Assignment 2 and 3- Human Activity Recognition

CSE 572: Introduction to Data Mining (Fall 2018)

To:

Prof. Ayan Banerjee

I.R.A. Fulton School of Engineering

Arizona State University

By:

Group - 4

Jayashree Subramanian (1214347796)

Mahathi Srinivasan (1215187155)

Mahima Chhalani (1215161558)

Arpan Vitthalbhai Patel (1215104930)

Vivek Ranjan Panigrahy (1215218901)

Abstract:

The project involved reducing the features of the eating activity dataset by performing PCA and classifying the eating and non-eating instances of a set of users using 3 specific classifying algorithms i.e. Decision Trees, Support Vector Machines and Artificial Neural Networks.

Keywords:

Feature, Myo Armband, Accelerometer, EMG Sensor, Gyrometer, feature matrix, PCA, Decision tree, Support vector machines, Neural networks, F1 score, precision, recall

1. Introduction

1.1 Terminologies:

a. *Feature:* A feature is any term in the document/dataset which is specific to it and helps in distinguishing the document/dataset from others.

b. *Gesture:* A gesture is a form of nonverbal communication or non-vocal communication in which visible bodily actions communicate messages, either in place of, or in conjunction with, speech. Gestures include movement of the hands, face, or other parts of the body.

c. *Gesture Recognition*: Gesture recognition is the mathematical interpretation of a human motion by a computing device. It is the ability of a computer to understand gestures, interpret them and analyze it.

d. *Human Activity Recognition (HAR)*: Human activity recognition is a key sub area of Human Computing Interaction (HCI) where a subject (usually a human) wears an array of sensors and these sensors record various input signals from the body which can be interpreted as gestures. Once the system has enough input signals with labels (Gestures) to differentiate one signal from the other, we can use the system/model to predict the next input signal without a label.

e. *Myo Armband*: It is a wearable gesture control and motion control device that records 8 different kinds of data like Accelerometer, Gyro meter, Electromyography, Orientation, Orientation-Euler etc.

Fig 1. Myo Armband [1]

f. *Accelerometer:* A sensor which measures proper acceleration of the object (in our case, hand) with respect to the x, y and z axis. Proper acceleration means the acceleration a body achieved by its own with respect to the instantaneous rest frame.

g. *Gyrometer*: This is used to measure the angular velocity i.e. the change in rotational angle per unit of time. These sensors collect input from the coriolis force applied to a vibrating element. For this reason, the accuracy of the angular velocity measured differs significantly depending on the element.

h. *EMG Sensor*: Electromyography sensor records the electrical activity produced by skeletal muscles in the body. Myoelectrical signals are formed by physiological variations in the muscle fibers. There are total of 8 EMG signals namely EMG1, EMG2, EMG3, EMG4, EMG5, EMG6, EMG7 and EMG8.

i. *Clustering:* It is a process where we club together similar data points having a common attribute or common goal which differentiates them from all other sets of samples/data points.

j. *PCA:* Principal Component Analysis is a statistical procedure that uses orthogonal transformation to convert a set of observations of linearly uncorrelated variables into principal components along maximum variance. [4]

k. *Feature Matrix*: It is a (Sample x Matrix) matrix which has a list of all features with respect to their sensors. This feature matrix is the input to the PCA algorithm which then outputs a set of Eigenvectors and Eigenvalues corresponding to the best features across which we have maximum variance.

l. *Roll, Pitch and Yaw*: These are Euler angles used to measure orientation of an object. For example, if we consider 3 straight orthogonal lines (intersecting each other at 90 degrees) along a central plane, then rotation along the front-to-back axis is called roll, rotation along the side-toside axis is called pitch and rotation along the vertical axis is called yaw.

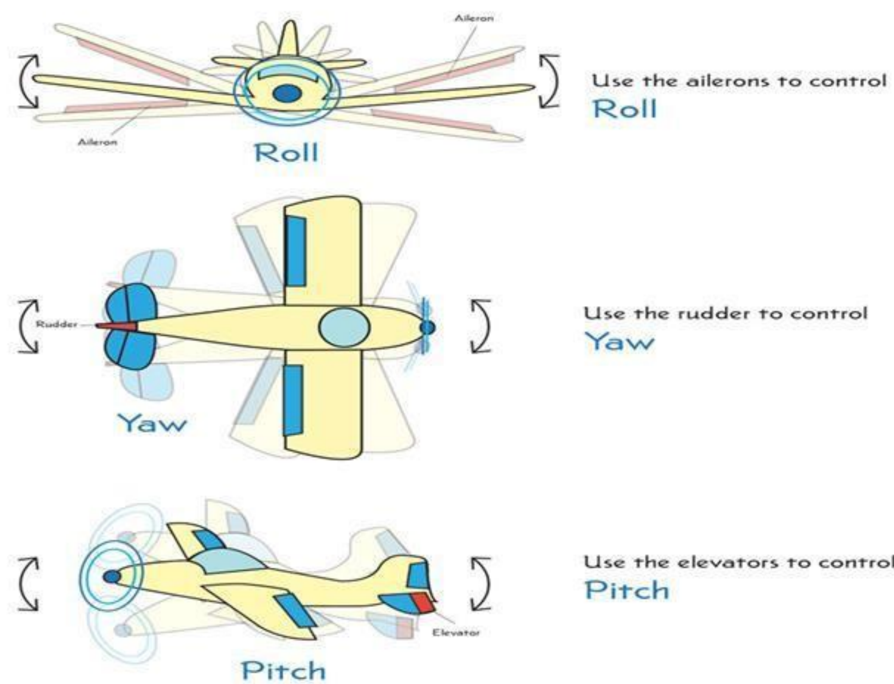


Fig 2. Representation of Pitch, Yaw and Roll[2]

1.2 Classification Algorithms:

a. *Decision tree:*

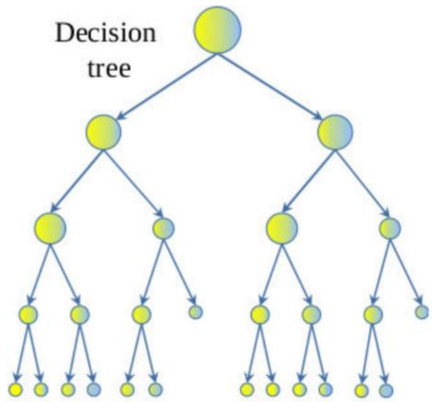


Fig 3. Decision Tree classifier[10]

Decision tree is a classification algorithm that takes the training data and returns a trained model in the form of decision nodes and leaf nodes. These decision nodes and leaf nodes contain the criteria to classify data. Decision trees can deal with numerical and categorical data.

b. *Support Vector Machine:*

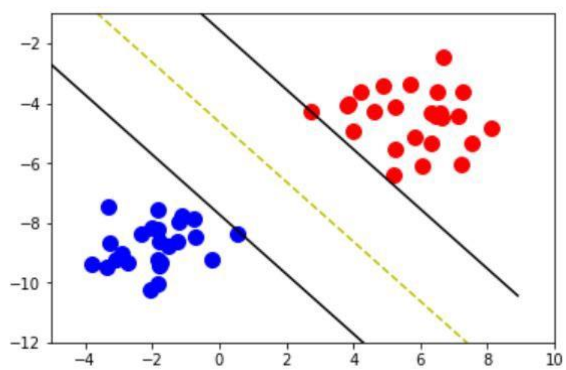


Fig 4. Support Vector Machine classifier[10]

A Support vector machine is a classifier that separates groups of data by a hyperplane. The hyperplane is determined by a set of support vectors. Support vectors are the closest points of each group of data that lie on the decision boundary. A good hyperplane is that which is equidistant from the nearest samples from each class.

c. *Neural Networks:*

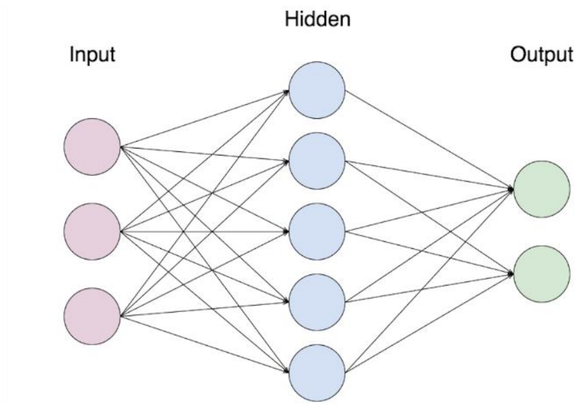


Fig 5. Neural Networks Classifier[10]

A neural network consists of an input layer, one or more hidden layers and an output layer. Each node in the input layer is connected to all nodes in the hidden layer by 'weights. The same can be said for all nodes in the hidden layer to the output layer. The inputs are passed on to the nodes in the next layer until the label is obtained by a linear or non-linear activation function. This is called forward propagation. The weights are also updated by a back-propagation algorithm, thus aiding in creating a well optimized and trained neural network.

1.3 Goal Description

During this phase of the project, we use the dataset that we got by multiplying the PCA output with our data feature set.

Phase 1

In this phase of the project first extract the data that we have obtained from PCA and separate the data for the eating and non-eating activity by assigning the labels (label 1 for eating activity and label 0 for non-eating activity).

Phase 2

In the second phase of the project, the separated data is divided into two parts. The data is split into training and testing data. 60% of the data is the training data and the rest 40% of the data is reserved for testing.

Phase 3

In the third phase of the project, we use three types of classification algorithms namely decision tree, support vector machine and neural networks. The model is built using these algorithms and is trained using the training data sand is tested using the tested data. Finally, we test its accuracy and report the metrics such as F1 score, precision and recall for each of the users. The train test split is taken as 80 - 20 for the first case and is compared with the 60 - 40 train and test split.

2. Data preprocessing

For the data preprocessing step, we first look at the ranges available in the ground truth file and then multiply each range in it with 50 and divided it by 30 to match the frequency of the frames of the video data and the same obtained for MYO band sensor. The columns in the EMG and IMU file corresponding to the eating activity will be the columns which are in the range that we got from the ground truth file. Finally, we labeled the eating activity as 1 and the remaining columns which corresponds to the non-eating activity as 0.

3. Feature Extraction and Selection

From assignment 1, five distinct features were selected as a result of feature extraction process for the sensors across all the dimensions (x, y, z and w). The number of features before performing feature reduction (PCA) was compared with the features obtained after PCA. The percentage of variance retained across the dimensions are measured, analysed and reported. [9]

3.1. Principal Component Analysis [9]

For tasks such as data mining, data in high dimensional space can often result in inaccurate and unreliable results due to the fact that density of the data decreases exponentially with the increase of the dimensionality, making it difficult to organize and analyze the data in high-dimensional spaces and hence data is preprocessed reduce the dimension.

Dimensionality reduction is the process of reducing the number of dimensions of a dataset under consideration, by obtaining a set of principal dimensions. Feature selection and Feature extraction are two techniques essentially used for reducing the dimensionality of the data.

Feature Extraction is a technique that transforms the original feature vectors in the high-dimensional space to a low-dimensional space by deriving informative and non-redundant features, whereas Feature Selection is a technique that tries to select a subset of the original features by removing redundant and irrelevant features of the data.

To overcome the curse of dimensionality and avoid overfitting, we use the feature extraction technique to reduce the dimensionality by applying the Principal Component Analysis algorithm that simplifies the complexity of high-dimensional data by performing an orthogonal linear transformation of data into fewer dimensions. It reduces data by geometrically projecting them onto lower dimensions called principal components, with the goal of preserving the variance of the data using a limited number of principal components. The first principal component is chosen to maximize the variance of the projected points. The second and subsequent principal components are selected similarly, with the additional requirement and advantage that they be uncorrelated with all previous principal components.

We have used the pca function defined as [coeff, score, latent, tsquared, explained, mu] = pca(X) in MATLAB, which takes the dataset as input and returns:

• The coefficient matrix (28-by-28 matrix) as ‘coeff’. For our n-by-d feature matrix, i.e. 626 x 28 feature matrix X, the corresponding coefficient 28-by-28 matrix is calculated and stored in the variable ‘coeff’. Each column of coeff contains loading/weights for one principal component, and the columns are in descending order of component variance, latent.

• The principal component scores as ‘score’. Principal component scores are the representations of X in the principal component space, where rows of score correspond to observations, and columns corresponds to components
• The principal component variances as ‘latent’, which stores the eigenvalues of the covariance matrix of X
• The Hoteling’s T-squared statistic for each observation X as ‘tsquared’, which is the sum of squares of the standardized scores for each observation
• The information regarding the percentage of the total variance retained by each principal

component as ‘explained’
• The estimated mean of each variable in X as ‘mu’

4. Methods and Experiments

4.1. Decision Tree

(i)fitctree(): It is a MATLAB method which is used for implementing the decision tree. The fitctree function is generally used when we want to fit the decision tree for the classification purpose. For the given input variable, the function will return a fitted decision tree [6].

4.2. Support Vector Machine

*(ii)fitcsvm():*It is a MATLAB method used to classify groups of data based on the distance of the closest points from each group. The given inputs are the training data matrix and its corresponding targets, while the output returned is a fitted SVM kernel[7].

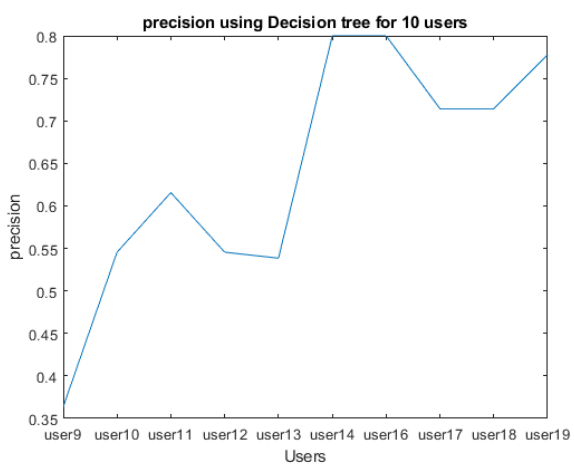
4.3. Neural Networks

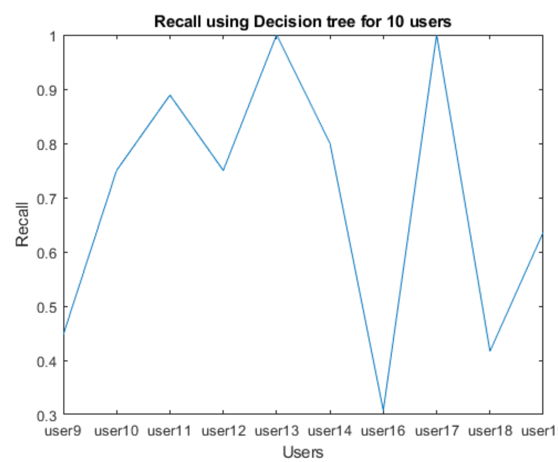
For this, MATLAB’s NNtoolbox was used to train a 1-hidden layer neural network with 10 nodes in the hidden layer. The inputs are the training data and its target variables while the output returned is a fully optimized neural network[8].

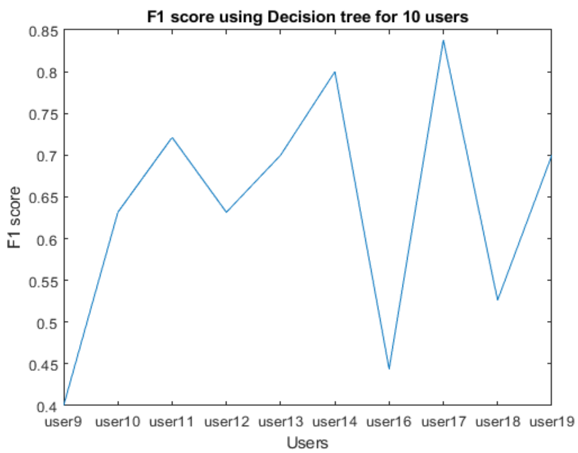
The 3 classifiers were trained and tested on the eating dataset of 10 users. Two different partition schemes were done on the dataset and the task was repeated. The results and their corresponding plots are given in the following section.

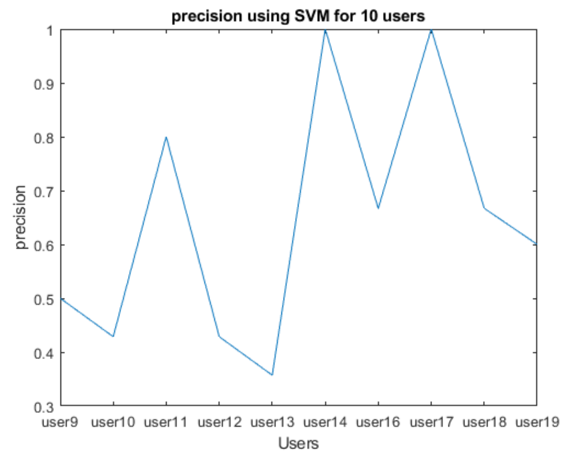
5. Results and Plots

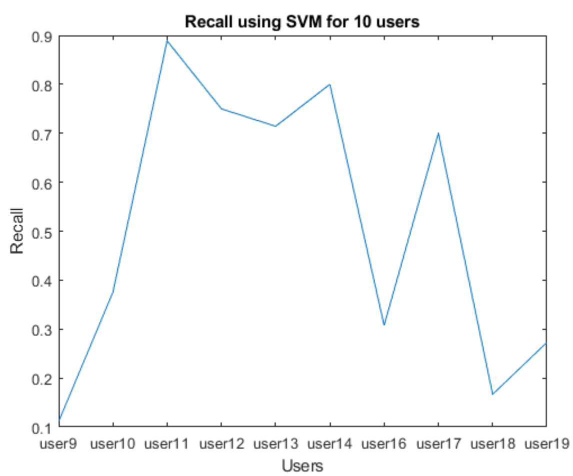
(i) When the Training dataset is 60% and Testing dataset is 40% (Assignment 2 and Assignment 3)

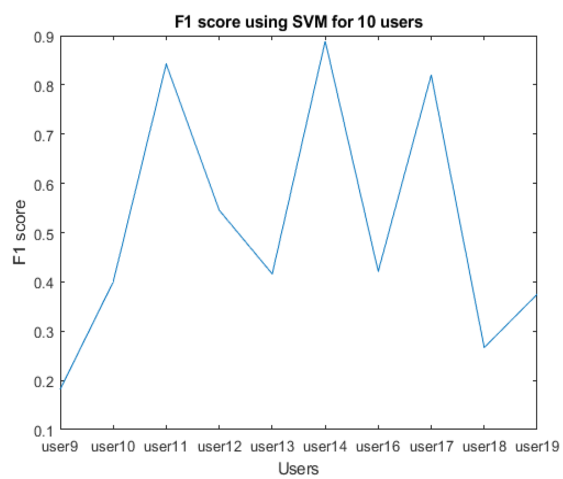


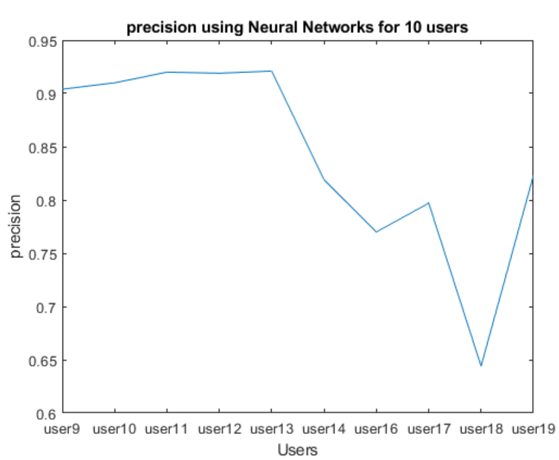


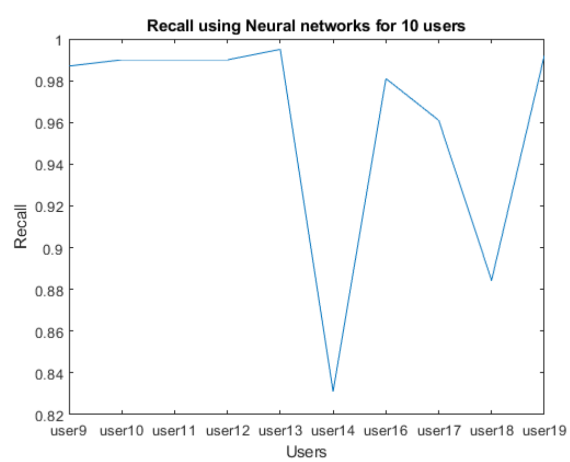

F1 score using Decision tree for 10 users
F1 score
user9 user10 user11 user12 user13 user14 user16 user17 user18 user19
Users


precision using SVM for 10 users
precision
user9 user10 user11 user12 user13 user14 user16 user17 user18 user19
Users


Recall using SVM for 10 users
Recall
user9 user10 user11 user12 user13 user14 user16 user17 user18 user19
Users


F1 score using SVM for 10 users
F1 score
user9 user10 user11 user12 user13 user14 user16 user17 user18 user19
Users


precision using Neural Networks for 10 users
precision
user9 user10 user11 user12 user13 user14 user16 user17 user18 user19
Users


Recall using Neural networks for 10 users
Recall
user9 user10 user11 user12 user13 user14 user16 user17 user18 user19
Users

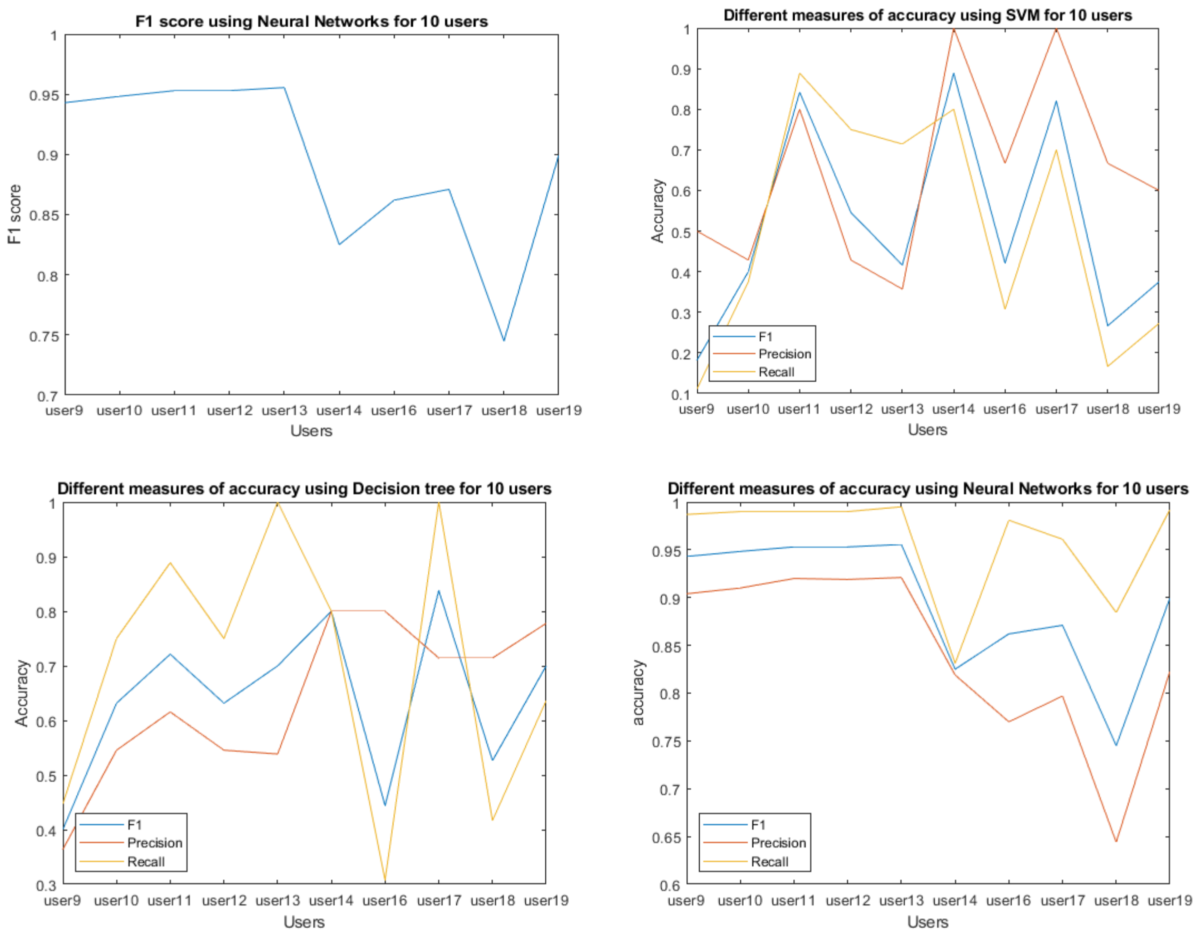
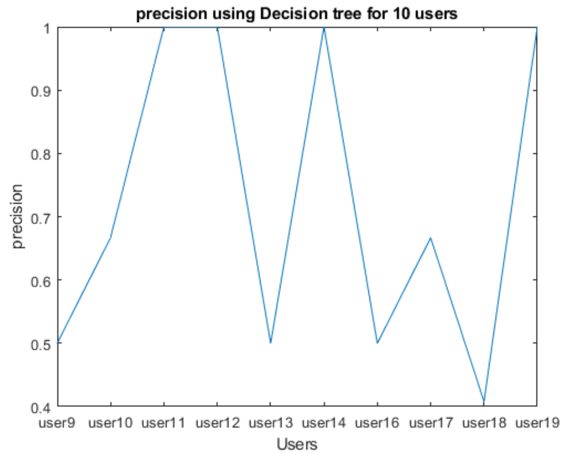
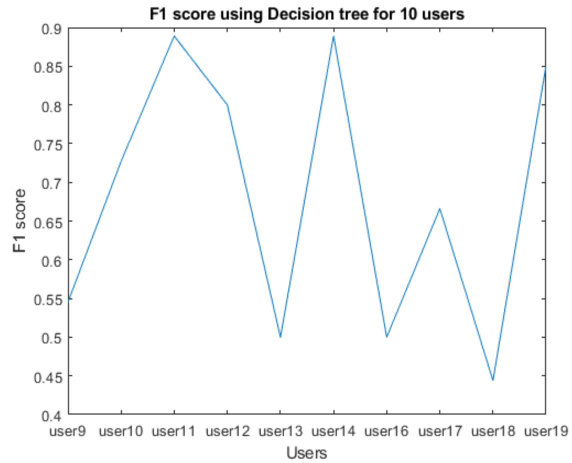
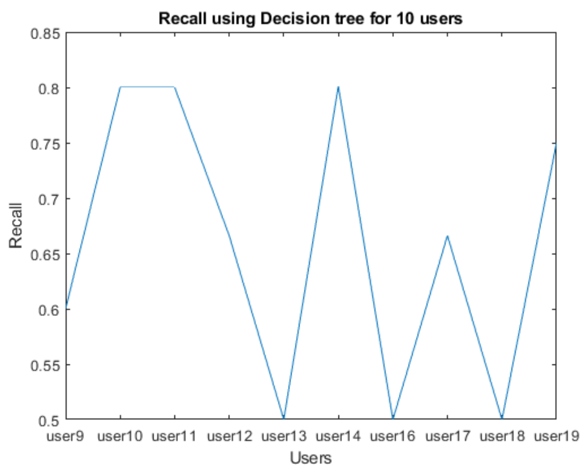


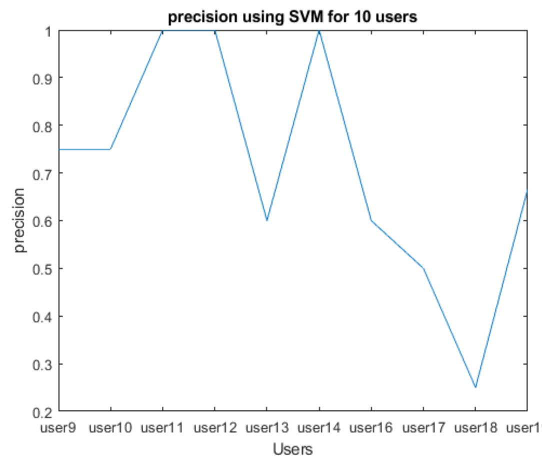
Fig 6. Plots of F1 score, precision, recall for three different classifiers for 60-40 split

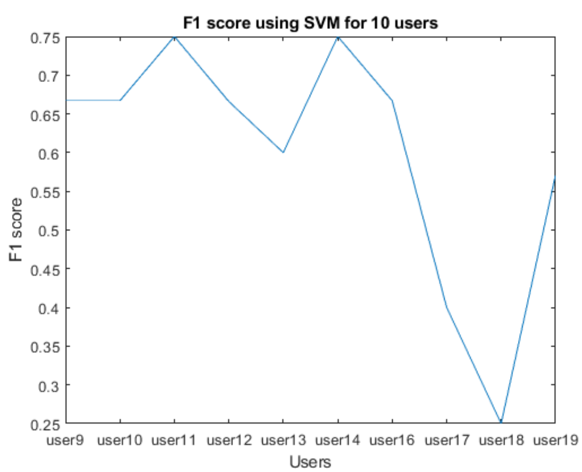
(ii) When the Training dataset is 80% and Testing dataset is 20%

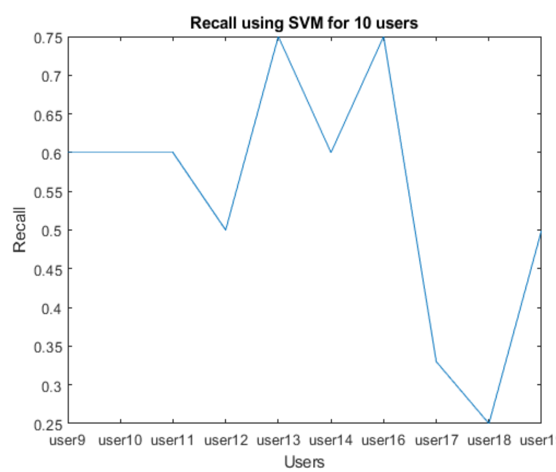

precision using Decision tree for 10 users
precision
1
0.9
0.8
0.7
0.6
0.5
0.4
user9 user10 user11 user12 user13 user14 user16 user17 user18 user19
Users


F1 score using Decision tree for 10 users
F1 score
0.9
0.85
0.8
0.75
0.7
0.65
0.6
0.55
0.5
0.45
0.4
user9 user10 user11 user12 user13 user14 user16 user17 user18 user19
Users


Recall using Decision tree for 10 users
Recall
0.85
0.8
0.75
0.7
0.65
0.6
0.55
0.5
user9 user10 user11 user12 user13 user14 user16 user17 user18 user19
Users


precision using SVM for 10 users
precision
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
user9 user10 user11 user12 user13 user14 user16 user17 user18 user19
Users


F1 score using SVM for 10 users
F1 score
0.75
0.7
0.65
0.6
0.55
0.5
0.45
0.4
0.35
0.3
0.25
user9 user10 user11 user12 user13 user14 user16 user17 user18 user19
Users


Recall using SVM for 10 users
Recall
0.75
0.7
0.65
0.6
0.55
0.5
0.45
0.4
0.35
0.3
0.25
user9 user10 user11 user12 user13 user14 user16 user17 user18 user19
Users

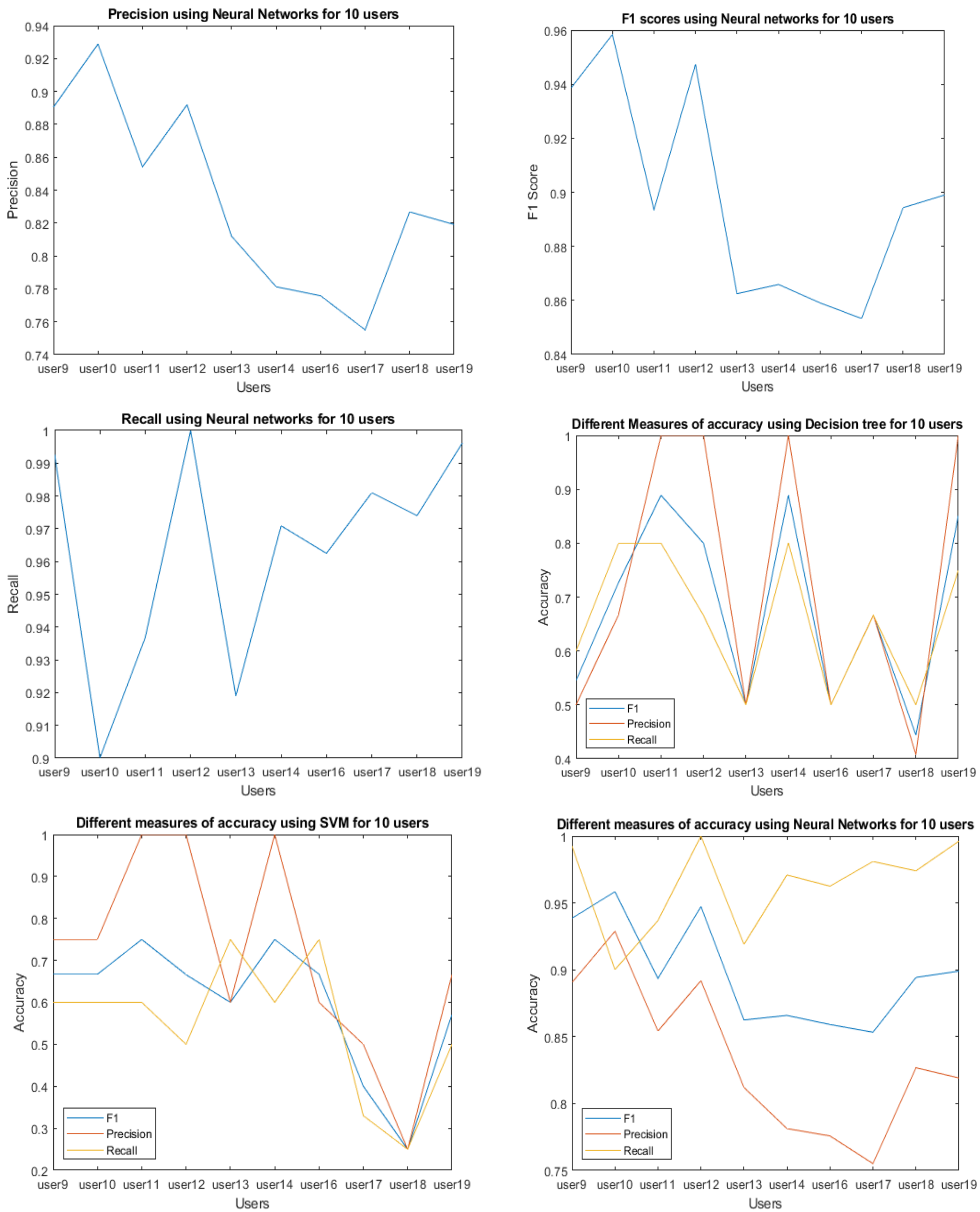


Fig 7. Plots of F1 score, precision, recall for three different classifiers for 80 - 20 split

5.1. Neural Network Plots for the 10 users:

(i) When the Training dataset is 60% and Testing dataset is 40%

User 9:

User 10:

User 11:

User 12:

User 13:

User 14:

User 16:

User 17:

User 18:

User 19:

5.2 Table of Results for comparison:

1. Train-Testing: 80-20

USER	SUPPORT VECTOR MACHINES			DECISION TREE			NEURAL NETWORKS		
	F1 Score	**Precision**	**Recall**	**F1 Score**	**Precision**	**Recall**	**F1 Score**	**Precision**	**Recall**
9	0.6670	0.7500	0.6000	0.5450	0.5000	0.6000	0.9385	0.8904	0.9926
10	0.6670	0.7500	0.6000	0.7270	0.6670	0.8000	0.9584	0.9288	0.9001
11	0.7500	1.0000	0.6000	0.8899	1.0000	0.8000	0.8935	0.8542	0.9368
12	0.6667	1.0000	0.5000	0.8000	1.0000	0.6660	0.9473	0.8999	1.0000
13	0.6000	0.5000	0.7500	0.5000	0.5000	0.5000	0.8625	0.8120	0.9190
14	**0.7500**	**1.0000**	**0.6000**	**0.8899**	**1.0000**	**0.8000**	**0.8659**	**0.7813**	**0.9709**
16	0.6667	0.6000	0.7500	0.5000	0.5000	0.5000	0.8591	0.7758	0.9625
17	0.4000	0.5000	0.3300	0.5000	0.6667	0.6667	0.8533	0.7551	0.9810
18	0.2500	0.2500	0.2500	0.4490	0.4000	0.5000	0.8943	0.8268	0.9740
19	0.5714	0.6661	0.5000	0.8591	1.0000	0.7500	0.8990	0.8191	0.9962

2. Training-Testing: 60-40

USER	SUPPORT VECTOR MACHINES			DECISION TREE			NEURAL NETWORKS		
	F1 Score	**Precision**	**Recall**	**F1 Score**	**Precision**	**Recall**	**F1 Score**	**Precision**	**Recall**
9	0.1818	0.5000	0.1110	0.4000	0.3636	0.4440	0.9430	0.9040	0.9870
10	0.4000	0.4286	0.3570	0.6316	0.5455	0.7500	0.9483	0.9100	0.9900
11	0.8421	0.8000	0.8889	0.7213	0.6154	0.8889	0.9530	0.9200	0.9900
12	0.5455	0.4286	0.7500	0.6316	0.5455	0.7500	0.9530	0.9190	0.9900
13	0.4162	0.3571	0.7143	0.7010	0.5388	1.0000	0.9556	0.9210	0.9950
14	**0.8889**	**1.0000**	**0.8001**	**0.8000**	**0.8000**	**0.8000**	**0.8250**	**0.8190**	**0.8311**
16	0.4211	0.6667	0.3077	0.4440	0.8000	0.3077	0.8620	0.7700	0.9810
17	0.8020	1.0000	0.7000	0.8380	0.7143	1.0000	0.8710	0.7970	0.1610
18	0.2267	0.6667	0.1667	0.5263	0.7143	0.4167	0.7450	0.6440	0.8844
19	0.3750	0.6000	0.2727	0.7000	0.7778	0.6364	0.8997	0.8230	0.9922

6. Conclusion:

For all the users, the feature matrix of the eating activity dataset was reduced using principal Component Analysis. The obtained reduced matrix was then classified based on the following partitions: Training-Testing: 80-20 and Training-Testing: 60-40. It is also observed that the 60-40 split works well for all the three classifiers and is able to distinguish the eating from non-eating activity. From the above results, Neural Networks seems to classify the data with higher accuracy than the other classifiers. From the users observed, **User 14** seems to have the highest values of F1 score, precision and recall when considering both partitions of the dataset.

7. References:

[1]Sushmalekha Shankar Birur [2018], *"Hand Gesture Recognition using Unsupervised Learning"*

[2]Smithsonian Air and Space Musuem, "Roll, Pitch and Yaw", https://howthingsfly.si.edu/flight-dynamics/roll-pitch-and-yaw

[3]Eider C. P. Silva, Esteban W. G. Clua, Anselmo A. Montenegro, 2015 14th Brazilian Symposium on Computer Games and Digital Entertainment, *"Sensor Data Fusion for full arm* tracking using Myo armband and Leap Motion"

[4]Principal Component Analysis, https://en.wikipedia.org/wiki/Principal_component_analysis

[5]MathWorks Documentation for PCA, https://www.mathworks.com/help/stats/pca.html

[6]MathWorks Documentation for Decision Tree, https://www.mathworks.com/help/stats/classificationtree.fit.html

[7]MathWorks Documentation for SVM, https://www.egr.msu.edu/classes/ece480/capstone/spring11/group04/application_Kan.pdf

[8]MathWorks Documentation for Neural networks, https://www.researchgate.net/publication/264742064_Neural_networks_in_Matlab

[9]Our group assignment 1 project (Group 4)

[10]Decision tree, SVM, neural networks images from Wikipedia.