Group 24

Project Phase 2 Report

Data Mining

CSE 572

Spring 2018

Submitted to:

Professor Ayan Banerjee

Ira A. Fulton School of Engineering

Arizona State University

Submitted by:

Anuhya Sai Nudurupati (anuduru@asu.edu**)**
Anvitha Dinesh Rao (arao30@asu.edu**)**
Rachana Kashyap (rnkashya@asu.edu**)**
Rohit Balachandran Menon (skolli6@asu.edu**)**
Vaishak Ramesh Vellore (vvellore@asu.edu**)**

March 23, 2018

Table of Contents

1 Introduction

In this project, we attempt to develop a system which can understand and recognize the American Sign Language(ASL) through human gestures. A wristband sensor worn on both hands is used to collect data related to acceleration, gyroscope, orientation, electromyography and kinect data and is mined to understand what gesture the person has made. This could help a person who does not understand ASL to be able to communicate with a deaf/dumb person who does communicate in ASL. We use MATLAB to develop this software.

2 Project Phase 1

In the first phase, we went to the IMPACT lab at Brickyard, Tempe in order to collect data. One person wore wrist bands on both arms and made the gestures, “ABOUT”, “AND”, “CAN”, “COP” ,“DEAF”,“DECIDE”, “FATHER”, “FIND”, “GO OUT” and “HEARING” about 20 times each. The data collected from the sensors is stored in the form of CSV files. The time series data is sampled every 3 seconds. The frequency of sensors was found to be 15Hz. The data headers of the collected data are Accelerometer, Electromyogram, Gyroscope and Orientation. The full list of 34 headers are as follows:

1. ALX: Accelerometer X axis of Left hand
2. ALY: Accelerometer Y axis of Left hand
3. ALZ: Accelerometer Z axis of Left hand
4. ARX: Accelerometer X axis of Right hand
5. ARY: Accelerometer Y axis of Right hand
6. ARZ: Accelerometer Z axis of Right hand
7. EMG0L: EMG 0 pod of Left hand
8. EMG1L: EMG 1 pod of Left hand
9. EMG2L: EMG 2 pod of Left hand
10. EMG3L: EMG 3 pod of Left hand
11. EMG4L: EMG 4 pod of Left hand
12. EMG5L: EMG 5 pod of Left hand
13. EMG6L: EMG 6 pod of Left hand
14. EMG7L: EMG 7 pod of Left hand
15. EMG0R: EMG 0 pod of Right hand

16. EMG1R: EMG 1 pod of Right hand
17. EMG2R: EMG 2 pod of Right hand
18. EMG3R: EMG 3 pod of Right hand
19. EMG4R: EMG 4 pod of Right hand
20. EMG5R: EMG 5 pod of Right hand
21. EMG6R: EMG 6 pod of Right hand
22. EMG7R: EMG 7 pod of Right hand
23. GLX: Gyroscope X axis of Left hand
24. GLY: Gyroscope Y axis of Left hand
25. GLZ: Gyroscope Z axis of Left hand
26. GRX: Gyroscope X axis of Right hand
27. GRY: Gyroscope Y axis of Right hand
28. GRZ: Gyroscope Z axis of Right hand
29. ORL: Orientation roll of Left hand
30. OPL: Orientation Pitch of Left hand
31. OYL: Orientation Yaw of Left hand
32. ORR: Orientation roll of Right hand
33. OPR Orientation Pitch of Right hand
34. OYR: Orientation Yaw of Right hand

3 Project Phase 2

The second phase of the project involves feature extraction and feature selection aspects of Data Mining. This phase involves 3 main tasks:

Task 1: Segment raw sensor data into 10 separate classes, each of which are stored in a csv file. The csv file is rearranged such that each action and its data must be present in rows. For example,

Action 1 Acc X 9 6 4 7 8 2 4 7 8 ----------------

Action 1 Acc Y 7 4 0 2 4 9 2 8 9-----------------

This is done using MATLAB code.

```
gesture_matrix=[];
sensor_names=["ALX","ALY","ALZ","ARX","ARY","ARZ","EMG0L","EMG1L","EMG2L","EM
G3L","EMG4L","EMG5L","EMG6L","EMG7L","EMG0R","EMG1R","EMG2R","EMG3R","EMG4R",
"EMG5R","EMG6R","EMG7R","GLX","GLY","GLZ","GRX","GRY","GRZ","ORL","OPL","OYL"
,"ORR","OPR","OYR"];

column_first=[];
files = dir('*.csv');
m=1;
for file = files'
    csv = readtable(file.name,'ReadRowNames',false);
    new_var1=csv(:,1:34);
    if size(new_var1,1)>45
        continue
    end
    if size(new_var1,1)<35
        continue
    end
    old_matrix=table2array(new_var1);
    size_old=size(old_matrix);
    if size_old(:,1)<45
        var1=[old_matrix; zeros((45-size_old(:,1)),34)];
    else
        var1=old_matrix;
    end
    gesture_matrix=[gesture_matrix;var1'];
    for k=1:34
       s=" ";
       column1=strcat("ACTION",s,num2str(m),s,sensor_names(k));
       column_first=[column_first;column1];
    end
end
final_matrix=[];
final_matrix=horzcat(column_first,gesture_matrix);
final_table=array2table(final_matrix);
writetable(final_table,'about.csv','WriteRowNames',false);
```

Task 2: Feature Extraction

Subtask (i): Feature extraction procedure

To show clear distinction between each gesture, we first need to do feature extraction on the dataset.

We have selected the following feature extraction methods:

1. Mean
2. Root Mean Square (RMS)
3. Zero Crossing Rate (ZCR)
4. Variance
5. Energy Function
6. Mean Absolute Variance (MAV)

To extract the feature, we have written a matlab script to sort the matrices such that, the data generated in task 1 is sorted sensor by sensor. Based on the sensor, different feature extraction techniques have been used:

(i) Accelerometer: RMS (Root Mean Square); variance; ZCR (Zero Crossing Rate).

(ii) Gyroscope data: Mean; Variance

(iii) EMG: MAV (Mean Absolute Variance); ZCR (Zero Crossing Rate)

(iv) Orientation: ZCR; Energy

Subtask (ii): Intuition behind features selection

We first found out which sensors could have changed while performing a particular gesture.

ABOUT: Orientation, Gyroscope, ARY, ARZ, EMGR.

AND: EMGR and ARX.

CAN - ARY, ALY, EMGR, EMGY, Orientation, Gyroscope

COP - EMGR, ARZ

DEAF - ARX, ARZ, EMGR, Gyroscope.

DECIDE : ARY, ALY, EMGL, EMGR.

FATHER - ARY, ARZ, EMGR.

FIND - ARY, ARZ, EMGR, Gyroscope, Orientation

GO OUT: ARZ, EMGR.

HEARING: ARY, ARZ, Orientation and Gyroscope for right hand.

From the above information, we gathered that accelerometer and gyroscope are very important, but EMG and orientation are less important.

The intuition behind choosing the feature extraction methods is as follows:

(i) **Mean**: The mean talks about the central tendency of the data set and may tell us what value a particular sensor may tend to have for a particular gesture. The mean value for different gesture's sensors may help distinguish one from each other.

(ii) **Root Mean Square (RMS)**: RMS gives the measure of the magnitude of a set of numbers. In case of a set of positive and negative numbers, the mean may cancel the signs out and give an average of zero which is not useful for analysing our data. RMS gives us the sense of the size of the numbers without having the effect that the mean does.

(iii) **Zero Crossing Rate (ZCR):**

The ZCR could be important in understanding when the hand is in motion and when it is still and thus help us deduce which gesture was made.

(iv) **Variance:** There could be various degree of variance in the hand gestures, especially since they involve moving each hands in different directions, speed and angle. For example, Acceleration in Z axis of the right hand might have large variation for the gesture "GO OUT".

(v) **Energy Function**: The energy function is a function of the configuration of latent variables in a data set. For example, in some gestures, left hand is not used at all and hence may have zero values for acceleration and EMG. These features could have been omitted due to lack of variance. Thus, the energy function may help us find such features.

(vi) **Mean Absolute Variance (MAV)** is the measure of the variability of a univariate sample of data. It is more resilient to outliers and noise in a data set. In case of a data set that has large noise and outliers (during data collection), this measure could be a better method to use than standard deviation and mean.

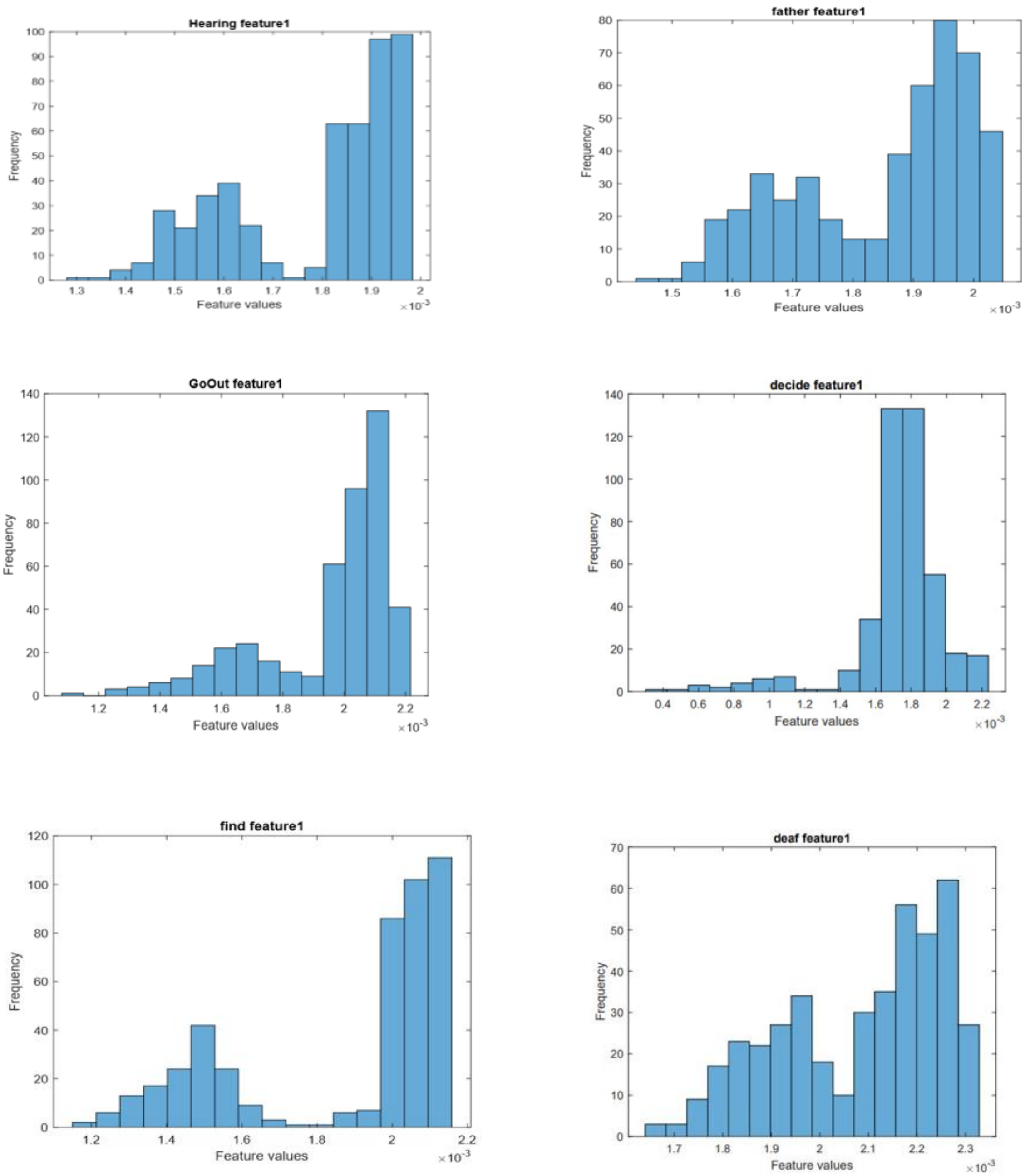
Subtask (iii): *Matlab code for feature extraction*

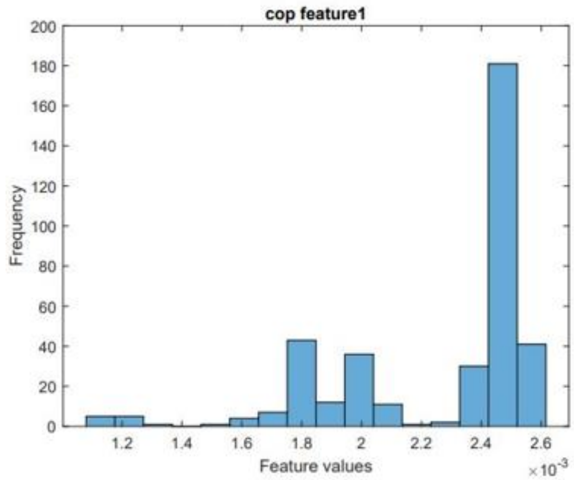
```
sorted_by_feature=[];
for feature=1:34
for i=1:size(gesture_matrix,1)
    if mod(i,34)==feature
        sorted_by_feature=[sorted_by_feature; gesture_matrix(i,:)];
    end

end
end
for i=1:size(about_matrix,1)
    if mod(i,34)==0
        sorted_by_feature=[sorted_by_feature; about_matrix(i,:)];
    end
end

%For accelerometer
feature_number=[1,2,3,4,5,6];
feature_extracted=[];
feature_acc=[];
for i=1:size(feature_number,2)
    factor=size(gesture_matrix,1)/34;
    feature_extracted=sorted_by_feature((feature_number(:,i)-
1)*factor+1:(feature_number(:,i))*factor,:)

   rms_feature                                                         =
sqrt(mean(feature_extracted.^2,2))/size(feature_extracted,1);
    var_feature=var(feature_extracted,0,2);
    ZCR=mean(abs(diff(sign(feature_extracted))),2);

    feature_acc=horzcat(feature_acc,rms_feature,var_feature,[ZCR;0]);
end

%For gyroscope: Standard deviation, RMS, ZCR and ABSDIFF
feature_number=[23,24,25,26,27,28];
feature_extracted=[];
feature_gyr=[];
for i=1:size(feature_number,2)
    factor=size(gesture_matrix,1)/34;
    feature_extracted=sorted_by_feature((feature_number(:,i)-
1)*factor+1:(feature_number(:,i))*factor,:)

    var_feature=var(feature_extracted,0,2);
    mean_abs_dev_feature=mad(feature_extracted,0,2);

    feature_gyr=horzcat(feature_gyr,var_feature,mean_abs_dev_feature);
end

%For  EMG:  RMS,  Integrated  EMG(IEMG)  ,Mean  Absolute  Value  (MAV)  ,Variance,
Zero Crossing
feature_number=[7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22];
feature_extracted=[];
feature_emg=[];
for i=1:size(feature_number,2)
    factor=size(gesture_matrix,1)/34;
    feature_extracted=sorted_by_feature((feature_number(:,i)-
1)*factor+1:(feature_number(:,i))*factor,:);
    mav_feature=mean(abs(feature_extracted),2);
```

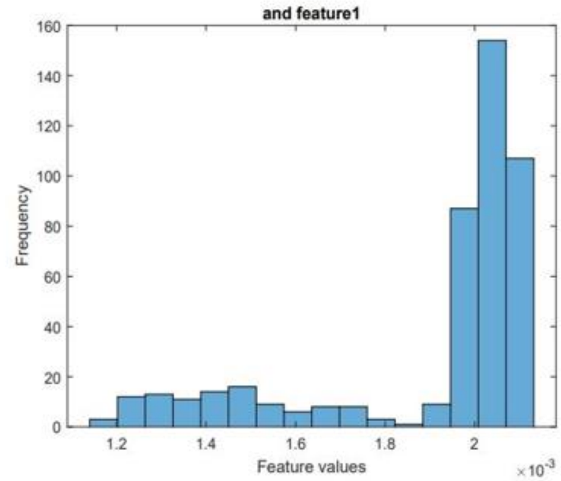
```
    ZCR=mean(abs(diff(sign(feature_extracted))),2);

    feature_emg=horzcat(feature_emg,mav_feature,[ZCR;0]);
end

%For OR 29,30,31,32,33,34 :First 5-FFT coefficients,Spectral energy,Standard
deviation
%ZCR,Zero-crossing rate,Absolute difference
feature_number=[29,30,31,32,33,34];
feature_extracted=[];
feature_or=[];
for i=1:size(feature_number,2)
    factor=size(gesture_matrix,1)/34;
    feature_extracted=sorted_by_feature((feature_number(:,i)-
1)*factor+1:(feature_number(:,i))*factor,:);
    fft_feature=fft(feature_extracted);
    ZCR=mean(abs(diff(sign(feature_extracted))),2);
    energy = sum(abs(fft_feature).^2,2);
    feature_or=horzcat(feature_or,[ZCR;0],energy);
end

feature_matrix=horzcat(feature_acc,feature_gyr,feature_emg,feature_or);
csvwrite('features_about.csv',feature_matrix,0,0);

%Plotting graphs for features
 for i=1:size(feature_matrix,2)
fig=figure(i);
histogram(feature_matrix(:,i),16);
title(strcat('can feature ', num2str(i)));
xlabel('Feature values');
ylabel('Frequency');
 filename = sprintf('Figure%02d.pdf', i);
      print( fig, '-dpdf', filename );
 end
```

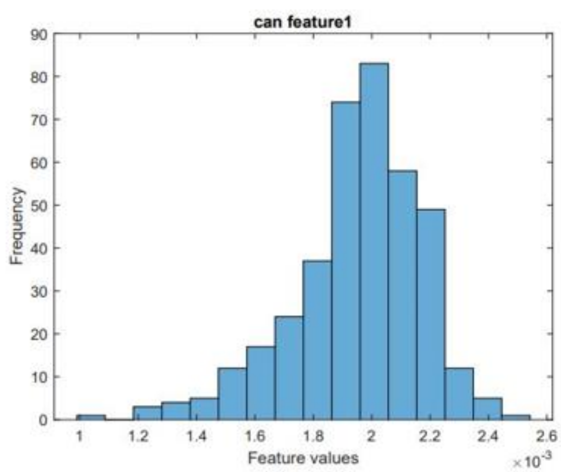
Subtask 4:

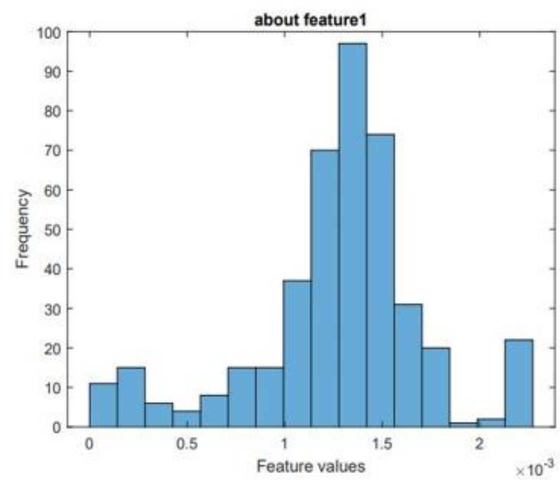
Feature 1 : RMS of Accelerometer X left hand


Hearing feature1
Frequency
Feature values
×10-3
father feature1
Frequency
Feature values
×10-3
GoOut feature1
Frequency
Feature values
×10-3
decide feature1
Frequency
Feature values
×10-3
find feature1
Frequency
Feature values
×10-3
deaf feature1
Frequency
Feature values
×10-3

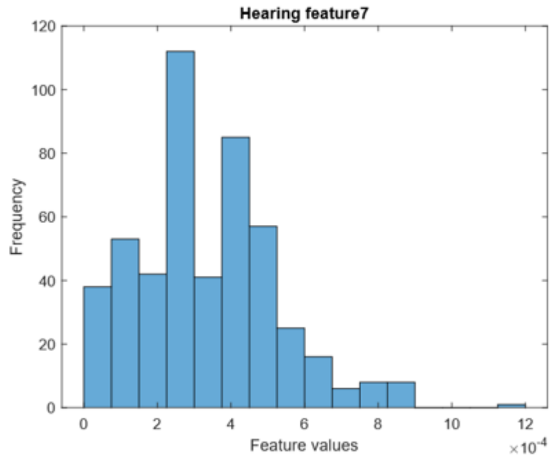


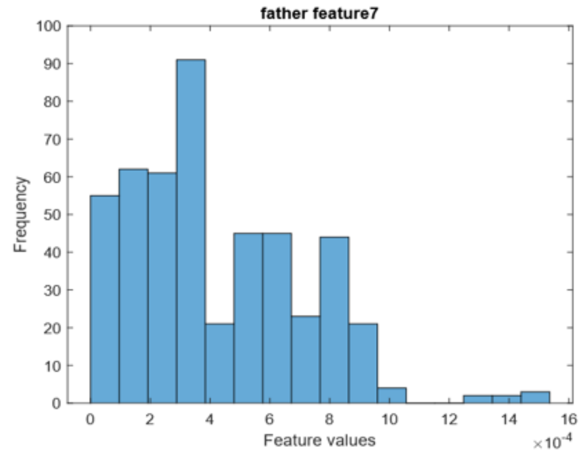


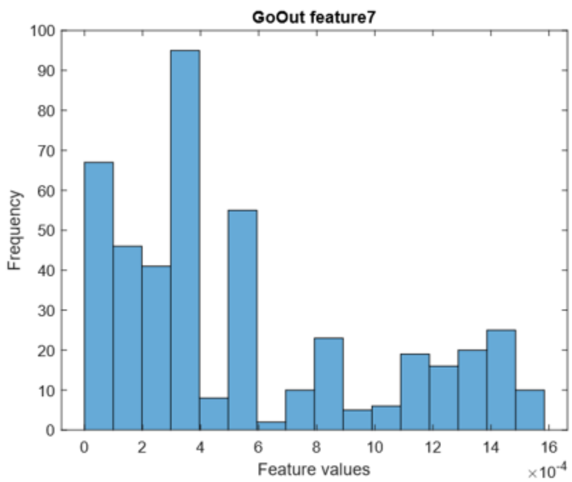


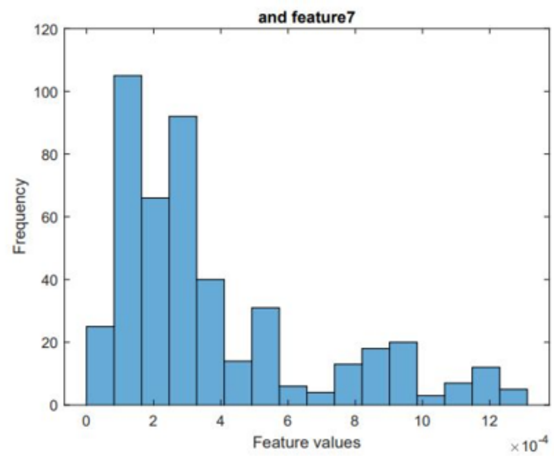


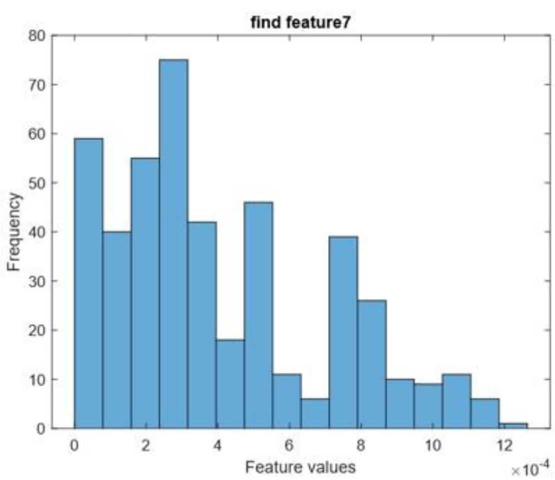
FEATURE 2: RMS of Accelerometer Z left hand

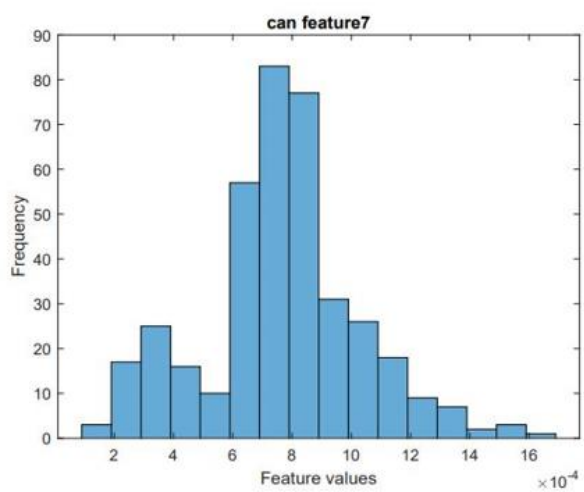

Hearing feature7
Frequency
Feature values
×10^-4

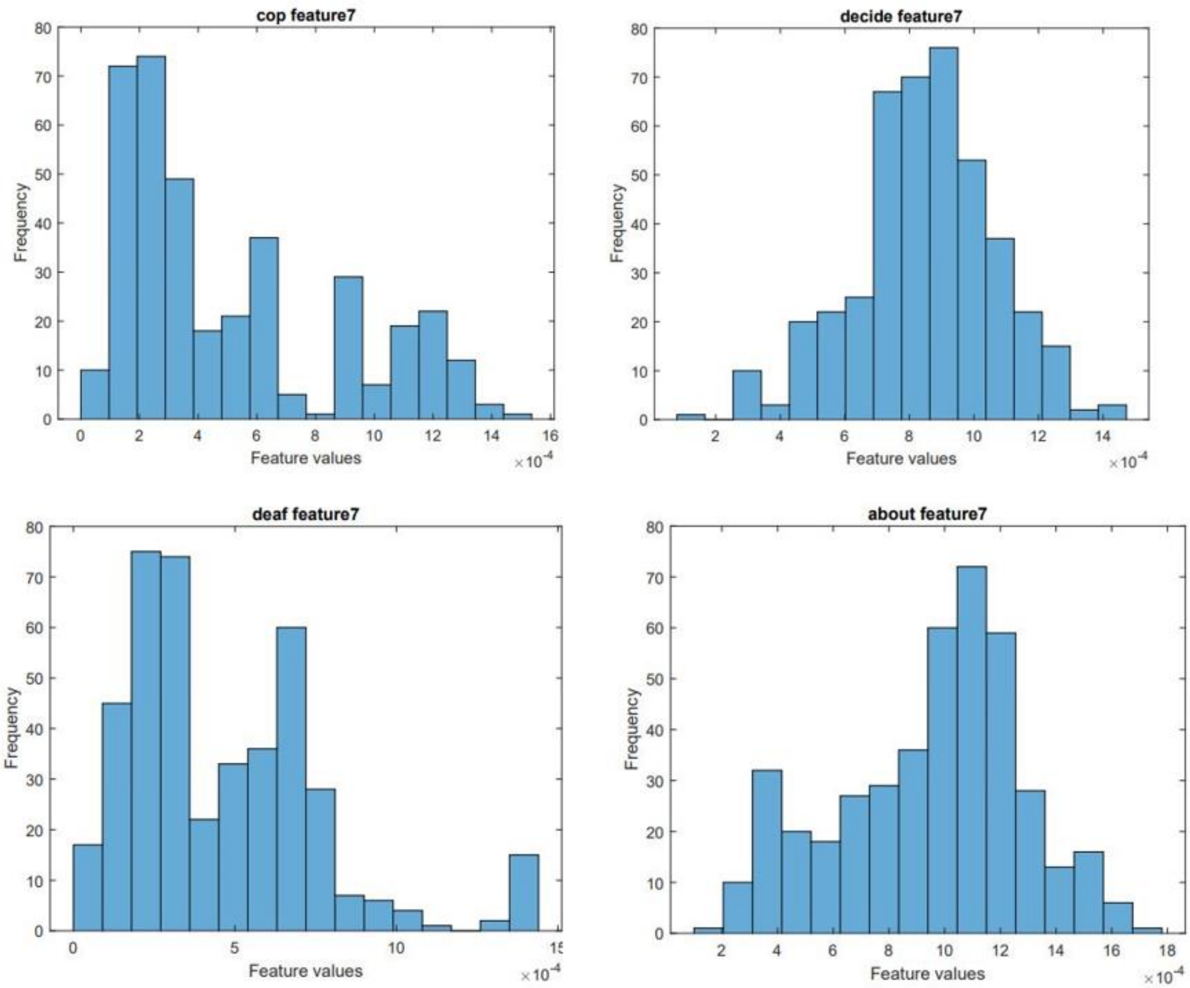

father feature7
Frequency
Feature values
×10^-4


GoOut feature7
Frequency
Feature values
×10^-4

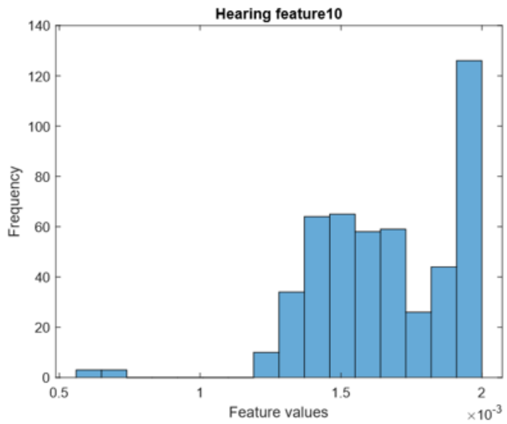

and feature7
Frequency
Feature values
×10^-4

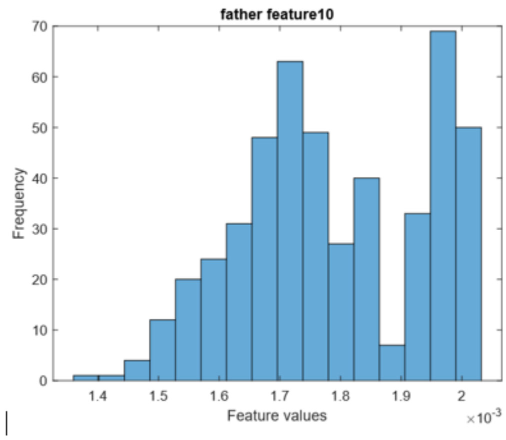

find feature7
Frequency
Feature values
×10^-4

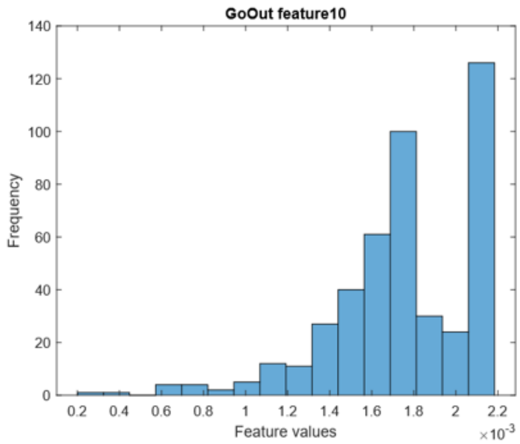

can feature7
Frequency
Feature values
×10^-4

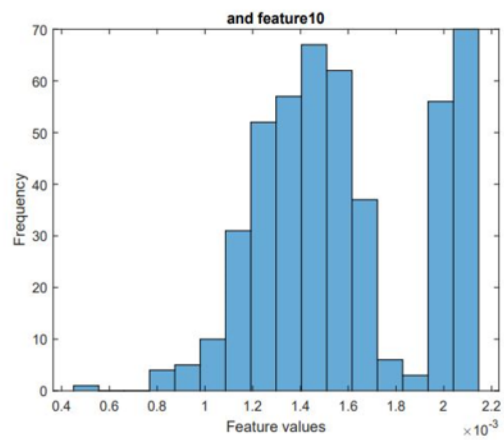

cop feature7
decide feature7
deaf feature7
about feature7
Frequency
Feature values
×10^-4

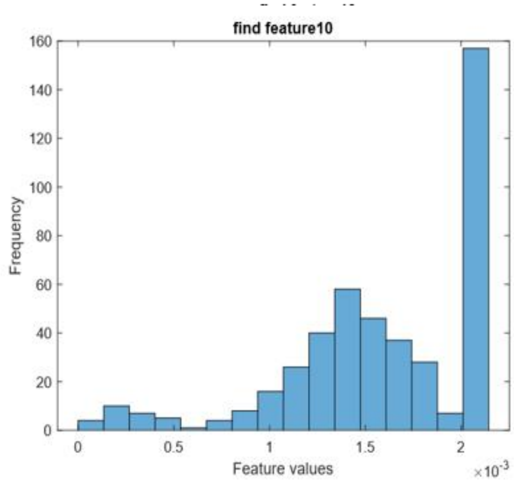
FEATURE 3: RMS of Accelerometer Right hand X axis

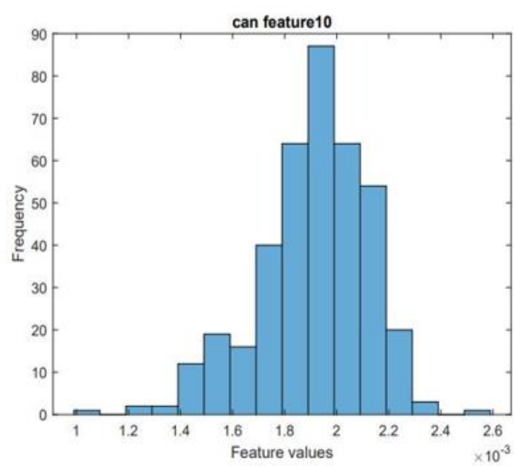


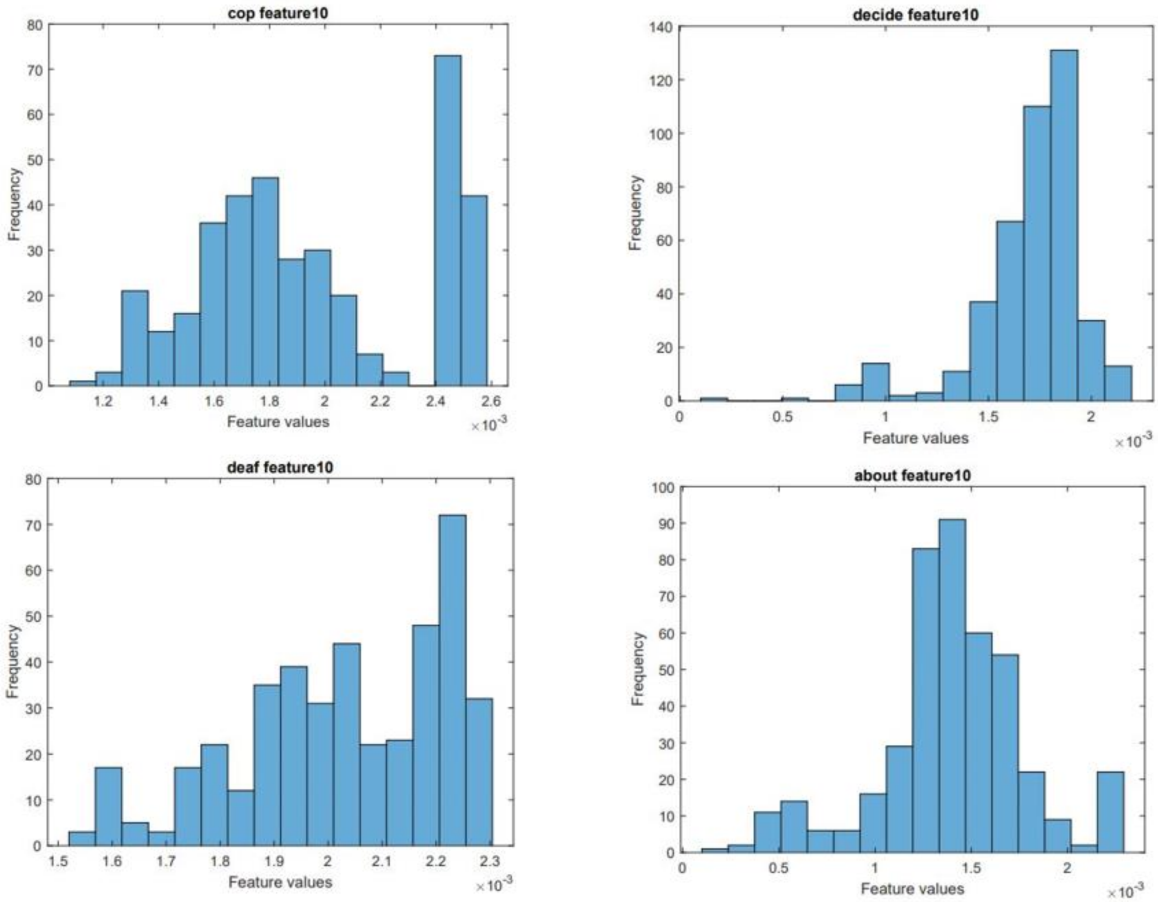




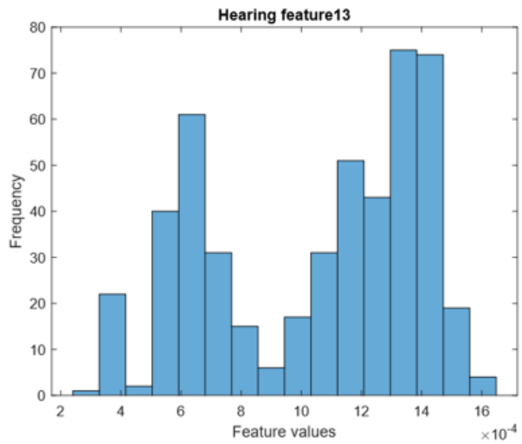


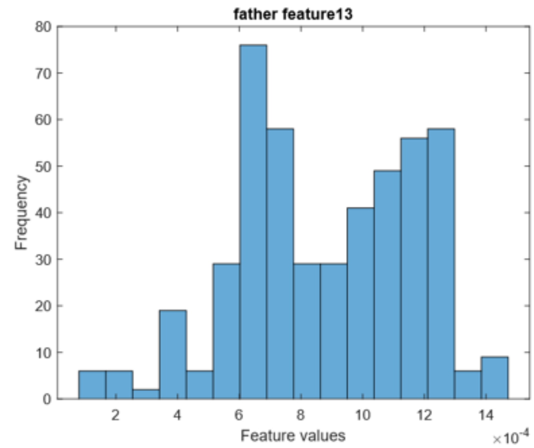


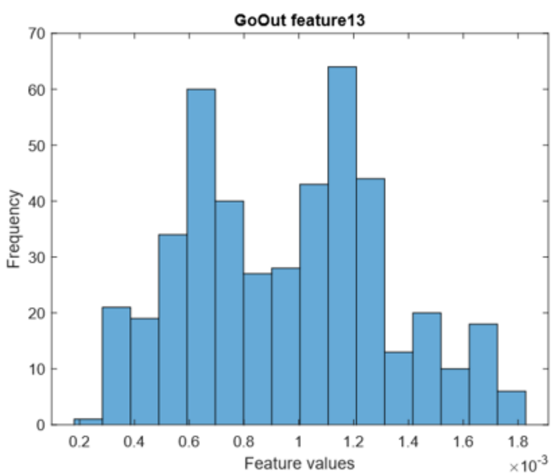


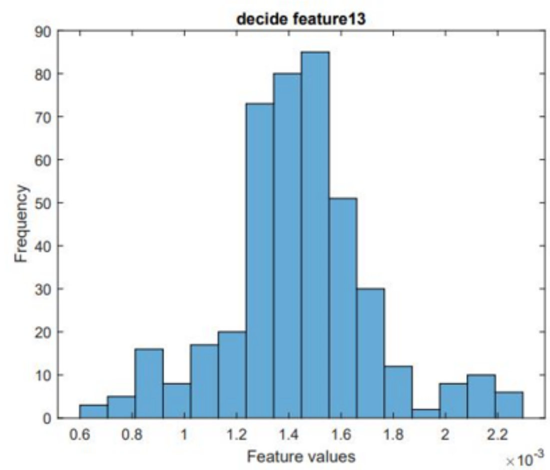


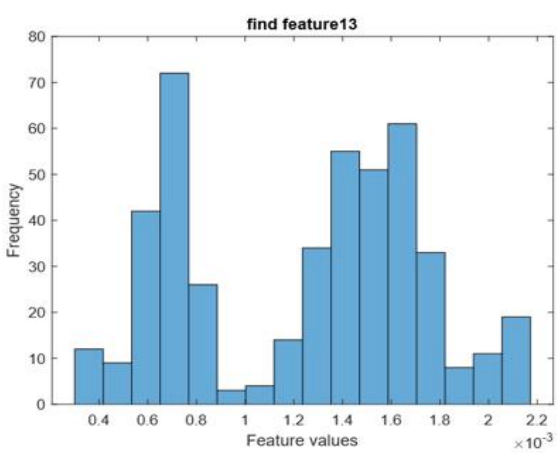
FEATURE 4: RMS of Accelerometer Y axis Right hand

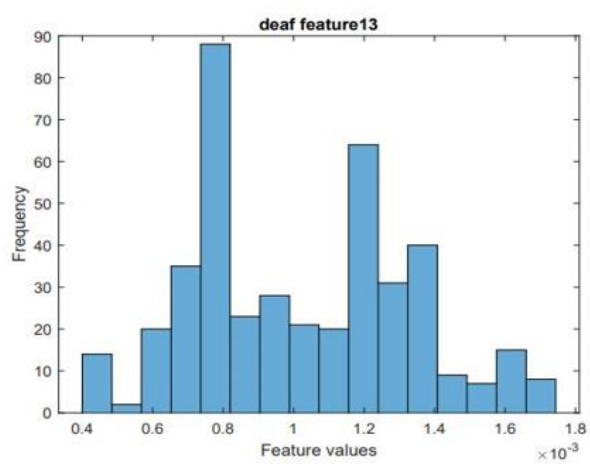

Hearing feature13
Frequency
Feature values
×10^-4

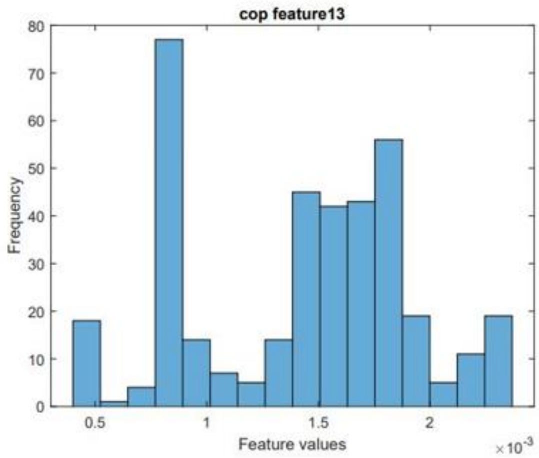

father feature13
Frequency
Feature values
×10^-4

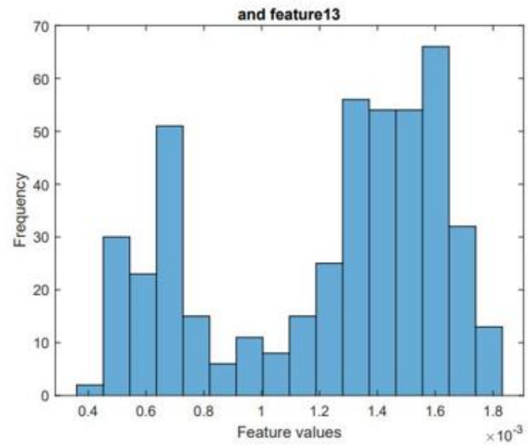

GoOut feature13
Frequency
Feature values
×10^-3

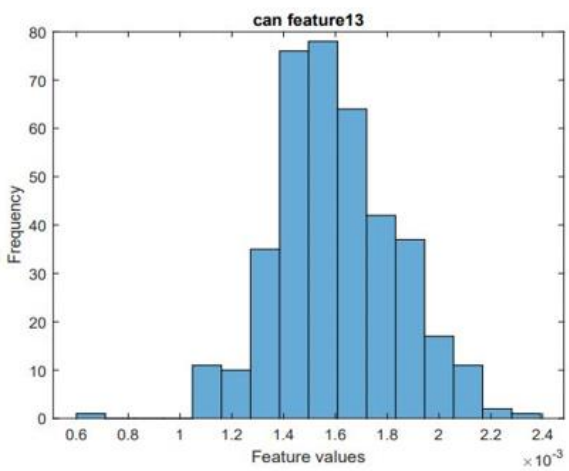

decide feature13
Frequency
Feature values
×10^-3

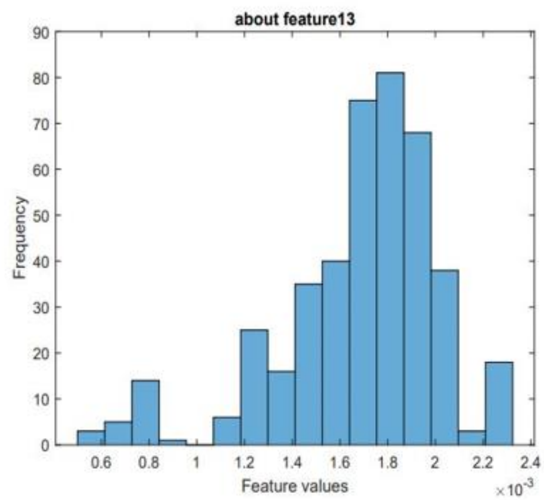

find feature13
Frequency
Feature values
×10^-3


deaf feature13
Frequency
Feature values
×10^-3

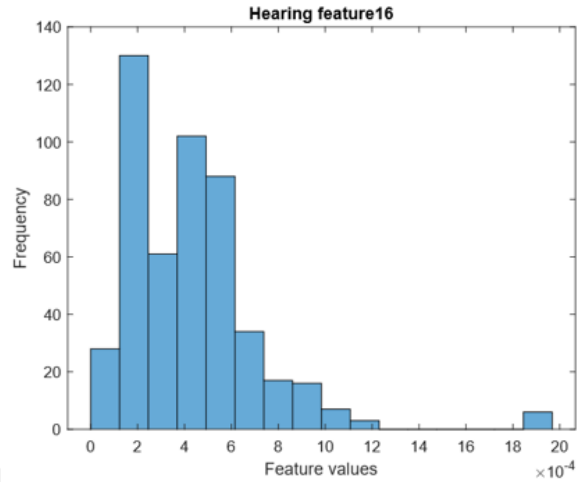


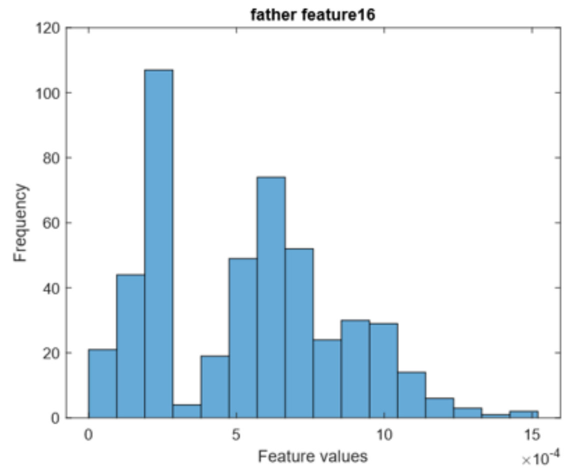


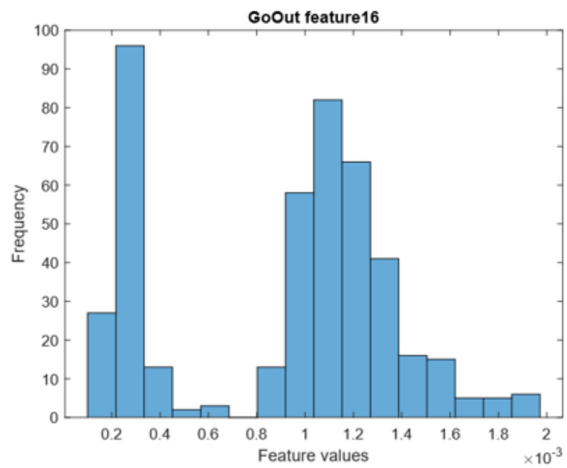


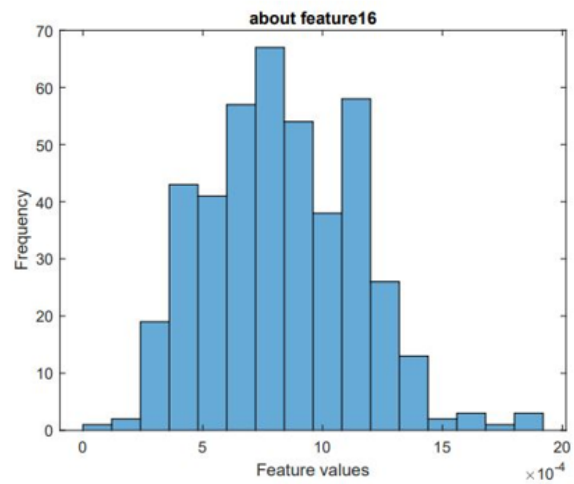


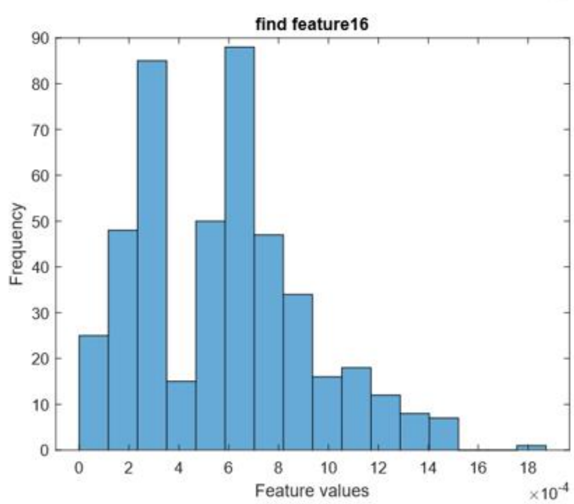
FEATURE 5: RMS of Accelerometer Right hand Z axis

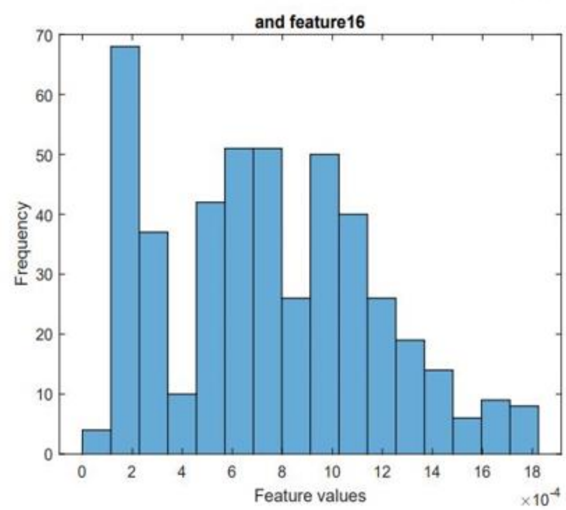

Hearing feature16
Frequency
Feature values
×10-4

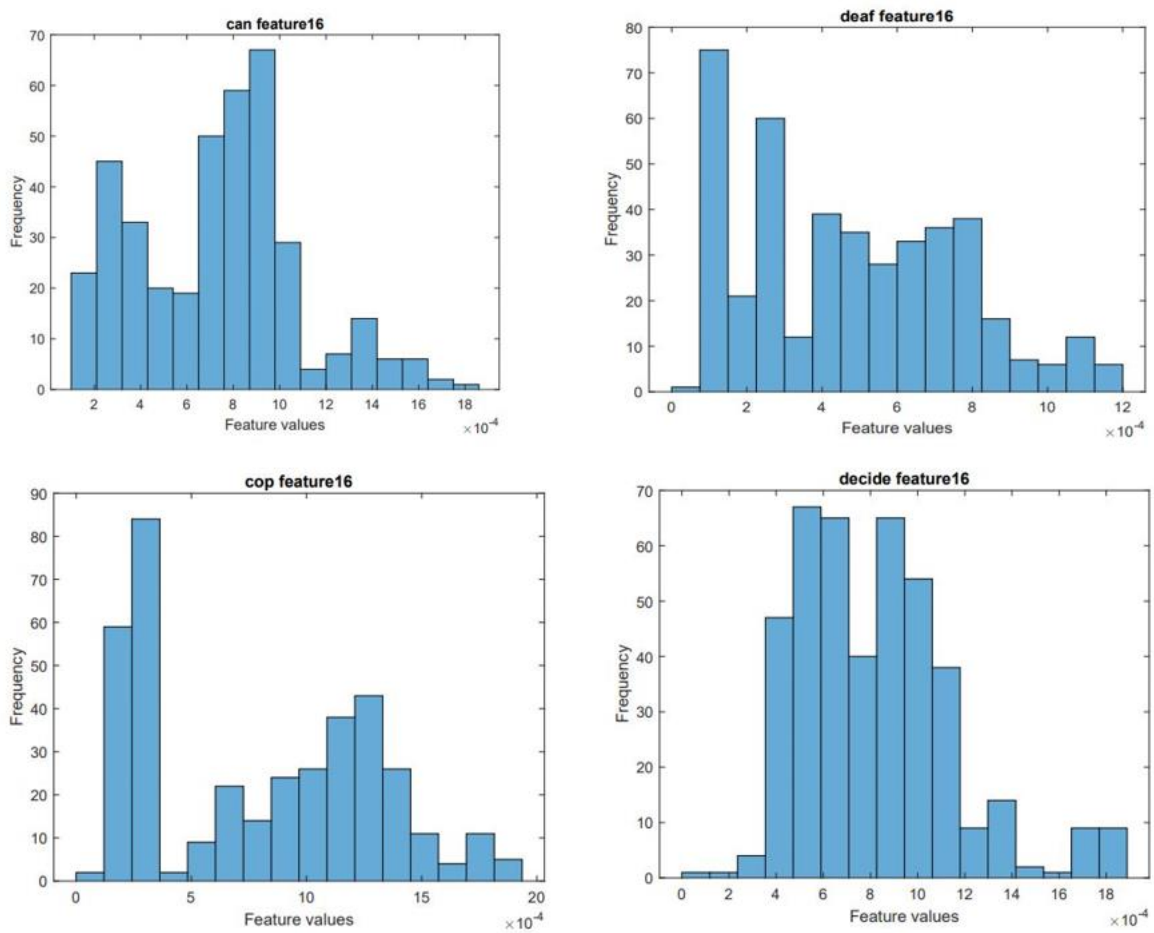

father feature16
Frequency
Feature values
×10-4


GoOut feature16
Frequency
Feature values
×10-3

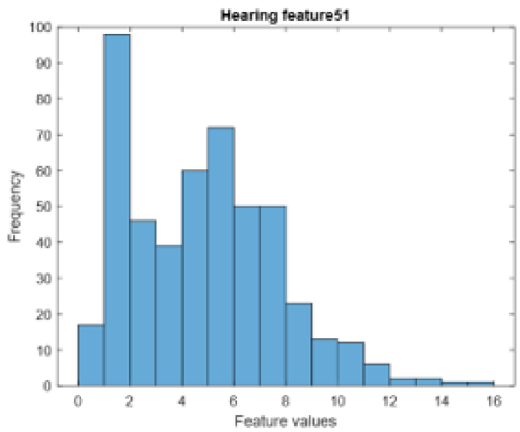

about feature16
Frequency
Feature values
×10-4

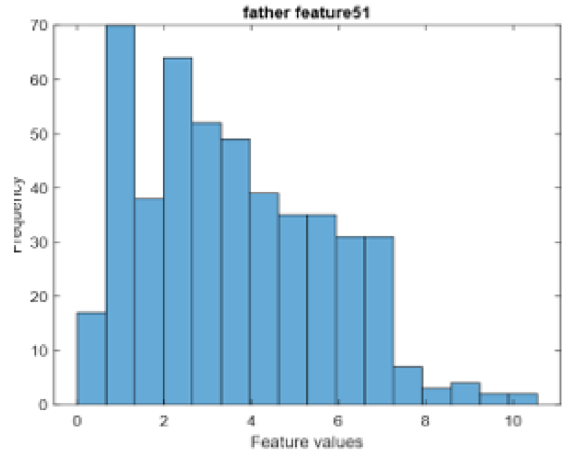

find feature16
Frequency
Feature values
×10-4

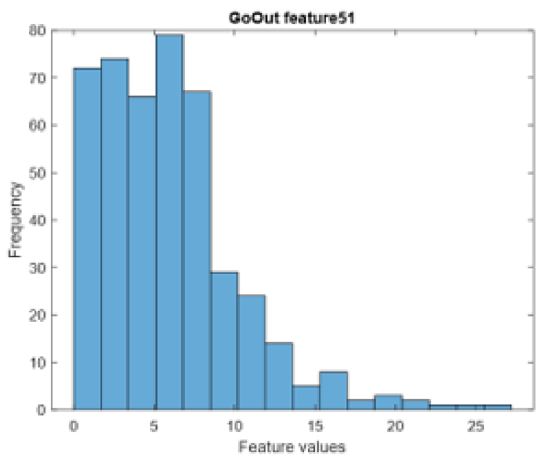

and feature16
Frequency
Feature values
×10-4

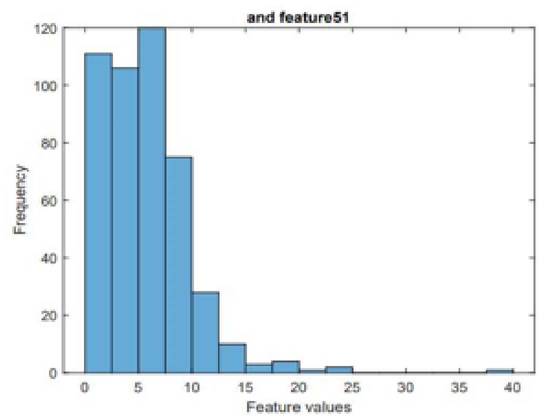


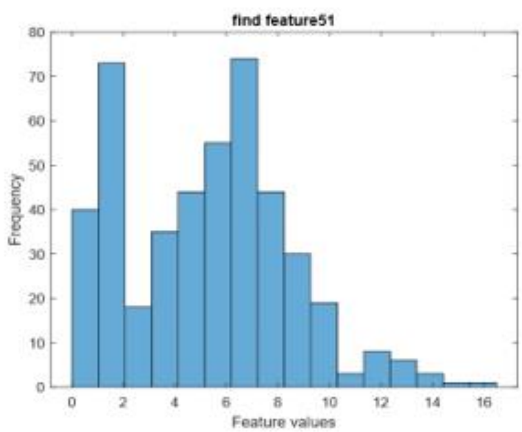
Feature 6: Mean Absolute Value for EMG 2 Right hand

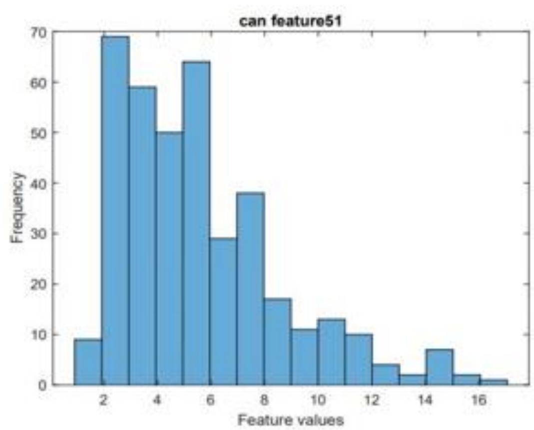

Hearing feature51
Frequency
Feature values

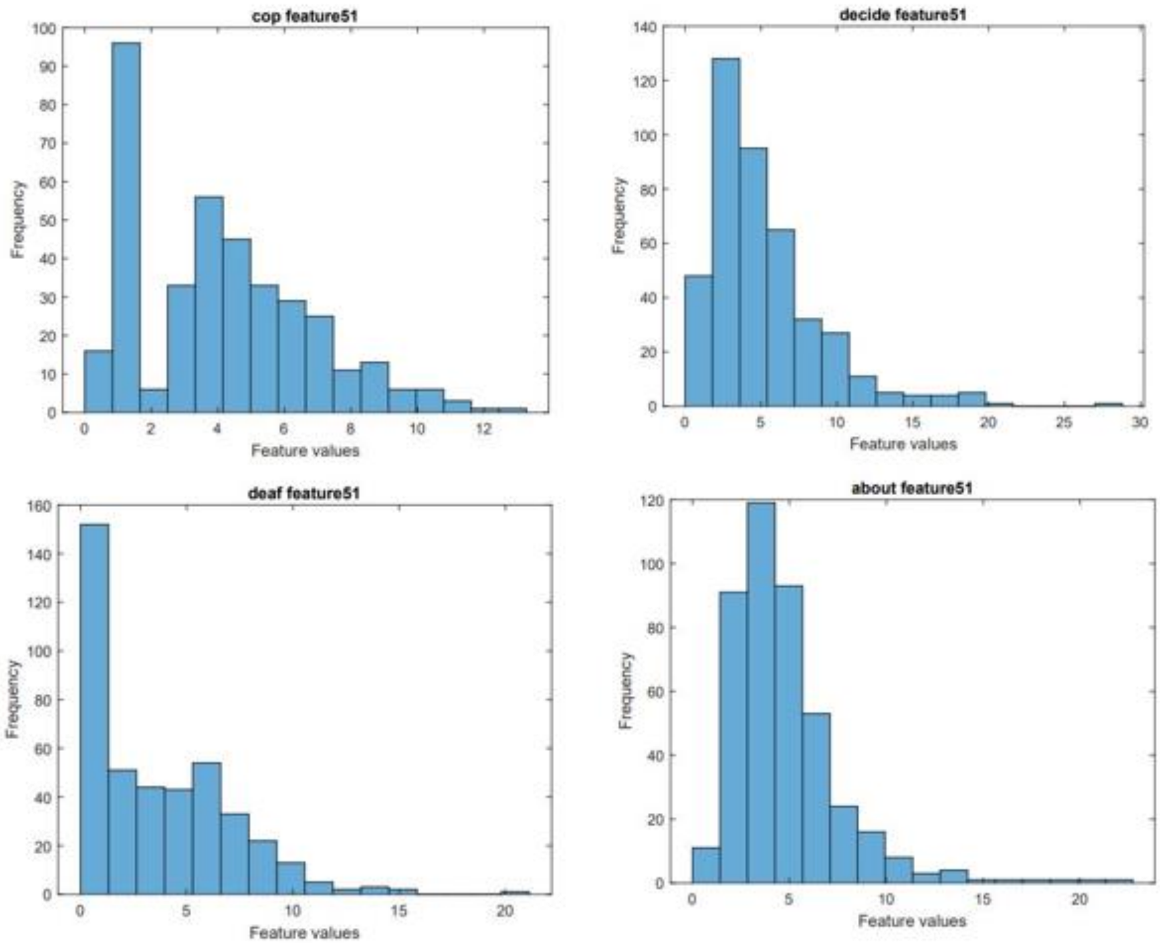

father feature51
Feature values


GoOut feature51
Frequency
Feature values

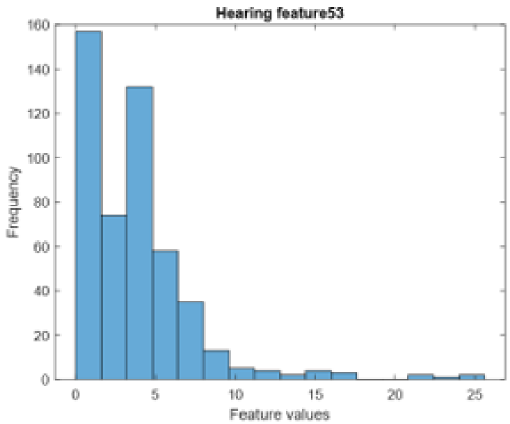

and feature51
Frequency
Feature values

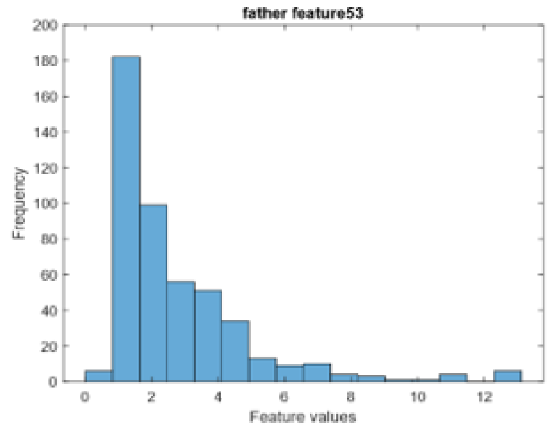

find feature51
Frequency
Feature values

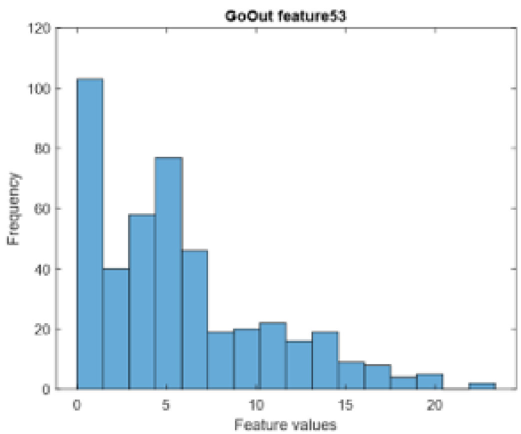

can feature51
Frequency
Feature values

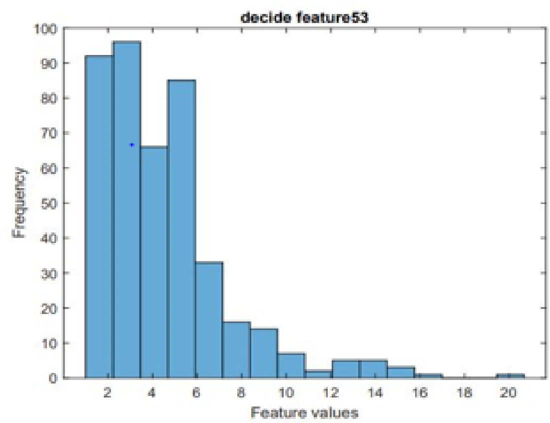


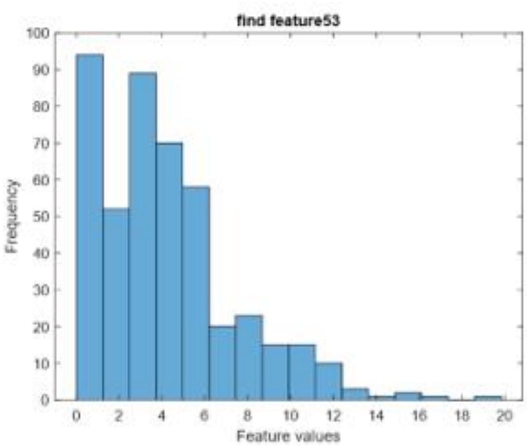
Feature 7: Mean Absolute Value for EMG 3 Right Hand

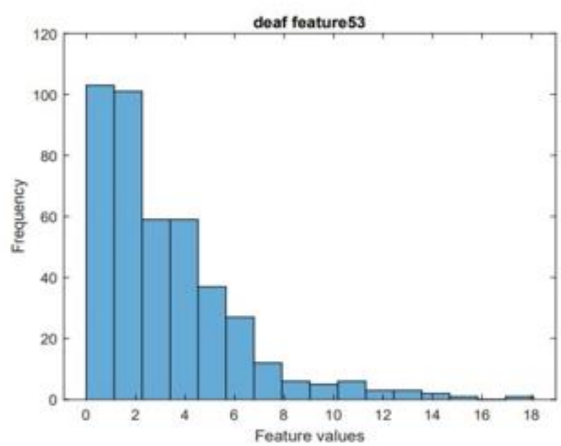

Hearing feature53
Frequency
0
20
40
60
80
100
120
140
160
0
5
10
15
20
25
Feature values

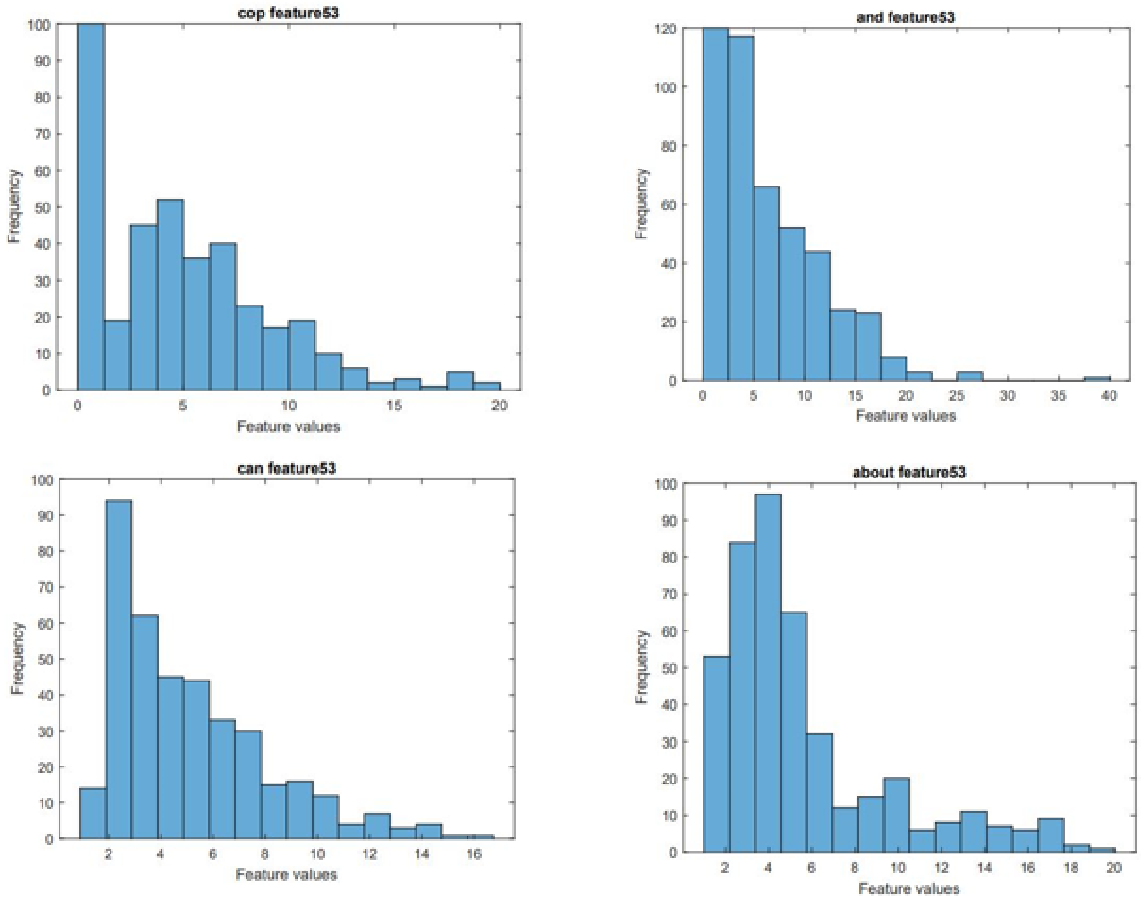

father feature53
Frequency
0
20
40
60
80
100
120
140
160
180
200
0
2
4
6
8
10
12
Feature values


GoOut feature53
Frequency
0
20
40
60
80
100
120
0
5
10
15
20
Feature values

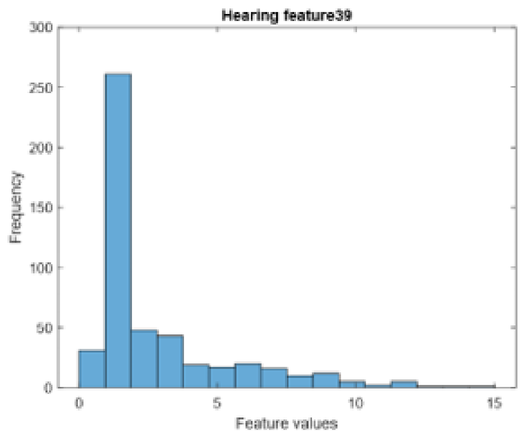

decide feature53
Frequency
0
10
20
30
40
50
60
70
80
90
100
2
4
6
8
10
12
14
16
18
20
Feature values

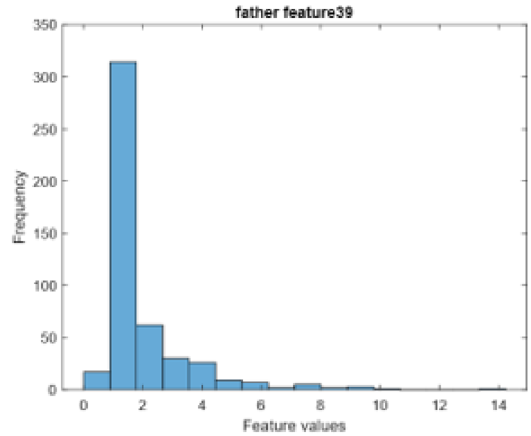

find feature53
Frequency
0
10
20
30
40
50
60
70
80
90
100
0
2
4
6
8
10
12
14
16
18
20
Feature values

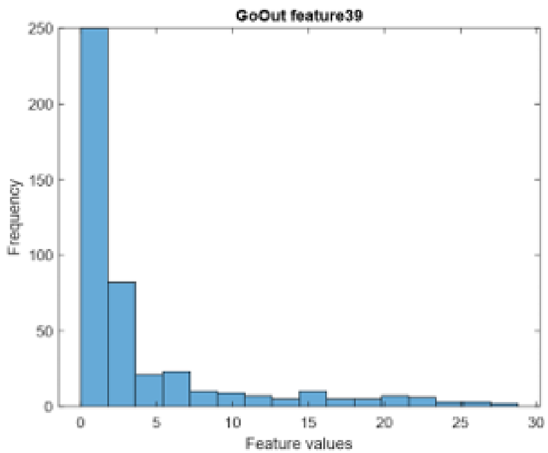

deaf feature53
Frequency
0
20
40
60
80
100
120
0
2
4
6
8
10
12
14
16
18
Feature values

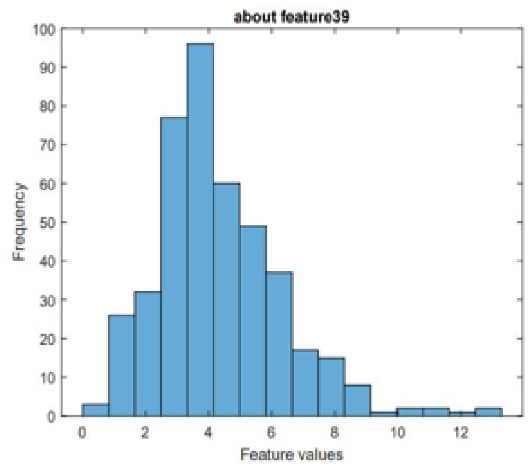


Feature 8: Mean Absolute Value for EMG 4 Left Hand

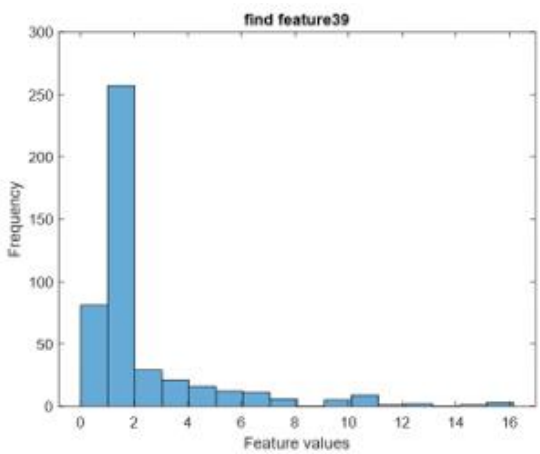


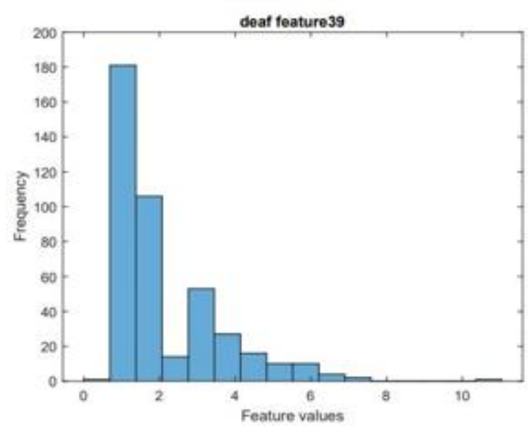


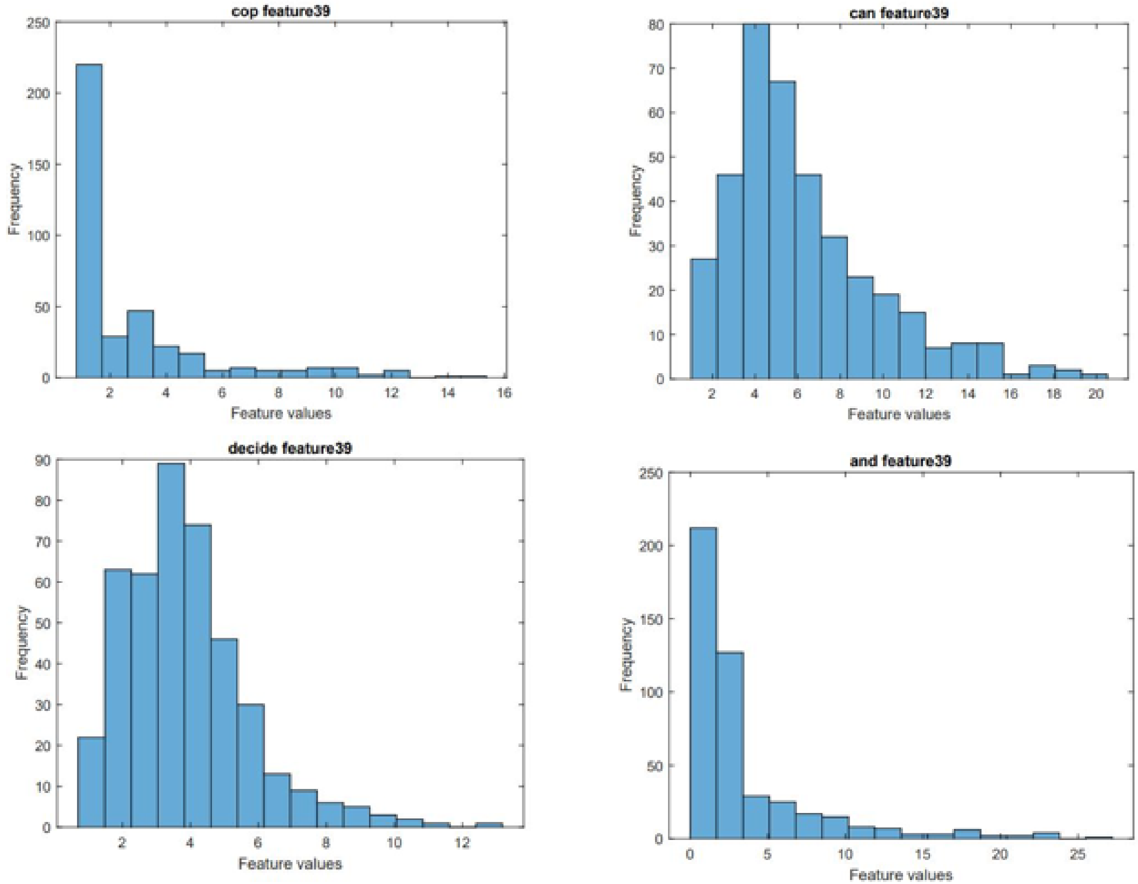




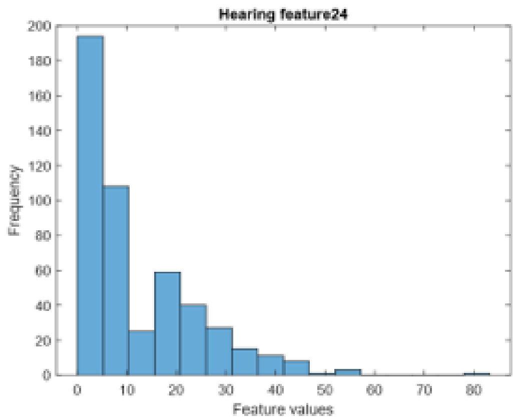
Feature values

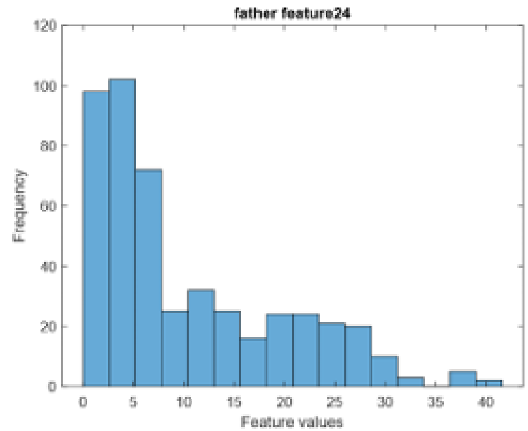


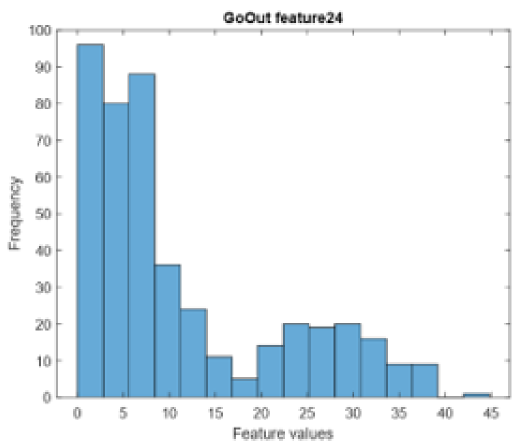


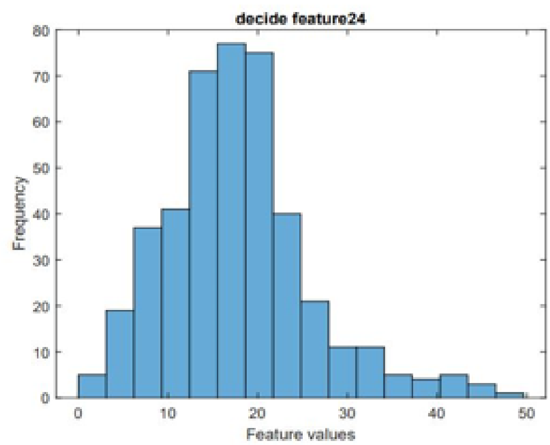


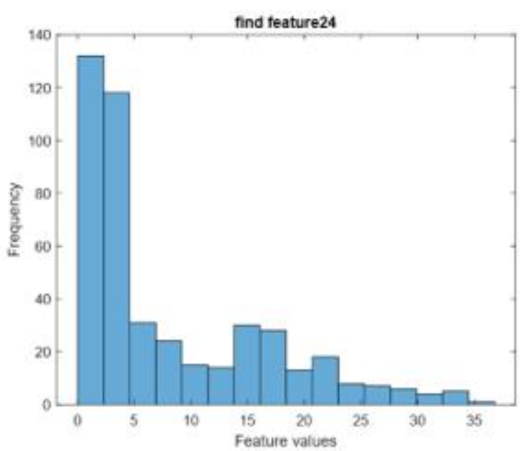
Feature 9: Mean for Gyroscope Y axis Left Hand

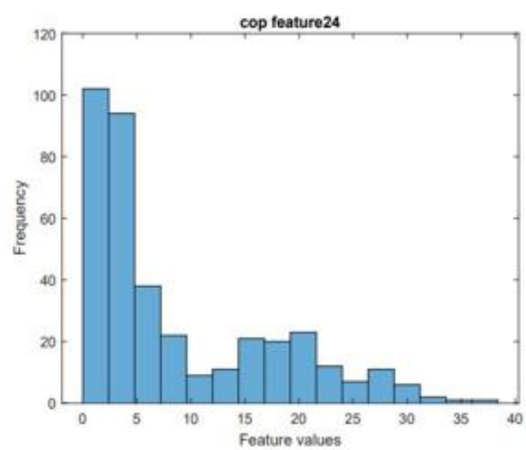

Hearing feature24
Frequency
Feature values

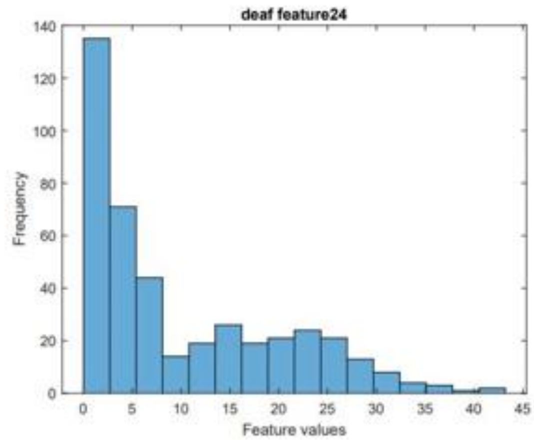

father feature24
Frequency
Feature values

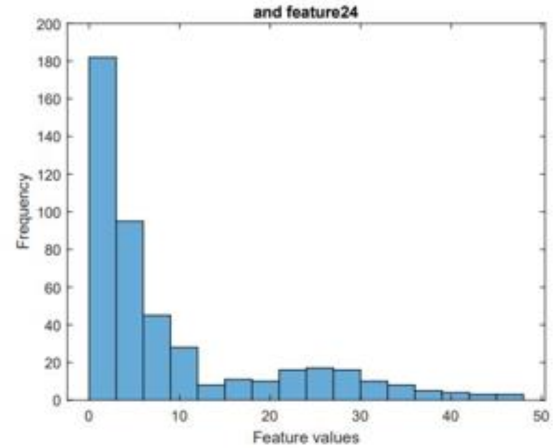

GoOut feature24
Frequency
Feature values

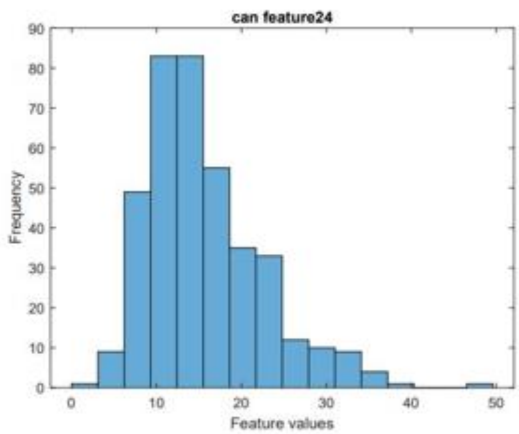

decide feature24
Frequency
Feature values

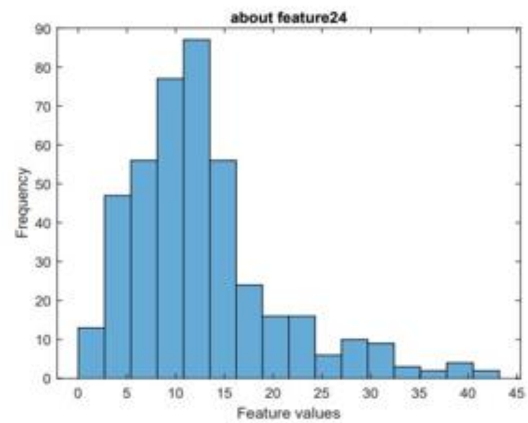

find feature24
Frequency
Feature values


cop feature24
Frequency
Feature values

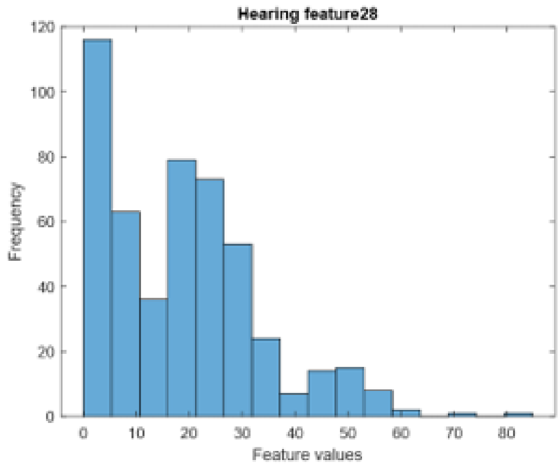


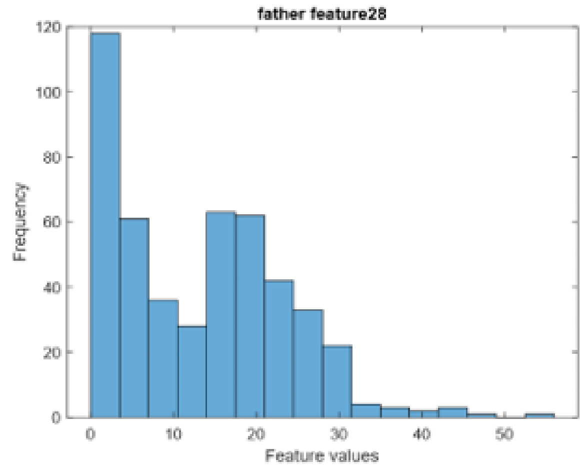


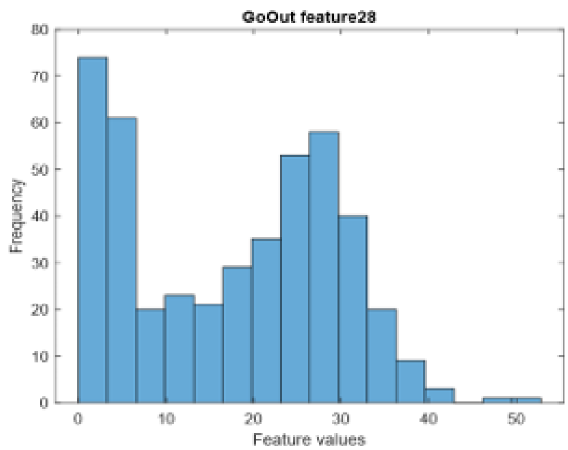


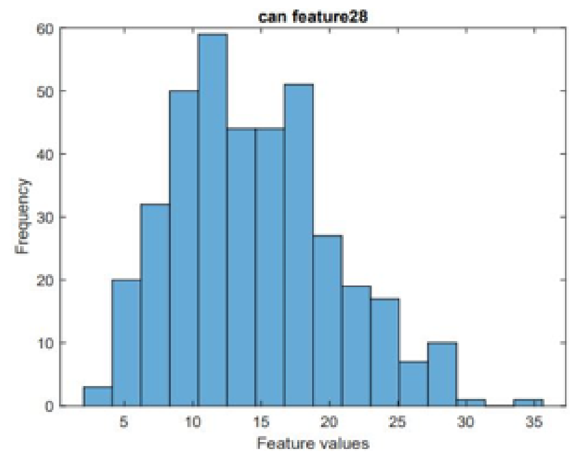


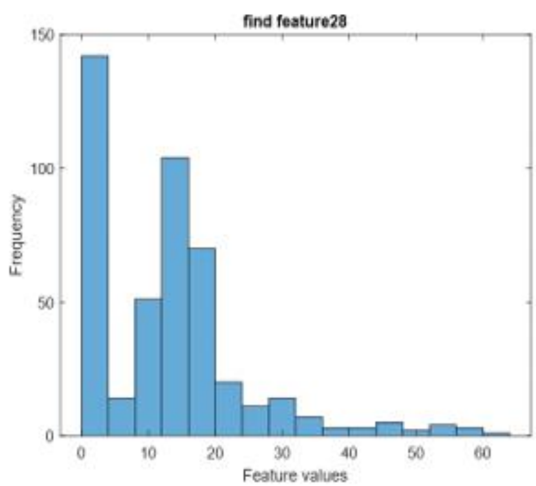
Feature 10: Mean for Gyroscope Y axis Right hand

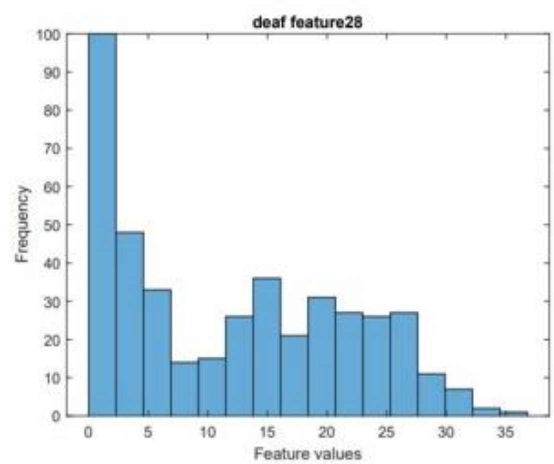

Hearing feature28
Frequency
Feature values

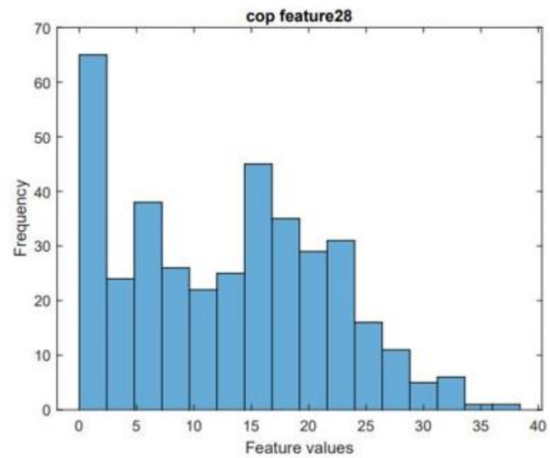

father feature28
Frequency
Feature values

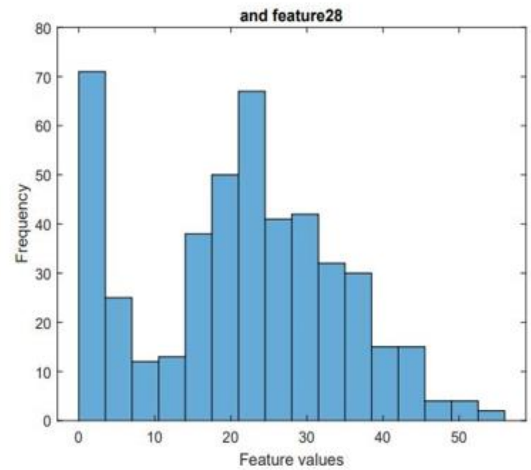

GoOut feature28
Frequency
Feature values

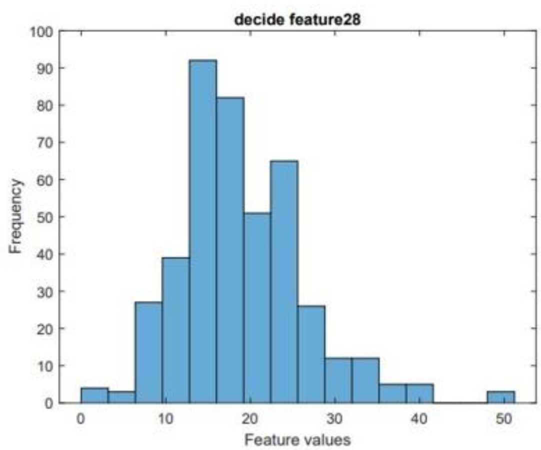

can feature28
Frequency
Feature values

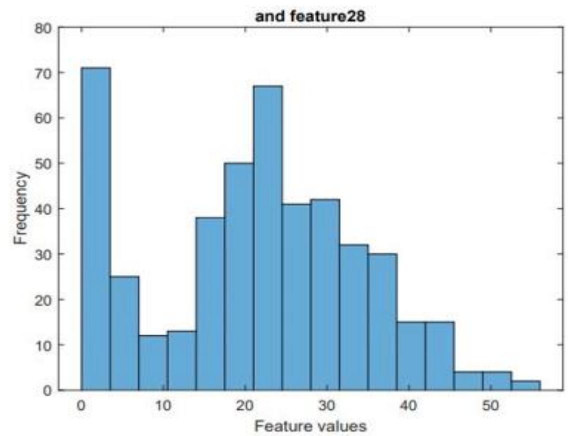

find feature28
Frequency
Feature values


deaf feature28
Frequency
Feature values









Task 3: Feature Selection:

PCA is a method for dimensionality reduction. The transformation **T** = **X** **W** maps a data vector $\mathbf{x}_{(i)}$ from an original space of p variables to a new space of p variables which are uncorrelated over the dataset. However, not all the principal components need to be kept. Keeping only the first L principal components, produced by using only the first L loading vectors, gives the truncated transformation.

Subtask 1: Arranging the feature matrix

The feature matrix is of the form **n** x **m** where **n** is the number of actions performed and **m** is the number of features in our feature matrix. The feature matrix is obtained by concatenating the features obtained using every sensor. PCA arranges eigenvectors in decreasing order of contribution of variance.

Subtask 2: Execution of PCA

We have plotted the biplot of first two eigenvectors and found certain features to be dominant than the others. Additionally, these features are found in the biplot of first 8 eigenvectors. The biplots for different features are given below.

Biplots:

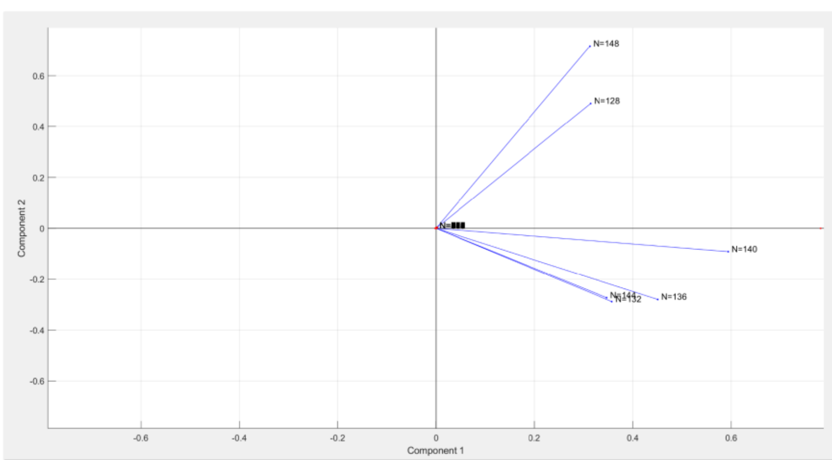


Fig: About

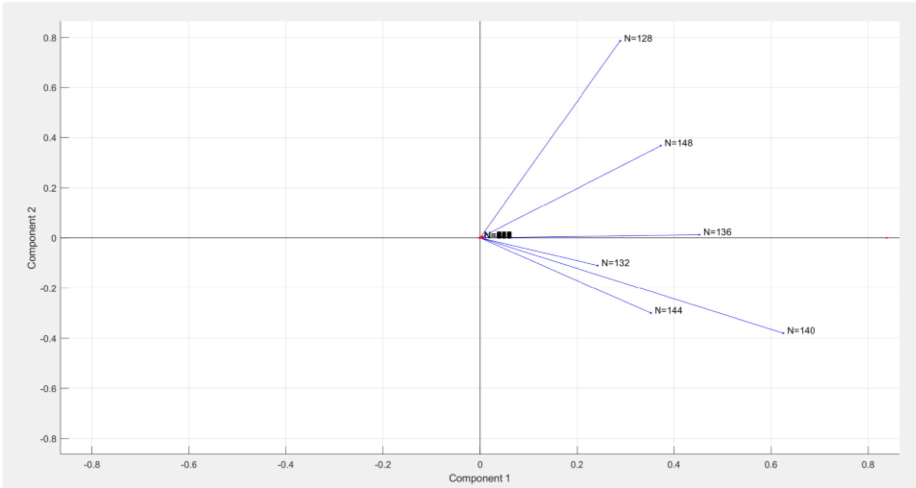


Fig: And

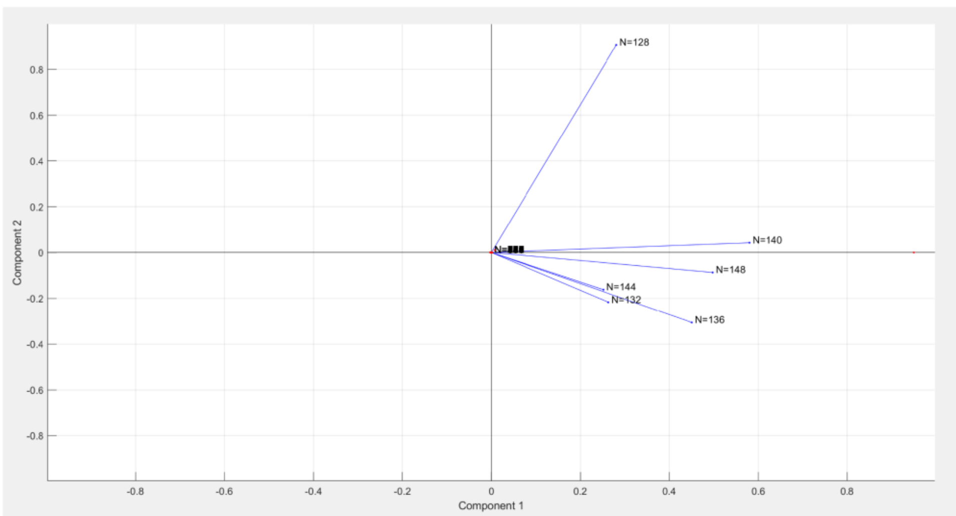


Fig: Can

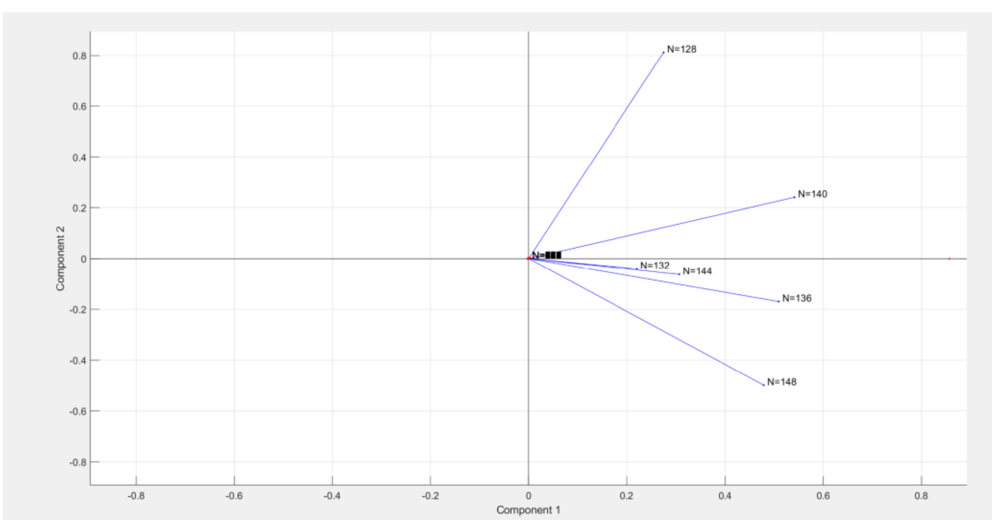


Fig: Cop

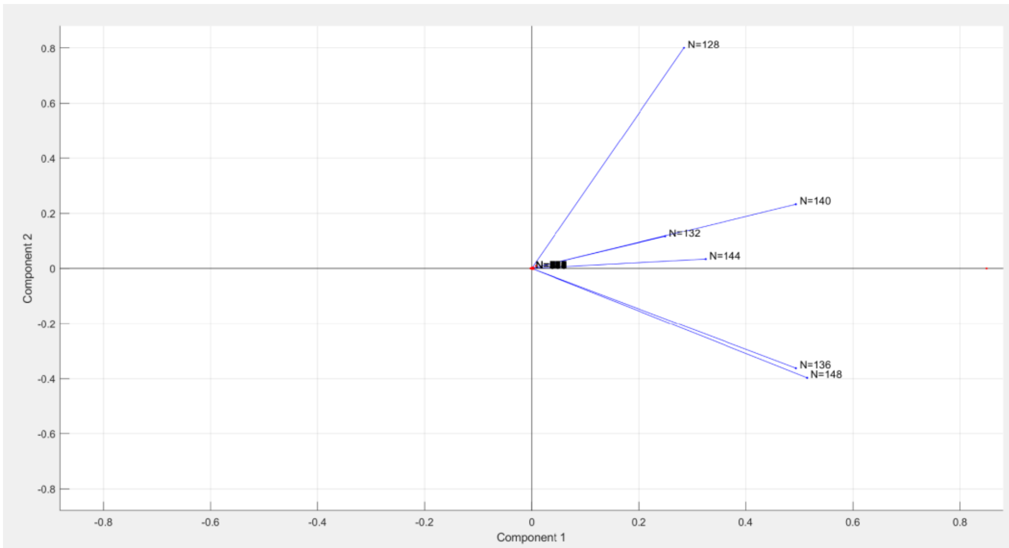


Fig: Deaf

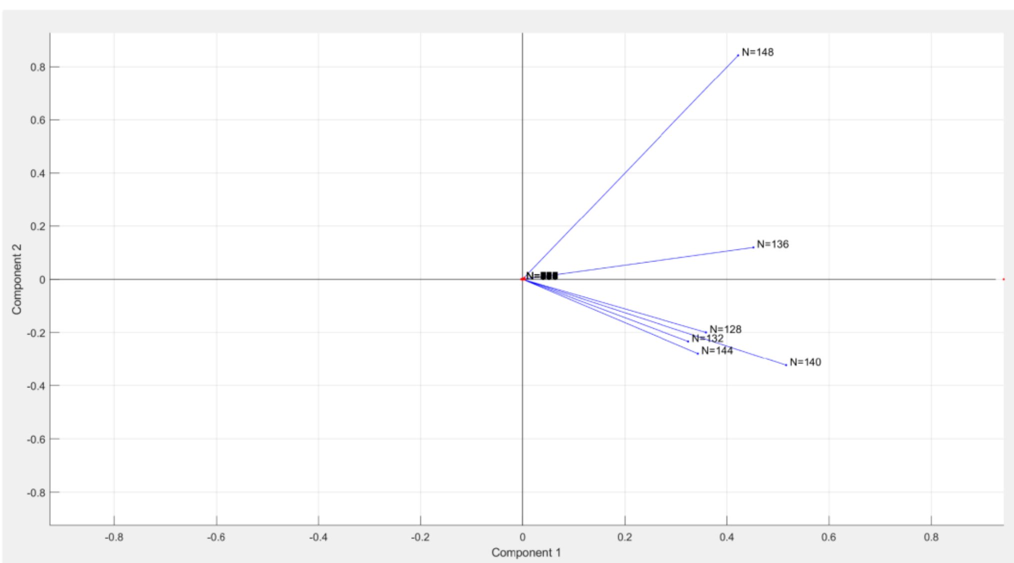


Fig: Decide

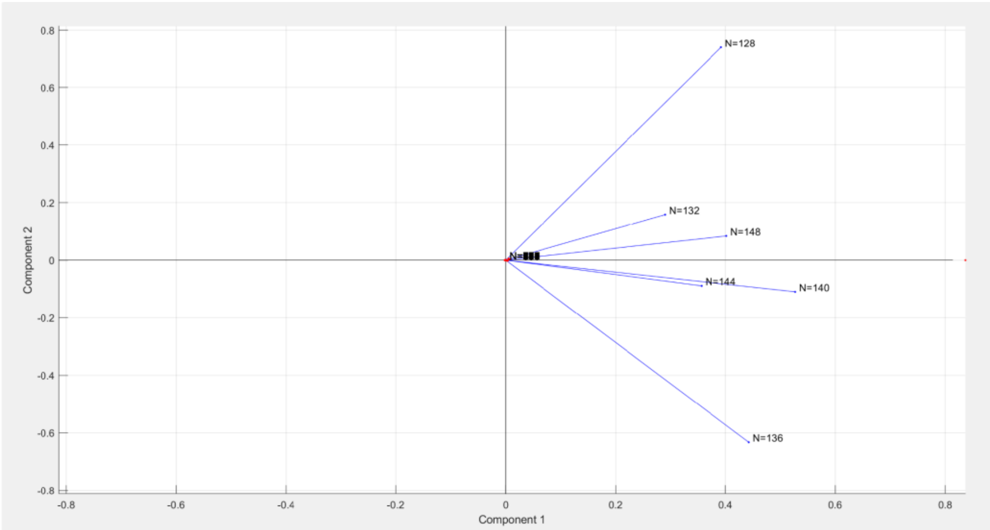


Fig: Father

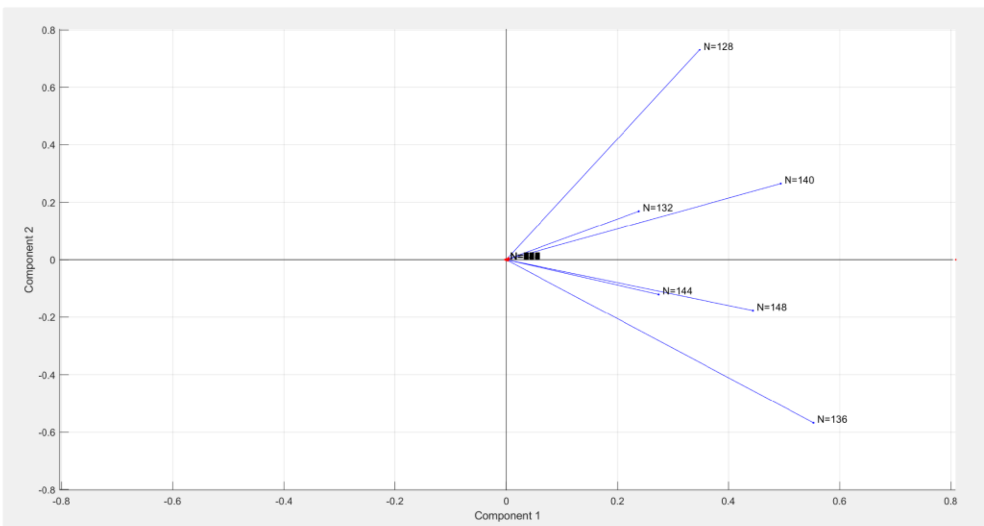


Fig: Find

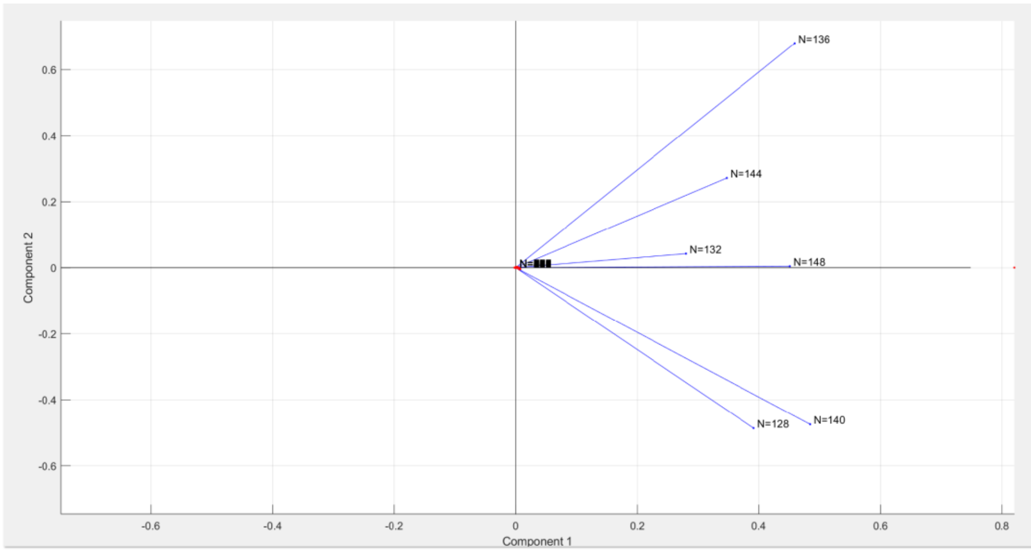
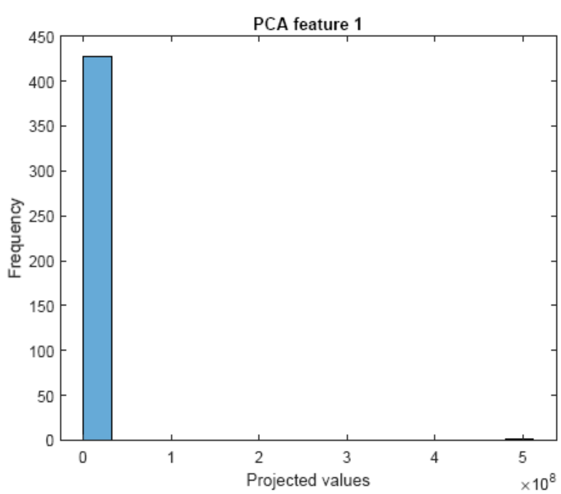
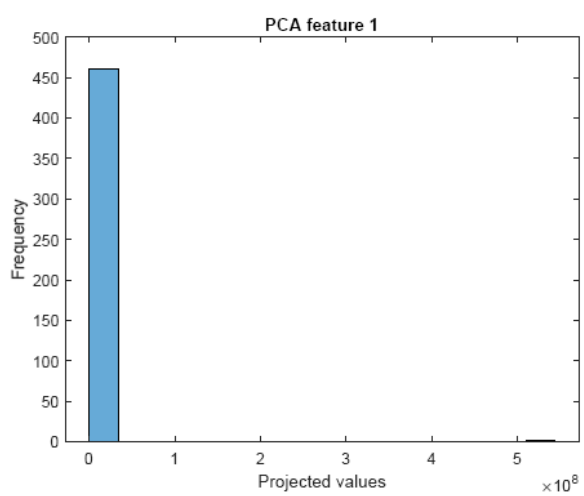


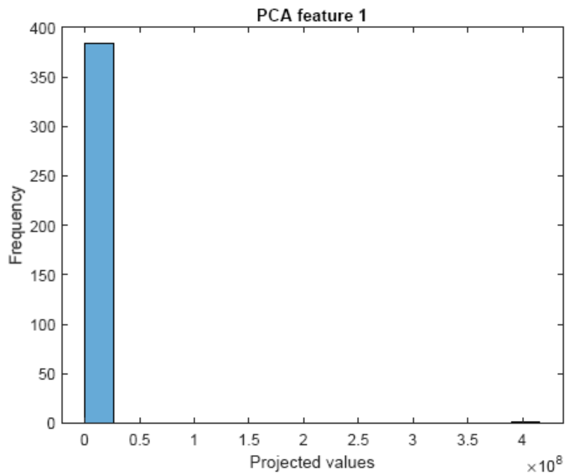
Fig: Hearing

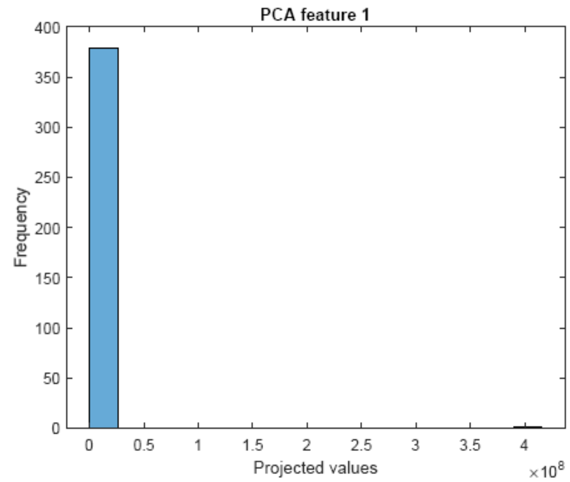
SUBTASK 4:

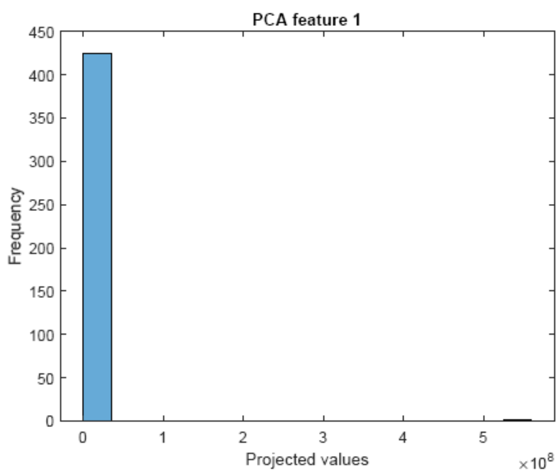
1. Feature 1

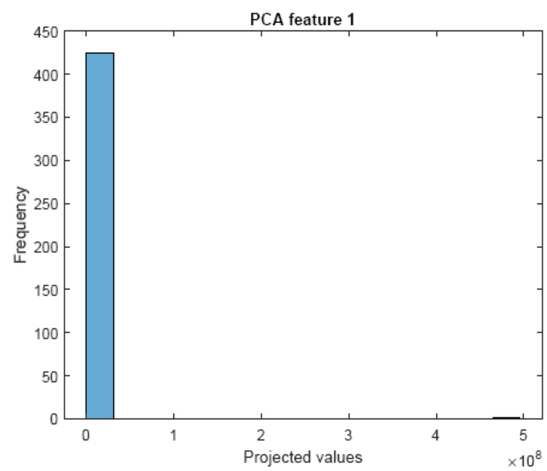


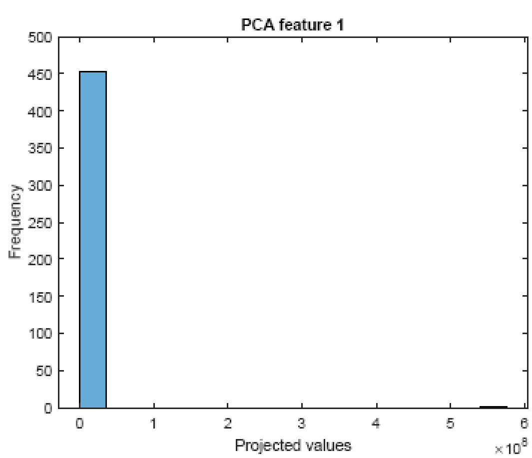


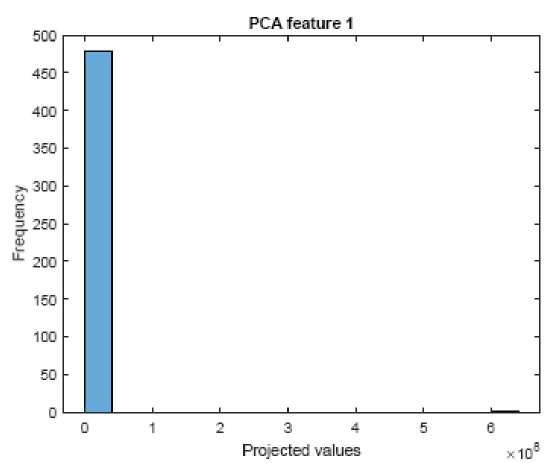

PCA feature 1
Frequency
Projected values
$\times 10^8$

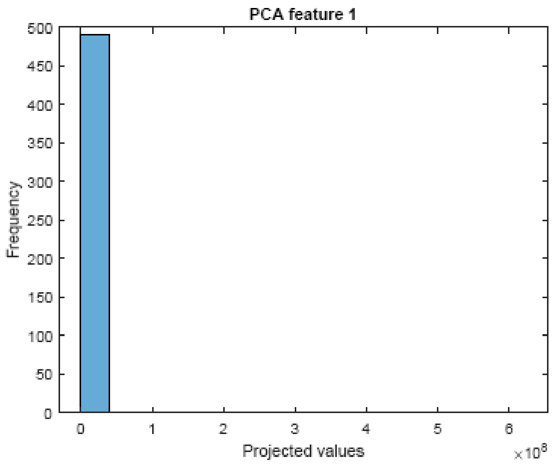

PCA feature 1
Frequency
Projected values
$\times 10^8$


PCA feature 1
Frequency
Projected values
$\times 10^8$

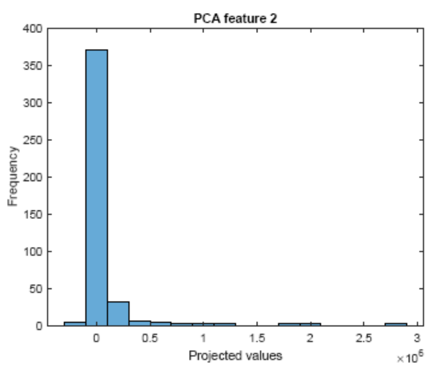

PCA feature 1
Frequency
Projected values
$\times 10^8$

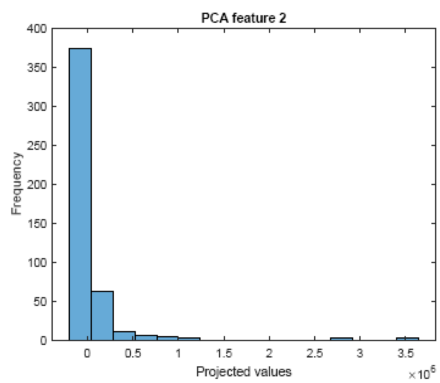

PCA feature 1
Frequency
Projected values
$\times 10^8$

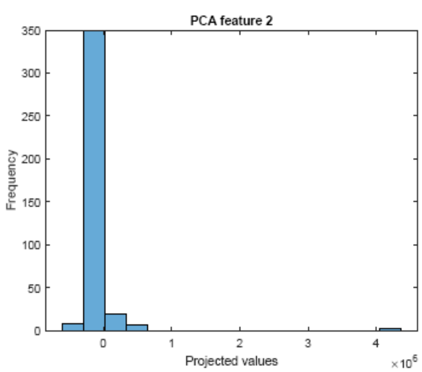

PCA feature 1
Frequency
Projected values
$\times 10^8$

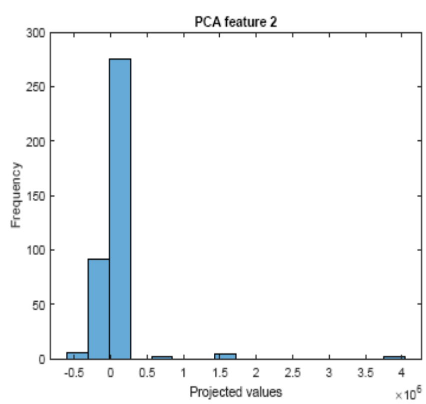

PCA feature 1
Frequency
Projected values
×10^8

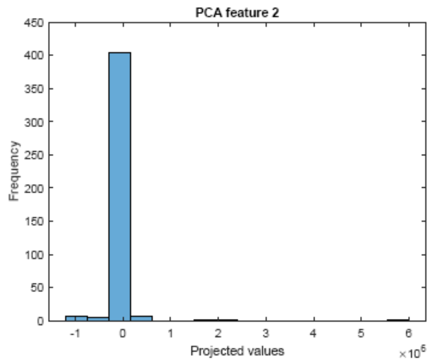
2.FEATURE 2:

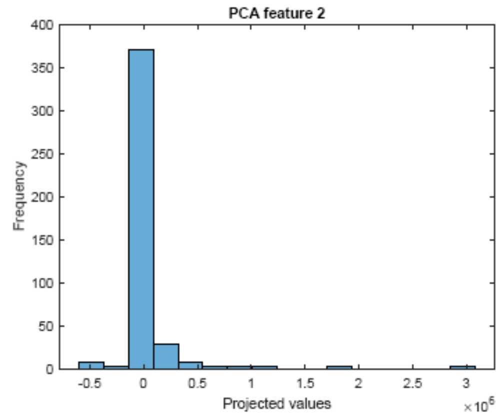

PCA feature 2
Frequency
Projected values

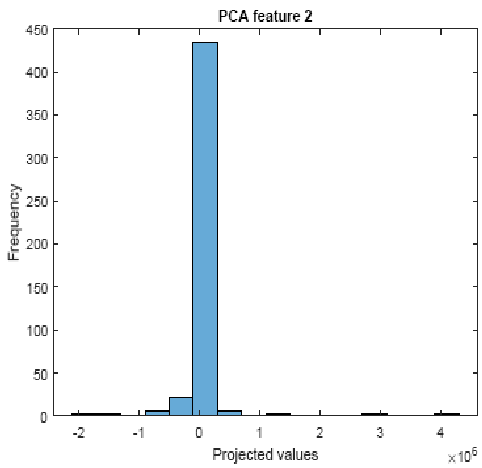

PCA feature 2
Frequency
Projected values

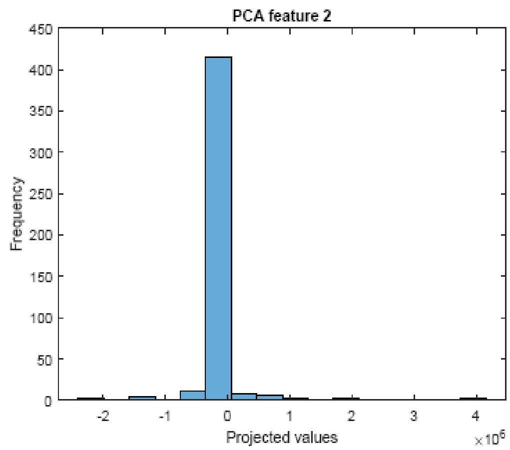

PCA feature 2
Frequency
Projected values

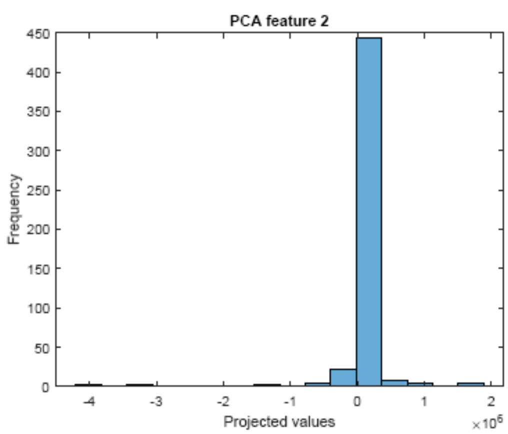

PCA feature 2
Frequency
Projected values


PCA feature 2
Frequency
0 50 100 150 200 250 300 350 400 450
-1 0 1 2 3 4 5 6
Projected values
×10^6

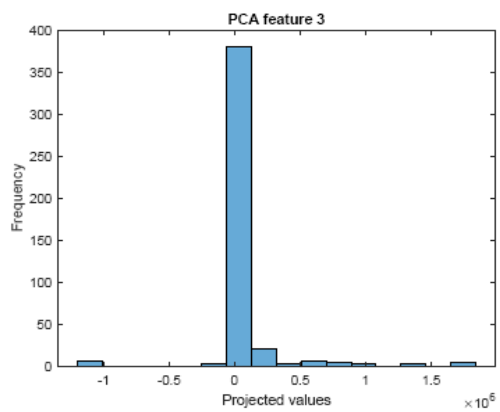

PCA feature 2
Frequency
0 50 100 150 200 250 300 350 400
-0.5 0 0.5 1 1.5 2 2.5 3
Projected values
×10^6

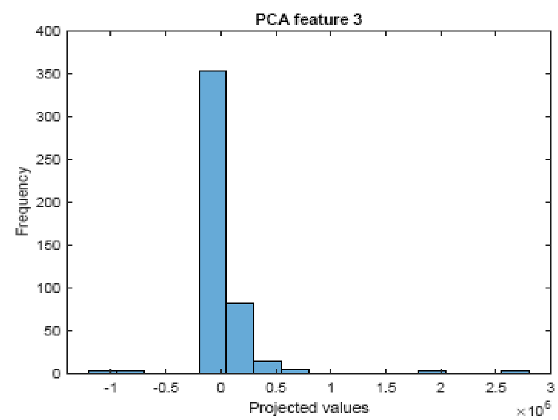

PCA feature 2
Frequency
0 50 100 150 200 250 300 350 400 450
-2 -1 0 1 2 3 4
Projected values
×10^6

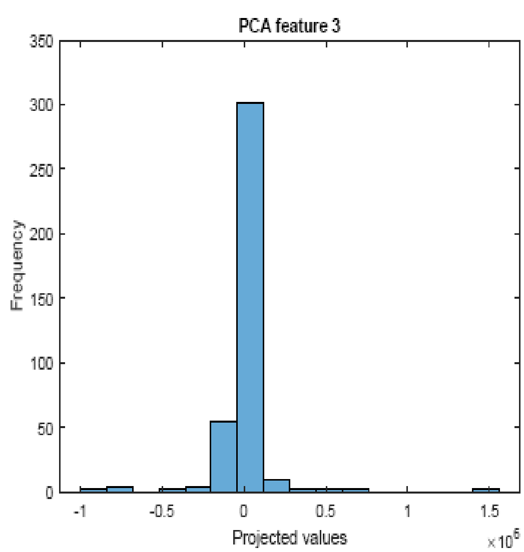

PCA feature 2
Frequency
0 50 100 150 200 250 300 350 400 450
-2 -1 0 1 2 3 4
Projected values
×10^6

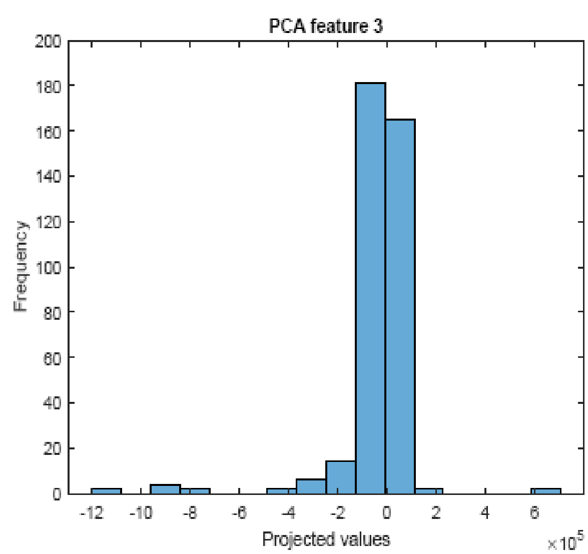

PCA feature 2
Frequency
0 50 100 150 200 250 300 350 400 450
-4 -3 -2 -1 0 1 2
Projected values
×10^6

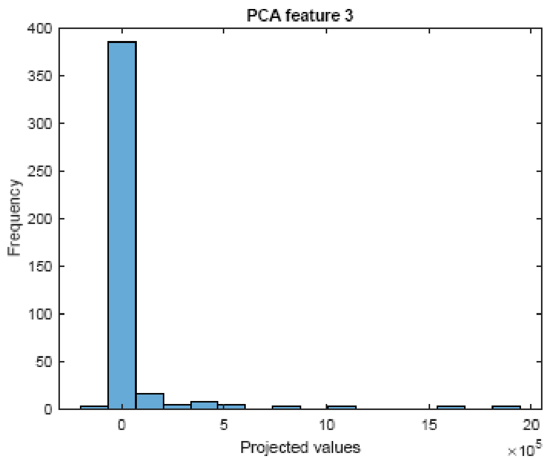
Feature 3:

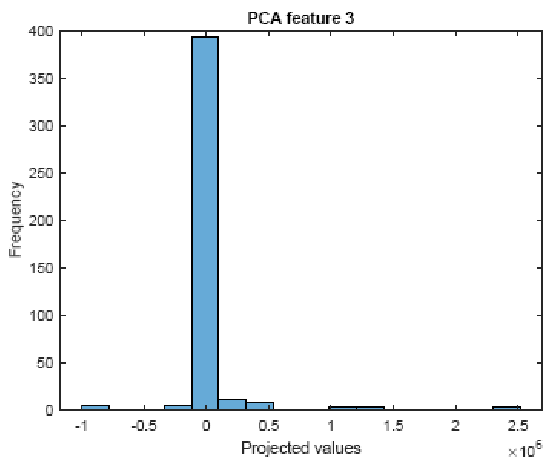

PCA feature 3
Frequency
Projected values
×10^6

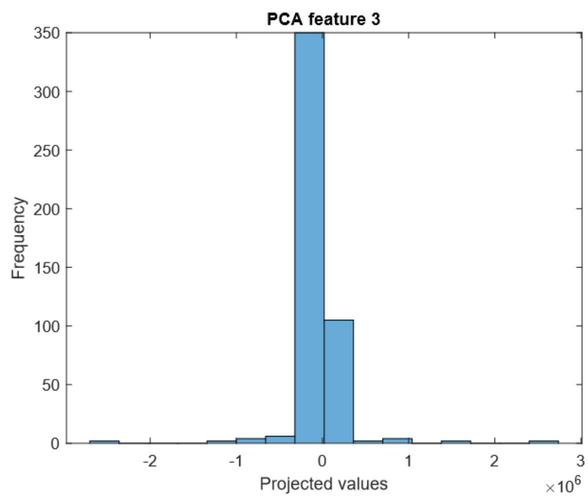

PCA feature 3
Frequency
Projected values
×10^6

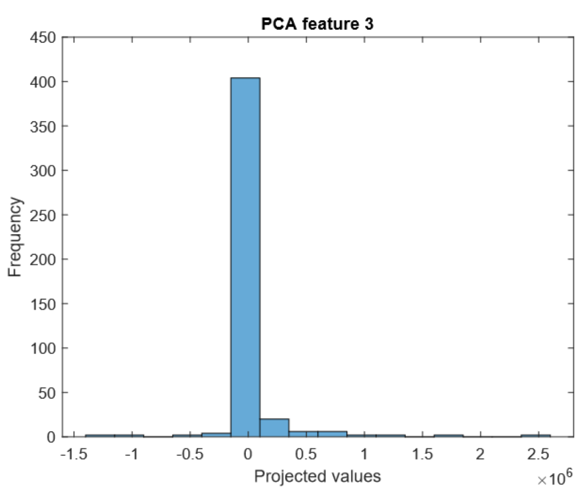

PCA feature 3
Frequency
Projected values
×10^6

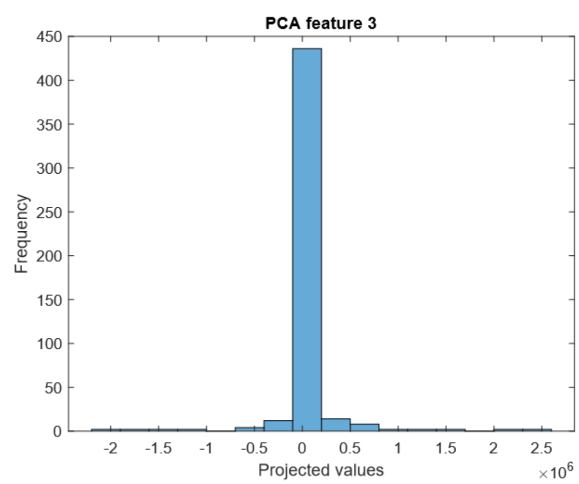

PCA feature 3
Frequency
Projected values
×10^5


PCA feature 3
Frequency
Projected values
$\times 10^5$


PCA feature 3
Frequency
Projected values
$\times 10^6$


PCA feature 3
Frequency
Projected values
$\times 10^6$


PCA feature 3
Frequency
Projected values
$\times 10^6$


PCA feature 3
Frequency
Projected values
$\times 10^6$

Subtask 3: Make sense of the PCA eigenvectors

From PCA eigenvectors we find that energy function on fft of orientation of right hand and left hand is very important. Since, PCA chooses the direction with maximum variance, it so happens that orientation happens to provide that. In all of the gestures we find that orientation varies from gesture to gesture.

Subtask 4: Results of PCA

Projection matrix is generated by multiplying the eigenvector with the feature matrix. The top 4 features of the projection matrix are pasted below:

Subtask 5: Usefulness of PCA and Comparison between feature matrix in task 2 vs PCA results in task 3

PCA was very helpful to distinguish between gestures as it captures the feature combination with highest variance. However, by intuition as well we were able to distinguish between gestures very well. RMS of accelerometer Z and variance of gyroscope right hand were able to nicely distinguish between different gestures.