AFFECTIVE STATE RECOGNITION THROUGH EEG SIGNALS FEATURE LEVEL FUSION AND ENSEMBLE CLASSIFIER

MOHAMMAD ABU SAJJAD

SHAMIM RAHMAN

AKASH PODDAR

A THESIS SUBMITTED

FOR THE DEGREE OF BACHELOR OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MILITARY INSTITUTE OF SCIENCE AND TECHNOLOGY

2019

SUPERVISOR'S APPROVAL

This thesis paper titled **"AFFECTIVE STATE RECOGNITION THROUGH EEG SIGNALS FEATURE LEVEL FUSION AND ENSEMBLE CLASSIFIER"**, submitted by 201614004 Mohammad Abu Sajjad, 201614005 Shamim Rahman and 201614051 Akash Poddar as mentioned below has been accepted as satisfactory in partial fulfillment of the requirements for the degree B.Sc. in Computer Science and Engineering in 2019.

Dr. Md. Mahbubur Rahman
Professor
Department of Computer Science and Engineering
Military Institute of Science and Technology

DECLARATION

This is to certify that the work presented in this thesis paper, titled, "AFFECTIVE STATE RECOGNITION THROUGH EEG SIGNALS FEATURE LEVEL FUSION AND ENSEMBLE CLASSIFIER", is the outcome of the investigation and research carried out by the following students under the supervision of Dr. Md. Mahbubur Rahman, Professor, Department of Computer Science and Engineering, Military Institute of Science and Technology.

It is also declared that neither this thesis paper nor any part thereof has been submitted anywhere else for the award of any degree, diploma or other qualifications.

Mohammad Abu Sajjad
Roll: 201614004
25 December 2019

Shamim Rahman
Roll: 201614005
25 December 2019

Akash Poddar
Roll: 201614051
25 December 2019

ABSTRACT

In the recent time the importance of emotion recognition has been extended and in this scope of several research works here a multimedia dataset for human emotion analysis is made ready for this paper. The electroencephalogram (EEG) of 100 participants are collected where they are given to watch one minute videos to create emotion in them. The videos with emotional tags have variety range of emotion including happy, sad, disgust, peaceful etc. The experimental stimuli collected are analyzed intensively. Interrelationship between the EEG signal frequencies and the ratings given by the participants are taken to consideration for study. Statistical feature extraction is applied on the dataset obtained and finally it is split to test and train set. Several machine learning algorithms are applied to determine the testing and training accuracy of data.

ACKNOWLEDGEMENT

We are thankful to Almighty Allah for his blessings for the successful completion of our thesis. Our heartiest gratitude, profound indebtedness and deep respect go to our supervisor, Dr. Md. Mahbubur Rahman, Professor, Department of Computer Science and Engineering, Military Institute of Science and Technology, for his constant supervision, affectionate guidance and great encouragement and motivation. His keen interest on the topic and valuable advices throughout the study was of great help in completing thesis. We are especially grateful to the Department of Computer Science and Engineering (CSE) of Military Institute of Science and Technology (MIST) for providing their all out support during the thesis work.

Finally, we would like to thank our families and our course mates for their appreciable assistance, patience and suggestions during the course of our thesis.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

Emotion is a mental state linked to the nervous system. Different emotions create different types of physiological signals which are generated from the brain. All these signals are called brain waves and can be distinguished by the features they carry. Thus the brain waves carry characteristics of the emotion and the emotional state of the brain or human can be identified by analysing the brain wave using reverse engineering.

Identification of emotion using brain waves can be utilized for different purposes. In medical science, this can be used to detect diseases, check the improvement of patients and also to detect the effectiveness of any special medicine.

1.1 Background and Motivation

In the industrial sector, the level of concentration of the workers and employees can be measured by the above system. By monitoring the state of anxiety of workers in various hazardous duties and also in the critical duties percentage of accidents can be reduced.

There had been much research in the field of emotion detection using physiological signal/visual image. Emotion detection using can be wrong by using visual images since the subject may not have all facial muscles similarly active like others and the subject may manipulate the result by using a fake facial expression.

All previous research in emotion detection using the physiological signals was done by utilizing a number of devices and sensors. All this research took place and need a lab environment.

In the proposed system emotion detection is done using physiological signals. The device is easy to operate and only two channels are used to read the brain waves. The device can be used in any environment and very easy to operate.

1.2 Research Problem

Human emotions create physiological signals which are generated from the brain. These are incorporated with thoughts, feelings, behavioral responses, and a degree of pleasure or displeasure [1] [2]. The emotions of ours can impel us to take action and dominate the decisions in

lives. Emotion is referred to as a list: anger, disgust, fear, joy, sadness and surprise [3]. Emotion detection is a technique used to read the emotions on a human face by using hi-tech image processing software. Human-computer interaction (HCI) is computer technology, focused on the interfaces between humans and computers [4]. There are different approaches of extensive scales of emotion, such like: Plutichik's emotion wheel [5], valence-arousal scale by Russell [6]. In this research this Russel model is applied. According to this process, each emotional state can be plotted into a two-dimensional plane where arousal and valence are represented by horizontal and vertical axes respectively. This system requires distinct features to be plotted in the plane.

There are many brain signals emitted from the brain and only very few are utilized in the above-stated system. In the proposed system in this paper total, seven types of signals are extracted from the device. Though all the signals do not carry distinct characteristics those can be analysed and the better system can be brought into the light.

1.3 Research Challenges

There had been many challenges the research team faced in different stages of the thesis. Few challenges are stated below:

- A noise-free lab environment is a prerequisite to get a strong dataset. For the data collection, a noise-free lab environment was not available.
- There are only two channels in the device to read the brain waves. This fact is a limitation for the thesis group as other devices got a number of brain wave reading channels. But it is otherwise a requirement of the research as this fact makes the thesis more acceptable in the industrial utilization of the workers.

1.4 Research Objectives

The objectives of our thesis are as following:

- To analyse the effect of different features on recognition of human emotion.
- To design affective domain classifier using feature level fusion.
- To propose Ensemble classifier for the classification of human emotion recognition.

1.5 Thesis Organization

The rest of the paper is organized as follows: Section 2 presents the related works in the relevant field, Section 3 talks about the environment which is created to ensure proper dataset collection and illustrates the total method followed in the process of data analysis, classification and accuracy result, Section 4 illustrates the performance of the result obtained through the experiment and finally Section 5 conclude the paper talking about the further expansion of the process, pitfalls and success of this research.

CHAPTER 2
DEFINITION AND RELATED WORKS

2.1 Definition of Important Terms

2.1.1 EEG

One of the most versatile brain imaging techniques is Electroencephalography (EEG). It records electrical activity and brain waves using electrodes placed on the scalp. Measuring electrical activity from the brain is useful because it reflects how the many different neurons in the brain network communicate with each other via electrical impulses. EEG has several benefits compared to other imaging techniques or pure behavioral observations. The most central benefit of EEG is its excellent time resolution, that is, it can take hundreds to thousands of snapshots of electrical activity across multiple sensors within a single second. This renders EEG an ideal technology to study the precise timecourse of cognitive and emotional processing underlying behavior

Few other reasons are [7]:

- EEG has very high time resolution and captures cognitive processes in the time frame in which cognition occurs.
- EEG directly measures neural activity.
- EEG is inexpensive, lightweight, and portable.
- EEG monitors cognitive-affective processing in absence of behavioral responses.

2.1.2 Band of Frequencies and Brain activities

The billions of neurons in the human brain have highly complex patters. The neural oscillations that can be measured with EEG are even visible in raw, unfiltered, unprocessed data. However, the signal is always a mixture of several underlying base frequencies. They are based on specific frequency ranges or frequency bands.

- **Delta Band (1-4 Hz):** The slowest and highest amplitude brainwaves, oscillations in the 1 – 4 Hz range are characterized as delta waves [8]. Delta waves are only present during deep

non-REM sleep (stage 3), also known as slowwave sleep (SWS). In sleep labs, delta band power is examined to assess the depth of sleep. The stronger the delta rhythm, the deeper the sleep. Delta frequencies are stronger in the right brain hemisphere, and the sources of delta are typically localized in the thalamus. Generally, this waves are examined for sleep and sleep disorders, alcoholism and sleep etc..

- **Theta Band (4-8 Hz):** Brain oscillations within the 4 – 8 Hz frequency range are referred to as theta band [8]. Studies consistently report frontal theta activity to correlate with the difficulty of mental operations, for example during focused attention and information uptake, processing and learning or during memory recall. Theta frequencies become more prominent with increasing task difficulty. Theta can be recorded from all over cortex. Generally, this waves are examined N-back task, Spatial navigation, Brain monitoring in operational environments etc.

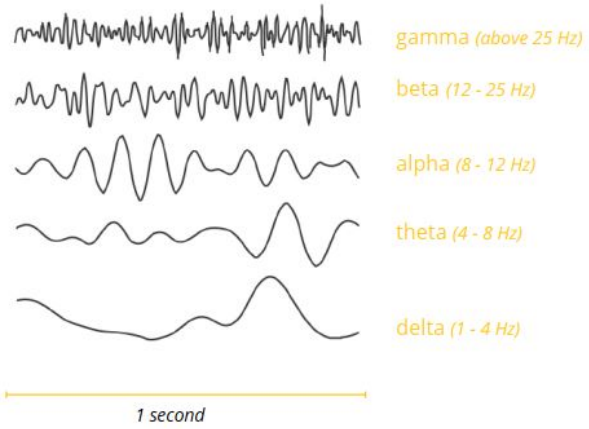


Figure 2.1: Sample Raw Frequencies of different Bands

- **Alpha Band (8-12 Hz):** First discovered by Hans Berger in 1929, alpha is defined as rhythmic oscillatory activity within the frequency range of 8 – 12 Hz [8]. Alpha is generated in posterior cortical sites, including occipital, parietal and posterior temporal brain regions. Alpha waves have several functional correlates reflecting sensory, motor and memory functions. Generally, Alpha wave is studied for examining meditation, Biofeedback training, Attention etc.

- **Beta Band (12-25 Hz):** Oscillations within the 12 – 25 Hz range are commonly referred to as beta band activity [8]. This frequency is generated both in posterior and frontal regions. Active, busy or anxious thinking and active concentration are generally known to correlate with higher beta power. Over central cortex (along the motor strip), beta power becomes stronger as we plan or execute movements, particularly when reaching or grasping requires fine finger movements and focused attention. Interestingly, this increase in beta power is also noticeable as we observe others' bodily movements.Generally, Beta wave is studied for understanding Motor control mechanisms, Stimulant-induced alertness etc.

- **Gamma Band (above 25 Hz):** At the moment, gamma frequencies are the black holes of EEG research as it is still unclear where exactly in the brain gamma frequencies are generated and what these oscillations reflect [8]. Some researchers argue that gamma, similar to theta, serves as a carrier frequency for binding various sensory impressions of an object together to a coherent form, therefore reflecting an attentional process. Others argue that gamma frequency is a by-product of other neural processes such as eye-movements and micro-saccades, and therefore do not reflect cognitive processing at all.

2.1.3 Statistical Features

- **Mean:** The mean (average) of a data set is found by adding all numbers in the data set and then dividing by the number of values in the set.
- **Median:** The median is the middle value when a data set is ordered from least to greatest.

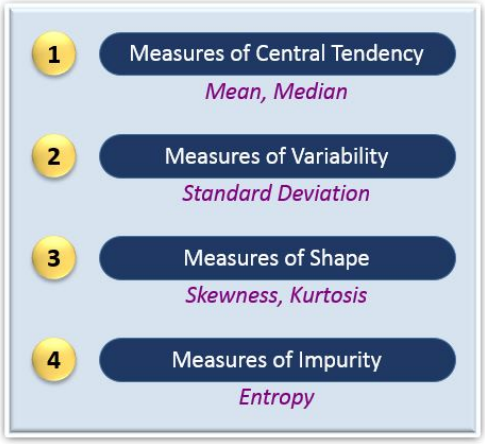


Figure 2.2: Statistical Features

- **Skewness:** Skewness refers to distortion or asymmetry in a symmetrical bell curve, or normal distribution, in a set of data. If the curve is shifted to the left or to the right, it is said to be skewed. Skewness can be quantified as a representation of the extent to which a given distribution varies from a normal distribution. A normal distribution has a skew of zero, while a lognormal distribution, for example, would exhibit some degree of right-skew.
- **Kurtosis:** Kurtosis is a statistical measure that is used to describe the distribution. Whereas skewness differentiates extreme values in one versus the other tail, kurtosis measures extreme values in either tail. Distributions with large kurtosis exhibit tail data exceeding the tails of the normal distribution (e.g., five or more standard deviations from the mean). Distributions with low kurtosis exhibit tail data that are generally less extreme than the tails of the normal distribution.

- **Standard Deviation:** Standard deviation is the measure of spread. It measures spread around the mean. Because of its close links with the mean, standard deviation can be greatly affected if the mean gives a poor measure of central tendency.

  Standard deviation is also influenced by outliers one value could contribute largely to the results of the standard deviation. In that sense, the standard deviation is a good indicator of the presence of outliers. This makes standard deviation a very useful measure of spread for symmetrical distributions with no outliers.

- **RMS:** Root mean square is a measure of the magnitude of a set of values. It is the square root of the arithmetic mean of the squares of the original values.

- **Entropy:** Entropy is a statistical model of the signal which can also provide physical information. Neurophysiological evidences show that it is possible to estimate the information from entropy as a measure of cortex performance. The change in entropy of EEG signals shows a real change in the function of the cerebral cortex. So, entropy, besides on a statistical measure of EEG pattern, in some cases, can reflect cortical information within the brain [9].

2.1.4 Feature Selection and Extraction

- **Definition:** Feature is an individual measurable property or characteristic of a phenomenon being observed. Choosing informative, discriminating and independent features is a crucial step for effective algorithms during classification. Again, if the number of features becomes similar than the number of observations stored in a dataset then this can most likely lead to a Machine Learning model suffering from overfitting. In order to avoid this type of problem, it is necessary to apply either regularization or dimensionality reduction techniques (Feature Extraction). In Machine Learning, the dimensionality of a dataset is equal to the number of variables used to represent it. Using Regularization could certainly help reduce the risk of overfitting, instead Feature Extraction techniques can also lead to other types of advantages such as:

  - Accuracy improvements.
  - Overfitting risk reduction.
  - Speed up in training.
  - Improved Data Visualization.
  - Increase in explainability of our model.

  Feature Extraction aims to reduce the number of features in a dataset by creating new features from the existing ones (and then discarding the original features). These new reduced set of features should then be able to summarize most of the information contained

in the original set of features. In this way, a summarised version of the original features can be created from a combination of the original set. Another commonly used technique to reduce the number of feature in a dataset is Feature Selection.

- **Differences:** The difference between Feature Selection and Feature Extraction is that feature selection aims instead to rank the importance of the existing features in the dataset and discard less important ones (no new features are created).

- **Feature Extraction Algorithms**

  - **Principle Components Analysis (PCA):** PCA is one of the most used linear dimensionality reduction technique. When using PCA, we take as input our original data and try to find a combination of the input features which can best summarize the original data distribution so that to reduce its original dimensions. PCA is able to do this by maximizing variances and minimizing the reconstruction error by looking at pair wised distances. In PCA, our original data is projected into a set of orthogonal axes and each of the axes gets ranked in order of importance. PCA is an unsupervised learning algorithm, therefore it doesn't care about the data labels but only about variation. This can lead in some cases to misclassification of data.
  - **Linear Discriminant Analysis (LDA):** LDA is supervised learning dimensionality reduction technique and Machine Learning classifier. LDA aims to maximize the distance between the mean of each class and minimize the spreading within the class itself. LDA uses therefore within classes and between classes as measures. It is a good choice because maximizing the distance between the means of each class when projecting the data in a lower-dimensional space can lead to better classification results.In LDA, it is assumed that the input data follows a Gaussian Distribution otherwise it may possibly lead to poor classification results.
  - **t-distributed Stochastic Neighbor Embedding (t-SNE):** t-SNE is non-linear dimensionality reduction technique which is typically used to visualize high dimensional datasets.t-SNE works by minimizing the divergence between a distribution constituted by the pairwise probability similarities of the input features in the original high dimensional space and its equivalent in the reduced low dimensional space. t-SNE makes then use of the Kullback-Leiber (KL) divergence in order to measure the dissimilarity of the two different distributions. The KL divergence is then minimized using gradient descent. In t-SNE, the higher dimensional space is modelled using a Gaussian Distribution, while the lower-dimensional space is modelled using a Student's t-distribution. This is done, in order to avoid an imbalance in the neighbouring points distance distribution caused by the translation into a lower-dimensional space.
  - **Short-term Fourier transform (STFT):** Basically, STFT extracts several frames of the original signal to be analyzed with a window frame that shifts with time. If the

time window utilized, is narrow enough, each frame extracted could be evaluated as stationary which enables using Fourier transform (FT). STFT calculates a time-varying spectrum by implementing the Discrete Fourier transform (DFT) to a framed part of the data and by moving the frame through the whole record. With this approach, the spectral characteristics are processed as constant for each constant frame time, but catches the switches in the spectral characteristics using various, possibly overlapping, frame locations throughout the time. The magnitude squared of STFT yields the spectrogram of the function. STFT spectrogram is the normalized, squared magnitude of the STFT coefficients produced by the STFT

- **Discrete Wavelet Transform (DWT):** DWT is the process of transforming time signal to a discrete wavelet representation. For processing Biological signals such as EEG in discrete wavelet transform digital filtering is used. DWT algorithm gives octave-scale frequency as well as spatial timing of the given signal. For this reason it is always used to address and resolve many sophisticated and complicated issues. In this process, Low Pass Filter (LFP) and High Pass Filter (HPF) are extracted. Scaling function can be acquired from the LPF while the wavelet function can be acquired from the HPF. The filter bank level varies with the availability of the bandwidth. The high frequency is analyzed by HPF and the low frequency is analyzed by LPF.
- **Discrete Cosine Transform (DCT):** DCT is used primarily for signal extraction features. In terms of a sum of sinusoids at multiple frequencies and amplitudes, it sends a signal like any linked Fourier transform. It was first introduced in 1974 in a research paper of Nasir Ahmed et al. In 1977 Wen-Hsiung Chen paved the way of this algorithm by developing it to work very fast. In 1987, Peincen, Johnson and Bradley suggested further adapted DCT. It is a way for transforming a frequency into different frequencies for summation of cosine functions. In this method the relevant coefficients of a signal is transformed from a whole signal. It performs energy compaction and decorrelate the data for the image. After decorrelation of every data, the coefficient can be cipher individualistic. The transformed signal is categorized in low, mid and high frequency each contains different details and information about the signal.
- **Fast Fourier Transform (FFT):** FFT is a kind of transforming from the basic Fourier that is much quicker. This is used to turn a signal from the time domain into a frequency domain. The inverse FFT does the opposite conversion. FFT is mostly used for signal processing as it consumes significantly less time than other feature extraction methods. It is an efficient algorithm of the (FT). It takes N2 multiplication to process a signal where the FFT takes only Nlog2(N). It is obtained by reducing the multiplication needed in the Fourier transform. The transformation of Fourier from N discrete points can be written as two N/2 discrete points. Thus if the number N is a power of two it is possible to divide the points until there is only one point left. It

greatly reduces the 18time complexity.

- **Wigner-Ville distribution (WVD):** The distribution of wigner implies, that the signal is split in a left and a right portion in relation to time t, and the right portion is folded over the left portion (equal to the spectrum because in both domains it is formally equal). The result is that, if the signal is always 0 before t or is always null after t, the Wigner distribution is undoubtedly 0 at t time, so the weak form is respected. For such a phenomenon, the word interference or cross term is used

- **Feature Selection Methods**

  In our study we have consulted XGBoost which stands for eXtreme Gradient Boosting. It is generally used for two reasons. Firstly, to improve the execution speed and secondly, to improve the model performance. We have also consulted mRMR (minimum Redundancy Maximum Relevance) feature selection method.

2.1.5 Model Validation

- **Definition:** In machine learning, model validation is referred to as the process where a trained model is evaluated with a testing data set. The testing data set is a separate portion of the same data set from which the training set is derived. The main purpose of using the testing data set is to test the generalization ability of a trained model [10]. We have included following methods:

  - **Hold Out:** The holdout method is the simplest kind of cross validation. The data set is separated into two sets, called the training set and the testing set. The function approximator fits a function using the training set only. Then the function approximator is asked to predict the output values for the data in the testing set (it has never seen these output values before). The errors it makes are accumulated as before to give the mean absolute test set error, which is used to evaluate the model. The advantage of this method is that it is usually preferable to the residual method and takes no longer to compute. However, its evaluation can have a high variance. The evaluation may depend heavily on which data points end up in the training set and which end up in the test set, and thus the evaluation may be significantly different depending on how the division is made [11].

  - **K-Fold:** K-fold cross validation is one way to improve over the holdout method. The data set is divided into k subsets, and the holdout method is repeated k times. Each time, one of the k subsets is used as the test set and the other k-1 subsets are put together to form a training set. Then the average error across all k trials is computed. The advantage of this method is that it matters less how the data gets divided. Every data point gets to be in a test set exactly once, and gets to be in a training set k-1 times. The variance of the resulting estimate is reduced as k is increased. The disadvantage

of this method is that the training algorithm has to be rerun from scratch k times, which means it takes k times as much computation to make an evaluation. A variant of this method is to randomly divide the data into a test and training set k different times. The advantage of doing this is that you can independently choose how large each test set is and how many trials you average over [11].

- **Leave One Out:** Leave-one-out cross validation is K-fold cross validation taken to its logical extreme, with K equal to N, the number of data points in the set. That means that N separate times, the function approximator is trained on all the data except for one point and a prediction is made for that point. As before the average error is computed and used to evaluate the model. The evaluation given by leave-one-out cross validation error (LOO-XVE) is good, but at first pass it seems very expensive to compute. Fortunately, locally weighted learners can make LOO predictions just as easily as they make regular predictions. That means computing the LOO-XVE takes no more time than computing the residual error and it is a much better way to evaluate models [11].

2.1.6 Classification of States

- **ML Algorithms:** There are different kinds of ML algorithms available. These are classified primarily into three main branches. Supervised, unsupervised and reinforced learning. We have used different ML algorithms to find-out higher accuracy results. They are: GradientBoostingClassifier, RandomForestClassifier, PassiveAggressiveClassifier, XGBClassifier, Perceptron, ExtraTreesClassifier, RidgeClassifierCV, BaggingClassifier, KNeighborsClassifier, LinearSVC, DecisionTreeClassifier, AdaBoostClassifier, GaussianNB, SGDClassifier, LogisticRegressionCV, GaussianProcessClassifier, BernoulliNB, SVC, NuSVC.

2.1.7 Ensemble Learning

In statistics and machine learning, ensemble methods use multiple learning algorithms to obtain better predictive performance than could be obtained from any of the constituent learning algorithms alone. An ensemble is itself a supervised learning algorithm, because it can be trained and then used to make predictions. The trained ensemble, therefore, represents a single hypothesis. This hypothesis, however, is not necessarily contained within the hypothesis space of the models from which it is built. Thus, ensembles can be shown to have more flexibility in the functions they can represent. ensembles tend to yield better results when there is a significant diversity among the models. Many ensemble methods, therefore, seek to promote diversity among the models they combine. In our study, we have included Max Voting technique.

- **Max Voting:** The max voting method is generally used for classification problems. In this technique, multiple models are used to make predictions for each data point. The

predictions by each model are considered as a 'vote'. The predictions which we get from the majority of the models are used as the final prediction.

2.2 Related Works

Currently, researches are conducted on emotion recognition or classification based on physiological signals or speech or face expression image processing.

Koelstra et al. in [12] has formulated a vast DEAP dataset based on EEG and peripheral physiological signals of 32 participants where each of them watched several videos. They worked with classification of arousal, valence, like/dislike ratings of the participants. Mavani et al. [13] in his research has worked on CFEE and RaFD datasets improving the existing CNN model and could reach accuracy of 65.39%.

Another work based on physiological signals like EDA, PPG, zEMG are fused to differentiate 5 major emotion classes using Fina Gaussian Support Vector Machine in [14]. Apart from that to extract features, they applied Deep Believe Network(DBN). They were able to reach at 89.53% accuracy. Tripati et al. [15] have also worked on DEAP dataset classifying emotion using EEG signals using both Deep Neural Network and Convolutional Neural Network (CNN) proving effectiveness over existing work on this arena. Another research on signal based emotion recognition is done by Alam et al. on [16] using EMG, EDA, ECG sensors are analyzed using deep CNN with accuracy of 87.5%. Meanwhile Mohammadi et al. in [17] used SVM and K-nearest neighbor classifiers to detect emotion using 10-channel EEG signal. Though researchers have worked with various sensors but their quality and emotion detection capabilities had a great doubt in research work. To check the grade and potentiality of lab based and wearable sensors Ragot et al. in [18] compared their accuracy. However this experiment proved reliability of both types of sensors.

Zhuang et al. in [19] used EMD and EEG signals for feature extraction and emotion recognition decomposing into empirical mode decomposition (EMD). Their multidimensional information is used as features. The researchers have checked their accuracy of classification comparing with several classical techniques, including fractal dimension (FD), sample entropy,differential entropy, and discrete wavelet transform (DWT). Though many research work has been done on EEG signals but its stability over time was a great question. The frequently used popular feature extraction, feature selection, feature smoothing and pattern classification methods are analyzed and evaluated in [20] using public data-set DEAP and their own developed data-set SEED. The emotion recognition model shows that the neural patterns are relatively stable within and between session. Xia et al. in [21] through their research has enforced activation and valence information for acoustic emotion recognition applying multi-task learning based on DBN. Besides, they have enforced activation and valence in two different ways: category level based classification and continuous level based regression. The fusion of the loss functions from both tasks is used as the

objective function in the multi-task learning framework. After iterative optimization, the values from the last hidden layer in the DBN are used as new features and made input into a support vector machine (SVM) classifier for emotion recognition.

Though conventional methods of using EEG signals for emotion recognition skips the use spatial characteristics of EEG signals which contain various salient features, Chao et al. in [22] have used frequency domain, spatial characteristics, and frequency band characteristics of the multichannel EEG signals to build up multiband feature matrix (MFM). Based on input MFM, a capsule network (CapsNet) classifies emotion states. Ullab et al. in [23] has proposed an ensemble learning algorithm for automatically computing the most discriminate subset of EEG channels for internal emotion recognition. This method describes an EEG channel using kernel-based representations computed from the training EEG recordings. This algorithm reduces the amount of data along with improving computational efficiency and classification accuracy at the same time. A complete different arena of emotion detection is represented on [24]. It works with emotion detection from surveillance camera video. The body shape and gesture from video identifies the emotion which is done on basis of publicly available data-set and modeled by Support Vector Machine (SVM) with accuracy of 93.39%. [25] performs continuous emotion prediction on three dimensions: Arousal, Valence, Likability based on audiovisual signals. They have measured the contrast between effectiveness of non-temporal model SVR and temporal model LSTM-RN.

Li et al. in [26] have developed a series of EEG Multidimensional Feature Image(EEG MFI) sequence from spacial characteristics, frequency domain and temporal characteristics mapping them into two-dimensional image. Besides, they have also constructed hybrid deep neural network along with CNN, LSTM, RNN to to work out with EEG MFI sequence to classify human emotion and finally got accuracy over 75.21%.

To overcome problem of learning efficiency and computational complexity of Deep Neural Network(DNN), [27] has worked on Softmax regression-based deep sparse auto encoder network(SRDSAN). Their proposed process has overcome local extreme and gradient diffusion problems and vigorously of neural network. They proved better efficiency and accuracy here.

Human facial expression is not the best way to determine human emotion. This is because, a person might be mentally sad but showing false happy emotional expression through his face. Thus working with facial emotion recognition might decrease the accuracy rate of the work. The best way to work with this problem is physiological signals. Brain impulse are used here which is EEG signal and it generates different types of signals for different types of emotions. As a result this process of emotion classification is more trustworthy and provides better accuracy than other processes. At the same time, we have tried to findout the accuracy of classifying emotion correctly only using two channels. Here the four emotion states funny, sad, disgust and peaceful are considered and customized dataset is collected of 100 persons. Based on the dataset statistical feature extraction is done and finally based on various machine learning algorithms the

accuracy result of testing is checked.

CHAPTER 3
METHODOLOGY

3.1 Data Collection

We have prepared our own dataset for the analysis of human affective states. The electroencephalogram (EEG) signals of 100 participants were recorded as each watched one-minute long videos of four different subjects. The objective was to record the values for four different emotion state.

- **Stimuli Selection:** The stimuli used in the experiment were selected in several steps. First, we have selected 15 initial stimuli manually. Then, a one-minute highlight part was determined for each stimulus. Finally, through the web-review process we have examined the content type. The videos were downloaded from famous YouTube channels. Basing on the result of the google engine for various tags, we have selected the most rated ones.

  The valence-arousal space to work on emotion classification can be split into 4 regions: low arousal/low valence(LALV), low arousal/high valence(LAHV), high arousal/low valence(HALV) and high arousal/high valence(HAHV) as illustrated in Fig 3.1.

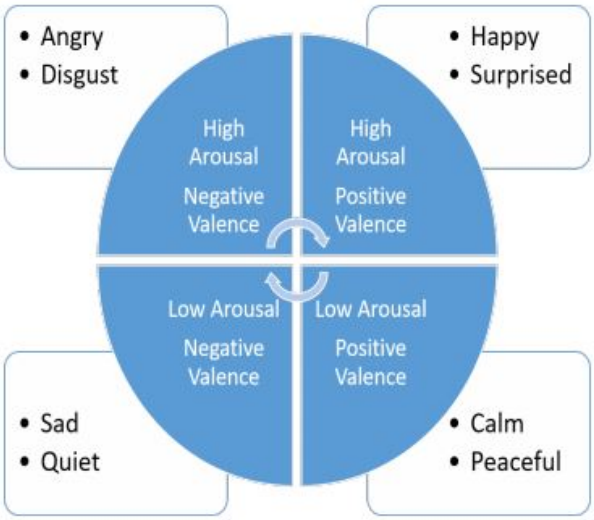


Figure 3.1: Arousal Valence Scale of Emotion

- **Distinguishing one minute climax:** Based on linear regression method proposed in [31],arousal and valence in each movies was determined. Loudness and energy of the audio signals,

motion component, visual excitement and shot duration were considered for arousal calculation. The regressors were trained using dataset of [31] for better valence and arousal estimation. Using the concept of Relevance Vector Machine(RVM) from RVM toolbox from [32] which is capable of refusing the vague features throughout the training process. The emotional climax score of i-th segment ei was calculated using the equation where ai and vi are arousal and valence respectively.

- **Equipment Used:** Mindwave Mobile 2. The electric signal emitted by neurons fired in the brain has different patterns and frequencies. This signals are measured by placing sensors in the scalp. This Mindwave NeuroSky Gear measures the analog electrical signals, commonly referred to as brainwaves, and processes them into digital signals. The frequencies that Mindwave can read are shown in the figure 3.2:

Brainwave Type	Frequency range	Mental states and conditions
Delta	0.1Hz to 3Hz	Deep, dreamless sleep, non-REM sleep, unconscious
Theta	4Hz to 7Hz	Intuitive, creative, recall, fantasy, imaginary, dream
Alpha	8Hz to 12Hz	Relaxed (but not drowsy) tranquil, conscious
Low Beta	12Hz to 15Hz	Formerly SMR, relaxed yet focused, integrated
Midrange Beta	16Hz to 20Hz	Thinking, aware of self & surroundings
High Beta	21Hz to 30Hz	Alertness, agitation

Figure 3.2: Mindwave Mobile2-Channel Types

This device has a sensor that touches the forehead, has contact points in the ear clip and on board chip that processes all the data. It has proprietary algorithms for characterizing mental states. During calculation of the data, it amplifies the raw brainwave signal and removes the ambient noise and muscle movement.

Experimental Setup: Mindwave Mobile2 apk for android is setup and used for the collection of the signals/data from EEG sensors. Since our study focuses on classifying four different emotions, therefore finally selected four stimuli videos were shown individually to the viewers. Each video continued for 60 seconds. We have collected the emotions of 100 different individuals. So the total number of data is 60 x 4 x 100=24000. The notations we have followed for the emotions are as shown in the table 3.1. In the raw data there had been few columns which does not have any significance with our research interest. Therefore primarily we have considered 12 columns (features). So the dimension of our dataset has become 24,000 x 12 = 2,88,000 values.

A catplot visualization of the raw collected data is shown in the fig 3.3

Table 3.1: Notation of Emotion

Label	Emotion
1	Disgust
2	Funny
3	Peaceful
4	Sad

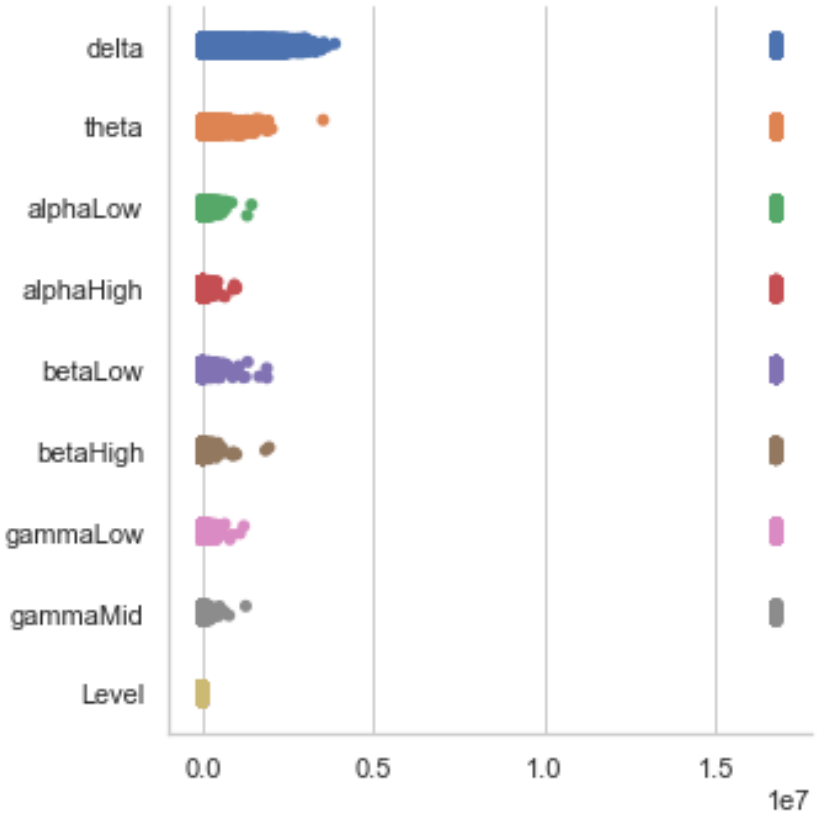


Figure 3.3: Raw Data Visualization

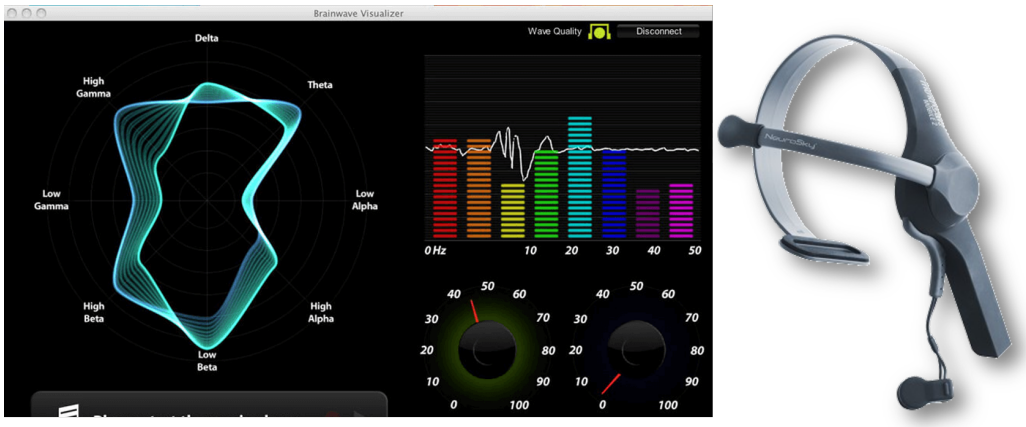
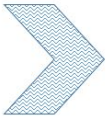


Figure 3.4: Mindwave Mobile2

Figure 3.5: Steps followed for Preprocessing

Serial	Initial Features
1.	eegRawValue
2.	Attention
3.	Meditation
4.	blinkStrength
5.	Delta
6.	Theta
7.	alphaLow
8.	alphaHigh
9.	betaLow
10.	betaHigh
11.	gammaLow
12.	gammaMid



Selection of Features

Serial	Selected Features
1.	Delta
2.	Theta
3.	alphaLow
4.	alphaHigh
5.	betaLow
6.	betaHigh
7.	gammaLow
8.	gammaMid

Figure 3.6: Selected Features

3.2 Data Preprocessing

After the completion of data collection, the next step that have been followed is preprocessing the data. Data re-scaling is an important part of data preparation before applying machine learning algorithms. It enhance the result and reduces unnecessary data. Therefore the steps followed by the group is as shown in the fig 3.5.

- **Selection of Significant Features:** In this step we have basically reduced the feature size for our experiment. The features which shows significance with the research topic we have kept those for detail investigation. Out of initial 12 features we have found relevance of 8 features as shown in the fig 3.6. So after this step our data size dimension has reduced to (row x column) = 24,000 x 8 = 1,92,000 values.
- **Reduction of Over fitting Data:** In this step we have removed the unnecessary data which are identified as irrelevant basing on their values.
- **Normalization:** Our data may contain attributes with a mixtures of scales for various quantities. Since many machine learning methods expect or are more effective if the data

$$z = \frac{x \text{ - } \min(x)}{\max(x) \text{ - } \min(x)}$$

Figure 3.7: Normalization

attributes have the same scale therefore, we have used data normalization process. Here, due to normalization, All the values are between 0 and 1, and the outliers are removed as well. All our features are more consistent with each other, which will allow us to evaluate the output of our future models better. This is done following the formula shown in formula 3.7 and Algorithm A.1.

3.3 Feature Extraction

- **Statistical Feature Selection:** After the following statistical feature applied to the dataset, total number of features stands 8(raw features) x 7(statistical features) = 56 (7 Statistical feature applied to each raw feature). So, now for one type of emotion one person has 56 features. Similarly, for one emotion 100 persons has 100 x 56 emotions. It means that for 4 different emotions the dimension has become (row x column) 400 x 56 = 22,400 values.
  - **Mean:** The arithmetic mean μ is the average of the values $x_1, x_2....x_m$ located within a time window. It was calculated by equation 3.1

$$\mu = \frac{1}{m}\sum_{i=1}^{m} x_i \tag{3.1}$$

  - **Median:** The median is the middle value when a data set is ordered from least to greatest. For n number of data it is the value of the median as calculated using the equation 3.2.

$$Median = (\frac{n+1}{2})th \tag{3.2}$$

  - **Skewness:** Skewness refers to distortion or asymmetry in a symmetrical bell curve, or normal distribution, in a set of data. If the curve is shifted to the left or to the right, it is said to be skewed. Skewness can be quantified as a representation of the extent to which a given distribution varies from a normal distribution. A normal distribution has a skew of zero, while a lognormal distribution, for example, would exhibit some degree of right-skew. It is calculated using equation 3.3

$$Skewness = E[\frac{(x-\mu)^3}{\sigma}] \tag{3.3}$$

Where E is the expectancy, μ is the mean and σ is the standard deviation.

- **Kurtosis**(K)**:** Kurtosis is a statistical measure that is used to describe the distribution. Whereas skewness differentiates extreme values in one versus the other tail, kurtosis measures extreme values in either tail. Coefficients of EEG signal do not follow the normal distribution, and have a heavy tail characteristic which is justified by the value of kurtosis parameters. Positive kurtosis indicates a relatively peaked distribution whereas negative kurtosis indicates a relatively flat distribution. It is calculated using equation 3.4.

$$K(s) = E(s^4) - 3E(s^2)^2 \tag{3.4}$$

Where s is the signal and E is the statistical expectation function of s. Kurtosis characterizes the relative peakedness of a distribution compared with the normal distribution.

- **Standard Deviation:** Standard deviation is the measure of spread. It measures spread around the mean. Because of its close links with the mean, standard deviation can be greatly affected if the mean gives a poor measure of central tendency. It is calculated using the following equation 3.5

$$STD = \sqrt{\frac{1}{N-1}\sum_{n=1}^{N}(X[n] - \frac{1}{N}\sum_{n=1}^{N}X[n])^2} \tag{3.5}$$

Standard deviation is also influenced by outliers one value could contribute largely to the results of the standard deviation. In that sense, the standard deviation is a good indicator of the presence of outliers. This makes standard deviation a very useful measure of spread for symmetrical distributions with no outliers.

- **RMS:** Root mean square is a measure of the magnitude of a set of values. It is the square root of the arithmetic mean of the squares of the original values. It is calculated using the equation 3.6

$$RMS = \sqrt{\frac{\sum_{i=1}^{m}(x_i)^2}{m}} \tag{3.6}$$

- **Entropy:** Entropy is a statistical model of the signal which can also provide physical information. It measures the signal complexity and and quantify regularity and order in the signal. It is observed that low entropy value of EEG signals represents less number of dominating process and the EEG signals with high entropy represent large number of dominating processes. Following equation 3.7 is followed for the

calculation.

$$E_n(n) = -\sum_{k=n}^{n+N} x(k) log_2 x(k) \tag{3.7}$$

After applying the above features the present dataset is saved for further cross references.

- **Advanced Features:** For better outcome calculation and performance evaluation, further study has been conducted for advanced features measurement. Following five different models are calculated using the algorithms shown below. Thereby, total 5 and other 3 (mixture of five features) total 8 features are taken. And each feature has 8 different sub features. So, after considering the advanced features over our raw dataset the dimension has stand to (row x column) 400 (for 4 emotions) x 64 = 25,600 values.

  - **Short-term Fourier transform (STFT):** Basically, STFT extracts several frames of the original signal to be analyzed with a window frame that shifts with time. If the time window utilized, is narrow enough, each frame extracted could be evaluated as stationary which enables using Fourier transform (FT). STFT calculates a time-varying spectrum by implementing the Discrete Fourier transform (DFT) to a framed part of the data and by moving the frame through the whole record. With this approach, the spectral characteristics are processed as constant for each constant frame time, but catches the switches in the spectral characteristics using various, possibly overlapping, frame locations throughout the time. The magnitude squared of STFT yields the spectrogram of the function. STFT spectrogram is the normalized, squared magnitude of the STFT coefficients produced by the STFT. For analysis and modelling of slowly changing signals, Fourier analysis have been widely used. To study frequency characteristics at moments t STFT is examined. Conversely, at a given frequency it is also possible to examine the time characteristics. In such cases, the spectrum is windowed and $S(w)$ if found, similarly the time transform with the frequency window function $H(w)$ is also calculated which provides inverse Fourier [28]. We get equation 3.8.

$$S_\omega(t) = \frac{1}{\sqrt{2\pi}} \int e^{j\omega' t} S(\omega') H(\omega - \omega') d\omega' \tag{3.8}$$

  when the window feature is related with time $h(t)$ to the frequency $H(w)$ window feature then it shows equation 3.9.

$$H(\omega) = \frac{1}{\sqrt{2\pi}} \int h(t) e^{-j\omega' t} dt \tag{3.9}$$

  - **Discrete Wavelet Transform (DWT):** DWT is the process of transforming time signal to a discrete wavelet representation. For processing Biological signals such as EEG in discrete wavelet transform digital filtering is used. DWT algorithm gives

octave-scale frequency as well as spatial timing of the given signal. For this reason it is always used to address and resolve many sophisticated and complicated issues. In this process, Low Pass Filter (LFP) and High Pass Filter (HPF) are extracted. Scaling function can be acquired from the LPF while the wavelet function can be acquired from the HPF. The filter bank level varies with the availability of the bandwidth. The high frequency is analyzed by HPF and the low frequency is analyzed by LPF [28]. Signal resolution is found by the filtering and scaling process. It is calculated using equation 3.10.

$$y[n] = (x * g)[n] = \sum_{k=-\infty}^{\infty} x[k]g[n-k] \tag{3.10}$$

Here x an input signal. This means that a series of filters can be used to measure the DWT of a signal x. Here g is impulse response of sample passing through LPF and h is the decomposition of the signal through HPF. n is the level of transform and the k is the type of the transformation. The detailed coefficient is found from the HPF and the approximation coefficients can be found from the LPF. These filters are also processed by another filters to step for achieve final signal resolution [28].

– **Discrete Cosine Transform (DCT):** DCT is used primarily for signal extraction features. In terms of a sum of sinusoids at multiple frequencies and amplitudes, it sends a signal like any linked Fourier transform. It was first introduced in 1974 in a research paper of Nasir Ahmed et al. In 1977 Wen-Hsiung Chen paved the way of this algorithm by developing it to work very fast. In 1987, Peincen, Johnson and Bradley suggested further adapted DCT. It is a way for transforming a frequency into different frequencies for summation of cosine functions. In this method the relevant coefficients of a signal is transformed from a whole signal. It performs energy compaction and decorrelate the data for the image. After decorrelation of every data, the coefficient can be cipher individualistic. The transformed signal is categorized in low, mid and high frequency each contains different details and information about the signal.

– **Fast Fourier Transform (FFT):** FFT is a kind of transforming from the basic Fourier that is much quicker. This is used to turn a signal from the time domain into a frequency domain. The inverse FFT does the opposite conversion. FFT is mostly used for signal processing as it consumes significantly less time than other feature extraction methods. It is an efficient algorithm of the (FT). It takes N2 multiplication to process a signal where the FFT takes only $Nlog2(N)$. It is obtained by reducing the multiplication needed in the Fourier transform. The transformation of Fourier from N discrete points can be written as two $\frac{N}{2}$ discrete points. Thus if the number N is a power of two it is possible to divide the points until there is only one point left. It greatly reduces the time complexity. If the even points are taken by dividing the

points by two then the equation should be as following equation 3.11. The algorithm we have followed is Algorithm A.2.

$$X[k] = \sum_{n=0evenn}^{N-1} x(n) e^{\frac{-2\pi jnk}{N}} \tag{3.11}$$

Here, the transform points are $x(n)$ and the amount of points is N. And other half of the equation would be

$$X[k] = \sum_{n=0oddn}^{N-1} x(n) e^{\frac{-2\pi jnk}{N}} \tag{3.12}$$

If both of the equations are added and let the X of even points be $X1$ and the odd point $X2$ then the equation becomes

$$X[k] = X_1(k) + e^{\frac{-2\pi jnk}{N}} X_2(k) \tag{3.13}$$

Thus N point discrete Furrier transform can be obtained from two $N/2$ transforms.

– **Wigner-Ville distribution (WVD):** This study group has added parts of the signal product to get Wigner distributed multiplied at some point at a certain point at a certain time by the signal, since past was equal to future. So we mentally fold the left part of the signal to the right and see if there are a simultaneous overlaps for determining Wigner distribution properties. If there is, then at the time t, those properties will now be present. Wigner Vile distribution compared the information of the signal with its own information at other times and frequencies. If this easy point is kept in mind, the distribution of Wigner becomes apparent with many problems and outcomes. For the frequency domain, Wigner distribution in both domains is essentially identical. Another significant point is that the distribution of Wigner is equally weighing the distant times to the close moments. The distribution of Wigner is therefore extremely non local. If a signal is $s(t)$, the corresponding Wigner distribution equation 3.14 becomes

$$W(t,w) = \frac{1}{2\Pi} \int_{-\infty}^{\infty} S^{\star}(t - \frac{1}{2}\tau) S(t + \frac{1}{2}\tau) \exp^{-\imath\tau\omega} dx \tag{3.14}$$

The distribution of wigner implies, that the signal is split in a left and a right portion in relation to time t, and the right portion is folded over the left portion (equal to the spectrum because in both domains it is formally equal). The result is that, if the signal is always 0 before t or is always null after t, the Wigner distribution is undoubtedly 0 at t time, so the weak form is respected. For such a phenomenon, the word interference or cross term is used [28].

After applying the advanced features we have brought out entropy for all the advanced

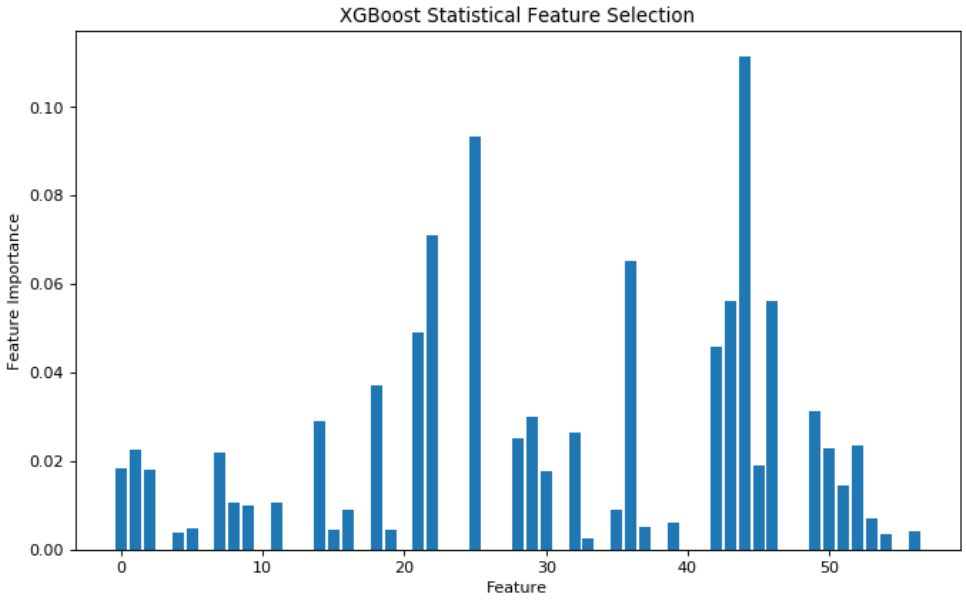


Figure 3.8: Important Statistical Features using XGBoost

features. Which are later used for further calculation and examinations.

3.4 Feature Selection

- **XGBoost:** XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. It implements machine learning algorithms under the Gradient Boosting framework. XGBoost provides a parallel tree boosting (also known as GBDT, GBM) that solve many data science problems in a fast and accurate way.

  After applying the algorithm A.2 in the Statistical features data following result is observed as shown in figure 3.8. Figure 3.9 shows the important features after applying the algorithm in the Advanced data.

  The statistical important features according to the f-score are as shown in figure 3.10 and in figure 3.11.

- **mRMR:** The mRMR is a feature selection approach that tends to select features with a high correlation with the class (output) and a low correlation between themselves. For continuous features, the F-statistic can be used to calculate correlation with the class (relevance) and the Pearson correlation coefficient can be used to calculate correlation between features (redundancy). Thereafter, features are selected one by one by applying a greedy search to maximize the objective function, which is a function of relevance and redundancy. Two commonly used types of the objective function are MID (Mutual Information Difference criterion) and MIQ (Mutual Information Quotient criterion) representing the difference or the quotient of relevance and redundancy, respectively. For temporal

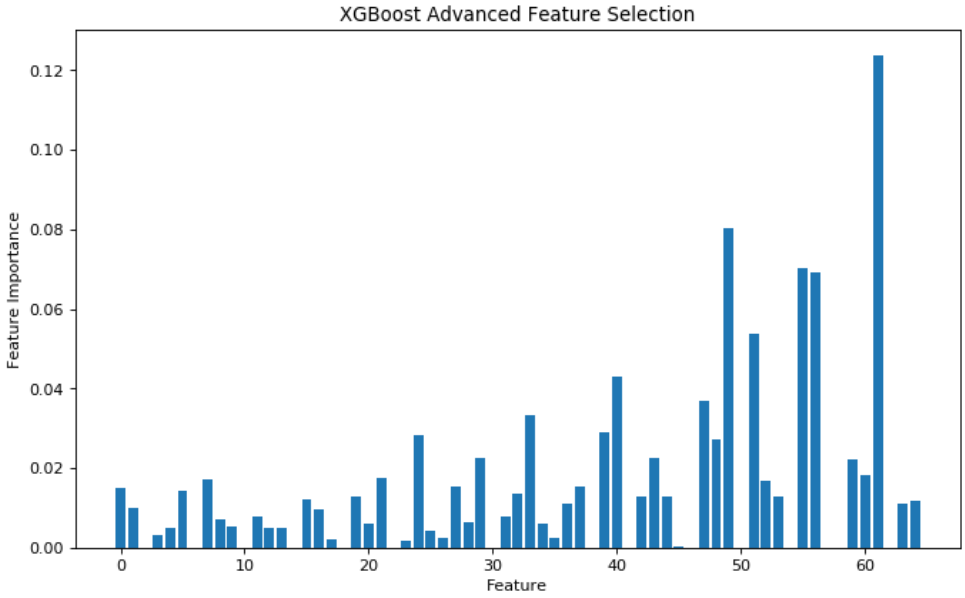


Figure 3.9: Important Advanced Features using XGBoost

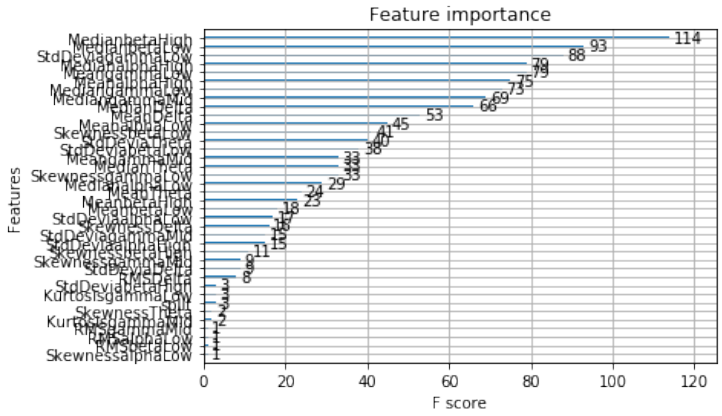


Figure 3.10: Important Statistical Features as per F-Score

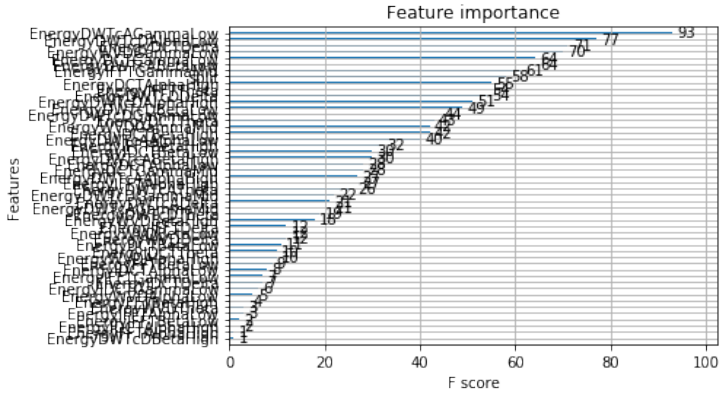


Figure 3.11: Important Advanced Features as per F-Score

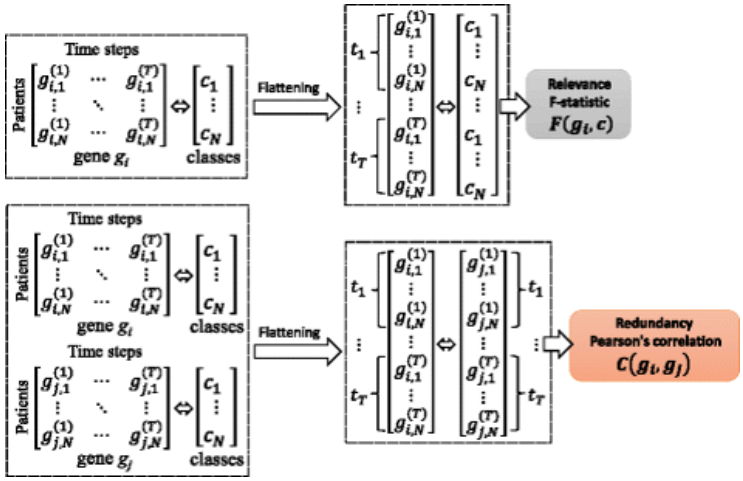


Figure 3.12: Data flattening commonly used as a preprocessing step to the mRMR

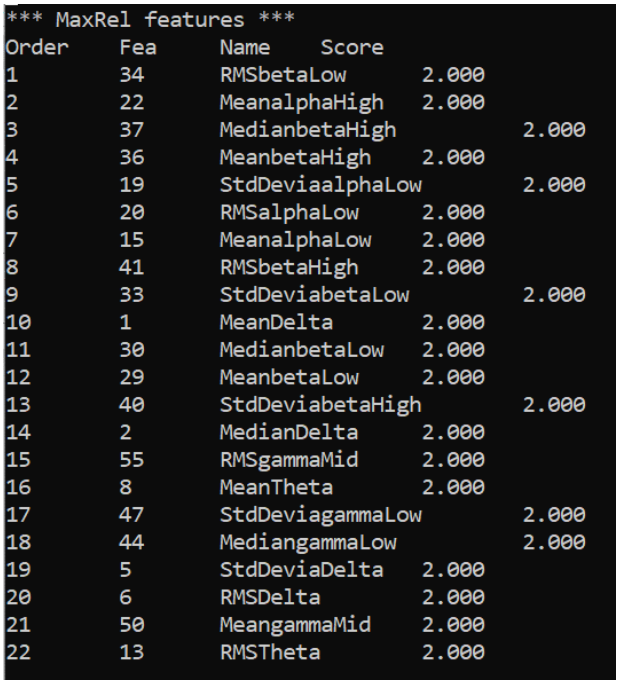
data, mRMR feature selection approach requires some preprocessing techniques that flatten temporal data into a single matrix in advance. This may result in a loss of possibly important information among temporal data (such as temporal order information). A common way for data flattening used as a preprocessing step to mRMR is depicted in figure 3.12. The Algorithm A.2 shows the steps followed.

After applying the mRMR algorithm to our statistical and advanced features the relevant features are identified as shown in figure 3.13 and 3.14 respectively.

3.5 Dimensionality Reduction

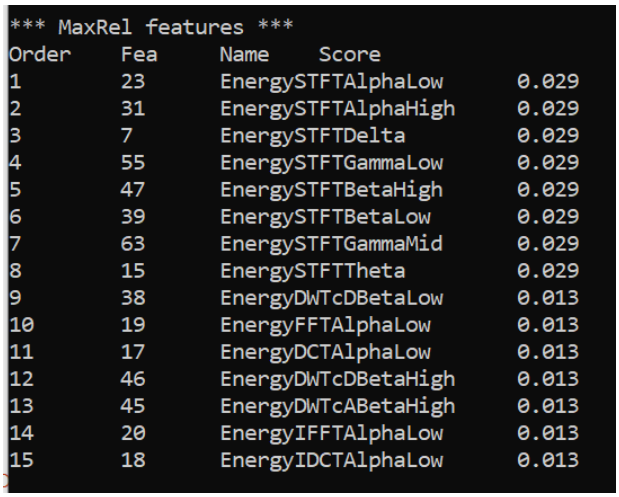
- **Principle Components Analysis (PCA):** PCA is one of the most used linear dimensionality reduction technique. When using PCA, we take as input our original data and try to find a combination of the input features which can best summarize the original data distribution so that to reduce its original dimensions. PCA is able to do this by maximizing variances and minimizing the reconstruction error by looking at pair wised distances. In PCA, our original data is projected into a set of orthogonal axes and each of the axes gets ranked in order of importance. PCA is an unsupervised learning algorithm, therefore it doesn't care about the data labels but only about variation. This can lead in some cases to misclassification of data. The main goal of a PCA analysis is to identify patterns in data; PCA aims to detect the correlation between variables. If a strong correlation between variables exists, the attempt to reduce the dimensionality only makes sense. In a nutshell, this is what PCA is all about: Finding the directions of maximum variance in high-dimensional data and project it onto a smaller dimensional subspace while retaining most of the information.

  The steps followed by the study group is, atfirst eigenvectors (the principal components) of a dataset are calculated and collected them in a projection matrix. Each of those eigenvec-



```
*** MaxRel features ***
Order    Fea      Name    Score
1        34       RMSbetaLow      2.000
2        22       MeanalphaHigh   2.000
3        37       MedianbetaHigh          2.000
4        36       MeanbetaHigh    2.000
5        19       StdDeviaalphaLow        2.000
6        20       RMSalphaLow     2.000
7        15       MeanalphaLow    2.000
8        41       RMSbetaHigh     2.000
9        33       StdDeviabetaLow         2.000
10       1        MeanDelta       2.000
11       30       MedianbetaLow   2.000
12       29       MeanbetaLow     2.000
13       40       StdDeviabetaHigh        2.000
14       2        MedianDelta     2.000
15       55       RMSgammaMid     2.000
16       8        MeanTheta       2.000
17       47       StdDeviagammaLow        2.000
18       44       MediangammaLow          2.000
19       5        StdDeviaDelta   2.000
20       6        RMSDelta        2.000
21       50       MeangammaMid    2.000
22       13       RMSTheta        2.000
```

Figure 3.13: Most Relevant Twenty Statistical Features using mRMR



```
*** MaxRel features ***
Order    Fea      Name    Score
1        23       EnergySTFTAlphaLow      0.029
2        31       EnergySTFTAlphaHigh     0.029
3        7        EnergySTFTDelta         0.029
4        55       EnergySTFTGammaLow      0.029
5        47       EnergySTFTBetaHigh      0.029
6        39       EnergySTFTBetaLow       0.029
7        63       EnergySTFTGammaMid      0.029
8        15       EnergySTFTTheta         0.029
9        38       EnergyDWTcDBetaLow      0.013
10       19       EnergyFFTAlphaLow       0.013
11       17       EnergyDCTAlphaLow       0.013
12       46       EnergyDWTcDBetaHigh     0.013
13       45       EnergyDWTcABetaHigh     0.013
14       20       EnergyIFFTAlphaLow      0.013
15       18       EnergyIDCTAlphaLow      0.013
```

Figure 3.14: Most Relevant Advanced Features using mRMR

tors is associated with an eigenvalue which are interpreted as the "length" or "magnitude" of the corresponding eigenvector. If some eigenvalues have a significantly larger magnitude than others, then the reduction of the dataset via PCA onto a smaller dimensional subspace by dropping the "less informative" eigenpairs is done. The Algorithm A.3 is followed.

- **Eigendecomposition - Computing Eigenvectors and Eigenvalues:** The eigenvectors and eigenvalues of a covariance (or correlation) matrix represent the "core" of a PCA: The eigenvectors (principal components) determine the directions of the new feature space, and the eigenvalues determine their magnitude. In other words, the eigenvalues explain the variance of the data along the new feature axes.

  Convariance Matrix: The classic approach to PCA is to perform the eigendecomposition on the covariance matrix $\sum$, which is a dd matrix where each element represents the covariance between two features. The covariance between two features is calculated as shown in the equation 3.15

$$\sigma_{jk} = \frac{1}{n-1}\sum_{i=1}^{n}(x_{ij} - \overline{x}_j)(x_{ik} - \overline{x}_k) \tag{3.15}$$

  It can be summarized via the following matrix equation 3.16

$$\sum = \frac{1}{n-1}((X - \overline{x})^T(X - \overline{x})) \tag{3.16}$$

  where $\overline{x}$ is the mean vector $\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$

  The mean vector is a d-dimensional vector where each value in this vector represents the sample mean of a feature column in the dataset.

- **Selection of Principal Components: Sorting Eigenpairs:** The typical goal of a PCA is to reduce the dimensionality of the original feature space by projecting it onto a smaller subspace, where the eigenvectors will form the axes. However, the eigenvectors only define the directions of the new axis., since they have all the same unit length 1. In order to decide which eigenvector(s) can dropped without losing too much information for the construction of lower-dimensional subspace, we need to inspect the corresponding eigenvalues: The eigenvectors with the lowest eigenvalues bear the least information about the distribution of the data; those are the ones can be dropped. So, the common approach is to rank the eigenvalues from highest to lowest in order choose the top k eigenvectors.

  Explained Variance: Now to choose the required principal components features, explained variance is calculated. This is calculated from the eigenvalues. The explained variance tells us how much information (variance) can be attributed to each of the principal components.

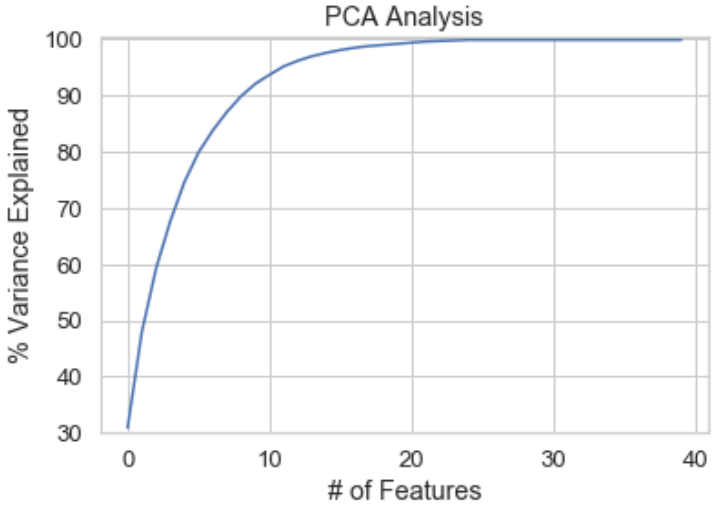


Figure 3.15: Number of Statistical Features reduced-PCA

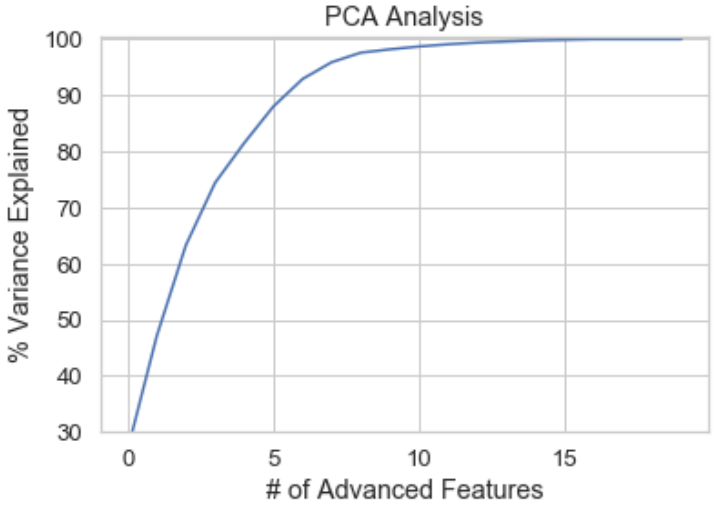


Figure 3.16: Number of Advanced Features reduced-PCA

  - **Projection on to the New Feature Space:** Finally, 2-dimensional projection matrix W is transformed onto the new subspace via the equation $Y = XW$.

After applying the PCA it is found to have 98% (as shown in figure 3.15) of the variance just applying only 20 plus statistical features and similarly 98% variance can also be achieved by applying 10 plus advanced features as shown in figure 3.16. Scatter plot shows in figure 3.17 how PCA separates features based on different emotion level.

- **Linear Discriminant Analysis (LDA):** LDA is supervised learning dimensionality reduction technique and Machine Learning classifier. LDA aims to maximize the distance between the mean of each class and minimize the spreading within the class itself in other words, with available information about class labels, LDA will seek to maximise the separation between the different classes by computing the component axes (linear discriminants

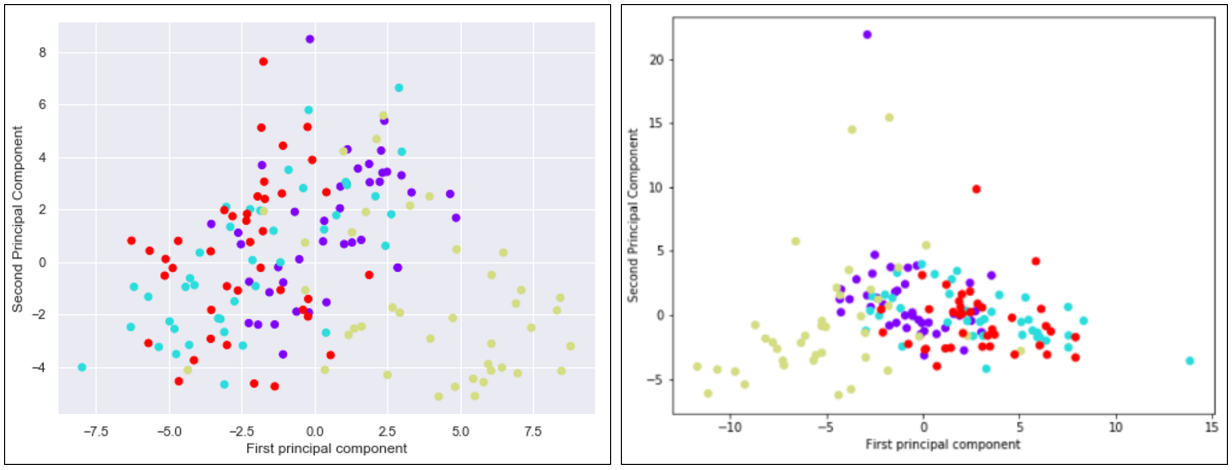


Figure 3.17: Distribution of Statistical and Advanced Features based on different emotional state after applying PCA

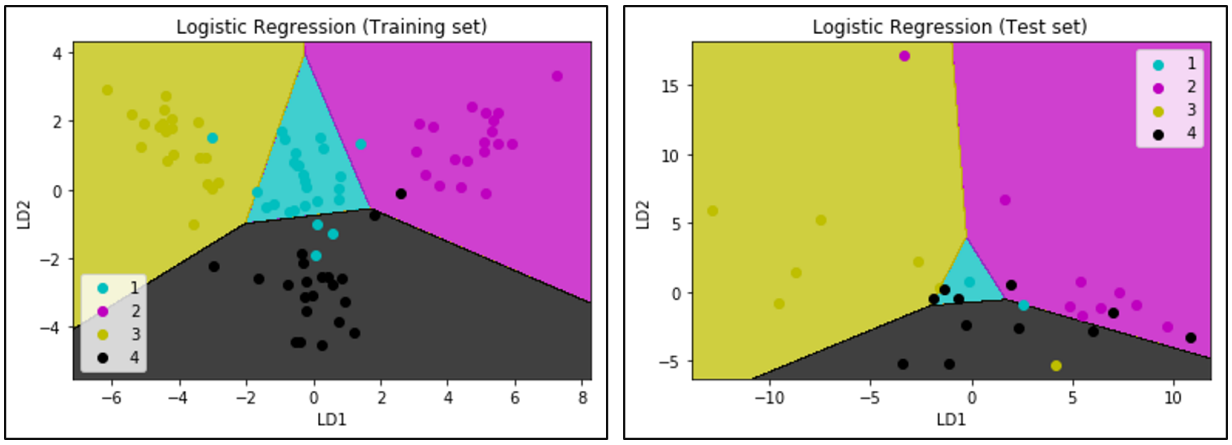


Figure 3.18: Train and Test Result on Statistical Features applying LDA

) which does this. LDA uses therefore within classes and between classes as measures. It is a good choice because maximizing the distance between the means of each class when projecting the data in a lower-dimensional space can lead to better classification results.In LDA, it is assumed that the input data follows a Gaussian Distribution otherwise it may possibly lead to poor classification results. Algorithm A.3 is followed for the operation.

After applying LDA in statistical features we get figure 3.18 and in advanced features we get figure 3.19

- **t-distributed Stochastic Neighbor Embedding (t-SNE):** t-SNE is non-linear dimensionality reduction technique which is typically used to visualize high dimensional datasets.t-SNE works by minimizing the divergence between a distribution constituted by the pairwise probability similarities of the input features in the original high dimensional space and its equivalent in the reduced low dimensional space. t-SNE makes then use of the Kullback-Leiber (KL) divergence in order to measure the dissimilarity of the two different

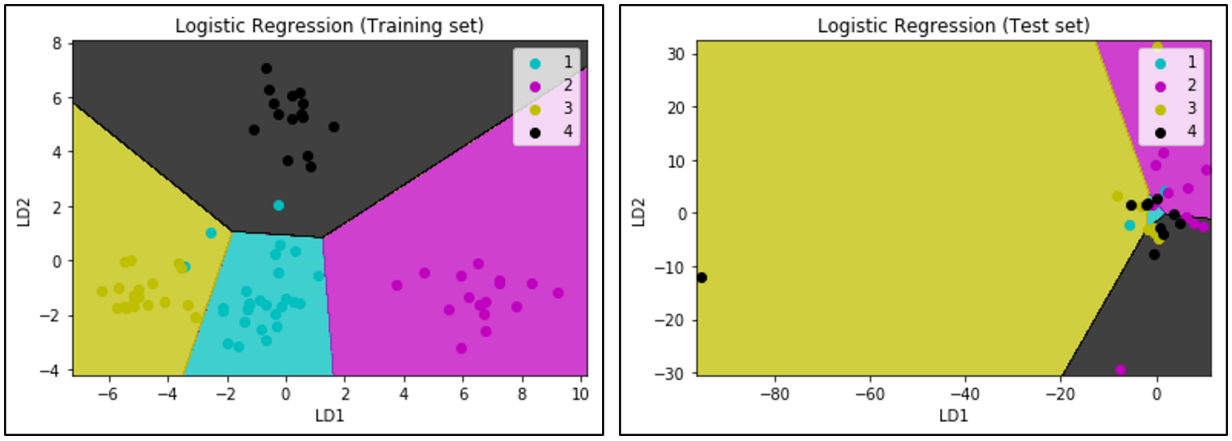


Figure 3.19: Train and Test Result on Advanced Features applying LDA

distributions. The KL divergence is then minimized using gradient descent. In t-SNE, the higher dimensional space is modelled using a Gaussian Distribution, while the lower-dimensional space is modelled using a Student's t-distribution. This is done, in order to avoid an imbalance in the neighbouring points distance distribution caused by the translation into a lower-dimensional space. SNE starts by converting the high-dimensional Euclidean distances between datapoints into conditional probabilities that represent similarities. In short, It is extremely good in capturing both the local and global structure of the highly-dimensional data. Instead of looking at directions/axes which maximise information or class separation, t-SNE converts the Euclidean distances between points into conditional probabilities. A Student-t distribution is then used on these probabilities which serve as metrics to calculate the similarity between one datapoint to another. The similarity of datapoint x_j to datapoint x_i is the conditional probability, $p_{j|i}$, that x_i would pick x_j as its neighbor if neighbors were picked in proportion to their probability density under a Gaussian centered at x_i. For nearby datapoints, $p_{j|i}$ is relatively high, whereas for widely separated datapoints, $p_{j|i}$ will be almost infinitesimal (for reasonable values of the variance of the Gaussian, ω_i). Mathematically, the conditional probability $p_{j|i}$ is given by equation 3.17

$$p_{j|i} = \frac{exp(\frac{-||x_i - x_j||^2}{2{\sigma_i}^2})}{\sum_{k \neq i} exp(\frac{-||x_i - x_k||^2}{2{\sigma_i}^2})} \tag{3.17}$$

where $_i$ is the variance of the Gaussian that is centered on datapoint x_i. The method for determining the value of i is presented later in this section. Because we are only interested in modeling pairwise similarities, we set the value of $p_{i|i}$ to zero. For the low-dimensional counterparts y_i and y_j of the high-dimensional datapoints x_i and x_j, it is possible to compute a similar conditional probability, which we denote by $q_{j|i}$. We set the variance of the Gaussian that is employed in the computation of the conditional probabilities $q_{j|i}$ to $\frac{1}{\sqrt{2}}$.

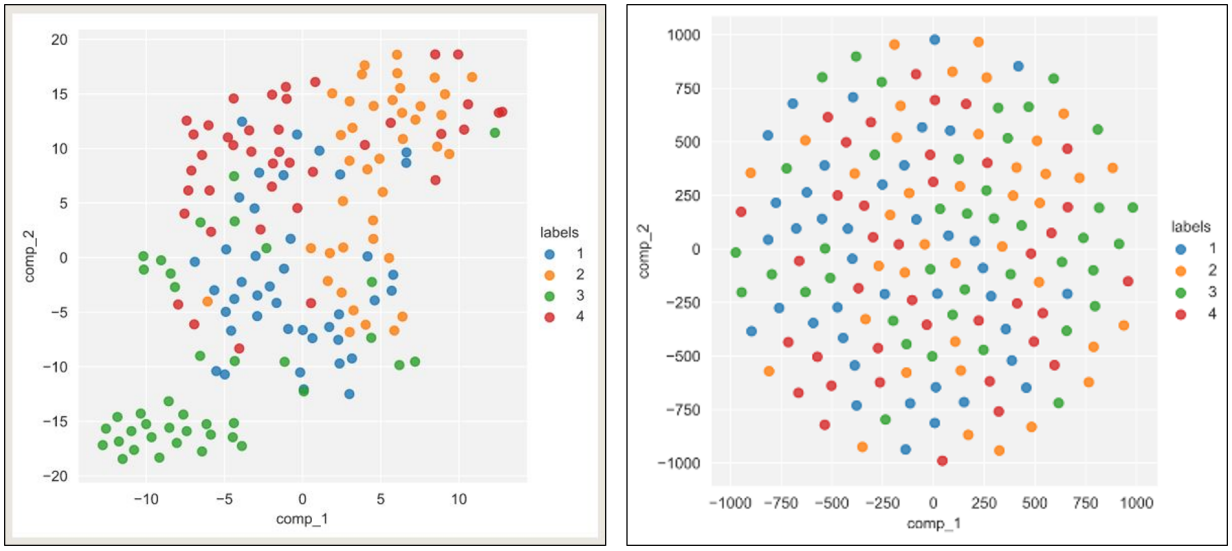


Figure 3.20: Distribution of Statistical and Advanced Features based on different Emotion State after applying t-SNE

Hence, we model the similarity of map point y_j to map point y_i by equation 3.18

$$q_{j|i} = \frac{exp(-||y_i - y_j||^2)}{\sum_{k \neq i} exp(-||y_i - y_k||^2)} \tag{3.18}$$

After applying t-SNE as shown in the Algorithm A.3 our dataset is visualized as figure 3.20

3.6 Model Specification

- **Hold Out:** In this method, our data is divided into two sets: Training and Test/Validation set i.e. a hold-out set. We then trained the model on the training dataset and evaluate the model on the Test/Validation dataset. Typically the training dataset is bigger than the hold-out dataset. Typical ratios used for splitting the data set include 60:40, 80:20 etc. Here, test and train is divided into 30/70 as shown in the figure 3.21

- **K-fold Cross-Validation:** In k-fold cross-validation, the original sample is randomly partitioned into k equal sized subsamples of the k subsamples, a single subsample is retained as the validation data for testing the model, and the remaining k 1 subsamples are used as training data. The cross-validation process is then repeated k times, with each of the k subsamples used exactly once as the validation data. The k results can then be averaged to produce a single estimation. The advantage is, all observations are used for both training and validation, and each observation is used for validation exactly once. Here, 10-fold cross-validation is used. Where k=10 results in 10-fold cross-validation. In 10-fold cross-validation, we have randomly shuffled the dataset into 10 sets d_0 to d_9, so that all sets are

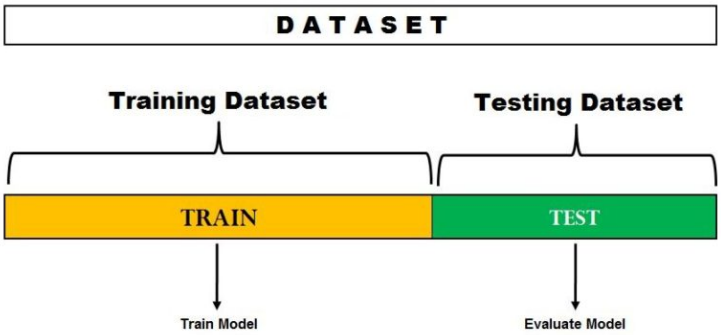


Figure 3.21: Hold Out Model Validation Technique

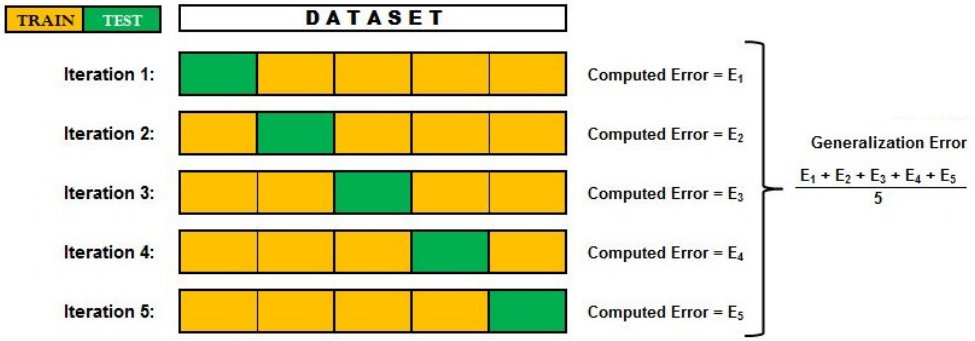


Figure 3.22: K(5)-Fold Cross Validation Technique

equal sized. Then the training and validations are performed. Following figure 3.22 shows for 5-fold cross validation.

- **Stratified K-fold Cross-Validation:** Here the limitation of K-fold validation techniques are minimized by introducing classes in our training and test dataset, for which we have use Stratified k-Fold Cross-Validation. For three class label and k=5 following figure 3.23 shows the mechanisms adopted.

  The comparative study shows the accuracy level as shown in the figure 3.24

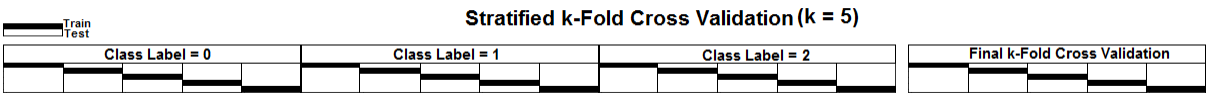


Figure 3.23: Example of Stratified K-fold Validation where k=5 and class=3

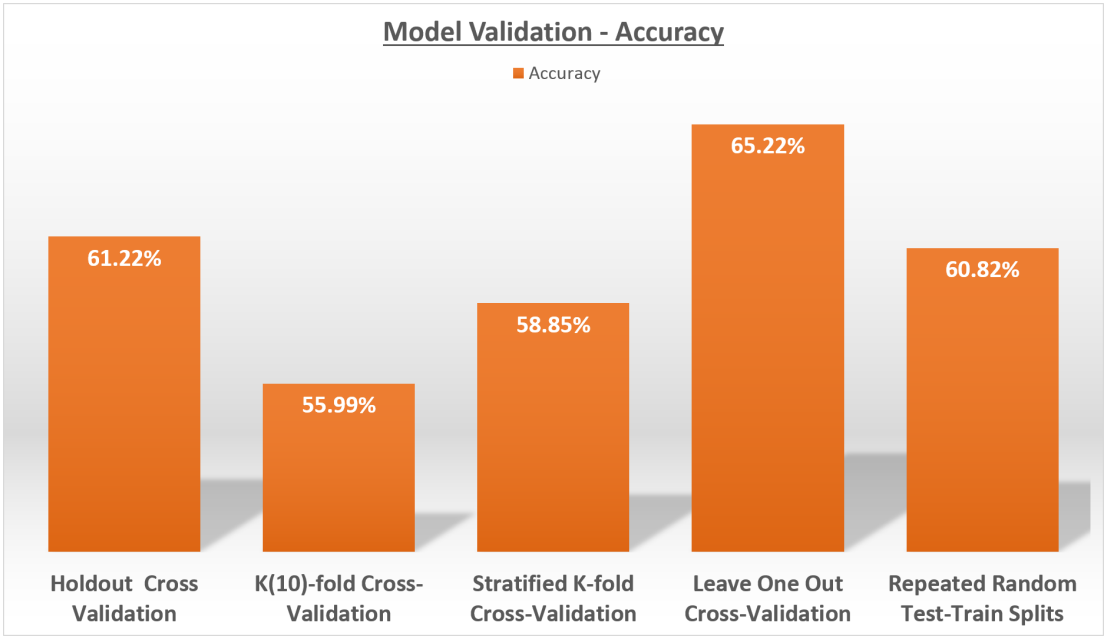


Figure 3.24: Comparative Study of Accuracy

CHAPTER 4
PERFORMANCE EVALUATION

The overall study that has been conducted with machine learning and emotion recognition applying multiple algorithms for classification, feature level fusion etc can be shown using figure 4.24.

Statistical Feature: We have applied RandomForestClassifier, DecisionTreeClassifier, PassiveAggressiveClassifier, BaggingClassifier, ExtraTreesClassifier, GradientBoostingClassifier, GaussianNB, Perceptron, KNeighborsClassifier, LogisticRegressionCV, RidgeClassifierCV, BernoulliNB, GaussianProcessClassifier, SGDClassifier, AdaBoostClassifier, LinearSVC, SVC, NuSVC on unscaled statistical feature dataset. But since all of them could not produce accuracy to the expected level so here we are mentioning in details about the best 5 algorithms for our case. The accuracy result in in Table 4.1 and in Figure 4.1. The pairplot of the accuracy result is in Figure 4.2. The confusion matrix of the accuracy result is in Figure 4.3. At next step precision, F1 score and recall values are shown in Table 4.2. The ROC curves generated for four types of emotions are shown:Disgust in Figure 4.4, Funny in Figure 4.5, Peaceful in Figure 4.6 and Sad in Figure 4.7.

We normalized/scaled the statistical features dataset and apply machine learning algorithms again to check whether any changes in accuracy found. Table 4.3 and Figure 4.8 shows the accuracy result for scaled statistical dataset. Since no major variations in between accuracy result of scaled and unscaled dataset. So we would proceed with unscaled data as we were doing earlier.

Advanced Feature: We have applied RandomForestClassifier, DecisionTreeClassifier, PassiveAggressiveClassifier, BaggingClassifier, ExtraTreesClassifier, GradientBoostingClassifier, GaussianNB, Perceptron, KNeighborsClassifier, LogisticRegressionCV, RidgeClassifierCV, BernoulliNB, GaussianProcessClassifier, SGDClassifier, AdaBoostClassifier, LinearSVC, SVC, NuSVC on unscaled advanced feature dataset. But since all of them could not produce accuracy to the expected level so here we are mentioning in details about the best 5 algorithms for our case. The accuracy result in in Table 4.4 and in Figure 4.9. The pairplot of the accuracy result is in Figure 4.10. The confusion matrix of the accuracy result is in Figure 4.11. At next step precision, F1 score and recall values are shown in Table 4.5. The ROC curves generated for four types of emotions are shown:Disgust in Figure 4.12, Funny in Figure 4.13, Peaceful in Figure 4.14 and Sad in Figure 4.15.

Statistical + Advanced Feature: We have applied RandomForestClassifier, DecisionTreeClassifier, PassiveAggressiveClassifier, BaggingClassifier, ExtraTreesClassifier, GradientBoostingClassifier, GaussianNB, Perceptron, KNeighborsClassifier, LogisticRegressionCV, RidgeClassifierCV, BernoulliNB, GaussianProcessClassifier, SGDClassifier, AdaBoostClassifier, LinearSVC, SVC, NuSVC on unscaled advanced and statistical combined feature dataset. But since all of them could not produce accuracy to the expected level so here we are mentioning in details about the best 5 algorithms for our case. The accuracy result in in Table 4.6 and in Figure 4.16. The pairplot of the accuracy result is in Figure 4.17. The confusion matrix of the accuracy result is in Figure 4.18. At next step precision, F1 score and recall values are shown in Table 4.7.

Mixture of Statistical and Advanced Selected Features: As discussed earlier that we applied XGBoost and mRMR on advanced and statistical feature dataset for data selection. The mentioned two techniques would select the features that have impact on emotion accuracy and neglect the excess features. We combine those selected features of advanced and statistical dataset and applied machine learning algorithm on them. Earlier we applied the machine learning algorithm on combined dataset of advanced and statisitcal features and found accuracy more than 86%. Now apply the machine learning algorithms on the dataset of selected features from both the dataset, we have also achieved accuracy more than 86%. So, it can depicted that using the selected features only, we can achieve the same accuracy, which means using lesser features in lesser time we can achieve the expected accuracy. We have applied RandomForestClassifier, DecisionTreeClassifier, PassiveAggressiveClassifier, BaggingClassifier, ExtraTreesClassifier, GradientBoostingClassifier, GaussianNB, Perceptron, KNeighborsClassifier, LogisticRegressionCV, RidgeClassifierCV, BernoulliNB, GaussianProcessClassifier, SGDClassifier, AdaBoostClassifier, LinearSVC, SVC, NuSVC on unscaled advanced and statistical combined selected feature dataset. But since all of them could not produce accuracy to the expected level so here we are mentioning in details about the best 4 algorithms for our case. The accuracy result in in Table 4.8 and in Figure 4.19. The pairplot of the accuracy result is in Figure 4.20. The confusion matrix of the accuracy result is in Figure 4.21. At next step precision, F1 score and recall values are shown in Table 4.9.

Selected Features from Statistical and Advanced Feature dataset applying mRMR: We select features from fusion of statistical and advanced feature using mRMR. From PCA we observe that around 30 features are required to achieve 100% variance as in Figure 4.29. We select the exact 30 features applying mRMR as in Figure 4.30. Applying machine learning algorithms on selected features we get the accuracy result in Table 4.13 and Figure 4.31. The pairplot is shown in Figure 4.32 and confusion matrix is on Figure 4.33. The precision, recall and F1 score is in Table 4.14. ROC is shown in Figure 4.34.

Statistical Feature Window: We apply overlapping window on statistical feature dataset with window size as 5. The accuracy result in in Table 4.10 and in Figure 4.26.

Advanced Feature Window: We apply overlapping window on advanced feature dataset with

window size as 5. The accuracy result in in Table 4.11 and in Figure 4.27.

Advanced and Statistical Fusion Feature Window: We apply overlapping window on advanced and statistical fusion feature dataset with window size as 5. The accuracy result in in Table 4.12 and in Figure 4.28.

MLA Name	MLA Train Accuracy(%)	MLA Test Accuracy(%)
XGBClassifier	100.00	86.21
RidgeClassifierCV	94.12	79.31
ExtraTreesClassifier	100.00	75.86
LinearSVC	78.82	72.41
DecisionTreeClassifier	100.00	65.51

Table 4.1: Accuracy Result for Algorithm applied on Statistical Feature Dataset

MLA Name	XGB Classifier	Ridge Classifier CV	Extra Trees Classifier	Gradient Boost Classidier
MLA Precision(Macro)(%)	83.65	75.21	70.33	62.91
MLA Precision(Micro)(%)	86.21	79.31	68.96	65.51
MLA Precision(Weighted)(%)	90.72	86.95	78.02	73.90
MLA Recall(Macro)(%)	86.67	81.89	76.74	62.34
MLA Recall(Micro)(%)	86.21	79.31	68.96	65.51
MLA Recall(Weighted)(%)	86.21	79.31	68.96	65.51
F1 Score(Macro)(%)	81.81	73.93	68.00	60.14
F1 Score(Micro)(%)	86.21	79.31	68.96	65.51
F1 Score(Weighted)(%)	86.57	81.44	71.41	68.65

Table 4.2: Precision, Recall, F1 Score for Algorithm applied on Statistical Feature Dataset

MLA Name	MLA Train Accuracy(%)	MLA Test Accuracy(%)
RidgeClassifierCV	81.32	87.1
GradientBoostingClassifier	100.00	83.87
LinearSVC	89.01	83.87
GaussianNB	73.63	80.65

Table 4.3: Accuracy Result for Algorithm applied on Scaled Statistical Feature Dataset

Finally, after evaluation of all studies we have found out that, the accuracy in recognizing human emotion using 2 channel EEG is 88%. Where, only considering the statistical features the accuracy level it can be increased to 86.21%. Our further study showed that, apart from statistical features, considering some advanced features the accuracy level has decreased to 85.21%.

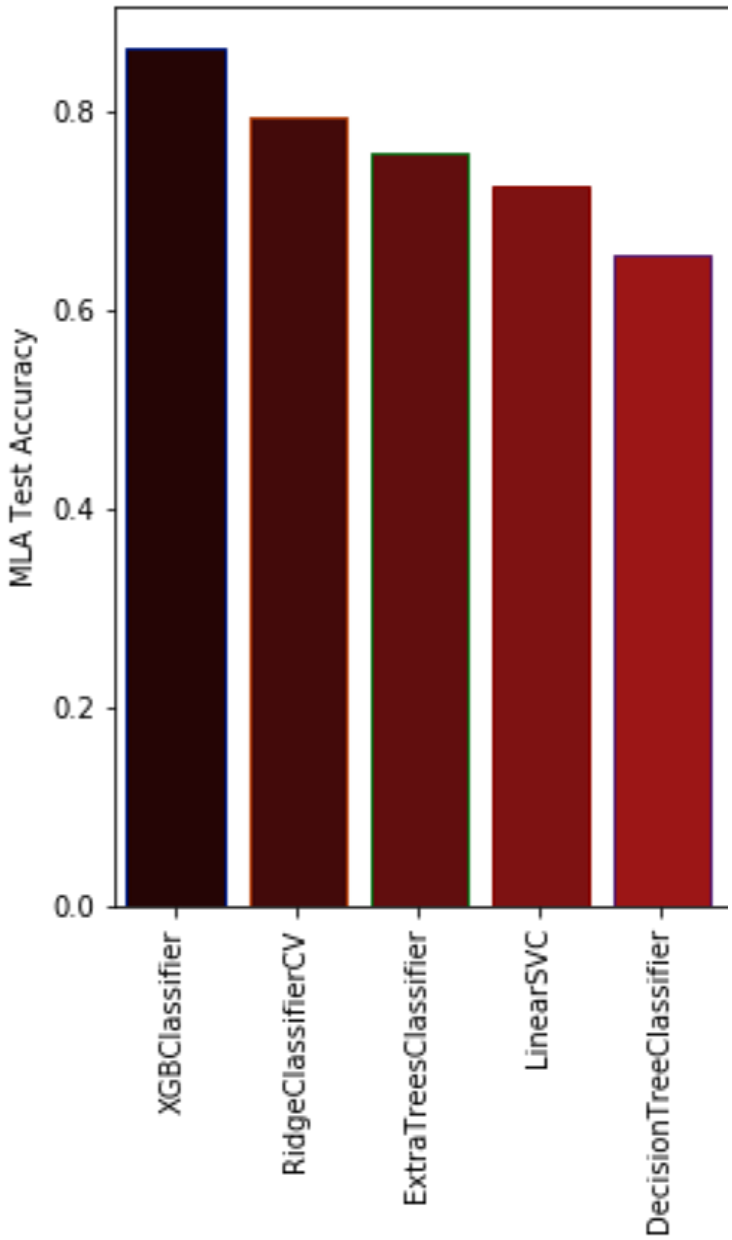


Figure 4.1: Accuracy Result for Algorithm applied on Statistical Feature Dataset

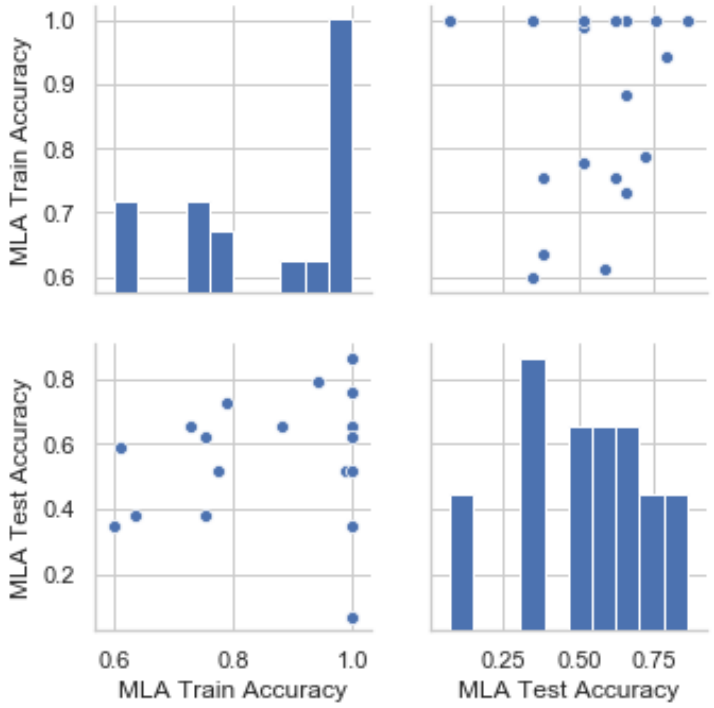


Figure 4.2: Accuracy Result for Algorithm applied on Statistical Feature Dataset (PairPlot)

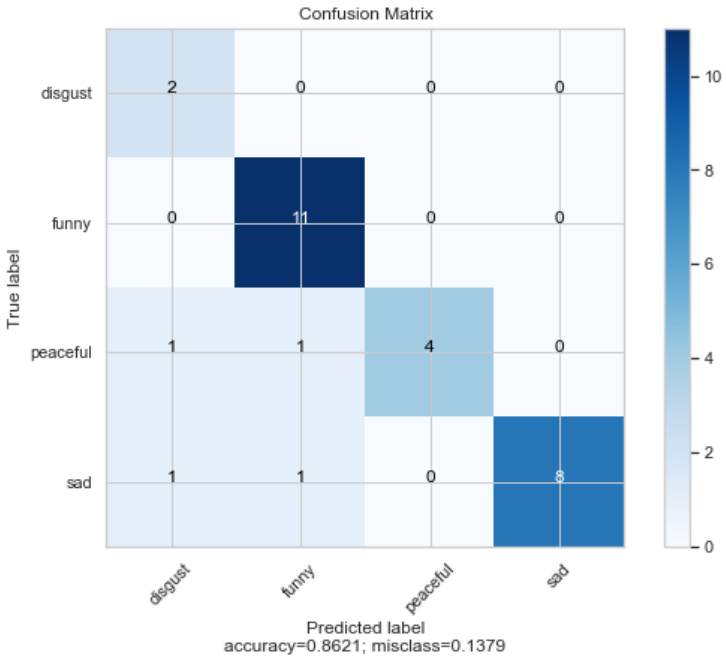


Figure 4.3: Accuracy Result for Algorithm applied on Statistical Feature Dataset (Confusion Matrix)

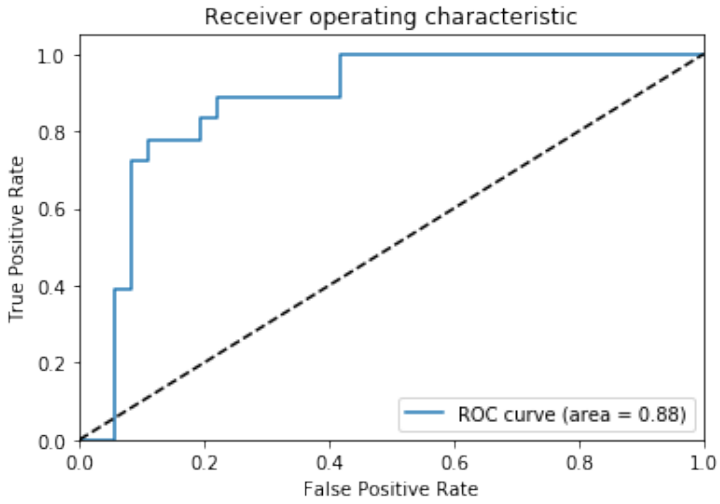


Figure 4.4: ROC curve for emotion disgust in Statistical Feature Dataset

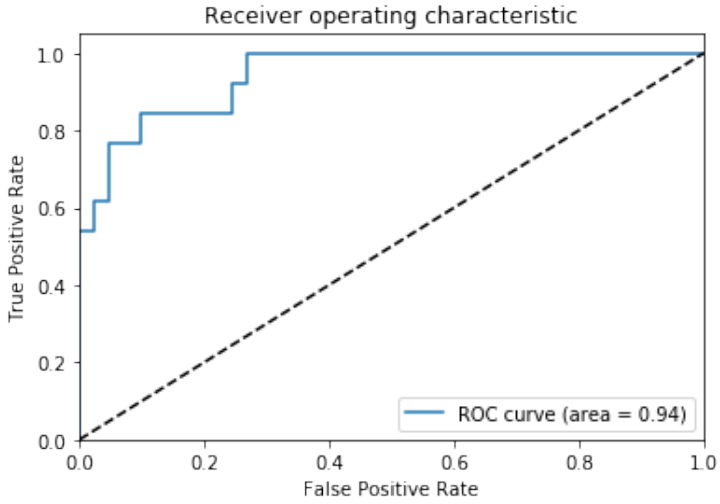


Figure 4.5: ROC curve for emotion funny in Statistical Feature Dataset

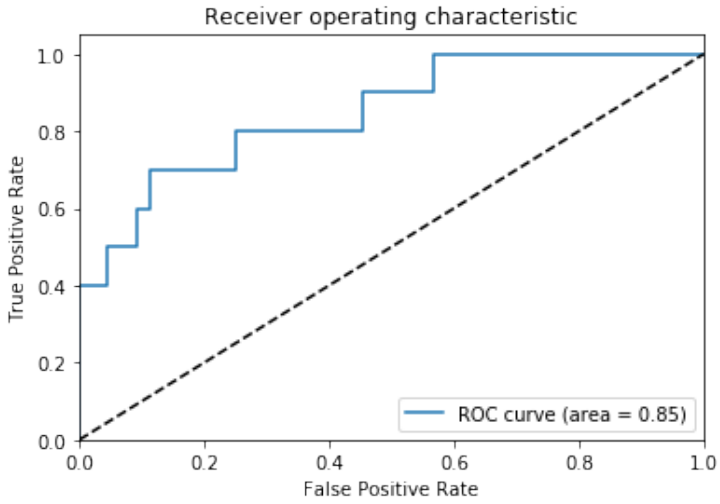


Figure 4.6: ROC curve for emotion Peaceful in Statistical Feature Dataset

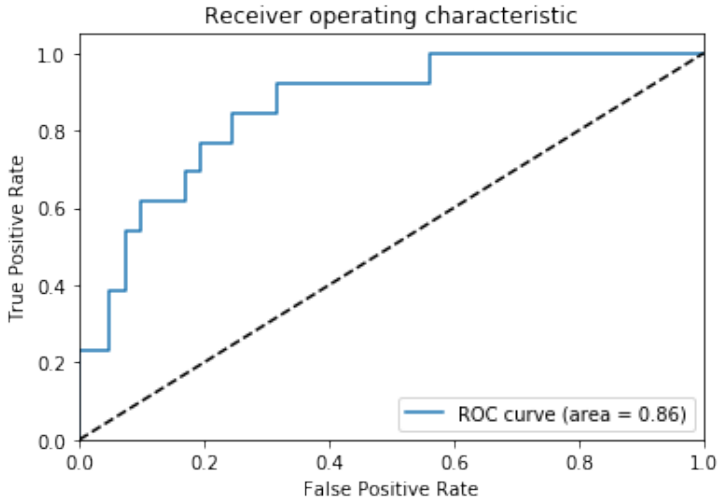


Figure 4.7: ROC curve for emotion Sad in Statistical Feature Dataset

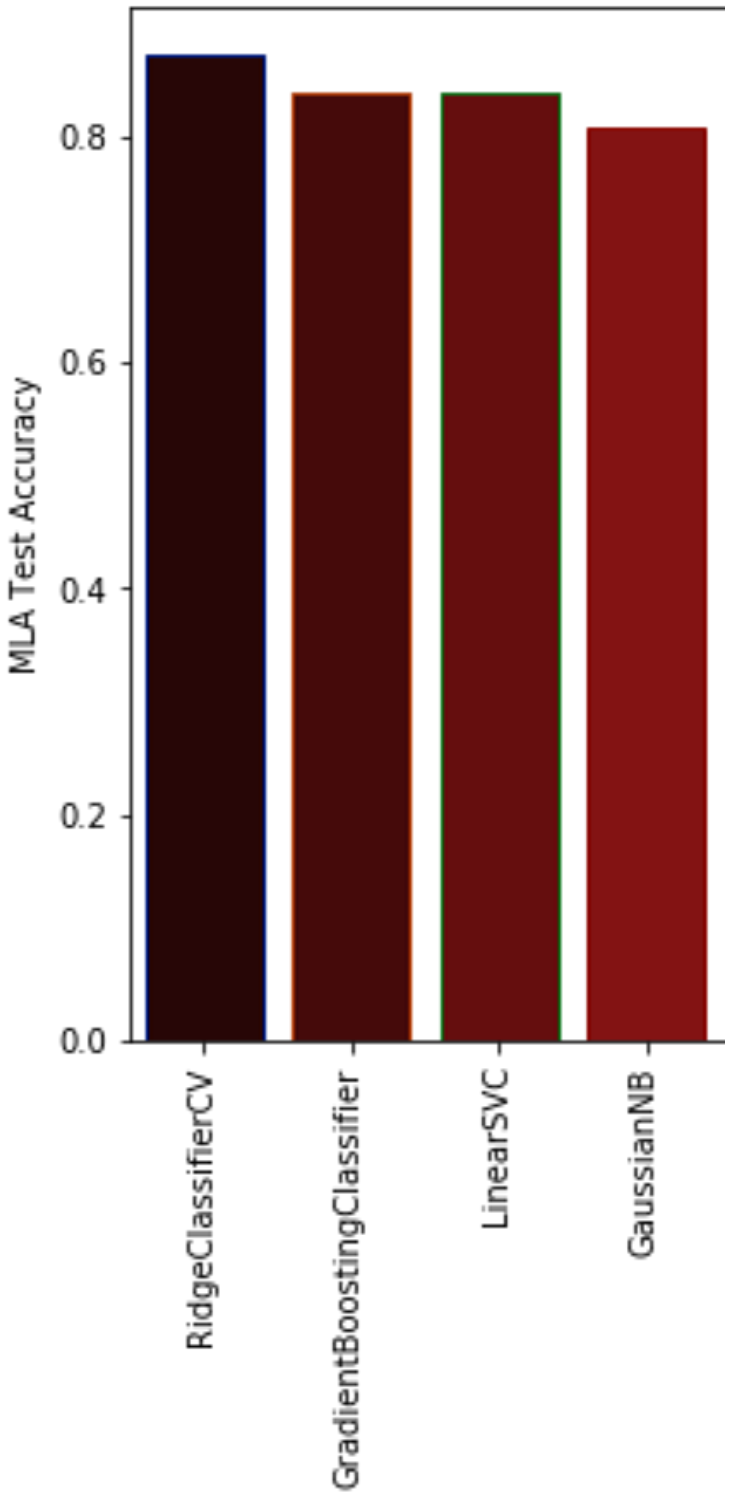


Figure 4.8: Accuracy Result for Algorithm applied on Scaled Statistical Feature Dataset

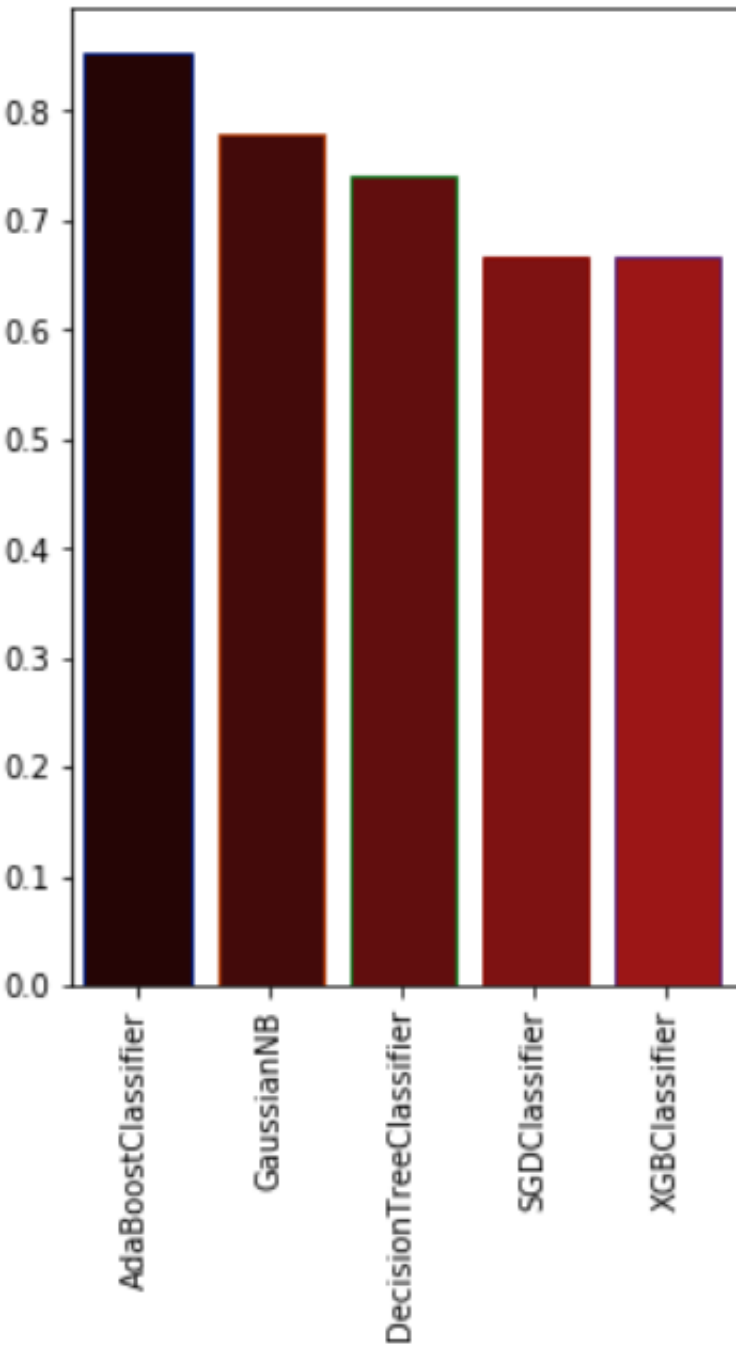


Figure 4.9: Accuracy Result for Algorithm applied on Advanced Feature Dataset

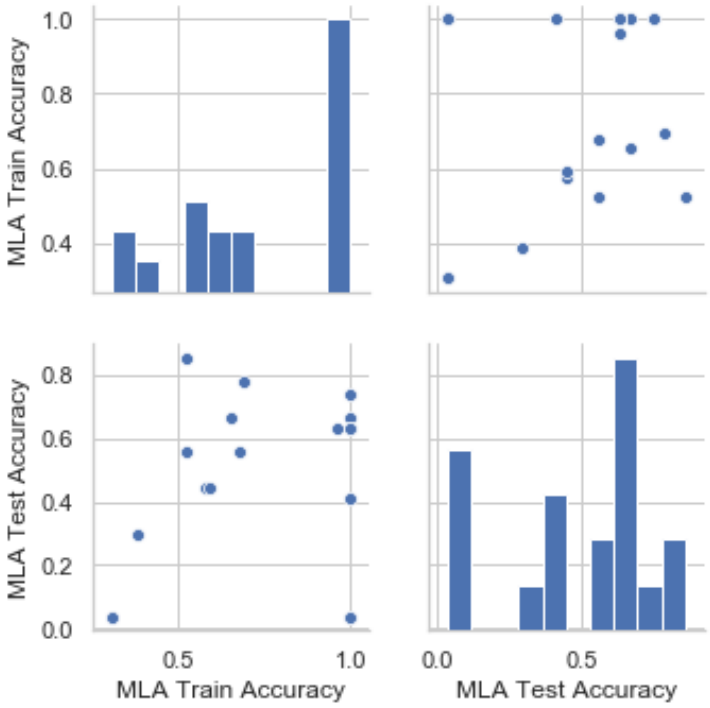


Figure 4.10: Accuracy Result for Algorithm applied on Advanced Feature Dataset (PairPlot)

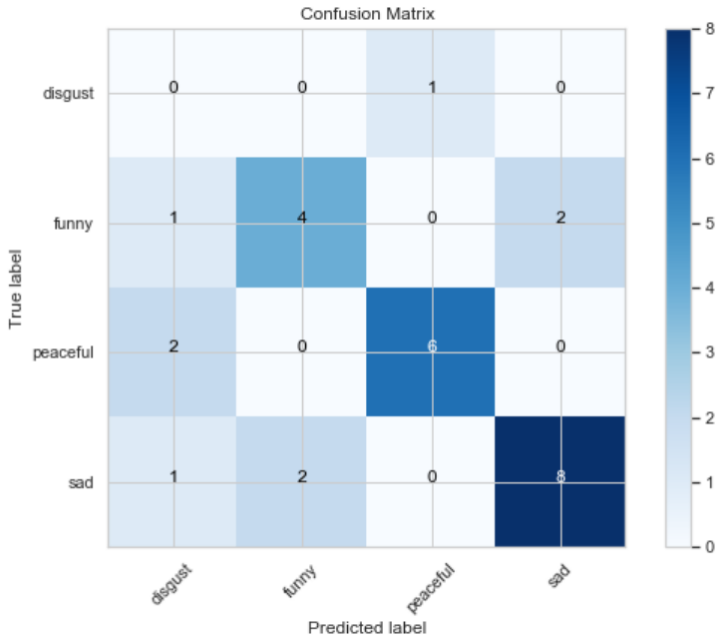


Figure 4.11: Accuracy Result for Algorithm applied on Advanced Feature Dataset (Confusion Matrix)

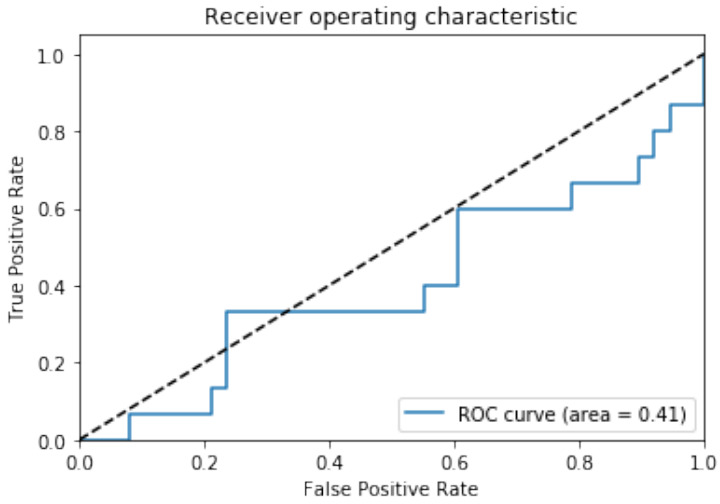


Figure 4.12: ROC curve for emotion disgust in Advanced Feature Dataset

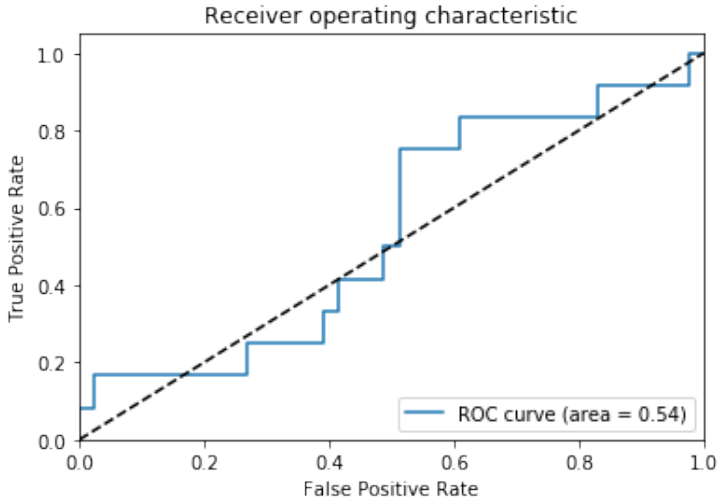


Figure 4.13: ROC curve for emotion funny in Advanced Feature Dataset

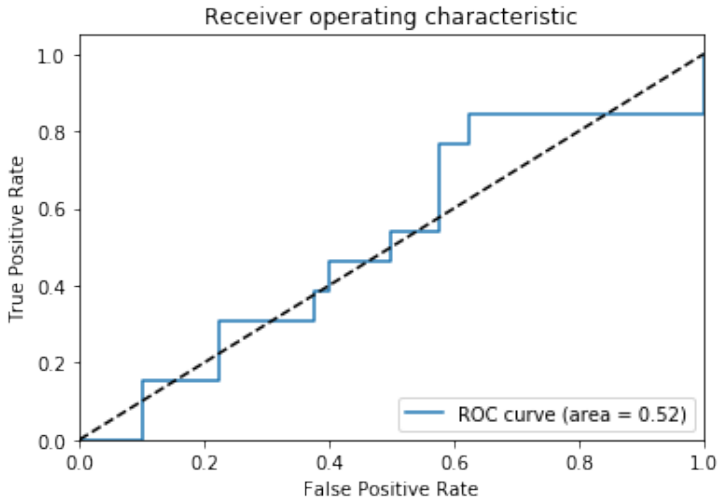


Figure 4.14: ROC curve for emotion Peaceful in Advanced Feature Dataset

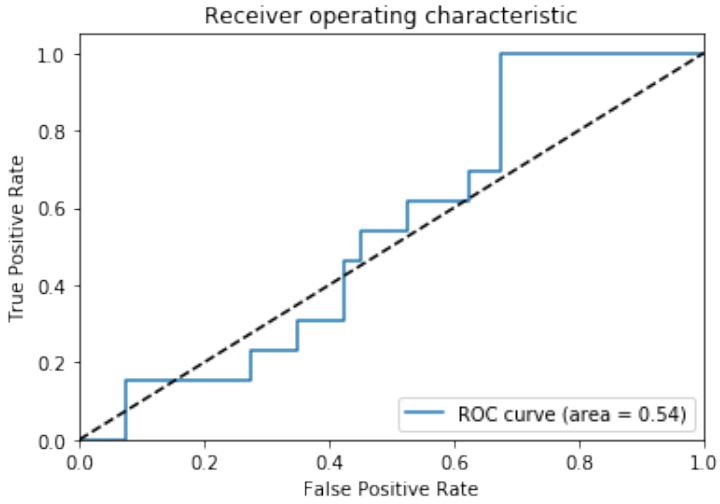


Figure 4.15: ROC curve for emotion Sad in Advanced Feature Dataset

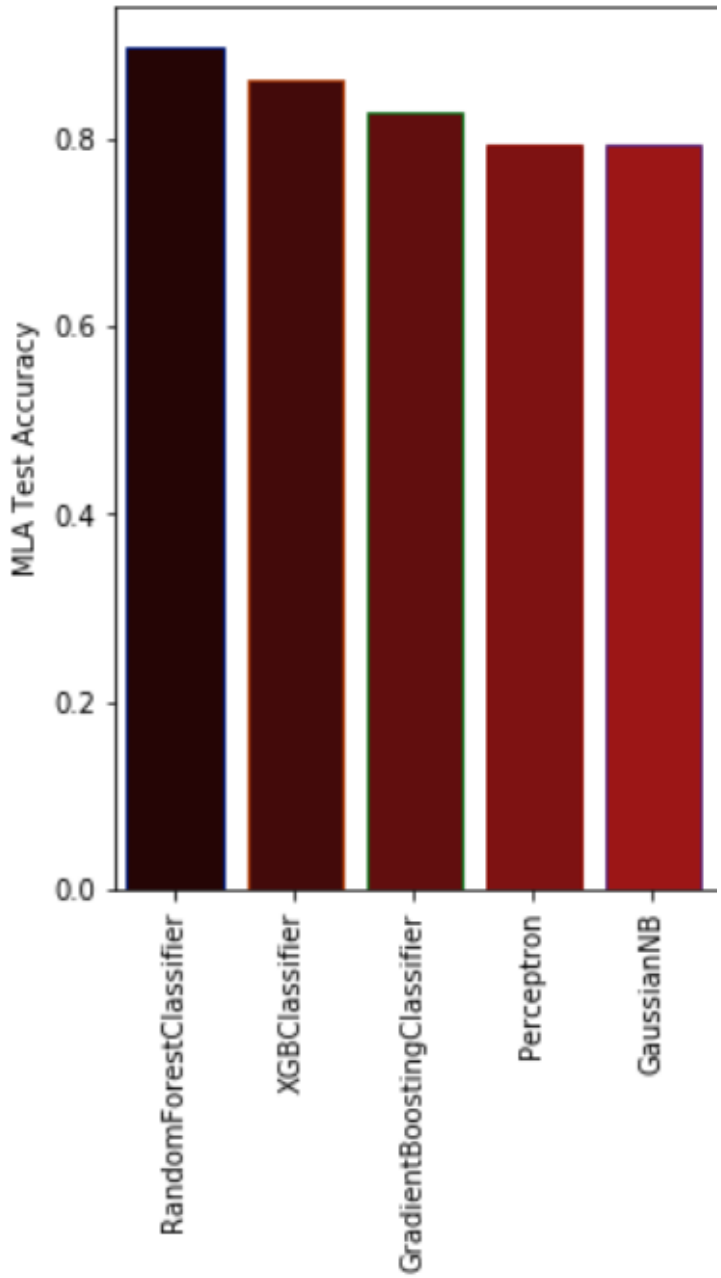


Figure 4.16: Accuracy Result for Algorithm applied on Advanced and Statistical Feature Dataset

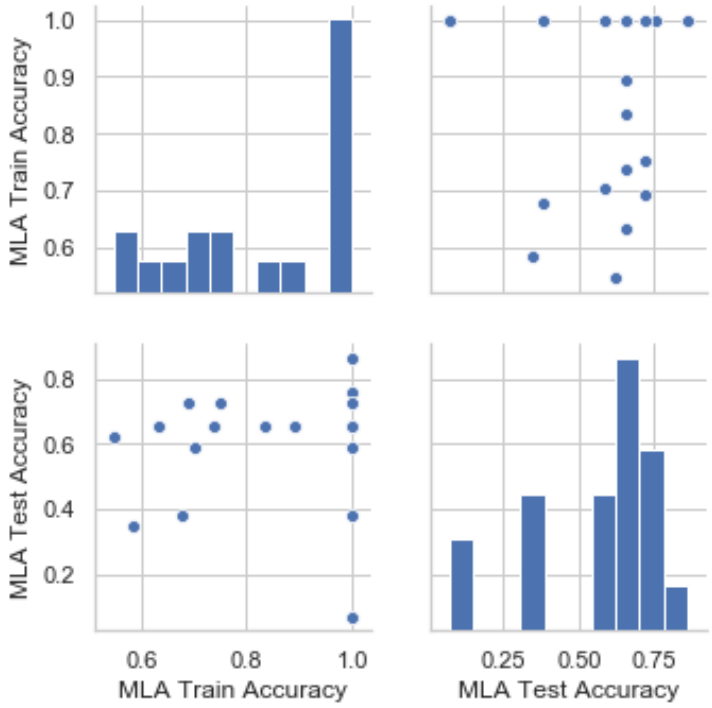


Figure 4.17: Accuracy Result for Algorithm applied on Advanced and Statistical Feature Dataset (PairPlot)

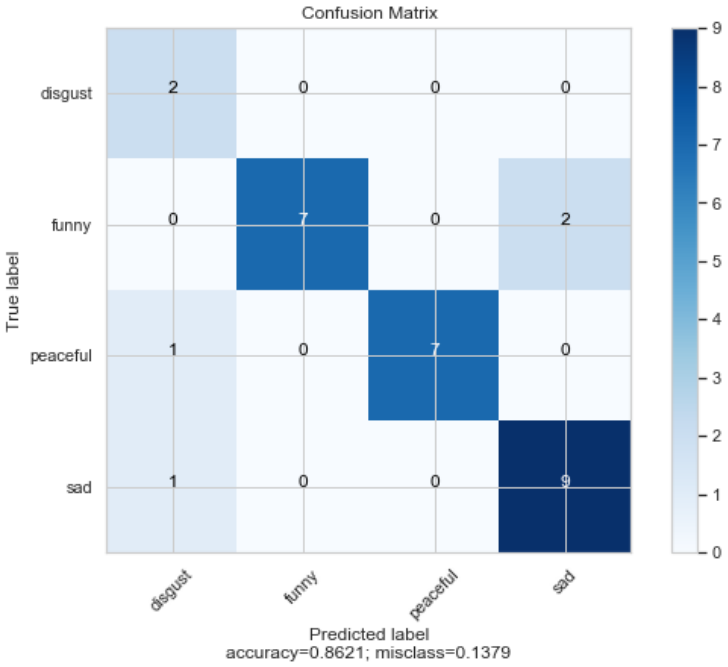


Figure 4.18: Accuracy Result for Algorithm applied on Advanced and Statistical Feature Dataset (Confusion Matrix)

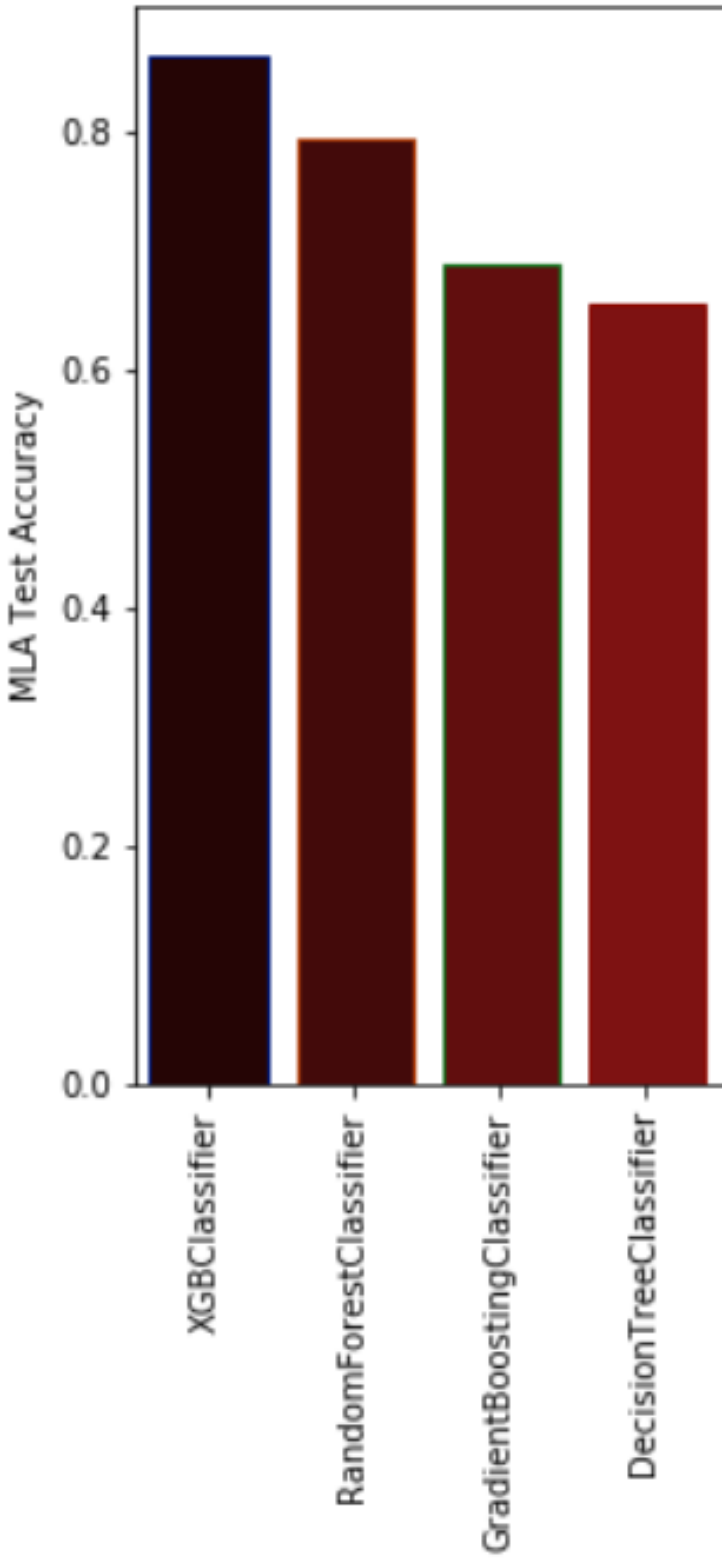


Figure 4.19: Accuracy Result for Algorithm applied on Advanced and Statistical Selected Feature Dataset

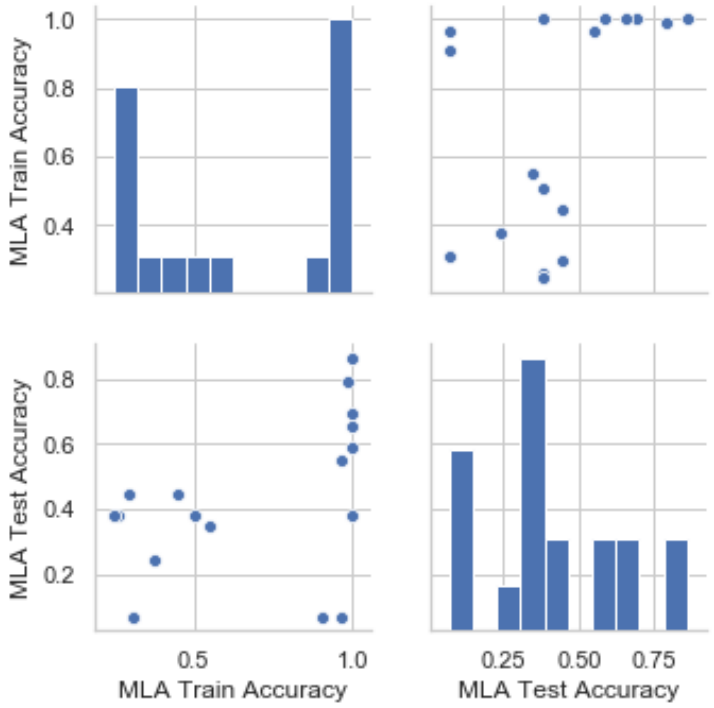


Figure 4.20: Accuracy Result for Algorithm applied on Advanced and Statistical Selected Feature Dataset (PairPlot)

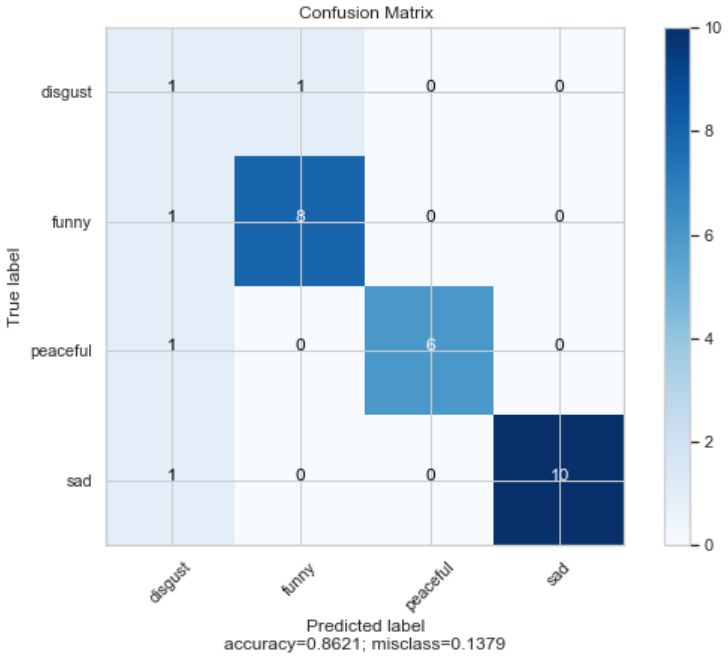


Figure 4.21: Accuracy Result for Algorithm applied on Advanced and Statistical Selected Feature Dataset (Confusion Matrix)

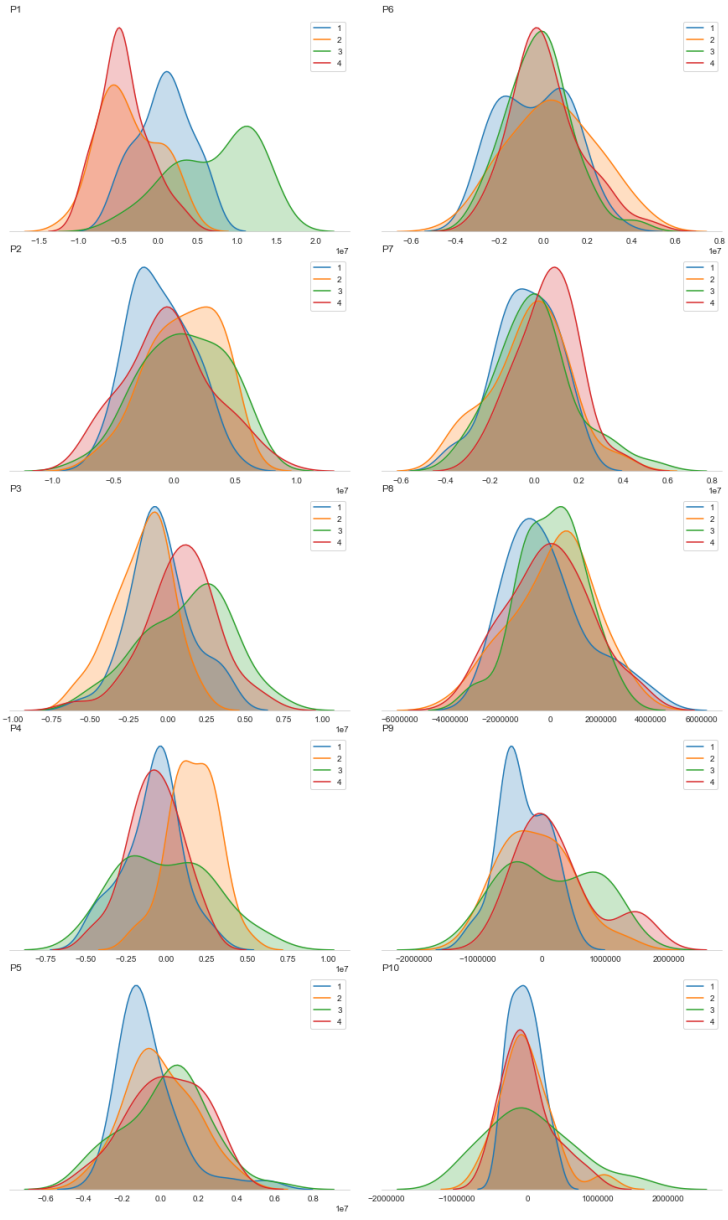


Figure 4.22: Distribution Plot of Different Emotion using Statistical Features after PCA

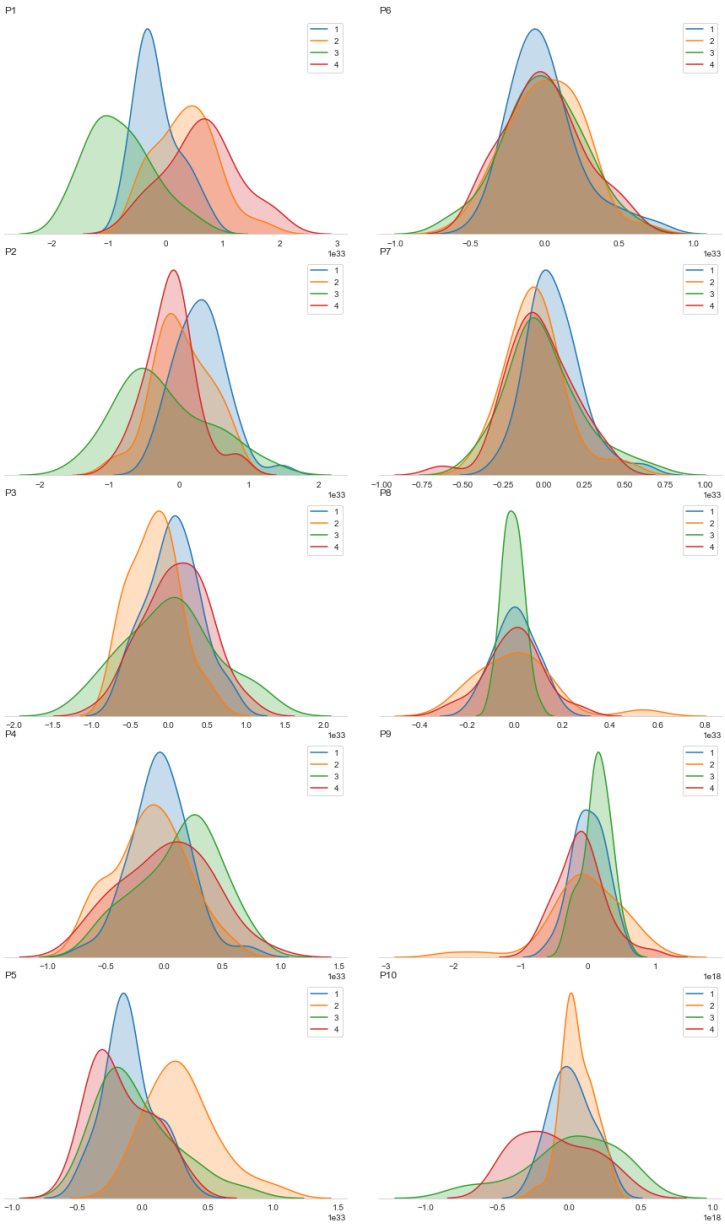


Figure 4.23: Distribution Plot of Different Emotion using Advanced Features after PCA

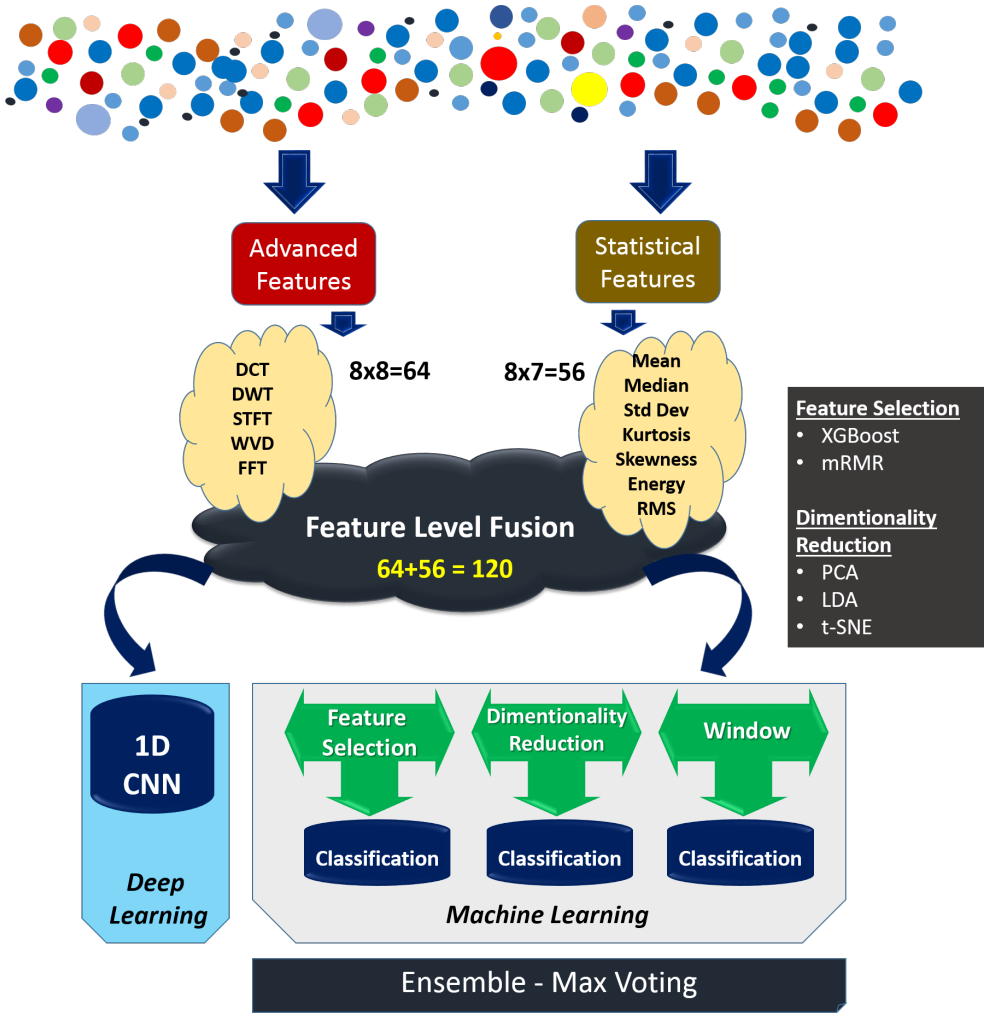


Figure 4.24: Steps followed for Performance Evaluation

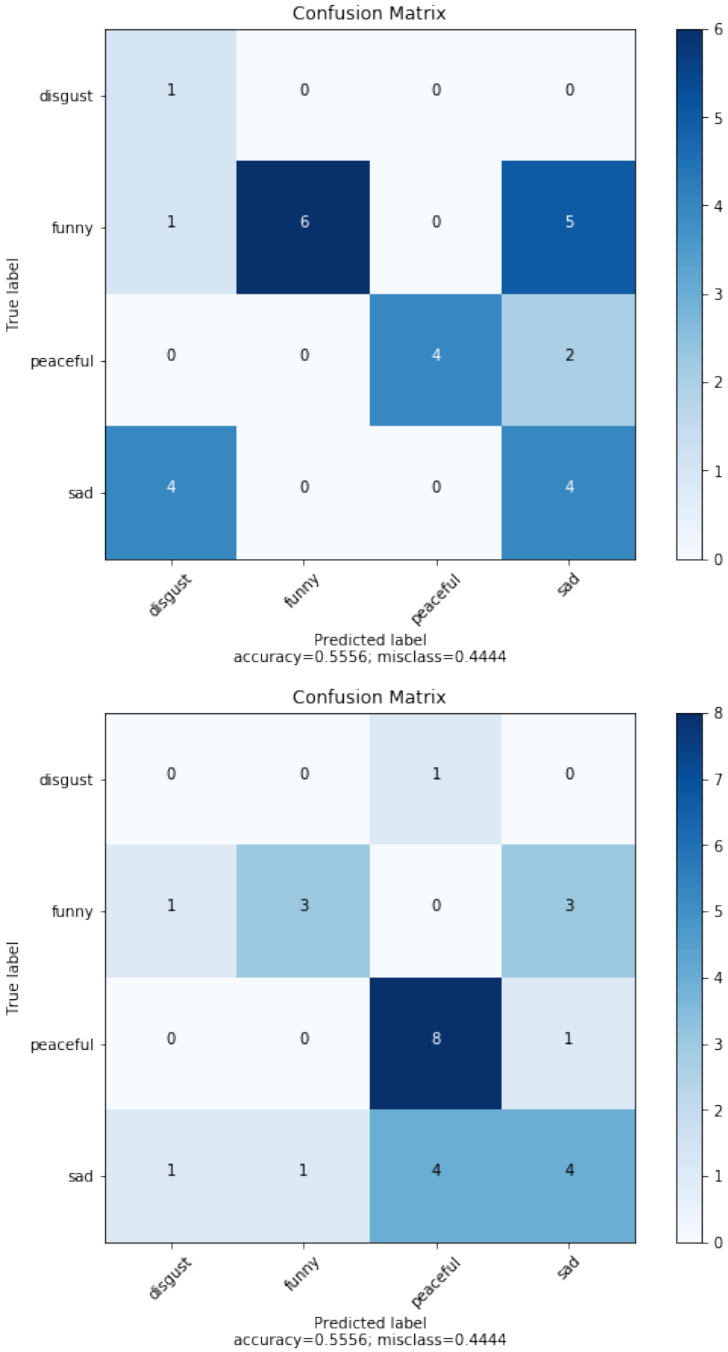


Figure 4.25: Confusion Matrix of Statistical and Advanced Features after LDA

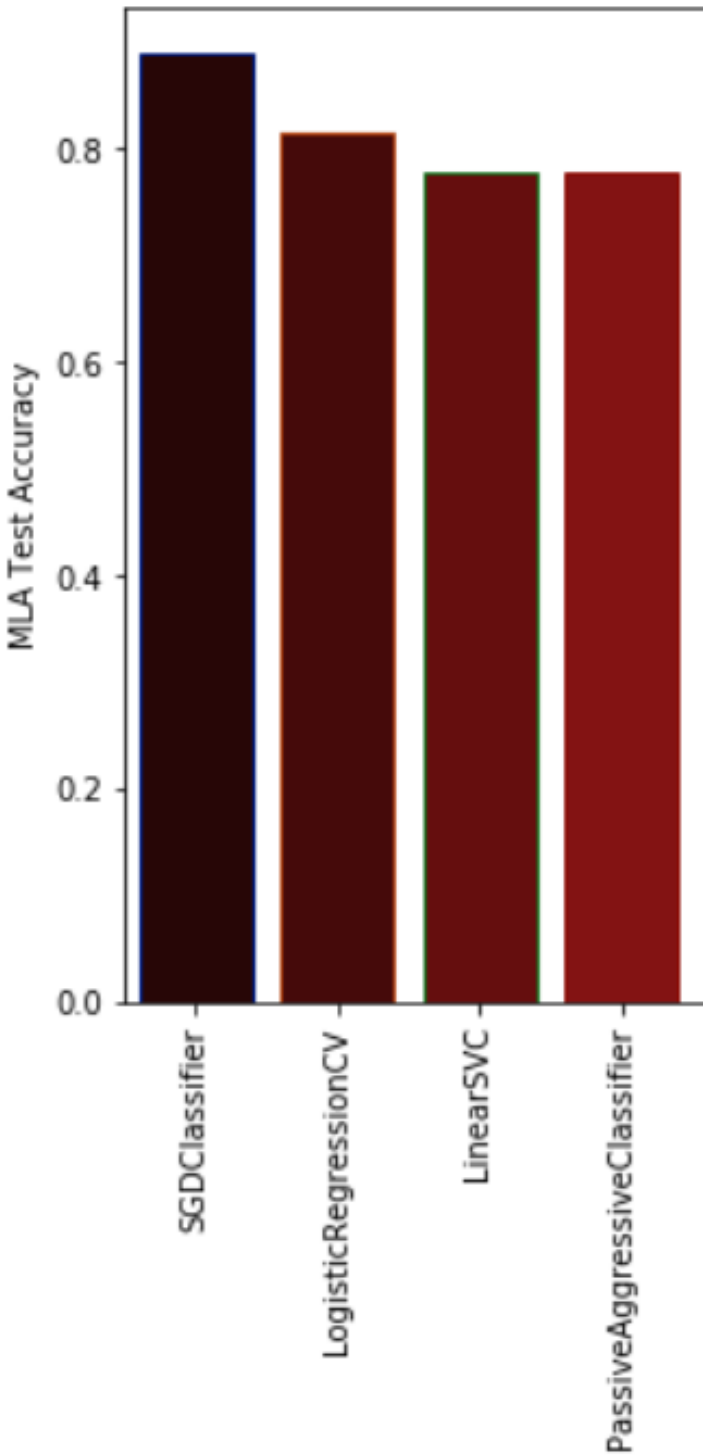


Figure 4.26: Accuracy Result for Algorithm applied on Statistical Windowing Feature Dataset

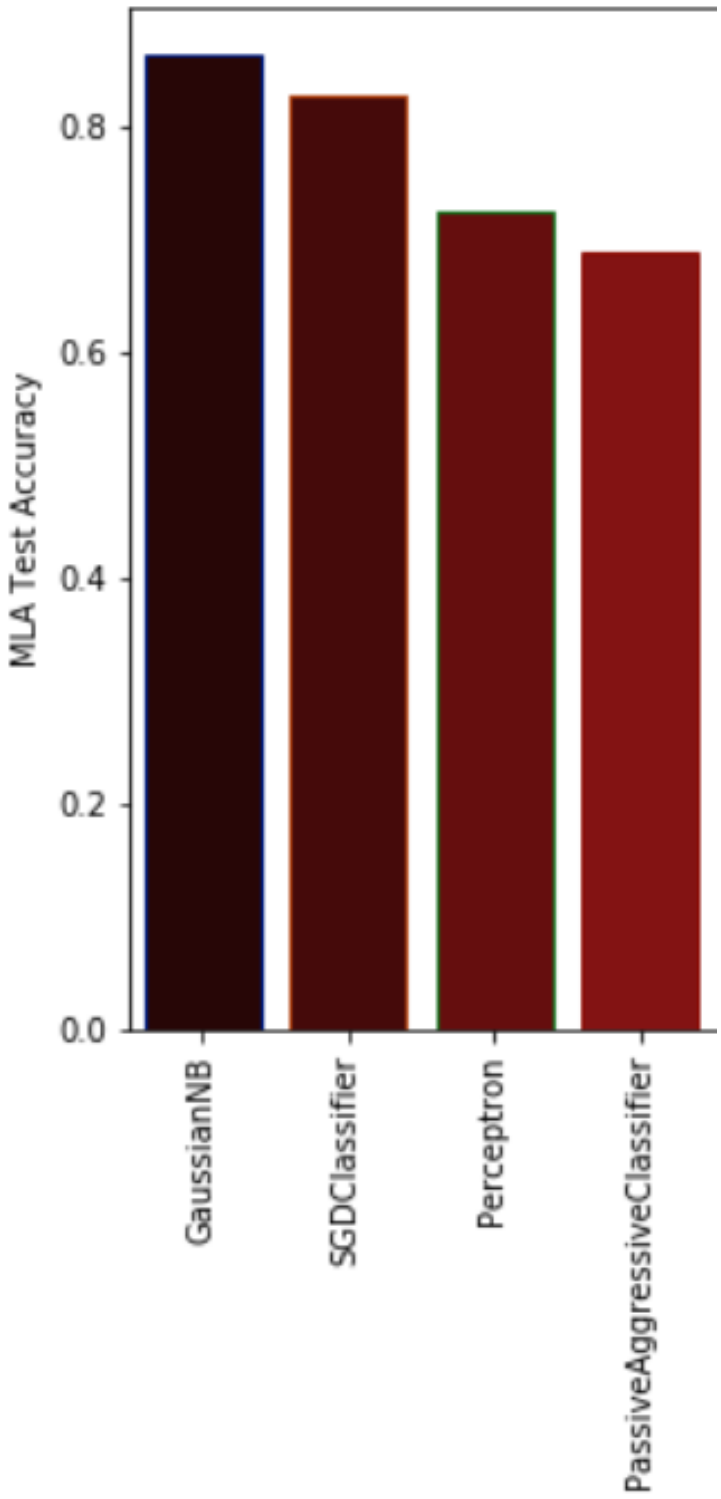


Figure 4.27: Accuracy Result for Algorithm applied on Advanced Windowing Feature Dataset

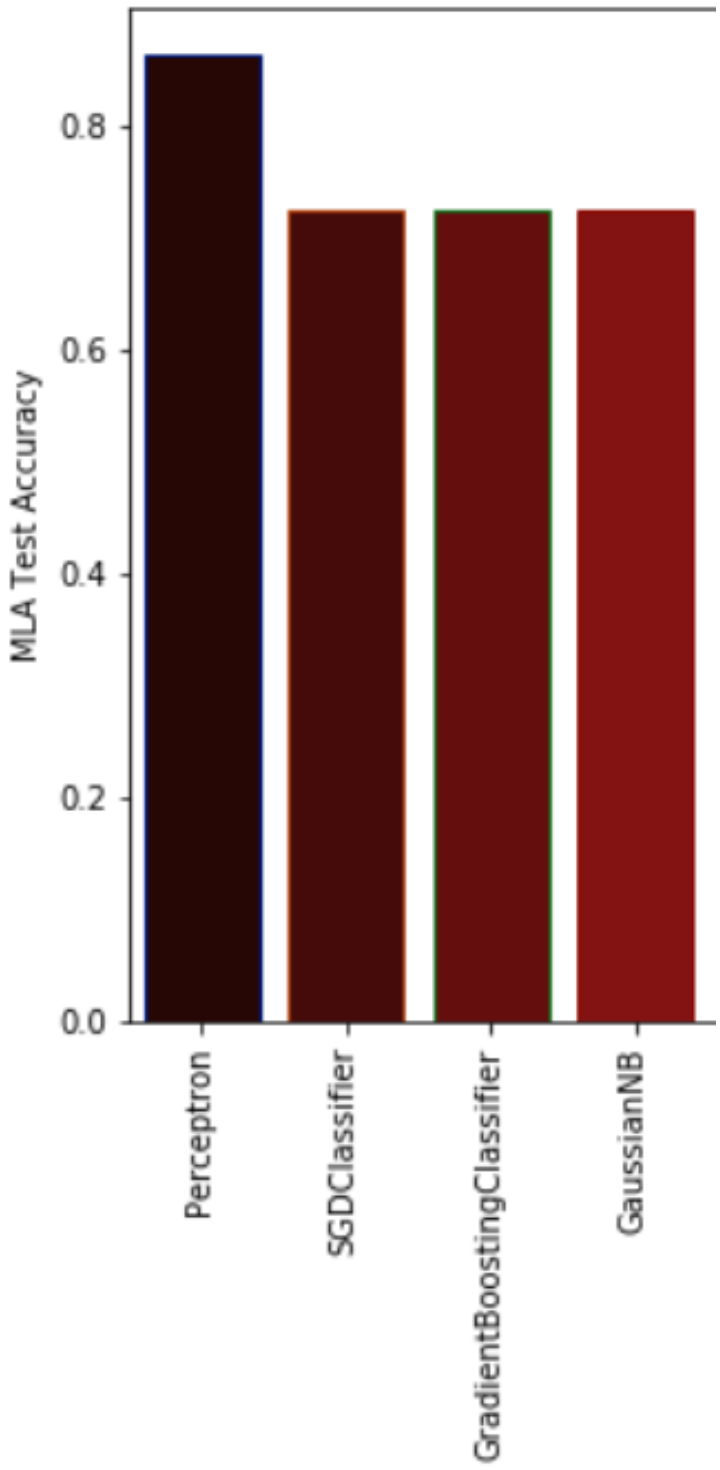


Figure 4.28: Accuracy Result for Algorithm applied on Advanced and Statistical Fusion Feature Windowing Dataset

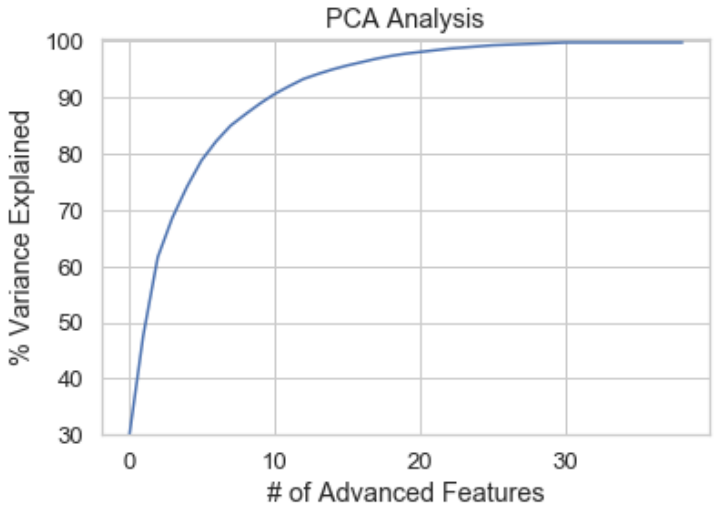


Figure 4.29: PCA on Fusion of Advanced and Statistical Feature Dataset

```
*** MaxRel features ***
Order    Fea     Name    Score
1        31      EnergySTFTAlphaHigh     0.029
2        7       EnergySTFTDelta         0.029
3        39      EnergySTFTBetaLow       0.029
4        55      EnergySTFTGammaLow      0.029
5        15      EnergySTFTTheta         0.029
6        63      EnergySTFTGammaMid      0.029
7        23      EnergySTFTAlphaLow      0.029
8        47      EnergySTFTBetaHigh      0.029
9        38      EnergyDWTcDBetaLow      0.013
10       79      SkewnessalphaLow        0.013
11       77      MeanalphaLow    0.013
12       80      KurtosisalphaLow        0.013
13       81      StdDeviaalphaLow        0.013
14       41      EnergyDCTBetaHigh       0.013
15       42      EnergyIDCTBetaHigh      0.013
16       43      EnergyFFTBetaHigh       0.013
17       44      EnergyIFFTBetaHigh      0.013
18       45      EnergyDWTcABetaHigh     0.013
19       46      EnergyDWTcDBetaHigh     0.013
20       48      EnergyWVDBetaHigh       0.013
21       24      EnergyWVDAlphaLow       0.013
22       82      RMSalphaLow     0.013
23       17      EnergyDCTAlphaLow       0.013
24       95      MeanbetaHigh    0.013
25       99      StdDeviabetaHigh        0.013
26       96      MedianbetaHigh          0.013
27       14      EnergyDWTcDTheta        0.013
28       100     RMSbetaHigh     0.013
29       18      EnergyIDCTAlphaLow      0.013
30       20      EnergyIFFTAlphaLow      0.013
```

Figure 4.30: Selected Features from Fusion of Advanced and Statistical Feature Dataset applying mRMR

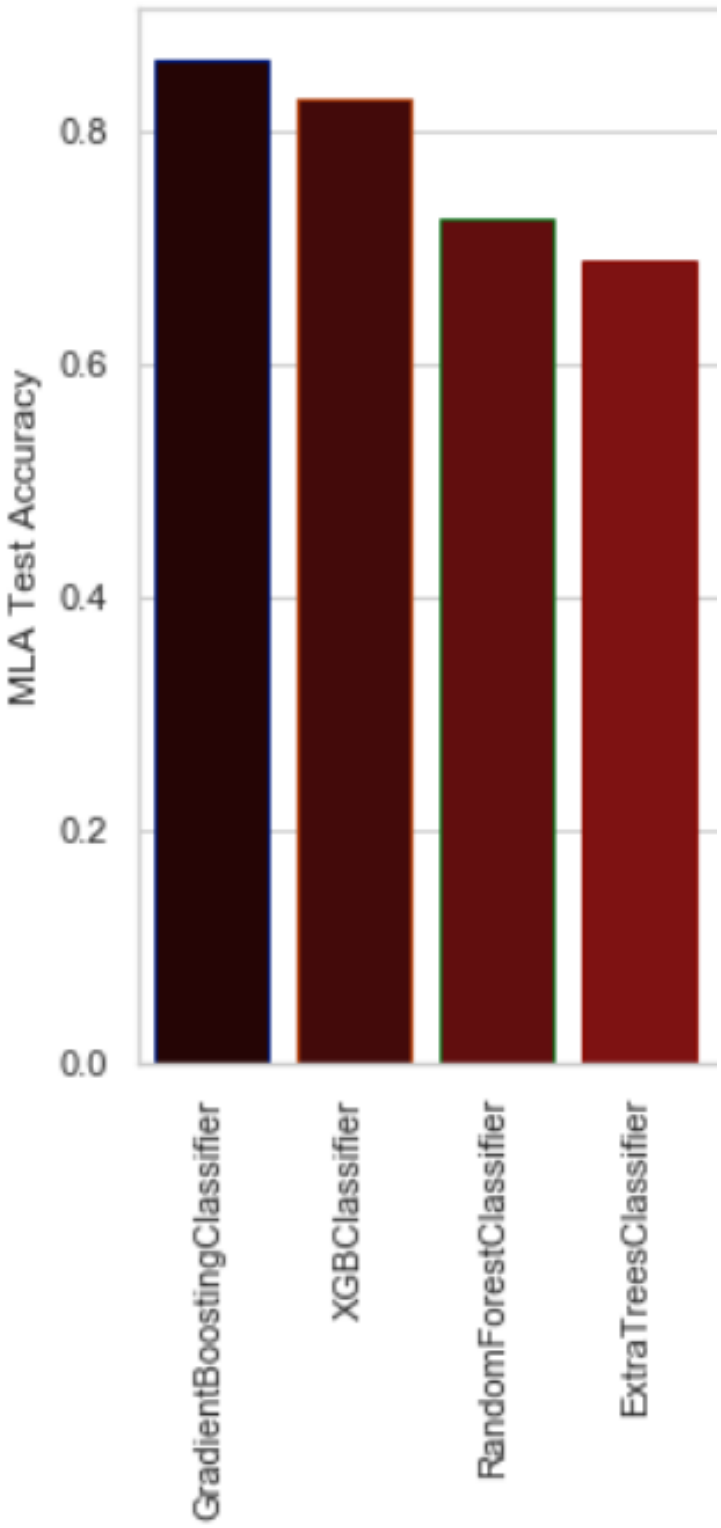


Figure 4.31: Accuracy Result applying Machine Learning Algorithms on Fusion of Advanced and Statistical mRMR Selected Feature Dataset

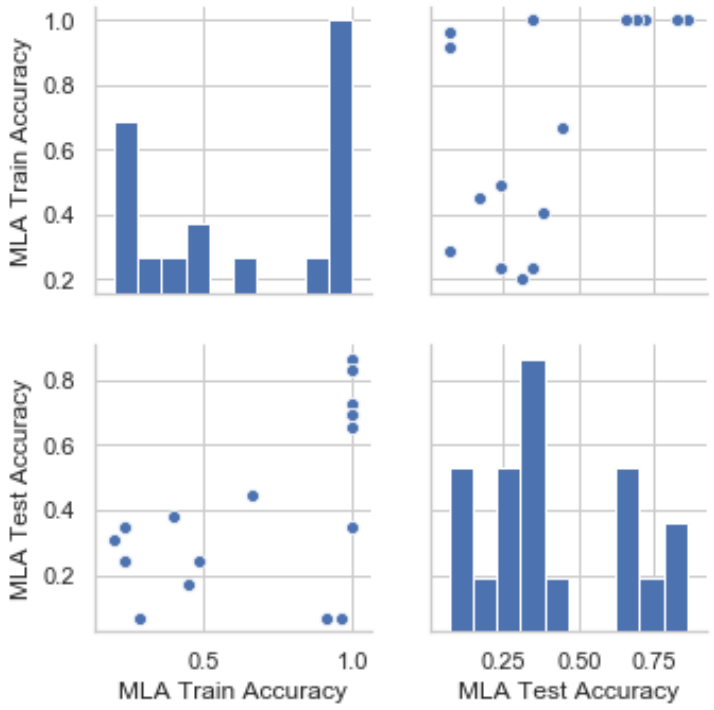


Figure 4.32: Pair Plot on Fusion of Advanced and Statistical mRMR Selected Feature Dataset

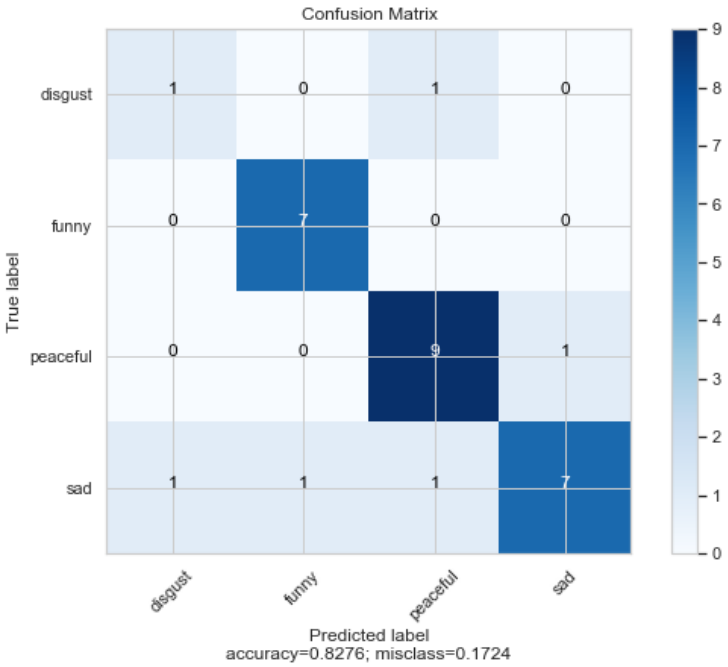


Figure 4.33: Confusion Matrix on Fusion of Advanced and Statistical mRMR Selected Feature Dataset

MLA Name	MLA Train Accuracy(%)	MLA Test Accuracy(%)
AdaBoostClassifier	52.56	85.19
GaussianNB	69.23	77.78
DecisionTreeClassifier	100.00	74.07
SGDClassifier	65.38	66.67
XGBClassifier	100.00	66.67

Table 4.4: Accuracy Result for Algorithm applied on Advanced Feature Dataset

MLA Name	ExtraTreesClassifier	XGBClassifier	BaggingClassifier
MLA Precision(Macro)(%)	75.8929	68.75	70.00
MLA Precision(Micro)(%)	75.8621	72.4138	72.4138
MLA Precision(Weighted)(%)	86.4532	87.069	83.3333
MLA Recall(Macro)(%)	81.4286	58.9286	67.8571
MLA Recall(Micro)(%)	75.8621	72.4138	72.4138
MLA Recall(Weighted)(%)	75.8621	72.4138	72.4138
F1 Score(Macro)(%)	73.0769	62.8105	66.069
F1 Score(Micro)(%)	75.8621	72.4138	72.4138
F1 Score(Weighted)(%)	78.9861	78.3593	76.2756

Table 4.5: Precision, Recall, F1 Score for Algorithm applied on Advanced Feature Dataset

MLA Name	MLA Train Accuracy(%)	MLA Test Accuracy(%)
Random Forest Classifier	100.00	89.66
XGB Classifier	100.00	86.21
Gradient Boosting Classifier	100.00	82.76
Perceptron	59.77	79.31
Gaussian NB	71.26	79.31

Table 4.6: Accuracy Result for Algorithm applied on Advanced and Statistical Feature Dataset

Thereby, we have considered to fusion the features and received accuracy level 89.66%, which is found to be the highest among all experiments. And this highest accuracy is found using Random Forest classification method. We have tried to further increase the accuracy using window method where we have observed good result 88.89% for statistical features and 86.21% for advanced features. The overall fusion features using window is found to be 86.21%. At the end, we have used ensemble max voting technique and found stable 88% accurate results at all times.

MLA Name	Gradient Boosting Classifier	Gaussian NB	Percentron	Bagging Classifier
MLA Precision (Macro)(%)	70.4212	68.1250	68.3333	67.3295
MLA Precision (Micro)(%)	75.8621	72.4138	72.4138	72.4138
MLA Precision (Weighted)(%)	79.7524	81.6379	78.1609	78.8793
MLA Recall (Macro)(%)	69.7511	67.4784	68.7771	67.4784
MLA Recall (Micro)(%)	75.8621	72.4138	72.4138	72.4138
MLA Recall (Weighted)(%)	75.8621	72.4138	72.4138	72.4138
F1 Score (Macro)(%)	68.6905	65.2381	66.6947	65.4545
F1 Score (Micro)(%)	75.8621	72.4138	72.4138	72.4138
F1 Score (Weighted)(%)	76.4368	75.4023	73.5246	74.2529

Table 4.7: Precision, Recall, F1 Score for Algorithm applied on Advanced and Statistical Feature Dataset

MLA Name	MLA Train Accuracy(%)	MLA Test Accuracy(%)
XGBClassifier	100.00	86.21
RandomForestClassifier	98.82	79.31
GradientBoostingClassifier	100.00	68.97
DecisionTreeClassifier	100.00	65.52

Table 4.8: Accuracy Result for Algorithm applied on Advanced and Statistical Selected Feature Dataset

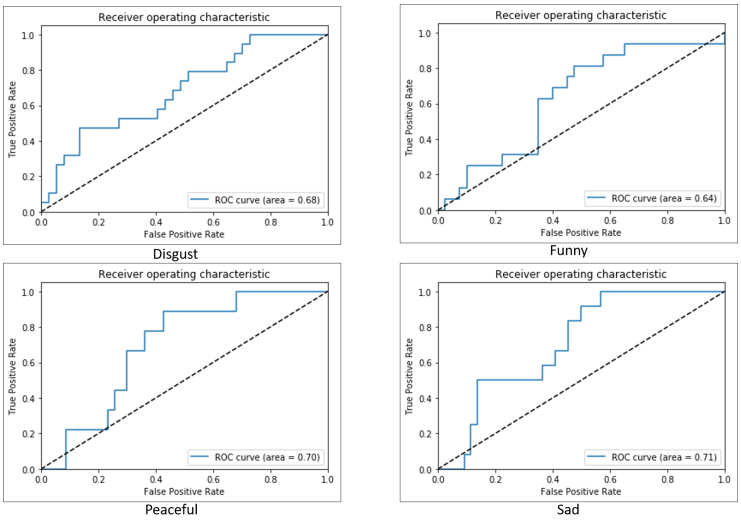


Figure 4.34: ROC on Fusion of Advanced and Statistical mRMR Selected Feature Dataset

MLA Name	XGB Classifier	Bagging Classifier	Gradient Boosting Classifier	Decision Tree Classifier
MLA Precision (Macro)(%)	78.4722	76.2500	68.4821	69.0972
MLA Precision (Micro)(%)	86.2069	75.8621	65.5172	65.5172
MLA Precision (Weighted)(%)	91.3793	90.0000	81.3054	81.9923
MLA Recall (Macro)(%)	78.8781	79.6898	61.1291	61.6342
MLA Recall (Micro)(%)	86.2069	75.8621	65.5172	65.5172
MLA Recall (Weighted)(%)	86.2069	75.8621	65.5172	65.5172
F1 Score (Macro)(%)	77.4420	71.2487	60.2354	60.8289
F1 Score (Micro)(%)	86.2069	75.8621	65.5172	65.5172
F1 Score (Weighted)(%)	88.2910	80.0014	70.8430	71.1857

Table 4.9: Precision, Recall, F1 Score for Algorithm applied on Advanced and Statistical Selected Feature Dataset

MLA Name	MLA Train Accuracy(%)	MLA Test Accuracy(%)
SGDClassifier	90.00	88.89
LogisticRegressionCV	100.00	81.48
LinearSVC	100.00	77.78
PassiveAggressiveClassifier	98.75	77.78

Table 4.10: Accuracy Result for Algorithm applied on Statistical Windowing Feature Dataset

MLA Name	MLA Train Accuracy(%)	MLA Test Accuracy(%)
GaussianNB	86.05	86.21
SGDClassifier	76.74	82.76
Perceptron	88.37	72.41
PassiveAggressiveClassifier	96.51	68.97

Table 4.11: Accuracy Result for Algorithm applied on Advanced Windowing Feature Dataset

MLA Name	MLA Train Accuracy(%)	MLA Test Accuracy(%)
Perceptron	82.14	86.21
SGDClassifier	88.10	72.41
GradientBoostingClassifier	100.00	72.41
GaussianNB	90.48	72.41

Table 4.12: Accuracy Result for Algorithm applied on Advanced and Statistical Fusion Feature Windowing Dataset

MLA Name	MLA Train Accuracy(%)	MLA Test Accuracy(%)
GradientBoostingClassifier	100.00	86.21
XGBClassifier	100.00	82.76
RandomForestClassifier	100.00	72.41
ExtraTreesClassifier	100.00	68.97

Table 4.13: Accuracy Result for Algorithm applied on Advanced and Statistical Fusion mRMR Selected Feature Dataset

MLA Name	MLA Precision(%)	MLA Recall(%)	F1 Score(%)
Random Forest Classifier	85.7143	85.7143	85.7143
Gradient Boosting Classifier	78.5714	78.5714	78.5714
XGB Classifier	75.00	75.00	75.00
Extra Trees Classifier	67.8571	67.8571	67.8571

Table 4.14: Precision, Recall, F1 Score on Advanced and Statistical Fusion mRMR Selected Feature Dataset

CHAPTER 5
CONCLUSION

5.1 Future Work

The research looks forward for introducing multi model approach. Besides, the emotion recognition research can lead to produce mind controlled car, robot with emotion. Besides the research leads to path on study of human emotion using human signals at advanced level.

5.2 Conclusion

Knowing about human emotion will milestone in the field of science for study of physiology and crime detection and also for development of applications and smart devices. Moreover, ti will help people with autism and other similar disorder to express their emotions. This research can lead to revolutionary change in treatment of people who have problem in expressing their emotion.

REFERENCES

[1] A. R. Damasio, "Emotion in the perspective of an integrated nervous system," *Brain research reviews*, vol. 26, no. 2-3, pp. 83–86, 1998.

[2] P. E. Ekman and R. J. Davidson, *The nature of emotion: Fundamental questions.* Oxford University Press, 1994.

[3] M. Cabanac, "What is emotion?," *Behavioural processes*, vol. 60, no. 2, pp. 69–83, 2002.

[4] J. Preece, Y. Rogers, H. Sharp, D. Benyon, S. Holland, and T. Carey, *Human-computer interaction*. Addison-Wesley Longman Ltd., 1994.

[5] R. Plutchik, "The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice," *American scientist*, vol. 89, no. 4, pp. 344–350, 2001.

[6] J. A. Russell, "A circumplex model of affect.," *Journal of personality and social psychology*, vol. 39, no. 6, p. 1161, 1980.

[7] L. Cohen, "It's about time," *Frontiers in Human Neuroscience*, vol. 5, no. 2, pp. 1–15, 2011.

[8] N. . L. da Silva, "Electroencephalography: Basic principles, clinical applications, and related fields," 2012.

[9] D. A. G. J. Sleigh JW, Olofsen E and S.-R. A, "Entropies of the eeg: the effects of general anesthesia," in *5th International Conf. on Memory, Awareness and Consciousness*, pp. 1–3, 2001.

[10] E. Alpaydin, *Introduction to machine learning*. MIT press, 2009.

[11] "Cross validation." https://www.cs.cmu.edu/~schneide/tut5/node42.htm.

[12] S. Koelstra, C. Muhl, M. Soleymani, J.-S. Lee, A. Yazdani, T. Ebrahimi, T. Pun, A. Nijholt, and I. Patras, "Deap: A database for emotion analysis; using physiological signals," *IEEE transactions on affective computing*, vol. 3, no. 1, pp. 18–31, 2011.

[13] V. Mavani, S. Raman, and K. P. Miyapuram, "Facial expression recognition using visual saliency and deep learning," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2783–2788, 2017.

[14] M. M. Hassan, M. G. R. Alam, M. Z. Uddin, S. Huda, A. Almogren, and G. Fortino, "Human emotion recognition using deep belief network architecture," *Information Fusion*, vol. 51, pp. 10–18, 2019.

[15] S. Tripathi, S. Acharya, R. D. Sharma, S. Mittal, and S. Bhattacharya, "Using deep and convolutional neural networks for accurate emotion classification on deap dataset.," in *Twenty-Ninth IAAI Conference*, 2017.

[16] M. G. R. Alam, S. F. Abedin, S. I. Moon, A. Talukder, and C. S. Hong, "Healthcare iot-based affective state mining using a deep convolutional neural network," *IEEE Access*, 2019.

[17] Z. Mohammadi, J. Frounchi, and M. Amiri, "Wavelet-based emotion recognition system using eeg signal," *Neural Computing and Applications*, vol. 28, no. 8, pp. 1985–1990, 2017.

[18] M. Ragot, N. Martin, S. Em, N. Pallamin, and J.-M. Diverrez, "Emotion recognition using physiological signals: laboratory vs. wearable sensors," in *International Conference on Applied Human Factors and Ergonomics*, pp. 15–22, Springer, 2017.

[19] N. Zhuang, Y. Zeng, L. Tong, C. Zhang, H. Zhang, and B. Yan, "Emotion recognition from eeg signals using multidimensional information in emd domain," *BioMed research international*, vol. 2017, 2017.

[20] W.-L. Zheng, J.-Y. Zhu, and B.-L. Lu, "Identifying stable patterns over time for emotion recognition from eeg," *IEEE Transactions on Affective Computing*, 2017.

[21] R. Xia and Y. Liu, "A multi-task learning framework for emotion recognition using 2d continuous space," *IEEE Transactions on Affective Computing*, vol. 8, no. 1, pp. 3–14, 2015.

[22] H. Chao, L. Dong, Y. Liu, and B. Lu, "Emotion recognition from multiband eeg signals using capsnet," *Sensors*, vol. 19, no. 9, p. 2212, 2019.

[23] H. Ullah, M. Uzair, A. Mahmood, M. Ullah, S. D. Khan, and F. A. Cheikh, "Internal emotion classification using eeg signal with sparse discriminative ensemble," *IEEE Access*, vol. 7, pp. 40144–40153, 2019.

[24] J. Arunnehru and M. K. Geetha, "Automatic human emotion recognition in surveillance video," in *Intelligent Techniques in Signal Processing for Multimedia Security*, pp. 321–342, Springer, 2017.

[25] S. Chen, Q. Jin, J. Zhao, and S. Wang, "Multimodal multi-task learning for dimensional and continuous emotion recognition," in *Proceedings of the 7th Annual Workshop on Audio/Visual Emotion Challenge*, pp. 19–26, ACM, 2017.

[26] Y. Li, J. Huang, H. Zhou, and N. Zhong, "Human emotion recognition with electroencephalographic multidimensional features by hybrid deep neural networks," *Applied Sciences*, vol. 7, no. 10, p. 1060, 2017.

[27] L. Chen, M. Zhou, W. Su, M. Wu, J. She, and K. Hirota, "Softmax regression based deep sparse autoencoder network for facial emotion recognition in human-robot interaction," *Information Sciences*, vol. 428, pp. 49–61, 2018.

[28] M. M.-E.-N. R. Lafiz Maruf Rahman, Zawad Alam, "Eeg signals analysis for motor imagery brain-computer interface," in *A thesis for BSc in Computer Science*, pp. 15–29, 2019.

APPENDIX A
ALGORITHMS

A.1 Data Preprocessing Algorithms

Normalization Process

Input: Matrix M

Output: Normalized Matrix N

Begin

for each line L in M do

a = average (L)

v = variance (L)

for each value x of L do

$x = (x - a) / \sqrt{v}$

endfor

endfor

End

Complexity=O (n*m)

Figure A.1: Normalization

A.2 Advanced Features Algorithms

```
RECURSIVE-FFT(a)
 1  n ← length[a]          ▷ n is a power of 2.
 2  if n = 1
 3     then return a
 4  ω_n ← e^(2πi/n)
 5  ω ← 1
 6  a^[0] ← (a_0, a_2, ..., a_{n-2})
 7  a^[1] ← (a_1, a_3, ..., a_{n-1})
 8  y^[0] ← RECURSIVE-FFT(a^[0])
 9  y^[1] ← RECURSIVE-FFT(a^[1])
10  for k ← 0 to n/2 − 1
11        do y_k ← y_k^[0] + ω y_k^[1]
12           y_{k+(n/2)} ← y_k^[0] − ω y_k^[1]
13           ω ← ω ω_n
14  return y               ▷ y is assumed to be column vector.
```

Figure A.2: FFT Analysis

A.3 Feature Extraction Algorithms

Input : Data set $\mathcal{D}$.
A loss function L.
A base learner $\mathcal{L}_{\Phi}$.
The number of iterations M.
The learning rate η.

Initialize $\hat{f}^{(0)}(x) = \hat{f}_0(x) = \hat{\theta}_0 = \arg\min_{\theta} \sum_{i=1}^{n} L(y_i, \theta)$;

for $m = 1,2,..,M$ **do**

$\hat{g}_m(x_i) = \left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f(x)=\hat{f}^{(m-1)}(x)}$;

$\hat{\phi}_m = \arg\min_{\phi \in \Phi, \beta} \sum_{i=1}^{n} \left[\left(-\hat{g}_m(x_i)\right) - \beta\phi(x_i)\right]^2$;

$\hat{\rho}_m = \arg\min_{\rho} \sum_{i=1}^{n} L(y_i, \hat{f}^{(m-1)}(x_i) + \rho\hat{\phi}_m(x_i))$;

$\hat{f}_m(x) = \eta\hat{\rho}_m\hat{\phi}_m(x)$;

$\hat{f}^{(m)}(x) = \hat{f}^{(m-1)}(x) + \hat{f}_m(x)$;

end

Output: $\hat{f}(x) \equiv \hat{f}^{(M)}(x) = \sum_{m=0}^{M} \hat{f}_m(x)$

Figure A.3: XGBoost-Gradient Boosting

Input: dataset $D = \{(X_i, c_i)\}_{i=1}^{N} \in \mathbb{R}^{G \times T} \times \{1,...,K\}$, $algorithm \in \{TMRMR-M, TMRMR-C\}$; number of features to select m, α
Output: Feature set S
1. $S_{ALL} \leftarrow \{1,2,...,G\}$
2. $S \leftarrow \varnothing$
3. $S_a \leftarrow \varnothing$
4. **for each** $g_j \in \mathbb{R}^{N \times T}$ **do**
5. $F(g_j, c) = temporalRelevance(g_j, c)$ // Eqs. (1)-(2)
6. **end for**
7. **for** $i = 1$ **to** $round(\alpha G)$ **do** // function $round(\alpha G)$ rounds αG to nearest integer
8. $S_a \leftarrow S_a \cup \arg\max_{j \in S_{ALL} \setminus S_a} F(g_j, c)$
9. **end for**
10. $S \leftarrow \arg\max_{j \in S_{ALL}} F(g_j, c)$
11. **while** $length(S) < m$ **do** // function $length(S)$ returns the number of elements in S
12. **for each** $k \in S_a \setminus S$ **do**
13. $S' \leftarrow S \cup k$
14. $W_{mw}(g_k) = \frac{1}{length(S')^2} \sum_{i,j \in S'} R(g_i, g_j)$ // $R(g_i, g_j) = R_c(g_i, g_j)$ if $algorithm = TMRMR-C$ (Eq. (3))
// $R(g_i, g_j) = R_m(g_i, g_j)$ if $algorithm = TMRMR-M$ (Eq. (4))
15. $V_F(g_k) = \frac{1}{length(S')} \sum_{i \in S'} F(g_i, c)$
16. **end for**
17. $S \leftarrow S \cup \arg\max_k \left(\frac{V_F(g_k)}{W_{mw}(g_k)}\right)$
18. **end while**
19. **return** S

Figure A.4: mRMR

Algorithm 1 The PCA algorithm

1: **Input**: a D-dimensional training set $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$ and the new (lower) dimensionality d (with $d \leq D$)

2: Compute the mean $\bar{\boldsymbol{x}} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i$

3: Compute the covariance matrix $\mathrm{Cov}(\boldsymbol{x}) = \frac{1}{N}\sum_{i=1}^{N} (\boldsymbol{x}_i - \bar{\boldsymbol{x}}_i)(\boldsymbol{x}_i - \bar{\boldsymbol{x}}_i)^T$

4: Find the spectral decomposition of $\mathrm{Cov}(\boldsymbol{x})$, obtaining the eigenvectors $\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_D$ and their corresponding eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_D$. Note that the eigenvalues are sorted, such that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_D \geq 0$

5: For any $\boldsymbol{x} \in \mathbb{R}^D$, its new lower dimensional representation is:

$$\boldsymbol{y} = \left(\boldsymbol{\xi}_1^T(\boldsymbol{x} - \bar{\boldsymbol{x}}), \boldsymbol{\xi}_2^T(\boldsymbol{x} - \bar{\boldsymbol{x}}), \ldots, \boldsymbol{\xi}_d^T(\boldsymbol{x} - \bar{\boldsymbol{x}})\right)^T \in \mathbb{R}^d, \tag{29}$$

and the original $\boldsymbol{x}$ can be approximated as

$$\boldsymbol{x} \approx \bar{\boldsymbol{x}} + (\boldsymbol{\xi}_1^T(\boldsymbol{x} - \bar{\boldsymbol{x}}))\boldsymbol{\xi}_1 + (\boldsymbol{\xi}_2^T(\boldsymbol{x} - \bar{\boldsymbol{x}}))\boldsymbol{\xi}_2 + \cdots + (\boldsymbol{\xi}_d^T(\boldsymbol{x} - \bar{\boldsymbol{x}}))\boldsymbol{\xi}_d \tag{30}$$

Figure A.5: PCA

Algorithm 6 LDA Algorithm

Require: data matrix $X \in R^{d\times N}$, $\mathbf{x}_i \in R^{d\times 1}$ is the i-th column of X. label vector $y_i \in \{1, 2, \cdots, C\}, i = 1, \cdots, N$, $N_c, c = 1, \cdots, C$ is the number of samples of each class.

Ensure: The projection matrix $P \in R^{p\times d}$

1: compute mean vector for each class: $\mathbf{m}_c = \frac{\sum_{i=1}^{N_c} \mathbf{x}_i}{N_c} \in R^{d\times 1}$

2: compute total mean vector: $\mathbf{m} = \frac{\sum_{i=1}^{N} \mathbf{x}_i}{N} \in R^{d\times 1}$

3: compute within-class scatter: $S_w = \sum_{c=1}^{C}\sum_{j=1}^{N_c} (\mathbf{x}_j - \mathbf{m}_c)(\mathbf{x}_j - \mathbf{m}_c)^T \in R^{d\times d}$

4: compute between-class scatter: $S_b = \sum_{c=1}^{C} N_c(\mathbf{m}_c - \mathbf{m})(\mathbf{m}_c - \mathbf{m})^T \in R^{d\times d}$

5: eigen-decomposition: $[V, D] = eig({S_w}^{-1}S_b)$

6: get the projection matrix: P is composed of the top-p eigenvectors corresponding to the largest eigenvalues.

Figure A.6: LDA

Algorithm: T-Distributed Stochastic Neighbor Embedding (t-SNE)

INPUT: high-dimensional data set $\mathbf{X} = (\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots\ldots \boldsymbol{x_n})$, where $\mathbf{x}_i \in \mathbb{R}^{K}$;
cost function parameters: perplexity ***Perp***;
optimization parameters: number of iterations **T**, learning rate $\boldsymbol{\eta}$, momentum $\boldsymbol{\alpha(t)}$.

INITIALIZE: sample initial solution $\boldsymbol{Y^{(0)}} = (\boldsymbol{y_1}, \boldsymbol{y_2}, \ldots\ldots \boldsymbol{y_n})$ from $\mathcal{N}(\mathbf{0}, \mathbf{10^{-4}}\boldsymbol{I})$, where $\mathbf{y}_i \in \mathbb{R}^{k}$.

BEGIN
compute $\mathbf{p}_{j|i}$ with perplexity ***Perp***;
set $\mathbf{p}_{ij} = \frac{p_{j|i} + p_{i|j}}{2n}$;
for t=1,2⋯ ⋯ **T**
compute low-dimensional affinities $\mathbf{q}_{ij}$;
compute gradient $\frac{\partial C}{\partial Y}$;
set $\mathbf{Y}^{(t)} = \mathbf{Y}^{(t-1)} + \boldsymbol{\eta}\frac{\partial C}{\partial Y} + \boldsymbol{\alpha(t)}\left(\mathbf{Y}^{(t-1)} - \mathbf{Y}^{(t-2)}\right)$;
end
END

OUTPUT: low-dimensional data representation $\boldsymbol{Y} = (\boldsymbol{y_1}, \boldsymbol{y_2}, \ldots\ldots \boldsymbol{y_n})$, where $\mathbf{y}_i \in \mathbb{R}^{k}$.

Figure A.7: t-SNE