Kelsey Nguyen

Matlab 4

Dec 7, 2017

**ECE 352: Final Project**

The purpose of this matlab is to simulate a radar detection system that has additive Gaussian(0,1) noise. The internal circuitry of the receiver introduces a “squared distortion” non-linearity in the received signal. The signal measured in volts is given by:

The were plotted as a function of the receiver threshold voltage , which is in between -2 and 6 for two transmit voltages. The target voltage is and three square distortion values are . To do so, a function hist that will return 1000 values of the variable X and store them as outputs. From figure 1 and figure 2, it shows that +=1 for any given value of voltage and distortion. Also, the and are not affected too much by the distortions. Figure 1 shows the distortion affects the the same way when voltage is 1 and voltage is 2. It increases in the beginning but then stay constant later on. Distortion also affects the . The distortion increases the voltage is decreasing when the voltage changes from 1 to 2. It misses less when the distortion increases. The code is shown below in %Question 1.

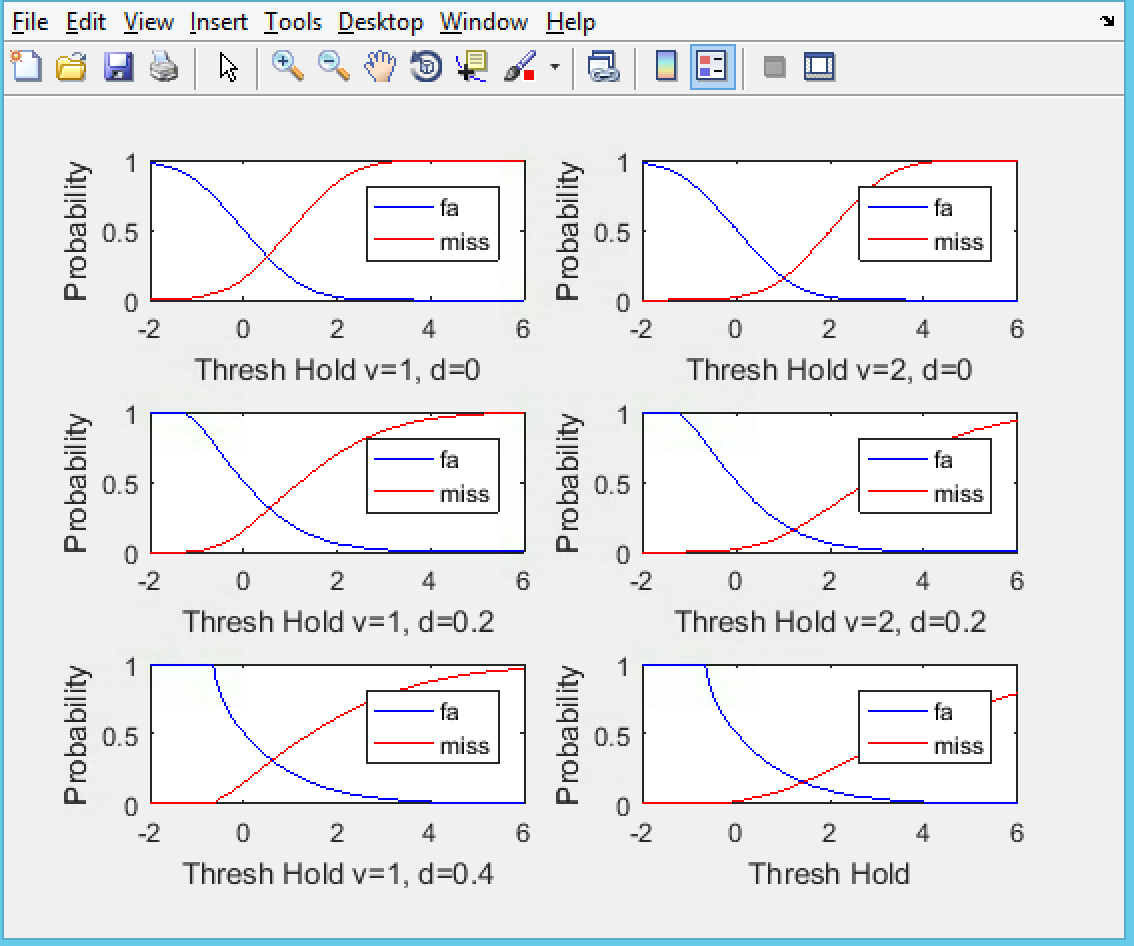


Figure 1: and results

The ROC was plotted for both transmit voltages and the three square distortion by set the probability of the miss over the probability of the false alarm. Looking at the results, the distortion did not change much. Figure 2 shows the rate rising decrease with presence of distortion and increase in function of voltages. The ROC improve as voltage increases. The code is shown below in %Question 2.

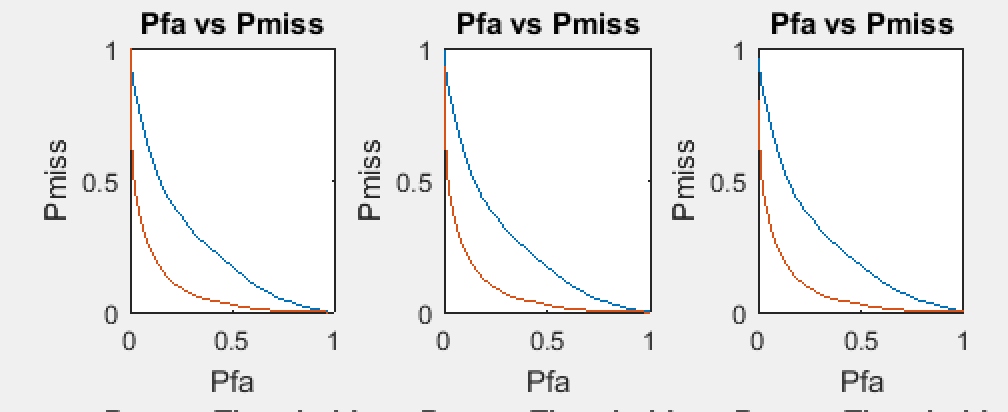


Figure 2: ROC results

The probability that the target is absent is , which is 0.8 and the probability that the target is present is . The maximum Posteriori Probability Test was used to compute . It was plotted against the threshold voltage for the transmit voltages and the three square distortion values in the previous problem. Figure 3 shows the distortions affects increase the probability of error. The percent error seems to be decreasing. The code is shown below in %Question 3.

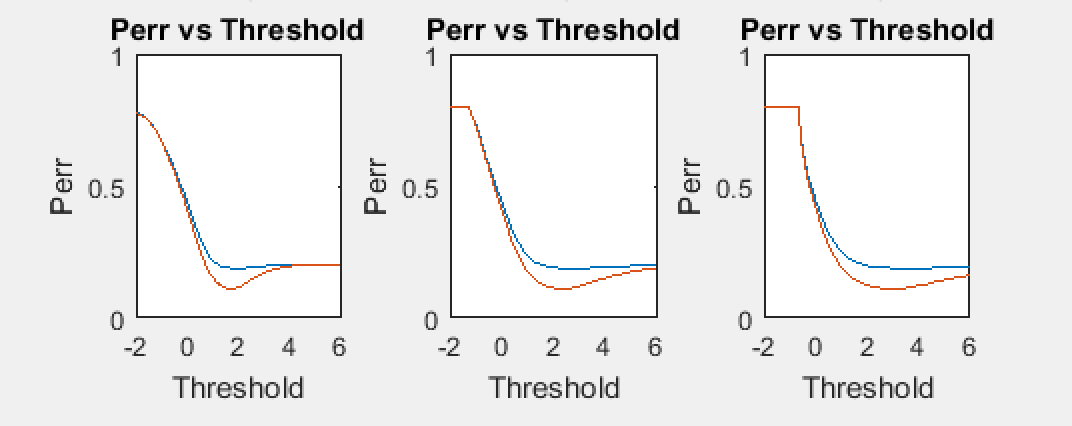
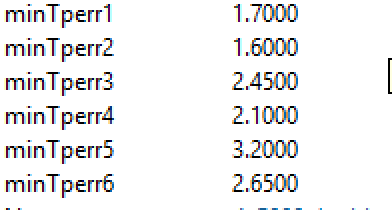
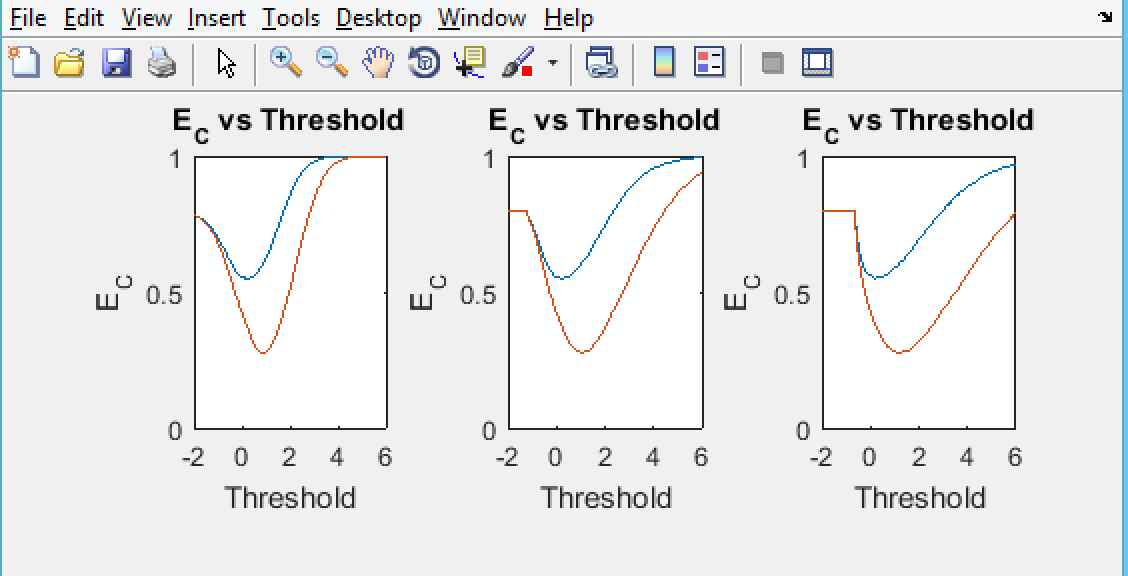


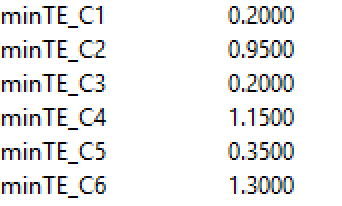
Figure 3:results



Threshold Voltage Minimized

The cost of the false alarm is 1 and the cost of a miss is 5. The expected cost of test errors was calculated by the minimum cost test equation for both transmit voltages and the three square distortion values in the previous problem. Figure 4shows the rate rising decrease with presence of distortion and increase in function of voltages. The curves of E[C] is much differ from . The voltages increase along with distortion in Figure 3 but in Figure 4 as the distortion increases the voltages decrease. The minimum values show that as d increase the values increases.The code is shown below in %Question 4.

Figure 4: results



Threshold Voltage minimized

The methodology for this project was to break thing down and hard code it. There was multiply way to attack the problem such as put it in a big for loop. The method was chosen to solve it was hard code it instead. It is time more time consuming to have to repeatedly copy and paste for every single one and change the variable for it, but it was the easiest thing to do. The X formula was hard coded by plug in d and v. The for loop is needed so it just can go through every single one. Then use the binary hypothesis formulas to find what the question asked for. It was still hard code by pluging in the given values instead of declaring it on top. The code is really repetitive.

Overall, this project was really challenging. The graphs look reasonable, so it was an assumption that the problems were solved correctly. The first problem was the most difficult problem to solve. This helped organize the ideas and made it easier to trouble shoot for problems. The rest was just applying the equation. There was a lot of research into the syntax and formula. There was some part that was really tough such as plot the ROC because there are so many syntaxes for it. The matlab website really helped a lot with the syntaxes. It gives full descriptions for the functions and a couple of examples. This project requires a decent amount of MATLAB skills and knowledge of the lectures to solve those problems. It gave students an opportunity to learn more and apply the knowledge of binary testing hypothesis.

**%Question 1**

**%Gaussian (Noise)**

**mean =0;**

**variance = 1;**

**numberPoints = 5000;**

**T = -2:.05:6**

**N = randn(1, numberPoints);**

**%distortion d= 0, voltage v = 1**

**X\_A1=( (N) + 0 \*(N.^2)); %Target Absent**

**P\_fa1=sum(X\_A1>T(1))/numberPoints;**

**for i=2 : numel(T)**

**P\_fa1=[P\_fa1,sum(X\_A1>T(i))/numberPoints];**

**end**

**X\_P1=( (1+N) + 0 \*((1+N).^2)); %target Presence**

**P\_miss1=sum(X\_P1 < T(1))/numberPoints;**

**for i=2 : numel(T)**

**P\_miss1=[P\_miss1,sum(X\_P1<T(i))/numberPoints];**

**end**

**%X\_P1= normcdf(T,1,variance)+0\*(normcdf(T,1,variance).^2); %Pfa**

**figure (1)**

**subplot(3,2,1)**

**plot(T,P\_fa1,'b');**

**hold on;**

**plot(T,P\_miss1, 'r');**

**xlabel ('Thresh Hold v=1, d=0')**

**ylabel ('Probability');**

**legend ('fa','miss')**

**%distortion d= 0, voltage v = 2**

**%X\_A2=(1- N) + 0 \*(1-N).^2;**

**%X\_P2= normcdf(T,2,variance)+0\*(normcdf(T,2,variance).^2);**

**%distortion d= 0, voltage v = 2**

**X\_A2=( (N) + 0 \*(N.^2)); %Target Absent**

**P\_fa2=sum(X\_A1>T(1))/numberPoints;**

**for i=2 : numel(T)**

**P\_fa2=[P\_fa2,sum(X\_A1>T(i))/numberPoints];**

**end**

**X\_P2=( (2+N) + 0 \*((2+N).^2)); %target Presence**

**P\_miss2=sum(X\_P2 < T(1))/numberPoints;**

**for i=2 : numel(T)**

**P\_miss2=[P\_miss2,sum(X\_P2<T(i))/numberPoints];**

**end**

**subplot(3,2,2)**

**plot(T,P\_fa2,'b');**

**hold on;**

**plot(T,P\_miss2, 'r');**

**xlabel ('Thresh Hold v=2, d=0')**

**ylabel ('Probability');**

**legend ('fa','miss')**

**%distortion d= 0.2, voltage v = 1**

**%X\_A3=(1- N) + .2 \*(1-N).^2;**

**%X\_P3= normcdf(T,1,variance)+.2\*(normcdf(T,1,variance).^2);**

**%distortion d= 0.2, voltage v = 1**

**X\_A3=( (N) + 0.2 \*(N.^2)); %Target Absent**

**P\_fa3=sum(X\_A3>T(1))/numberPoints;**

**for i=2 : numel(T)**

**P\_fa3=[P\_fa3,sum(X\_A3>T(i))/numberPoints];**

**end**

**X\_P3=( (1+N) + 0.2\*((1+N).^2)); %target Presence**

**P\_miss3=sum(X\_P3 < T(1))/numberPoints;**

**for i=2 : numel(T)**

**P\_miss3=[P\_miss3,sum(X\_P3<T(i))/numberPoints];**

**end**

**subplot(3,2,3)**

**plot(T,P\_fa3,'b');**

**hold on;**

**plot(T,P\_miss3, 'r');**

**xlabel ('Thresh Hold v=1, d=0.2')**

**ylabel ('Probability');**

**legend ('fa','miss')**

**%distortion d= 0.2, voltage v = 2**

**%X\_A4=(1- N) + .2 \*(1-N).^2;**

**%X\_P4= normcdf(T,2,variance)+.2\*(normcdf(T,2,variance).^2);**

**X\_A4=( (N) + 0.2 \*(N.^2)); %Target Absent**

**P\_fa4=sum(X\_A4>T(1))/numberPoints;**

**for i=2 : numel(T)**

**P\_fa4=[P\_fa4,sum(X\_A4>T(i))/numberPoints];**

**end**

**X\_P4=( (2+N) + 0.2 \*((2+N).^2)); %target Presence**

**P\_miss4=sum(X\_P4 < T(1))/numberPoints;**

**for i=2 : numel(T)**

**P\_miss4=[P\_miss4,sum(X\_P4<T(i))/numberPoints];**

**end**

**subplot(3,2,4)**

**plot(T,P\_fa4,'b');**

**hold on;**

**plot(T,P\_miss4, 'r');**

**xlabel ('Thresh Hold v=2, d=0.2')**

**ylabel ('Probability');**

**legend ('fa','miss')**

**%distortion d= 0.4, voltage v = 1**

**%X\_A5=(1- N) + 0.4 \*(1-N).^2;**

**%X\_P5= normcdf(T,1,variance)+.4\*(normcdf(T,1,variance).^2);**

**%distortion d= 0.4, voltage v = 1**

**X\_A5=( (N) + 0.4 \*(N.^2)); %Target Absent**

**P\_fa5=sum(X\_A5>T(1))/numberPoints;**

**for i=2 : numel(T)**

**P\_fa5=[P\_fa5,sum(X\_A5>T(i))/numberPoints];**

**end**

**X\_P5=( (1+N) + 0.4 \*((1+N).^2)); %target Presence**

**P\_miss5=sum(X\_P5 < T(1))/numberPoints;**

**for i=2 : numel(T)**

**P\_miss5=[P\_miss5,sum(X\_P5<T(i))/numberPoints];**

**end**

**subplot(3,2,5)**

**plot(T,P\_fa5,'b');**

**hold on;**

**plot(T,P\_miss5, 'r');**

**xlabel ('Thresh Hold v=1, d=0.4')**

**ylabel ('Probability');**

**legend ('fa','miss')**

**%distortion d= 0.4, voltage v = 2**

**X\_A6=( (N) + 0.4 \*(N.^2)); %Target Absent**

**P\_fa6=sum(X\_A6>T(1))/numberPoints;**

**for i=2 : numel(T)**

**P\_fa6=[P\_fa6,sum(X\_A6>T(i))/numberPoints];**

**end**

**X\_P6=( (2+N) + 0.4 \*((2+N).^2)); %target Presence**

**P\_miss6=sum(X\_P6 < T(1))/numberPoints;**

**for i=2 : numel(T)**

**P\_miss6=[P\_miss6,sum(X\_P6<T(i))/numberPoints];**

**end**

**subplot(3,2,6)**

**plot(T,P\_fa6,'b');**

**hold on;**

**plot(T,P\_miss6, 'r');**

**xlabel ('Thresh Hold')**

**ylabel ('Probability');**

**legend ('fa','miss')**

**%Question2**

**%Distortion = 0 ROC voltage = 1 & 2**

**figure (2)**

**subplot(2,3,1);**

**plot(P\_fa1,P\_miss1)**

**hold on;**

**plot(P\_fa2,P\_miss2)**

**hold on;**

**xlabel('Pfa');**

**ylabel('Pmiss');**

**title('Pfa vs Pmiss');**

**%Distortion = 0.2 ROC voltage = 1 & 2**

**figure (2)**

**subplot(2,3,2);**

**plot(P\_fa3,P\_miss3)**

**hold on;**

**plot(P\_fa4,P\_miss4)**

**hold on;**

**xlabel('Pfa');**

**ylabel('Pmiss');**

**title('Pfa vs Pmiss');**

**%Distortion = 0.4 ROC voltage = 1 & 2**

**figure (2)**

**subplot(2,3,3);**

**plot(P\_fa5,P\_miss5)**

**hold on;**

**plot(P\_fa6,P\_miss6)**

**xlabel('Pfa');**

**ylabel('Pmiss');**

**title('Pfa vs Pmiss');**

**hold on;**

**%Question 3**

**%distortion = 0**

**Perr\_1= .8\*P\_fa1 + 0.2\*P\_miss1;**

**%calculating the min**

**r1 = min(Perr\_1);**

**w1 = 0;**

**for i = 1 : numel (Perr\_1)**

**if(Perr\_1(i) == r1)**

**w1= i;**

**end**

**end**

**minTperr1 = T(w1);**

**Perr\_2= .8\*P\_fa2 + 0.2\*P\_miss2;**

**r2 = min(Perr\_2);**

**w2 = 0;**

**for i = 1 : numel (Perr\_2)**

**if(Perr\_2(i) == r2)**

**w2= i;**

**end**

**end**

**minTperr2 = T(w2);**

**figure (2)**

**subplot(2,3,4);**

**plot(T,Perr\_1)**

**hold on;**

**plot(T,Perr\_2)**

**hold on;**

**xlabel('Threshold');**

**ylabel('Perr');**

**title('Perr vs Threshold');**

**axis ([-2 6 0 1.0]);**

**%distortion =.2**

**Perr\_3= .8\*P\_fa3 + 0.2\*P\_miss3;**

**r3 = min(Perr\_3);**

**w3 = 0;**

**for i = 1 : numel (Perr\_1)**

**if(Perr\_3(i) == r3)**

**w3= i;**

**end**

**end**

**minTperr3 = T(w3);**

**Perr\_4= .8\*P\_fa4 + 0.2\*P\_miss4;**

**r4 = min(Perr\_4);**

**w4 = 0;**

**for i = 1 : numel (Perr\_4)**

**if(Perr\_4(i) == r4)**

**w4= i;**

**end**

**end**

**minTperr4 = T(w4);**

**figure (2)**

**subplot(2,3,5);**

**plot(T,Perr\_3)**

**hold on;**

**plot(T,Perr\_4)**

**hold on;**

**xlabel('Threshold');**

**ylabel('Perr');**

**title('Perr vs Threshold');**

**axis ([-2 6 0 1.0]);**

**%distortion =.4**

**Perr\_5= .8\*P\_fa5 + 0.2\*P\_miss5;**

**r5 = min(Perr\_5);**

**w5 = 0;**

**for i = 1 : numel (Perr\_5)**

**if(Perr\_5(i) == r5)**

**w5= i;**

**end**

**end**

**minTperr5 = T(w5);**

**Perr\_6= .8\*P\_fa6 + 0.2\*P\_miss6;**

**r6 = min(Perr\_6);**

**w6 = 0;**

**for i = 1 : numel (Perr\_6)**

**if(Perr\_6(i) == r6)**

**w6= i;**

**end**

**end**

**minTperr6 = T(w6);**

**figure (2)**

**subplot(2,3,6);**

**plot(T,Perr\_5)**

**hold on;**

**plot(T,Perr\_6)**

**hold on;**

**xlabel('Threshold');**

**ylabel('Perr');**

**title('Perr vs Threshold');**

**axis ([-2 6 0 1.0]);**

**%Question 4**

**%d=0**

**E\_1= .8\*P\_fa1\*1 + 0.2\*P\_miss1\*5;**

**t1 = min(E\_1);**

**s1 = 0;**

**for i = 1 : numel (E\_1)**

**if(E\_1(i) == t1)**

**s1= i;**

**end**

**end**

**minTE\_C1 = T(s1);**

**E\_2= .8\*P\_fa2\*1 + 0.2\*P\_miss2\*5;**

**t2 = min(E\_2);**

**s2 = 0;**

**for i = 1 : numel (E\_2)**

**if(E\_2(i) == t2)**

**s2= i;**

**end**

**end**

**minTE\_C2 = T(s2);**

**figure (3)**

**subplot(2,3,1);**

**plot(T,E\_1)**

**hold on;**

**plot(T,E\_2)**

**hold on;**

**axis([-2 6 0 1.0]);**

**xlabel('Threshold');**

**ylabel('E\_C');**

**title('E\_C vs Threshold');**

**%d=.2**

**E\_3= .8\*P\_fa3\*1 + 0.2\*P\_miss3\*5;**

**t3 = min(E\_3);**

**s3 = 0;**

**for i = 1 : numel (E\_3)**

**if(E\_3(i) == t3)**

**s3= i;**

**end**

**end**

**minTE\_C3 = T(s3);**

**E\_4= .8\*P\_fa4\*1 + 0.2\*P\_miss4\*5;**

**t4 = min(E\_4);**

**s4 = 0;**

**for i = 1 : numel (E\_4)**

**if(E\_4(i) == t4)**

**s4= i;**

**end**

**end**

**minTE\_C4 = T(s4);**

**subplot(2,3,2);**

**plot(T,E\_3)**

**hold on;**

**plot(T,E\_4)**

**hold on;**

**axis([-2 6 0 1.0]);**

**xlabel('Threshold');**

**ylabel('E\_C');**

**title('E\_C vs Threshold');**

**%d=.4**

**E\_5= .8\*P\_fa5\*1 + 0.2\*P\_miss5\*5;**

**t5 = min(E\_5);**

**s5 = 0;**

**for i = 1 : numel (E\_5)**

**if(E\_5(i) == t5)**

**s5= i;**

**end**

**end**

**minTE\_C5 = T(s5);**

**E\_6= .8\*P\_fa6\*1 + 0.2\*P\_miss6\*5;**

**t6 = min(E\_6);**

**s6 = 0;**

**for i = 1 : numel (E\_6)**

**if(E\_6(i) == t6)**

**s6= i;**

**end**

**end**

**minTE\_C6 = T(s6);**

**subplot(2,3,3);**

**plot(T,E\_5)**

**hold on;**

**plot(T,E\_6)**

**hold on;**

**axis([-2 6 0 1.0]);**

**xlabel('Threshold');**

**ylabel('E\_C');**

**title('E\_C vs Threshold');**