HUMAN EMOTION RECOGNITION FROM PHYSIOLOGICAL BIOSIGNALS

by

Priyank G. Trivedi, B.E.

A thesis submitted to the Graduate Council of
Texas State University in partial fulfillment
of the requirements for the degree of
Master of Science
with a Major in Computer Science
December 2018

Committee Members:

Vangelis Metsis, Chair

Jelena Tesic

Moonis Ali

COPYRIGHT

FAIR USE AND AUTHOR'S PERMISSION STATEMENT

Fair Use

Duplication Permission

DEDICATION

I dedicate this thesis to my parents who have always been a constant support in my life. They are my source of inspiration. Their love, support, and encouragement have inspired me to be a very strong person, despite of many obstacles. Without them, none of my success would be possible.

ACKNOWLEDGMENTS

I want to express my highest appreciation to my Thesis advisor, Dr. Vangelis Metsis for his willingness to give his valuable time, constant guidance and to keep my progress on schedule. I am incredibly grateful for his words of encouragements, suggestions, and support.

I am highly indebted to Virtual Reality Technology Lab (VRTL) for providing me the opportunity and the environment to collect data; I would especially like to thank Dr. Kenneth Scott Smith and Dr. Mark Trahan for interviewing veterans during the data collection session. It wouldn't have been possible to complete the 'Using Virtual Reality Exposure Treatment and Real-Time Physiological Monitoring to Address PTSD Symptoms in Veterans' without the help of students who were working in developing Virtual Reality environment at VRTL.

I would also like to express my gratitude towards my thesis committee members, Dr. Moonis Ali and Dr. Jelena Tesic who helped me through the research by pointing me in the right direction.

My thanks and appreciation also goes to the people who directly or indirectly helped me in developing the Veteran Dataset; which includes all the participating veterans and computer science department at Texas state university.

TABLE OF CONTENTS

Page

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figure **Page**

LIST OF ABBREVIATIONS

EEG	Electroencephalogram
ECG	Electrocardiogram
EMG	Electromyography
EOG	Electrooculography
EDA	Electrodermal Activity
ReLU	Rectified Linear Units
SVM	Support Vector Machine
LG	Logistic Regression
PPG	Photoplethysmogram
RIP	Respiratory inductance plethysmography
ML	Machine Learning
SpO2	Peripheral capillary oxygen saturation

ABSTRACT

The goal of this research is to develop and evaluate methods of classifying human emotions using physiological bio-signals. As computing technology has been steadily growing and becoming more ubiquitous, it is essential that intelligent computer systems can determine the affective state of human subjects and adjust accordingly. Moreover, the assessment of a subject's emotional state in real-time can facilitate advanced therapeutic intervention tools, such as Virtual Reality Exposure Therapy (VRET), which can complement the traditional approaches. Although physiological responses in humans have been shown to correlate with their affective state, accurately determining someone's emotion from bio-signals remains a challenge. In this work, we have developed and experimented with a set of machine learning tools to optimize the task of emotion classification using physiological bio-signals. In a series of experiments, using a publicly available dataset as well as a dataset we collected during sessions of virtual reality exposure therapy with veterans suffering from social anxiety, we evaluate the utility of automatic emotion recognition for improved human-computer interaction as well as its use as an objective metric of emotional response monitoring during therapeutic interventions.

1. INTRODUCTION

The primary goal of this research is to determine human emotional stress or arousal through physiological data. Bio-signals or Physiological signals are signals in any living being which can be continuously measured such as skin conduction, breathing, temperature, etc. We as humans have gained the ability to perceive emotional stress and arousal via observation. Humans can observe environmental or appearance changes to detect the change in a state of being within ourselves and more notably the things around us with relative ease. This detection that humans perform very well is somehow a challenging task for machines. Machines, however, have the ability to measure small electrical pulse changes in the human body, but they lack the innate ability to understand what those changes mean. We take the machines ability to observe these bio-signals while introducing machine learning algorithms to recognize the patterns; this way we can try to identify what the changes mean concerning stress and arousal.

Computing has come a long way in the last few decades and technology has touched almost all aspects of life, but with regards to emotional recognition related research, technology's help is still somewhat limited. The limitation is due to the fact that biology-based research requires very high precision and bio-signals are highly subjective to a person and their vitals.

1.1 Motivation

The ability for a machine to understand its surrounding can provide new means to interact with the technology. As technology has touched all the areas of our life it will be meaningful to create human aware computing which can understand the need of the user or can act according to the surrounding situation. This can provide us with a wide range of possibilities of interacting or monitoring. Apple watch can be a great

example of this; the watch can detect irregular heart rhythms and can possibly warn the user that they are in distress, about to be in distress, or help the user receive help during an emergency in which the user is incapacitated by acting automatically. Clearly, the ability for a device to understand the physiological signals can save lives. Another example is the evaluation of therapeutic treatments. If we can provide machines with the ability to understand emotions, we can use it to quantify the effectiveness of treatments.

This research can be applicable to many real-world scenarios including: evaluations of different phycological treatments, creating more immersive computing experiences and monitoring human emotional state.

1.2 Challenges

This section describes, in brief, some of the challenges that we faced, and how we tried to resolve them. This is discussed in more detail within the sections to be followed.

The first stage of our research takes us to the basic concept, yet the most complicated part in practice, to obtain bio-signals with relevant stress markers that describe stress level at any given time. There are many facets to be considered, we do not only need reliable bio-signals, but we also need reliable labels to train our machine learning algorithms. One of the main problems with this is that bio-signals are subject to tiny changes in electrical signals and our body produces many electrical pulses every moment, which results in noisy readings. On top of that, bio-signals are highly subjective, varying with the health and physical activity of a person. The subjectivity could be more easily explained by stating that not everyone experiences the same difficulty when given the same task. To obtain labeled and reliable data, we decided to collect our own data as part of ‘Developing a Therapeutic Intervention

Environment using 360 Video and Virtual Reality' project which has been approved by under 'IRB 2017019'. Moreover, we have also used publicly available 'DEAP dataset.' The details on this are available in later sections.

The second stage is to pre-process the raw signal data and convert it into a feature dataset which can be fed to a machine learning algorithm. One of the challenges that we faced at this stage included an unbalanced dataset. If the dataset is favoring one of the classification labels, it can lead to lousy classification model. For example, if there were 99 positive samples out of 100 samples, to stay accurate, it is best for the model to predict everything as positive.

The third and final stage is to feed these data to machine learning algorithms, we have used algorithms such as Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Random Forest (RF) and a few other classifiers. Moreover, we have also utilized perceptron, and Long Short-Term Memory (LSTM) based neural network. As the data is difficult to classify, a lot of parameter tuning has been done to improve the accuracy which is covered in the later sections.

The recent developments made in the field of machine learning have proven to be a promising solution to a multifaceted problem that enables us to classify highly complex data easily. The objective of this research is to use these machine learning algorithms to our advantage in classifying bio-signals and develop non-invasive methods to get reliable physiological signals to determine the emotional state. The aim is to create a more streamlined method to collect and process the bio-signals in a non-invasive, reliable and more accessible way.

1.3 Research Findings

In this research we built a process pipeline to collect and process bio-signals. We also tested out different classification methods to classify this bio-signals to detect

stress or arousal. We have performed different feature extraction methods and a p-value based feature selection method with machine learning algorithms to achieve up to 70% accuracy. During the research we also curated a dataset of bio-signals with stress labels, which is collected from Veteran which are suffering from PTSD. We have also performed stress analysis during Virtual Reality sessions. From the results one of the observations is that Blood Oxygen level is the least effective bio-signal in determining emotional arousal or stress among the signals we collected, while we can also conclude from our results that Blood volume and Galvanic Skin response are a good indicator of stress or arousal.

We have also applied our method on a publicly available dataset and have achieved measurable gains using our feature selection and extraction methods.

2. BACKGROUND AND RELATED WORK

In 1842, Carlo Matteucci recorded the first electric activity of the heart of a frog. Since then, bio-electrical signals have been widely studied in medicine and are used in daily medical practice. The ability to record this small electrical activity has given us access to many different physiological signals which helps us understand heart activity, sleep activity, nervous system activity and many more.

There have been some efforts to automate the analysis of bio-signals using computers; early studies include the Pan-Tompkins algorithm [1] to detect QRS complex out of ECG. Most of the early bio-medical research has an association with either understanding the bio-signal or to automate the process of collecting bio-signals.

Our goal here is to detect stress/arousal through these bio-signals, and in order to do so, it's important to first understand the physiological response to stress in the human body. Our nervous system can be divided into two types of responses- Parasympathetic and Sympathetic system. The sympathetic nervous system (SNS) is one of two main divisions of the autonomic nervous system (ANS). Its general action is to mobilize the body's fight-or-flight response. The parasympathetic nervous system is one of the two main divisions of the autonomic nervous system (ANS). Its general function is to keep the body in a state of homeostasis and is also the body's rest-and-digest response. Mostly, the sympathetic nervous system is the one to control the body's response regarding receiving a threat. It is also responsible for controlling pupil dilation, salivary glands, heartbeat acceleration, bronchi dilation in lungs and glucose release. Work in this facet includes Emotion Recognition Using Bio-Sensors: First Steps Towards an Automatic System in which they included ECG, GSR and

EMG [2] to classify emotions. In our research we have collected and processed similar bio-signals as mentioned below:

2.1 Physiological Signals

- **Electrocardiogram (ECG)**: A measure of the electrical activity of the heart. ECG is measured using electrodes placed on the skin. Typically, electrodes are placed on both wrists or on both shoulders with a ground reference on the elbow. Additional electrodes are placed at predefined locations on the chest. These electrodes detect the electrical impulses generated from the heart. These electrical signals can interpret the rate and rhythm of the heartbeat.
- **Electrodermal Activity (EDA)**: Previously referred to as Galvanic Skin Resistance (GSR). It is a measurement of conductivity of the skin caused by sweating. It is a reliable indicator of sympathetic nervous system activation that changes by stress, anger excitement or joy. During periods of increased arousal, the electrical conductance of the skin changes especially when measured across the palm or sole of the feet. It is recognized as a key signal in lie detector equipment. EDA measurement is performed by placing two electrodes strapped to the tips of two fingers on the same hand.
- **Electromyography (EMG)**: A measure of the electrical activity of the muscle. The potential between two points along a muscle indicates the level of muscle activity. EMG measurement is performed by inserting a needle electrode directly into the muscle. These electrodes translate the signal into graphs or numerical values that can be interpreted. To detect an emotional response, EMG signals are typically made on muscles that tense under stress such as the muscles of jaw and shoulder.

- **Electrooculography (EOG)**: It is the measurement of eye movement based on corner-retinal potential. It is performed by placing electrodes above/below and left/right of the eye. It produces an electrical signal that corresponds to eye position. The eye is tightly coupled with our vision system that represents a significant portion of our brain factors. EOG is useful for sleep studies where eye movement cannot be determined visually. The speed of movement and duration of held gaze are likely more critical to the emotional state than the actual direction the subject is looking.
- **Plethysmography**: Measures the changes in volume in different parts of the body. Commonly, plethysmography is performed to check the blood flow in the arteries, especially in arms and legs. It is performed by using blood pressure cuffs or other sensors attached to a plethysmograph
- **Pulse oximetry:** Pulse Oximeter is a non-invasive method of monitoring oxygen saturation in blood. It is the most widely used method for collecting blood oxygen saturation. The device passes multiple wavelengths of light through the body to a photodetector, the photodetector measures change in absorption of all the wavelength, allowing it to detection absorption though blood.

2.2 Machine Learning Algorithms

We have used several traditional machine learning methods as well as deep learning to run various tests on our dataset. We will discuss all the machine learning algorithms used in brief:

- **Support Vector Machines:**

More commonly known as SVM, mostly it is used for classification tasks; however, the usage has also been extended for regression tasks. SVMs generate hyperplanes to separate different bins of data. The algorithm optimizes separating

hyperplanes to provide an accurate classification for new data [3]. This separation is optimized by placing the hyperplane between different classes of points and maximizing the separating distance from both.

- **Random Forest Classifier**

The Random Forest algorithm is based on the assembly of decision trees. The algorithm takes random subsets of the training data and generates decision trees based on the subset. Each decision tree makes a prediction, in this sense also known as a vote, and then the algorithm tallies those predictions and whichever prediction has the most votes is most likely the correct prediction.

- **K-Nearest Neighbors algorithm**

K-Nearest Neighbor is a non-parametric method; it can be used for classification and regression tasks. The classification or regression is done by taking a majority vote of surrounding neighbors in the feature space. The number of votes that must be made is K, which is an arbitrarily assigned number, that must be taken to perform the classification or regression.

- **Neural Network**

A Neural Network was made by loosely modeling the human brain, a simple neural network is known as a multi-class perceptron. The original Perceptron is a binary classifier; a perceptron will only fire if the input meets certain threshold that satisfies the activation function. To adjust the input values to the output neuron the network's weights must be updated. These weights multiply against the input, propagate forward in the network, produce an error based on an assigned loss function, and then back propagate throughout the network.

- **Long short-term memory Network**

LSTM cells also sometimes referred as LSTM units are part of the Recurrent neural network. An RNN, which is build using LSTM cells, is mostly referred as LSTM network. LSTM networks are very well suited for time series data. This is because LSTM networks have the ability to learn the patterns that only hold true for a series of inputs rather than an instantaneous input.

- **Naive Bayes classifier**

Naïve Bayes is a simple probabilistic classifier based on the Bayes theorem [4]. It has been one of the oldest classifiers and has been studied since the 1950s. It has remained popular method for text classification and span filtering. It used probability theory to classify data.

2.3 Related Research

Effect of emotional change in human physiology has been a long-debated topic, initially in 1927, James Lang was one of the first to support the theory of effect of human emotions on human physiology [5]. In 1964 Schachter and Singer argued that the experience of physiological changes was not sufficient in response to emotions [6]. In 1983, Ekman and Friesen challenged the view of Schachter and Singer and tried to divide emotional states into six different emotional states, which includes happiness, sadness, fear, anger, disgust, and surprise. The first effort to use pattern recognition to classify emotions were A.J. Fridlund and C.E. Izard in 1990 [7]. In 2014, Eun-Hye, Byoung-Jun, Sang-Hyeob, Myung-Ae Chung, Mi-Sook, Jin-Hun published research about emotion classification using machine learning. They used Electrodermal activity (EDA), electrocardiogram (ECG), skin temperature (SKT), and photoplethysmography (PPG) [8].

There have also been efforts made in identifying emotional responsive from visual changes, In 2004 Carlos Busso, Zhigang Deng, Serdar Yildirim, Murtaza Bulut, Chul Min Lee, Abe Kazemzadeh, Sungbok Lee, Ulrich Neumann, and Shrikanth Narayanan used computer vision approach to classify emotions in four different classes: Anger, Sadness, Happiness & Neutral [9]. In recent machine learning, the deep learning has gained a lot of attention; Jonghwa Kim and Elisabeth Andre tried emotions recognition based on physiological changes. They tried to classify emotions in positive and negative valence and positive, and negative arousal using EEG, EMG, respiration and GSR via neural networks [10].

3. METHODOLOGY

The activities involved in this project are: Data Collection, Processing, Labeling, Synchronizing Labels with Data, Feature Selection, and Feature Extraction and Training and evaluating different classifiers. We will go through each step-in detail within this section.

In order to get bio-signals, we have relied upon two sources. First, we have used publicly available 'DEAP dataset', and secondly, we have developed our dataset from veterans who are suffering from Post-Traumatic Stress Disorder (PTSD). Let's first get a basic idea about the bio-signals we are going to process and collect:

3.1 Data Collection - Veteran Dataset

The majority of our research time was dedicated in the collection of datasets that we gathered as an outcome of the concluded project namely 'Using Virtual Reality Exposure Treatment and Real-Time Physiological Monitoring to Address PTSD Symptoms in Veterans'. The overall methodology applied in the implementation of the project is described in detail in the following sections:

3.1.1 Background and opportunity

With prolonged military operations over the past decade and multiple deployments as well as harsh combat and environmental conditions, the number of veterans transitioning out of the military and reporting PTSD has been on the rise in recent years [11]. An increasing amount of combat veterans report struggles with "maladaptive patterns of social functioning" [12], due to a strong relationship between PTSD and Social Anxiety Disorder [13], characterized by distress in social interactions, social avoidance patterns, and impaired social relationships. Furthermore, the presence of anxiety for reintegrating combat soldiers is a crucial factor in externalized behavior problems such as aggression and substance abuse [14].

This research explores whether post-combat exposure stimulating anxiety, resulting from non-combat situations, increases resilience and decreases PTSD symptoms in combat veterans. Specifically, it addresses the avoidance distress of veterans with PTSD symptoms resulting in better engagement in day-to-day situations where overwhelming emotions of threat might be perceived (e.g. crowded classrooms, shopping centers etc.)

Personalized VR-based therapy, combined with real-time emotional and physiological response monitoring, will provide researchers with a physiological baseline for understanding the impact of VET on avoidance and social anxiety response. This research enables the future study of the use of personalized mindfulness training (Baer, 2015) assisting veterans in moment-by-moment awareness of their thoughts, emotions, bodily sensations, and the surrounding environment.

3.1.2 Project plan

The project - 'Using Virtual Reality Exposure Treatment and Real-Time Physiological Monitoring to Address PTSD Symptoms in Veterans' as a whole was designed to collect as well as process physiological data and to build VR environments for phobia therapy. But as a relevant portion to this thesis we have only performed collection and processing of physiological data. The remainder of the project is explained here to get a greater understanding of the entire project that was performed. Moreover, it shall also help to gain an understanding of why bio-signal processing is essential for the project.

In this study, we have employed a mixed-method approach utilizing a focus group, interviews, survey data, and quantitative emotional response metrics to

develop and evaluate a treatment for PTSD in combat veterans. The study has three objectives implemented in 3 partially overlapping stages (one stage per objective):

Objective 1: To identify non-combat related scenarios that cause heightened anxiety in veterans with combat-related PTSD. The goal is to characterize the experiences of combat veterans with PTSD in social situations. For this end, focus groups including 5-10 people identified 3-5 environmental scenarios/stimuli that reinforce low, medium and high anxiety for combat veterans.

Objective 2: To develop quantitative metrics to measure stress and emotional response. The goal is to pinpoint physiological biomarkers that can be effectively used for emotion recognition and further identify computer-based methods can be utilized to accurately analysis these biomarkers. The research team developed the VR content as well as the physiological bio-signal collection and analysis tools. Data from 8 participants is collected and manually annotated for system training purposes.

Objective 3: To develop immersive virtual environments that stimulate feelings of heightened graded anxiety. The goal is to understand what non-combat virtual environment scenarios elicit anxiety in a) combat veterans and b) general population, at three levels: low, medium, and high. It is expected that users will be able to identify low, medium, and high anxiety environments. Towards this end, the researchers developed three crowd encounter scenarios that produce low, medium and high anxiety. Five to ten participants were used for the performance evaluation of our methods. To further elaborate, the research team developed computer-generated imagery (CGI) and animations corresponding to the identified “high anxiety scenarios”. The CGI content is developed using free, open-source technologies such as Unity3D [15] and Python.

As we have gathered a general idea about the project, we will get into details of data collection and analysis aspect of the project.

3.1.3 Data Collection

We collected data in multiple incremental sessions, we learned from our mistakes and tried making changes to improvise on our data collection methods in every session. We collected data from veterans who are suffering from PTSD. The data were collected during interview and VR sessions; was aimed to keep the methods as non-invasive and non-intrusive as possible. Hence, we decided to use a device called BioRadio which is made by Great Lakes Neurotechnology. The device provides 8 or 4 different inputs to be gained in differential or single-channel mode respectively. The device is rechargeable and allows us to store data locally or live to stream the channels over Bluetooth. As per the official device manual, it is described as "The BioRadio is worn by the person and is designed for acquiring physiological signals from sensors attached to the body. Physiological signals are amplified, sampled, and digitized, which can be wirelessly transmitted to a computer Bluetooth receiver or recorded to onboard memory for post-analysis." [7]. Bio-radio also allows us to attach a pod which can be connected to thermistor and Pulse-Oximeter. The following table represents all the devices we used during the data collection process:

Table 1: List of used devices

Device Name and Make	Device purpose
BioRadio™ by Great Lakes NeuroTechnologies	Digitises and amplifies data. Can also act as storage and streaming device.
Respiratory Inductive Effort (RIP) belt by SleepSense (S.L.P. Inc).	It is used to collect data related to breathing activity.
Pulse Oximeter (3012LP) by Nonin Technologies	Used to collect pulse count and blood oxygen level.
MVAP-II electrodes by MVAP Medical Supplies	These electrodes are used to connect bio-radio with skin.

3.1.4 Electrode placement and preparations

Each participant was briefed with the overall process, their rights and obligations and their overall role with regards to the project. They were also provided with information about the collection process of the bio-signals. After receiving their consent to the whole process; we first cleansed the skin where the electrodes are supposed to be placed. We used alcohol swabs in this process to help remove dead skin cells as well as oils. Doing so gives the electrode better grip and accurate readings. We used Hydron Gel/ Silver Chloride based electrodes which provides good conductance and sufficient grip. Electrode placement for data collection can be inferred from below images:

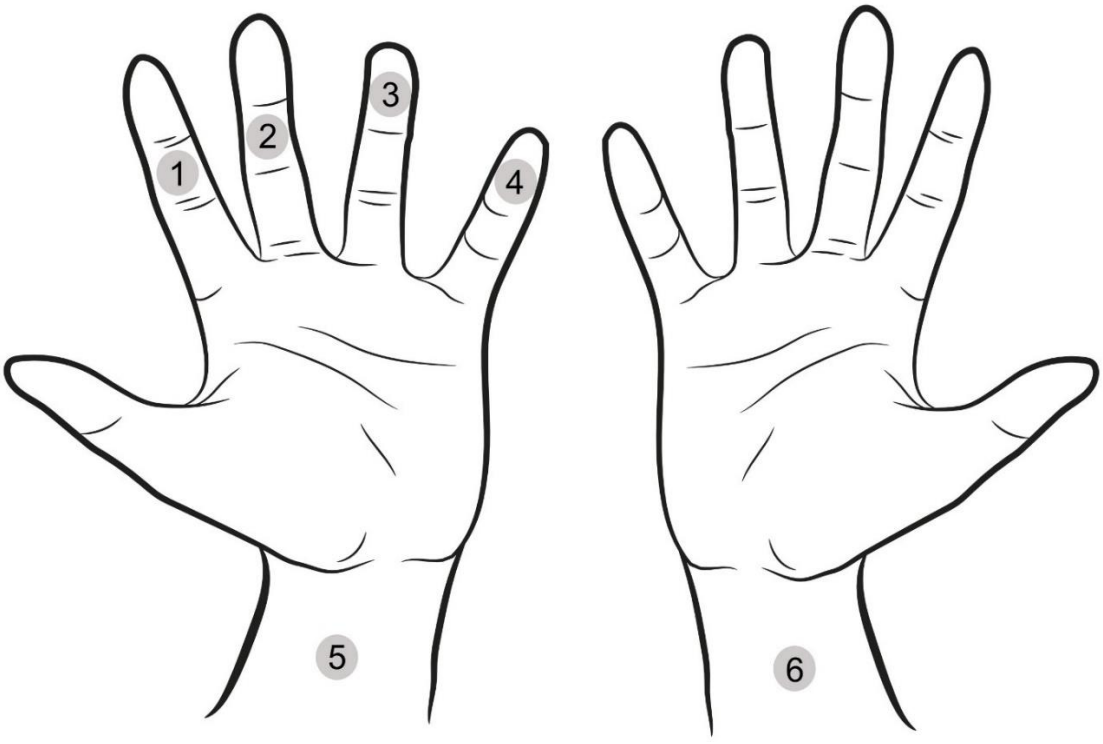


Figure 1: Electrode placement in left and right hand (For right-handed person)

3.1.4.1 Hand & Wrist Electrode Placement

To provide less intrusion of wires and electrodes, we first ask the participant about their dominant hand and place the majority of the electrode on their non-dominant hand. This gives them greater mobility during their VR sessions as they use

the controller to interact with the environment. Figure-1 represents the placement of electrodes for a Right-handed person. Below table explains number in Figure-1 and their relevant bio-signal channels:

Table 2: Hand electrode placement

Electrode Number from the image	Placement	Channel(c) or Device(d)
1	Middle segment of Index Finger	GSR (1)(c)
2	Middle segment of Middle finger	GSR (2)(c)
3	Tip of Ring Finger	Pulse Oximeter (d)
4	Tip of Pinky Finger	Thermistor (d)
5	The wrist of one of the hands (Close to ulnar artery)	ECG (1)(c)
6	The wrist of the other hand (Close to ulnar artery)	ECG (2)(c)

3.1.4.2 Face electrode Placement:

On the face, we want to track eye movement, and for the same, we rely on EMG. The placement for hEMG (Horizontal EMG) looks like the following image:

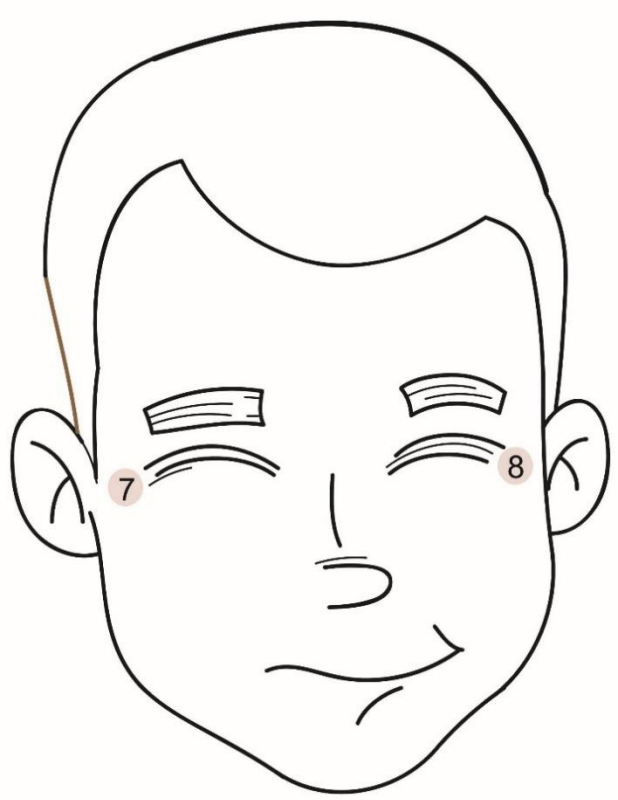


Figure 2: Electrode placement on the face

Below table represents the electrode's association with the appropriate channels which are obtained from the face.

Table 3: Head electrode placement

Electrode Number from the image	Placement	Channel
7	Front of left temporal muscle	hEOG (1)
8	Front of right temporal muscle	hEOG (2)

Figure 3: Bio-radio device

All the sensors are attached to the participant; we also mount Oculus VR headset; the entire setup looks like below image:

Figure 4: A veteran during a VR session with bio-radio attached.

3.1.5 Bio-signal Collection and storage process

We previously mentioned that bio-radio device allows us to store data locally on inbuilt memory or even stream it live via Bluetooth and store it to a computer, we preferred the later. Streaming the data live allows us to have a clear view of each channel in real time which allowed us to diagnose possible issues. On the computer,

we used ‘BioCapture Software’ which is also provided by ‘Great Lakes NeuroTechnologies’, the software comes as a package with the device and allows us to program and stream data from the BioRadio device. Moreover, the device also provides us with visualization capability which is shown in below figure:

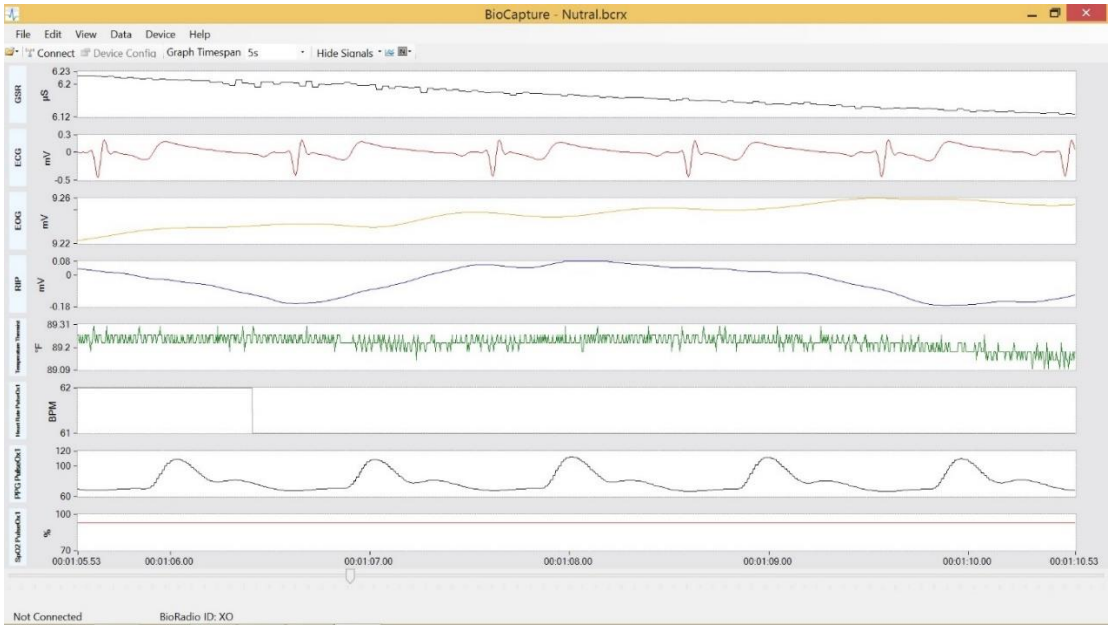


Figure 5: Screenshot of Bio-capture software

The BioCapture software stores each session in .bcrx files, which is a proprietary format; fortunately, it also allows us to export collected data in CSV format.

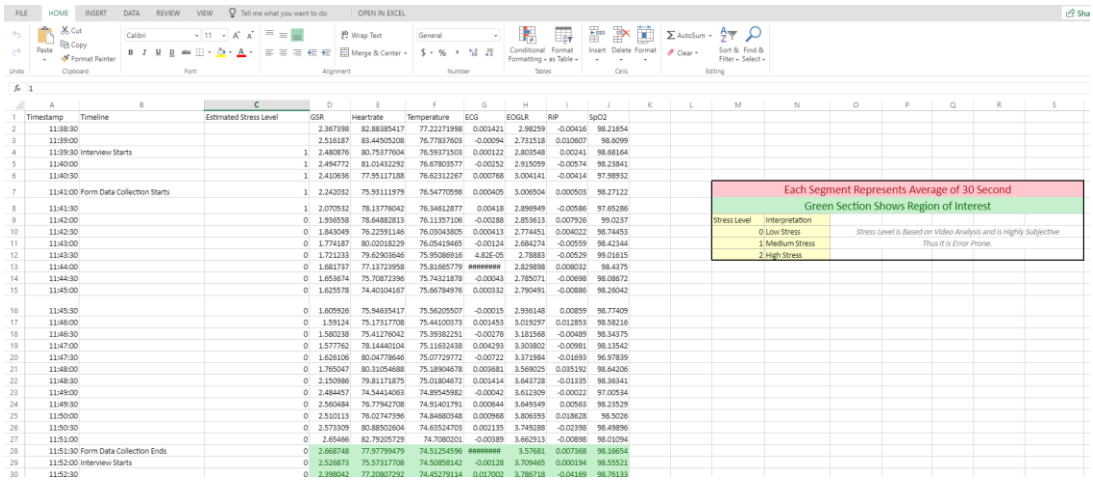
For each participant we took two data samples, the first sample named ‘Neutral’ was created to take appropriate baseline data for lower stress level. During the collection of this neutral file, participants were instructed to relax for 2-3 minutes. The second sample named “Interview” was created to record the rest of the entire session. At the end of all the data-collection sessions the session files were stored in a folder, and then another copy of the session was exported and stored in .csv format for further processing.

After conversion to .csv format, each of the data files was imported into MATLAB table and stored on the University workstation. MATLAB Version R2017b

was used for this analysis. Product details can be found on the MathWorks website [16].

3.1.6 Labeling

Since we tried to achieve supervised learning, we needed appropriate stress labels with the data. Unfortunately, BioCapture does not allow us to mark stress levels or additional details to the collected data. Moreover, getting the right stress labels is also a challenging task as stress is highly subjective. In our initial sessions, we planned to record the video of the entire session and planned to synchronize it with the data collected and mark appropriate labels. For the same, we also created analysis files which contained timestamps of each event. Screenshot of this analysis file is shown below:



Timestamp	Timeline	Estimated Stress Level	GSR	Heartrate	Temperature	ECG	EOGLR	RIP	SpO2
11:38:30			2.367398	82.88385417	77.22271998	0.001421	2.98259	-0.00416	98.21654
11:39:00			2.516187	83.44505208	76.77837603	-0.00094	2.731518	0.010607	98.6099
11:39:30	Interview Starts	1	2.480876	80.75377604	76.59371503	0.000122	2.805548	0.00241	98.68164
11:40:00		1	2.494772	81.01432292	76.67803577	-0.00252	2.915059	-0.00574	98.23841
11:40:30		1	2.410636	77.95117188	76.62312267	0.000768	3.004141	-0.00414	97.98932
11:41:00	Form Data Collection Starts	1	2.242032	75.93111979	76.54770598	0.000405	3.006504	0.000503	98.27122
11:41:30		1	2.070532	78.13776042	76.34612877	0.00418	2.890949	-0.00586	97.65286
11:42:00		0	1.936558	78.64882813	76.11357106	-0.00288	2.859613	0.007926	99.0237
11:42:30		0	1.843049	76.22591146	76.03043805	0.000413	2.774451	0.004022	98.74453
11:43:00		0	1.774187	80.02018229	76.05419465	-0.00124	2.684274	-0.00559	98.42344
11:43:30		0	1.721233	79.62903646	75.95086916	4.82E-05	2.78883	-0.00529	99.01615
11:44:00		0	1.681737	77.13723958	75.81665779	########	2.829898	0.008032	98.4375
11:44:30		0	1.653674	75.70872396	75.74321878	-0.00043	2.785071	-0.00698	98.08672
11:45:00		0	1.625578	74.40104167	75.66784976	0.000332	2.790491	-0.00886	98.28042
11:45:30		0	1.605926	75.94635417	75.56205507	-0.00015	2.936148	0.00859	98.77409
11:46:00		0	1.59124	75.17317708	75.44100373	0.001453	3.019297	0.012853	98.58216
11:46:30		0	1.580238	75.41276042	75.39382251	-0.00278	3.181568	-0.00489	98.34375
11:47:00		0	1.577762	78.14440104	75.11632438	0.004293	3.303802	-0.00981	98.13542
11:47:30		0	1.626106	80.04778646	75.07729772	-0.00722	3.371984	-0.01693	96.97839
11:48:00		0	1.765047	80.31054688	75.18904678	0.003681	3.569025	0.035192	98.64206
11:48:30		0	2.150986	79.81171875	75.01804672	0.001414	3.643728	-0.01335	98.36341
11:49:00		0	2.484457	74.54414063	74.89545982	-0.00042	3.612309	-0.00022	97.00534
11:49:30		0	2.560484	76.77942708	74.91401791	0.000644	3.649349	0.00503	98.23529
11:50:00		0	2.510113	76.02747396	74.84680348	0.000968	3.806393	0.018628	98.5026
11:50:30		0	2.573309	80.88502604	74.63524703	0.002135	3.749288	-0.02398	98.49896
11:51:00		0	2.65466	82.79205729	74.7080201	-0.00389	3.662913	-0.00898	98.01094
11:51:30	Form Data Collection Ends	0	2.668748	77.97799479	74.51254596	########	3.57681	0.007368	98.16654
11:52:00	Interview Starts	0	2.526873	75.57317708	74.50858142	-0.00128	3.709465	0.000194	98.55521
11:52:30		0	2.398042	77.20807292	74.45279114	0.017002	3.786718	-0.04169	98.78133

Each Segment Represents Average of 30 Second	
Green Section Shows Region of Interest	
Stress Level	Interpretation
0	Low Stress
1	Medium Stress
2	High Stress

Stress Level is Based on Video Analysis and is Highly Subjective
Thus It is Error Prone.

Figure 6: Screenshot of Data Summary

Regrettably, the collected video did not provide us with all the details necessary to generate accurate stress labels. These files have been kept in the university provided workstations for future use-case.

To provide a better and reliable solution, we developed a small utility which can store the input with stress level and timestamp during the interview. All the data used in this research work is the one we collected after switching to this application.

The application is created using Python and Qt (PyQt5) framework. Below is the screenshot of the utility:

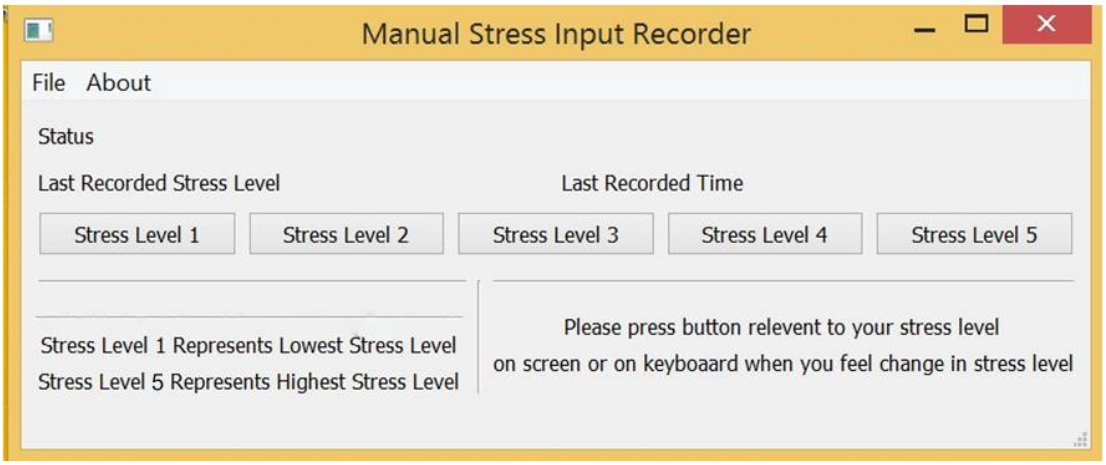


Figure 7: Screenshot of manual stress input recorder utility

Thanks to this small utility we were able to record stress markers during the interview by asking the participants for feedback and recording it. After each session, we used recorded data and stress input from this utility, and we were able to label the data and synchronize it accurately. As we can see from the screenshot, we recorded stress on a scale of 1 to 5 where 1 is the lowest stress and 5 being the highest stress. One thing to note here is that the label is still dependent on the feedback provided by the participant, and we cannot accurately determine how honest the input is.

3.1.7 Institutional Review Board

As this research involved human participants, review, and approval by the Texas State Intuitional Review Board was required.

IRB Application '**2017019'** was approved by the Institutional Review Board of Texas State University.

All participants have signed an informed consent form before the interview sessions. Participants were given information regarding the duration and content of each data collection session. Participants were reminded during the data collection

session that they are free to stop at any moment without repercussion or harm to the research or the reward. Participants were also told to be aware of the possibility of nausea due to VR usage and electrode removal process.

The collected dataset does not contain subject names or other known identifiable information. All data is stored on the Texas State TRACS workstations with controlled access by Dr. Vangelis Metsis.

3.2 Usage of DEAP Dataset

Apart from running and processing Veteran Dataset which was made by us, we also decided to use a publicly available dataset for reference and to compare our approach's ability.

DEAP Dataset contains data that have been collected from 32 subjects. The data were recorded as each subject watched 40 one-minute long excerpts of music videos. The data consist of 40 channels, 32 of them are EEG channels and the remaining 8 are EOG Horizontal and Vertical, EMG of Zygomaticus and Trapezius muscles, GSR, Respiration belt, Plethysmograph, and Temperature. The labels of these data are on a 1-9 scale for each class of the four valences, arousal, dominance, and liking. Out of all these channels we have used only peripheral channels due to the similarities to our dataset; we have also utilized arousal rating of the label as it is the most relevant to stress.

DEAP dataset is provided in .mat files with labels in the same format. To process it with our pipeline we have converted it .csv files for each subject. The provided data is downsampled at 128Hz but were collected at 512Hz. DEAP dataset also has an associated research paper in which they have achieved 57% accuracy, they have used same user's data in testing and training dataset; more detail regarding testing and training dataset can be found at section 4.3.

3.3 Feature Extraction

Feature extraction is the process of transforming the input data in a set of features. Features are distinguishing characteristic of the input pattern that can help in differentiating between categories of input data. In other words, features extraction converts our input data into information which is easier for the machine learning algorithms to classify.

In our initial experiments, we started with simple features like standard deviation, mean, etc. Then we switched to more advanced and signal-specific features like QRS complex detection from ECG or calculating the breathing rate. Due to data being noisy we did not succeed in this channel specific feature extraction.

A more complex set of features were extracted using feature extraction library with a set of 64 features; We will discuss the features extraction process and the challenges we faced during the process in the following section:

3.3.1 Initial experiments

In initial experiments, we tried to extract bio-signal specific features; we decided to extract the following features:

- **ECG:**

From ECG we extracted QRS complex, A combination of the Q wave, R wave and S wave, the “QRS complex” represents ventricular depolarization. It looks like the following image:

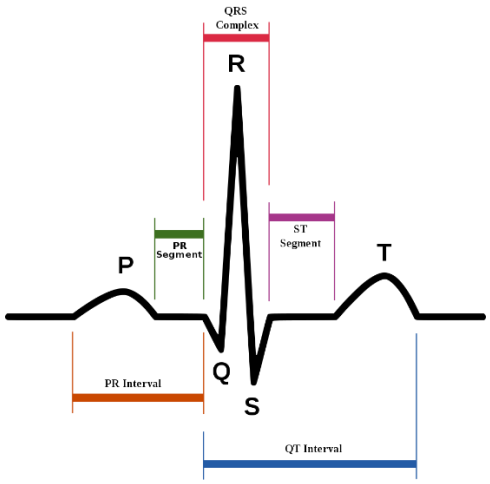


Figure 8: PQRST complex from ECG signal [17]

In medical science QRS complex analysis provides a vital role in patient diagnosis; heart activity and blood flow patterns can be learned from the QRS complex. We implemented the Pan-Tompkins [1] algorithm to detect these patterns and calculate RR-Interval, Heartbeat calculation, and RR peak detection.

- **EOG & RIP belt data:**

From EOG we tried to derive eye crossing rate and tried to detect rapid eye movement. For the Respiratory belt data, we tried to calculate breathing ration and zero crossing rate.

Our initial feature extraction efforts came to a halt due to inconsistency in the dataset. In medical procedures ECG is taken using 12-lead electrodes; in this process, electrodes are placed on to the chest which gives much more accurate and less noisy data [18]. In our data collection efforts, to try to keep the process non-invasive, we have collected ECG from the wrist as mentioned previously. This has caused data to be noisier than normal ECG, on top of that, we accumulated data while participants were constantly moving their hands and interacting with Oculus controller. All this led to more noisy data than expected and thus we quickly switched to python tsfresh library. All the extracted features using this library have been described in brief in the following section.

3.3.2 Final Feature Extraction

All the features which we have extracted from our dataset using python and tsfresh library are explained here in brief. The description for all this function can be found in more details on their website [19]. Feature extraction expects time series data with chunk size; it transforms this chunk of time series into more representable value for classification by applying mathematical functions. We have derived the following functions using tsfresh library, we have listed brief description of each feature calculator from library's documentation [19]:

- **Absolute Energy**

Computes the absolute energy of the time series chunk which can be given by following formula:

$$E = \sum_{i=1,\ldots,n} x_i^2$$

Where x is the time series chunk.

- **Absolute sum of changes**

Calculates the sum over the absolute value of consecutive changes in the chunk x of time series, this calculation can be mathemetically represented by the following equation:

$$\sum_{i=1,\ldots,n-1} \mid x_{i+1} - x_i \mid$$

- **Aggregation autocorrelation**

This function computes the value of an aggregation function f_{agg} which can be variance or mean, over the autocorrelation R(l) for different lags. The autocorrelation R(l) for lag l is defined as:

$$R(l) = \frac{1}{(n-l)\sigma^2} \sum_{t=1}^{n-l} (X_t - \mu)(X_{t+l} - \mu)$$

where X_i are the values of the time series chunk, n is its size of chunk. And, $(Sigma)^2$ and μ are estimators for its variance and mean.

All the autocorrelation R (1….max) are aggregated for different leg and sizes.

- **Approximate entropy**

  Computes Approximate entropy.

- **Autoregressive coefficient**

  This feature computes max probability of an autoregressive AR(k) process. k is the maximum lag of the process, mathematically it can be represented by:

$$X_t = \varphi_0 + \sum_{i=1}^{k} \varphi_i X_{t-i} + \varepsilon_t$$

For the configurations from parameter which should contain the maxlag "k" and such an AR process is calculated. Then the coefficients φ_i whose index i contained from "coeff" are returned.

- **Augmented Dickey fuller**

The Augmented Dickey-Fuller test is a hypothesis test which checks whether a unit root is present in a time series sample. This feature calculator returns the value of the corresponding test statistic.

- **Autocorrelation**

Calculates the autocorrelation of the specified lag, according to the formula:

$$\frac{1}{(n-l)\sigma^2} \sum_{t=1}^{n-l} (X_t - \mu)(X_{t+l} - \mu)$$

Where n is the length of the time series chunk X_i, μ its mean and σ^2 its variance. l denotes the lag.

- **Binned entropy**

First bins the values of x into max_bins equidistant bins. Then calculates the value of

$$-\sum_{k=0}^{min(max\_bins,len(x))} p_k log(p_k) \cdot \mathbf{1}_{(p_k>0)}$$

where p_k is the percentage of samples in bin k.

- **Measure of nonlinearity**

This function calculates the value of

$$\frac{1}{n-2lag}\sum_{i=0}^{n-2lag} x_{i+2\cdot lag}^2 \cdot x_{i+lag} \cdot x_i$$

which is

$$\mathbb{E}[L^2(X)^2 \cdot L(X) \cdot X]$$

Where $\mathbb{E}$ is mean and L is the lag operator. It was proposed in [20] as a measure of non-linearity in the time series.

- **Change quantiles**

It first creates a segment of quantiles ql and qh of the spread of x. Then it computes the absolute value, an average of continuous changes of the series x inside this segment.

Imagine selecting a partial segment on the y-Axis and only computing the mean of the absolute change of the time series inside this segment.

- **Efficient complexity-invariant distance for time series**

This computes an estimate for a time series complexity [21]. The calculation can be represented using the following formula:

$$\sqrt{\sum_{i=0}^{n-2lag} (x_i - x_{i+1})^2}$$

- **Count above mean**

Returns the number of values in a series chunk that are higher than the mean of that series chunk.

- **Count below mean**

Returns the number of values in series chunk which are lower than the mean of that series chunk.

- **Continuous wavelet transform of coefficients**

Computes Continuous wavelet transform for the Ricker wavelet, is can be mathematically defined by:

$$\frac{2}{\sqrt{3a}\pi^{\frac{1}{4}}}(1-\frac{x^2}{a^2})exp(-\frac{x^2}{2a^2})$$

Where a is the width parameter of the wavelet function.

- **Energy ratio by chunks**

Calculates the sum of squares of chunk I out of N chunks expressed as a ratio with the sum of squares over the whole series.

- **Fast Fourier transform aggregated**

Returns the spectral centroid (mean), variance, skew, and kurtosis of the absolute Fourier transform spectrum.

- **Fast Fourier transform coefficient**

Calculates the Fourier coefficients of the one-dimensional discrete Fourier Transform for real input by FFT algorithm, which can be mathematically represented by:

$$A_k = \sum_{m=0}^{n-1} a_m \exp\left\{-2\pi i \frac{mk}{n}\right\}, \qquad k = 0, \ldots, n-1.$$

The resulting values will be complex number.

- **The first location of maximum**

Returns the first location of the maximum value of a time series chunk. The position is calculated relative to the length of individual chunk.

- **The first location of minimum**

Returns the first location of the minimum value of a time series chunk. The position is represented relative to the length of individual chunk.

- **Friedrich coefficients**

Coefficients of polynomial $h(x)$, which has been fitted to the deterministic dynamics of Langevin model

$$\dot{x}(t) = h(x(t)) + \mathcal{N}(0, R)$$

as described by [22].

- **Has duplicate**

Checks if any value in time series chunk occurs more than once.

- **Has a duplicate of the maximum value**

Checks if the maximum value of time series chunk is observed more than once.

- **Has a duplicate of the minimum value**

Checks if the minimal value of time series chunk is observed more than once.

- **Index mass quantile**

These feature computes the relative index i where q% of the mass of the time series chunk lie left of i. For example, for q = 50% this feature calculator will return the mass center of the time series.

- **Large standard deviation**

Computes a Boolean variable representing if the standard deviation of time series chunk(x) is higher than ‘r’ times the range = difference between max and min of x(chunk). Hence it checks if

$$std(x) > r * (max(X) - min(X))$$

- **Last location of maximum**

Returns the relative previous location of the maximum value of within chunk time series chunk(x). The position is calculated relative to the length of that chunk.

- **Last location of minimum**

Returns the previous location of the minimal value of time series chunk(x). The position is calculated relative to the length of time series chunk(x).

- **Length**

Returns length of time series chunk(x).

- **Linear trend**

Calculate a linear least-squares regression for the values of the time series versus the sequence from 0 to length of the time series minus one. This feature assumes the signal to be uniformly sampled. It will not use the timestamps to fit the model. The parameters control which of the characteristics are returned.

Possible extracted attributes are “pvalue”, “rvalue”, “intercept”, “slope”, “stderr”.

- **Longest strike above mean**

Computes the length of the longest consecutive subsequence in chunk x that is bigger than the mean of x.

- **Longest strike below mean**

Computes the length of the longest consecutive subsequence in chunk x that is smaller than the mean of x.

- **Max langevin fixed point**

Largest fixed point of dynamics: math: argmax_x {h(x)=0}` estimated from polynomial $h(x)$, which has been fitted to the deterministic dynamics of Langevin model

$$\dot{(}x)(t) = h(x(t)) + R(N)(0,1)$$

as described by [22].

- **Maximum**

Calculates the highest value of the time series x.

- **Mean**

Returns the mean of x.

- **Mean absolute change**

Returns the mean over the absolute differences between subsequent time series values which is

$$\frac{1}{n} \sum_{i=1,\ldots,n-1} |x_{i+1} - x_i|$$

- **Mean change**

Returns the mean over the absolute differences between following time series values which is

$$\frac{1}{n} \sum_{i=1,\ldots,n-1} x_{i+1} - x_i$$

- **Mean second derivative of central approximation**

Returns the mean value of a central approximation of the second derivative

$$\frac{1}{n} \sum_{i=1,\ldots,n-1} \frac{1}{2}(x_{i+2} - 2 \cdot x_{i+1} + x_i)$$

- **Median**

  Computes the median of time series chunk.

- **Minimum**

  Finds the lowest value of the time series chunk.

- **Number of crossing value**

Calculates the number of crossings of x on m. A crossing is defined as two sequential values where the first value is lower than m and the next is greater, or vice-versa. If you set m to zero, you will get the number of zero crossings.

- **Number of cwt peaks**

This feature calculator searches for different peaks in x. To do so, x is smoothed by a ricker wavelet and for widths ranging from 1 to n. This feature calculator returns the number of peaks that occur at enough width scales and with sufficiently high Signal-to-Noise-Ratio (SNR).

- **Number of peaks**

Calculates the number of peaks of at least support n in the time series x. A peak of support n is defined as a subsequence of x where a value occurs, which is bigger than its n neighbors to the left and to the right.

- **Partial autocorrelation**

Calculates the value of the partial autocorrelation function at the given lag. The lag *k* partial autocorrelation of a time series $\{x_t, t = 1 \ldots T\}$ equals the partial correlation of x_t and x_{t-k}, adjusted for the intermediate variables $\{x_{t-1}, \ldots, x_{t-k+1}\}$([23]). Following [24], it can be defined as:

$$\alpha_k = \frac{Cov(x_t, x_{t-k} | x_{t-1}, \ldots, x_{t-k+1})}{\sqrt{Var(x_t | x_{t-1}, \ldots, x_{t-k+1}) Var(x_{t-k} | x_{t-1}, \ldots, x_{t-k+1})}}$$

With $x_{t-k} = f(x_{t-1}, \ldots, x_{t-k+1})$ being AR(k-1) models that can be fitted by OLS. Be aware that in (a), the regression is done on past values to predict x_t whereas in (b), future values are used to calculate the past value x_{t-k}. It is said in [23] that "for an AR(p), the partial autocorrelations [α_k] will be nonzero for k<=p and zero for k>p." With this property, it is used to determine the lag of an AR-Process.

- **Percentage of reoccurring datapoints to all data points**

Returns the percentage of unique values, that are present in the time series more than once.

len(different values occurring more than once) / len(different values)

This means the percentage is normalized to the number of unique values, in contrast to the 'percentage of reoccurring values to all values'.

- **Percentage of reoccurring values to all values**

Returns the ratio of unique values, that are present in the time series more than once.

num of data points occurring more than once / num of all data points

This means the ratio is normalized to the number of data points in the time series, in contrast to the 'percentage of reoccurring datapoints to all data points'.

- **Quantile**

Calculates the q quantile of x. This is the value of x greater than q% of the ordered values from x.

- **Range Count**

Count observed values within the interval (min, max).

- **Ratio beyond r sigma**

The ratio of values that are more than r*std(x) (so r sigma) away from the mean of x.

- **The ratio of value number to time series length**

Returns a factor which is 1 if all values in the time series occur only once, and below one if this is not the case. In principle, it just returns

Num of unique values/num of values

- **Sample entropy**

Calculate and return sample entropy of time series.

- **Skewness**

Calculates the sample skewness of x (calculated with the adjusted Fisher-Pearson standardized moment coefficient G1).

- **Spectral density**

Computes estimates of the cross power spectral density of the time series chunk at diverse frequencies. In order to achieve this, the time series chunk is first converted from the time domain to the frequency domain.

- **Standard deviation**

Returns the standard deviation of x.

- **Sum of reoccurring data points**

Sum of all the non-unique datapoints in the entire chunk.

- **Sum of reoccurring values**

Calculates the sum of all values, that are repeating in the present in the time series chunk.

- **Sum of values**

Computes the sum of the time series chunk values

- **Symmetry looking**

Boolean variable denoting if the distribution of x *looks symmetric*. This is the case if

$$|mean(X) - median(X)| < r * (max(X) - min(X))$$

Time reversal asymmetry statistic. This function calculates the value of

$$\frac{1}{n - 2lag} \sum_{i=0}^{n-2lag} x_{i+2 \cdot lag}^2 \cdot x_{i+lag} - x_{i+lag} \cdot x_i^2$$

which is

$$\mathbb{E}[L^2(X)^2 \cdot L(X) - L(X) \cdot X^2]$$

Where $\mathbb{E}$ is the mean and L is the lag operator. It was proposed in [25] as a promising feature to extract from time series.

- **Value count**

Count occurrences of value in time series chunk.

- **Variance**

Returns the variance of time series chunk.

- **Variance larger than the standard deviation**

Boolean calculator representing if the variance of time series chunk is greater than its standard deviation.

3.4 Feature Selection

Feature selection is also called variable selection or attribute selection. In other words, it is the automatic selection of attributes from the feature set that is most relevant to the predictive modeling problem at hand. Feature selection aids the process of creating a more accurate predictive model. It can be used to remove irrelevant and redundant attributes from the feature set. “The objective of variable selection is three-fold: improving the prediction performance of the predictors,

providing faster and more cost-effective predictors, and providing a better understanding of the underlying process that generated the data." [26]

Feature selection is an essential technique to improve accuracy for machine learning classifier. As we discussed, extracted features in the previous section, it is not necessary to keep all of the features as they might not be directly relevant for machine learning task. For the same, we have used FRESH (FeatuRe Extraction based on Scalable Hypothesis tests) algorithm [27].

It is efficient and scalable feature extraction and selection algorithm; it helps in filtering the extracted features in an earlier stage of the machine learning pipeline with respect to their significance on bases of the p-value for the classification or regression task in hand while controlling the selection of necessary features.

The FRESH algorithm uses the Mann-Whitney U test for calculating feature significance vector for each feature. The result of this processes is a vector of p-values which quantifies significance for each feature for predicting the label.

The FRESH algorithm next uses Benjamini-Yekutieli procedure [28] to decide which features are most important. For this the test regards the features p-values and controls the global irrevalent feature discovery rate, which is the ratio of false rejections by all rejections:

$$FDR = \mathbb{E}\left[\frac{|\text{false rejections}|}{|\text{all rejections}|}\right]$$

We can tweak this FDR perimeter to keep a threshold to increase or decrease selected features. The entire process of FRESH algorithm can be summarized as per the following image:

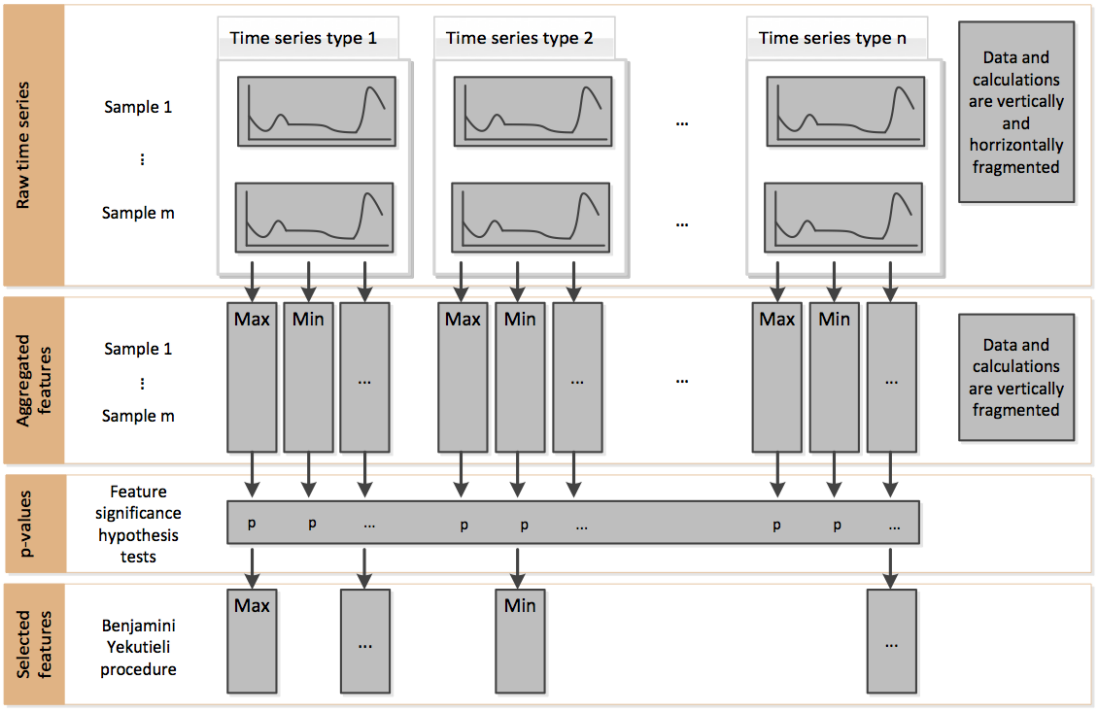


Figure 9: FRESH Feature extraction and selection process [27]

When we use FRESH algorithm the dimension of the dataset is reduced vertically as each chunk time series will result in one representable value, but the size increases horizontally as each extracted feature represents one column of data.

4. EXPERIMENTS AND RESULTS

We have processed DEAP and Veteran dataset in a similar pipeline. Thus common processes related to both datasets are explained here in details.

4.1 Updating the labels for binary classification

DEAP and Veteran Datasets both have labels in different scales, out aim here to classify Stress or Arousal in high or low level. For the same, we have updated the labeling scheme between '0' and '1'. In this, '0' represents low stress while '1' represents high stress. In veteran dataset stress value between '3-5' is considered as high-stress value and 1-2 is regarded as low-stress values. In DEAP dataset the arousal level has been recorded in 0-9 scale, all the value larger than four is considered as a high-stress label while lower than four is regarded as low-stress values.

Hence, for both the datasets we will be performing binary classification.

4.2 Cross-Validation

To overcome overfitting problem, we have divided our dataset in test-train split. We have used Leave-one-out cross validation method. Before we feed any data to our machine learning classifier we have to extract and select same features from all the different splits of data to maintain consistency. All the result presented in the result section are from the leave-one-out method.

4.3 Result calculations & Confusion matrices

We have split our data in test & train datasets; the algorithm is trained on the training dataset. After the training, we feed the test portion of the data to trained machine learning model and generate the predicting labels. We compare the results of predicted labels and the actual labels of the test chunk and calculate the accuracy as per the following formula:

accuracy = (correctly predicted class / total testing class) × 100%

In more detail, the predicted labels can be divided into four classes by comparing with the actual test labels as follow:

True positive (TP) = the number of instances correctly identified as true.

False positive (FP) = the number of instances incorrectly identified as true.

True negative (TN) = the number of instances correctly identified as false.

False negative (FN) = the number of instances incorrectly identified as false.

From all these measures we can calculate accuracy, precision, recall and F1 score as per the following formulas:

Accuracy= (TP+TN)/(TP+TN+FP+FN)

Precision = (TP)/(TP+FP)

Recall = (TP)/(TP+FN)

F1 Score = 2*((Precision * Recall)/ (Precision + Recall))

Support = Number of samples to support a class of labels (In our case, zero and one in actual test class labels).

To represent our results in the graphical form, we have used standard confusion matrix. In the standard confusion matrix, these four labels are represented in percentage form (normalized confusion matrix) as follows:

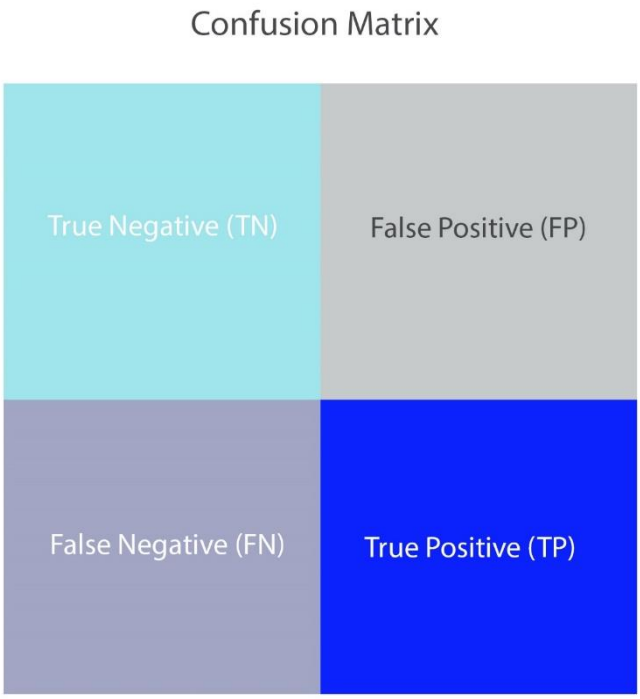


Figure 10: Representation of results as Confusion Matrix

4.4 Veteran Dataset Results and Experiments

After extracting and selecting features from raw data as explained in section 3.3 & 3.4 we normalize the range of feature values between 0 and 1. The transformation for this can be given by:

$$X\_std = (X - X.min(axis=0)) / (X.max(axis=0) - X.min(axis=0))$$

$$X\_scaled = X\_std * (max - min) + min$$

where min and max are feature range.

4.4.1 List of selected features

After feature extraction and selection process is done which is explained in 3.3 and 3.4 section, we have gathered our training dataset. We have used FRESH algorithm [19] which considers features listed in following table to be most important for classification, we will be using only these features for machine learning task:

Table 4: List of selected features from veteran dataset

Channel	**Feature selected**
PPG	Aggregate linear trend, Quantile Variance, Standard deviation, Linear trend, Fast Fourier transform, Minimum, Change quantiles
RIP	Autoregressive coefficient
GSR	Efficient complexity-invariant distance for time series, Change quantiles

In total the algorithm picks 32 features using different parameters from the above table at FDR value of 1e-20.

4.4.2 Classifier and Results

We have implemented all the classifiers using scikit-learn library. We have also listed the parameters used with the results. More information regarding model's and parameters can be found at 'https://scikit-learn.org' [29]. Baseline for the classifier will be 50% as we are performing binary classification.

- **Support vector machine:**

We have trained Support Vector Machine classifier with the following parameter: Kernal: ‘RBF’, C: 1 and Gamma: 20

The confusion matrix for this classifies can be seen below:

	precision	recall	f1-score	support
0	0.80	0.53	0.64	124
1	0.62	0.85	0.72	109
micro avg	0.68	0.68	0.68	233
macro avg	0.71	0.69	0.68	233
weighted avg	0.72	0.68	0.68	233

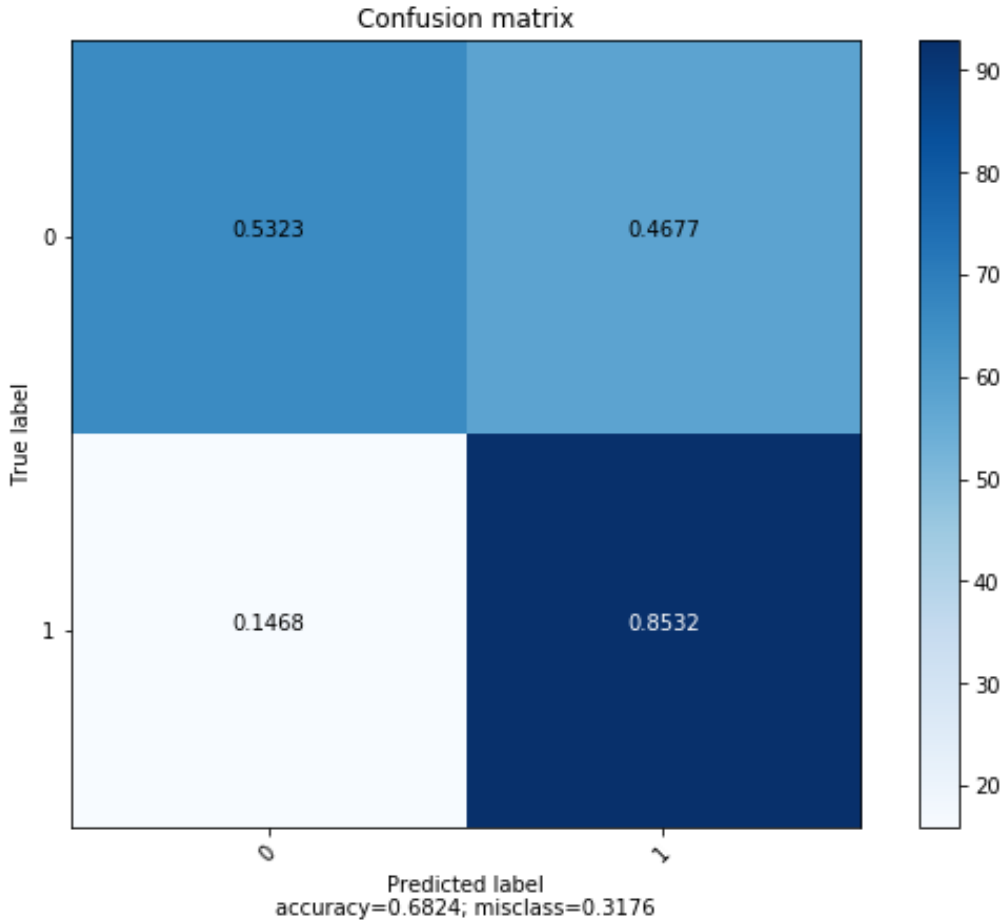


Figure 11: Confusion matrix for SVM on Veteran dataset

- **Random Forest**

We have trained Random Forest using these parameters, N_estimators: 400, max_depth: 20, min_samples_leaf: 40.

The confusion matrix for this classifies can be seen below:

	precision	recall	f1-score	support
0	0.68	0.77	0.72	150
1	0.70	0.60	0.65	135
micro avg	0.69	0.69	0.69	285
macro avg	0.69	0.69	0.69	285
weighted avg	0.69	0.69	0.69	285

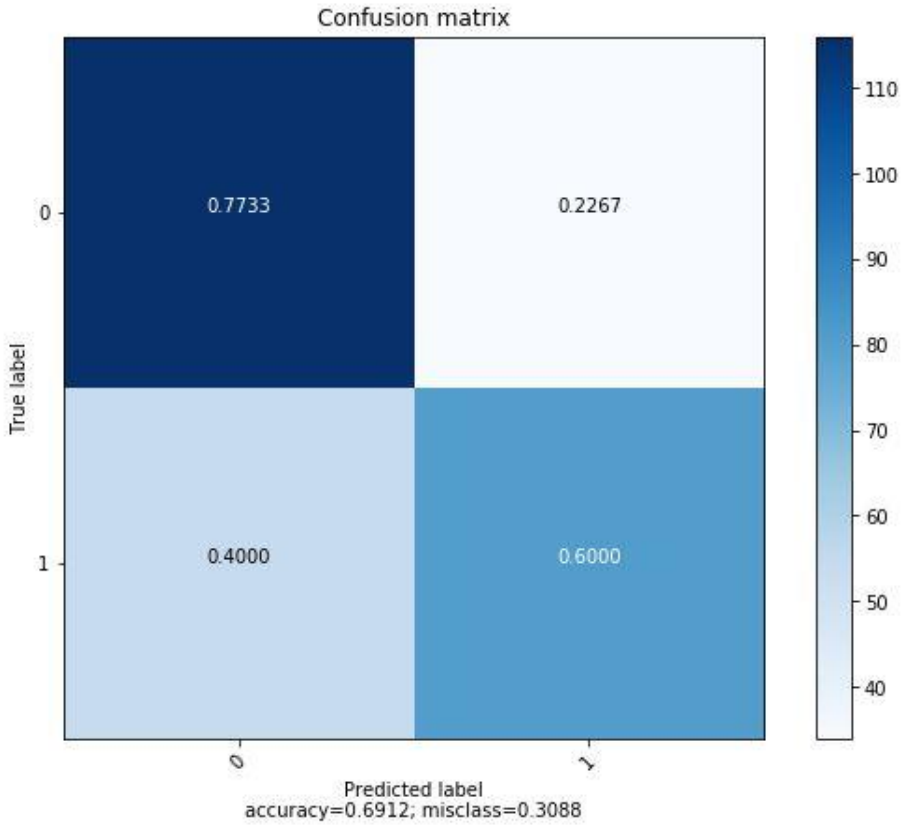


Figure 12: Confusion matrix of Random Forest on Veteran dataset

- **K-Nearest Neighbors algorithm**

We have trained K-Nearest Neighbors using the following parameters: n_neighbors:10, P: 1.

The confusion matrix for this classifies can be seen below:

	precision	recall	f1-score	support
0	0.78	0.63	0.69	206
1	0.65	0.80	0.72	183
micro avg	0.71	0.71	0.71	389
macro avg	0.72	0.71	0.71	389
weighted avg	0.72	0.71	0.71	389

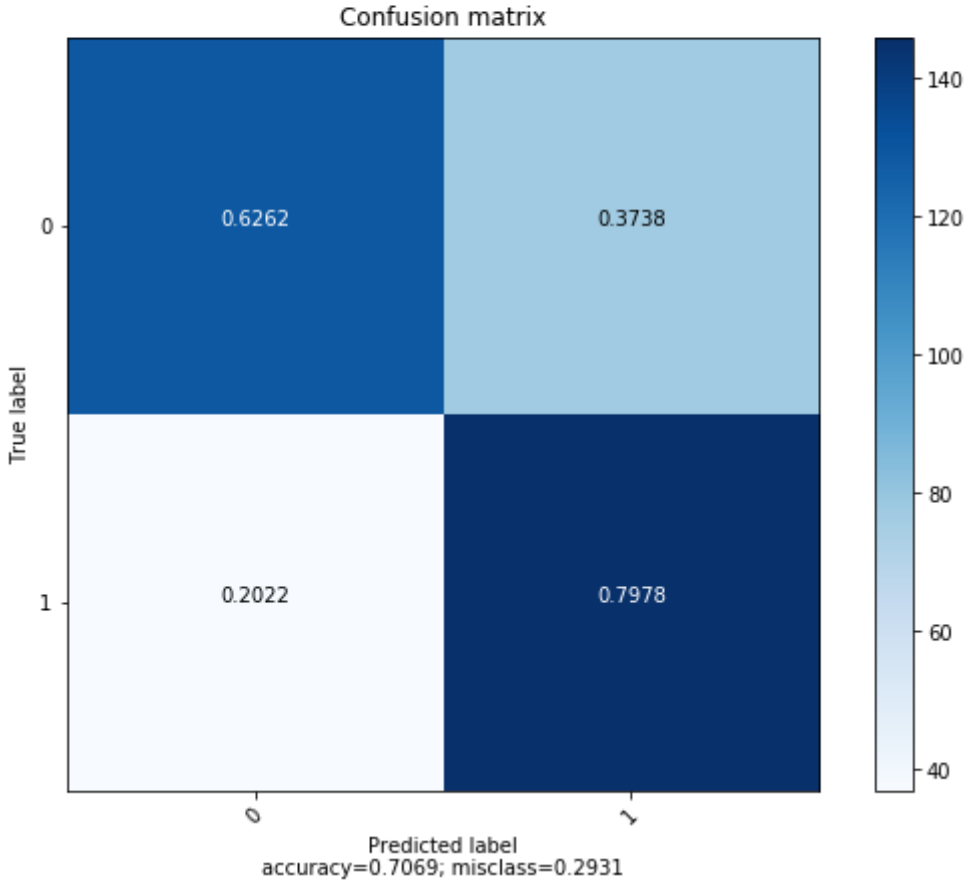


Figure 13: Confusion matrix of KNN on Veteran dataset

- **LSTM Neural Network**

We have implemented a simple LSTM neural network using Keras Library. The network consists of two dense LSTM layers which contain fifty cells. In order to avoid overfitting, we have also introduced dropout layers after each dense layer at drop-out ratio of 0.3. We have used sigmoid as the activation layer. The network was ran on 60 epochs with batch size of 512. The architecture of the network can be represented as per the following graph:

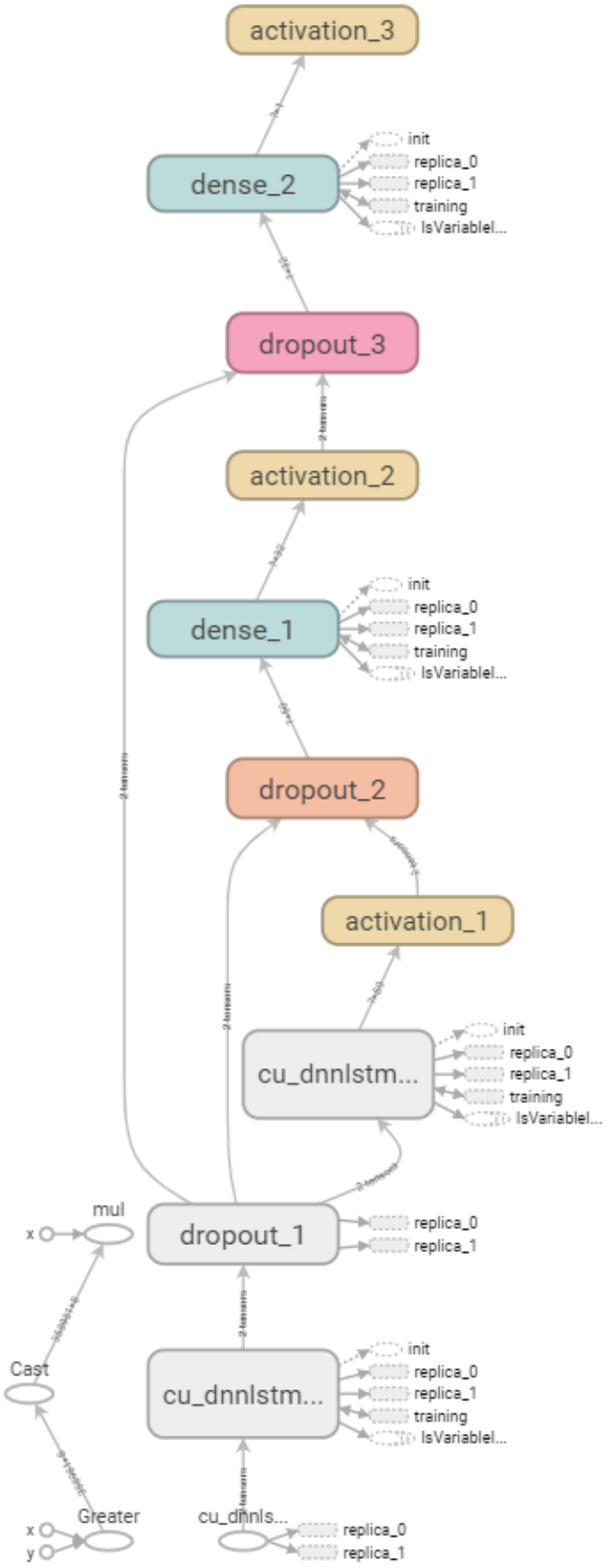


Figure 14: LSTM Network Diagram

More details regarding the parameters and layers can be found at keras website ‘www. keras.io’.

The resulting confusion matrix of this network can be seen below:

	precision	recall	f1-score	support
0	0.24	0.48	0.32	6667
1	0.89	0.73	0.80	38192
micro avg	0.70	0.70	0.70	44859
macro avg	0.57	0.61	0.56	44859
weighted avg	0.79	0.70	0.73	44859

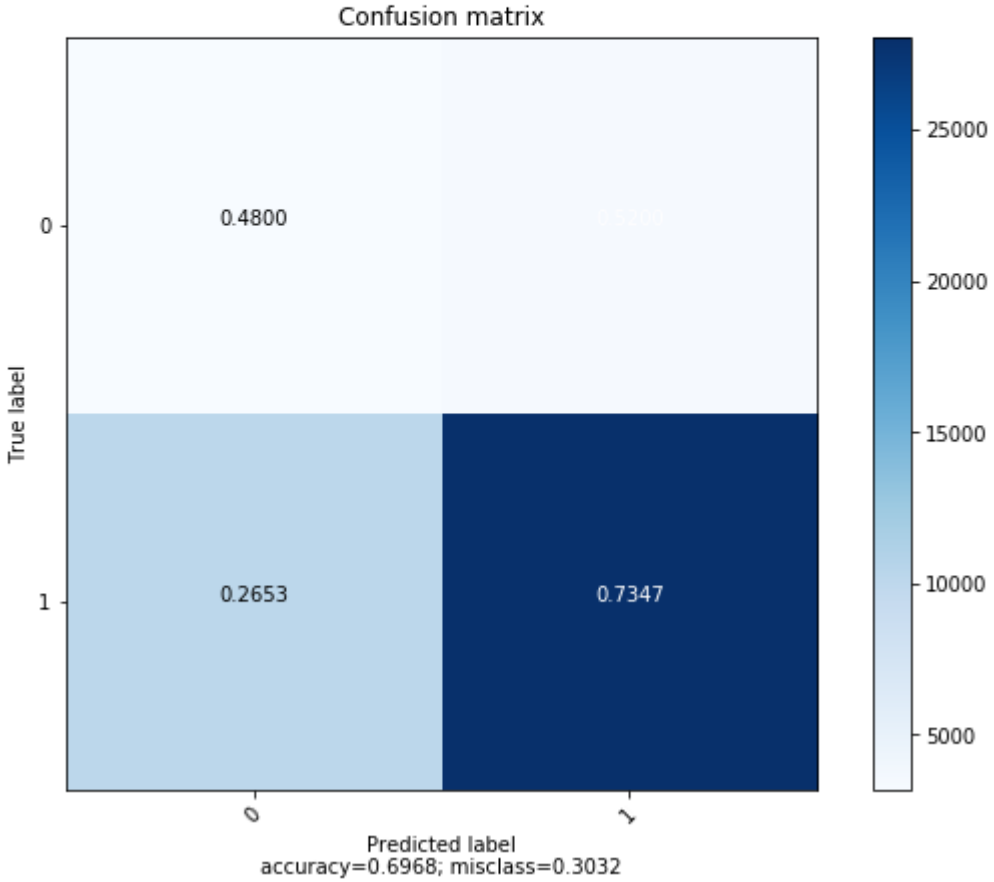


Figure 15: Confusion matrix of LSTM on Veteran dataset

- **Naive Bayes**

We have trained Naïve Bayes classifier by setting alpha value to 20.

The confusion matrix for this classifier can be seen in following image:

	precision	recall	f1-score	support
0	0.70	0.63	0.66	206
1	0.63	0.69	0.66	183
micro avg	0.66	0.66	0.66	389
macro avg	0.66	0.66	0.66	389
weighted avg	0.66	0.66	0.66	389

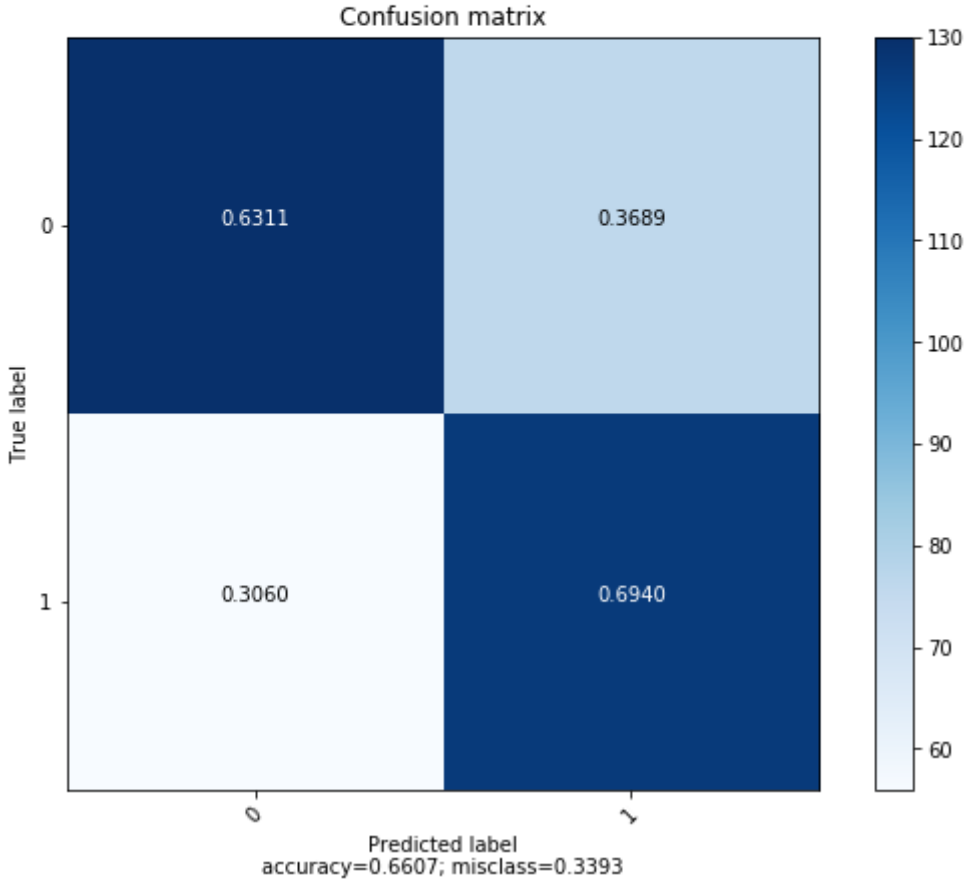


Figure 16: Confusion matrix of Naive Bayes on Veteran dataset

4.4.3 Independent Bio-signal Channel Analysis

We analyzed each bio-signal channel with the same feature extraction and selection method to see which bio-signal individually a better indicator for stress classification is. We have extracted and selected features for each individual Bio-signal channel. We have used FDR values which gives around 10-20 features per bio-signal, except for the SpO2 signal which results in less than 5 relevant features due to its stagnant value.

The value of FDR values are picked by the brute force method to find feature set from around 10-25 per channel.

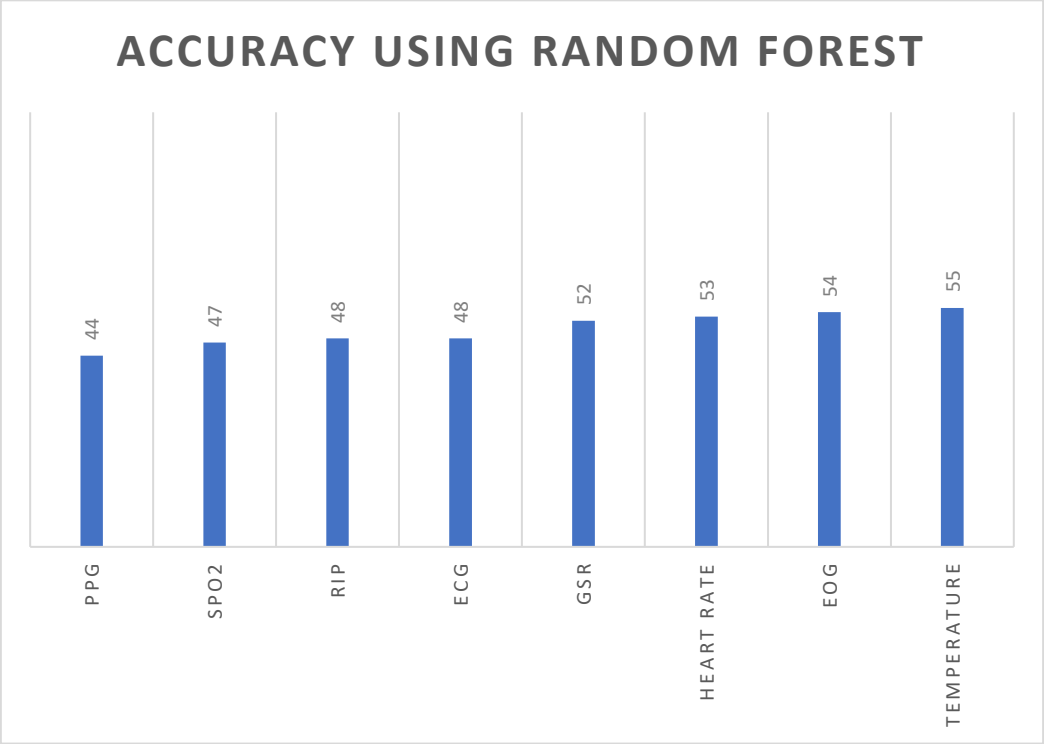


Figure 17: Independent bio-signal analysis

The above chart represents all the individual channels accuracy using Random Forest classifier at number of estimators count of 200 and a maximum depth of 30. We have picked Random forest classifier due to its reliable results in both the datasets and its ability to parallelize which results in faster execution. Random forest also allows us to gain reliable results without aggressive tuning. As we mentioned previously the features selection process has selected 10-20 features per signal, the list of all the selected features without parameters can be found in the following tables:

Table 5: List of selected features from each physiological signal in Veteran dataset (1)

GSR	**ECG**	**PPG**	**EOG**
Change quantiles	Ratio beyond r sigma	Aggregated linear trend	Change quantiles
Variance	Minimum	Variance	
Aggregated linear trend	Change quantiles	Fast fourier transformation	
Aggregated linear trend	Approximate entropy	Change quantiles	
	Large standard deviation		
	Aggregated linear trend		
	Autoregressive Coefficient		

Table 6: List of selected features from each physiological signal in Veteran dataset (2)

RIP	**Temperature**	**Heart Rate**	**SpO2**
Autoregressive Coefficient	Aggregated linear trend	Minimum	Autoregressive Coefficient
	Linear Trend	Sum of reoccurring data points	
	Partial autocorrelation	Sum values	
	FFT Coefficient	FFT Coefficient	
		Median	
		Continuous wavelet transform	

In the above table we have only written unique features without listing the parameters, as previously mentioned we have collected 10 to 20 features per channel, but some of them are same feature selectors at different parameters.

4.4.4 Combination of bio-signals

We have selected four best performing bio-signals from the individual bio-signal analysis. We have tried to train our Random forest classifier with the same parameters as in section 4.4.2; on best four channels, the best two channels and rest of the channels from the best four channels. We have also added the highest recurring selected features channels from section 4.5.2. The results of accuracy from them can be found in the following image:

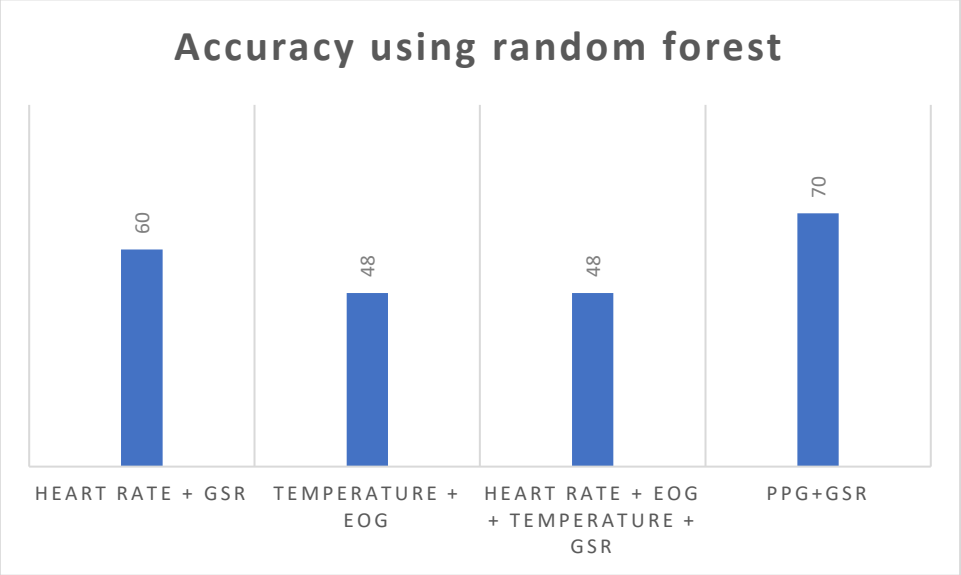


Figure 18: Combination bio-signal analysis

The result above shows that the feature selection method using p-values is providing better selection of features. Each individual bio-signal on its own cannot correctly represent the stress level for classification, but the Bio-signal channels picked by the algorithm during previous experiments seems to be proving a better accuracy at classification than selecting the highest performing individual channels.

4.4.5 Summary of Veteran Dataset Results

We have achieved around 66-71% accuracy on this dataset using our feature selection and extraction methods. The result from all the classifier can be seen in below chart:

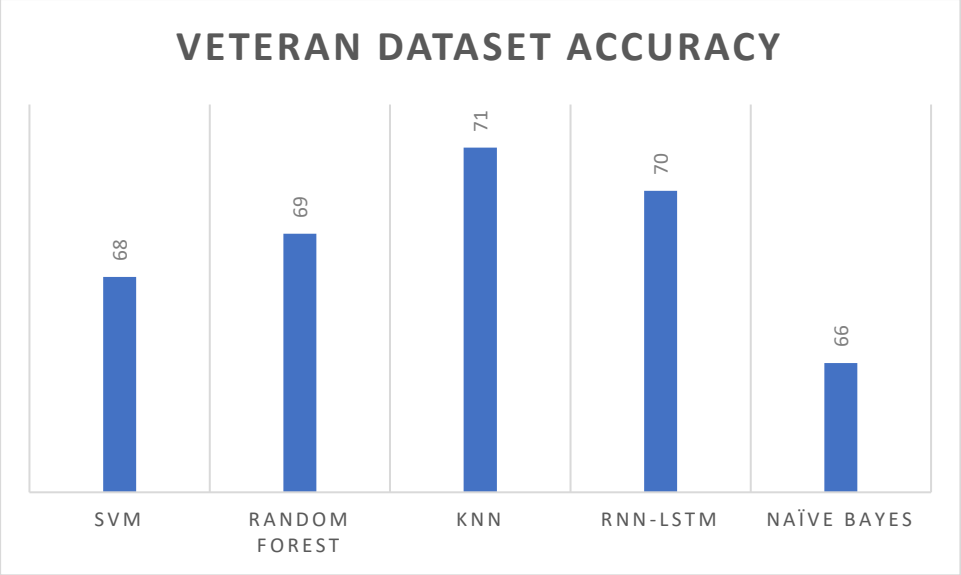


Figure 19: Summary of Veteran dataset results

In our experiment we also observed that Blood oxygen level (SpO2) is not a great bio-signal for classifying emotion, as its value doesn't seem to get affected by human stress during our experiments. Blood volume (PPG) and Galvanic skin response (GST) have proven to be highly effective physiological-signals for classifying stress in these experiments.

4.5 DEAP Dataset

4.5.1 Data visualization & Preprocessing

Before we further process data, let's visualize the '.csv' Files we created with DEAP dataset. In the following figure, we have created a combined file of all the DEAP users and visualized it after removing outliers.

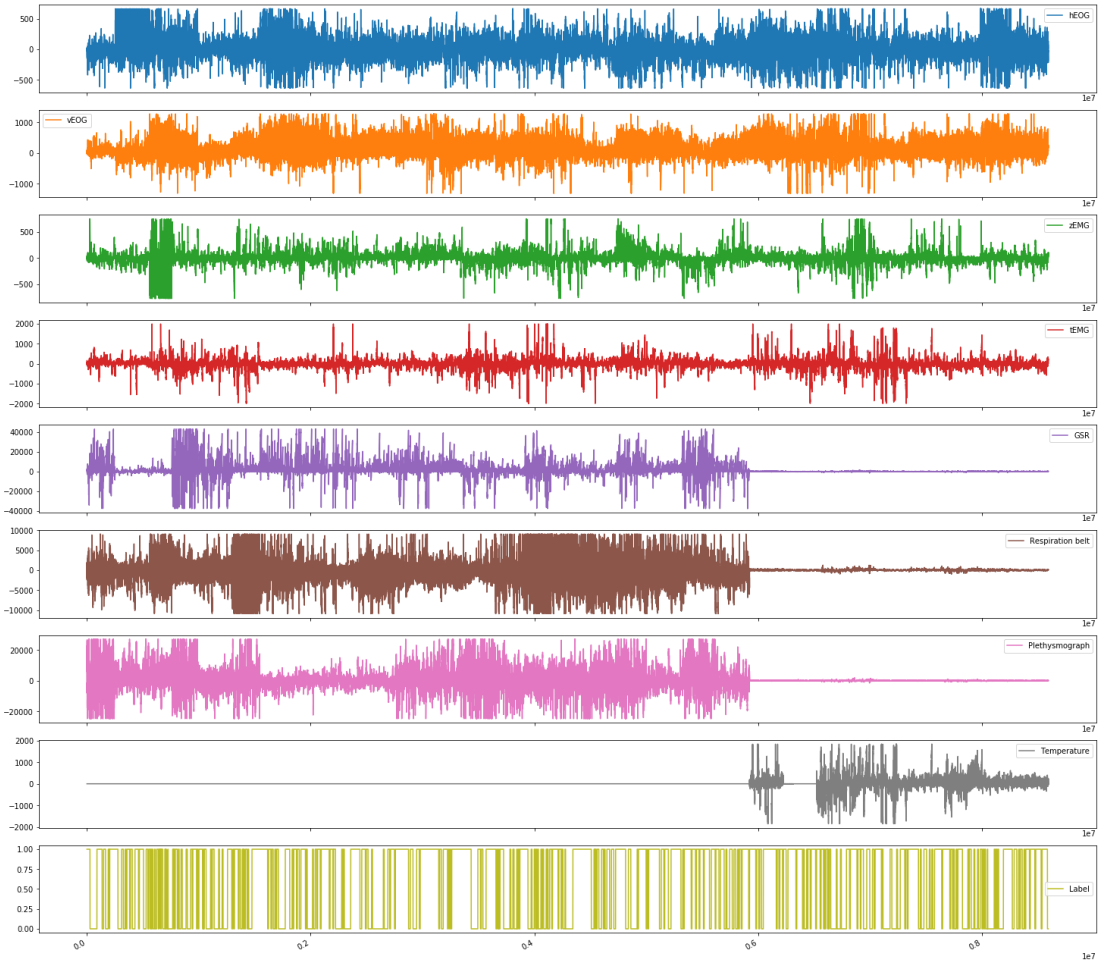


Figure 20: DEAP data analysis

We have used DEAP dataset to test our method and for result reference, DEAP research team also has published research paper in which they have gained 57% accuracy using Naïve Bayes classifier [30]. This dataset suffers from class imbalance [30] which is mentioned in the cited paper and can be seen in following chart:

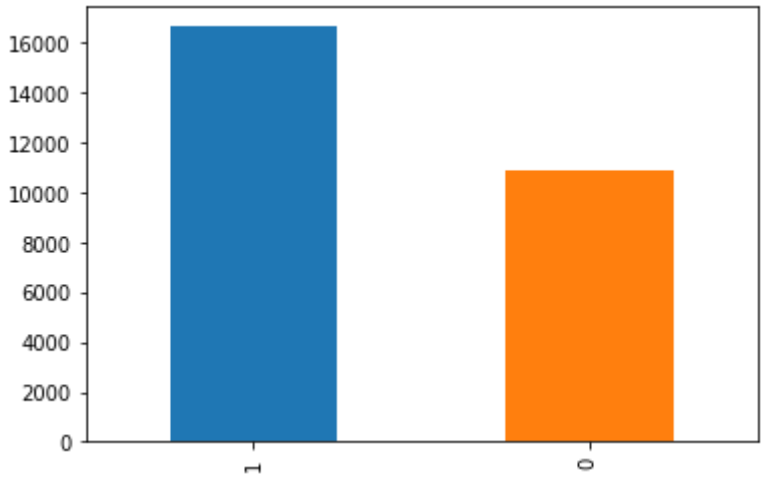


Figure 21: DEAP class balance

After removing outliers and scaling the database between 0 and 1, we feed the data to machine learning classifier in a similar fashion as previously mentioned.

4.5.2 List of selected features

After feature extraction and selection which is described in sections 3.3 and 3.4; FRESH algorithm [19] considers features listed in following table to be most important for classification. In total it has derived 40 features with different parameter values for each feature. we will be using only these features for machine learning task:

Table 7: List of selected features from DEAP dataset

Channel	Feature selected
Zygomaticus EMG	Autoregressive coefficient, Partial autocorrelation, change quantiles, mean absolute change, Number of peaks, fast fourier transform aggregated, approximate entropy
Horizontal EOG	Change quantiles, Absolute sum of changes, Mean absolute change

In total the algorithm picks 40 features using different features from the features listed in the above table at FDR parameter set at 1e-50. We have run the experiments with a greater number of features as well, but the results seem to provide

minor changes, while performance improvement we gain with a reduced number of feature set is significant.

4.5.3 Classifier and results

- Support vector machine

We have trained Support Vector Machine classifier with the following parameter: Kernel: ‘RBF’, C: 5, Gamma: 20. The result can be seen below:

	precision	recall	f1-score	support
0	0.48	0.63	0.55	3974
1	0.70	0.57	0.62	6099
micro avg	0.59	0.59	0.59	10073
macro avg	0.59	0.60	0.59	10073
weighted avg	0.61	0.59	0.59	10073

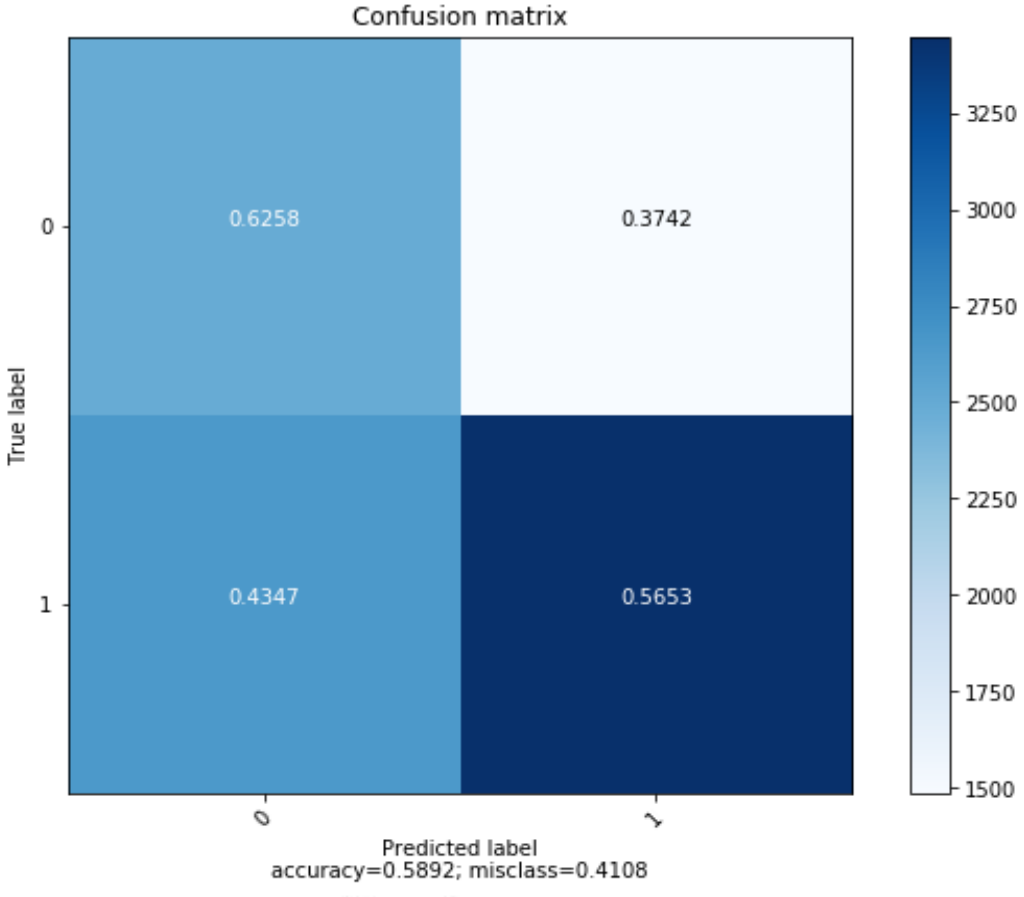


Figure 22: Confusion matrix of SVM on DEAP dataset

- **Random Forest**

We have used Random Forest classifier with following parameters: N_estimator : 100, max_depth : 100, min_samples_leaf : 40. Below is the resulting confusion matrix:

	precision	recall	f1-score	support
0	0.52	0.53	0.53	3974
1	0.69	0.69	0.69	6099
micro avg	0.63	0.63	0.63	10073
macro avg	0.61	0.61	0.61	10073
weighted avg	0.63	0.63	0.63	10073

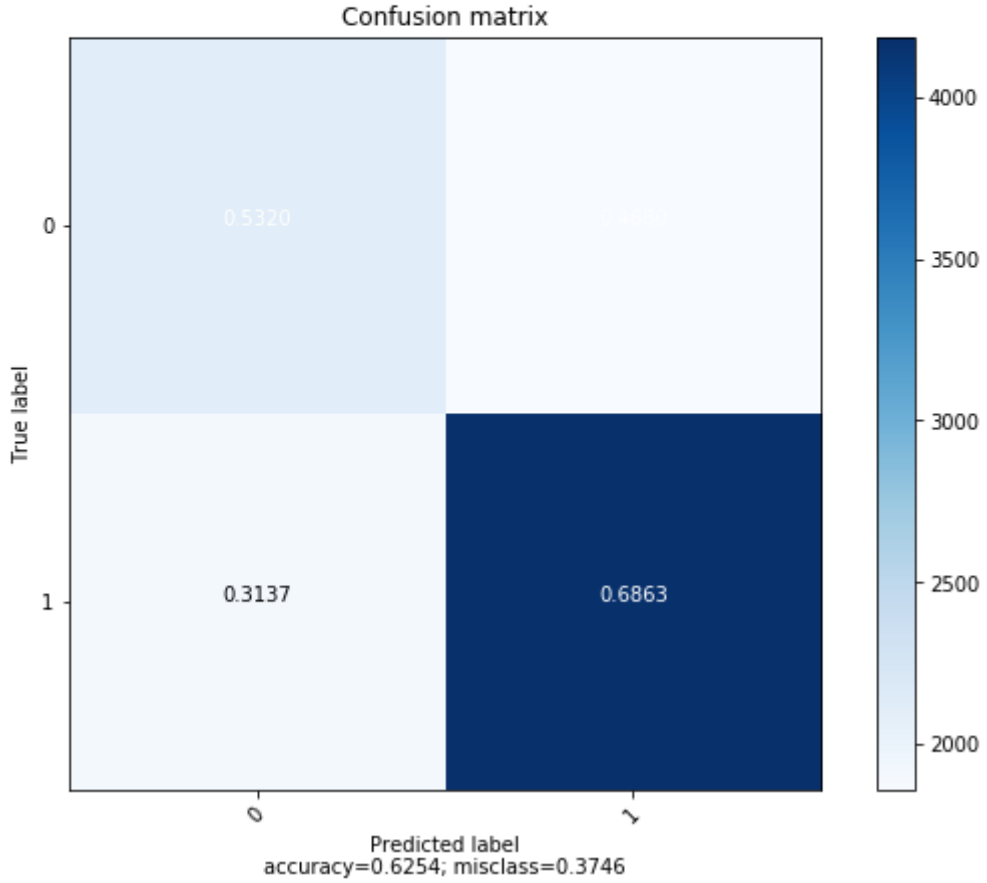


Figure 23: Confusion matrix of Random Forest on DEAP dataset

- **K-Nearest Neighbors algorithm:**

We trained K-Nearest Neighbors classifier using the following parameter: n_neighbours: 10, P: 1. The results can be seen in the following confusion matrix:

	precision	recall	f1-score	support
0	0.50	0.37	0.42	3974
1	0.65	0.76	0.70	6099
micro avg	0.60	0.60	0.60	10073
macro avg	0.57	0.56	0.56	10073
weighted avg	0.59	0.60	0.59	10073

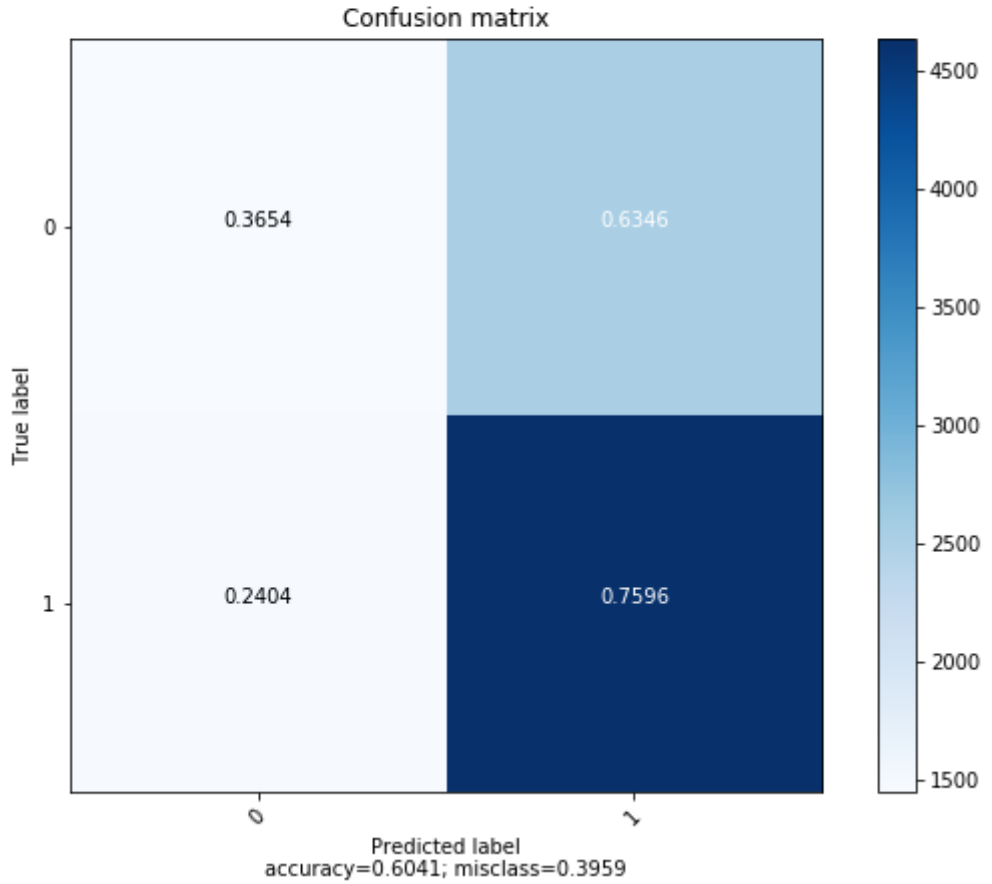


Figure 24: Confusion matrix of KNN on DEAP dataset

- **LSTM Neural Network**

We have used the same network as in Veteran Dataset which is described in Section 4.4.2. The resulting confusion matrix looks like below:

	precision	recall	f1-score	support
0	0.53	0.18	0.27	363635
1	0.54	0.86	0.66	409940
micro avg	0.54	0.54	0.54	773575
macro avg	0.54	0.52	0.47	773575
weighted avg	0.54	0.54	0.48	773575

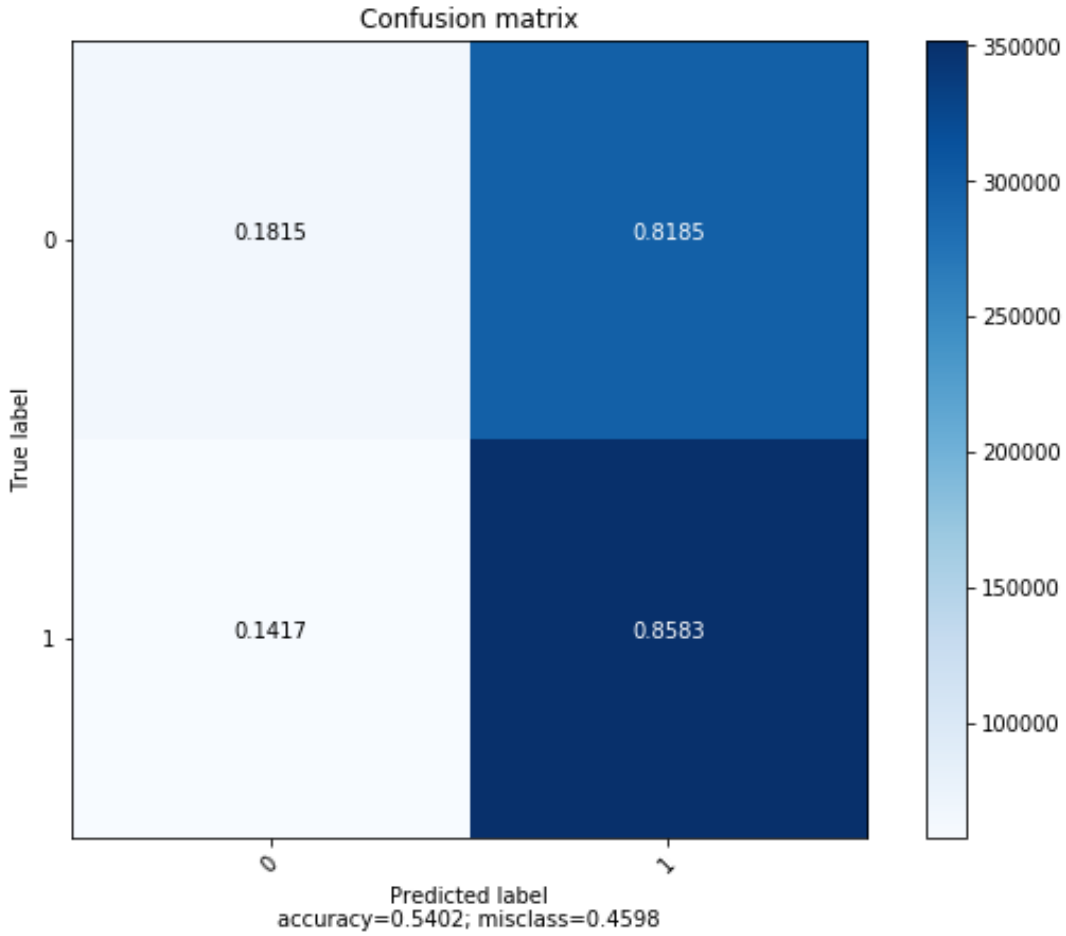


Figure 25: Confusion matrix of LSTM on DEAP dataset

➢ **Naive Bayes**

We have trained Naïve Bayes classifiers while alpha parameter set to 200.

The resulting confusion matrix looks like following:

	precision	recall	f1-score	support
0	0.44	0.22	0.29	2857
1	0.62	0.81	0.70	4378
micro avg	0.58	0.58	0.58	7235
macro avg	0.53	0.52	0.50	7235
weighted avg	0.54	0.58	0.54	7235

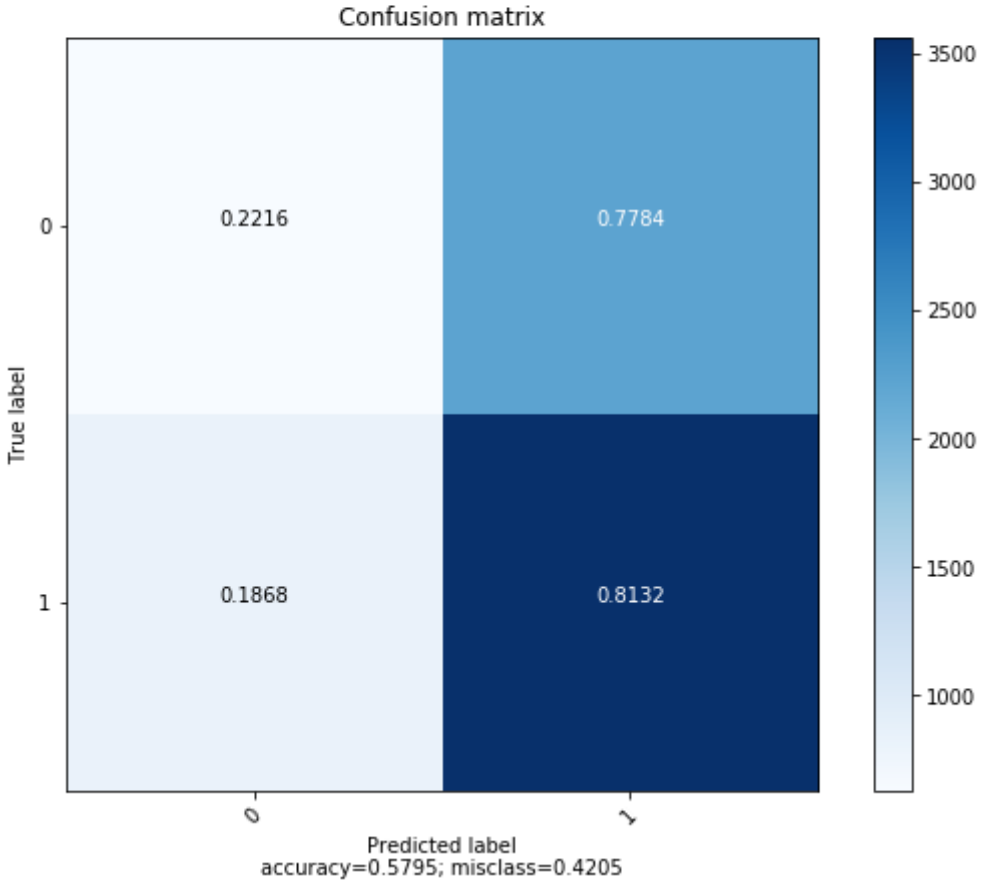


Figure 26: Confusion matrix of Naive Bayes on DEAP dataset

4.5.4 Independent Bio-signal analysis

We analyzed each bio-signal with same feature extraction and selection method to see which bio-signal individually a better indicator for stress classification is. In this case, we have brute forced the FDR values to obtain ten to twenty features

from each bio-signal, which allows us to keep more features than the previous experiments.

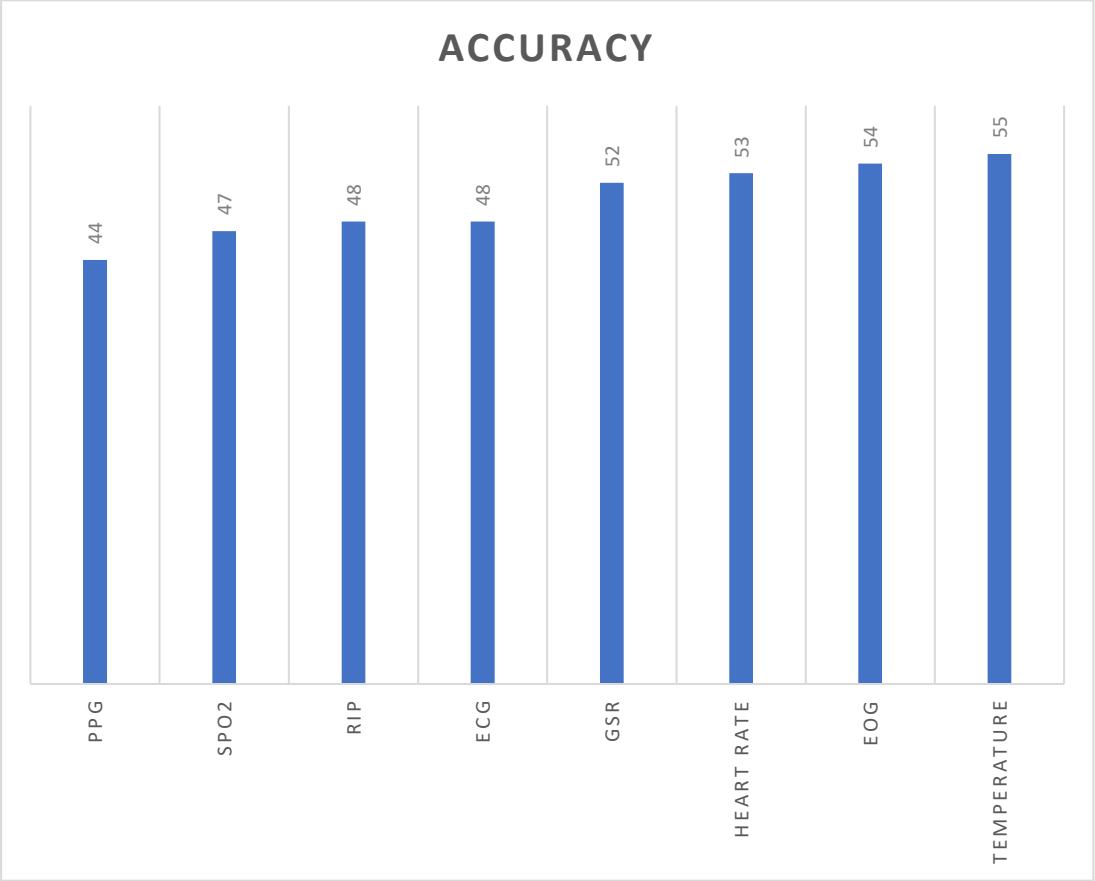


Figure 27: Independent bio-signal analysis of DEAP dataset

The above chart represents all the individual channels accuracy using Random Forest classifier at number of estimators count of 200 and a maximum depth of 30. We have picked Random forest classifier due to its reliable results in both the datasets and its ability to parallelize which results in faster execution. Random forest also allows us to gain reliable results without aggressive tuning. From the results, we can see supporting result with the features we extracted from all the bio-signal channels. The EMG signal collected from zygomaticus major muscle seems to be providing a good information about the arousal, vertical eye movement is also giving promising results for arousal classification. As we mentioned previously the features selection

process has selected 10-20 features per signal, the list of all the selected features without parameters can be found in the following tables:

Table 8: List of selected features from each physiological signal in DEAP dataset (1)

hEOG	**vEOG**	**tEMG**	**zEMG**
Change quantiles	Number of Peaks	Change quantiles	Partial autocorrelation
Absolute sum of changes	Autoregressive coefficient	Absolute sum of changes	Autoregressive coefficient
Mean absolute change	Time series complexity	Autoregressive coefficient	Change quantiles
Time series complexity	Approximate entropy	Aggregated linear trend	Mean absolute change
	Partial autocorrelation		
	Sample entropy		

Table 9: List of selected features from each physiological signal in DEAP dataset (2)

RIP	**PPG**	**GSR**	**Temperature**
Autoregressive coefficient	Change quantiles	Autoregressive coefficient	Change quantiles
Partial autocorrelation	Efficient complexity-invariant distance for time series	Fast fourier transform coefficient	Mean absolute change
Change quantiles	Cwt peaks	Large standard deviation	Last location of minimum
Mean absolute change	Spkt welch density	Augmented dickey fuller	First location of maximum
Fast fourier transform coefficient	Mean absolute change	Symmetry looking	Aggregated linear trend
Number of Peaks	Aggregated linear trend	Mean second derivative central	Fast fourier transform coefficient
Mean absolute change	Large standard deviation	Efficient complexity-invariant distance for time series	
Aggregated linear trend		Has duplicate	

4.5.5 DEAP dataset after data reduction

As can be seen in the DEAP dataset introduction at Data visualization & Preprocessing section, some of the user data does not contain proper values. As it can be seen from Figure 20 some uses have bad data on Temperature, GSR, RIP and PPG data. To better understand the effects of those noisy data, we have removed those users. One thing to note here is that with the removal of bad data, we are also losing some of the bio-signal channels which contains healthy data. You can find the visualization of this reduced used data in the following image:

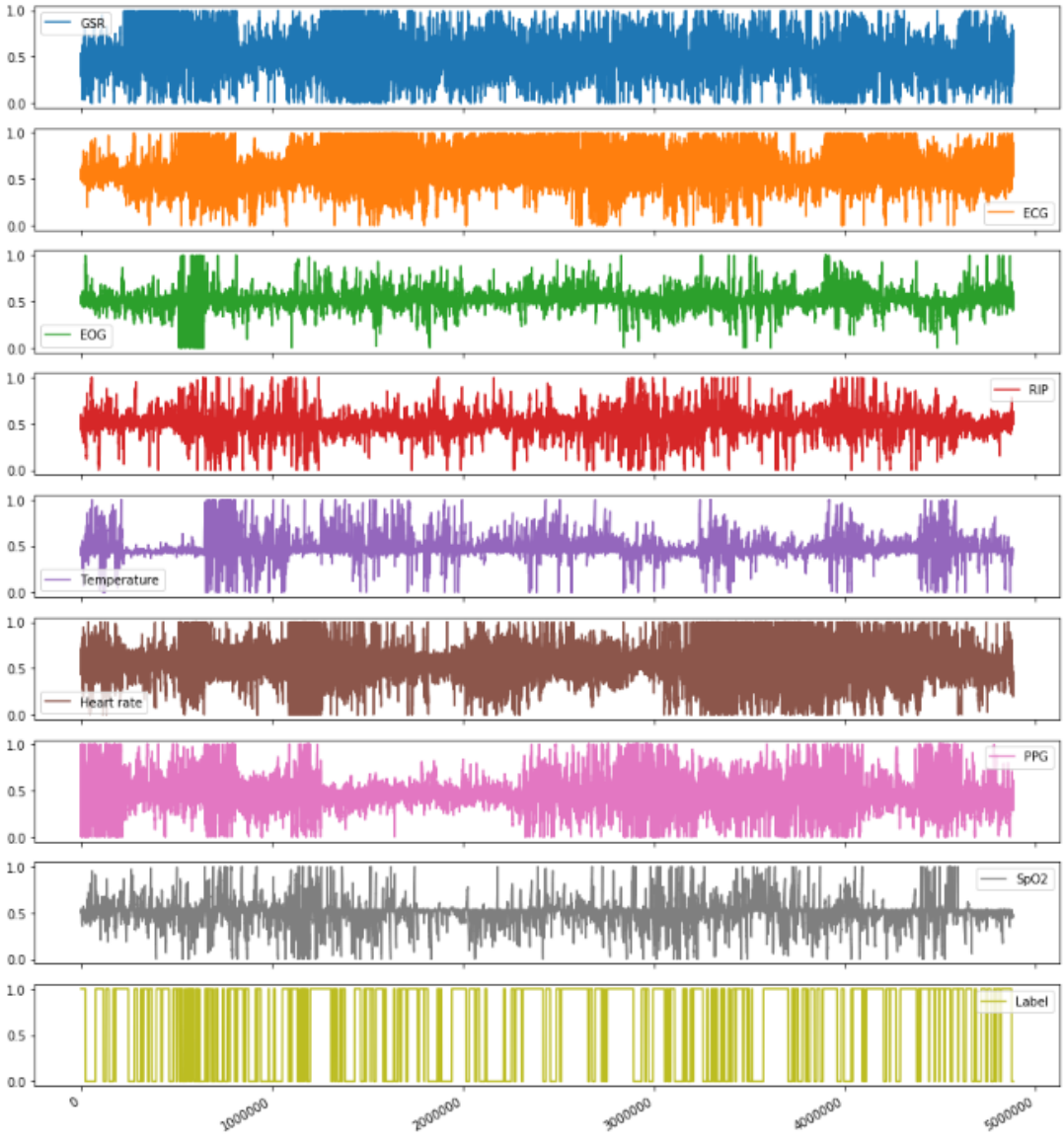


Figure 28: Reduced DEAP data analysis

We have removed last eight user data to achieve this result. This means that the above image contains data from 24 users. We have used Random Forest classifier with similar parameter of: N_estimator : 400, max_depth : 200, min_samples_leaf : 30. The results can be seen in the following confusion

	precision	recall	f1-score	support
0	0.40	0.24	0.30	447
1	0.62	0.77	0.68	710
micro avg	0.57	0.57	0.57	1157
macro avg	0.51	0.51	0.49	1157
weighted avg	0.53	0.57	0.54	1157

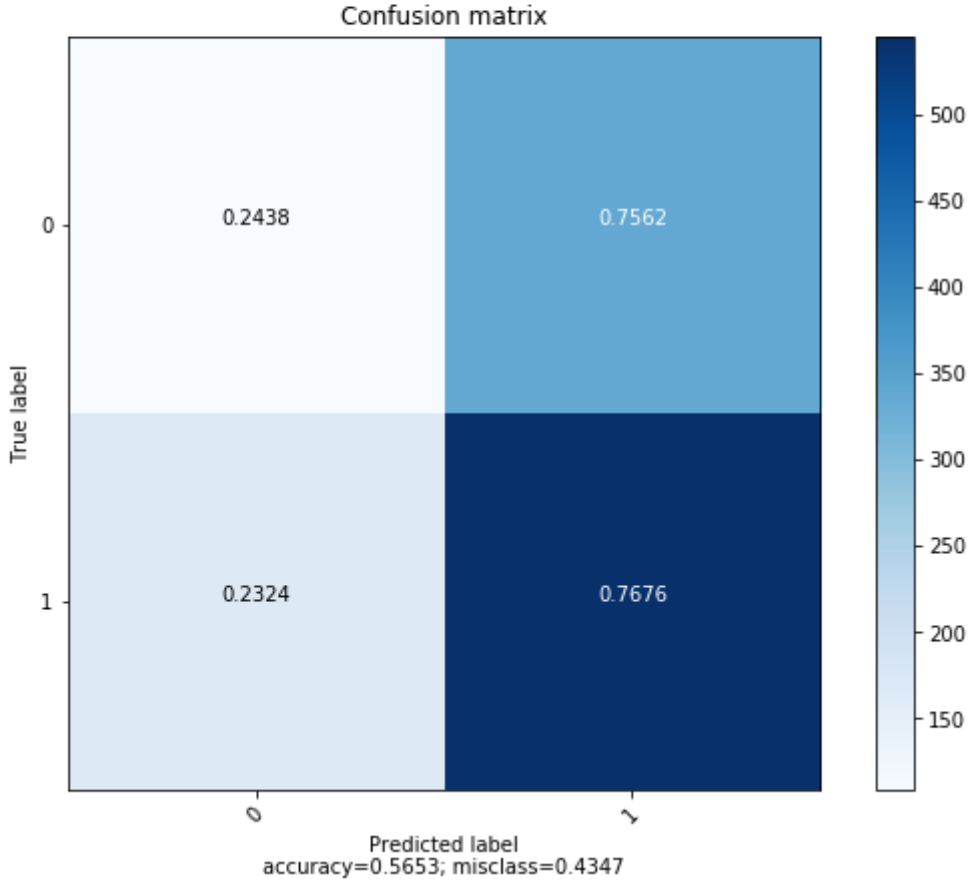


Figure 29: Confusion matrix of random forest on reduced DEAP dataset

As we can see from the result the accuracy seems to be decreasing, this can be due to the reduction of overall size of data as well due to increase in imbalance of class in the dataset, which can be seen in below chart:

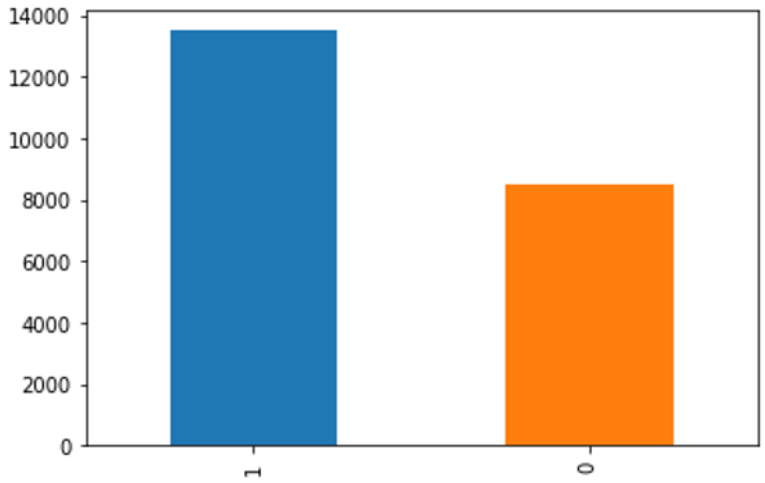


Figure 30 : DEAP dataset class imbalance after data-reduction

4.5.6 DEAP dataset summary

DEAP research team has also generated results with 57% accuracy using Naïve Bayes classifier. DEAP research team have used different method to divide dataset in testing and training set. They have used each user's data collected during one video as testing set and rest of the videos (39) in training set. More detail on the dataset is given in Usage of DEAP Dataset section. On our experiments we have divided our data using leave-one-out method.

Below figure shows all the results we gathered from DEAP dataset with their result for reference.

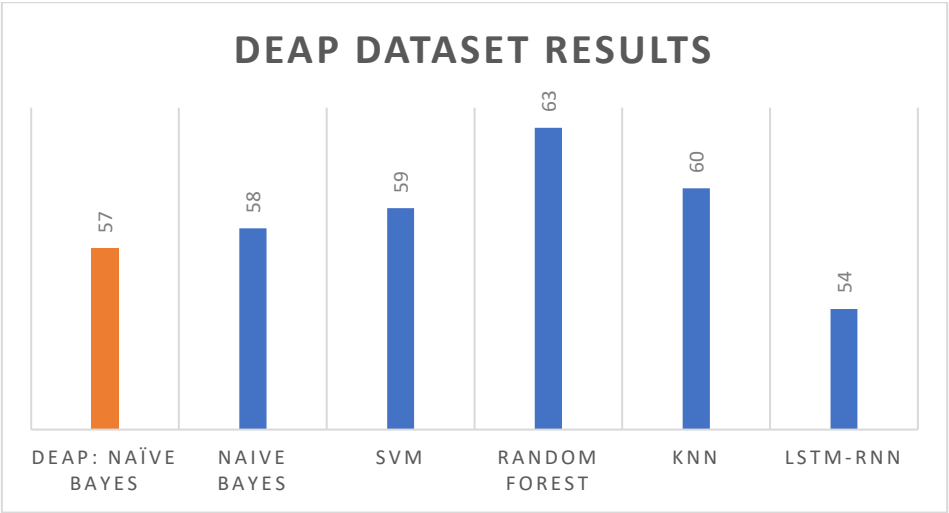


Figure 31: DEAP dataset result summary

5. CONCLUSION

In this work, we proposed a non-invasive data-collection and classification pipeline for Bio-signal data. The primary goal of this research is to classify human emotions using bio-signal and machine learning techniques, we have been able to achieve this goal as demonstrated in the Results section. We also employed our pipeline in the real-world scenario during the project - 'Using Virtual Reality Exposure Treatment and Real-Time Physiological Monitoring to Address PTSD Symptoms in Veterans' that resulted in the development of VR environment to help veterans with PTSD symptoms; which we have successfully used and tested on more than 30 veterans.

Our data processing methods also displayed significant improvement over the publicly available dataset; we gained marginal improvement in all classification task with p-value based feature selection techniques.

Even though results are encouraging, certain aspects of these experiments can be improved like the data classification label still entirely relies on participants feedback, which is subjective and thus cannot be guaranteed to be true.

Experiment design is a crucial component to success in this research, the quality of data can be improved using more controlled data collection environment. Experiment with more standardized tests like the cold-pressor test, word association test and more can be carried out to gain more robust data with reliable labels.

A significantly larger dataset is needed, certain bio-signals changed significantly between participants, for a machine learning classifier to identify this pattern, more data is key.

A more complex and signal depended feature extraction method can also result in better results. For example, change in GSR is fairly easy to spot by human

observation; on graph however, the feature extraction process would require trend analysis to detect this type of event.

We have successfully collected and classified bio-signals for stress, we are aware that the current method of obtaining the bio-signals requires specialize hardware. However, considering the current trend of wearable computing, it can be expected that the bio-sensors will be tiny enough to be used in more natural way. An important component for future work is to increase the number of emotions to investigate while reducing the number of bio-signals that we obtained which may allow us hassle free sensor arrangement.

LITERATURE CITED

[1] J. pan and W. J. Tompkins, *A Real-Time QRS Detection Algorithm,* IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING, 1985.

[2] A. Haag, S. Goronzy, P. Schaich and J. Williams, "Emotion Recognition Using Bio-Sensors: First Steps Towards an Automatic System," *Sony Corporate Laboratories Europe.*

[3] C. CORTES and V. VAPNIK, "Support-Vector Networks," Kluwer Academic Publishers, 1995.

[4] R. Routledge, "Bayes Theorem," Encyclopeadia Britannica , [Online]. Available: https://www.britannica.com/topic/Bayess-theorem. [Accessed 2018].

[5] W. B. Cannon, "The James-Lange Theory of Emotions: A Critical Examination and an Alternative Theory," University of Illinois Press, 1927.

[6] R. Reisenzein, "The Schachter Theory of Emotion: Two Decades Later," American Psychological Association, Inc., 1983.

[7] C. E. Izard and A. J. Fridlund, "Facial expressions and the regulation of emotions," Journal of Personality and Social Psychology, 1990.

[8] Eun-Hye, Byoung-Jun, Sang-Hyeob, M.-A. Chung, Mi-Sook and Jin-Hun, "Emotion Classification Using Physiological Signals," Department of Psychology/Brain Research Institute, Chungnam National University, 2014.

[9] C. Busso, Z. Deng, S. Yildirim, M. Bulut, C. M. Lee, A. Kazemzadeh, S. Lee, U. Neumann and S. Narayanan, "Analysis of emotion recognition using facial expressions, speech and multimodal information," 6th international conference on Multimodal interfaces, 2004.

[10] J. Kim and E. Andre, "Emotion Recognition Based on Physiological Changes in Music Listening," IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, 2008.

[11] B. A. Sharpless and J. P. Barber, "A Clinician's Guide to PTSD Treatments for Returning Veterans," *US National Library of Medicine ,* 2012.

[12] B. C. T. Frueh, S. M. Beidel, D. C. Cahill and S. P., "Assessment of social functioning in combat veterans with PTSD," *American Psychological Association,* 2001.

[13] K. TB, F. BC, K. RG, H. R and M. KM., *Social anxiety disorder in veterans affairs primary care clinics,* Department of Psychology, George Mason University, 2006.

[14] S. M. Miller, E. R. Pedersen and G. N. Marshall, *Combat Experience and Problem Drinking in Veterans: Exploring the roles of PTSD, coping motives, and perceived stigma,* National Institute of Health, 2016.

[15] U. Technology, "Unity Game Engine," 2018. [Online]. Available: https://unity3d.com/. [Accessed 2018].

[16] MathWorks, "Matlab," MathWorks , 2018. [Online]. Available: https://www.mathworks.com/products/matlab.html.

[17] A. (. Atkielski), Artist, *Schematic diagram of normal sinus rhythm for a human heart as seen on ECG (with English labels).*. [Art]. 2007.

[18] "EKG Interpretation," nurseslearning, [Online]. Available: https://www.nurseslearning.com/courses/nrp/nrp1619/section%205/index.htm.

[19] tsfresh, "Overview on extracted features," 2018. [Online]. Available: https://tsfresh.readthedocs.io/en/latest/text/list_of_features.html.

[20] T. Schreiber and A. Schmitz, "Discrimination power of measures for nonlinearity in a time series," PHYSICAL REVIEW E, 1997.

[21] Batista and Gustavo, "CID: an efficient complexity-invariant distance for time series.," Data Mining and Knowledge Difscovery, 2014.

[22] Friedrich, "Extracting model equations from experimental data," Physics Letters A 271, p. 217-222, 2000.

[23] G. M. Jenkins, G. C. Reinsel and G. M. Ljung, "Time series analysis: forecasting and control," John Wiley & Sons, 2015.

[24] "Partial Autocorrelation Function," The Pennsylvania State University.

[25] B. Fulcher and N. Jones, "Highly comparative feature-based time-series classification," Knowledge and Data Engineering, IEEE Transactions on 26, 3026–3037., 2014.

[26] I. Guyon and A. Elisseeff, "An Introduction to Variable and Feature Selection," Journal of Machine Learning Research, 2003.

[27] M. Christ, A. W. Kempa-Liehr and M. Feindt, "Distributed and parallel time series feature extraction for industrial big data applications," eprint arXiv, 2016.

[28] Y. Benjamini and D. Yekutieli, "The control of the false discovery rate in multiple testing under dependency," The Annals of Statistics, 2001.

[29] D. Cournapeau, M. Brucher, F. Pedregosa, G. Varoquaux, A. Gramfort and V. Michel, "scikit-learn: machine learning in Python," Open source, [Online]. Available: https://scikit-learn.org. [Accessed 2018].

[30] S. Koelstra, C. Muehl, M. Soleymani, J.-S. Lee, A. Yazdani, T. Ebrahimi, A. N. T. Pun and I. Patras, "DEAP: A Database for Emotion Analysis using Physiological Signals," IEEE Transaction on Affective Computing, 2011.

[31] G. L. NeuroTechnologies, "glneurotech.com," [Online]. Available: https://glneurotech.com/bioradio/wp-content/uploads/2017/05/392-0050-Rev-F-BioRadio-User-Guide.pdf.

[32] "MATLAB Product Overview," 15 January 2016. [Online]. Available: https://www.mathworks.com/products/matlab/index.html.

[33] "Institutional Review Board," Texas State University, [Online]. Available: http://www.txstate.edu/research/orc/IRB-Resources.html. [Accessed 4 November 2016].

[34] T. S. University, "Institute Review Board - Office of Research and Sponsored Programs," 2018. [Online]. Available: https://www.txstate.edu/research/orc/IRB-Resources.html.

[35] L. Liu and M. T. Özsu, Encyclopedia of Database Systems, Springer, 2009.

[36] C. Busso, Z. Deng, S. Yildirim, M. Bulut, C. M. Lee, A. Kazemzadeh, S. Lee, U. Neumann and S. Narayanan, "Analysis of Emotion Recognition using Facial Expressions, Speech and Multimodal Information," Proceedings of the 6th international conference on Multimodal interfaces, 2004.