CSE 572: Data Mining Fall 2017
Assignment 4

Feature Space Selection:

PCA was carried out on the best features obtained from the raw sensor data In the assignment 3. Using the best features' matrix was multiplied with the eigenvector containing highest eigenvalue to obtain dataset in the direction where it showed most variance. Thus, the feature space was obtained. The feature space obtained is used for the assignment 4.

Input Data (for all classification models):

Number of dimensions: 18 (same as from assignment 3)
Class (Eating action): 1 (last column)
Class (Non Eating action): 0 (last column)
For each user the data is picked from both the Eat_user and Noneat_user and shuffled, so that we train and test both the classes accurately.

Labelling of the classes:

In assignment-3 the raw data was filtered as eating action data and non eating action data. The two types of data were stored as two separate matrices (and csv files). Hence, it became simpler to assign the classes to the given dataset. For SVM and decision tree classifiers, the eating action is denoted as 1 and non eating as 0. For neural networks, one-hot encoding approach is used which is the standard used by neural network libraries in matlab. For e.g. label 0 becomes [1, 0] and label 1 becomes [0, 1]. In other words, the index at which the value is 1 in the newly created array is used to classify the instances. This is done because it is easier for softmax activation function to classify instances with a confidence/probability score.

Accuracy Metrics:

The accuracy models which are being used are as follows:

Precision: Total Number of true positives divided by the Total number of true positives and Total number of false positives. In other words, it is the number of positive predictions made divided by the number of total positive classes predicted.

Recall: Total Number of true positives divided by the Total number of true positives and Total number of false negatives. In other words, it is the number of positive predictions made divided by the actual number of total positive classes in the dataset.

F1 Score: The harmonic mean of the precision and recall values. Higher F1 score means higher precision and recall and better model.

ROC curve: When TP (true positives) are plotted against FP (false positives), then one obtains an ROC curve. A model is classified as a better model when the curve goes straight up the Y-axis and then along the X-axis.

Area under the curve (AUC): The AUC metric is used to compare two or models together which can give an idea of the better model among all of the models.

Three different machine learning models are used to perform binary classification for the PCA transformed dataset. The models are run for the following variations of the dataset:

1. All 33 users and reporting the matrix containing the above classification metrics for each and every individual user. The data for each user was shuffled during training and testing.
2. The whole dataset of 33 users is considered as one and split as training data (10 users) and testing data (remaining 23 users) and reporting the matrix containing the above classification metrics for each and every individual user in the testing set.

Model configurations:

The configuration table contains the parameters of the commands we have used for each algorithms. They were extracted from the matlab workspace.

1. Decision Tree:

# of Input features	18
Classification or Regression tree	Classification tree (fitctree)
Hyperparameters	Default hyperparameter values
Class names	0: Non Eat and 1: Eating Action
Num of Observations	1540
Algorithm used	CART algorithm (Default)

A decision tree is a tree-like representation of various data points segregated into classes based on the attributes they possess. In Matlab, the default functionality for binary classification using decision tree is given by the command fitctree. Here each row corresponds to an observation of the user dataset and each column represents the predictor variable. Two classes are formed in the decision tree, namely Eating and Non Eating. In decision trees, the model is trained and two classes are formed based on the attributes of the training dataset. Once all the datasets are trained, testing is carried out where in each test data is compared to the most similar class attribute, the higher the similarity in attributes, the testing dataset is classified in that class. Thus in our case, two classes of Eating and Non Eating were created based on the training data we provided in both the phases and results were predicted. In case of a missing value, fitctree considers it as NaN. In decision tree, if the training dataset is very high or if it's very similar to the testing dataset, then the classification takes place efficiently.

2. Support Vector Machines:

# of Input features	18
Type of SVM	Classification SVM (fitcsvm)
Class names	0: Non Eat and 1: Eating Action
Num of Observations	1540
Kernel Used	Linear (Default)
Bias	2.3472
Solver	Sequential minimal optimization(SMO)

By default SVM uses linear kernel to define a model. SVM uses the classification score to classify an instance x to be of a particular class (binary outcome in this case). The signed distance of instance x is computed from the decision boundary ranging from -inf to +inf. A positive score means class belonging to label 1 and negative score means it belongs to label 0.
The SVM predict method gives probabilities for different classes which is then used to classify instances into the classes based on the highest posterior probability score.

However, by plotting the graphs of the PCA transformed dataset, it can be inferred that the data is not linearly separable and hence linear SVM won't be able to arrive at better results. The SVM by default is a hard margin classifier which means it can separate the two classes with a linear hyperplane. But, the soft margin classifier has to be used because the data is messy. Hence, we can allow some points in the training data to cross the margin. How many data points can cross the margin depends on the value of some coefficients (slack variables). This increases the complexity of the model and can lead to overfitting. We have to be careful here not to introduce overfitting and accordingly adjust the coefficients. One such coefficient is called C which defines the magnitude of the margin allowed in all dimensions. It defines the amount of violation of the margin.

- Smaller value of C means more prone to overfitting (high variance, lower bias)
- Larger value of C means less prone to overfitting (low variance, high bias)

We will want to minimise the sum of prediction error as well as the slack error in training to categorise the data points in test data correctly.

3. Neural Networks:

# of input layer neurons (# of features)	18
Type of Neural Network	Feed-forward NN
Number of hidden layers	1 (default)
Number of neurons in each hidden layer	5 (default)
Epochs	25 (Number of full iterations over trainset)
Activation function	Sigmoid for hidden layer, softmax (in this case sigmoid, again) for output layer
Labels	One-hot encoding

In a feedforward neural network, the input features are fed to hidden layers. The connection to each neuron in the hidden layer contains some weight which are multiplied with each input feature and a bias term is added. Each neuron is then activated depending upon the weights and the features. During the training phase, the model collects all the errors using binary cross entropy function and after each epoch it back propagates the error to all the weights in the respective layers and slightly changes the weights in the direction of minimal error. The softmax function is used to squish the output of the fully connected layer a layer of number of classes which gives result in terms of probability. This feedforward and backpropagation keeps on taking place for 15 epochs (in this case) such that the model learns the pattern in the data pertaining to the respective class labels.

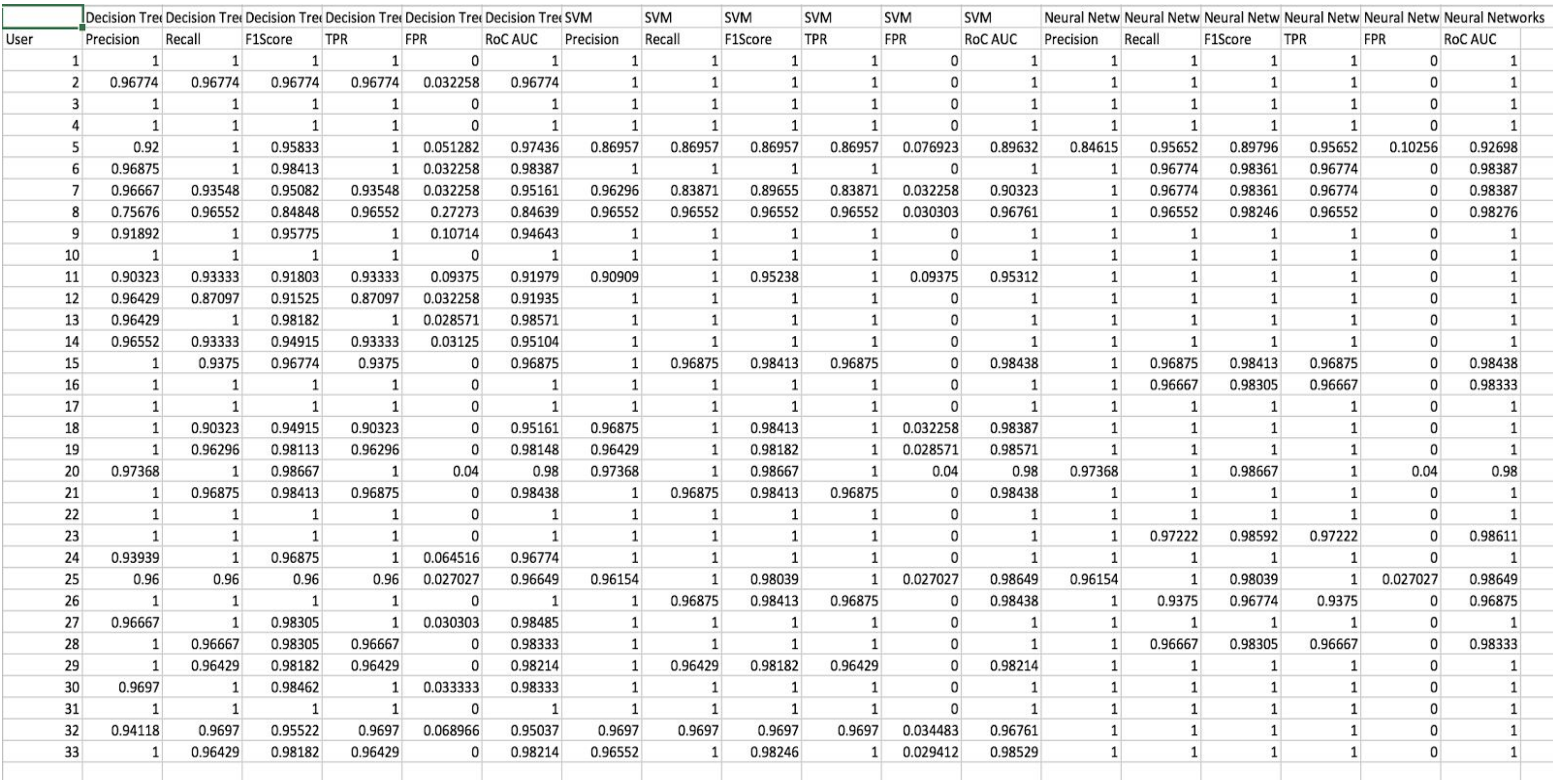
Phase 1: User dependent analysis:

In this task, we have used the dataset of total 33 users, to individually test and train on each user. So in this task, we split the dataset of each user, such that **per user** we have:

Training dataset: 60%
Testing Dataset: 40%

In this case, the metrics defined would be quiet efficient and easily classifiable because, we are training and testing using the same set of dataset for each of the 33 users. We can even interpret that, if testing and training is performed on the same user data, it is highly possible that the tests that we perform on the dataset, were already trained and so the accuracy of classification will be really high in this case. Since we had a feature space of all users combined, taken from assignment 3, we appended the user database from both eat and Non-eat matrix and randomized it based on each user. We then applied the machine learning algorithms to each user such that, we used 60% of it for training and 40% for testing.
We have calculated Precision, Recall, F1 Score, TPR, FPR and AUC under the ROC curve and shown in the matrix.

The matrix are defined based on three algorithms: Decision Tree, SVM and Neural Networks based on the **parameters and explanation of the working of each ML algorithm is defined above.**



	Decision Tre	Decision Tre	Decision Tre	Decision Tre	Decision Tre	Decision Tre	SVM	SVM	SVM	SVM	SVM	SVM	Neural Netw	Neural Netw	Neural Netw	Neural Netw	Neural Netw	Neural Networks
User	Precision	Recall	F1Score	TPR	FPR	RoC AUC	Precision	Recall	F1Score	TPR	FPR	RoC AUC	Precision	Recall	F1Score	TPR	FPR	RoC AUC
1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1
2	0.96774	0.96774	0.96774	0.96774	0.032258	0.96774	1	1	1	1	0	1	1	1	1	1	0	1
3	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1
4	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1
5	0.92	1	0.95833	1	0.051282	0.97436	0.86957	0.86957	0.86957	0.86957	0.076923	0.89632	0.84615	0.95652	0.89796	0.95652	0.10256	0.92698
6	0.96875	1	0.98413	1	0.032258	0.98387	1	1	1	1	0	1	1	0.96774	0.98361	0.96774	0	0.98387
7	0.96667	0.93548	0.95082	0.93548	0.032258	0.95161	0.96296	0.83871	0.89655	0.83871	0.032258	0.90323	1	0.96774	0.98361	0.96774	0	0.98387
8	0.75676	0.96552	0.84848	0.96552	0.27273	0.84639	0.96552	0.96552	0.96552	0.96552	0.030303	0.96761	1	0.96552	0.98246	0.96552	0	0.98276
9	0.91892	1	0.95775	1	0.10714	0.94643	1	1	1	1	0	1	1	1	1	1	0	1
10	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1
11	0.90323	0.93333	0.91803	0.93333	0.09375	0.91979	0.90909	1	0.95238	1	0.09375	0.95312	1	1	1	1	0	1
12	0.96429	0.87097	0.91525	0.87097	0.032258	0.91935	1	1	1	1	0	1	1	1	1	1	0	1
13	0.96429	1	0.98182	1	0.028571	0.98571	1	1	1	1	0	1	1	1	1	1	0	1
14	0.96552	0.93333	0.94915	0.93333	0.03125	0.95104	1	1	1	1	0	1	1	1	1	1	0	1
15	1	0.9375	0.96774	0.9375	0	0.96875	1	0.96875	0.98413	0.96875	0	0.98438	1	0.96875	0.98413	0.96875	0	0.98438
16	1	1	1	1	0	1	1	1	1	1	0	1	1	0.96667	0.98305	0.96667	0	0.98333
17	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1
18	1	0.90323	0.94915	0.90323	0	0.95161	0.96875	1	0.98413	1	0.032258	0.98387	1	1	1	1	0	1
19	1	0.96296	0.98113	0.96296	0	0.98148	0.96429	1	0.98182	1	0.028571	0.98571	1	1	1	1	0	1
20	0.97368	1	0.98667	1	0.04	0.98	0.97368	1	0.98667	1	0.04	0.98	0.97368	1	0.98667	1	0.04	0.98
21	1	0.96875	0.98413	0.96875	0	0.98438	1	0.96875	0.98413	0.96875	0	0.98438	1	1	1	1	0	1
22	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1
23	1	1	1	1	0	1	1	1	1	1	0	1	1	0.97222	0.98592	0.97222	0	0.98611
24	0.93939	1	0.96875	1	0.064516	0.96774	1	1	1	1	0	1	1	1	1	1	0	1
25	0.96	0.96	0.96	0.96	0.027027	0.96649	0.96154	1	0.98039	1	0.027027	0.98649	0.96154	1	0.98039	1	0.027027	0.98649
26	1	1	1	1	0	1	1	0.96875	0.98413	0.96875	0	0.98438	1	0.9375	0.96774	0.9375	0	0.96875
27	0.96667	1	0.98305	1	0.030303	0.98485	1	1	1	1	0	1	1	1	1	1	0	1
28	1	0.96667	0.98305	0.96667	0	0.98333	1	1	1	1	0	1	1	0.96667	0.98305	0.96667	0	0.98333
29	1	0.96429	0.98182	0.96429	0	0.98214	1	0.96429	0.98182	0.96429	0	0.98214	1	1	1	1	0	1
30	0.9697	1	0.98462	1	0.033333	0.98333	1	1	1	1	0	1	1	1	1	1	0	1
31	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1
32	0.94118	0.9697	0.95522	0.9697	0.068966	0.95037	0.9697	0.9697	0.9697	0.9697	0.034483	0.96761	1	1	1	1	0	1
33	1	0.96429	0.98182	0.96429	0	0.98214	0.96552	1	0.98246	1	0.029412	0.98529	1	1	1	1	0	1

The Result_phase1.csv file can be found in the main folder. Below is the snapshot of the same.

Explanation for Decision tree:

The decision tree is based on the binary classification of classes 0 and 1 where class 0 denotes Non Eating action and class 1 represents Eating actions. We have used the inbuilt function of Matlab, fitctree with which we are splitting the data into binary classification of two classes. As we can observe from the above matrix, the F1 score is very high for almost all users, as we perform testing on a similar dataset on which the training was done. It varies from mainly from 0.95 to 1, where values closer to 1 denote better representation of the dataset by the model used. Comparing it with all the algorithms, it has a lesser Area Under the Curve, which shows, it is not as optimal an algorithm as others.

Explanation for Support Vector Machine:

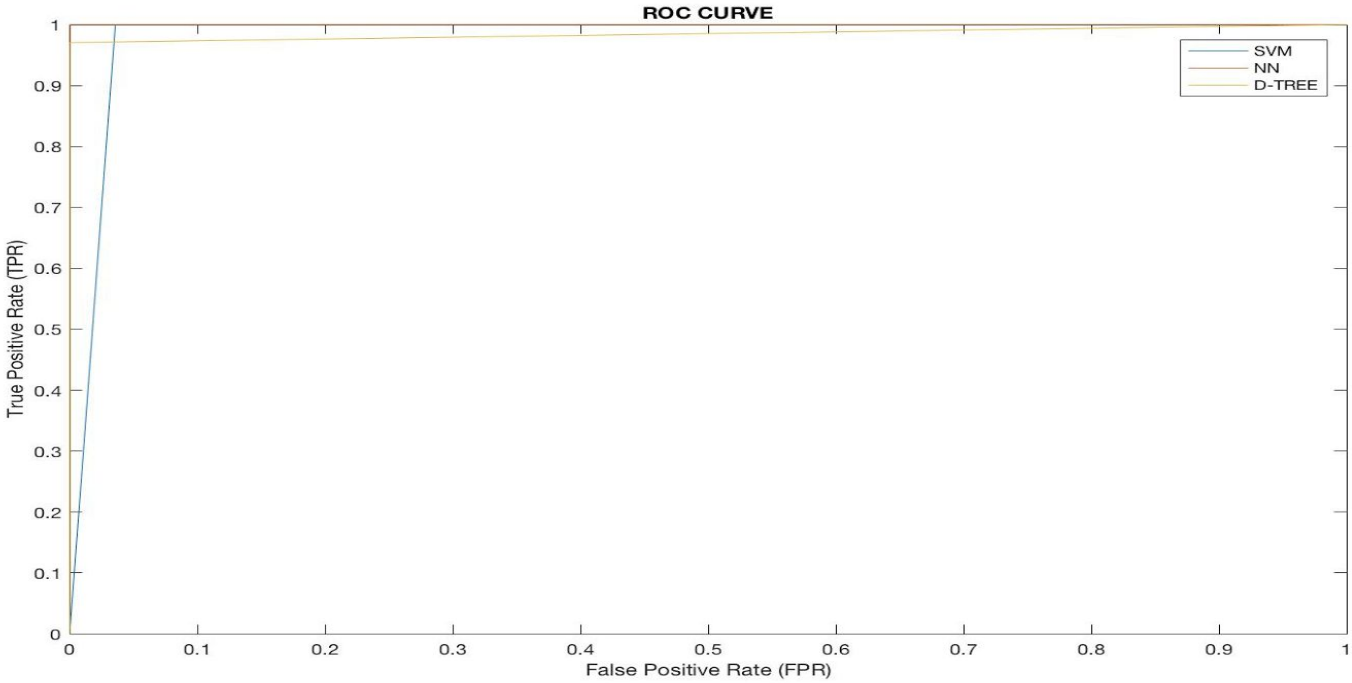
Similar to decision trees, its class 0 represents Non Eating Action and class 1 represents Eating Actions. Based on the predictor features algorithms used by fitcsvm, it calculates probabilities for a particular class. Based on the output of the F1 score, we can see how well is the user data classified.The F1 score varies from 0.96 to 1, where the higher the F1 score, the better classification of the dataset is performed by the model. Comparing it to other models, it classifies better than Decision trees, and almost similar to Neural Networks in this case.

Explanation for Neural Network:

The number of features (18) corresponds to the number of neurons in the input layer of the neural network. With a single hidden layer (default) and 5 neurons in each layer, the Matlab library was used for Neural Network. Feed forward neural network was used for the processing part which was activated by the sigmoid function. The F1 score of the Neural network, is almost similar to SVM and bit better than Decision tree algorithm for this dataset.

Based on the final results,thus, we were able to say that the accuracy of predictions made by the Neural Networks and SVM algorithm are the very good. We verified this by plotting the ROC curve of a single user.

We plotted the comparison between each of this algorithms using the ROC curve where we captured the overall efficiency based on one of the user as the sample set. The best classification accuracy is shown by the graph which has the maximum area under the curve.



In this case each of the algorithms perform really well and it is hard to distinguish the best of them. But analysing the results, based on every user graph and the one plotted, we can derive that Neural Networks and SVM perform better than Decision tree in this case.

Phase 2: User independent analysis

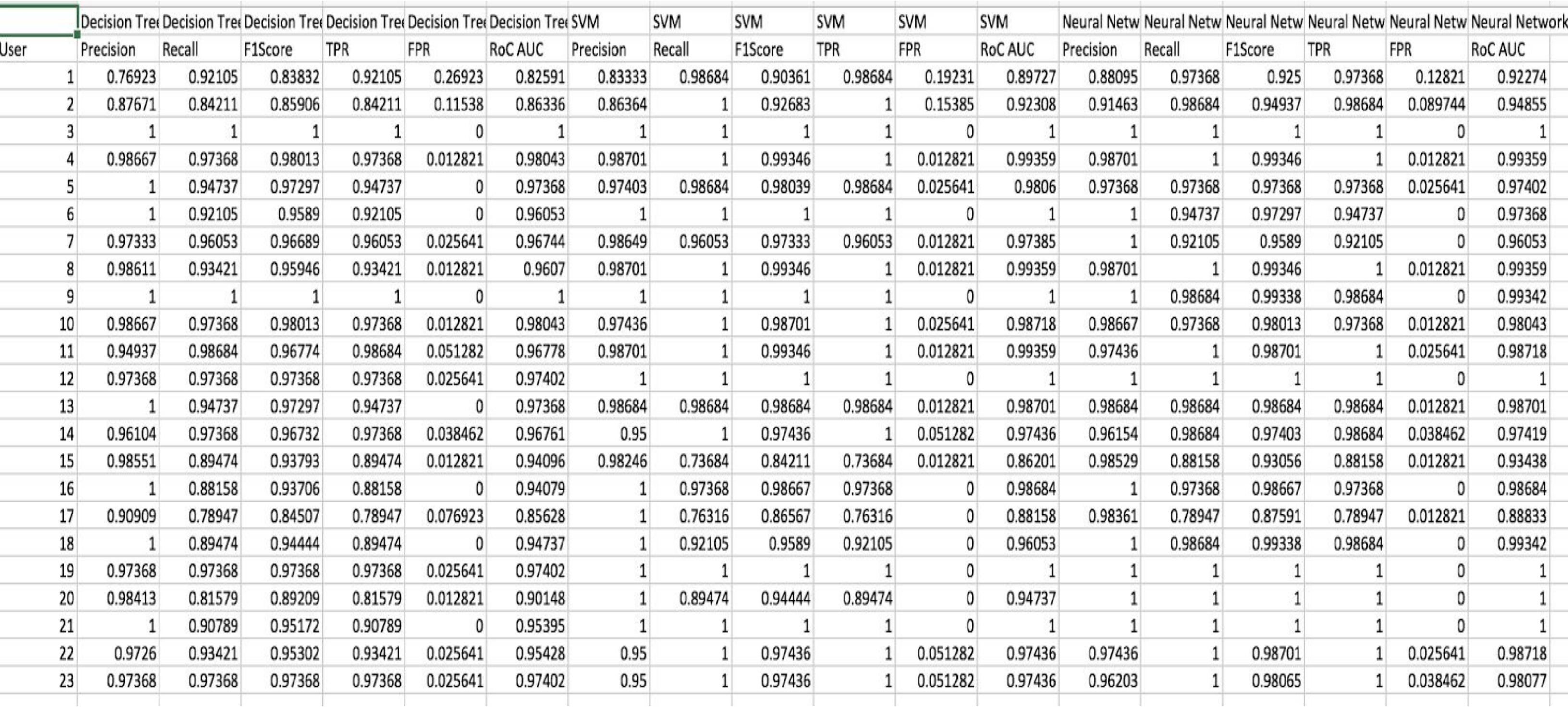
In this phase, we have used the total dataset of 33 users to train and test the given input of the feature space for different machine learning algorithms using the following as the training and testing datasets. For this phase:

Training on each algorithm using 10 users dataset.
Test using each algorithm for the remaining 23 users.

Using the above mentioned divisions, we ran three Machine Learning algorithms based on the above mentioned parameters. In this case, total 23 users were tested based on the training model generated by the 10 users dataset. We got the following matrix consisting of various metrics:

As shown in the below matrix, we have calculated Precision, Recall, F1 Score, TPR, FPR and AUC under the ROC curve and shown in the matrix.

Each of this are given based on three algorithms: Decision Tree, SVM and Neural Networks based on the **parameters and explanation of the working of each ML algorithm is defined above.**



	Decision Tre	Decision Tre	Decision Tre	Decision Tre	Decision Tre	Decision Tre	SVM	SVM	SVM	SVM	SVM	SVM	Neural Netw	Neural Netw	Neural Netw	Neural Netw	Neural Netw	Neural Network
User	Precision	Recall	F1Score	TPR	FPR	RoC AUC	Precision	Recall	F1Score	TPR	FPR	RoC AUC	Precision	Recall	F1Score	TPR	FPR	RoC AUC
1	0.76923	0.92105	0.83832	0.92105	0.26923	0.82591	0.83333	0.98684	0.90361	0.98684	0.19231	0.89727	0.88095	0.97368	0.925	0.97368	0.12821	0.92274
2	0.87671	0.84211	0.85906	0.84211	0.11538	0.86336	0.86364	1	0.92683	1	0.15385	0.92308	0.91463	0.98684	0.94937	0.98684	0.089744	0.94855
3	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1
4	0.98667	0.97368	0.98013	0.97368	0.012821	0.98043	0.98701	1	0.99346	1	0.012821	0.99359	0.98701	1	0.99346	1	0.012821	0.99359
5	1	0.94737	0.97297	0.94737	0	0.97368	0.97403	0.98684	0.98039	0.98684	0.025641	0.9806	0.97368	0.97368	0.97368	0.97368	0.025641	0.97402
6	1	0.92105	0.9589	0.92105	0	0.96053	1	1	1	1	0	1	1	0.94737	0.97297	0.94737	0	0.97368
7	0.97333	0.96053	0.96689	0.96053	0.025641	0.96744	0.98649	0.96053	0.97333	0.96053	0.012821	0.97385	1	0.92105	0.9589	0.92105	0	0.96053
8	0.98611	0.93421	0.95946	0.93421	0.012821	0.9607	0.98701	1	0.99346	1	0.012821	0.99359	0.98701	1	0.99346	1	0.012821	0.99359
9	1	1	1	1	0	1	1	1	1	1	0	1	1	0.98684	0.99338	0.98684	0	0.99342
10	0.98667	0.97368	0.98013	0.97368	0.012821	0.98043	0.97436	1	0.98701	1	0.025641	0.98718	0.98667	0.97368	0.98013	0.97368	0.012821	0.98043
11	0.94937	0.98684	0.96774	0.98684	0.051282	0.96778	0.98701	1	0.99346	1	0.012821	0.99359	0.97436	1	0.98701	1	0.025641	0.98718
12	0.97368	0.97368	0.97368	0.97368	0.025641	0.97402	1	1	1	1	0	1	1	1	1	1	0	1
13	1	0.94737	0.97297	0.94737	0	0.97368	0.98684	0.98684	0.98684	0.98684	0.012821	0.98701	0.98684	0.98684	0.98684	0.98684	0.012821	0.98701
14	0.96104	0.97368	0.96732	0.97368	0.038462	0.96761	0.95	1	0.97436	1	0.051282	0.97436	0.96154	0.98684	0.97403	0.98684	0.038462	0.97419
15	0.98551	0.89474	0.93793	0.89474	0.012821	0.94096	0.98246	0.73684	0.84211	0.73684	0.012821	0.86201	0.98529	0.88158	0.93056	0.88158	0.012821	0.93438
16	1	0.88158	0.93706	0.88158	0	0.94079	1	0.97368	0.98667	0.97368	0	0.98684	1	0.97368	0.98667	0.97368	0	0.98684
17	0.90909	0.78947	0.84507	0.78947	0.076923	0.85628	1	0.76316	0.86567	0.76316	0	0.88158	0.98361	0.78947	0.87591	0.78947	0.012821	0.88833
18	1	0.89474	0.94444	0.89474	0	0.94737	1	0.92105	0.9589	0.92105	0	0.96053	1	0.98684	0.99338	0.98684	0	0.99342
19	0.97368	0.97368	0.97368	0.97368	0.025641	0.97402	1	1	1	1	0	1	1	1	1	1	0	1
20	0.98413	0.81579	0.89209	0.81579	0.012821	0.90148	1	0.89474	0.94444	0.89474	0	0.94737	1	1	1	1	0	1
21	1	0.90789	0.95172	0.90789	0	0.95395	1	1	1	1	0	1	1	1	1	1	0	1
22	0.9726	0.93421	0.95302	0.93421	0.025641	0.95428	0.95	1	0.97436	1	0.051282	0.97436	0.97436	1	0.98701	1	0.025641	0.98718
23	0.97368	0.97368	0.97368	0.97368	0.025641	0.97402	0.95	1	0.97436	1	0.051282	0.97436	0.96203	1	0.98065	1	0.038462	0.98077

The Result_phase2.csv file can be found in the main folder. Below is the snapshot of the same.

Explanation for Decision tree:

The decision tree is based on the binary classification of classes 0 and 1 where class 0 denotes Non Eating action and class 1 represents Eating actions. We have used the inbuilt function of Matlab, fitctree with which we are splitting the data into binary classification of two classes. As we can observe from the above matrixs, the F1 score varies from the range of 0.83 to 1, where values closer to 1 denote better representation of the dataset by the model used. Comparing it with all the algorithms, it has a lesser Area Under the Curve, which shows, it is not as optimal an algorithm as others.

Explanation for Support Vector Machine:

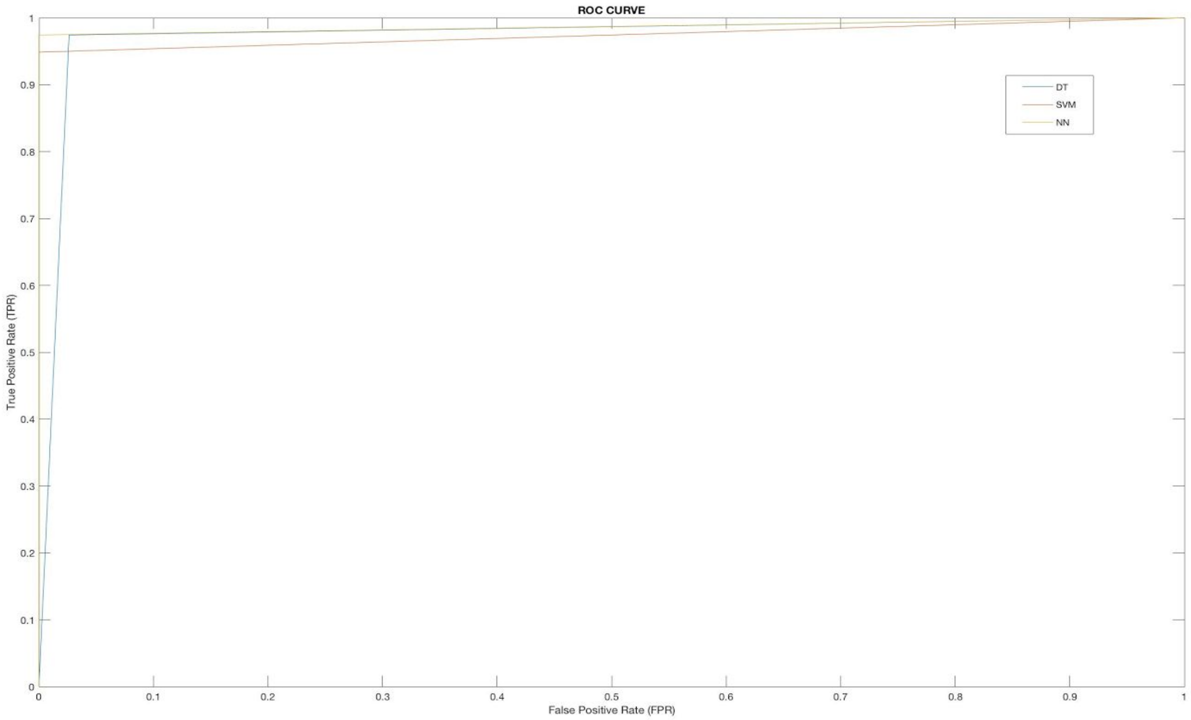
Similar to decision trees, its class 0 represents Non Eating Action and class 1 represents Eating Actions. Since the data is non separable, so the matlab function fitcsvm minimize the L1 norm problem using various methodologies which are defined in the function directly in the Matlab. Based on the predictor features algorithms used by fitcsvm, it calculates probabilities for a particular class. Based on the output of the F1 score, we can see how well is the user data classified. The higher the F1 score, the better classification of the dataset is performed by the model. Comparing it to other models, it classifies better than Decision trees, and almost similar to Neural Networks in this case.

Explanation for Neural Network:

The number of features (18) corresponds to the number of neurons in the input layer of the neural network. With a single hidden layer (default) and 5 neurons in each layer, the Matlab library was used for Neural Network. Feed forward neural network was used for the processing part which was activated by the sigmoid function. The F1 score of the Neural network, is comparatively the highest amongst all the three algorithms for this dataset.

Based on the final results,thus, we were able to say that the accuracy of predictions made by the Neural Networks algorithm were the best of all. We verified this by plotting the ROC curve.

We were also able to plot the comparison between each of this algorithms using the ROC curve which captured the overall efficiency based on the training done on 10 users and testing done on remaining 23 users. The best classification accuracy is shown by the graph which has the maximum area under the curve.



As verified using the ROC curve plotted for all the users, we see that Neural Network is the best Machine Learning algorithm in this case, while SVM and Decision tree have a little less Area under the curve.

References:

1. https://www.mathworks.com