Lecture 1: The Fundamental Questions and some bounds

Topics in Error-Correcting Codes (Fall 2022)
University of Toronto
Swastik Kopparty
Scribes: Anvith Thudi and Arkaprava Choudhury

1 Course outline

Instructor's contact: `swastik.kopparty@utoronto.ca`

Grading will be based on the following components:

1. Scribing lecture notes: each student scribes at least half a weekly lecture
2. Final project: an approximately 10-page write-up about a recent paper
3. 1 homework problem to be handed in before the end of the semester

2 Introduction to Error-Correcting Codes

In what follow we will use the following notation: Σ is a finite alphabet (may be any size: $\{1, 2, 3, \ldots, \log n, \sqrt{n}, n, \ldots, 2^n, \ldots, \infty\}$) and Σ^n denotes the set of strings of length n (n huge $\to \infty$) made from elements in Σ.

Over the space Σ we define the following metric:

Definition 1 (Hamming Distance)**.** $\forall x, y \in \Sigma^n$, $\Delta(x, y) :- |\{i \in [n] : x_i \neq y_i\}|$

One can trivially check the following fact which shows the Hamming distance is in fact a metric.

Fact 2. $\Delta(x, z) \leq \Delta(x, y) + \Delta(y, z)$ $\forall x, y, x \in \Sigma^n$

An error correcting code is then something that is spread out in this metric space, which is formally defined as follows:

Definition 3 (Error-Correcting Code (of distance d))**.** *An error correcting code is a subset $C \subset \Sigma^n$ s.t $\forall x, y \in C,\ x \neq y$ we have $\Delta(x, y) \geq d$. We say such subsets have "distance d".*

Given this setup and the definition of error-correcting codes, we can proceed to state the fundamental questions that motivate the study of error-correcting codes (i.e maximality and "construction") and for which we will give some preliminary results during the rest of this lecture.

1st Fundamental Question of Error Correcting Codes: What is the largest $|C|$ which has distance d given n, Σ.

2nd Fundamental Question of Error Correcting Codes: "Construct"[1] a set C reaching the largest $|C|$ that answers the first question.

3 Elementary upper and lower bounds

Consider the sets (or "balls") $B(x, r) \coloneqq \{z \in \Sigma^n : \Delta(x, y) \leq r\}$. We have the following fact:

Fact 4. *A code C has distance d for odd d iff $B(x, \frac{d-1}{2}) \cap B(y, \frac{d-1}{2}) = \emptyset$ for all $x, y \in C$.*

Proof. For the forward direction, suppose there is a point $z \in B(x, \frac{d-1}{2}) \cap B(y, \frac{d-1}{2})$. We would then have $d - 1 \geq \Delta(x, z) + \Delta(y, z) \geq \Delta(x, y) \geq d$ giving a contradiction.

For the backwards direction, for any $x, y \in C$ let $\Delta(x, y) = l$ and note there always $\exists z$ s.t $\Delta(x, z) = \lfloor l/2 \rfloor$ and $\Delta(y, z) = \lceil l/2 \rceil$. But then if $\Delta(x, y) \leq d$ we have z is in both balls, giving a contradiction. □

This fact gives an upper-bound on the size of any error correcting code, where we will let $B_n(r) = |B(x, r)|$.

Theorem 5 (Hamming Bound). *If $C \subset \Sigma^n$ has distance d, then $|C| \leq \frac{|\Sigma^n|}{B_n(\lfloor \frac{d-1}{2} \rfloor)}$*

A question now is if we could obtain lower-bounds that match the upper-bound (or come close), which would then answer our first fundamental question. In what follows we will specifically consider $\Sigma = 0, 1$ and $d = 1, 2, \cdots$ constant. Do note then that $B_n(r) = 1 + n + \binom{n}{2} + \cdots \binom{n}{r} \approx n^r$ for large n.

A first approach is to explicitly define a large error correcting code. For example, in the case of $d = 3$ we could consider $C = \{x \in \Sigma^n : x_{3i+1} = x_{3i+2} = x_{3i+3} \ \forall i\}$ which has size $2^{n/3}$. In the case of $d = 2$ we could take C as the set off all string with an even number of $1's$ (which has size $2^n/2$), and in $d = 1$ we just note Σ^n satisfies the condition (has size 2^n).

However, we could alternatively consider an algorithm that we know will produce an error correcting code, and give a lower bound for how large that final set is. Consider the following greedy approach:

1. Start with $C = \emptyset$
2. While $\exists x \in \Sigma^n \setminus C$ s.t $\Delta(x, y) \geq d \ \forall y \in C$, redefine $C = C \cup \{x\}$

Note that the while loop will keep running so long as $|C| B_n(d-1) \leq \Sigma^n$, i.e so long as $|C| \leq \frac{|\Sigma|^n}{B_n(d-1)}$ and this actually holds for all Σ. This then gives a lower bound on the size of the final state and hence a lower bound to our first fundamental question.

Theorem 6 (Gilbert-Varshamov Bound). *$\exists C$ with distance d s.t $|C| \geq \frac{|\Sigma|^n}{B_n(d-1)}$*

[1] what construction means will be defined later, but one can think of it as efficient computation of the set

d	1	2	3	...	$\theta(1)$
max $\lvert C\rvert \leq$	2^n	2^n	$2^n/(n+1)$	...	$\frac{\lvert\Sigma^n\rvert}{B_n(\lfloor\frac{d-1}{2}\rfloor}) \approx \theta(\frac{2^n}{n^{\lfloor\frac{d-1}{2}\rfloor}})$
max $\lvert C\rvert \geq$	2^n	$2^n/2$	$2^{n/3}$	...	$\frac{\lvert\Sigma^n\rvert}{B_n(\lfloor d-1\rfloor}) \approx \Omega(\frac{2^n}{n^{\lfloor d-1\rfloor}})$

Table 1: table of lower and upper bounds on the maximal size of error-correcting codes

Note that this does not match the upper-bound, and an open question is if we could somehow be more selective in our greedy approach and do better.

A summary of all our results are given in Table **??**

4 Hamming codes and tight bounds

As mentioned earlier, in the table above there exist gaps between the obtained upper and lower bounds when $d > 1$. However, at least for $d = 3$, we know that there are some error-correcting codes achieving better distance than the greedy code.

In fact, when $d = 3$, there exists a code C with $|C| = \frac{2^n}{n+1}$, thus achieving the upper bound obtained by the Hamming bound. These codes are called *Hamming codes*, and we shall require n to be of the form $n = 2^t - 1$ for these codes to be well-defined. However, we need not be concerned with this limitation as, after all, there are infinitely many such n possible and we are interested in the behaviour as n becomes large.

For this section, we will need to use some $\mathbb{F}_2$ linear algebra.

Construct a binary matrix $M \in \mathbb{F}_2^{t\times n}$ of size $t \times n$ such that the columns of M are all the non-zero binary strings in $\mathbb{F}_2^t$ (not necessarily in any particular order). Then, define C as

$$C = \{x \in \mathbb{F}_2^n : Mx = 0 \in \mathbb{F}_2^t\} \tag{1}$$

Theorem 7. *We claim the following two results.*

1. $|C| = \frac{2^n}{n+1}$
2. *C has distance 3*

Proof. Observe the following.

1. Note that C is a $\mathbb{F}_2$-linear subspace, and $|C| = 2^{\dim C}$. Also, note that C is the space of solutions to t linear homogeneous equations. Thus, $\dim C \geq n - t$, with equality being achieved if all the t equations are linearly independent (which, as a fact, they are not). Thus, putting these observations together, we get

$$|C| \geq 2^{n-t} = \frac{2^n}{2^t} = \frac{2^n}{n+1} \tag{2}$$

2. Now, choose $x \in C$ and $y \in C \backslash \{x\}$ arbitrarily. We want to show that x and y differ in at least 3 bits. Since $x, y \in C$, we already know that $Mx = My = 0$, and since these equations are linear, it follows that $M(x - y) = 0$.

   (a) Suppose, for contradiction, that $\Delta(x, y) = 1$. Then, $x - y$ has only one non-zero coordinate. Hence, it follows that $M(x - y)$ is simply some column of M. However, since $M(x - y) = 0$, this column would have to be 0 throughout, contradicting the definition of M.

   (b) Similarly, suppose that $\Delta(x, y) = 2$. Then, $x - y$ has only two non-zero coordinates. Thus, $M(x - y)$ can be expressed as the sum of two columns as $Me_i + Me_j$ for two one-bit strings $e_i, e_j \in \mathbb{F}_2^n$.

   However, since addition is the same as subtraction over $\mathbb{F}_2$, this would imply that these two columns in M are equal, which is also a contradiction.

   Thus, $\Delta(x, y) \geq 3$. Since x, y were chosen arbitrarily, by generalization, C has distance 3.

3. The inequality $|C| \leq \frac{2^n}{n+1}$ follows from the Hamming bound. This gives us the desired result that $|C| = \frac{2^n}{n+1}$.

□

Exercise 8. *Construct a Hamming code over $\mathbb{F}_q$ for a general finite field $\mathbb{F}_q$.*

Fact 9. *As a matter of fact, the upper bound of $\mathcal{O}\left(\frac{2^n}{n^{\left\lfloor\frac{d-1}{2}\right\rfloor}}\right)$ is tight when working over $\mathbb{F}_2$ and with $d \in \Theta(1)$.*

5 Larger d

Now, we consider some results over the same binary alphabet $\Sigma = \{0, 1\}$ but with $d \in \Theta(n)$ instead of being some small constant. In fact, let $d = \delta n$ for some fixed $\delta \in (0, 1)$.

Theorem 10 (Hamming bound)**.** *If C is a code with distance d, then*

$$|C| \leq \frac{2^n}{B(\delta n/2)} \tag{3}$$

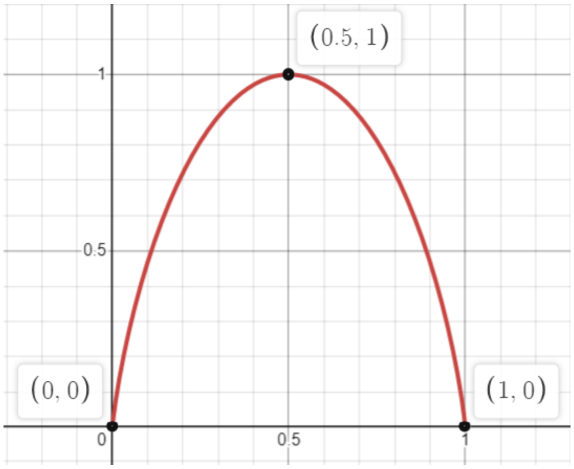
Fact 11. *If $0 < \delta < 1/2$, then we have*

$$\begin{aligned} B(\delta n/2) &= \binom{n}{0} + \binom{n}{1} + \ldots + \binom{n}{\lfloor \delta n/2 \rfloor} \qquad (4) \\ &= 2^{(H(\delta/2)+o(1))n} \qquad (5) \end{aligned}$$

where $H(\cdot)$ denotes the binary entropy function defined below.

Definition 12 (Binary entropy function)**.** *Define $H(x) = x\log(1/x) + (1 - x)\log\left(\frac{1}{1-x}\right)$ for $x \in [0, 1]$.*

The following diagram shows a plot of H over the interval $[0,1]$, drawn using Desmos.



Again, like the case for constant d, we also have a lower bound on the distance for codes, as follows.

Theorem 13 (Gilbert-Varshamov bound)**.** *There exists a code with distance δn such that*

$$|C| \geq \frac{2^n}{B(\delta n - 1)} \geq 2^{n(1-H(\delta)+o(1))} \tag{6}$$

6 Motivation and further definitions

We now look at some more motivation for why we study error-correcting codes in the first place, and in particular, why Hamming began studying error-correcting codes.

The original motivating example which necessitated the use of error-correction was in sending messages over wire, back in the 1940s, so that a corruption in one or more bits of the message did not prevent the receiver from retrieving the contents of the message. These messages could be thought of as arbitrary strings in $\{0,1\}^k$.

The error-correcting procedure was then thought of as *encoding* the original k-bit binary string into an n-bit binary string reversibly. This encoded message was then passed along to the intended receiver, who would then *decode* it, thus retrieving the original k-bit message.

This leads us to the following third fundamental question we consider when looking at error-correcting codes.

3rd Fundamental Question of Error Correcting Codes: Give an optimal code C with an efficient decoding algorithm.

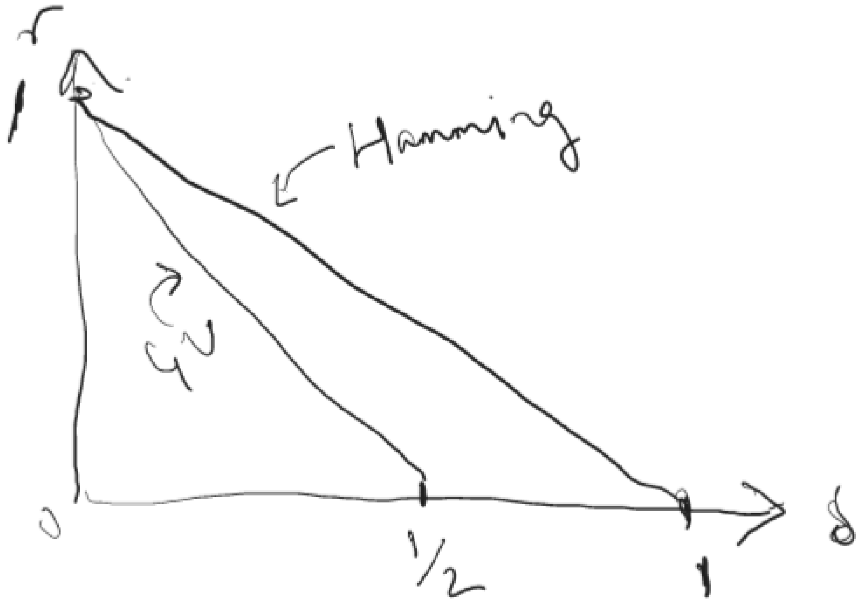
More formally, the error-correcting procedure of distance d would consist in the following two maps:

1. An encoding map $\text{Enc} : \{0,1\}^k \to \{0,1\}^n$, which would be injective onto $C \subseteq \{0,1\}^n$
2. A decoding map $\text{Dec} : \{0,1\}^n \to \{0,1\}^k$ such that $\text{Dec} \circ \text{Enc}$ would serve as the identity, as long as at most $\frac{d-1}{2}$ errors had occurred.

Definition 14 (Rate)**.** *In the framework described above, the rate of the code C would be defined as $r = k/n$, which can be thought of as how much you are using the channel.*

Observation 15. *The rate of any $C \subseteq \{0,1\}^n$ is simply $r = \frac{\lceil \log |C| \rceil}{n}$.*

Note that there is an interesting tradeoff between the rate of a code versus the value of δ, as given by the Hamming and the Gilbert-Varshamov bounds. The following diagram gives a (very) rough sketch of this tradeoff.


r
Hamming
GV
1/2
δ