Project Phase II Report

Data Mining

CSE 572

Submitted to:

Professor Ayan Banerjee

Ira A. Fulton School of Engineering

Arizona State University

Submitted By:

Abhijith Shreesh (ashreesh@asu.edu)

Abishek Ravichandran (aravic14@asu.edu)

Aditya Chayapathy (achayapa@asu.edu)

Chandni Shrivatsava (cshrivas@asu.edu)

Kevin Thomas (kthoma46@asu.edu)

Table Of Contents

Introduction

This project is a part of CSE 572 - Data Mining course for Spring 2018 semester. It is an attempt at developing a computing system that can understand human gestures based on American Sign Language. We aim to do this in three parts:

- record and identify a set of known gestures
- segment sequence of gestures
- identify unknown gestures based on our previous observations

Identification of the gestures was done by aggregating the annotations into separate csv files based on gestures and apply feature extraction methods on each of them to build a distinction amongst the different gesture classes. Subsequently PCA will be applied on selected features that represent maximum distinction between the classes.

Project Phase I

The main task pertaining to Phase I, was data collection. One person from the group had to perform each action twenty times by wearing a wristband sensor and in front of a Kinect. The wristband sensor records the Accelerometer, Gyroscope, EMG and Orientation sensor values pertaining to each action, whilst the Kinect records the video of the same. The sensor values are regularly recorded and stored in a comma separated values file.

Project Phase II

This phase involves feature extraction and application of Principal Component Analysis (PCA) of the Phase I data collected. The number of raw sensor data used are as follows:

a) 6 from Accelerometer,
b) 6 from Gyroscope,
c) 6 from Orientation and,
d) 16 from EMG sensors.

The following tasks were performed in this phase of the project:

Task I: Transformation of Input Data Files into Gesture CSVs

Among the thirty-seven folders of data provided to us, we chose ten folders of data as our dataset i.e. *DM01, DM02, DM03, DM04, DM05, DM07, DM08, DM10, DM11* and *DM12*. With these as our dataset, we used ***Task1.m****,* to extract all the data from the files present in these folders and convert them into the Gesture CSVs. Each row in each Gesture CSV consist of an action count column, sensor name column and many columns of time instances. The MATLAB code parses the name of all the files in the dataset to figure out which gesture the file belongs to and then converts the data present in it to the above format. The code limits the data to thirty-four sensors needed i.e. *ALX, ALY, ALZ, ARX, ARY, ARZ, EMG0L, EMG1L, EMG2L, EMG3L, EMG4L, EMG5L, EMG6L, EMG7L, EMG0R, EMG1R, EMG2R, EMG3R, EMG4R, EMG5R, EMG6R, EMG7R, GLX, GLY, GLZ, GRX, GRY, GRZ, ORL, OPL, OYL, ORR, OPR* and *OYR*. When combining the data between different files there might be a mismatch in the number of time instances. In this case, the code pads the empty cells with zeros after combining the tables of the combined data.

Task II: Feature Extraction

Based on our intuition, given below are the list of sensors we have chosen, which best represent each action.

Action	Sensors Considered		Intuition
	Sensor	**Subset of Sensors**	
About	Orientation	OPR	Since the right hand is rotating around the left hand, we are considering the orientation of the right hand.
And	Electromyograph (EMG)	EMG4R	Since the fingers are converging, and there is muscle tension, EMG is considered.
	Accelerometer	ARX	Since the hand moves from left to right, we are considering Accelerometer.
Can	Orientation	OPR	Since, there involves movement of both hands from top to bottom, the Orientation and Accelerometer are the best fit to represent the action.
	Accelerometer	ARZ	
Cop	Orientation	OPR	Since there is movement of the arm with the elbow as the fulcrum, Orientation is considered.
Deaf	Orientation	OPR	Since there is movement of the arm with the elbow as the fulcrum, Orientation is considered.
		ORR	
	Accelerometer	ARX	Since there is movement of the hand from mouth to the ear, Accelerometer is considered.
Decide	Orientation	OPL	Since there is movement of the arm with the elbow as the fulcrum, Orientation is considered.
		OPR	
	Accelerometer	ALY	Since there is movement of the hand from temple to the midriff, Accelerometer is considered.
		ARY	
Father	Orientation	OPR	Since there is movement of the arm with the elbow as the fulcrum, Orientation is considered.
	Accelerometer	ARX	Since there is movement of the arm oscillating along an axis, Accelerometer is considered.
Find	Electromyograph (EMG)	EMG0R	Since there is muscle tension in the forearm, EMG is considered.
	Accelerometer	ALX	Since there is movement of the wrist along an axis, Accelerometer is considered.
Go Out	Orientation	OPR	Since there is movement of the arm with the elbow as the fulcrum, Orientation is considered.
	Electromyograph (EMG)	EMG2R	Since the fingers are converging, and there is muscle tension, EMG is considered.
	Accelerometer	ARZ	Since the hand moves away from the body, we are considering Accelerometer.

Hearing	Electromyograph (EMG)	EMG2R EMG3R	Constant activity is observed in the fingers and arm muscles as there is muscle tension caused by the movement of the fingers.

As we observe the above table, we see that there are three types of sensors which are common to all the actions: **Accelerometer**, **Electromyograph (EMG)** and **Orientation**. For each sensor belonging to these three types, we have used a specific type of transformation listed below:

- **Root Mean Square (RMS):** EMG0R, EMG2R, EMG3R, EMG4R
- **Standard Deviation (STD):** ARZ, OPL, ORR
- **Discrete Wavelet Transform (DWT):** ARX, ALY, OPR

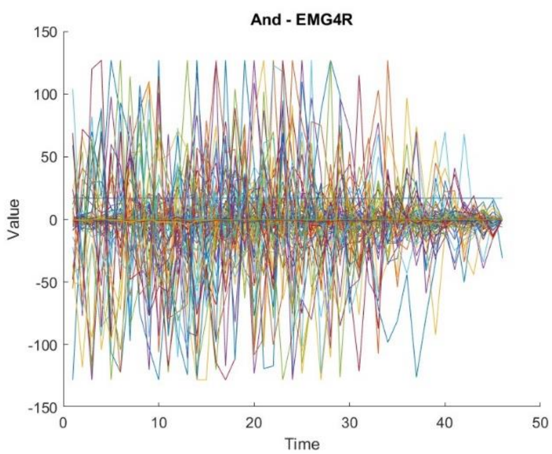
The explanation, intuition and analysis behind the choices of these feature extraction methods are explained in the following sub-sections.

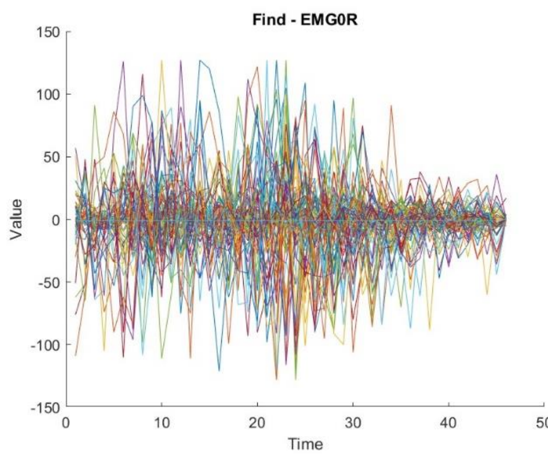
Root Mean Square (RMS)

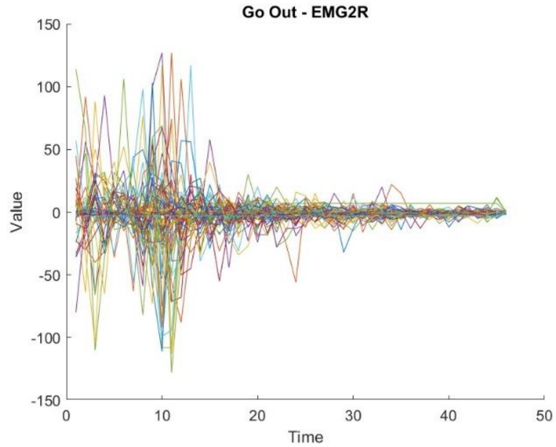
In statistics and its applications, the root mean square or RMS is defined as the arithmetic mean of the squares of a set of number. The RMS value of a signal is a way of expressing its average/mean power. It is the square root of the mean of the squared value of the signal.

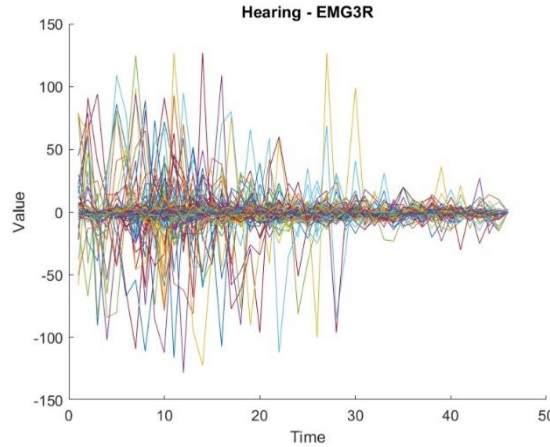
$$\mathrm{RMS} = \sqrt{\frac{a_1^2 + a_2^2 + a_3^2 + \cdots + a_n^2}{n}}$$

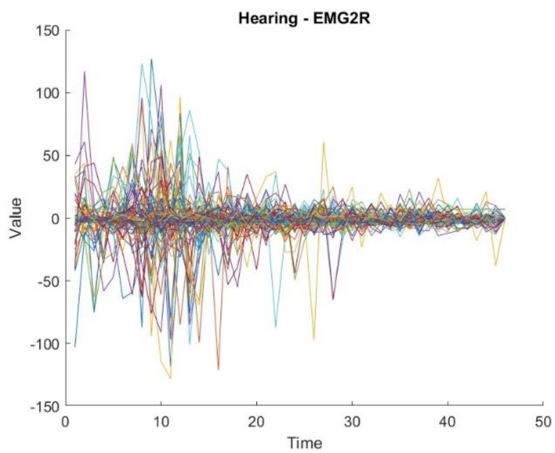
The following plots depict the time series data of the gestures and EMG sensors of interest. They are an overlap of different instances of the same gesture performed.





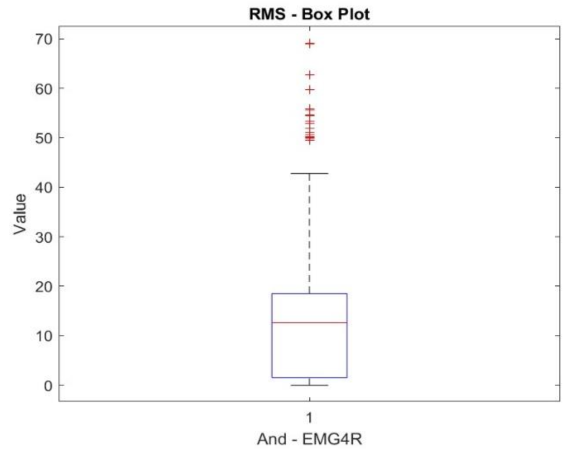




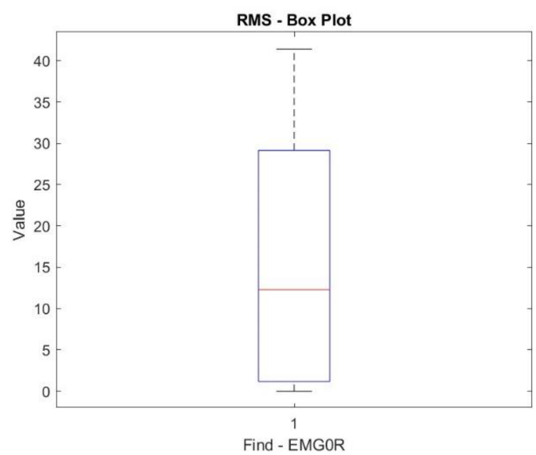


Intuition: Upon observing the plots for the EMG sensors, we can see that the values of the sensors oscillate around the value "0". The extent to which the values oscillate around the value "0" is of interest to us. Hence, our intuition is that RMS would be a good feature extraction method to extract pattern from this data.

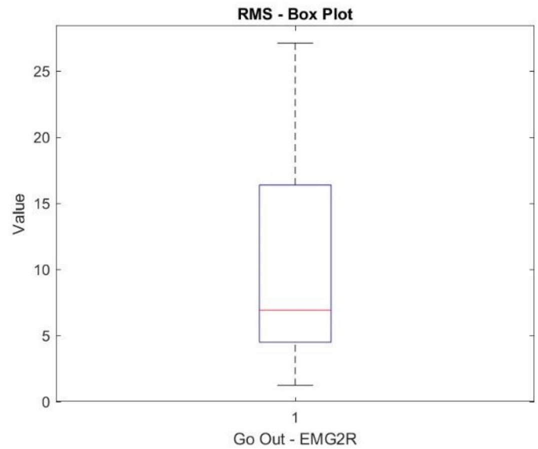
The figures shown below depict the range of values for RMS for their respective gestures. Each figure is a box-plot which contains the data (the 10th percentile, Inter-quartile range, Mean, 90th percentile) and the outliers (indicated by + in the graphs).



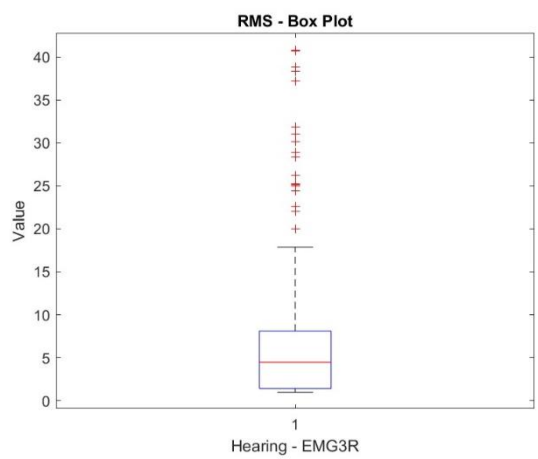
Inference: Thus, for gesture "And", we expect the value of RMS on sensor EMG4R to be in the range as depicted on the graph above.



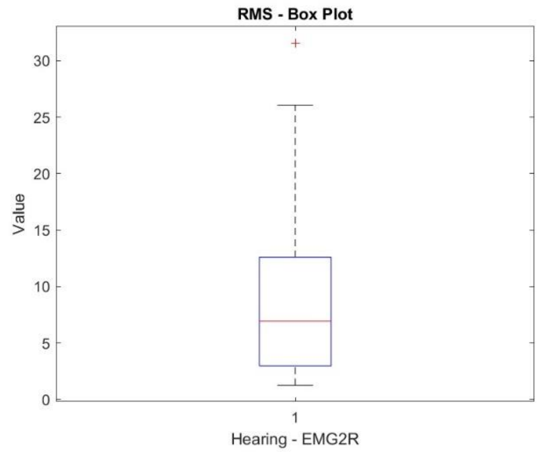
Inference: Thus, for gesture "Find", we expect the value of RMS on sensor EMG0R to be in the range as depicted on the graph above.



Inference: Thus, for gesture "Go out", we expect the value of RMS on sensor EMG2R to be in the range as depicted on the graph above.



Inference: Thus, for gesture "Hearing", we expect the value of RMS on sensor EMG3R to be in the range as depicted on the graph above.



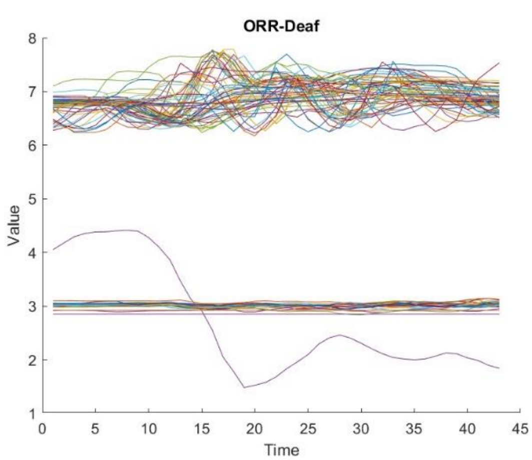
Inference: Thus, for gesture "Hearing", we expect the value of RMS on sensor EMG2R to be in the range as depicted on the graph above.

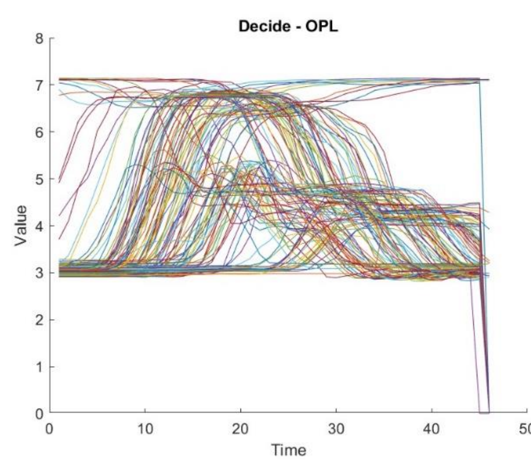
Standard Deviation (STD)

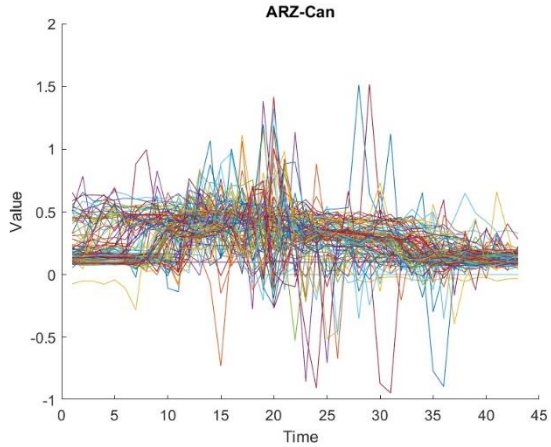
In statistics, the standard deviation is a measure that is used to quantify the amount of variation or dispersion of a set of data values. A lower standard deviation indicates that the data points tend to be closer to the mean, while a higher standard deviation indicates that the data points are spread out over a wide range of values.

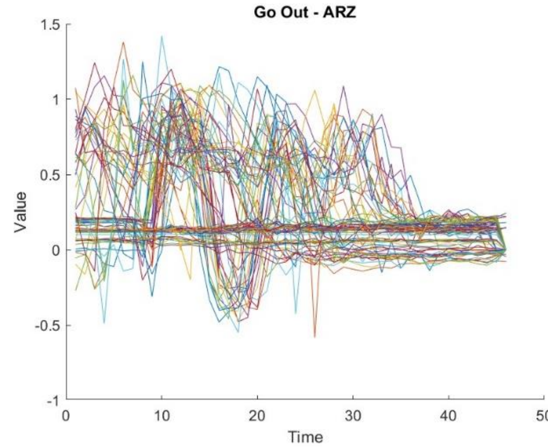
$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2}$$

The following plots depict the time series data of the gestures and sensors (OPL, ORR and ARZ) of interest.



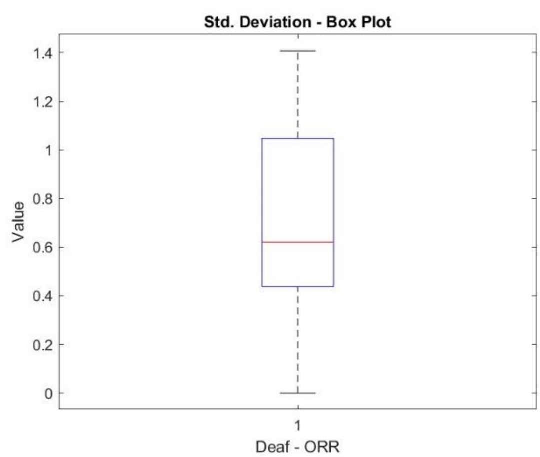




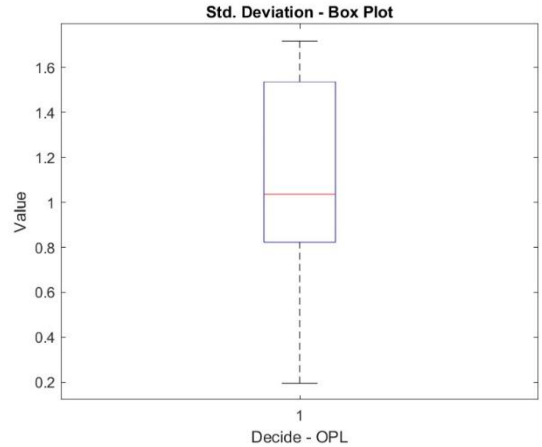


Intuition: Upon observing the above plot, we see that the values of the sensor lie within a certain range. We are interested in the spread of these values across time. Thus, our intuition is that Standard Deviation would be a good feature extraction method to extract this pattern from the data.

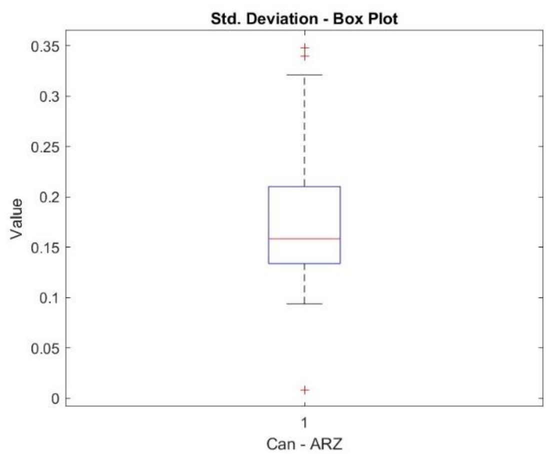
The figures shown below depict the range of values for Standard Deviation for their respective gestures. Each figure is a box-plot which contains the data (the 10th percentile, Inter-quartile range, Mean, 90th percentile) and the outliers (indicated by + in the graphs).



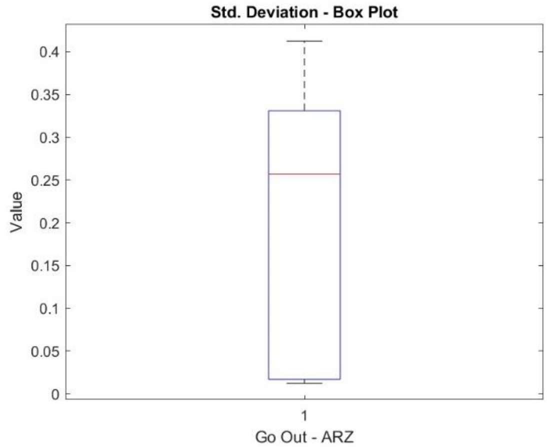
Inference: Thus, for gesture "Deaf", we expect the value of Standard Deviation on sensor ORR to be in the range as depicted on the graph above.



Inference: Thus, for gesture "Decide", we expect the value of Standard Deviation on sensor OPL to be in the range as depicted on the graph above.



Inference: Thus, for gesture "Can", we expect the value of Standard Deviation on sensor ARZ to be in the range as depicted on the graph above.

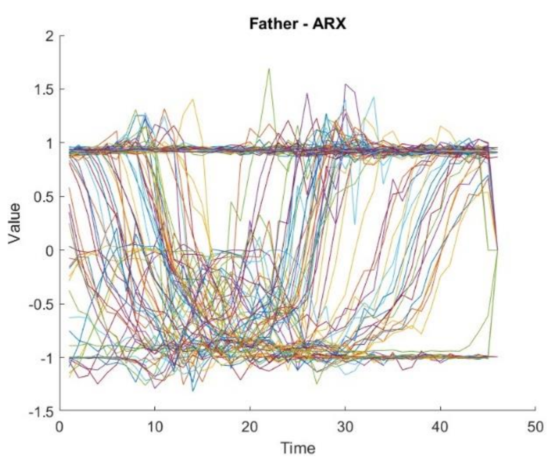


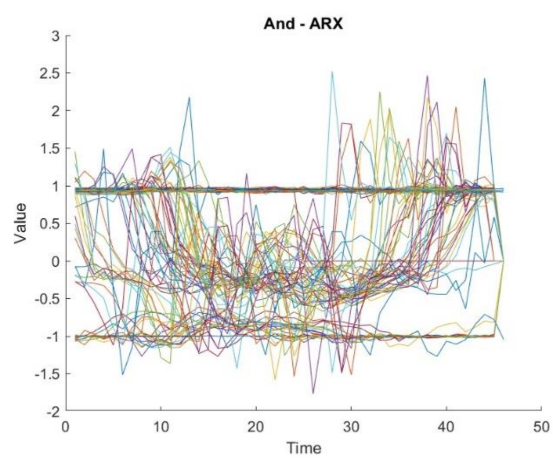
Inference: Thus, for gesture "Go Out", we expect the value of Standard Deviation on sensor ARZ to be in the range as depicted on the graph above.

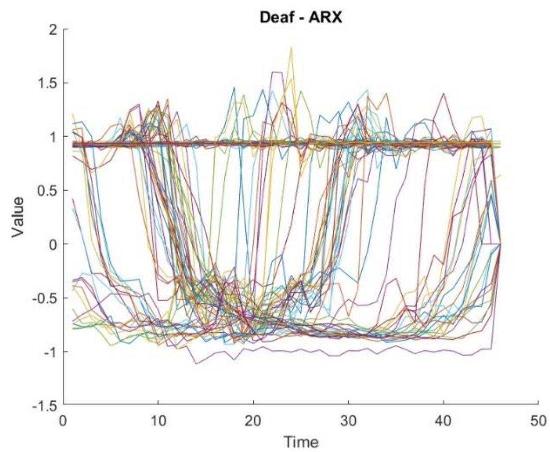
Discrete Wavelet Transform (DWT)

In mathematics, a wavelet series is a representation of a square-integrable function by a certain orthonormal series of generated by a wavelet. A discrete wavelet transform (DWT) is an implementation of the wavelet transform using a discrete set of the wavelet scales and translations obeying some defined rules.

Consider the following plots:

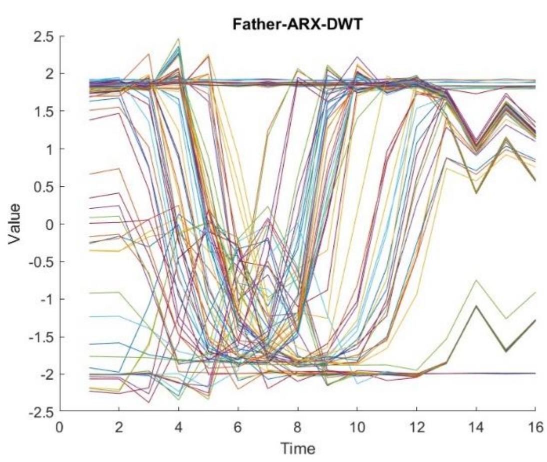


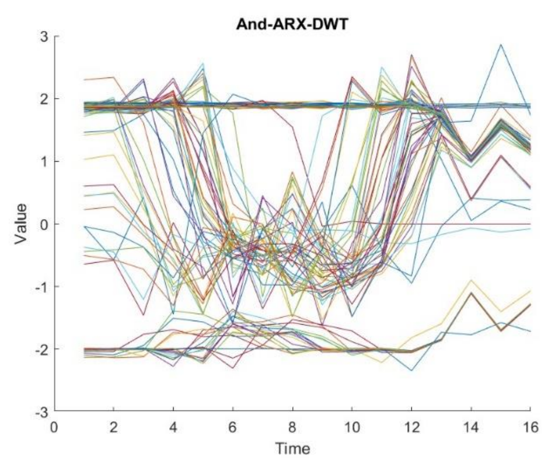




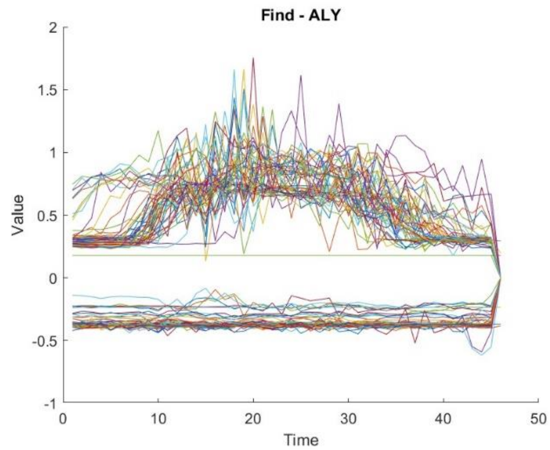
In these plots, we can see a certain distinctive pattern that are being followed. For example, in the first plot, we can see that there is a dip in the values between the time 10 and 30. Similarly, we can see from second and third plot, that there is a certain pattern observed in the range (10-30) and (10-40). Thus, our intuition is that DWT would be a good feature extraction method to extract this pattern from the data.

The plots below depict the DWT transformation on the graphs shown above. In specific, we are interested in the values obtained at points 6 and 12. These values help us in identifying the pattern we are trying to extract.



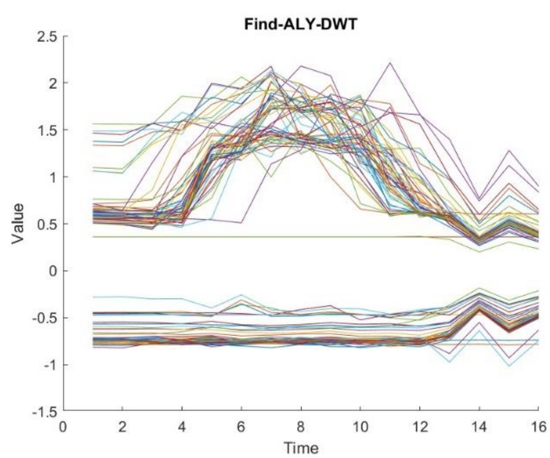


Consider the following plot:

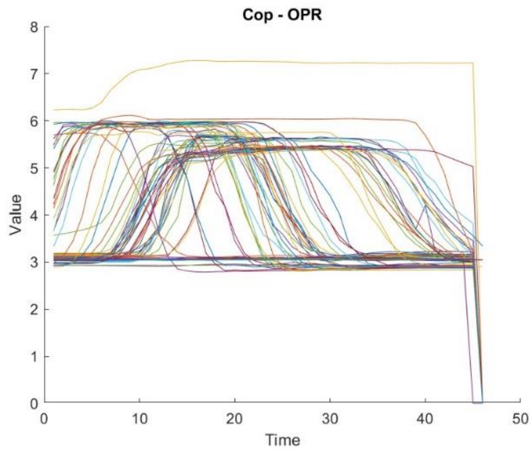


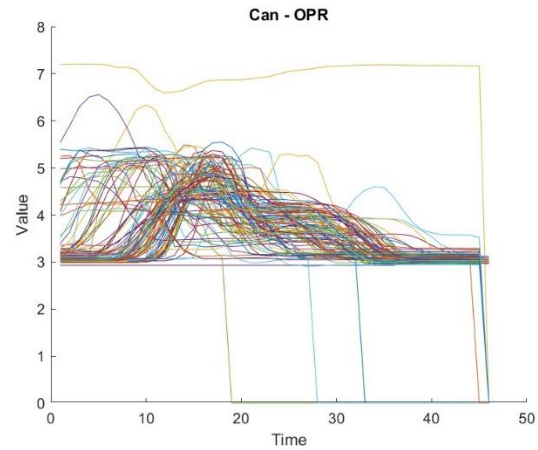
In the above plot, we can see a certain distinctive pattern that are being followed. Thus, our intuition is that DWT would be a good feature extraction method to extract this pattern from the data.

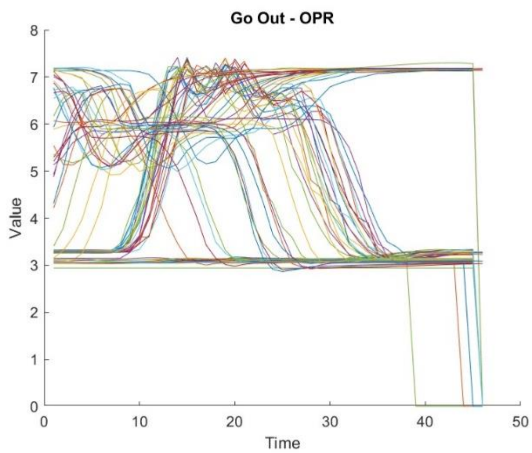
The plot below depicts the DWT transformation on the graph shown above. In specific, we are interested in the values obtained at points 8, 9 and 10. These values help us in identifying the pattern we are trying to extract.

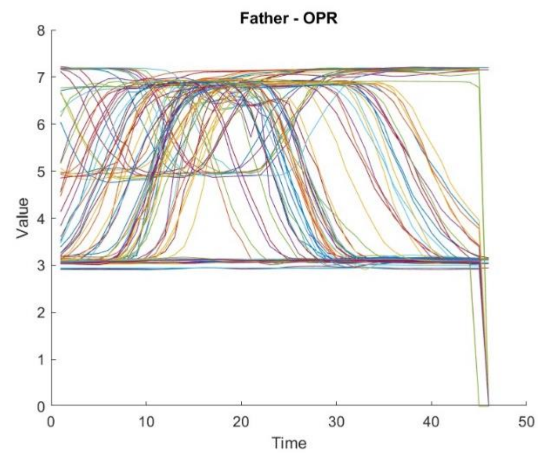


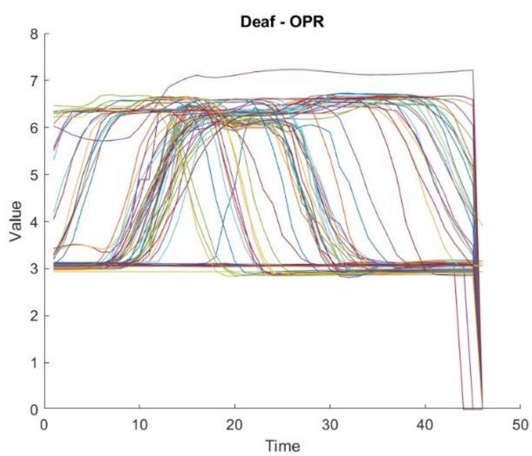
Consider the following plots:

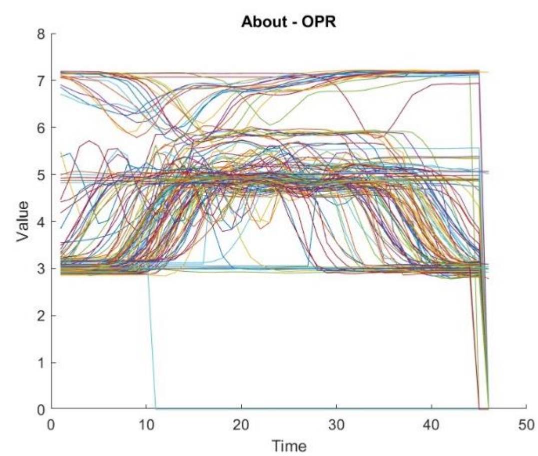






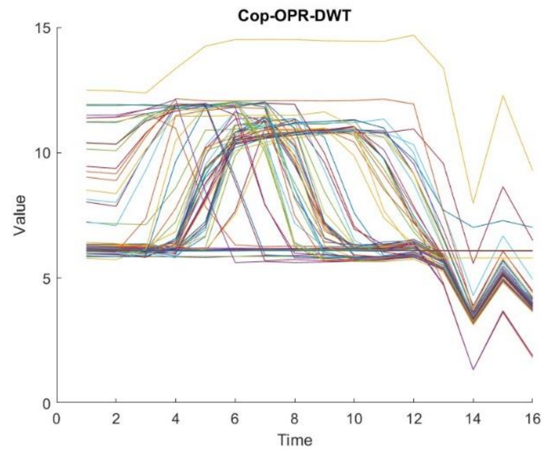


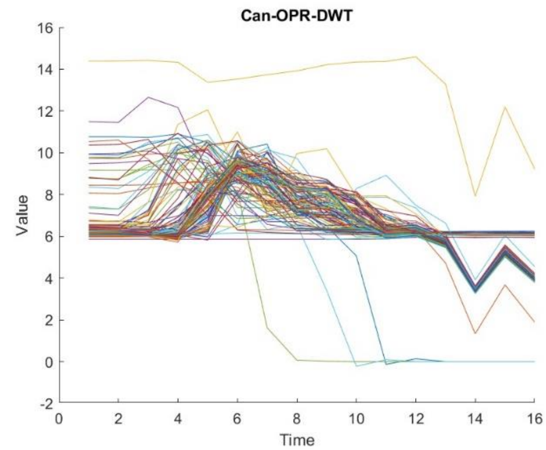


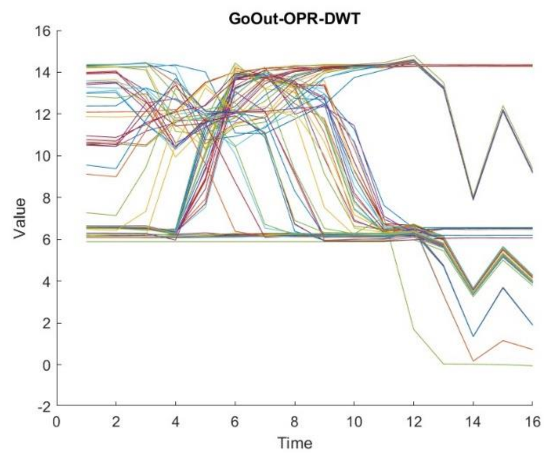


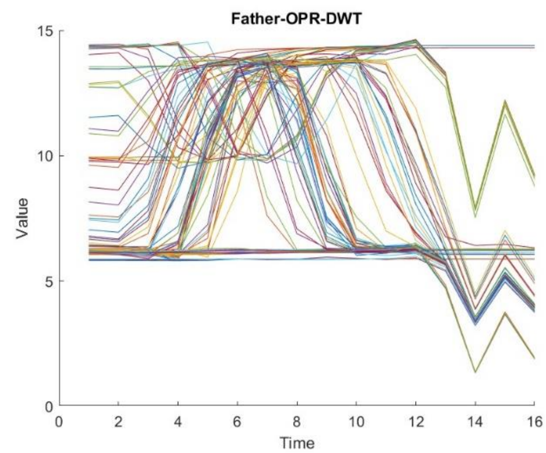
In the above plots, we can see a certain distinctive pattern that are being followed. Thus, our intuition is that DWT would be a good feature extraction method to extract this pattern from the data.

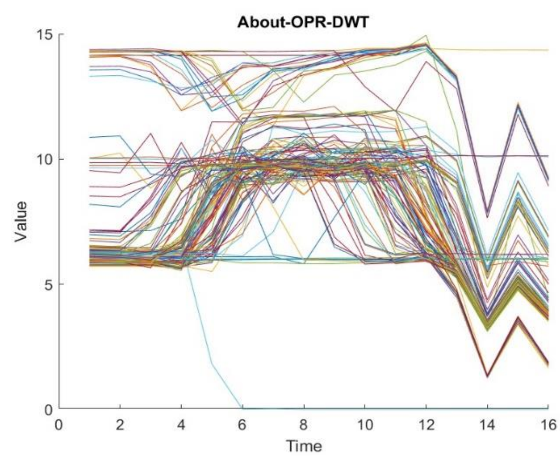
The plot below depicts the DWT transformation on the graph shown above. In specific, we are interested in the values obtained at points 8. This value would help us in identifying the pattern we are trying to extract.

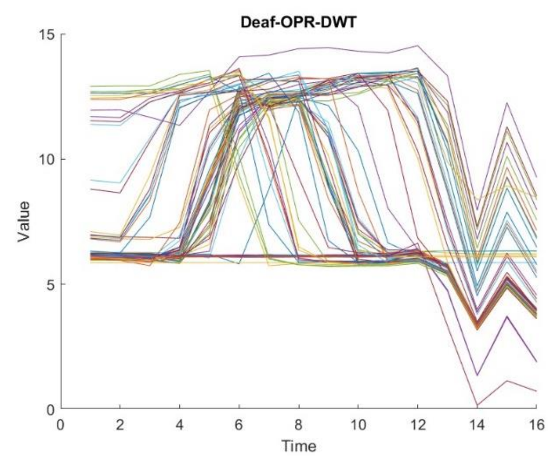












Task III: Feature Selection

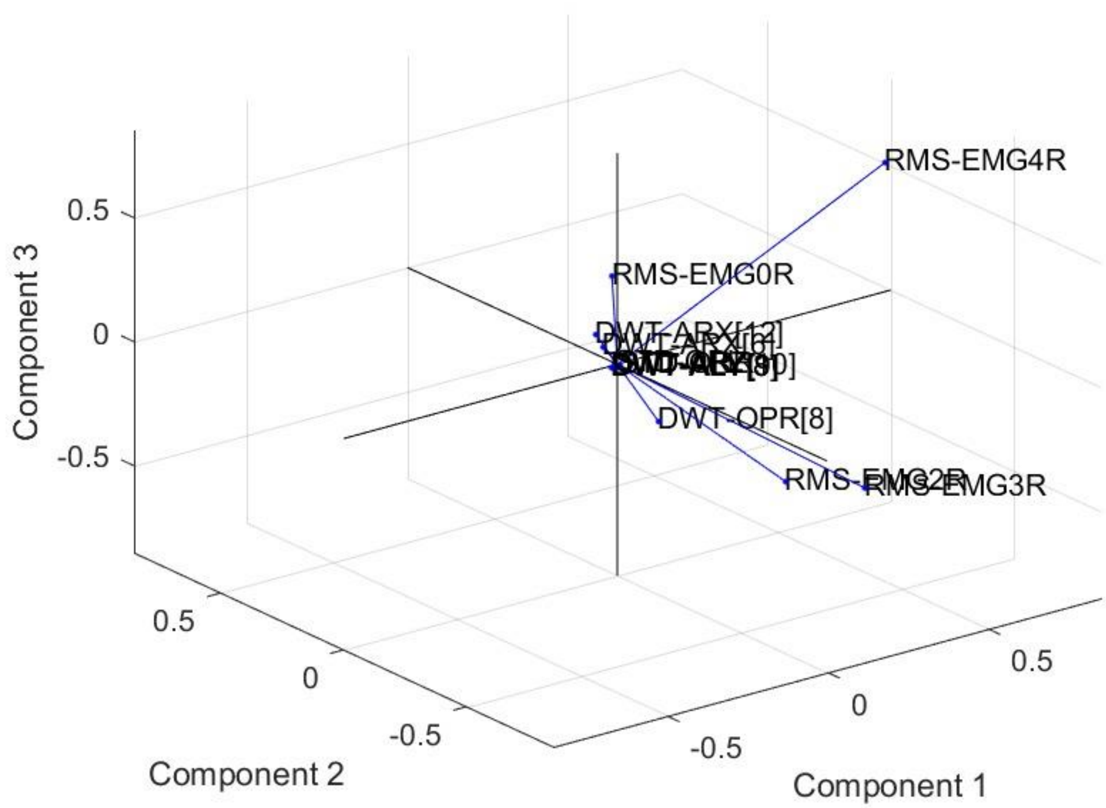
Upon extraction features in Task II, we try to find which of these features yields the most variance on the data set. This can be accomplished by performing *Principal Component Analysis (PCA)*.

Arranging the Feature Matrix

Before we can proceed and perform Principal Component Analysis, we must arrange the features extracted in the previous task, into a single 2-dimensional matrix. It should be noted that this matrix should contain values pertaining to all actions, as Principal Component Analysis *(PCA)* must be run over the all of them. This two-dimensional matrix is obtained by column-wise concatenation of all the feature matrices obtained in the previous task. So, the dimensions of this would be 400x13, where there are 400 data samples and there are 5 (*Discrete Wavelet Transform*) + 3 (*Root Mean Square*) + 3 (*Standard Deviation*) = 13 columns in total.

Execution of Principal Component Analysis (PCA)

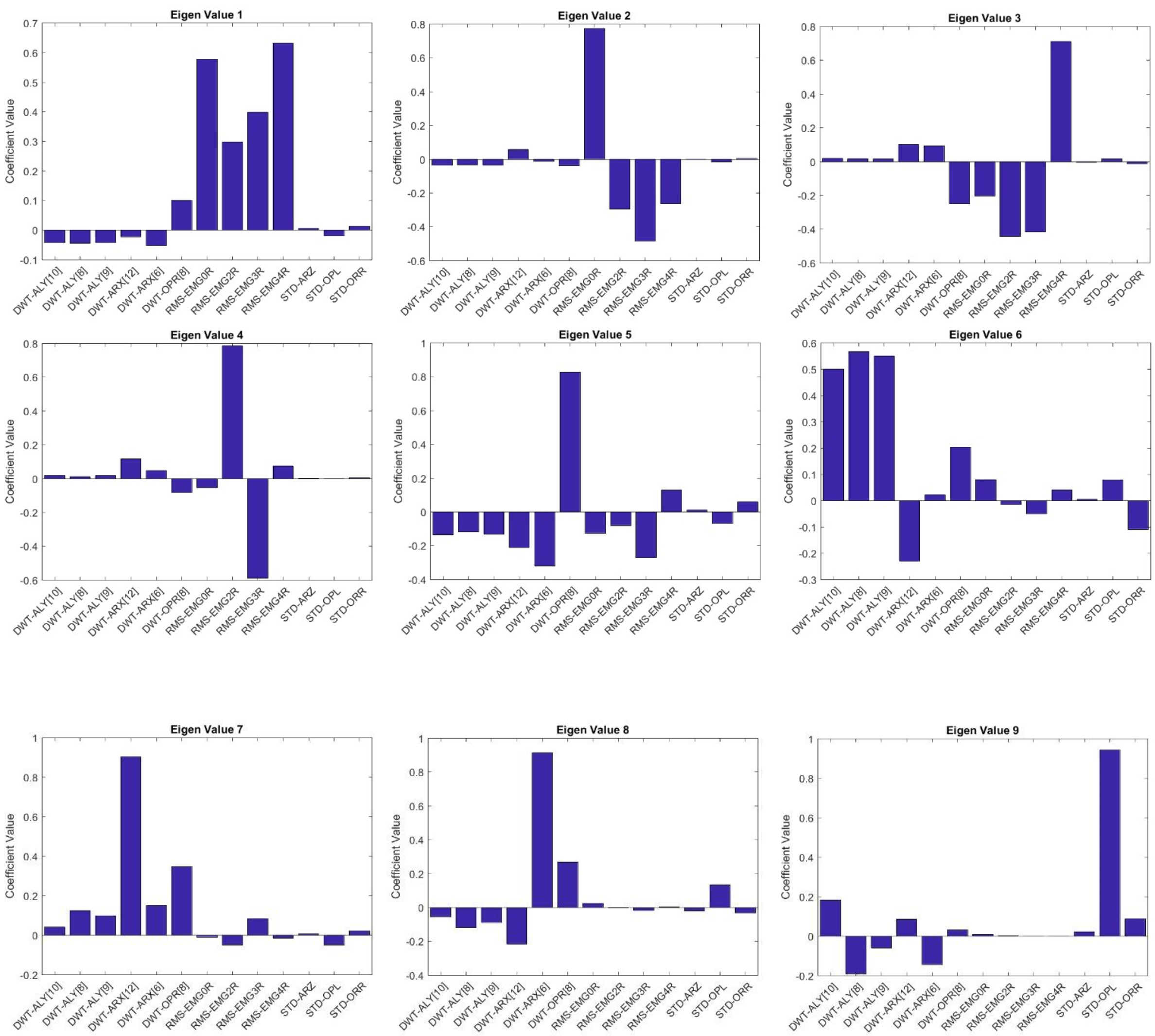
The eigenvectors of the three principal components are shown in the plot given below:

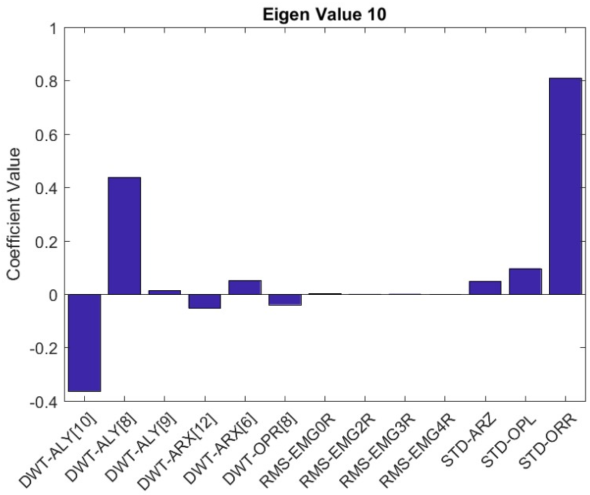


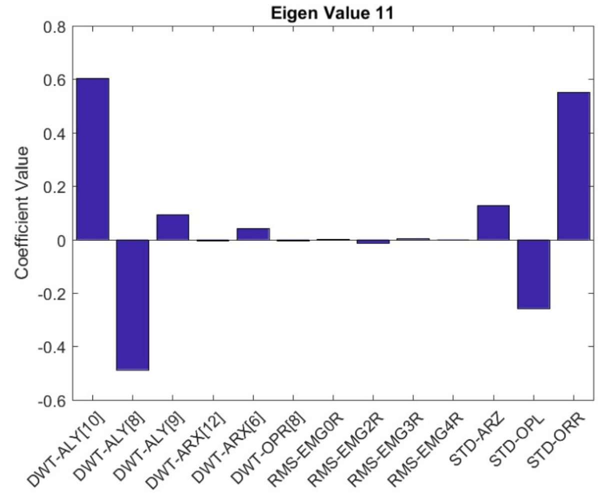
This plot shows the contribution of each individual feature to the three principal components. *RMS-EMG4R, RMS-EMG0R, DWT-OPR, RMS-EMG2R* and *RMS-EMG3R* refers to the variances pertaining to the *EMG* and *Orientation* features. This plot shows that:

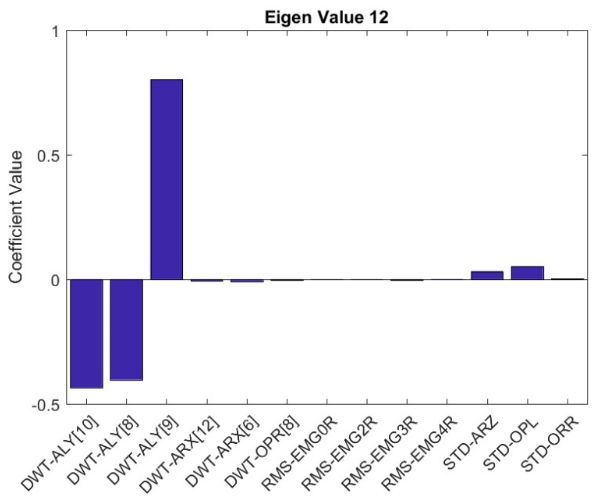
- *RMS-EMG0R, RMS-EMG2R, RMS-EMG3R* and *RMS-EMG4R* have a strong positive co-efficient in principal component I.
- *RMS-EMG0R* has a strong positive co-efficient in principal component II.
- *RMS-EMG2R, RMS-EMG3R* and *RMS-EMG4R* have a strong negative co-efficient in principal component II.
- *DWT-OPR, RMS-EMG0R, RMS-EMG2R, RMS-EMG3R* have strong negative influence, and *RMS-EMG4R* has a strong positive influence in principal component III.
- The other features all contribute significantly lesser than the above features and their labels are overlapping near the origin.

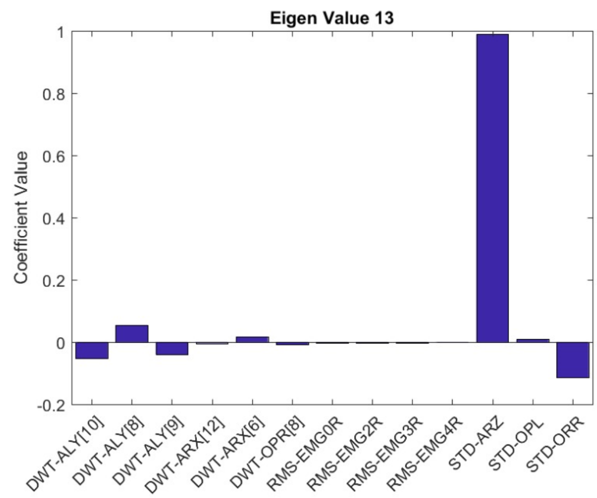
Discussion on Eigen Vectors







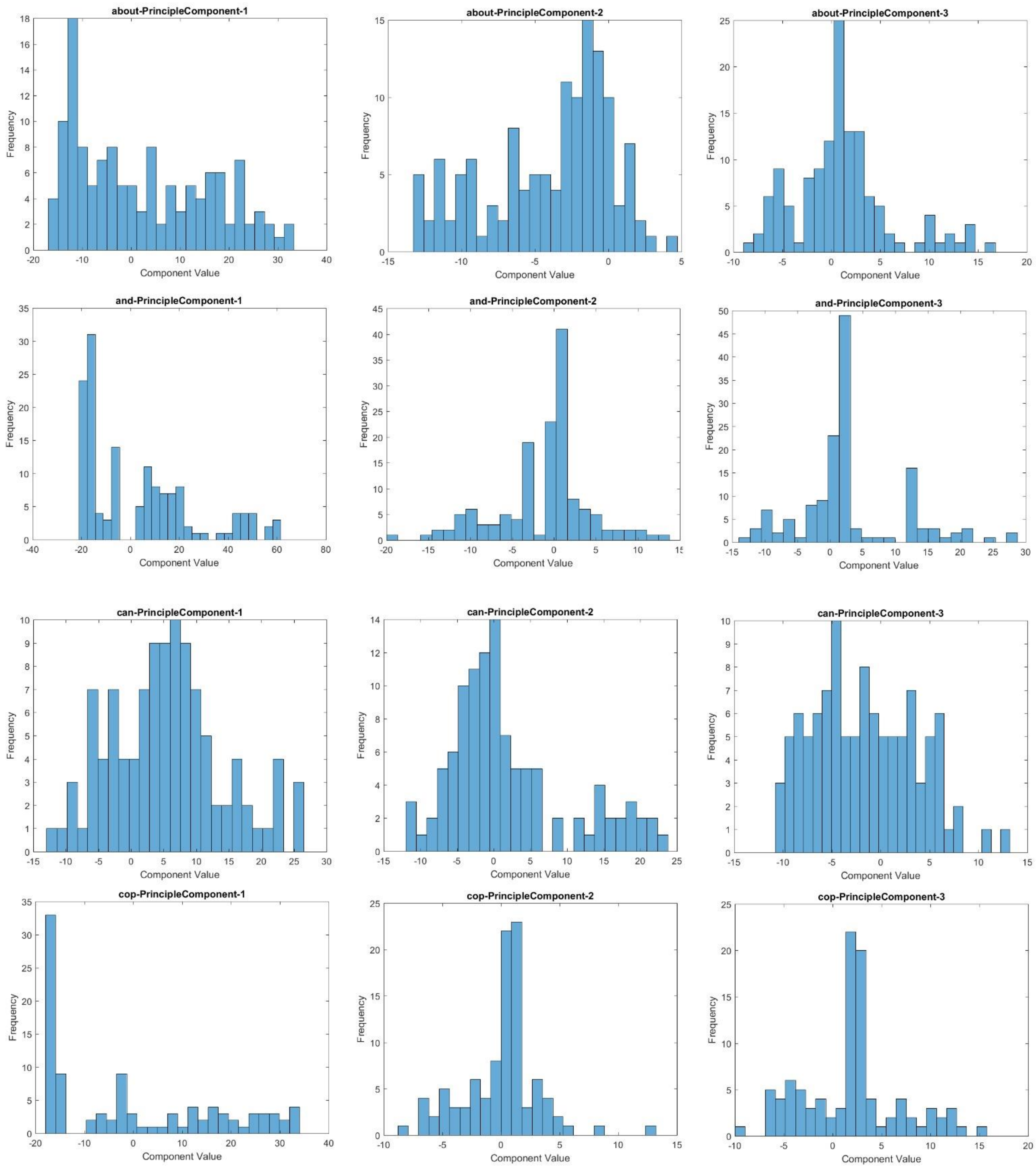


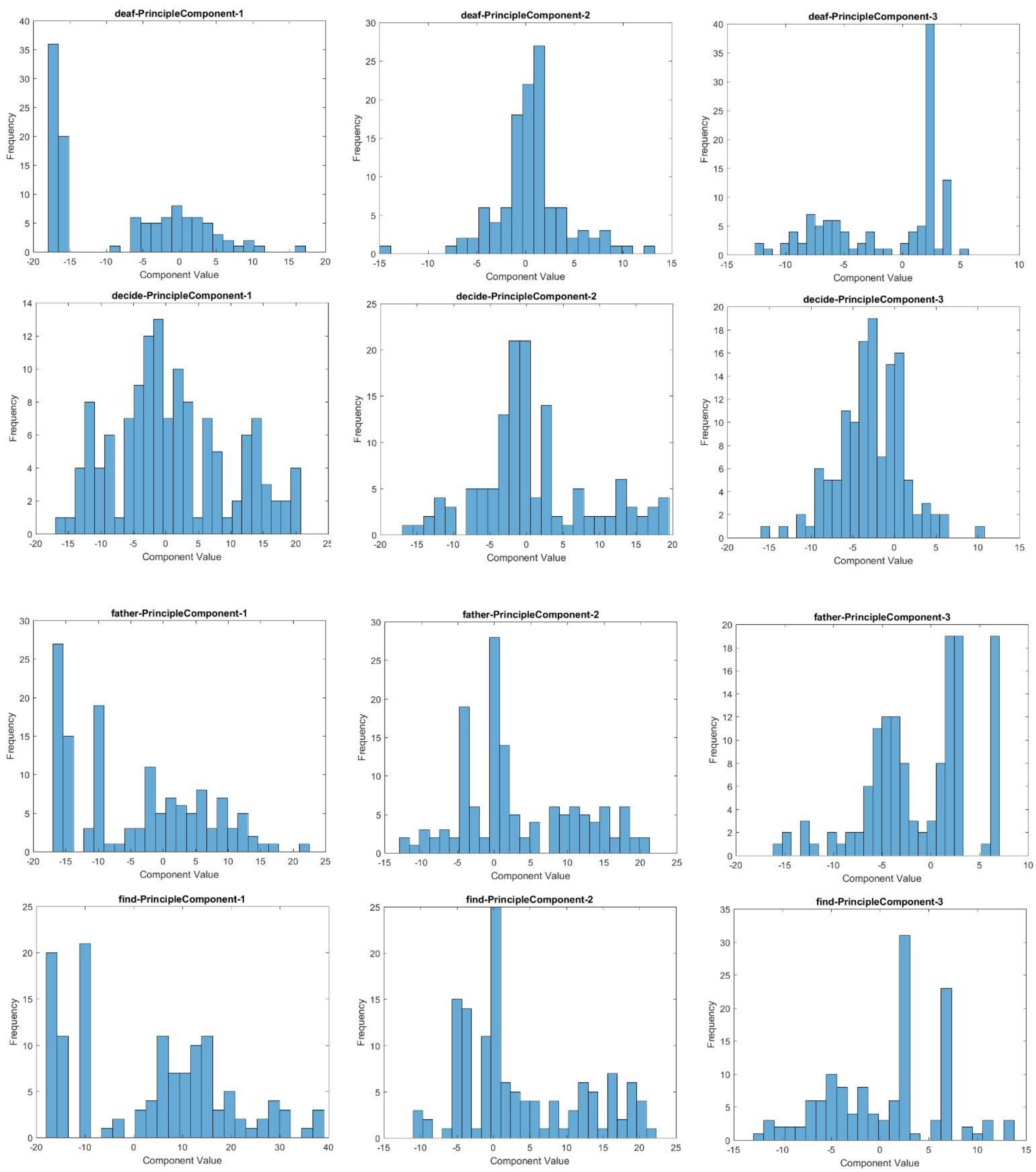


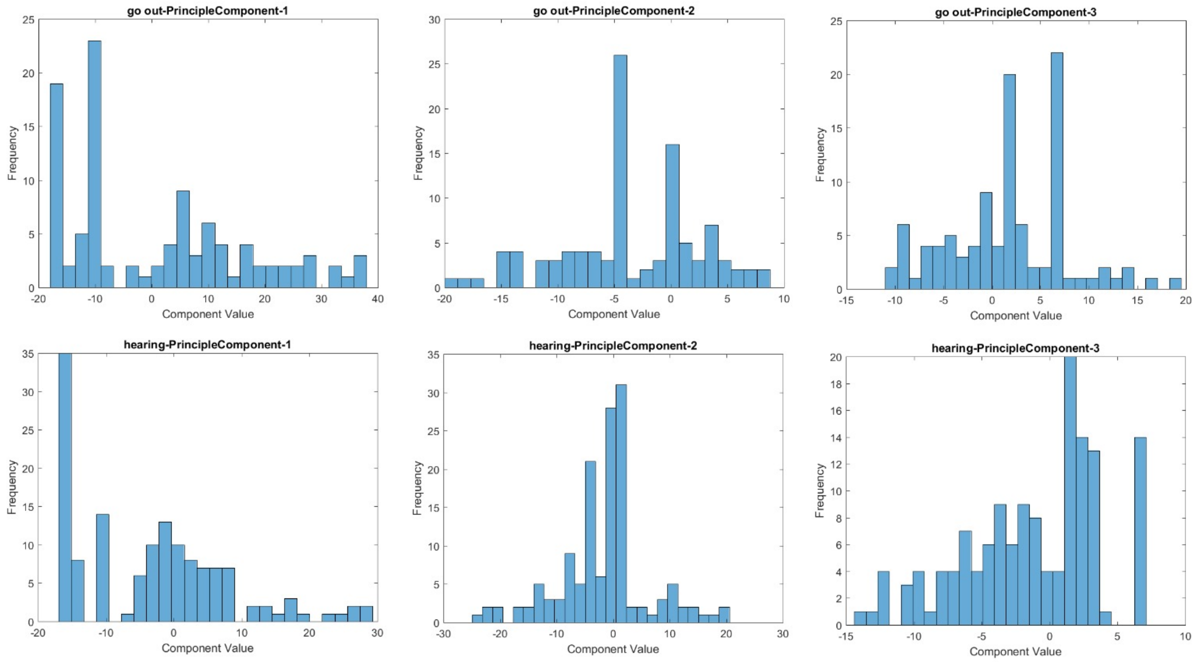
As seen in the above bar graphs, the variances of the *EMG* sensors are weighted very heavily in the first three eigen vectors. The feature with the next highest influence is the *OPR* sensor. This comes as a surprise especially since the first principal component captures *64.7%*, the second captures *15.35%* and third captures *10.57%*, of the data. We expected features like *Accelerometer* to make an appearance, but their contribution is rather infinitesimal. This tells us that most of our information can be captured using the *EMG* sensors. This does make sense as each gesture do consist of muscle contractions which are captured by the *EMG* sensors.

Results of Principal Component Analysis (PCA)

The distributions of the original data, projected onto the new feature space all the actions are given below.




deaf-PrincipleComponent-1
deaf-PrincipleComponent-2
deaf-PrincipleComponent-3
decide-PrincipleComponent-1
decide-PrincipleComponent-2
decide-PrincipleComponent-3
father-PrincipleComponent-1
father-PrincipleComponent-2
father-PrincipleComponent-3
find-PrincipleComponent-1
find-PrincipleComponent-2
find-PrincipleComponent-3
Frequency
Component Value



Conclusion of Principal Component Analysis (PCA)

Principal Component Analysis was certainly helpful as we can reduce the feature matrix from a *Nx13* to a *Nx3* matrix, where *N* is the number of data samples, and still capture *90.62%* of the variance. This is important because features with high variance are required to distinguish between gestures. Also, if we use the top five principal components, we can capture *99.01%* of the entire variance.

Since we can capture almost the entire variance with around 38% of the features chosen, PCA was helpful in reducing the number of features.

Bibliography

1. https://en.wikipedia.org/wiki/Wavelet_transform
2. https://www.mathsisfun.com/data/standard-deviation-formulas.html
3. https://en.wikipedia.org/wiki/Standard_deviation
4. http://www.mathwords.com/r/root_mean_square.htm
5. https://www.mathworks.com/matlabcentral/answers/131353-relation-between-fft-and-rms
6. https://en.wikipedia.org/wiki/Root_mean_square
7. "Project Phase 3 Report – Data Mining" – Resource provided on Blackboard.