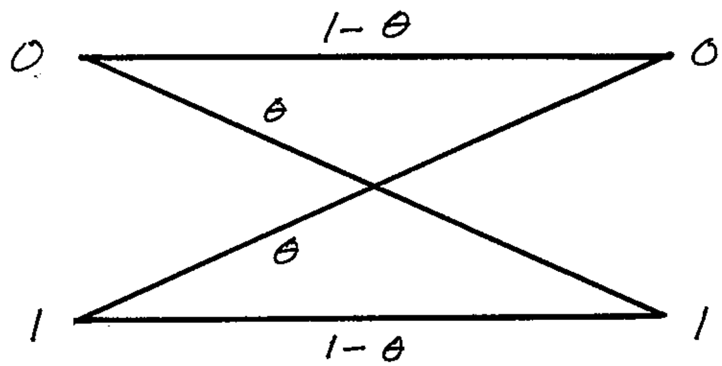
APPLICATION: BINARY SYMMETRIC CHANNEL

In a binary symmetric channel (BSC), whenever we transmit a bit (0 or 1), the bit is <u>flipped</u> with probability $\theta \in [0,1]$.



BSC is a very common model in digital comm. systems

If we denote

x = transmitted bit

y = received bit

then

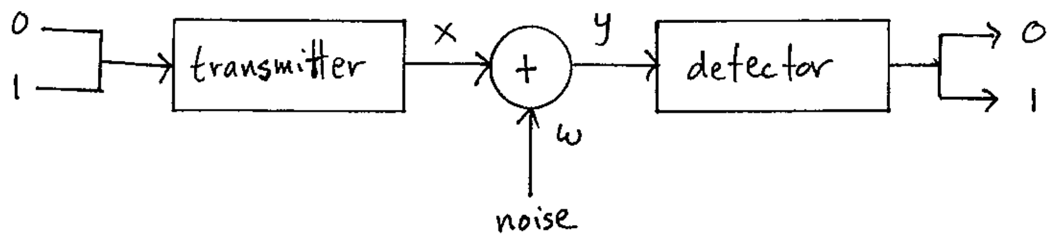
$$y = x + w$$

modular arithmetic: $1+1=0$

where

$$w \sim \text{Bernoulli}(\theta)$$

Our job is to build a detector to determine the transmitted bit from the received bit



Let's set this up as a hypothesis testing problem:

$$H_0: \ y = 0 + w$$
$$H_1: \ y = 1 + w$$

Equivalently

$$H_0: \ y \sim \text{Bernoulli}(\theta)$$
$$H_1: \ y \sim$$

ⓐ

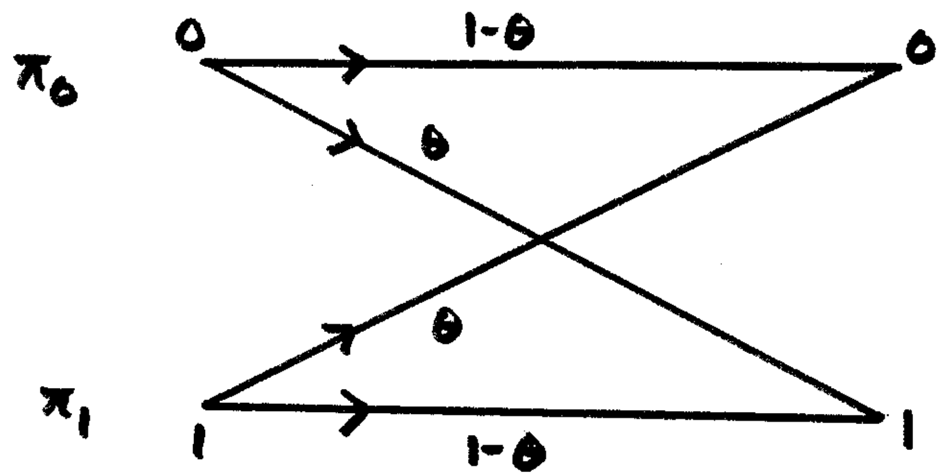
Our goal is effectively to form an estimate $\hat{x}$ of the transmitted bit x.

Let's assume $\theta < \frac{1}{2}$: more often than not a bit is uncorrupted.

Also assume prior probabilities

$$\pi_i = \Pr\{X = i\}$$

are known.



How should we decode/detect y so as to minimize the <u>probability of error</u>?

P_e is minimized by the LRT:

$$\Lambda(y) = \frac{f_1(y)}{f_0(y)} \overset{H_1}{\underset{H_0}{\gtrless}} \frac{\pi_0}{\pi_1} = \eta$$

where

$$f_1(y) = \begin{cases} 1-\theta & \text{if } y=1 \\ \theta & \text{if } y=0 \end{cases}$$

$$f_0(y) = \begin{cases} \theta & \text{if } y=1 \\ 1-\theta & \text{if } y=0 \end{cases}$$

So

(b)

$$\Lambda(y) =$$

$$=$$

Let's apply <u>monotonic transformations</u> to simplify the LRT:

$$\frac{\theta}{1-\theta}\cdot\left(\frac{1-\theta}{\theta}\right)^{2y} \underset{H_0}{\overset{H_1}{\gtrless}} \eta$$

$$\Updownarrow$$

$$\left(\frac{1-\theta}{\theta}\right)^{2y} \underset{H_0}{\overset{H_1}{\gtrless}} \eta\cdot\frac{1-\theta}{\theta}$$

$$\Updownarrow$$

$$2y\ln\left(\frac{1-\theta}{\theta}\right) \underset{H_0}{\overset{H_1}{\gtrless}} \ln(\eta)+\ln\left(\frac{1-\theta}{\theta}\right)$$

$$\Updownarrow$$

$$y \underset{H_0}{\overset{H_1}{\gtrless}} \frac{1}{2}+\frac{1}{2}\cdot\frac{\ln(\eta)}{\ln\left(\frac{1-\theta}{\theta}\right)} \equiv \gamma$$

$$\boxed{\text{Note: } \theta<\tfrac{1}{2} \Rightarrow \frac{1-\theta}{\theta}>1 \Rightarrow \ln\left(\frac{1-\theta}{\theta}\right)>0}$$

Three cases:

threshold	decision rule
$\gamma > 1$	$\hat{x} = 0$
$\gamma < 0$	$\hat{x} = 1$
$0 < \gamma < 1$	$\hat{x} = y$

Question Do we need to worry about $\gamma = 0$ or $\gamma = 1$?

Let's consider each case: Recall $\gamma = \frac{1}{2} + \frac{1}{2}\frac{\ln(\eta)}{\ln\left(\frac{1-\theta}{\theta}\right)}$

$$\boxed{\gamma > 1} \iff \frac{\ln(\eta)}{\ln\left(\frac{1-\theta}{\theta}\right)} > 1$$

$$\iff \ln(\eta) > \ln\left(\frac{1-\theta}{\theta}\right)$$

$$\iff \eta > \frac{1-\theta}{\theta} \qquad \left[\text{Recall } \eta = \frac{\pi_0}{\pi_1} = \frac{1-\pi_1}{\pi_1}\right]$$

$$\iff \frac{1-\pi_1}{\pi_1} > \frac{1-\theta}{\theta}$$

$$\iff \frac{1}{\pi_1} - 1 > \frac{1}{\theta} - 1$$

$$\iff \frac{1}{\pi_1} > \frac{1}{\theta} \iff \boxed{\pi_1 < \theta}$$

$$\boxed{\gamma < 0} \iff \frac{\ln(\eta)}{\ln\left(\frac{1-\theta}{\theta}\right)} < -1$$

$$\iff \ln(\eta) < \ln\left(\frac{\theta}{1-\theta}\right)$$

$$\iff \eta < \frac{\theta}{1-\theta}$$

$$\iff \frac{\pi_0}{1-\pi_0} < \frac{\theta}{1-\theta}$$

$$\iff \frac{1-\pi_0}{\pi_0} > \frac{1-\theta}{\theta}$$

$$\iff \frac{1}{\pi_0} - 1 > \frac{1}{\theta} - 1$$

$$\iff \frac{1}{\pi_0} > \frac{1}{\theta} \iff \boxed{\pi_0 < \theta}$$

$\boxed{0 < \gamma < 1}$ From the previous two cases, we conclude

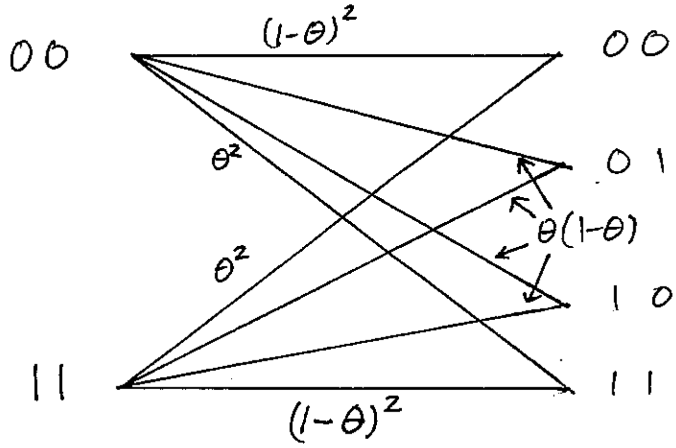
$$0 < \gamma < 1 \iff \boxed{\theta < \pi_1 \text{ and } \theta < \pi_2.}$$

What is the intuition behind these cases?

Binary Repitition Code

In an effort to decrease P_e, let's send N copies of each "information" bit:

<u>Example</u> $N=2$



As a hypothesis testing problem, we have

$$H_0: \ \underline{y} = 00\ldots0 + \underline{\omega}$$

$$H_1: \ \underline{y} = 11\ldots1 + \underline{\omega}$$

where

$$\underline{y} = (y_1, \ldots, y_N)^T$$

$$\underline{\omega} = (\omega_1, \ldots, \omega_N)^T$$

$$\omega_n \overset{iid}{\sim} \text{Bernoulli}(\theta)$$

Exercise | Apply the LRT to determine the min P_E detector. Express your answer in terms of a one dimensional test statistic. Derive a formula for P_E in terms of N, θ, and $\eta = \frac{\pi_0}{\pi_1}$

Solution 1

$$\Lambda(\underline{y}) = \frac{f_1(\underline{y})}{f_0(\underline{y})} = \frac{\prod_{n=1}^{N} (1-\theta)^{y_n} \theta^{1-y_n}}{\prod_{n=1}^{N} \theta^{y_n} (1-\theta)^{1-y_n}}$$

$$\boxed{k = \sum_{n=1}^{N} y_n}$$

sufficient statistic

$$= \frac{(1-\theta)^k \theta^{N-k}}{\theta^k (1-\theta)^{N-k}} = \left(\frac{1-\theta}{\theta}\right)^{2k-N}$$

After some simplification

$$\Lambda(\underline{y}) \underset{H_0}{\overset{H_1}{\gtrless}} \eta \iff (2k-N) \ln\left(\frac{1-\theta}{\theta}\right) \underset{H_0}{\overset{H_1}{\gtrless}} \ln(\eta)$$

$$\iff k \underset{H_0}{\overset{H_1}{\gtrless}} \frac{N}{2} + \frac{1}{2} \frac{\ln(\eta)}{\ln\left(\frac{1-\theta}{\theta}\right)} \equiv \gamma$$

In the special case $\eta = 1$ $(\pi_0 = \pi_1 = \frac{1}{2})$,

$$k \underset{H_0}{\overset{H_1}{\gtrless}} \frac{N}{2}$$

$\boxed{\text{“majority vote”}}$

In case $k = \gamma$, let's declare H_0.

Let's compute the probability of error:

$$P_e = \pi_0 P(H_1|H_0) + \pi_1 P(H_0|H_1)$$

$$= \pi_0 P_F + \pi_1 P_M$$

$$P_F = P(k > \gamma \mid H_0)$$

$$P_M = P(k \leq \gamma \mid H_1)$$

Under H_0,

c

$k \sim$

$\Rightarrow f_0(k) =$

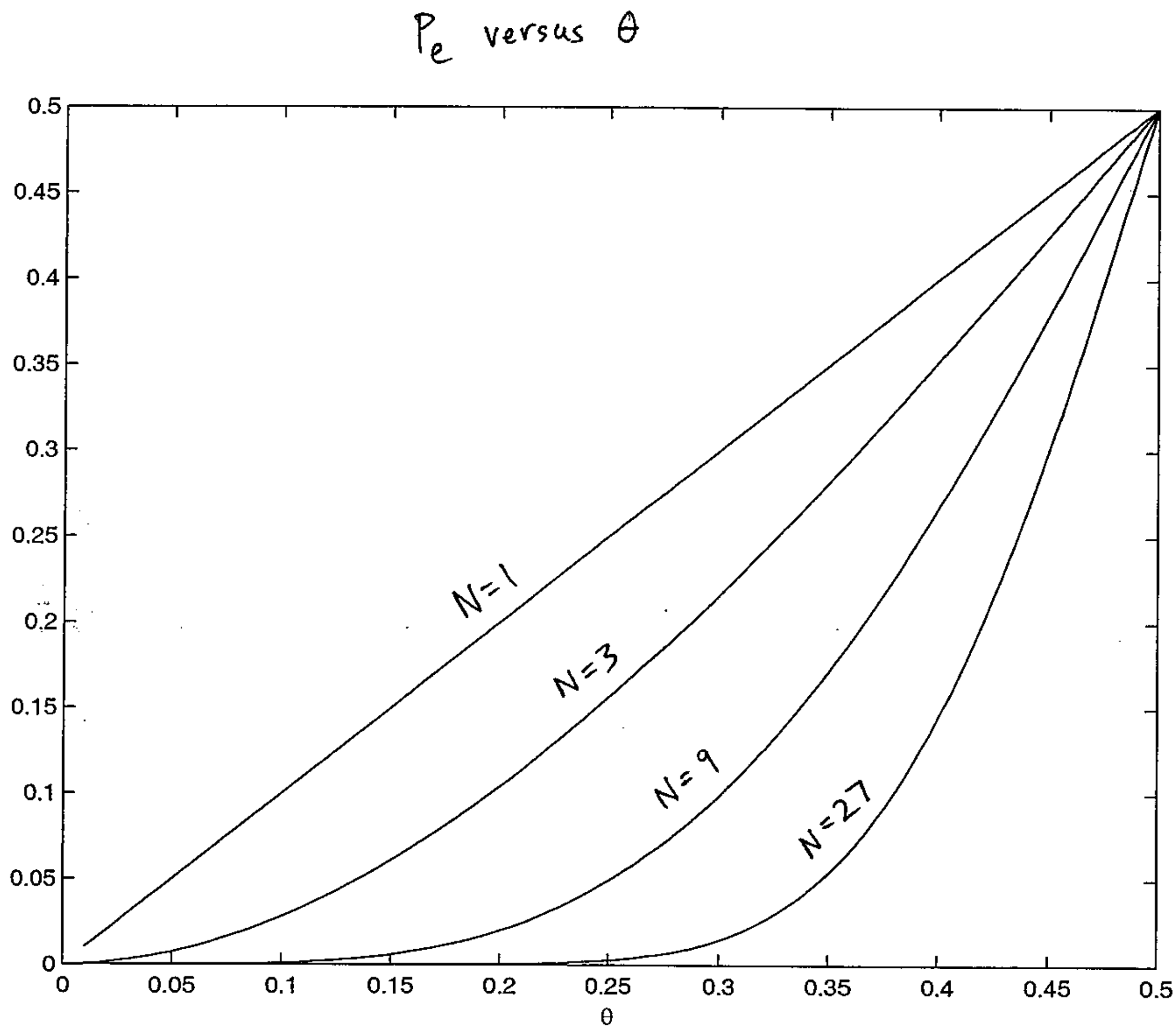
$\Rightarrow P_F =$

Under H_1,

$k \sim$

$\Rightarrow f_1(k) =$

$\Rightarrow P_M =$



Pe versus θ
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45
0.5
θ
N=1
N=3
N=9
N=27

```
% Binary repitition code

close all

for N=[1 3 9 27]

    Npts=100;
    p=linspace(.01,.5,Npts);

    pi0=1/2;
    pi1=1-pi0;
    eta=pi0/pi1;

    for j=1:Npts
        gam=N/2 +.5*(log(eta)/log((1-p)/p));

        pmf0=zeros(1,N+1);
        for k=0:N
            pmf0(k+1)=nchoosek(N,k)*(p(j)^k)*((1-p(j))^(N-k));
        end

        Pf = sum(pmf0(find(0:N>gam)));

        pmf1=zeros(1,N+1);
        for k=0:N
            pmf1(k+1)=nchoosek(N,k)*((1-p(j))^k)*(p(j)^(N-k));
        end

        Pm = sum(pmf1(find(0:N<=gam)));

        Pe(j)=pi0*Pf+pi1*Pm;
    end

    plot(p,Pe)
    hold on
end

xlabel('\theta');
```

Now let's design a Neyman-Pearson detector:

To be concrete, let's say $N=15$, $\theta = \frac{1}{4}$, $\alpha = 0.2$

We need to find γ, ρ such that

$$P(k > \gamma \mid H_0) + \rho P(k = \gamma \mid H_0) = \alpha$$

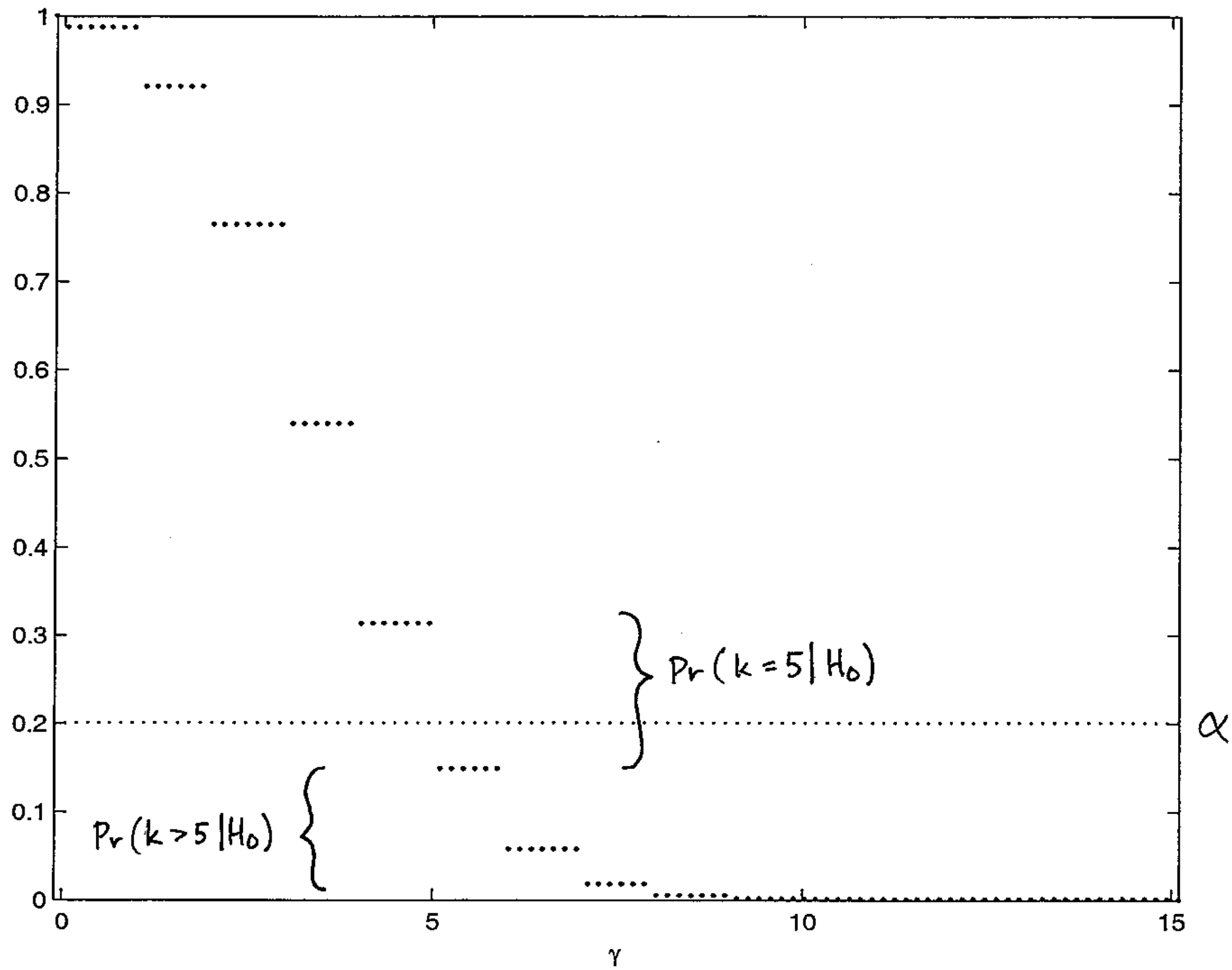
where $\gamma \in \{0, 1, \ldots, N\}$, $\rho \in [0, 1)$.

[see plot on next page]

So the NP detector is

- If $k > 5$, declare H_1
- If $k < 5$, declare H_0
- If $k = 5$, flip a coin that turns up heads (H_1) with probability $\rho = .3126$

$$Pr(k > \gamma \mid H_0)$$



$$\rho = \frac{\alpha - Pr(k>5 \mid H_0)}{Pr(k=5)} = .3126$$

$$\Longleftrightarrow \quad P_F = Pr(k>5 \mid H_0) + \rho\, Pr(k=5 \mid H_0) = \alpha$$

```
% Binary repitition code

close all

N=15;
alpha=0.2;
p=.25;

pmf0=zeros(1,N+1);
for k=0:N
    pmf0(k+1)=nchoosek(N,k)*(p^k)*((1-p)^(N-k));
end

Npts=100;
gam = linspace(-.1,N+.1,Npts);
for j=1:Npts
    Pf(j) = sum(pmf0(find(0:N>gam(j))));
end

plot(gam,Pf,'.')
hold on
plot(gam,alpha,'--')

xlabel('\gamma');
axis tight

% gamma = 5 by inspection of graph
rho = (alpha - sum(pmf0(0:N>5)))/pmf0(5+1)  % recall Matlab indexing starts↙
 at 1
```

Error Correcting Codes

There are more sophisticated ways of introducing <u>redundancy</u> in order to reduce the probability of error per "information bit."

Example Hamming (7,4) code

$\underline{x}_1$	0000 000	$\underline{x}_9$	1000 011
$\underline{x}_2$	0001 111	$\underline{x}_{10}$	1001 100
$\underline{x}_3$	0010 110	$\underline{x}_{11}$	1010 101
$\underline{x}_4$	0011 001	$\underline{x}_{12}$	1011 010
$\underline{x}_5$	0100 101	$\underline{x}_{13}$	1100 110
$\underline{x}_6$	0101 010	$\underline{x}_{14}$	1101 001
$\underline{x}_7$	0110 011	$\underline{x}_{15}$	1110 000
$\underline{x}_8$	0110 100	$\underline{x}_{16}$	1111 111

"information bits" "parity check bits"

<u>M-ary hypothesis test</u>

$$H_1 : \underline{y} = \underline{x}_1 + \underline{w}$$

$$\vdots$$

$$H_M : \underline{y} = \underline{x}_M + \underline{w}$$

Recall the <u>MAP detector</u>:

$$\boxed{\text{Choose } H_i \text{ such that } \pi_i f_i(\underline{y}) \text{ is maximal}}$$

In communication systems, we often know

$$\pi_1 = \pi_2 = \cdots = \pi_M = \frac{1}{M}$$

in which case we get the <u>maximum likelihood detector</u>:

$$\boxed{\text{Choose } H_i \text{ such that } f_i(\underline{y}) \text{ is maximal}}$$

In general, $f_i(\underline{y})$ is easily computed as a product of Bernoulli likelihoods, since $\underline{x}_i$ is known.

$\boxed{\text{Key}}$

a. Bernoulli $(1-\theta)$

b. $\Lambda(y) = \dfrac{(1-\theta)^y \theta^{1-y}}{\theta^y (1-\theta)^{1-y}} = \dfrac{\theta}{1-\theta} \cdot \left(\dfrac{1-\theta}{\theta}\right)^{2y}$

c. Under H_0,

$$k \sim \text{binom}(N, \theta)$$

$$f_0(k) = \binom{N}{k} \theta^k (1-\theta)^{N-k}$$

$$P_F = \sum_{k > \gamma} \binom{N}{k} \theta^k (1-\theta)^{N-k}$$

Under H_1,

$$k \sim \text{binom}(N, 1-\theta)$$

$$f_1(k) = \binom{N}{k} (1-\theta)^k \theta^{N-k}$$

$$P_M = \sum_{k \leq \gamma} \binom{N}{k} (1-\theta)^k \theta^{N-k}$$