as not Happy.

Some of the more popular metrics used are:

- Sensitivity

$$\text{Sensitivity} = [TP / (TP + FN)] \times 100 \tag{6}$$

- Recall or Sensitivity

$$\text{Recall} = TP / (TP + FN) \tag{7}$$

- Specificity

$$\text{Specificity} = TN / (TN + FP) \times 100 \tag{8}$$

- Accuracy

$$\text{Accuracy} = (TP + TN) / (TP + FP + TN + FN) \tag{9}$$

- Precision

$$\text{Precision} = TP/\ (TP + FP) \tag{10}$$

4.Technical Specification

4.1 Hardware Specification

4.1.1 Emotiv Epoc+

It is a 14-channel mobile EEG is designed for scalable and contextual human brain research and advanced brain-computer interface applications and provides access to professional grade brain data with a quick and easy to use design. It provides access to high-quality raw EEG data and helps in leveraging detections for mental commands, performance metrics or facial expressions.

- **EEG sensors**
  14 channels: AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4
  2 references: CMS/DRL references at P3/P4; left/right mastoid process alternative
  Sensor material: Saline soaked felt pads
- **Connectivity**
  Wireless: Bluetooth Low Energy
  Proprietary USB receiver: 2.4GHz band
  USB: to change headset settings

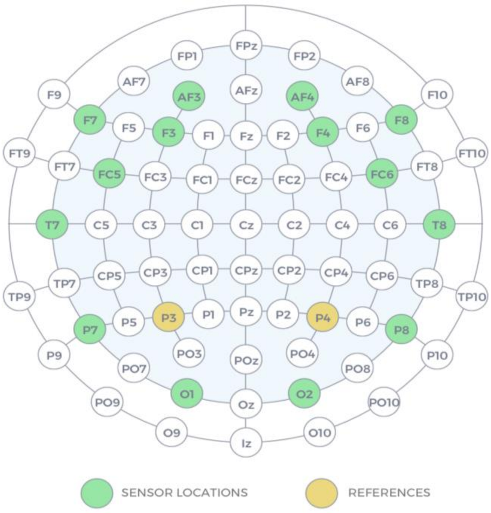


Figure 11: Standard Sensor Location

- **Motion sensors**

  IMU part: ICM-20948

  Quaternions: normalized, 4D

  Accelerometer: 3-axis +/-4g

  Magnetometer: 3-axis +/-4900uT

  Sampling rate: 0 / 32 / 64 Hz (user configured)

- **EEG signals**

  Sampling method: Sequential sampling, single ADC

  Sampling rate: 2048 internal down sampled to 128 SPS or 256 SPS (user configured)

  Resolution: 14 bits with 1 LSB = 0.51μV (16 bit ADC, 2 bits instrumental noise floor discarded), or 16 bits (user configured)

  Bandwidth: 0.16 – 43Hz, digital notch filters at 50Hz and 60Hz

  Filtering: Built in digital 5th order Sinc filter

  Dynamic range (input referred): 8400 μV(pp)

  Coupling mode: AC coupled

  Supported platforms

  Windows: 7,8,10; 8GB RAM; 500MB available disk space

  MAC: OS X; 8GB RAM; 500MB available disk space

  iOS: 9 or above; iPhone 5+, iPod Touch 6, iPad 3+, iPad mini

  Android: 4.43+ (excluding 5.0); device with Bluetooth Low Energy

- **Power**

  Battery: Internal Lithium Polymer battery 640mAh

  Battery life: up to 12 hours using USB receiver, up to 6 hours using Bluetooth Low Energy

Figure 12: 1 Top View of EMOTIV EPOCH +, 2 LATERAL VIEW

4.2 Software Specification

S. No	Software	Version	UseCase
1	Python	3.6	Primary scripting for developing different pipelines
2	Matlab	2018	EEG Data Analysis and Visualisation
3	Emotiv-PRO	2017	EEG Data Acquisition
4	Keras	2.1.6	Scripting Deep Neural Network
5	Tensorflow	1.8.0	Backend for Deep Neural Networks
6	Scikit-Learn	1.3.2	Optimised Implementation of various Machine learning Algorithm

7	Electron JavaScript	NA	UI interaction for Data Recording
8	OpenCV	3.3	Creating Pipeline for Emotion Inducing Pipeline

Table 1 : Software Specifications

5. Design Approach and Details

5.1 Design Approach

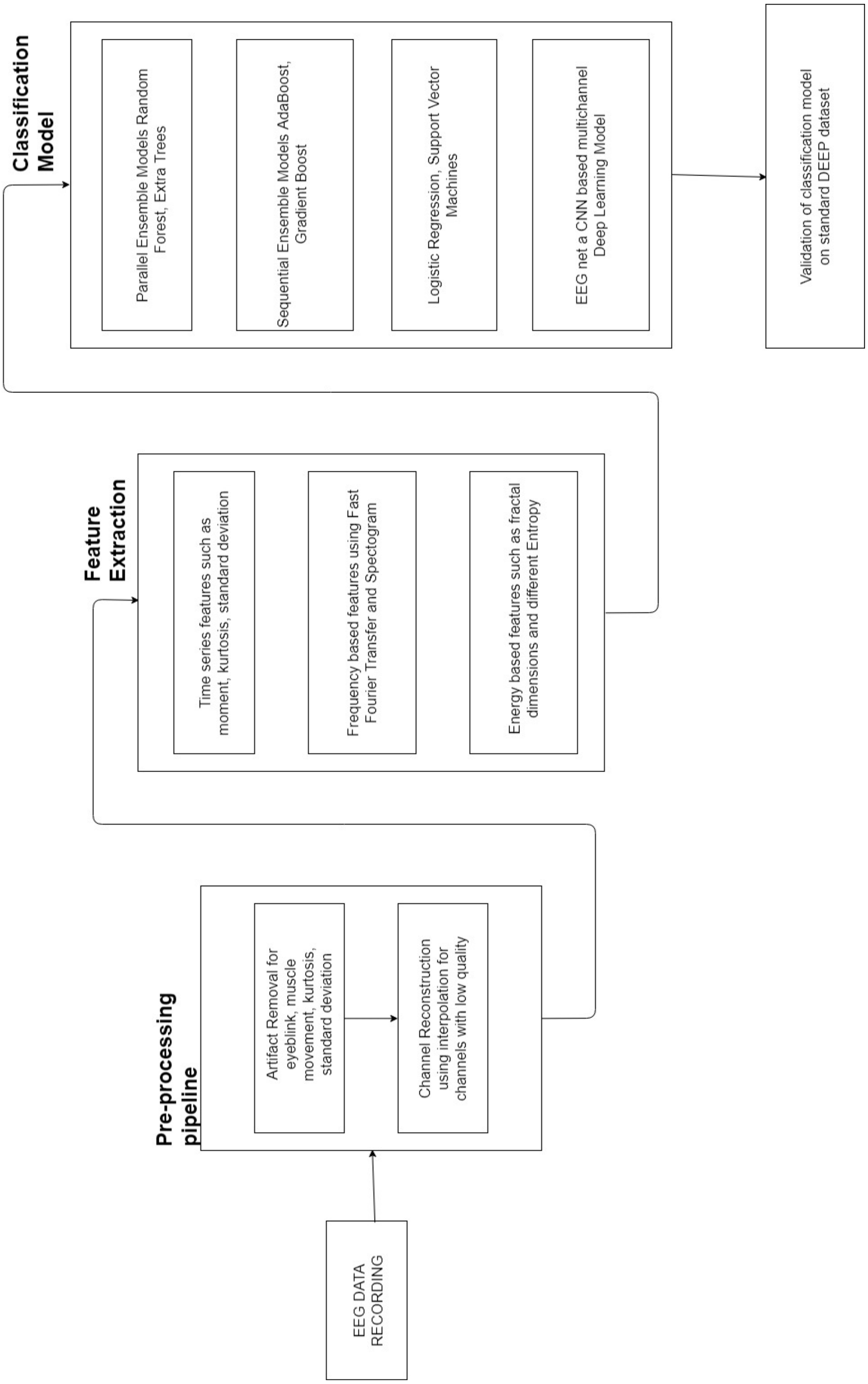


Figure 13: Architecture for project

The project has been divided into 5 modules: EEG data recording, Pre-processing pipeline, Feature Extraction, Classification Model and Benchmarking with DEEP dataset

Using the Emotiv Epoc hardware EEG data is recorded. Process involved in recording of

EEG data includes:

- Dataset - International Affective Picture System (IAPS) is developed and distributed by the NIMH Center for Emotion and Attention (CSEA) at the University of Florida. IAPS intends to provide a set of standardised, emotionally visual stimuli. The IAPS database contains color photos from a different area, and all photos are labeled with the valence, arousal and dominance level ratings which were assessed by a large number of subjects.
- In the process of finding the artefacts for different emotion we first take two baseline recordings for two minutes. First baseline recording corresponds to subject closing eyes for two minutes sitting in a relaxed state and the second baseline recording corresponds to subject with minimum blinking sitting a relaxed state.
- A pipeline has been setup that shows images corresponding to different emotions it also logs the time keeping track of the image id the emotion and time it was shown for.

The images are shown under 4 categories of happy, disgust, calm, sad. The contrasting emotions are shown so that the effect of the emotions developed are maximum. Each image is shown for 12 secs where each emotion has 20 pictures which is enough to gather data for the corresponding emotions.

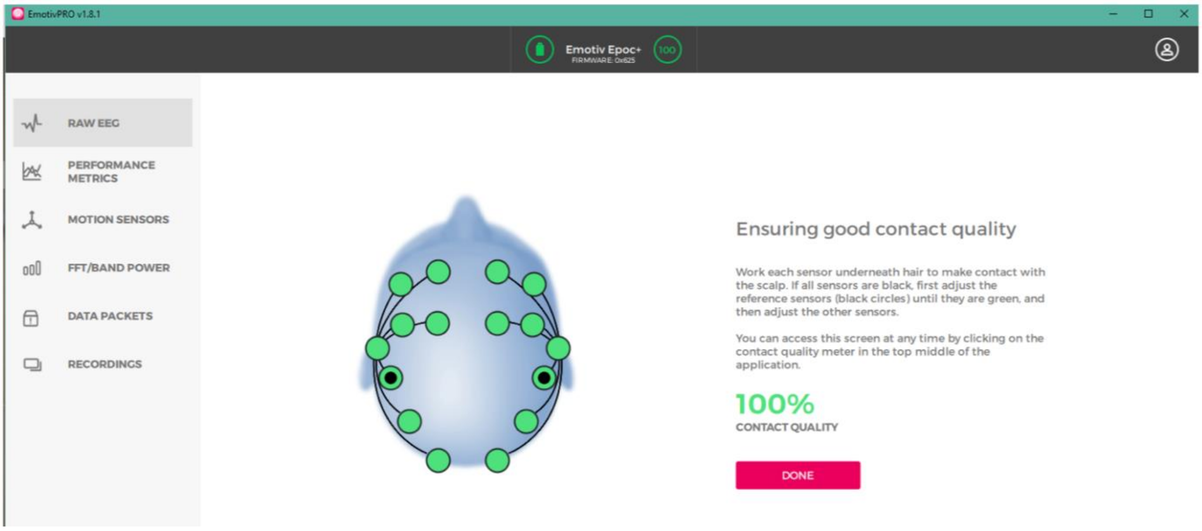


Figure 14: Emotiv Epoc device with 100% contact with human brain

Change in stimulus is referred to as artifact and this need to be removed from the data recording. Hence Pre-processing pipeline is used.

5.2 Use Case Diagram

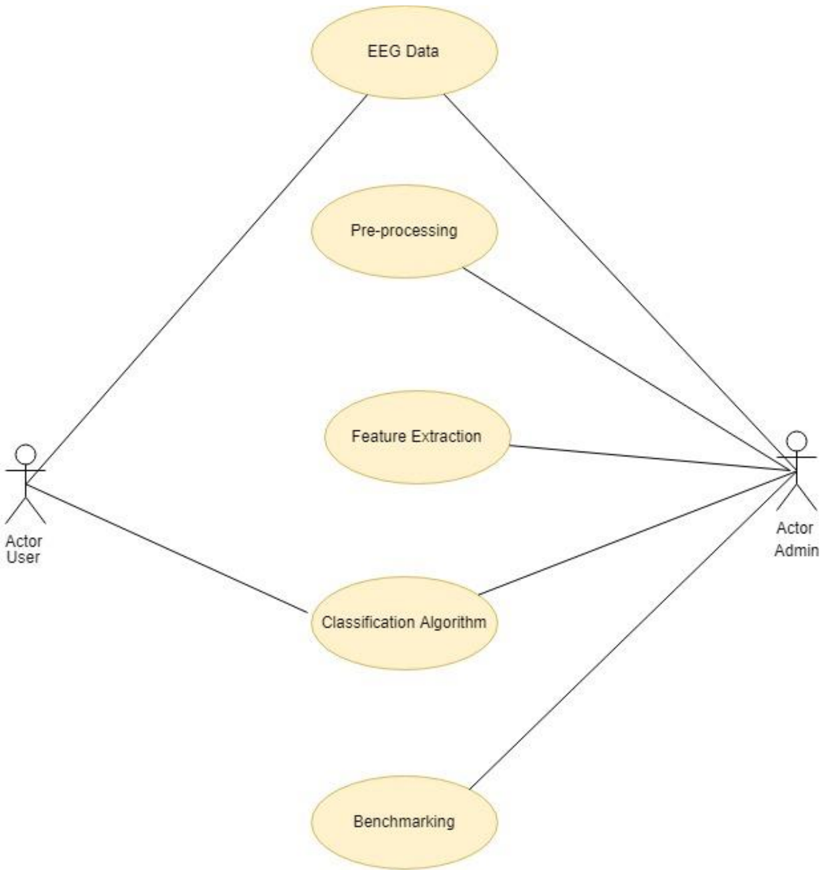


Figure 15: Usecase Diagram of project modules

5.2.1 Modular Description

EEG Data	User provide his EEG using Emotiv Epoc device
Pre-processing	Using ICA and Cubic Interpolation Algorithm admin removes artifacts from input EEG of user
Feature Extraction	Using various transforms Time Based, Frequency Based and Energy Based features are extracted from EEG after artifact removal
Classification Algorithm	Using Machine Learning Algorithms classification of Emotion among Happy, Sad, Disgust and Calm is done
Benchmarking	Using DEEP dataset results obtained from input EEG are benchmarked

Table 2: Modular Description based on Usecase diagram

5.3. Constraints and alternatives

5.3.1 Design Constraints

The system has been built keeping the following design constraints in mind:

- The subject's EEG information has to be kept confidential.
- The most optimal technologies have to be used for signal analysis and models.
- The availability of connection to the internet.
- The differences in the needs of different classification algorithms.
- Optimal use of existing resources and minimal additional resources and expenditure for the users involved.

5.3.2 Components Constraints

Open BCI could have been used in place of Emotive. The advantages and disadvantage of using this Emotive are mentioned below.

Name	Advantages	Disadvantages
Emotive	The signal quality is upto medical standards	Since the electrodes are gel based wet electrodes they need tone cleaned after every 3 hrs
	Simple API and library wrappers for accessing raw egg data	Model with 32 electrodes is considerably costly and can't be used due to budget constraints

Table 3: Component Constraints

5.3.3 Budget Constraints

- **Hardware Component**

Component	Price
Emotiv Headset	799$ = Rs 55,157 (FUNDED)
Import/Custom Duty and Shipping	Rs 15,000 (approx.) (FUNDED)
Total	Rs 70,157 (FUNDED)

Table 4: Hardware Components Price

- **Software Component**

Component	Price
Emotiv Epoc Plus	89$ /monthly = Rs 6144 (FUNDED) (For 4 months)
License (Number of Devices as 3)	44$ (FUNDED)
Total	400$ = Rs 27,616 (FUNDED)

Table 5: Software Components Price

6. Project Modules

6.1 Data Acquisition

6.1.1 Test Subjects

A total of 25 samples has been collected from our EEG signal acquisition experiment and used for the study reported in this report. The test subjects were selected to have the following characteristics;

1) males between 19-23 years old
2) no personal history of neurological or psychiatric illness

6.1.2 Stimuli and Design

Each subject is first introduced to the process of recording data and a NDA is signed In this current study, an international standardized emotion induction tool IAPS was used to induce sustained, reliable emotions. Then 72 colored pictures were finally selected from the IAPS database 18 images per emotions and since the focus is on 4 emotions hence 72 images. Each picture was presented on a 13 Inch computer screen which was for 10sec in front of the viewer.

The experiment was divided into 6 blocks where different aspects where captured for the subject. The subjects would have a short break after each block, the blocks were as follows

a) Baseline Eyes Open:

EEG data recorded for x mins where the subject is in a relaxed position with eyes open with minimum blinking and head resting.

b) Baseline Eyes Open:

EEG data recorded for x mins where the subject is in a relaxed position with eyes closed and head resting.

c) Emotion Calm:

EEG data recorded for 3 mins 18 images * 10secs where the subject is in a relaxed position focusing on images displayed corresponding too images bringing out calm emotion. Followed by a 20 sec break

d) Emotion Disgust:

EEG data recorded for 3 mins 18 images * 10secs where the subject is in a relaxed position focusing on images displayed corresponding too images bringing out Disgust emotion. Followed by a 20 sec break

f) Emotion Happy:

EEG data recorded for 3 mins 18 images * 10secs where the subject is in a relaxed position focusing on images displayed corresponding too images bringing out Happy emotion. Followed by a 20 sec break

g) Emotion Sad:

EEG data recorded for 3 mins 18 images * 10secs where the subject is in a relaxed position focusing on images displayed corresponding too images bringing out Sad emotion. Followed by a 20 sec break

The contrasting emotions were kept one after the another to capture considerable change in EEG recording.

6.2 Pre-processing Pipeline

6.2.1 Channel artifacts

An electrode or set of electrodes in an EEG recording may move during an EEG session, resulting in poor contact with the scalp and hence a poor-quality signal. Electrodes may also have mechanical faults which can partially or wholly degrade the signal received. To classify channels as artifactual, three parameters of each channel were calculated:

The first parameter was the mean of the channel's correlation coefficients with other channels. Most channels, especially in a high-density system, should correlate highly with neighboring channels. Therefore, a channel with contaminated data will likely have a low correlation with other channels.

Parameter 1 , $\sum$ rxn, xm
The mean correlation coefficient of channel n, where r_{xn},x_m is the Pearson correlation coefficient between channels n and m. A contaminated channel may correlate well with other channels but have a higher variance (due to additive noise) and therefore the second parameter is the variance of the channel.

Parameter 2: $S_x{}^2{}_n$
The variance of channel n, the third parameter was the Hurst exponent. The Hurst exponent is a measure of long-range dependence within a signal. Human phenomena such as EEG have values of H ≈ 0.7, and signals that deviate from this number are more likely to be artifacts.

Parameter 3: H_{xn},
The Hurst exponent of channel n, channels identified as contaminated were removed and data at this electrode were reconstructed by interpolating from neighboring electrodes while keeping the data from the contaminated signal as reference.

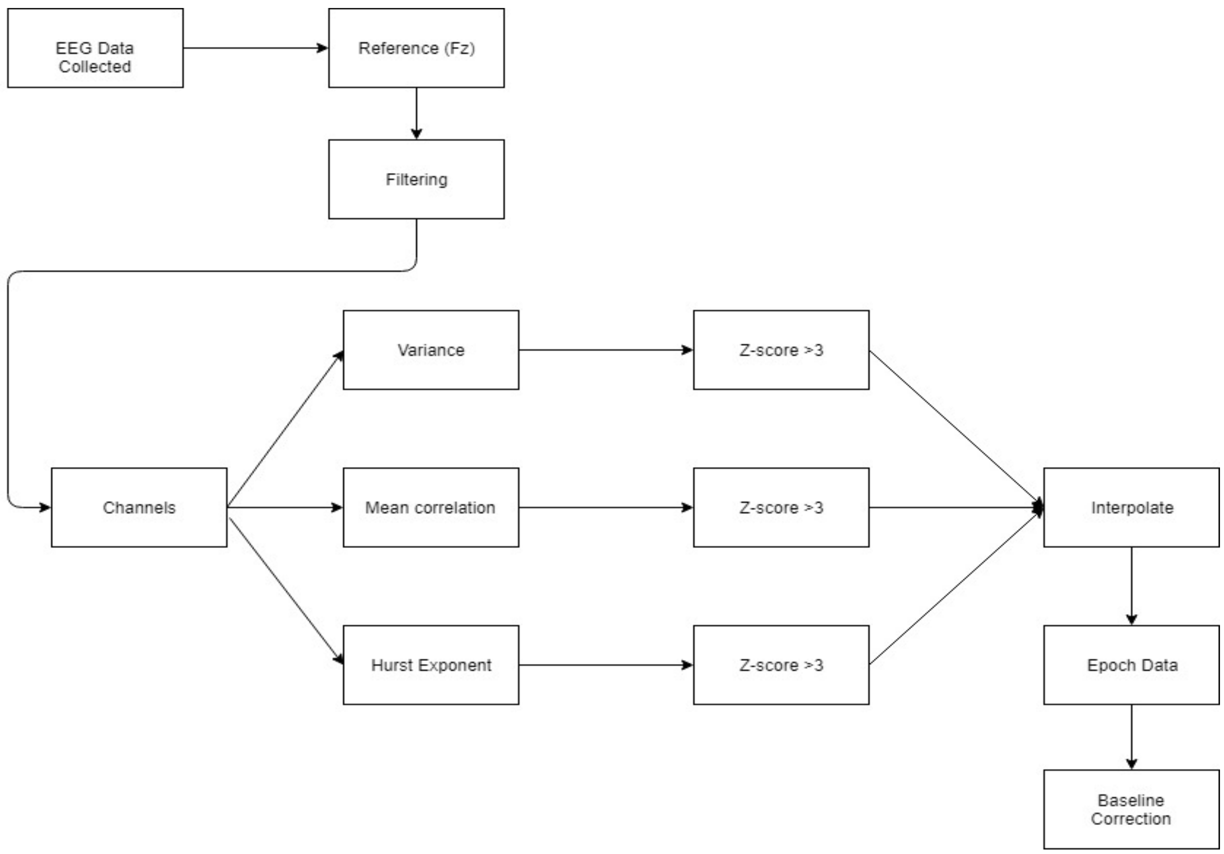


Figure 16 : Channel Artifact Pipeline

6.2.2 Epoch artifacts

An epoch in an EEG dataset may at times be contaminated with all-channel noise, caused due to subject movement, and physical movement of the electrodes. To detect such epochs, 3 parameters were computed for each channel within the epoch. Movement of electrodes on scalp results in a change in impedance between the scalp and the electrodes, which then affects the electrode voltage offsets. This change contaminates epochs and can be identified by its high amplitude. To detect this contamination, the first parameter computed was the amplitude range of the epoch.

Parameter 4: $\left\langle \max(x_{n_e}) - \min(x_{n_e}) \right\rangle$
The amplitude range in epoch e Shifting electrodes may also produce fewer extreme movements that may not have sufficient amplitude range to exceed the threshold of a Z-score equal to 3, but still contaminate an epoch. This type of artifact may be reflected in a high deviation of that epoch's average value from the average values across all channels. The second parameter computed was the deviation from each channel's average value.

Parameter 5 $\left\langle \left\langle x_{n_e} \right\rangle - \langle x_n \rangle \right\rangle$
The deviation from the channel average in epoch ***e*** Subject movement also produces EMG interference. A high variance will reflect such activity, and so the third parameter calculated was the variance.

Parameter 6: $\langle S^2_{Xne} \rangle$

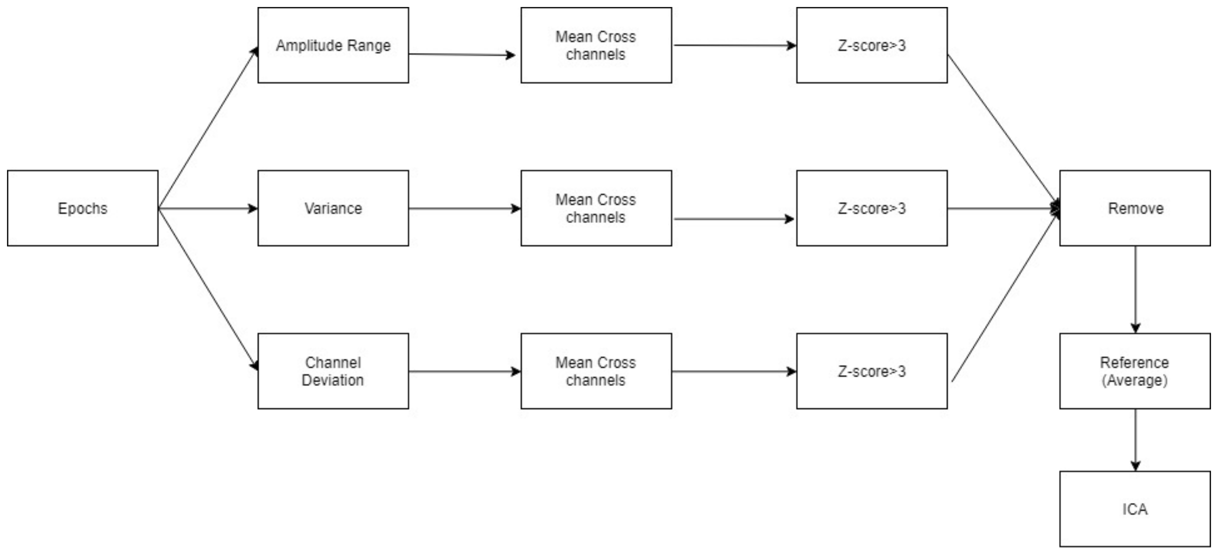


Figure 17: Epoch Artifact Pipeline

6.2.3 IC artifacts

The Infomax algorithm was employed to perform the ICA decomposition. The number of data points needed to find C stable components from ICA is typically kC^2 for each data channel, where k is a multiplier. The k value was set to 25, as recommended in. For example, our real data were of length 512×205=104960 points, given 128-scalp channels, 4 EOG channels and 2 mastoid channels, the maximum possible number of ICs would be 134. However, this would not have met the k=25 criterion, which would necessitate $25\times(134)^2$ =448900 data points. Therefore, the value of C was reduced to C_{pca} by performing Principal Component Analysis (PCA) on the EEG data and keeping nly the first C_{pca} principal components. C_{pca} was calculated as:

$$C_{pca} = \text{floor}\left(\sqrt{\frac{L}{k}}\right) \qquad (11)$$

where L is the length of the EEG dataset (in samples), and floor indicates rounding down to the nearest integer. This reduces the rank of the data and so a smaller number of ICs are computed.

ICA often produces ICs which consist entirely of artifactual data. These can then be subtracted from the dataset, leaving EEG data without the artifact. To classify ICs, five parameters were computed.

To identify artifacts caused by eye blinks (vertical EOG, VEOG) or saccades (horizontal EOG, HEOG), the correlation coefficients of each IC time series with the four recorded EOG (two VEOG and two HEOG) data channels was calculated, and the maximum absolute value was taken as the first parameter. The absolute was taken to account for possible differences in polarity between the EOG channel data and the IC time series.

Parameter 7: $\max(r_{xct}, x_{EOG^{1,2,3,4}})$,

the maximum of the absolute correlation coefficient

between component c time-course and EOG channels

Another common type of artifact singled out by ICA is a short, high-amplitude, single-electrode offset, often termed a "pop-off". An IC consisting of a pop-off has spatial data which shows activity in a single-channel and none otherwise. This is reflected in a high kurtosis value in the spatial data, as kurtosis measures the peakedness of data. The second parameter computed was the kurtosis of the spatial data.

Parameter 8: $(\mu_4/\mu^2{}_2) - 3$,

where $_i$ gives the ith central moment of the spatial data, is the equation for kurtosis of the

spatial information in component c

There is typically white noise in the acquired data due to hardware properties. White noise has a close-to-flat frequency power spectrum, as opposed to EEG components which have a 1/f power spectrum distribution. Residual white noise may remain after filtering, albeit with a very low contribution. Independent components consisting of white noise were identified by calculating the slope of the spectrum over the low-pass filter band as the third parameter.

Parameter 9: $\left\langle \frac{dF(x_{c_t})}{df} \right\rangle \Big| f_{LP1} < f < f_{LP2}$

The mean slope of the power spectrum of the component c time-course, between the band edges of the low-pass filter band The fourth parameter estimated was the Hurst exponent.

Parameter 10: H_{xct}, the Hurst exponent of component c time course The fifth parameter was the median gradient value, which is above threshold if the IC contains considerable high frequency content, was also calculated for each IC time series.

Parameter11: $\mathrm{median}\left(\frac{d(x_{c_t})}{dt}\right)$ the median slope of the component c time-course

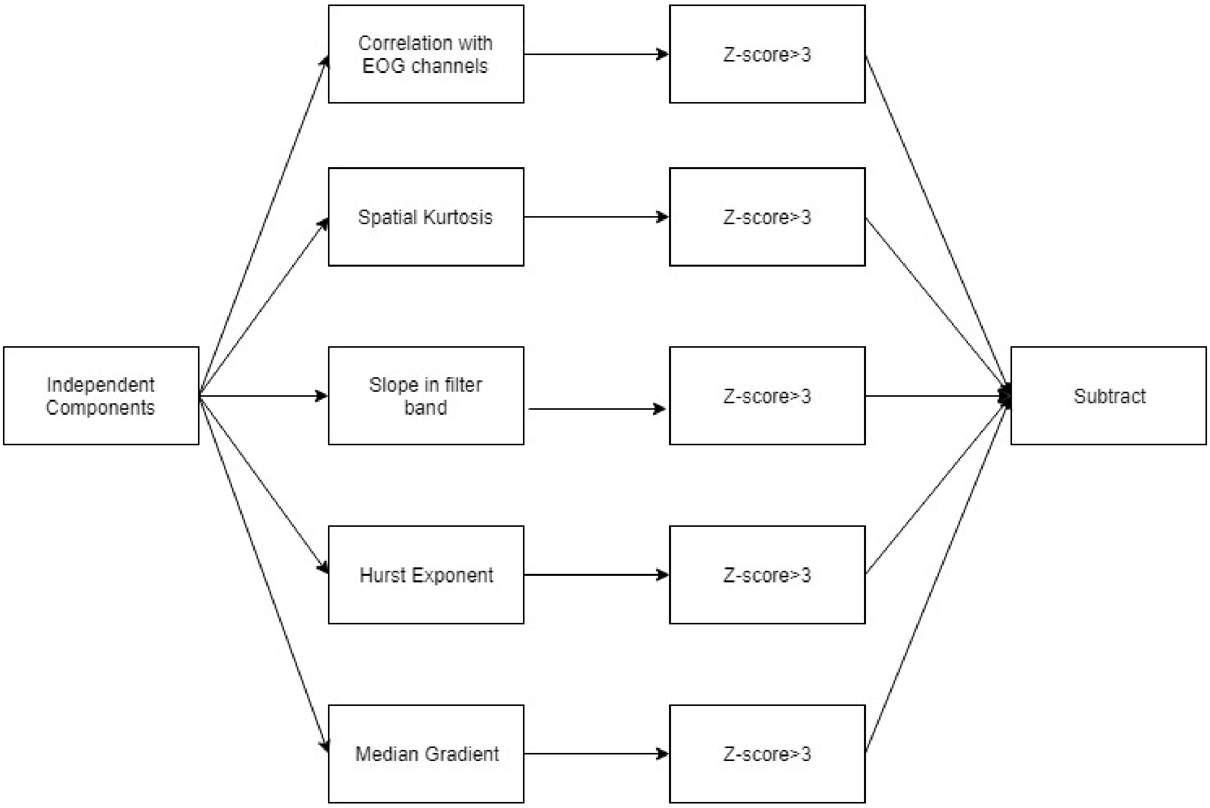


Figure 18 : IC Artifact Pipeline

6.2.4 Single-channel, single-epoch artifacts

Following the previous three steps, a high percentage of artifacts will have been removed. Some small transient artifacts may remain on single channels, within single epochs – for example, short bursts of white noise due to transient electrical faults, or electrodes that lost contact during a recording and were not sufficiently noisy to be detected as bad channels. Such artifacts were corrected by interpolating single channels within single epochs, using spherical splines. To detect the artifacts, four parameters were computed for each channel within each epoch. The first parameter was the variance, to detect single channels in single epochs with additive noise.

Parameter 12: $S^2_{xne,}$ the variance of channel n in epoch the second was the median gradient, to detect another high-frequency activity.
Parameter 13: median $\left(\frac{d(x_{n_e})}{dt}\right)$ the median slope of the channel n in epoch the third was the amplitude range of the channel, to detect pop-offs.

Parameter 14: $\max(x_{ne}) - \min(x_{ne})$, the amplitude range of channel n in epoch e Fourth, in order to detect electrical drift, the deviation of the mean amplitude in the epoch for each channel from the whole-channel mean amplitude was calculated.

Parameter15: $\langle x_{n_e} \rangle - \langle x_n \rangle$,the deviation from the channel average of channel n in epoch e

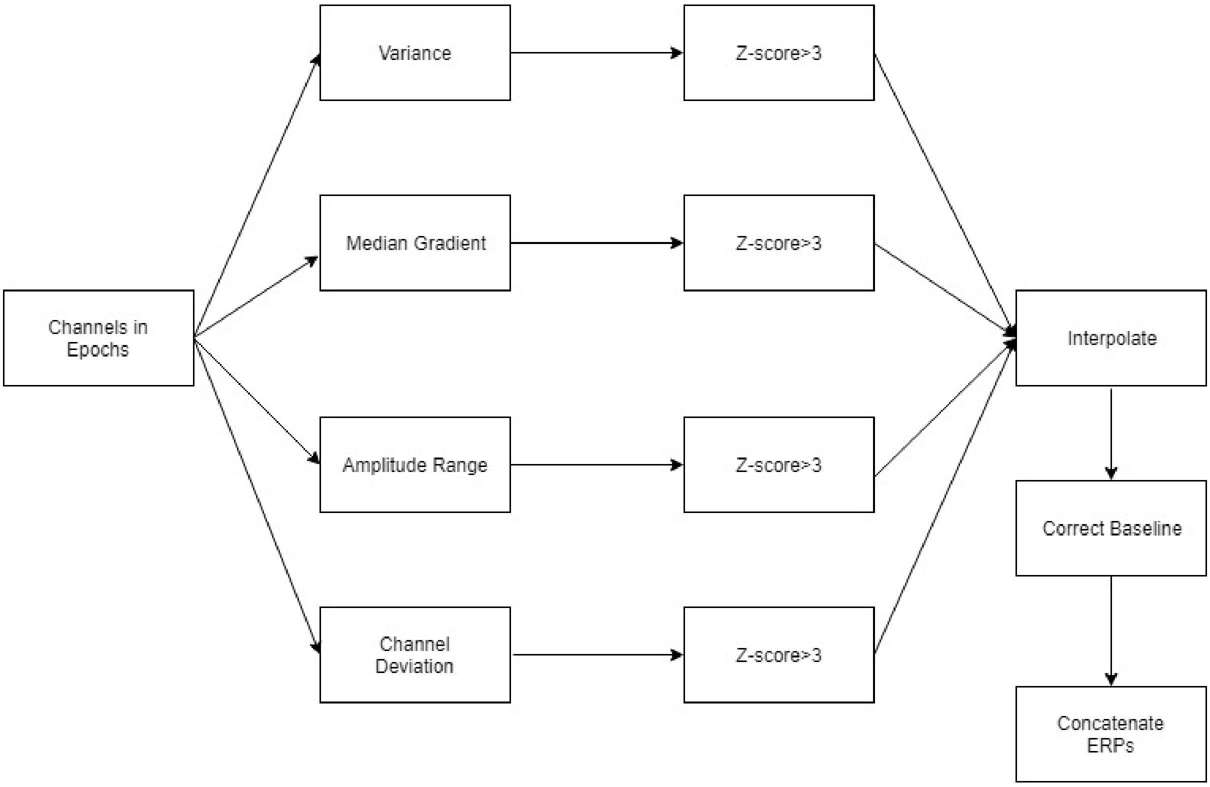


Figure 19: Single Channel ,Single Epoch Artifact Pipeline

6.2.5 Contaminated datasets

After each file had been processed, a grand average dataset was created (i.e. all subjects' data were aggregated) so that each epoch was the ERP of a processed file. In a typical EEG study there are often subjects whose data are contaminated by artifacts to the extent that their data are not a true reflection of neural processes, and therefore distort the grand average data. These subjects' data are often removed entirely from the grand average. In order to identify these subjects, the epoch artifact detection method was repeated for the
grand average:

Parameter 16: $\max(x_{ne})$ - $\min(x_{ne})$, the amplitude range in epoch e.

Parameter 17: $\langle S^2_{x_{n_e}} \rangle$ the variance in epoch e. N

Parameter 18: $<<X_{ne} - <X_n>>_N$ the deviation from the channel average in epoch e. An additional parameter – the maximum absolute value of the EOG channels in the ERP – was computed for each epoch in order to determine whether eye movement artifacts remained.

Parameter 19: $\max(x_{EOG1,2,3,4e})$, the maximum value in the EOG channels in epoch. Thresholds were calculated for each parameter, and any epoch (i.e. subject) that surpassed that threshold was considered contaminated and removed from the grand average file.

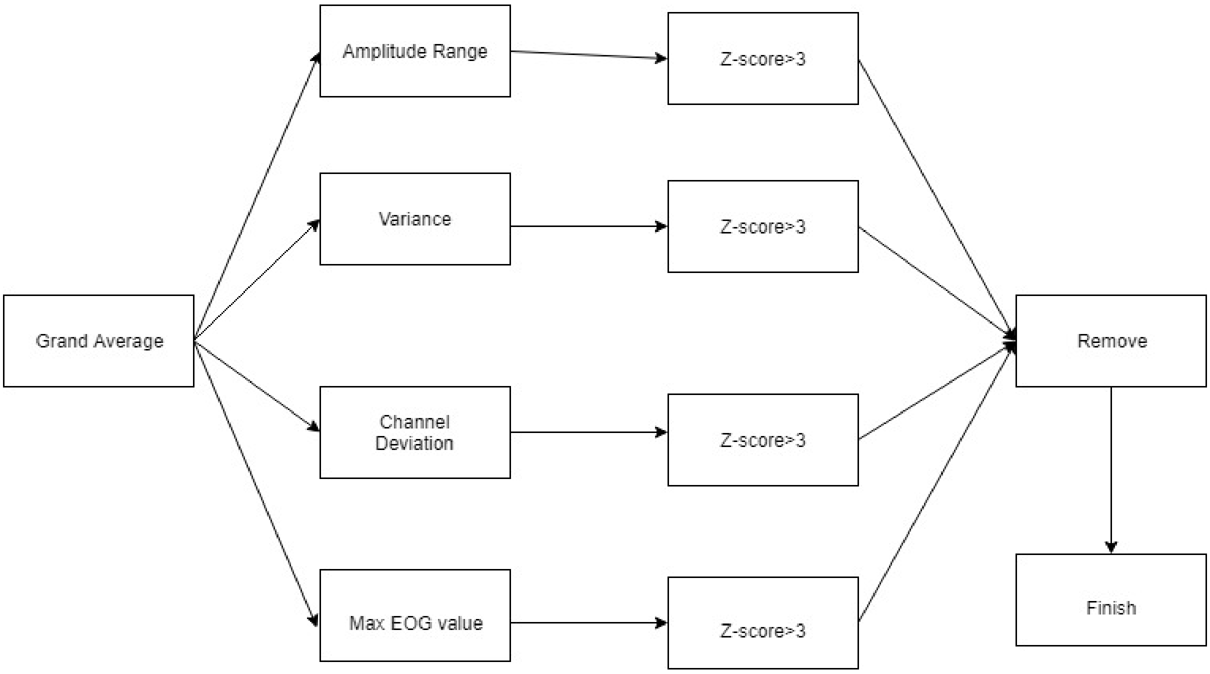


Figure 20: Contaminated Dataset Correction Pipeline

6.3 Feature Extraction

Artefacts in EEG recordings are forms of outliers and are considered as disturbances in a regular brain-signal, not originating from the brain, which usually show up in the signal as noisy frequency bands. Binning and outlying frequency removal is carried out. Any frequency below 0.5Hz and above 200 Hz is removed as they are assumed to give no significant information gain. The Band Gaps taken are Delta; 0.5-4, Theta; 4-8, Alpha; 8-12, Beta; 12-30, Gamma; 30-100 frequencies. The data is then windowed into 2 sec segments non overlapping windows.

After artefact removal, the features described in the following paragraph were then extracted for each segment for each of the subjects. Power spectral density and energy (Quasiperiodic fluctuations or rhythmic behavior characterized by a peak in the power spectrum) at specific frequencies are extracted as they may be used to identify epileptic seizures in some cases. Spectral entropy is a measure of the spectral power distribution of a signal treating it as a probability distribution in the frequency domain. The spectral entropy is lower when there is information in the signal. This aspect is used for feature extraction in biomedical signals. We extract signal energy-based features like entropies and spectral densities. Statistical quantitative features as moments, particularly skewness, kurtosis, are measure of the shape of the distribution of a set of points, standard deviation quantifies the amount of variation or dispersion of data points of the signal. These statistics are applied as features. Further measures like Hjorth parameters which indicate statistical properties used in signal processing in the time domain are also used. The parameters are Activity, which indicates the surface of power spectrum in the frequency domain, Mobility, which is the proportion of standard deviation of the power spectrum, and Complexity, which represents a measure of similarity between the signal and a pure sine wave are also computed. Fractal dimension (FD) estimates are obtained from the segment to capture self-similar unvaried repetitive patterns in the EEG signal. Fractal dimension is shown to characterizes the nonlinear behavior and state of many chaotic systems. In analysis of chaotic time series such as EEG, this feature is used to discern specific states of physiological function. Fractal Dimensions for each segment are computed and used as features. The Hear wavelet was used to derive DWT for each segment. Additional features from Spectral Bands which measure the signal energy in a specific frequency range, as calculated through Fourier transform, and Spectral Frequency which distinguishes the signal's energy distribution in terms of how signal power is concentrated in the frequency spectrum, were also taken in.

A total of 47 features were extracted for every channel second of the EEG signals. For each 10 minutes recording the 56 features derived from the 30 sec Segment where concatenated to derive a 1220-dimensional feature vector representing the recording.

6.3.1 Correlation dimension:

For any set of *N* points in an *m*-dimensional space x(i) = [x1(i),….,xn(i)], i=0,1,2….n, then the correlation integral $C(\varepsilon)$ is calculated by:

$$C(\varepsilon) = \lim_{N\to\infty} \frac{g}{N^2} \tag{12}$$

6.3.2 Power Spectral Intensity:

The average power of a signal x(t) over all time is therefore given by the following time average:

$$P = \lim_{T\to\infty} \frac{1}{T} \int_0^T |x(t)|^2 \, dt. \tag{13}$$

Spectral Entropy:

$$PSE = -\sum_{i=1}^{n} p_i \ln p_i \tag{14}$$

where p_i is the probability density function

6.3.3 Hjorth Parameters:

Hjorth Activity

The activity parameter represents the signal power, the variance of a time function. This can indicate the surface of power spectrum in the frequency domain. This is represented by the following equation:

$$\text{Activity} = \text{var}(y(t)). \tag{15}$$

Where y(t) represents the signal.

Hjorth Mobility

The mobility parameter represents the mean frequency or the proportion of standard deviation of the power spectrum. This is defined as the square root of variance of the first derivative of the signal y(t) divided by variance of the signal y(t).

$$\text{Mobility} = \sqrt{\frac{\text{var}(\frac{dy(t)}{dt})}{\text{var}(y(t))}}. \tag{16}$$

Hjorth Complexity

X signal's similarity to a pure sine wave, where the value converges to 1 if the signal is more similar.

$$\text{Complexity} = \frac{\text{Mobility}(\frac{dy(t)}{dt})}{\text{Mobility}(y(t))}. \tag{17}$$

6.3.4 Hurst Exponent:

The Hurst exponent, H, is defined in terms of the asymptotic behavior of the rescaled range as a function of the time span of a time series as follows;

$$\mathrm{E}\left[\frac{R(n)}{S(n)}\right] = Cn^H \text{ as } n \to \infty\,, \tag{18}$$

where;

R(n) is the range of the first n cumulative deviations from the mean, and S(n) is their standard deviation E[x] is the expected value n is the time span of the observation (number of data points in a time series) C is a constant.

Detrended Fluctuation Analysis: Given a bounded time series x_t of length N, where t€N, integration or summation first converts this into an unbounded process X_t:

$$X_t = \sum_{i=1}^{t}(x_i - \langle x \rangle) \tag{19}$$

Where x denotes the mean value of the time series. X_t is called cumulative sum or profile. This process converts, for example, an i.i.d. white noise process into a random walk.

6.3.5 Discreet Wavelet Transform:

The DWT of a signal x is calculated by passing it through a series of filters. First the samples are passed through a low pass filter with impulse response g resulting in a convolution of the two:

$$y[n] = (x * g)[n] = \sum_{k=-\infty}^{\infty} x[k]g[n-k] \tag{20}$$

6.3.6 Standard Deviation:

Standard Deviation represents how much elements of a group differ from the mean of the group

$$s = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \bar{x})^2}{N-1}} \tag{21}$$

6.3.7 Variance:

The degree of divergence among a set of elements

$$\sigma \equiv \left(\mathrm{E}\left[(x-\mu)^2\right]\right)^{\frac{1}{2}}. \tag{22}$$

6.3.8 Skewness

The skewness of a random variable X is the third standardized moment γ_1, defined as:

$$\gamma_1 = \mathrm{E}\left[\left(\frac{X-\mu}{\sigma}\right)^3\right] = \frac{\mu_3}{\sigma^3} = \frac{\mathrm{E}\left[(X-\mu)^3\right]}{(\mathrm{E}[(X-\mu)^2])^{3/2}} = \frac{\kappa_3}{\kappa_2^{3/2}} \quad (23)$$

6.3.9 Kurtosis

The kurtosis is the fourth standardized moment, defined as

$$\mathrm{Kurt}[X] = \mathrm{E}\left[\left(\frac{X-\mu}{\sigma}\right)^4\right] = \frac{\mu_4}{\sigma^4} = \frac{\mathrm{E}[(X-\mu)^4]}{(\mathrm{E}[(X-\mu)^2])^2}, \quad (24)$$

6.3.10 Fast Fourier Transform:

Let x0, ..., xN−1 be complex numbers. The DFT is defined by the formula

$$X_k = \sum_{n=0}^{N-1} x_n e^{-i2\pi kn/N} = \sum_{n=0}^{N-1} x_n w^{-kn} \qquad k = 0, \dots, N-1. \quad (25)$$

Where w=e^{i2\pi /N} is the first complex N-th root of 1.

7. Results

After artefact removal and feature extraction from the signal, the features are scaled, so that the coefficients are penalized based on their predictive power and not their amplitude. These features are used to train the following four broad categories of models; Generalized Regression models (Logistic Regression) Support Vector Machines (Linear and RBF Kernel Support Vector Machine Classifiers), Decision Trees (Decision Tree Classifier) and Tree Ensemble models (Parallel ensemble models: Random Forests and Sequential ensemble model Gradient Boosting Classifiers) and also K-neighbors' clustering algorithm

The following different classification tasks have been performed:

Classifications of data based on four different emotions of happy, calm, sad and disgust. Classifications of data based on negative and positive emotions where disgust and sad are clubbed as negative and happy and calm are clubbed as positive negative.

Classification of each emotion against baseline data where the baseline corresponds to the subject being in a completely relaxed phase with eyes closed.

Classification amongst the 4 emotions.

This is the main task of the classification framework where we aim to identify the emotions by training data for a 2 sec window which corresponds to a 256 data points (2 secs into sampling frequency of 128) per channel. The training can be performed in two ways were the model is trained by providing data for 2 sec windows for all channels as one feature row while the other procedure involves providing data for all the channels as individual features. For our the purpose we have implemented the latter mainly due to the fact that emotions when represented in EEG are not evident in all channels. Clubbing data of different channels would then mask the peculiarity exhibited by the channel which exhibits change when a subject sink into an emotion. The features are scaled such the mean is zero to make sure the huge change in the features value doesn't hamper with the training of the model and it initially gives equal importance to every feature. The results are mentioned in the table below.

Model	Accuracy	Precision	Recall	F1 Score
Decision Tree	35.91	63.60	63.16	63.38
Logistic Regression	33.71	57.48	72.98	84.28
Multi-Layer Perceptron	58.74	78.07	78.35	78.21
K Neighbours	**85.06**	**90.56**	**91.60**	**91.08**
Gradient Boost	38.22	60.17	77.18	67.62
Random Forest	45.83	70.21	71.94	71.06

Table 6: Disgust vs happy vs clam vs sad

Classification between positive and negative emotions

This classification was done basically by keeping mind its application in neural marketing and neural feedback. Classifying the experience of the user of a product or some media into a positive or negative using EEG would be help in product evolution. For this purpose happiness and calm were clubbed into one treated as positive emotion while disgust and sad were clubbed together to be considered as negative emotions. The training method was similar to the procedure followed in the first case. The results are mentioned in the table below.

Model	Accuracy	Precision	Recall	F1 Score
Decision Tree	57.14	60.07	60.02	63.04
Logistic Regression	55.67	56.50	80.20	66.29
Multi Layer Perceptron	72.42	74.21	75.50	74.83
K Neighbours	**85.44**	**90.51**	**81.79**	**85.93**
Gradient Boost	57.63	57.15	88.12	69.34
Random Forest	63.85	66.45	67.67	67.05

Table 7: Positive vs Negative Emotion

Classification of emotions with baseline

This task is solely performed so as to determine biomarkers that differentiate a subjects neutral state and a state where the subject Is experiencing some emotion. In this approach different models are trained to classify different emotions against baseline recording where baseline

recording corresponds to a resting and relaxed state of the user with eye closed and in comfortable conditions. After models are trained we perform feature importance, thresholding and correlation analysis to identify features which are most decisive, the range of those features and the corresponding emotion associated to that range combined with their biological significance. Below is mentioned the best results from the top two classifiers.

Classification Type	Model	Accuracy	Precision	Recall	F1Score
Happy vs Baseline	K Neighbours	89.61	**96.10**	89.93	91.28
	Multi Layer Perceptron	**91.37**	91.74	**94.74**	**93.22**
Sad vs Baseline	K Neighbours	87.85	81.08	**97.18**	88.40
	Multi Layer Perceptron	**91.28**	**90.49**	91.28	**90.89**
Calm vs Baseline	K Neighbours	90.02	**95.67**	87.96	91.65
	Multi Layer Perceptron	**91.95**	92.22	**94.85**	**93.52**
Disgust vs Baseline	K Neighbours	**92.27**	92.57	**95.75**	**94.13**
	Multi Layer Perceptron	92.02	**92.64**	95.25	93.93

Table 8: Emotions VS Baseline

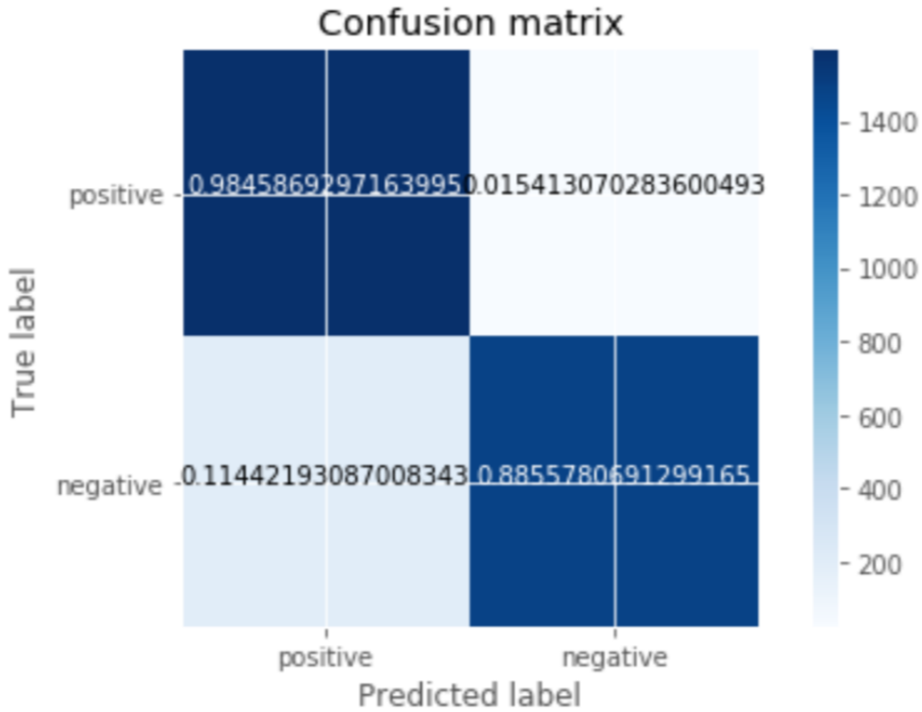


Figure 21: Positive Emotion vs Negative Emotion

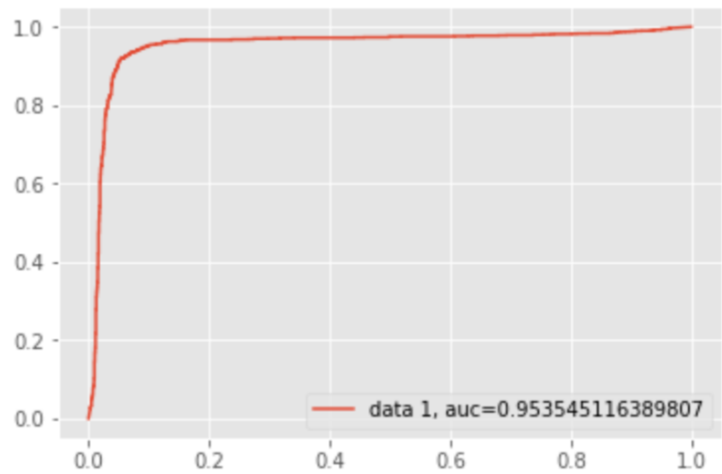


Figure 22: Area Under Curve for KNN

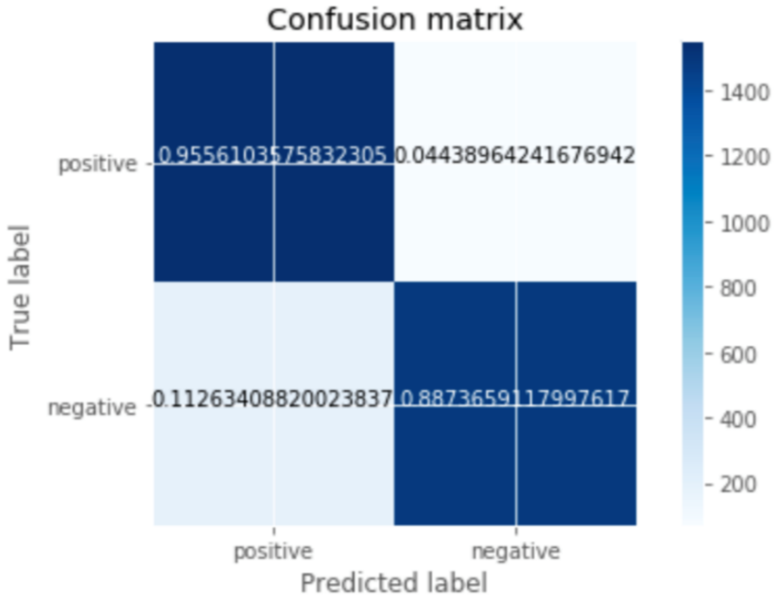


Figure 23: Kernel SVM Positive Emotion vs Negative Emotion

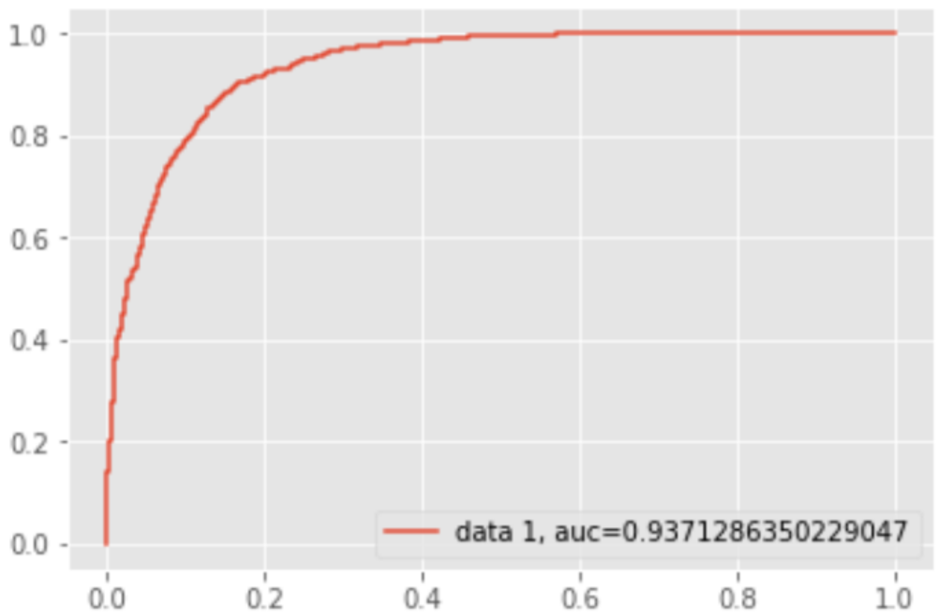


Figure 24: Area Under Curve for Kernel SVM

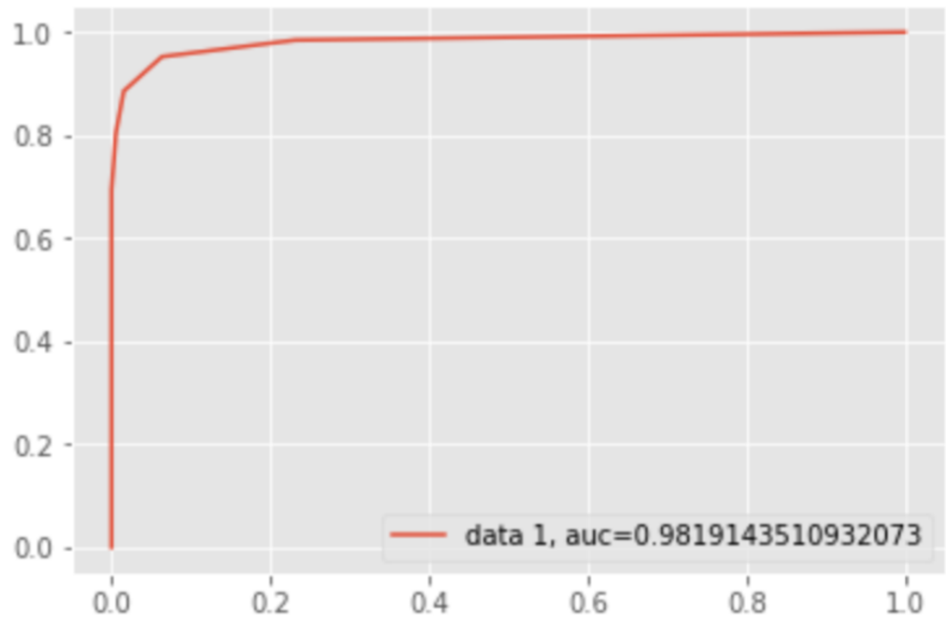


Figure 25: AUC Emotion Classification

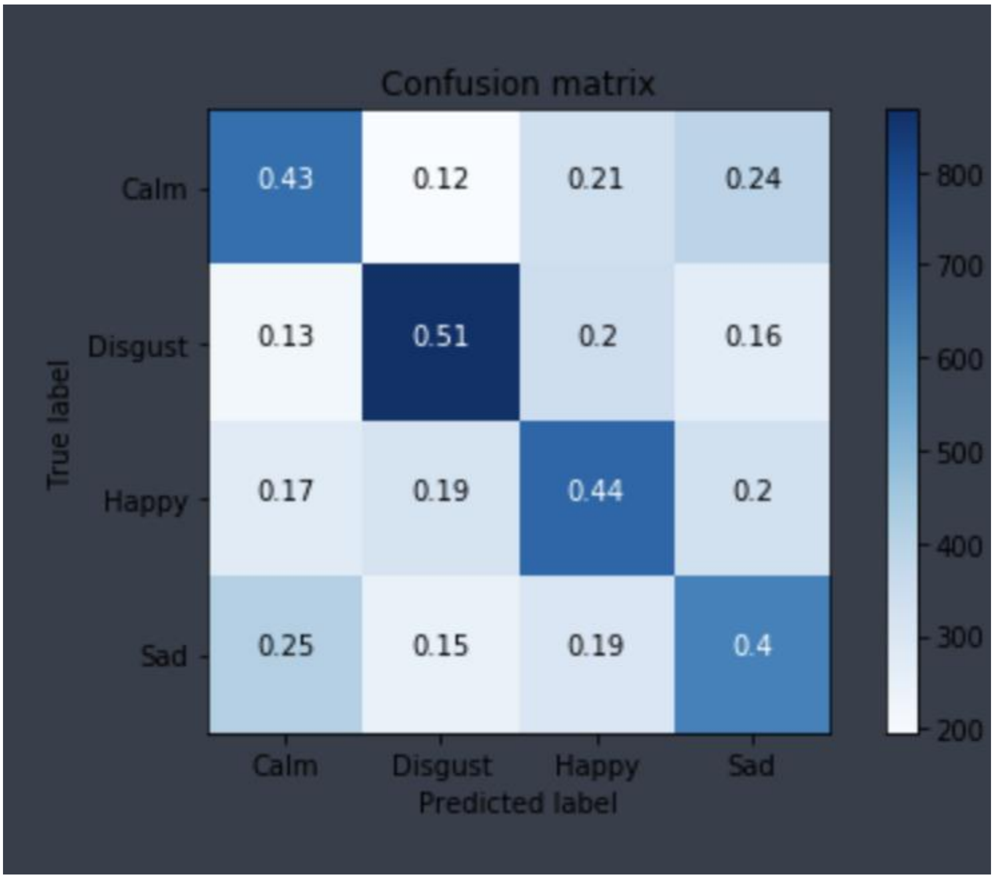


Figure 26: Emotion Classification

8. Conclusion

In this project, an emotion recognition method based on features from different domains is proposed. An extensive analysis has been carried out to investigate the effectiveness of the features for emotion classification and identifying the biomarkers for the same. The results draw a mapping of time-series, frequency and energy-based features for emotion recognition. Our study confirms the deduction that emotion is more relative to high frequency component. This consists with findings in that Power Spectral analysis for Beta (16–32 Hz) and Gamma (32–64 Hz) have more feature importance than the others. It also identifies a high positive correlation of energy-based features such as spectral entropy and fractal dimensions.

9. Codes and Standards

EmoEngine Code	Hex Value	Description
EDK_OK	0x0000	Operation has been carried out successfully.
EDK_UNKNOWN_ERROR	0x0001	An internal fatal error occurred.
EDK_INVALID_PROFILE_ARCHIVE	0x0101	Most likely returned by EE Set User Profile when the content of the supplied buffer is not a valid serialized EmoEngine profile.
EDK_NO_USER_FOR_BASE_PROFILE	0x0102	Returns when trying to query the user ID of a base profile.
EDK_CANNOT_ACQUIRE_DATA	0x0200	Returns when Emo Engine is unable to acquire any signal from Emotiv epoc™ for processing
EDK_BUFFER_TOO_SMALL	0x0300	Most likely returned by EE Get User Profile() when the size of the supplied buffer is not large enough to hold the profile.
EDK_OUT_OF_RANGE	0x0301	One of the parameters supplied to the function is out of range.
EDK_INVALID_PARAMETER	0x0302	One of the parameters supplied to the function is invalid (e.g. null pointers, zero size buffer)
EDK_PARAMETER_LOCKED	0x0303	The parameter value is currently locked by a running detection and cannot be modified at this time.

EmoEngine Code	Hex Value	Description
EDK_COG_INVALID_TRAINING_ACTION	0x0304	The specified action is not an allowed training action at this time.
EDK_COG_INVALID_TRAINING_CONTROL	0x0305	The specified control flag is not an allowed training control at this time.
EDK_COG_INVALID_ACTIVE_ACTION	0x0306	An undefined action bit has been set in the actions bit vector.
EDK_COG_EXCESS_MAX_ACTIONS	0x0307	The current action bit vector contains more than maximum number of concurrent actions.
EDK_EXP_NO_SIG_AVAILABLE	0x0308	A trained signature is not currently available for use – some actions may still require training data.
EDK_INVALID_USER_ID	0x0400	The user ID supplied to the function is invalid.
EDK_EMOENGINE_UNINITIALIZED	0x0500	Emo Engine™ needs to be initialized via calling EE Engine Connect() or EE Eng Remote Connect before calling any other APIs.
EDK_EMOENGINE_DISCONNECTED	0x0501	The connection with Emo Engine™ via EE Eng Remote Connect has been lost.
EDK_EMOENGINE_PROXY_ERROR	0x0502	Returned by EE Eng Remote Connect when the connection to the EmoEngine™ cannot be established.
EDK_NO_EVENT	0x0600	Returned by EE_Engine Get Next Event when there is no pending event.

EmoEngine Code	Hex Value	Description
EDK_GYRO_NOT_CALIBRATED	0x0700	The gyroscope is not calibrated. Please ask the user to remain still for .5 seconds.
EDK_OPTIMIZATION_IS_ON	0x0800	Operation failed due to algorithm optimization settings.

Table 9 : Codes and Standards for Emotiv Epoc Plus

10. Schedules, Tasks and Milestones

Module - 1	Finalizing Test
Module - 2	Data Acquisition
Module - 3	Data Cleaning Pipeline
Module - 4	Feature Extraction Pipeline
Module - 5	Model Development and Training
Module - 6	Results Comparision and Report

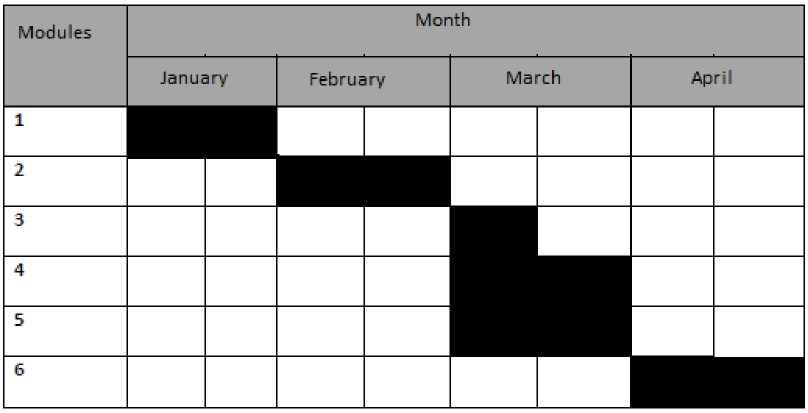


Figure 27: Gantt Chart for timeline

11. Future Work

As we discuss in the results section that Ensemble methods when regularised outperform other approaches, the follow up work should constitute better regularisation approaches on Ensemble models, using other regularised forms of Adaboost (RegBoost, AdaBoostReg, LPBoost, QPBoost) should help. Greater insights can be gained by using Convolutional Neural Networks as a feature extractor over the spectrogram of the EEG signals, also using autoencoders over the raw EEG data for noise removal or on extracted features for removing redundant features could be highly productive.

12. References

[1] Bethel, C.L., Salomon, K., Murphy, R.R., Burke, J.L.: Survey of psychophysiology measurements applied to human-robot interaction. In: The 16th IEEE International Symposium on Robot and Human interactive Communication, RO-MAN 2007, pp. 732–737.IEEE (2007)

[2] Hagelbäck, J., Hilborn, O., Jerčić, P., Johansson, S.J., Lindley, C.A., Svensson, J., Wen, W.: Psychophysiological Interaction and Empathic Cognition for Human-Robot Cooperative Work (PsyIntEC),pp.283–299.Springer,Cham(2014)

[3] Liu, Y., Sourina, O., Nguyen, M.K.: Real-time EEG-based human emotion recognition and visualization. In: International Conference on Cyberworlds (CW), pp. 262–269, October2010

[4] Lin, Y.-P., Wang, C.-H., Wu, T.-L., Jeng, S.-K., Chen, J.-H.: EEG-based emotion recognition in music listening: a comparison of schemes for multiclass support vector machine. In: International Conference on Acoustics, Speech and Signal Processing, ICASSP2009,pp.489–492.IEEE(2009)

[5] Bos, D.O.: EEG-based emotion recognition. Influ. Vis. Audit. Stimuli 56(3), 1–17 (2006)

[6] Murugappan, M., Rizon, M., Nagarajan, R., Yaacob, S., Zunaidi, I., Hazry, D.: Lifting scheme for human emotion recognition using EEG. In: International Symposium on Information Technology, ITSim2008, vol.2, pp.1–7. IEEE(2008)

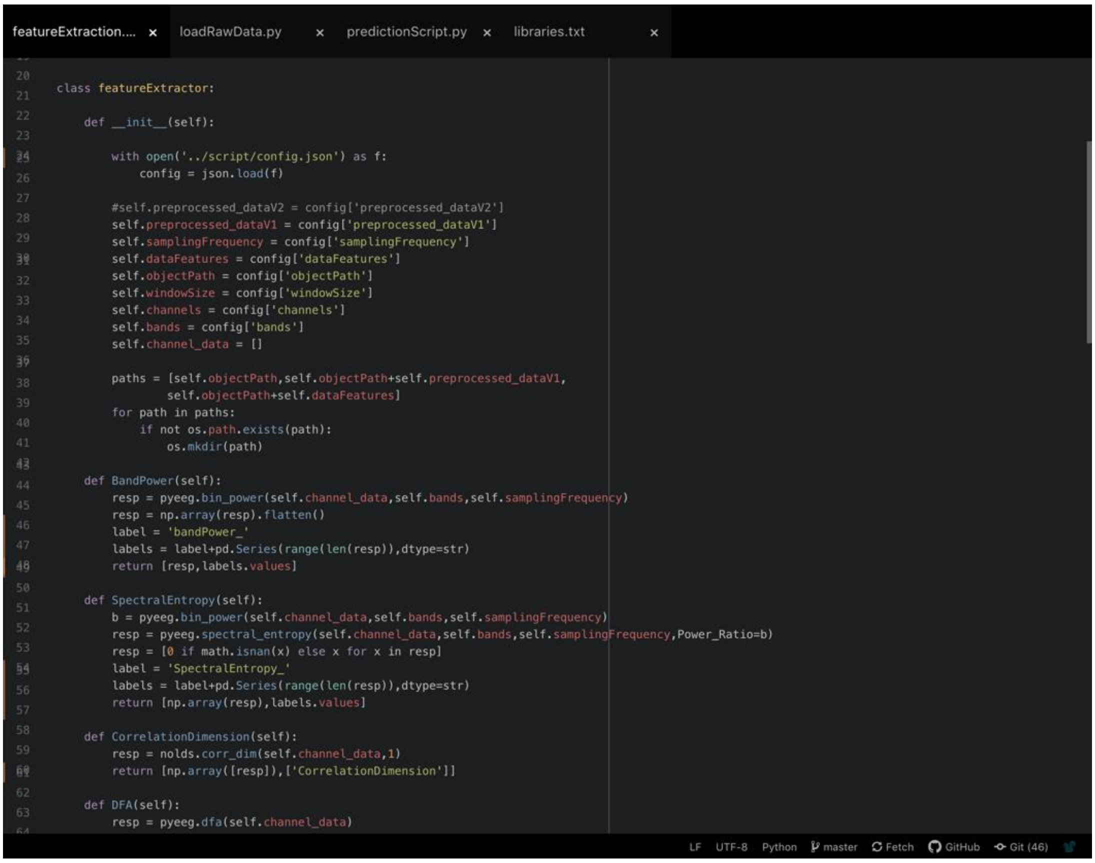
[7] Schaaff, K.: EEG-based emotion recognition. Ph.D. dissertation, Ph.D. thesis, UniversitatKarlsruhe(TH)(2008)

[8] Li, M., Chai, Q., Kaixiang, T., Wahab, A., Abut, H.: EEG emotion recognition system. In: In-vehicle Corpus and Signal Processing for Driver Behavior, pp. 125–135. Springer, Boston (2009)

[9] Sørensen, A.S., Nielsen, J., Maagaard, J., Nielsen, J.L., Rasmussen, G., Day, D.: Natural kinesthetic interaction and social relations between training-robots and their users.

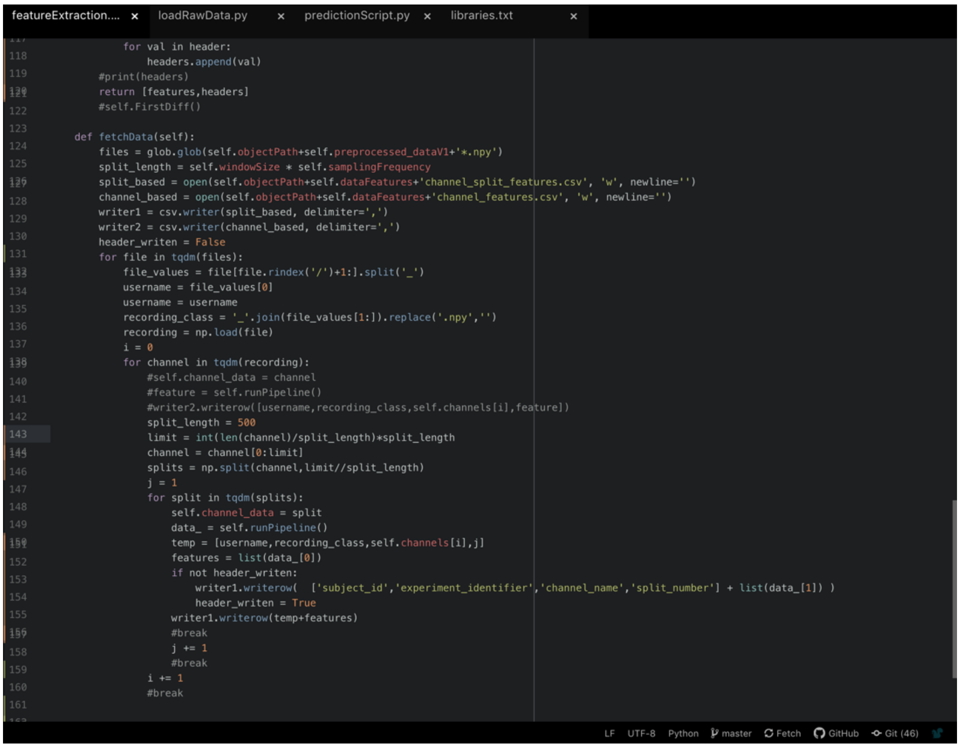
[10] Fisher, R.A.: The use of multiple measurements in taxonomic problems. Ann. Eugen. 7(2),179–188(1936)

APPENDIX 1: CODE SNIPPETS



```python
class featureExtractor:

    def __init__(self):

        with open('../script/config.json') as f:
            config = json.load(f)

        #self.preprocessed_dataV2 = config['preprocessed_dataV2']
        self.preprocessed_dataV1 = config['preprocessed_dataV1']
        self.samplingFrequency = config['samplingFrequency']
        self.dataFeatures = config['dataFeatures']
        self.objectPath = config['objectPath']
        self.windowSize = config['windowSize']
        self.channels = config['channels']
        self.bands = config['bands']
        self.channel_data = []

        paths = [self.objectPath,self.objectPath+self.preprocessed_dataV1,
                self.objectPath+self.dataFeatures]
        for path in paths:
            if not os.path.exists(path):
                os.mkdir(path)

    def BandPower(self):
        resp = pyeeg.bin_power(self.channel_data,self.bands,self.samplingFrequency)
        resp = np.array(resp).flatten()
        label = 'bandPower_'
        labels = label+pd.Series(range(len(resp)),dtype=str)
        return [resp,labels.values]

    def SpectralEntropy(self):
        b = pyeeg.bin_power(self.channel_data,self.bands,self.samplingFrequency)
        resp = pyeeg.spectral_entropy(self.channel_data,self.bands,self.samplingFrequency,Power_Ratio=b)
        resp = [0 if math.isnan(x) else x for x in resp]
        label = 'SpectralEntropy_'
        labels = label+pd.Series(range(len(resp)),dtype=str)
        return [np.array(resp),labels.values]

    def CorrelationDimension(self):
        resp = nolds.corr_dim(self.channel_data,1)
        return [np.array([resp]),['CorrelationDimension']]

    def DFA(self):
        resp = pyeeg.dfa(self.channel_data)
```

Figure 31: Feature Extraction Pipeline



```python
            for val in header:
                headers.append(val)
        #print(headers)
        return [features,headers]
        #self.FirstDiff()

    def fetchData(self):
        files = glob.glob(self.objectPath+self.preprocessed_dataV1+'*.npy')
        split_length = self.windowSize * self.samplingFrequency
        split_based = open(self.objectPath+self.dataFeatures+'channel_split_features.csv', 'w', newline='')
        channel_based = open(self.objectPath+self.dataFeatures+'channel_features.csv', 'w', newline='')
        writer1 = csv.writer(split_based, delimiter=',')
        writer2 = csv.writer(channel_based, delimiter=',')
        header_writen = False
        for file in tqdm(files):
            file_values = file[file.rindex('/')+1:].split('_')
            username = file_values[0]
            username = username
            recording_class = '_'.join(file_values[1:]).replace('.npy','')
            recording = np.load(file)
            i = 0
            for channel in tqdm(recording):
                #self.channel_data = channel
                #feature = self.runPipeline()
                #writer2.writerow([username,recording_class,self.channels[i],feature])
                split_length = 500
                limit = int(len(channel)/split_length)*split_length
                channel = channel[0:limit]
                splits = np.split(channel,limit//split_length)
                j = 1
                for split in tqdm(splits):
                    self.channel_data = split
                    data_ = self.runPipeline()
                    temp = [username,recording_class,self.channels[i],j]
                    features = list(data_[0])
                    if not header_writen:
                        writer1.writerow(  ['subject_id','experiment_identifier','channel_name','split_number'] + list(data_[1]) )
                        header_writen = True
                    writer1.writerow(temp+features)
                    #break
                    j += 1
                    #break
                i += 1
                #break
```

Figure 32: Feature Extraction Pipeline

```python
from pyprep.noisy import Noisydata,find_bad_epochs
import matplotlib.pyplot as plt
from mne import Epochs
import warnings
import mne.io
import glob
warnings.filterwarnings('ignore')
```

```python
def getBadChannels(eegChannels,sfreq = 128,n_chans = 14):
    nd = Noisydata(eegChannels)
    nd.find_all_bads(ransac=False)
    bads = nd.get_bads(verbose=False)
    return(bads)
```

```python
test = None
folders = glob.glob('../dataset/*/')
for folder in folders:
    files = glob.glob(folders[3]+'*.edf')
    for file in files:
        rawData = mne.io.read_raw_edf(files[3],preload=True,verbose=50)
        rawData.set_montage('standard_1020',verbose=50)
        channels = ["AF3", "F7", "F3", "FC5","T7","P7","O1",
           "O2","P8", "T8","FC6","F4", "F8", "AF4"]
        eegChannels = rawData.pick_channels(channels)
        test = eegChannels.copy()
        #epoched = Epochs(rawData,10)
        badChannels = getBadChannels(rawData)
        eegChannels.info['bads'] = badChannels
        eegChannels.interpolate_bads(reset_bads=True,verbose=False)
        break
    break

```

```python
test.plot()
```

Figure 33: Feature Extraction Pipeline

APPENDIX 2: LOG BOOK

SEPTEMBER 2018

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
26	27	28	29	30	31	1
2	3	4	5 Group Finalization	6	7	8 Topic Finalization
9	10	11	12	13	14	15
16	17 Approaching Capstone Mentor	18	19	20 Check For Funding	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

Notes:

OCTOBER 2018

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
30	1	2	3 Guide Finalizing Capstone Topic	4	5	6 Going through research papers
7 Selecting trending issues	8 Understanding EEG	9 Going through EEG documentations	10 Analysing available technology	11 Going thorough various EEG projects	12	13
14	15 Analysing Capstone requirements	16	17 Reporting all current findings to the Guide	18	19 Selecting Hardware component	20
21	22	23 Check for available funding	24	25	26	27
28	29	30	31	1	2	3

Notes:

NOVEMBER 2018

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
28	29	30	31	1 Selecting Machine Learning algorithms	2	3
4	5	6	7	8	9	10
11	12	13	14	15 Exploring ANN architectures	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30 Looking into IEEE papers	1

Notes: FAT was schedule in this month

DECEMBER 2018

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
25	26	27	28	29	30	1
2	3 Finalizing budget required	4	5 Reading Emotiv documentation	6 Subscribe to EmotivPlus software	7 Searching EQ and IQ tests	8
9	10	11 Looking for dataset	12	13 Requesting dataset	14 Brushing up ML concepts	15
16 Setting up Emotiv hardware	17 Finalizing subjects for EEG dataset	18 Contacting subjects	19 Testing Emotiv hardware	20 Reporting to Guide	21 Studying Emotiv architecture	22
23	24	25 Selecting proper channels for Emotiv	26	27	28	29
30	31	1	2	3	4	5

Notes: From 25th winter vaction started

JANUARY 2019

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
30	31	1	2	3 Studying Frequency emitted from brain	4 Classifying frequencies	5
6	7	8 Receiving requested dataset	9 Initiating EEG dataset collection	10 Contacting subjects	11	12
13	14 EEG data recording	15 Analysing problems faced during EEG	16 Continuing with EEG data recording	17 Contacting more subjects	18 Reading artifacte removal documents EEG data recording	19 Going through artifact removal research papers
20	21 EEG data recording	22	23 Preparing status report for review 1	24 Verifying report with Guide	25 Submiting report	26 Preparing for review 1
27 Preparing for review 1	28 **REVIEW 1**	29 Team brainstorming session	30	31 Planning next step for capstone	1	2

Notes:

FEBRUARY 2019

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
27	28	29	30	31	1 Going through ICA algorithm EEG data recording	2 IEEE papers based on ICA EEG data recording
3	4 Reporting to Guide EEG data recording	5 Contacting more subjects for EEG	6 EEG data recording	7 Going through cubic interpolation EEG data recording	8 Feature extraction algorithms EEG data recording	9 IEEE papers based on EEG feature extraction
10	11 Studying python EEG modules EEG data recording	12	13 EEG data recording	14	15 Reporting to Guide EEG data recording	16 Installing all required python modules
17	18 Scripts for EEG EEG data recording	19 Initiate coding EEG data recording	20 Coding ANN architecure EEG data recording	21 Computing computational cost EEG data recording	22 Applying data structures EEG data recording	23
24	25	26 Reporting to Guide EEG data recording	27	28	1	2

Notes:

MARCH 2019

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
24	25	26	27	28	1 Report for review 2 EEG data recording	2 Verifying report with Guide EEG data recording
3	4 **REVIEW 2**	5 Team brainstorming session	6	7 Planning next step for capstone	8	9
10	11 Contacting more Female participants	12 EEG data recording	13 Completion of ML scripts EEG data recording	14 Implementing dataset correction algorithms	15 Implementing ICA algorithm EEG data recording	16 Testing ICA algorithm on EEG EEG data recording
17	18 Implementing cubic Interpolation EEG data recording	19 Testing interpolation on EEG EEG data recording	20 Initiating artifact removal EEG data recording	21	22	23
24	25 Completing Feature extraction EEG data recording	26 EEG data recording	27 Benchmarking results EEG data recording	28 Final Thesis documentation EEG data recording	29 Reporting to Guide EEG data recording	30 Completing Thesis EEG data recording
31	1	2	3	4	5	6

Notes:

APRIL 2019

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
31	1 EEG data recording	2 Verifying report with Guide EEG data recording	3 Making necessary changes in Thesis EEG data recording	4 Compiling all the results in Thesis	5 Submitting Thesis	6
7	8	9 Verification of Thesis from Capstone co-ordinator	10 Preparing for review	11 Preparing for review	12 **FINAL REVIEW**	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1	2	3	4

Notes: