Physiological Sensor based Affective State Recognition

Fahim Fazle Habib
16201048
Khaled Mohammad
19141028
Sikder Shadman Sami
20141031

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
BRAC University
January 2021

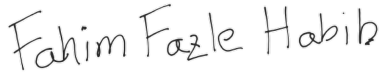
Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at BRAC University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Khaled Mohammad
19141028



Fahim Fazle Habib
16201048

Sikder Shadman Sami
20141031

Approval

The thesis/project titled "Physiological Sensor Based Affective State Recognition" submitted by

1. Khaled Mohammad (19141028)
2. Fahim Fazle Habib (16201048)
3. Sikder Shadman Sami (20141031)

Of Fall, 2020 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on (January 12, 2021)?.

Examining Committee:

Supervisor:
(Member)

Dr. Md. Golam Rabiul Alam
Associate Professor
Department of Computer Science and Engineering
BRAC University

Thesis Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
Associate Professor
Department of Computer Science and Engineering
BRAC University

Head of Department:
(Chair)

Dr. Mahbub Majumdar
Professor and Chairperson
Department of Computer Science and Engineering
BRAC University

Abstract

With rapid advancements of Medical IoT sensors in recent years, using them to recognize an individual's affective state has become more easily attainable. If an individual's physiological signals are recorded while they are made to experience certain feelings, the data can be used to create a model that can recognize those feelings using the sensor data. In this paper, a system is created to use data collected from physiological sensors to predict the affective state of the individual the data is extracted from. First, the sensor data was trimmed down to just the portions where the participants experience the feeling and filtered to get rid of unnecessary features and bad data. Then, the data was processed to condense the sensor readings of the entire time a user experienced a feeling into a single row that represents that time period. Finally, the data was mapped to the feeling felt. Instead of using generic colloquial terms for emotions, more abstract notions of defining emotions were used - specifically, the Valence-Arousal-Dominance space which defines emotions using these three parameters. Using that data-set, feature selection was done to find the most important features to feed to Machine Learning Models to detect the affective state of the patient in the Valence-Arousal-Dominance space. The novelty of our research comes from the features used to predict the emotions, which include statistical representations of the raw signal data and special domain features that give further insight into the signal data from EEG and ECG.

Acknowledgement

Firstly, all praise to the Great Allah for whom our thesis have been completed without any major interruption.

Secondly, we thank our supervisor Dr. Md. Golam Rabiul Alam for providing us the opportunity to research this topic and assisting us through the process in every possible way. His help guided us through the research process and allowed us to do what we did. We would like to express our special gratitude and appreciation to him for giving us such attention and time.

Lastly, acknowledgement must go to the BRAC university Computer Science curriculum and faculty members for providing us with the knowledge and equipping us with the critical thinking required to make it this far.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

CV Cross-fold Value

ECG Electrocardiogram

EEG Electroencephalography

KNN K Nearest Neighbours

RBF Radial Basis Function

RFE Recursive Feature Elimination

SVM Support Vector Machine

Chapter 1

Introduction

There is much merit in the idea of computationally reading the emotions of humans. Phones with increasingly powerful processors and rapidly improving cameras are carried by people everyday, everywhere they go. That, coupled with an increase in the popularity of wearable devices with heart rate and other M-IoT sensors in them that pair with the phones, means that there is, at the very least, potential for these to become good sources of data for emotional state recognition in the future. With the phone and wearables gathering data and sending them to powerful servers for processing, it is not difficult to imagine a future where data about users' feelings can be extrapolated and used to gauge user satisfaction or frustrations.

In terms of more clinical usage, the aforementioned pair can be used to easily track moods of patients with psychological disorders that could be useful in outpatient settings or just to constantly monitor problematic patients. In addition, even disregarding the future of mobile human data-set gathering, the traditional methods of gathering data currently in use for research can be used for many applications. In the media industry, advertisements and movies that are meant to elicit specific emotions can use automatic tagging of emotions using sensor data to get accurate reads and judge the effectiveness of their product.

However, there is also the problem with different languages not having a word defining a specific emotion that the model might try to predict. The Valence-Arousal-Dominance space of defining emotions eliminates this problem. It creates a standardized form of defining different emotions based on the combination of ratings on a 9 point scale for the different dimensions. The valence scale can have ratings ranging from pleasant to unpleasant. Similarly, the arousal scale can have ratings ranging from passive to active. Lastly, dominance scale can have ratings ranging from submissive to dominant. Creating a model to recognize these features rather than an emotion defined by colloquial words has merit in defining emotions universally and can also help to identify more complex feelings from recorded data.

1.1 Motivation

Active adoption of wearable technology with advanced hardware to record the wearer's physiological signals opens up numerous possibilities of using them. One such usage is the ability to determine the wearer's emotions using these sensor data.

Learning the emotional responses of people using these sensors have uses ranging from gauging if a product or service is providing the appropriate emotional responses, to finding out if a wearer is in a state of emotional distress and preventing emergencies. It can also have uses in the medical field to track at-risk individuals. As such, investigating different models to process the data from M-IoT sensors and provide estimations of affective state is required to eventually standardize the procedure of predicting emotions from such sensor data.

1.2 Problem Statement

It is often difficult to correctly identify an individual's underlying feelings. In general, people can attempt to piece together a person's feelings based on facial expressions and the way a person communicates - whether it is by spoken or written words. However, there are still issues with these methods. Firstly, people do not always effectively communicate what they are feeling in any form or even necessarily communicate about other things unrelated to the feeling that could be interpreted to point towards an affective state. This can have issues ranging from mental health concerns to people being dishonest about their feelings in cases like - giving emotional feedback for advertisements and movie previews' test audiences. Second, reading facial expressions to identify emotions is difficult for both humans and computers in its current state. Fortunately, all emotions also have physiological patterns that are associated with each emotion that could be recorded and processed to computationally predict the emotion that is felt.

1.3 Thesis Objectives

The objective of this paper is to create a model that can accurately label an individual's affective state by using physiological signals that are provided by M-IoT sensors. In order to do this, the first step is to gather M-IoT sensor data recorded while an individual is experiencing certain feelings and having the user self-report the feelings to be mapped to the recorded data.

The next step is to interpret the data-set and preprocess it to prepare for it to be fed to a machine learning model. The raw sensor data recorded must have data points that are not of any use to the project and need to be trimmed and features that need to be dropped. Lastly, recorded data needs to be simplified to a single row of data to map the users' self reported emotional ratings to.

After the necessary processing of the data is done, the final data-set must then be divided into three parts for training and testing the SVM and XGBoost model to obtain and verify the results.

In short, the objectives can be summarized as follows:

- Using data collected from M-IoT sensors recorded while subjects are made to feel specific emotions

- Cleaning the data-set to prepare for processing by getting rid of rows containing erroneous or null values.
- Using statistical functions like mean, skewness and variance of the data to condense the full array of sensor readings from the time period the individual experienced the emotion into a single row of features.
- Mapping the participants' self reported ratings for valence, arousal and emotion felt to the features extracted from the session.
- Dividing the data-set to train the model and avoid underfitting or overfitting issues.
- Training various machine learning models such as Support Vector Machine, KNN, Gaussian Naive Bayes, Random Forest, Decision Tree with the training data-set to predict the affective state in terms of valence, arousal and emotion felt.
- Testing the completed model with the test data-set to verify results.

1.4 Thesis Organization

The paper is structured to outline a model to predict the affective state of an individual using data collected from physiological signals that are recorded as they experience the emotion. The main contribution to predicting the affective state that the authors hope to achieve is to engineer a set of features that can assist in the classification of the affective state.

First, in the Introduction section (Chapter 1), the motivation behind the research is made clear, highlighting the many uses this could have. It also clearly states the problem that this solves, in that, humans are not able to correctly identify others' emotions without external aid. Lastly, it gives a brief overview of what the paper's objectives are to outline what comes in the next sections.

After that, in the Literature review section (Chapter 2), further context is added to the background of the work. Firstly, it covers other work that the authors researched to identify how physiological signals are affected by an individual's feelings. It also covers work from external sources that outline different databases containing the necessary physiological data and what approaches others used to make similar classifications. Therefore, this section provides background on how these physiological signals can be used for this purpose and allowed the authors to come up with a novel approach to the problem.

Following that, in the Methodology section (Chapter 3), a more detailed description of the process is outlined and visualized. Also, the data-set and the features extracted from it is explored in detail. Lastly, the feature selection process is outlined, with explanations of how each model used functions.

Then, in the Performance Evaluation section (Chapter 4), the models that were

used for classification are listed and briefly described. After that, the performance metrics that were used are described and the training and testing procedure is outlined. Then, the accuracy for each classifier is visualized using plots and discussed in detail. Finally, the best accuracy results are presented and conclusions are drawn.

Finally, the Conclusion section (Chapter 5), concludes the paper by summarizing the work and speculating what future work using the same approach could entail.

Chapter 2

Literature Review

To gather information about how the research should be conducted, we had to review existing work on how emotions are correlated to physiological signals, what kind of data exists that can be used to research the prediction of emotion from physiological signals and what approaches others took when doing similar work.

To gather knowledge on how physiological signals can be correlated to different emotions, or affective states, of human subjects, several papers were studied. First, it was discovered that the eye, specifically the pupil, has several responses to what an individual is feeling. In the paper [7] Margaret M., et al. monitored the pupil diameter and the effects of emotional arousal and hedonic valence on it as pupillary response during image viewing. Heart Rate and skin conductance was simultaneously measured to identify whether parasympathetic or sympathetic activation mediate pupillary changes. Preliminary conclusions showed that pupillary changes were larger when viewing emotionally arousing images, be it pleasant or unpleasant. The pupillary changes in correspondence to the skin conductance change, solidified the interpretation that in terms of affective image viewing, sympathetic nervous system activity modulates these changes. Considering all these, the hypothesis that pupil's response during affective image viewing shows emotional arousal associated, is solidified with the data obtained. After that, information about the phenomonon of hippus was learned as H. Bouma, et al. described hippus as a sustained oscillation of the pupil with a period of 5 sec or so in their paper [2]. Hippus was found in almost every tested subject, although hippus shows a tendency of random occurrence, it is more evidently found when subjects are at a relaxed or passive state. Disappearance of hippus is directly correlated to mental activity in their paper. They concluded a continuous monitoring of the pupil as a good option to prevent hippus affecting pupillary measurements.

Other physiological signals that could be analyzed to detect emotional states include cardiorespiratory patterns like P. Rainville et al. suggests that specific emotions have specific cardiorespiratory activity patterns that can be identified and used to automatically detect said emotions in their paper [6]. The researchers focused on four emotions – fear, anger, happiness, sadness – and gathered 43 test subjects consisting of healthy adults. The test subjects were instructed to visualize a specific moment in their life where they felt one of these emotions and had their cardiorespiratory activity recorded while they thought about that memory. Similarly, they

also followed the same procedure while thinking of a neutral memory and were all made to self report the intensity of the feeling on a scale of $0 - 4$. The researchers then extracted features like changes in heart rate, heart rate variability, changes in respiration and respiration variability using exploratory principal component analysis Then, they created a model to map these feelings to a combination of these features using stepwise discriminant analysis.

It was also found that skin temperature also changes in response to feelings, as R. A. McFarland showcases the phenomenon of human skin temperatures changing with respect to emotions accompanying music they listen to[3]. The author worked with a test case consisting of one hundred college psychology students who are made to listen to two pieces of music evoking different two types of emotions. The first piece was to elicit arousing, negative feelings and the second was to elicit calming, positive feelings. In the first test run, the music evoking negative emotions caused the listeners' skin temperatures to decrease while the other piece, evoking positive feelings, caused the listeners' skin temperatures to increase. In a second run, with the same pieces, both music cased temperature increases but the calm, positive music had a negligible increase in the test subjects. The author concluded that music does affect skin temperatures predictably based on positive or negative feelings elicited by the piece, but the effect is dependable on whether the subject's prior familiarity with the piece.

M. Pantic et al, establishes the idea of "Implicit Human-Centered Tagging" in their paper [8]. They describe it as the generation of tags for multimedia data by the spontaneous, subconsciously done actions and behaviors of the users interacting with it. In fact, the tagging is termed implicit because the users exhibit these features without any conscious effort. In this study, they focus mainly on features extracted from behavior captured by audio-visual sensors, they acknowledge that physiological sensors can further aid in this and even solve some problems with using audio-visual sensors. Some examples of the features extracted would be body postures, like head tilts or nods, and facial expressions like frowns or smiles. They also describe the issues with this, like different cultures and backgrounds, or even the location where the data is taken from, affecting the spontaneous reactions to the data and not having the proper equipment to extract these features effectively.

After researching how physiological signals can affect emotions, research began on finding a set of data that could be used for our purpose. One such data base was created by M. Soleymani et al. who sought to create a database that could be used for research on recognizing and implicitly tagging emotions felt by humans [11]. In their paper, they provide detailed descriptions on their methods of data collection and provide reasoning behind the data they chose. To create the database, they arranged a multimodal setup to simultaneously record the participants' physiological data - like ecg, eeg, galvanic skin response and skin temperatures – along with others like face video from multiple angles and eye gaze data. Using this setup, 27 healthy adult participants of both genders were shown videos that are meant to elicit specific feelings and their data was recorded. After that, the participants were asked to report the feelings they felt in terms of the 3D valence-arousal-dominance space of defining emotions. The combined data was then tested to show that they are usable

in models to predict emotions using features from this dataset. In combination to it, J. Lichtenauer et al. details the different readings in the MAHNOB HCI database in their manual [10]. It provides context on how to read certain keywords, acting as a comprehensive guide for users. It also provides extra detail on the process of data collection, like which title in the bdf data maps to which electrode and the time buffer recorded before and after each session.

Aside from that, another dataset was by T. Kanade et al., who discuss the various issues faced while analyzing facial expressions and points out that there have been limited data sets to work with up till that point, with the methods they used for data collection being uncertain in terms of being generalized in their paper [4]. Specifically, they present a problem space for analyzing facial expressions with 8 dimensions that they discuss at length and believe must be adequately sampled to create robust systems of analyzing facial expressions. Two examples from the dimensions mentioned are the level at which each facial expression is described and the orientation of a subject's head and the background in which they are being pictured or videotaped etc. After that, they present their database, which was evaluated to be consistent with the established problem space during data collection and includes 2105 digital images from 182 adults of different ethnic backgrounds.

Finally, a multitude of papers were reviewed to gauge how others conducted similar research. For example, in their paper [14], Mimoun Ben Henia Weim et al. classified emotional statements using peripheral physiological signals based on arousal-valence evaluation. They used the MAHNOB-HCI Tagging multimodal database, from which the Electrocardiogram, Respiration Volume, Skin Temperature and Galvanic Skin response signals were utilized. Emotions were defined in three different ways, two ways included using self rating scales and and one used 9 emotional keywords. 169 features were extracted after removing artefacts and noise from the signals, the SVM model was used to classify emotional states. The results showed two signals to be most relevant in terms of detecting human emotions, they were respiration volume and electrocardiogram signals, the obtained accuracy was also substantial compared to related work on it.

C.Godin et al. discussed in the paper [13] how the development of wearable physiological sensors bring a lot of interest to the topic of emotion detection. Many databases of physiological signals exist and machine learning algorithms are applied to them, still there is scope to identify the most relevant signals to detect emotion. To identify these most relevant signals, they applied several feature selection algorithms to two databases DEAP and MAHNOB-HCI. Both the databases are available for research and use short video clips shown to the participants to instigate various emotions. Finally it was concluded that the Galvanic Skin response, Eye closing rate, Variance of Zygomatic EMG, Heart rate variability were the most relevant signals to identify emotions accurately.

Detecting Stress During Real-World Driving Tasks Using Physiological Sensors [5] is a paper that offers methods for interpreting and collecting physiological data during real world driving scenarios that helps in determination of relative stress levels of the driver. Jennifer A. Healey, et al. continuously recorded electrocardiogram, elec-

tromyogram, skin conductance and respiration of the drivers during a drive through the greater Boston Area on a set route. Data of 24 different drivers were analysed in two different ways, in Analysis 1, the features of five minute intervals were used to identify three levels of stress during three segments of the route, rest, highway and city driving conditions. Analysis 1 achieved an accuracy of 97%. Analysis 2 used continuous features at one second intervals through the entire drive with a metric of observable stressors, the results indicate that physiological signals can provide a measure of driver stress in future cars with ability of monitoring physiological data.

Upon reviewing these papers, we found that the combination of special domain features of Hjorth Components, Wavelets and Fourier Transform for EEG and ECG data, along with the use of statistical features to condense the session data was not explicitly used in others. Thus, making our approach novel.

Chapter 3

Methodology

The process begins with preparing the data for the final training and testing of the model. To begin with, each dataset - both the eye gaze and physiological data - in every session needs to be trimmed down to only contain the readings from when the participants are actually viewing the video. After that, rows that contain bad data needs to be dropped along with columns and features that are irrelevant.

Once that is done, both datasets need to be condensed into a single row of features that can then both be combined into a single row of features for that session and the user's self reported ratings can be mapped to it as the expected output. After that, this must be repeated for all the sessions and all the sessions' data must be combined to form a single dataset that contains all the sessions' data. This provides the final dataset that can be used with the model.

After that, the next step would be to find the most important features using feature selection algorithms, then use that as the dataset to feed the models. The feature selection methods that are to be used are: SelectKBest, Chi Square, Feature Importance and Recursive Feature Elimination.

Then, split up the data and feed the training data for classification to the model. Once the model is trained, the data that was split can be used to test the model to predict the affective state and crosscheck with the existing outputs. The models that are to be used are: SVM Linear, SVM Polynomial, SVM RBF, SVM Sigmoid, Decision Tree, KNN, Gaussian Bayes and Random Forest.
The entire process is outlined in fig.3.1 as the workflow diagram.

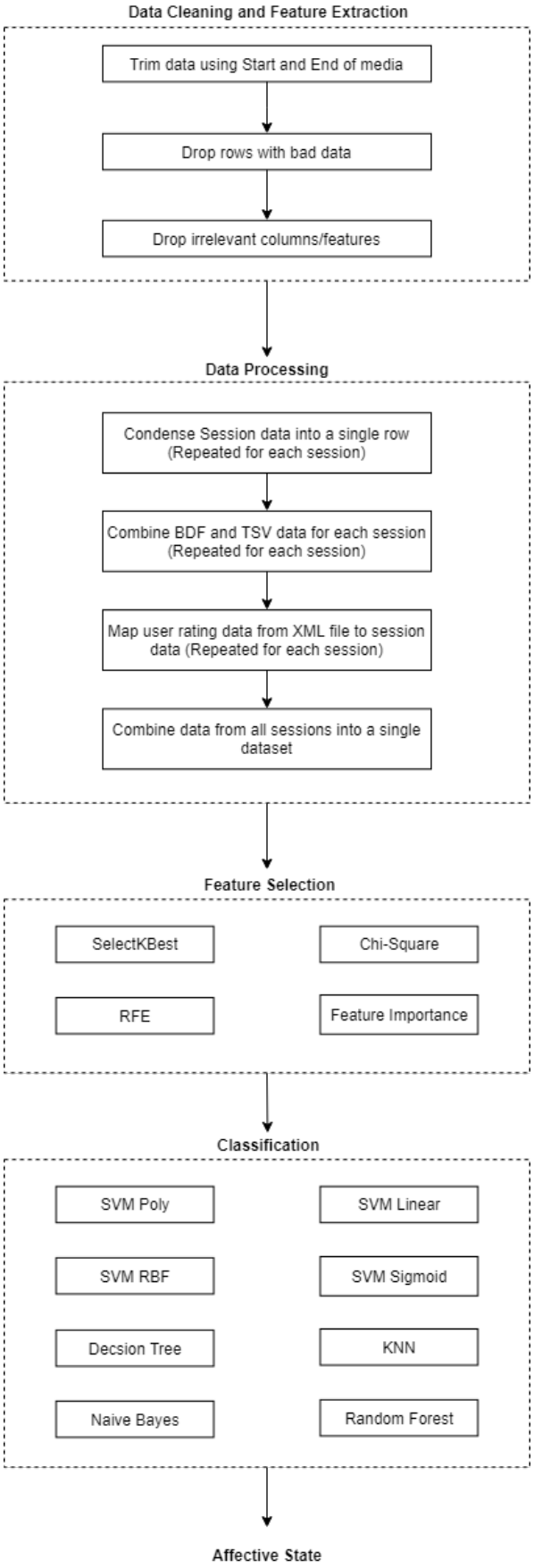


Figure 3.1: Workflow Diagram

3.1 Dataset Details

The MAHNOB HCI database was released to the public for use in research in tagging and predicting emotional states, it was the benchmark dataset used in this research in place of physically collecting physiological data from individuals. Specifically, the database provides multimodal data recorded while showing participants clips that are meant to elicit different emotions. There are 20 sessions for each participant and the participant pool consisted of healthy adults of both genders. Information about these participants are available in a directory named subjects and contain information about each subject like - date of birth, nationality and gender.

In case of sessions, there are directories for each session, with each session consisting of two separate datasets - one for the eye gaze data and the other for the eeg and physiological data. The eye gaze data is stored in a TSV file format and the physiological data is stored in a BDF file format.

The eye gaze data consists of data from the Tobii Eye Tracker and provides data like pupil size, the location on the screen where the participant is looking and location of the pupils in the camera image. It also contains the timestamps recording the entire duration of the session and has event indicators under the columns named "Event" and "EventKey". The event indicators signify actions the participants took, like a mouse click or keypress and when the media starts and stops playing. The self reported ratings of the user also appear in this column.

The physiological data contains the data from 32 EEG electrodes from the EEG cap worn by participants, three ECG sensors on the participants' chest, a respiration belt, Galvanic Skin Response from the left and middle finger, temperature taken from the left pinky and a status channel. All this data is mapped to channels named 1 to 47. The status channel is used to determine events like the "Events" column in the eye gaze data.

There is also a XML file containing all the user inputs and other specific data, like what media was played, in every session directory. Since this file has all the compiled user inputs, including the self reported ratings for emotions and valence-arousal-dominance, it is ideal for extracting the user ratings for that session. The ratings for Valence-Arousal-Dominance reported by the user range from 1-9.

3.2 Data Pre-Processing

In this section, all the processing done to prepare the data in order to train the model is covered in detail - exploring the thought process that was used. This aims to elaborate on the previously covered methodology and explain the current progress of the research.

First, the eye gaze data in the tsv and physiological sensor data in the tsv need to be trimmed to only have the readings from when the media plays. To do this, the beginning and end of the media being played must be identified for both datasets. In case of the TSV, there are events showing the beginning and ending of the media in the "Event" column termed "MovieStart" and "MovieEnd". For the BDF, it was mentioned that there was a 30 second buffer in the beginning and ending of the session. So, in each case both can be used to identify the index of the start and end of the media and trim the recordings before and after.

Next, certain columns which hold no significance for this research must be dropped, like the timestamps and events columns. Also, rows with NaN values are dropped. Then, the remaining columns must have the entire session's readings summarized and condensed into a single row for both the TSV and BDF. This is done using a combination of statistical functions to summarize the readings for each feature, condensing it to a single row with multiple columns for each feature. For the BDF data, there are additional features, the HJORTH Parameters, Wavelets and Fast Fourier Transform, for the wave data.

After that, the two datasets in the session need to be merged into a single row and the user reported ratings must be extracted from the XML file to be mapped to the row of data for the session.

Finally, this is all to be repeated for every session and compiled into a single csv file to get a row of data corresponding to each session. This final combined csv contains 512 session data, i.e rows, and 2241 features, i.e columns.

3.3 Feature Extraction

For the data pre-processing we used many statistical features as well as many special domain features depending on the type of data we were working on. We used only statistical features for features such as Eye Gaze, Temperature and Respiration, where as for features such as ECG readings, EEG readings and Galvanic Skin Response, special domain features were also used alongside statistical features. After pre-processing the data, we feed the data into Machine Learning Models to be used for classification of affective states.

3.3.1 Statistical Features

The statistical features were mainly used to provide a representation of the entire session data to be condensed into a single row of features that can be mapped to the used ratings for Arousal and Valence for that session. They provide relevant information about the data values recorded throughout the session in a concise form.

Mean

The arithmetic mean for a set of numbers is found by first summing up all the values in the set, then dividing the summation by the total number of values that are in the set. It represents the average of that specific set of values.

$$\mathbf{m} = \frac{\textbf{sum of terms}}{\textbf{number of terms}}$$

Median

Median is the value that seperates the higher half of a set of numbers from the lower half. It represents the middle value of the set, provided that its values are sorted.

$$\textbf{median}(\mathbf{x}) = \frac{\mathbf{1}}{\mathbf{2}}(\mathbf{x}_{\lfloor \mathbf{(n+1)/2} \rfloor} + \mathbf{x}_{\lceil \mathbf{(n+1)/2} \rceil})$$

Standard Deviation

Standard deviation in a set of values indicates the level of variation or dispersion from the mean of that set of values. A lower standard deviation indicates that the values are generally closer to the mean, that is, they are spread over a smaller range from the mean value. Similarly, a higher standard deviation indicates that they are spread further from the mean, over a larger range.

$$\mathbf{s} = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i) - \overline{x})^2}$$

Variance

Variance represents the square deviation from the mean of a set of values. It indicates how spread out the values in a dataset are from its arithmetic mean.

$$\mathbf{Var}(\mathbf{x}) = E[(X - \mu)^2]$$

Kurtosis

Kurtosis measures the degree of "tailedness" in the probability distribution of a random variable with real value. It defines the shape of the probability distribution. Kurtosis can be measured in several different ways for a theoretical distribution and have corresponding methods of estimation when extracting it from a population sample.
The fourth standardized moment, defined by the following equation, refers to kurtosis. Here, the fourth central moment is defined by μ_4 and σ is the standard deviation.

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

Skewness Skewness measures the degree of asymmetry in the probability distribution of a real-valued random variable. The asymmetry is measured about the mean of the set of data. The skew can have values that are positive, negative, zero, or undefined - each indicating the shape of the skew. For example, a positive value means that the tail of an unimodal distribution is on the right and a negative value means that it is on the left. There is no simple rule in skewness for situations where one tail is flat and the other is tall. If the value is zero, for instance, it means that on both sides of the mean, the tails balance out in case of a symmetric distribution. However, in an asymmetric distribution, if one tail is short and fat while the other is long and thin, the same can be true.

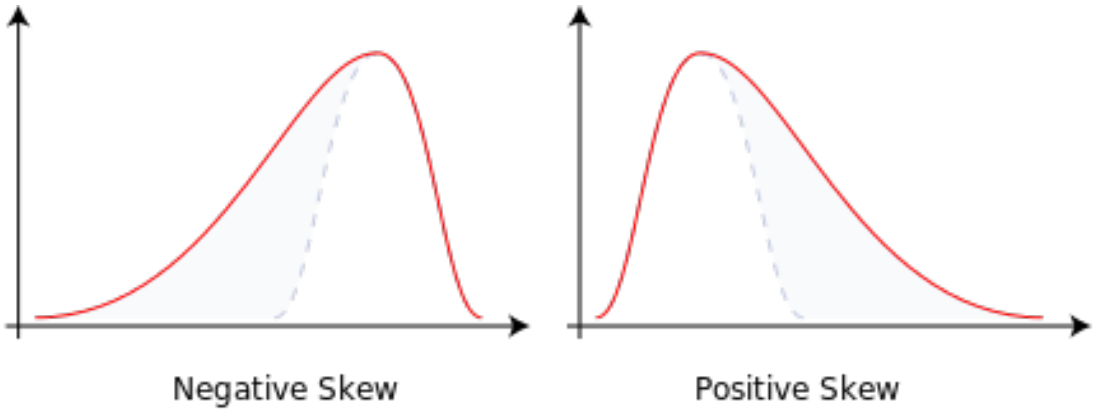


Figure 3.2: Skewness Diagram

$$\tilde{\mu}_3 = \mathrm{E}\left[\left(\frac{X-\mu}{\sigma}\right)^3\right] = \frac{\mu_3}{\sigma^3} = \frac{\mathrm{E}\left[(X-\mu)^3\right]}{\left(\mathrm{E}\left[(X-\mu)^2\right]\right)^{3/2}} = \frac{\kappa_3}{\kappa_2^{3/2}}$$

The above equation defines skewness. Here, the mean is denoted by , standard deviation is denoted by σ, expectation operator is denoted by E, the third central moment is denoted by μ_3, and the t-th cumulants are denoted by κt.

3.3.2 Special Domain Features

The Special Domain Features are used to add more texture to the EEG and ECG data, allowing the generation of more features which could contribute to the classification. These generated features also spanned the whole session, so they also had to be condensed using the statistical features to a single row of data.

Wavelets

A common tool for computational harmonic analysis is Wavelets, as detailed in the manual for Gwyddion [15], a tool for spectral probe microscopy analysis and visualization, that explains how wavelets function. In both the temporal (or spatial) domain and the frequency domain, they provide localization. The ability to conduct a multi-resolution assessment is a popular attribute. In a small fraction of the coefficients, the wavelet transformation of natural signals and images appears to have much of its energy concentrated. In applications such as data compression and denoising, this sparse representation property is fundamental to the good performance of wavelets. The wavelet transform, for instance, is a key component of the standard for JPEG 2000 image compression.

Two types of wavelet transformations were used: continuous and discrete.

In continuous wavelet transformations, the projection of a specific signal of finite energy is done on a continuous family of frequency bands (or subspaces of the L_p function space L2(R)). For example, the signal maybe represented on each band of frequency in the form [f, 2f] for all frequencies that are positive, that is, f ¿ 0. Then, the original signal can be restored by effectively integrating across all the frequency components that resulted from the original signal.

Frequency bands or subspaces (sub-bands) are the scaled representations of a subspace at scale 1. In turn, changing the mother wavelet, which is one of the functions in L2(R), generates this subspace. This feature is an example of the one-frequency band scale [1,2].

$$\psi(t) = 2\,\mathrm{sinc}(2t) - \mathrm{sinc}(t) = \frac{\sin(2\pi t) - \sin(\pi t)}{\pi t}$$

The functions, often referred to as child wavelets, generate the subspace of the scale or frequency band [1/a, 2/a].

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right)$$

In the above equation, a defines the scale and is positive, and b defines the shift and can be any real number. A point in the right half plane R+ x R is defined by the pair (a, b).

The mapping of a function x onto the subspace of scale a then has the form

$$x_a(t) = \int_R WT_\psi\{x\}(a,b) \cdot \psi_{a,b}(t)db$$

with wavelet coefficients

$$WT_{\psi}\{x\}(a,b) = \langle x, \psi_{a,b} \rangle = \int_R x(t)\psi_{a,b}(t)dt$$

Discrete transformations of wavelets are any transformation of wavelets for which the wavelets are discreetly sampled. As with other wavelet transformations, temporal resolution is a key benefit it has over Fourier transformations: it gathers information about both frequency and position.

In discrete transformations, wavelets that are formed are mutually orthogonal wavelets decomposed from the input signal. The wavelets could be created by using a scaling function that describes the scaling properties of the wavelets. The implication that certain mathematical conditions are applied on them are stated everywhere comes from the constraint that the functions used for scaling must be orthogonal to their separate translations. An example is the equation for dilation, as shown below.

$$\phi(x) = \sum_{k=-\infty}^{\infty} a_k \phi(Sx - k)$$

In the above equation, S acts as a scaling factor (usually valued at 2). Also, normalization but be done for the area in-between the function and function for scaling must be made to be orthogonal to the translations of its integers, i.e.

$$\int_{-\infty}^{\infty} \phi(x)\phi(x + l)dx = \delta_{0,l}$$

The results of all these equations can be obtained after some more conditions are introduced, for example - a set of coefficients a_k which are finite and simultaneously define both the scaling function and the wavelet. This needs to be done as the above restrictions do not produce a solution that is unique. Finally, the wavelet is obtained from the scaling function as

$$\psi(x) = \sum_{k=-\infty}^{\infty} (-1)^k a_{N-1-k} \psi(2x - k)$$

In the above equation, N is an even integer. The set of wavelets then forms an orthonormal basis which we use to decompose signals. Note that usually only few of the coefficients a_k are nonzero which simplifies the calculations.

For our dataset, the PyWavelets python library[16] was used for wavelet transformations.

Fast Fourier Transform (FFT)

Fast Fourier Transform (FFT) is an algorithm to calculate either the discrete Fourier transform (DFT) or the inverse discrete Fourier transform (IDFT) of a sequence. Fourier transform is used to transform a signal from its original domain, usually space or time, to represent it in the frequency domain. The reverse can also be done using the inverse Fourier transform. For instance, discrete Fourier transform (DFT) is used on a series of values to decompose them into components of varying frequencies. While this function has uses in a lot of fields, the computation from the general definition is impractical due to the time it takes to complete. As a result, fast Fourier transforms (FFT) are used to do these transformations with greater speed by performing factorization on the DFT matrix to obtain factors that are sparse (mostly zero). In fact, when applying FFT, the computation time complexity for calculating DFT is reduced to O(NlogN) from its original value of O(N^2) that arises from calculating it using its definition. In the big O notation, the N is used as a stand in for the size of the data. Therefore, for data that is very large i.e., with N with valued at several thousand or even several million, the speed at which the computation is completed for FFT or regular DFT can have extreme differences.

Hjorth Parameters

Hjorth Parameters were introduced in 1970 by Bo Hjorth [1] to further the analysis of signals in the domain of time by using statistical properties, mainly using the standard deviation of specific elements of the signal. Specifically, the Hjorth parameters consist of Activity, Complexity and Mobility. When working with analyzing electroencephalography signals, the parameters can be used to extract features from the signal data.

Hjorth Activity measures the square of the standard deviation of the amplitude. It could also be described as the signal power or as the surface of the power spectrum, as it is a measure of the variance of the time function in the frequency domain. This is expressed by the equation below, where the signal is represented by y(t):

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

The mobility parameter is calculated by dividing the standard deviation of the first derivative of the slope with that of the amplitude, that is, the standard deviation of the first derivative of the amplitude. The ratio here is only affected by the shape

of the curve itself so it acts as a measure of the relative average of the slope. The equation below showcases the calculation of Mobility.

$$\text{Mobility } = \sqrt{\frac{\text{var}\left(\frac{dy(t)}{dt}\right)}{\text{var}(y(t))}}$$

In the Hjorth Complexity parameter, using the sine wave as a reference, any extra details that differentiate it from the reference is measured to find the shift in frequencies. This is done by dividing the mobility parameters of the first derivative or the signal with that of the signal itself.

$$\text{Complexity } = \frac{\text{Mobility } \left(\frac{dy(t)}{dt}\right)}{\text{Mobility } (y(t))}$$

For our dataset, we used both Hjorth Mobility and Hjorth Complexity parameters. The compute_hjorth_complexity and compute_hjorth_mobility functions were used from the MNE-Features python library.

3.4 Feature Selection

A total of 2241 features for 512 sessions were attained after the data prepossessing stage, from which the more important features had to be found to be fed to the machine learning models. This was done using various feature selection algorithms described in the earlier section, i.e. SelectKBest, Chi-square, Feature Importance and Recursive Feature Elimination.

The number of features to be selected were adjusted various times to give the best accuracy for the models. At first, a smaller number like 20 were selected and then tested. After that, iteration was made with larger numbers to check for accuracy increases. It was found that, for our models, a larger number of features led to a better result.

The conclusion reached was that the number of sessions are not enough to find a smaller number of features that are strong enough to predict the outcomes by training, especially because the initial data-set is further divided to test and train data-sets. As such, a larger number, closer to the original size of the data-set was used to train and test the models. Figure 3.3 shows the difference in accuracy for the models using various number of features selected, using Valence. Figure 3.4 shows the difference in accuracy for the models using various number of features selected, using Arousal.

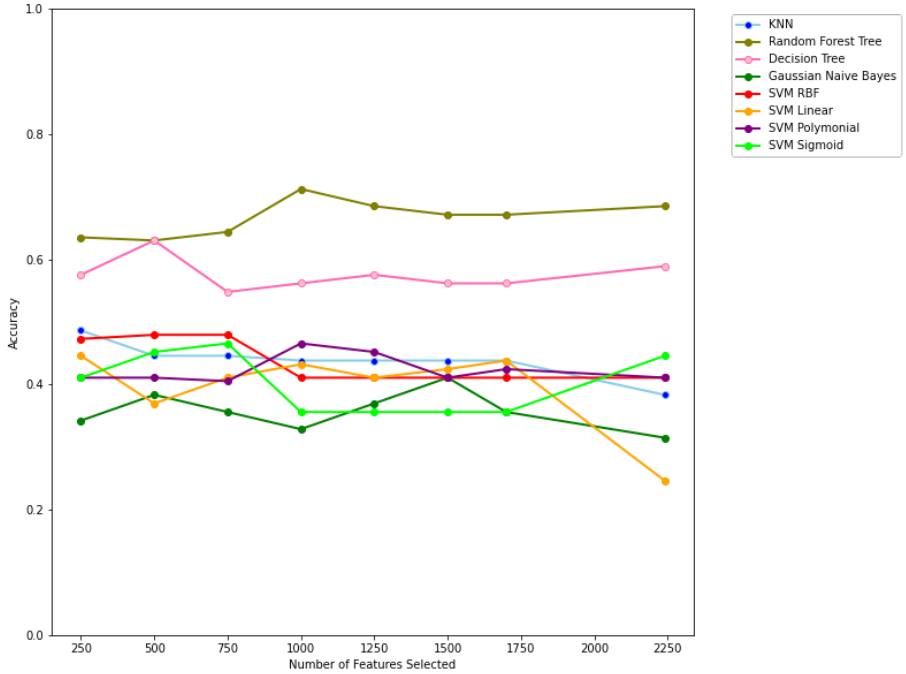


Figure 3.3: Accuracy of Models for various numbers of features selected (Valence)

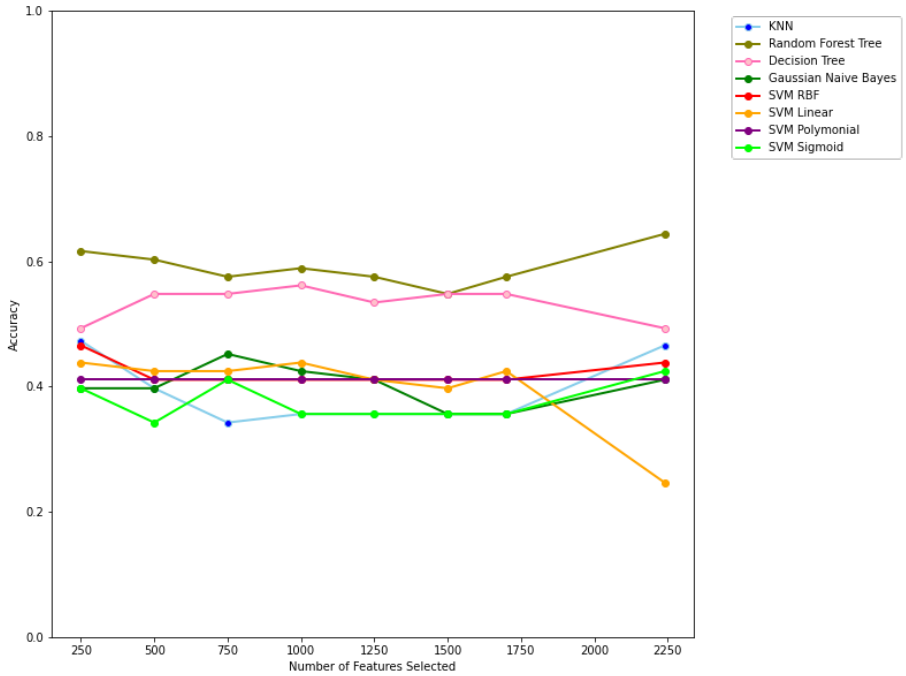


Figure 3.4: Accuracy of Models for various numbers of features selected (Arousal)

All the feature selection methods listed were used from the Python Scikit-Learn[9] Library, using mostly default settings. The methods also informed the understanding of the inner workings of the feature selection algorithms.

3.4.1 SelectKBest

SelectKBest feature selection has an internal function that is run to assign the features in the data-set given to it a score. Then, only the highest scored features, K number of them as stated when running the algorithm, are selected from the dataset.

3.4.2 Chi-Square Feature Selection

Chi-Square is a statistical feature that tests the degree of independence of two variables. Using this, the features that have the higher degree of dependence to the results are more likely to contribute to the prediction of the result and are picked. The following is the Chi-Squared Test equation, where E is the expected value and O is the Observed value.

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i}$$

3.4.3 Recursive Feature Elimination

In recursive feature elimination (RFE), first, a data-set is used in its entirety to run a model internally and weights are assigned to each feature. After that, features with the lowest weights are dropped and the process is done again. This continues until the desired number of features are found.

3.4.4 Feature Importance

Similar to Recursive Feature Elimination, Feature Importance has a model run internally that assigns weights to the features in the data-set that it is used on. After that, the most important features can be sorted to use the most important features and weed out the ones that are not as important, picking as many features as needed.

Chapter 4

Performance Evaluation

4.1 Model Specification

This section lists all the models used and briefly outlines how each model used in this paper make classifications.

4.1.1 Support Vector Machines (SVM)

In several classifications and regression problems, Support Vector Machines (SVM) are used as they are supervised machine learning algorithms. The support vector machine works to correctly distinguish 2 or more different classes in a classification problem by seeking an ideal separation line called a 'hyper-plane'. Using the SVM algorithm, data that is linearly separable is trained to find the optimal hyper-plane separation.

More formally, the algorithm (SVM) produces a higher dimensional space hyper-plane (if not linearly separable) that will assist with classification, outlier detection, regression, etc. By having a hyper-plane with the greatest distance to the data points used for training that are closest to it, a successful separation of classes is achieved.

In cases that are linearly separable, SVM is used to ensure that the hyper-plane has each observation on the correct side and correctly pick the optimal line such that the distance between the points closest to the hyper-plane and the hyper-plane itself is maximum. The hyper-plane acts as the decision boundary between the classes. The distance between the separating line and the data points closest to it is the margin [12].

For data that is not linearly separable, SVM can adjust itself using the concepts of Soft Margin and Kernel Tricks. Soft Margins try to find a line that separates classes with some added tolerance for datapoints that don't quite fall under the classification. To do this, they tolerate points that do not fall on the right side of the decision boundary but fall on either the correct or incorrect side of the margin. Kernel Tricks create new features that aid in correctly producing a decision boundary by using the features that already exist and applying some transformations on them.

The SVM kernels used for classifying in this case were: Linear, Polynomial, Radial Basis Function and Sigmoid.

The linear kernel is defined by the following equation:

$$K(x_i, x_j) = x_i \cdot x_j$$

The polynomial kernel is defined by the following equation, where d is the degree of the polynomial:

$$K(x_i, x_j) = (x_i \cdot x_j + c)^d$$

The RBF kernel is defined by the following equation:

$$K(x_i, x_j) = \exp\left(-\gamma \left\|x_i - x_j\right\|^2\right)$$

The Sigmoid kernel is defined by the following equation:

$$k(x, y) = \tanh\left(\alpha x^T y + c\right)$$

4.1.2 Gaussian Naive Bayes

Using Bayes' Theorem, which is used to calculate the probability of an event occurring by using the probability of another event which is assumed to have occurred, Naive Bayes classifiers are used to classify data. The assumption made when applying Naive Bayes classifiers is that all the features are independent and that they all equally contribute to the result, that is, they have the same weight. For Gaussian Naive Bayes classification, it is assumed that the features are distributed in a Gaussian distribution or normal distribution.

4.1.3 K Nearest Neighbour

K Nearest Neighbours is a supervised machine learning algorithm that can be used to classify data. To do this, it uses data points nearest to the given value it is meant to classify. The number of data points nearest to the value that the algorithm considers is denoted by k and the nearness of the points is calculated based in the Euclidean distance from the given point.

In short, when a new point is introduced and needs to be labelled under a class, it uses k number of points nearest to it in order to place it in a specific class.

4.1.4 Decision Tree

Decision Trees are used for classification of data. It is a supervised machine learning algorithm that classifies data using a directed tree that is rooted where the classifications are on the leaf node. All the connected nodes in-between that are not leaf nodes are the features and the branches contain the collections of features that get to a specific classification.

The branching features contribute to the classification, finally reaching the leaf nodes that contain a specific classification once the branch conditions are satisfied.

4.1.5 Random Forest

Random Forest operates as an ensemble with each tree in the forest giving a class prediction. There are a large amount of decision trees in the forest that individually work and give out a prediction, then the one with the most votes is picked as the final result.

The forest as a whole being relatively uncorrelated allows it to have better performance by protecting each tree from individual trees' errors. As a whole, they have better performance than any individual tree might have produced.

4.2 Performance Evaluation Measure

A very renowned performance measure, F-Score, has been used to evaluate the results of our research. To calculate F-Score precision and recall of the test is used. The precision is the total number of correct identification of positive results, divided by the number of all positive results inclusive of incorrect results. The recall is the fraction where the divisor is the total number of samples which were expected to be identified correctly, and the dividend is the total number of results that were identified correctly. The maximum possible value for F-Score is 1, which is the best accuracy, and the lowest possible value for F-Score is 0, which is the worst accuracy.

$$\text{Precision } = \frac{TP}{TP + FP}, \qquad \text{Recall } = \frac{TP}{TP + FN}$$

F-Score can be calculated using the following equation:

$$F - Score = 2 \cdot \frac{\text{precision } \cdot \text{ recall}}{\text{precision } + \text{ recall}} = \frac{\text{TP}}{\text{TP} + \frac{1}{2}(\text{FP} + \text{FN})}$$

TP = number of true positives
FP = number of false postitive
FN = number of false negatives

4.3 Training and Testing the Models

After the final data-set is found, it is further divided into two parts, a training set and a testing set. The training set is used to train the models and the test set is used to test the model. Also, to evaluate the models at different cross-fold values (CV) a script was written to to test each model at CV ranging from 2 to 9 and note the accuracy for each. The accuracy numbers for each model were then used to generate graphs that show their accuracy at each level.

The graphs detail the accuracy of the model named in the figure at each CV. The plot has CV in the X axis and accuracy in the Y axis. The peak shows the highest

accuracy achieved by that model and shows the CV at which it was found in an easily readable format.

There was a possibility that the number of sessions collected were not enough to accurately predict the Valence and Arousal classes in a scale or 1-9 as the users reported, i.e with 9 classes. This was confirmed after preliminary tests were done with 9 classes for each, which had very low accuracy scores.

Following that, the number of classes were reduced to test it further and it was decided to settle on these two classifications. The tables show how the Arousal and Valence ratings were translated from values ranging from 1-9 to 3 classes, table 7.3, and 2 classes, table 7.4, respectively.

Arousal	Valence	Rating (r)
Calm	Unpleasant	$1 \geq r \geq 3$
Medium	Neutral	$4 \geq r \geq 6$
Excited	Pleasant	$7 \geq r \geq 9$

Table 4.1: Classification of Valence and Arousal from Initial User Ratings (3 Classes)

Arousal	Valence	Rating (r)
Low	Low	$r \leq 4.5$
High	High	$r \geq 9$

Table 4.2: Classification of Valence and Arousal from Initial User Ratings (2 Classes)

The tests run with the above classes had the most promising results and were comparable to other research done on emotional classification using physiological data. As such, those classifications were chosen for training and testing. The results of the testing are detailed in the next section.

4.4 Accuracy for Three Classes

This section discusses the accuracy found for testing with three classes for Valence and Arousal. Separate subsections are dedicated to discussing the accuracy found for Valence and Arousal. In each subsection, the resulting accuracy of the models are discussed in terms of accuracy score, visible similarity in accuracy, consistency of accuracy and peaks of accuracy.

4.4.1 Accuracy for Valence

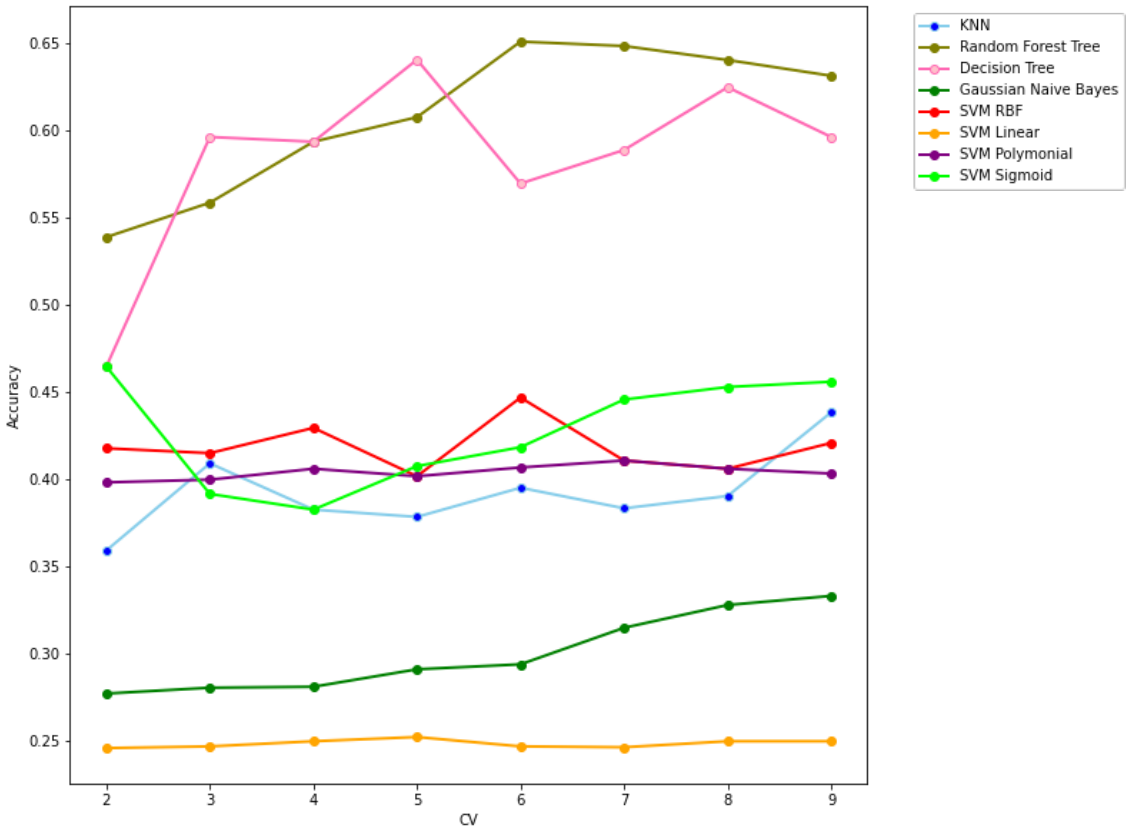


Figure 4.1: Accuracy of Models for Valence

Figure 4.1 illustrates the accuracy at different CV for all the classifiers tested for Valence with 3 Classes.The plot shows that for the CV tested, Gaussian Naive Bayes and SVM with the Linear kernel gave the lowest accuracy, with the Linear SVM being the worst amongst all the classification models. It is also visible that Random Forest and Decision Tree classifiers had the highest peaks and therefore the highest accuracy of the remaining classifers, both peaking above 0.60, at different CV. The remaining models formed a cluster, with performance in the accuracy range of 0.35 and 0.45. They also gave more stable accuracy with changing CV, with less dramatic peaks. It is also worth noting that the two models with the worst accuracy also have stable accuracy results with no dramatic peaks.

4.4.2 Accuracy for Arousal

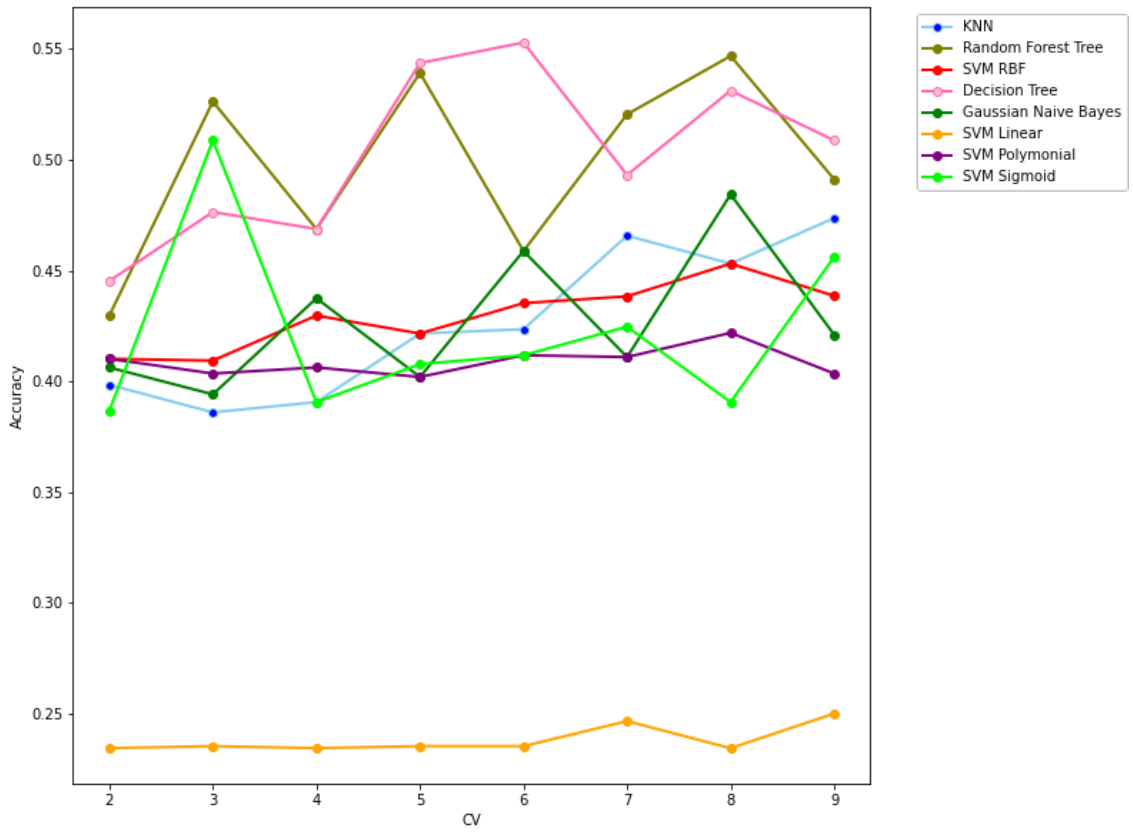


Figure 4.2: Accuracy of Models for Arousal

Figure 4.2 illustrates the accuracy at different CV for all the classifiers tested for Arousal with 3 Classes. Once again, SVM with the Linear kernel performed the worst in terms of accuracy. However, unlike for Valence, Gaussian Naive Bayes has risen to the middle of the pack for Arousal, joining the rest of the classifiers in the accuracy range of around 0.40 to 0.45. Leaving that aside, all the classifiers barring SVM with the RBF kernel and SVM with Polynomial kernel have dramatic peaks. Returning to similarities, like the results found in Valence, Random Forest and Decision Tree classifiers have the highest accuracy with peaks close to 0.55, which is higher than the rest of the classifiers for Arousal.

4.5 Accuracy for Two Classes

This section discusses the accuracy found for testing with two classes for Valence and Arousal. Separate subsections are dedicated to discussing the accuracy found for Valence and Arousal. In each subsection, the resulting accuracy of the models are discussed in terms of accuracy score, visible similarity in accuracy, consistency of accuracy and peaks of accuracy.

4.5.1 Accuracy for Valence

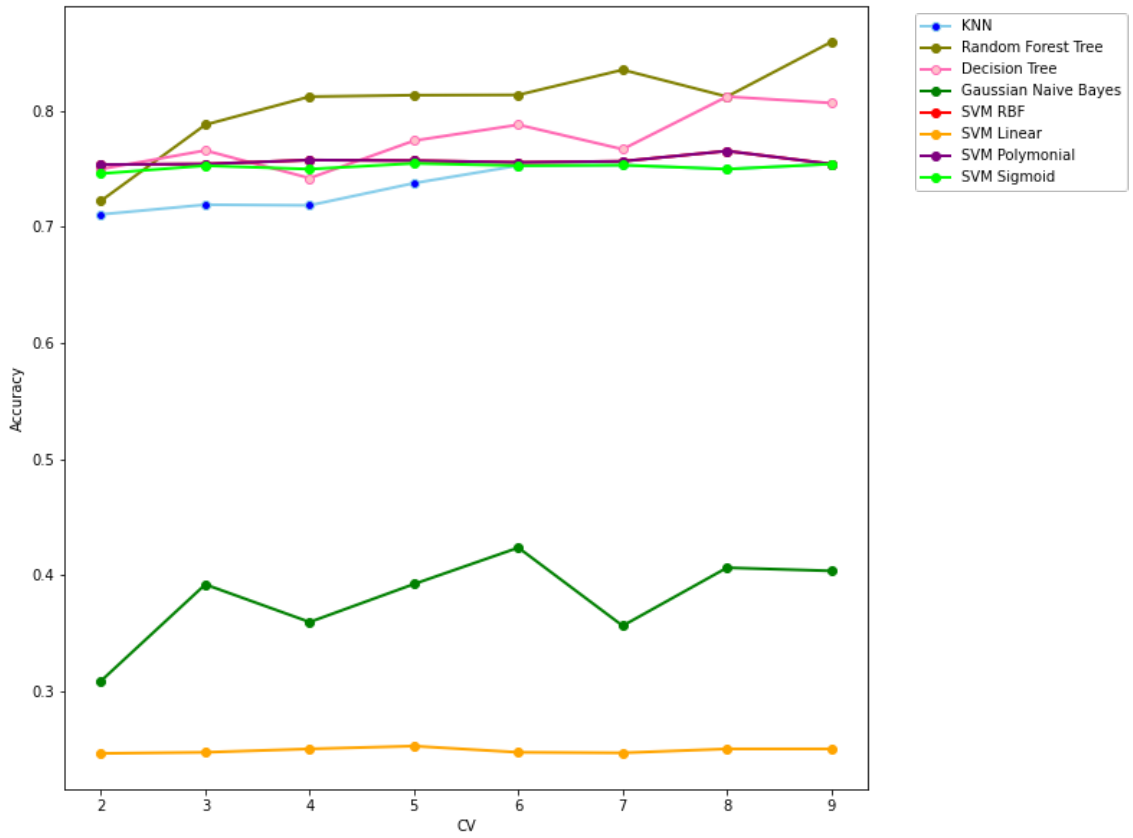


Figure 4.3: Accuracy of Models for Valence

Figure 4.3 illustrates the accuracy at different CV for all the classifiers tested for Valence with 2 Classes. In general, the accuracy plots show less significant peaks, with more consistent accuracy with different CV when compared to the 3 class figures. There are also more classifiers clustered towards the 0.7-0.8 range. The highest accuracy peaks come once again from the Random Forest and Decision Tree classifiers both peaking above 0.8 at different CV. Once again, SVM with the linear kernel had the lowest accuracy of all the models and it is joined by Gaussian Naive Bayes to be two models with lowest recorded accuracy of the rest. They also happen to have lower accuracy by a significant margin, Gaussian Naive Bayes peaking at around 0.4 and Linear SVM peaking around 0.2. The SVM classifiers with Polynomial, RBF and Sigmoid kernels also had nearly identical accuracy numbers in this case, both strictly staying near the middle of 0.7 and 0.8.

4.5.2 Accuracy for Arousal

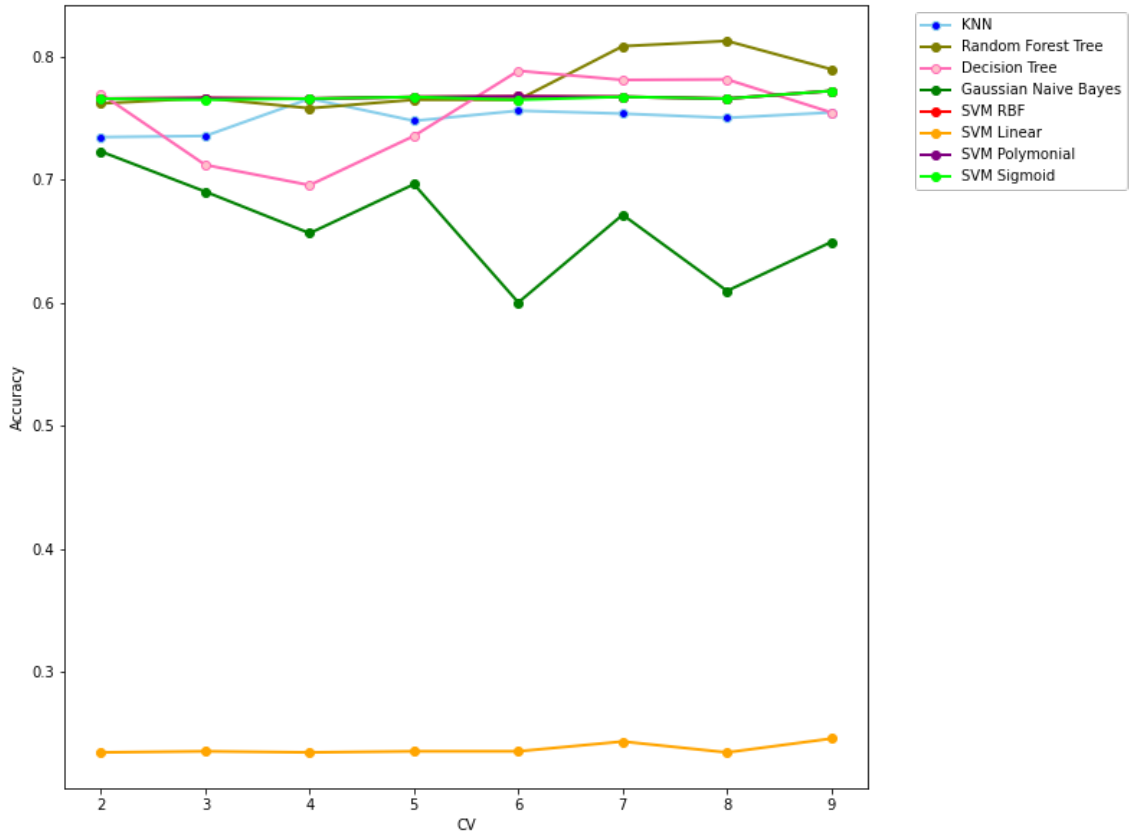


Figure 4.4: Accuracy of Models for Arousal

Figure 4.4 illustrates the accuracy at different CV for all the classifiers tested for Arousal with 2 Classes. Starting with similarities to the figure 4.4, the SVM Linear Kernel once again gives the lowest accuracy for all values for CV consistently. Gaussian Naive Bayes has accuracy results closer to the rest but have extreme peaks and dips bringing it farther down in accuracy from the range where the others fall. The rest fall firmly between 0.7 and 0.8 in terms of accuracy, with Decision Tree and Random Forest classifiers once again having the highest peaks of the bunch. However, Decision Tree does have some dips in accuracy in the CV range 2 to 4 that bring it out of line with the rest which fall squarely between 0.7 and 0.8 in accuracy. Only the Random Forest Classifier peaks above 0.8 in this case, the rest of the classifiers not named having a consistent accuracy around the 0.75 accuracy threshold with little peaks or dips. In fact, the SVM classifiers with RBF, Polynomial and Sigmoid kernels having accuracy that is nearly identical over all the CV.

4.6 Results and Discussions

From the observations in the previous section, it is evident that in general, using two classes for Valence and Arousal yields more accurate results at different CV when compared to using three classes. There are also less peaks and more consistent accuracy at different CV observed for most of the classifiers when only two classes are used. Migrating to specificity, the comparison now moves to the best accuracy from each classifier.

Model	Arousal	Valence
SVM POLY	0.422	0.411
SVM LINEAR	0.250	0.252
SVM RBF	0.453	0.447
SVM SIGMOID	0.509	0.465
DECISION TREE	0.553	0.641
KNN	0.474	0.439
BAYES	0.484	0.333
RANDOM FOREST	0.547	0.651

Table 4.3: Best Results for all the models using three defined classes for Arousal and Valence

Table 7.1 shows the highest accuracy found for each model for both Arousal and Valence, when using 3 classes for each. For predicting both Arousal, Decision tree had the greatest accuracy of all the models tested, with an accuracy of 0.553, while for Valence, Random Forest had the greatest accuracy of all models tested with an accuracy of 0.651. The rest hovered around the 0.4 to 0.5 range, with the worst accuracy coming from SVM Linear.

Model	Arousal	Valence
SVM POLY	0.772	0.766
SVM LINEAR	0.246	0.252
SVM RBF	0.772	0.766
SVM SIGMOID	0.772	0.755
DECISION TREE	0.788	0.812
KNN	0.766	0.754
BAYES	0.723	0.424
RANDOM FOREST	0.812	0.860

Table 4.4: Best Results for all the models using three defined classes for Arousal and Valence

Table 7.2 shows the highest accuracy found for each model for both Arousal and Valence, when using 2 classes for each. For predicting both Arousal and Valence, Random Forest Classifier had the greatest accuracy of all the models tested, with an accuracy of 0.812 for Arousal and 0.860 for Valence. The rest hovered around the 0.7 to 0.8 range, with the worst accuracy coming once again from SVM with the

Linear kernel.

The results show a substantial jump in accuracy as the classes are decreased. A jump from average accuracy goes to the range of 0.7-0.8 from an average accuracy of 0.4-0.5 as the classes are decreased from 3 to 2. The highest classification accuracy for the model also increases from 0.641 to 0.860 for Valence and 0.55 to 0.812 for Arousal. In both cases, the Decision Tree and Random Forest Classifiers performed provided the highest accuracy. On the other end, SVM using Linear kernel consistently had the lowest accuracy recorded.

The results of our efforts are comparable to other papers covering the same subject. In fact, the approach taken here produced better results than other work we reviewed [14], which worked with the same data-set and produced a highest accuracy of 0.6503 for Valence with two classes and a highest accuracy of 0.6875 for Arousal with two classes. Whereas in our case, with two classes, the highest achieved accuracy for Valence was 0.860 and for Arousal, the highest achieved accuracy was 0.812.

Lastly, the increase in accuracy observed over the course of testing as classes are reduced - first from the user reported nine classes, then reduced to three classes and finally the reduction to two classes - could indicate that there aren't enough sessions to extensively train and test the model to its fullest potential. The relatively low number of data existing in the data-set used had to be further divided for testing and training, which further reduces the available data for training. It could also explain why a larger number of features yielded better accuracy than when attempting with a smaller number of features.

Chapter 5

Conclusion

In conclusion, in this paper we have used physiological signals collected from participants experiencing an emotion for affective state recognition of the participants. The sensor readings were filtered and combined in the pre-processing stage. Once that is done, the readings for the entire session was condensed to a single row using statistical features. To gain further information on the EEG and ECG signal data, special domain features like Wavelets, Fourier Transformation and Hjorth Components were added, which also had to be condensed using statistical features to a single row for each session. Then, the user ratings were appended to each session and all sessions were compiled into a single data set. The data-sets then undergo feature selection to further identify important features and the final trimmed data-set is used with the models. The final data-set is divided into test and train sets that are used with the models. Lastly, The models were trained and tested twice, once with Arousal-Valence classification of two and another with a classification of three. The models used were SVM, KNN, Gaussian Naive Bayes, Decision Tree and Random Forest.

Finally, the models were able to accurately predict the Arousal-Valence classification in case of two classes, but had lower accuracy when trained to predict three classes. This is speculated to be due to the relatively low number of sessions, or rows, present in the data-set that are further reduced when the data-set is split to be used for training and testing. In the future, with more sessions to collect data using a similar setup, a more favourable result could possibly be found with three classes or even a higher number of classes for Arousal-Valence.

References

[1] B. Hjorth, "Eeg analysis based on time domain properties," *Electroencephalography and Clinical Neurophysiology*, vol. 29, 3 1970.

[2] H. Bouma and L. C. J. Baghuis, "Hippus of the pupil: Periods of slow oscillations of unknown origin," *Vision Research*, vol. 11, no. 11, pp. 1345–1351, 1971.

[3] R. A. McFarland, "Relationship of skin temperature changes to the emotions accompanying music," *Applied Psychophysiology and Biofeedback*, vol. 10, pp. 255–267, 1985.

[4] T. Kanade, J. F. Cohn, and T. Yingli, "Comprehensive database for facial expression analysis," *Proc. IEEE Fourth Int'l Conf. Automatic Face and Gesture Recognition*, pp. 46–53, 2000.

[5] J. A. Healey and R. W. Picard, "Detecting stress during real-world driving tasks using physiological sensors," *IEEE Trans. Intelligent Transportation Systems*, vol. 6, no. 2, pp. 156–166, Jun. 2005.

[6] P. Rainville, A. Bechara, N. Naqvi, and A. R. Damasio, "Basic emotions are associated with distinct patterns of cardiorespiratory," 2006.

[7] M. M. Bradley, Miccoli, Laura, Escrig, A. Miguel, Lang, and J. Peter, "The pupil as a measure of emotional arousal and autonomic activation," *Psychophysiology*, vol. 45, no. 4, pp. 602–607, Jul. 2008.

[8] M. Pantic and A. Vinciarelli, "Implicit human-centered tagging," *IEEE Signal Processing Magazine*, vol. 26, no. 6, pp. 173–180, Nov. 2009.

[9] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[10] J. Lichtenauer and M. Soleymani, "Mahnob-hci-tagging database," 2012. [Online]. Available: https://mahnob-db.eu/hci-tagging/media/uploads/manual.pdf.

[11] M. Soleymani, J. Lichtenauer, T. Pun, and M. Pantic, "A multimodal database for affect recognition and implicit tagging," *IEEE Transactions on Affective Computing*, vol. 3, pp. 42–55, 1 Apr. 2012.

[12] V. Apostolidis-Afentoulis, "Svm classification with linear and rbf kernels," 2015. [Online]. Available: https://www.researchgate.net/publication/279913074_SVM_Classification_with_Linear_and_RBF_kernels.

[13] C. Godin, F. Prost-Boucle, A. Campagne, S. Charbonnier, S. Bonnet, and A. Vidal, "Proceedings in international conference on physiological computing systems," PhyCS, Angers, France: 2nd International Conference on Physiological Computing Systems, 2015.

[14] M. B. H. Wiem and Z. Lachiri, "Emotion classification in arousal valence model using mahnob-hci database," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 3, 2017.

[15] P. Klapetek, D. Nečas, and C. Anderson, "Gwyddion user guide," 2019. [Online]. Available: http://gwyddion.net/documentation/user-guide-en/wavelet-transform.html.

[16] G. R. Lee, R. Gommers, F. Wasilewski, K. Wohlfahrt, and A. O'Leary, "Pywavelets: A python package for wavelet analysis," *Journal of Open Source Software*, vol. 4, 36 Apr. 2019.