ERROR-CORRECTING CODE

$C \subseteq \Sigma^n$ $\quad \Sigma$ alphabet

$(n, k, d)_q$ - code

- block length n
- message length k $\quad (= \log_q |C|)$
- minimal distance $d = \Delta(C)$
- alphabet size $q = |\Sigma|$

Linear code: $\Sigma = \mathbb{F}_q$

$C \subseteq \mathbb{F}_q^n$ linear subspace

Notation $[n, k, d]_q$

SINGLETON BOUND: Any $(n, k, d)_q$ - code has $d \leq n - k + 1$

Codes meeting this bound are maximum distance separable codes (MDS codes)

REED-SOLOMON CODE $\quad RS_{q,n,k}$

q = alphabet size $= |\mathbb{F}_q|$

$n \leq q$ $\quad$ block length

$k \leq n$ $\quad$ message length

- Pick n distinct elements $\alpha_1, \alpha_2, \ldots, \alpha_n \in \mathbb{F}_q$
- Identify message $(c_0, c_1, \ldots, c_{k-1}) \in \mathbb{F}_q^k$ with $P_{\vec{c}} = \sum_{j=0}^{k-1} c_j x^j$
- Codeword is $(P_{\vec{c}}(\alpha_1), P_{\vec{c}}(\alpha_2), \ldots, P_{\vec{c}}(\alpha_n))$

$RS_{q,n,k}$ is an $[n, k, n-k+1]_q$ - code $\quad$ (MDS)

REED-MULLER CODES $RM_{m,\ell,q}$

q alphabet size $= |\mathbb{F}_q|$

ℓ $\geq$ total degree of polynomials

m-variate polynomials (over $x_1, x_2, \ldots, x_m$)

Case 1 $\ell < q$

Message $M(\vec{x}) = \sum_{i_1+\ldots+i_m \leq \ell} m_{i_1,\ldots,i_m} x_1^{i_1} \ldots x_m^{i_m}$

Codeword $(M(\alpha))_{\alpha \in \mathbb{F}_q^m}$

Block length $n = q^m$

Message length $\binom{m+\ell}{m}$

Distance $\left(1 - \frac{\ell}{q}\right) n$

Follows from

SCHWARTZ-ZIPPEL LEMMA

Non-zero $f \in \mathbb{F}_q[x_1, \ldots, x_m]$, $\deg(f) \leq d$.

$S \subseteq \mathbb{F}_q$. f is non-zero on at least

$1 - \frac{d}{|S|}$ fraction of points in $S^m \subseteq \mathbb{F}_q^m$

Case 2 $\ell \geq q$

Messages polynomials of total degree ℓ

individual degree $\leq q-1$

More complicated definitions and bounds

See notes from Lecture 9.

- Today will talk about what it means that a code is "good"
- Will see target parameters to shoot for in explicit constructions
- Talk a little bit about how to compose codes to get new code
- Finally discuss algorithmic challenge of efficient decoding (probably next 2 lectures)

But first... one more code you should know about (turns up in lots of different contexts)

HADAMARD CODES

Obtained from self-orthogonal matrices over $\{\pm 1\}$

<u>DEF 1</u> An $n \times n$ matrix $H = \{h_{ij}\}$ is a HADAMARD MATRIX if $h_{ij} \in \{+1, -1\}$ $\forall i,j$ and $HH^T = n \cdot I$ (in regular integer arithmetic).

Can view rows of Hadamard matrix as binary code of blocklength n with n codewords (i.e., message length $\log n$)

Binary alphabet $\{0,1\} \longleftrightarrow \{+1,-1\}$

Distance? $HH^T = nI$ means that for $i \neq j$ $\sum_{k=1}^{n} h_{ik} h_{jk} = 0$

I.e. ith row and jth row equal in exactly half of pos
distinct in exactly half of pos

So distance <u>exactly</u> $n/2$

Can make code twice as large while keeping distance by adding all complements

<u>DEF 2</u> Given $n \times n$ Hadamard matrix H, the HADAMARD CODE of block length n, Had_n, is the binary code whose codewords are the rows of H (with $\{+1,-1\}$ replaced by $\{0,1\}$) and the complements of the rows of H.

<u>PROP 3</u> For every $n \times n$ Hadamard matrix exists, the Hadamard code Had_n is an $(n, \log(2n), n/2)_2$ - code.

These parameters look somewhat similar to something achieved last time...
Namely, take Reed-Muller code $RM_{m,1,2}$
- Total degree 1 (linear/affine functions)
- Over $\mathbb{F}_2$
- gives a $[2^m, m+1, 2^{m-1}]_2$ code (same parameters with $m = \log n$)

Is this a Hadamard code?
i.e., is there an underlying Hadamard matrix?
Yes!
The messages are coefficients $(c_0, c_1, \ldots, c_m)$ representing $RM_{\vec{c}}(\vec{x}) = c_0 + \sum_{i=1}^{m} c_i x_i$

Look at codewords for $c_0 = 0$.
Will differ in <u>exactly</u> half the places $\Leftrightarrow$ rows of Hadamard matrix

Can view Hadamard code as all linear functions on $(x_1, x_2, \ldots, x_m)$
[complement $\Leftrightarrow$ affine functions]

Usual construction of Hadamard matrices for $n = 2^m$ is inductive

$$H_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad H_{m+1} = \begin{bmatrix} H_m & H_m \\ H_m & -H_m \end{bmatrix}$$

Summing up what codes we've seen so far

HAMMING CODES

Good relationship between message and block lengths
Packs codewords perfectly. Binary alphabet (and non-binary)
But only distance 3

REED-SOLOMON CODES

Optimal distance / message length behaviour
But require large alphabets
(Reed Muller decreases alphabet size a bit)

HADAMARD CODES

Binary alphabet
Great distance!
But very poor relationship between message length and block length

Would like to find codes with
- constant ratio between message length & block length
- constant ratio between distance & block length

Need to study <u>asymptotics</u> of codes
Consider infinite families of codes

$$\mathcal{C} = \{ (n_i, k_i, d_i)_{q_i} \}_{i=1}^{\infty}$$

with $\lim_{i \to \infty} \{ n_i \} = \infty$
(and we will want to think of q_i as fixed, usually).

<u>DEF 4</u> (MESSAGE) RATE $R(\mathcal{C}) = \liminf_{n \to \infty} \left\{ \frac{k_i}{n_i} \right\}$

RELATIVE DISTANCE $\delta(\mathcal{C}) = \liminf_{n \to \infty} \left\{ \frac{d_i}{n_i} \right\}$

<u>DEF 5</u> A family of codes is ASYMPTOTICALLY GOOD if $R(\mathcal{C}) > 0$ and $\delta(\mathcal{C}) > 0$.

Do asymptotically good codes exist? Yes.
Can they be constructed explicitly?
Yes, and this was in fact achieved early on.
(Hope to see one such construction before wrapping up our coding theory excursion)

Results in coding theory tend to have asymptotic versions / interpretations.
Not seldom, these versions are more succinct.

Singleton bound $\quad d \le n-k+1$

SINGLETON BOUND ASYMPTOTIC VERSION $\quad \boxed{\delta \le 1 - R}$

Hamming bound (for binary codes)

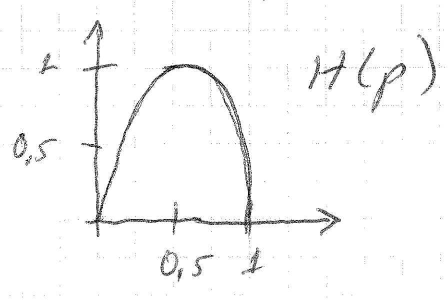
$$2^k \, Vol_2\left(\left\lfloor \frac{d-1}{2} \right\rfloor, n\right) \le 2^n$$

$$\left\lfloor \frac{d-1}{2} \right\rfloor \approx \delta n/2$$

$$Vol_2(pn, n) \approx 2^{H(p)\cdot n}$$

for $H(p)$ being the BINARY ENTROPY FUNCTION

$$H(p) = -p \log_2 p - (1-p) \log_2 (1-p)$$
$$= p \log(1/p) + (1-p) \log(1/(1-p))$$



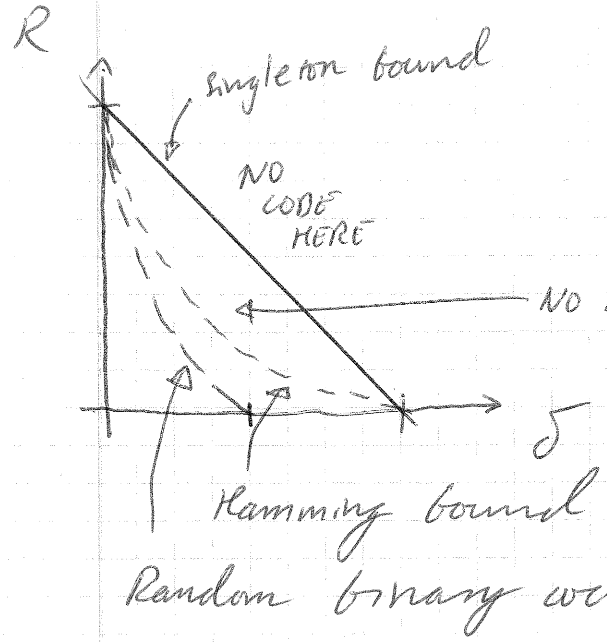
HAMMING BOUND, ASYMPTOTIC VERSION $\quad$ For $q=2$ $\quad \boxed{R + H(\delta/2) \le 1}$

Random binary codes satisfy

$$R \ge 1 - H(\delta) \qquad (*)$$

(Doesn't meet Hamming bound)

There are <u>no explicit constructions</u> known achieving $(*)$



sketch of bounds on previous page

GILBERT - VARSHAMOV BOUND

Gilbert '52 : Random (→ or greedy construction; kind of the same) code satisfies (*)

Varshamov '57 : Random linear code satisfies (*)

Gilbert:

GREEDY (n, d)

$S := \{0,1\}^n$

$C := \emptyset$

while $S \neq \emptyset$

Pick $x \in S$

$C := C \cup \{x\}$

$S := S \setminus B(x,d)$ ← ball of radius d around x

PROP 6 Fix $\delta \in (0, 1/2)$ and $\varepsilon > 0$ and let $R = 1 - H(\delta) - \varepsilon$. Then for all sufficiently large n GREEDY $(n, \lceil \delta n \rceil)$ produces a code with at least 2^{Rn} codewords

Need that

$$\mathrm{Vol}_2(pn, n) = 2^{(H(p)+o(1))n}$$

Proof of Prop 6: Pick n large enough so that $\mathrm{Vol}_2(d,n) \leq 2^{(H(\delta)+\varepsilon)n}$.

Assume the algorithm picks K codewords. At every step, at most $\mathrm{Vol}_2(d,n)$ elements removed from S. Hence

$$K \geq \frac{2^n}{\mathrm{Vol}(d,n)} \geq 2^{(1-H(\delta)-\varepsilon)n} = 2^{Rn}$$

$\blacksquare$

<u>Varshamov</u>:

RANDOM-LINEAR (n, k)

Pick entries of $G \in \mathbb{F}_2^{k\times n}$ uniformly and independently at random

Let $C = \{ yG \mid y \in \mathbb{F}_2^k \}$

<u>PROP 7</u> Fix $\delta \in (0, 1/2)$ and $\varepsilon > 0$ and let $R = 1 - H(\delta) - \varepsilon$, then for all sufficiently large n and $k = \lceil Rn \rceil$ the procedure RANDOM-LINEAR (n,k) produces a code with 2^k codewords and distance at least δn asymptotically almost surely

<u>Proof</u> Need to prove two things

1. G has full ~~column~~ row rank k so that 2^k codewords
2. No codewords are closer than distance δn

Combine two claims into one: $\boxed{X}$

For every $y \neq 0$ $yG \notin B(0, \delta n)$

From this:

① follows since $yG \neq 0$ for $y \neq 0$ so all rows linearly independent

② ~~fo~~ follows since min distance of linear code = min weight of nonzero code word.

Pick n large enough so that

$$Vol_2(\delta n, n) \leq 2^{(H(\delta)+\varepsilon/2)n}$$

Let $d = \lfloor \delta n \rfloor$

For any fixed $y \neq 0$, yG is random vector in $\mathbb{F}_2^n = \{0,1\}^n$

Hence

$$\Pr[wt(yG) \leq d] = \Pr[yG \in B(0,d)] = \frac{Vol_2(d,n)}{2^n} \leq 2^{(H(\delta)+\varepsilon/2-1)n}$$

Take union bound over all $y \in \mathbb{F}_2^k \setminus \{0\}$

$$\Pr[\exists y \neq 0 \ \ wt(yG) \leq d] \leq 2^k 2^{(H(\delta)+\varepsilon/2-1)n} \qquad (+)$$

If $R = k/n = 1 - H(\delta) - \varepsilon$, then $(+)$ becomes

$$\leq 2^{-\varepsilon/2 \cdot n} \to 0 \quad \text{as} \quad n \to \infty$$

and asymptotically almost surely C has min distance $\geq \delta n$. $\boxed{\ }$

How can one build asymptotically good codes? By combining two good codes to get an even better bound

Outline from a few lectures ago
1. Build great code with too large alphabet
2. Shrink alphabet size while keeping overall goodness.

We saw a very simple example of this
1. Build Reed-Solomon code
2. Get down to binary alphabet by encoding binary strings as field elements

This increased block length but did <u>not</u> improve distance. Can we do something smarter?
Yes

Have code over large alphabet/field $\mathbb{F}_{q^k}$
Want to get down to alphabet $\mathbb{F}_q$

<u>FACT</u>: $\mathbb{F}_{q^k}$ can be viewed as vector space $\mathbb{F}_q^k$

There are linear bijections $\mathbb{F}_{q^k} \leftrightarrow \mathbb{F}_q^k$

Will use this heavily (but implicitly)

CONCATENATION (somewhat sloppy definition, but captures the essentials)

Have two codes

C_1 $[n_1, k_1, d_1]_{q^{k_2}}$ outer code, large alphabet

C_2 $[n_2, k_2, d_2]_q$ inner code, small alphabet

CONCATENATION $C_1 \diamond C_2$ is $[n_1 n_2, k_1 k_2, d_1 d_2]_q$-code as defined next, using encoding functions

$$E_1 : \mathbb{F}_{q^{k_2}}^{k_1} \to \mathbb{F}_{q^{k_2}}^{n_1}$$

$$E_2 : \mathbb{F}_q^{k_2} \to \mathbb{F}_q^{n_2}$$

① Take input α in $\mathbb{F}_q^{k_1 k_2}$ (message)

② View as element α' in $\left(\mathbb{F}_{q^{k_2}}\right)^{k_1}$ by linear bijection

③ Encode by E_1 to outer codeword β in $\left(\mathbb{F}_{q^{k_2}}\right)^{n_1}$

④ Interpret as n_1 messages $\beta' \left(\mathbb{F}_q^{k_2}\right)^{n_1}$ by linear bijection again

⑤ Encode coordinatewise using E_2 to get element γ in $\left(\mathbb{F}_q^{n_2}\right)^{n_1}$

⑥ The final codeword is this element γ' in $\left(\mathbb{F}_q^{n_2}\right)^{n_1} = \mathbb{F}_q^{n_1 n_2}$

PROP 8 $C_1 \diamond C_2$ is an [as described above] $[n_1 n_2, k_1 k_2, d_1 d_2]_q$ - code

Proof sketch:

Things to check

- The code $C_1 \diamond C_2$ depends only on C_1 & C_2 and not on E_1, E_2, or implicit linear bijections. We will completely ignore this.
- Message length and block length — follows from construction.
- Linearity — follows since all operations are linear.
- Distance: Need to show that non-zero codeword has weight $\geq d_1 d_2$

  Suppose $\alpha \in \mathbb{F}_q^{k_1 k_2}$, $\alpha \neq 0$

  Then $\alpha' \in (\mathbb{F}_{q^{k_2}})^{k_1}$ is also $\neq 0$

  C_1 has distance d_1, so β is nonzero in d_1 coordinates

  $\Rightarrow \beta'$ has d_1 nonzero messages

  Every such message turns into a weight $\geq d_2$ codeword under E_2

  $\Rightarrow \gamma$ has weight $\geq d_1 d_2$, QED.

Example 9 RS ◇ Hadamard

Assume $n = 2^m$

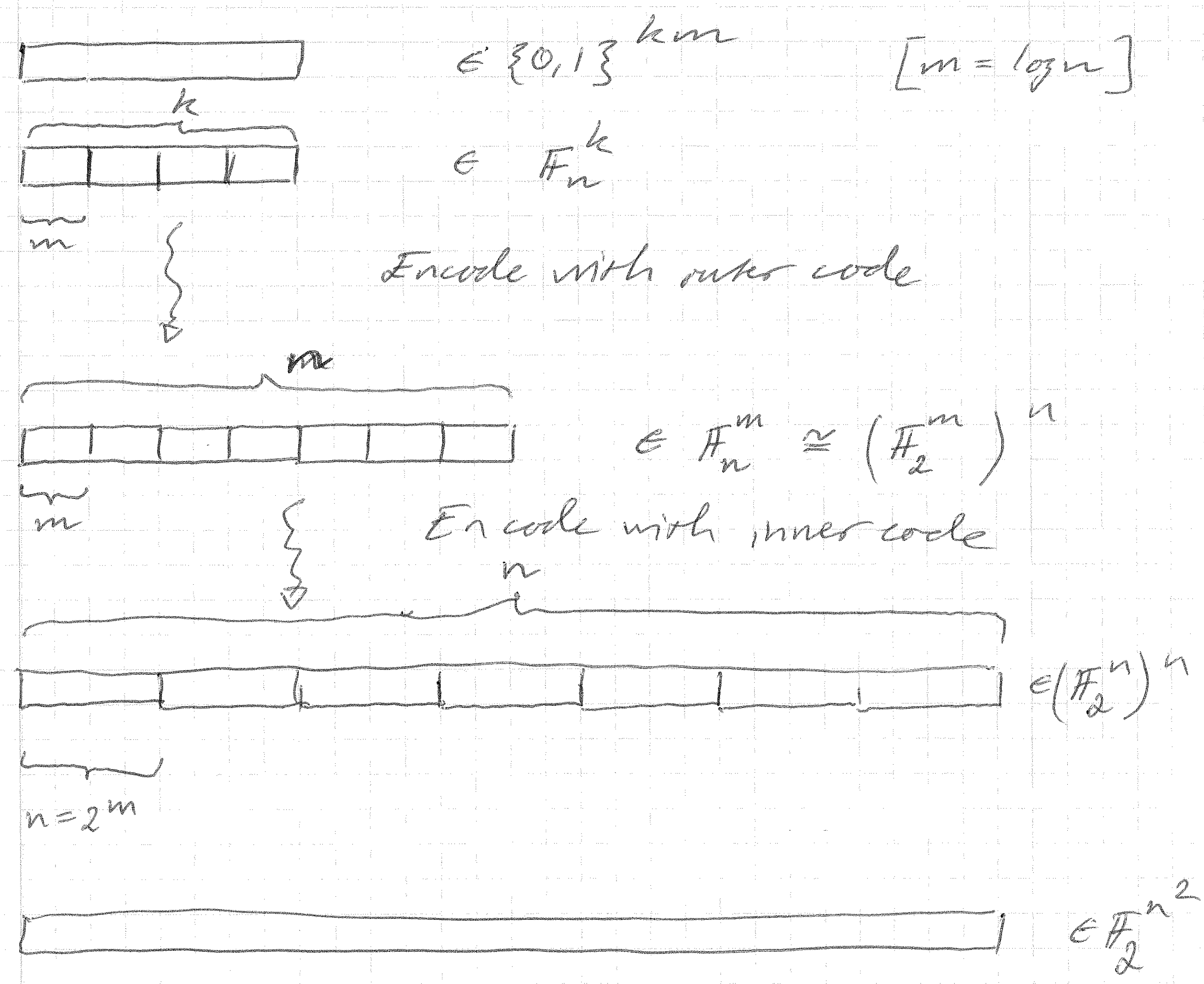
Take $[n, k, n-k+1]_n$ Reed-Solomon code as outer code C_1

Take $[n, \log n, n/2]_2$ Hadamard code as inner code C_2 (i.e., skipping complements for simplicity)

Obtain by concatenation

$$\left[n^2,\ k \log n,\ \frac{n}{2}(n-k+1)\right]_2 \text{ - code}$$

Illustration



Suppose we pick $k = \Theta(n)$ in outer code

Then $C_1 \diamond C_2$ has

- constant relative distance

$$\frac{n(n-k+1)/2}{n^2} \approx 1 - k/n$$

- inverse polynomial rate

$$\frac{k \log n}{n^2} \approx \frac{\log n}{n}$$

New range of parameters compared to codes we've seen earlier

What is the point of concatenation?
Outer code can be over large alphabet
$\Rightarrow$ easier to construct
Inner code allows us to shrink alphabet size

Can use multiple levels of concatenation to get better and better parameters
Won't get us all the way to asymptotically good codes, though

Informally, for this we need both outer and inner codes to be "asymptotically good" in some sense.

Can use RS codes as outer codes, but need better inner codes. Details not hard given what we know now, but still beyond scope of our limited survey of coding theory