Ain Shams University
Faculty of Computer & Information Sciences
Computer Science Department

Emotion Recognition using Brainwaves

August 2020

Ain Shams University
Faculty of Computer & Information Sciences
Computer Science Department

Emotion Recognition using Brainwaves

By:

Belal Hany Kamal [Computer Science]
Pola Farag Asaad [Computer Science]
Peter Maged Mounir [Computer Science]
George Kameel Barsoum [Computer Science]
John Emil Youhanna [Computer Science]

Under Supervision of :

Dr.Mohammed Marey,
Assistant Professor.
Scientific Computing Department,
Faculty of Computer and Information Sciences,
Ain Shams University.

Acknowledgement

We would like to express our deepest and sincerest gratitude to our research supervisor, Professor Mohammed Marey, for giving us the opportunity to do research and providing invaluable guidance throughout this research. His dynamism, vision, sincerity and motivation have deeply inspired us. He has taught us the methodology to carry out the research and to present the research works as clearly as possible. It was a great privilege and honor to work and study under his guidance. We are extremely grateful for what he has offered us. We would also like to thank him for his friendship, dedication ,empathy, and great sense of humor. We could not have imagined having a better advisor and mentor for our research journey. We would also like to express our sincerest thanks to Ain Shams University for accepting us into the graduate program. We had the pleasure to be one of Ain Shams University students, as it was an incredible chance to gain lots of proficiency in conducting empirical projects all over our voyage in the university.

Also, we would like to give thanks for Eng. Abdalla Amjad (Bio 2019) for his generous help for us. He has dedicated so much from his time to help us. His presence was very helpful for us during our implementation and practical work. His expertise in the domain of our project and research helped us to get over some issues that we faced.

Abstract

Emotions play a major role in humans' life, that emphasizes the importance of detecting our emotions. Multiple algorithms, techniques and approaches are used to classify emotions. This project was an attempt to implement the most famous algorithms in that domain. Multiple approaches were used including Machine Learning models and Deep Learning models. Deep Learning models were trained using the Raw Data with EMD Augmentation, Raw Data and time augmentation and Discrete Wavelet Transform with Power Spectral Density features. The shallow machine learning models including Bagging Tree (BT), Random Forest (RF), Adaptive Boost (AdaBoost), Support Vector Machine (SVM), K-Nearest Neighbor (KNN) and Logistic Regression (LR) models and deep CNN models were used to make emotional binary classification experiments on DEAP datasets in valence and arousal dimensions. The experimental results showed that the deep CNN models which required feature engineering using Discrete Wavelet Transform achieved the best recognition performance on frequency features in valence dimension, which is 7% higher than the performance of the best traditional LR classifier in valence dimension.
On the contrary, the traditional LR classifier which required feature engineering using Discrete Wavelet Transform achieved the best recognition performance on frequency features in arousal dimension, which is 3% higher than the performance of the deep CNN models in arousal dimension.

Table of Contents

List of Figures

Note: Always number your figures and include a caption under each one like this. Then if you update the list above it'll be updated automatically.

When adding a figure, right click on the image -> insert caption.
After you finish the document, write click on the table and choose update field, then update entire table.

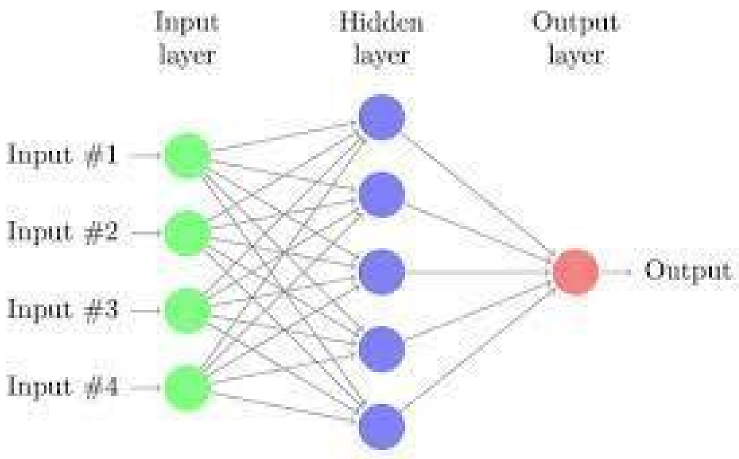


Figure 1- Neural Network general architecture

Add list of Tables if you have tables in your text

List of Abbreviations

Abbreviation	Meaning
CNN	Convolution Neural Network
DEAP	Database for Emotion Analysis using
DWT	Discrete wavelet Transform
EEG	Electroencephalography
EMD	Empirical Mode decomposition
ENG	Energy
ENT	**Entropy**
FFT	Fast Fourier Transform
IMF	Intrinsic Mode Function
RMS Prop	Root Mean Square Propagation
SGD	Sophisticated Gradient descent

Introduction

Chapter One

1.1 Motivation

Everything we hear, read or even think about affects our emotions. This made the emotion recognition a critical need in many fields and affect our lives dramatically, it helps:

- The Psychologist in treating the patients
- Communicating with the people who have difficulties in talking
- Mothers understanding their babies, when they are angry or when they are happy,
- Dealing with old people and many other fields.

This made us feel that helping in getting more information in that field would help a lot. So that people could just know themselves more and try to help themselves if they can or at least ask for help.

The domain of Brain-Computer Interface (BCI) has been given a lot of interest and attention during the past couple of years, as many universities have dedicated their research labs and resources just to explore more in the human and his brain. This also helped us to gain more information about that domain and get more interested in it.

Helping the people was one of the most important motives that made us research and work in that domain, and in our opinion any effort done by anyone just to help others is worth the pain behind.

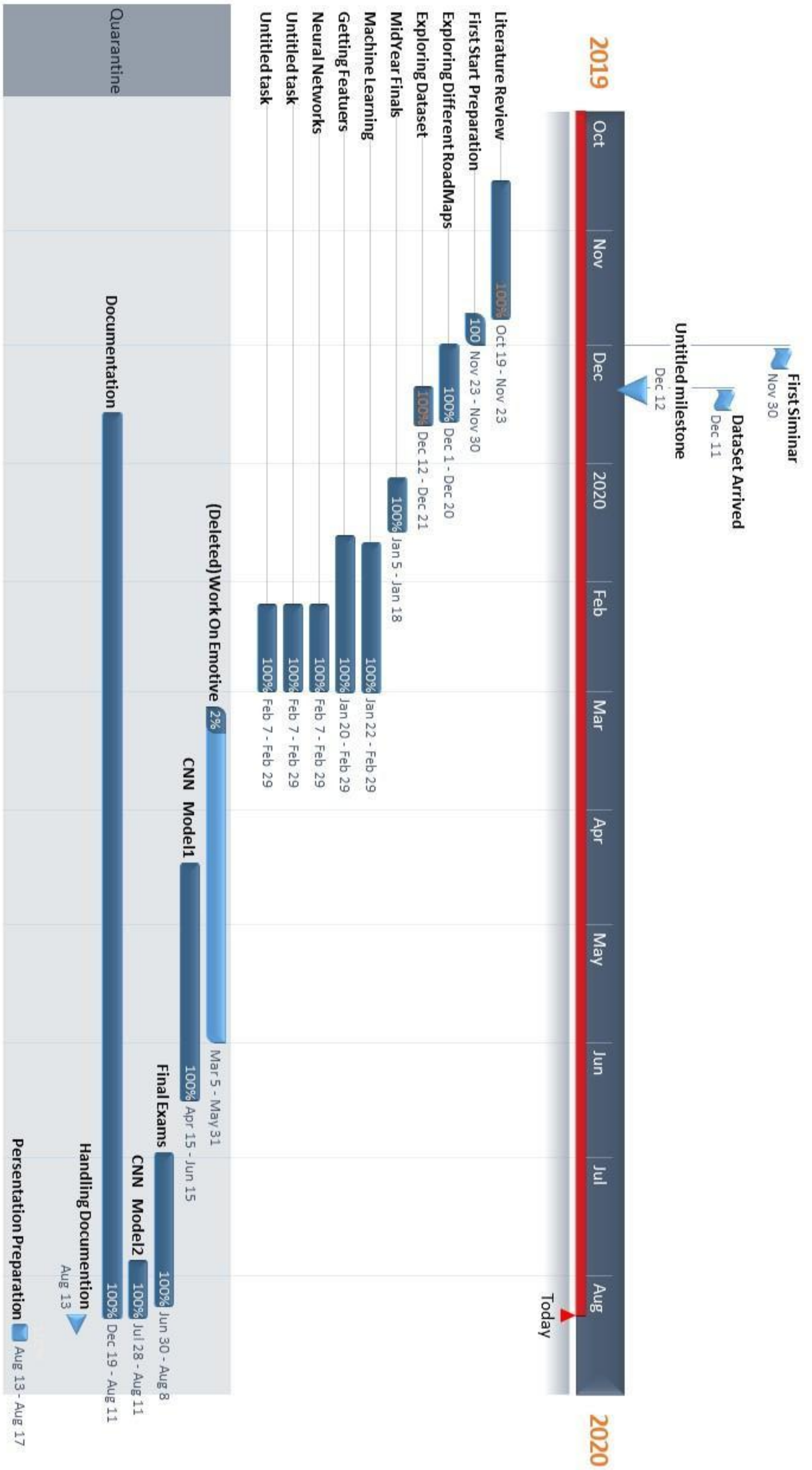
1.2 Problem Definition

Emotion recognition is very complicated. That made many systems try to detect it by using some techniques like voice analysis or facial expression that require very strict conditions, and can be deceived easily. But on the other hand, no one can deceive his own mind, so recognition using EEG signals is an effective yet a complex way to detect emotion, due to the complexity of the signals.

1.3 Objective

Deliver a system that could classify human emotions using EEG (Brainwaves) signals and to have the ability to compete with other sentiment analysis techniques (Face and Voice Recognition).

1.4 Time Plan


2019
Oct
Nov
Dec
2020
Feb
Mar
Apr
May
Jun
Jul
Aug
2020
First Siminar
Nov 30
DataSet Arrived
Dec 11
Untitled milestone
Dec 12
Today
Literature Review
100% Oct 19 - Nov 23
First Start Preparation
100 Nov 23 - Nov 30
Exploring Different RoadMaps
100% Dec 1 - Dec 20
Exploring Dataset
100% Dec 12 - Dec 21
MidYear Finals
100% Jan 5 - Jan 18
Machine Learning
100% Jan 22 - Feb 29
Getting Features
100% Jan 20 - Feb 29
Neural Networks
100% Feb 7 - Feb 29
Untitled task
100% Feb 7 - Feb 29
Untitled task
100% Feb 7 - Feb 29
(Deleted) Work On Emotive
2% Mar 5 - May 31
CNN Model1
100% Apr 15 - Jun 15
Final Exams
100% Jun 30 - Aug 8
CNN Model2
100% Jul 28 - Aug 11
Documentation
100% Dec 19 - Aug 11
Handling Documention
Aug 13
Persentation Preparation
Aug 13 - Aug 17
Quarantine

1.5 Document Organization

Chapter 2 is the Background Information needed by anyone to understand the definitions and theories used in this project. In that chapter we discussed in detail the emotions and their effect on people's behaviour. Also, we discussed where the emotions reside in the brain and how the brain translates those emotions in order to produce the corresponding hormone that helps the person in that action. Some technical background is discussed also in that chapter so that if someone doesn't know any of the techniques and algorithms used in this project he can know more about it from that section. Technical background contains some information about SVM, KNN, Logistic Regression, Random Forests, AdaBoost and CNNs.

Chapter 3 is talking about the analysis and design of the project. We discussed in detail the structure of the dataset used in training and testing our models (DEAP dataset), the properties of the people participated in recording the data and the preprocessing done. And because we used so many architectures in our program so the system architecture is moved into the implementation section along with the inputs, outputs and results.

Chapter 4 is the Implementation and Testing chapter, in that chapter we discussed the features extracted to be used in our models (either the machine learning models or the deep learning models), the algorithm of extracting it and why it might be useful to be used. Also, we discussed an algorithm we used to augment the data so the input data will be larger and better, and we wrote an example of augmenting some signals along with the constraints on it and why it is good in augmenting the data. Lastly, we discussed the different models we implemented for our program, with the architecture used (if available) and the results for different inputs and outputs and differences in between.

Chapter 5 is the user manual, where at every part we discuss how to use the attached scripts in order to get the data for each part whether it is augmentation, feature extraction or running the model with different ideas.

Chapter 6 is the conclusion and future work where we discuss the results we got from all the models and which is better to be used in our case. And also discuss what needs to be done after to increase the accuracy and make our project better.

Background

Chapter Two

Emotions in Humans[1]

Emotions can play an important role in how we think and behave. The emotions we feel each day can compel us to take action and influence the decisions we make about our lives, both large and small.

Emotions also affect our biological system as it controls the secretion of many hormones that help the body reply back to the external stimuli we got.

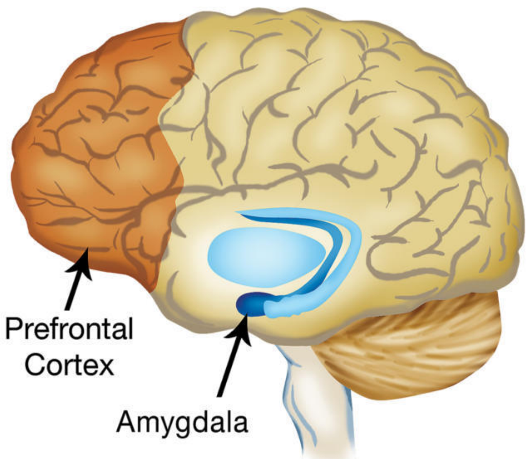
Emotions also help in making our decision, from choosing what to eat on breakfast to choosing your vote in elections for example. Researchers found that brain damage that affects emotion expression can lead to reducing the ability to make a decision.

Emotions help us to understand others and helps others understand us. Emotions can be translated into body language that helps others understand that you got some problems or not feeling good that they may help you, also people can directly state to others how they feel so that others can help them or at least try not to burden them more.

Lastly, Emotions are so important for human beings because it helps them understand themselves and others.

Emotions in the Brain [24]

It has been proven by a number of neuropsychological studies that there are, in fact, correlations between EEG signals and emotions. There are two main areas in the brain that are related to emotional activity: the amygdala (one of two almond-shaped clusters of nuclei located deep and medially within the temporal lobes of the brain); and the pre-frontal cortex (the cerebral cortex which covers the front part of the frontal lobe). The amygdala's activation is believed to be more related to negative emotions than positive ones.



[1] These information are summarized from the article written by Kendra Cherry and reviewed by Amy Morin, LCSW on verywellmind.com. Link for the article is found in reference number [22]

The asymmetry between different parts of the brain in different power bands is related to emotions. In the alpha band, a relatively higher right frontal activation is associated with negative emotions, such as fear or disgust. However, a relatively higher left frontal activation is associated with positive emotions, such as joy or happiness. Thus, the asymmetry in the frontal activation of the brian is associated with valence in the alpha band as well as the beta band. Changes in the gamma band is also associated with the emotions happiness and sadness

It has been suggested in previous studies that the male and the female brains have different ways of processing emotional activity. The studies suggest that the male brain relies on past emotional experience in evaluating new emotional stimuli, whereas the female brain seemed to engage the emotional stimuli more readily.

In conclusion, it is clear that the frontal and parietal lobes in the brain are the most useful when it comes to emotional recognition and that the alpha, beta and gamma frequency bands contain the most information in regards to the matter.

Electroencephalography (EEG) [25]

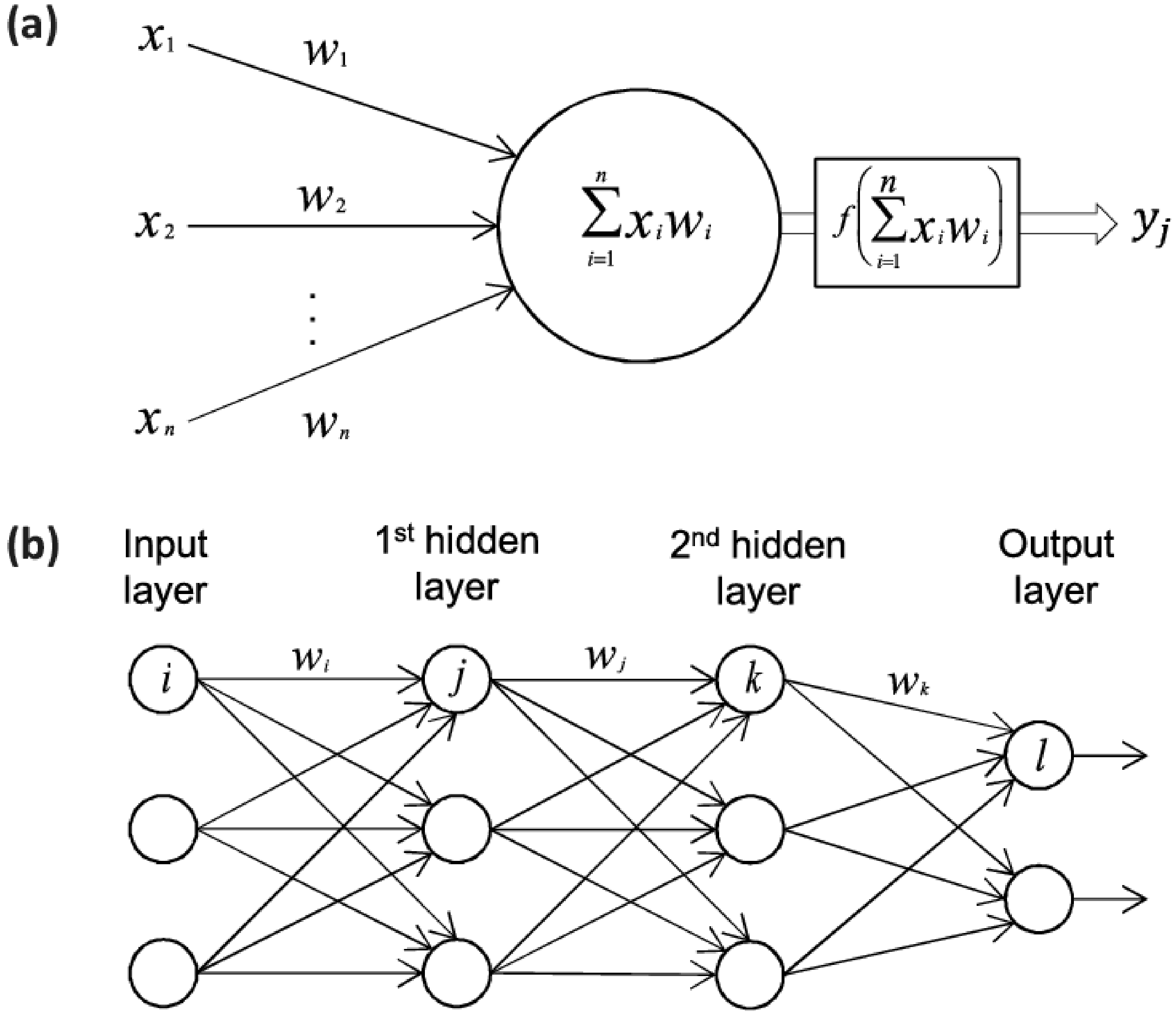
The largest portion of the human brain, the cortex, is divided into the frontal, temporal, parietal, and occipital lobes. The frontal lobe is responsible for the conscious thought. The temporal lobe is responsible for the senses of smell and sound, and the processing of complex stimuli such as faces and scenes. The parietal lobe is responsible for integrating sensory information from various senses, as well as the manipulation of objects. Finally, the occipital lobe is responsible for the sense of sight.

The cortex subdivided into the frontal, temporal, parietal, and occipital lobes. EEG is a medical imaging technique that reads scalp electrical activity generated by brain structures, i.e., it measures voltage fluctuations resulting from ionic current flows within the neurons of the brain. A typical adult EEG signal, when measured from the scalp, is about 10-100 V. These signals observed in the scalp are divided into specific ranges that are more prominent in certain states of mind, namely the delta (1-4 Hz), theta (4-7 Hz), alpha (8-13 Hz), beta (13-30Hz), and gamma (>30 Hz) bands. The beginning and the end of the bands varies a few hertz among different authors.

Delta waves are associated with the unconscious mind, and occur during a deep dreamless sleep. Theta brain waves are associated with the subconscious mind, for instance with activities such as sleeping and dreaming. Alpha waves are typically associated to a relaxed mental state, yet aware, and are more visible over the parietal and occipital lobes. High alpha activity has been correlated to brain inactivation. Beta waves are related to an active state of mind, more prominent in the frontal cortex and over other areas during intense focused mental activity. Finally, gamma waves are associated with an hyper brain activity.

Convolutional Neural Networks (CNN)

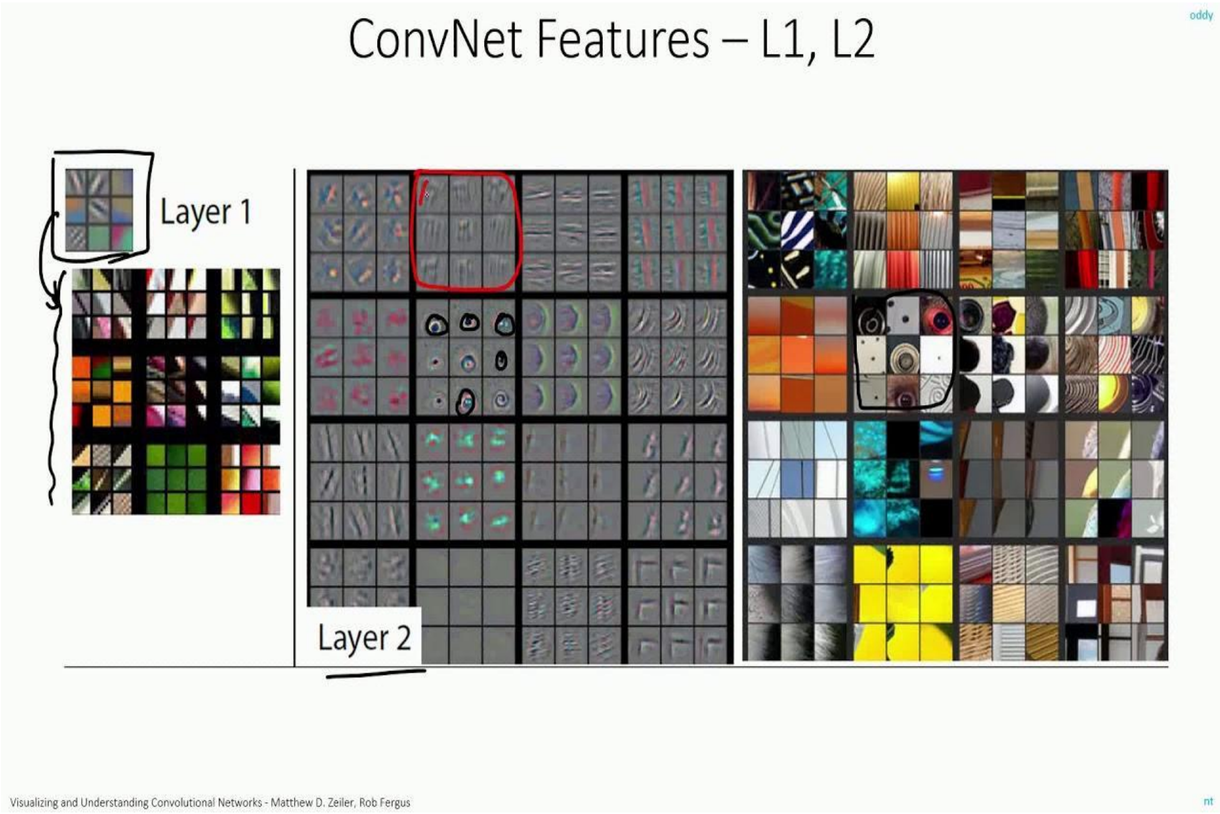
Convolutional Neural Networks is an Artificial Intelligence algorithm that is mostly used with signal data like images and sound tracks and is based on the Artificial Neural Network algorithm and follows the same general concepts of multiplying the weights by the inputs, applying activation functions and repeating this process until the output layer is reached and the output is calculated, calculating the errors (gradients) and propagating them back through the network, then updating the weights of the network to minimize the loss function and improve the weights of the network to better suit the function of the current problem. The general deep neural network architecture is shown in the following figure.



$$y_j = f\left(\sum x_i w_i\right) \quad y_k = f\left(\sum x_j w_j\right) \quad y_l = f\left(\sum x_k w_k\right)$$

The difference between artificial neural networks and convolutional neural networks is that the latter introduces a new type of layer called the convolution layer which, instead of having a weight attached to each neuron in the previous layer, it has filters which are essentially shared weights that are used by convoluting the signal with them in order to produce feature maps which make up the next layer to be convoluted again with a different set of filters. The convolutional neural networks train by adjusting the values of the filters in order to better extract features from the feature maps.

The main advantage of using convolutional neural networks over traditional artificial neural networks is that convolutional networks are much better at extracting complex features from the data and using it in the classification process, and that is caused by the existence of the filters in the convolutional layers which, by combining data points in the signal, are able to extract complex features from them and the deeper the convolutional network goes the more complex the features described by the layers are. It is also worth noting that it is usually important to break the linearity of the feature between the layers and to do that a non-linear activation function is used like the ‘sigmoid’ function or ,more commonly, the ‘relu’ function The following figure illustrates how the filters contribute to the feature extraction process.



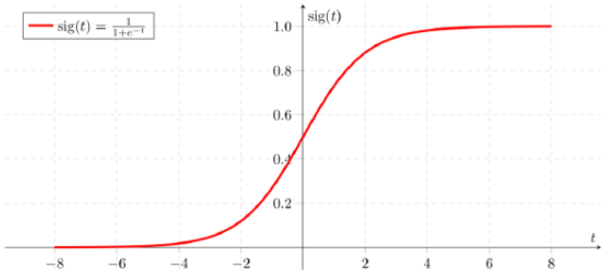
Logistic Regression

Logistic regression is named for the function used in its method logistic function also called sigmoid function

Logistic Regression Model

Want $0 \leq h_\theta(x) \leq 1$

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$



Logistic regression is a statistical method similar to linear regression since they find an equation that predicts an outcome for a binary variable, *Y,* from one or more response variables, *X*
key difference from linear regression is the output value is binary value not numeric and it uses log function in loss function instead of mean square error.
logistic regression predict probability of being from the default class
we are modeling the probability that an input (X) belongs to the default class (Y=1), we can write this formally as:

$$P(X) = P(Y=1|X)$$

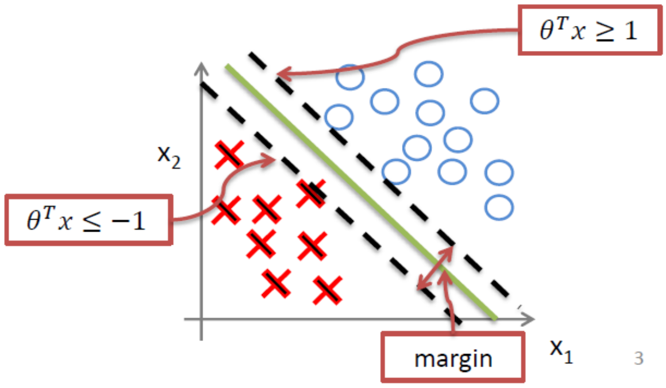
Logistic regression is a linear method, but the predictions are transformed using the logistic function. The impact of this is that we can no longer understand the predictions as a linear combination of the inputs as we can with linear regression

Support vector machine (SVM)

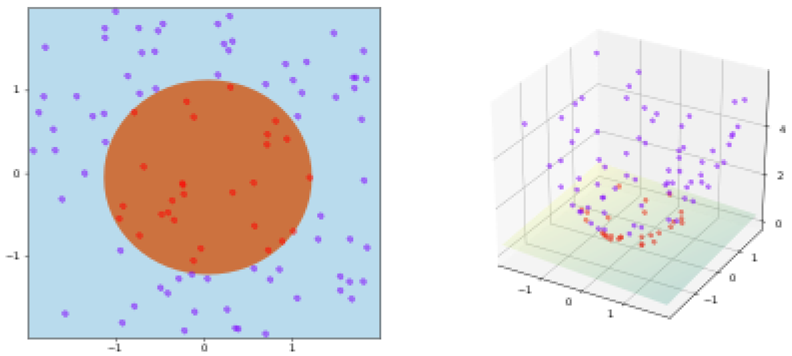
Support Vector Machine (SVM) is a supervised machine learning algorithm which can be used for classification or regression problems. classification is performed by finding the hyper-plane that differentiates the two classes maximizing the distances between the nearest data points of the two classes (this can be generalized to more than two classes). This enables the SVMs to learn reasonably good decision boundaries using small datasets and still be able to generalize very well on the unseen data because of the margin around the decision boundary The following figure shows how the SVM works by using margins around the decision boundary.

$$\min_{\theta} C \underbrace{\sum_{i=1}^{m} \left[y^{(i)} cost_1(\theta^T x^{(i)}) + (1 - y^{(i)}) cost_0(\theta^T x^{(i)}) \right]}_{\text{Classification error}} + \underbrace{\frac{1}{2} \sum_{i=1}^{n} \theta_j^2}_{\text{Margin error}}$$

$$\text{s.t.} \quad \begin{aligned} \theta^T x^{(i)} &\geq 1 \quad && \text{if } y^{(i)} = 1 \\ \theta^T x^{(i)} &\leq -1 \quad && \text{if } y^{(i)} = 0 \end{aligned}$$



SVMs can also be very effective in dealing with non-linear data by mapping this data to a higher dimensional space to be able to find a suitable hyperplane to separate the data with by using the kernel trick. The kernel trick makes use of kernel functions which enable them to operate in a higher dimensional space without computing the coordinates of a point in that space, instead it computes the similarity between different data points, that makes them computationally cheaper than calculating the coordinates of the data points in the feature space. The following figure shows the basic idea of the kernel trick in action.



K-Nearest Neighbors (KNN)

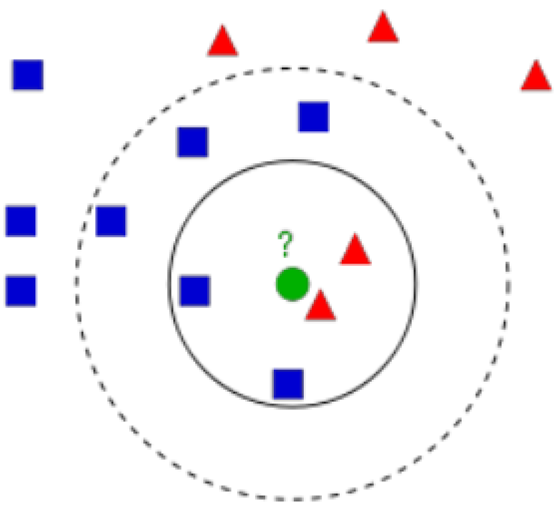
classification algorithms used in supervised learning and is called Lazy learning algorithm because it does not have a specialized training phase and uses all the data for training while classification

The idea is to search for the closest match(es) of the test data in the feature space by measure similarity to the n nearest neighbor

In the classification phase, k is a user-defined constant, and an unlabeled vector (test point) is classified by assigning the label which is most frequent among the k training samples nearest to that query point.

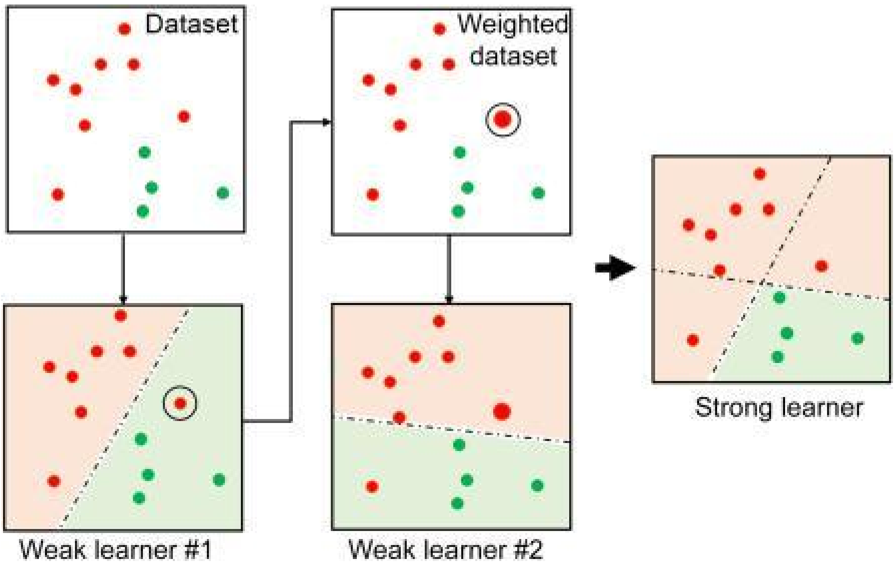
A commonly used distance metric is Euclidean or Hamming distance

The best choice of k depends upon the data generally, larger values of k reduces effect of the noise on the classification, but make boundaries between classes less distinct



AdaBoost

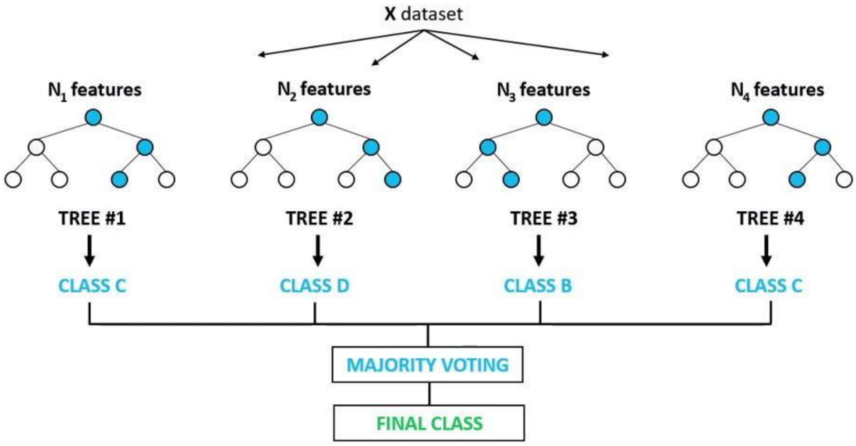
The Adaboost classifier belongs to a family of classifiers called Ensemble classifiers which work by training multiple classifiers and using them in the prediction process. Adaboost uses multiple copies of other ,possibly weak, classifiers in order to increase the accuracy of the output. The way Adaboost works is that it trains an initial classifier on the dataset and then trains more copies of the same classifier on the same dataset but with adjusted weights for the incorrectly classified samples so that the following classifiers focus more on the more difficult samples which, in theory, should be the wrongly classified samples in the previous classifiers. The following figure shows a basic example of the Adaboost classifiers in work.



Random Forest

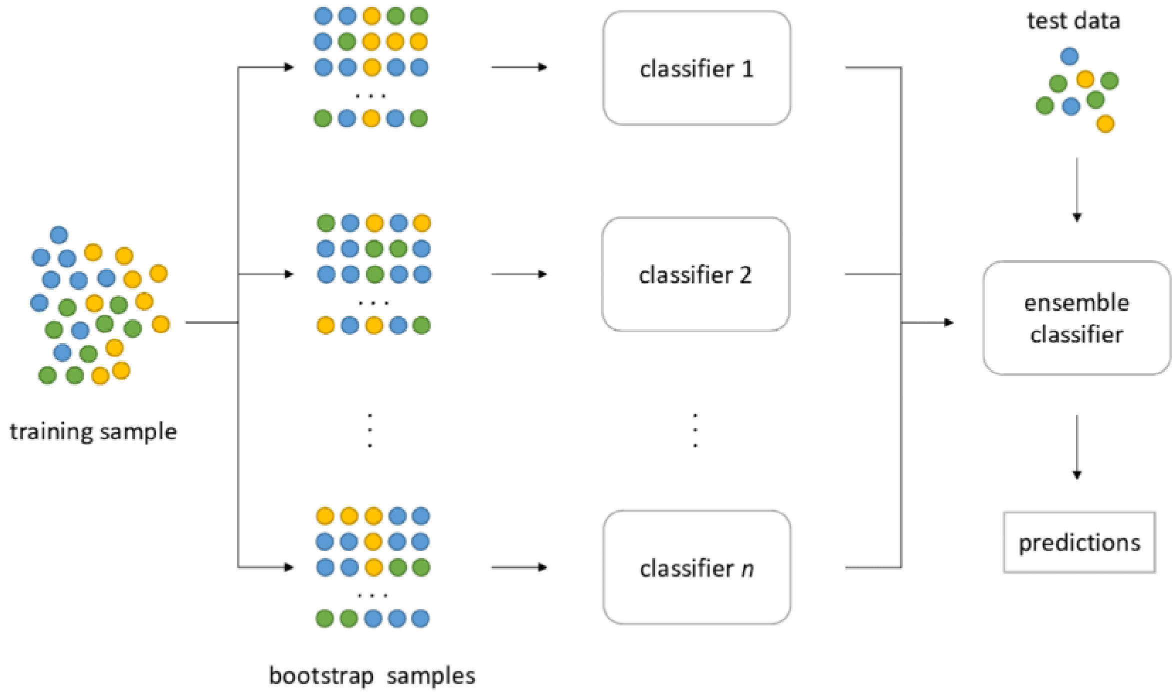
The Random Forest Classifier is another type of ensemble classifier that trains a number of decision trees on different subsets of the main dataset and averages the results in order to improve the accuracy of the predictions and the classifier's generalization ability. The following figure illustrates how the Random Forest Classifier works with 4 decision trees.

Random Forest Classifier



Bagging Classifier

The Bagging Classifier is another ensemble classifier that trains multiple classifiers on random subsets of the original dataset and then uses all of their individual predictions by averaging their results or by using their votes in order to make the final prediction of the classifier. The Bagging Classifier has the same idea as the Random Forest Classifier but instead of always using decision trees as the base estimator it can use any kind of classifier and combine the robustness of that classifier with the benefits of having multiple copies of it in an ensemble to improve its predictive results. The following figure illustrates the workflow of the Bagging Classifier approach.


training sample
bootstrap samples
classifier 1
classifier 2
classifier n
test data
ensemble classifier
predictions

Analysis and Design

Chapter Three

3.1 System Overview

3.1.1 System Architecture

Include a figure of the system architecture and a description of all modules.
You may add Functional and non-functional requirements section –If needed–

3.2 Dataset Description

DEAP Dataset [1]

This data is captured from 32 participant 16 males and 16 females
this data consists of 32 .dat file each file of those consists data and labels
the data is consists of 40 session each consists 0f 48 channels
as follows:

Channel no.	Ch. name Twente	Ch. name Geneva
1	Fp1	Fp1
2	AF3	AF3
3	F7	F3
4	F3	F7
5	FC1	FC5
6	FC5	FC1
7	T7	C3
8	C3	T7
9	CP1	CP5

10	CP5	CP1
11	P7	P3
12	P3	P7
13	Pz	PO3
14	PO3	O1
15	O1	Oz
16	Oz	Pz
17	O2	Fp2
18	PO4	AF4
19	P4	Fz
20	P8	F4
21	CP6	F8
22	CP2	FC6
23	C4	FC2
24	T8	Cz
25	FC6	C4
26	FC2	T8
27	F4	CP6
28	F8	CP2
29	AF4	P4
30	Fp2	P8

31	Fz	PO4
32	Cz	O2
33	EXG 1	$hEOG_1$ (to the left of left eye)
34	EXG 2	$hEOG_2$ (to the right of right eye)
35	EXG 3	$vEOG_1$ (above right eye)
36	EXG 4	$vEOG_4$ (below right eye)
37	EXG 5	$zEMG_1$ (Zygomaticus Major, +/- 1cm from left corner of mouth)
38	EXG 6	$zEMG_2$ (Zygomaticus Major, +/- 1cm from $zEMG_1$)
39	EXG 7	$tEMG_1$ (Trapezius, left shoulder blade)
40	EXG 8	$tEMG_2$ (Trapezius, +/- 1cm below $tEMG_1$)

41	GSR1	Galvanic skin response, left middle and ring finger
42	GSR2	Unused
43	Erg1	Unused
44	Erg2	Unused
45	Resp	Respiration belt
46	Plet	Plethysmograph, left thumb
47	Temp	Temperature, left pinky
48	Status	Status channel containing markers

The first 32 channels are the electrodes of the EEG signals the we are dealing with in the rest of the paper each channel has 8064 record taken from 1 min stimuli and 3 seconds self assessment taken with 512 hz than sampled into 128 and the other are biological identifiers used to be sure of the values of the labels the the label part (velancy ,Arousal ,Dominance ,liking) each scaled from 0 to 9

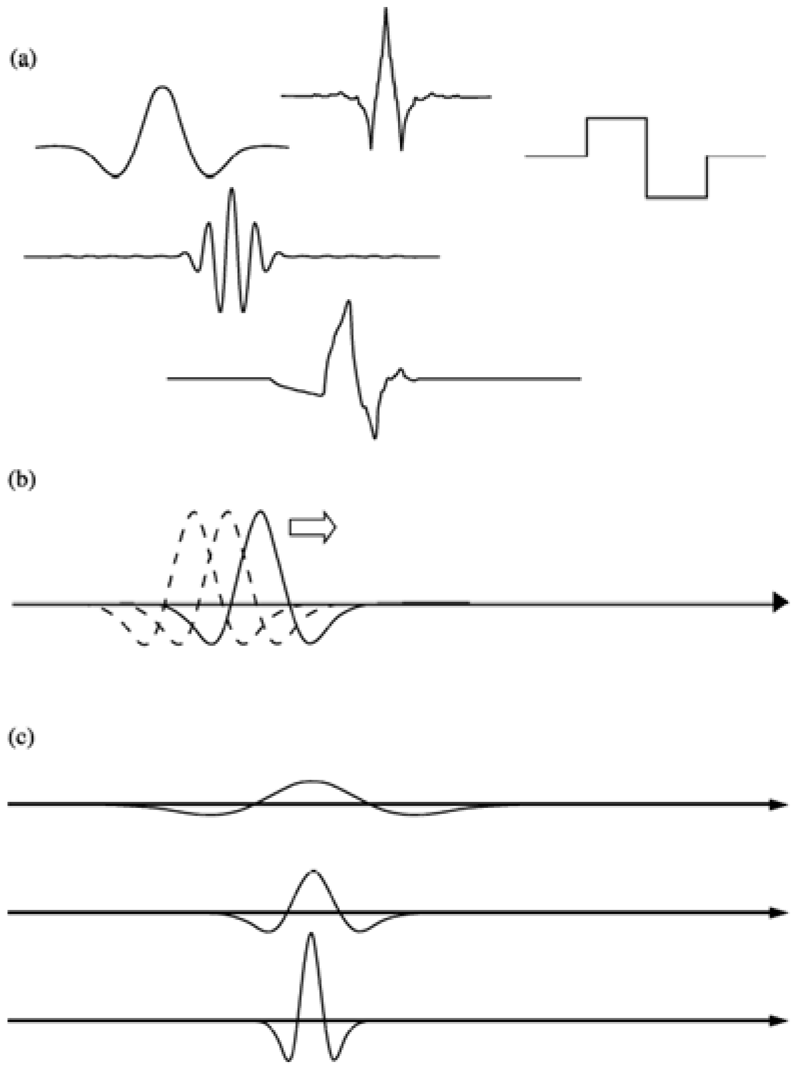
Implementation and Testing

Chapter Four

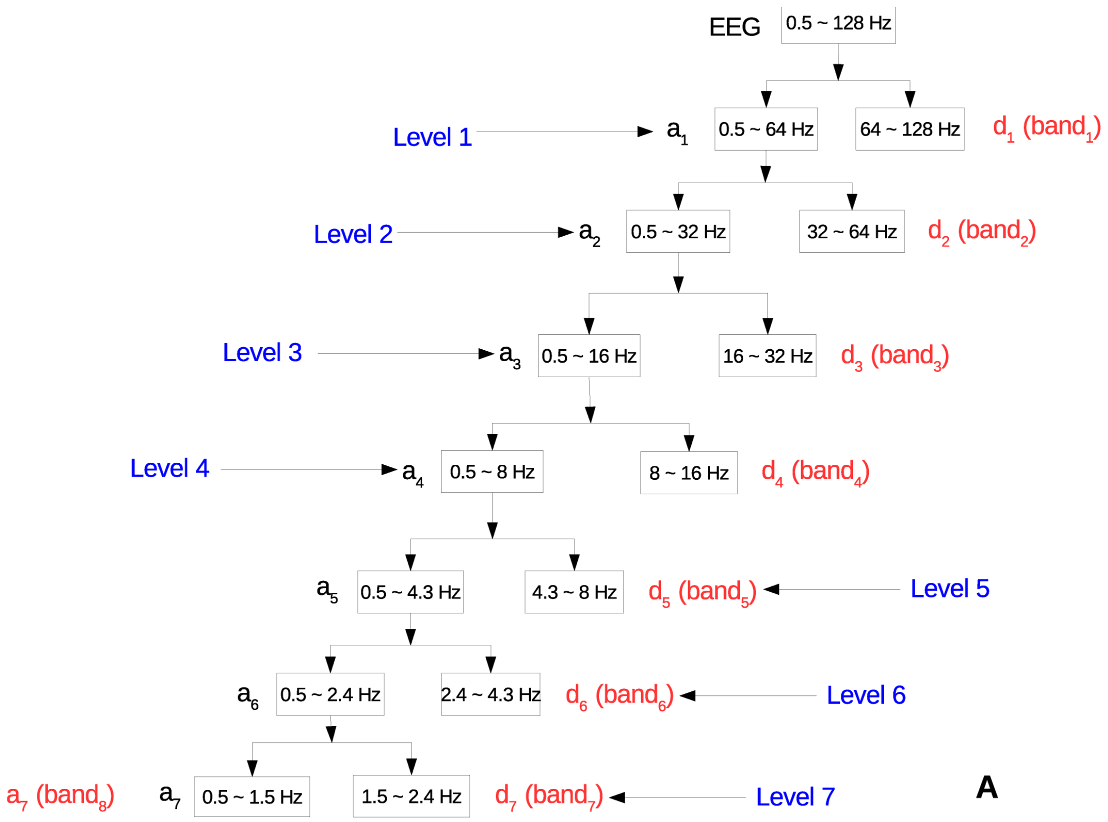
4.1 Features

Discrete Wavelet Transform [13,14,16,17]

The wavelet transform is a mapping technique that is used widely because it is very powerful in time-frequency localization wavelet transform is simply the projection of a signal into smaller functions called wavelets



this wavelets is divided is high and low levels as show we took the advantage of this to divide it to theta (4-8 Hz), alpha (8-16 Hz), beta (16-32 Hz), gamma(32-64 Hz) and noises (> 64 Hz) as this bands are the only bands extracted from brain



and used dp4 wavelet due to it resemblance with the EEG Signal

Entropy and Energy[4]

EEG signals are very complex signal , non linear ,non stationary so we used Decimated Wavelet transform DWT entropy and we extracted the entropy by from windowed EGG signal over 4 seconds by overlapping in 2 seconds to help in quick detection of the emotion state these EEG signal is transformed by db4 only because it is the mother wavelet function from; theta (4-8 Hz), alpha (8-16 Hz), beta (16-32 Hz), gamma (32-64 Hz) and noises (> 64 Hz) then we calculated the entropy because it gives indication of the amount of information that is carried by EEG signal in our case the amount of information carried by each band of the EEG signal calculated as following:

$$ENT_j = -\sum_{k=1}^{N} (D_j(k)^2)\log(D_j(k)^2) \quad (1)$$

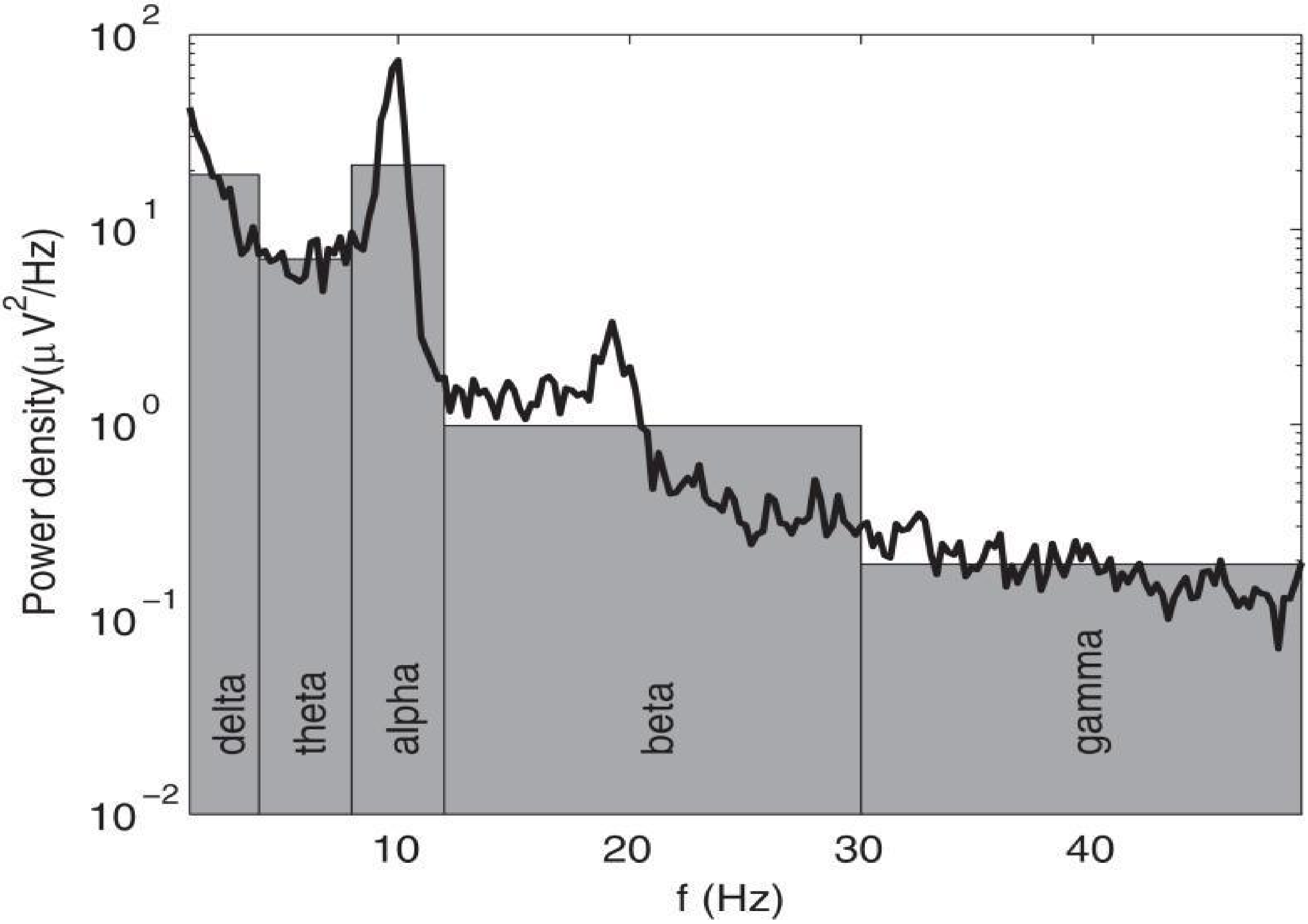
by summing the coefficients of dwt transform we can compute the amount of energy carried by each band this energy is calculated as following:

$$ENG_j = \sum_{k=1}^{N} (D_j(k)^2) \quad k=1,2,\ldots,N. \quad (2)$$

j is the frequency band , and k is the number of wavelet coefficients within the j frequency band.

Power Spectrum[5,6]

Spectral analysis is a popular method to describe the EEG signals as Power spectrum emphasizes information about the properties of the eeg signal like the EEG. In the stationary case this information and it plays an important role to determine the difference in activity between different frequency bands,
e.g., between alpha and beta activities. we calculated it
for each spectral band power theta (4-8 Hz), alpha (8-16 Hz), beta (16-32 Hz)
and gamma (32-64 Hz)the power spectrum is calculates the number that each of frequency bands appears in the fourier transform



4.2 Data Augmentation:

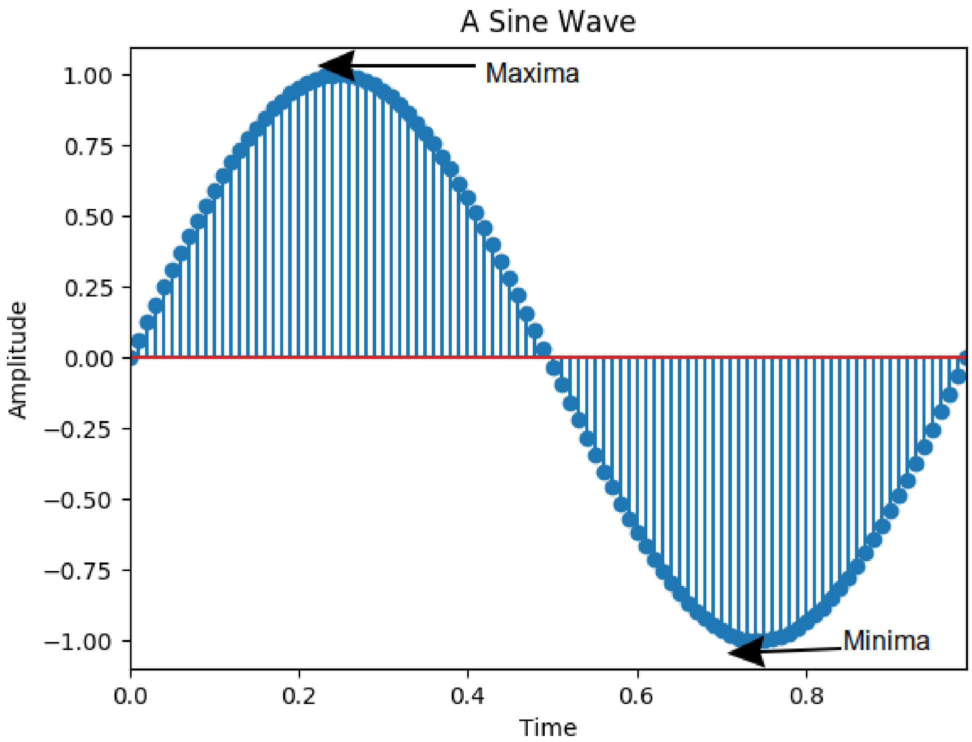
- Empirical Mode Decomposition (EMD)[7][8][9]:[2]

  EMD is an algorithm to convert our signal (Time-Series Data). **We inputting a signal to the EMD and we will get some decomposed signal** a.k.a 'basic ingredient' of our signal input. It's similar to the Fast Fourier Transform (FFT). FFT assumes our signal is periodic and it's 'basic ingredient' is various simple sine waves. In FFT, our signal is changed from the time spectrum to the frequency spectrum.

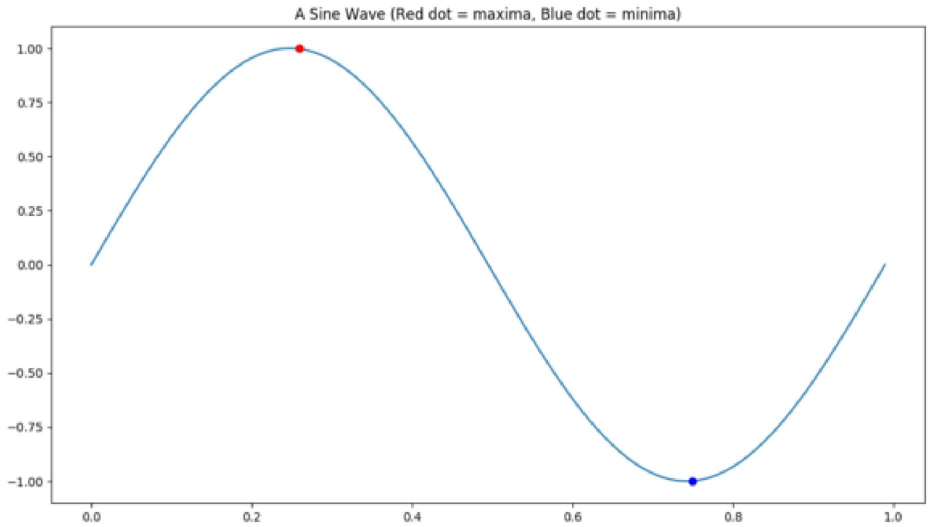
[2] Details of the EMD was take from the the referenced papers and from an article by Muhammed Ryan on towardsdatascience.com link is found on reference number [23]

Difference between EMD and FFT, is that it doesn't consider the signal as a periodic signal or a sinusoidal wave instead it's a **Intrinsic Mode Function (IMF)**.
Also, the EMD is based on the dataset it has no assumption about our data that is why it is called Empirical.

The IMFs has to have 2 properties:
1- The number of local maximas and local minimas to be equal or at most with difference one
(i.e. |local maximas - local minimas| =<1)
Noting that the maximas are not always above zero and minimas are not always below zero it is just the value where the value tends to change either from downwards to upwards or from upwards to downwards.

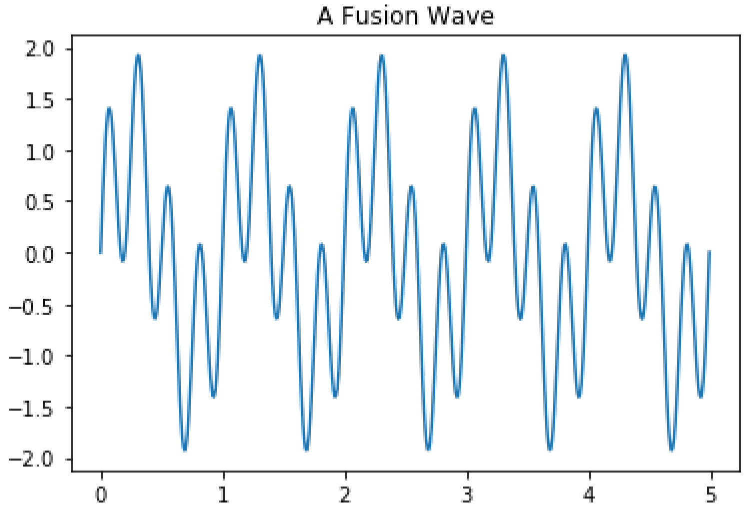


2- The mean of the signal is equal to zero.

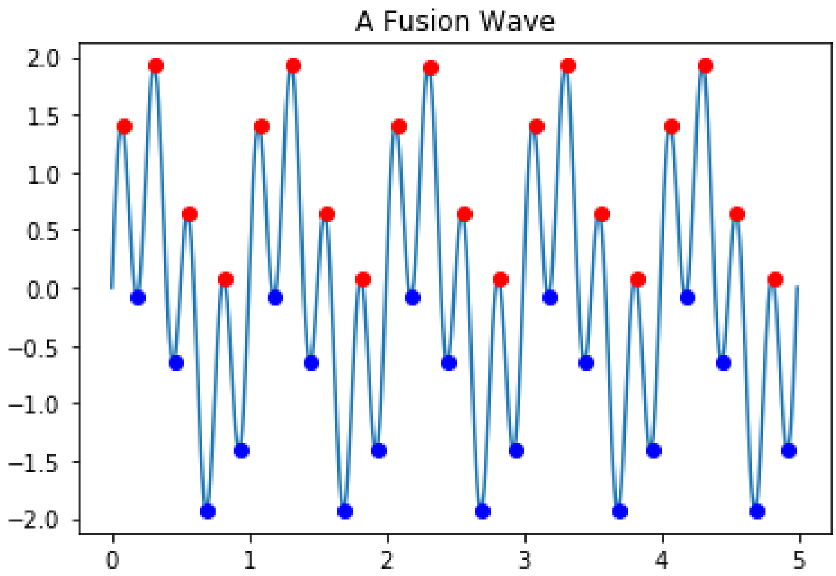


The algorithm of the EMD is taken on 10 steps:

1. Set s (t) = ri−1 (t). Initially, i = 1 and r0 (t) = x(t).

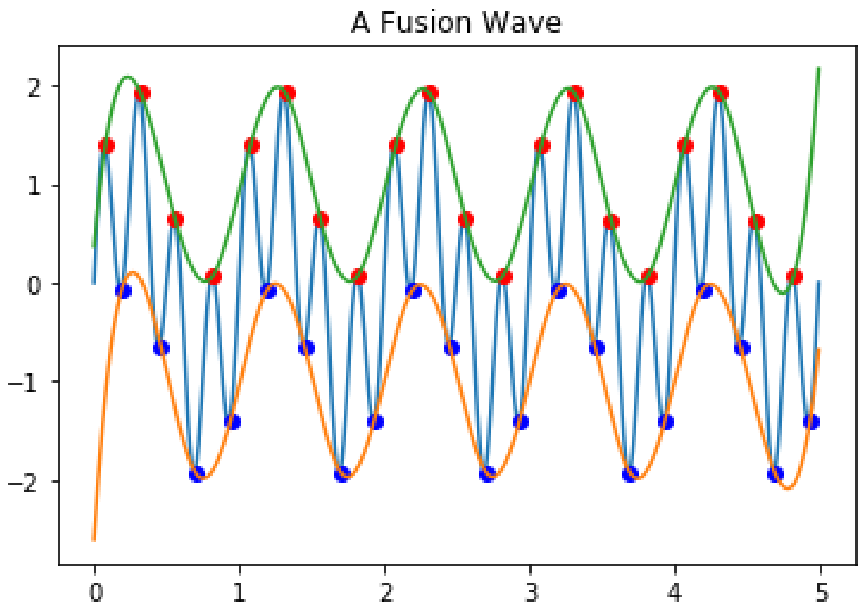


2. Detect the local maxima and the local minima of s(t).

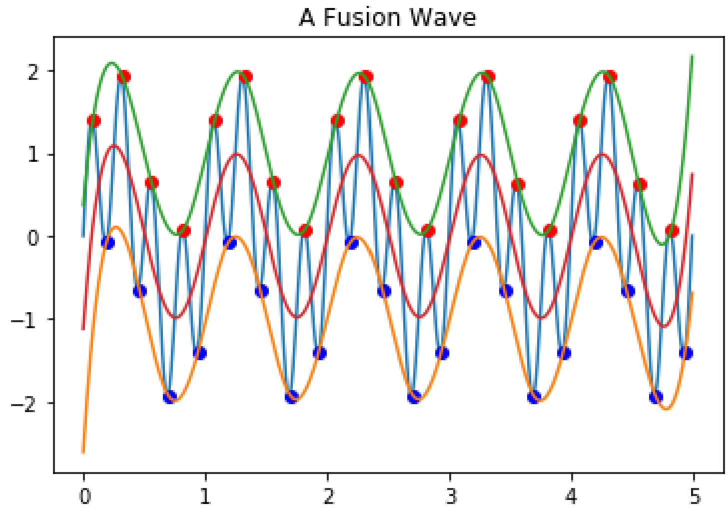


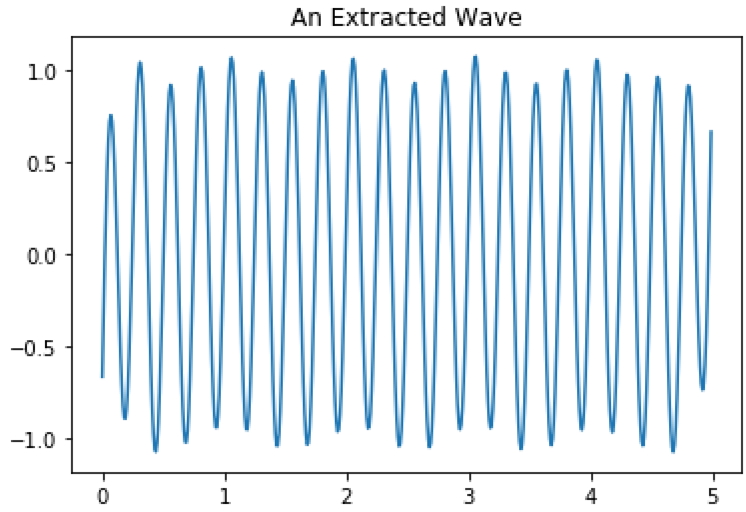
3. Interpolate all local maxima to generate the upper envelope.

4. Interpolate all local minima to generate the lower envelope.



5. Obtain the local mean m(t) by averaging the upper and lower envelopes.





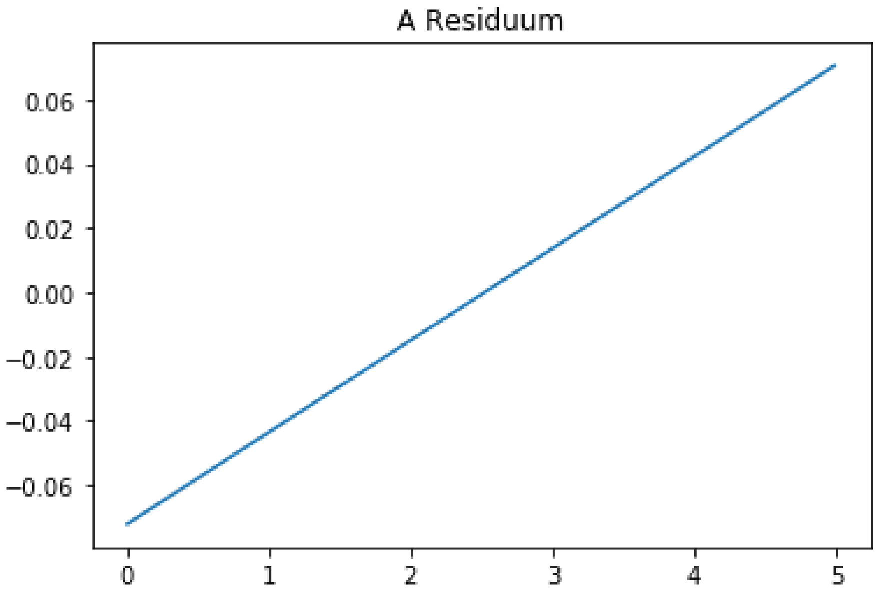
6. Get a candidate for IMF by subtracting the local mean m(t) from the signal: h (t) = s (t) − m(t).

7. If h(t) does not satisfy the IMF's conditions, begin a new loop from step 2, setting s (t) = h (t).

8. Otherwise, h(t) is defined as an IMF: IMFi (t) = h (t).

9. ri (t) = ri−1 (t) − IMFi(t).

10. If ri (t) is a monotonic function or does not have enough extrema to calculate the upper and lower envelopes, then IMFi(t) is the last IMF function of x(t) and the decomposition ends.



11. Otherwise, set s (t) = ri (t) and start a new loop from step 2 in order to obtain IMFi+1 (t).

4.3 Implementation:

Machine Learning Approaches:

1. **Logistic Regression (LR)**

   Using the Logistic Regression algorithm on three types of features extracted from the the preprocessed data to classify the data to the output classes (HV vs. LV) and (HA vs. LA) using :
   1. the Discrete Wavelet Transform algorithm using the db4 wavelet
   2. the Discrete Wavelet Transform algorithm using the db10 wavelet
   3. the combination of theDiscrete Wavelet Transform algorithm using the db4 wavelet and Power Spectral Density
   4. The energy and entropy of power spectrum dp4 wavelet Transform

   This model deals with all electrodes only in the brain.

Algo\Feature	Wavelet Transform(db4)	Wavelet Transform(db10)	Wavelet Transform(db4) & PSD	The energy and entropy of power spectrum
Arousal	71.8%	73%	70.3%	59.3%
Valence	67.2%	67.6%	65.6%	53.9%

This model deals with all electrodes only in the brain for each person individually and outputs the average accuracy.

Algo\Feature	Wavelet Transform(db4)	Wavelet Transform(db10)	Wavelet Transform(db4) & PSD	The energy and entropy of power spectrum
Logistic Regression	65.2%	62.9%	59.8%	60.9%
Valence	62.1%	62.1%	60.1%	62.1%

2. **Support Vector Machine (SVM)**

Using the Support Vector Machine algorithm on three types of features extracted from the preprocessed data to classify the data to the output classes (HV vs. LV) and (HA vs. LA) using :

1. the Discrete Wavelet Transform algorithm using the db4 wavelet
2. the Discrete Wavelet Transform algorithm using the db10 wavelet
3. the combination of the Discrete Wavelet Transform algorithm using the db4 wavelet and Power Spectral Density
4. The energy and entropy of power spectrum dp4 wavelet Transform

This model deals with all electrodes only in the brain.

Algo\Feature	Wavelet Transform(db4)	Wavelet Transform(db10)	Wavelet Transform(db4) & PSD	The energy and entropy of power spectrum
Arousal	71.5%	71.1%	68.4%	63.2%
Valence	67.5%	66.4%	68.7%	63.3%

This model deals with all electrodes only in the brain for each person individually and outputs the average accuracy.

Algo\Feature	Wavelet Transform(db4)	Wavelet Transform(db10)	Wavelet Transform(db4) & PSD	The energy and entropy of power spectrum
Arousal	68.8%	66%	67.5%	63.2%
Valence	60.5%	59.4%	59.8%	65.6%

3. **K-Nearest Neighbor (KNN)**

Using the K-Nearest Neighbor algorithm on three types of features extracted from the preprocessed data to classify the data to the output classes (HV vs. LV) and (HA vs. LA) using :

1. the Discrete Wavelet Transform algorithm using the db4 wavelet
2. the Discrete Wavelet Transform algorithm using the db10 wavelet
3. the combination of theDiscrete Wavelet Transform algorithm using the db4 wavelet and Power Spectral Density

4. The energy and entropy of power spectrum dp4 wavelet Transform

This model deals with all electrodes only in the brain.

Algo\Feature	Wavelet Transform(db4)	Wavelet Transform(db10)	Wavelet Transform(db4) & PSD	The energy and entropy of power spectrum
Arousal	69.1%	67.6%	71.9%	57.4%
Valence	61.7%	62.5%	60.9%	54.7%

This model deals with all electrodes only in the brain for each person individually and outputs the average accuracy.

Algo\Feature	Wavelet Transform(db4)	Wavelet Transform(db10)	Wavelet Transform(db4) & PSD	The energy and entropy of power spectrum
K-Nearest Neighbor	65.2%	61.3%	62.1%	61.3%
Valence	58.2%	58.2%	60.9%	60.1%

4. Adaptive Boosting (AdaBoost)

Using the Adaptive Boosting algorithm using Decision Trees on three types of features extracted from the preprocessed data to classify the data to the output classes (HV vs. LV) and (HA vs. LA) using :

1. the Discrete Wavelet Transform algorithm using the db4 wavelet
2. the Discrete Wavelet Transform algorithm using the db10 wavelet
3. the combination of theDiscrete Wavelet Transform algorithm using the db4 wavelet and Power Spectral Density
4. The energy and entropy of power spectrum db4 wavelet Transform

This model deals with all electrodes only in the brain.

Algo\Feature	Wavelet Transform(db4)	Wavelet Transform(db10)	Wavelet Transform(db4) & PSD	The energy and entropy of power spectrum
Arousal	55.3%	63.2%	60.9%	53.5%
Valence	58.9%	61.7%	61.7%	53.3%

This model deals with all electrodes only in the brain for each person individually and outputs the average accuracy.

Algo\Feature	Wavelet Transform(db4)	Wavelet Transform(db10)	Wavelet Transform(db4) & PSD	The energy and entropy of power spectrum
Arousal	58.6%	60.2%	60.9%	58.2%
Valence	56.6%	63.7%	56.6%	58.2%

5. Random Forest

Using the Random Forest algorithm using Decision Trees on three types of features extracted from the preprocessed data to classify the data to the output classes (HV vs. LV) and (HA vs. LA) using :

1. the Discrete Wavelet Transform algorithm using the db4 wavelet
2. the Discrete Wavelet Transform algorithm using the db10 wavelet
3. the combination of theDiscrete Wavelet Transform algorithm using the db4 wavelet and Power Spectral Density
4. The energy and entropy of power spectrum dp4 wavelet Transform

This model deals with all electrodes only in the brain.

Algo\Feature	Wavelet Transform(db4)	Wavelet Transform(db10)	Wavelet Transform(db4) & PSD	The energy and entropy of power spectrum
Arousal	70.3%	72.7%	70.3%	58.6%
Valence	67.2%	69.1%	68.4%	60.2%

This model deals with all electrodes only in the brain for each person individually and outputs the average accuracy.

Algo\Feature	Wavelet Transform(db4)	Wavelet Transform(db10)	Wavelet Transform(db4) & PSD	The energy and entropy of power spectrum
Arousal	67.8%	59.7%	67.5%	62.5%
Valence	66.8%	69.1%	67.6%	61.3%

6. **Bagging Classifier**

Using the Bagging Classifier algorithm using Support Vector Machines as the base classifier on three types of features extracted from the preprocessed data to classify the data to the output classes (HV vs. LV) and (HA vs. LA) using :

1. the Discrete Wavelet Transform algorithm using the db4 wavelet
2. the Discrete Wavelet Transform algorithm using the db10 wavelet
3. the combination of theDiscrete Wavelet Transform algorithm using the db4 wavelet and Power Spectral Density
4. The energy and entropy of power spectrum dp4 wavelet Transform

This model deals with all electrodes only in the brain.

Algo\Feature	Wavelet Transform(db4)	Wavelet Transform(db10)	Wavelet Transform(db4) & PSD	The energy and entropy of power spectrum
Arousal	70.3%	69.9%	69.9%	64.8%
Valence	62.5%	64%	63.7%	64.8%

This model deals with all electrodes only in the brain for each person individually and outputs the average accuracy.

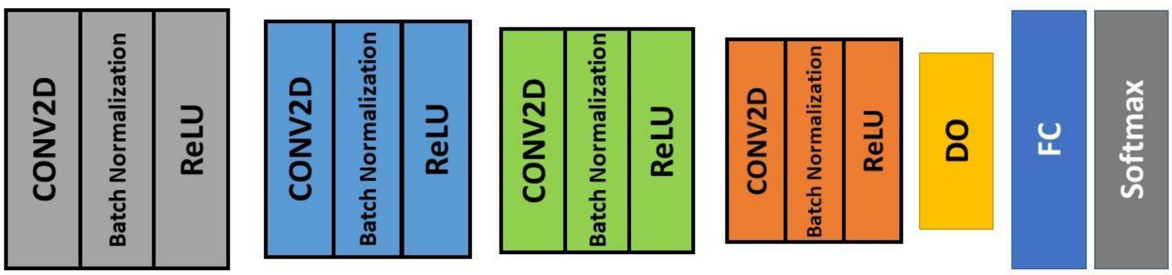
Algo\Feature	Wavelet Transform(db4)	Wavelet Transform(db10)	Wavelet Transform(db4) & PSD	The energy and entropy of power spectrum
Arousal	67.2%	64.8%	64.8%	64.8%
Valence	55.5%	55%	55.5%	57%

Deep Learning Approaches:

1. Approach one:

   Using the CNNs to try to map input (which consists of Raw and Augmented data using EMD) to the output classes (HV vs. LV) and (HA vs. LA).
   This model deals with the frontal electrodes only in the brain as they are more accurate in the results. Each electrode is augmented using the EMD and only the top 4 IMFs are selected.

   The proposed CNN architecture:[3]
   The CNN Architecture consists of 4 blocks of convolutions where each block consists of a CONV2D layer followed by a Batch Normalization followed by a non-linear activation layer (ReLU).
   The number of kernels for the first block equals to 64, has a receptive field of (3x3), and with no strides. Second block has 32 kernels, receptive field of (3x3) and no strides. Third block has 16 kernels, receptive field of (3x3) and no strides. Fourth and last block contains 8 kernels, receptive field of (5x5) and no strides.
   The output of the 4 convolution blocks is then flattened and then passed by a DropOut layer of 0.5 and then finally into a fully connected layer that has 2 nodes and a Softmax activation function.
   The input size for the model is (12x8064x4). The first axis is the number of electrodes (frontals only), Second axis is the electrode reading in 63 seconds with sampling rate 128 and the last axis is the augmented data for each electrode.



[3] This approach was inspired from the model by paper number [10] in references.

Layer	Parameters
CONV2D - 1	(3x3), 64, 0 Strides
	Batch Normalization
	ReLU
CONV2D - 2	(3x3), 32, 0 Strides
	Batch Normalization
	ReLU
CONV2D - 3	(3x3), 16, 0 Strides
	Batch Normalization
	ReLU
CONV2D - 4	(5x5), 8, 0 Strides
	Batch Normalization
	ReLU
Dropout	0.5
FC	2
Classifier	Softmax

We convolve with the 12 channels in order to get the common features between the 12 electrodes. And also convolving with 4 augmented layers in order to increase the input data as the number of data is so low. Each convolution block is used to extract some features, then it is normalized and activated using a ReLU function to be entered to the next block to extract some new features.
After 4 blocks of features extraction these features are then classified using a Softmax.

The testing and results:

We have applied the previous approach with two different hypotheses. The first one considers that all the EEG readings for the people are alike and therefore we trained a one model for valence and one model for arousal for the whole input.

For the first hypothesis, both of our valence and arousal models trained on 50 epochs and with batch size = 10. Also, we used RMSProp as our optimizer. And Categorical Cross Entropy for the loss calculation.

We trained the model two times, one with Early stopping on validation loss with 7 steps patience and one without.

Test results:

Model	Valence Accuracy	Arousal Accuracy
With Callback	53.5%	56.6%
Without Callback	60.5%	55%

The second idea, considers that all the EEG signals of the people are not alike and for that we will need to train 2 models (one for valence and one for arousal) for each person (i.e. 32x2 models). The same settings as the previous model are used in each model of the 64 models. 50 epochs, batch size = 10, RMSProp optimizer and Categorical Cross Entropy for loss calculation.

Each model was trained two times, one with Early stopping on validation loss with 7 steps patience and one without.

	No Callbacks		Callback	
	Valence Accuracy	Arousal Accuracy	Valence Accuracy	Arousal Accuracy
Person 1	50%	50%	37.5%	75%
Person 2	62.5%	62.5%	62.5%	50%
Person 3	37.5%	62.5%	62.5%	50%
Person 4	37.5%	37.5%	37.5%	37.5%

Person 5	50%	50%	62.5%	62.5%
Person 6	62.5%	75%	50%	25%
Person 7	75%	75%	25%	75%
Person 8	50%	37.5%	75%	50%
Person 9	50%	75%	37.5%	25%
Person 10	37.5%	50%	50%	37.5%
Person 11	62.5%	37.5%	75%	37.5%
Person 12	37.5%	25%	37.5%	50%
Person 13	37.5%	50%	37.5%	50%
Person 14	50%	50%	62.5%	62.5%
Person 15	62.5%	50%	50%	50%
Person 16	62.5%	62.5%	37.5%	50%
Person 17	37.5%	62.5%	50%	37.5%
Person 18	62.5%	50%	50%	62.5%
Person 19	75%	75%	75%	62.5%
Person 20	50%	62.5%	75%	37.5%
Person 21	50%	50%	50%	50%
Person 22	75%	50%	50%	62.5%
Person 23	50%	50%	75%	50%
Person 24	100%	37.5%	100%	50%
Person 25	87.5%	50%	62.5%	50%
Person 26	87.5%	75%	75%	62.5%
Person 27	62.5%	75%	75%	25%
Person 28	50%	62.5%	75%	62.5%
Person 29	62.5%	50%	87.5%	75%

Person 30	50%	75%	87.5%	50%
Person 31	75%	37.5%	75%	50%
Person 32	37.5%	87.5%	62.5%	50%
Averages	59%	56%	63%	50%

2. **Approach Two:**

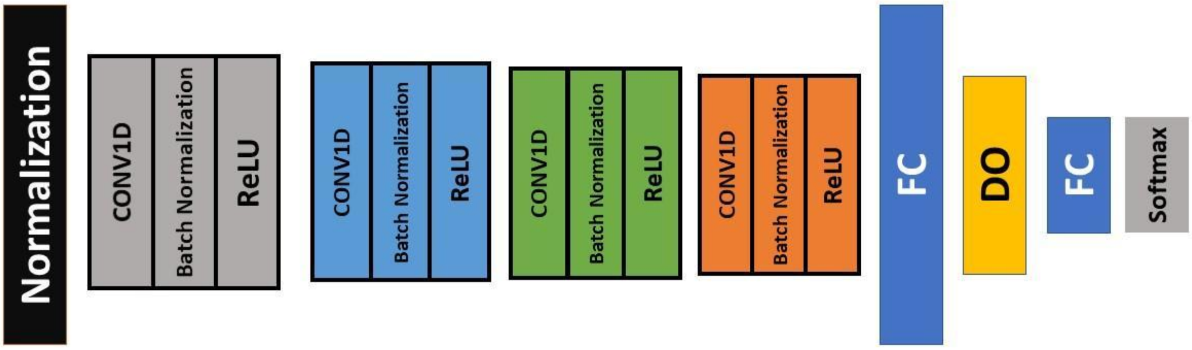
   Using the CNNs to try to map input (which consists of Raw data divided into 12 time samples only) to the output classes (HV vs. LV) and (HA vs. LA).
   This model deals with the frontal electrodes only in the brain as they are more accurate in the results.

   The proposed CNN architecture:[4]

   The CNN Architecture consists ofNormalization layer followed by 4 blocks of convolutions where each block consists of a CONV1D layer

[4] This approach was inspired from the model by paper number [10] in references.

followed by a Batch Normalization followed by a non-linear activation layer (ReLU).
The number of kernels for the first block equals to 32, has a receptive field of (3x1), and with strides of (3). Second block has 24 kernels, receptive field of (3x1) and strides of (2). Third block has 16 kernels, receptive field of (3x1) and strides of (2). Fourth and last block contains 8 kernels, receptive field of (5x1) and strides of(3).
The output of the 4 convolution blocks is then flattened and then , fully connected layer that has 20 nodes and (ReLU) activation function passed by a DropOut layer of 0.5 and then finally into a fully connected layer that has 2 nodes and a Softmax activation function.
The input size for the model is (672x1). The first axis is the number of readings captured by electrode in 1/12 timesamples of electrode electrodes (frontals only)



Layer	Parameters
Normalization	Batch Normalization
CONV1D - 1	(5), 32 filter, 3 Strides
	Batch Normalization
	ReLU
CONV1D - 2	(3), 24 filter, 2 Strides
	Batch Normalization
	ReLU
CONV1D - 3	(3), 16 filter, 2 Strides
	Batch Normalization
	ReLU
CONV1D - 4	(5), 8 filter, 3 Strides
	Batch Normalization
	ReLU
FC	20 Nodes
Dropout	0.5
FC	2 Nodes
Classifier	Softmax

We convolve with the 12 channels 12 times in each extracted time sample in order to get the common features between the 12 electrodes along the time series. in order to increase the input data as the number of data is so low. Each convolution block is used to extract some features relevant to the electrode and emotion along all time samples then it is normalized and activated using a ReLU function to be entered to the next block to extract some new features.

After 4 blocks of features extraction these features are then classified using a Softmax.

The testing:

We test with a session this session is divided into 12 then we enter the value of each session to the model of its electrode then take the max vote of all the time samples among the electrode repeat it until the electrodes are finished then max vote between all the electrodes this is how the result is extracted

The results:

We have applied the previous approach with two different ideas. The first one considers that all the EEG readings for the people are alike and therefore we trained a one model for valence and one model for arousal for the whole input.

For the first idea, both of our valence and arousal models trained on 25 epochs and with batch size = 1. Also, we used SGD as our optimizer. And Categorical Cross Entropy for the loss calculation.

The second idea, considers that all the EEG signals of the people are not alike and for that we will need to train 2 models (one for valence and one for arousal) for each person (i.e. 32x2 models). The same settings as the previous model are used in each model of the 64 models. 10 epochs, batch size = 1, SGD optimizer and Categorical Cross Entropy for loss calculation.

Each model was trained two times, one with Early stopping on validation loss with 7 steps patience and one without.

Test results:

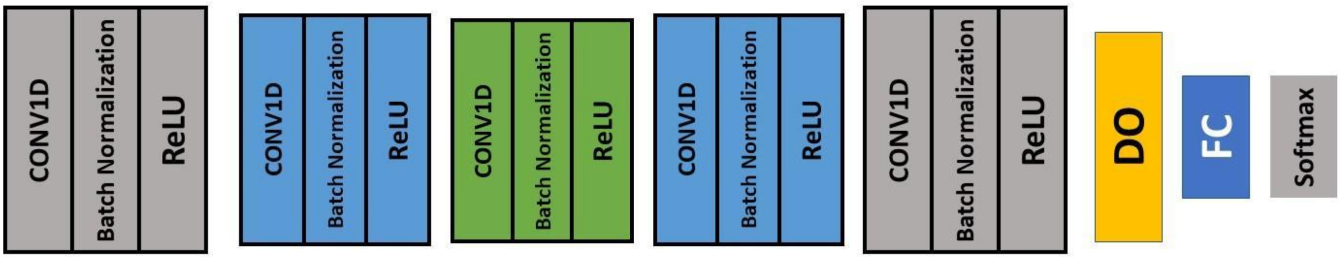
Model	Valence Accuracy	Arousal Accuracy
First Idea	39.06%	43.75%
Second Idea	75.7%	26%

Approach Three:

Using CNNs to try to map input which is Discrete Wavelet Decomposition of all electrodes as an input.

The proposed CNN Architecture:[5]

The architecture is composed of 5 convolution blocks. Each block contains 1D convolution layer and a ReLU activation function. After the 5 blocks we use a dropout layer then a Fully connected layer with 1 neuron to classify the output.



The first block contains 64 filters with a receptive field of (3) and no strides. The second block contains 32 fiters, with receptive field (3) and no strides. The third block contains 16 filters, with receptive field (3) and no strides. The fourth block contains 32 filters, receptive field of (3) and no strides. The fifth and last block contains 64 filters with a receptive field of (3) and no strides.

Layer	Properties
CONV-1D	(3), 64 filter
	ReLU

[5] This architecture was inspired from the Auto Encoders found in reference number [26]

CONV-1D	(3), 32 filter
	ReLU
CONV-1D	(3), 16 filter
	ReLU
CONV-1D	(3), 32 filter
	ReLU
CONV-1D	(3), 64 filter
	ReLU
DO	0.9
FC	1 Nodes
Classifier	Sigmoid

The input for that architecture is the DWT features with input size 165 value.
By testing the CNN we got the accuracy as shown in the next table.

Accuracy	Valence	Arousal
Train	**87%**	**71%**
Test	**76%**	**70%**

We also tested another idea for the same architecture, but instead inputting the data in 1D vector we inputted the data in 2D shape where we reshaped the DWT input to (5x33).
The same convolutions are used but instead of using 1D convolution layer we will use 2D convolution layer as follows.

Layer	Properties

CONV-2D	(3x3), 64 filter
	ReLU
CONV-2D	(3x3), 32 filter
	ReLU
CONV-2D	(1x3), 16 filter
	ReLU
CONV-2D	(1x3), 32 filter
	ReLU
CONV-2D	(1x3), 64 filter
	ReLU
DO	0.9
FC	1 Nodes
Classifier	Sigmoid

After training the above model the accuracies are as follows.

Accuracy	Valence	Arousal
Train	**85%**	**70%**
Test	**67%**	**62%**

User Manual

Chapter Five

1. the first step to load the data and keep in mind to change all the path variables to the path of the original data on your local machine.
   1.1. if we want to run the project in either machine learning or deep learning approach on all the electrodes we first run the “Allelectrode.py” file from “ReadingTheOriginalData” folder
   1.2. if we want to run the project in either machine learning or deep learning approach on all the electrodes we first run the “Frontalelectrode.py” file from “ReadingTheOriginalData” folder

        then “Labels.py” from “ReadingTheOriginalData” folder.

2. (optional step) if you want to increase the data, you can do so by running the “EMDAugmentation.py” from the “EMDAugmentation” which uses the Empirical Mode Decomposition to augment the raw input loaded in csv format.
3. The third step is to extract the features from the loaded data in csv format using any feature extraction method of your choice from the offered methods.
   3.1. choose which feature extraction method you want to work on ex. wavelet transform using db4 wavelets and run it’s file which in this case is “WaveletTransformdb4.py” and run it.
4. The fourth step is to choose which approach to train and classify the extracted features from the input data and run it.
   4.1. If you choose the machine learning approach, select if you want to work on the dataset as a whole or to work on the data of each person individually.
      4.1.1. If you want to work on the data of each person individually, choose which machine learning model you want to train ex.KNN then run “KNN.py” file from the “individuals” “Machine Learning Approach”.
      4.1.2. If you want to work on the data as a whole, choose which machine learning model you want to train ex.KNN then run “KNN.py” file from the “whole” “Machine Learning Approach”.

4.2. If you choose the deep learning approach, select which deep learning approach you want to train ex.CNN1 then run "Raw+Aug_CNN1Model.py" file from the "DeepLearning-Approach1" .

1. Reading The Original Data

    All the scripts in this Folder is dedicated to read the .dat files of DEAP dataset and write its EEG signal readings and Labels into .csv

    A. All electrode it is used to extract the data of each person session and write it into csv file
    B. Frontal Electrode it is used to extract the data of each person session of the12 frontal electrodes [FP1,AF3,F7,F3,FC1,FC5,C4,T8,FC6,FC2,F4,F8] and write it into csv file
    C. Labels is used to extract the labels of each session each person into 4 files (Velancy ,Arousal ,Dominance ,Liking) to label0,label1,label2,label3

2. Features

    All the scripts in this Folder is dedicated to read the previously prepared csv files made by Allelectrode in Reading The Original "DO NOT RUN IT ON THE DATA OBTAINED FROM FRONTAL ELECTRODE "

    Don't forget to resolve the paths as well

    A. WavletTransformdp4 This script project dp4 wavelet of level 6 on the signal obtained from the electrodes of the person in the session
    B. WavletTransformdp10 This script project dp4 wavelet of level 6 on the signal obtained from the electrodes of the person in the session
    C. WavletTransformdp4&PowerSpectrum This script project dp4 wavelet of level 6 on the signal obtained from the electrodes of the person in the session then calculate the the power spectrum density on the extracted wave
    D. ARM(energy,entropyandDWT) This script project dp4 wavelet of level 6 on the signal obtained from the electrodes of the person in the session then calculate its energy and entropy of 2 seconds is 50% overlap window

3. EMDAugmentation
   the script in this Folder is dedicated to read the previously prepared csv files made by Frontal electrode in Reading The Original then augment it by using EMD

4. Machine Learning Approach
   All the scripts in this Folder is dedicated to read the previously prepared csv files made by scripts a the Features Folder then enter then as input in Machine Learning Algorithms
   Don't forget to resolve the paths as well
   A. Adaboost : This Script enters all the previous extracted features on The Adaboost
   B. Bagging Tree:This Script enters all the previous extracted features on The Bagging Tree
   C. KNN:This Script enters all the previous extracted features on The KNN
   D. Logistic Regression:This Script enters all the previous extracted features on The Logistic Regression
   E. Random Forest:This Script enters all the previous extracted features on The Random Forest
   F. SVM:This Script enters all the previous extracted features on The SVM

5. Machine Learning Approach 2
   All the scripts in this Folder is dedicated to read the previously prepared csv files made by scripts a the Features Folder then enter then as input in Machine Learning Algorithms but run them individual by individual
   Don't forget to resolve the paths as well
   a. Adaboost : This Script enters all the previous extracted features on The Adaboost for each individual
   b. Bagging Tree:This Script enters all the previous extracted features on The Bagging Tree for each individual
   c. KNN:This Script enters all the previous extracted features on The KNN for each individual

d. Logistic Regression:This Script enters all the previous extracted features on The Logistic Regression for each individual
e. Random Forest:This Script enters all the previous extracted features on The Random Forest for each individual
f. SVM:This Script enters all the previous extracted features on The SVM for each individual

6. Deep Learning Approach 1

This file is dedicated to Run the deep learning model with the first approach to you need to Run the frontal and Labels script from the Reading The Original Data Folder
Please check the Paths

A. Raw+Aug_CNN.1Model this script the first Deep learning approach 1 to use the callback just uncomment it
B. Raw+Aug_CNN.1MultipleModels this script the first Deep learning approach 1 , but with a model for every individual to use the callback just uncomment it

7. Deep Learning Approach 2

This file is dedicated to Run the deep learning model with the second approach to you need to Run the frontal and Labels script
from the Reading The Original Data Folder
Please check the Paths

A. FirstIdea this script the Deep learning approach 2 to use
B. SecondIdea this script the Deep learning approach 2 , but with a model for every individual

8. Deep Learning Approach 3

This file is dedicated to Run the deep learning model with the second approach to you need to Run the frontal and Labels script
from the Reading The Original Data Folder
Please check the Paths

A. DWT_CNN this script the Deep learning approach 3 with 1D Convolution Architecture
B. DWT_CNN_2d this script the Deep learning approach 3 with 2D Convolution Architecture

Conclusion and Future Work

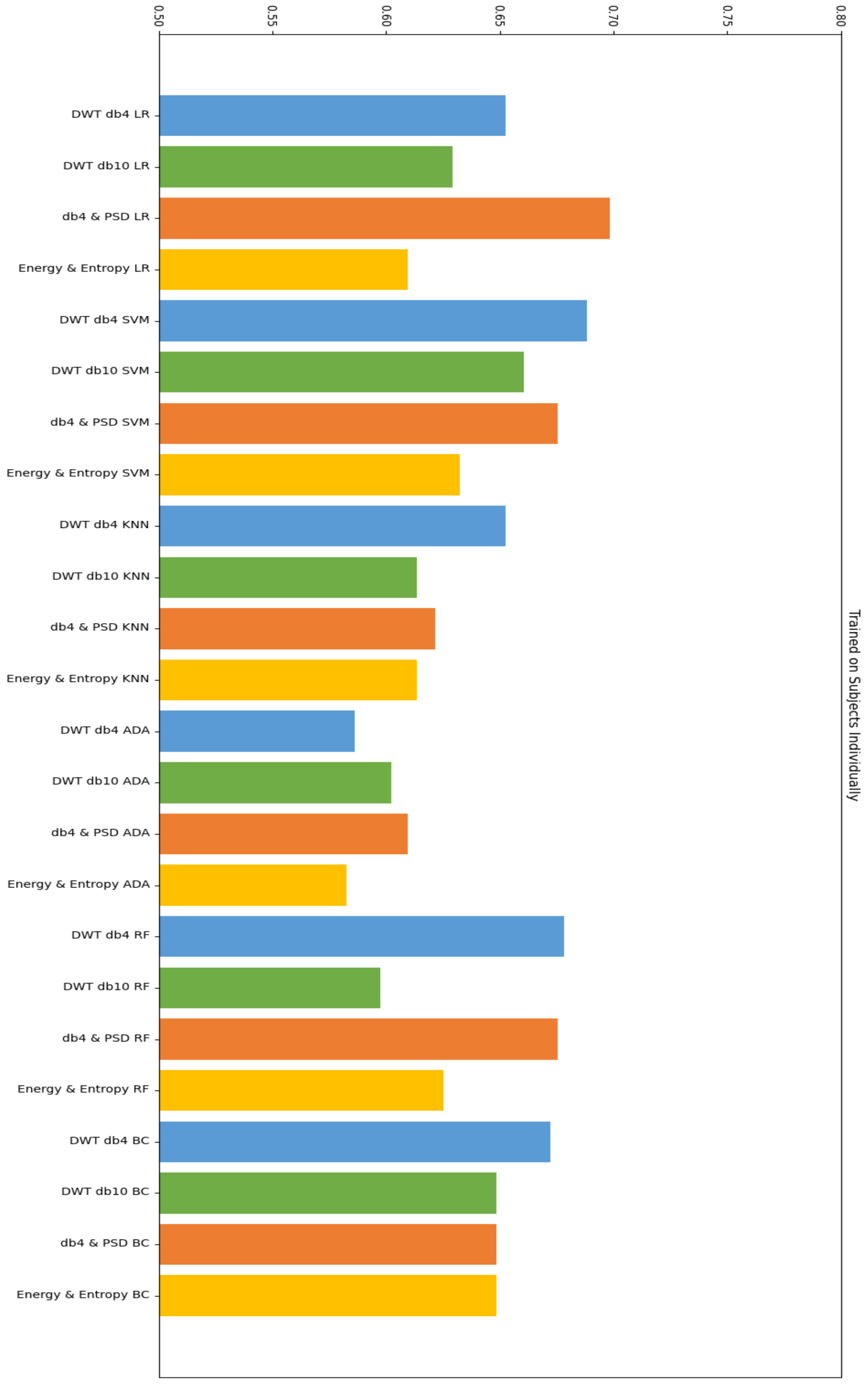
Chapter Six

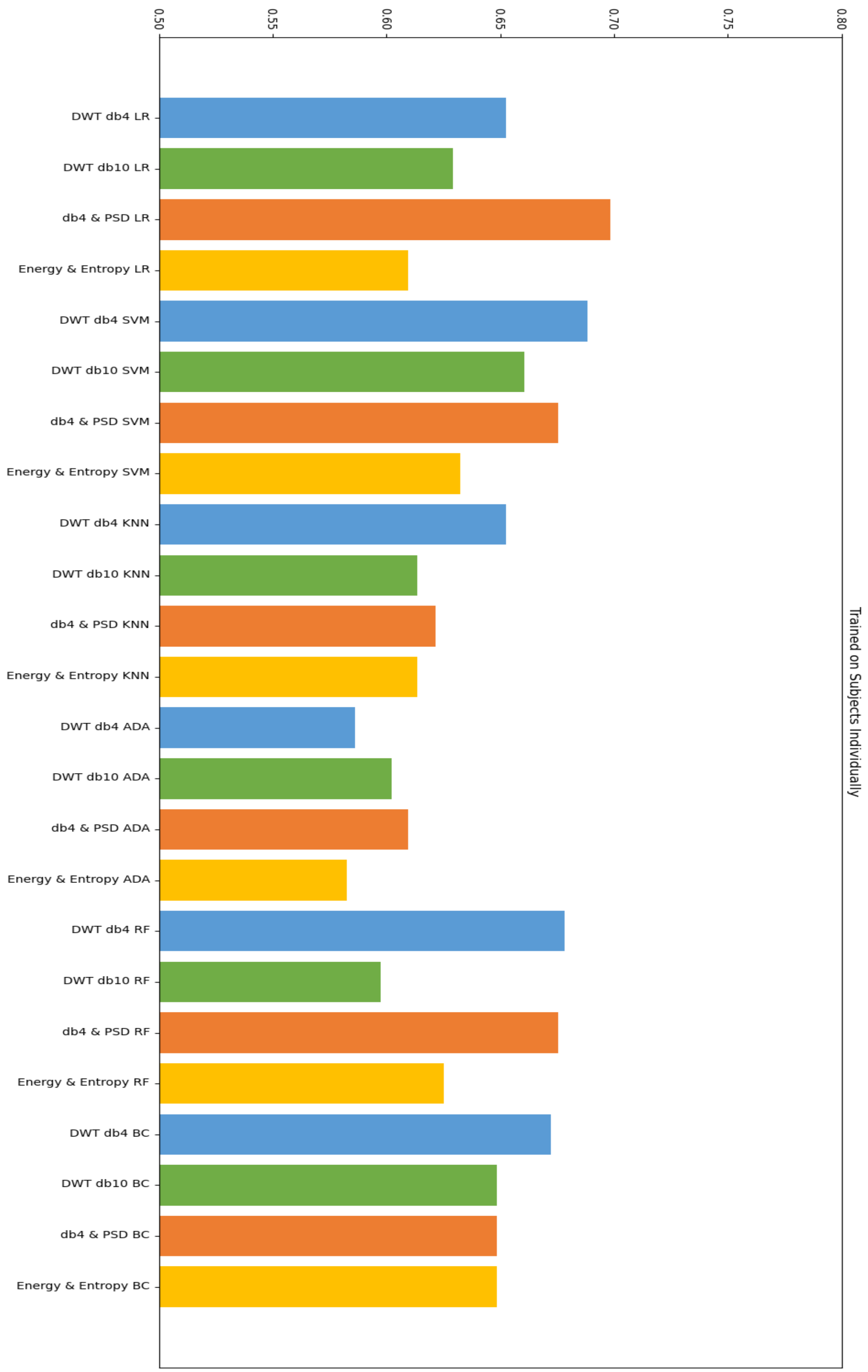
6.1 Conclusion

In this project, multiple machine learning and deep learning approaches have been explored and experimented with using different features associated with the EEG data and combinations between these features in order to train a model that can classify emotions in the human brain.

For the machine learning approaches used in this project, the logistic regression using the discrete wavelet transform feature on the db10 wavelet produced the best accuracy among the models trained on the data of all the subjects together. It was concluded that The discrete wavelet transform feature on the db10, db4, and db4 with power spectral density produce good results with most of the algorithms used, however the energy and entropy features produce the least accuracy among all the other features.

When training the same models with the same features on each one of the subjects in the data individually, the model with the most accuracy produced was the logistic regression model trained on the db4 discrete wavelet transform features with the power spectral density features. The following two figures show the achieved accuracy using Logistic Regressors, Support Vector Machine, K-Nearest Neighbours, AdaBoost using Decision Trees as base estimators, Random Forest and Bagging Classifiers using SVMs as base estimators with the db4 , db10 Discrete Wavelet Transform features, db4 combined with Power Spectral Density features, and Energy and Entropy features, all of which is trained on both the subjects individually and together.


0.50
0.55
0.60
0.65
0.70
0.75
0.80
DWT db4 LR
DWT db10 LR
db4 & PSD LR
Energy & Entropy LR
DWT db4 SVM
DWT db10 SVM
db4 & PSD SVM
Energy & Entropy SVM
DWT db4 KNN
DWT db10 KNN
db4 & PSD KNN
Energy & Entropy KNN
DWT db4 ADA
DWT db10 ADA
db4 & PSD ADA
Energy & Entropy ADA
DWT db4 RF
DWT db10 RF
db4 & PSD RF
Energy & Entropy RF
DWT db4 BC
DWT db10 BC
db4 & PSD BC
Energy & Entropy BC
Trained on Subjects Individually

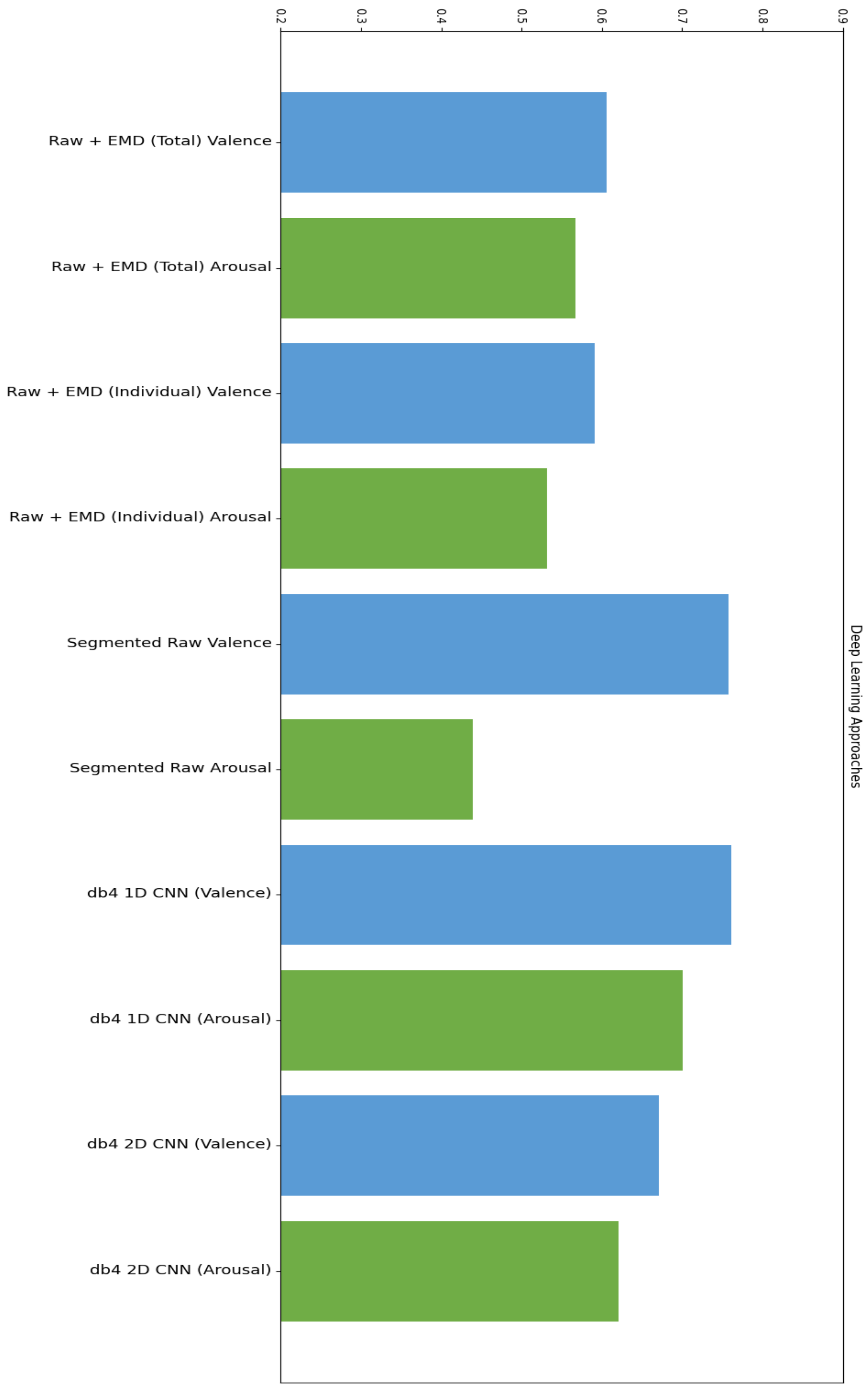

0.50
0.55
0.60
0.65
0.70
0.75
0.80
DWT db4 LR
DWT db10 LR
db4 & PSD LR
Energy & Entropy LR
DWT db4 SVM
DWT db10 SVM
db4 & PSD SVM
Energy & Entropy SVM
DWT db4 KNN
DWT db10 KNN
db4 & PSD KNN
Energy & Entropy KNN
DWT db4 ADA
DWT db10 ADA
db4 & PSD ADA
Energy & Entropy ADA
DWT db4 RF
DWT db10 RF
db4 & PSD RF
Energy & Entropy RF
DWT db4 BC
DWT db10 BC
db4 & PSD BC
Energy & Entropy BC
Trained on Subjects Individually

For the deep learning part in the project, three approaches were explored with different features and network architectures.

The first approach used uses convolutional neural networks in order to map the raw frontal electrodes as they are the most accurate when it comes to emotions and augments this data using EMD. The best accuracy achieved with this approach was 60.5% in valence, 56.6% in arousal. This accuracy was achieved by training the network using the data of all the subjects together. This approach was used again by training it using the data of each subject individually to achieve an average accuracy of 59% in valence and 63% in arousal.

For the second approach, CNN was used. The data used to train the model was the raw EEG data, and each sample was divided into 12 segments of about 5 seconds each, this is done for the 12 frontal electrodes only as they are the most relevant in emotion recognition. This approach was used twice, once with the data
of all the subjects together and once with each subject individually. The best accuracies produced from this approach was 75.7% for valence and 43.75% for arousal.

For the third approach, the features used in training the CNN was the db4 Discrete Wavelet Decomposition for all of the input electrodes. This approach produced an accuracy of 76% for valence and 70% for arousal when trained with 1D CNN . The data was reshaped and a 2D CNN was used to train on the data and produced an accuracy of 67% for valence and 62% for arousal. The following figure shows the accuracy among all the approaches used for deep learning


Deep Learning Approaches
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
Raw + EMD (Total) Valence
Raw + EMD (Total) Arousal
Raw + EMD (Individual) Valence
Raw + EMD (Individual) Arousal
Segmented Raw Valence
Segmented Raw Arousal
db4 1D CNN (Valence)
db4 1D CNN (Arousal)
db4 2D CNN (Valence)
db4 2D CNN (Arousal)

6.2 Future Work

For the machine learning approaches according to one of the latest papers written [25] it was concluded that the accuracy of most of the machine learning approaches do not passes the 75% (Specifically the SVM with topped accuracy in frequency domain 75.96% for valence and 55.31% for the arousal models. In our opinion the machine learning approaches has reached its top accuracy for us to be implemented.

For the deep learning models working with Raw Data only it reached its top accuracy as the most of the papers top accuracy is around 62% for the valence and 60% for the arousal.

On the other hand the deep learning models that work with combination of features has promising results as its top accuracy reached the 99% in some models.

In order to enhance our deep learning models we will need to combine our Raw Data with other features and augmentations and train our models on it so it can produce more accurate results.

Once we reach a good accuracy which we can rely on, we can start working on the practical approach for the project where we can start reading some data from the user and start classifying it. Noting that in order to reach a good accuracy in the project we will need to have a lot of good data to be trained on and tested.

References

Chapter Seven

01. Koelstra, S., Muhl, C., Soleymani, M., Lee, J.S., Yazdani, A., Ebrahimi, T., Pun, T., Nijholt, A. and Patras, I., 2011. Deap: A database for emotion analysis; using physiological signals. *IEEE transactions on affective computing*, *3*(1), pp.18-31.
02. Liang, Z., Wang, Y., Sun, X., Li, D., Voss, L.J., Sleigh, J.W., Hagihira, S. and Li, X., 2015. EEG entropy measures in anesthesia. *Frontiers in computational neuroscience*, *9*, p.16.
03. Liang, Z., Wang, Y., Sun, X., Li, D., Voss, L., Sleigh, J., Hagihira, S. and Li, X., 2020. *EEG Entropy Measures In Anesthesia*.
04. Bazgir, O., Mohammadi, Z. and Habibi, S.A.H., 2018, November. Emotion recognition with machine learning using EEG signals. In *2018 25th National and 3rd International Iranian Conference on Biomedical Engineering (ICBME)* (pp. 1-5). IEEE.
05. Dressler, O., Schneider, G., Stockmanns, G. and Kochs, E.F., 2004. Awareness and the EEG power spectrum: analysis of frequencies. *British journal of anaesthesia*, *93*(6), pp.806-809.
06. Dumermuth, G., Huber, P.J., Kleiner, B. and Gasser, T., 1971. Analysis of the interrelations between frequency bands of the EEG by means of the bispectrum a preliminary study. *Electroencephalography and clinical neurophysiology*, *31*(2), pp.137-148.
07. Zeiler, A., Faltermeier, R., Keck, I. R., Tomé, A. M., Puntonet, C. G., & Lang, E. W. (2010, July). Empirical mode decomposition-an introduction. In *The 2010 International Joint Conference on Neural Networks (IJCNN)* (pp. 1–8). IEEE.
08. Dinarès-Ferran, J., Ortner, R., Guger, C., & Solé-Casals, J. (2018). *A New Method to Generate Artificial Frames Using the Empirical Mode Decomposition for an EEG-Based Motor Imagery BCI. Frontiers in Neuroscience, 12. (pp 3-4)*
09. Zhang, Z., Duan, F., Sole-Casals, J., Dinares-Ferran, J., Cichocki, A., Yang, Z., & Sun, Z. (2019). A Novel Deep Learning Approach with Data Augmentation to Classify Motor Imagery Signals. IEEE Access, 1–1. pp(3-5)
10. Qazi, E.U.H., Hussain, M., AboAlsamh, H. and Ullah, I., 2019. Automatic Emotion Recognition (AER) System based on Two-Level Ensemble of Lightweight Deep CNN Models. *arXiv preprint arXiv:1904.13234*.

11. Akansu, A. and Haddad, R., 2001. *Multiresolution Signal Decomposition*. 2nd ed. San Diego: Academic Press, pp.391-442.
12. Kehtarnavaz, N., Kim, N. and Peng, Q., 2008. *Digital Signal Processing System Design*. 2nd ed. Amsterdam: Elsevier, pp.175-196.
13. A., G., Manuel, J., Marty, J. and Munoz, R., 2020. *Implementation Of The Discrete Wavelet Transform Used In The Calibration Of The Enzymatic Biosensors*. [online] Available at: <https://www.intechopen.com/books/discrete-wavelet-transforms-biomedical-applications/implementation-of-the-discrete-wavelet-transform-used-in-the-calibration-of-the-enzymatic-biosensors> [Accessed 10 August 2020].
14. Ahmet Taspinar. 2020. *A Guide For Using The Wavelet Transform In Machine Learning*. [online] Available at: <http://ataspinar.com/2018/12/21/a-guide-for-using-the-wavelet-transform-in-machine-learning/> [Accessed 10 August 2020].

15. Chen, D., Wan, S., Xiang, J. and Bao, F., 2020. *A High-Performance Seizure Detection Algorithm Based On Discrete Wavelet Transform (DWT) And EEG*. [online] PlOSONE. Available at: <https://journals.plos.org/plosone/article/figure?id=10.1371/journal.pone.0173138.g003> [Accessed 10 August 2020].
16. Vieira, Sandra & Pinaya, Walter & Mechelli, Andrea. (2017). Using deep learning to investigate the neuroimaging correlates of psychiatric and neurological disorders: Methods and applications. Neuroscience & Biobehavioral Reviews. 74. 10.1016/j.neubiorev.2017.01.002.
17. .Zeiler, M.D. and Fergus, R., 2014, September. Visualizing and understanding convolutional networks. In *European conference on computer vision* (pp. 818-833). Springer, Cham.
18. .Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.
19. .Siddharth Misra, Hao Li, in Machine Learning for Subsurface Characterization, 2020
20. .Galdi, Paola & Tagliaferri, Roberto. (2018). Data Mining: Accuracy and Error Measures for Classification and Prediction. 10.1016/B978-0-12-809633-8.20474-3.
21. .https://medium.com/@ar.ingenious/applying-random-forest-classification-machine-learning-algorithm-from-scratch-with-real-24ff198a1c57
22. Cherry, K., 2020. *The Important Role Of Emotions*. [online] Verywell Mind. Available at: <https://www.verywellmind.com/the-purpose-of-emotions-2795181> [Accessed 11 August 2020].

23. Ryan, M., 2020. *Decomposing Signal Using Empirical Mode Decomposition — Algorithm Explanation For Dummy*. [online] Medium. Available at: <https://towardsdatascience.com/decomposing-signal-using-empirical-mode-decomposition-algorithm-explanation-for-dummy-93a93304c541> [Accessed 7 August 2020].
24. Alarcao, S.M. and Fonseca, M.J., 2017. Emotions recognition using EEG signals: A survey. IEEE Transactions on Affective Computing, 10(3), pp.374-393.
25. Chen, J.X., Zhang, P.W., Mao, Z.J., Huang, Y.F., Jiang, D.M. and Zhang, Y.N., 2019. Accurate EEG-based emotion recognition on combined features using deep convolutional neural networks. IEEE Access, 7, pp.44317-44328.
26. Wen, T. and Zhang, Z., 2018. Deep convolution neural network and autoencoders-based unsupervised feature learning of EEG signals. IEEE Access, 6, pp.25399-25410.