Week 9

1 Lemma

If $\mathscr{C}$ is a linear code, then

$$d_{min}(\mathscr{C}) = \min_{\underline{c} \neq 0} w_H(\underline{c})$$

where, Hamming weight, $w_H(\underline{c})$ = number of non zero positions in $\underline{c}$.

Proof:
By definition,

$$\begin{aligned} d_{min}(\mathscr{C}) &= \min(d_H(\underline{c}_1, \underline{c}_2)) \qquad \underline{c}_1, \underline{c}_2 \in \mathscr{C}; \underline{c}_1 \neq \underline{c}_2 \\ \min(d_H(\underline{c}_1, \underline{c}_2)) &= \min(w_H(\underline{c}_1 - \underline{c}_2) \qquad \underline{c}_1, \underline{c}_2 \in \mathscr{C}; \underline{c}_1 \neq \underline{c}_2 \\ &= \min(w_H(\underline{c})) \qquad \underline{c} \neq 0; \underline{c} \in \mathscr{C} \end{aligned}$$

Hence proved.

2 Examples

We know that every subspace of a vector space has a basis, i.e a set of linearly independent vectors from the subspace which span the subspace.

1. Suppose $\mathscr{C} = \mathbb{F}_2^n$,
   - Then any set of n linearly independent vectors from $\mathbb{F}_2^n$ will be a basis of $\mathscr{C}$.
   - In particular we can choose the standard basis, $\underline{c}_1 = (1, 0, \cdots, 0), \underline{c}_2 = (0, 1, 0, \cdots, 0), \cdots, \underline{c}_n = (0, \cdots, 0, 1)$
2. Suppose $\mathscr{C} = \{(0, \cdots, 0), (1, \cdots, 1)\}$
   - As this code is closed under addition, this is a valid linear code.
   - The basis for $\mathscr{C}$ will be $\{(1, \cdots, 1)\}$.
   - This code encodes 1 bit.
3. Suppose $B = \{g_1, \cdots, g_k\}$, $k < n$ are a set of linearly independent vectors in $\mathbb{F}_2^n$. What is linear code $\mathscr{C}$ for which B is a basis?
   - Set of all linear combinations of vectors in B, i.e.

$$\mathscr{C} = \text{span}(B) = \left\{ \sum_{i=1}^{k} \alpha_i g_i : \alpha_i \in \mathbb{F}_2 \right\}$$

- $|\mathscr{C}| = 2^k$, k is called the dimension of the subspace.
  Hence, $k = \log_2 |\mathscr{C}|$.
- Rate of the code $= k/n$.
- This code encodes k bits.
- Encoding is a linear operator, hence implementation is simple.

$$(\alpha_1, \alpha_2, \cdots, \alpha_k) \xrightarrow{\text{encoded}} \sum_{i=1}^{k} \alpha_i g_i$$

$$(\alpha_1, \alpha_2, \cdots, \alpha_k) \xrightarrow{\text{linear}} (\alpha_1, \alpha_2, \cdots, \alpha_k)_{1\times k} G_{k\times n}$$

where, $G_{k\times n} = \begin{pmatrix} g_1 \\ g_2 \\ \vdots \\ g_k \end{pmatrix}$

2.1 Generator matrix

Pick any collection of k linearly independent from $\mathbb{F}_2^n$ $\{g_1, \cdots, g_k\}$.

$G_{k\times n} = \begin{pmatrix} g_1 \\ g_2 \\ \vdots \\ g_k \end{pmatrix}$

Rowspace(G) = span(rows of G) = k dimensional subspace of $\mathbb{F}_2^n$

$$d_{min}(\mathscr{C}) = \min_{\underline{c} \neq 0} w_H(\underline{c})$$

Encoding is the operation of mapping $2^n R$ length messages to the n-length codewords in a unique manner. It is the mapping from k-length vectors over $\mathbb{F}_2$ to $\mathscr{C}$.

For linear codes, we can do this encoding as a linear mapping. Encoding operation for linear codes requires polynomial in n, unlike non-linear codes require exponential complexity.

2.2 Example

- Repetition code (eg. 2):

$$G = [1, 1, \cdots, 1]_{1\times n}$$

$$\text{Rowspace}(G) = \{(0, 0, \cdots, 0), (1, 1, \cdots, 1)\}$$

$$d_{min}(\mathscr{C}) = n \qquad dim(\mathscr{C}) = 1 \qquad R = \frac{k}{n} = \frac{1}{n}$$

How can we implement minimum distance decoding more efficiently?

$$\hat{\underline{c}} = argmin_{\underline{c} \in \mathscr{C}} d_H(y, \underline{c})$$

For n=5: Suppose $y = (1\,1\,1\,0\,0)$, then minimum distance decoder output is $\hat{c} = (1\,1\,1\,1\,1)$.

$$MDD(y) = \begin{cases} \underline{0} = (0, \cdots, 0) & w_H(y) < \frac{n}{2} \\ \underline{1} = (1, \cdots, 1) & w_H(y) > \frac{n}{2} \end{cases}$$

This is the majority decoding rule.

3 Binary Hamming code

This is a class of codes, we take up a particular example, let:

$$n = 2^r - 1 \qquad k = 2^r - 1 - r \qquad d = 3 \quad \forall \quad r \geq 3$$

$$\text{if } r = 3 \Rightarrow \qquad n = 7, k = 4, d = 3$$

$$G_{4\times 7} = \begin{bmatrix} I_4 : \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \end{bmatrix}$$

(we are appending I_4 with 3 columns to the right)

Note: The 4 rows of G are linearly independent vectors of $\mathbb{F}_2^7$. Rank(G) = number of linearly independent vectors in rows or columns = 4.

$\mathscr{C}$=Rowspace(G) is a 4-dim linear code. Rate= 4/7.

$|\mathscr{C}| = 2^k = 2^4 = 16$

$d_{min}(\mathscr{C}) = 3$