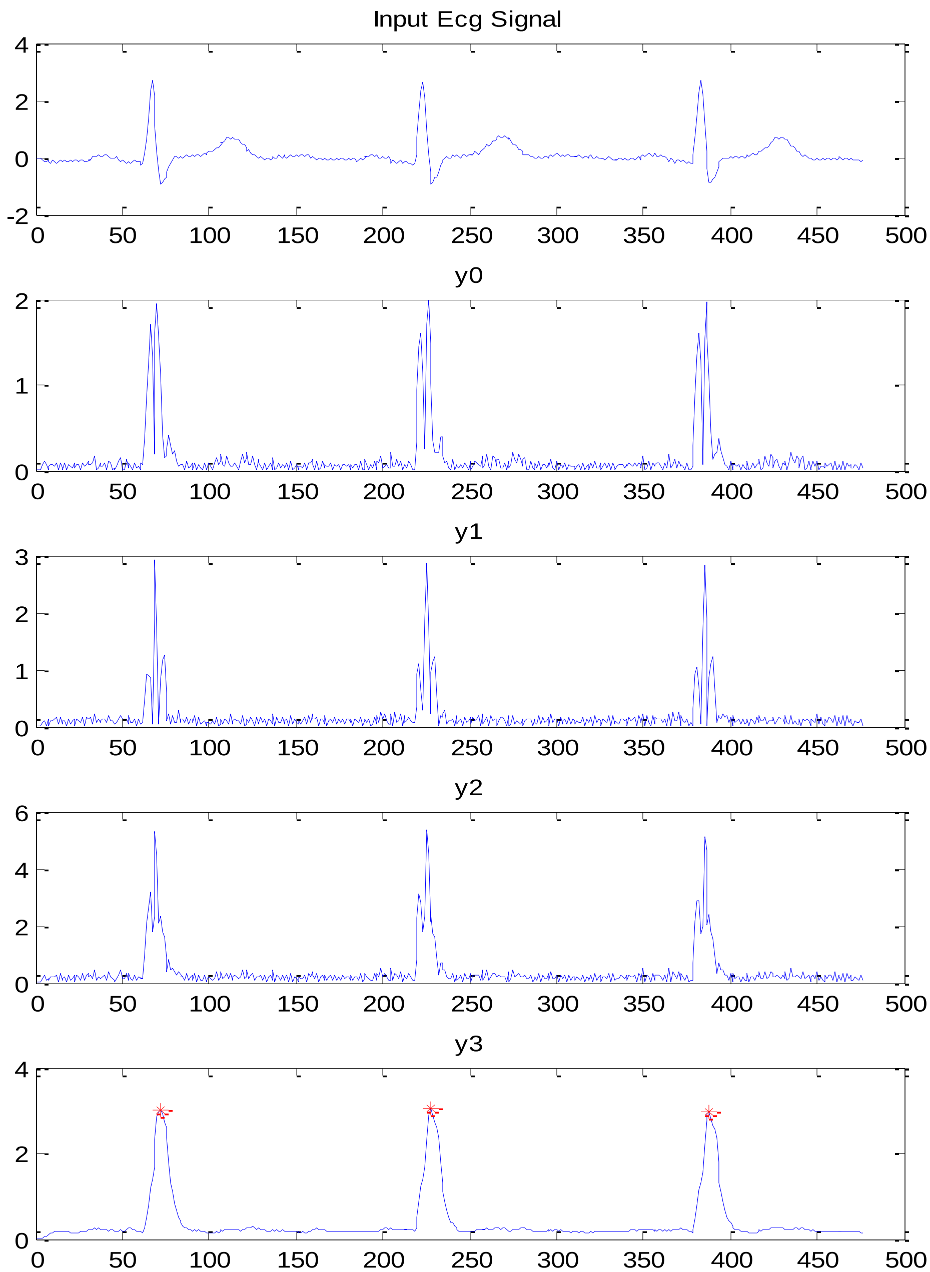
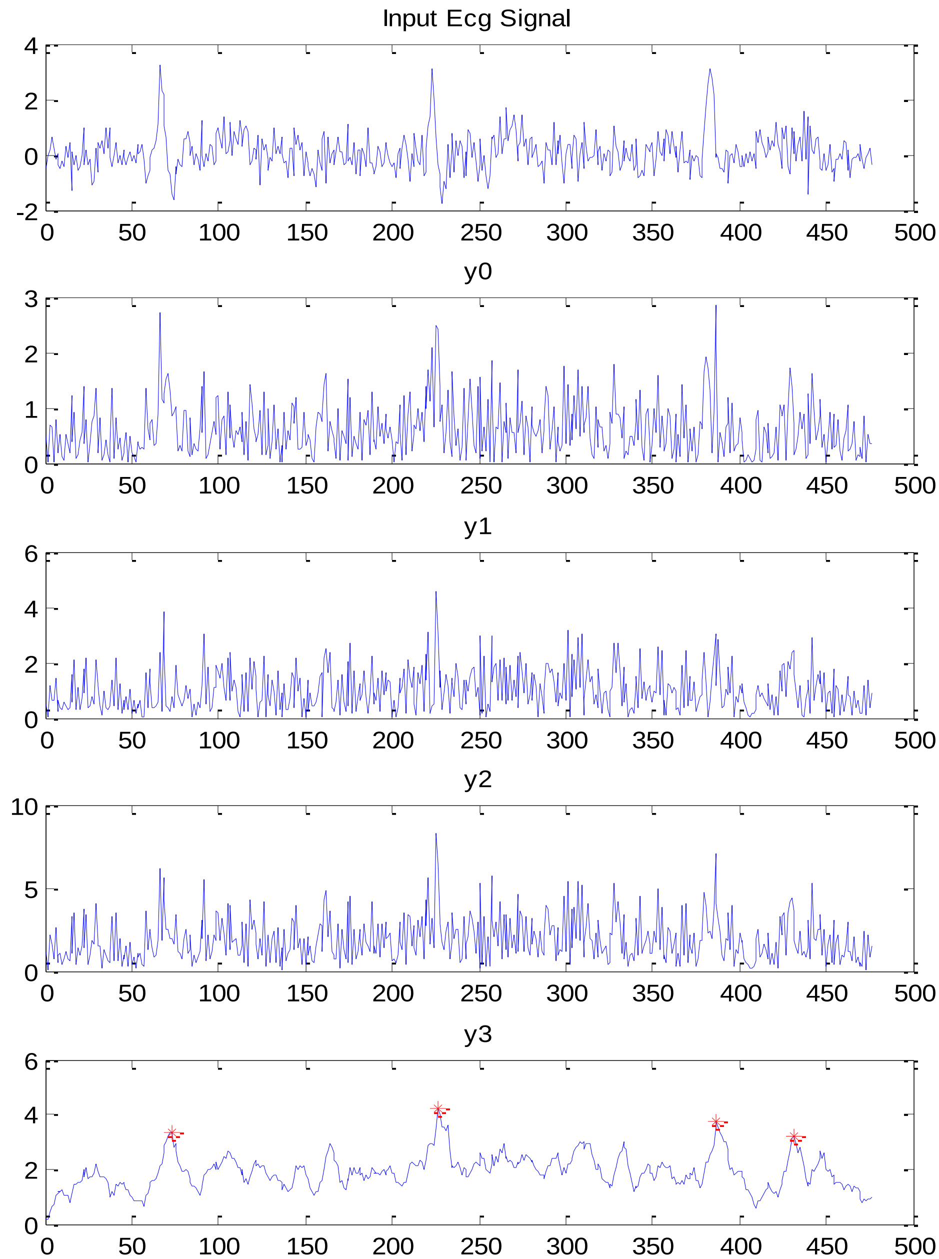
Jonathan Georgino – EE497A Biomedical Signal Analysis – Lab Report #9
Event Detection Lab – QRS Complex

## Exercise 1: Balda Method

1) x(n) is fairly clean



2) x(n) with white noise, x=signal + .5*randn(N,1);




Notice that there was an additional peak detected along with the 3 true QRS peaks. This is at x=signal + .5*randn(N,1); it can be seen that this method cannot stand a lot of noise. In 10 trials, only 3 trials detected the three QRS complex perfectly in the signal.

Code:

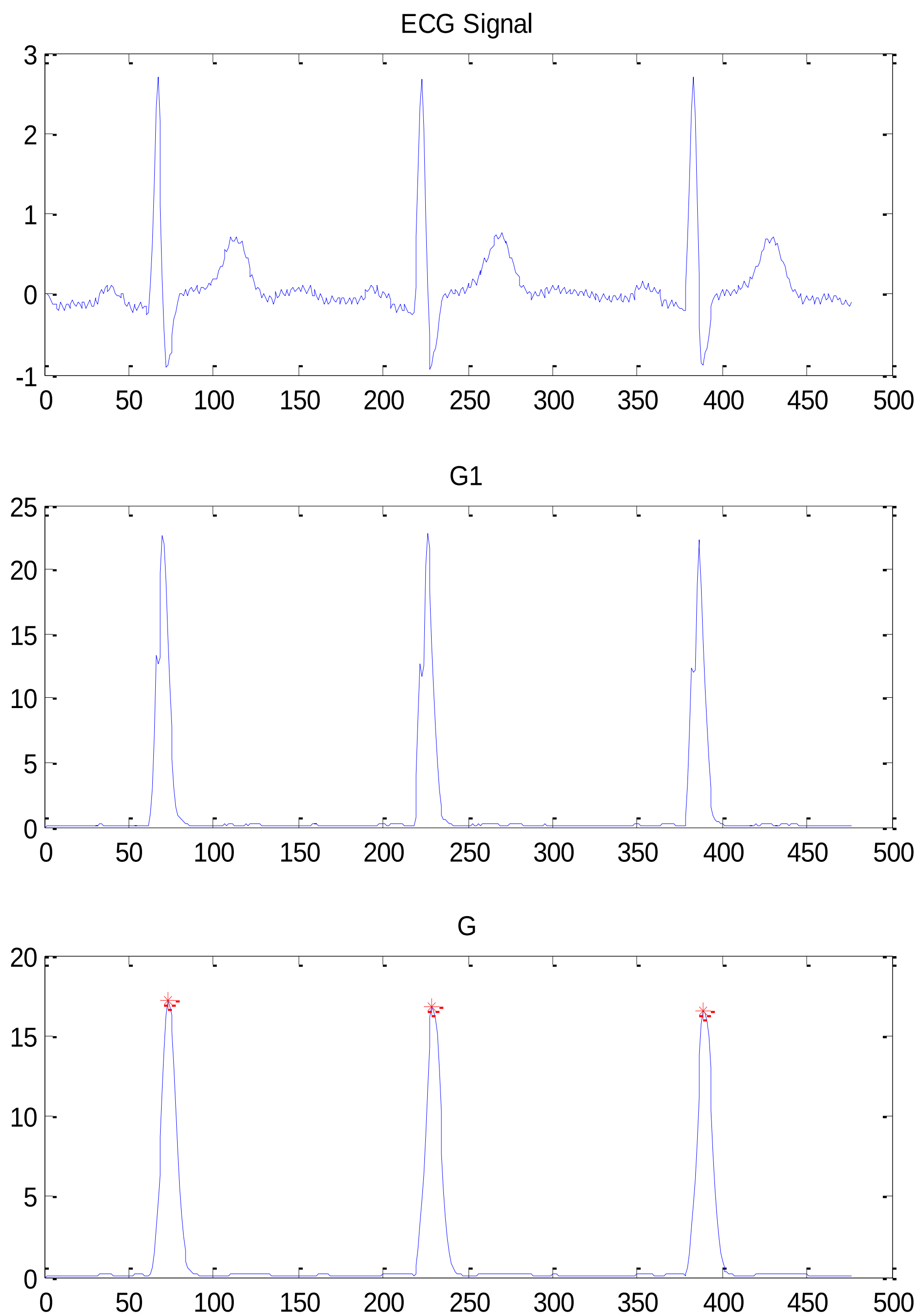
```
%Jonathan Georgino
%EE497A Biomedical Signal Analysis
%Event Detection Lab
clear all;
close all;
clc;

%load signal and determine length
x=load('ecg_60hz_200.dat');
N=length(x);

%add white noise
x=x+.5*randn(N,1);

%construct y0
y0(1)=abs(x(1));
y0(2)=abs(x(2));
for j = 3:N
    y0(j)=abs(x(j)-x(j-2));
end

%construct y1
y1(1)=abs(x(1));
y1(2)=abs(x(2));
y1(3)=abs(x(3)-2*x(3-2));
y1(4)=abs(x(4)-2*x(4-2));

for j = 5:N
    y1(j)=abs(x(j)-2*x(j-2)+x(j-4));
end

%construct y2
for j=1:N
    y2(j)=1.3*y0(j)+1.1*y1(j);
end

%construct y3
p=0;
for j=1:7
    p=y2(j)+p;
   y3(j)=(1/8)*p;
end

for j=8:N
p=0;

    for k=0:7
    p=y2(j-k)+p;
    end

y3(j)= (1/8)*p;
end
```

```
%peak detect
sensitivity = 2;
%calculate peaks based on peakdet.m function
[maxtab, mintab] = peakdet(y3, sensitivity);

%plot
subplot(511)
plot(x);
title('Input Ecg Signal');
subplot(512)
plot(y0);
title('y0');
subplot(513)
plot(y1);
title('y1');
subplot(514)
plot(y2);
title('y2');
subplot(515)
plot(y3);
title('y3');
hold on;
plot(maxtab(:,1), maxtab(:,2), 'r*');
```

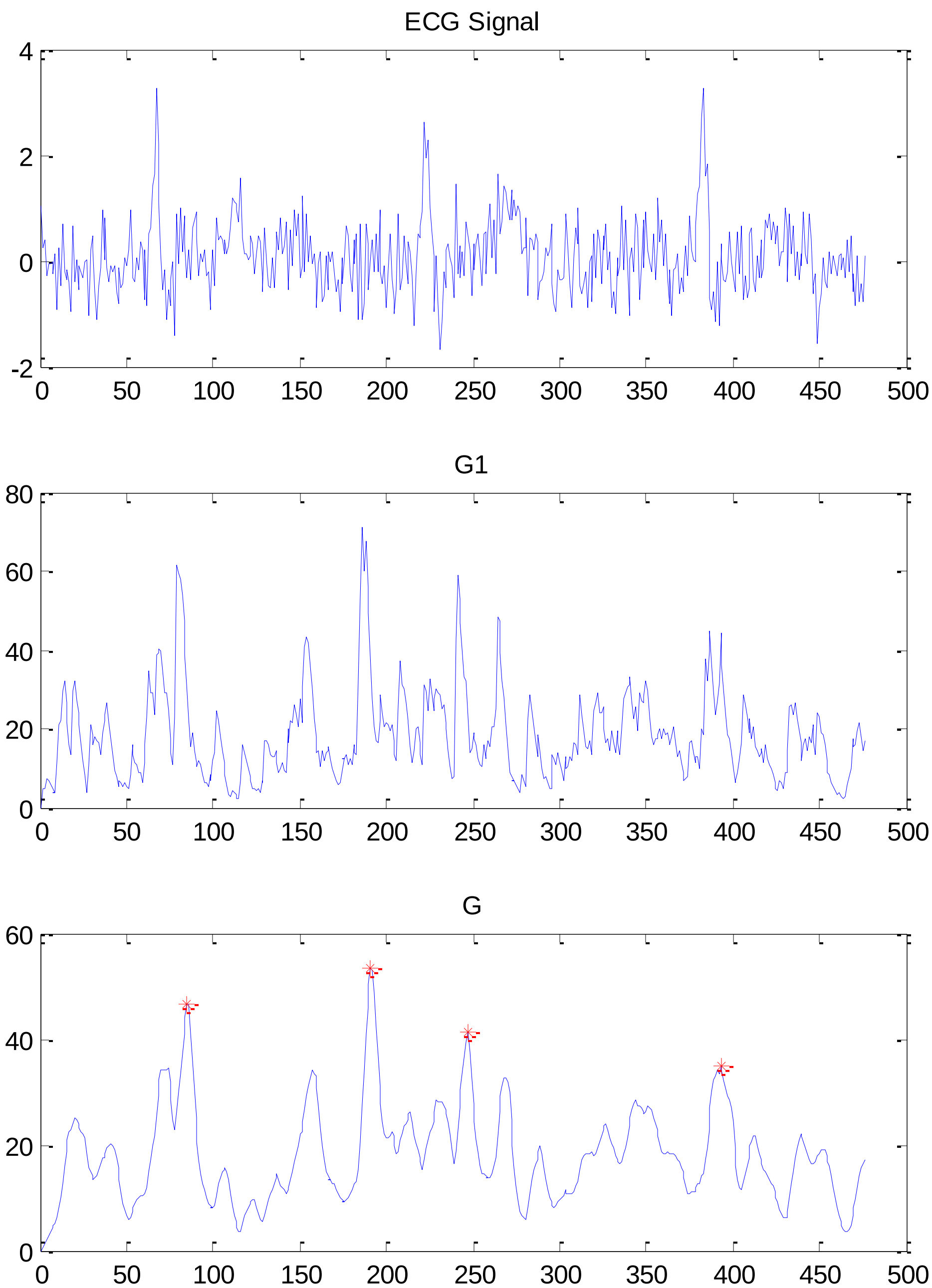
## Exercise 2: Murthy and Rangaraj Method

**I. N=M=8**

1) x(n) is fairly clean

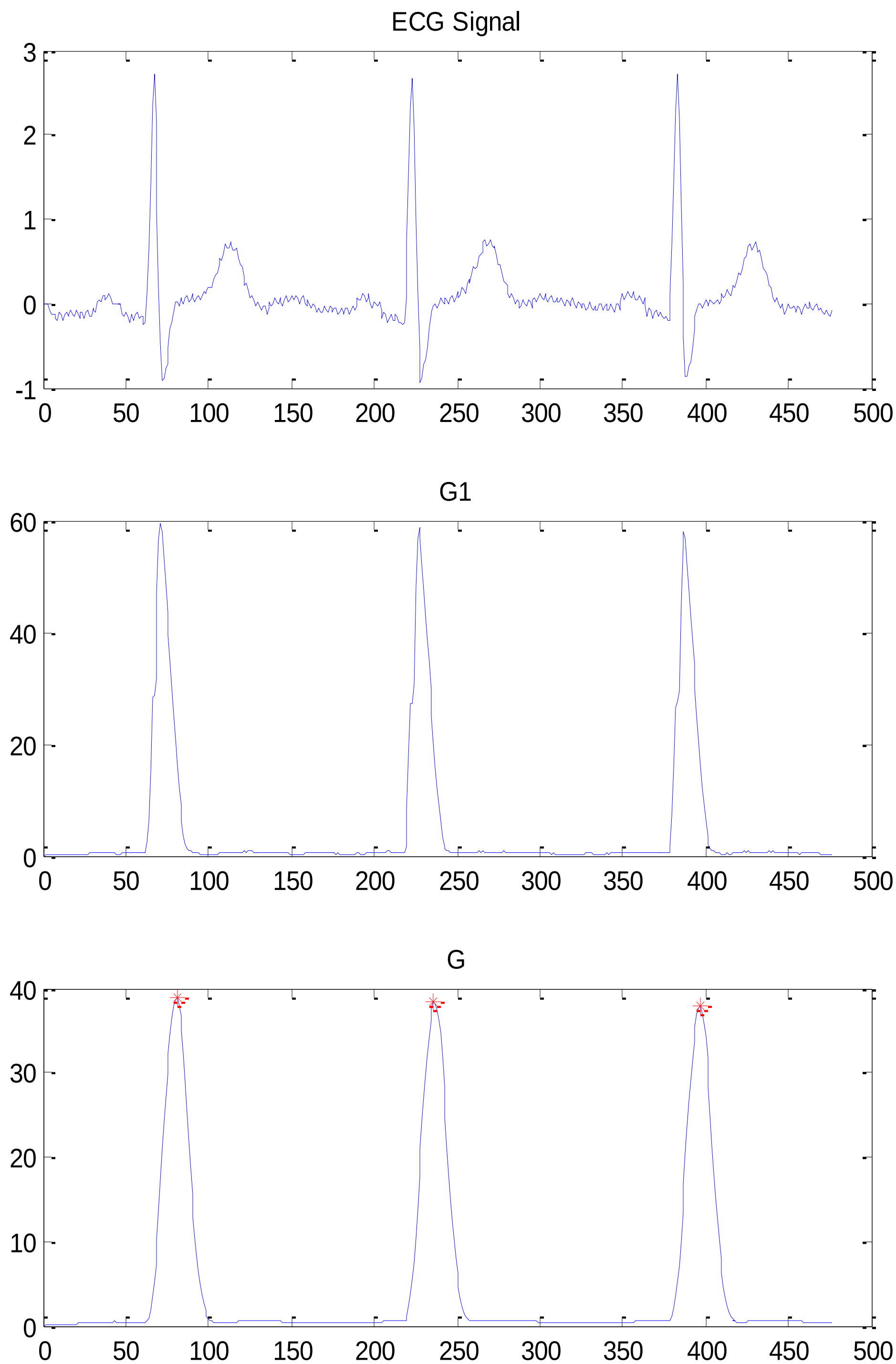


2) x(n) with white noise, x=signal + .5*randn(N,1);

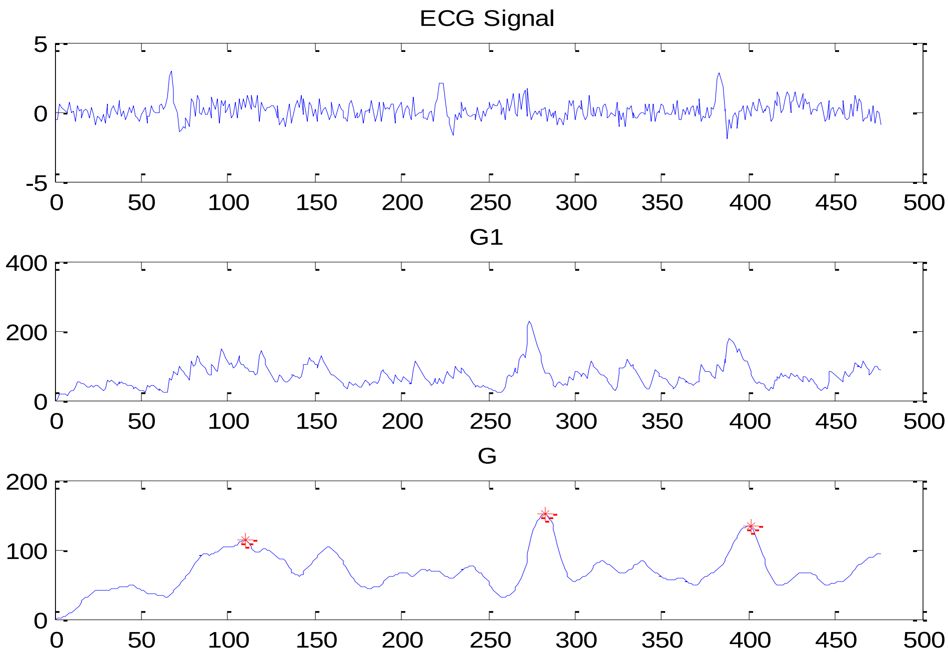


**II. N=M=16**

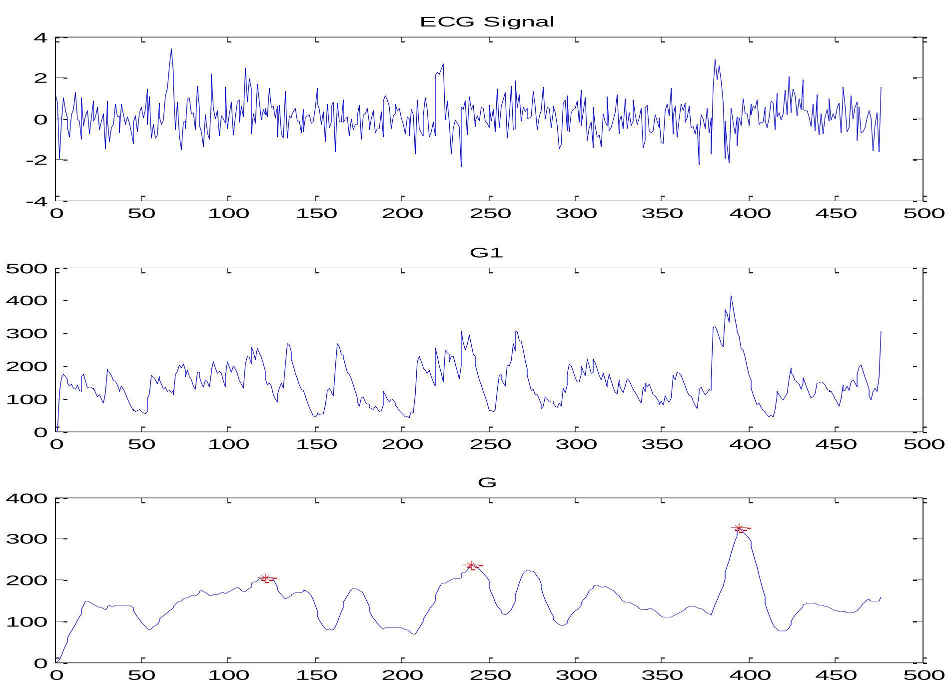
1) x(n) is fairly clean



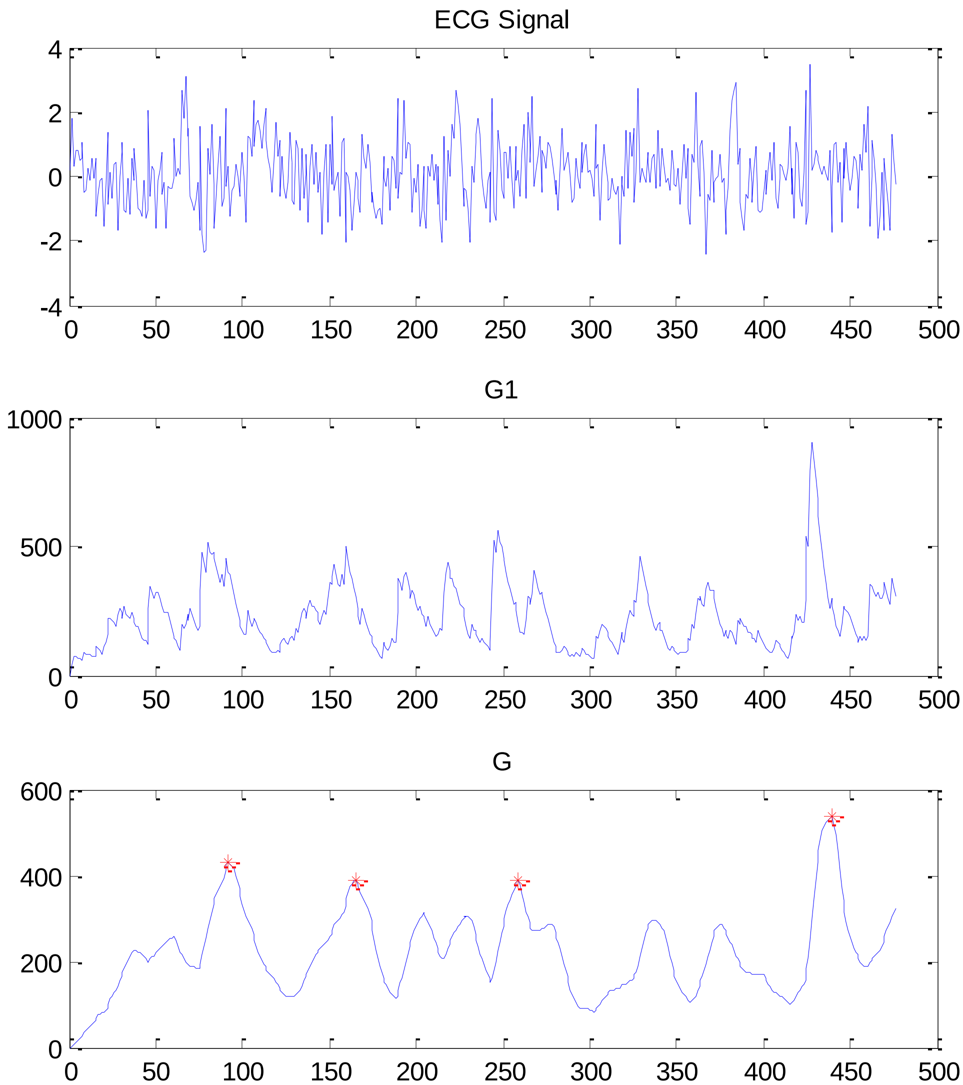
2) x(n) with white noise, x=signal + .5*randn(N,1);




3) x(n) with white noise, x=signal + .75*randn(N,1);



4) x(n) with white noise, x=signal + 1*randn(N,1);



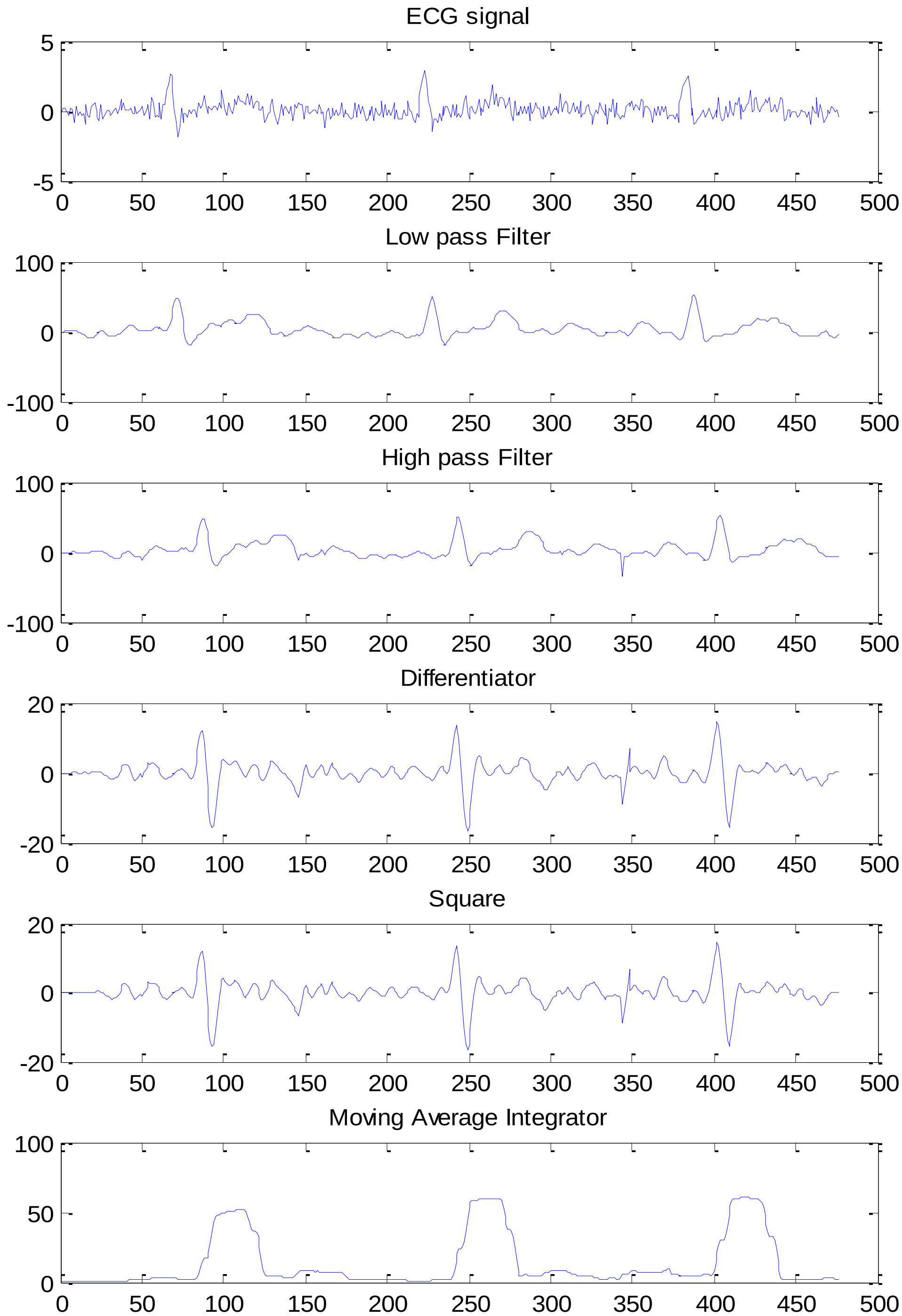

Notice that there was an additional peak detected along with the 3 true QRS peaks. This is at x=signal + 1*randn(N,1); it can be seen that this method can stand a lot more noise than the Baldan method.  This method also works much better when N=M=16 than when it's N=M=8.

**Code:**

```
%Jonathan Georgino
%EE497A Biomedical Signal Analysis
%Event Detection Lab
clear all;
close all;
clc;

%load signal and determine length
x=load('ecg_60hz_200.dat');
len=length(x);

%add white noise
x=x+1*randn(len,1);

N = 16;
M = 16;

%calculate g1
for j=1:N+1

    p=0;
    for k=1:j-1
    p=(abs(x(j-k+1)-x(j-k))^2)*(N-k+1) +p;
    end

    g1(j)=p;

end

for j=N+1:len

    p=0;
    for k=1:N
    p=(abs(x(j-k+1)-x(j-k))^2)*(N-k+1) +p;
    end

    g1(j)=p;

end

%calculate g

for j=1:M-1

    p=0;
    for k=0:j-1
    p=g1(j-k)+p;
    end

    g(j)=(1/M)*p;
end
```

```
for j=M:len

    p=0;
    for k=0:M-1
    p=g1(j-k)+p;
    end

    g(j)=(1/M)*p;
end

%peak detect
sensitivity = 220;
%calculate peaks based on peakdet.m function
[maxtab, mintab] = peakdet(g, sensitivity);

subplot(311)
plot(x)
title('ECG Signal')
subplot(312)
plot(g1)
title('G1')
subplot(313)
plot(g)
title('G')
hold on;
plot(maxtab(:,1), maxtab(:,2), 'r*');
```

**Exercise 3: The Pan Tompkins Algorithm**



**Code:**

```
%Jonathan Georgino
%EE497A Biomedical Signal Analysis
%Event Detection Lab
clear all;
close all;
clc;

%load signal and determine length
x=load('ecg_60hz_200.dat');
len=length(x);

%add white noise
x=x+.45*randn(len,1);

%low pass filter
alp = [1 -2 1];
blp = [1 0 0 0 0 0 -2 0 0 0 0 0 1];
hlp = filter(blp, alp, x);


%high pass filter
hhp(1)=0-(1/32)*(1-0)/(1-0);

for j=2:16
    hhp(j)=0-(1/32)*(1-0)/(1-hlp(j-1));
end

for j=17:32
    hhp(j)=hlp(j-16)-(1/32)*(1-0)/(1-hlp(j-1));
end

for j=33:len
   hhp(j)=hlp(j-16)-(1/32)*(1-hlp(j-32))/(1-hlp(j-1));
end

%differentiator
hd(1)=(1/8)*(2*hhp(1)+0-0-0);
hd(2)=(1/8)*(2*hhp(2)+hhp(1));
hd(3)=(1/8)*(2*hhp(3)+hhp(2));
hd(4)=(1/8)*(2*hhp(4)+hhp(3)-hhp(1));
for j=5:len
    hd(j)=(1/8)*(2*hhp(j)+hhp(j-1)-hhp(j-3)-2*hhp(j-4));
end

%squarer
hsq=hd.^2;

%moving average integrator
N=30;

for j=1:N-1
p=0;
```

```
    for k=0:j-1
    p=hsq(j-k)+p;
    end
hmai(j)=(1/N)*p;
end

for j=N:len
p=0;
for k=0:N-1
p=hsq(j-k) + p;
end
hmai(j)=(1/N)*p;
end

subplot(611)
plot(x)
title('ECG signal')
subplot(612)
plot(hlp)
title('Low pass Filter')
subplot(613)
plot(hhp)
title('High pass Filter')
subplot(614)
plot(hd)
title('Differentiator')
subplot(615)
plot(hd)
title('Square')
subplot(616)
plot(hmai)
title('Moving Average Integrator')
```