

Lecture 4: Hamming code and Hamming bound

September 5, 2007

Lecturer: Atri Rudra

Scribe: Kanke Gao & Atri Rudra

In the last couple of lectures, we have seen that the repetition code $C_{3,rep}$, which has distance $d = 3$, can correct ≤ 1 error. On the other hand the parity code C_{\oplus} , which has distance $d = 2$, can detect ≤ 1 error, but can not correct 1 error. In the last lecture, we extended these observations to the general case: code with distance d can correct $\lfloor \frac{d-1}{2} \rfloor$ errors and can detect $d - 1$ errors. Thus, the fundamental tradeoff that we are interested in (the amount of redundancy in the code vs. the number of error that it can correct) is equivalent to the one between rate and distance of the code (for worst-case errors).

1 Hamming Code

We have seen that the repetition code $C_{3,rep}$ has distance 3 and rate $1/3$. A natural question to ask is whether we can have distance 3 with a larger rate. With this motivation, we will now consider the so called *Hamming code* (named after its inventor, Richard Hamming), which we will denote by C_H . Given a message $(x_1, x_2, x_3, x_4) \in \{0, 1\}^4$, its corresponding codeword is given by

$$C_H(x_1, x_2, x_3, x_4) = (x_1, x_2, x_3, x_4, x_2 \oplus x_3 \oplus x_4, x_1 \oplus x_3 \oplus x_4, x_1 \oplus x_2 \oplus x_4),$$

where the \oplus denotes the EXOR operator. It is easy to check that this code has the following parameters:

$$C_H : q = 2, k = 4, n = 7, R = 4/7.$$

Before we move onto determining the distance of C_H , we will need another definition.

Definition 1.1 (Hamming Weight). *Let $q \geq 2$. Given any vector $\mathbf{v} \in \{0, 1, 2, \dots, q - 1\}^n$, its Hamming weight, denoted by $wt(\mathbf{v})$ is the number of non-zero symbols in \mathbf{v} .*

We now look at the distance of C_H .

Proposition 1.2. C_H has a distance 3.

Proof. We will prove the claimed property via two properties of C_H :

$$\forall c \in C_H, c \neq \mathbf{0} : wt(c) \geq 3, \tag{1}$$

and

$$\min_{c \in C_H, c \neq \mathbf{0}} wt(c) = \min_{c_1 \neq c_2 \in C_H} \Delta(c_1, c_2) \tag{2}$$

We begin with the proof of (1), which follows from a case analysis on the Hamming weight of the message bits. Below we will use $\mathbf{x} = (x_1, x_2, x_3, x_4)$ to denote the message vector.

- **Case 1:** If $wt(\mathbf{x}) = 1$ then at least two parity check bits in $(x_2 \oplus x_3 \oplus x_4, x_1 \oplus x_2 \oplus x_4, x_1 \oplus x_3 \oplus x_4)$ are 1. So in this case, $wt(C_H(\mathbf{x})) \geq 3$.
- **Case 2:** If $wt(\mathbf{x}) = 2$ then at least one parity check bit in $(x_2 \oplus x_3 \oplus x_4, x_1 \oplus x_2 \oplus x_4, x_1 \oplus x_3 \oplus x_4)$ is 1. So in this case, $wt(C_H(\mathbf{x})) \geq 3$.
- **Case 3:** If $wt(\mathbf{x}) \geq 3$ then obviously $wt(C_H(\mathbf{x})) \geq 3$.

Thus, we can conclude that $\min_{c \in C_H, c \neq \mathbf{0}} wt(c) \geq 3$

We now turn to the proof of (2). For the rest of the proof, let $\mathbf{x} = (x_1, x_2, x_3, x_4)$ and $\mathbf{y} = (y_1, y_2, y_3, y_4)$ denote two distinct messages. Using associativity and commutativity of the \oplus operator, we obtain that $C_H(\mathbf{x}) + C_H(\mathbf{y}) = C_H(\mathbf{x} + \mathbf{y})$, where the “+” operator is just the bit-wise \oplus of the operand vectors. Further, it is easy to verify that for two vectors $\mathbf{u}, \mathbf{v} \in \{0, 1\}^n$, $\Delta(\mathbf{u}, \mathbf{v}) = wt(\mathbf{u} + \mathbf{v})$. Thus, we have

$$\begin{aligned} \min_{\mathbf{x} \neq \mathbf{y} \in \{0,1\}^4} \Delta(C_H(\mathbf{x}), C_H(\mathbf{y})) &= \min_{\mathbf{x} \neq \mathbf{y} \in \{0,1\}^4} wt(C_H(\mathbf{x} + \mathbf{y})) \\ &= \min_{\mathbf{x} \neq \mathbf{0} \in \{0,1\}^4} wt(C_H(\mathbf{x})), \end{aligned}$$

where the second equality follows from the observation that $\{\mathbf{x} + \mathbf{y} | \mathbf{x} \neq \mathbf{y} \in \{0, 1\}^n\} = \{\mathbf{x} \in \{0, 1\}^n | \mathbf{x} \neq \mathbf{0}\}$. The proof of (2) is complete by recalling that $wt(C_H(\mathbf{x})) = 0$ if and only if $\mathbf{x} = \mathbf{0}$ and $C_H = \{C_H(\mathbf{x}) | \mathbf{x} \in \{0, 1\}^4\}$. Combining (1) and (2), we conclude that C_H has a distance 3. \square

In fact the second part of the proof could also have been shown in the following manner. It can be verified easily that the Hamming code is the set $\{\mathbf{x} \cdot G_H | \mathbf{x} \in \{0, 1\}^4\}$, where G_H is the following matrix.

$$G_H = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

In fact, any binary code (of dimension k and block length n) that is generated¹ by a $k \times n$ matrix is called a *binary linear code*. This implies the following simple fact.

Lemma 1.3. *For any binary linear code C and any two messages \mathbf{x} and \mathbf{y} , $C(\mathbf{x}) + C(\mathbf{y}) = C(\mathbf{x} + \mathbf{y})$.*

Proof. For any binary linear code, we have a generator matrix G . The following sequence of equalities (which follow from the distributivity and associativity properties of \oplus and AND operators) proves the lemma.

$$\begin{aligned} C(\mathbf{x}) + C(\mathbf{y}) &= \mathbf{x} \cdot G + \mathbf{y} \cdot G \\ &= (\mathbf{x} + \mathbf{y}) \cdot G \\ &= C(\mathbf{x} + \mathbf{y}) \end{aligned}$$

¹That is, $C = \{\mathbf{x} \cdot G | \mathbf{x} \in \{0, 1\}^k\}$, where addition is the \oplus operation and multiplication is the AND operation.

□

The above lemma along with the arguments used to prove (2) in the proof of Proposition 1.2 imply the following result.

Proposition 1.4. *For any binary linear code, minimum distance is equal to minimum Hamming weight of any non-zero codeword.*

Thus, we have seen that C_H has distance $d = 3$ and rate $R = \frac{4}{7}$ while $C_{3,rep}$ has distance $d = 3$ and rate $R = \frac{1}{3}$. Thus, the Hamming code is strictly better than the repetition code (in terms of the tradeoff between rate and distance). The next natural question is can we have a distance 3 code with a rate higher than C_H . We address this question in the next section.

2 Hamming Bound

We now switch gears to present our first tradeoff between redundancy (in the form of dimension of a code) and its error-correction capability (in form of its distance). In particular, we will first prove a special case of the so called Hamming bound for a distance of 3.

We begin with another definition.

Definition 2.1 (Hamming Ball). *For any vector $\mathbf{x} \in [q]^n$,*

$$B(\mathbf{x}, e) = \{\mathbf{y} \in [q]^n \mid \Delta(\mathbf{x}, \mathbf{y}) \leq e\}.$$

Next we prove an upper bound on the dimension of *every* code with distance 3.

Theorem 2.2. *Every binary code with block length n , dimension k , distance $d = 3$ satisfies*

$$k \leq n - \log_2(n + 1).$$

Proof. Given any two codewords, $c_1 \neq c_2 \in C$, the following is true (as C has distance² 3):

$$B(c_1, 1) \cap B(c_2, 1) = \emptyset. \tag{3}$$

See Figure 1 for an illustration. Note that for all $\mathbf{x} \in \{0, 1\}^n$,

$$|B(\mathbf{x}, 1)| = n + 1. \tag{4}$$

Now consider the union of all Hamming balls centered around some codeword. Obviously their union is a subset of $\{0, 1\}^n$. In other words,

$$\left| \bigcup_{c \in C} B(c, 1) \right| \leq 2^n. \tag{5}$$

²Assume that $\mathbf{y} \in B(c_1, e) \cap B(c_2, e)$, that is $\Delta(\mathbf{y}, c_1) \leq e$ and $\Delta(\mathbf{y}, c_2) \leq e$. Thus, by the triangle inequality, $\Delta(c_1, c_2) \leq 2e \leq d - 1$, which is a contradiction.

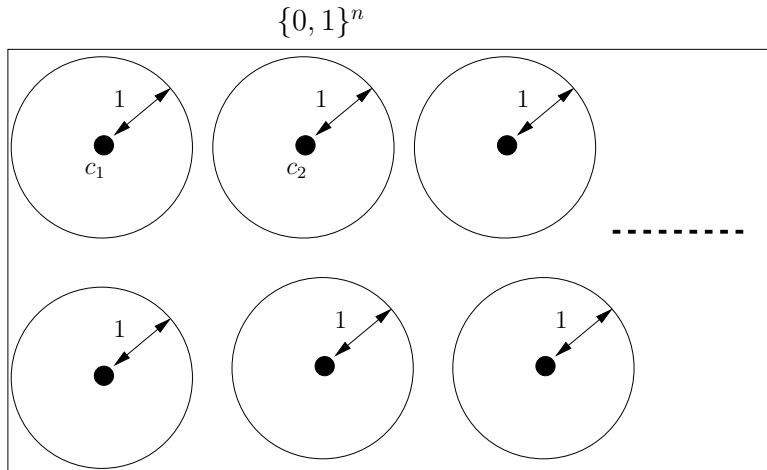


Figure 1: Hamming balls of radius 1 are disjoint.

As (3) holds for every pair of distinct codewords,

$$\begin{aligned} \left| \bigcup_{c \in C} B(c, 1) \right| &= \sum_{c \in C} |B(c, 1)| \\ &= 2^k \cdot (n + 1), \end{aligned} \tag{6}$$

where (6) follows from (4) and the fact that C has dimension k . Combining (6) and (5) and taking \log_2 of both sides we will get the desired bound:

$$k \leq n - \log_2(n + 1).$$

□

Thus, the Hamming bound implies that C_H has the largest possible dimension for any binary code of block length 7 and distance 3.