FEB 10

1 MIN CHECKIN

REMINDERS

(•) HW 0 is out. Due 11:59pm Wed, Feb 15
  → ONLY submit Q1
  → Optional
  → Please read the instructions/hints carefully

(•) General Comment: Any proof technique is fine (as long as proof is correct of course!)

(•) Team registation due 11:59pm Wed, Feb 15 ← Missing this deadline ⇒ you lose 50% of your grade!

RECAP

(•) Big Q: Given a distance d, what is the largest rate of a code w/ dist d.?

(•) Simpler Q: Above with $d=3$

(•) Hamming code (C_H)

$$(x_1,x_2,x_3,x_4) \mapsto (x_1,x_2,x_3,x_4)\begin{pmatrix} 1&0&0&0&0&1&1\\ 0&1&0&0&1&0&1\\ 0&0&1&0&1&1&0\\ 0&0&0&1&1&1&1 \end{pmatrix}$$

← addition is XOR

$\boxed{n=7,\ k=4,\ d=3}$

PLAN for TODAY: (1) Hamming bound
(2) Linear codes

Hamming code: First non-trivial upper bound on R. (or k)

Trivial: $k \le n$ ($\equiv R \le 1$)

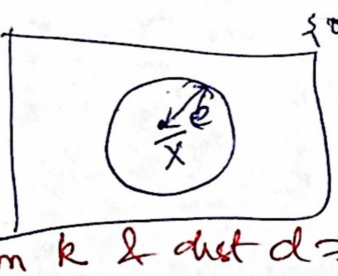
THM: For any binary code $C \subseteq \{0,1\}^n$ with $\dim(C)=k$ and $d=3$ $\Rightarrow$ $k \le n - \log_2(n+1)$

Ex: $n=7, d=3 \Rightarrow k \le 7-\log_2(7+1) = 7-\log_2 8 = 7-3=4.$

$\Rightarrow k \le 4 \Rightarrow$ C_H is the best possible for $n=7, d=3$.

Def: (Hamming ball) $\overline{x} \in \{0,1\}^n$ ← Σ^n

$B(\overline{x}, e) = \{ y \in \{0,1\}^n \mid \Delta(\overline{x}, \overline{y}) \le e \}$ ← Σ^n



Let C be a code of len n & dim k & dist $d=3$

Pf of THM: **Lemma 1:** Hamming balls of radius 1 ~~around~~ around codewords of C_H have to be disjoint

⇒ Limits how many Hamming balls can be packed in $\{0,1\}^n$

⇒ Limit how many centers I can have

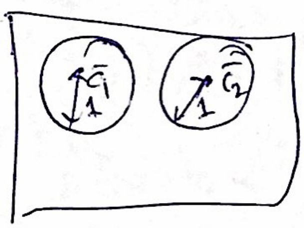
⇒ Limits $|C|$

⇒ $|C| \le q^{blah}$ ⇒ $\dim(C) \le \log_q(blah)$

Lemma 1: $\forall\ \overline{c_1} \ne \overline{c_2} \in C \qquad B(\overline{c_1}, 1) \cap B(\overline{c_2}, 1) = \phi$

Lemma 2: $\forall\ x \in \{0,1\}^n, \ |B(\overline{x}, 1)| = n+1$

(Punt on the proofs for a bit)



Note:

$$\bigcup_{\overline{c} \in C} B(\overline{c}, 1) \subseteq \{0,1\}^n$$

$$\Rightarrow \left| \bigcup_{\overline{c} \in C} B(\overline{c}, 1) \right| \le 2^n \quad \text{— (1)}$$

OTOH

$$\left| \bigcup_{\overline{c} \in C} B(\overline{c}, 1) \right| \overset{\text{Lem 1}}{=} \sum_{\overline{c} \in C} |B(\overline{c}, 1)| \overset{\text{Lem 2}}{=} \sum_{\overline{c} \in C} (n+1)$$

$$= (n+1)|C| \quad \text{— (2)}$$

$$(1) + (2) \Rightarrow (n+1) \cdot |C| \le 2^n$$

$$\Rightarrow |C| \le \frac{2^n}{n+1} \Rightarrow 2^k \le \frac{2^n}{n+1}$$

$$\Rightarrow k \le \log_2 \frac{2^n}{n+1} = \log_2 2^n - \log_2(n+1)$$

$$= n - \log_2(n+1) \ \blacksquare$$

$\boxed{k \le n - \log_2(n+1)}$

Pf: By contradiction. Assume $\exists\ \bar{c}_1 \neq \bar{c}_2 \in C$
but $B(\bar{c}_1, 1) \cap B(\bar{c}_2, 1) \neq \phi$

$\exists\ \bar{y} \in B(\bar{c}_1, 1) \cap B(\bar{c}_2, 1)$

$\Rightarrow \Delta(\bar{c}_1, \bar{y}) \leq 1 \quad \Delta(\bar{c}_2, \bar{y}) \leq 1$

$$\Delta(\bar{c}_1, \bar{c}_2) \leq \Delta(\bar{c}_1, \bar{y}) + \Delta(\bar{c}_2, \bar{y})$$
$$\leq 1 + 1 = 2 \Rightarrow \text{contradict } d(C) = 3 \ \blacksquare$$

Def: Code of block length n, dim k, dist d over alphabet Σ / alphabet size q will be denoted by $(n, k, d)_\Sigma$ / $(n, k, d)_q$

E.g.: C_H is $(7, 4, 3)_2$ code
$C_{3,rep}$ is $(12, 4, 3)_2$ —

Generalized Hamming bound: for any $(n, k, d)_q$ code,

$$k \leq n - \log_q \left[\sum_{i=0}^{\lfloor \frac{d-1}{2} \rfloor} \binom{n}{i} (q-1)^i \right]$$

$\Rightarrow q = 2, d = 3$
$$k \leq n - \log_2 \left[\binom{n}{0}(2-1)^0 + \binom{n}{1}(2-1)^1 \right]$$
$$= n - \log_2(1+n) \ \blacksquare$$

Reading Assignment: Sec 1.7

$\boxed{\text{Linear Codes}}$

Consider an arbitrary $(n, k, d)_q$ code / $(n, k)_q$
(Σ)

$\rightarrow$ list all q^k codewords

$\Rightarrow n \cdot q^k$ symbols over Σ
or $n \cdot q^k \cdot \lceil \log_2 q \rceil$ bits

Linear codes can be represented with $O(n^2)$ symbols.
Any $(n,k)_q$ or $O(n^2 \log q)$ bits

Def (Linear code) Let $q = p^s$ $p \Rightarrow$ prime
$s \geq 1$ is an int.

$C \subseteq$ "$\{0,\ldots,q-1\}^n$" is a linear code $\Longleftrightarrow$ C is a linear subspace

on Mon

First: "$\{0,\ldots,q-1\}$" ?

Want set of q numbers that we can add/subtract/mult./divide

Fields $\mathbb{F} = (S, +, \cdot) \Rightarrow$ add/subtract/mult/divide & remain inside S