

## Contents

- Inter session Based Classification Binary Case
- Input
- Initiate some Structures
- Two main loops
- Load the Training Data
- Data Selection
- Calculate Covariance Matrix
- Train Classifier and Test

## Inter session Based Classification Binary Case

a simple pipeline to demonstrate how the data could be handled

```
clear all;
close all;
```

## Input

```
SBJ={ '01' };% Participants

TASK={ 'MATBdiff' 'MATBmed' 'MATBeasy' }; % Here all the names of the conditions

% The next two variables indicate what conditions you want to compare with
% each other. For example here the first test will be TASK{1} --> Difficult
% against TASK{3} --> Easy
TESTGR1={ (1) (1) (2)};

TESTGR2={ (3) (2) (3) }; % And the second condition to compare with

CLUSTER=[1,2,3,4,29,30,31,32]; % What electrodes you would like to keep
```

## Initiate some Structures

```
ALLDATA=struct;
accuracy=struct;
ACC=zeros(length(SBJ),length(TESTGR1));
```

## Two main loops

One for each Testgroup (Easy against Difficult)

```
for s=1:length(SBJ) % for all subjects

    disp('Hey I am now importing data for Subject '+string(SBJ{s}));

    for t=1:length(TESTGR1) % across all the test groups
```

## Load the Training Data

```
    disp('Hey I am now doing Machine Learning for Subject '+string(SBJ{s}));

    traindatal=pop_loadset(char('/Volumes/TOSHIBA EXT/comeptition_done/P'+string(SBJ{s})+'/S1/eeg/alldata_sb'+string(SBJ{s})+'_sess1_'+string(TASK{TESTGR1{t}})+'.set'));
    traindata2=pop_loadset(char('/Volumes/TOSHIBA EXT/comeptition_done/P'+string(SBJ{s})+'/S1/eeg/alldata_sb'+string(SBJ{s})+'_sess1_'+string(TASK{TESTGR2{t}})+'.set'));

    testdata1=pop_loadset(char('/Volumes/TOSHIBA EXT/comeptition_done/P'+string(SBJ{s})+'/S2/eeg/alldata_sb'+string(SBJ{s})+'_sess2_'+string(TASK{TESTGR1{t}})+'.set'));
    testdata2=pop_loadset(char('/Volumes/TOSHIBA EXT/comeptition_done/P'+string(SBJ{s})+'/S2/eeg/alldata_sb'+string(SBJ{s})+'_sess2_'+string(TASK{TESTGR2{t}})+'.set'));
```

```
Hey I am now doing Machine Learning for Subject 01
pop_loadset(): loading file /Volumes/TOSHIBA EXT/comeptition_done/P01/S1/eeg/alldata_sb01_sess1_MATBdiff.set ...
pop_loadset(): loading file /Volumes/TOSHIBA EXT/comeptition_done/P01/S1/eeg/alldata_sb01_sess1_MATBeasy.set ...
pop_loadset(): loading file /Volumes/TOSHIBA EXT/comeptition_done/P01/S2/eeg/alldata_sb01_sess2_MATBdiff.set ...
pop_loadset(): loading file /Volumes/TOSHIBA EXT/comeptition_done/P01/S2/eeg/alldata_sb01_sess2_MATBeasy.set ...
```

```
Hey I am now doing Machine Learning for Subject 01
pop_loadset(): loading file /Volumes/TOSHIBA EXT/comeptition_done/P01/S1/eeg/alldata_sb01_sess1_MATBdiff.set ...
pop_loadset(): loading file /Volumes/TOSHIBA EXT/comeptition_done/P01/S1/eeg/alldata_sb01_sess1_MATBmed.set ...
pop_loadset(): loading file /Volumes/TOSHIBA EXT/comeptition_done/P01/S2/eeg/alldata_sb01_sess2_MATBdiff.set ...
pop_loadset(): loading file /Volumes/TOSHIBA EXT/comeptition_done/P01/S2/eeg/alldata_sb01_sess2_MATBmed.set ...
```

```
Hey I am now doing Machine Learning for Subject 01
pop_loadset(): loading file /Volumes/TOSHIBA EXT/comeptition_done/P01/S1/eeg/alldata_sb01_sess1_MATBmed.set ...
pop_loadset(): loading file /Volumes/TOSHIBA EXT/comeptition_done/P01/S1/eeg/alldata_sb01_sess1_MATBeasy.set ...
pop_loadset(): loading file /Volumes/TOSHIBA EXT/comeptition_done/P01/S2/eeg/alldata_sb01_sess2_MATBmed.set ...
pop_loadset(): loading file /Volumes/TOSHIBA EXT/comeptition_done/P01/S2/eeg/alldata_sb01_sess2_MATBeasy.set ...
```

## Data Selection

```
%extract only data
class1=traindatal.data;
class2=traindata2.data;

testclass1=testdata1.data;
testclass2=testdata2.data;

%concatenate
allclass=cat(3,class1,class2);
testallclass=cat(3,testclass1,testclass2);

%reduce electrodes
allclass=allclass(CLUSTER,:,:);

% same for testset
testallclass=testallclass(CLUSTER,:,:);

N=size(allclass,3);
% same for testset
testN=size(testallclass,3);

%make binary vector for labels
classbin=zeros(N,1);
classbin(1:size(class1,3))=1;
classbin((size(class1,3)+1):N)=0;
% same for testset
testbin=zeros(N,1);
testbin(1:size(class1,3))=1;
testbin((size(class1,3)+1):N)=0;

% Randomize order of training and testing set
trainrand=randperm(N);
trainingbin=classbin(trainrand);
trainingset=allclass(:, :, trainrand);

% same for testset
testrand=randperm(testN);
testbin=testbin(testrand);
testset=testallclass(:, :, testrand);
```

## Calculate Covariance Matrix

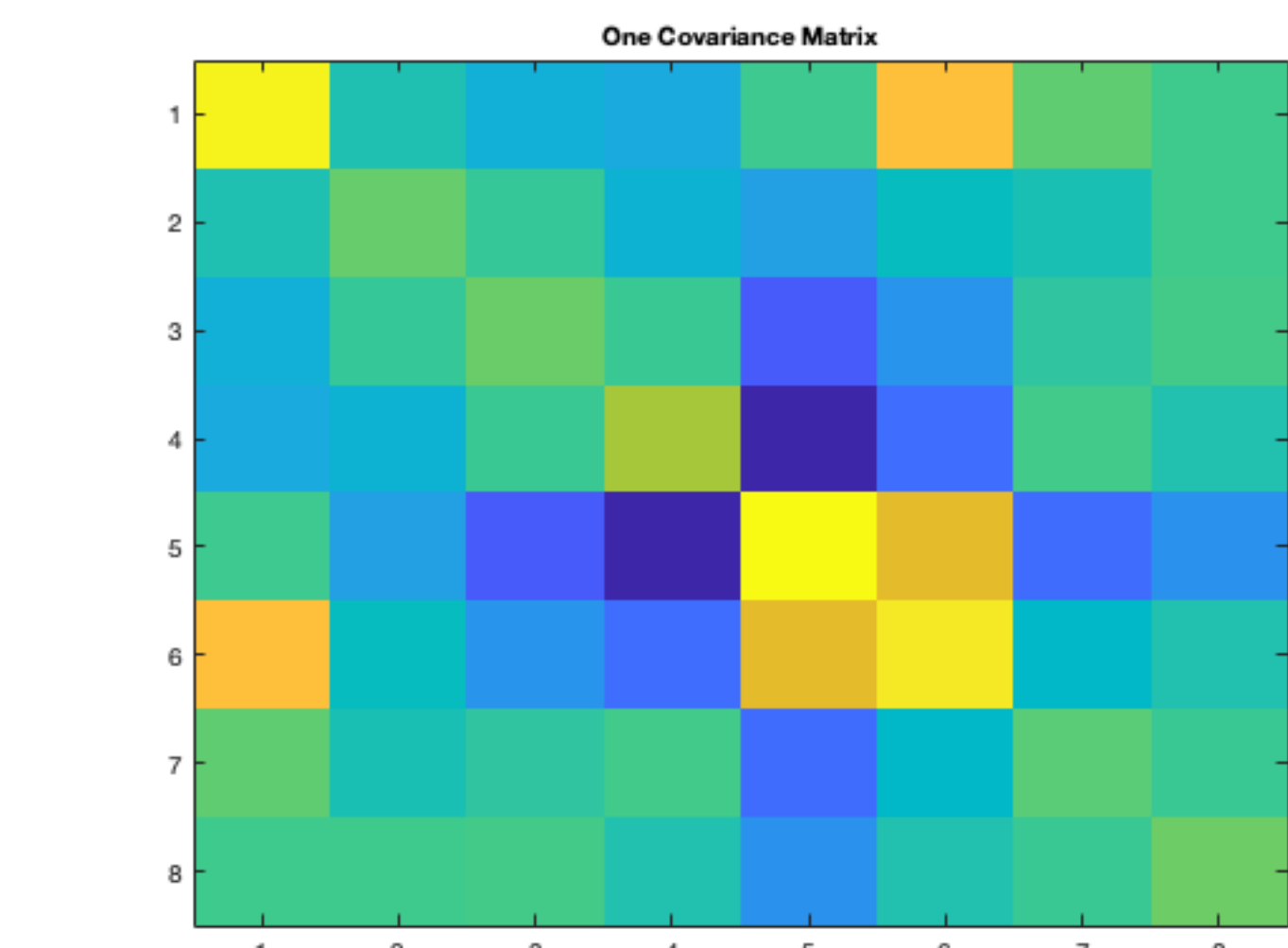
```
for ep=1:size(trainingset,3)

    trainingcov(:, :, ep)=cov(trainingset(:, :, ep));
end

for ep=1:size(testset,3)

    testcov(:, :, ep)=cov(testset(:, :, ep));
end

if t=1
    imagesc(testcov(:, :, 1));
    title('One Covariance Matrix', 'FontSize', 10);
end
```



## Train Classifier and Test

```
% Tidy Up
clear trainingset; trainingset=trainingcov; clear trainingcov;
clear testset; testset=testcov; clear testcov;

% Classifier training and prediction
Ytest =mdm(testset,trainingset,trainingbin, 'riemann', 'riemann');

% Results
ACC(s,t) = 100*mean(Ytest==testbin);

% Tidy up
clear trainingset; clear trainingsetn; clear testsetn; clear trainingbin; clear testset; clear testbin; clear testres; clear allfourier;
clear EBGsignals;
```

```
end

end

for i=1:length(TESTGR1)
    disp('Accuracy of : '+string(ACC(i))+ ' for '+string(TASK{TESTGR1{i}})+' versus '+string(TASK{TESTGR2{i}}));
end

disp('How Much Better Can you do');
WaitSecs(1.5);

disp('Good Luck!');
```

```
Hey I am now importing data for Subject 01
Accuracy of : 46.9799 for MATBdiff versus MATBeasy
Accuracy of : 49.6644 for MATBdiff versus MATBmed
Accuracy of : 50.6711 for MATBmed versus MATBeasy
How Much Better Can you do
Good Luck!
```