Class 1 Exercises

CS250/EE387, Winter 2022

1. Let $\mathcal{C} \subset \{0,1\}^n$ be a code over the alphabet $\{0,1\}$ with block length n. What distance does $\mathcal{C}$ need to have to correct up to two errors?

   Come up with a code $\mathcal{C} \subset \{0,1\}^n$ that can correct two errors and that has $n \leq 10$. There is a straightforward construction of such a $n = 10$ and with message length $k = 2$. Once you find that construction, can you do better? (For example, can you find a code that can correct up to two errors, with $k = 2$ and $n < 10$?)

   Bonus. What's the best you can do? Can you prove that it's the best?

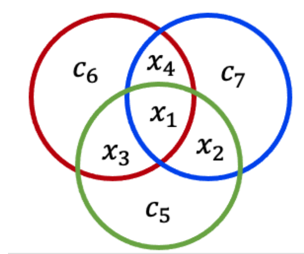
   Extra Bonus. What's the best you can do if $k = 3$? $k = 4$? Does your solution generalize?

   [Break for a bit of lecture before moving on]

2. In the lecture, we saw a binary code that had message length $k = 4$, codeword length $n = 7$, and distance $d = 3$. The encoding map was:

$$(x_1, x_2, x_3, x_4) \mapsto (x_1, x_2, x_3, x_4, x_1 + x_2 + x_3, x_1 + x_3 + x_4, x_1 + x_2 + x_4),$$

   (all mod 2), and we had this picture of circles:



   (a) We asserted that this code has distance 3. Convince yourself of this. (You don't need to give a formal proof, just stare at it until you are convinced and/or can convince each other).

   Hint. It suffices to show that there are no codewords with fewer than 3 ones. (Do you see why?)

   (b) It turns out that this is optimal – for example, there is no binary code with $k = 5, d = 3$ and $n = 7$. Prove this!

   Hint. Suppose that there were such a code. Consider the *Hamming balls* of radius 1 given by

$$B(c, 1) = \{x \in \{0,1\}^7 : \Delta(x, c) \leq 1\}$$

   for each $c \in C$. Do any of these Hamming balls overlap? How many points do they cover in total?

   (c) **(Bonus).** Generalize your logic on the previous problem to give an upper bound on k, in terms of n and d, if a code $C \subset \{0,1\}^n$ with message length k and distance d exists.

3. **(Bonus.)** How would you show formally that the distance of the Hamming code in the previous problem has distance 3. (There are several ways – try to find the most general way you can! What abstractions might be useful?)

4. **(More bonus).** The code in the previous problem suggests a general recipe for creating codes (with $k = 4$ and $n = 7$):

$$(x_1, x_2, x_3, x_4) \mapsto (x_1, x_2, x_3, x_4, f_5(\vec{x}), f_6(\vec{x}), f_7(\vec{x})),$$

   where f_5, f_6, f_7 are some linear functions mod 2. (That is, $f_i(x_1, x_2, x_3, x_4)$ is the sum of some of the message bits, mod 2.)

   What properties should f_5, f_6, f_7 have in order to make sure that the code we get has distance 3? How many possibilities are there?