Examination of signal parameters

Sojung Son

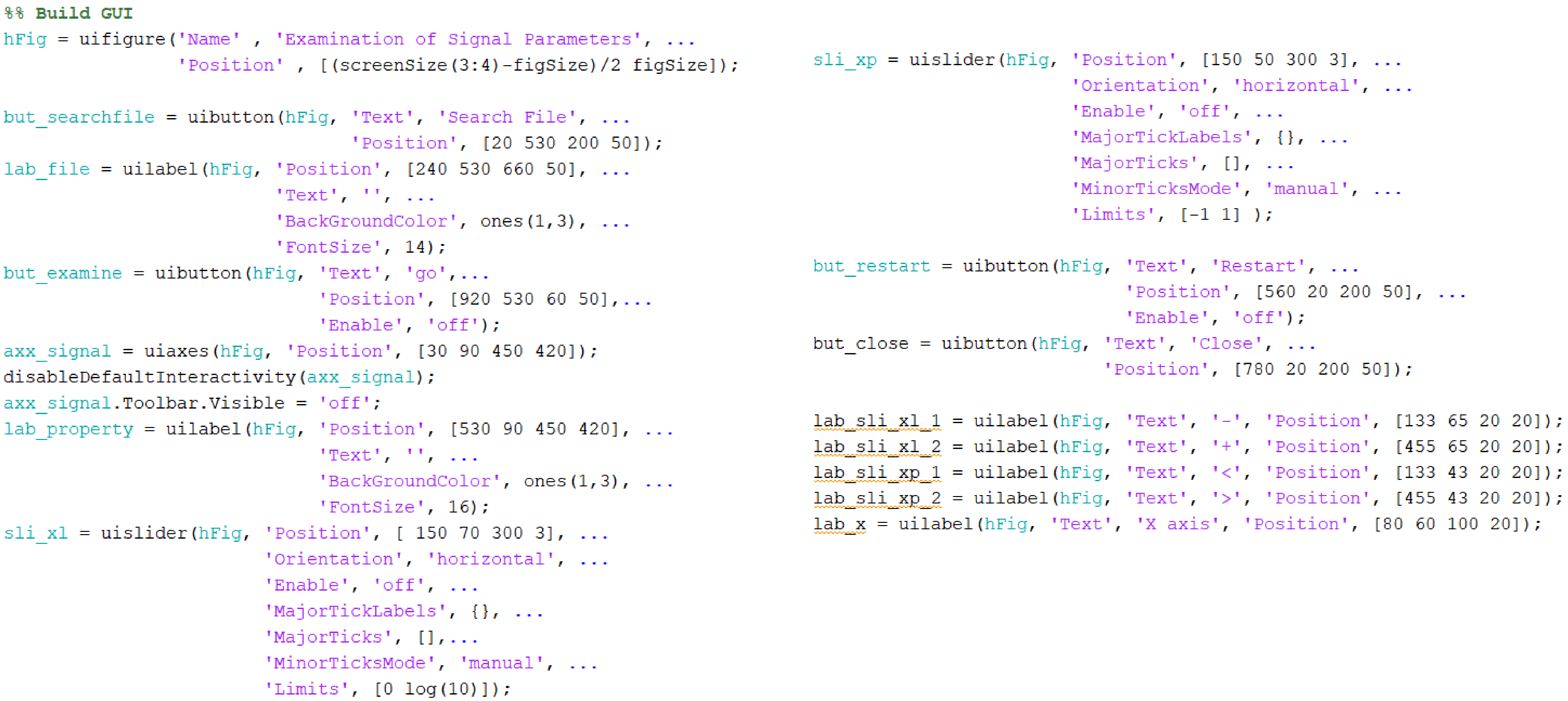
sonsojun@fel.cvut.cz

Examination of signal parameters

From loaded signal/data (voltage, sound, blood pressure, wind speed, …) examine its basic parameters as amplitude, DC component, peak-peak voltage, rms voltage, frequency content, overshoot, … Examined parameters depends on type of the signal. Source of the signal is up to you.

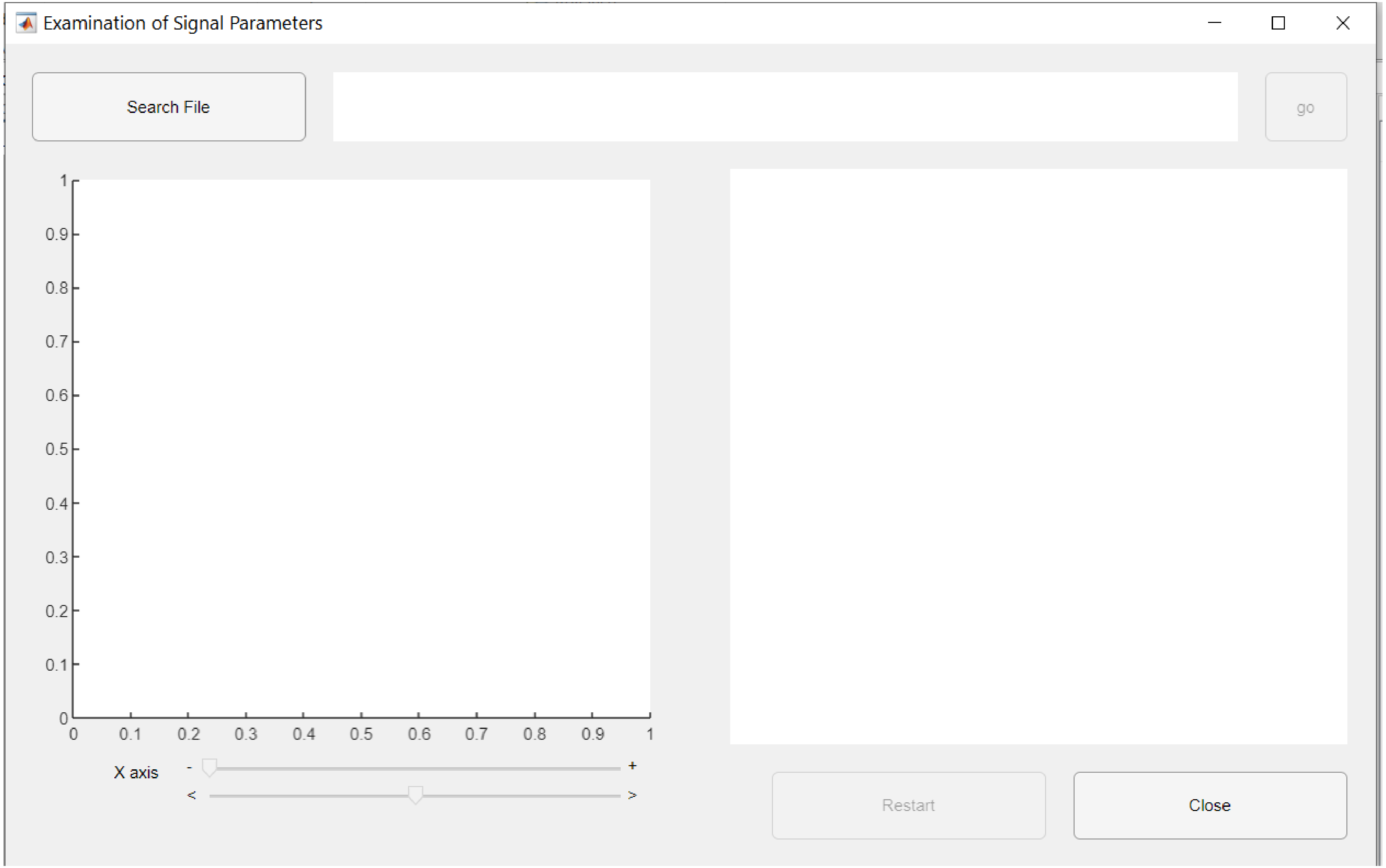
→ ECG signal (Electrocardiography)

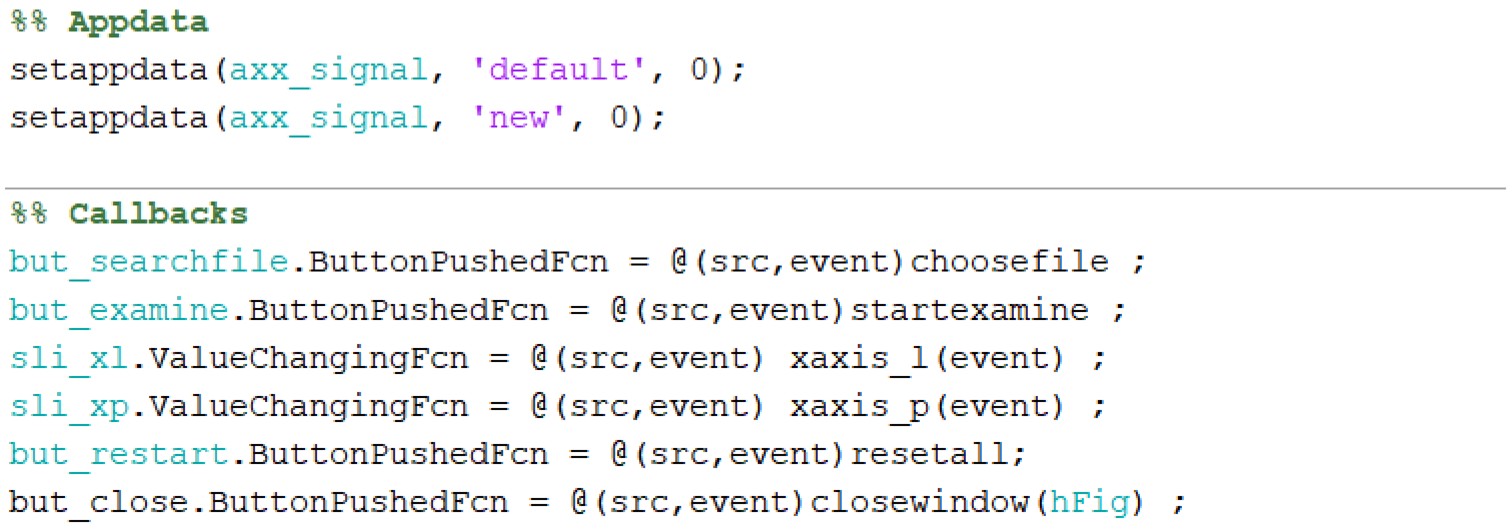
How does it work?
run the program
search ECG data (xlsx file)
push button to start examination
go through some correction of data
show signal graph and several paramemters
parameters : R-R interval, P-R interval, QRS width
end program or reset to examine another data



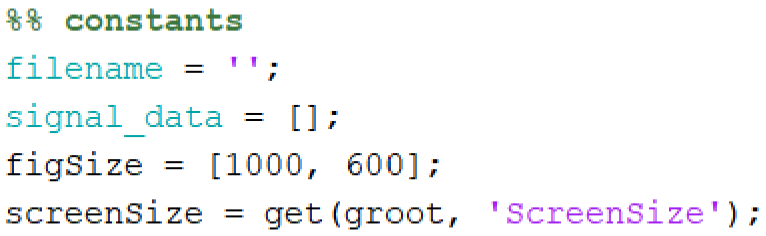
```
%% Build GUI
hFig = uifigure('Name' , 'Examination of Signal Parameters', ...
                'Position' , [(screenSize(3:4)-figSize)/2 figSize]);

but_searchfile = uibutton(hFig, 'Text', 'Search File', ...
                          'Position', [20 530 200 50]);
lab_file = uilabel(hFig, 'Position', [240 530 660 50], ...
                   'Text', '', ...
                   'BackGroundColor', ones(1,3), ...
                   'FontSize', 14);
but_examine = uibutton(hFig, 'Text', 'go',...
                       'Position', [920 530 60 50],...
                       'Enable', 'off');
axx_signal = uiaxes(hFig, 'Position', [30 90 450 420]);
disableDefaultInteractivity(axx_signal);
axx_signal.Toolbar.Visible = 'off';
lab_property = uilabel(hFig, 'Position', [530 90 450 420], ...
                       'Text', '', ...
                       'BackGroundColor', ones(1,3), ...
                       'FontSize', 16);
sli_xl = uislider(hFig, 'Position', [ 150 70 300 3], ...
                  'Orientation', 'horizontal', ...
                  'Enable', 'off', ...
                  'MajorTickLabels', {}, ...
                  'MajorTicks', [],...
                  'MinorTicksMode', 'manual', ...
                  'Limits', [0 log(10)]);
```

```
sli_xp = uislider(hFig, 'Position', [150 50 300 3], ...
                  'Orientation', 'horizontal', ...
                  'Enable', 'off', ...
                  'MajorTickLabels', {}, ...
                  'MajorTicks', [], ...
                  'MinorTicksMode', 'manual', ...
                  'Limits', [-1 1] );

but_restart = uibutton(hFig, 'Text', 'Restart', ...
                       'Position', [560 20 200 50], ...
                       'Enable', 'off');
but_close = uibutton(hFig, 'Text', 'Close', ...
                     'Position', [780 20 200 50]);

lab_sli_xl_1 = uilabel(hFig, 'Text', '-', 'Position', [133 65 20 20]);
lab_sli_xl_2 = uilabel(hFig, 'Text', '+', 'Position', [455 65 20 20]);
lab_sli_xp_1 = uilabel(hFig, 'Text', '<', 'Position', [133 43 20 20]);
lab_sli_xp_2 = uilabel(hFig, 'Text', '>', 'Position', [455 43 20 20]);
lab_x = uilabel(hFig, 'Text', 'X axis', 'Position', [80 60 100 20]);
```

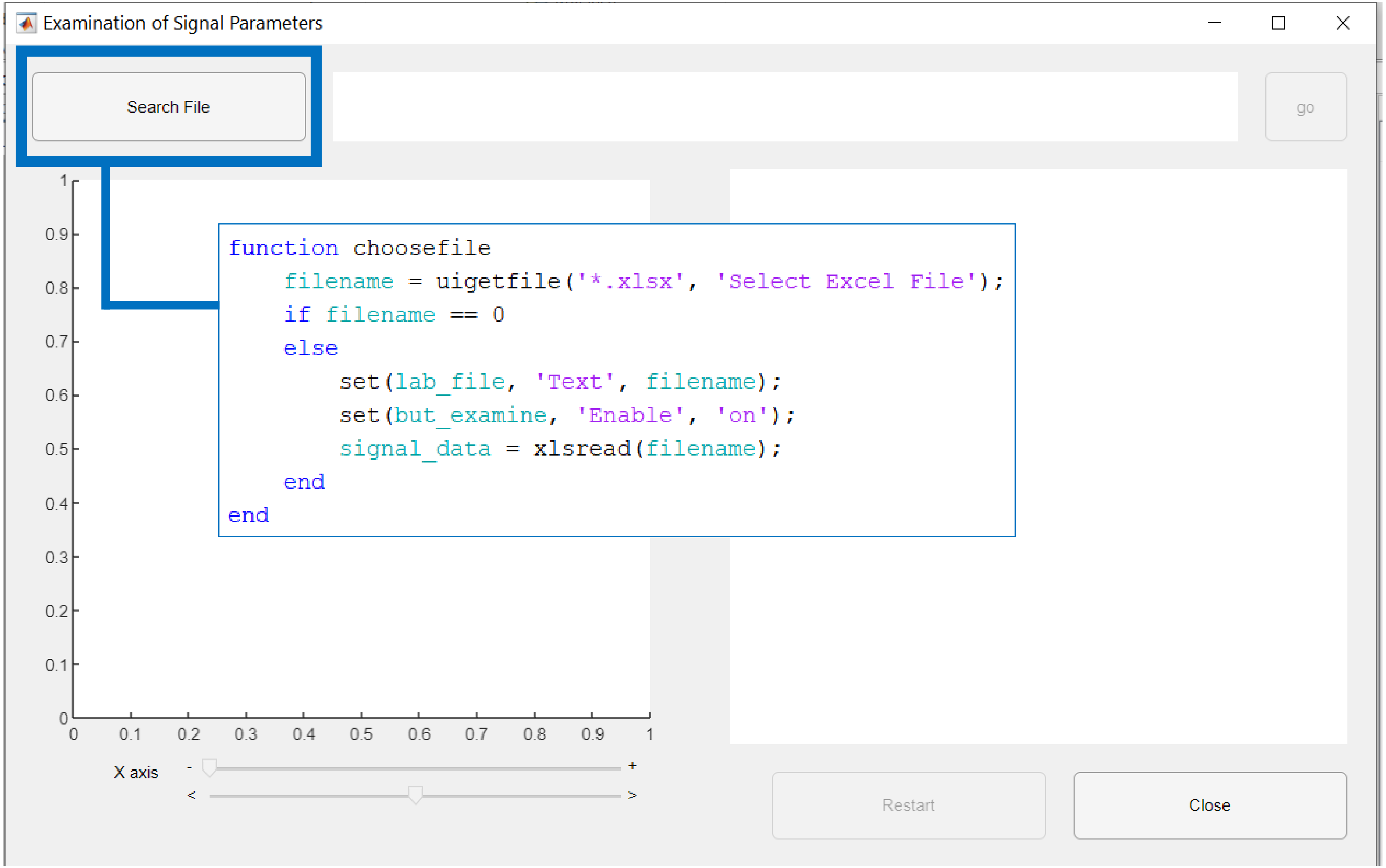

Examination of Signal Parameters
Search File
go
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
X axis
-
+
<
>
Restart
Close



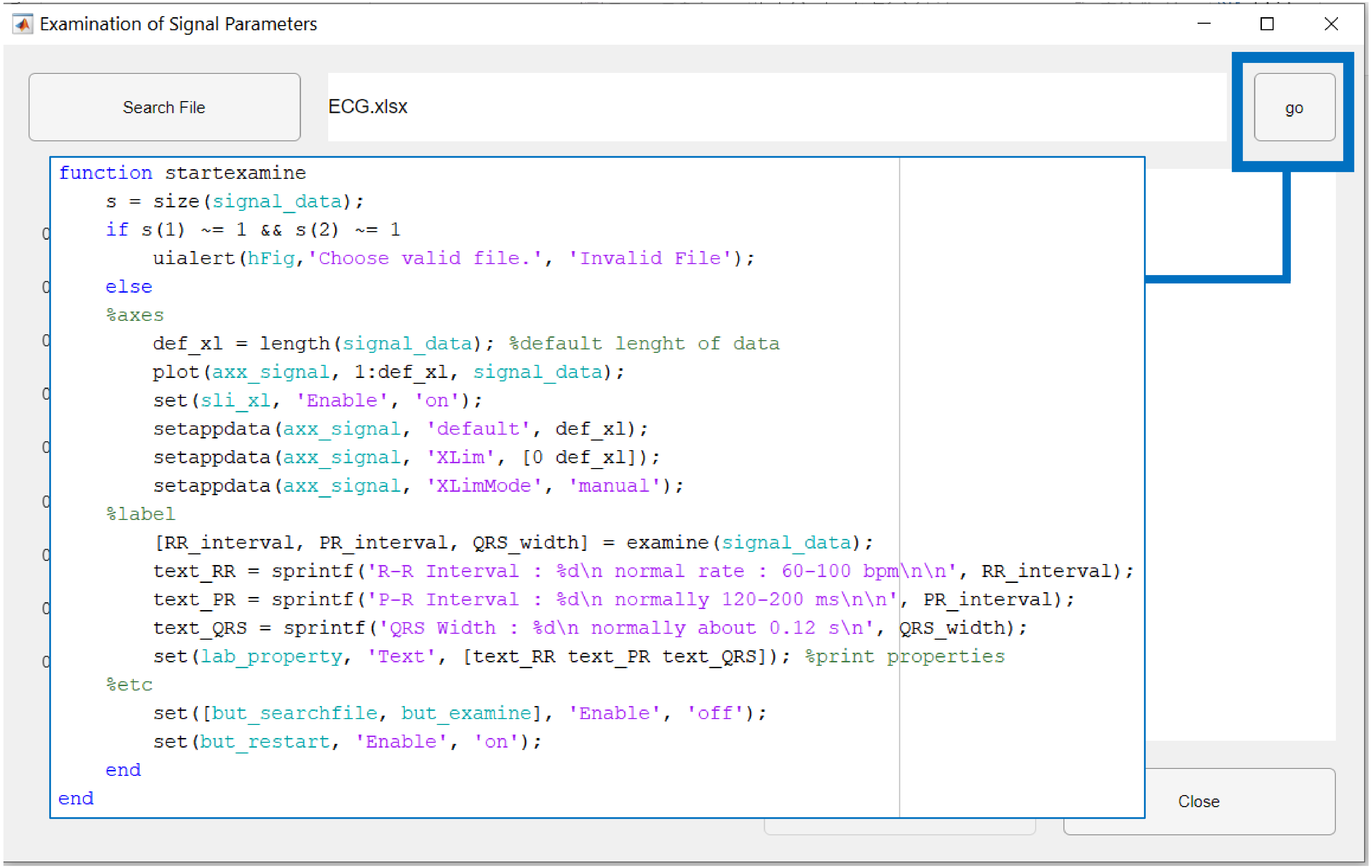
```
%% Appdata
setappdata(axx_signal, 'default', 0);
setappdata(axx_signal, 'new', 0);

%% Callbacks
but_searchfile.ButtonPushedFcn = @(src,event)choosefile ;
but_examine.ButtonPushedFcn = @(src,event)startexamine ;
sli_xl.ValueChangingFcn = @(src,event) xaxis_l(event) ;
sli_xp.ValueChangingFcn = @(src,event) xaxis_p(event) ;
but_restart.ButtonPushedFcn = @(src,event)resetall;
but_close.ButtonPushedFcn = @(src,event)closewindow(hFig) ;
```



```
%% constants
filename = '';
signal_data = [];
figSize = [1000, 600];
screenSize = get(groot, 'ScreenSize');
```

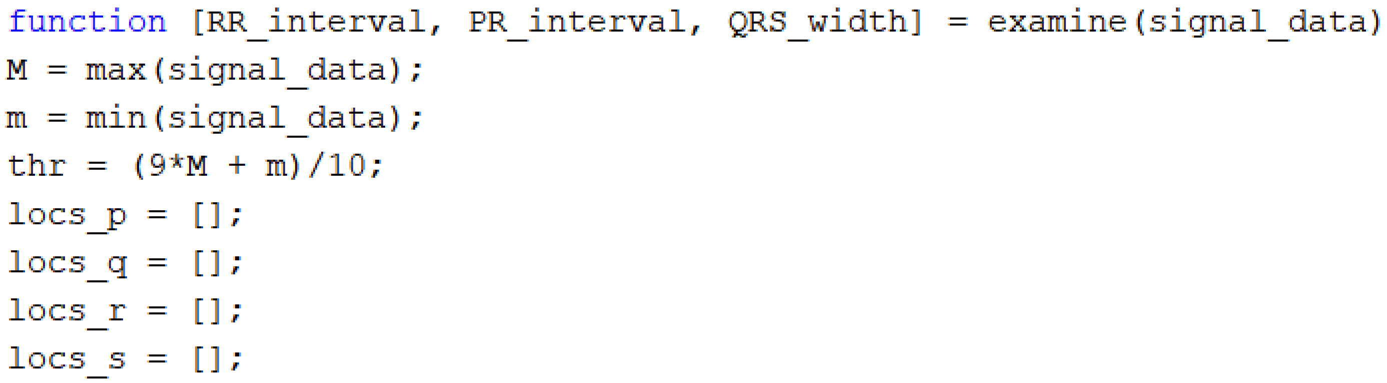


```
function choosefile
    filename = uigetfile('*.xlsx', 'Select Excel File');
    if filename == 0
    else
        set(lab_file, 'Text', filename);
        set(but_examine, 'Enable', 'on');
        signal_data = xlsread(filename);
    end
end
```

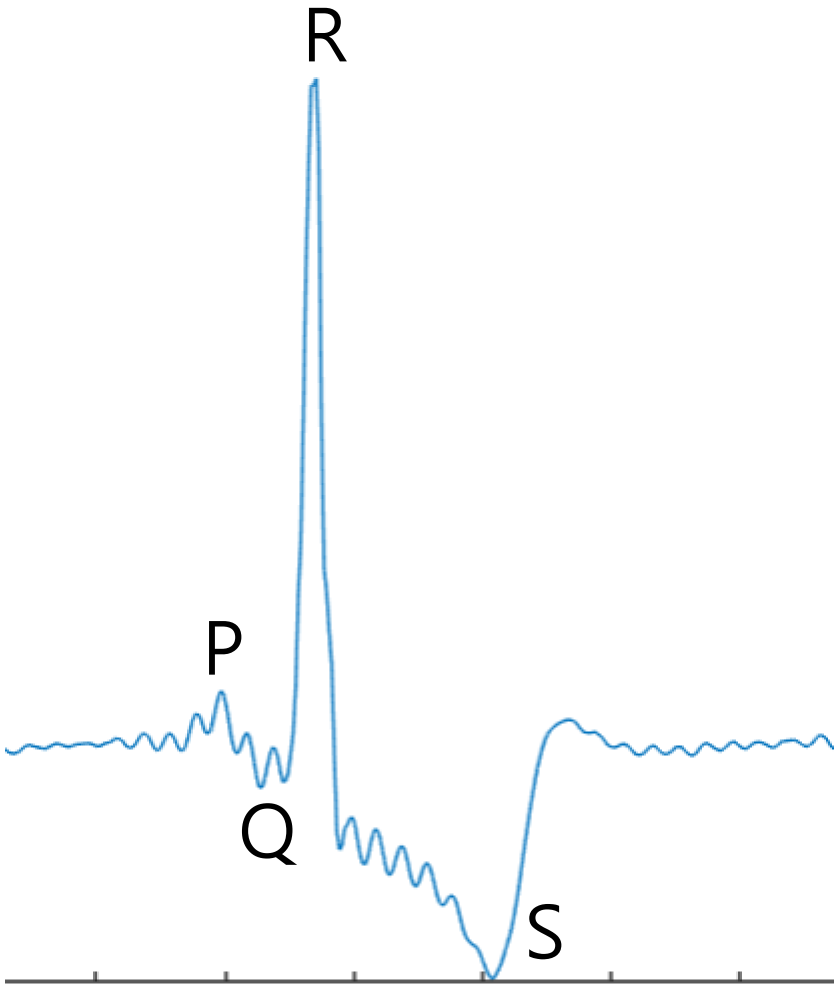


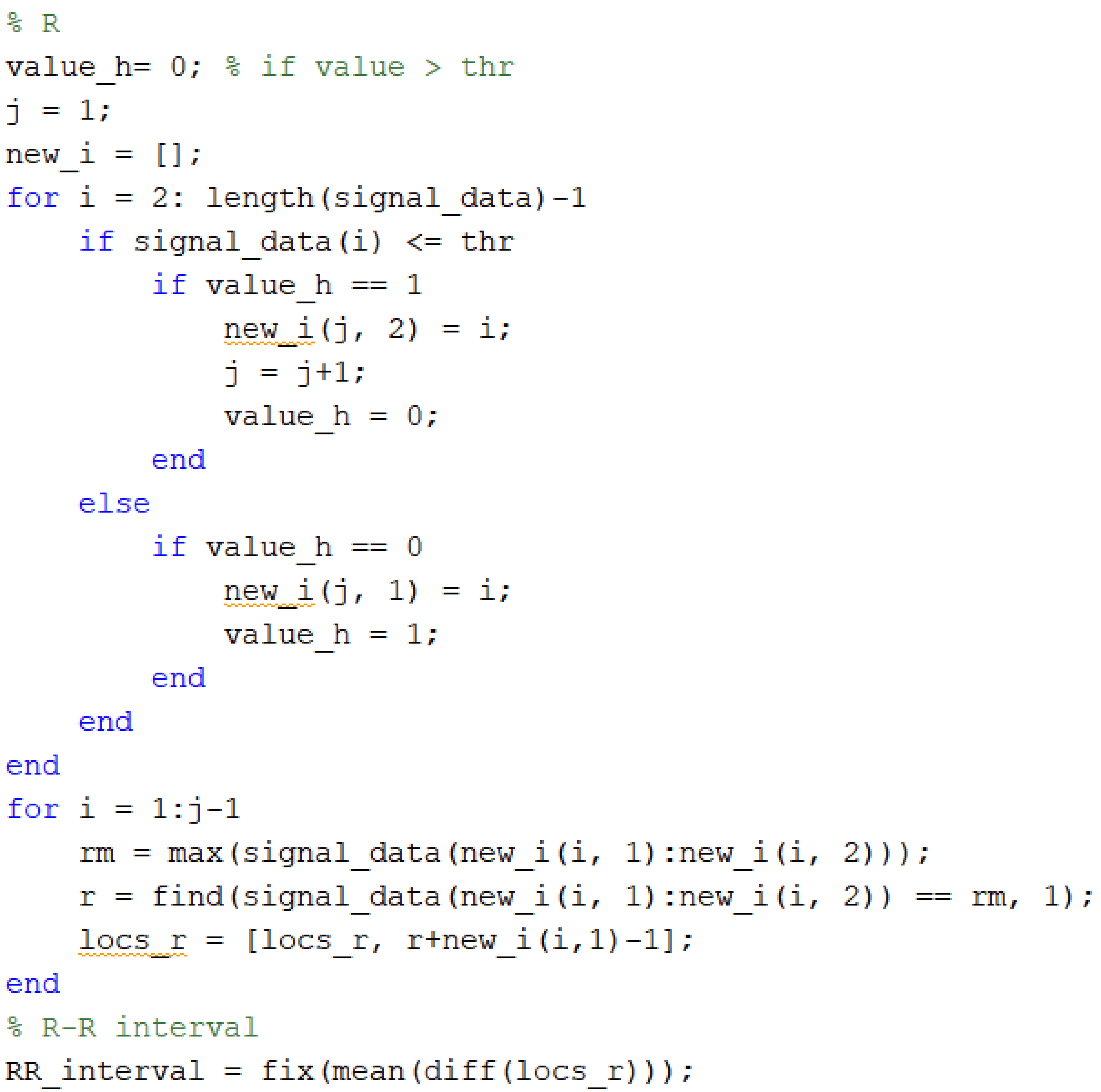
ECG.xlsx

```
function startexamine
    s = size(signal_data);
    if s(1) ~= 1 && s(2) ~= 1
        uialert(hFig,'Choose valid file.', 'Invalid File');
    else
    %axes
        def_xl = length(signal_data); %default lenght of data
        plot(axx_signal, 1:def_xl, signal_data);
        set(sli_xl, 'Enable', 'on');
        setappdata(axx_signal, 'default', def_xl);
        setappdata(axx_signal, 'XLim', [0 def_xl]);
        setappdata(axx_signal, 'XLimMode', 'manual');
    %label
        [RR_interval, PR_interval, QRS_width] = examine(signal_data);
        text_RR = sprintf('R-R Interval : %d\n normal rate : 60-100 bpm\n\n', RR_interval);
        text_PR = sprintf('P-R Interval : %d\n normally 120-200 ms\n\n', PR_interval);
        text_QRS = sprintf('QRS Width : %d\n normally about 0.12 s\n', QRS_width);
        set(lab_property, 'Text', [text_RR text_PR text_QRS]); %print properties
    %etc
        set([but_searchfile, but_examine], 'Enable', 'off');
        set(but_restart, 'Enable', 'on');
    end
end
```

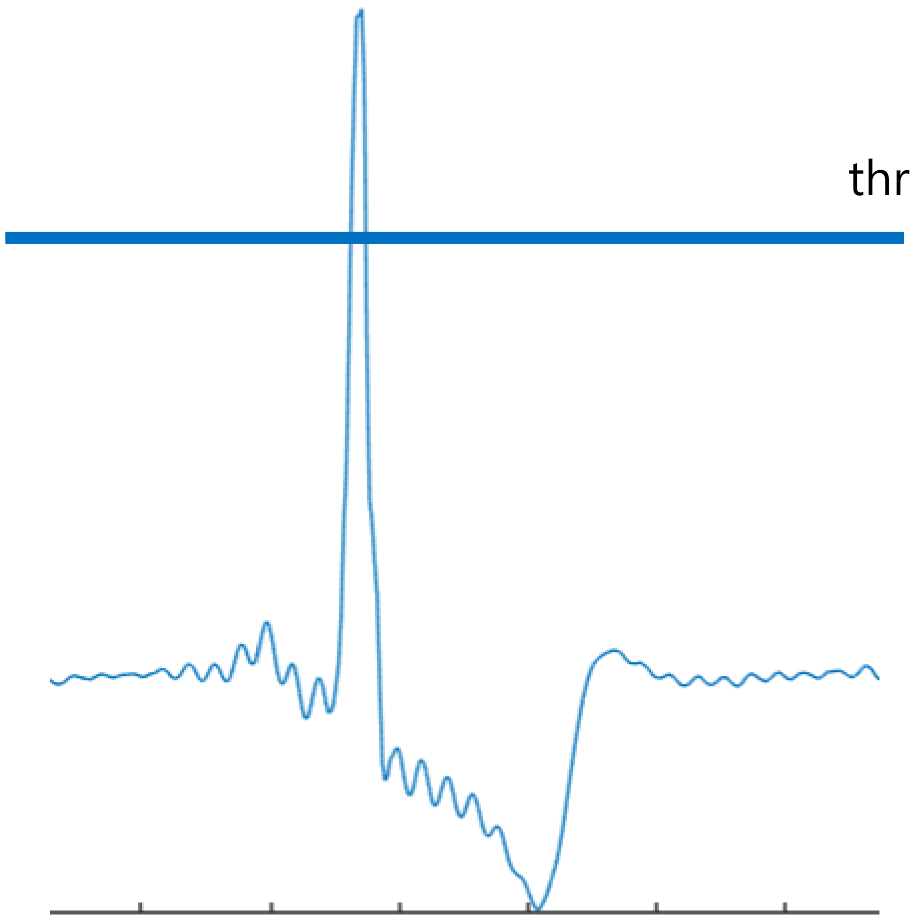


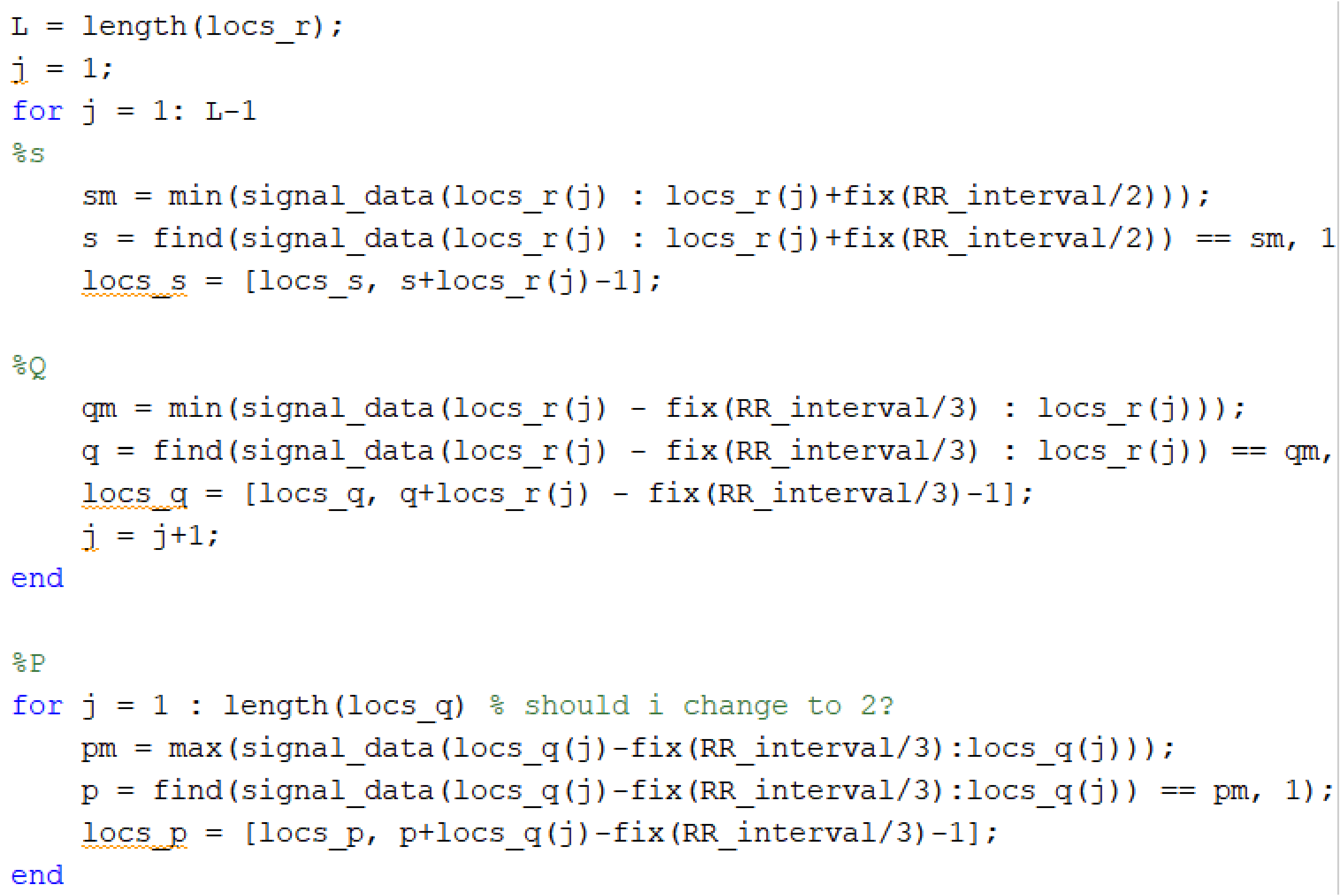
```
function [RR_interval, PR_interval, QRS_width] = examine(signal_data)
M = max(signal_data);
m = min(signal_data);
thr = (9*M + m)/10;
locs_p = [];
locs_q = [];
locs_r = [];
locs_s = [];
```



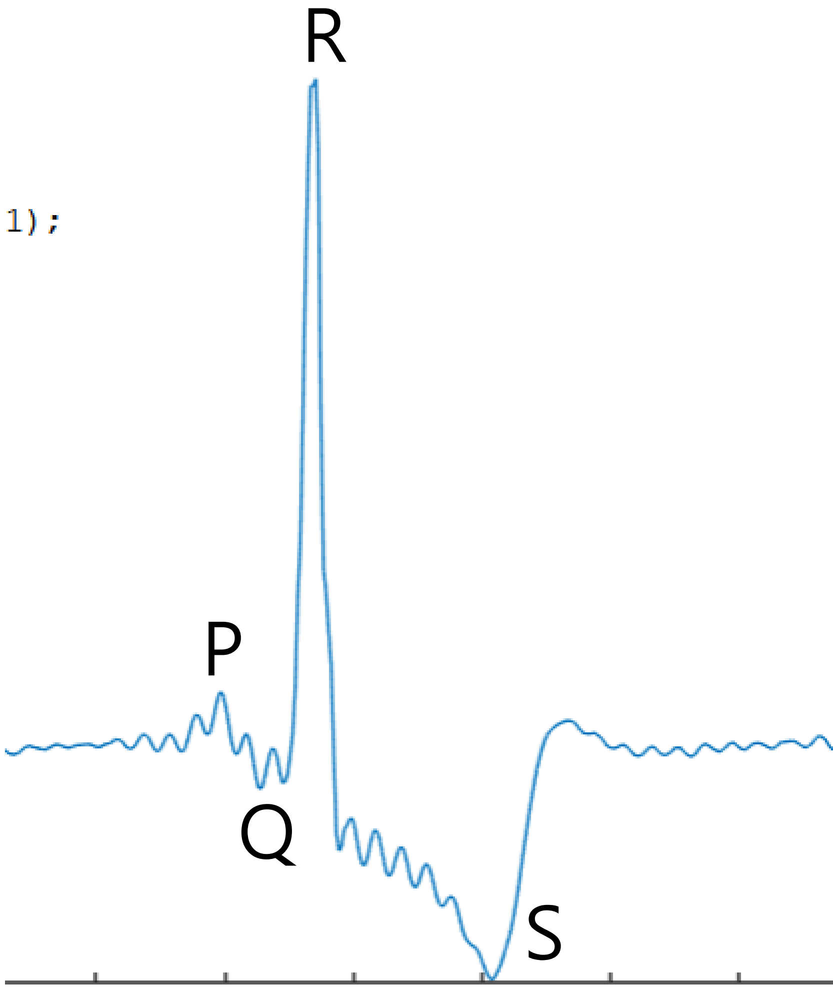


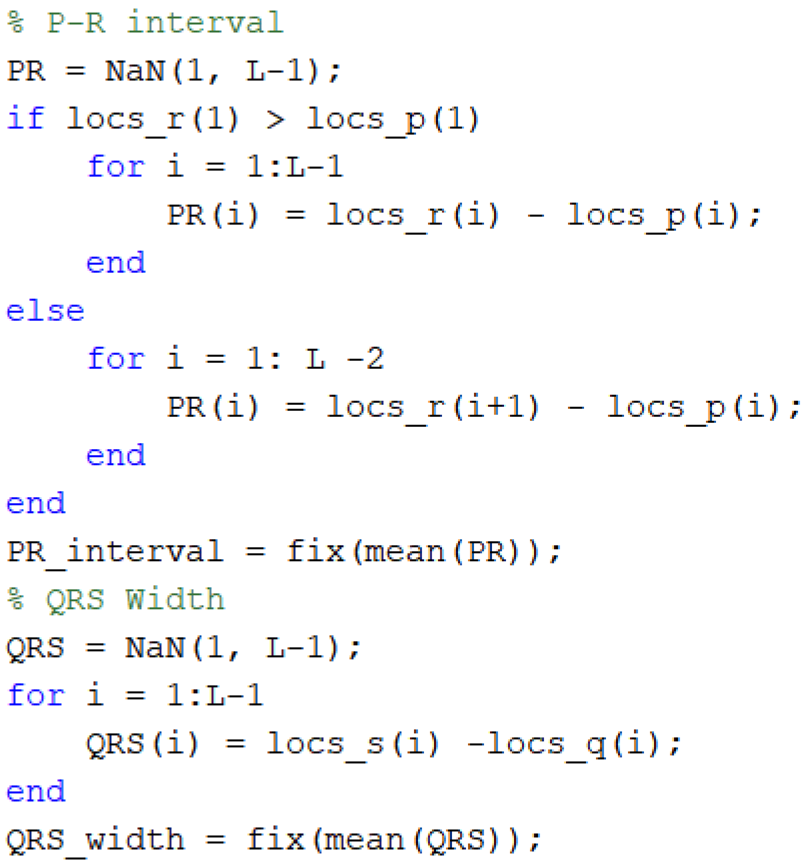
```
% R
value_h= 0; % if value > thr
j = 1;
new_i = [];
for i = 2: length(signal_data)-1
    if signal_data(i) <= thr
        if value_h == 1
            new_i(j, 2) = i;
            j = j+1;
            value_h = 0;
        end
    else
        if value_h == 0
            new_i(j, 1) = i;
            value_h = 1;
        end
    end
end
for i = 1:j-1
    rm = max(signal_data(new_i(i, 1):new_i(i, 2)));
    r = find(signal_data(new_i(i, 1):new_i(i, 2)) == rm, 1);
    locs_r = [locs_r, r+new_i(i,1)-1];
end
% R-R interval
RR_interval = fix(mean(diff(locs_r)));
```



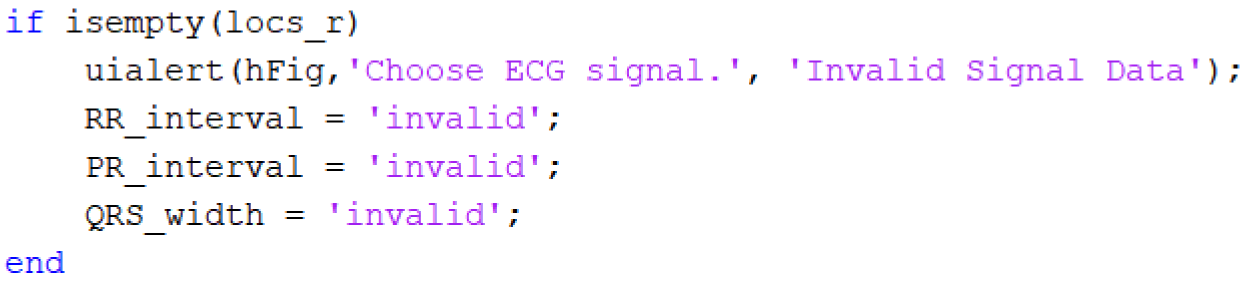


```
L = length(locs_r);
j = 1;
for j = 1: L-1
%s
    sm = min(signal_data(locs_r(j) : locs_r(j)+fix(RR_interval/2)));
    s = find(signal_data(locs_r(j) : locs_r(j)+fix(RR_interval/2)) == sm, 1);
    locs_s = [locs_s, s+locs_r(j)-1];

%Q
    qm = min(signal_data(locs_r(j) - fix(RR_interval/3) : locs_r(j)));
    q = find(signal_data(locs_r(j) - fix(RR_interval/3) : locs_r(j)) == qm, 1);
    locs_q = [locs_q, q+locs_r(j) - fix(RR_interval/3)-1];
    j = j+1;
end

%P
for j = 1 : length(locs_q) % should i change to 2?
    pm = max(signal_data(locs_q(j)-fix(RR_interval/3):locs_q(j)));
    p = find(signal_data(locs_q(j)-fix(RR_interval/3):locs_q(j)) == pm, 1);
    locs_p = [locs_p, p+locs_q(j)-fix(RR_interval/3)-1];
end
```

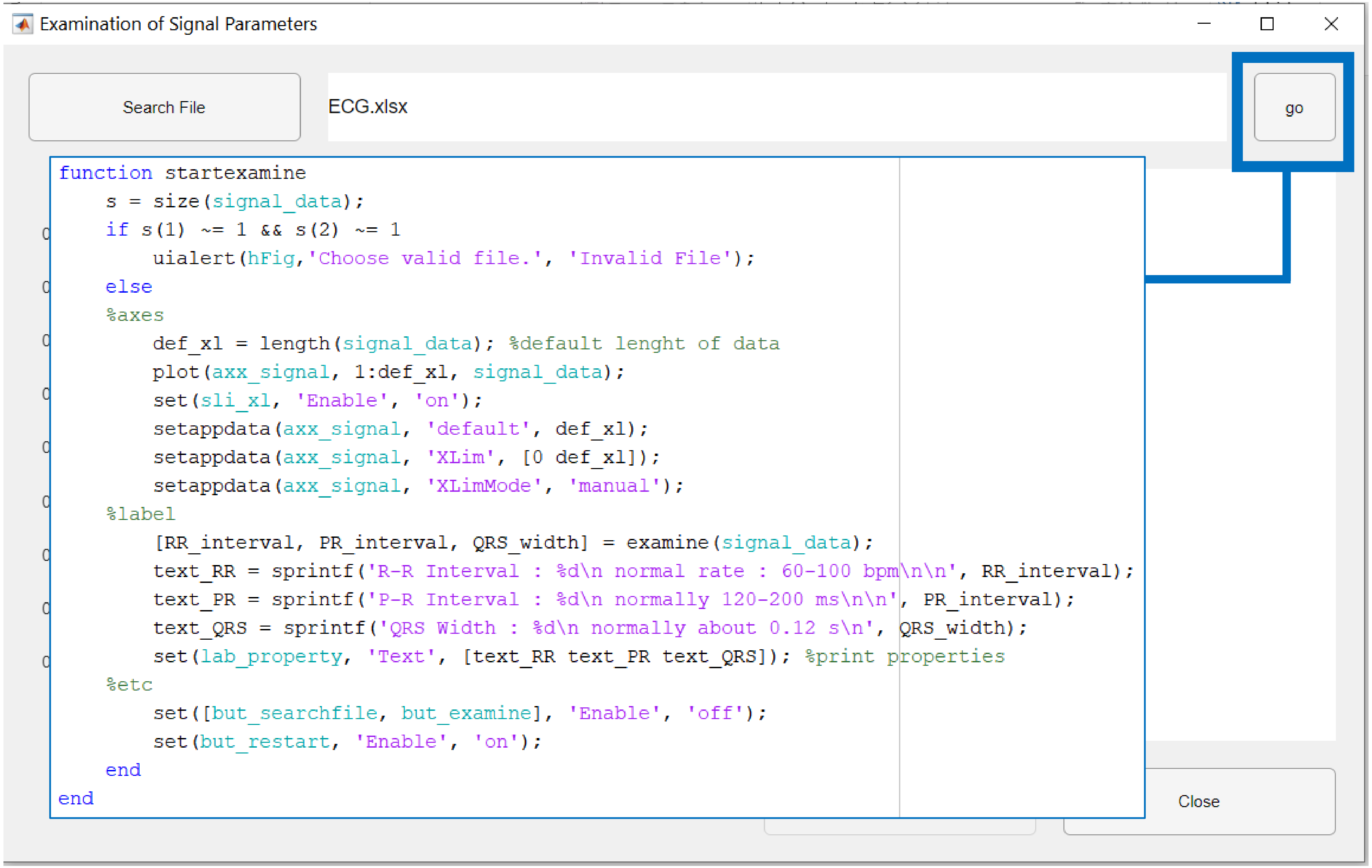




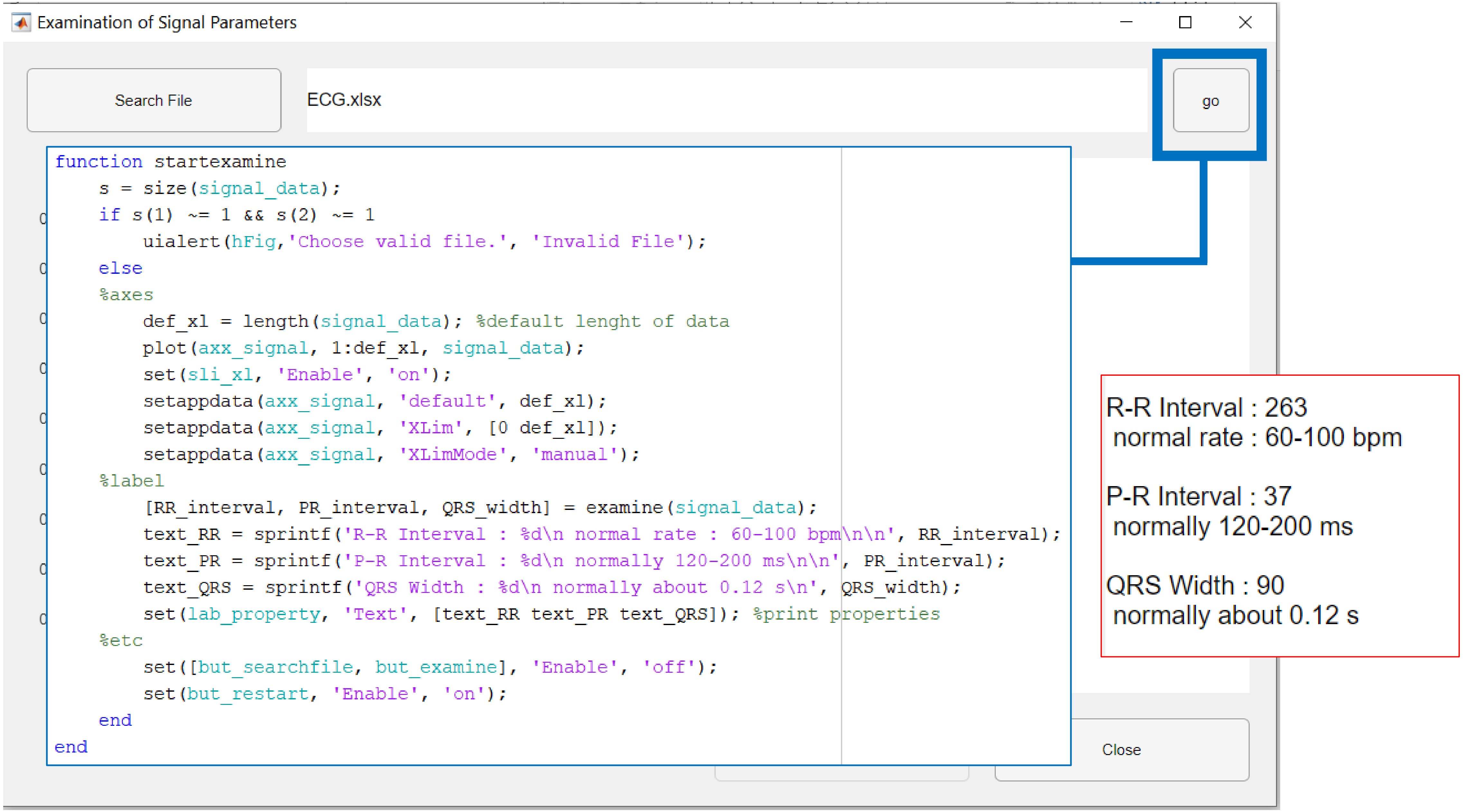
```
% P-R interval
PR = NaN(1, L-1);
if locs_r(1) > locs_p(1)
    for i = 1:L-1
        PR(i) = locs_r(i) - locs_p(i);
    end
else
    for i = 1: L -2
        PR(i) = locs_r(i+1) - locs_p(i);
    end
end
PR_interval = fix(mean(PR));
% QRS Width
QRS = NaN(1, L-1);
for i = 1:L-1
    QRS(i) = locs_s(i) -locs_q(i);
end
QRS_width = fix(mean(QRS));
```



```
if isempty(locs_r)
    uialert(hFig,'Choose ECG signal.', 'Invalid Signal Data');
    RR_interval = 'invalid';
    PR_interval = 'invalid';
    QRS_width = 'invalid';
end
```

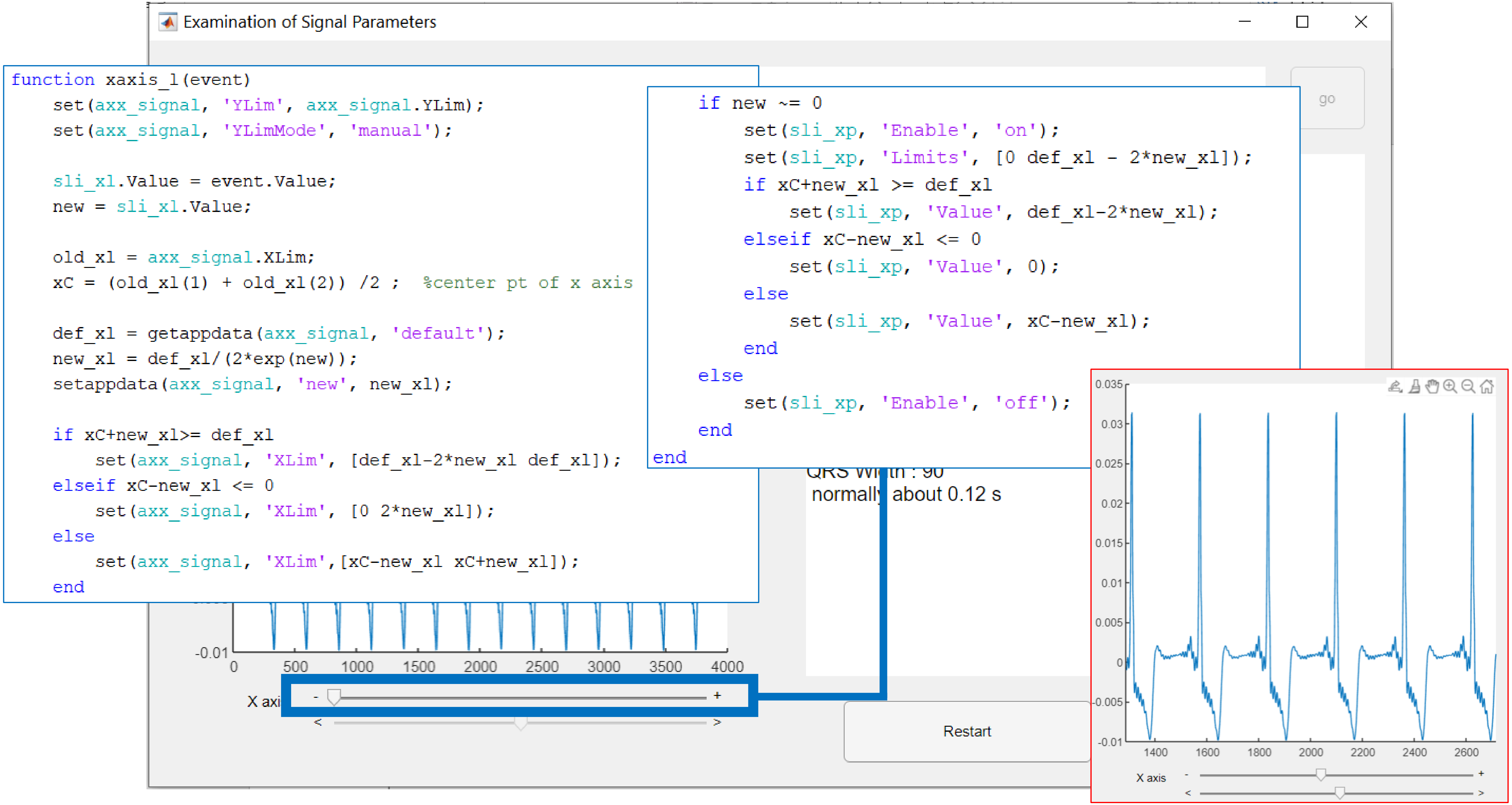


```
function startexamine
    s = size(signal_data);
    if s(1) ~= 1 && s(2) ~= 1
        uialert(hFig,'Choose valid file.', 'Invalid File');
    else
    %axes
        def_xl = length(signal_data); %default lenght of data
        plot(axx_signal, 1:def_xl, signal_data);
        set(sli_xl, 'Enable', 'on');
        setappdata(axx_signal, 'default', def_xl);
        setappdata(axx_signal, 'XLim', [0 def_xl]);
        setappdata(axx_signal, 'XLimMode', 'manual');
    %label
        [RR_interval, PR_interval, QRS_width] = examine(signal_data);
        text_RR = sprintf('R-R Interval : %d\n normal rate : 60-100 bpm\n\n', RR_interval);
        text_PR = sprintf('P-R Interval : %d\n normally 120-200 ms\n\n', PR_interval);
        text_QRS = sprintf('QRS Width : %d\n normally about 0.12 s\n', QRS_width);
        set(lab_property, 'Text', [text_RR text_PR text_QRS]); %print properties
    %etc
        set([but_searchfile, but_examine], 'Enable', 'off');
        set(but_restart, 'Enable', 'on');
    end
end
```



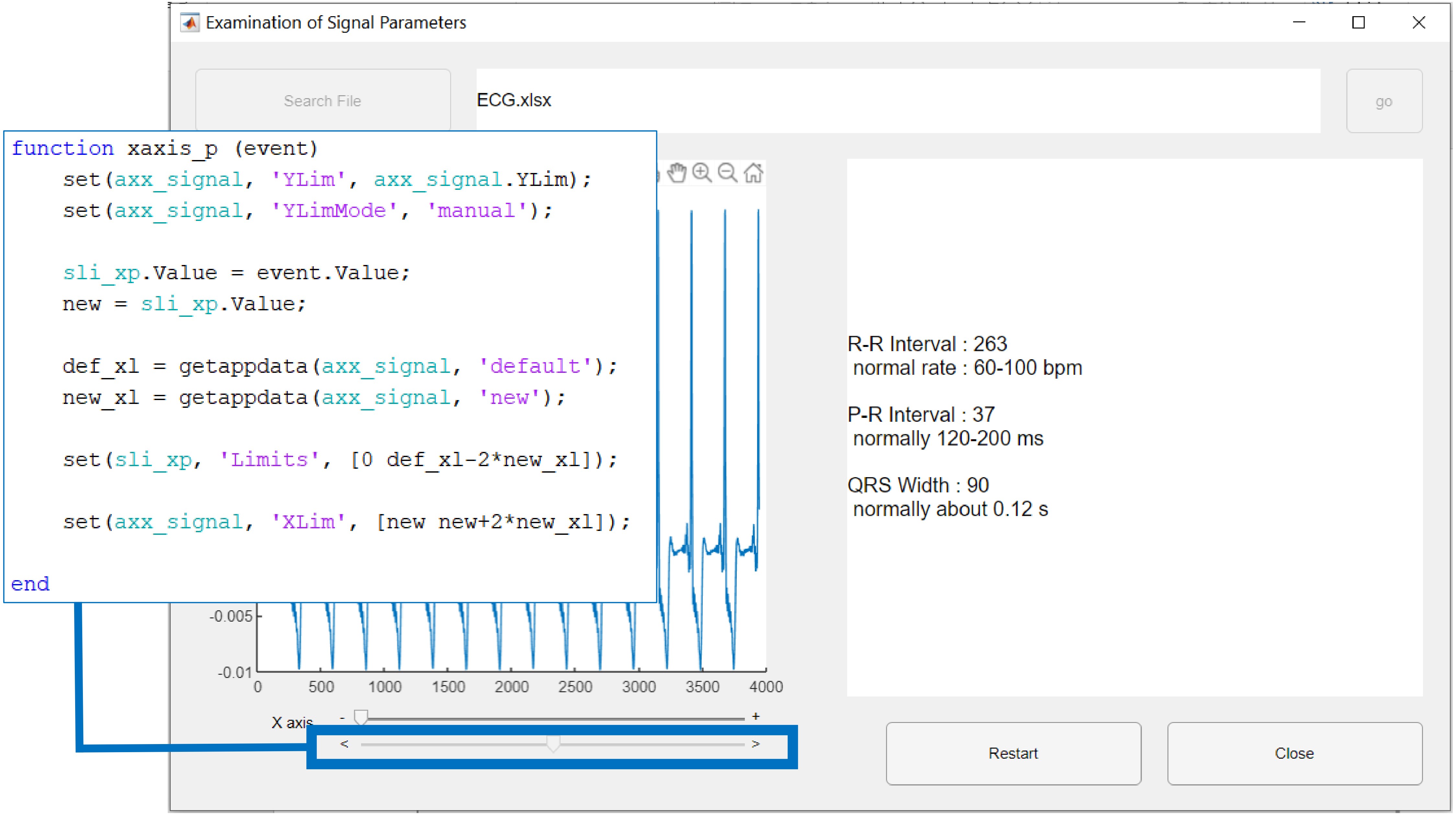
```
function startexamine
    s = size(signal_data);
    if s(1) ~= 1 && s(2) ~= 1
        uialert(hFig,'Choose valid file.', 'Invalid File');
    else
    %axes
        def_xl = length(signal_data); %default lenght of data
        plot(axx_signal, 1:def_xl, signal_data);
        set(sli_xl, 'Enable', 'on');
        setappdata(axx_signal, 'default', def_xl);
        setappdata(axx_signal, 'XLim', [0 def_xl]);
        setappdata(axx_signal, 'XLimMode', 'manual');
    %label
        [RR_interval, PR_interval, QRS_width] = examine(signal_data);
        text_RR = sprintf('R-R Interval : %d\n normal rate : 60-100 bpm\n\n', RR_interval);
        text_PR = sprintf('P-R Interval : %d\n normally 120-200 ms\n\n', PR_interval);
        text_QRS = sprintf('QRS Width : %d\n normally about 0.12 s\n', QRS_width);
        set(lab_property, 'Text', [text_RR text_PR text_QRS]); %print properties
    %etc
        set([but_searchfile, but_examine], 'Enable', 'off');
        set(but_restart, 'Enable', 'on');
    end
end
```

```
R-R Interval : 263
 normal rate : 60-100 bpm

P-R Interval : 37
 normally 120-200 ms

QRS Width : 90
 normally about 0.12 s
```



```
function xaxis_l(event)
    set(axx_signal, 'YLim', axx_signal.YLim);
    set(axx_signal, 'YLimMode', 'manual');

    sli_xl.Value = event.Value;
    new = sli_xl.Value;

    old_xl = axx_signal.XLim;
    xC = (old_xl(1) + old_xl(2)) /2 ;  %center pt of x axis

    def_xl = getappdata(axx_signal, 'default');
    new_xl = def_xl/(2*exp(new));
    setappdata(axx_signal, 'new', new_xl);

    if xC+new_xl>= def_xl
        set(axx_signal, 'XLim', [def_xl-2*new_xl def_xl]);
    elseif xC-new_xl <= 0
        set(axx_signal, 'XLim', [0 2*new_xl]);
    else
        set(axx_signal, 'XLim',[xC-new_xl xC+new_xl]);
    end
```

```
    if new ~= 0
        set(sli_xp, 'Enable', 'on');
        set(sli_xp, 'Limits', [0 def_xl - 2*new_xl]);
        if xC+new_xl >= def_xl
            set(sli_xp, 'Value', def_xl-2*new_xl);
        elseif xC-new_xl <= 0
            set(sli_xp, 'Value', 0);
        else
            set(sli_xp, 'Value', xC-new_xl);
        end
    else
        set(sli_xp, 'Enable', 'off');
    end
end
```



```
function xaxis_p (event)
    set(axx_signal, 'YLim', axx_signal.YLim);
    set(axx_signal, 'YLimMode', 'manual');

    sli_xp.Value = event.Value;
    new = sli_xp.Value;

    def_xl = getappdata(axx_signal, 'default');
    new_xl = getappdata(axx_signal, 'new');

    set(sli_xp, 'Limits', [0 def_xl-2*new_xl]);

    set(axx_signal, 'XLim', [new new+2*new_xl]);

end
```

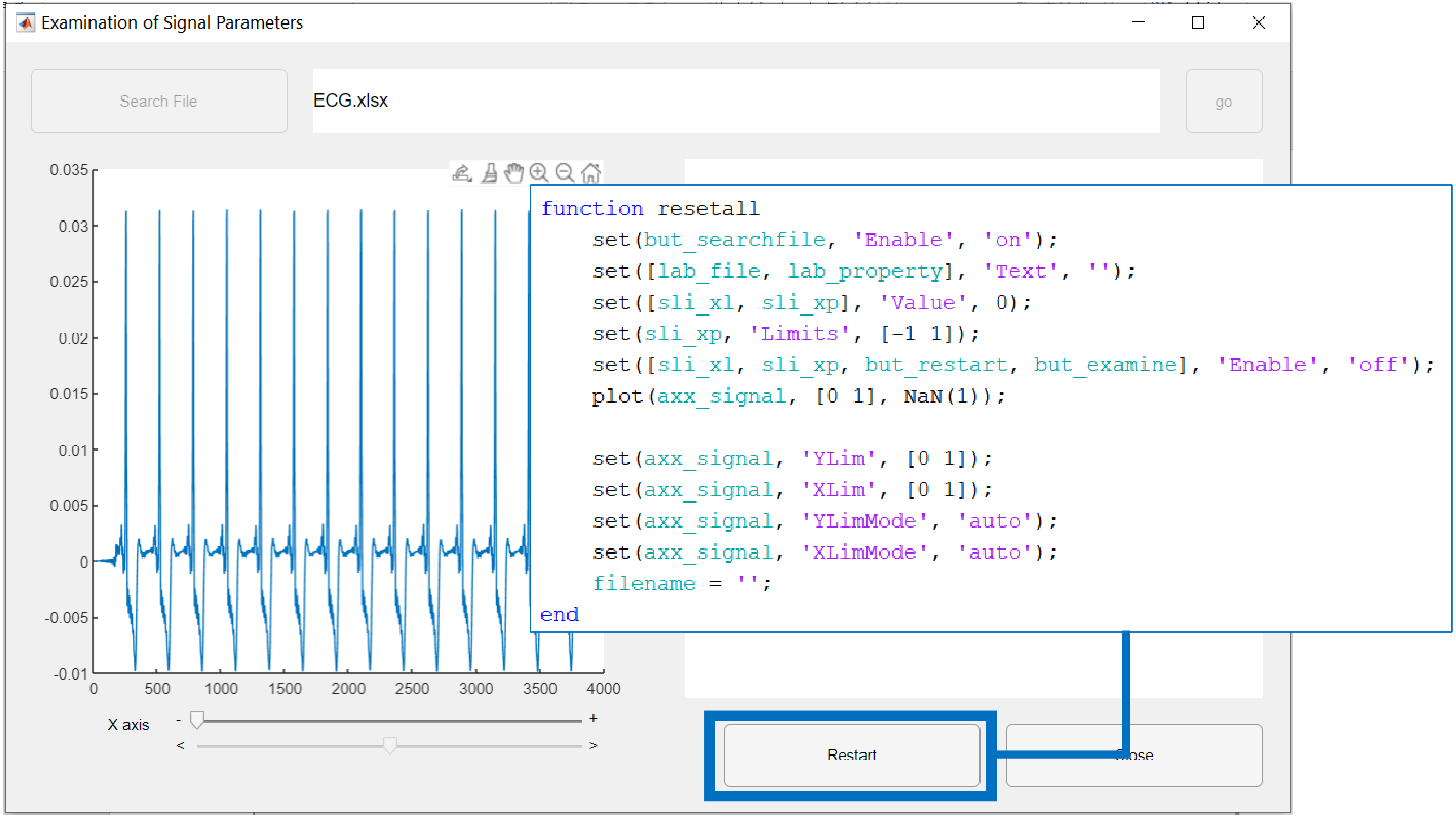
Examination of Signal Parameters

ECG.xlsx

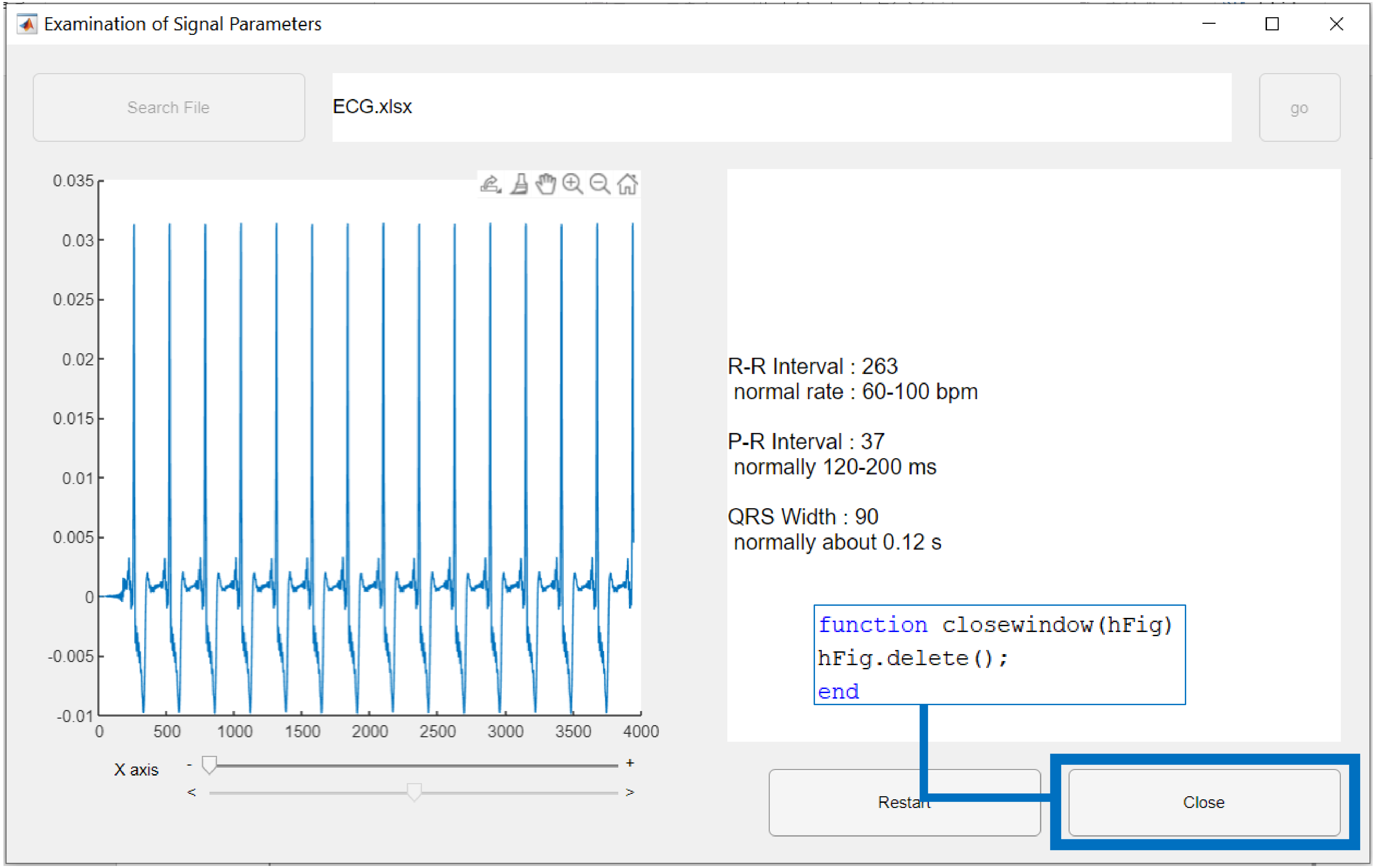
R-R Interval : 263
normal rate : 60-100 bpm

P-R Interval : 37
normally 120-200 ms

QRS Width : 90
normally about 0.12 s



```
function resetall
    set(but_searchfile, 'Enable', 'on');
    set([lab_file, lab_property], 'Text', '');
    set([sli_xl, sli_xp], 'Value', 0);
    set(sli_xp, 'Limits', [-1 1]);
    set([sli_xl, sli_xp, but_restart, but_examine], 'Enable', 'off');
    plot(axx_signal, [0 1], NaN(1));

    set(axx_signal, 'YLim', [0 1]);
    set(axx_signal, 'XLim', [0 1]);
    set(axx_signal, 'YLimMode', 'auto');
    set(axx_signal, 'XLimMode', 'auto');
    filename = '';
end
```



ECG.xlsx

R-R Interval : 263
 normal rate : 60-100 bpm

P-R Interval : 37
 normally 120-200 ms

QRS Width : 90
 normally about 0.12 s

```
function closewindow(hFig)
hFig.delete();
end
```