ASSIGNMENT 1
CSE 571 SPRING 2020

BY

SAI KASYAP KAMARAJU

Introduction:

This is assignment is an attempt understand human activities that can identify eating with spoon or fork or not eating with them. The real-world Mayo data consists of Wristband Sensor, IMU and EMG sensors, but in this project only one sensor data is used for all users i.e EMG sensor which has a sampling rate of 100 Hz. This data is preprocessed in Phase 1; useful features are extracted in phase 2 and finally PCA is applied in Phase 3 to understand which one of these features are dominant ones. The Project is being on done MATLAB R2019b version.

Phase 1: Data cleaning and Data organization

The EMG sensor also called as electromyography sensor is used for detecting small electrical signals generated which are generated by muscles when say arms are lifted, fist clenching or any movement by fingers while eating action is performed. The EMG sensor data consists of Timestamps (which are ignored in this assignment) and sensor data EMG1, EMG2… EMG8 for each user. Also, ground Truth data is provided to know beginning and end of an eating action using where frame numbers in them.[1]

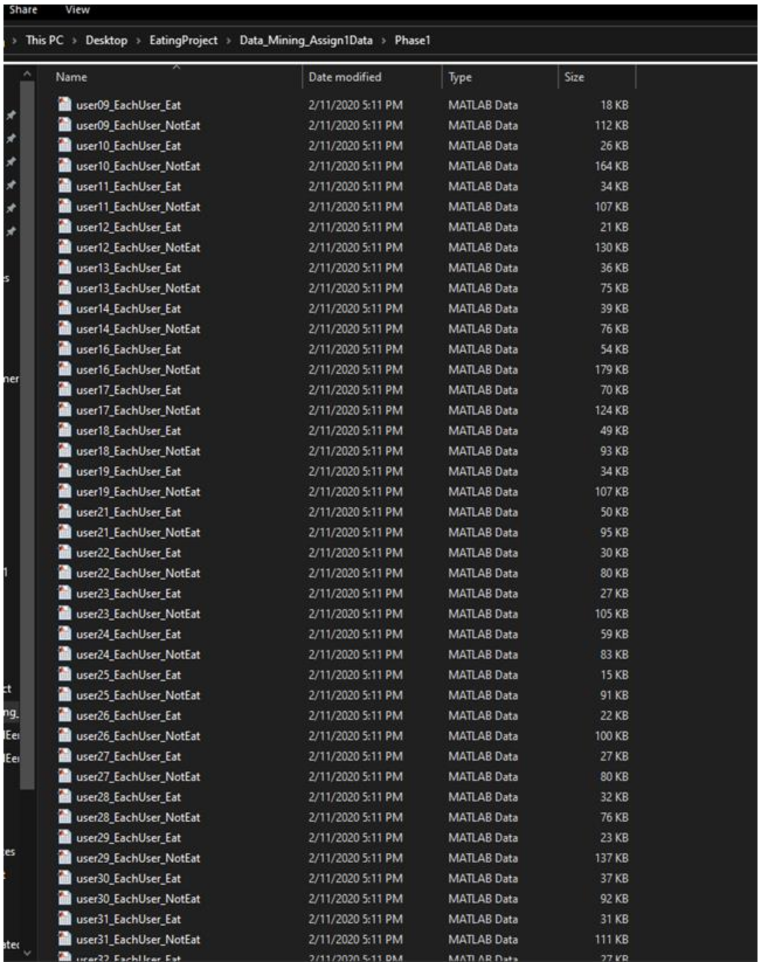
There two distinguished actions in this data set either eating by spoon or eating by fork. The video of the data though not used is taken at 30 samples per second.

The logic of Phase 1 is as follows:

1. The Timestamps are ignored in this assignment instead sample numbers are used for plotting.
2. The ground Truth data consists of 3 columns out of which the third column is ignored while the first two denote the beginning and end of the row numbers of for eating action either by spoon or fork when multiplied by 100 (EMG sampling rate) and divided by 30 (video frame sampling rate).
3. The above computations are mapped to EMG Sensor data for all the users by creating a list of them.
4. The row numbers of EMG sensor data are further divided based on the flooring the mapped values from the groundTruth in to 4 tasks i.e Fork eating, Fork not eating, Spoon Eating and Spoon Noneating for better understanding for each user.
5. The code for phase1 is named as DataCleaningAndOrganinization.m.

Also, renaming of User9 folder of groundTruth to User09 (same as EMG folder) is done to compute for this user.

After running the code, the data is saved into .mat files for each user as eating or non-eating as shown below.



Share View

This PC › Desktop › EatingProject › Data_Mining_Assign1Data › Phase1

Name	Date modified	Type	Size
user09_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	18 KB
user09_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	112 KB
user10_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	26 KB
user10_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	164 KB
user11_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	34 KB
user11_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	107 KB
user12_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	21 KB
user12_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	130 KB
user13_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	36 KB
user13_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	75 KB
user14_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	39 KB
user14_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	76 KB
user16_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	54 KB
user16_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	179 KB
user17_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	70 KB
user17_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	124 KB
user18_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	49 KB
user18_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	93 KB
user19_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	34 KB
user19_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	107 KB
user21_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	50 KB
user21_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	95 KB
user22_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	30 KB
user22_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	80 KB
user23_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	27 KB
user23_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	105 KB
user24_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	59 KB
user24_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	83 KB
user25_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	15 KB
user25_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	91 KB
user26_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	22 KB
user26_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	100 KB
user27_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	27 KB
user27_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	80 KB
user28_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	32 KB
user28_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	76 KB
user29_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	23 KB
user29_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	137 KB
user30_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	37 KB
user30_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	92 KB
user31_EachUser_Eat	2/11/2020 5:11 PM	MATLAB Data	31 KB
user31_EachUser_NotEat	2/11/2020 5:11 PM	MATLAB Data	111 KB

Figure 1: Partitioning Eating and Not Eating activities for each User

Phase 2: Feature Extraction

The analysis is performed on Eating and Noneating tasks of many users using the EMG sensor data. The Feature extraction implemented are as follows:

1. ***Max:***
   a) Max is a math function which is in built in MATLAB (max) which returns the greatest value. For the given signal, Max return the peak value in the given signal data.[5]

   b) In this case the Max function is computed across all the EMG signals (EMG1, EMG2, ..EMG8) for each row or sample number. The highest value of EMG1, EMG2, … EMG8 is returned using Max function say EMG1 for example. This makes sense to capture the highest signal EMG1 as a dominant feature for that particular row discarding the rest.

c) Code for Max function :

```
for k = 1 : size(User_Eat_Data, 1)
  % Calculation along X - Axis
  userEatMax = max(User_Eat_Data(k,1:8));
  userEatMax_data = [userEatMax_data;userEatMax];
end
```

%Where K variable loops through all the rows for user and max is computed across EMG signals 1:8.
The full code is available in Phase2Max.m

d) The Max plot for user 21 (eating and Noneating actions) and user 10 (eating and Non eating actions) are as shown below.
Where X- axis indicates the sample / row numbers and y- axis Max values.

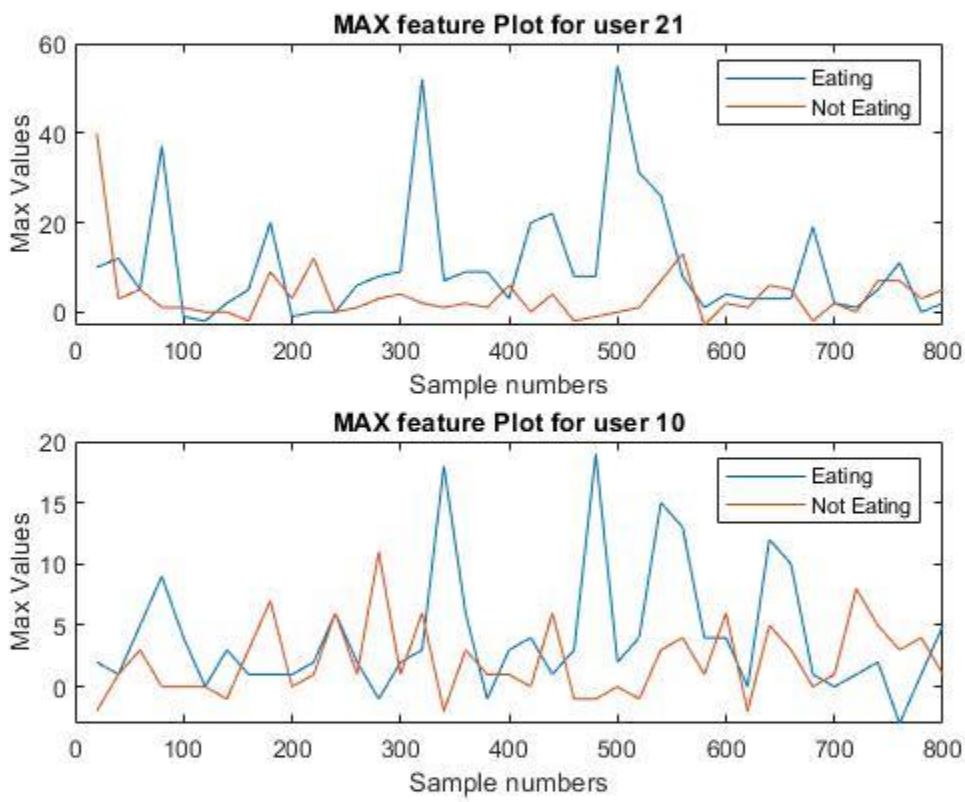


Figure 2: MAX as feature selection plot for different users for eating and non-eating activities.

e) Intuition: The eating action has generally has higher value signal across time when compared to that of non – eating.

2. Mean

a) As per definition, Mean is defined as sum of all the data points per number of data points. Statistically, mean refers to the expectation or average that is used to derive the central tendency of the given data. The formula for the statistical mean can be expressed as: Sample mean,[2]

$$\bar{x} = (\Sigma x_i) / n$$

b) For example, to calculate the mean of (EMG1, EMG2, ..EMG8) as a feature, the input will be EMG sensor signal value as input for each eating activity. Hence, for n sensor values EMG sensor mean can be considered as one of the important features.

c) The code for Mean function is in Phase2Mean.m file.

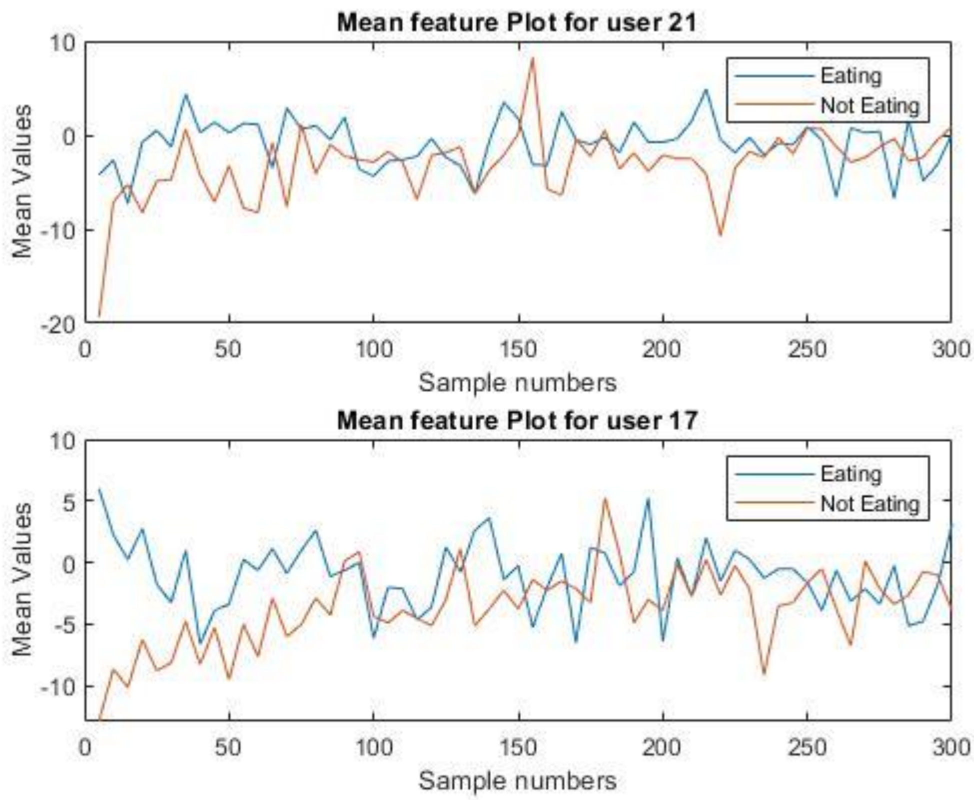


Figure 3: Mean as feature selection plot for different users for eating and non-eating activities.

d) The Mean plot for user 21 (eating and Noneating actions) and user 17 (eating and Noneating actions) are as shown above in figure 3. Where X- axis indicates the sample / row numbers and y- axis Max values.

e) The EMG signals of a human change across the time when a person is eating. Thus, reemphasis is made on mean values along the x-axis (time) to understand and differentiate an eating activity from the non-eating one. Non-eating activity will have a higher value as the time goes by as user can be move his fingers or muscles while noneating.

3. *Min:*

a) Min is quite the opposite of Max, a math function where in it returns the lowest value for a given set of data points. For the given signal, Min returns the least(global minima) value in the given signal data.

b) In our scenario, Min function is computed across all the EMG signals (EMG1, EMG2, ..EMG8) for each row or sample number in similar fashion as Max. The lowest value of EMG1, EMG2, … EMG8 is returned using Max function say EMG1 for example. This helps to figure out which is the relatively weak signal received across time

c) Code for Min Function: The code for Min function is named as Phase2Min.m

d) The Min plot for user 25 (eating and Noneating actions) and user 16 (eating and Noneating actions) are as shown below in figure 4.
Where X- axis indicates the sample / row numbers and y- axis Max values.

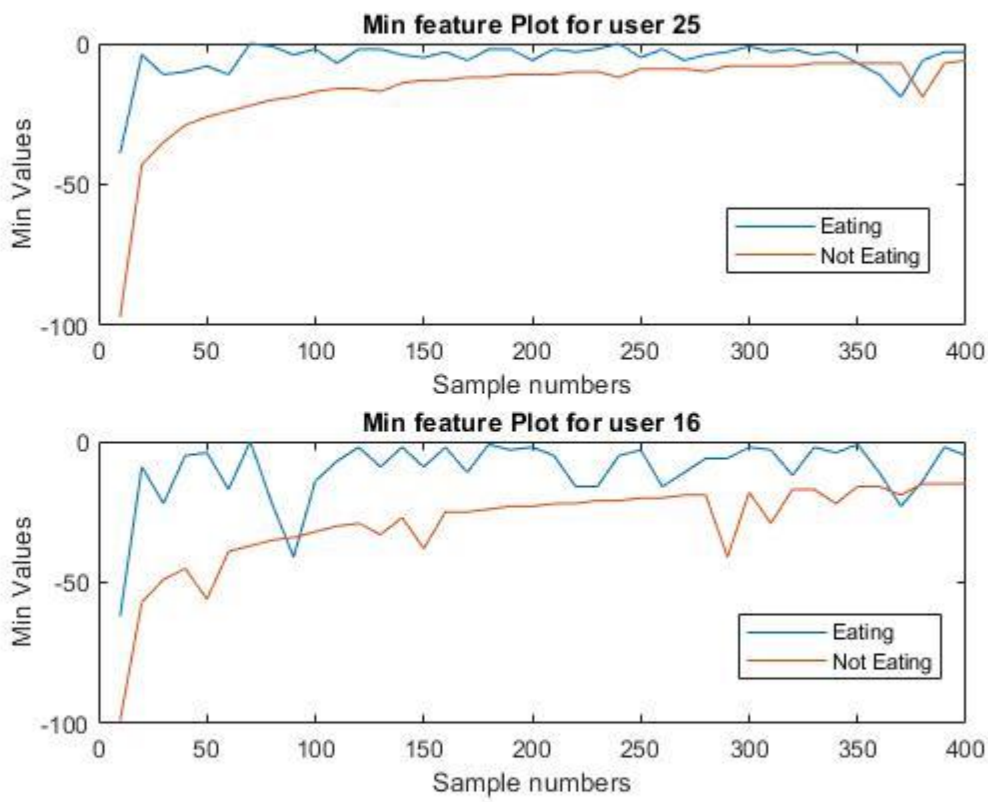


Figure 4: MIN as feature selection plot for different users for eating and non-eating activities.

e) Intuition: The eating action has higher minimum value signal across time when compared to that of non – eating one which is a useful feature for distinguishing activities.

4. *Standard Deviation:*

a) Statistically, Standard deviation is defined as measure that is used to calculate the amount of variation for a given set of data points. A low standard deviation implies that the data points are concentrated around mean while high standard deviation implies the data points are scattered out. The standard deviation is also defined as a square root of variance which is given by: [3]

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}$$

b) The standard deviation in our case gives an idea about the spread of EMG1, EMG2, … EMG8 signals across time i.e how close they are with respect each other.

c) The code for standard deviation is defined in the file Phase2Std.m

d) The standard deviation plot for user 30 (eating and Non eating actions) and user 40 (eating and Non eating actions) are as shown below in figure 5.
Where X- axis indicates the sample / row numbers and y- axis Max values.

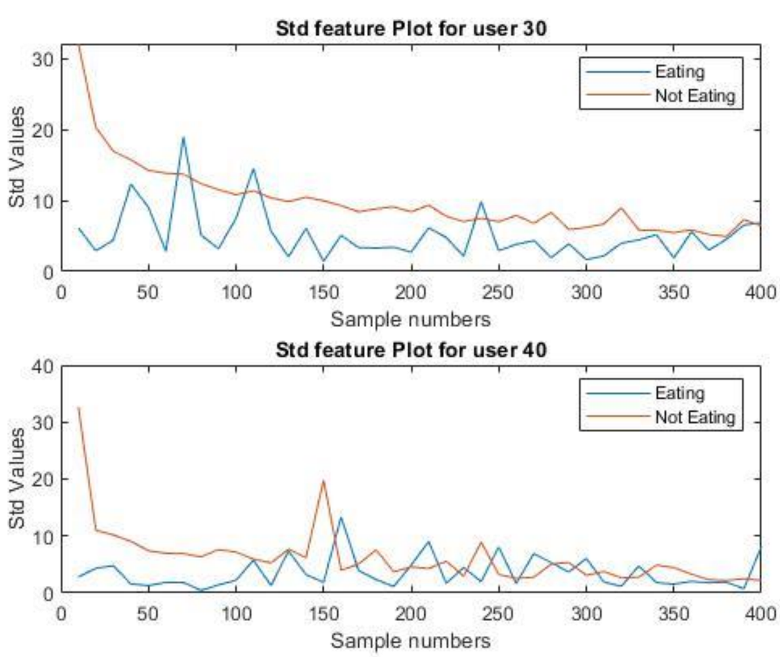


Figure 5: Standard deviation as feature selection plot for different users for eating and non-eating activities.

Intuition: The EMG sensor values show different range for different activities. Thus, it is easier to distinguish eating and non-eating activities using this parameter. Non-eating activities will have higher standard deviation across time when compared to eating activities, as eating actions are like one other while noneating are not.

5. *Fast Fourier Transform:*

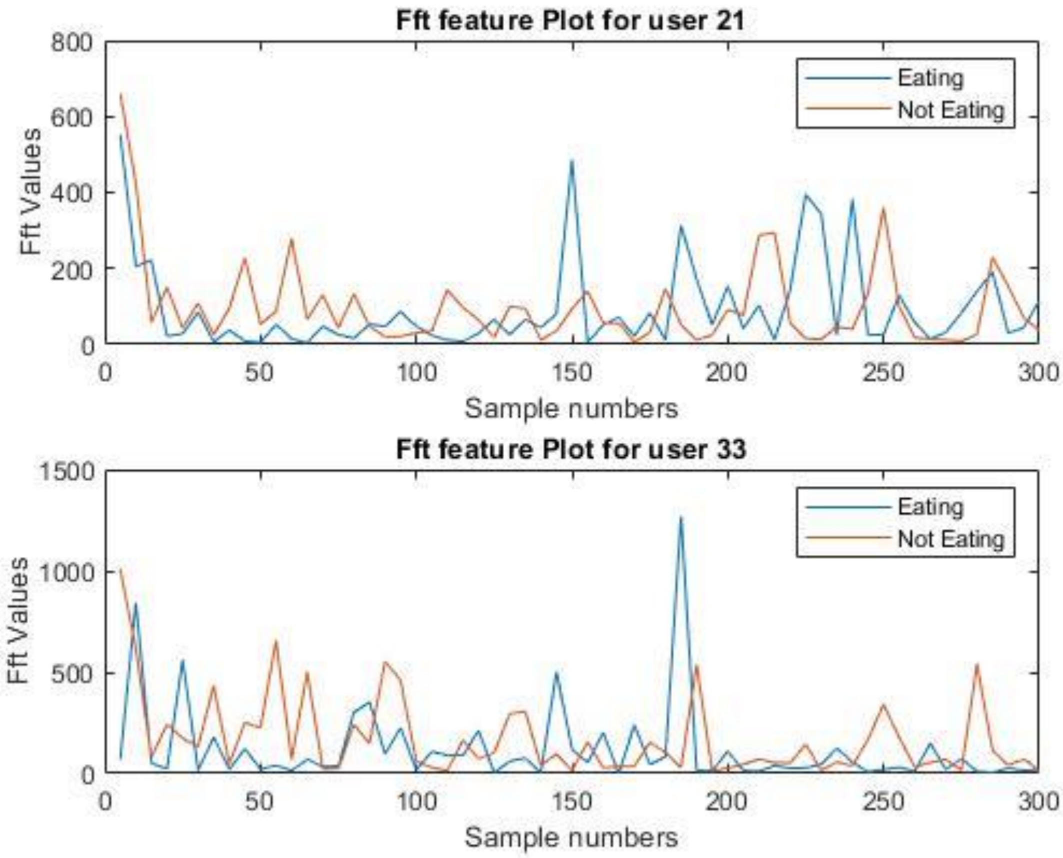
a) A fast Fourier Transform is used calculate the discrete Fourier transform (DFT) or its inverse (IDFT) for a given sequence. Fourier analysis converts a signal in time domain to frequency domain and vice versa. It is an algorithm that samples a signal over a period (or space) and divides it into its frequency components. The components of single sinusoidal oscillations are unique frequencies having their own amplitude and phase.

b) The input of an FFT is one signal there by generating frequencies which correspond to maximum values as output which is used as feature extraction technique. So, in our case we will know the peak value frequencies of signals EMG1, EMG2,… EMG8 and their variance with one and other.

c) The code for fft is defined in the file Phase2fft. m (variance of abs (fft value of signals) is taken across time.

d) The FFT plot for user 21 (eating and Noneating actions) and user 33 (eating and Noneating actions) are as shown below in figure 6.
Where X- axis indicates the frequency sample / row numbers and y- axis Max values.

Figure 6: FFT as feature selection plot for different users for eating and non-eating activities as shown below.



e) Intuition:
The time domain data samples are converted to frequency domain. Variance of FFT values are collected for each EMG signal and shows a clear distinction between the activities as it maps the signal to a new frequency space for each activity.

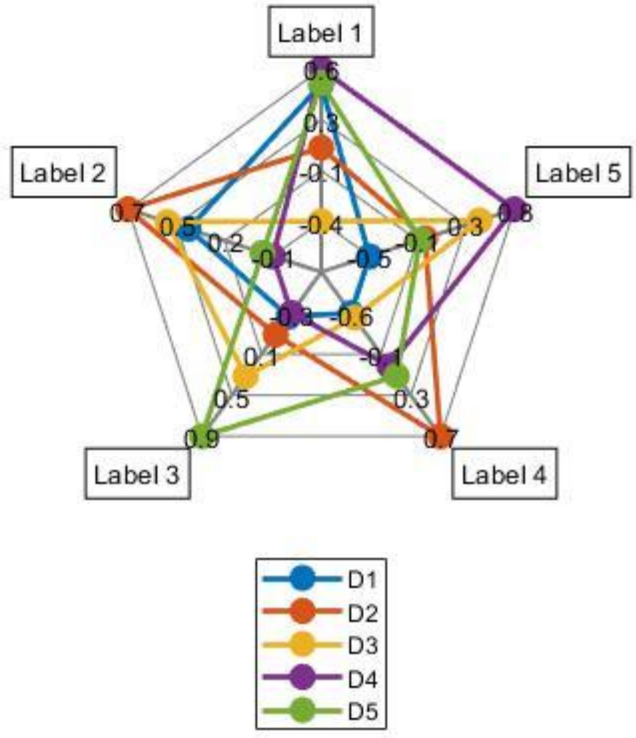
Phase 3: Feature Selection

In this Phase, PCA is used as feature extraction technique on the above 5 selected techniques across sample row numbers. Applying PCA, reduces the dimensions of selected features and helps to identify the topmost dominant features that could give a possible explanation to various activities such as eating and non-eating for each user.

Here, the PCA is applied on say mx n matrix where m refers to time period and n refers to number of feature columns that are taken from the phase 2.

a) The 5 features extracted from the above form the columns of the feature matrix. The number of rows are taken to be 2000 each for user for eating and non-eating activities. The plots have been generated separately for eating and non-eating activities.
b) The code of PCA is written phase3.m file

Figure 6: For Eating activity, Spyder plot for user 41



Here D1, D2, D3, D4, D5 refers to generated top 5 eigenvectors where in Max, Mean, Min, Std and FFT features of the feature matrix.

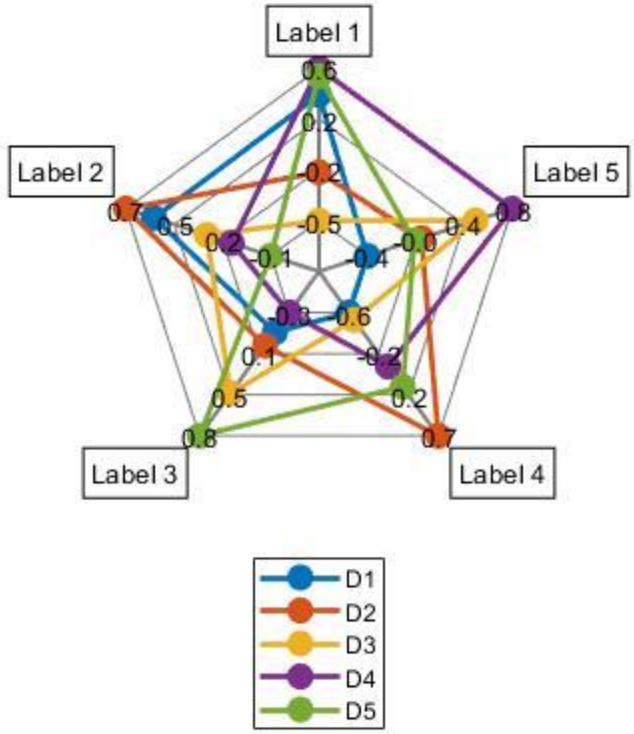


Figure 8: For Non-Eating activity, Spyder plot for user 41

PCA Result: Dimensionality gets reduced (Across time in this case).

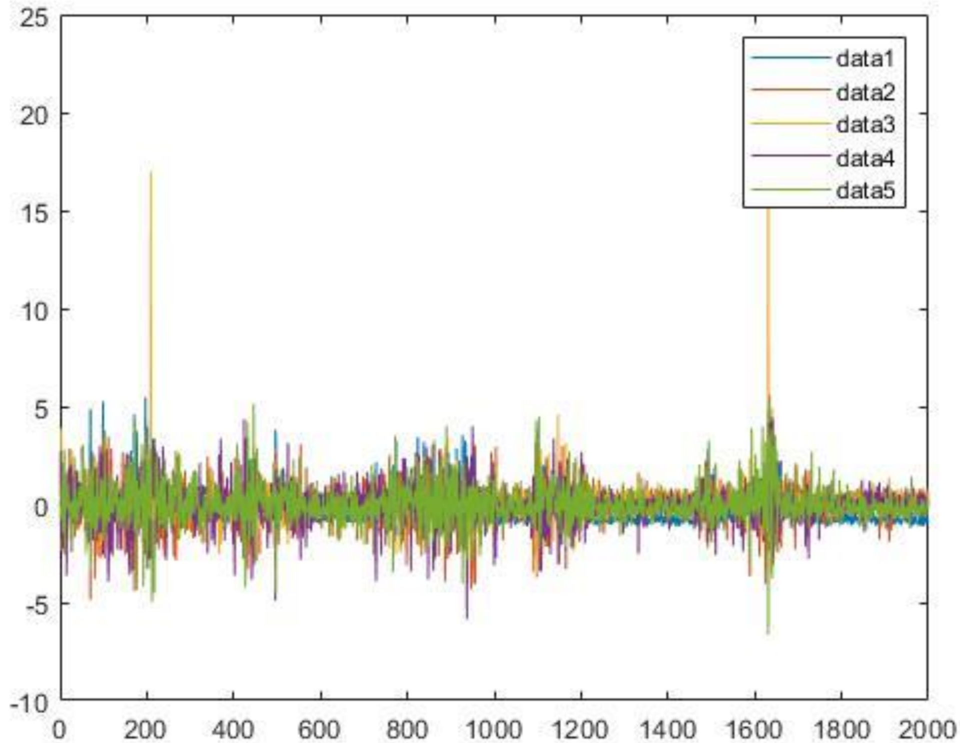


Figure 9: PCA multiplication with Feature Matrix for user 41 Eating activity

Multiplying PCA with the feature Matrix for user 41 eating activity as shown in the code ComparePCa.m

PCA helpful or not:

PCA is helpful as a dimensionality reduction technique where in the dimensions of the feature Matrix have been reduced to its principal components which show good enough variance in eating and non-eating for each user. The above plots indicate that though data complexity is reduced by PCA, the data is still intact

References:

1. https://www.seeedstudio.com/blog/2019/12/27/what-is-emg-sensor-myoware-and-how-to-use-with-arduino
2. https://www.statisticshowto.datasciencecentral.com/sample-mean/
3. https://www.mathsisfun.com/data/standard-deviation-formulas.html
4. Mathworks.com
5. Wikipedia.org