COMPSCI 690T Coding Theory and Applications Jan 30, 2017

Lecture 3

Instructor: Arya Mazumdar *Scribe: Names Redacted*

1 Linear Codes (Recall)

Linear codes are formed by linear mapping of a k-bit vector to an n-bit vector (codewords).

$$\phi : \{0,1\}^k \to \{0,1\}^n$$

Codes are the sets $C = \{\phi(x) : x \in \{0,1\}^k\}$. For a k length vector $x \in \{0,1\}^k$, the corresponding codeword is calculated by $x^T G$, where $G \in \{0,1\}^{n\times k}$ is the generator matrix . For general, including nonlinear codes, $k = \lfloor \log_2 |C| \rfloor$.

$$x : k \times 1 \text{ vextor}$$
$$G : k \times n \text{ matrix}$$
$$x^T G : 1 \times n \text{ vector}$$

Review: For a linear code $C = \{x^T G : x \in \{0,1\}^k\}$, we have that: $\forall y_1, y_2 \in C \Rightarrow y_1 + y_2 \in C$
A linear code of size k, length n and minimum distance d is represented as $[n,k,d]$-code.

Let G be the generator matrix of a $[6,3,3]$-code:

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

The generated code words are

$$000 \to 000000$$
$$001 \to 001111$$
$$010 \to 010110$$
$$011 \to 011001$$
$$100 \to 100101$$
$$101 \to 101010$$
$$110 \to 110011$$
$$111 \to 111100$$

2 Dual Codes and Parity Check Matrix

Each linear code is the linear subspace spanned by G, which means rows of G are bases of this subspace. That's why we call G a generator matrix. Each linear subspace has a dual subspace. It(dual subspace) is the set of the vectors that are orthogonal to the vectors of linear codes (spanned by the row space of the generator matrix). The dual code of C is

$$C^\perp = \{z | \langle x, z \rangle = 0, \forall x \in C\}$$

$\langle x, z \rangle$ is the inner product of two vectors. Say $x = [x_1, x_2, \ldots, x_n]^T$ and $z = [z_1, z_2, \ldots, z_n]^T$ then $\langle x, z \rangle = x^T z = \sum_{i=1}^n x_i z_i$. (Each operation is modulo 2).

Vectors in the dual space are called dual code $C^\perp$. The linear codes (primal) and dual codes may have same codewords. It is possible that an element of $C^\perp$ is in C. In particular, it is possible to have

$C = C^{\perp}$. In this case we call code C a self-dual code. If the original subspace has dimension k, then the dimension of the dual subspace is $(n-k)$.
We call the generator matrix of the dual code, parity-check matrix. The minimum distance of the dual code of a code with minimum distance d is denoted by by $d^{\perp}$. In general, the relation between d and $d^{\perp}$ is unclear.

For example for the given parity check matrix

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

For any codeword $x \in C$ the equation $Hx = 0$ holds. In other words, any codewords in C is valid a solution of the following set of equation

$$\begin{aligned} x_1 + x_2 + x_3 + x_4 &= 0 \\ x_2 + x_3 + x_5 &= 0 \\ x_1 + x_3 + x_6 &= 0 \end{aligned}$$

.

$\forall x, y \in C : Hx = 0,\ Hy = 0 \to H(x+y) = 0$

2.1 Single Parity Check Codes

For the single parity check code with generator matrix $G = [I_{(n-1)\times(n-1)}, \underline{1}]$, the corresponding parity check matrix is $H = [1, 1, \ldots, 1]$ which is the generator matrix for repetition code.

2.2 A Claim

The vectors $x^{(1)}, x^{(2)}, \ldots, x^{(l)}$ are linearly independent if for all $a_1, a_2, \ldots, a_l \in \{0, 1\}$, $\sum_{i=1}^{l} a_i x^{(i)} \neq 0$.

Theorem 1 *A linear code has distance d, if and only if any $(d-1)$ columns of the parity check matrix is linearly independent and there exist d columns that are linearly dependent.*

Proof $(\Rightarrow)$ Consider $[n, k, d]-$ code C. By contradiction, there are $d-1$ linearly dependent columns in the parity check matrix H. Let $h^{(1)}, h^{(2)}, \ldots h^{(d-1)}$ be $d-1$ linearly dependent columns in H. Consider a binary vector x with ones in entries corresponding to columns $h^{(1)}, h^{(2)}, \ldots h^{(d-1)}$. Since $h^{(1)}, h^{(2)}, \ldots h^{(d-1)}$ are linearly dependent, we have that $Hx = 0$. We have that $w(x) = d-1$, which contradicts the fact that minimum distance of code C is d. So any $(d-1)$ column are linearly independent. The distance of the code is d. So, there exist a code with d ones in the vector x . Now from $Hx = 0$, we get the linear combination of the corresponding d columns of the parity check matrix H equal to 0. So there exists d linearly dependent columns.
$(\Leftarrow)$ Following a similar argument, we can show that no code word can have weight less than d. Since other wise there exist $d-1$ linearly dependent columns corresponding to the 1 in this codeword. ■

Note: As a result of this claim, if we find a parity-check matrix in which every $d-1$ columns are independent, then we have a code of distance d.

2.3 Hamming Code

Hamming code has a minimum distance of $d = 3$. As a result any 2 column of the parity check matrix are different.

$$H^{(1)} = [1,0] \qquad\qquad n = 2$$

$$H^{(2)} = \begin{bmatrix}0 & 1 & 1\\ 1 & 0 & 1\end{bmatrix} \qquad\qquad \text{C}= \{000, 111\}$$

$$H^{(3)} = \begin{bmatrix}0 & 0 & 0 & 1 & 1 & 1 & 1\\ 0 & 1 & 1 & 0 & 0 & 1 & 1\\ 1 & 0 & 1 & 0 & 1 & 0 & 1\end{bmatrix} \qquad\qquad \begin{gathered} n = 7\\ k = 4 \rightarrow 16 \ \mathit{codewords}\\ d = 3\\ R = \tfrac{4}{7}\end{gathered}$$

In general $H^{(m)}$ columns are all binary vector of length $m \Rightarrow n = 2^m - 1$.

$$m = n - k \Rightarrow k = 2^m - m - 1 \approx \log n$$

We have a $[2^m - 1, 2^m - m - 1, 3]$-code .

2.4 Decoding

Upon receiving the vector $y = x + e_i$ where e_i is the n-length error vector with $(n-1)$ zeros and one in i^{th} place where the bit got flipped.

$$Hy = H(x + e_i) = Hx + He_i = 0 + He_i = h_i \ (i^{th} \ \mathit{column\ of\ H)}$$

$Hx = 0$, because x is orthogonal to the column space of H. A binary to decimal conversion of the binary vector h_i will produce the error position.

2.5 Optimality

The maximum possible size (number of codewords) of a code with length n and minimum distance d is denoted by $A(n, d)$.By singleton bound we have $A(n, d) \leq 2^{n-d+1}$, or equivalently $\log_2 A(n, d) \leq n-d+1$. As we can see singleton bound does not imply that hamming code is optimal.

Next, we are going to study the sphere packing bound which implies optimality of Hamming code.