Neurophotonics HW1

1. Writing a function that calculates ΔHbR and ΔHbO over time

**Main function :**

clc;clear;close all;

% Loading the data

data1 = load('FN\_032\_V1\_Postdose1\_Nback.mat');

data2 = load('FN\_031\_V2\_Postdose2\_Nback.mat');

SDS = 3; %Sourse-Detector Separation distance [cm]

relDPF = "RelativeDPFCoefficients.csv";

DPFperTissue = "DPFperTissue.txt"; %relative DPF according to wavelength

extinctionCoefficients = "ExtinctionCoefficientsData.csv"; %.csv file with the following columns : wavelength, Water, HbO2, HHb, FatSoybean

tissueType = 'adult\_head'; %Options: 'adult\_forearm' \ 'baby\_head' \ 'adult\_head' \ 'adult\_leg'

plotChannelIdx = [1,2,3]; %vector indicating channels to plot.

[ dHbR , dHbO, fig ] = CalcNIRS(data1, SDS, tissueType, plotChannelIdx, extinctionCoefficients , DPFperTissue, relDPF );

**dHbR & dHbO Calc function**

function [ dHbR , dHbO, fig ] = CalcNIRS(dataFile, SDS, tissueType, plotChannelIdx, extinctionCoefficientsFile, DPFperTissueFile, relDPFfile )

%% Extract relevant data from the input dataFile

wavelengths =dataFile.SD.Lambda; %Two wavelengths [nm]

time = dataFile.t; %Time vector

intensities = dataFile.d;

%intensityLow = dataFile.d(:,1:20); %intensity levels at low WL

%intensityHigh = dataFile.d(:,21:end); %intensity levels at high WL

%% Set default values

if nargin < 7 || isempty(relDPFfile)

relDPFfile = '.\RelativeDPFCoefficients.csv';

end

if nargin < 6 || isempty(DPFperTissueFile)

DPFperTissueFile = '.\DPFperTissue.txt';

end

if nargin < 5 || isempty(extinctionCoefficientsFile)

extinctionCoefficientsFile = '.\ExtinctionCoefficientsData.csv';

end

if nargin < 4 || isempty(plotChannelIdx)

plotChannelIdx = [];

end

%% Load extinction coefficients and DPF data

extinctionCoefficients = readtable(extinctionCoefficientsFile);

DPFperTissue = readtable(DPFperTissueFile);

relDPF = readtable(relDPFfile);

%% Calculate DPF for the given tissue type and wavelengths

DPF\_807nm = DPFperTissue.DPF(strcmp(DPFperTissue.Tissue, tissueType));

relDPF\_factors = interp1(relDPF.wavelength, relDPF{:, 2:end}, wavelengths);

DPF = DPF\_807nm .\* relDPF\_factors;

%% Calculate optical densities

I0 = intensities(1, :);

OD = log10(I0 ./ intensities);

%% Calculate extinction coefficients for the given wavelengths

epsilon\_HbR = interp1(extinctionCoefficients.wavelength, extinctionCoefficients.HHb, wavelengths);

epsilon\_HbO = interp1(extinctionCoefficients.wavelength, extinctionCoefficients.HbO2, wavelengths);

%% Output

dHbR = OD(:,1:20) ./ (epsilon\_HbR(1) \* DPF(1) \* SDS);

dHbO = OD(:,21:end) ./ (epsilon\_HbO (2)\* DPF(2) \* SDS);

%% Plot the specified channels

figs = [];

if ~isempty(plotChannelIdx) && isvector(plotChannelIdx) && all(plotChannelIdx >= 1 & plotChannelIdx <= 20 & rem(plotChannelIdx, 1) == 0)

for ch = plotChannelIdx

fig = figure;

plot(time, dHbR(:, ch), 'b');

hold on;

plot(time, dHbO(:, ch), 'r');

hold off;

title(sprintf('Channel %d', ch));

xlabel('Time [s]');

ylabel('Concentration Change');

legend('dHbR', 'dHbO', 'Location', 'best');

figs = [figs, fig]; % Store the figure handle

end

elseif isempty(plotChannelIdx)

disp('No channels specified for plotting.');

else

disp('Invalid input for plotChannelIdx. Please provide a vector with values in the range [1-20].');

end

First 2 channels plots for “FN\_031\_V2\_Postdose2\_Nback.mat” :



First 2 channels plots for “ FN\_032\_V1\_Postdose1\_Nback.mat” :