Today

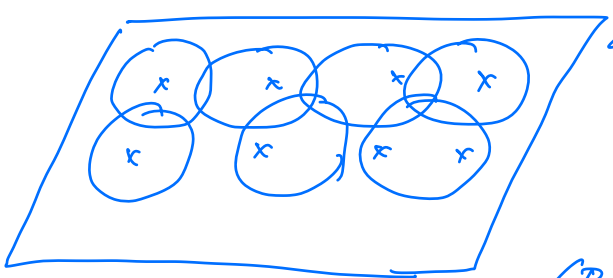
- Recap Elias-Bassalygo Bound
- Johnson Radius
- Reed-Muller Code.

CSS.318.1
Coding Theory
Lecture 8 (2022-9-21)
Instructor: Prahladh Harsha.

Recall from a few lectures ago

Elias-Bassalygo Bound:
$\forall \delta \in (0, 1/2)$ if C is $[n, Rn, \delta n]_2$-code then
$R + h_2(J_2(\delta)) \leq 1 + o(1)$.
where $J_2(\delta) = \frac{1}{2}(1 - \sqrt{1-2\delta})$

(Strengthening of the Hamming / Packing Bd.



$\{0,1\}^n$

$\forall x,$
$|Ball(x, \frac{\delta n}{2}) \cap C| \leq 1$

$Ball(x, r) = \{y \in \{0,1\}^n \mid \Delta(x,y) < r\}$

Suppose (r, L) satisfied the following

$\forall x \quad |Ball(x, rn) \cap C| \leq L \quad \ldots (*)$

$(*) \Rightarrow$ Balls of radius rn around codewords have a max overlap of L at any pt.

$$\sum_{c \in C} |Ball(c, \tau n)| \le L \cdot 2^n$$

$$2^k \cdot 2^{H_2(\tau) n} \le 2^{n + \log_2 L}$$

ie, $R + H_2(\tau) \le 1 + \frac{\log_2 L}{n}$ | As long as $L = 2^{o(n)}$ it hardly affects the bound

---

List-decoding: Given x, find all the codewords in $Ball(x, \tau n)$.

Johnson Radius: $J_2(\delta) := \frac{1}{2}\left(1 - \sqrt{1 - 2\delta}\right) > \delta/2$

Given any $\delta \in (0, 1/2)$, a $C \subseteq \{0,1\}^n$ code w/ distance δ, then for $\tau = J_2(\delta)$

$\forall x \in \{0,1\}^n$, $|Ball(x, \tau n) \cap C| \le n + 1$.

Proof: Similar to that of Plotkin Bd (using the Geometric Lemma).

Suppose $\exists\ x \in \{0,1\}^n$ & $c_1 \ldots c_m \in C$ s.t. $\Delta(c_i, x) < \tau n$. $\forall i$

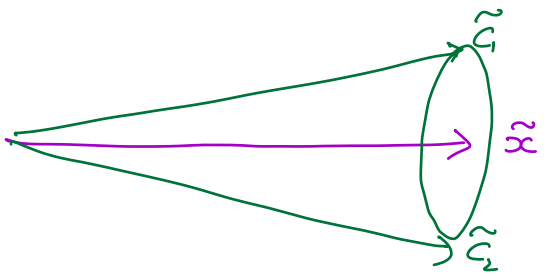
Hamming $\longmapsto$ Eucl.

b $\longmapsto$ $(-1)^b$

$x, c_1 \ldots c_m \longmapsto \tilde{x}, \tilde{c}_1 \ldots \tilde{c}_m \in \{\pm 1\}^n$

What do we know

$$\langle \tilde{c}_i, \tilde{c}_j \rangle \le n(1-2\delta) \quad , \forall i \neq j$$
$$\langle \tilde{c}_i, \tilde{c}_i \rangle = n$$
$$\langle \tilde{c}_i, \tilde{x} \rangle > n(1-2\tau) \quad , \forall i$$



Can we find an $\alpha \in \mathbb{R}$

$$v_i = \tilde{c}_i - \alpha \tilde{x}$$

s.t

$$\forall\, i \neq j \quad \langle v_i, v_j \rangle < 0$$

then can conclude

$$m \le n+1$$

$i \neq j$

$$\langle v_i, v_j \rangle = \langle \tilde{c}_i - \alpha\tilde{x}, \tilde{c}_j - \alpha\tilde{x} \rangle$$
$$= \langle \tilde{c}_i, \tilde{c}_j \rangle - \alpha(\langle \tilde{x}, \tilde{c}_i \rangle + \langle \tilde{x}, \tilde{c}_j \rangle) + \alpha^2 \langle \tilde{x}, \tilde{x} \rangle$$
$$< n[(1-2\delta) - 2\alpha(1-2\tau) + \alpha^2]$$
$$= n[(\alpha - (1-2\tau))^2 + (1-2\delta) - (1-2\tau)^2]$$

Set $\alpha = 1-2\tau$ $\;\Rightarrow\;$ τ s.t $(1-2\delta) < (1-2\tau)^2$

i.e $\tau < \frac{1}{2}(1 - \sqrt{1-2\delta})$ ☒

Reed-Muller Code:

RM: polynomial evaluation

message $\longrightarrow$ codeword

coeffs of univariate poly $\longrightarrow$ eval of poly on some set

Reed-Muller: Generalization to multivariate polynomials.

m-variate $X_1, X_2 \ldots, X_m$; $\mathbb{F} = \mathbb{F}_q$ $(q = p^B)$

$\mathbb{F}[X_1, X_2 \ldots X_m]$ - ring of polynomials

$\mathbb{F}_{\le r}[X_1, X_2 \ldots X_m]$ - vector space of m-variate poly of total degree $\le r$.

$$p(X_1 \ldots X_m) = \sum_{\substack{e_1 \ldots e_m \\ \sum e_i \le r \\ 0 \le e_i \le q-1}} P_{e_1 \ldots e_m} X_1^{e_1} X_2^{e_2} \ldots X_m^{e_m}$$

$(r < m(q-1))$

Evaluation Set $= \mathbb{F}_q^m$ (the entire space) (most of the time)

$$RM_q(m, r) = \left\{ Eval(p(X_1 \ldots X_m))\Big|_{\mathbb{F}_q^m} \;\middle/\; p \in \mathbb{F}_{\le r}[X_1 \ldots X_m] \right\}$$

$\mathbb{F}_q$-linear

dimension $K_q(m, r) = \left| \left\{ (e_1 \ldots e_m) \in \mathbb{Z}_{\ge 0}^m \;/\; 0 \le e_i \le q-1, \forall i, \; \sum e_i \le r \right\} \right|$

Two Settings: (1) Low-degree $r < q$

(2) Binary Setting $q = 2$.

(1) Low-degree setting ($d < q$)

$\sum e_i \le d$ implies $0 \le e_i \le q-1$

$$K_q(m,d) = \binom{m+d}{d}$$

Distance of code:

Lemma (SZ). $p \in \mathbb{F}_q[x_1 \dots x_m]$; $d < q$. $p \neq 0$

$$\Pr_{a \leftarrow \mathbb{F}_q^m}[p(a) \neq 0] \ge 1 - \frac{d}{q}$$

(Generalization of degree mantra in univariate case)

Proof does not require eval set to be $\mathbb{F}_q^m$

Any product set S^m will suffice

$$1 - \frac{d}{|S|}$$

(2) Binary setting : $q = 2$

$$K_2(m,d) = |\{(e_1 \dots e_m) \in \mathbb{Z}_{\ge 0} \mid e_i \in \{0,1\}, \sum e_i \le d\}|$$

$$= \sum_{i \ge 0}^{d} \binom{m}{i} = \binom{m}{\le d}$$

Lemma (SZ). $p \in \mathbb{F}_2[x_1 \dots x_m]$; $\mathbb{F} = \mathbb{F}_2$, $p \neq 0$

$$\Pr_{a \leftarrow \mathbb{F}_2^m}[p(a) \neq 0] \ge \frac{1}{2^d}$$ (tight case $\prod_{i=1}^{d} x_i$)

Common Generalization to the above 2 SZ Lemmas

Lemma (SZ) $p \in \mathbb{F}_q[x_1 .. x_m]$; $p \neq 0$

$n = a(q-1)+b$ where $0 \le b < q-1$

$$\Pr_{a \in \mathbb{F}_q^m}[p(a) \neq 0] \ge \frac{1}{q^a}\left(1 - \frac{b}{q}\right)$$

$\mathbb{F}_q = \{\alpha_1 \dots \alpha_q\}$

Right: $p(x_1 .. x_m) = \prod_{j=1}^{b}(x_{a+1} - \alpha_j)\left(\prod_{i=1}^{a}\prod_{\alpha \neq 0}(x_i - \alpha)\right)$

$$\Pr[p(a) \neq 0] = \frac{1}{q^a}\left(1 - \frac{b}{q}\right)$$

Dual of RM code:

We know $\sum_{\alpha \in \mathbb{F}_q} \alpha^i = \begin{cases} 0 & \text{if } 0 \le i < q-1 \\ -1 & \text{if } i = q-1 \end{cases}$

$$\sum_{(\alpha_1 .. \alpha_m) \in \mathbb{F}_q^m} \alpha_1^{e_1} \alpha_2^{e_2} .. \alpha_m^{e_m} = \prod_{i=1}^{m}\left(\sum_{\alpha_i \in \mathbb{F}_q} \alpha_i^{e_i}\right)$$

$= 0$ if $(e_1 \dots e_m) \neq (q-1, q-1 \dots q-1)$

If $x_1^{e_1} \cdot x_m^{e_m}$ and $g = x_1^{f_1} .. x_m^{f_m}$

$\sum e_i + \sum f_i < m(q-1)$

$$\sum_{\bar{\alpha} \in \mathbb{F}_q^m} f(\bar{\alpha}) \cdot g(\bar{\alpha}) = 0$$

$$RM_q(m,r)^{\perp} \supseteq RM_q(m, m(q-1)-r-1)$$

Matching dimensions

$$RM_q(m,r)^{\perp} = RM_q(m, m(q-1)-r-1).$$

Special Cases of $RM_q(m,r)$

① $q=2$, $r=1$, m - arbitrary

$RM_2(m,1)$ $\quad [2^m, m+1, 2^{m-1}]_2$-code.

Hadamard Code

$[2^m, m, 2^{m-1}]_2$-code

$$H_{2^m} = \begin{bmatrix} \vdots & \vdots \\ \xi^i & \tilde{C}_{2^m} \\ \vdots & \vdots \end{bmatrix}$$

$$H \cdot H^T = I \cdot 2^m$$

Drop Eval at $\bar{0}^m$ $\quad [2^m-1, m+1, 2^{m-1}]_2$-code.

(Simplex code)

What is the dual of $RM_2(m,1)$

$= RM_2(m, m-2)$

Drop the eval at origin $\leftarrow [2^m, 2^m-m-1, 4]_2$-code.

Hamming Code. $[2^m-1, 2^m-m-1, 3]_2$-code