2/8

Block Linear Code

Let $F = GF(q)$ be the underlying alphabet. (we will mainly work with $GF(2)$ & $GF(2^m)$)

An (n, M, d) code C over F is said to be linear if C is a subspace of F^n over F.

$\Rightarrow \forall c_1, c_2 \in C, \; a_1, a_2 \in F, \; a_1 c_1 + a_2 c_2 \in C$

Dimension of a linear code C is the dimension of the subspace C. We say that the code is an (n, k, d) - linear code if the dimension is k.

Since every basis of C will contain k codewords, whose linear combinations are all distinct, we have $|C| = M = q^k$

Rate $R = \dfrac{\log_{|F|} M}{n} = \dfrac{k}{n}$

Eg Simple parity-check code over $GF(2)$. $\rightarrow [3, 2, 2]$

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

a subspace of $\{0,1\}^3$ over $GF(2)$

Spanned by $\begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$

$k = 2, \; n = 3, \;$ Rate $= \dfrac{2}{3}$

A generator matrix of a $[n, k, d]$ - linear code C over F is a $k \times n$ matrix whose rows form a basis of C.

Eg 1 - Parity check code

$$G = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \quad \text{or} \quad G' = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

Note that rank $G = k$.

Eg 2 - For a $(3,1,3)$ - repetition code

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

Minimum distance

Claim → For an (n, k, d) - linear code over F,

$$d = \min_{c \in C \setminus \{\bar{0}\}} \omega(c)$$

, where $\omega(c)$ = # non-zero entries in c. called Hamming weight of c.

Pf →

$$d = \min_{\substack{c_1, c_2 \in C \\ c_1 \neq c_2}} d(c_1, c_2) = \min_{\substack{c_1, c_2 \in C \\ c_1 \neq c_2}} \omega(c_1 - c_2)$$

Sin C is linear, $c_1 - c_2$ is a codeword as well

So $$d = \min_{c \neq \bar{0}} \omega(c)$$

Eg -
$$C = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

$$\min_{c \neq \bar{0}} \omega(c) = 2$$

$$d(C) = 2$$

<u>Encoding</u> – q^k codewords in (n,k,d) linear code

One to one map to ~~i~~ source codeword u by thinking of u as all possible vectors in F^k

$u = (u_0, u_1, \ldots u_{k-1})$ and mapping

$u \rightarrow uG$. since G is full-rank, this is a one-to-one map.

<u>Eg</u> $(7, 4, 3)$ Hamming code over $GF(2)$

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

u	c
0000	0000000
0001	0001101
0010	0010110
0100	⋮
1000	⋮
1100	⋮
0011	0011010 → $w(c)=3$.
⋮	
1111	

$C = \{\ldots\}$

$2^4 = 16$. messages

$$\begin{array}{cc} k\times k & k\times n-k \\ \uparrow & \uparrow \end{array}$$

Note that G has the form $G = [I \mid A]$. Such a generator matrix is said to be systematic.

$$\text{Codeword} = \Big[\underbrace{\text{Message bit}}_{k} \mid \underbrace{\overset{\text{Parity-check}}{\text{bits}}}_{n-k}\Big]$$

$$uG = (u_0, u_1, \dots u_k, P_0, P_1 \dots P_{n-k-1})$$

For (7,4,3)-Hamming code, each codeword $(c_0, c_1 \dots c_6)$:

$c_0 = u_0$, $c_1 = u_1$, $c_2 = u_2$, $c_3 = u_3$,

$c_4 = u_0 + u_2 + u_3$, $c_5 = u_0 + u_1 + u_2$, $c_6 = u_1 + u_2 + u_3$

For any linear code & generator matrix G, it is always possible to create an equivalent code with a systematic generator matrix G' by permuting coordinates of codeword & using elementary row operations.

Eg - (5,3,2) code $G = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix}$

Add R_2 to R_3

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix} \xrightarrow{\text{Add } R_3 \text{ to } R_1} \begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix} \rightarrow G^{sys.}$$

Verify codeword for $(0\ 1\ 1) = u$

$uG = (0\ 0\ 1\ 1\ 1) \neq uG^{sys.} = (0\ 1\ 1\ 1\ 0)$

Parity-check Matrix

C is a subspace and let $C^\perp$ be the dual subspace which consists of all vectors which are orthogonal to C.

So $C^\perp$ itself is a subspace and can be used as a code, called dual code.

$\mathrm{Dim}(C^\perp) = n-k \Rightarrow$ Any basis has $n-k$ lin. indep. vectors

Let H be an $n-k \times n$ matrix whose rows form a basis of $C^\perp$. Then, by definition, $\forall c \in C$, we have

$Hc^T = \bar{0}$. ~~Also~~ Also implies $HG^T = \bar{0}$ $[n-k \times k]$

H is called the parity-check matrix of code C.

Eg $(7,4,3)$ - HC. over $GF(2)$

$$H = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}.$$

Codeword for (0011) is 0011010

check that $Hc^T = \bar{0}$

Particularly for a systematic $G = [I \mid A]$, $H = [-A^T \mid I]$

Eg - Parity check matrix of $(3,2,2)$ simple parity check code is $(1\ 1\ 1)$, which is generator matrix of $(3,1,3)$ repetition code. And vice-versa. These are dual codes.

Claim – Let H be the parity-check matrix of a ~~(n,k)~~ linear code C. Then, minimum distance of C is the largest integer d s.t. every set of $d-1$ columns in H are lin. ind.

Pf – Say $c = (c_1, c_2 \ldots c_n)$ with weight t and let $H = [h_1 \; h_2 \ldots h_n]$. Then $Hc^T = 0$ means

$$\sum_{j=1}^{n} c_j h_j = \bar{0}$$ $\Rightarrow$ t columns of H corresponding to non-zero elements of c are linearly dependent.

$\Rightarrow \exists$ a set of d columns of H which are linearly dep.

Conversely, if $\exists$ a set of t columns of H which are linearly dependent. Then $\exists$ some linear comb. of these columns which sum upto the zero-vector. Then taking these coefficients, we can create a codeword c of weight t s.t $Hc^T = 0$

Since min. weight of any codeword is d, only possible for d or more columns. So any subset of $d-1$ columns is lin. independent.

Qn: → is subspace a subgroup?
In particular, does subspace contain additive inverse

Decoding of linear codes

Consider an (n, k, d)-linear code over $GF(q)$. Received word

$r = c + e$ ← an element of $[q^n]$. A decoding scheme partitions them into

$e \to$ error vector

q^k sets $D_1, D_2 \ldots D_{q^k}$ such that all vectors in D_i are decoded to codeword c_i.

Standard array → A method to partition the possible received words, used to implement nearest codeword decoding.

Based on idea of coset decomposition

q^{n-k} rows

$$\begin{array}{ccccc}
c_1 = (0, 0 \ldots 0) & c_2 & c_3 & c_4 \; \ldots & c_{q^k} \\
e_2 & e_1 + c_2 & \ldots & & e_1 + c_{q^k} \\
\vdots & & & & \\
e_{q^{n-k}} & e_{q^{n-k}} + c_2 & \ldots & & e_{q^{n-k}} + c_{q^k}
\end{array}$$

e_2 chosen as an n-tuple not seen in first row. In general, e_j chosen as an n-tuple not seen before. As argued before, this partitions set of n-tuples into q^{n-k} disjoint cosets, each associated with a coset leader.

Eg $(5, 2, 3)$ code over $GF(2)$

$$G = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix} \qquad C = \begin{cases} 00000 \\ 10111 \\ 01101 \\ 11010 \end{cases}$$

00000	10111	01101	11010
00001	10110	01100	11011
00010	10101	01111	11000
00100	10011	01001	11110
01000	11111	00101	10010
10000	00111	11101	01010
00011	10100	01110	11001
00110	10001	01011	11100

We will use each column i as $D_i \Rightarrow$ each received word is decoded to its corresponding codeword at top of column. Note that if c_j is tx-ed, then received word r_j will be in D_j if the error pattern is the coset leader. However if the error pattern is not a coset leader, then decoding will be incorrect.

Note that error pattern x will be in same coset as r_j since their difference is a codeword. Say

$x = e_l + c_i$ (l^{th} coset, c_i^{th} codeword)

Then, $r_j = c_j + x = c_j + e_l + c_i = e_l + \hat{c} \Rightarrow r_j$ decoded to $\hat{c} + c$

So we have the following claim:

- Under standard array decoding, (n, k, d) - linear code can correct the q^{n-k} error patterns corr. to coset leaders in standard array.

So, how to select coset leaders to minimize prob. of error? In many channels like BSC, error patterns of lower weight are more likely. So makes sense to select coset leader as vector of least weight from remaining available vectors. This in fact will correspond to nearest codeword (or minimum distance) decoding. To see this, consider received word r; say it is found in l^{th} coset, i^{th} column. So $r = e_l + c_i$ and decode to codeword c_i. So $d(r, c_i) = \omega(e_l)$, where $\omega(e_l)$ is # non-zero elements of e_l. Now, considering $d(r, c_j)$ for some $j \neq i$. $d(r, c_j) = d(e_l + c_i, c_j)$
$= \omega(e_l + c_i - c_j)$

$c_i - c_j \in C$

$e_l + c_i - c_j$ is also in l^{th} coset & so by construction,

$\omega(e_l + c_i - c_j) \geq \omega(e_l) \Rightarrow d(r, c_j) \geq d(r, c_i)$.

So minimum distance decoding.

In fact,

Claim → For (n,k,d) - linear code & $t = \left\lfloor \frac{d-1}{2} \right\rfloor$, all n-tuples of weight $\leq t$ can be used as coset leaders of a standard array of C.

⇒ All error patterns of weight $\leq t$ can be corrected using this approach.

Pf → Argument follows by showing that no two tuples x, y of weight $\leq t$ each can belong to the same coset.
If x, y are in the same coset, then $x - y \in C$ & so by defn. $\omega(x-y) \geq d$. However wt(x), wt(y) $\leq t$ and So $wt(x-y) \leq 2t < d$. Contradiction

Recall q^{n-k} error patterns can be corrected.

what about detection

Claim — An (n,k,d) - linear code can detect upto $q^n - q^k$ errors.

Pf → For decoder to miss an error, the error pattern should convert one codeword into another. But $c_1 - c_2 \in C$ & so the error patterns correspond to the $q^k - 1$ non-zero codewords.
Total possible # of error patterns → $q^n - 1$.
& # error patterns which can be detected → $q^n - q^k$

Eg → ~~Pet~~ : (5,2) Code error correction

Pr (Error Corr) , Pr (Err Det)

So for an (n, k, d) - lin code, prob. of error not being corrected over a BSC is ~~given~~ given by prob. that error pattern does not match a coset leader. If $\alpha_0, \alpha_1, \dots \alpha_n$ denote the weight dist. of coset leaders (α_i = # coset leaders with weight i), then

$$Pr_c(\varepsilon) = 1 - \sum_{i=0}^{n} \alpha_i \, p^i (1-p)^{n-i}$$

On the other hand, for prob of undetected error,

$$P_d(\varepsilon) = \sum_{i=1}^{n} A_i \, p^i (1-p)^{n-i} \quad \text{where}$$

A_i is weight distribution of code C.

~~Coset decomp~~ for Standard array is of size $q^{n-k} \times q^k$ which can be very large depending on values of n, k. A better implementation is using ideas of syndromes.

Syndrome Decoding

$(1\times n-k) \quad 1\times n \quad n\times n-k$

For any received word r, syndrome is $S = r H^T$
where H is the parity check matrix of code C. Note that syndrome for all codewords is 0 ($(n-k)$-tuple)

Furthermore

Claim – All the q^k n-tuples of a coset have the same syndrome, while syndromes for different cosets are different.

Pf – Consider coset l with leader e_l. Then any tuple in this coset is of form $c_i + e_l$, & has syndrome

$$(c_i + e_l)\cdot H^T = c_i H^T + e_l H^T = e_l H^T.$$

So same for all elements of coset.

Now, say syndrome same for l^{th} & j^{th} cosets with $j < l$. Then

$$e_l H^T = e_j H^T \Rightarrow (e_l - e_j) H^T = 0$$

$$\Rightarrow e_l - e_j \in C \Rightarrow e_l \text{ lies in } j^{th} \text{ coset \& is not a leader}$$

Contradiction ■

So syndrome & cosets / coset-leaders have a one-to-one map & instead of entire standard array, we can just store a table mapping syndromes to coset leaders.

Decoding will follow:

1. Calculate syndrome of received word r, $S = r \cdot H^T$
2. Use table to find coset leader, say e_ℓ.
3. Decode as $\hat{c} = r - e_\ell$.

Eg For the $(\overset{n}{5}, \overset{k}{2}, \overset{d}{3})$ code with $G = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix}$

$$H = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{bmatrix}.$$

$$C = \left\{ \begin{matrix} 00000 \\ 10111 \\ 01101 \\ 11010 \end{matrix} \right\}$$

Syndrome Table

Coset leader	Syndrome
00000	000
00001	001
00010	010
00100	100
01000	101
10000	111
00011	011
00110	110

Say $c = 11010$ tx

err $= 01000$

So $r = 10010$

$S = rH^T = 101$. Get e_ℓ as 01000, so

Correctly recover $\hat{c} = 11010$.

Instead say

$c = 00000$

$e = 10100$

$r = 10100$

$S = rH^T = 011 \rightarrow e_\ell = 00011$

Wrong decoding as 10111