Mainft.m

clear sample;

M = cell(32,1);

for i=1:32;

%figure;

file = ['/Users/gavinckoalagesan/Library/Mobile Documents/com~apple~CloudDocs/Year4/BME 772/LABS/Project/Alcoholics/SMNI\_CMI\_TRAIN/Data', num2str(i), '.csv']

%file = 'Data1.csv';

file = readtable(file);

structarray = table2struct(file);

sample = [(0:1:255) ; zeros(1,256)]';

sample(:,1) = sample(:,1)/256;

% diff\_var = [(0:1:255) ; zeros(1,256)]';

% mobility = [(0:1:255) ; zeros(1,256)]';

% diff\_mobility = [(0:1:255) ; zeros(1,256)]';

% formfactor = [(0:1:255) ; zeros(1,256)]';

% electrode = zeros(255,2);

%sample = [zeros(1,256)]';

freq = zeros(129,1);

% figure;

for j = 0:63

j

% [sample, freq(:,j+1)] = electrodeplot(j, structarray, sample);

% [sample, freq(:,j+1)] = electrodeplot(j, structarray, sample);

%

%

% electrode\_mean(i,j+1) = mean(sample(:,j+1));

% electrode\_var(i,j+1) = std(sample(:,j+1));

% electrode\_diff = diff(sample(:,j+1));

% diff\_var(i,j+2) = std(electrode\_diff);

% mobility(i,j+2) = diff\_var(i,j+2)/electrode\_var(i,j+1);

% diff\_mobility(i,j+2) = diff(mobility(i, j+2);

% formfactor (i,j+2) = diff\_mobility(i,j+2)/(mobility(i,j+2));

%

[sample, freq(:,j+1)] = electrodeplot(j, structarray, sample);

% electrode\_mean(i,j+1) = mean(sample(:,j+2))

% electrode\_std(i,j+1) = std(sample(:,j+2))

% da = [1, -1] %denom

% db = [1, -0.995] %numerator

% %filtered

% derivativeeeg = filter(da, db, sample(:,j+1));

% variance\_of\_der(i,j+1) = std(derivativeeeg);

% mobility(i,j+1) = variance\_of\_der(i,j+1)/electrode\_std(i,j+1);

% % derivative\_mobility(i,j+1) = filter(da,db, mobility(i,j+1));

% deriv\_dersig = filter(da, db, derivativeeeg);

% mobility\_dersig(i, j+1) = (std(deriv\_dersig))/variance\_of\_der(i, j+1);

% formfactor (i,j+1) = (mobility\_dersig(i,j+1))/(mobility(i,j+1))

% %

% approxEnt(i,j+1) = approximateEntropy(sample(:, j+1));

% energy(i, j+1) = 0;

% entropy(i, j+1) =0;

% for f=1:256

% energy(i, j+1) = energy(i, j+1)+ (sample(f, j+2)^2);

% entropy(i, j+1) = entropy(i, j+1) + ((sample(f, j+2).^2) \* log((sample(f, j+2).^2)));

% end

end

features = zeros(64,1);

[deltawave, features(1:64, 1:3)] = extractwave(1, 4, sample, "Delta", i);

[thetawave, features(1:64, 4:6)] = extractwave(4, 7, sample, "Theta", i);

[alphawave, features(1:64, 7:9)] = extractwave(8, 12, sample, "Alpha", i);

[betawave, features(1:64, 10:12)] = extractwave(12, 30, sample, "Beta", i);

[gammawave, features(1:64, 13:15)] = extractwave(30, 100, sample, "Gamma", i);

M30{i-20} = features;

end

%%

j=1;

figure;

for i=1:20

alcohalic = cell2mat(realdata(i));

control = cell2mat(realdata(i+20));

%figure

i

%for j=53:54

for n=2:3:15

scatter(alcohalic(54,n),alcohalic(54,n+1),'+g');hold on

scatter(control(54,n),control(54,n+1),'.r');

%X((i\*2)-1:(i\*2), 1:2) = [alcohalic(54,n) alcohalic(54,n+1)

X(j,1:2) = [alcohalic(54,n) alcohalic(54,n+1)];

Y(j, 1:2) = [control(54,n) control(54,n+1)];

j = j+1;

end

%end

% scatter(sample(:,1),alcohalic(:,3), '+g')

% scatter(saple(:,1),control(:,3), '.r');

xlabel('Form Factor')

ylabel('Entropy')

%legend('Alcohalics','Control')

title('Data for classification')

%hold off

hold on;

end

electrodeplot.m

function [sample,xsing] = electrodeplot(j,structarray, sample)

%UNTITLED3 Summary of this function goes here

% Detailed explanation goes here

%sample = [(0:1:255) ; zeros(1,256)]';

Fs = 256;

f = Fs\*(0:(256/2))/256;

sum = ((j + 1) \* 3) -2;

sum2 = ((j+1) \* 3) -1;

sum3 = ((j+1)\*3);

t = [0:255]/Fs;

% sum2 = sum+1;

for n=1:16384

if structarray(n).channel == j

for i=1:1:256

if structarray(n).sampleNum == i-1

sample(i, j+2) = structarray(n).sensorValue;

end

end

% subplot(4, 3, sum)

% % figure;

% plot(sample(:,1),sample(:,j+2));

%plot(t, sample(:,j+2):

axis tight

ylabel (j);

title ('A');

% figure;

% subplot(5,1,sum)

% plot(fft(sample(:,2)))

x = abs(fft(sample(:,j+2))/255); %255 the number of samples

xsing = x(1:(256/2+1)); % cutting the sammple by half

% subplot(4,3,sum2);

% plot(f,xsing);

y = abs(stft(sample(:,j+2))/255);

ysing = y(1:(256/2+1));

% subplot(4,3,sum3);

% % %plot(f,ysing);

% stft((sample(:,j+2))/255)

%

% x=1;

% plot(x);

% axis tight

% ylabel (j);

% title ('B')

end

% end

% sample = [(0:1:255) ; zeros(1,256)]'

end

end

extractwave.m

function [Wave, features] = extractwave(f1,f2, sample, titleg,i)

%UNTITLED2 Summary of this function goes here

% Detailed explanation goes here

%figure;

for j=0:63

bpFilt = designfilt('bandpassfir','FilterOrder',20, ...

'CutoffFrequency1',f1,'CutoffFrequency2',f2, ...

'SampleRate',256);

%fvtool(bpFilt);

[b,a] = tf(bpFilt);

%.,?freqz(b, a, 255, 256);title("Band Pass Filter");

Wave(:,j+1) = filter(b,a, sample(:, j+2));

% subplot(4,1,j+1);

% plot(sample(:,1), Wave(:,j+1));xlabel("Samples");title(titleg);

electrode\_mean(i,j+1) = mean(Wave(:,j+1))

electrode\_std(i,j+1) = std(Wave(:,j+1))

da = [1, -1] %denom

db = [1, -0.995] %numerator

%filtered

derivativeeeg = filter(da, db, Wave(:,j+1));

variance\_of\_der(i,j+1) = std(derivativeeeg);

mobility(i,j+1) = variance\_of\_der(i,j+1)/electrode\_std(i,j+1);

% derivative\_mobility(i,j+1) = filter(da,db, mobility(i,j+1));

deriv\_dersig = filter(da, db, derivativeeeg);

mobility\_dersig(i, j+1) = (std(deriv\_dersig))/variance\_of\_der(i, j+1);

formfactor (i,j+1) = (mobility\_dersig(i,j+1))/(mobility(i,j+1))

%

approxEnt(i,j+1) = approximateEntropy(Wave(:, j+1));

energy(i, j+1) = 0;

entropy(i, j+1) =0;

for f=1:256

energy(i, j+1) = energy(i, j+1)+ (sample(f, j+2)^2);

entropy(i, j+1) = entropy(i, j+1) + ((sample(f, j+2).^2) \* log((sample(f, j+2).^2)));

end

features(j+1, 1:3) = [electrode\_mean(i,j+1) formfactor(i,j+1) entropy(i, j+1)];

end

end