ICM- PHYSIOME CODING PROGRESS ASSIGNMENT
SUBMITTED BY SEJAL GHATE, ZIXU HAN AND YONGZHI SUN

MAIN CODE UTILIZING ALL FUNCTIONS:

```
%%%%%%%%%%%%%%%%%%% ICM PHYSIOME CODING PROGRESS ASSIGNMENT %%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%% SEJAL GHATE, ZIXU HAN, YONGZHI SUN %%%%%%%%%%%%%%%%%%%%

clc;
clear all;
close all;


addpath(genpath('PhysioToolkitCardiacOutput_MatlabCode'));

%% Question 1:
% Enter the subject name here: input should be string

%%%%%%%%%% In question 1: subject number 20  %%%%%%%%%%%%%
subj_name = "s00020";

% Plot 20 peaks starting from the 10th hour
plot_peaks_abp(10, subj_name);

% Plot 20 peaks starting from the 11th hour
plot_peaks_abp(11, subj_name);

%% Question 3:

subj_name = "s00020";

% Using estimator 5 for the first 12 hours:
estimated_plots(12, 5, subj_name);
```

FUNCTION plot_peaks_abp to plot ABP waveform and respective features:

```
%% First Function
%%%%%%%%%%%%%%%% ICM Question 1 and 2 %%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%% Plot Peaks and point %%%%%%%%%%%%%%%%%%%%%%%%%%%

function [] = plot_peaks_abp(hours, subj_name)

    % Need folder of Physionet Toolkit
    %addpath(genpath(physio_link));

    % Need ABP file of subject
    path_abp = dir((fullfile(subj_name,'*_ABP.txt')));
    ABP = table2array(readtable(((fullfile(subj_name,path_abp.name)))));

    % Need n file of subject
    path_n = dir((fullfile(subj_name,'*n.txt')));
    n_data = readtable((fullfile(subj_name,path_n.name)));
```

```
    %
    time_s = hours*10*10;
    start_dur = find(ABP(:,1)>=time_s);
    t1 = start_dur(1);
    new_abp = ABP(t1:t1+1990,2);
    figure();
    plot(new_abp);
    title(sprintf("First 20 peaks of ABP wave starting at the %d th hour",
hours));
    xlabel("Time (s)");
    ylabel("Amplitude");

    %% Obtain onset time (x hours)

    r = wabp(new_abp);
    r_feat = abpfeature(new_abp',r);

    figure();
    plot(1:1:1991,new_abp,'LineWidth',1.5);
    hold on;
    sz = 100;

    for i = 1:length(r)
        if i < length(r)
            scatter(r(i,1),new_abp(r(i)),sz,'*','k','LineWidth',2);

scatter(r_feat(i,9),new_abp(r_feat(i,9)),sz,'x','r','LineWidth',2);

scatter(r_feat(i,11),new_abp(r_feat(i,11)),sz,'o','b','LineWidth',2);
        end
    end

    xlabel('Time Index','FontSize',12,'FontWeight','bold');
    ylabel('ABP (mmHg)','FontSize',12,'FontWeight','bold');
    title(sprintf("Visualization of ABP at %d hours and analyzed
features",hours),'FontSize',14,'FontWeight','bold');
    legend('ABP Waveform','Onset point','End of systole (0.3 beatperiod
^{0.5} method)','End of systole (lowest nonnegative slope method)');

    end
```

FUNCTION estimated_plots to estimate CO value using estimate_CO_v3 and plotting respective measurements at stem plots along side continuous values of other features:

```
%% Second Function estimated plots for assessing conitnuous vs. stem plots

%%%%%%%%%%%%%%%%% ICM Question 3 and 4 %%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%% Sejal Ghate, Zixu Han, Yongzhi Sun %%%%%%%%%%%%%%%

function [] = estimated_plots(hours, est, subj_name)    % Need folder of
Physionet Toolkit


    % Need ABP file of subject
    path_abp = dir((fullfile(subj_name,'*_ABP.txt')));
```

```
    ABP = table2array(readtable(((fullfile(subj_name,path_abp.name)))));

    % Need n file of subject
    path_n = dir((fullfile(subj_name,'*n.txt')));
    n_data = readtable((fullfile(subj_name,path_n.name)));


    end_time = hours*60*60; %the time of first 12 hours (in seconds)
    end_index = find(ABP(:,1)==end_time); %the index where the time (the
first column) is the end of first 12 hours
    abp_hrs = ABP(1:end_index,:);%ABP from first 12 hours

    %% Values of N_data that are non Zero and ABP within 12 hours:

    n_data_hrs_idx = find(n_data.ElapsedTime == end_time); %find the index
when 12 hour ends
    n_data_hrs = n_data(1:n_data_hrs_idx,:); % subjdata in first 12 hours

    cotd_idx_hrs = find(n_data_hrs.CO ~= 0); %find index of cotd that are not
equal to zero in first 12 hours
    cotd_hrs = n_data_hrs.CO(cotd_idx_hrs); %values of non-zero cotd in first
12 hours
    cotd_hrs_time = n_data_hrs.ElapsedTime(cotd_idx_hrs); %time when cotd was
measured in first 12 hours
    cotd_12_time_hr = (cotd_hrs_time)./3600; %time when cotd measured in
first 12 hours are converted to hours


    %% find first non 0 value of TCO and time corresponding to first TCO

    tco1 = cotd_hrs(1);%the first non-zero cotd value (used for calibration)
    tco1_time = n_data.ElapsedTime(n_data.CO == tco1);% the time when the
first measurement of cotd was conducted

    %% Applying estimateCO_v3 on data

    % to on
    t_on = wabp(abp_hrs(:,2)); %the onset times of first 12 hours of ABPs

    % getting features
    features = abpfeature(abp_hrs(:,2),t_on);%get features of ABP_12hours

    % getting beats
    beats = jSQI(features,t_on,abp_hrs(:,2));%beats quality

    %using the fifth estimator
    est=5;

    %set filter order as zero
    filt_order = 0;

    %get uncalibrated estimated co
    [co, to, told, fea] = estimateCO_v3(t_on,features,beats,est,filt_order);


    %% Calibration of estimated co values from estimate function
```

```
    to_s = round(to*60); %convert time to seconds: to is orignally in minutes

    index_time = find(to_s == tco1_time); %the index of the first
co_uncalibrated from co when the first tco occurs in n_data
    est_co_uncalibrated = co(index_time); %the value of uncalibrated co when
the firsted cotd is measured

    k = tco1/est_co_uncalibrated; %get calibration factor k=first cotd
measured/the corresponding uncalibrated estimated co at the same time

    est_co_calibrated = k*co; %input k to the L-function to obtain calibrated
co

    %% non zero tco values table

    non0_idx = find(n_data_hrs.CO ~= 0 );
    non0_values = n_data_hrs(non0_idx,:);
    non0_pp = non0_values.ABPSys - non0_values.ABPDias;
    non0_map = non0_values.ABPMean;
    non0_hr = non0_values.HR;

    %%  plot Cotd in first 12 hours with a function of time in hour
    figure();
    tiledlayout(4,1);
    ax1=nexttile;

    % continuous plots
    plot(ax1,(to_s)./3600,est_co_calibrated); %plot calibrated co with a
function of time in first hours
    hold on;
    % stem plot
    stem(ax1,cotd_12_time_hr,cotd_hrs,'MarkerEdgeColor',[0 .5
.5],'MarkerFaceColor','#D95319','LineWidth',1.5); %stem plot of cotd

    title(ax1,'Estimated CO versus thermodilution CO measurements');
    ylabel(ax1,'CO');
    hold off;

    %% plot of pulse pressure

    %stem plot
    ax2=nexttile;

    %continuous plot
    plot(ax2,(to_s)./3600,fea(:,5));
    hold on;
    %stem plot
    stem(ax2,cotd_12_time_hr,non0_pp,'MarkerEdgeColor',[0 .5
.5],'MarkerFaceColor','#D95319','LineWidth',1.5);

    title(ax2,'Estimated pulse pressure vs PP measurements');
    ylabel(ax2,'PP');
    hold off;

    %% plot of MAP

    ax3=nexttile;
```

```
    %continuous plot
    plot(ax3,(to_s)./3600,fea(:,6));
    hold on;
    %stem plot
    stem(ax3,cotd_12_time_hr,non0_map,'MarkerEdgeColor',[0 .5
.5],'MarkerFaceColor','#D95319','LineWidth',1.5);

    title(ax3,'Estimated mean arterial pressure vs. MAP measurements');
    ylabel(ax3,'MAP');
    hold off;

    %% plot of HR

    ax4=nexttile;

    %continuous plot
    plot(ax4,(to_s)./3600,60*125./fea(:,7));
    hold on;
    %stem plot
    stem(ax4,cotd_12_time_hr,non0_hr,'MarkerEdgeColor',[0 .5
.5],'MarkerFaceColor','#D95319','LineWidth',1.5);

    title(ax4,'Estimated heart rate vs HR measurements');
    xlabel(ax4,'time[hours]');
    ylabel(ax4,'HR');
    hold off;
end
```