Project 3: Feature Selection

Due on Friday, March 6, 2020 at 11.59PM

Name: Ke LIANG ID: 926791183 email: kul660@psu.edu

contents

1 Introduction

The goal of this project is to get you familiar with two common methods to get the feature selection. The first is the filter feature selection and the second is wrapper feature selection.

The dataset we use is Face Dataset and EEG Dataset.

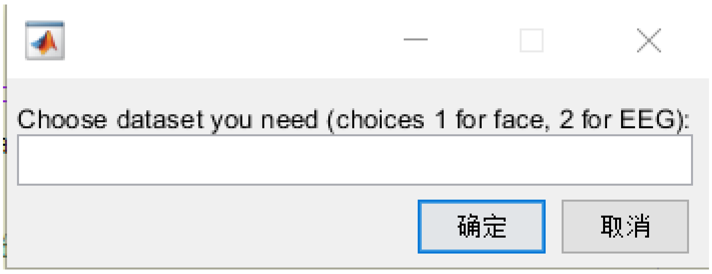
2 Methods

2.1 Dataset

As for **Face** dataset, there are 2 classes for the whole datasets and the number of the person is 125 where combining 15500 features.

As for **EEG** dataset, there are Fourier Coefficient Features and Temporal Features. And 2 sets of features are counted together to have 49920 features. While the raw data contains NAN's and all 0 entries for features and observations. In my implement I apply a search on the datasets to clean up all of these useless data.[1]

In my code when you print in you can follow the step to run the code as follow, where 1 is for Face dataset, and 2 is for EEG dataset.



2.2 Filter Feature Selection with Criterion

Filter Feature selection method is to select best features according to a criterion.[2]

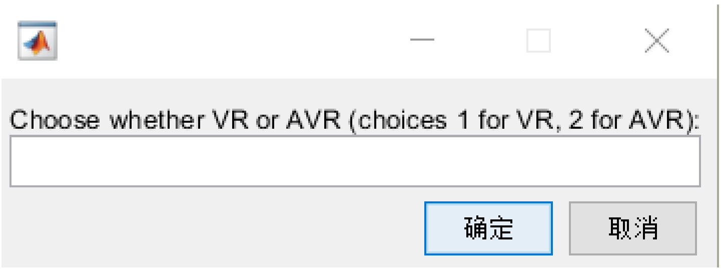
For me I used both Variation Ratio and Augmented Variance Ratio as the criterion on this part.

We name the meaning of F is feature F, and S_F is its value, C is the number of the total classes, and $Var_k(S_F)$ means the variance of the subset of values from feature F which belongs to class k. The formula for VR and AVR are shown below.[3]

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

$$\mathrm{AVR}(F) = \frac{\mathrm{Var}(S_F)}{(1/C) \sum_{i=1,\ldots,C} (\mathrm{Var}_i(S_F) / \min_{i \neq j}(|\mathrm{mean}_i(S_F) - \mathrm{mean}_j(S_F)|)}) \tag{2}$$

When you are running the code, follow the instructions it mentions you can choose which method you want to finish the filter part, and the instructions in the program is shown below. 1 is for VR and 2 is for AVR.



2.3 Wrapper based on Sequential Forward Selection

For this Wrapper Methods part, we combine it with forward selection. And the total idea of this method is that “for each subset of features, we solve the discrimination problem with quantitative results and select the best feature subset based on the classification accuracy” [2].

For me I choose the Linear discriminant analysis LDA classifiers and use the in-build function in Matlab code and use the Sensitivity Optimization Criteria as my evaluation criterion for wrapper and for search part we use sequential forward selection (SFS).

We first loop for all the feature we can choose in “topfeatures” which can also satisfy the stopping condition in the slides that “When no more feature left to be selected”, and for each subset of features, we solving the discrimination problem with the function “fitcdiscr” and “predict” then we can get the MdlLinear and train_pred, and we use classperf function in matlab to get the evaluation results for this features, and then compare it to last time loop evaluation value which is "temprate" in my code, and there are 3 situations. When this time is smaller then the value of last time, we do not do any updates and directly break out of the loop; when it is equal to the value of last time, we use an array "mark" to mark it, and if there are continuous 3 loops (which is used to speed up the program but makes the accuracy not that great)for the same value, we directly break out of the loop without update; if it is bigger than last time value, we keep updates and go to next loop.

2.4 Classification

We directly use the classification in project 2 to help us finish this part which is based on the in-build matlab function "fitcdiscr" and "predict".

In total, we can get the flowchart of all of my algorithm combined with filter selection method and wrapper method as shown below.

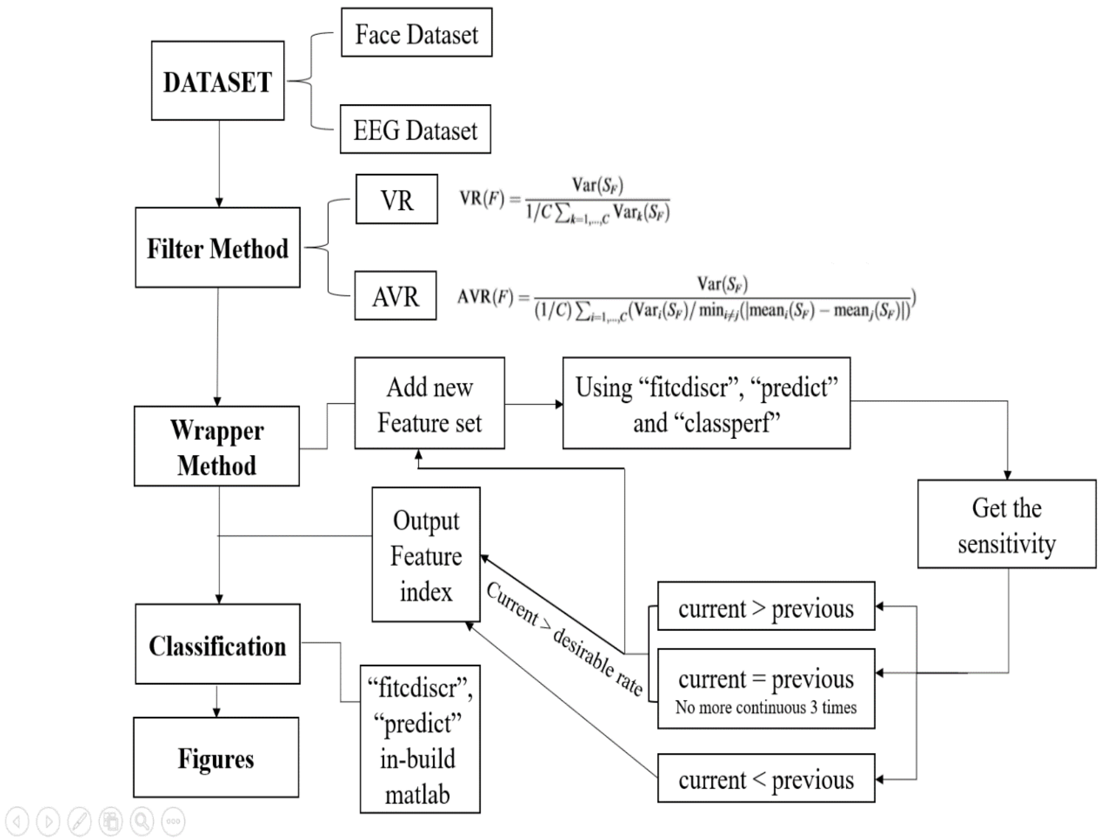
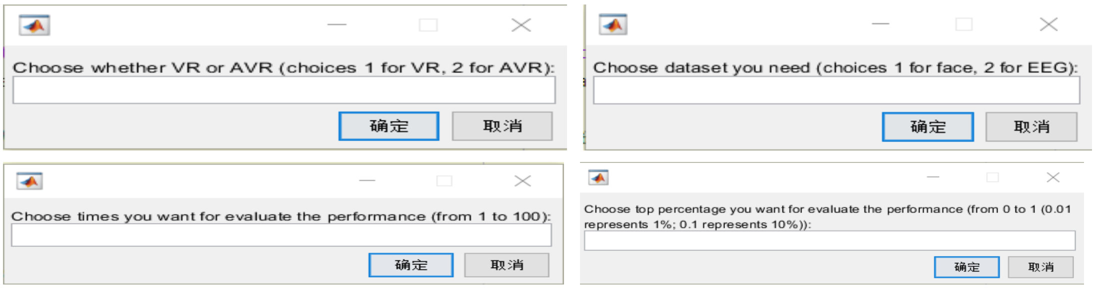


Figure 1 Algorithm Flowchart

3 Results

When we run the code start.m, there will appear some instructions as follow.



3.1 Results on Face dataset

We choose 1 for “Choose dataset you need” in this section.

We try method VR and AVR both on Face Dataset.

3.1.1 Results for VR method in filter part

We choose 1 for “choose whether VR or AVR”, 100 for “Choose times you want for evaluate the performance”, and 0.01 for “Choose top percentage you want for evaluate the performance” which means 1%. We can get the Figure 2, Figure 3, Figure 4 and Figure 9 below, and 4 other figures for classification matrix and confusion matrix.

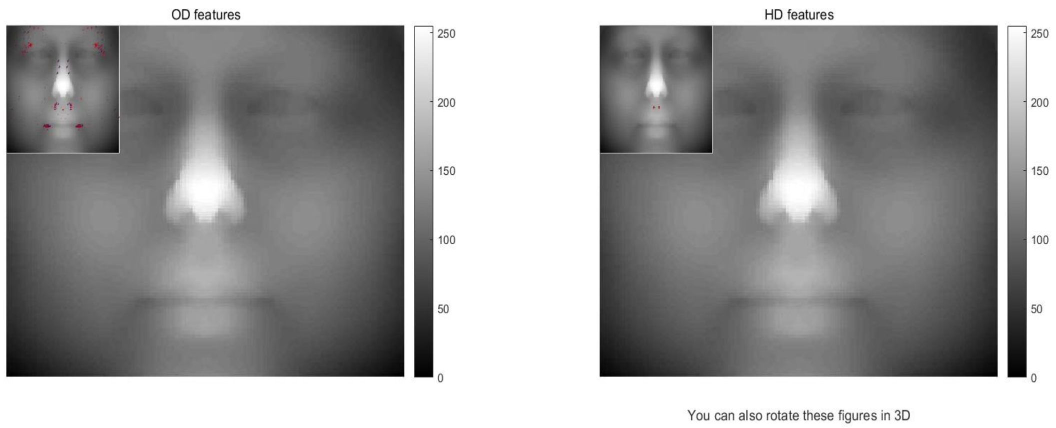


Figure 2 FACE of Method VR visualize the variance ratio of the top 1% features

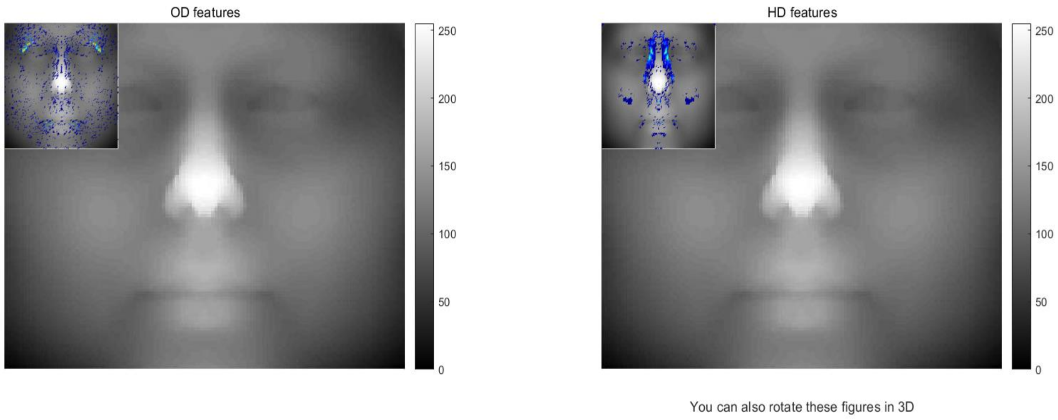


Figure 3 FACE of Method VR visualize the features that have ranked within top 1% most during 100 times of feature ranking

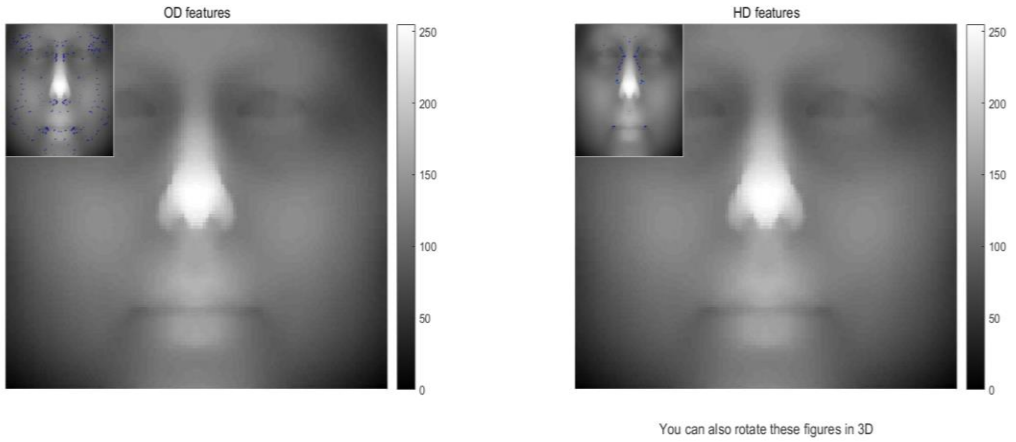


Figure 4 FACE of Method VR visualize the features that have been selected most during 100 times of forward selection

trainConfMat_ave =

17.9100	0.0900
3.6600	31.3400

Figure 5 FACE_VR Confusion Matrix for Training

trainClassMat_ave =

0.9950	0.0050
0.1046	0.8954

Figure 6 FACE_VR Classification Matrix for Training

testConfMat_ave =

6.2800	10.7200
3.7600	30.2400

Figure 7 FACE_VR Confusion Matrix for Testing

testClassMat_ave =

0.3694	0.6306
0.1106	0.8894

Figure 8 FACE_VR Classification Matrix for Testing

And besides, we can also get the accuracy of the training part and the testing part which are 94.52% and 62.94%. The accuracy of the training part is bigger than the testing part. And the standard deviation is 7.04% for testing part and it is same as training part which is 7.04%.

The visualization of the histogram of the top 10% features of FACE dataset with VR is shown below.

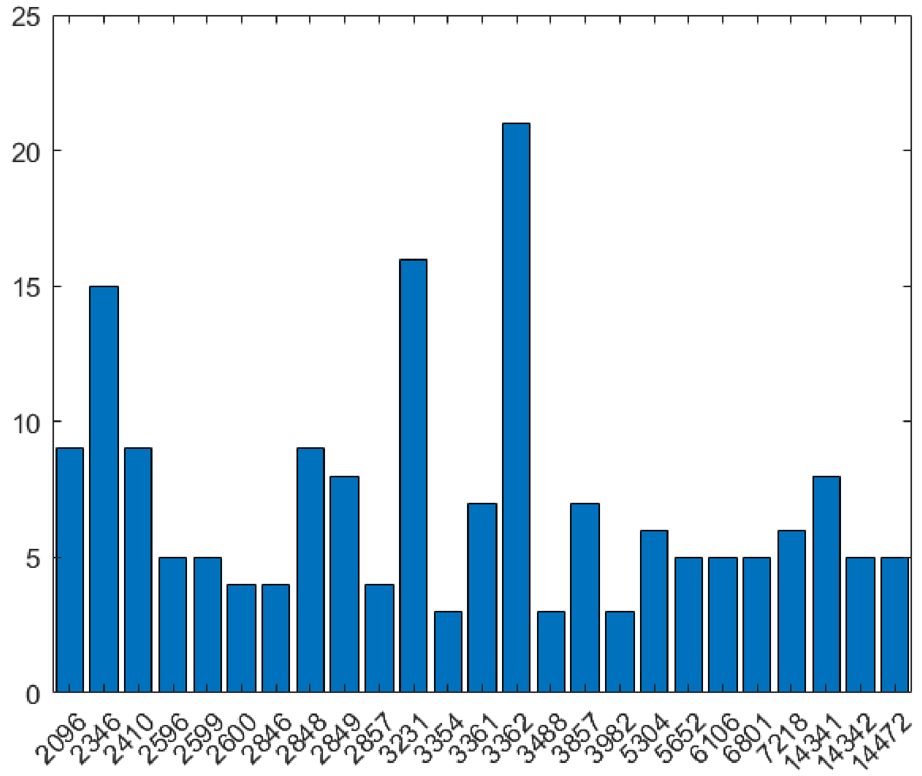


Figure 9 FACE of Method VR visualization of the histogram of the top 10% features

The bottom of the figure is the index of the features and the y coordinate is the number of the occurrence of the features. We can find that the feature 3361 shows up most frequently in top 10% features.

3.1.2 Results for AVR method in filter part

We choose 2 for “choose whether VR or AVR”, 100 for “Choose times you want for evaluate the performance”, and 0.01 for “Choose top percentage you want for evaluate the performance” which means 1%. We can get the Figure 9, Figure 10, Figure 11 and

Figure 16 below, and 4 other figures for classification matrix and confusion matrix.

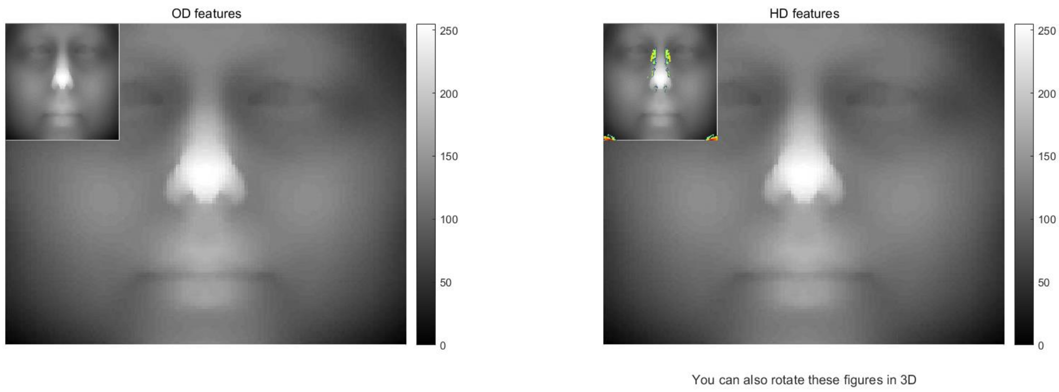


Figure 9 FACE of Method AVR visualize the variance ratio of the top 1% features

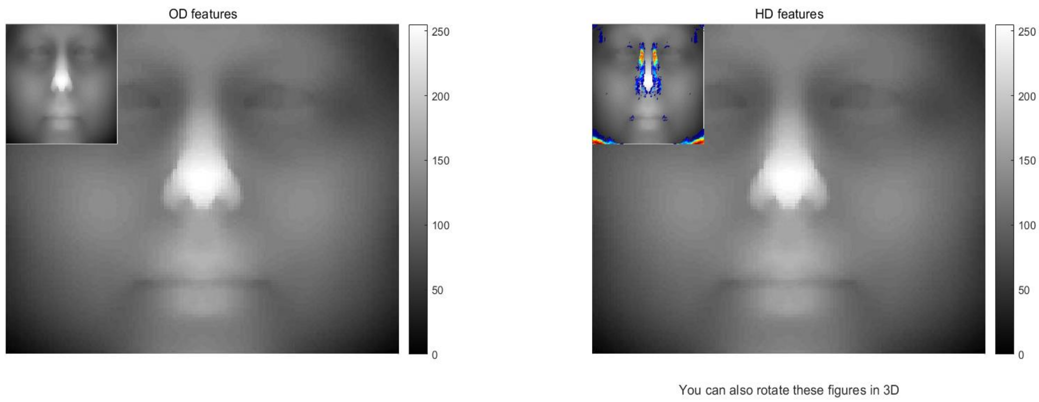


Figure 10 FACE of Method AVR visualize the features that have ranked within top 1% most during 100 times of feature ranking

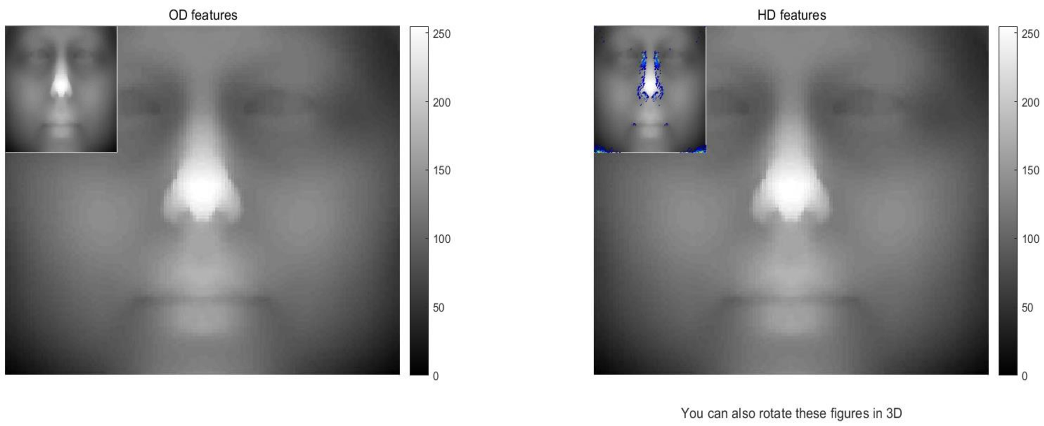


Figure 11 FACE of Method AVR visualize the features that have been selected most during 100 times of forward selection

trainConfMat_ave =

16.3300	1.6700
4.0500	30.9500

Figure 12 FACEAVR Confusion Matrix for Training

trainClassMat_ave =

0.9072	0.0928
0.1157	0.8843

Figure 13 FACEAVR Classification Matrix for Training

testConfMat_ave =

8.9400	8.0600
3.1900	30.8100

Figure 14 FACEAVR Confusion Matrix for Testing

testClassMat_ave =

0.5259	0.4741
0.0938	0.9062

Figure 15 FACEAVR Classification Matrix for Testing

And besides, we can also get the accuracy of the training part and the testing part which are 89.58% and 71.6%. The accuracy of the training part is bigger than the testing part. And the standard deviation is 4.61% for testing part and it is same as training part which is 4.61%.

The visualization of the histogram of the top 10% features of FACE dataset with VR is shown below.

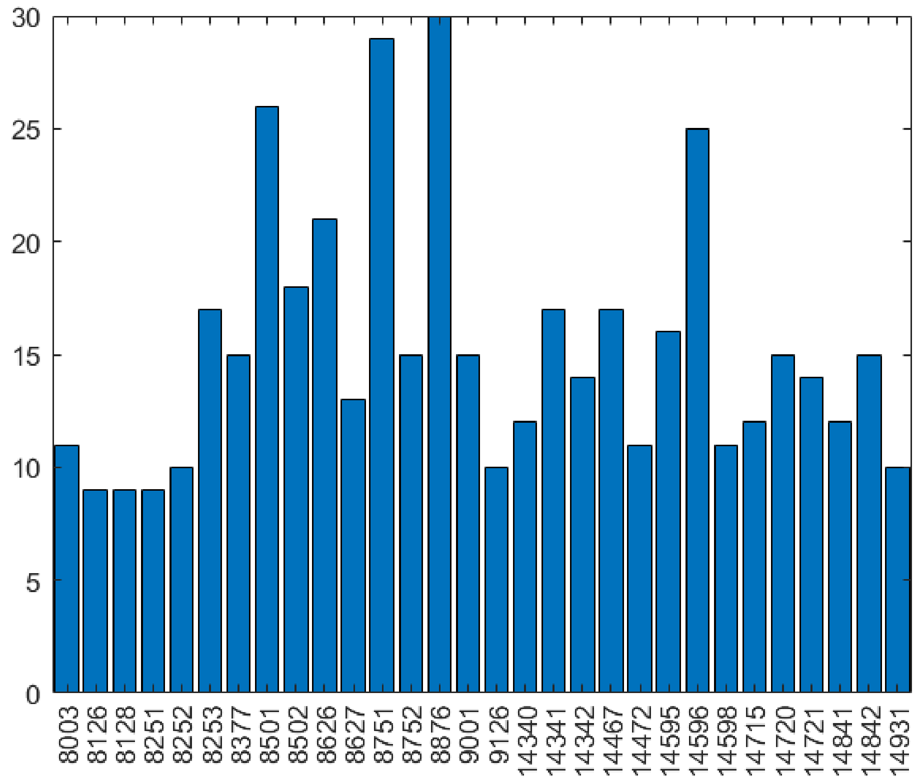


Figure 16 FACE of Method VR visualization of the histogram of the top 10% features

The bottom of the figure is the index of the features and the y coordinate is the number of the occurrence of the features. We can find that the feature 8876 shows up most frequently in top 10% features.

Comparing the results above, we can notice that it is more accurate when we using AVR rather than using VR for filter methods.

3.2 Results on EEG dataset

We choose 2 for "Choose dataset you need" in this section.
We try method VR and AVR both on Face Dataset.

3.1.1 Results for VR method in filter part

We choose 1 for "choose whether VR or AVR", 100 for "Choose times you want for evaluate the performance", and 0.01 for "Choose top percentage you want for evaluate the performance" which means 1%. We can get the Figure 17, Figure18, Figure19, Figure 20, Figure 21, Figure 22 and Figure 27 below, and 4 other figures for classification matrix and confusion matrix.

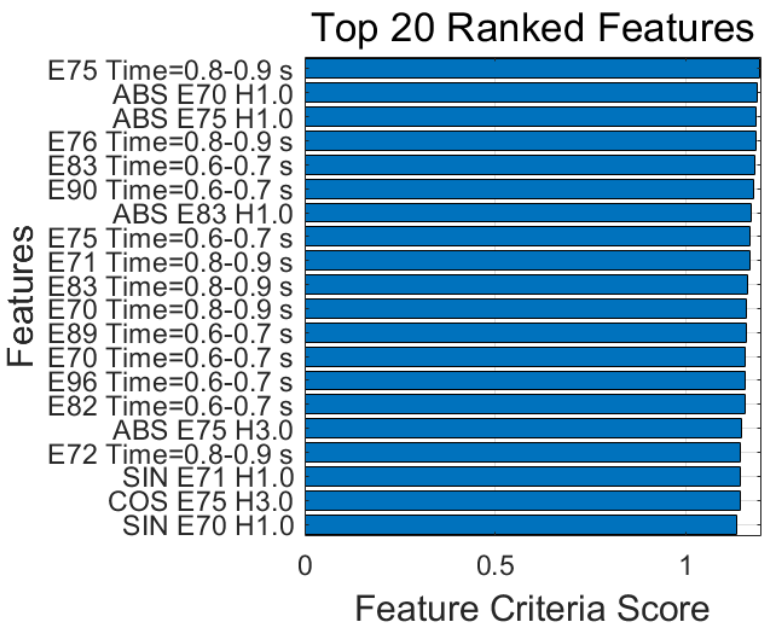


Figure 17 Histogram EEG of Method VR visualize the variance ratio of the top 1% features

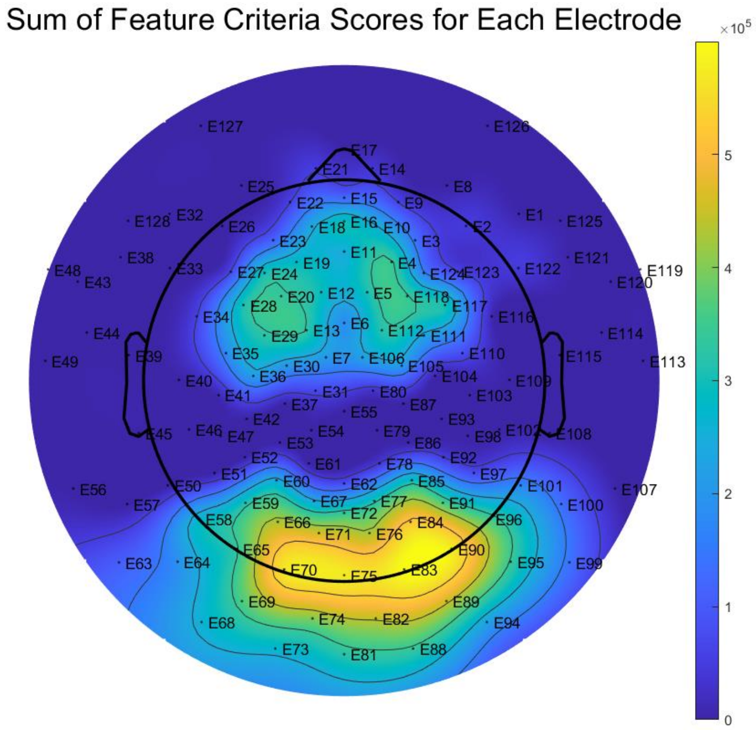


Figure 18 EEG of Method VR visualize the variance ratio of the top 1% features

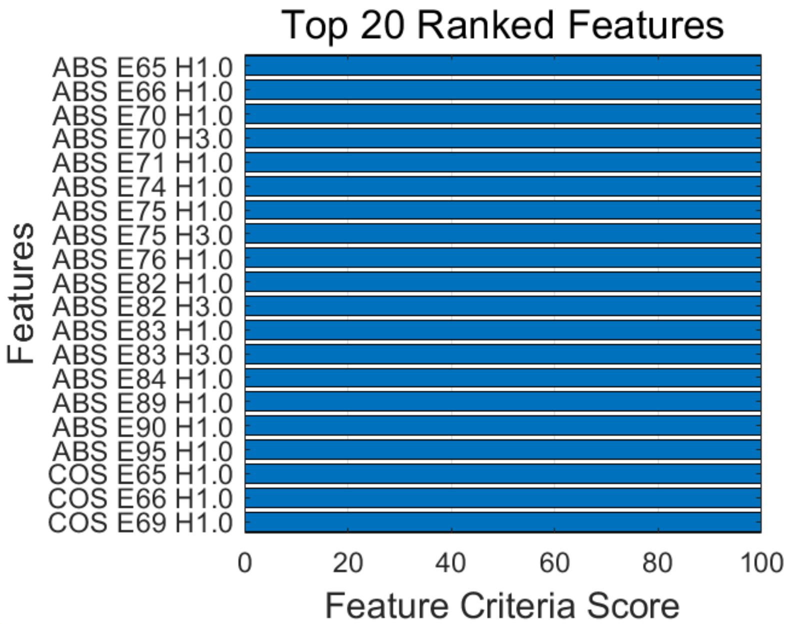


Figure 19 Histogram EEG of Method VR visualize the features that have ranked within top 1% most during 100 times of feature ranking

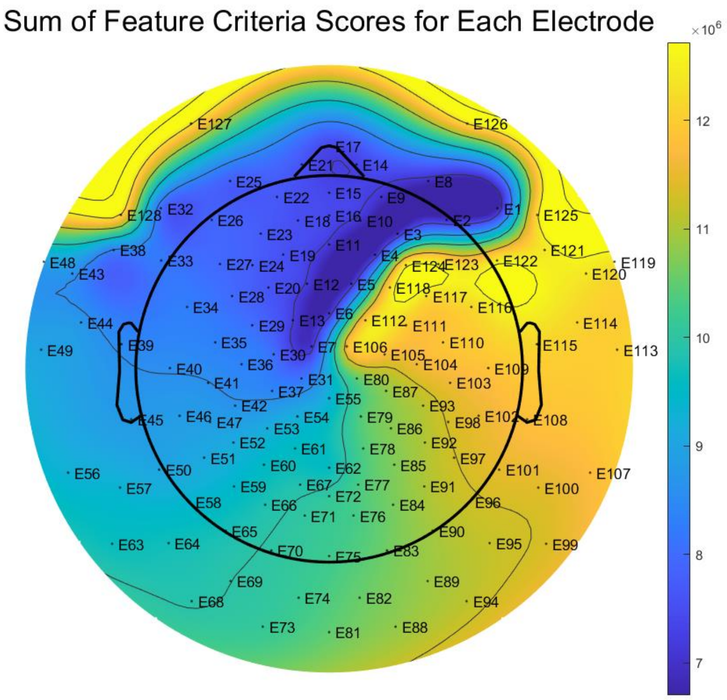


Figure 20 EEG of Method VR visualize the features that have ranked within top 1% most during 100 times of feature ranking

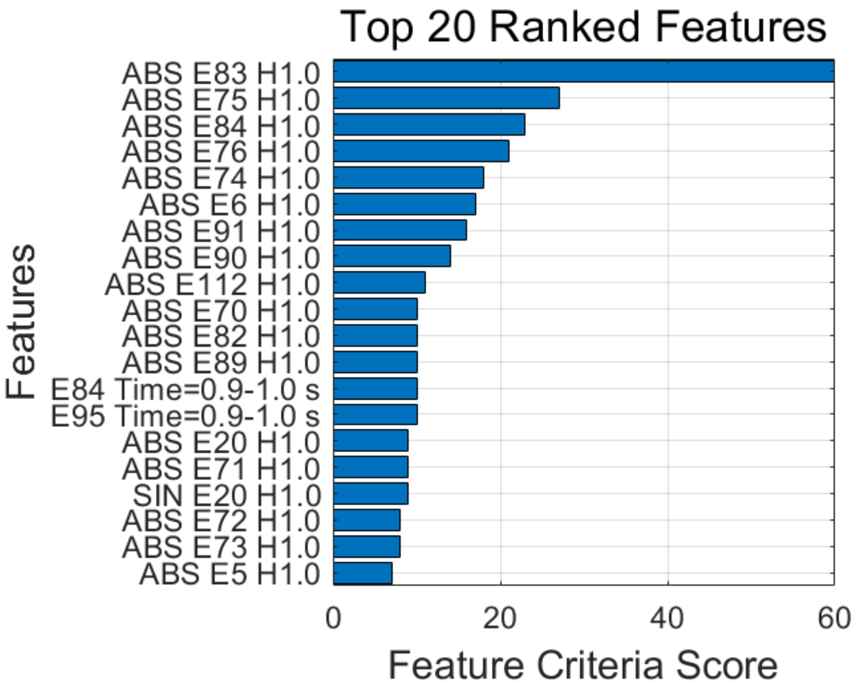


Figure 21 Histogram EEG of Method VR visualize the features that have been selected most during 100 times of forward selection

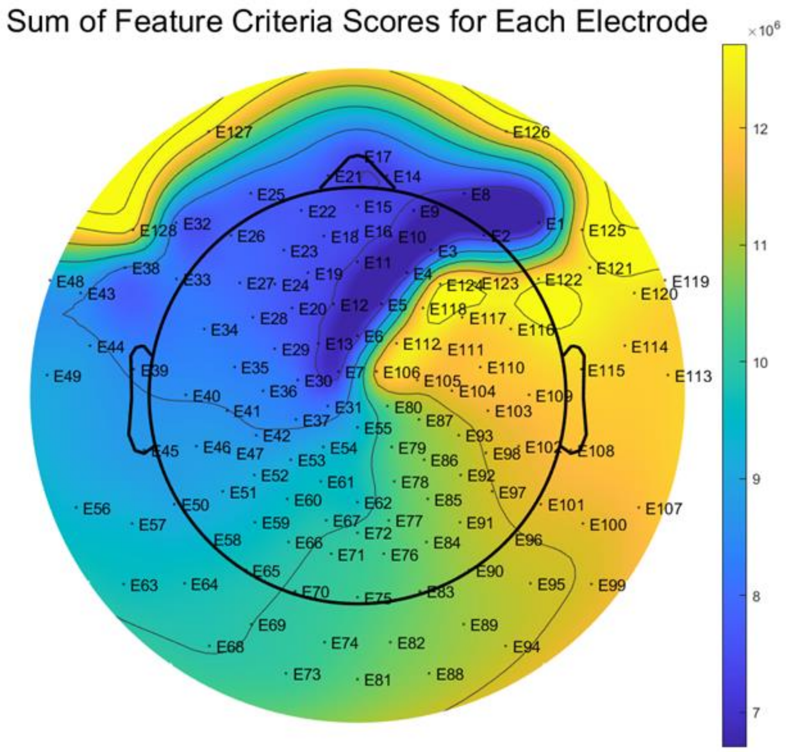


Figure 22 EEG of Method VR visualize the features that have been selected most during 100 times of forward selection

trainConfMat_ave =

161.9100	16.0200	17.0700
83.5400	51.5200	59.9400
44.7500	38.4600	110.7900

Figure 23 EEG_VR Confusion Matrix for Training

trainClassMat_ave =

0.8303	0.0822	0.0875
0.4284	0.2642	0.3074
0.2307	0.1982	0.5711

Figure 24 EEG_VR Classification Matrix for Training

testConfMat_ave =

137.6900	34.4700	21.8400
76.0200	61.2900	57.6900
43.1200	40.6500	110.2300

Figure 25 EEG_VR Confusion Matrix for Testing

testClassMat_ave =

0.7097	0.1777	0.1126
0.3898	0.3143	0.2958
0.2223	0.2095	0.5682

Figure 26 EEG_VR Classification Matrix for Testing

And besides, we can also get the accuracy of the training part and the testing part which are 55.52% and 53.07%. The accuracy of the training part is bigger than the testing part. And the standard deviation is 28.41% for testing part and it is same as training part which is 28.41%.

The visualization of the histogram of the top 10% features of EEG dataset with VR is shown below.

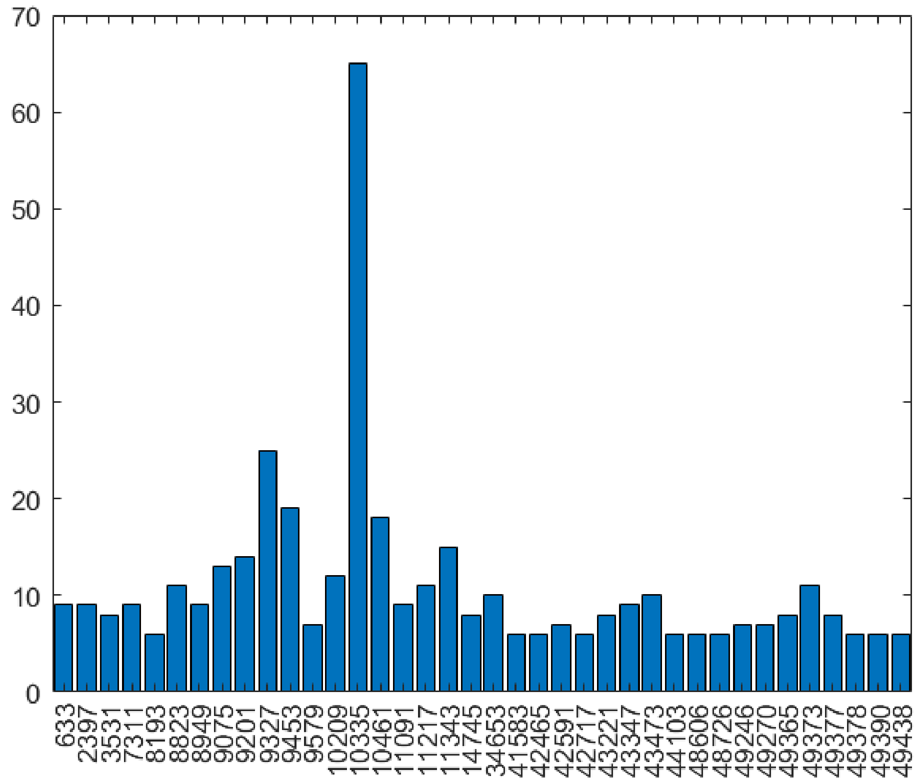


Figure 27 EEG of Method VR visualization of the histogram of the top 10% features

The bottom of the figure is the index of the features and the y coordinate is the number of the occurrence of the features. We can find that the feature 10335 shows up most frequently in top 10% features.

3.1.2 Results for AVR method in filter part

We choose 2 for “choose whether VR or AVR”, 100 for “Choose times you want for evaluate the performance”, and 0.01 for “Choose top percentage you want for evaluate the performance” which means 1%. We can get the Figure 28, Figure 29, Figure 30, Figure 31, Figure 32, Figure 33 and Figure 38 below, and 4 other figures for classification matrix and confusion matrix.

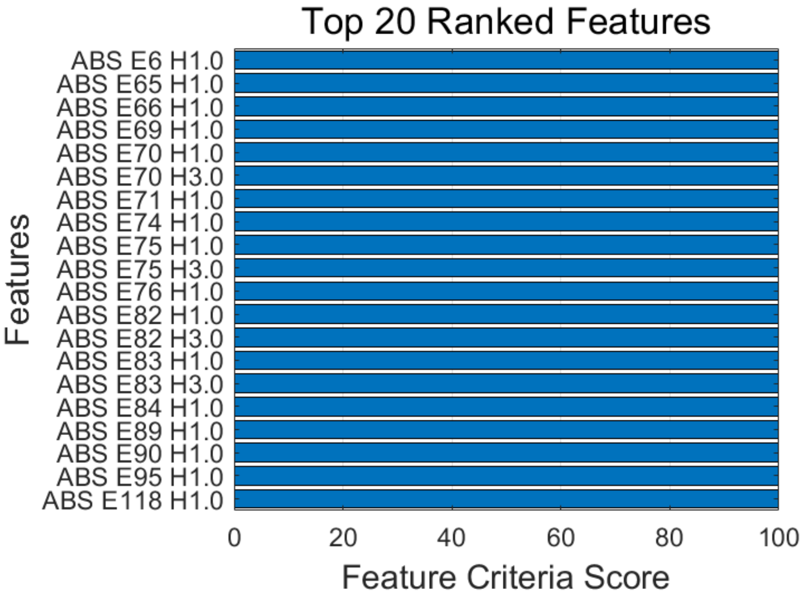


Figure 28 Histogram EEG of Method AVR visualize the variance ratio of the top 1% features

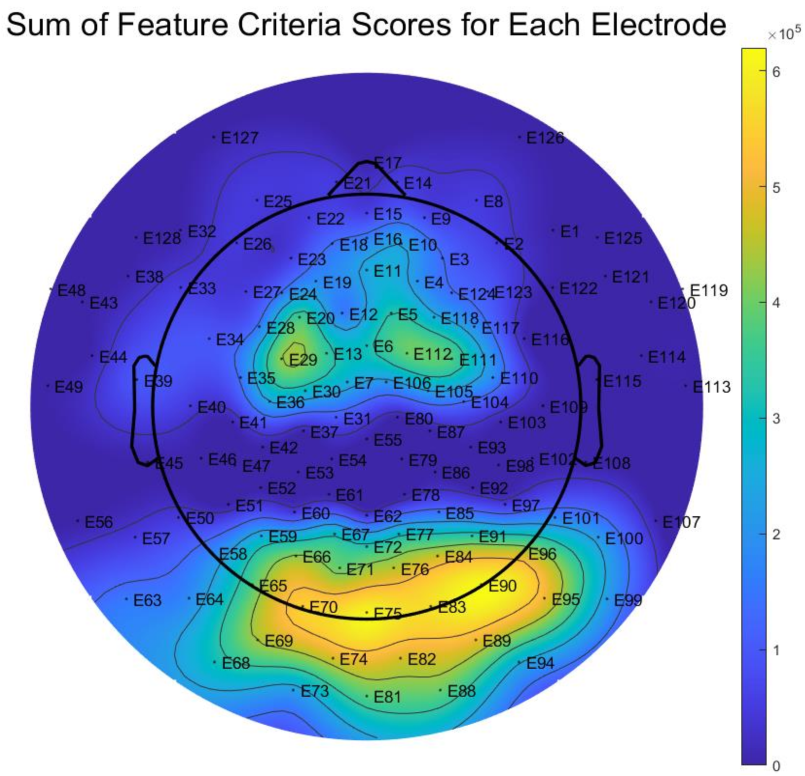


Figure 29 EEG of Method AVR visualize the variance ratio of the top 1% features

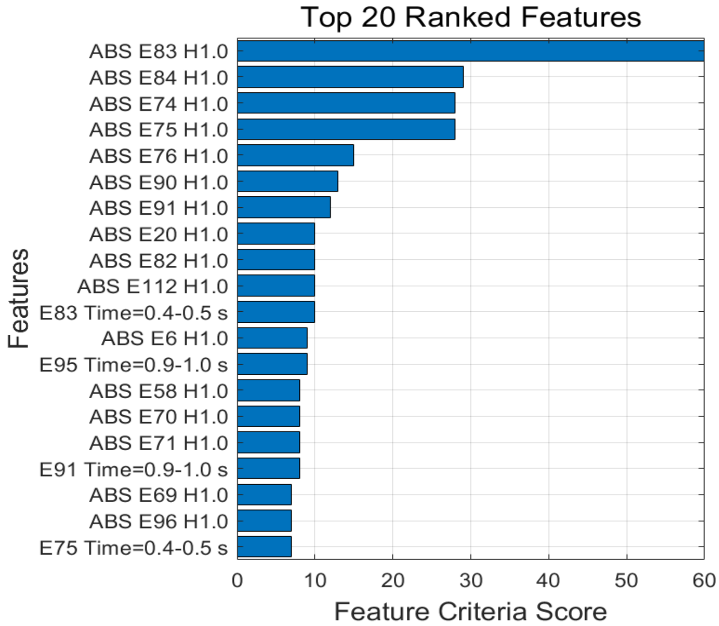


Figure 30 Histogram EEG of Method AVR visualize the features that have ranked within top 1% most during 100 times of feature ranking

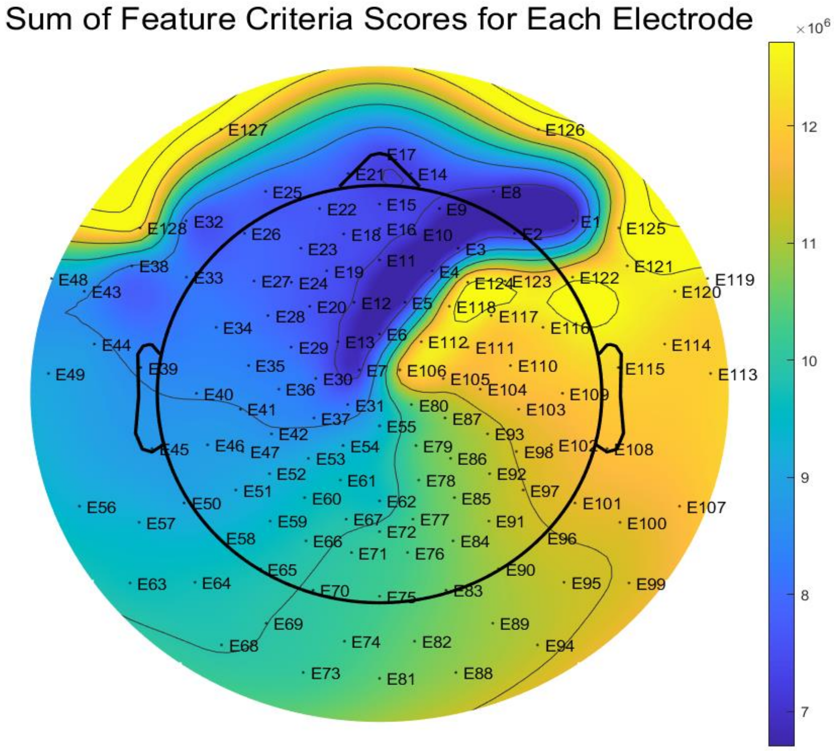


Figure 31 EEG of Method AVR visualize the features that have ranked within top 1% most during 100 times of feature ranking

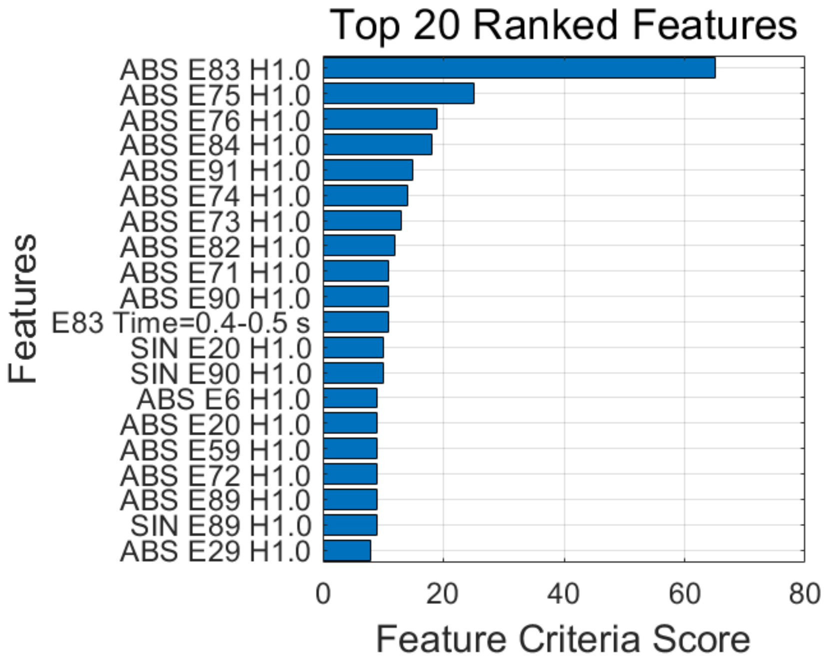


Figure 32 Histogram EEG of Method AVR visualize the features that have been selected most during 100 times of forward selection

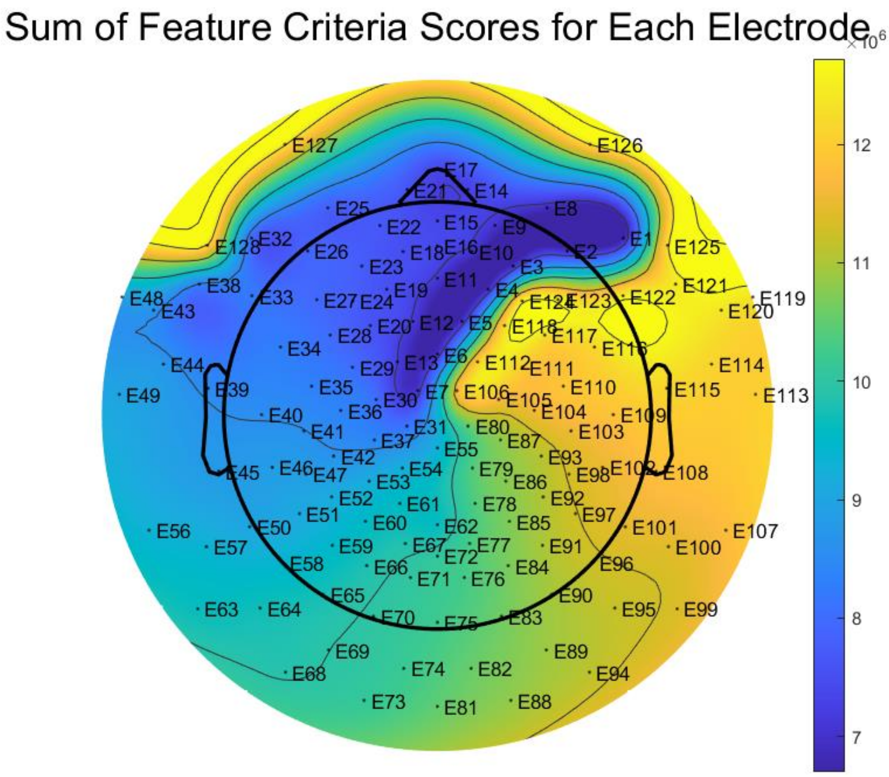


Figure 33 EEG of Method VR visualize the features that have been selected most during 100 times of forward selection

trainConfMat_ave =

161.0200	15.9000	18.0800
83.7400	51.8600	59.4000
43.6200	38.2200	112.1600

Figure 34 EEGAVR Confusion Matrix for Training

trainClassMat_ave =

0.8257	0.0815	0.0927
0.4294	0.2659	0.3046
0.2248	0.1970	0.5781

Figure 35 EEGAVR Classification Matrix for Training

testConfMat_ave =

135.2800	35.2200	23.5000
75.3400	61.2000	58.4600
43.7100	40.7500	109.5400

Figure36 EEGAVR Confusion Matrix for Testing

testClassMat_ave =

0.6973	0.1815	0.1211
0.3864	0.3138	0.2998
0.2253	0.2101	0.5646

Figure 37 EEGAVR Classification Matrix for Testing

And besides, we can also get the accuracy of the training part and the testing part which are 55.66% and 52.53%. The accuracy of the training part is bigger than the testing part. And the standard deviation is 28.13% for testing part and it is same as training part which is 28.13%.

The visualization of the histogram of the top 10% features of EEG dataset with VR is shown below.

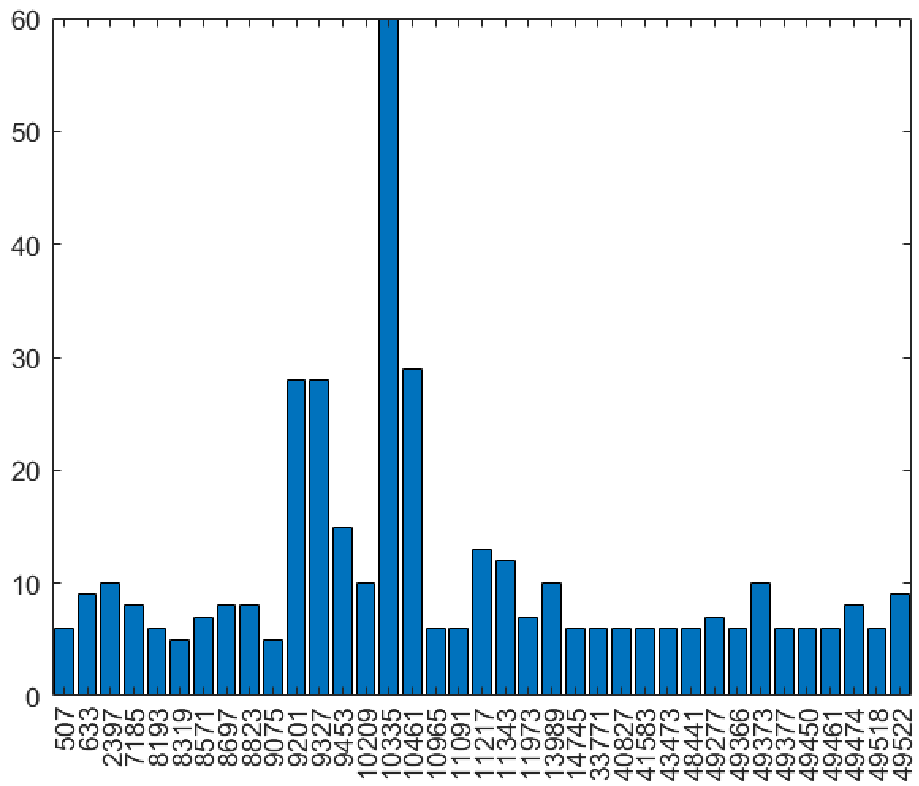


Figure 28 EEG of Method AVR visualization of the histogram of the top 10% features

The bottom of the figure is the index of the features and the y coordinate is the number of the occurrence of the features. We can find that the feature 10335 shows up most frequently in top 10% features.

4 Conclusions

We try to implement the feature selection with filter method and wrapper method, for filter method we use both VR and AVR to implement, and for Wrapper method we use SFS and the in-build matlab classification to test.

It seems that the algorithm test greater when using AVR than VR and it is great on FACE dataset than EEG dataset.

This project helps me learn more deep in feature selection, and hope this will be helpful to term-proj.

Reference

1 Description of the EEG dataset(professor posts on Canvas)

2 Slides for project 3 (professor posts on Canvas)

3 Yanxi Liu, Karen L. Schmidt, et al. Facial asymmetry quantification for expression invariant human identification. Computer Vision and Image Understanding, 2003.

4 Christopher, M. Bishop. PATTERN RECOGNITION AND MACHINE LEARNING. Springer-Verlag New York, 2016.