Breathing extracted from cardiac signals

Table of Contents

This code was developed by Miodrag Bolic for the book PERVASIVE CARDIAC AND RESPIRATORY MONITORING DEVICES: https://github.com/Health-Devices/CARDIAC-RESPIRATORY-MONITORING

We acknowledge using RRest toolbox provided by P. Charlton [1].


```
% Changing the path from main_folder to a particular chapter
main_path=fileparts(which('Main_Content.mlx'));
if ~isempty(main_path)
    %addpath(append(main_path,'/Chapter2'))
    cd (append(main_path,'/Chapter9/RespRatefromECG'))
    addpath(append(main_path,'/Service'))
end
SAVE_FLAG=0; % saving the figures in a file
```

Introduction

In this notebook we first generate artificial ECG and PPG signals. Their freqeuncy is 60 beat per minute. Then we add different modulations and noise. Breathing is simulated as a sinewave signal. Signals are generated based on [1].

Please note that only ECG signal is used for further processing. The same analysis can be done with the PPG signal too.

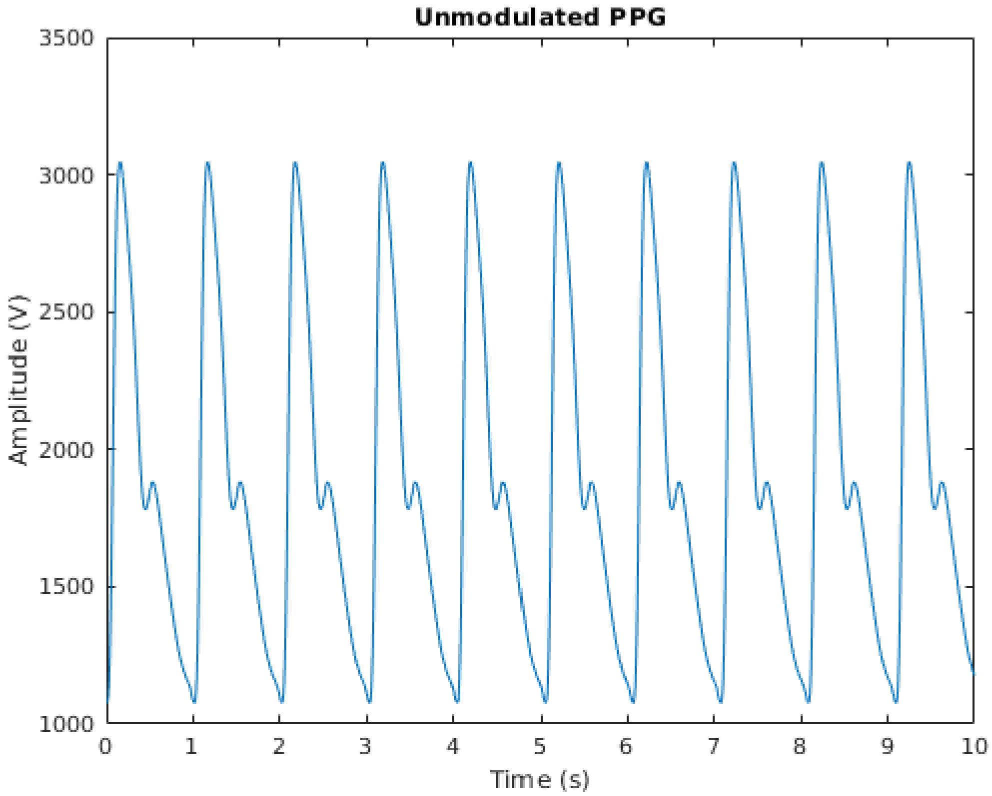
These steps were followed in order to determine the breathing rate:

1. Detecting fiducial points on the cardiac signal of interest (ECG or PPG)
2. Extracting the breathing signal
3. Estimating breathing rate

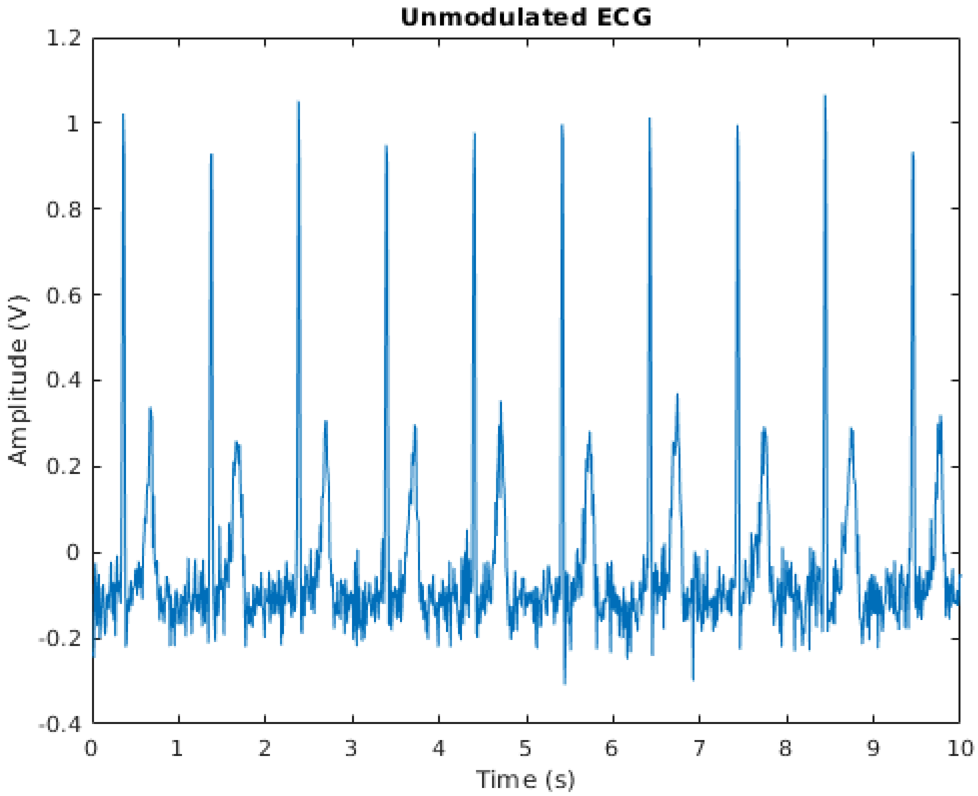
Of course, it is possible to extract the breathing signal automatically, for example by using Wavelet transform, without determining the fiducial points on the ECG signal.

Generating signals

```
fs=100;
%N=10000;
br_rate=0.15; %in Hz
br_rate_bpm=br_rate*60;
fm=0.05;
bm=0.1;
am=0.1;
% EPG beat data
ecg_data.beat.t = 0:0.01:1;
ecg_data.beat.v = [-0.120000000000000,-0.120000000000000,-0.125000000000000,-0.1225001

% PPG beat data
ppg_data.beat.t = 0:0.01:1;
ppg_data.beat.v = [1077,1095.27676998271,1150.39696200268,1253.64011384634,1410.014306

ecg_data.sig.v=repmat(ecg_data.beat.v,1,100);
N=length(ecg_data.sig.v);
ecg_data.sig.t=1/fs:1/fs:N/fs;
ppg_data.sig.t=1/fs:1/fs:N/fs;
ppg_data.sig.v=repmat(ppg_data.beat.v,1,100);

T=1/fs;
figure
plot(ppg_data.sig.t,ppg_data.sig.v);
    title('Unmodulated PPG')
    xlabel('Time (s)')
    ylabel('Amplitude (V)')
        xlim([0,10])
```



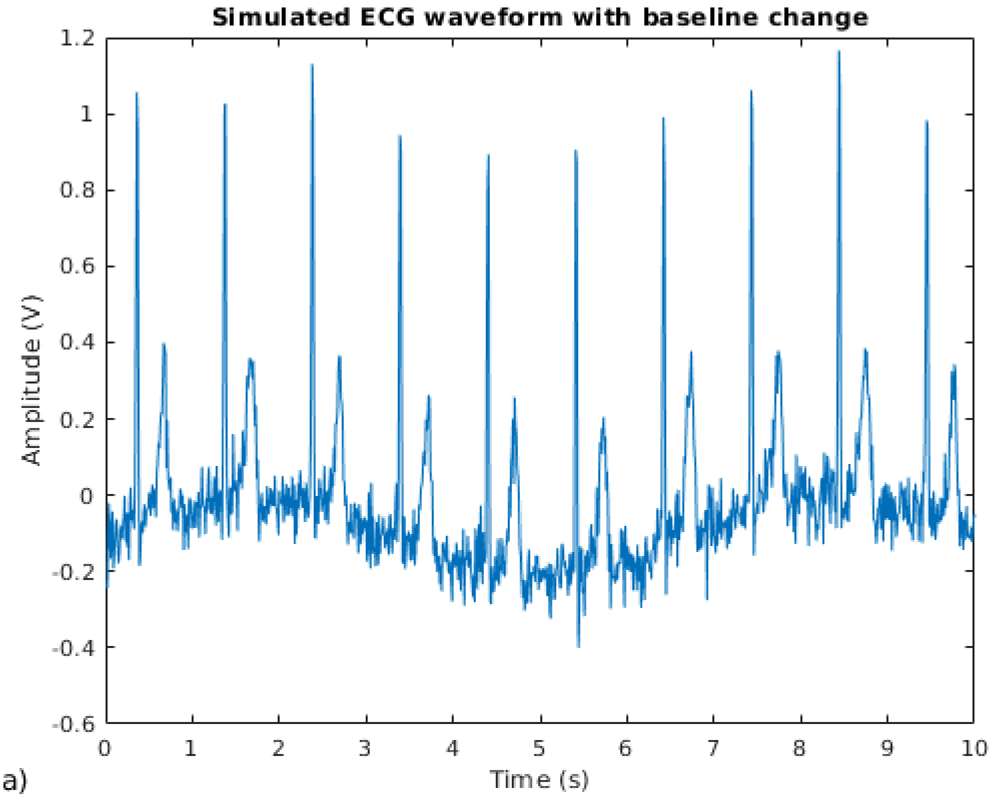

```
    ecg_data.sig.v=ecg_data.sig.v+0.05*randn(1,length(ecg_data.sig.v));
     figure
     plot(ecg_data.sig.t,ecg_data.sig.v);
     title('Unmodulated ECG')
     xlabel('Time (s)')
     ylabel('Amplitude (V)')
         xlim([0,10])
```



Simulating breathing modulation on the ECG signal

ECG BM

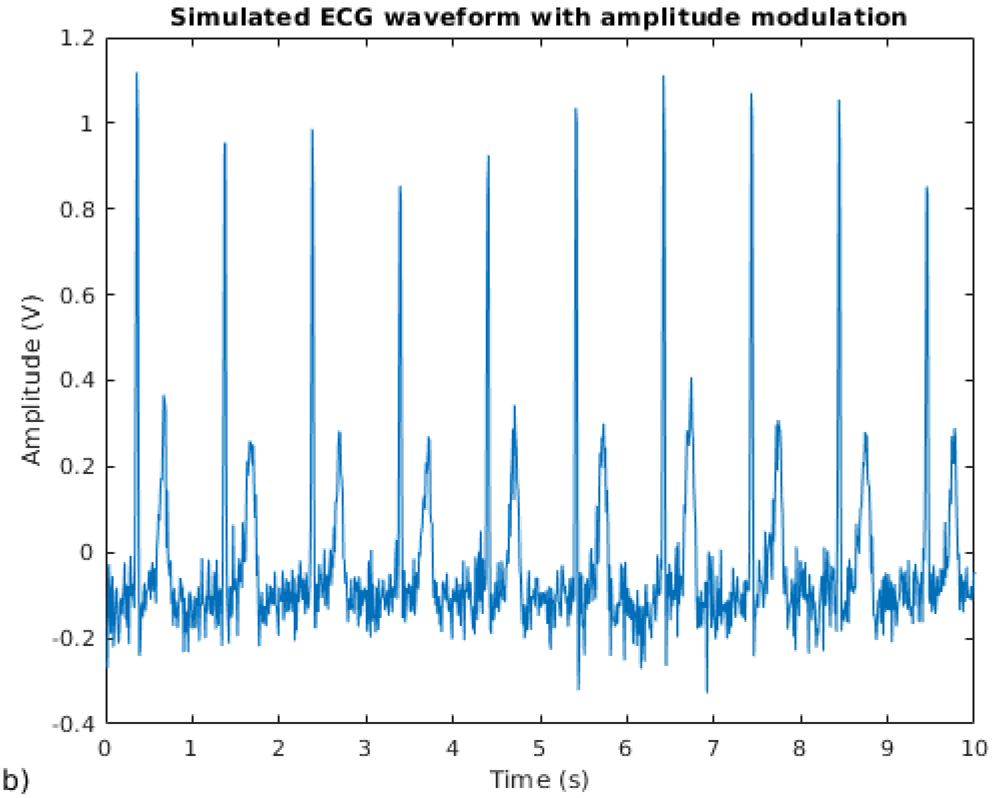
```
ecg_data.sig.v_bm=ecg_data.sig.v+bm*sin(2*pi*ecg_data.sig.t*br_rate);
figure
plot(ecg_data.sig.t,ecg_data.sig.v_bm)
title('Simulated ECG waveform with baseline change')
    xlabel('Time (s)')
    ylabel('Amplitude (V)')
    xlim([0,10])
    annonation_save('a)',"Fig9.6a.jpg", SAVE_FLAG);
```



a)

ECG AM

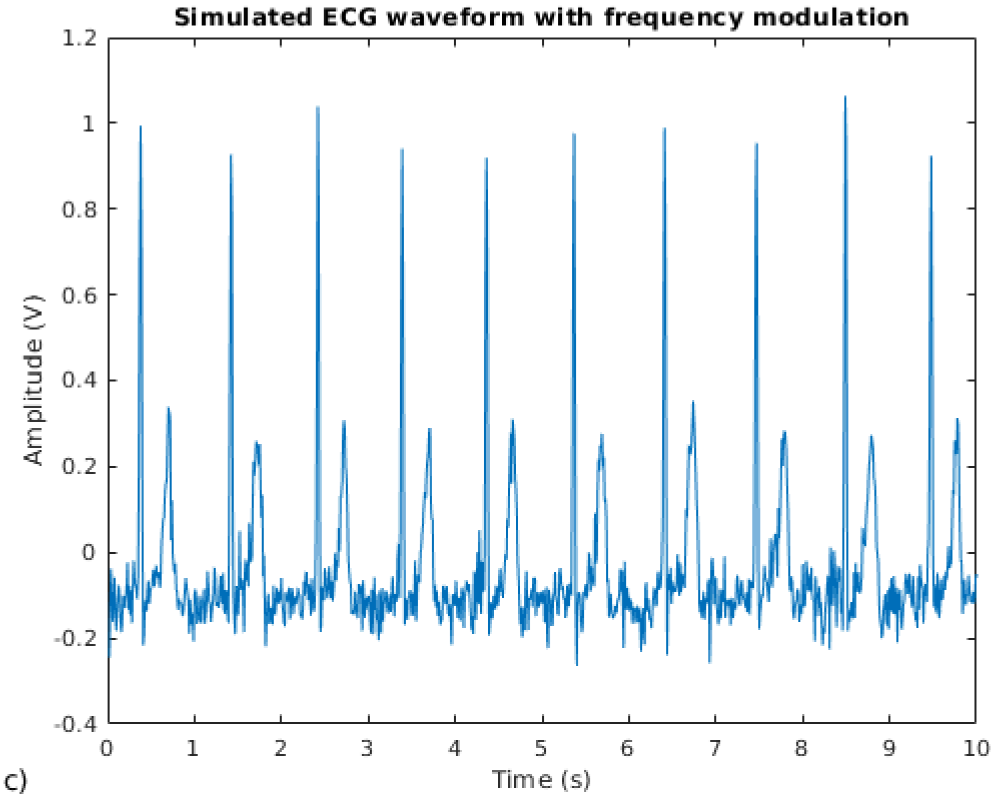
```
ecg_data.sig.v_am=ecg_data.sig.v.*(1+am*cos(2*pi*ecg_data.sig.t*br_rate));
figure
plot(ecg_data.sig.t,ecg_data.sig.v_am)
title('Simulated ECG waveform with amplitude modulation')
    xlabel('Time (s)')
    ylabel('Amplitude (V)')
    xlim([0,10])
    annonation_save('b)',"Fig9.6b.jpg", SAVE_FLAG);
```



b)

ECG FM

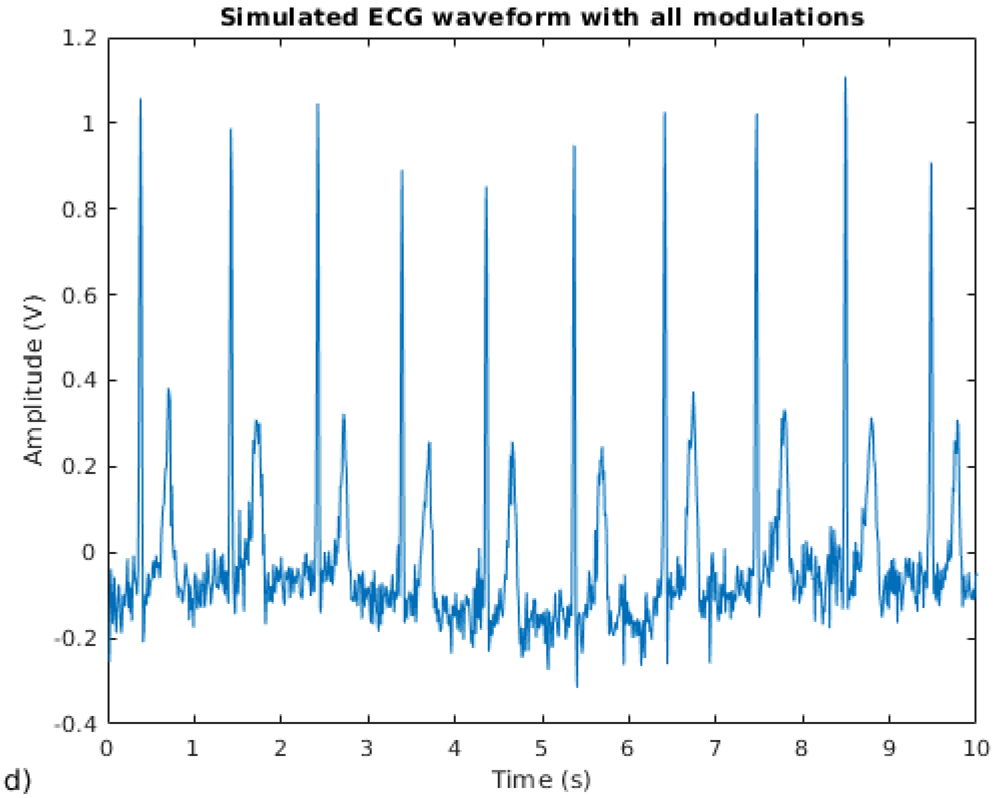
```
mod_time = ecg_data.sig.t + fm*sin(2*pi*br_rate*ecg_data.sig.t);
ecg_data.sig.v_fm = interp1(mod_time, ecg_data.sig.v, ecg_data.sig.t);

figure
plot(ecg_data.sig.t,ecg_data.sig.v_fm)
title('Simulated ECG waveform with frequency modulation')
    xlabel('Time (s)')
    ylabel('Amplitude (V)')
    xlim([0,10])
    annonation_save('c)',"Fig9.6c.jpg", SAVE_FLAG);
```



c)

ECG all modulations

```
ecg_data.sig.v_all=interp1(mod_time, (ecg_data.sig.v_bm+ecg_data.sig.v_am)/2, ecg_data.
figure
plot(ecg_data.sig.t,ecg_data.sig.v_all)
title('Simulated ECG waveform with all modulations')
    xlabel('Time (s)')
    ylabel('Amplitude (V)')
    xlim([0,10])
    annonation_save('d)',"Fig9.6d.jpg", SAVE_FLAG);
```



d)

Extracting the breathing signal

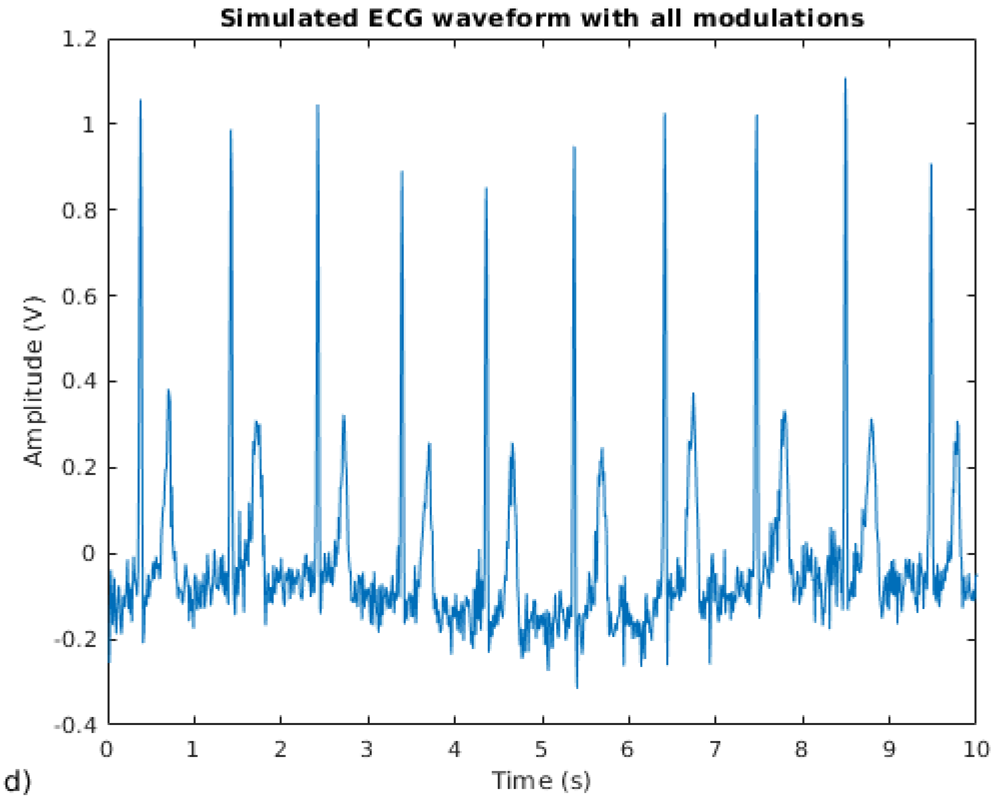
Finding R peaks

```
[R1,Q,S,T,P_w, Te, ecg1] = MTEO_qrst(ecg_data.sig.v_all(2:end),fs,0);

figure,plot(ecg_data.sig.t(2:end),ecg1);
hold on,scatter(R1(:,1)/fs,R1(:,2),'r');
hold on,scatter(Q(:,1)/fs,Q(:,2),'g');
hold on,scatter(S(:,1)/fs,S(:,2),'k');
hold on,scatter(T(:,1)/fs,T(:,2),'m');
hold on,scatter(P_w(:,1)/fs,P_w(:,2),'MarkerEdgeColor',[.7 .5 0]);
ylim([-0.35,1.25])
legend({'ECG signal','R wave','Q wave','S wave','T wave', 'P wave'},'NumColumns',2)
title('Delineated ECG signal')
xlabel('Time (s)')
ylabel('Amplitude (V)')
xlim([0 10])
```

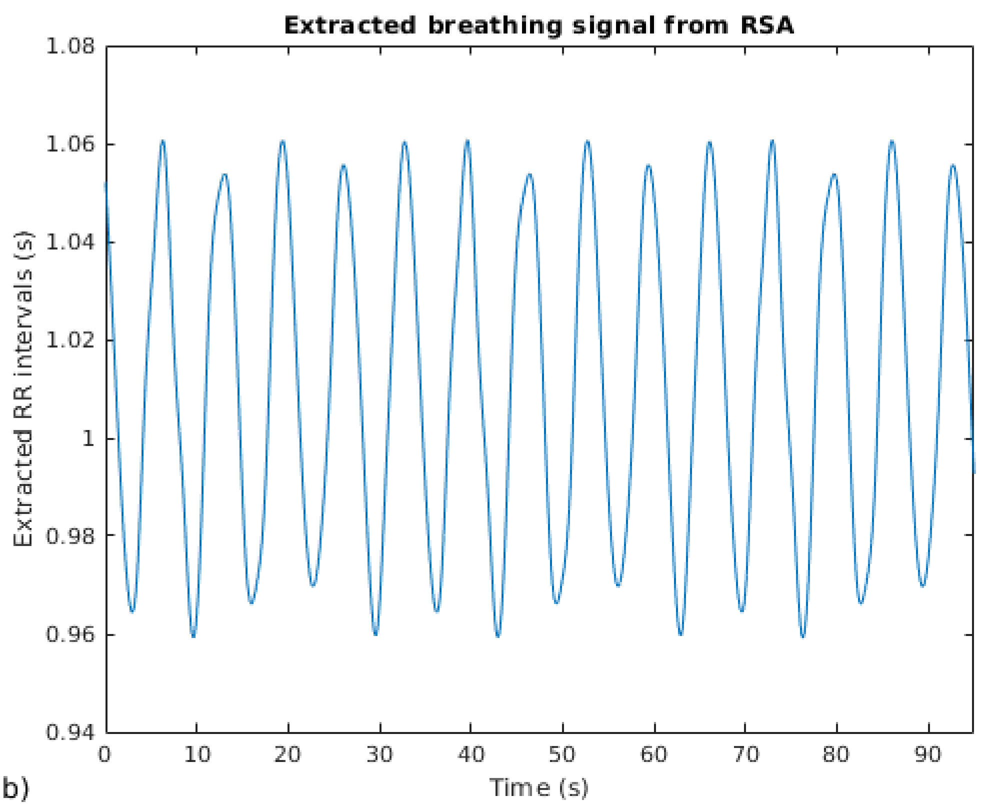
Extracting Respiratory Sinus Arrhythmia (RSA) Signal

```
RSA  = diff(R1(:,1));
n=1:length(ecg1);
RSA_interp = interp1(R1(1:end-1,1),RSA,n, 'spline');
figure
plot(n/fs,RSA_interp/fs)
title('Extracted breathing signal from RSA')
```

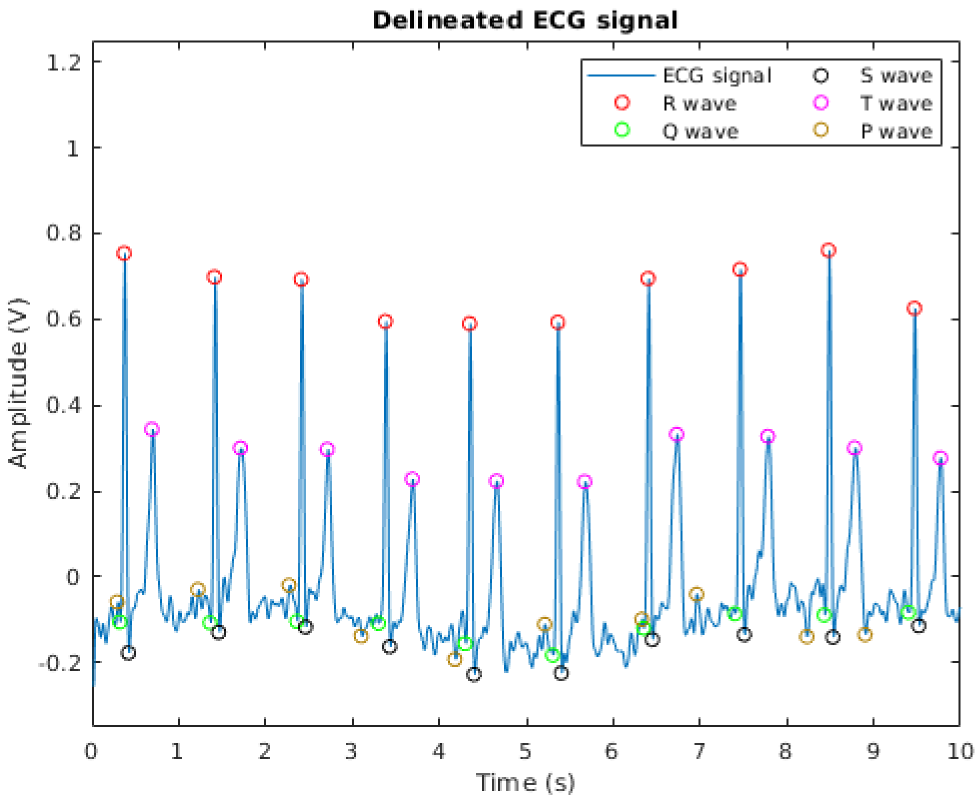
```
xlabel('Time (s)')
ylabel('Extracted RR intervals (s)')
xlim([0 95])
annonation_save('b)',"Fig9.7b.jpg", SAVE_FLAG);
```



d)

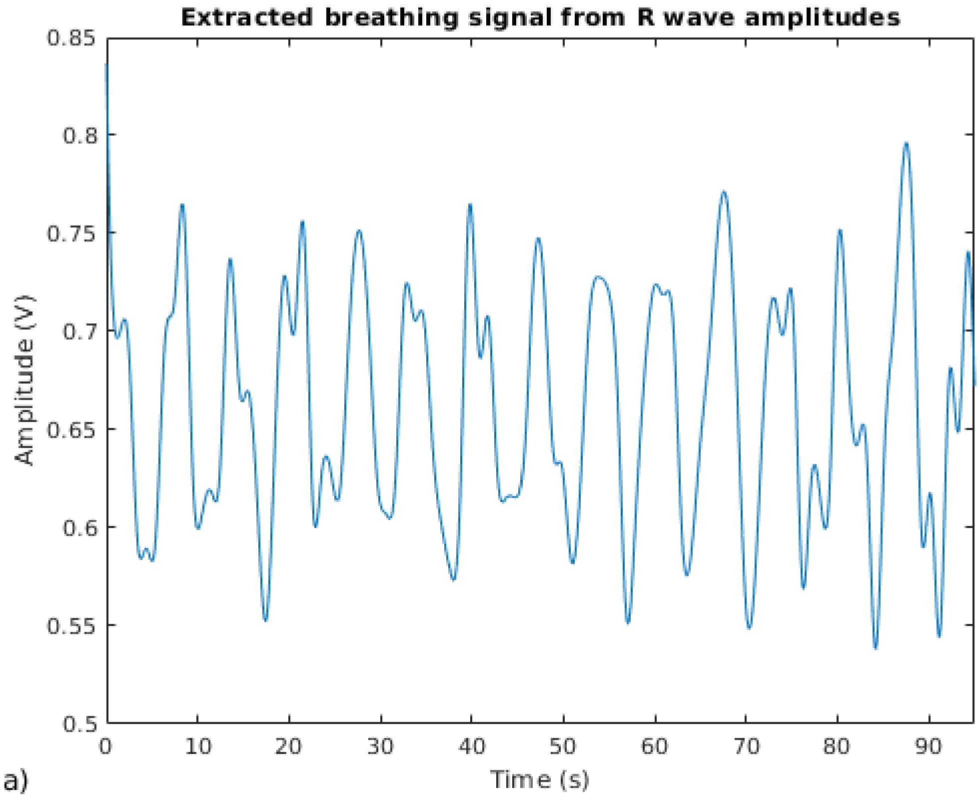


b)



Extracting Respiratory Signal based on R peak amplitude (RPA)

```
%%
RPA_interp = interp1(R1(:,1),R1(:,2),n, 'spline');
figure
plot(n/fs,RPA_interp)
title('Extracted breathing signal from R wave amplitudes')
xlabel('Time (s)')
ylabel('Amplitude (V)')
xlim([0 95])
annonation_save('a)',"Fig9.7a.jpg", SAVE_FLAG);
```



a)

Estimating breathing rate

```
[breathing_rate_peaks] = br_rate_est(RSA_interp',fs)
```

```
breathing_rate_peaks = 9.0493
```

```
[breathing_rate_peaks] = br_rate_est(RPA_interp',fs)
```

```
breathing_rate_peaks = 9.1919
```

```
function [breathing_rate_peaks] = br_rate_est(sig,fs)
[m1,j1]=findpeaks(rescale1(medfilt1(sig,5)), 'MinPeakProminence',0.1,'MinPeakDistance',
[m2,j2]=findpeaks(rescale1(-medfilt1(sig,5)), 'MinPeakProminence',0.1,'MinPeakDistance'
%breathing_rate_peaks=60*(0.5*(legth(j1)+legth(j2)))/segment_length;
if length(j1)>1 & length(j2)>1
     breathing_rate_peaks=2*(60*fs/(mean(diff(j1))+mean(diff(j2)))); %0.5*(length(j1)+l
elseif length(j1)>1
     breathing_rate_peaks=2*(60*fs/(mean(diff(j1)))) ;
elseif length(j2)>1
     breathing_rate_peaks=2*(60*fs/(mean(diff(j2)))) ;
else
     breathing_rate_peaks=0;
end
```

```
end
```

References

[1] Charlton, P. H., Bonnici, T., Tarassenko, L., Clifton, D. A., Beale, R., & Watkinson, P. J. (2016). An assessment of algorithms to estimate respiratory rate from the electrocardiogram and photoplethysmogram. Physiological Measurement, 37(4), 610–626. [DOI: 10.1088/0967-3334/37/4/610](http://doi.org/10.1088/0967-3334/37/4/610)