IERG 6120 Coding Theory for Storage Systems	Lecture 2 - 13/09/2016

The Minimum Distance of a Code

Lecturer: Kenneth Shum *Scribe: Yulin Shao*

1 Hamming distance

- The Hamming distance between two strings of equal length, $d_H(\mathbf{u}, \mathbf{v})$, is defined as the number of positions at which the corresponding symbols are different.
- Triangular Inequality of Hamming distance:
$$d_H(\mathbf{u}, \mathbf{v}) \leq d_H(\mathbf{u}, \mathbf{w}) + d_H(\mathbf{v}, \mathbf{w}). \tag{1}$$
- The Hamming weight of a string, $wt_H(\mathbf{u})$, is the number of symbols that are different from the zero-symbol of the alphabet used, i.e.
$$wt_H(\mathbf{u}) = wt(\mathbf{u}) \triangleq \big|\{i : u_i \neq 0\}\big|. \tag{2}$$
It is thus equivalent to the Hamming distance from the all-zero string of the same length.
- When the alphabet is *binary*, given two binary string $\mathbf{u}, \mathbf{v}$ of equal length, we have
$$d_H(\mathbf{u}, \mathbf{v}) = wt(\mathbf{u}) + wt(\mathbf{v}) - 2(\mathbf{u} \cdot \mathbf{v}), \tag{3}$$
where '$\cdot$' denotes inner product, by treating the components of $\mathbf{u}$ and $\mathbf{v}$ as integer 0 and 1.

Example 1: Given $\mathbf{u} = (111110000)$ and $\mathbf{v} = (000111110)$, we have $wt(\mathbf{u}) = wt(\mathbf{v}) = 5$, $\mathbf{u} \cdot \mathbf{v} = 2$ and $d_H(\mathbf{u}, \mathbf{v}) = 5 + 5 - 4 = 6$.

2 The Minimum Distance of a Code

- The minimum distance of a code C is defined as the smallest Hamming distance between two distinct codewords in C. Specifically,
$$d(C) \triangleq \min \big\{ d_H(\mathbf{c}, \mathbf{c}') : \mathbf{c}, \mathbf{c}' \in C, \mathbf{c} \neq \mathbf{c}' \big\}. \tag{4}$$
- The notation $(n, M, d)_q$−code C indicates that the code C is a q-ary code with length n, size M and minimum distance d. For binary code, i.e. $q = 2$, we simply write (n, M, d)-code.

Example 2: The rows of the following matrix are the codewords of a $(11, 12, 6)_2$-code.

$$\begin{bmatrix} 00000000000 \\ 10100011101 \\ 11010001110 \\ 01101000111 \\ 10110100011 \\ 11011010001 \\ 11101101000 \\ 01110110100 \\ 00111011010 \\ 00011101101 \\ 10001110110 \\ 01000111011 \end{bmatrix}$$

Matlab program: Calculate the Hamming distance of a code C using Matlab.

Matlab codes can be found in `https://piazza.com/class/isgy6spmwwm3ba?cid=15`.

Theorem 1 (Error Correction)**.** *An $(n, M, d)_q$-code C can correct t errors if $d \geq 2t+1$.*

Proof Suppose codeword $\mathbf{c}$ is transmitted and there are t errors, for some integer t satisfying $2t+1 \leq d$. Denote the received sequence by $\mathbf{y}$. We have

$$d_H(\mathbf{c}, \mathbf{y}) = t. \tag{5}$$

The decoder we use is "nearest neighbor decoder":

$$\mathrm{Dec}(\mathbf{y}) = \arg\min_{\mathbf{u}\in C} d_H(\mathbf{u}, \mathbf{y}). \tag{6}$$

Suppose that there is decoding error, say codeword $\mathbf{c}'$ is decoded erroneously, i.e., $\mathbf{c}' \neq \mathbf{c}$, and

$$d_H(\mathbf{c}', \mathbf{y}) = \min_{\mathbf{u}\in C} d_H(\mathbf{u}, \mathbf{y}). \tag{7}$$

Since $d_H(\mathbf{c}, \mathbf{y}) = t$, we must have $d_H(\mathbf{c}', \mathbf{y}) \leq t$. Then, we get

$$\begin{aligned} d &\leq d_H(\mathbf{c}, \mathbf{c}') && (8)\\ &\leq d_H(\mathbf{c}, \mathbf{y}) + d_H(\mathbf{c}', \mathbf{y}) && (9)\\ &\leq t + t = 2t, && (10) \end{aligned}$$

where (8) follows since $\mathbf{c} \neq \mathbf{c}'$ and the minimum distance of code C is d; (9) follows since Hamming distance satisfies Triangular Inequality.

Notice that (10) is a contradiction to our assumption that $d \geq 2t+1$. Thus, given $d \geq 2t+1$, any t errors can be corrected. □

Theorem 2 (Error Detection)**.** *An $(n, M, d)_q$-code C can detect any s errors if $s \leq d-1$.*

Proof Suppose that a codeword $\mathbf{c} \in C$ is transmitted, s errors occur, and $\mathbf{y}$ is received. The decoder we use is

$$\mathrm{Dec}(\mathbf{y}) = \begin{cases} \mathbf{c}' & \exists\, \mathbf{c}' \in C, \text{ s.t. } \mathbf{c}' = \mathbf{y}; \\ \text{error} & \text{otherwise.} \end{cases} \tag{11}$$

If there is an undetectable error, then we have $\mathbf{c}' = \mathbf{y}$ and $\mathbf{c}' \neq \mathbf{c}$. That is

$$s = d_H(\mathbf{c}, \mathbf{y}) = d_H(\mathbf{c}, \mathbf{c}') \geq d. \tag{12}$$

Thus, the decode could detect any $d-1$ errors. □

Theorem 3 (Erasure Correction)**.** *An $(n, M, d)_q$-code C can recover r erasures if $r \leq d-1$.*

Proof Notation: For $\mathbf{J} \subseteq \{1, 2, ..., n\}$, let $\mathbf{u_J} \triangleq (u_j, j \in \mathbf{J})$

Example: If $\mathbf{y} = (0, 1, 1, 1, 0, 0, 1)$ and $\mathbf{J} = \{1, 2, 5, 6, 7\}$, then $\mathbf{u_J} = (0, 1, 0, 0, 1)$.

Suppose codeword $\mathbf{c}$ is transmitted, r erasures occur and $\mathbf{y}$ is received. Let $\mathbf{J}$ be the set of indices of the unerased symbol. We note that $|\mathbf{J}| = n - r$.

The decoder we use is

$$\mathrm{Dec}(\mathbf{y}) = \begin{cases} \mathbf{c}' & \exists \text{ a unique } \mathbf{c}' \in C, \text{ s.t. } \mathbf{c}'_{\mathbf{J}} = \mathbf{y}_{\mathbf{J}}; \\ \text{error} & \text{otherwise.} \end{cases} \tag{13}$$

If there is decoding error, then the decoder's output $\mathrm{Dec}(\mathbf{y}) = \mathbf{c}'$ is a codeword different from the transmitted codeword $\mathbf{c}$, satisfying $\mathbf{c}'_{\mathbf{J}} = \mathbf{c}_{\mathbf{J}}$. Since the components of codewords $\mathbf{c}$ and $\mathbf{c}'$ with indices in $\mathbf{J}$ are the same, we get

$$d_H(\mathbf{c}, \mathbf{c}') \leq n - |\mathbf{J}| = n - (n - r) = r. \tag{14}$$

This contradicts the assumption that the Hamming distance between two distinct codewords is at least d. Thus, the decoder can recover any $d - 1$ erasures. □

Example 3: Using the $(11, 12, 6)$-code in Example 2, we can

- correct 2 errors,
- detect 5 errors, or
- correct 5 erasures.

Exercises: Show that the converses of the above three theorems hold.

1. If we can design an error-correcting decoder for an $(n, M, d)_q$-code that can correct any t errors, then $d \geq 2t + 1$.
2. If we have an error-detecting decoder for an $(n, M, d)_q$-code that can detect any s errors, then $d \geq s + 1$.
3. If we have an erasure-correcting decoder for an $(n, M, d)_q$-code that can correct any r erasures, then $d \geq r + 1$.

Theorems 1 to 3, together with the above exercises, imply that an $(n, M, d)_q$ code can

1. correct up to $\lfloor (d-1)/2 \rfloor$ errors, but there exists an uncorrectable error pattern with $\lfloor (d-1)/2 \rfloor + 1$ errors.
2. detect up to $d - 1$ errors, but there exists an undetectable error pattern with d errors.
3. recover up to $d - 1$ erasures, but there exists an unrecoverable erasure pattern consisting of d erasures.