Practicals, Week 1

1st academic week, 27 Sep 2024
Codes, Error-correction / -detection, Hamming distance

Exercises to be discussed in the practicals session

Problem P1 (How many codes are there?)

Suppose you were hired for a company that needs a very simple, but very large code (i.e., a code with many many many codewords) for accurate internal transmission of data. But management discarded previous codes and say they ran out of ideas and **claim** it is impossible to construct any new large codes. Prepare an argument, to be presented in the next meeting, that *it is always possible to construct new codes.*
(You can do so breaking things down according to the following steps.)

a) Recall to the team the definition of a code.

b) Given a natural number $q \in N$, consider the set $\{0, 1, \ldots, q-1\}$, which we also denote by $\mathbb{Z}/q\mathbb{Z}$. (In case q is **a prime number**, we denote $\{0, 1, \ldots, q-1\}$ by $\mathbb{F}_q$.) Use such sets to argue that there exist infinitely many possible alphabets for codes.

c) Given a finite set A with q elements, recall the definition of Cartesian products to determine the number of elements of the Cartesian product A^n.

d) Recalling (again) how codes are defined, explain that for any 'size' $x \in \mathbb{N}$ that your boss might come up with, one can always define (at least) one code that has *at least* x elements (i.e., codewords).

Problem P2 (Designing a code)

Now suppose the technical team of the company you work in needs a simple code as follows:

- The code has to encode single instructions in English (such as "skip", "compute", "print") whose length is at most 10;
- Due to the efficiency of the network of the company, it suffices that the code detects and corrects 1 error.

With the intuitive tip that 'repetition, repetition, repetition' is a way of making sure a spoken or written message comes across, design a code for your company that fulfils the requirements.
(Hint: you can use the following steps.)

a) First determine which alphabet should be used for the code.

b) Create a first attempted code by repeating each symbol once per transmission, and describe how many elements this code has.

c) Check whether the code from item (b) detects and corrects errors. If not, try again with repetition of symbols to design a code that eventually works.

Problem P3 (Chance of errors)

The shareholders of the company would like to understand why your code is reliable enough in practice. Knowing that the network uses a symmetric channel with symbol error probability $p = 0.000000001$ for its transmissions, explain to them that the chance of an arbitrary message being sent without errors is very high.

Problem P4 (Hamming distance)

Recall the notation we have set $\mathbb{F}_q = \{0, 1, \ldots, q-1\}$ when q is a prime number.

a) How many words in $\mathbb{F}_2^5$ have Hamming distance exactly two from the word 0 1 0 1 0? List all of them.

b) In a code of length 6 on the alphabet $\mathbb{F}_5$, what is the Hamming distance between 0 3 2 3 1 4 and 2 2 1 4 4 4?

Problem H1 (ISBN-10)

Recall that the ISBN-10 code is an 11-ary code of length 10 defined as

$$C = \{(x_1, \ldots, x_{10}) \in \mathbb{F}_{11}^{10} \mid x_{10} = \sum_{k=1}^{9} k x_k \text{ and } x_k < 10 \text{ for } k \leq 9\},$$

where the symbol '10' is (in practice) typically denoted by the capital letter X.

Remebering that calculations in $\mathbb{F}_{11}$ are performed 'modulo 11', solve the following problems:

a) The following two ISBN codewords have been received with smudges. What are the missing digits?

- 0 1 3 1 3 9 □ 3 9 9
- 0 0 2 3 2 9 9 □ 0 0

b) Give an example of a valid ISBN codeword where the last symbol is X.