Chapter 1

The concept of coding

1.1 Bitstrings and binary operations

Exercises 1.1

1.1.1. *Compute the sum of* 11001 *and* 01110

$$11001 + 01110 = 10111$$

1.1.2. *Assume* 000000 *was sent and two errors occurred. List all possible received messages.*

There are $\binom{6}{2} = 15$ possibilities: all bitstrings of length 6 with precisely two ones.

1.1.3. *Let* $x = 1101$ *be the message to be sent. Encode* x *using the repetition code of length* 3.

Encode

$$x = 1101 \longrightarrow 111\ 111\ 000\ 111.$$

1.1.4. *Assume the repetition code of length* 3 *is used and* 000110111101 *is received. What is the result of decoding?*

$$000110111101 \longrightarrow 0111.$$

1.1.5. *Why does the Morse code represent letters E and T by bitstrings of length* 1, *whereas letters like Q,V,Z are represented by longer bitstrings?*

Because E and T are frequent letters in standard text, whereas Q,V,Z typically occur less often.

1.2 The Hamming distance

Exercises 1.2

1.2.1. *Compute* $d(11001, 01110)$ *and* $d(0000, 0110)$.

$$d(11001, 01110) = 4,\ d(0000, 0110) = 2$$

1.2.2. *Find* $wt(00110)$ *and* $wt(10111)$.

$$wt(00110) = 2,\ wt(10111) = 4.$$

1.2.3. *List all vectors in* $\mathbb{F}_2^6$ *at distance* 3 *from* 111000.

The list consists of $\binom{6}{3} = 20$ vectors: all the vectors which have the same weight in the first and in the last half of the coordinates.

1.2.4. *The alphabet has* 26 *letters. If we want to represent all possible words of length* ≤ 3 *(all letters, pairs of letters and triples of letters) as bitstrings of the same length* n*, what is the smallest* n *such that this is possible?*

There are $26^3 = 17,576$ words of length 3, $26^2 = 676$ words of length 2 and of course 26 words of length 1. We have to choose n such that $2^n \geq 26^3 + 26^2 + 26 = 18,278$. The minimal value of n for which this is satisfied is $n = 15$.

1.2.5. *Prove that the Hamming distance is a metric.*

Only the triangle inequality is in doubt. For arbitrary x, y, z we have to prove

$$d(x, z) \leq d(x, y) + d(y, z).$$

Addition of a fixed string does not change the distance: $d(x, y) = d(x+c, y+c)$. This follows directly from the definition of the Hamming distance. We can therefore assume that $y = 0$ is the 0-string. What we have to prove is

$$d(x, z) \leq wt(x) + wt(z).$$

This is clear from the definition.

1.2.6. *Assume* x *is sent and* $y = x + e$ *is received. What can we say about* $d(x, y)$ *and about* $wt(e)$ *if not more than* 3 *errors have occurred?*

If x is sent and $y = x + e$, then $d(x, y) = wt(e)$. This is ≤ 3 if no more than 3 errors have occurred.

1.3 Binary codes

1.3.1. *If we want to correct* 8 *bit errors, what would the minimum distance of the code have to be?*

In order to correct 8 bit errors, distance $d \geq 17$ is needed.

1.3.2. *Using our code* $(6, 8, 3)_2$*, decode the following received vectors:*

$$111100,\ 111011,\ 000001,\ 011110.$$

Decode:

$$111100 \mapsto 111000,\ 111011 \mapsto 110011,\ 000001 \mapsto 000000,\ 011110 \mapsto 011110.$$

1.3.3. *Does a code* $(5, 6, 3)_2$ *exist?*

It does not exist. In fact we can prove the stronger statement that a $(5, 5, 3)_2$-code cannot exist.

Assume such a code C does exist. Assume at first there are two codewords at distance 5. Adding the same word to all codewords we can assume that 00000 and 11111 belong to C. There is no further word at distance ≤ 3 from both, contradiction.

Assume next there are two words at distance 4 in C. These can be chosen without restriction as $x = 00000$ and $y = 11110$. Every word ending in 0 has distance < 3 from either x or y. All remaining words in C must therefore end in 1 and they must have weight 3. We can choose without restriction $z = 11001$ as a word of C. The fourth word is 00111, and there is no fifth word. Contradiction.

We have shown that any two different words in C must be at distance 3. This allows us to choose as codewords $x = 00000,\ y = 11100$. There is no further word of weight 3 at distance 3 from y.

1.4 Error-correcting codes in general

Exercises 1.4

1.4.1. *Find the smallest length* n *such that an* $(n, 27, 2)_3$ *exists.*

We have seen a $(4, 27, 2)_3$. Can a $(3, 27, 2)_3$ exist? As it has 27 codewords it would have to consist of all ternary triples. This is impossible as the minimum distance is only 1. The answer is 4.

1.4.2. *Prove the following: if there is an* $(n, M, d)_q$*, then there is an* $(n+1, M, d)_q$*.*

Write out the words of the $(n, M, d)_q$, lengthen each codeword by appending an arbitrary $(n+1)$-st coordinate.

1.4.3. *Prove: If there is an* $(n, M, d)_{q-1}$*, then there is an* $(n, M, d)_q$*.*

An $(n, M, d)_{q-1}$ can be seen as an $(n, M, d)_q$. The entries are from an alphabet of size $q-1$, which can be seen as a subset of an alphabet of size q.

1.5 The binary symmetric channel

Exercises 1.5

1.5.1. *Compute the probability that no more than* 1 *error occurs in the transmission of a bitstring of length* 10*, when the bit error probability is* $p = 10^{-3}$*, or* $p = 10^{-4}$*, or* $p = 10^{-5}$*.*

The probability of no errors is $(1-p)^{10}$, the probability of exactly one error is $10p(1-p)^9$. The probability of at most one error is therefore

$$P = (1-p)^{10} + 10p(1-p)^9 = (1+9p)(1-p)^9 = (1+9p)(1-9p+36p^2-+\dots),$$

which is

$$P = 1 - 45p^2 + \text{ higher terms}$$

For $p = 10^{-3}$ we have $P \sim 1 - 0.000045 = 0.999955$. For $p = 10^{-4}$ this is $P \sim 1 - 0.00000045 = 0.99999955$. In case $p = 10^{-5}$ finally $P \sim 1 - 0.0000000045 = 0.9999999955$.

1.5.2. *Describe a generalization of the BSC from the binary to the general q-ary case.*

The alphabet $\mathcal{A}$ has q elements. The most obvious generalization of the BSC is the following: fix a probability p, the error probability. When $x \in \mathcal{A}$ is sent, define the probability that x' is received as $1-p$ if $x' = x$, as $p/(q-1)$ if $x' \neq x$.

1.5.3. *Sketch a formal picture of the binary erasure channel, analogous to Figure 2.3 for the BSC.*

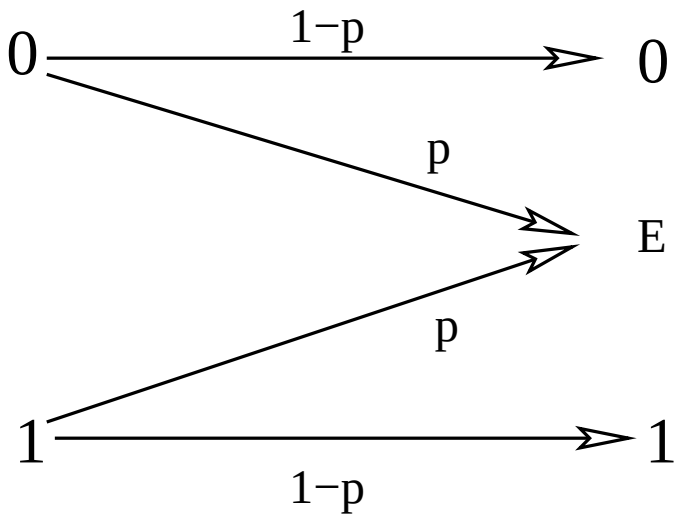

FIGURE 1.1: The erasure channel

1.5.4. *Show that if a code of minimum distance d is used for the erasure channel, then any $d-1$ errors can be corrected.*

Assume the 0-word was sent and errors occur in the first $d-1$ coordinates. The erasure channel is so kind to tell us that the final $n-(d-1)$ zeroes are correct whereas there may be problems with the first $d-1$ bits. The 0-word is then the only codeword which explains this behaviour as any other codeword would have to have at least one entry 1 in the last coordinate section.

1.5.5. *Use the subset interpretation of the binomials to prove the binomial formula*

$$(a+b)^n = \sum_{i=0}^{n} \binom{n}{i} a^i b^{n-i}.$$

Let $S = \{1, 2, \ldots, n\}$ and imagine $(a+b)^n$ as a product of n copies of $(a+b)$ where factor number j corresponds to $j \in S$. Multiplying out yields 2^n products. Associate to the choice of a from factor number j the choice of $j \in S$, to the choice of b to not choosing $j \in S$. In this way we obtain a one-to-one correspondence between subsets $U \subseteq S$ and products of n copies of a and b. The factor $a^i b^{n-i}$ results if and only if the corresponding subset U of S has cardinality i. The formula results as there are $\binom{n}{i}$ subsets of cardinality i.

1.5.6. *Prove*

$$(1-b)^n = \sum_{i=0}^{n} (-1)^i \binom{n}{i} b^i.$$

This follows from the preceding formula by substituting $a \mapsto -b, b \mapsto 1$.

1.6 The sphere packing bound

Exercises 1.6

1.6.1. *Construct a code* $(4,2,3)_2$ *(this is really trivial).*

We need only two codewords. Choose them as 0000 and 1111. This even is a $(4,2,4)_2$, obviously.

1.6.2. *Show that there is no* $(4,3,3)_2$*-code*

If there are two words at distance 4, we can choose them as 0000 and 1111. The third codeword cannot be found. So we can choose 0000 and 1110 as the first two words. The third word needs last entry 1, but the string of first three bits needs to have distance ≥ 2 from both 000 and 111, which is not possible.

1.6.3. *Construct a code* $(5,4,3)_2$ *(hint: use our code* $(6,8,3)_2$*).*

In our code $(6,8,3)$ there are 4 words ending in 0. Write out these words and omit the final 0. This is a $(5,4,3)_2$.

1.6.4. *Show that there is no* $(5,5,3)_2$.

Assume a $(5,5,3)_2$ exists. Either we can find 3 words ending in 0 or we can find 3 words ending in 1. Imagine these 3 words written out, with the last bit cancelled. This is a $(4,3,3)_2$. We know from Exercise 2.6.2 that it cannot exist.

1.6.5. *Using the preceding exercise, show that there is no* $(6,9,3)_2$.

Assume there is a $(6,9,3)_2$. Either there are 5 words ending in 0 or there are 5 words ending in 1. Erasing the last entry we obtain $(5,5,3)_2$, contradiction by the preceding exercise.

1.6.6. *What does the sphere-packing bound tell us about the length* n *of a binary code* $(n,2^7,5)_2$?

Assume $(n,2^7,5)_2$ exists. The sphere packing bound says

$$M = 2^7 \leq 2^n/(1+n+\binom{n}{2}),$$

equivalently

$$\binom{n}{2} + n + 1 \leq 2^{n-7}.$$

For $n = 13$ this is not satisfied ($13 + 78 \leq 64$ is not true), for $n = 14$ it is satisfied ($15 + 91 \leq 128$ is true). The answer is: $n \geq 14$.