False Detection Rates

A false detection, with respect to the Acoustic Awareness Enabler is when an interrupt is detected when there is no physical reality that would suggest that an interrupt had occurred. A false detection may be frustrating for the user of the Acoustic Awareness Enabler as this means that their audio was turned off for no good reason. In order to assess the performance of the Acoustic Awareness Enabler, a demonstration of a sufficiently low (less than 1%) false detection rate is needed, as per engineering requirement (XYZ). The following information is needed to determine a false detection rate (FDR).

Title header

Variable	Name	Description
Environment Data (Ambient)	n/a	Collection of data points. Running intake information to the AAE.
2 second average	2secAve	The most recent 2 seconds of data (typically 200 data points) are averaged to determine an instantaneous sound level.
20 Seconds Averages Array	Data20sec	The 20-second averages array is typically made of 10 numbers, which are 2-second averages of the most recent 20 seconds of data.
20 Second Average	20secAve	The average sound level from the last 20 seconds of data.
20 second Standard Deviation	20secStd	The standard deviation of the environment data is calculated.
Threshold Setting	T	The threshold setting is High, medium or low as selected by the user. The threshold setting is not a sound level, but a is the number of standard deviations from an ambient average that will theoretically trigger an interrupt.
Running Threshold	RT	The running threshold is the instantaneous sound level required for an interrupt to be detected. This value is updated after each new 2-second average. The running threshold is calculated using the following formula: $RT = 20secAve + 20secStd * T$
Interrupt detected	ID	A detected interrupt means that the running threshold has been exceeded by the instantaneous 2 second average. $ID = \begin{cases} 1 & 2secAve > RT \\ 0 & 2secAve < RT \end{cases}$
Real Interrupt	RI	A real interrupt is an event that may be detected. This is the case in which the source of an interrupt is worthy of being detected. The value of RI is determined by previous knowledge of the event, which was acquired through data collection. If the

		interrupt is real, RI=1. If the interrupt is not real, RI=0.
True Interrupt Detection	TID	A true interrupt detection, TID happens when an interrupt is detected that, through collected data is found to be a verified real interrupt. $TID = \begin{cases} 1 & ID = RI = 1 \\ 0 & otherwise \end{cases}$
False Detection	FD	False detection is an event where an interrupt is detected, while the interrupt is unfounded. $FD = \begin{cases} 1, & ID = 1\,, RI = 0 \\ 0, & otherwise \end{cases}$
False Negative	FN	A false negative means that an interrupt was not detected, despite the existence of a real interrupt. $FN = \begin{cases} 1, & ID = 0, RI = 1 \\ 0, & otherwise \end{cases}$
False Detection Rate	FDR	The false detection rate is the probability of the event $RI = 0$, given that $ID = 1$. $\boldsymbol{FDR = Pr[(RI = 0)\|(ID = 1)]}$

In order to calculate a false detection rate, a data set is required that possesses both real interrupts and interrupt detection. Data samples were collected for six environments containing samples for all possible events. Using data from these environments, false detection rates may be calculated. One parameter that the designers of the Acoustic Awareness Enabler may easily manipulate is the Threshold setting (T) values. There exist three values that are used for the threshold setting, depending on the user-selected sensitivity setting of high, medium or low.

$$T = \begin{cases} 4.00, & High \\ 8.00, & Medium \\ 12.00, & Low \end{cases}$$

As shown next, the false detection rate (FDR) is a function of T. The false detection rate should be as low as possible, though it is an engineering requirement that it be below 1%. Since T is an array of three integers, there will also exist three different false detection rates.

$$FDR = Pr[(RI = 0)|(ID = 1)] = Pr[(RI = 0)|\big(2secAve > (20secAve + 20secStd * T)\big)]$$

Senior Design Lab

```
%ECE458 - Senior Design
%Michael Benker
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% FALSE DETECTION RATES %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clf;clear all; clc; close all;

%VARIABLES
Ave2sec = 0;    %2-second average
T_high = 4;     %standard deviations for high sensitivity setting
T_med = 8;      %standard deviations for medium sensitivity setting
T_low = 12;     %standard deviations for low sensitivity setting
Data20sec = zeros(10,1);
history = zeros(100,1);

RI_med = zeros(100,1); %1 if real interrupt, 0 if false interrupt
RI_high = zeros(100,1);
RI_low = zeros(100,1);
ID_low = zeros(100,1);   %1 if interrupt detected, 0 if interrupt not
 detected
ID_med = zeros(100,1);
ID_high = zeros(100,1);
TID_low = zeros(100,1);    %1 if ID = RI = 1, 0 otherwise
TID_med = zeros(100,1);
TID_high = zeros(100,1);
FD_low = zeros(100,1);     %1 if ID = 1 & RI = 0, 0 otherwise
FD_med = zeros(100,1);
FD_high = zeros(100,1);
FN_low = zeros(100,1);     %1 if ID = 0 & RI = 1, 0 otherwise
FN_med = zeros(100,1);
FN_high = zeros(100,1);
FDR_low = 0;    %probability of false interrupt given interrupt
 detection
FDR_med = 0;
FDR_high = 0;

%IMPORT DATA
SoundData1 = 'Book1.xlsx'; %Read excel file in folder
DataMat = zeros(30,4); %Predefine Data Matrix
Ambients = xlsread(SoundData1, 'A2:A31'); %Ambient 1st col
Quiets = xlsread(SoundData1, 'B2:B31'); %Quiet is 2nd col
Mediums = xlsread(SoundData1, 'C2:C31'); %Medium is 3rd col
Louds = xlsread(SoundData1, 'D2:D31'); %Loud is 4th col
[k,DataLoc] = xlsread(SoundData1, 'E1:E1');
DataLoc
%Define past 20 seconds (ambients)

for c =1:10
    Data20sec(c,1) = Ambients(randi([1 30],1,1),1);
    history(c,1)=Data20sec(c,1);
end
Ave20sec = mean(Data20sec)
```

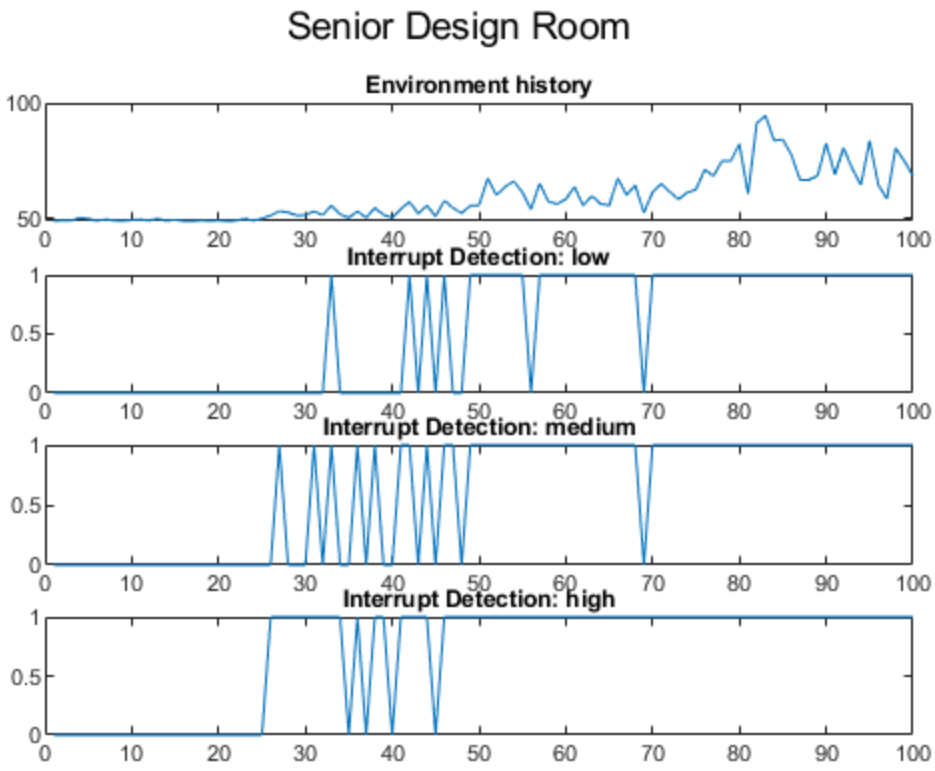
```
Std20sec = std(Data20sec)
RT_high = Ave20sec+Std20sec*T_high %Running threshold level (high
 sens)
RT_med = Ave20sec+Std20sec*T_med %Running threshold level (med sens)
RT_low = Ave20sec+Std20sec*T_low %Running threshold level (low sens)

for c=11:25
    new = Ambients(randi([1 30],1,1),1);
    history(c,1)=new;

    if new>RT_high
            ID_high(c,1)=1;
            FD_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%Quiet interrupts
%Only high sensitivity should activate interrupt
for c=26:50
    new = Quiets(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%medium interrupts
%Only high and medium sensitivity should activate interrupt

for c=51:75
    new = Mediums(randi([1 30],1,1),1);
    history(c,1)=new;
```

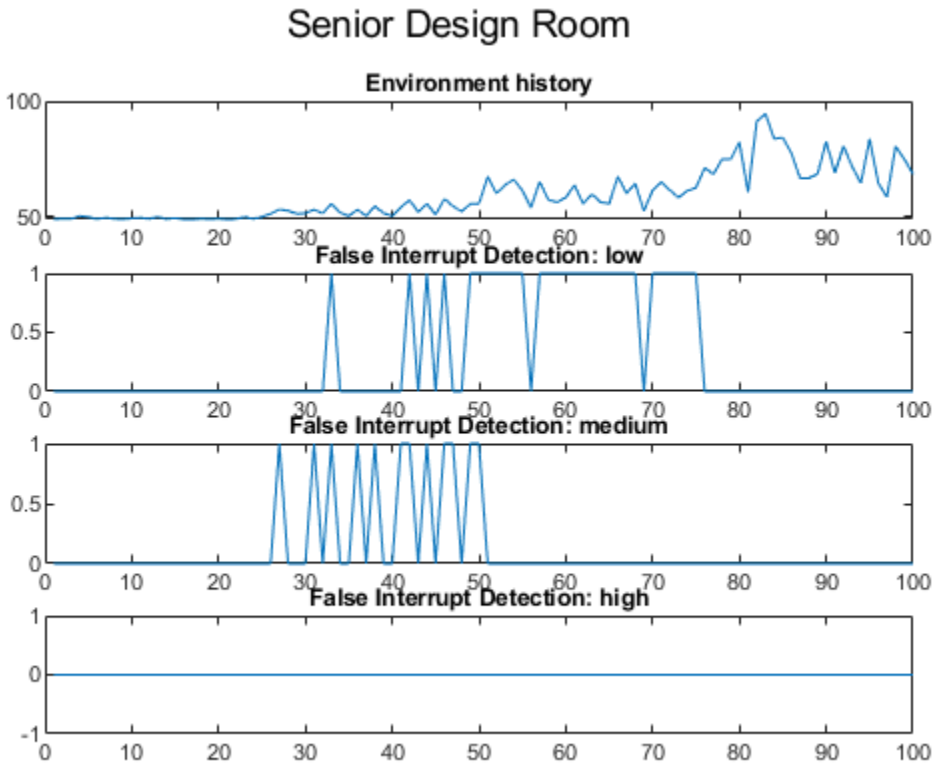
```
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end



%loud interrupts
%all activate interrupt
for c=76:100
    new = Louds(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    RI_low(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            TID_low(c,1)=1;
    else
        FN_low(c,1)=1;
    end

end

%False Interrupt Detection Rate - Print all
```

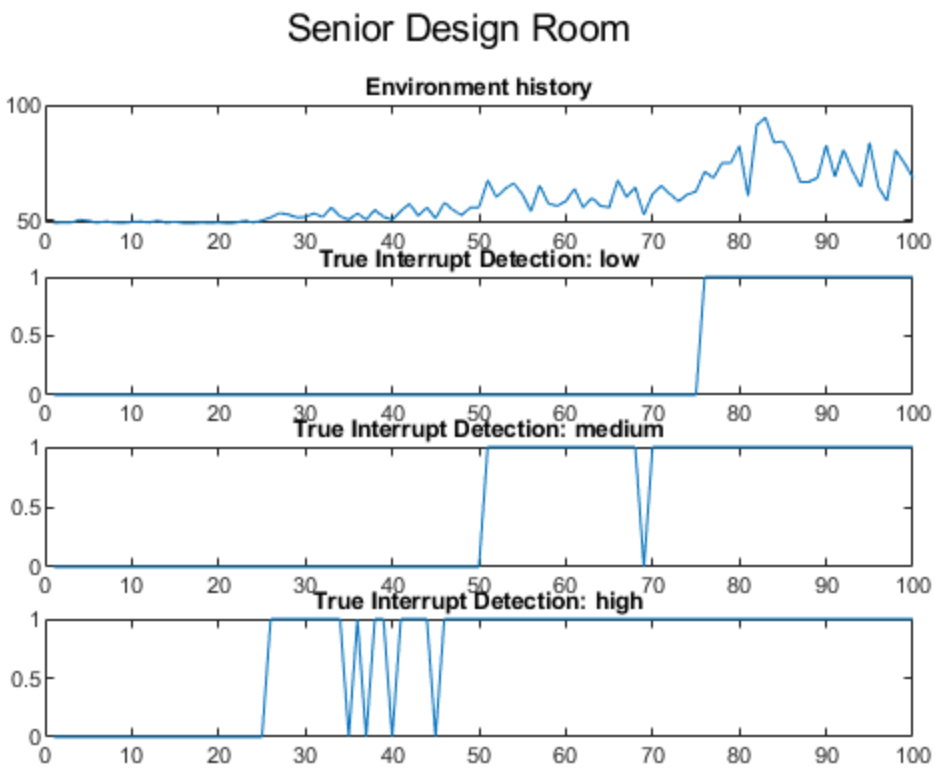
```
FDR_low = sum(FD_low, 'all')/sum(ID_low, 'all')    %probability of
 false interrupt given interrupt detection
FDR_med = sum(FD_med, 'all')/sum(ID_med, 'all')
FDR_high = sum(FD_high, 'all')/sum(ID_high, 'all')
```

```
figure(1)
subplot(4,1,1)
plot(history)
title('Environment history')
subplot(4,1,2)
plot(ID_low)
title('Interrupt Detection: low')
subplot(4,1,3)
plot(ID_med)
title('Interrupt Detection: medium')
subplot(4,1,4)
plot(ID_high)
title('Interrupt Detection: high')
suptitle(DataLoc)

figure(2)
subplot(4,1,1)
plot(history)
title('Environment history')
subplot(4,1,2)
plot(FD_low)
title('False Interrupt Detection: low')
subplot(4,1,3)
plot(FD_med)
title('False Interrupt Detection: medium')
subplot(4,1,4)
plot(FD_high)
title('False Interrupt Detection: high')
suptitle(DataLoc)

figure(3)
subplot(4,1,1)
plot(history)
title('Environment history')
subplot(4,1,2)
plot(TID_low)
title('True Interrupt Detection: low')
subplot(4,1,3)
plot(TID_med)
title('True Interrupt Detection: medium')
subplot(4,1,4)
plot(TID_high)
title('True Interrupt Detection: high')
suptitle(DataLoc)
```

```
DataLoc =

  1×1 cell array

    {'Senior Design Room'}

Ave20sec =

   49.3400

Std20sec =

    0.4502

RT_high =

   51.1407

RT_med =

   52.9415

RT_low =

   54.7422

FDR_low =

    0.5370

FDR_med =

    0.1967

FDR_high =

     0
```


Senior Design Room
Environment history
Interrupt Detection: low
Interrupt Detection: medium
Interrupt Detection: high


Senior Design Room
Environment history
False Interrupt Detection: low
False Interrupt Detection: medium
False Interrupt Detection: high



Published with MATLAB® R2019b

```
%ECE458 - Senior Design
%Michael Benker
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% FALSE DETECTION RATES %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clf;clear all; clc; close all;

samples=1000;
T_h_begin = 3; %Begin and end for parameter sweep of threshold numbers
T_h_end = 6;
T_m_begin = 6;
T_m_end = 15;
T_l_begin = 12;
T_l_end = 40;


%VARIABLES
Ave2sec = 0;    %2-second average
T_high = 4;     %standard deviations for high sensitivity setting
T_med = 8;      %standard deviations for medium sensitivity setting
T_low = 12;     %standard deviations for low sensitivity setting
Data20sec = zeros(10,1);
history = zeros(100,1);

RI_med = zeros(100,1); %1 if real interrupt, 0 if false interrupt
RI_high = zeros(100,1);
RI_low = zeros(100,1);
ID_low = zeros(100,1);   %1 if interrupt detected, 0 if interrupt not
 detected
ID_med = zeros(100,1);
ID_high = zeros(100,1);
TID_low = zeros(100,1);    %1 if ID = RI = 1, 0 otherwise
TID_med = zeros(100,1);
TID_high = zeros(100,1);
FD_low = zeros(100,1);     %1 if ID = 1 & RI = 0, 0 otherwise
FD_med = zeros(100,1);
FD_high = zeros(100,1);
FN_low = zeros(100,1);     %1 if ID = 0 & RI = 1, 0 otherwise
FN_med = zeros(100,1);
FN_high = zeros(100,1);
FDR_low = 0;    %probability of false interrupt given interrupt
 detection
FDR_med = 0;
FDR_high = 0;
FDR_low_array = zeros(samples,1);
FDR_med_array = zeros(samples,1);
FDR_high_array = zeros(samples,1);
T_high_array = zeros(samples,1);     %standard deviations for high
 sensitivity setting
T_med_array = zeros(samples,1);      %standard deviations for medium
 sensitivity setting
```

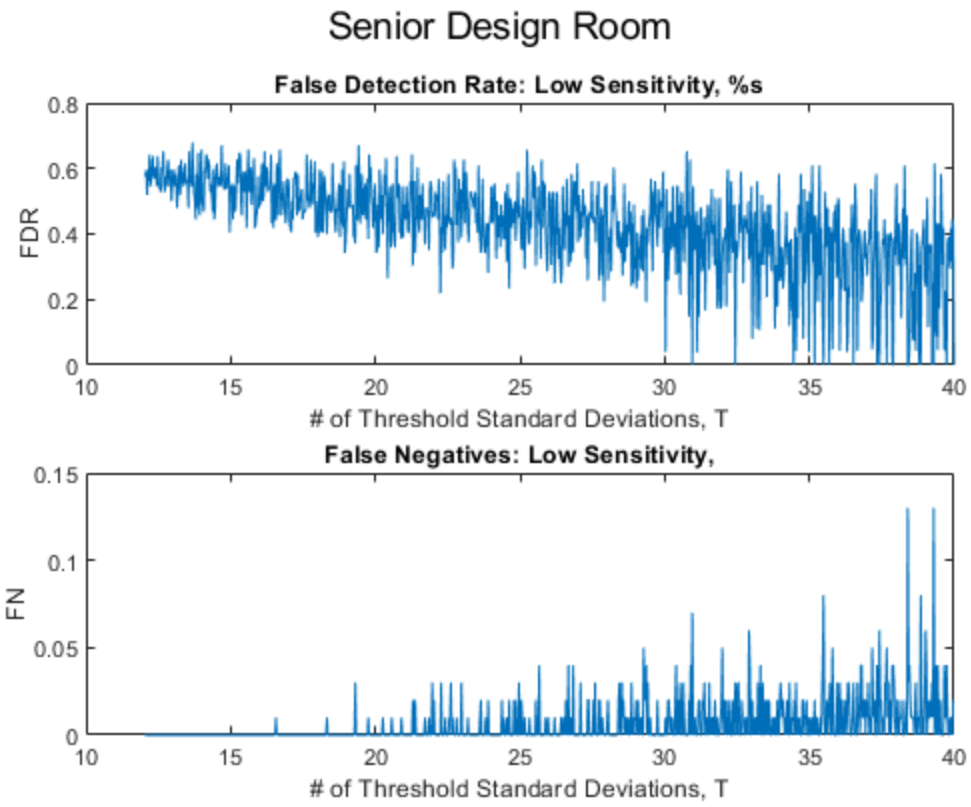
```
T_low_array = zeros(samples,1);     %standard deviations for low
 sensitivity setting
FN_low_array = zeros(samples,1);
FN_med_array = zeros(samples,1);
FN_high_array = zeros(samples,1);
Opti_low = zeros(samples,1);
Opti_med = zeros(samples,1);
Opti_high = zeros(samples,1);


%IMPORT DATA
SoundData1 = 'Book1.xlsx'; %Read excel file in folder
DataMat = zeros(30,4); %Predefine Data Matrix
Ambients = xlsread(SoundData1, 'A2:A31'); %Ambient 1st col
Quiets = xlsread(SoundData1, 'B2:B31'); %Quiet is 2nd col
Mediums = xlsread(SoundData1, 'C2:C31'); %Medium is 3rd col
Louds = xlsread(SoundData1, 'D2:D31'); %Loud is 4th col
[k,DataLoc] = xlsread(SoundData1, 'E1:E1');

%Define past 20 seconds (ambients)


for s=1:samples
    T_high=T_h_begin+s*(T_h_end-T_h_begin)/samples;
    T_med=T_m_begin+s*(T_m_end-T_m_begin)/samples;
    T_low=T_l_begin+s*(T_l_end-T_l_begin)/samples;

    T_high_array(s,1)=T_high;
    T_med_array(s,1)=T_med;
    T_low_array(s,1)=T_low;

    RI_med = zeros(100,1); %1 if real interrupt, 0 if false interrupt
    RI_high = zeros(100,1);
    RI_low = zeros(100,1);
    ID_low = zeros(100,1);   %1 if interrupt detected, 0 if interrupt
 not detected
    ID_med = zeros(100,1);
    ID_high = zeros(100,1);
    TID_low = zeros(100,1);    %1 if ID = RI = 1, 0 otherwise
    TID_med = zeros(100,1);
    TID_high = zeros(100,1);
    FD_low = zeros(100,1);     %1 if ID = 1 & RI = 0, 0 otherwise
    FD_med = zeros(100,1);
    FD_high = zeros(100,1);
    FN_low = zeros(100,1);     %1 if ID = 0 & RI = 1, 0 otherwise
    FN_med = zeros(100,1);
    FN_high = zeros(100,1);



for c =1:10
    Data20sec(c,1) = Ambients(randi([1 30],1,1),1);
    history(c,1)=Data20sec(c,1);
```

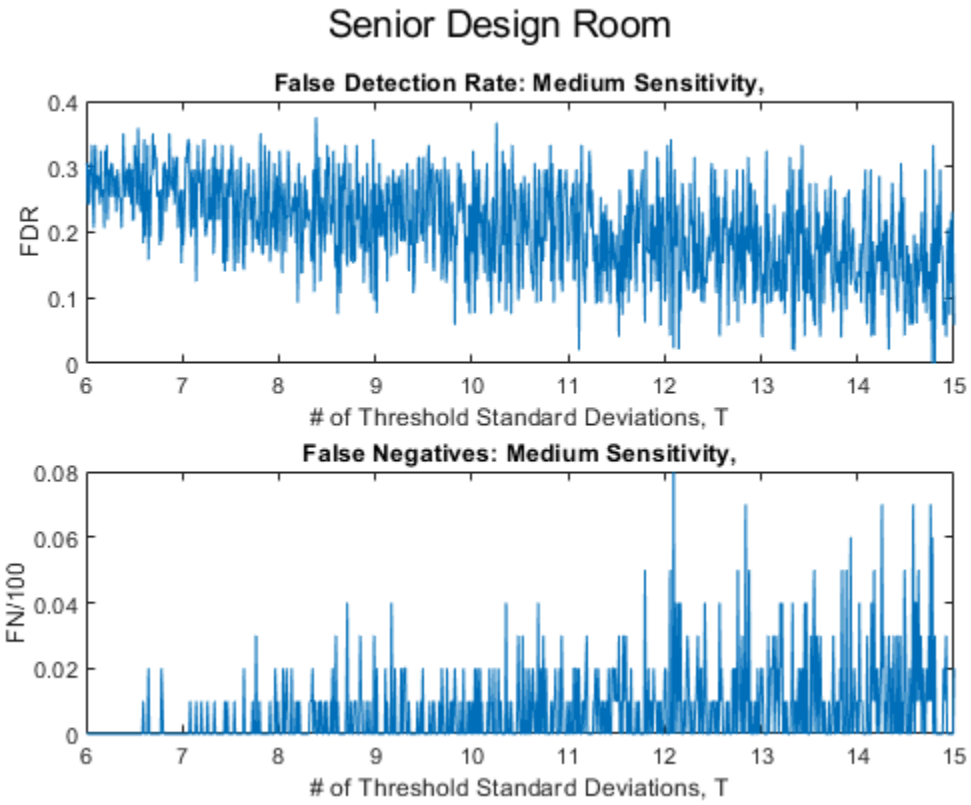
```
end
Ave20sec = mean(Data20sec);
Std20sec = std(Data20sec);
RT_high = Ave20sec+Std20sec*T_high; %Running threshold level (high
 sens)
RT_med = Ave20sec+Std20sec*T_med; %Running threshold level (med sens)
RT_low = Ave20sec+Std20sec*T_low; %Running threshold level (low sens)

for c=11:25
    new = Ambients(randi([1 30],1,1),1);
    history(c,1)=new;

    if new>RT_high
            ID_high(c,1)=1;
            FD_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%Quiet interrupts
%Only high sensitivity should activate interrupt
for c=26:50
    new = Quiets(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%medium interrupts
%Only high and medium sensitivity should activate interrupt

for c=51:75
```

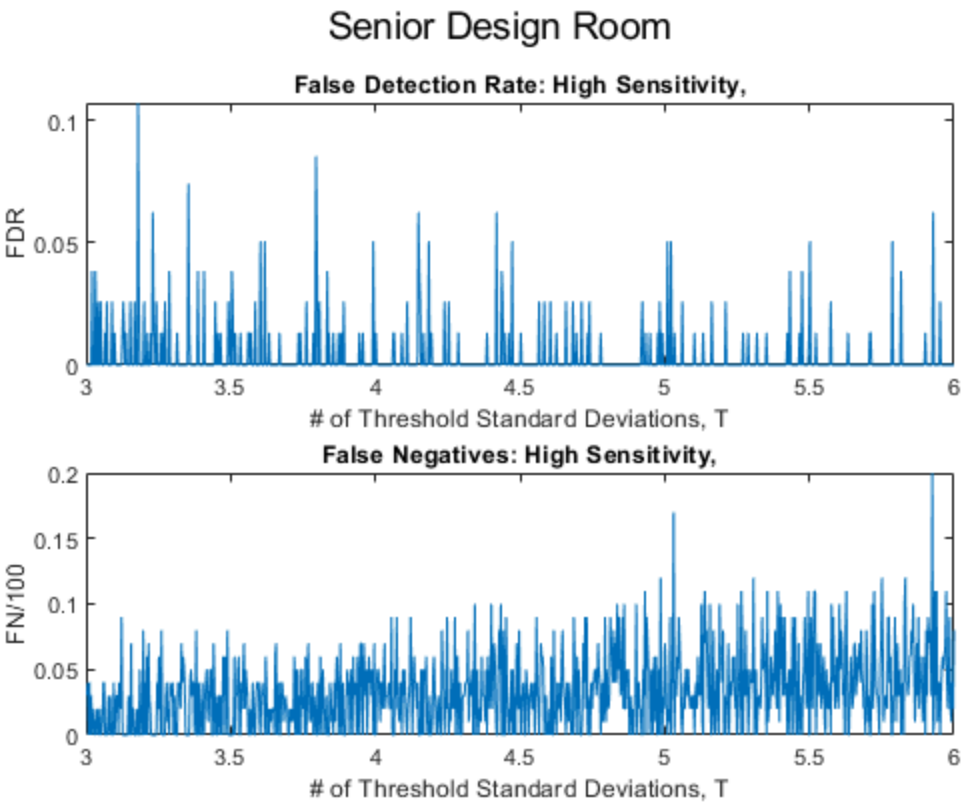
```
    new = Mediums(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end



%loud interrupts
%all activate interrupt
for c=76:100
    new = Louds(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    RI_low(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            TID_low(c,1)=1;
    else
        FN_low(c,1)=1;
    end

end
```

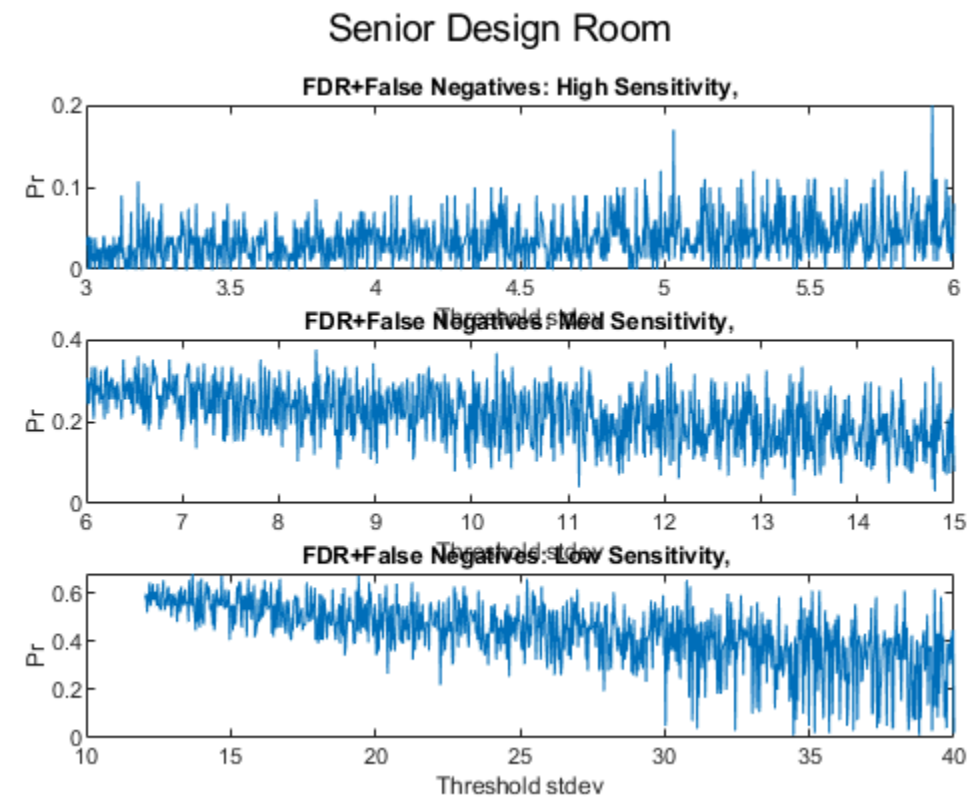
```
%False Interrupt Detection Rate - Print all
FDR_low = sum(FD_low, 'all')/sum(ID_low, 'all');    %probability of
 false interrupt given interrupt detection
FDR_med = sum(FD_med, 'all')/sum(ID_med, 'all');
FDR_high = sum(FD_high, 'all')/sum(ID_high, 'all');

FDR_low_array(s,1) = FDR_low;
FDR_med_array(s,1) = FDR_med;
FDR_high_array(s,1) = FDR_high;

FN_low_array(s,1)=sum(FN_low,'all')/100;
FN_med_array(s,1)=sum(FN_med,'all')/100;
FN_high_array(s,1)=sum(FN_high,'all')/100;

Opti_low(s,1)= FDR_low_array(s,1)+FN_low_array(s,1);
Opti_med(s,1)=FDR_med_array(s,1)+FN_med_array(s,1);
Opti_high(s,1)=FDR_high_array(s,1)+FN_high_array(s,1);


end

%!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!


DataLoc = char(DataLoc);
figure(1)
subplot(2,1,1)
plot(T_low_array,FDR_low_array)
title('False Detection Rate: Low Sensitivity, %s')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FDR')
subplot(2,1,2)
plot(T_low_array,FN_low_array)
title('False Negatives: Low Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FN')
suptitle(DataLoc)

figure(2)
subplot(2,1,1)
plot(T_med_array,FDR_med_array)
title('False Detection Rate: Medium Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FDR')
subplot(2,1,2)
plot(T_med_array,FN_med_array)
title('False Negatives: Medium Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FN/100')
suptitle(DataLoc)

figure(3)
subplot(2,1,1)
plot(T_high_array,FDR_high_array)
```

```
title('False Detection Rate: High Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FDR')
subplot(2,1,2)
plot(T_high_array,FN_high_array)
title('False Negatives: High Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FN/100')
suptitle(DataLoc)

figure(4)
subplot(3,1,1)
plot(T_high_array,Opti_high)
title('FDR+False Negatives: High Sensitivity,')
xlabel('Threshold stdev')
ylabel('Pr')
subplot(3,1,2)
plot(T_med_array,Opti_med)
title('FDR+False Negatives: Med Sensitivity,')
xlabel('Threshold stdev')
ylabel('Pr')
subplot(3,1,3)
plot(T_low_array,Opti_low)
title('FDR+False Negatives: Low Sensitivity, ')
xlabel('Threshold stdev')
ylabel('Pr')
suptitle(DataLoc)
```


Senior Design Room
False Detection Rate: Low Sensitivity, %s
FDR
# of Threshold Standard Deviations, T
False Negatives: Low Sensitivity,
FN
# of Threshold Standard Deviations, T


Senior Design Room
False Detection Rate: Medium Sensitivity,
FDR
# of Threshold Standard Deviations, T
False Negatives: Medium Sensitivity,
FN/100
# of Threshold Standard Deviations, T


Senior Design Room
False Detection Rate: High Sensitivity,
FDR
# of Threshold Standard Deviations, T
False Negatives: High Sensitivity,
FN/100
# of Threshold Standard Deviations, T


Senior Design Room
FDR+False Negatives: High Sensitivity,
Pr
FDR+False Negatives: Med Sensitivity,
Pr
FDR+False Negatives: Low Sensitivity,
Pr
Threshold stdev

Published with MATLAB® R2019b

RF Photonics Lab

```
%ECE458 - Senior Design
%Michael Benker
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% FALSE DETECTION RATES %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clf;clear all; clc; close all;

%VARIABLES
Ave2sec = 0;    %2-second average
T_high = 4;     %standard deviations for high sensitivity setting
T_med = 8;      %standard deviations for medium sensitivity setting
T_low = 12;     %standard deviations for low sensitivity setting
Data20sec = zeros(10,1);
history = zeros(100,1);

RI_med = zeros(100,1); %1 if real interrupt, 0 if false interrupt
RI_high = zeros(100,1);
RI_low = zeros(100,1);
ID_low = zeros(100,1);   %1 if interrupt detected, 0 if interrupt not
 detected
ID_med = zeros(100,1);
ID_high = zeros(100,1);
TID_low = zeros(100,1);    %1 if ID = RI = 1, 0 otherwise
TID_med = zeros(100,1);
TID_high = zeros(100,1);
FD_low = zeros(100,1);     %1 if ID = 1 & RI = 0, 0 otherwise
FD_med = zeros(100,1);
FD_high = zeros(100,1);
FN_low = zeros(100,1);     %1 if ID = 0 & RI = 1, 0 otherwise
FN_med = zeros(100,1);
FN_high = zeros(100,1);
FDR_low = 0;    %probability of false interrupt given interrupt
 detection
FDR_med = 0;
FDR_high = 0;

%IMPORT DATA
SoundData1 = 'Book5.xlsx'; %Read excel file in folder
DataMat = zeros(30,4); %Predefine Data Matrix
Ambients = xlsread(SoundData1, 'A2:A31'); %Ambient 1st col
Quiets = xlsread(SoundData1, 'B2:B31'); %Quiet is 2nd col
Mediums = xlsread(SoundData1, 'C2:C31'); %Medium is 3rd col
Louds = xlsread(SoundData1, 'D2:D31'); %Loud is 4th col
[k,DataLoc] = xlsread(SoundData1, 'E1:E1');
DataLoc
%Define past 20 seconds (ambients)

for c =1:10
    Data20sec(c,1) = Ambients(randi([1 30],1,1),1);
    history(c,1)=Data20sec(c,1);
end
Ave20sec = mean(Data20sec)
```

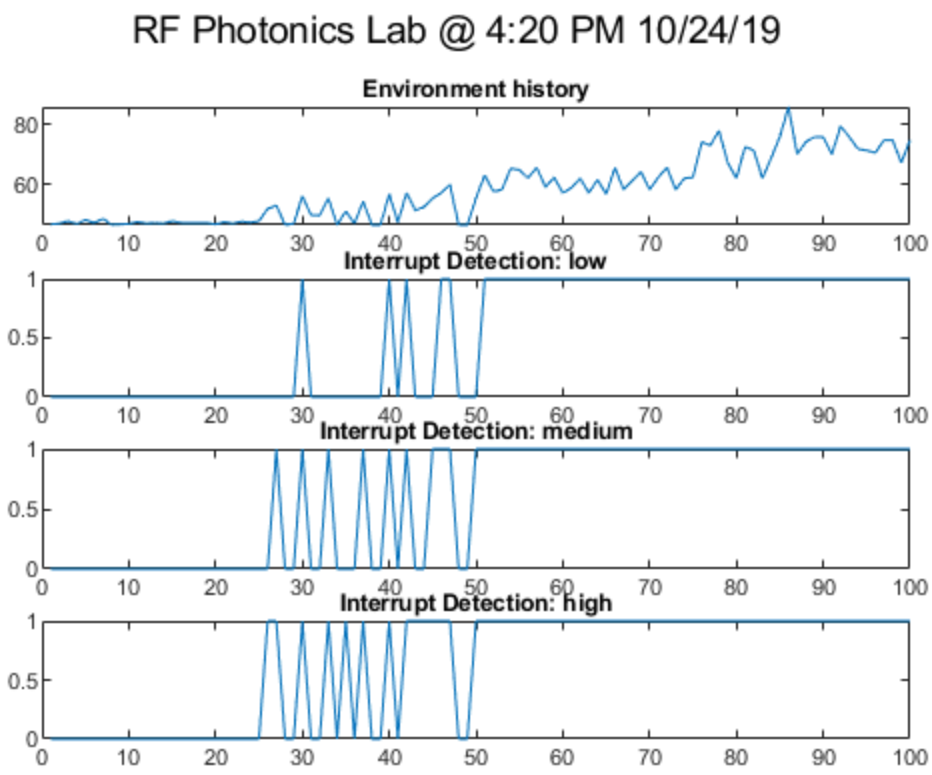
```
Std20sec = std(Data20sec)
RT_high = Ave20sec+Std20sec*T_high %Running threshold level (high
 sens)
RT_med = Ave20sec+Std20sec*T_med %Running threshold level (med sens)
RT_low = Ave20sec+Std20sec*T_low %Running threshold level (low sens)

for c=11:25
    new = Ambients(randi([1 30],1,1),1);
    history(c,1)=new;

    if new>RT_high
            ID_high(c,1)=1;
            FD_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%Quiet interrupts
%Only high sensitivity should activate interrupt
for c=26:50
    new = Quiets(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%medium interrupts
%Only high and medium sensitivity should activate interrupt

for c=51:75
    new = Mediums(randi([1 30],1,1),1);
    history(c,1)=new;
```

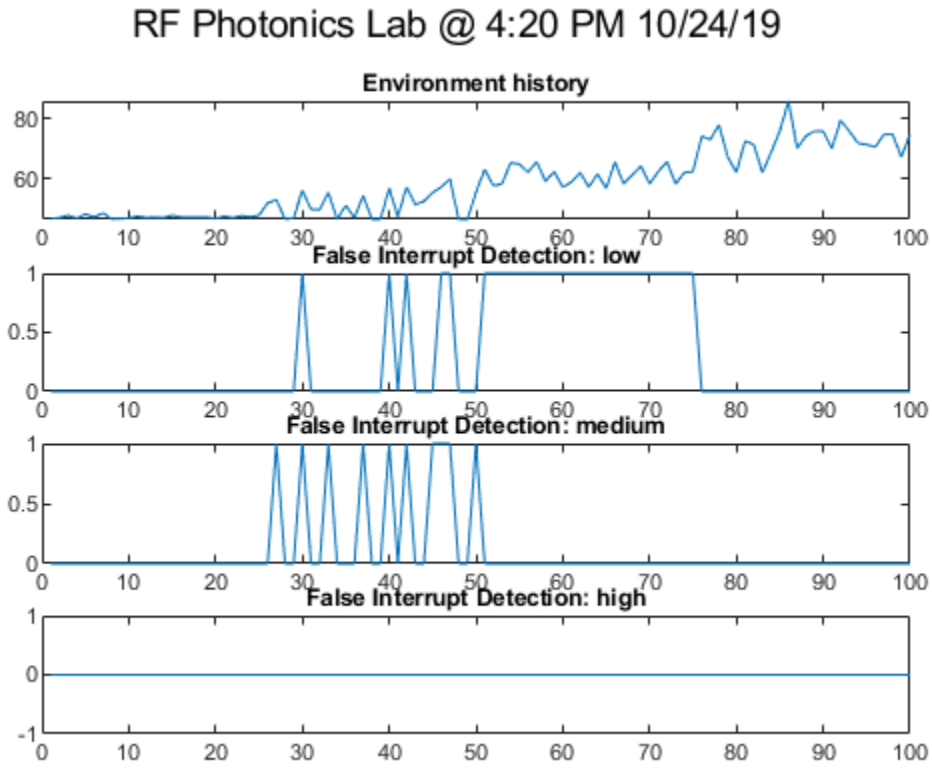
```
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end



%loud interrupts
%all activate interrupt
for c=76:100
    new = Louds(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    RI_low(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            TID_low(c,1)=1;
    else
        FN_low(c,1)=1;
    end

end

%False Interrupt Detection Rate - Print all
```

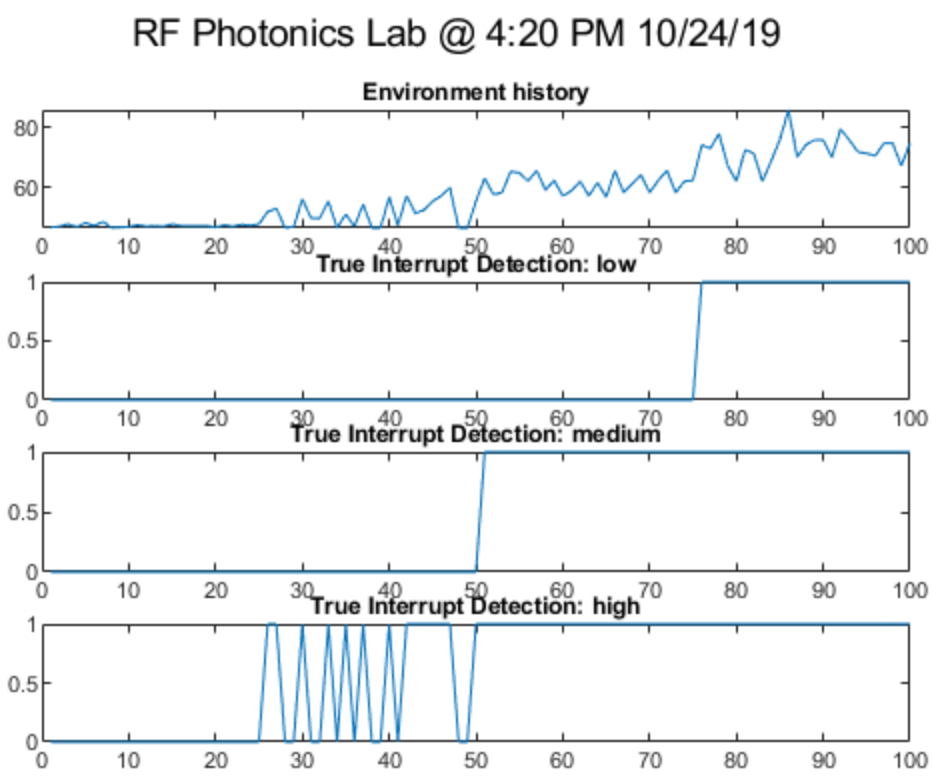
```
FDR_low = sum(FD_low, 'all')/sum(ID_low, 'all')    %probability of
 false interrupt given interrupt detection
FDR_med = sum(FD_med, 'all')/sum(ID_med, 'all')
FDR_high = sum(FD_high, 'all')/sum(ID_high, 'all')
```

```
figure(1)
subplot(4,1,1)
plot(history)
title('Environment history')
subplot(4,1,2)
plot(ID_low)
title('Interrupt Detection: low')
subplot(4,1,3)
plot(ID_med)
title('Interrupt Detection: medium')
subplot(4,1,4)
plot(ID_high)
title('Interrupt Detection: high')
suptitle(DataLoc)

figure(2)
subplot(4,1,1)
plot(history)
title('Environment history')
subplot(4,1,2)
plot(FD_low)
title('False Interrupt Detection: low')
subplot(4,1,3)
plot(FD_med)
title('False Interrupt Detection: medium')
subplot(4,1,4)
plot(FD_high)
title('False Interrupt Detection: high')
suptitle(DataLoc)

figure(3)
subplot(4,1,1)
plot(history)
title('Environment history')
subplot(4,1,2)
plot(TID_low)
title('True Interrupt Detection: low')
subplot(4,1,3)
plot(TID_med)
title('True Interrupt Detection: medium')
subplot(4,1,4)
plot(TID_high)
title('True Interrupt Detection: high')
suptitle(DataLoc)
```

```
DataLoc =

  1×1 cell array

    {'RF Photonics Lab @ 4:20 PM 10/24/19'}

Ave20sec =

   47.3600

Std20sec =

    0.6963

RT_high =

   50.1454

RT_med =

   52.9307

RT_low =

   55.7161

FDR_low =

    0.5455

FDR_med =

    0.1667

FDR_high =

     0
```


RF Photonics Lab @ 4:20 PM 10/24/19
Environment history
Interrupt Detection: low
Interrupt Detection: medium
Interrupt Detection: high


RF Photonics Lab @ 4:20 PM 10/24/19
Environment history
False Interrupt Detection: low
False Interrupt Detection: medium
False Interrupt Detection: high



Published with MATLAB® R2019b

```
%ECE458 - Senior Design
%Michael Benker
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% FALSE DETECTION RATES %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clf;clear all; clc; close all;

samples=1000;
T_h_begin = 3; %Begin and end for parameter sweep of threshold numbers
T_h_end = 6;
T_m_begin = 6;
T_m_end = 15;
T_l_begin = 12;
T_l_end = 40;


%VARIABLES
Ave2sec = 0;    %2-second average
T_high = 4;     %standard deviations for high sensitivity setting
T_med = 8;      %standard deviations for medium sensitivity setting
T_low = 12;     %standard deviations for low sensitivity setting
Data20sec = zeros(10,1);
history = zeros(100,1);

RI_med = zeros(100,1); %1 if real interrupt, 0 if false interrupt
RI_high = zeros(100,1);
RI_low = zeros(100,1);
ID_low = zeros(100,1);   %1 if interrupt detected, 0 if interrupt not
 detected
ID_med = zeros(100,1);
ID_high = zeros(100,1);
TID_low = zeros(100,1);    %1 if ID = RI = 1, 0 otherwise
TID_med = zeros(100,1);
TID_high = zeros(100,1);
FD_low = zeros(100,1);     %1 if ID = 1 & RI = 0, 0 otherwise
FD_med = zeros(100,1);
FD_high = zeros(100,1);
FN_low = zeros(100,1);     %1 if ID = 0 & RI = 1, 0 otherwise
FN_med = zeros(100,1);
FN_high = zeros(100,1);
FDR_low = 0;    %probability of false interrupt given interrupt
 detection
FDR_med = 0;
FDR_high = 0;
FDR_low_array = zeros(samples,1);
FDR_med_array = zeros(samples,1);
FDR_high_array = zeros(samples,1);
T_high_array = zeros(samples,1);     %standard deviations for high
 sensitivity setting
T_med_array = zeros(samples,1);      %standard deviations for medium
 sensitivity setting
```

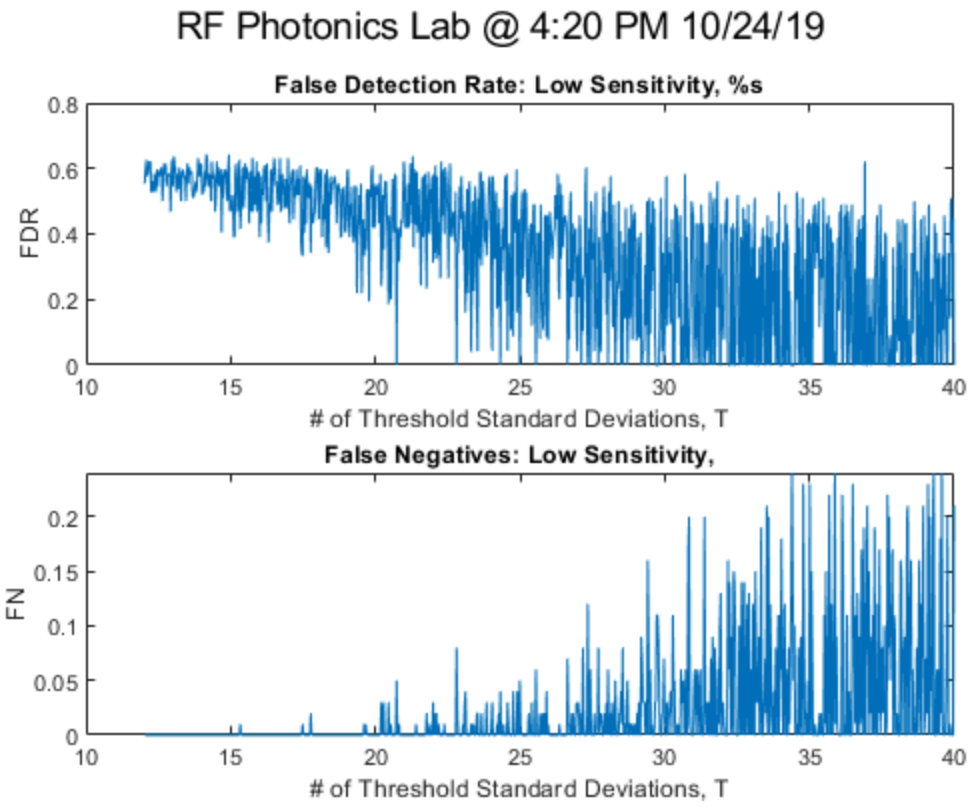
```
T_low_array = zeros(samples,1);     %standard deviations for low
 sensitivity setting
FN_low_array = zeros(samples,1);
FN_med_array = zeros(samples,1);
FN_high_array = zeros(samples,1);
Opti_low = zeros(samples,1);
Opti_med = zeros(samples,1);
Opti_high = zeros(samples,1);


%IMPORT DATA
SoundData1 = 'Book5.xlsx'; %Read excel file in folder
DataMat = zeros(30,4); %Predefine Data Matrix
Ambients = xlsread(SoundData1, 'A2:A31'); %Ambient 1st col
Quiets = xlsread(SoundData1, 'B2:B31'); %Quiet is 2nd col
Mediums = xlsread(SoundData1, 'C2:C31'); %Medium is 3rd col
Louds = xlsread(SoundData1, 'D2:D31'); %Loud is 4th col
[k,DataLoc] = xlsread(SoundData1, 'E1:E1');

%Define past 20 seconds (ambients)


for s=1:samples
    T_high=T_h_begin+s*(T_h_end-T_h_begin)/samples;
    T_med=T_m_begin+s*(T_m_end-T_m_begin)/samples;
    T_low=T_l_begin+s*(T_l_end-T_l_begin)/samples;

    T_high_array(s,1)=T_high;
    T_med_array(s,1)=T_med;
    T_low_array(s,1)=T_low;

    RI_med = zeros(100,1); %1 if real interrupt, 0 if false interrupt
    RI_high = zeros(100,1);
    RI_low = zeros(100,1);
    ID_low = zeros(100,1);   %1 if interrupt detected, 0 if interrupt
 not detected
    ID_med = zeros(100,1);
    ID_high = zeros(100,1);
    TID_low = zeros(100,1);    %1 if ID = RI = 1, 0 otherwise
    TID_med = zeros(100,1);
    TID_high = zeros(100,1);
    FD_low = zeros(100,1);     %1 if ID = 1 & RI = 0, 0 otherwise
    FD_med = zeros(100,1);
    FD_high = zeros(100,1);
    FN_low = zeros(100,1);     %1 if ID = 0 & RI = 1, 0 otherwise
    FN_med = zeros(100,1);
    FN_high = zeros(100,1);



for c =1:10
    Data20sec(c,1) = Ambients(randi([1 30],1,1),1);
    history(c,1)=Data20sec(c,1);
```

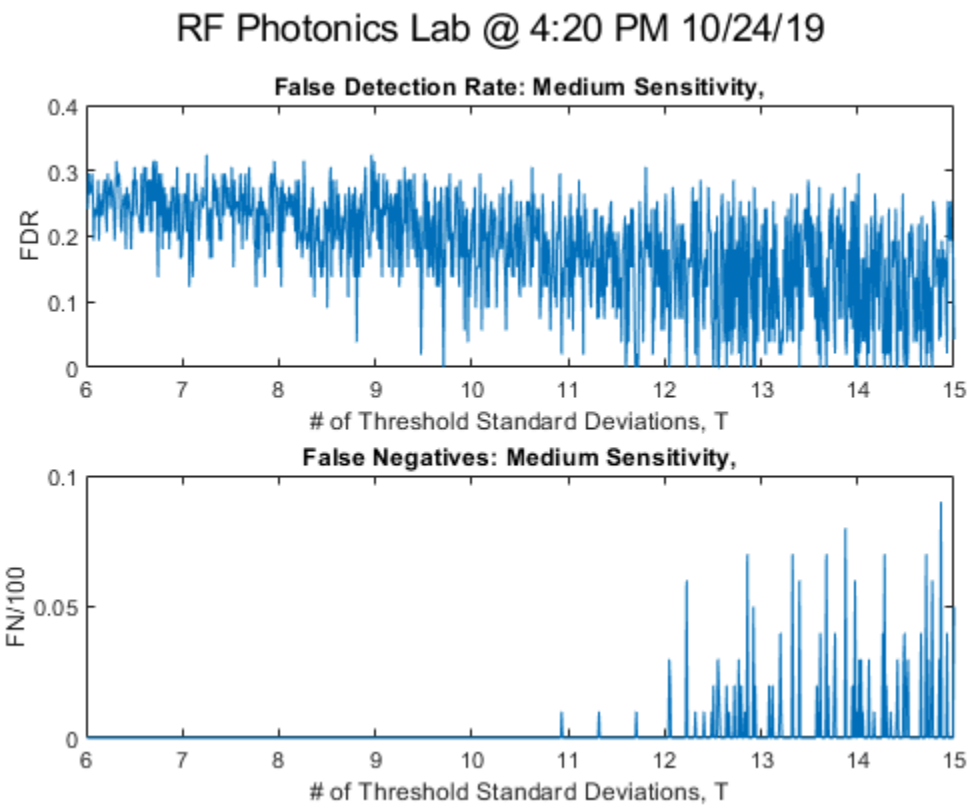
```
end
Ave20sec = mean(Data20sec);
Std20sec = std(Data20sec);
RT_high = Ave20sec+Std20sec*T_high; %Running threshold level (high
 sens)
RT_med = Ave20sec+Std20sec*T_med; %Running threshold level (med sens)
RT_low = Ave20sec+Std20sec*T_low; %Running threshold level (low sens)

for c=11:25
    new = Ambients(randi([1 30],1,1),1);
    history(c,1)=new;

    if new>RT_high
            ID_high(c,1)=1;
            FD_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%Quiet interrupts
%Only high sensitivity should activate interrupt
for c=26:50
    new = Quiets(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%medium interrupts
%Only high and medium sensitivity should activate interrupt

for c=51:75
```

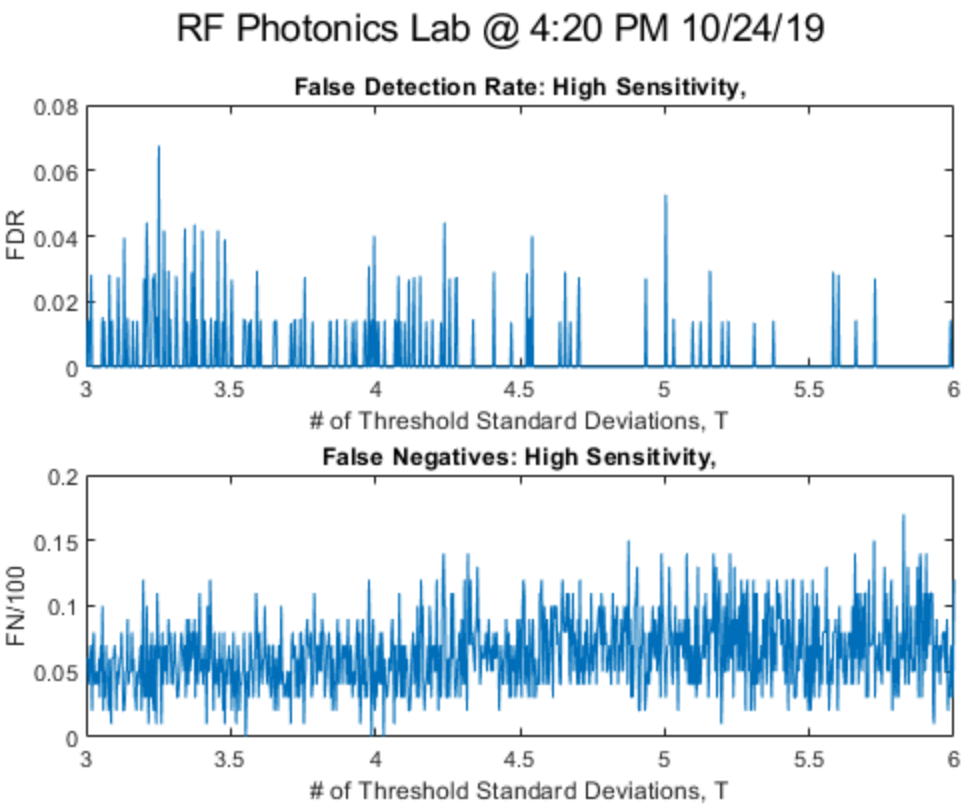
```
    new = Mediums(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end



%loud interrupts
%all activate interrupt
for c=76:100
    new = Louds(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    RI_low(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            TID_low(c,1)=1;
    else
        FN_low(c,1)=1;
    end

end
```

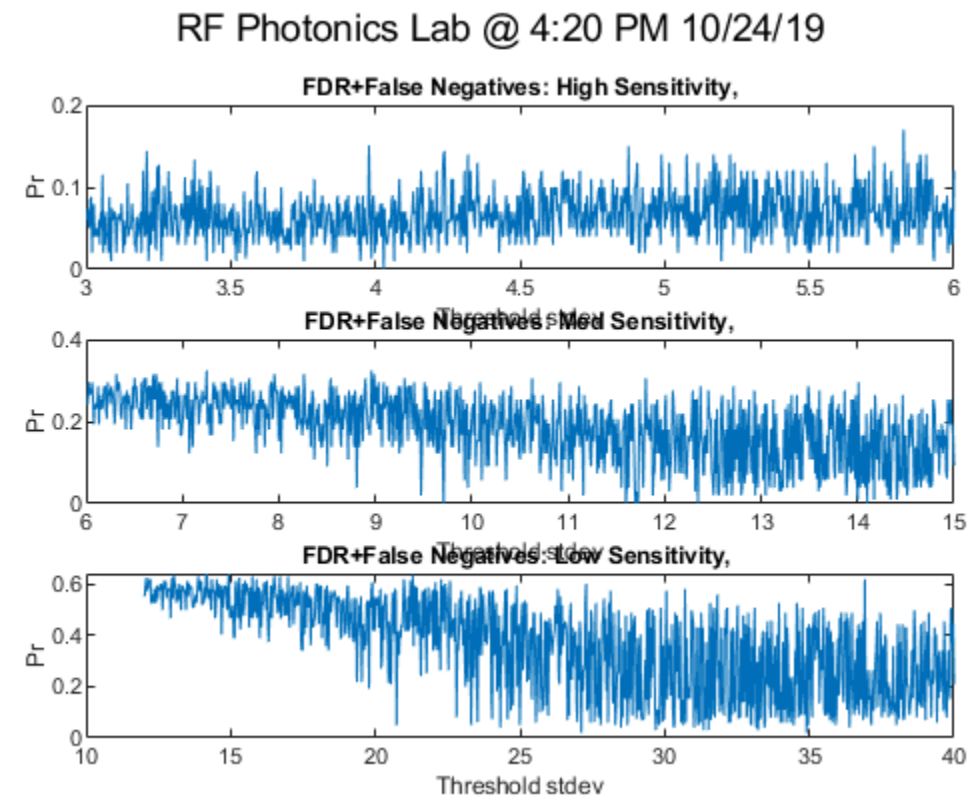
```
%False Interrupt Detection Rate - Print all
FDR_low = sum(FD_low, 'all')/sum(ID_low, 'all');    %probability of
 false interrupt given interrupt detection
FDR_med = sum(FD_med, 'all')/sum(ID_med, 'all');
FDR_high = sum(FD_high, 'all')/sum(ID_high, 'all');

FDR_low_array(s,1) = FDR_low;
FDR_med_array(s,1) = FDR_med;
FDR_high_array(s,1) = FDR_high;

FN_low_array(s,1)=sum(FN_low,'all')/100;
FN_med_array(s,1)=sum(FN_med,'all')/100;
FN_high_array(s,1)=sum(FN_high,'all')/100;

Opti_low(s,1)= FDR_low_array(s,1)+FN_low_array(s,1);
Opti_med(s,1)=FDR_med_array(s,1)+FN_med_array(s,1);
Opti_high(s,1)=FDR_high_array(s,1)+FN_high_array(s,1);


end

%!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!


DataLoc = char(DataLoc);
figure(1)
subplot(2,1,1)
plot(T_low_array,FDR_low_array)
title('False Detection Rate: Low Sensitivity, %s')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FDR')
subplot(2,1,2)
plot(T_low_array,FN_low_array)
title('False Negatives: Low Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FN')
suptitle(DataLoc)

figure(2)
subplot(2,1,1)
plot(T_med_array,FDR_med_array)
title('False Detection Rate: Medium Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FDR')
subplot(2,1,2)
plot(T_med_array,FN_med_array)
title('False Negatives: Medium Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FN/100')
suptitle(DataLoc)

figure(3)
subplot(2,1,1)
plot(T_high_array,FDR_high_array)
```

```
title('False Detection Rate: High Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FDR')
subplot(2,1,2)
plot(T_high_array,FN_high_array)
title('False Negatives: High Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FN/100')
suptitle(DataLoc)

figure(4)
subplot(3,1,1)
plot(T_high_array,Opti_high)
title('FDR+False Negatives: High Sensitivity,')
xlabel('Threshold stdev')
ylabel('Pr')
subplot(3,1,2)
plot(T_med_array,Opti_med)
title('FDR+False Negatives: Med Sensitivity,')
xlabel('Threshold stdev')
ylabel('Pr')
subplot(3,1,3)
plot(T_low_array,Opti_low)
title('FDR+False Negatives: Low Sensitivity, ')
xlabel('Threshold stdev')
ylabel('Pr')
suptitle(DataLoc)
```


RF Photonics Lab @ 4:20 PM 10/24/19
False Detection Rate: Low Sensitivity, %s
FDR
# of Threshold Standard Deviations, T
False Negatives: Low Sensitivity,
FN
# of Threshold Standard Deviations, T


RF Photonics Lab @ 4:20 PM 10/24/19
False Detection Rate: Medium Sensitivity,
FDR
# of Threshold Standard Deviations, T
False Negatives: Medium Sensitivity,
FN/100
# of Threshold Standard Deviations, T


RF Photonics Lab @ 4:20 PM 10/24/19
False Detection Rate: High Sensitivity,
FDR
# of Threshold Standard Deviations, T
False Negatives: High Sensitivity,
FN/100
# of Threshold Standard Deviations, T


RF Photonics Lab @ 4:20 PM 10/24/19
FDR+False Negatives: High Sensitivity,
Pr
FDR+False Negatives: Med Sensitivity,
Pr
FDR+False Negatives: Low Sensitivity,
Pr
Threshold stdev

Published with MATLAB® R2019b

WENDY'S

```
%ECE458 - Senior Design
%Michael Benker
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% FALSE DETECTION RATES %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clf;clear all; clc; close all;

%VARIABLES
Ave2sec = 0;    %2-second average
T_high = 4;     %standard deviations for high sensitivity setting
T_med = 8;      %standard deviations for medium sensitivity setting
T_low = 12;     %standard deviations for low sensitivity setting
Data20sec = zeros(10,1);
history = zeros(100,1);

RI_med = zeros(100,1); %1 if real interrupt, 0 if false interrupt
RI_high = zeros(100,1);
RI_low = zeros(100,1);
ID_low = zeros(100,1);   %1 if interrupt detected, 0 if interrupt not
 detected
ID_med = zeros(100,1);
ID_high = zeros(100,1);
TID_low = zeros(100,1);    %1 if ID = RI = 1, 0 otherwise
TID_med = zeros(100,1);
TID_high = zeros(100,1);
FD_low = zeros(100,1);     %1 if ID = 1 & RI = 0, 0 otherwise
FD_med = zeros(100,1);
FD_high = zeros(100,1);
FN_low = zeros(100,1);     %1 if ID = 0 & RI = 1, 0 otherwise
FN_med = zeros(100,1);
FN_high = zeros(100,1);
FDR_low = 0;    %probability of false interrupt given interrupt
 detection
FDR_med = 0;
FDR_high = 0;

%IMPORT DATA
SoundData1 = 'Book3.xlsx'; %Read excel file in folder
DataMat = zeros(30,4); %Predefine Data Matrix
Ambients = xlsread(SoundData1, 'A2:A31'); %Ambient 1st col
Quiets = xlsread(SoundData1, 'B2:B31'); %Quiet is 2nd col
Mediums = xlsread(SoundData1, 'C2:C31'); %Medium is 3rd col
Louds = xlsread(SoundData1, 'D2:D31'); %Loud is 4th col
[k,DataLoc] = xlsread(SoundData1, 'E1:E1');
DataLoc
%Define past 20 seconds (ambients)

for c =1:10
    Data20sec(c,1) = Ambients(randi([1 30],1,1),1);
    history(c,1)=Data20sec(c,1);
end
Ave20sec = mean(Data20sec)
```

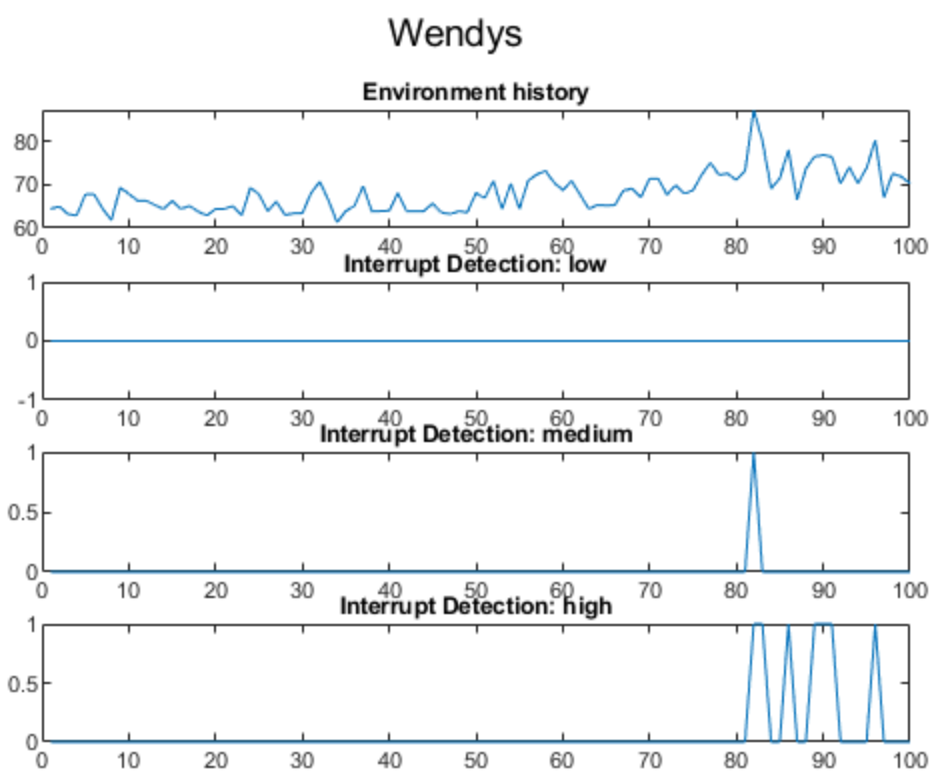
```
Std20sec = std(Data20sec)
RT_high = Ave20sec+Std20sec*T_high %Running threshold level (high
 sens)
RT_med = Ave20sec+Std20sec*T_med %Running threshold level (med sens)
RT_low = Ave20sec+Std20sec*T_low %Running threshold level (low sens)

for c=11:25
    new = Ambients(randi([1 30],1,1),1);
    history(c,1)=new;

    if new>RT_high
            ID_high(c,1)=1;
            FD_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%Quiet interrupts
%Only high sensitivity should activate interrupt
for c=26:50
    new = Quiets(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%medium interrupts
%Only high and medium sensitivity should activate interrupt

for c=51:75
    new = Mediums(randi([1 30],1,1),1);
    history(c,1)=new;
```

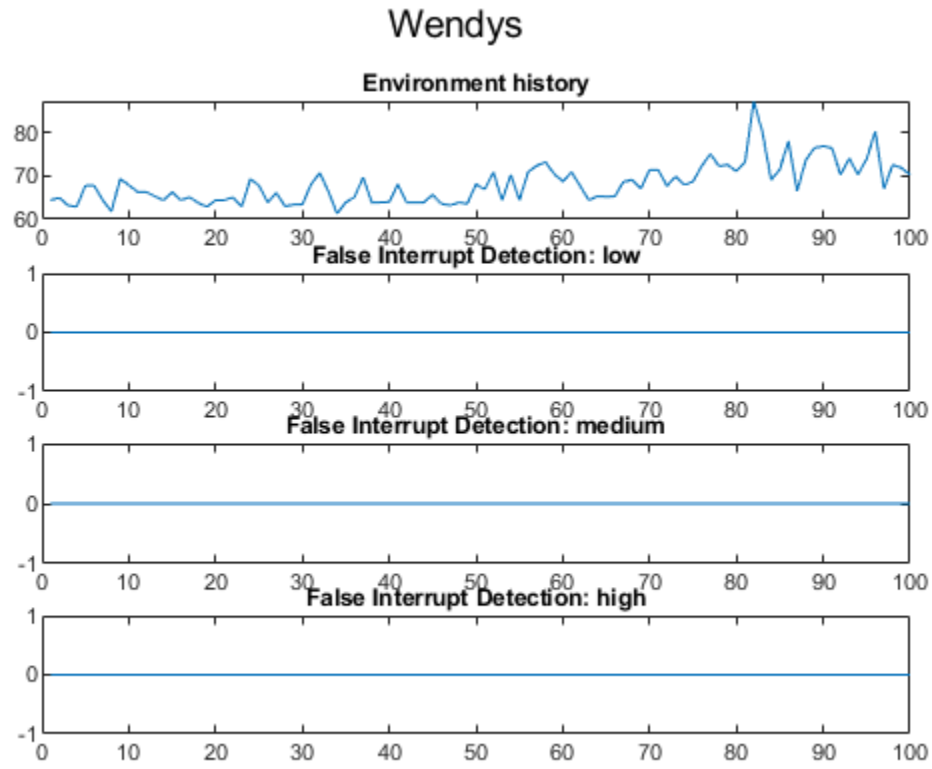
```
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end



%loud interrupts
%all activate interrupt
for c=76:100
    new = Louds(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    RI_low(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            TID_low(c,1)=1;
    else
        FN_low(c,1)=1;
    end

end

%False Interrupt Detection Rate - Print all
```

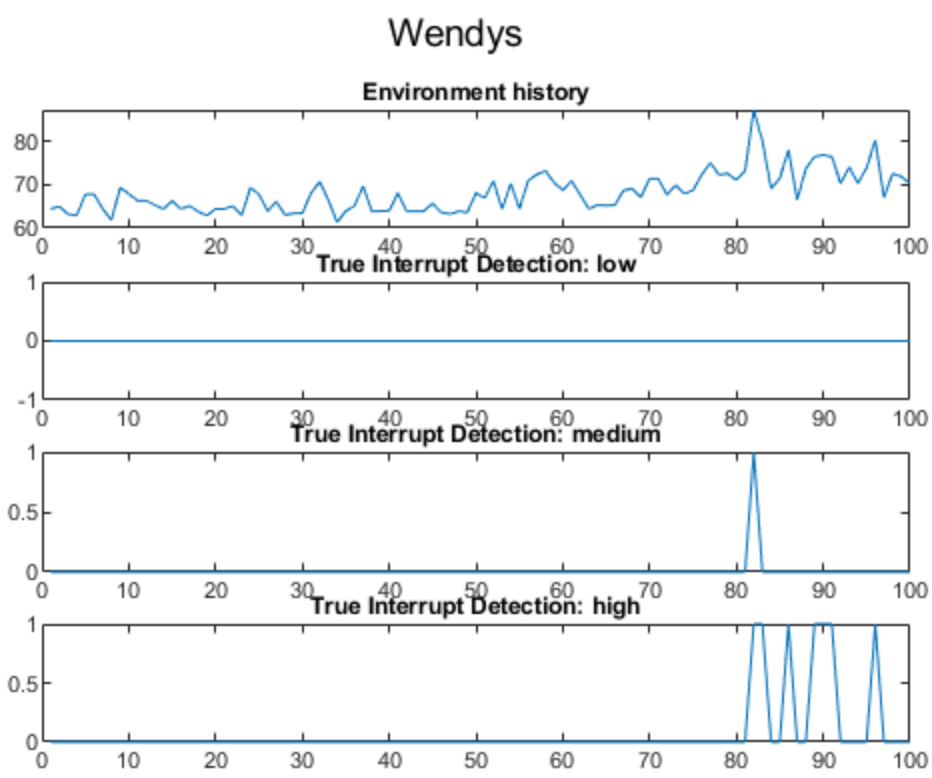
```
FDR_low = sum(FD_low, 'all')/sum(ID_low, 'all')    %probability of
 false interrupt given interrupt detection
FDR_med = sum(FD_med, 'all')/sum(ID_med, 'all')
FDR_high = sum(FD_high, 'all')/sum(ID_high, 'all')
```

```
figure(1)
subplot(4,1,1)
plot(history)
title('Environment history')
subplot(4,1,2)
plot(ID_low)
title('Interrupt Detection: low')
subplot(4,1,3)
plot(ID_med)
title('Interrupt Detection: medium')
subplot(4,1,4)
plot(ID_high)
title('Interrupt Detection: high')
suptitle(DataLoc)

figure(2)
subplot(4,1,1)
plot(history)
title('Environment history')
subplot(4,1,2)
plot(FD_low)
title('False Interrupt Detection: low')
subplot(4,1,3)
plot(FD_med)
title('False Interrupt Detection: medium')
subplot(4,1,4)
plot(FD_high)
title('False Interrupt Detection: high')
suptitle(DataLoc)

figure(3)
subplot(4,1,1)
plot(history)
title('Environment history')
subplot(4,1,2)
plot(TID_low)
title('True Interrupt Detection: low')
subplot(4,1,3)
plot(TID_med)
title('True Interrupt Detection: medium')
subplot(4,1,4)
plot(TID_high)
title('True Interrupt Detection: high')
suptitle(DataLoc)
```

```
DataLoc =

  1×1 cell array

    {'Wendys'}

Ave20sec =

   65.4500

Std20sec =

    2.5640

RT_high =

   75.7058

RT_med =

   85.9617

RT_low =

   96.2175

FDR_low =

   NaN

FDR_med =

     0

FDR_high =

     0
```


Wendys
Environment history
Interrupt Detection: low
Interrupt Detection: medium
Interrupt Detection: high


Wendys
Environment history
False Interrupt Detection: low
False Interrupt Detection: medium
False Interrupt Detection: high



Published with MATLAB® R2019b

```
%ECE458 - Senior Design
%Michael Benker
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% FALSE DETECTION RATES %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clf;clear all; clc; close all;

samples=1000;
T_h_begin = 3; %Begin and end for parameter sweep of threshold numbers
T_h_end = 6;
T_m_begin = 6;
T_m_end = 15;
T_l_begin = 12;
T_l_end = 40;


%VARIABLES
Ave2sec = 0;    %2-second average
T_high = 4;     %standard deviations for high sensitivity setting
T_med = 8;      %standard deviations for medium sensitivity setting
T_low = 12;     %standard deviations for low sensitivity setting
Data20sec = zeros(10,1);
history = zeros(100,1);

RI_med = zeros(100,1); %1 if real interrupt, 0 if false interrupt
RI_high = zeros(100,1);
RI_low = zeros(100,1);
ID_low = zeros(100,1);   %1 if interrupt detected, 0 if interrupt not
 detected
ID_med = zeros(100,1);
ID_high = zeros(100,1);
TID_low = zeros(100,1);    %1 if ID = RI = 1, 0 otherwise
TID_med = zeros(100,1);
TID_high = zeros(100,1);
FD_low = zeros(100,1);     %1 if ID = 1 & RI = 0, 0 otherwise
FD_med = zeros(100,1);
FD_high = zeros(100,1);
FN_low = zeros(100,1);     %1 if ID = 0 & RI = 1, 0 otherwise
FN_med = zeros(100,1);
FN_high = zeros(100,1);
FDR_low = 0;    %probability of false interrupt given interrupt
 detection
FDR_med = 0;
FDR_high = 0;
FDR_low_array = zeros(samples,1);
FDR_med_array = zeros(samples,1);
FDR_high_array = zeros(samples,1);
T_high_array = zeros(samples,1);     %standard deviations for high
 sensitivity setting
T_med_array = zeros(samples,1);      %standard deviations for medium
 sensitivity setting
```

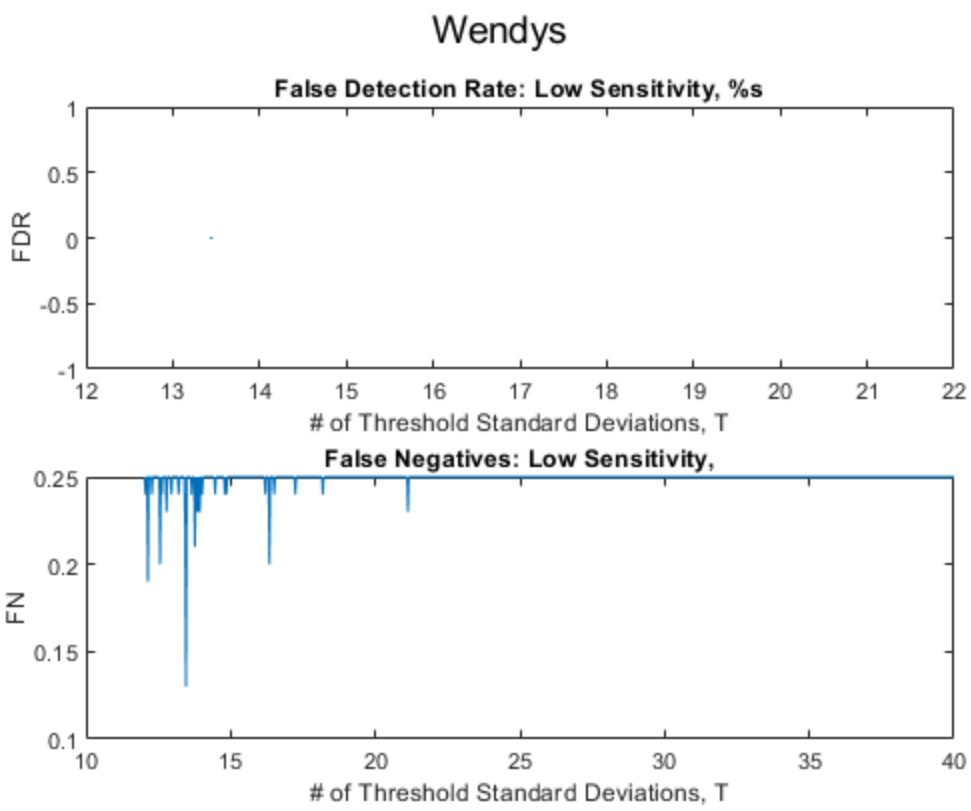
```
T_low_array = zeros(samples,1);     %standard deviations for low
 sensitivity setting
FN_low_array = zeros(samples,1);
FN_med_array = zeros(samples,1);
FN_high_array = zeros(samples,1);
Opti_low = zeros(samples,1);
Opti_med = zeros(samples,1);
Opti_high = zeros(samples,1);


%IMPORT DATA
SoundData1 = 'Book3.xlsx'; %Read excel file in folder
DataMat = zeros(30,4); %Predefine Data Matrix
Ambients = xlsread(SoundData1, 'A2:A31'); %Ambient 1st col
Quiets = xlsread(SoundData1, 'B2:B31'); %Quiet is 2nd col
Mediums = xlsread(SoundData1, 'C2:C31'); %Medium is 3rd col
Louds = xlsread(SoundData1, 'D2:D31'); %Loud is 4th col
[k,DataLoc] = xlsread(SoundData1, 'E1:E1');

%Define past 20 seconds (ambients)


for s=1:samples
    T_high=T_h_begin+s*(T_h_end-T_h_begin)/samples;
    T_med=T_m_begin+s*(T_m_end-T_m_begin)/samples;
    T_low=T_l_begin+s*(T_l_end-T_l_begin)/samples;

    T_high_array(s,1)=T_high;
    T_med_array(s,1)=T_med;
    T_low_array(s,1)=T_low;

    RI_med = zeros(100,1); %1 if real interrupt, 0 if false interrupt
    RI_high = zeros(100,1);
    RI_low = zeros(100,1);
    ID_low = zeros(100,1);   %1 if interrupt detected, 0 if interrupt
 not detected
    ID_med = zeros(100,1);
    ID_high = zeros(100,1);
    TID_low = zeros(100,1);    %1 if ID = RI = 1, 0 otherwise
    TID_med = zeros(100,1);
    TID_high = zeros(100,1);
    FD_low = zeros(100,1);     %1 if ID = 1 & RI = 0, 0 otherwise
    FD_med = zeros(100,1);
    FD_high = zeros(100,1);
    FN_low = zeros(100,1);     %1 if ID = 0 & RI = 1, 0 otherwise
    FN_med = zeros(100,1);
    FN_high = zeros(100,1);



for c =1:10
    Data20sec(c,1) = Ambients(randi([1 30],1,1),1);
    history(c,1)=Data20sec(c,1);
```

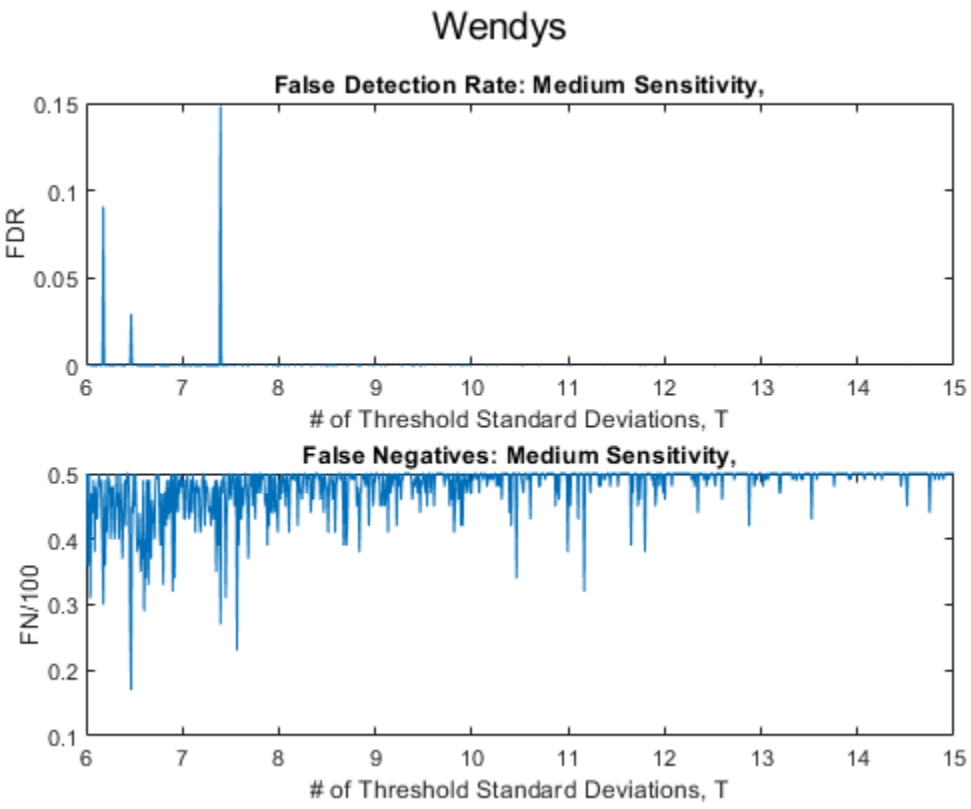
```
end
Ave20sec = mean(Data20sec);
Std20sec = std(Data20sec);
RT_high = Ave20sec+Std20sec*T_high; %Running threshold level (high
 sens)
RT_med = Ave20sec+Std20sec*T_med; %Running threshold level (med sens)
RT_low = Ave20sec+Std20sec*T_low; %Running threshold level (low sens)

for c=11:25
    new = Ambients(randi([1 30],1,1),1);
    history(c,1)=new;

    if new>RT_high
            ID_high(c,1)=1;
            FD_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%Quiet interrupts
%Only high sensitivity should activate interrupt
for c=26:50
    new = Quiets(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%medium interrupts
%Only high and medium sensitivity should activate interrupt

for c=51:75
```

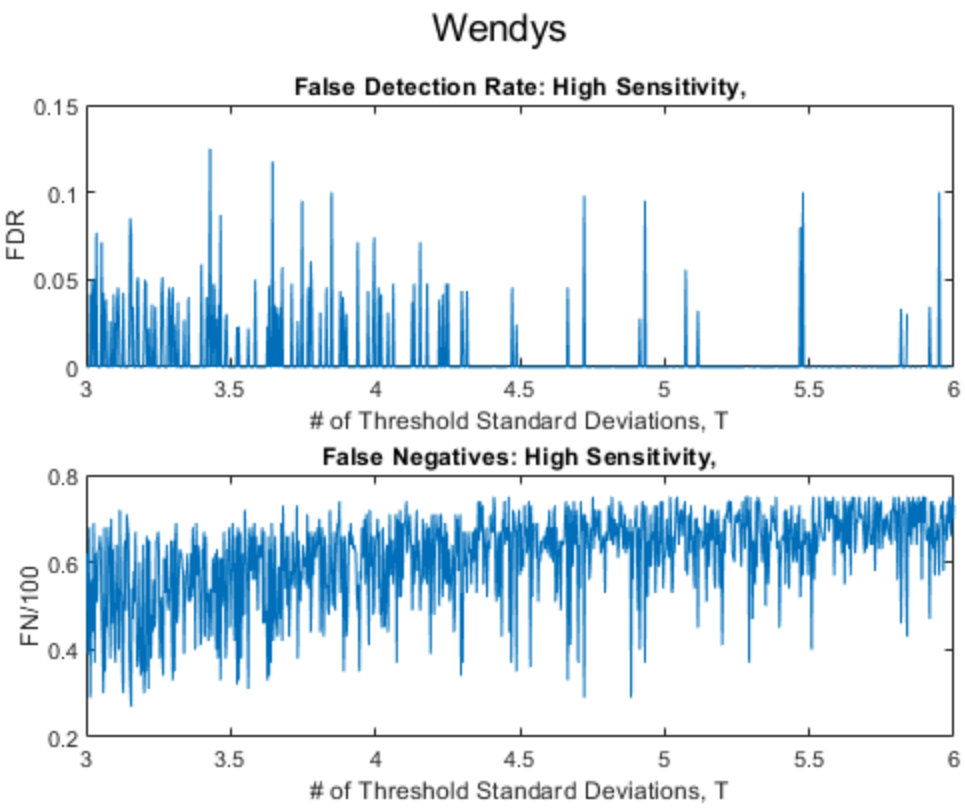
```
    new = Mediums(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end



%loud interrupts
%all activate interrupt
for c=76:100
    new = Louds(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    RI_low(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            TID_low(c,1)=1;
    else
        FN_low(c,1)=1;
    end

end
```

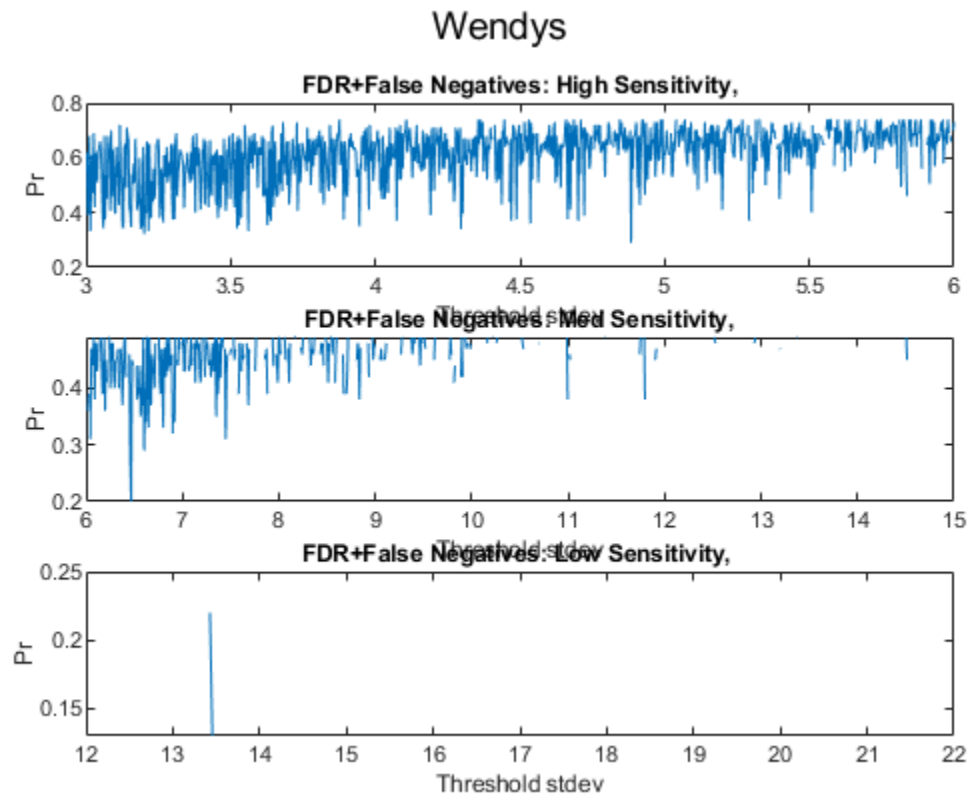
```
%False Interrupt Detection Rate - Print all
FDR_low = sum(FD_low, 'all')/sum(ID_low, 'all');    %probability of
 false interrupt given interrupt detection
FDR_med = sum(FD_med, 'all')/sum(ID_med, 'all');
FDR_high = sum(FD_high, 'all')/sum(ID_high, 'all');

FDR_low_array(s,1) = FDR_low;
FDR_med_array(s,1) = FDR_med;
FDR_high_array(s,1) = FDR_high;

FN_low_array(s,1)=sum(FN_low,'all')/100;
FN_med_array(s,1)=sum(FN_med,'all')/100;
FN_high_array(s,1)=sum(FN_high,'all')/100;

Opti_low(s,1)= FDR_low_array(s,1)+FN_low_array(s,1);
Opti_med(s,1)=FDR_med_array(s,1)+FN_med_array(s,1);
Opti_high(s,1)=FDR_high_array(s,1)+FN_high_array(s,1);


end

%!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!


DataLoc = char(DataLoc);
figure(1)
subplot(2,1,1)
plot(T_low_array,FDR_low_array)
title('False Detection Rate: Low Sensitivity, %s')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FDR')
subplot(2,1,2)
plot(T_low_array,FN_low_array)
title('False Negatives: Low Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FN')
suptitle(DataLoc)

figure(2)
subplot(2,1,1)
plot(T_med_array,FDR_med_array)
title('False Detection Rate: Medium Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FDR')
subplot(2,1,2)
plot(T_med_array,FN_med_array)
title('False Negatives: Medium Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FN/100')
suptitle(DataLoc)

figure(3)
subplot(2,1,1)
plot(T_high_array,FDR_high_array)
```

```
title('False Detection Rate: High Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FDR')
subplot(2,1,2)
plot(T_high_array,FN_high_array)
title('False Negatives: High Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FN/100')
suptitle(DataLoc)

figure(4)
subplot(3,1,1)
plot(T_high_array,Opti_high)
title('FDR+False Negatives: High Sensitivity,')
xlabel('Threshold stdev')
ylabel('Pr')
subplot(3,1,2)
plot(T_med_array,Opti_med)
title('FDR+False Negatives: Med Sensitivity,')
xlabel('Threshold stdev')
ylabel('Pr')
subplot(3,1,3)
plot(T_low_array,Opti_low)
title('FDR+False Negatives: Low Sensitivity, ')
xlabel('Threshold stdev')
ylabel('Pr')
suptitle(DataLoc)
```


Wendys
False Detection Rate: Low Sensitivity, %s
FDR
# of Threshold Standard Deviations, T
False Negatives: Low Sensitivity,
FN
# of Threshold Standard Deviations, T


Wendys
False Detection Rate: Medium Sensitivity,
FDR
# of Threshold Standard Deviations, T
False Negatives: Medium Sensitivity,
FN/100
# of Threshold Standard Deviations, T


Wendys
False Detection Rate: High Sensitivity,
FDR
# of Threshold Standard Deviations, T
False Negatives: High Sensitivity,
FN/100
# of Threshold Standard Deviations, T


Wendys
FDR+False Negatives: High Sensitivity,
Pr
FDR+False Negatives: Med Sensitivity,
Pr
FDR+False Negatives: Low Sensitivity,
Pr
Threshold stdev

Published with MATLAB® R2019b

Campus Center

```
%ECE458 - Senior Design
%Michael Benker
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% FALSE DETECTION RATES %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clf;clear all; clc; close all;

%VARIABLES
Ave2sec = 0;    %2-second average
T_high = 4;     %standard deviations for high sensitivity setting
T_med = 8;      %standard deviations for medium sensitivity setting
T_low = 12;     %standard deviations for low sensitivity setting
Data20sec = zeros(10,1);
history = zeros(100,1);

RI_med = zeros(100,1); %1 if real interrupt, 0 if false interrupt
RI_high = zeros(100,1);
RI_low = zeros(100,1);
ID_low = zeros(100,1);   %1 if interrupt detected, 0 if interrupt not
 detected
ID_med = zeros(100,1);
ID_high = zeros(100,1);
TID_low = zeros(100,1);    %1 if ID = RI = 1, 0 otherwise
TID_med = zeros(100,1);
TID_high = zeros(100,1);
FD_low = zeros(100,1);     %1 if ID = 1 & RI = 0, 0 otherwise
FD_med = zeros(100,1);
FD_high = zeros(100,1);
FN_low = zeros(100,1);     %1 if ID = 0 & RI = 1, 0 otherwise
FN_med = zeros(100,1);
FN_high = zeros(100,1);
FDR_low = 0;    %probability of false interrupt given interrupt
 detection
FDR_med = 0;
FDR_high = 0;

%IMPORT DATA
SoundData1 = 'Book2.xlsx'; %Read excel file in folder
DataMat = zeros(30,4); %Predefine Data Matrix
Ambients = xlsread(SoundData1, 'A2:A31'); %Ambient 1st col
Quiets = xlsread(SoundData1, 'B2:B31'); %Quiet is 2nd col
Mediums = xlsread(SoundData1, 'C2:C31'); %Medium is 3rd col
Louds = xlsread(SoundData1, 'D2:D31'); %Loud is 4th col
[k,DataLoc] = xlsread(SoundData1, 'E1:E1');
DataLoc
%Define past 20 seconds (ambients)

for c =1:10
    Data20sec(c,1) = Ambients(randi([1 30],1,1),1);
    history(c,1)=Data20sec(c,1);
end
Ave20sec = mean(Data20sec)
```

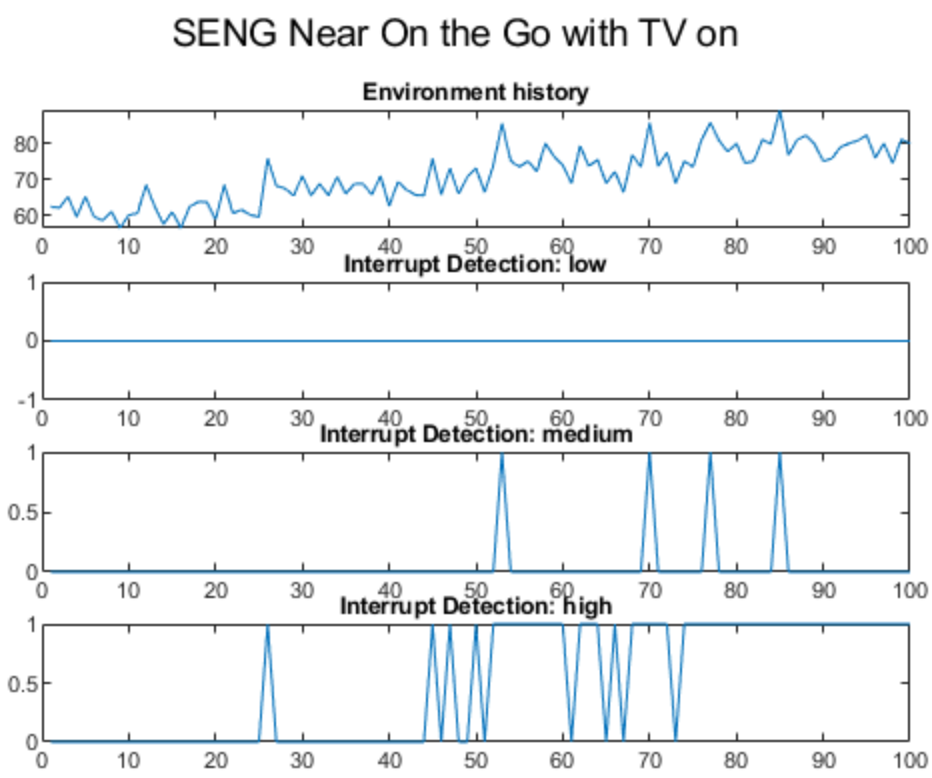
```
Std20sec = std(Data20sec)
RT_high = Ave20sec+Std20sec*T_high %Running threshold level (high
 sens)
RT_med = Ave20sec+Std20sec*T_med %Running threshold level (med sens)
RT_low = Ave20sec+Std20sec*T_low %Running threshold level (low sens)

for c=11:25
    new = Ambients(randi([1 30],1,1),1);
    history(c,1)=new;

    if new>RT_high
            ID_high(c,1)=1;
            FD_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%Quiet interrupts
%Only high sensitivity should activate interrupt
for c=26:50
    new = Quiets(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%medium interrupts
%Only high and medium sensitivity should activate interrupt

for c=51:75
    new = Mediums(randi([1 30],1,1),1);
    history(c,1)=new;
```

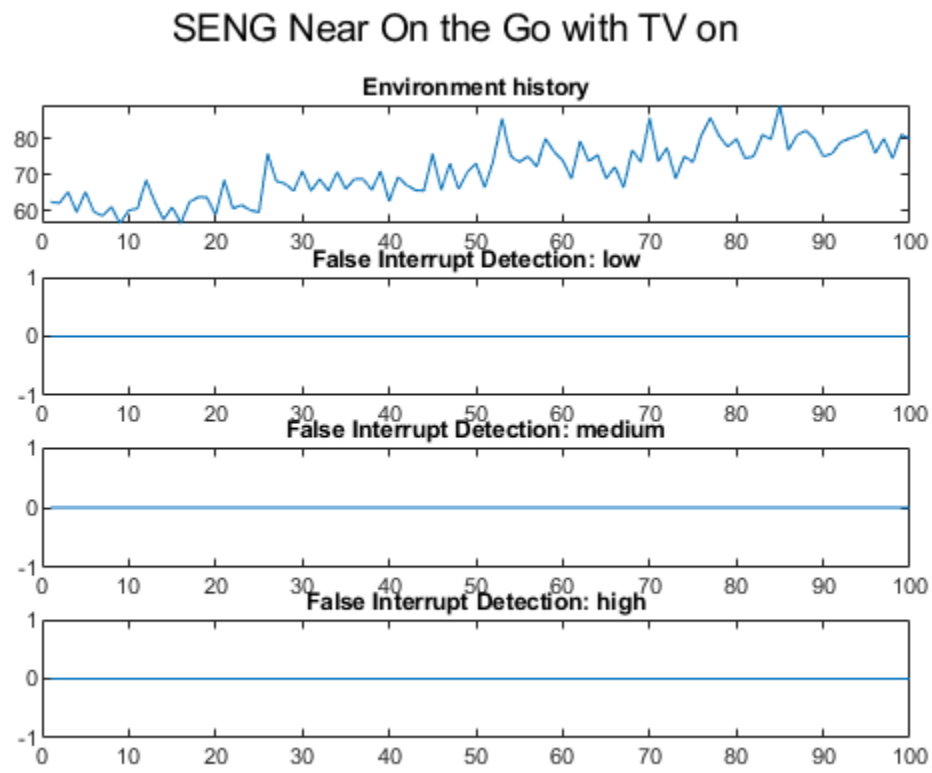
```
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end



%loud interrupts
%all activate interrupt
for c=76:100
    new = Louds(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    RI_low(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            TID_low(c,1)=1;
    else
        FN_low(c,1)=1;
    end

end

%False Interrupt Detection Rate - Print all
```

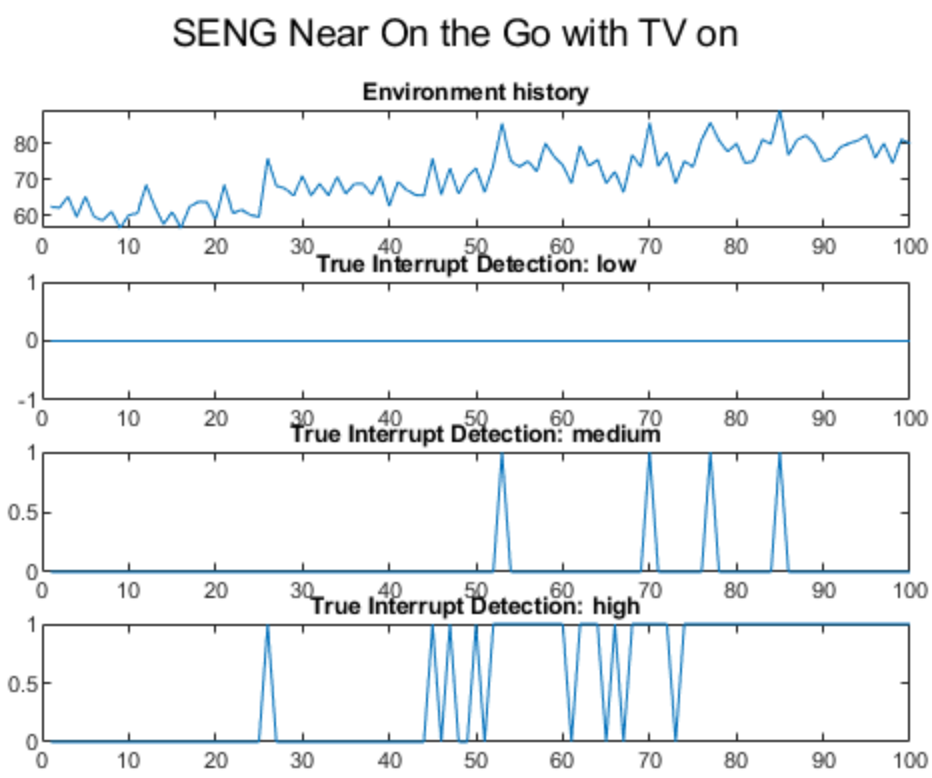
```
FDR_low = sum(FD_low, 'all')/sum(ID_low, 'all')    %probability of
 false interrupt given interrupt detection
FDR_med = sum(FD_med, 'all')/sum(ID_med, 'all')
FDR_high = sum(FD_high, 'all')/sum(ID_high, 'all')
```

```
figure(1)
subplot(4,1,1)
plot(history)
title('Environment history')
subplot(4,1,2)
plot(ID_low)
title('Interrupt Detection: low')
subplot(4,1,3)
plot(ID_med)
title('Interrupt Detection: medium')
subplot(4,1,4)
plot(ID_high)
title('Interrupt Detection: high')
suptitle(DataLoc)

figure(2)
subplot(4,1,1)
plot(history)
title('Environment history')
subplot(4,1,2)
plot(FD_low)
title('False Interrupt Detection: low')
subplot(4,1,3)
plot(FD_med)
title('False Interrupt Detection: medium')
subplot(4,1,4)
plot(FD_high)
title('False Interrupt Detection: high')
suptitle(DataLoc)

figure(3)
subplot(4,1,1)
plot(history)
title('Environment history')
subplot(4,1,2)
plot(TID_low)
title('True Interrupt Detection: low')
subplot(4,1,3)
plot(TID_med)
title('True Interrupt Detection: medium')
subplot(4,1,4)
plot(TID_high)
title('True Interrupt Detection: high')
suptitle(DataLoc)
```

```
DataLoc =

  1×1 cell array

    {'SENG Near On the Go with TV on'}

Ave20sec =

   61.1300

Std20sec =

    2.7725

RT_high =

   72.2200

RT_med =

   83.3100

RT_low =

   94.4000

FDR_low =

   NaN

FDR_med =

     0

FDR_high =

     0
```


SENG Near On the Go with TV on
Environment history
Interrupt Detection: low
Interrupt Detection: medium
Interrupt Detection: high


SENG Near On the Go with TV on
Environment history
False Interrupt Detection: low
False Interrupt Detection: medium
False Interrupt Detection: high



Published with MATLAB® R2019b

```
%ECE458 - Senior Design
%Michael Benker
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% FALSE DETECTION RATES %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clf;clear all; clc; close all;

samples=1000;
T_h_begin = 3; %Begin and end for parameter sweep of threshold numbers
T_h_end = 6;
T_m_begin = 6;
T_m_end = 15;
T_l_begin = 12;
T_l_end = 40;


%VARIABLES
Ave2sec = 0;    %2-second average
T_high = 4;     %standard deviations for high sensitivity setting
T_med = 8;      %standard deviations for medium sensitivity setting
T_low = 12;     %standard deviations for low sensitivity setting
Data20sec = zeros(10,1);
history = zeros(100,1);

RI_med = zeros(100,1); %1 if real interrupt, 0 if false interrupt
RI_high = zeros(100,1);
RI_low = zeros(100,1);
ID_low = zeros(100,1);   %1 if interrupt detected, 0 if interrupt not
 detected
ID_med = zeros(100,1);
ID_high = zeros(100,1);
TID_low = zeros(100,1);    %1 if ID = RI = 1, 0 otherwise
TID_med = zeros(100,1);
TID_high = zeros(100,1);
FD_low = zeros(100,1);     %1 if ID = 1 & RI = 0, 0 otherwise
FD_med = zeros(100,1);
FD_high = zeros(100,1);
FN_low = zeros(100,1);     %1 if ID = 0 & RI = 1, 0 otherwise
FN_med = zeros(100,1);
FN_high = zeros(100,1);
FDR_low = 0;    %probability of false interrupt given interrupt
 detection
FDR_med = 0;
FDR_high = 0;
FDR_low_array = zeros(samples,1);
FDR_med_array = zeros(samples,1);
FDR_high_array = zeros(samples,1);
T_high_array = zeros(samples,1);     %standard deviations for high
 sensitivity setting
T_med_array = zeros(samples,1);      %standard deviations for medium
 sensitivity setting
```

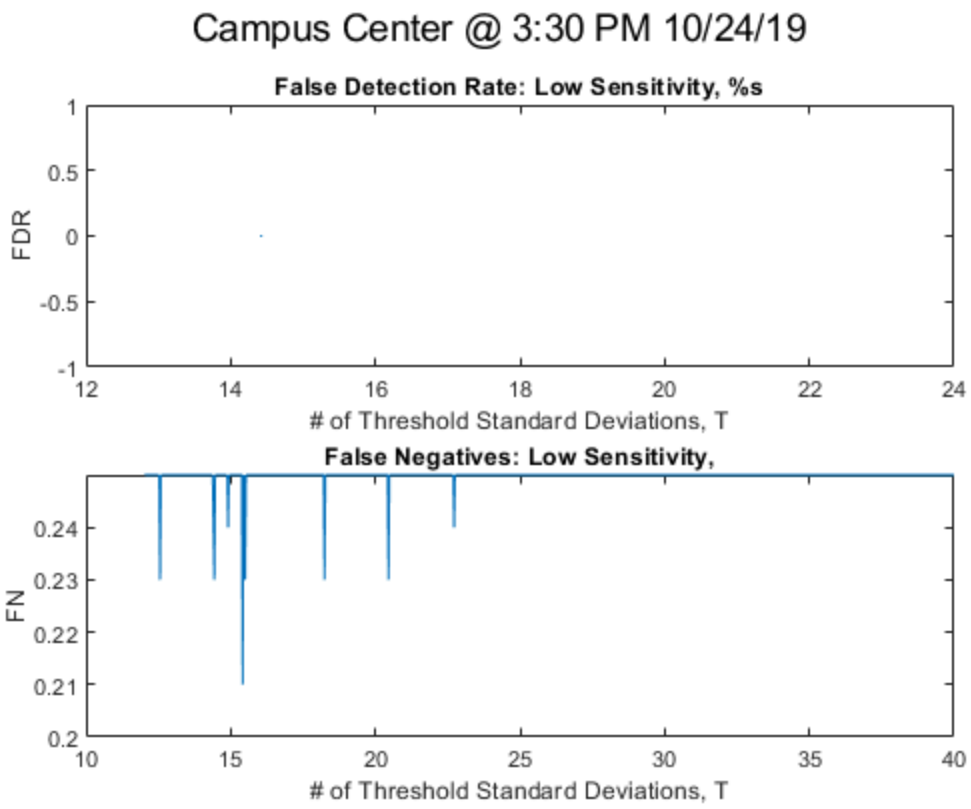
```
T_low_array = zeros(samples,1);     %standard deviations for low
 sensitivity setting
FN_low_array = zeros(samples,1);
FN_med_array = zeros(samples,1);
FN_high_array = zeros(samples,1);
Opti_low = zeros(samples,1);
Opti_med = zeros(samples,1);
Opti_high = zeros(samples,1);


%IMPORT DATA
SoundData1 = 'Book4.xlsx'; %Read excel file in folder
DataMat = zeros(30,4); %Predefine Data Matrix
Ambients = xlsread(SoundData1, 'A2:A31'); %Ambient 1st col
Quiets = xlsread(SoundData1, 'B2:B31'); %Quiet is 2nd col
Mediums = xlsread(SoundData1, 'C2:C31'); %Medium is 3rd col
Louds = xlsread(SoundData1, 'D2:D31'); %Loud is 4th col
[k,DataLoc] = xlsread(SoundData1, 'E1:E1');

%Define past 20 seconds (ambients)


for s=1:samples
    T_high=T_h_begin+s*(T_h_end-T_h_begin)/samples;
    T_med=T_m_begin+s*(T_m_end-T_m_begin)/samples;
    T_low=T_l_begin+s*(T_l_end-T_l_begin)/samples;

    T_high_array(s,1)=T_high;
    T_med_array(s,1)=T_med;
    T_low_array(s,1)=T_low;

    RI_med = zeros(100,1); %1 if real interrupt, 0 if false interrupt
    RI_high = zeros(100,1);
    RI_low = zeros(100,1);
    ID_low = zeros(100,1);   %1 if interrupt detected, 0 if interrupt
 not detected
    ID_med = zeros(100,1);
    ID_high = zeros(100,1);
    TID_low = zeros(100,1);    %1 if ID = RI = 1, 0 otherwise
    TID_med = zeros(100,1);
    TID_high = zeros(100,1);
    FD_low = zeros(100,1);     %1 if ID = 1 & RI = 0, 0 otherwise
    FD_med = zeros(100,1);
    FD_high = zeros(100,1);
    FN_low = zeros(100,1);     %1 if ID = 0 & RI = 1, 0 otherwise
    FN_med = zeros(100,1);
    FN_high = zeros(100,1);



for c =1:10
    Data20sec(c,1) = Ambients(randi([1 30],1,1),1);
    history(c,1)=Data20sec(c,1);
```

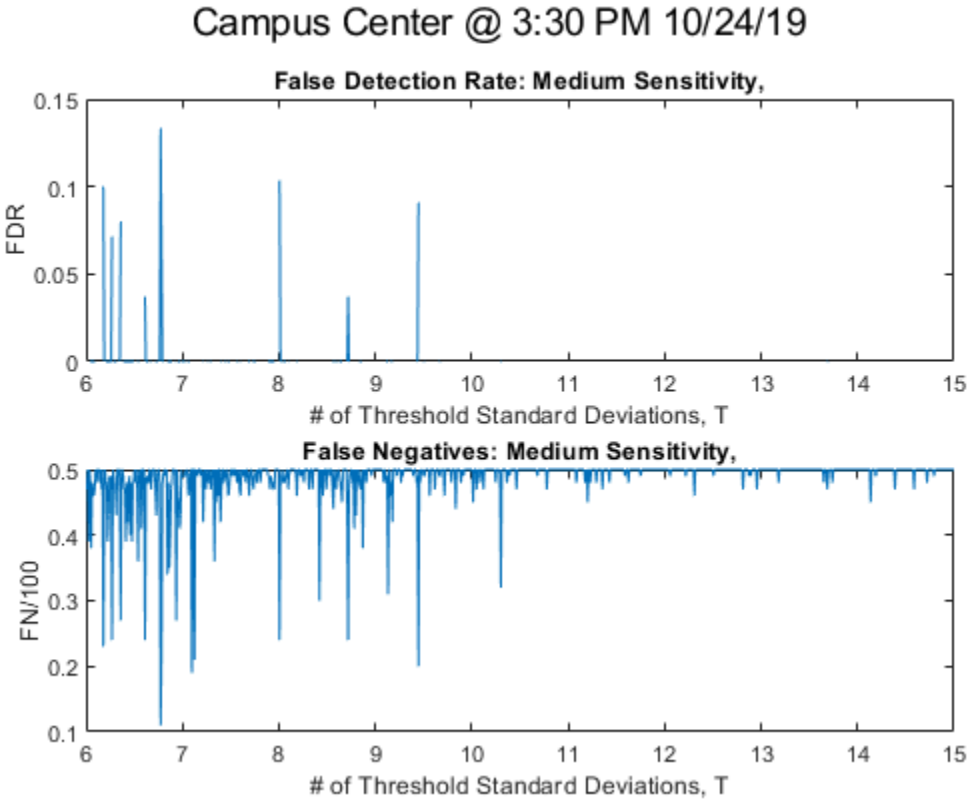
```
end
Ave20sec = mean(Data20sec);
Std20sec = std(Data20sec);
RT_high = Ave20sec+Std20sec*T_high; %Running threshold level (high
 sens)
RT_med = Ave20sec+Std20sec*T_med; %Running threshold level (med sens)
RT_low = Ave20sec+Std20sec*T_low; %Running threshold level (low sens)

for c=11:25
    new = Ambients(randi([1 30],1,1),1);
    history(c,1)=new;

    if new>RT_high
            ID_high(c,1)=1;
            FD_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%Quiet interrupts
%Only high sensitivity should activate interrupt
for c=26:50
    new = Quiets(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%medium interrupts
%Only high and medium sensitivity should activate interrupt

for c=51:75
```

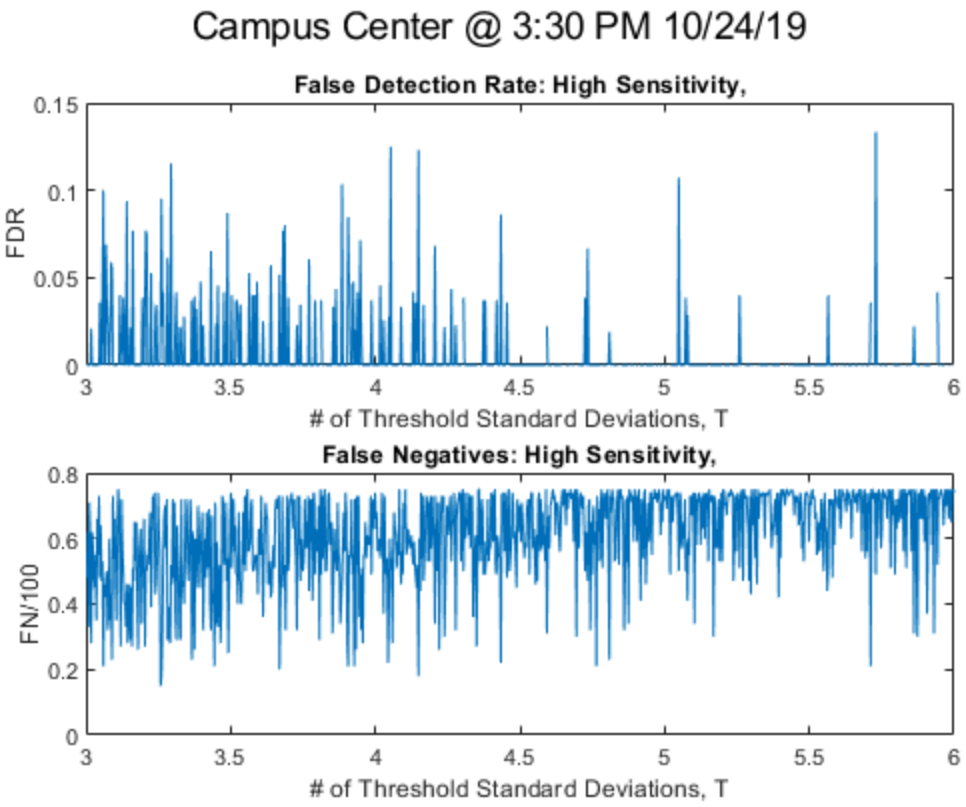
```
    new = Mediums(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end



%loud interrupts
%all activate interrupt
for c=76:100
    new = Louds(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    RI_low(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            TID_low(c,1)=1;
    else
        FN_low(c,1)=1;
    end

end
```

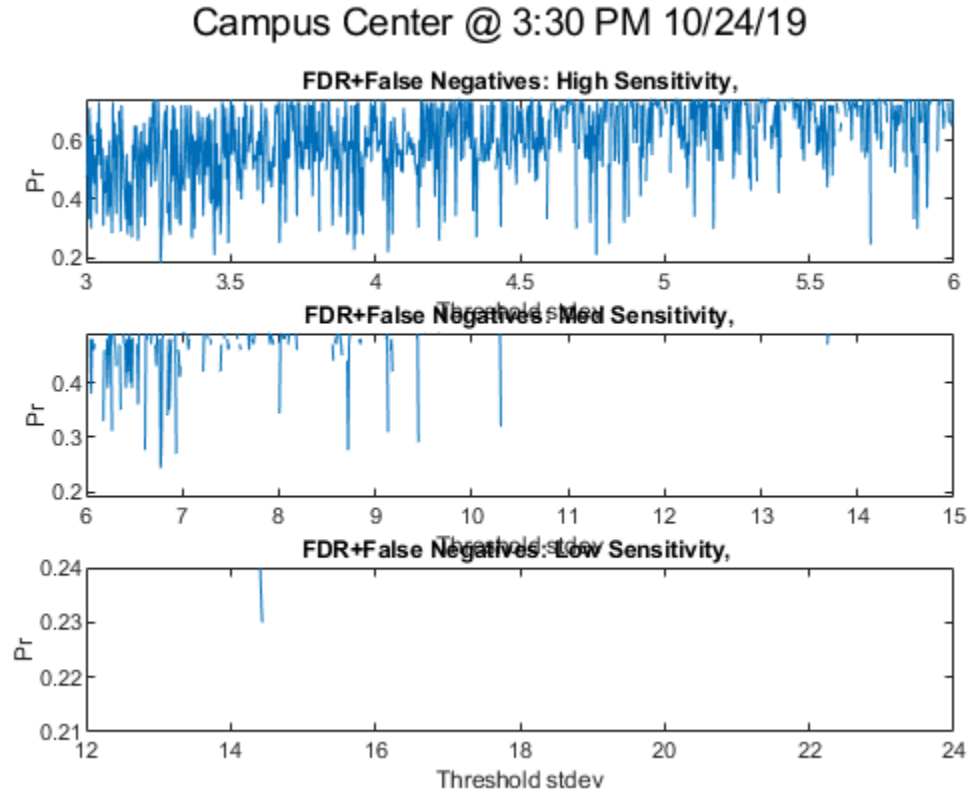
```
%False Interrupt Detection Rate - Print all
FDR_low = sum(FD_low, 'all')/sum(ID_low, 'all');    %probability of
 false interrupt given interrupt detection
FDR_med = sum(FD_med, 'all')/sum(ID_med, 'all');
FDR_high = sum(FD_high, 'all')/sum(ID_high, 'all');

FDR_low_array(s,1) = FDR_low;
FDR_med_array(s,1) = FDR_med;
FDR_high_array(s,1) = FDR_high;

FN_low_array(s,1)=sum(FN_low,'all')/100;
FN_med_array(s,1)=sum(FN_med,'all')/100;
FN_high_array(s,1)=sum(FN_high,'all')/100;

Opti_low(s,1)= FDR_low_array(s,1)+FN_low_array(s,1);
Opti_med(s,1)=FDR_med_array(s,1)+FN_med_array(s,1);
Opti_high(s,1)=FDR_high_array(s,1)+FN_high_array(s,1);


end

%!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!


DataLoc = char(DataLoc);
figure(1)
subplot(2,1,1)
plot(T_low_array,FDR_low_array)
title('False Detection Rate: Low Sensitivity, %s')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FDR')
subplot(2,1,2)
plot(T_low_array,FN_low_array)
title('False Negatives: Low Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FN')
suptitle(DataLoc)

figure(2)
subplot(2,1,1)
plot(T_med_array,FDR_med_array)
title('False Detection Rate: Medium Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FDR')
subplot(2,1,2)
plot(T_med_array,FN_med_array)
title('False Negatives: Medium Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FN/100')
suptitle(DataLoc)

figure(3)
subplot(2,1,1)
plot(T_high_array,FDR_high_array)
```

```
title('False Detection Rate: High Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FDR')
subplot(2,1,2)
plot(T_high_array,FN_high_array)
title('False Negatives: High Sensitivity,')
xlabel('# of Threshold Standard Deviations, T')
ylabel('FN/100')
suptitle(DataLoc)

figure(4)
subplot(3,1,1)
plot(T_high_array,Opti_high)
title('FDR+False Negatives: High Sensitivity,')
xlabel('Threshold stdev')
ylabel('Pr')
subplot(3,1,2)
plot(T_med_array,Opti_med)
title('FDR+False Negatives: Med Sensitivity,')
xlabel('Threshold stdev')
ylabel('Pr')
subplot(3,1,3)
plot(T_low_array,Opti_low)
title('FDR+False Negatives: Low Sensitivity, ')
xlabel('Threshold stdev')
ylabel('Pr')
suptitle(DataLoc)
```


Campus Center @ 3:30 PM 10/24/19
False Detection Rate: Low Sensitivity, %s
FDR
# of Threshold Standard Deviations, T
False Negatives: Low Sensitivity,
FN
# of Threshold Standard Deviations, T


Campus Center @ 3:30 PM 10/24/19
False Detection Rate: Medium Sensitivity,
FDR
# of Threshold Standard Deviations, T
False Negatives: Medium Sensitivity,
FN/100
# of Threshold Standard Deviations, T


Campus Center @ 3:30 PM 10/24/19
False Detection Rate: High Sensitivity,
FDR
# of Threshold Standard Deviations, T
False Negatives: High Sensitivity,
FN/100
# of Threshold Standard Deviations, T


Campus Center @ 3:30 PM 10/24/19
FDR+False Negatives: High Sensitivity,
Pr
FDR+False Negatives: Med Sensitivity,
Pr
FDR+False Negatives: Low Sensitivity,
Pr
Threshold stdev

Published with MATLAB® R2019b

On the Go (SENG)

```
%ECE458 - Senior Design
%Michael Benker
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% FALSE DETECTION RATES %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clf;clear all; clc; close all;

%VARIABLES
Ave2sec = 0;    %2-second average
T_high = 4;     %standard deviations for high sensitivity setting
T_med = 8;      %standard deviations for medium sensitivity setting
T_low = 12;     %standard deviations for low sensitivity setting
Data20sec = zeros(10,1);
history = zeros(100,1);

RI_med = zeros(100,1); %1 if real interrupt, 0 if false interrupt
RI_high = zeros(100,1);
RI_low = zeros(100,1);
ID_low = zeros(100,1);   %1 if interrupt detected, 0 if interrupt not
 detected
ID_med = zeros(100,1);
ID_high = zeros(100,1);
TID_low = zeros(100,1);    %1 if ID = RI = 1, 0 otherwise
TID_med = zeros(100,1);
TID_high = zeros(100,1);
FD_low = zeros(100,1);     %1 if ID = 1 & RI = 0, 0 otherwise
FD_med = zeros(100,1);
FD_high = zeros(100,1);
FN_low = zeros(100,1);     %1 if ID = 0 & RI = 1, 0 otherwise
FN_med = zeros(100,1);
FN_high = zeros(100,1);
FDR_low = 0;    %probability of false interrupt given interrupt
 detection
FDR_med = 0;
FDR_high = 0;

%IMPORT DATA
SoundData1 = 'Book2.xlsx'; %Read excel file in folder
DataMat = zeros(30,4); %Predefine Data Matrix
Ambients = xlsread(SoundData1, 'A2:A31'); %Ambient 1st col
Quiets = xlsread(SoundData1, 'B2:B31'); %Quiet is 2nd col
Mediums = xlsread(SoundData1, 'C2:C31'); %Medium is 3rd col
Louds = xlsread(SoundData1, 'D2:D31'); %Loud is 4th col
[k,DataLoc] = xlsread(SoundData1, 'E1:E1');
DataLoc
%Define past 20 seconds (ambients)

for c =1:10
    Data20sec(c,1) = Ambients(randi([1 30],1,1),1);
    history(c,1)=Data20sec(c,1);
end
Ave20sec = mean(Data20sec)
```

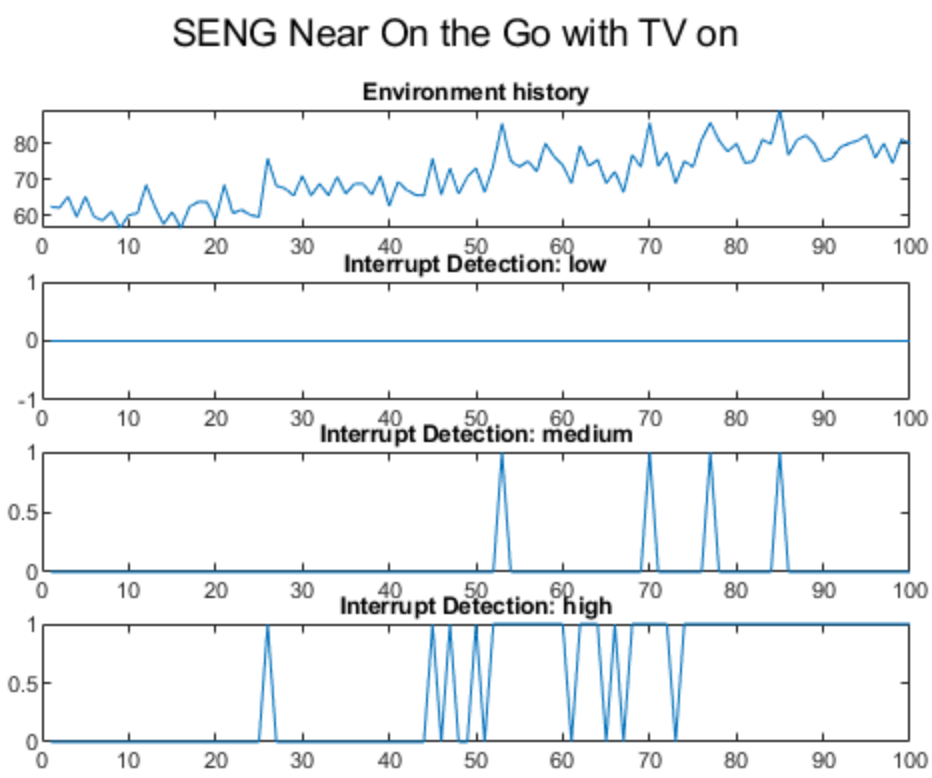
```
Std20sec = std(Data20sec)
RT_high = Ave20sec+Std20sec*T_high %Running threshold level (high
 sens)
RT_med = Ave20sec+Std20sec*T_med %Running threshold level (med sens)
RT_low = Ave20sec+Std20sec*T_low %Running threshold level (low sens)

for c=11:25
    new = Ambients(randi([1 30],1,1),1);
    history(c,1)=new;

    if new>RT_high
            ID_high(c,1)=1;
            FD_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%Quiet interrupts
%Only high sensitivity should activate interrupt
for c=26:50
    new = Quiets(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            FD_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end

%medium interrupts
%Only high and medium sensitivity should activate interrupt

for c=51:75
    new = Mediums(randi([1 30],1,1),1);
    history(c,1)=new;
```

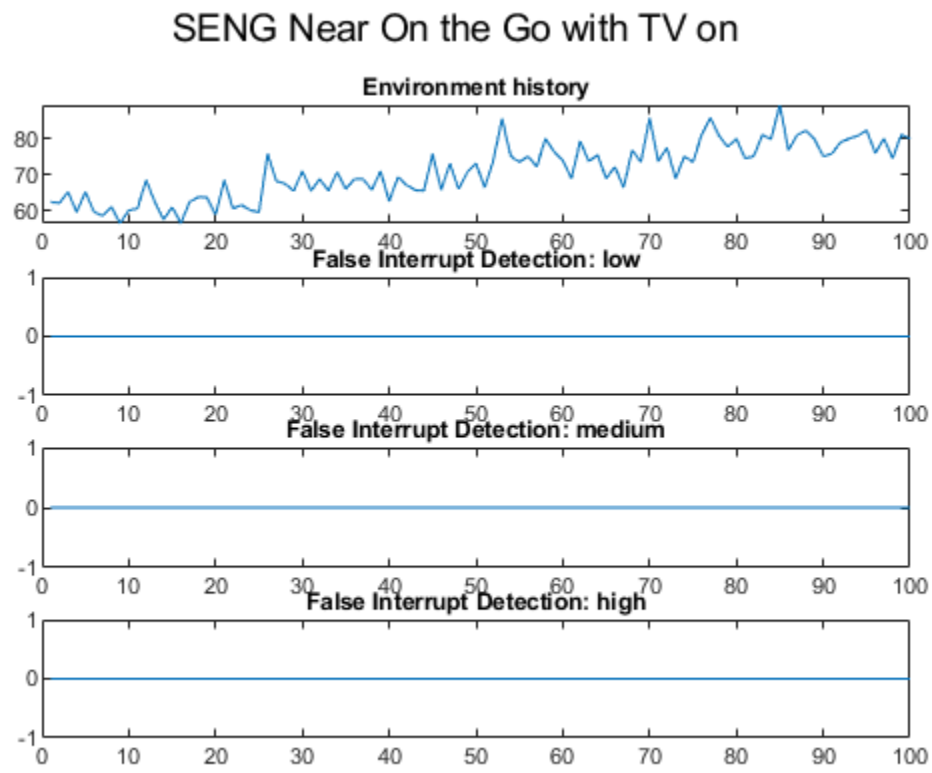
```
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            FD_low(c,1)=1;
    end

end



%loud interrupts
%all activate interrupt
for c=76:100
    new = Louds(randi([1 30],1,1),1);
    history(c,1)=new;
    RI_high(c,1)=1;
    RI_med(c,1)=1;
    RI_low(c,1)=1;
    if new>RT_high
            ID_high(c,1)=1;
            TID_high(c,1)=1;
    else
        FN_high(c,1)=1;
    end
    if new>RT_med
            ID_med(c,1)=1;
            TID_med(c,1)=1;
    else
        FN_med(c,1)=1;
    end
    if new>RT_low
            ID_low(c,1)=1;
            TID_low(c,1)=1;
    else
        FN_low(c,1)=1;
    end

end

%False Interrupt Detection Rate - Print all
```

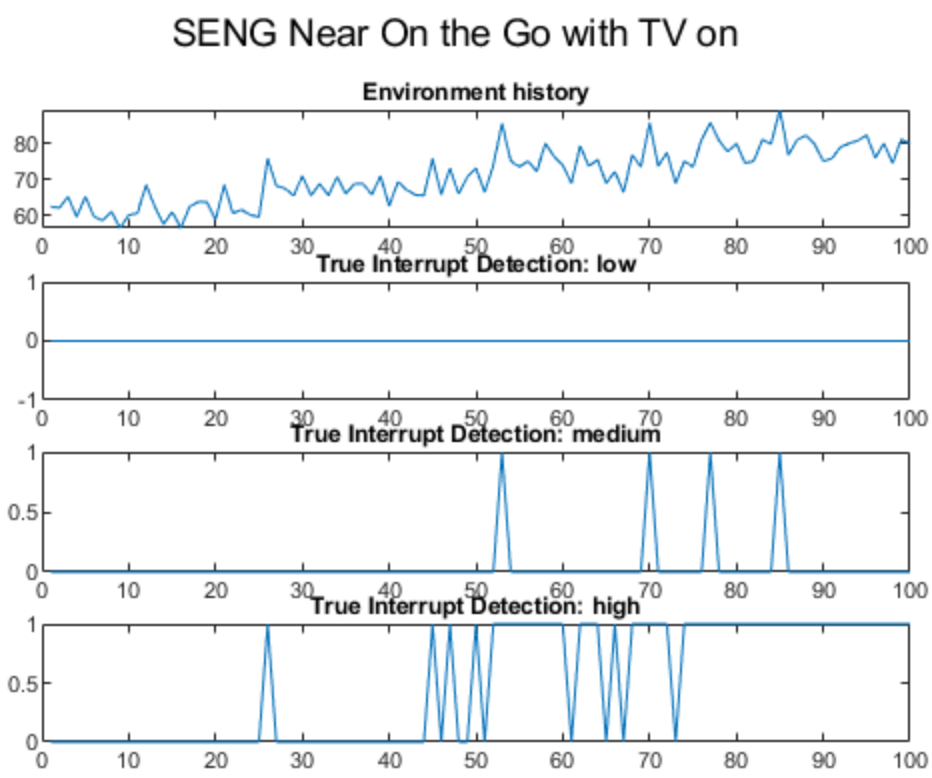
```
FDR_low = sum(FD_low, 'all')/sum(ID_low, 'all')    %probability of
 false interrupt given interrupt detection
FDR_med = sum(FD_med, 'all')/sum(ID_med, 'all')
FDR_high = sum(FD_high, 'all')/sum(ID_high, 'all')
```

```
figure(1)
subplot(4,1,1)
plot(history)
title('Environment history')
subplot(4,1,2)
plot(ID_low)
title('Interrupt Detection: low')
subplot(4,1,3)
plot(ID_med)
title('Interrupt Detection: medium')
subplot(4,1,4)
plot(ID_high)
title('Interrupt Detection: high')
suptitle(DataLoc)

figure(2)
subplot(4,1,1)
plot(history)
title('Environment history')
subplot(4,1,2)
plot(FD_low)
title('False Interrupt Detection: low')
subplot(4,1,3)
plot(FD_med)
title('False Interrupt Detection: medium')
subplot(4,1,4)
plot(FD_high)
title('False Interrupt Detection: high')
suptitle(DataLoc)

figure(3)
subplot(4,1,1)
plot(history)
title('Environment history')
subplot(4,1,2)
plot(TID_low)
title('True Interrupt Detection: low')
subplot(4,1,3)
plot(TID_med)
title('True Interrupt Detection: medium')
subplot(4,1,4)
plot(TID_high)
title('True Interrupt Detection: high')
suptitle(DataLoc)
```

```
DataLoc =

  1×1 cell array

    {'SENG Near On the Go with TV on'}

Ave20sec =

   61.1300

Std20sec =

    2.7725

RT_high =

   72.2200

RT_med =

   83.3100

RT_low =

   94.4000

FDR_low =

   NaN

FDR_med =

     0

FDR_high =

     0
```


SENG Near On the Go with TV on
Environment history
Interrupt Detection: low
Interrupt Detection: medium
Interrupt Detection: high


SENG Near On the Go with TV on
Environment history
False Interrupt Detection: low
False Interrupt Detection: medium
False Interrupt Detection: high



Published with MATLAB® R2019b