Mathematics 342. Solutions to Problem Set 1. January 2019

(1) How many errors can be detected with the following \( q \)-ary codes? How many errors can be corrected? Explain your answers.

(a) \( C_1 = \{(0, 0, 0, 0, 1), (0, 1, 1, 1, 1), (1, 1, 1, 0, 0)\} \). Here \( q = 2 \).
(b) \( C_2 = \{(0, 1, 2, 0, 1, 2), (2, 1, 0, 2, 1, 0), (2, 2, 2, 2, 2)\} \). Here \( q = 3 \).
(c) \( C_3 = \{(0, 1, 2, 3, 4, 5, 6), (1, 2, 3, 4, 5, 6, 0), (2, 3, 4, 5, 6, 0, 1), (3, 4, 5, 6, 0, 1, 2), (4, 5, 6, 0, 1, 2, 3), (5, 6, 0, 1, 2, 3, 4), (6, 0, 1, 2, 3, 4, 5)\} \). Here \( q = 7 \).

**Solution:** The number of errors detected or corrected can be deduced from the minimum distance of the code. If \( d \) is the minimum distance then it can detect up to \( d - 1 \) errors and correct up to \( \lfloor (d - 1)/2 \rfloor \) errors.

(a) By inspection \( d(C_1) = 3 \). Thus \( C_1 \) can detect up to 2 errors and correct 1.
(b) By inspection \( d(C_2) = 4 \). Thus \( C_2 \) can detect up to 3 errors and correct 1.
(c) No two codewords agree in any position. Thus \( d(C_3) = 7 \). This code can detect up to 6 errors and correct up to 3.

(2) Assume the code \( C_1 \) from Problem 1 was used in transmission, and the following words were received. Decode each of these words using the nearest neighbour decoding algorithm. (The incomplete decoding version: if there is more than one nearest neighbour, declare an error.)

(a) \( (0, 0, 1, 1, 1) \), (b) \( (1, 1, 0, 0, 0) \), (c) \( (1, 1, 1, 1, 1) \), (d) \( (1, 0, 1, 0, 1) \).

**Solution:** In each case look for the word in \( C_1 \) that is closest to the received word.

(a) \( (0, 0, 1, 1, 1) \mapsto (0, 1, 1, 1, 1) \).
(b) \( (1, 1, 0, 0, 0) \mapsto (1, 1, 1, 0, 0) \).
(c) \( (1, 1, 1, 1, 1) \mapsto (0, 1, 1, 1, 1) \).
(d) \( (1, 0, 1, 0, 1) \) is equidistant from \( (0, 0, 0, 0, 1) \) and \( (1, 1, 1, 0, 0) \). Declare an error.

Recall that the triangle inequality for the Hamming distance says that

\[ d(a, b) + d(b, c) \geq d(a, c). \]

Here \( a, b \) and \( c \) are \( q \)-ary words of length \( n \). We will say that \( b \) lies between \( a \) and \( c \) if equality holds in the above formula, i.e.,

\[ d(a, b) + d(b, c) = d(a, c). \]

The purpose of the next four exercises is to discover and prove a formula for the number of words that lie between \( a \) and \( c \).
(3) How many words lie between \(a\) and \(a\)?

**Solution:** Suppose \(b = (b_1, \ldots, b_n)\) is between \(a\) and \(a\). Then by \((\ast)\),
\[
d(a, b) + d(b, a) = d(a, a) = 0.
\]
This simplifies to \(2d(a, b) = 0\) or equivalently, \(d(a, b) = 0\). Thus \(b = a\).

Answer: There is exactly one word between \(a\) and \(a\).

(4) Assume \(q = 2\) and \(n = 3\). How many words lie
(i) between \((0, 0, 0)\) and \((1, 1, 1)\)?
(ii) between \((0, 0, 0)\) and \((1, 1, 0)\)?
(iii) between \((0, 0, 0)\) and \((1, 0, 0)\)?

**Solution:** In each case there are 8 possibilities for \(b = (b_1, b_2, b_3)\), where each
of \(b_1, b_2, b_3\) is either 0 or 1. For each of these words we can compute both sides of
\((\ast)\) and see whether equality \((\ast)\) holds or not. Here are the answers obtained by
this method. \(b\) lies between \(a\) and \(c\) if and only if it does.

(a) Here \(a = (0, 0, 0)\) and \(c = (1, 1, 1)\). All 8 words in \(F_2^3\),
\[
(0, 0, 0), (0, 0, 1), (0, 1, 0), (0, 1, 1), (1, 0, 0), (1, 0, 1), (1, 0, 1), (1, 1, 1),
\]
lie between \(a\) and \(c\).

(b) Here \(a = (0, 0, 0)\) and \(c = (1, 1, 0)\). Only 4 words \((0, 0, 0), (0, 1, 0), (1, 0, 0), (1, 1, 0)\),
lie between \(a\) and \(c\).

(c) Here \(a = (0, 0, 0)\) and \(c = (1, 0, 0)\). Only 2 words, \((0, 0, 0)\) and \((1, 0, 0)\), lie
between \(a\) and \(c\).

(5) Suppose \(q = 2\), \(a\) and \(c\) are binary words of length \(n\) and \(d(a,c) = d\). Based on
your answers in Problems 3 and 4, guess a formula for the number of binary words
of length \(n\) lying between \(a\) and \(c\). Prove this formula.

(6) We now allow \(q\) to be arbitrary and ask the same question as in Problem 5. Given
\(q\)-ary words \(a\) and \(b\) of length \(n\) and at Hamming distance \(d\), how many \(q\)-ary
words of length \(n\) lie between \(a\) and \(c\)? Prove your answer.

**Solutions to Problems 5 and 6:** Let \(a = (a_1, \ldots, a_n)\), \(b = (b_1, \ldots, b_n)\), and
and \(c = (c_1, \ldots, c_n)\).

Then
\[
d(a, b) + d(b, c) = d(a, c)
\]
can be rewritten as
\[
\sum_{i=1}^{n} (d(a_i, b_i) + d(b_i, c_i)) = \sum_{i=1}^{n} d(a_i, c_i).
\]
Here for any \(s, t \in F_q\),
\[
d(s, t) = \begin{cases} 0, & \text{if } s = t, \\ 1, & \text{if } s \neq t. \end{cases}
\]
In other words, \( d(s, t) \) is the usual Hamming distance if we view \( s \) and \( t \) as \( q \)-ary words of length 1.

By the triangle inequality for words of length 1,
\[
d(a_i, b_i) + d(b_i, c_i) \geq d(a_i, c_i)
\]
for every \( i = 1, \ldots, n \). Thus \((\star)\) holds if and only if
\[
d(a_i, b_i) + d(b_i, c_i) = d(a_i, c_i)
\]
for every \( i \).

If \( a_i = c_i \), then the right hand side of \((\star\star)\) is 0. Thus both terms on the left hand side should be 0, i.e., \( b_i = a_i = c_i \).

If \( a_i \neq c_i \), the right hand side of \((\star\star)\) is 1. In this case one of the terms on the left hand side is 1, and the other is 0. That is, \( b_i = a_i \) or \( c_i \).

There are exactly \( d \) positions where \( a_i \neq c_i \), and \( (n - d) \) positions where \( a_i = c_i \). Thus the words \( b \) that lie between \( a \) and \( c \) can be described as follows. In the \( n - d \) positions where \( a \) and \( c \) agree, \( b \) has to agree with both of them. In the \( d \) positions, where they disagree, there are two possibilities for the entry \( b_i \) of \( b \), it can be either \( a_i \) or \( c_i \).

This shows that there are exactly \( 2^d \) codewords between \( a \) and \( c \).

Note that the answer depends only on \( d \), not on \( n \) or \( q \). Note also that this formula gives the answers to Problems 3 and 4 as special cases. I assigned Problems 3 and 4 to build up your intuition for Problems 5 and 6.

(7) How many binary words of length 6 are at Hamming distance
(a) at Hamming distance 6 from \((1, 0, 1, 0, 1, 0)\)?
(b) at Hamming distance 5 from \((1, 0, 1, 0, 1, 0)\)?

**Solution:** (a) Suppose \( b \) is at distance 6 from \( a = (1, 0, 1, 0, 1, 0) \). Then \( b \) has to differ from \( a \) in every position. There is only one such word \( b = (0, 1, 0, 1, 0, 1) \).

(b) Here \( b \) has to agree with \( a \) in exactly one position and disagree in the remaining 5. There are 6 choices for the position, where they agree, so there are exactly 6 such words \( b \). They are \((1, 1, 0, 1, 0, 1), (0, 0, 0, 1, 0, 1), (0, 1, 1, 1, 0, 1), (0, 1, 0, 0, 0, 1), (0, 1, 0, 1, 1, 1), and (0, 1, 0, 1, 0, 0)\).

(8) Is the code \( C_3 \) in Problem 1 equivalent to the 7-ary repetition code
\[
C_4 = \{(0, \ldots, 0), (1, \ldots, 1), \ldots (6, \ldots, 6)\}
\]
of length 7? Prove your answer.

**Solution:** Yes, these two codes are equivalent. To see this, permute the elements in the second column of \( C_4 \) via the permutation \( f \) taking \( i \) to \( i + 1 \) modulo 7. That is, we count 7 as 0, 8 as 1, etc., so \( f \) takes 0 to 1, 1 to 2, \ldots, 5 to 6 and 6 back to 0.

Similarly in the third column send \( i \) to \( i + 2 \) modulo 7, in the 4th column, \( i \) to \( i + 3 \), etc. This will change \( C_4 \) into \( C_3 \).