2.1 (a) There are $\binom{100}{0}+\binom{100}{1}+\binom{100}{2}+\binom{100}{3} = 166751$ length 100 sequences containing 3 or fewer 1s.

The smallest power of 2 greater than 166751 is 2^{18}. So 18 is the minimum length necessary.

(b) The prob. of setting a source sequence for which ther is no codeword is

$$\Pr\{\text{no codeword}\} = 1 - \Pr\{\text{codeword provided}\}$$
$$= 1 - \sum_{k=0}^{3}\binom{100}{k}(0.005)^k(0.995)^{100-k} = 0.0017$$

2.2 By defn., a sequence $(X_1,\ldots,X_8)\in A^8$ is ε-typical if $\left|\frac{1}{8}\sum_{k=1}^{8}\log\frac{1}{p(X_k)} - H\right| \le \varepsilon$

For this source $H = \frac{14}{8} = 1.75$ bits. The possible values of $\log\frac{1}{p(a_j)}$ = 1, 2, or 3 for this source.

For $\varepsilon = 0$, we must have

$$\sum_{k=1}^{8}\log_2\frac{1}{p(X_k)} = 8H = 8\cdot 1.75 = 14 \Rightarrow \text{"}\textstyle\sum = 14\text{"} \quad (\varepsilon=0)$$

For $\varepsilon = 0.25$, we have

$$\left|\sum_{k=1}^{8}\log_2\frac{1}{p(X_k)} - 8H\right| \le 8\varepsilon \Rightarrow \text{"}12 \le \textstyle\sum \le 16\text{"} \quad (\varepsilon=0.25).$$

So in order to determine the number of ε-typical sequences for $\varepsilon=0$ and $\varepsilon=0.25$, we must count the number of sequences for which $\sum_{k=1}^{8}\log\frac{1}{p(X_k)}$ falls within the collect ranges as determined above.

Example: For $\sum = 15$ bits

no. a_1's	no. a_2's	no. a_3's and a_4's	
4	1	3	$\Rightarrow \binom{8}{4,1,3}\cdot 2^3 = 2240$ possibilities
3	3	2	$\Rightarrow \binom{8}{3,3,2}\cdot 2^2 = 2240$ possibilities
2	5	1	$\Rightarrow \binom{8}{2,5,1}\cdot 2^1 = 336$ "
1	7	0	$\Rightarrow \binom{8}{1,7,0}\cdot 2^0 = 8$ possibilities

Total of 4824 sequences.

Doing this for $\Sigma = 12, \ldots, 16$, we get

Σ:	12	13	14	15	16
No. sequences	518	1288	2716	4824	7393

So we have

$\underline{\varepsilon = 0.0}$: 2716 sequences. $2^{\lceil Rn \rceil} \geq 2716 \Rightarrow Rn \geq 12 \Rightarrow R = \frac{3}{2}$.

$\underline{\varepsilon = 0.25}$: 16739 sequences. $2^{\lceil Rn \rceil} \geq 16739 \Rightarrow Rn \geq 15 \Rightarrow R = \frac{15}{8}$.

2.3 $\underline{\text{Case 1}}$: Checking K-M Inequality, $\sum_i 4^{-n_i} = 3\left(\frac{1}{4} + \frac{1}{16} + \frac{1}{64} + \frac{1}{256}\right) + \frac{4}{1024} = \frac{1024}{1024} = 1$

$\Rightarrow$ $\exists$ prefix code satisfying these lengths. One such code is given below.

$\underline{\text{Case 2}}$: Checking K-M Inequality $\Rightarrow \sum_i 4^{-n_i} = \frac{1025}{1024} > 1$

$\Rightarrow$ There is no UD code with these lengths

$\underline{\text{Case 3}}$: Checking K-M Inequality $\Rightarrow \sum_i 4^{-n_i} = \frac{784}{1024} < 1$

$\Rightarrow$ A prefix code with these lengths exists. One such code is given below.

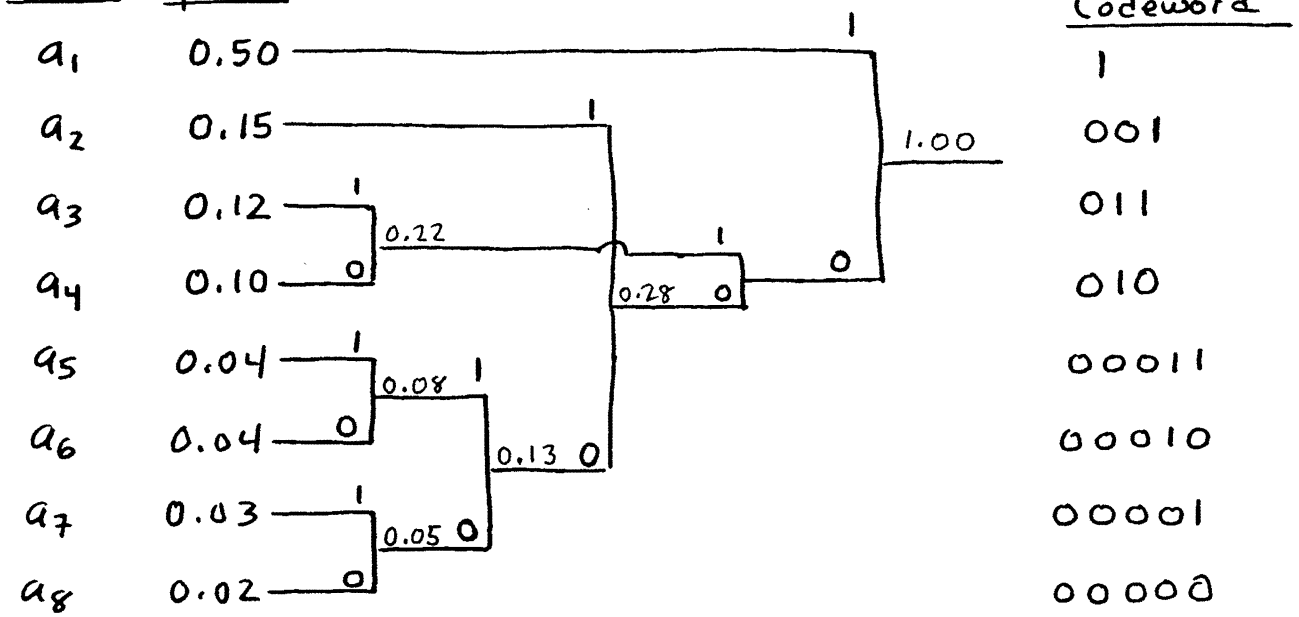
$\underline{\text{Case 4}}$: checking K-M $\Rightarrow \sum_i 4^{-n_i} = \frac{17}{32} < 1$. $\Rightarrow$ Prefix code exists. One given below.

Case 1 Code:
0
1
2
30
31
32
330
331
332
3330
3331
3332
33330
33331
33332
33333

Case 3 Code:
0
10
11
12
13
20
21
22
230
231
232
3010
3011
3012
3013
3110
3111
3112
33300
33301
33310
33311

Case 4 Code:
00
01
02
03
10
11
12
130
131
132
1331
1332
1333
1330
2330
2331
2332
2333
3330
3331
3332

2.4



$H = 2.25$ bits/symbol

$L(C) = 2.26$ bits/symbol (Very close to H).

For a fixed length code, we would need 3 bits/symbol.

2.5 The number of unused terminal nodes can be calculated using the formula derived in class:

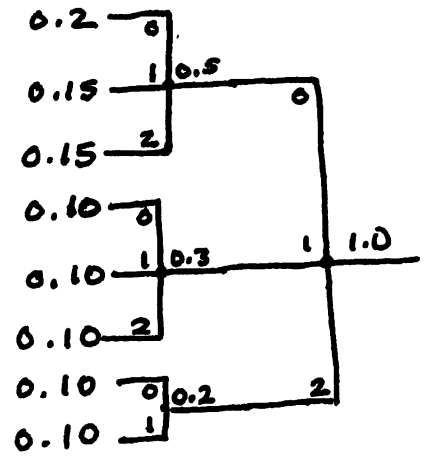
$$B = D - 2 - \lceil (m-2) \bmod (D-1) \rceil$$
$$= 3 - 2 - [\,6 \bmod 2\,] = 1 - 0 = 1.$$

This means we only merge $D - B = 2$ terminal nodes on the first merger. After that, we merge 3 nodes at each stage.

Using the Huffman algorithm, we can generate the following two codes:

Code 1:

Source Letter	$P(a_i)$	Codeword
a_1	0.2	00
a_2	0.15	01
a_3	0.15	02
a_4	0.10	10
a_5	0.10	11
a_6	0.10	12
a_7	0.10	20
a_8	0.10	21

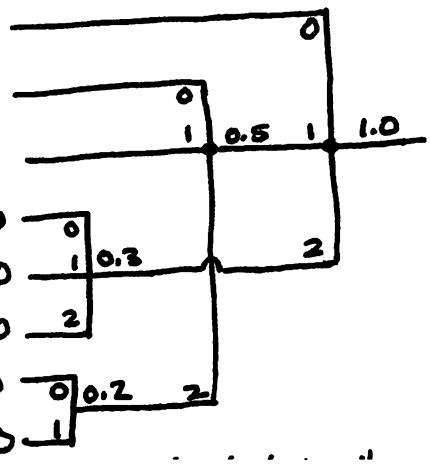


$\bar{n} = 2$

$\sigma_n^2 = 0$

Code 2:

Source Letter	$P(a_i)$	Codeword
a_1	0.2	0
a_2	0.15	10
a_3	0.15	11
a_4	0.10	20
a_5	0.10	21
a_6	0.10	22
a_7	0.10	120
a_8	0.10	121



$\bar{n} = 0.2(1) + (0.6)2 + 0.2(3) = 2$

$\sigma_n^2 = 0.4$

Code 1 is preferable for several reasons: (1) Fixed block length makes decoding simpler; (2) with fixed length codewords no buffering is required; (3) Code 1's average codeword length $\bar{n}$ is insensitive to changes in the input probabilities; (4) It is easier to recover from occasional errors in code 1.

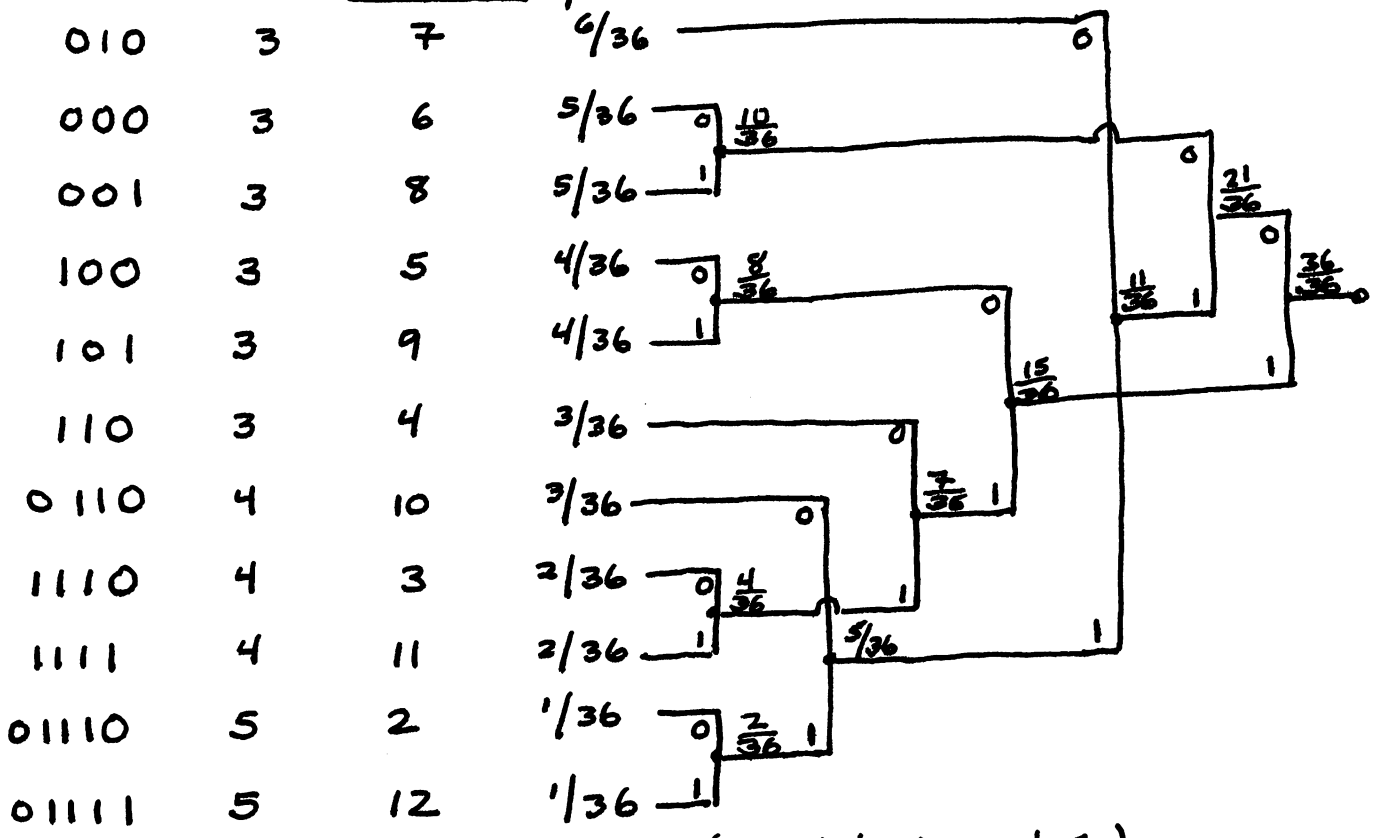
Code 2 has one major advantage: If we get out of word-sync in Code 2, it is easier to get back in sync. In code 1 the only way to get back in sync is to have a_3 or a_6 followed by a_7 or a_8.

2.6 There are 11 possible outcomes $\{2, 3, \ldots, 12\}$ for the sum of two dice, with probabilities as determined in homework problem 1.1.

We can think of the sequence of yes/no questions leading to any outcome as a binary codeword or encoding corresponding to that outcome. As a result, we can construct a questioning strategy, or equivalently a binary encoding, in which the minimum number of questions asked on average can be found using the Huffman algorithm.

Applying the Huffman algorithm, one such code is:

Codeword	n	Outcome	prob.
010	3	7	6/36
000	3	6	5/36
001	3	8	5/36
100	3	5	4/36
101	3	9	4/36
110	3	4	3/36
0110	4	10	3/36
1110	4	3	2/36
1111	4	11	2/36
01110	5	2	1/36
01111	5	12	1/36



$\bar{n} = 3.3056$, $H = 3.2744$ bits (see Hwk. 1, prob. 1).

So the average number of questions that must be asked to determine the outcome is 3.3056 questions, using the questioning scheme corresponding to the Huffman code above. This is the minimum possible average number of questions for any questioning scheme. The entropy of the outcome is 3.2744 bits, so the minimum average number of questions (3.3056) is just slightly larger than the entropy.

2.7. Cover & Thomas 2.4: We must justify the following chain of inequalities to show that $H(g(X)) \leq H(X)$:

$$\left.\begin{aligned} H(X, g(X)) &\overset{(a)}{=} H(X) + H(g(X)|X) \\ &\overset{(b)}{=} H(X) \end{aligned}\right\} (*)$$

$$\left.\begin{aligned} H(X, g(X)) &\overset{(c)}{=} H(g(X)) + H(X|g(X)) \\ &\overset{(d)}{\geq} H(g(X)). \end{aligned}\right\} (**)$$

a: This follows from the chain rule for entropy, which states that $H(X,Y) = H(X) + H(Y|X)$.

b: Since $g(X)$ is a deterministic function of X, knowledge of X completely specifies $g(X)$

$$\Rightarrow H(g(X)|X) = 0.$$

c: Again, the chain rule for entropy

d: Entropy is nonnegative, so $H(X|g(X)) \geq 0$

$$\Rightarrow H(g(X)) + H(X|g(X)) \geq H(g(X))$$

Now equating $H(X, g(X))$, the LHS of $(*)$ and $(**)$, we have

$$H(X) \overset{(a,b)}{=} H(X, g(X)) \overset{(c)}{=} H(g(X)) + H(X|g(X)) \overset{(d)}{\geq} H(g(X))$$

$$\Rightarrow H(g(X)) \leq H(X).$$

2.8. Cover and Thomas 7 (a):

For n coins, there are $2n+1$ possible situations

One of the n coins is heavier:	n
One of the n coins is lighter:	n
All coins have equal weight:	$+\ 1$
	$2n+1$

Each weighing has 3 possible outcomes:

(i) equal weight

(ii) left pan heavier

(iii) right pan heavier

So with k weighings, there are 3^k possible outcomes. In order to determine uniquely which of the $2n+1$ situations above holds, we must have $3^k \geq 2n+1$, because k weighings can determine at most 3^k states. So

$$2n+1 \leq 3^k$$
$$\Rightarrow n \leq \frac{3^k-1}{2}.$$

Another way of looking at this is that each weighing gives at most $\log_2 3$ bits of information. Since there are $2n+1$ possible situations, the maximum entropy is $\log_2(2n+1)$. So we need at least $\frac{\log_2(2n+1)}{\log_2 3}$ weighings to determine the odd coin

$$\Rightarrow k \geq \frac{\log_2(2n+1)}{\log_2 3} \Rightarrow \log_2 3^k \geq \log_2(2n+1)$$

monotone ↑

$$\Rightarrow 3^k \geq 2n+1 \Rightarrow n \leq \frac{3^k-1}{2}.$$

2.9 (Cover and Thomas 2.8)

By the chain rule for entropy, we have

$$H(X_1, \ldots, X_k) = H(X_1) + H(X_2|X_1) + \cdots + H(X_k|X_1, \ldots, X_{k-1}) \quad (*)$$

With replacement: In this case the state of the urn is identical with each draw and hence the outcome of each draw is statistically independent of the previous draws. Thus

$$H(X_j|X_1, \ldots, X_{j-1}) = H(X_j), \quad j = 2, \ldots, k.$$

Since

$$X_j = \begin{cases} \text{red}, & \text{with probability } \dfrac{r}{r+w+b} \\ \text{white}, & \text{with probability } \dfrac{w}{r+w+b} \\ \text{black}, & \text{with probability } \dfrac{b}{r+w+b} \end{cases}$$

we have

$$\begin{aligned} H(X_j) &= -p_X(\text{red}) \log p_X(\text{red}) - p_X(\text{white}) \log p_X(\text{white}) \\ &\quad - p_X(\text{black}) \log p_X(\text{black}) \\ &= \log(r+w+b) - \frac{r}{r+w+b} \log r - \frac{w}{r+w+b} \log w - \frac{b}{r+w+b} \log b \end{aligned}$$

and thus

$$\begin{aligned} H(X_1, \ldots, X_k) &= H(X_1) + \cdots + H(X_k) \\ &= k\left[\log(r+w+b) - \frac{1}{r+w+b}\left[r \log r + w \log w + b \log b\right]\right] \end{aligned}$$

Without Replacement: The unconditional entropy of the j-th draw is still $H(X_j)$ as given above, but now

$$H(X_j|X_1, \ldots, X_{j-1}) \leq H(X_j), \quad \text{with equality iff } X_j \perp\!\!\!\perp X_1, \ldots, X_{j-1}$$

but X_j is clearly dependent on $X_1, \ldots X_{j-1}$. Thus the entropy of drawing without replacement is smaller.

2.10. Cover and Thomas, Ch. 3, Prob. 8:

Let $G_n = [X_1 X_2 \cdots X_n]^{1/n}$. Then taking logs, we have

$$\log G_n = \frac{1}{n} \log [X_1 \cdots X_n]$$

$$= \frac{1}{n} [\log X_1 + \log X_2 + \cdots + \log X_n]$$

$$= \frac{1}{n} \sum_{k=1}^{n} \log X_k = \frac{1}{n} \sum_{k=1}^{n} Y_k, \quad Y_k = \log X_k$$

By the WLLN, we have that

$$\frac{1}{n} \sum_{k=1}^{n} Y_k \xrightarrow{(p)} E\{Y\} \quad \text{as } n \to \infty$$

or equivalently

$$\frac{1}{n} \sum_{k=1}^{n} \log X_k \xrightarrow{(p)} E\{\log X\} \quad \text{as } n \to \infty.$$

So we have $\log G_n \xrightarrow{(p)} E\{\log X\}$ as $n \to \infty$,

or $G_n \xrightarrow{(p)} 2^{E \log_2 X}$ as $n \to \infty$.

Now

$$E\{\log_2 X\} = \frac{1}{2} \log_2 1 + \frac{1}{4} \log_2 2 + \frac{1}{4} \log_2 3$$

$$= \frac{1}{4} \cdot 1 + \frac{1}{4} \log_2 3 = \frac{1 + \log_2 3}{4}$$

$$(= 0.646241) \quad \text{Let me further simplify}$$

$$= \frac{\log_2 2 + \log_2 3}{4} = \frac{\log_2 6}{4}$$

Thus we have $G_n \xrightarrow{(p)} 2^{\frac{1}{4} \log_2 6} = 1.56508.$

(n.b. Recall from the statement of the Strong Law (Borel) in class, we would also have convergence (a.e.), although as I noted in class, we will not generally concern ourselves with the strong law.)

2.11 Cover and Thomas, Ch. 3: Problem 9.

(a) Since $X_1, \ldots, X_n$ are i.i.d. RVs, so are the RVs $Y_1, \ldots, Y_n$, where

$$Y_k = \log q(X_k), \quad k = 1, \ldots, n.$$

Applying the WLLN to Y_k, we get

$$\frac{1}{n}\sum_{k=1}^{n} Y_k \xrightarrow{(p)} E[Y] = E[\log q(X)]$$

(n.b., Technically, for i.i.d Y_k having finite variance the Borel form of the strong law of large numbers (SLLN) holds, and we could say

$$\frac{1}{n}\sum_{k=1}^{n} Y_k \xrightarrow{(a.e.)} E[Y] = E[\log q(X)].$$

But we are not going to stress convergence (a.e.) and the SLLN in this course, even though Cover and Thomas sometimes do. When reading Cover and Thomas, just remember that convergence (a.e) (or with probability 1 (wp1)) implies convergence (p).)

But $$\frac{1}{n}\sum_{k=1}^{n} Y_k = \frac{1}{n}\sum_{k=1}^{n} \log q(X_k) = \frac{1}{n}\log\left[q(X_1)\cdots q(X_n)\right]$$

$$= \frac{1}{n}\log q(X_1, \ldots, X_n)$$

So we have

$$-\frac{1}{n}\log q(X_1, \ldots, X_n) \xrightarrow{(p)} -E[\log q(X)]$$

But $$-E[\log q(X)] = -\sum_{x\in\mathcal{X}} p(x)\log q(x) = -\sum_{x\in\mathcal{X}} p(x)\log\frac{q(x)p(x)}{p(x)}$$

$$= -\sum_{x} p(x)\left[\log\frac{q(x)}{p(x)} + \log p(x)\right]$$

$$= -\sum_{x} p(x)\log\frac{q(x)}{p(x)} - \sum_{x} p(x)\log p(x)$$

$$= \sum_{x} p(x)\log\frac{p(x)}{q(x)} + \sum_{x} p(x)\log\frac{1}{p(x)}$$

$$= \underbrace{D(p\|q)}_{\text{Relative Entropy between } p \text{ and } q.} + \underbrace{H(X)}_{\text{entropy of } X.}$$

(b) Now by the WLLN (or by the SLLN if you insist)

$$\lim_{n\to\infty} -\frac{1}{n}\log\frac{q(X_1,\ldots,X_n)}{p(X_1,\ldots,X_n)} = \lim_{n\to\infty} -\frac{1}{n}\sum_x \log\frac{q(X_i)}{p(X_i)} \xrightarrow[(p)]{\text{(a.e)}} E\left[\log\frac{q(X)}{p(X)}\right]$$

$$= -\sum_x p(x)\log\frac{q(x)}{p(x)} = +\sum_x p(x)\log\frac{p(x)}{q(x)}$$

$$= D(p\|q)$$

Thus we have that the log likelihood ratio

$$\log\frac{q(X_1,\ldots,X_n)}{p(X_1,\ldots,X_n)} \xrightarrow{(p)} -nD(p\|q) \text{ as } n\to\infty.$$

under hypothesis H_0 that $p(X_1,\ldots,X_n)$ is actually in effect.

Aside: Because in statistical hypothesis testing problems using the Neyman–Pearson Lemma, we compare the log-likelihood ratio

$$\log\frac{q(X_1,\ldots,X_n)}{p(X_1,\ldots,X_n)}$$

to a Threshold, because

$$\lim_{n\to\infty}\log\frac{q(X_1,\ldots,X_n)}{p(X_1,\ldots,X_n)} \xrightarrow{(p)} -nD(p\|q),$$

the asymptotic error probability can be related to $D(p\|q)$. For example, the probability of [error Type I] (false alarm can be shown to decrease exponentially in the number of observations n, such that the n sample probability of false alarm can be bounded by

$$P_{FA}(n) \le \exp\{-nD_e(p\|q)\} \text{ as } n\to\infty.$$

Similarly, the probability of a miss (Type II error) can be bounded by

$$P_M(n) \le \exp\{-nD_e(q\|p)\} \text{ as } n\to\infty.$$

We will not cover this topic in ECE 642, but this topic (Large Deviations as Statistical Decision Theory) are discussed in Chapter 12 of Cover and Thomas and Ch. 3 of B.C. Levy, Principles of Signal Detection and Parameter Estimation, Springer, 2010.