UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN
Department of Electrical and Computer Engineering

ECE 544NA Pattern Recognition

Solutions 1

Fall 2013

Assigned: Thursday, August 29, 2013 Due: Thursday, September 5, 2013

Reading: Bishop Chapter 1

Problem 1.1

$$\epsilon = \vec{y}^T\vec{y} - 2\vec{a}^T X^T\vec{y} + \vec{a}^T X^T X\vec{a}$$
$$\nabla_{\vec{a}}\epsilon = -2X^T\vec{y} + 2X^T X\vec{a}$$
$$\nabla_{\vec{a}}\epsilon = 0 \ \text{ if } \ \vec{a} = (X^T X)^{-1}X^T\vec{y}$$

Problem 1.2

(a) $P_{FA} = Q(\theta) = 1 - \Phi(\theta)$

(b) $P_{MD} = \Phi(\theta - d') = Q(d' - \theta)$

(c) $P_{EER} = Q(d'/2) = 1 - \Phi(d'/2)$

(d) $P_E = \pi_A Q(\theta) + \pi_B Q(d' - \theta)$

(e) Minimum error is achieved by the rule that chooses B for any x such that $\pi_A p(x|A) < \pi_B p(x|B)$, thus the threshold is given by $\pi_A p(x = \theta|A) = \pi_B p(x = \theta|B)$. Expanding, we find

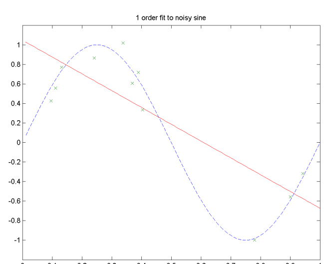
$$\pi_A \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\theta^2} = \pi_B \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(d'-\theta)^2}$$
$$\theta = \frac{d'}{2} + \frac{1}{d'}\ln\left(\frac{\pi_A}{\pi_B}\right)$$
$$P_{BR} = \pi_A Q\left(\frac{d'}{2} + \frac{1}{d'}\ln\left(\frac{\pi_A}{\pi_B}\right)\right) + \pi_B Q\left(\frac{d'}{2} - \frac{1}{d'}\ln\left(\frac{\pi_A}{\pi_B}\right)\right)$$

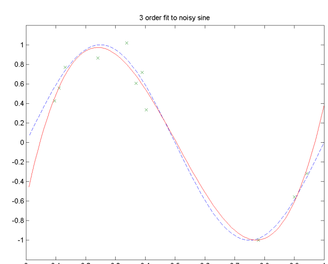
In this problem, the likelihood functions have the same form and standard deviation, and are just shifted versions of one another. In this circumstance, when the classes are equally likely, then the whole problem becomes symmetric around its midpoint $\theta = d'/2$, and in particular, the false alarm and missed-detection rates become equal. We have already shown that the equal error rate for this problem is $Q(d'/2)$. Since the total probability of error is a weighted average of the false-alarm and missed-detection rates, equality of the latter two means equality of the former as well.

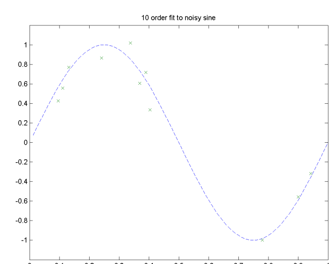
Matlab Exercises

Problem 1.3

```
% random variables
t = rand(11,1);
z = sin(2*pi*t);
y = z + 0.1*randn(11,1);

% plotting convenience variables
tau = [0.01:0.01:1]';
orders=[1,3,10];

% create the figures
for k=1:3,
  X = exp(log(t)*[0:orders(k)]);
  a = inv(X'*X)*X'*y;

  Xtau = exp(log(tau)*[0:orders(k)]);
  ftau = Xtau*a;
  ytau = sin(2*pi*tau);

  figure(k);
  plot(tau,ytau,'--',t,y,'x',tau,ftau,'-');
  title(sprintf('%d order fit to noisy sine',orders(k)));
  axis([0 1 -1.2 1.2]);
  print('-dpng',sprintf('fig3_%d.png',k));
end
```





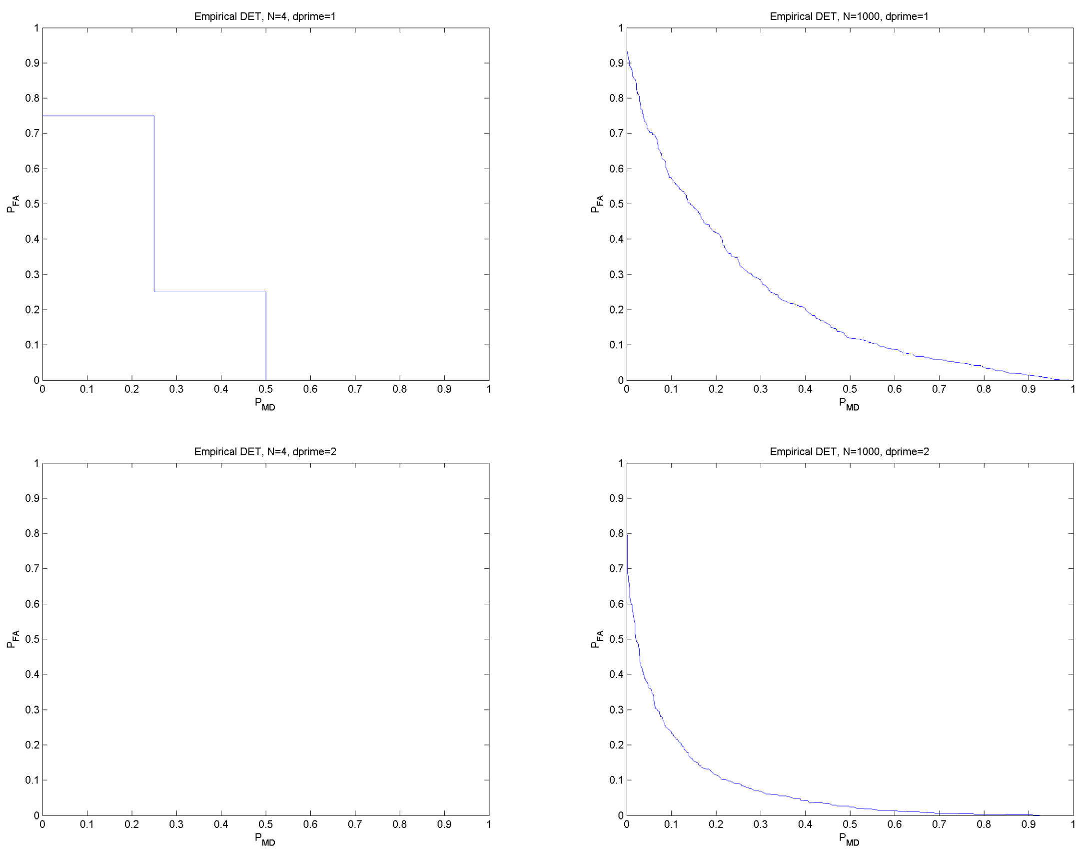


Notice that the last plot has no well-fit polynomial, because `X'*X` could not be inverted using the `inv` function. In this case, `pinv` would have done a better job, since matlab's implementation of `pinv` only inverts the non-zero eigenvalues.

Problem 1.4

```
function [PMD,PFA,x_sorted, C_sorted]=prob4(dprime,N)
% problem 1.4: generate empirical DET for given dprime, N

% x are data, C are class labels
x = [randn(N,1); dprime + randn(N,1)];
C = [zeros(N,1); ones(N,1)];

% sort them to find the possible thresholds
[x_sorted,sort_indices]=sort(x);
```

```
C_sorted = C(sort_indices);

% MD = ones to left of theta, FA = zeros to right of theta
PMD = [0; cumsum(C_sorted)/N; 1];
PFA = [1; 1 - cumsum(1-C_sorted)/N; 0];
plot(PMD,PFA);
title(sprintf('Empirical DET, N=%d, dprime=%g',N,dprime));
xlabel('P_{MD}');
ylabel('P_{FA}');
print('-dpng',sprintf('fig4_%d_%d.png',N,dprime));
```



The lower-left DET ($N = 4, d' = 2$) hugs the axes: there is no threshold for which both $P_{MD} > 0$ and $P_{FA} > 0$. Your plots may differ slightly because you are generating random data, but your plots should have roughly the characteristics shown above.