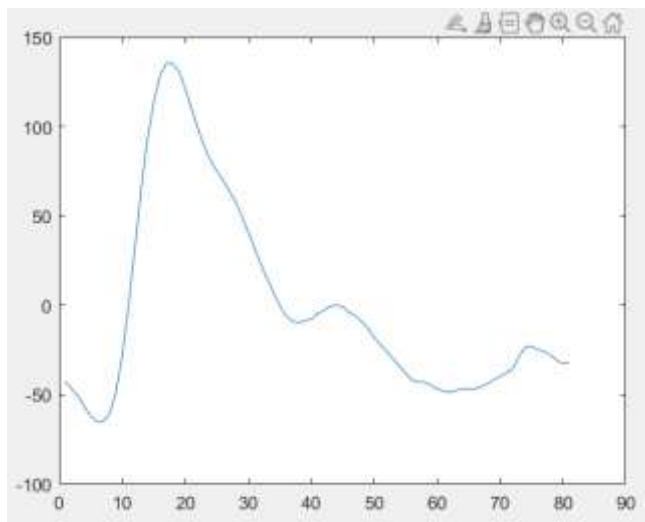


Práctica Calificada – Control de lectura

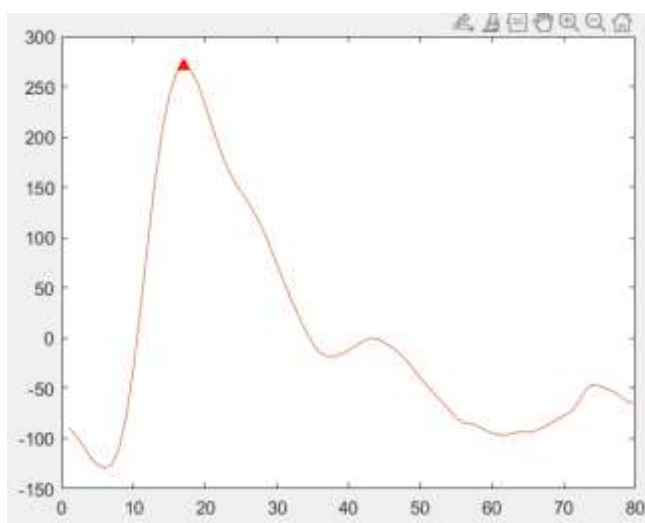
Nombre: Davis Bremdow Salazar Roa

En esta pregunta se busca definir el tiempo de inicio de la presión sistólica y el punto máximo de presión de la primera derivada de la forma de onda de la señal de presión periférica $T_{dmaxdpdt}$,

Para ello se realizó el procesamiento de la información mediante Matlab en el cual se eliminó la primera columna de los datos en los cuales se define el tiempo de cada muestra y se procesó la segunda columna con los datos de presión, seguido a esto se graficó la señal de presión eliminando el componente dc o valor medio de la señal obteniendo valores aproximados o considerados ideales para la presión.



En la gráfica se puede apreciar la gráfica de los datos en crudo y luego para el cálculo de $T_{dmaxdpdt}$ se tuvo en cuenta la aplicación de la derivada para obtener este valor el cual ocurre en los 0.17 segundos



En la figura se aprecia la primera derivada de la señal y el pico máximo Tmaxdpdt con una valor de 270.5185 .

El código Matlab para todo este proceso es:

1. Lectura de datos y eliminar el valor medio

```
clear, clc, close all;
t = 0: 0.01: 0.8;

%% Objetivos
% 1. Calcular en que tiempo inicia la sistole
% 2. Calcular Tmaxdpdt de la señal de presión (Punto máximo en la curva de la primera derivada)

%% Obtención y gráfica de la señal
% Recuperando el archivo como una matriz
datos_presion = dlmread('presion_periferica.txt');
% Ignorando o eliminando la primera columna
datos_presion(:, 1) = [];
% Eliminando el valor medio
datos_presion = datos_presion - mean(datos_presion);
% Mostrando los datos de la presión periférica
plot(datos_presion);
hold on
```

2. Calculo de Tmaxdpdt y la primera derivada

```
%% Tmaxdpdt Punto máximo en la curva de la primera derivada
primera_derivada(1) = 0;
for i = 1:length(datos_presion) - 1
    primera_derivada(i) = datos_presion(i + 1) - datos_presion(i);
end
% Obtención de valor máximo

% Selección del rango umbral (determinación del máximo)
umbral_minimo = 100;

% Inicializar variables
max_locales = []; % Almacenar valores de los máximos locales
posiciones = []; % Almacenar posiciones de los máximos locales

% Algoritmo para encontrar máximos locales
for i = 2:length(primera_derivada)-1
    % Comprobar si el elemento actual es mayor que sus vecinos y supera el umbral
    if primera_derivada(i) >= primera_derivada(i-1) && primera_derivada(i) >= primera_derivada(i+1) && primera_derivada(i) >= umbral_minimo
        max_locales = [max_locales, primera_derivada(i)]; % Guardar el valor del máximo
        posiciones = [posiciones, i]; % Guardar la posición del máximo
    end
end

% Mostrar resultados
disp(['Máximos locales encontrados (con umbral mínimo de ', num2str(umbral_minimo), '):']);
disp(length(max_locales));
disp('Máximo local y Posición de los máximos locales:');
disp([max_locales, posiciones]);
```

