Feature Selection and Classification.

Project Report 3: Rutvij Dhotey
Pattern Recognition and Machine Learning.

In this project , instead of projecting all the features down to a lower dimensional space, we learned how we have to select a specific set of features of the given one, so as to obtain the best classification results. I approached the project in the same way stated in the project description.

Part 1: Selection and Ranking

I used Variation Ratio to select and rank my top features according to the discriminative power calculated of every feature. I also calculated the values using Augmented Variance Ratio, but as I got better results using Variance Ratio, I used the same for final calculations. The graphs for both the VR and AVR (Face Data Set) and VR(EEG dataset) are attached below.

As you can clearly see every feature has its own discriminative value and we can rank the features according to the same.

The formulae used for AVR and VR are stated as follows:

$$\mathrm{VR}(F) = \frac{\mathrm{Var}(S_F)}{1/C \sum_{k=1,\ldots,C} \mathrm{Var}_k(S_F)},$$

$$\mathrm{AVR}(F) = \frac{\mathrm{Var}(S_F)}{(1/C) \sum_{i=1,\ldots,C} (\mathrm{Var}_i(S_F) / \min_{i \neq j}(|\mathrm{mean}_i(S_F) - \mathrm{mean}_j(S_F)|)},$$

Figures:

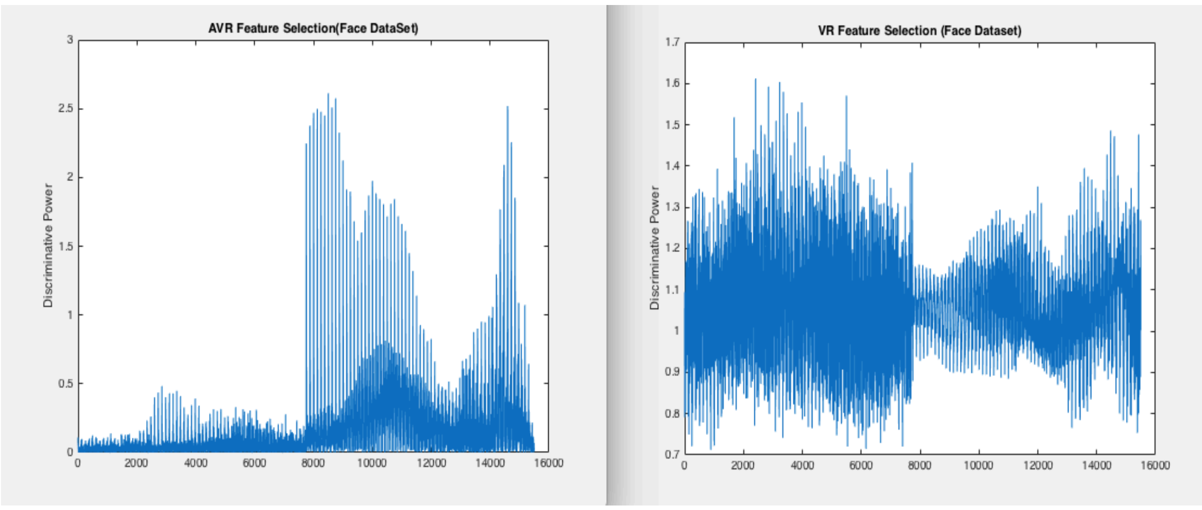


Figure 1: VR and AVR results for Face DataSet

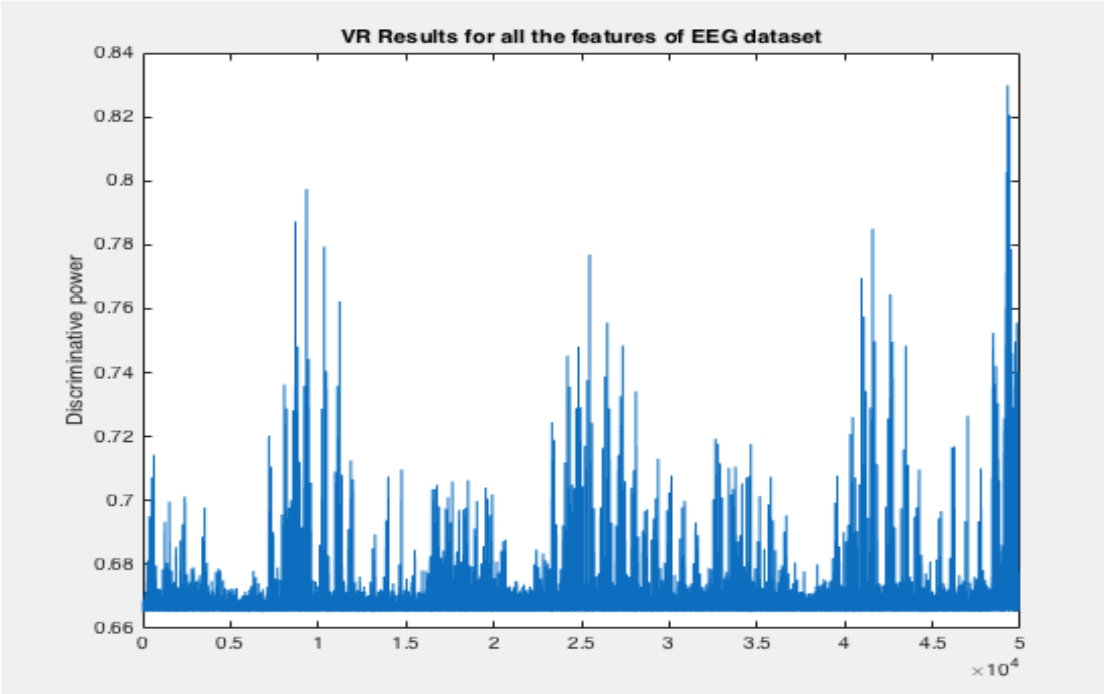


Figure 2: VR Results for EEG dataset.

Observations for the Feature Visualisation for both Datasets:

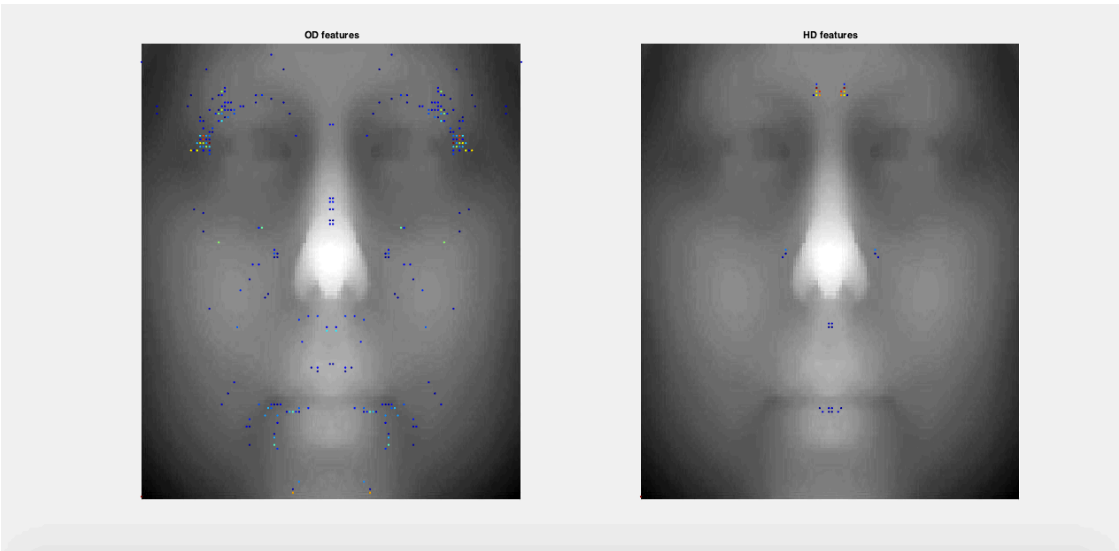


Figure 3. Top 10% Features of the VR Results (Face Dataset)

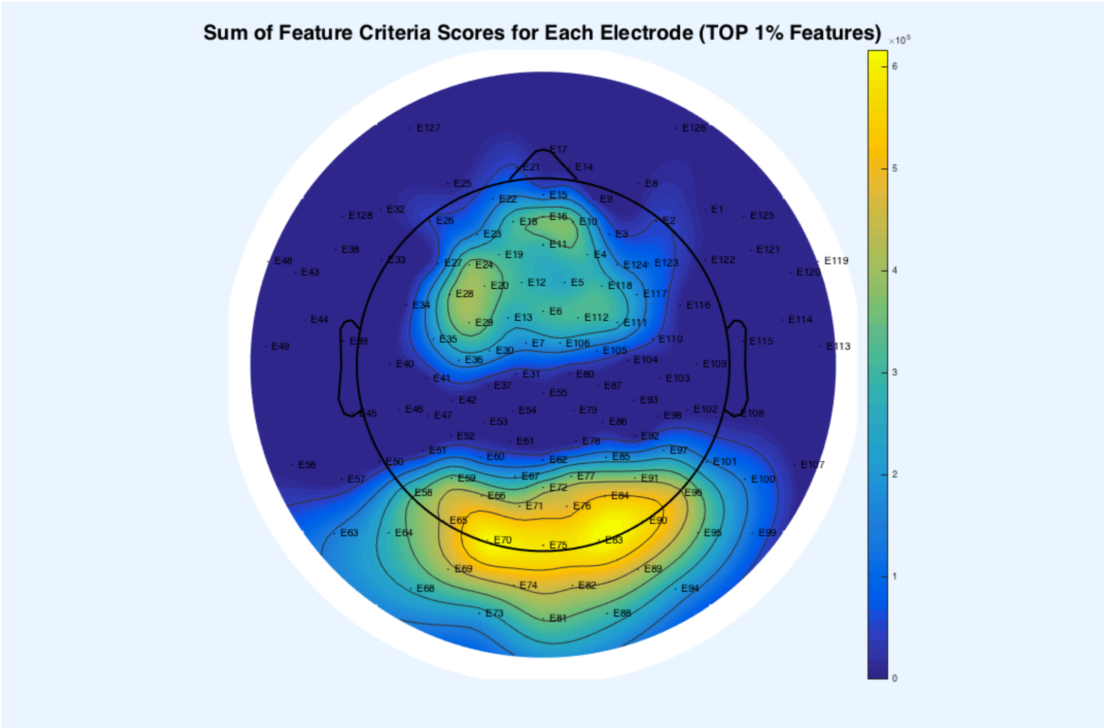


Figure 4: Top 1% Feautes of VR results (EEG Dataset)

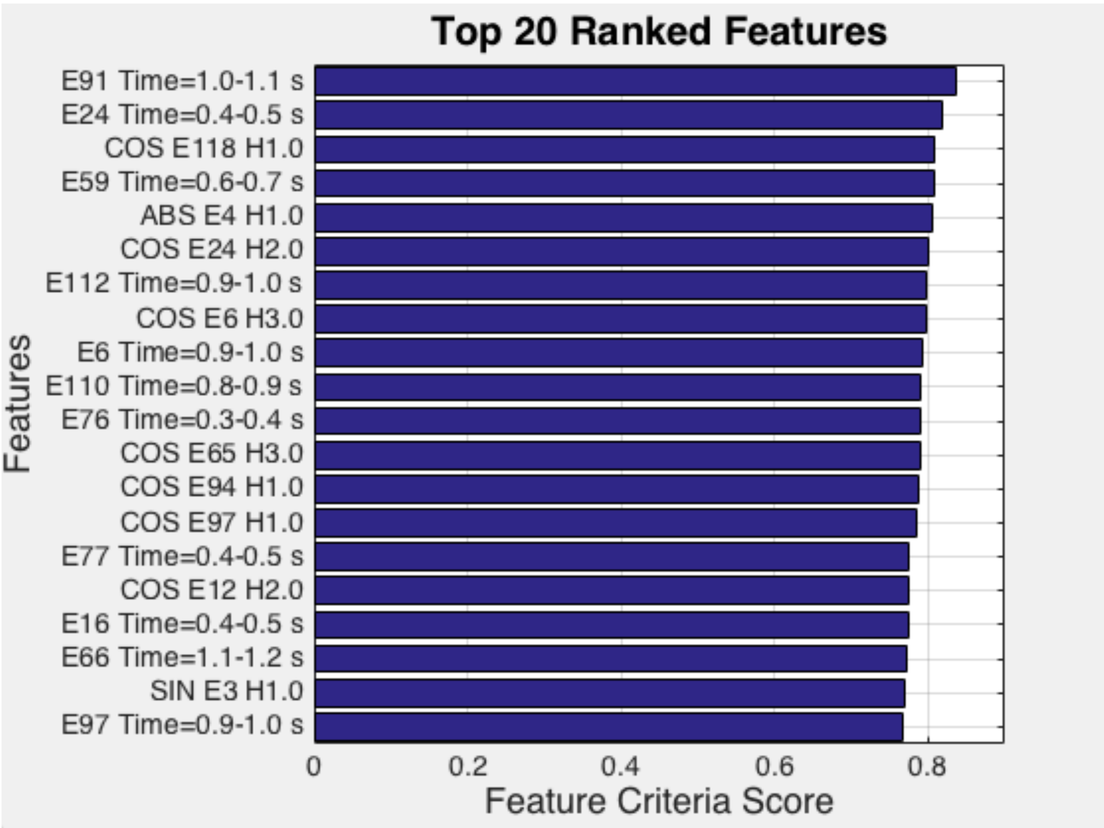


Figure 5

The above three figures give us an idea visually to which features are more likely to be used in the actual classification. In simple terms, which features would suffice to classify the data into their respective classes. We then use these features and build on to our wrapper to obtain the forward selected features.

From figure 5 we can see clearly that the feature criteria score decreases for the features stated and we used these features for forward selection.

Part 2: Wrapper (Forward Feature Selection)

After selecting top 10% features of the ranked matrix, we implement the wrapper function . This is where the Forward Feature Selection comes in. The algorithm is shown in the form of a flowchart drawn below. I have used 2 Classifiers in my entire working of the project. For the forward feature selection I have used my own classifier of LSE or MSE as its called. And for the final classification I have used the KNN(K nearest Neighbor) classifier.

After running the ranking feature function, we obtain a top selected feature matrix giving us the top 10%(face) and top 1%(EEG) features according to their discriminative powers. Using those top features, we input them into our fowardselection function which uses MSE classification to calculate the error rates after classification and at the end return us the final forwardselected feature vector that can be used for final data selection.

The equation used for MSE weight vector calculation is :

$$\mathbf{W^{*}= (X^{T}X)^{-1} * X^{T} * T}$$

1. Initially we solve the problem of linear regression by the technique of error minimisation. The minimising of sum of squares error function will give us a straight forward vector formula as calculated in the first homework.
   The Formula can be stated as: X is the training input and T is the training output.
   $$\mathbf{W^{*}= (X^{T}X)^{-1} * X^{T} * T}$$
2. Thus here W* is the coefficient parameters that we calculate to train wrt. the input / training data.
4. As we have to calculate the output y(X,W*)

$$\mathbf{y(X,W^{*})}= \sum_{i=0}^{M} \boldsymbol{W0 + W1 * Xi + W2 * Xi^{2} \ldots\ldots WM * Xi\text{^}M}$$

By using y(X,W*) we calculate the values of the classes(K encoding done). After that I calculate the error for the mis-classified inputs and then repeat this MSE.

The algorithm is stated below:

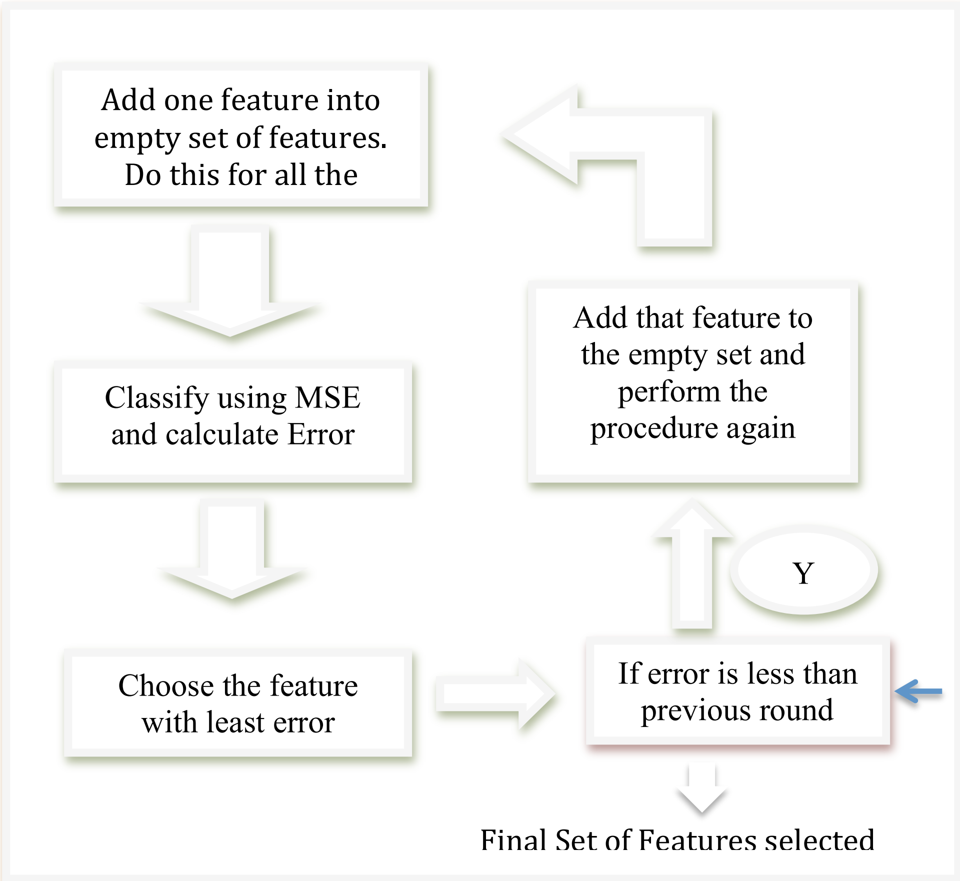


Figure 5: Algorithm for Forward Selection Features.

This algorithm is has the stopping condition stated in the blue. Whenever we stop , the forwardselected features are calculated and then these a new training dataset is tailored corresponding to these features.

During the forward selection we get the following graphs and figures telling us how many times they have occurred in the calculation of the forwardselected.

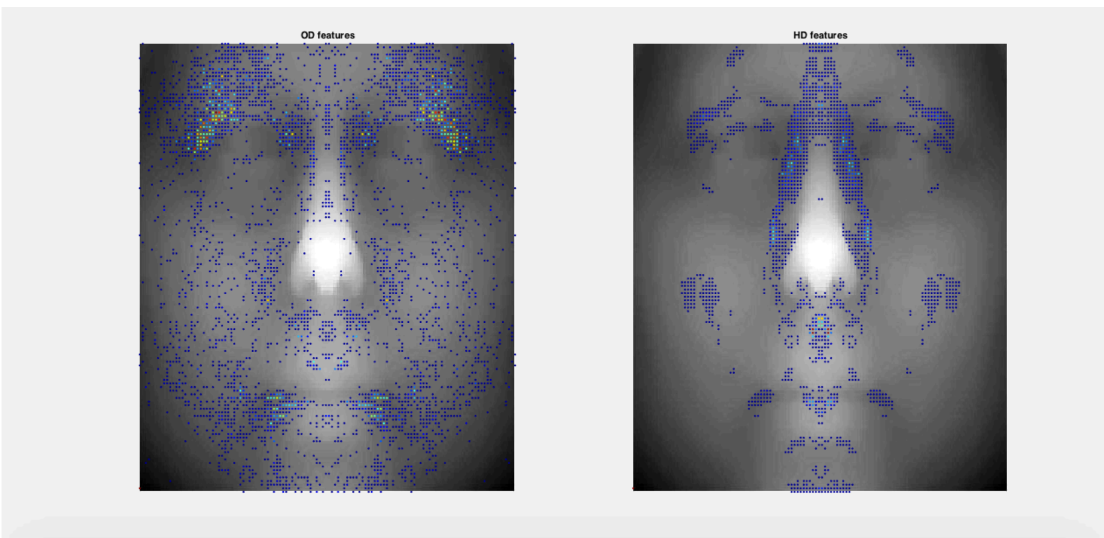


Figure 6: Top 20% features appearing in the forward selection (Face Dataset).

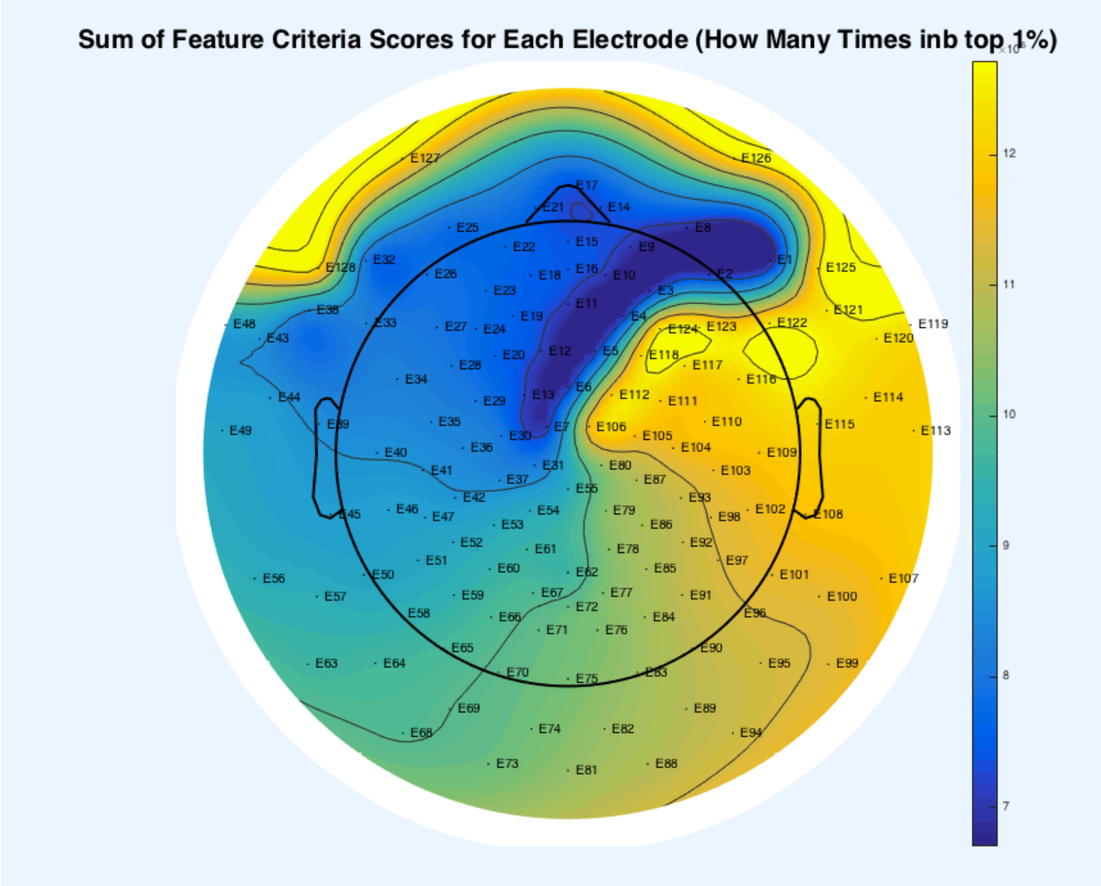


Figure 7: EEG dataset topfeatures in top 1%.

The forward selected features are then run on for 100 iterations(Face dataset) and 20 Iterations(EEG dataset) and then we plot visuals to see how many came in those iterations. These figures state exactly the same.

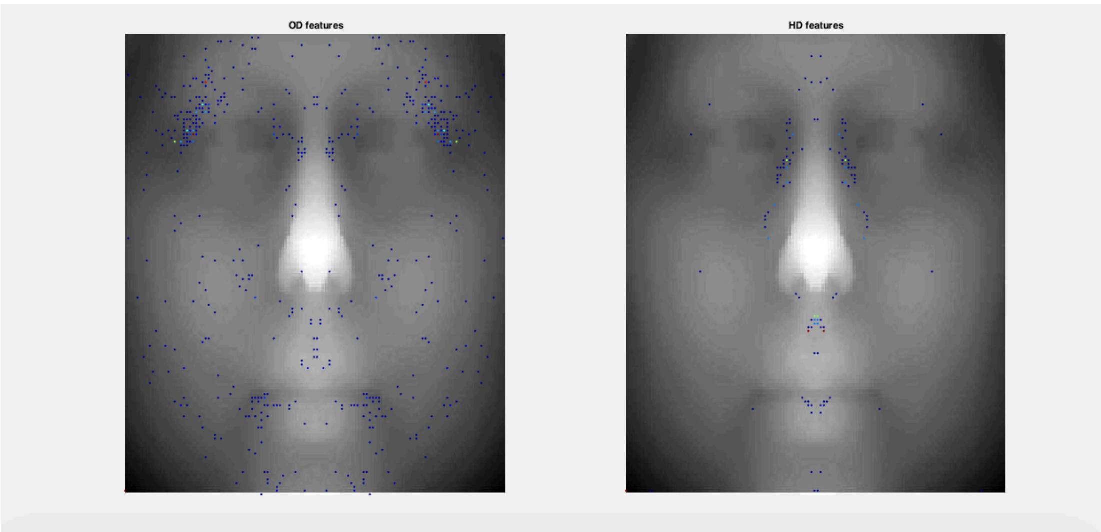


Figure 8: How many times by forward selection ? (Face).

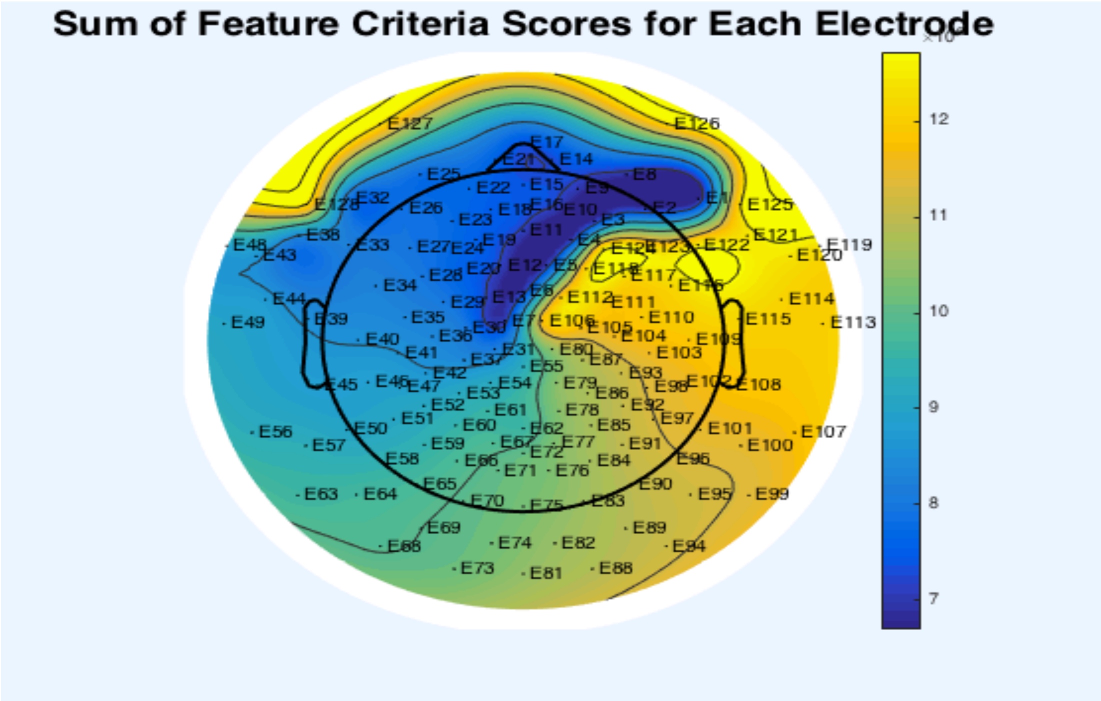


Figure 9: How many times by forward selection ? (EEG Dataset).

Part 3: Actual Classification using KNN.

After getting the forward selected features, I make new training and testing vectors according to the specific features and use a KNN classifier to classify these. The results are stated as below.

I did cross validation on the Face dataset and these were the results obtained for the respective iterations. As you can clearly see that the results change for every iteration . I calculated the mean of my error rate. Training and Testing data using the final KNN classifier on the selected features data set.

Iterations	Error_Train Mean	Error_Test Mean
100	6.96	33.94
80	6.72	33.25
60	7.51	33.75
40	6.50	33.33
20	5.2	36.32

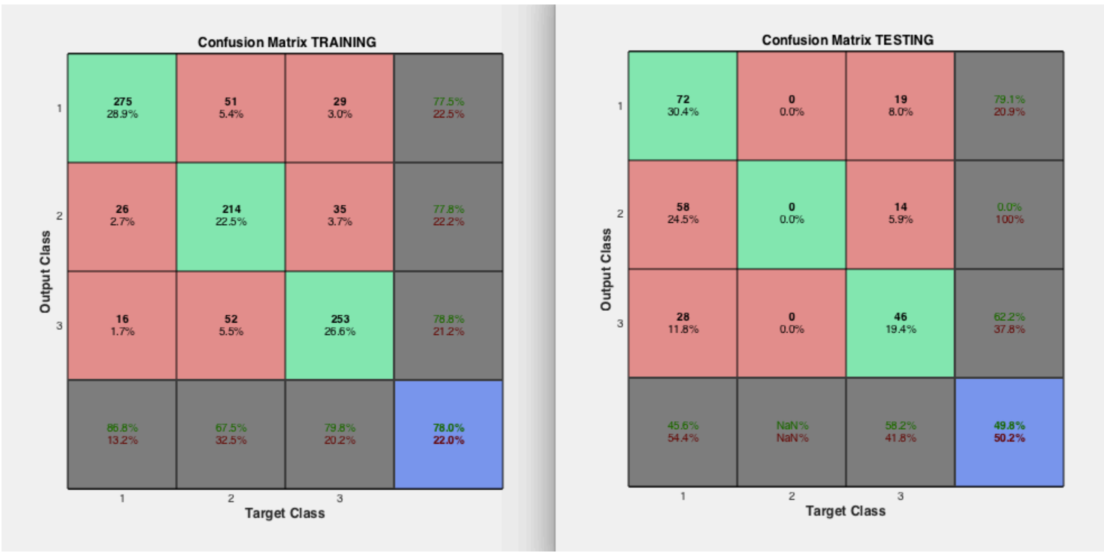
Thus as we can see the mean remains almost the same. With a slight varation in them, if we look at the error matrices obtained in matlab we can see the real change in the error rates. It can go to as low as 19% but also as high as 49% in some cases.

For the EEG data I calculated for three iteration variations : 1, 2 ,20,100.

Iteration	Error_Train Mean	Error_Test Mean
1	21.23	48.25
2	19.22	47.23
20	21.87	49.12

100	20.11	51.12

Classification Results: Confusion Matrix



EEG DataSet Classification

Face Dataset Classification.

Conclusion: I have also added a small part to my code so as to calculate the Error Rate means for Classification using all the features vs Classification using all the forward selected features, thus comparing what we learnt in the forward selection project. Thus as we the two values are quite same for the testing data(varying for Training data) we can say that by only selecting a chosen few out of the thousands of features we still achieve reasonably well classification for the same data set.