Encoding, decoding with linear codes (§12.7, 12.8)

Having $\mathcal{C} \subset (\mathbb{F}_q)^n$ a linear code simplifies many things.

PROPOSITION: For a linear code $\mathcal{C} \subset (\mathbb{F}_q)^n$, one can compute the minimum distance

$$d(\mathcal{C}) \;\left[:= \min\{d(x,x') : x, x' \in \mathcal{C},\; x \neq x'\}\right]$$

as

$$d(\mathcal{C}) = \min\{\underbrace{d(y,\underline{0})}_{=\#\{i\,:\,y_i \neq 0\} =: wt(y)} : x \in \mathcal{C} - \{\underline{0}\}\}$$

called the Hamming weight of y

proof: Note by definition that

$$d(x,x') := \#\{i : x_i \neq x_i'\} = \#\{i : x_i - x_i' = 0\}$$
$$= d(x-x', \underline{0}) = wt(x-x')$$

Also when $\mathcal{C}$ is linear, since $\underline{0} \in \mathcal{C}$,

$$\{\underset{\substack{\| \\ d(x-x',\underline{0})}}{d(x,x')} : \underset{x \neq x'}{x, x' \in \mathcal{C}}\} = \{d(y,\underline{0}) : \underset{y \neq \underline{0}}{y \in \mathcal{C},}\} \quad \blacksquare$$

let $y = x-x'$

EXAMPLE The binary Hamming $[7,4,3]$-code (§12.4) was the basis for the parlor trick on the 1st day.

It has generator matrix

$$G = \begin{bmatrix} 1&0&0&0&1&1&0 \\ 0&1&0&0&1&0&1 \\ 0&0&1&0&0&1&1 \\ 0&0&0&1&1&1&1 \end{bmatrix}$$

row vectors	wt(–)
r_1	3
r_2	3
r_3	3
r_4	4

and also contains non-zero vectors

	wt(–)
$r_1+r_2 = [1\,1\,0\,0\,0\,1\,1]$	4
$r_1+r_4 = [1\,0\,0\,1\,1\,0\,0]$	3
$r_1+r_2+r_3 = [1\,1\,1\,0\,0\,0\,0]$	3
$r_1+r_2+r_4 = [1\,1\,0\,1\,0\,0\,1]$	4
$r_1+r_2+r_3+r_4 = [1\,1\,1\,1\,1\,1\,1]$	7

and a few more, but $d(\mathcal{C}) = \min\{3,4,7\}$
$= 3$
(as claimed in $[7,4,3]$)

How many in total, that is, what is $m = |\mathcal{C}|$?

PROPOSITION: A k-dim'l subspace $\mathcal{C} \subset (\mathbb{F}_q)^n$ has size $m = |\mathcal{C}| = q^k$.

So $[n,k,d]$ $\mathbb{F}_q$-linear codes are (n, q^k, d) q-ary codes.

with q-ary $\text{rate}_q(\mathcal{C}) = \dfrac{\log_q(m)}{n} = \dfrac{k}{n}$

---

proof: Pick any basis $w_1, \dots, w_k$ for $\mathcal{C}$.

Then we claim (checked below) that the map

$$(\mathbb{F}_q)^k \xrightarrow{f} \mathcal{C}$$

$$\underline{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_k \end{bmatrix} \longmapsto f(\underline{c}) = c_1 w_1 + c_2 w_2 + \dots + c_k w_k$$

is a bijection, so $|\mathcal{C}| = |(\mathbb{F}_q)^k| = \underbrace{q \cdot q \cdots q}_{k \text{ times}} = q^k$.

Surjectivity comes from the fact that $w_1, \dots, w_k$ span $\mathcal{C}$, by definition of spanning.

Injectivity comes from the lin. independence of the $w_1, \dots, w_k$: if $f(\underline{c}) = f(\underline{d})$ for some $\underline{c}, \underline{d}$

then $c_1 w_1 + \dots + c_k w_k = d_1 w_1 + \dots + d_k w_k$

$\Rightarrow (c_1 - d_1) w_1 + \dots + (c_k - d_k) w_k = \underline{0}$

$w_1, \dots, w_k$ lin. indep.

$\Rightarrow c_1 - d_1 = \dots = c_k - d_k = 0$

$\Rightarrow \underline{c} = \underline{d}$ $\blacksquare$

It's easier to work with generator matrices in...

DEF'N: **Standard form** for a generator matrix G of an $[n,k,d]$ q-ary code:

$$G = \left[\begin{array}{c|c} \underbrace{\begin{matrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 \end{matrix}}_{k \times k \text{ identity matrix } I_k} & \underbrace{\quad A \quad}_{n-k \text{ columns}} \end{array}\right] \Bigg\} \, k \text{ rows}$$

A: an arbitrary $k \times (n-k)$ matrix with entries in $\mathbb{F}_q$

EXAMPLES (1) We just gave $[7,4,3]$ **Hamming code** via a standard form generator matrix

$$G = \left[\begin{array}{c|c} \underbrace{\begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{matrix}}_{I_4} & \underbrace{\begin{matrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{matrix}}_{A} \end{array}\right]$$

(2) The **binary parity check code** $C = \left\{ \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} : \begin{array}{l} x_i \in \mathbb{F}_2 \\ \sum x_i = 0 \end{array} \right\}$ has a standard form generator matrix

$$G = \left[\begin{array}{c|c} \underbrace{\begin{matrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{matrix}}_{I_{n-1}} & \underbrace{\begin{matrix} 1 \\ 1 \\ \vdots \\ 1 \end{matrix}}_{A} \end{array}\right]$$

PROPOSITION Not every linear code $\mathcal{C}$ has a generator matrix G in standard form, but if we apply a single permutation to its columns, we can make a new code $\mathcal{C}'$ that does (and has all the same parameters $[n,k,d]$).

Proof: 1. Start with any generator matrix for $\mathcal{C}$.

$$G = \begin{bmatrix} * & * & \cdots & * \\ * & * & \cdots & * \\ \vdots & \vdots & & \vdots \\ * & * & \cdots & * \end{bmatrix}$$

2. Use Gaussian elimination
$= \text{row operations} \begin{cases} \text{swapping rows} \\ \text{scaling rows by } c \in \mathbb{F}_q^{\times} \\ \text{adding rows to each other} \end{cases}$

to put it in row-reduced echelon form

$$G = \begin{bmatrix} 0 & \cdots & 0 & 1 & * & \cdots & * & 0 & * & \cdots & * & 0 & * & \cdots & * \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & 1 & * & \cdots & * & 0 & * & \cdots & * \\ & \vdots & & & & & & & & & & & & & \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & 1 & * & \cdots & * \end{bmatrix}$$

all zeroes here

3. If needed, apply a permutation of columns to make the pivot columns all to the left:

$$G = \left[\begin{array}{ccc|ccc} 1 & & O & * & \cdots & * \\ & 1 & & * & \cdots & * \\ O & & \ddots \; 1 & * & \cdots & * \end{array}\right]$$

∎

EXAMPLE The 3-fold ternary repetition code $\mathcal{C}$ in $(\mathbb{F}_3)^3$

$$\mathcal{C} = \left\{ \begin{bmatrix}0\\0\\0\end{bmatrix}, \begin{bmatrix}1\\1\\1\end{bmatrix}, \begin{bmatrix}2\\2\\2\end{bmatrix} \right\}$$ has 2nd extension

$$\mathcal{C}^{(2)} = \left\{ (\omega_1, \omega_2) : \omega_i \in \mathcal{C} \right\} = \underbrace{\left\{ \begin{bmatrix}0\\0\\0\\0\\0\\0\end{bmatrix}, \begin{bmatrix}0\\0\\0\\1\\1\\1\end{bmatrix}, \begin{bmatrix}0\\0\\0\\2\\2\\2\end{bmatrix}, \begin{bmatrix}1\\1\\1\\0\\0\\0\end{bmatrix}, \dots, \begin{bmatrix}2\\2\\2\\2\\2\\2\end{bmatrix} \right\}}_{m = 3\cdot 3 = 9}$$

is a $[6,2,3]$ ternary linear code.

$x_1 = x_2 = x_3, \ x_4 = x_5 = x_6$

$G = \begin{bmatrix} 2&2&2&1&1&1 \\ 1&1&1&2&2&2 \end{bmatrix}$ is **not** a generator matrix for it, (why?)

but $G = \begin{bmatrix} 2&2&2&0&0&0 \\ 1&1&1&2&2&2 \end{bmatrix}$ is, although not in standard form.

swap rows

$\begin{bmatrix} 1&1&1&2&2&2 \\ 2&2&2&0&0&0 \end{bmatrix}$ ← pivot

subtract 2 (row 1) from row 2

$\begin{bmatrix} 1&1&1&0&0&0 \\ 0&0&0&2&2&2 \end{bmatrix}$

scale row 2 by $2^{-1} = 2$

$\begin{bmatrix} 1&1&1&0&0&0 \\ 0&0&0&1&1&1 \end{bmatrix}$ ← row-reduced echelon form but not standard form

swap columns 2 & 4

$x_1 = x_3 = x_4, \ x_2 = x_5 = x_6$

$G = \left[\begin{array}{cc|cccc} 1&0&1&1&0&0 \\ 0&1&0&0&1&1 \end{array}\right]$ generates a **different** code than $\mathcal{C}^{(2)}$, but both are $[6,2,3]$ ternary codes

Encoding becomes particularly simple if $\mathcal{C}$ has generator $G = \left[\begin{array}{c|c} I_k & A \end{array}\right]$ in standard form

$$= \left[\begin{array}{cccc|cccc} 1 & & & & | & | & & | \\ & 1 & & & c_1 & c_2 & \cdots & c_{n-k} \\ & & \ddots & 1 & | & | & & | \end{array}\right].$$

Given a word $v = (v_1, \dots, v_k)$ with k letters in $(\mathbb{F}_q)^k$, apply the encoding map

$$\begin{aligned} (\mathbb{F}_q)^k &\longrightarrow (\mathbb{F}_q)^n \\ \underset{[v_1, \dots, v_k]}{\overset{v}{\shortparallel}} &\longmapsto vG \\ &= \left[\, vI_k \mid vA \,\right] \\ &= [\underbrace{v_1, \dots, v_k}_{\text{called the information digits}} \mid \underbrace{v \cdot c_1, \dots, v \cdot c_{n-k}}_{\text{called the check digits}}] \end{aligned}$$

usual dot product: $v \bullet w = v_1 w_1 + \dots + v_k w_k = [v_1 \cdots v_k] \begin{bmatrix} w_1 \\ \vdots \\ w_k \end{bmatrix}$

EXAMPLES

(1) Binary parity check code $\mathcal{C} \subset (\mathbb{F}_2)^n$

had $G = \left[\begin{array}{c|c} \underbrace{\begin{matrix} 1 & & 0 \\ & 1 & \\ & & \ddots \\ 0 & & 1 \end{matrix}}_{I_{n-1}} & \begin{matrix} 1 \\ 1 \\ \vdots \\ 1 \end{matrix} \end{array}\right]$ in standard form,

and encodes $v = [v_1, \dots, v_{n-1}] \in (\mathbb{F}_2)^{n-1}$

as $vG = [\underbrace{v_1, \dots, v_{n-1}}_{\text{info bits}}, \mid \underbrace{v_1 + v_2 + \dots + v_{n-1}}_{\text{parity check bit}}] \in (\mathbb{F}_2)^n$

---

(2) The binary Hamming $[7,4,3]$ code $\mathcal{C}$ had

$$G = \left[\begin{array}{cccc|ccc} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{array}\right]$$

so it encodes $v = [v_1, v_2, v_3, v_4] \in (\mathbb{F}_2)^4$

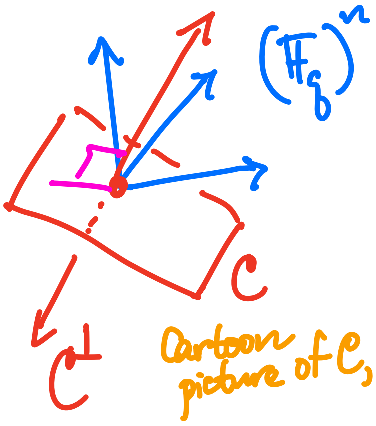
as $vG = [\underbrace{v_1\ v_2\ v_3\ v_4}_{\text{info bits}} \mid v_1+v_2+v_4 \quad v_1+v_3+v_4 \quad v_2+v_3+v_4]$

3 check bits

Dual codes (§12.8)

DEF'N: Given a linear code $\mathcal{C} \subset (\mathbb{F}_q)^n$, its dual code (perp) (perpendicular) $\mathcal{C}^\perp := \{ y \in \mathbb{F}_q^n : x \cdot y = 0 \ \forall x \in \mathcal{C} \}$

usual dot product

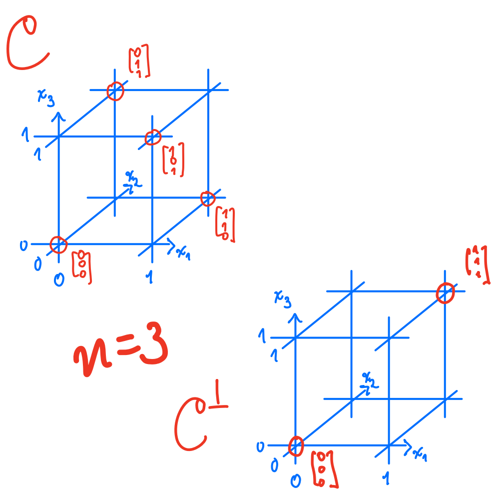


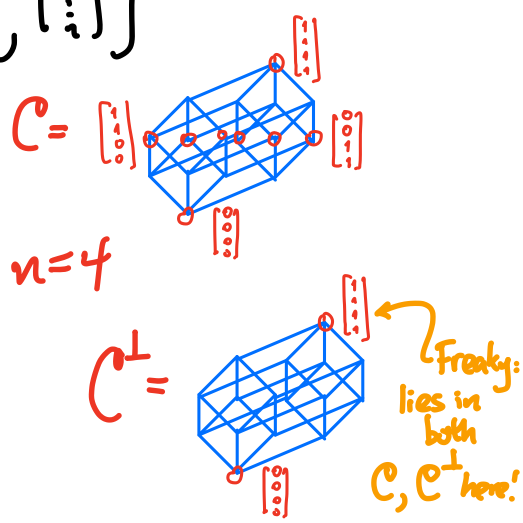
Cartoon picture of $\mathcal{C}, \mathcal{C}^\perp$

We think of the vectors $y \in \mathcal{C}^\perp$ as being the parity checks (over $\mathbb{F}_2$) on the vectors $x \in \mathcal{C}$.

EXAMPLE The binary parity check code $\mathcal{C} \subset (\mathbb{F}_2)^n$ has $\mathcal{C}^\perp$ = the binary repetition code of length n

$$= \left\{ \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \right\}$$





PROPOSITION

(i) If $\mathcal{C}$ is a k-dim'l linear code in $(\mathbb{F}_q)^n$ then $\mathcal{C}^\perp$ is an $(n-k)$-dim'l linear code in $(\mathbb{F}_q)^n$.

(ii) Furthermore, if $\mathcal{C}$ has generator matrix

$$G = \left[\; I_k \;\middle|\; A \;\right]$$ in standard form,

then $\mathcal{C}^\perp$ has generator matrix (not in standard form)

$$H = \left[\, -A^t \;\middle|\; I_{n-k} \,\right]$$ (sometimes called a check matrix for $\mathcal{C}$).

(iii) Lastly, $(\mathcal{C}^\perp)^\perp = \mathcal{C}$.

EXAMPLE The 3-dim'l linear code $\mathcal{C} \subset (\mathbb{F}_3)^5$ with generator matrix $G = \left[\begin{array}{ccc|cc} 1 & 0 & 0 & 1 & 2 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 2 \end{array}\right]$

has dual code $\mathcal{C}^\perp \subset (\mathbb{F}_3)^5$ of dimension $\overset{n-k=}{5-3=2}$

and generator matrix

$$H = \left[\begin{array}{ccc|cc} -1 & -0 & -0 & 1 & 0 \\ -2 & -1 & -2 & 0 & 1 \end{array}\right] = \left[\begin{array}{ccc|cc} 2 & 0 & 0 & 1 & 0 \\ 1 & 2 & 1 & 0 & 1 \end{array}\right]$$

(sketch) proof of PROP:

$\mathcal{C}^\perp$ is always a subspace since $y, y' \in \mathcal{C}^\perp$

$$\Rightarrow \begin{array}{l} y\cdot x = 0 \\ y'\cdot x = 0 \end{array} \quad \forall x \in \mathcal{C} \Rightarrow \begin{array}{l} (cy)\cdot x = c(y\cdot x) = c\cdot 0 = 0 \\ (y+y')\cdot x = y\cdot x + y'\cdot x \\ \qquad = 0+0=0 \end{array}$$

For the rest of the proof assume, by re-indexing coordinates in $(\mathbb{F}_q)^n$, that $\mathcal{C}$ has generator matrix

$$G = [\, I_k \mid \underbrace{A}_{n-k} \,]\}k$$ in standard form

and let $H = [-A^t \mid I_{n-k}]$ as in the PROP.

It's easy to check the rows of H lie in $\mathcal{C}^\perp$, that is, they dot to 0 with rows of G:

$$\underset{i=1,\dots,k}{(\text{row } i \text{ of } G)} \cdot \underset{j=1,\dots,n-k}{(\text{row } j \text{ of } H)} = [\overbrace{0 \cdots \underset{\uparrow\, i^{th}}{1} \cdots 0}^{k} \mid \overbrace{(\text{row } i \text{ of } A)}^{n-k}] \cdot [\overbrace{-(\text{row } j \text{ of } A^t)}^{k} \mid \overbrace{0 \cdots \underset{\uparrow\, j^{th}}{1} \cdots 0}^{n-k}]$$

$$= -a_{ij} + a_{ij} = 0$$

The rows of $r_1, \dots, r_{n-k}$ of H are lin. indep. inside $\mathcal{C}^\perp$ because of the I_{n-k} in the rightmost columns of H.

Thus it only remains to show $r_1, \ldots, r_{n-k}$ span $\mathcal{C}^\perp$, and then they would be a basis for $\mathcal{C}^\perp$, showing all of the rest of (i) & (ii) (and then (iii) follows by swapping roles of $\mathcal{C}, \mathcal{C}^\perp$).

To see the spanning, given $y = [d_1 \cdots d_k \; c_1 \cdots c_{n-k}] \in \mathcal{C}^\perp$, we claim $y = c_1 r_1 + \cdots + c_{n-k} r_{n-k}$:

Note $y' := y - (c_1 r_1 + \cdots + c_{n-k} r_{n-k})$ also lies in $\mathcal{C}^\perp$ and has the form $y' = [d_1' \cdots d_k' \; 0 \cdots 0]$,

but then $\underset{\text{for } i=1,2,\ldots,k}{0 = (\text{row } i \text{ of } G) \cdot y' = d_i'}$ forces $y' = 0$ $\boxtimes$

This has a useful consequence (discussed in §14.1).

COROLLARY: Given dual linear codes $\mathcal{C}$ and $\mathcal{C}^\perp$, the min. distance $d(\mathcal{C})$ has this reformulation:

$d(\mathcal{C})$ = smallest number d of columns in the generator matrix H for $\mathcal{C}^\perp$ involved in a nontrivial lin. dependence

$$H = \underbrace{\begin{bmatrix} | & | & & | \\ v_1 & v_2 & \cdots & v_n \\ | & | & & | \end{bmatrix}}_{\text{columns of } H}$$

proof: Since $\mathcal{C} = (\mathcal{C}^\perp)^\perp = (\text{row space of } H)^\perp$,

the (nonzero) vectors $x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \in \mathcal{C}$

are the same as (nonzero) vectors in the nullspace of H

i.e. $\underline{0} = Hx = \begin{bmatrix} | & & | \\ v_1 & \cdots\cdots & v_n \\ | & & | \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = x_1 v_1 + \cdots + x_n v_n$

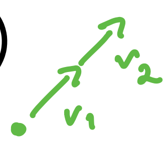
i.e. (non-trivial) linear dependences among $v_1, \ldots, v_n$

and the Hamming weight $wt(x) = d$ tells us how many v_i's are actually used in the dependence.

So minimizing the d gives $d(\mathcal{C}) = \min\{wt(x) : x \in \mathcal{C} - \{\underline{0}\}\}$. $\blacksquare$

Note this says H the $(n-k) \times n$ gen. matrix for $\mathcal{C}^\perp$ having

- no zero columns $\Rightarrow d(\mathcal{C}) \neq 1$, so $d(\mathcal{C}) \geq 2$
- no pair of dependent columns (parallel) $\Rightarrow d(\mathcal{C}) \neq 2$, so $d(\mathcal{C}) \geq 3$.



IDEA: Try to find such H with $n-k$ small, so k is large and $\text{rate}(\mathcal{C}) = \frac{k}{n}$ is large.

EXAMPLE: This is exactly how Hamming cooked up his $[7,4,3]$ binary code, and more generally, the Hamming $[\underbrace{2^r-1}_{n}, 2^r-1-r, 3]$ codes $\mathcal{C}_r$:

pick $\mathcal{C}_r^\perp$ to have $r \times (2^r-1)$ generator matrix H_r whose columns are all nonzero vectors in $(\mathbb{F}_2)^r$:

$$H_2 = \left[\left.\underbrace{\begin{matrix}1\\1\end{matrix}}_{-A^t}\,\right|\,\underbrace{\begin{matrix}1&0\\0&1\end{matrix}}_{I_2}\right]\Big\}\, r=2 \;\Rightarrow\; G_2 = \left[\underbrace{1}_{I_1}\,\middle|\,\underbrace{1\quad 1}_{A}\right]$$

generates binary 3-fold repetition $[3,1,3]$-code

$$H_3 = \left[\left.\underbrace{\begin{matrix}1&1&0&1\\1&0&1&1\\0&1&1&1\end{matrix}}_{-A^t}\,\right|\,\underbrace{\begin{matrix}1&0&0\\0&1&0\\0&0&1\end{matrix}}_{I_3}\right]\Bigg\}\, r=3 \;\Rightarrow\; G_3 = \left[\left.\underbrace{\begin{matrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{matrix}}_{I_4}\,\right|\,\underbrace{\begin{matrix}1&1&0\\1&0&1\\0&1&1\\1&1&1\end{matrix}}_{A}\right]$$

generates binary Hamming $[7,4,3]$-code

$$H_r = \left.\left[\begin{array}{c|c} \underbrace{\quad -A^t \quad}_{2^r-1-r} & \underbrace{\begin{matrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \end{matrix}}_{r} \end{array}\right]\right\} r=3 \Rightarrow G_r = \left[\begin{array}{c|c} \underbrace{\begin{matrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \end{matrix}}_{2^r-1-r} & \underbrace{A}_{r} \end{array}\right]$$

other nonzero vectors in $(\mathbb{F}_2)^r$

generates binary Hamming $[2^r-1, 2^r-1-r, 3]$-code

Their rates quickly improve as r grows:

$$\text{rate}(\mathcal{C}_r) = \frac{k}{n} = \frac{2^r-1-r}{2^r-1} = 1 - \frac{r}{2^r-1} \xrightarrow[\text{as } r\to\infty]{} 1$$

But their min. dist. $d(\mathcal{C}_r) = 3 \ \forall r$, which doesn't lead to any better error-correction than $1 = \left\lfloor \frac{3-1}{2} \right\rfloor$.

REMARK Hamming more generally defined his $\mathbb{F}_q$-linear $[n, k, d]$-codes the same way:

$$n = \frac{q^r-1}{q-1}, \quad k = \frac{q^r-1}{q-1} - r, \quad d = 3$$

$\mathcal{C}^\perp$ has generator matrix H whose columns pick one vector from each line through $\underline{0}$ in $(\mathbb{F}_q)^r$.

EXERCISE: Why are there $\frac{q^r-1}{q-1}$ such lines?

Syndrome decoding (§12.8)

Given our $[n,k,d]$ linear code $\mathcal{C} \subset (\mathbb{F}_q)^n$, after the transmitter encodes their message as some $x \in \mathcal{C}$, suppose some noise in transmission lets us receive $y \in (\mathbb{F}_q)^n$.

Q: How do we do min. distance decoding of $y \in (\mathbb{F}_q)^n$ efficiently, that is, how to find some $x' \in \mathcal{C}$ minimizing $d(x',y)$?

The method called syndrome decoding works pretty well, and starts by having us pre-compute

$$H = {\scriptstyle n-k}\Big\{\left[\, -A^t \mid I_{n-k} \,\right] \text{ generating } \mathcal{C}^\perp$$

from $G = {\scriptstyle k}\Big\{\left[\, I_k \mid A \,\right]$ generating $\mathcal{C}$.

DEF'N: The syndrome for $y \in (\mathbb{F}_q)^n$ is the vector $Hy \in (\mathbb{F}_q)^{n-k}$

$$Hy = n-k\left\{\begin{bmatrix} -A^t & | & I_{n-k} \end{bmatrix}\right. \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} y \cdot (\text{row } 1 \text{ of } H) \\ y \cdot (\text{row } 2 \text{ of } H) \\ \vdots \\ y \cdot (\text{row } n-k \text{ of } H) \end{bmatrix}$$

NOTE: Garrett calls yH^t the syndrome of y. This is just the same row vector instead of a column vector.

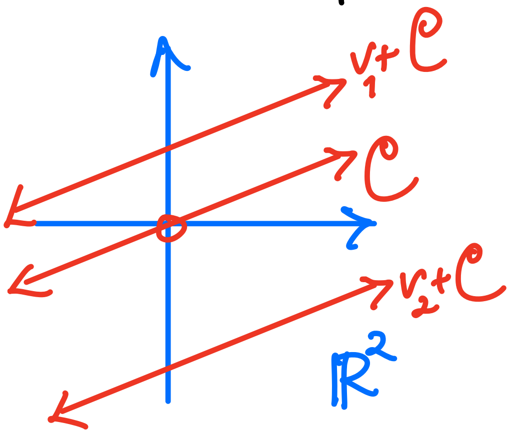
How does the syndrome Hy help decode y?

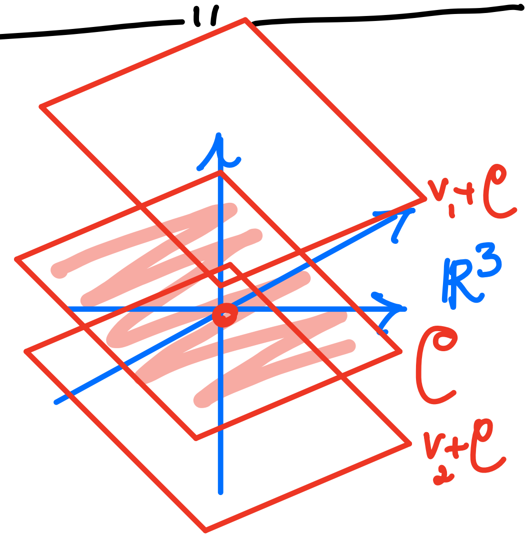
It turns out that $(\mathbb{F}_q)^n$ decomposes disjointly into sets (affine subspaces parallel to $\mathcal{C}$) called the cosets $v + \mathcal{C} := \{v + x : x \in \mathcal{C}\}$ of the subspace $\mathcal{C}$, and we can read off which coset y lies in from its syndrome Hy.

EXAMPLES:

(1) Cosets of lines $\mathcal{C}$ through $\{0\}$ are its parallel lines

— " — planes ——— " ——— planes

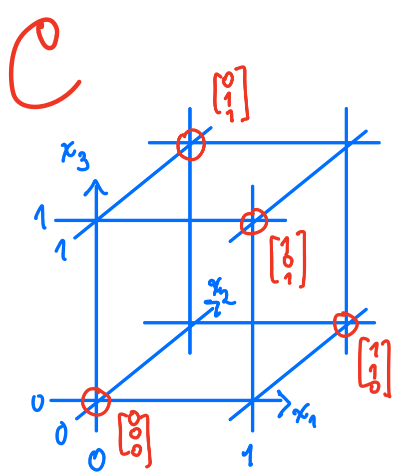


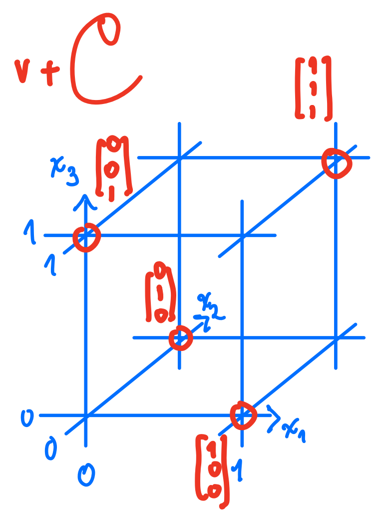


(2) Similar idea over finite fields $\mathbb{F}_q$,

e.g. binary parity check code $\mathcal{C} = \left\{ \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} : \begin{array}{l} x_i \in \mathbb{F}_2 \\ x_1+\dots+x_n=0 \end{array} \right\}$

has one other coset $v+\mathcal{C} = \left\{ \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} : \begin{array}{l} x_i \in \mathbb{F}_2 \\ x_1+\dots+x_n=1 \end{array} \right\}$





PROPOSITION:

(i) Two cosets $v+\mathcal{C}$, $v'+\mathcal{C}$ intersect at all

$\overset{(a)}{\iff}$ the cosets are the same: $v+\mathcal{C} = v'+\mathcal{C}$

$\overset{(b)}{\iff}$ $v-v' \in \mathcal{C}$

$\overset{(c)}{\iff}$ $Hv = Hv'$ in $(\mathbb{F}_q)^{n-k}$, i.e. v, v' have same syndrome

(ii) All cosets $v+\mathcal{C}$ have same size as $\mathcal{C}\ (= \underline{0}+\mathcal{C})$, namely $|v+\mathcal{C}| = |\mathcal{C}| = q^k$ if $k = \dim_{\mathbb{F}_q}(\mathcal{C})$

So the cosets $v+\mathcal{C}$ disjointly decompose $\mathbb{F}_q^n$ into q^{n-k} sets, each of size q^k

---

proof: For (i), certainly if $v+\mathcal{C} = v'+\mathcal{C}$ then they intersect, but conversely if $w \in (v+\mathcal{C}) \cap (v'+\mathcal{C})$ then $w = v+x = v'+x'$ for some $x, x' \in \mathcal{C}$

so $v-v' = x'-x \in \mathcal{C}$

and then $v+\mathcal{C} = v' + \underbrace{(v-v')+\mathcal{C}}_{=\mathcal{C} \text{ since } v-v' \in \mathcal{C}} = v'+\mathcal{C}$.

This shows (a), (b).

For (c), note $v - v' \in \mathcal{C}$

$\iff v - v' \in (\mathcal{C}^\perp)^\perp$

$\iff v - v'$ has zero dot product with all vectors in $\mathcal{C}^\perp$ = row space of H

$\iff (v - v') \cdot (\text{row } i \text{ of } H)$ $\forall i = 1, \ldots, n-k$

$\iff H(v - v') = \underline{0}$

$\iff Hv = Hv'$

For (ii), note that the maps

$$\begin{array}{ccc} \mathcal{C} & \underset{g}{\overset{f}{\rightleftarrows}} & v + \mathcal{C} \\ x & \overset{f}{\longmapsto} & v + x \\ x = y - v & \underset{g}{\longleftarrow\!\shortmid} & y = v + x \end{array}$$

are mutually inverse bijections,

so $|v + \mathcal{C}| = |\mathcal{C}| = q^k$ $\boxtimes$

SYNDROME DECODING FOR $\mathcal{C}$:

Given H a $(k-n) \times n$ matrix generating $\mathcal{C}^\perp$, do a (one-time) precomputation to find in each of the q^{n-k} cosets $v + \mathcal{C}$ a coset leader e_{min} such that $wt(e_{min}) = \min\{wt(v) : v \in e_{min} + \mathcal{C}\}$.

Tabulate these coset leaders v_{min} and their syndromes He_{min} in a syndrome table.

Then when you receive the transmitted word $y \in (\mathbb{F}_q)^n$, compute its syndrome Hy, find the unique coset leader e_{min} having $Hy = He_{min}$, and

decode y as $x' = y - e_{min}$.

PROPOSITION: Syndrome decoding is min. distance decoding, that is,

$$d(x', y) \leq d(x, y) \quad \forall x \in \mathcal{C} \text{ if } x' = y - e_{min}$$

where $Hy = He_{min}$ and e_{min} has smallest Hamming weight in $e_{min} + \mathcal{C}$.

---

proof: Suppose not, that is, there exists some $x \in \mathcal{C}$ with

$$\begin{array}{ccc} d(y, x) & < & d(y, x') \\ \| & & \| \\ d(\underline{0}, y-x) & & d(y, y-e_{min}) \\ \| & & \| \\ wt(y-x) & & d(\underline{0}, e_{min}) \\ & & \| \\ & & wt(e_{min}) \end{array}$$

$v := y - x$ lies in $y + \mathcal{C} = e_{min} + \mathcal{C}$ since $Hy = He_{min}$. Contradiction $\boxtimes$

EXAMPLE of syndrome decoding.

Suppose $\mathcal{C}$ is the $[5,3,2]$ code in $(\mathbb{F}_2)^5$ with generator matrix $G = \left[\begin{array}{ccc|cc} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{array}\right] = [I_3 \mid A]$

so $\mathcal{C}^\perp$ has gen. matrix $H = \left[\begin{array}{ccc|cc} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{array}\right] = [-A^t \mid I_2]$

We pre-compute a syndrome table by brute force:

a coset leader e_{min}	syndrome He_{min}
$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$
$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$
$\begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 1 \end{bmatrix}$
$\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1 \end{bmatrix}$

On transmitter's end, say they encode info $v = [1\,1\,1]$ as

$$x = vG = [1\,1\,1]\left[\begin{array}{ccc|cc} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{array}\right] = [1\,1\,1\,|\,1\,0]$$

and in transmission it is corrupted and received as one of these:

compute syndrome Hy

subtract e_{min}

$$y = [1\,0\,1\,1\,0]$$

$$Hy = \left[\begin{array}{ccc|cc} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{array}\right]\begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

matching He_{min} for $e_{min} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$

$$x' = y - e_{min} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

$= x$

Success!

$$y' = [1\,1\,0\,1\,0]$$

$$Hy' = \left[\begin{array}{ccc|cc} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{array}\right]\begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

matching same e_{min}

$$x' = y - e_{min} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

$\neq x$

failure

(inevitable since 1 error occurred and $d(\mathcal{C}) = 2$)