1. [10=5+5] Find code parameters, that is, length, size, dimension, rate and the minimum distance of a code \( C \) when \( C \) is given below.

(a) \( C \) is the repetition code
\[
C = \{0000000, 1111111\}.
\]

(b) \( C \) is a single parity check code
\[
C = \{x = x_1x_2 \cdots x_6x_7 \mid \sum_{i=1}^{7} x_i \equiv 0 \pmod{2}\}
\]

Solutions

(a) Here \( F = \{0, 1\} \). The length is \( n = 7 \). The size is \( M = 2 \). The dimension is \( k = \log_{|F|} M = \log_2 2 = 1 \) and the rate is \( R = k/n = 1/7 \). The minimum distance is \( d = 7 \).

(b) Here \( F = \{0, 1\} \). The length is \( n = 7 \). The size is \( M = 2^6 \). The dimension is \( k = \log_{|F|} M = \log_2 2^6 = 6 \). The rate is \( R = k/n = 6/7 \). The minimum distance is \( d = 2 \).

2. [10=2+6+2]

(a) Define the Hamming distance.

(b) Show that the Hamming distance obeys the triangle inequality: for any codewords \( x, y, z \in F^n \),
\[
d(x, z) \leq d(x, y) + d(y, z).
\]

(c) When does the equality hold?

Solutions Let \( F \) be an alphabet.

(a) The Hamming weight \( w(x) \) of a word \( x \in F^n \) is defined by
\[
w(x) := \text{the number of non-zero components in } x.
\]

The Hamming distance \( d(x, y) \) between two words \( x, y \in F^n \) is defined by
\[
d(x, y) := w(y - x) = \text{the number of components on which } x \text{ and } y \text{ differ.}
\]

If \( F = \{0, 1\} \), then \( d(x, y) = w(y + x) \).
(b) Let \( x = x_1 x_2 \cdots x_n, y = y_1 y_2 \cdots y_n \) and \( z = z_1 z_2 \cdots z_n \) be three arbitrary words in \( F^n \). Let

\[
U := \{ i \mid x_i \neq z_i \}, \quad S := \{ i \mid x_i \neq y_i \}, \quad T := \{ i \mid y_i \neq z_i \}.
\]

Then

\[
|U| = d(x, z), \quad |S| = d(x, y), \quad |T| = d(y, z).
\]

It is easily seen that

\[
U \subseteq S \cup T
\]

since if \( x_i \neq z_i \), then we must either have \( x_i \neq y_i \) or \( y_i \neq z_i \) (contra-positively, if \( x_i = y_i \) and \( y_i = z_i \), then \( x_i = z_i \)).

Hence

\[
d(x, z) = |U| \leq |S \cup T| = |S| + |T| - |S \cap T| \leq |S| + |T| = d(x, y) + d(y, z).
\]

(c) The equality holds if and only if \( U = S \cup T \) and \( S \cap T = \emptyset \). In other words,

\[
d(x, z) = d(x, y) + d(y, z)
\]

if and only if \( S \) and \( T \) form a partition of \( U \).

3. \([40=10+10+10+10]\) The international Standard Book Number (ISBN) system, used to uniquely identify books, is a non-binary code defined over the 11-letter alphabet \( \{0, 1, 2, \cdots, 9, X\} \), where \( X \) represents the number 10. An ISBN codeword \( c_1 c_2 \cdots c_{10} \) is ten digits in length. The first nine digits are “information digits” that record data such as publisher, title and edition. The tenth digit is a check digit, chosen so that

\[
\sum_{i=1}^{10} i \cdot c_i \equiv 0 \pmod{11}.
\]

(a) In this application, the common types of error are digits being wrong, and transpositions, in which some two digits \( c_i \) and \( c_j \) are interchanged. Determine the error detection/correction capability of the ISBN code against these types of errors.

(b) Show that the ISBN code is a linear code.

(c) Determine the dimension and minimum distance of the ISBN code.

(d) Check if 0521845041 is a valid ISBN. The following ISBN is received with smudges. What is the missing digit \( \square \) in 013139\( \square \)99?

Solutions
The ISBN code can detect all error patterns in which a single digit is in error or two digits have been transposed. It can fail to detect two or more digits errors or a combination of digit errors and transpositions, if such an error pattern is not equivalent to a transposition. Since we are only claiming detection of certain kinds of error patterns, we only need to show that in such cases the resulting words do not satisfy the parity check condition defining the code. In what follows, it is to be understood that all operations are being performed modulo 11.

For \( x = x_1x_2 \cdots x_{10} \), \( x_i \in \{0, 1, 2, \ldots, 9, X\} \), define

\[
h(x) := \sum_{i=1}^{10} i \cdot x_i.
\]

Thus, \( x \) is a valid ISBN codeword if and only if \( h(x) = 0 \). Let \( c = c_1c_2 \cdots c_{10} \) be a valid ISBN codeword, and let \( \hat{c} = \hat{c}_1\hat{c}_2 \cdots \hat{c}_{10} \) be the word that results when \( c \) is affected by an error pattern in which the \( j \)-th digit of \( c \) is affected. To be precise, \( \hat{c}_j \neq c_j \) and \( \hat{c}_i = c_i \) for \( i = 1, 2, \ldots, 10, i \neq j \). Note that, since \( h(c) = 0 \), we have

\[
h(\hat{c}) = h(\hat{c}) - h(c) = \sum_{i=1}^{10} i \cdot (\hat{c}_i - c_i) = j \cdot (\hat{c}_j - c_j) \neq 0
\]

since \( j \neq 0 \) and \( \hat{c}_j \neq c_j \). Thus, \( \hat{c} \) is not a valid ISBN codeword, which shows that all error patterns in which a single digit is in error can be detected.

Next, let \( \hat{c} \) be the word that results when \( c \) is affected by an error pattern in which the \( j \)-th and \( \ell \)-th digits are transposed. Thus

\[
\hat{c}_j = c_\ell, \hat{c}_\ell = c_j \quad \text{for some} \quad j \neq \ell \quad \text{for} \quad i \neq \ell.
\]

In this situation, we have

\[
h(\hat{c}) = h(\hat{c}) - h(c) = \sum_{i=1}^{10} i \cdot (\hat{c}_i - c_i)
\]

\[
= j(c_\ell - c_j) + \ell(c_j - c_\ell) = (j - \ell)(c_\ell - c_j).
\]

Since \( j \neq \ell \), we can have

\[
h(\hat{c}) = 0 \quad \text{if and only if} \quad c_\ell = c_j.
\]

In this case, the "errors" have not actually changed the codeword since \( \hat{c} \) is the same as \( c \). Thus, if a transposition error actually results in a \( \hat{c} \) different from \( c \), then \( h(\hat{c}) \) must be nonzero, and hence the transposition error is detected.

Double (or higher order) error patterns that are not equivalent to single transpositions are not detectable in general. In fact, for any codeword \( c \) and any pair
of positions $i$ and $j$, there are several valid codewords $\hat{c}$ that differ from $c$ in exactly the positions $i$ and $j$. (In fact, 10 codewords.) For instance, if $c$ is the all zeros codeword $00\cdots0$, then the ten valid codewords $\hat{c}$ that differ from $c$ in the first two positions are $(x, 5x, 0, 0, 0, 0, 0, 0, 0, 0), x = 1, 2, \ldots, X$. Thus, double-error patterns affecting $c = 00\cdots0$ that results in one of these ten words will go undetected.

A similar analysis holds for higher-order error patterns.

(b) First $\mathbb{F}_{11} = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ becomes a field. Addition and multiplication are defined modulo 11:

<table>
<thead>
<tr>
<th>+</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>9</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>10</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\times$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>8</td>
<td>10</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>3</td>
<td>6</td>
<td>9</td>
<td>1</td>
<td>4</td>
<td>7</td>
<td>10</td>
<td>2</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>4</td>
<td>8</td>
<td>1</td>
<td>5</td>
<td>9</td>
<td>2</td>
<td>6</td>
<td>10</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>5</td>
<td>10</td>
<td>4</td>
<td>9</td>
<td>3</td>
<td>8</td>
<td>2</td>
<td>7</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>6</td>
<td>1</td>
<td>7</td>
<td>2</td>
<td>8</td>
<td>3</td>
<td>9</td>
<td>4</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>7</td>
<td>3</td>
<td>10</td>
<td>6</td>
<td>2</td>
<td>9</td>
<td>5</td>
<td>1</td>
<td>8</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>8</td>
<td>5</td>
<td>2</td>
<td>10</td>
<td>3</td>
<td>7</td>
<td>4</td>
<td>1</td>
<td>9</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>9</td>
<td>7</td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>10</td>
<td>8</td>
<td>6</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>10</td>
<td>9</td>
<td>8</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

The ISBN code is a subspace of $\mathbb{F}_{11}^{10}$.

The parity check constraint defining the code is a linear over the finite field $\mathbb{F}_{11}$. A parity-check matrix for the code is

$$H = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{bmatrix}.$$
(c) An ISBN code will have block length \( n = 11 \). The code size is \( M = 11^{10} \). The dimension is \( k = \log_{|F|} M = \log_{11} 11^{10} = 10 \) and the rate is \( R = k/n = 10/11 \). The minimum distance is \( d = 2 \). This is because, there is no codeword of weight 1. In fact, \( c = c_1c_2 \cdots c_{10} \) is a codeword if and only if
\[
\sum_{i=1}^{10} i \cdot c_i = c_1 + 2c_2 + 3c_3 + \cdots + 9c_9 + 10c_{10} \equiv 0 \pmod{11}.
\]
But if only one component is nonzero then 11 divides \( ic_i \) for some \( i, 1 \leq i \leq 10 \). This forces 11 divides \( c_i \) which implies that \( c_i = 0 \). However, there is a codeword of weight 2, for example, \((1, 0, 0, 0, 0, 0, 0, 0, 0, 1)\). Hence the minimum distance is \( d = 2 \).

(d) We check if
\[
1\cdot0+2\cdot5+3\cdot2+4\cdot1+5\cdot8+6\cdot4+7\cdot5+8\cdot0+9\cdot4+10\cdot1 = 165 = 11\cdot15 \equiv 0 \pmod{11}.
\]
So it is a valid ISBN.

For the second code to be an ISBN, it must satisfy
\[
1 \cdot 0 + 2 \cdot 1 + 3 \cdot 3 + 4 \cdot 1 + 5 \cdot 3 + 6 \cdot 9 + 7 \cdot \Box + 8 \cdot 3|9 \cdot 9 + 10 \cdot 9 \equiv 0 \pmod{11}.
\]
So
\[
7\Box \equiv 5 \pmod{11} \iff \Box = 5.
\]

4. [20=10+5+5] A \( k \)-bit message is to be transmitted across a *memoryless* binary symmetric channel (BSC) with crossover probability \( p = 0.1 \). The error-protection scheme used is an \( r \)-fold repetition code, in which each message bit is repeated \( r \) times before transmission, and a majority rule is used to decoding at the channel output. Given a value of \( k \), let \( r(k) \) denote the minimum value of \( r \) for which the probability that the entire \( k \)-bit message will be received and decoded correctly is at least 0.999.

(a) Determine \( r(1) \), \( r(2) \) and \( r(3) \).

(b) Show that \( r(k) \) is a non-decreasing function of \( k \), that is, if \( k_1 \leq k_2 \) then \( r(k_1) \leq r(k_2) \).

(c) Show that \( \lim_{k \to \infty} r(k) = \infty \).

**Solutions**
(a) The memoryless BSC in question has error probability $p = 0.1$. The probability that a $k$-bit message, encoded using an $r$-fold repetition code and transmitted across the channel, is received and decoded successfully is $(P_C(r))^k$ where $P_C(r)$ is given by the expression

$$P_C(r) = \sum_{j=0}^{\lfloor(r-1)/2\rfloor} \binom{r}{j} p^j (1-p)^{r-j} = \sum_{j=0}^{\lfloor(r-1)/2\rfloor} \binom{r}{j} (0.1)^j (0.9)^{r-j}.$$ 

Thus,

$$r(k) = \min\{r \mid P_C(r) \geq (0.999)^{1/k}\}.$$ 

A table listing the values of $P_C(r)$ for $1 \leq r \leq 20$ is given here.

<table>
<thead>
<tr>
<th>$r$</th>
<th>$P_C(r)$</th>
<th>$r$</th>
<th>$P_C(r)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.900000000</td>
<td>11</td>
<td>0.99970429</td>
</tr>
<tr>
<td>2</td>
<td>0.810000000</td>
<td>12</td>
<td>0.99945877</td>
</tr>
<tr>
<td>3</td>
<td>0.972000000</td>
<td>13</td>
<td>0.99990071</td>
</tr>
<tr>
<td>4</td>
<td>0.947700000</td>
<td>14</td>
<td>0.99981864</td>
</tr>
<tr>
<td>5</td>
<td>0.991440000</td>
<td>15</td>
<td>0.99996638</td>
</tr>
<tr>
<td>6</td>
<td>0.984150000</td>
<td>16</td>
<td>0.99993867</td>
</tr>
<tr>
<td>7</td>
<td>0.997272000</td>
<td>17</td>
<td>0.99998854</td>
</tr>
<tr>
<td>8</td>
<td>0.99497565</td>
<td>18</td>
<td>0.99997912</td>
</tr>
<tr>
<td>9</td>
<td>0.99910908</td>
<td>19</td>
<td>0.99999607</td>
</tr>
<tr>
<td>10</td>
<td>0.99836506</td>
<td>20</td>
<td>0.99999285</td>
</tr>
</tbody>
</table>

From the above table, it is clear that $r(1) = 0$, since 9 is the least value for $r$ for which $P_C(r) \geq 0.999$. Similarly, from the facts that $0.999^{1/2} = 0.99949987$ and $0.999^{1/3} = 0.9996656$, we see that $r(2) = r(3) = 11$. The table below lists $r(k)$ for the first few values of $k$.

<table>
<thead>
<tr>
<th>$k$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>19</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r(k)$</td>
<td>9</td>
<td>11</td>
<td>11</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>13</td>
<td>15</td>
</tr>
</tbody>
</table>

(b) The above table indicates that $r(k)$ is a non-decreasing function of $k$. Now we will give a rigorous mathematical proof. For this, define for $k = 1, 2, 3, \cdots$ the sets

$$R(k) := \{ r \geq 1 \mid P_C(r) \geq 0.999^{1/k} \}.$$ 

Note that $r(k)$ is, by definition, the least element of $R(k)$. It is sufficient to show that

$$r(k) \leq r(k+1) \quad \text{for all } k \geq 1.$$
Now, for all \( k \geq 1 \), since \( 0.999^{1/k} < 0.999^{1/(k+1)} \), we see that for any \( r \) satisfying \( P_C(r) \geq 0.999^{1/(k+1)} \) also automatically satisfies \( P_C(r) \geq 0.999^{1/k} \). It follows that

\[
R(k + 1) \subseteq R(k) \quad \text{for all } k \geq 1. 
\]

Therefore, the least element of \( R(k + 1) \) can be no smaller than the least element of \( R(k) \). This shows that \( r(k + 1) \geq r(k) \) as claimed.

(c) Since \( r(k) \) is a non-decreasing function, it must have a limit (which may be \( \infty \)).

Suppose that

\[
\lim_{k \to \infty} r(k) = \rho < \infty. 
\]

Since \( r(k), k = 1, 2, 3, \cdots \) is a sequence of integers, \( \rho \) must be an integer and \( r(k) = \rho \) for sufficiently large \( k \). Therefore, \( P_C(r) \geq 0.999^{1/k} \) for all sufficiently large \( k \). Letting \( k \to \infty \), we see that this is possible only when \( P_C(r) \geq 1 \). However, for any finite positive integer \( \rho \), it is clear from the expression for \( P_C(r) \) given above that \( 0 < P_C(r) < 1 \). This is a contradiction. Thus, \( \lim_{k \to \infty} r(k) = \rho \) cannot be finite.

5. [20=10+10] Let \( C \) be a block code over an alphabet of size \( q \).

(a) Show that over a \( q \)-ary symmetric channel with crossover probability \( p < 1 - \frac{1}{q} \), maximum-likelihood decoding of \( C \) is equivalent to minimum distance decoding.

(b) Show that the maximum a-posteriori probability (MAP) decoder minimizes among all possible decoders for \( C \), the average probability of decoding error \( \overline{P_{err}} \). Here

\[
\overline{P_{err}} = \sum_{c \in C} P_{err}(c)p(c) 
\]

where for each \( c \in C \), \( P_{err}(c) \) is the probability of decoding error given that \( c \) was transmitted, and \( p(c) \) is the a-priori probability of transmitting \( c \).

Solutions

(a) The memoryless \( q \)-ary symmetric channel is specified by a triple \((F, F, \text{Prob})\). Here \( F = \mathbb{F}_q \) is the finite field of \( q \)-elements, and \( \text{Prob} \) is the channel probability transition function defined via a crossover probability \( p \in [0, 1] \) as follows: For \( x, y \in F \)

\[
\text{Prob}(y|x) := \begin{cases} 
1 - p & \text{if } x = y \\
\frac{p}{(q - 1)} & \text{if } x \neq y
\end{cases}
\]
and for any \( m > 1 \) and any \( x = (x_1, x_2, \ldots, x_m) \), \( y = (y_1, y_2, \ldots, y_m) \in F^m \),

\[
\text{Prob}(y|x) = \prod_{i=1}^{m} \text{Prob}(y_i|x_i).
\]

The maximum-likelihood decoding (MLD) rule is: given a received word \( y \in F^m \), decode to a word \( c \in C \) that maximizes \( \text{Prob}(y|c) \). It is easily verified that

\[
\text{Prob}(y|c) = (1 - p)^{m-d} \left( \frac{p}{q-1} \right)^d = (1 - p)^m \left( \frac{p}{(1-p)(q-1)} \right)^d,
\]

where \( d = d(y, c) \) is the Hamming distance between \( y \) and \( c \). Now observe that

\[
\frac{p}{(1-p)(q-1)} < 1 \iff p < (1-p)(q-1) \iff p + p(q-1) < q - 1
\]

\[
\iff p < \frac{q-1}{q} = 1 - \frac{1}{q}.
\]

Hence for \( p < 1 - \frac{1}{q} \), we have that \( \text{Prob}(y|c) \) is maximized (for fixed \( p, n \)) precisely when \( d = d(y, c) \) is minimized. Hence for \( p \) in this range, MLD is equivalent to minimum-distance decoding.

(b) We consider MAP decoding of \( C \) over an arbitrary discrete probability channel \((F, Y, \text{Prob})\) where \( F = \mathbb{F}_q \). Recall that for any received word \( y \), the MAP decoding rule is: decode to a \( c \in C \) that maximizes the probability \( P(c \text{ transmitted} | y \text{ received}) \). Also recall that for any \( y \), this rule minimizes, among all decoders for \( C \), the probability of decoding error given that \( y \) was received. (A quick proof of this: for an arbitrary decoder \( D : Y^m \rightarrow C \), \( P(\text{decoding error} | y \text{ received}) \) is given by

\[
\sum_{c \in C; D(y) \neq c} P(c \text{ transmitted} | y \text{ received})
\]

\[
= \sum_{c \in C} P(c \text{ transmitted} | y \text{ received}) - P(D(y) \text{ transmitted} | y \text{ received})
\]

and it is the MAP decoder that maximizes the probability \( P(D(y) \text{ transmitted} | y \text{ received}) \) among all decoders \( D \).

Now, for an arbitrary decoder \( D : Y^m \rightarrow C \), consider the average probability of decoding error \( \overline{P}_{err} = \sum_{c \in C} P_{err}(c)p(c) \), where

\[
P_{err}(c) = \sum_{y : D(y) \neq c} \text{Prob}(y|c)
\]
is the probability of decoding error given that $c$ is transmitted, and $p(c)$ is the a-priori probability of transmitting $c$. Thus, we have
\[
P_{err} = \sum_{c \in C} \sum_{y : D(y) \neq c} \text{Prob}(y|c)p(c) = \sum_{y} \sum_{c \in C} \text{Prob}(y|c)p(c)
\]
the second equality just being a change in the order of summation. Using the fact that
\[
\text{Prob}(y|c)p(c) = P(c \text{ transmitted} | y \text{ received})P(y \text{ received}),
\]
the above reduces to
\[
P_{err} = \sum_{y \text{ received}} P(y \text{ received}) \sum_{c \in C : D(y) \neq c} P(c \text{ transmitted} | y \text{ received})
\]
\[
= \sum_{y} P(y \text{ received})P(\text{decoding error} | y \text{ received}).
\]
We have already noted that for each $y$, the MAP decoder minimizes $P(\text{decoding error} | y \text{ received})$. Therefore, it also minimizes $\sum_{y} P(y \text{ received})P(\text{decoding error} | y \text{ received})$. 
