Lecture 2: key definitions and results

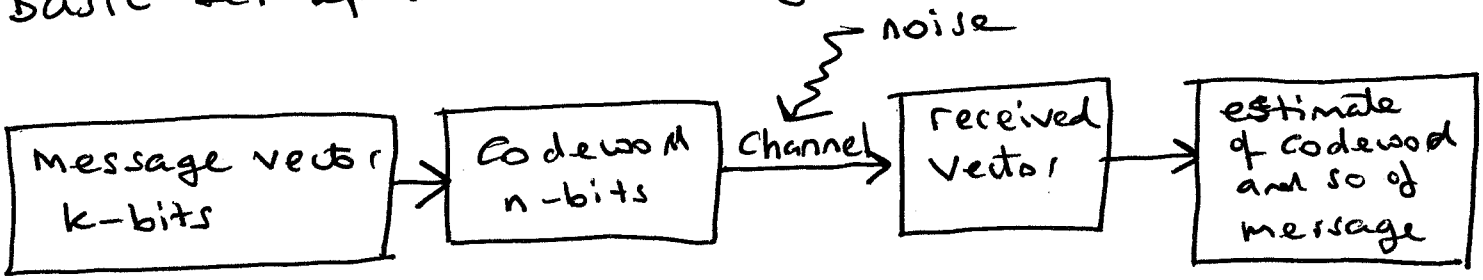
Any non-empty subset $C \subseteq \mathbb{Z}_2^n$ is called a (binary) code.

In most of our examples, $|C| = 2^r$ for some r. The number $\frac{r}{n}$ is called the rate of the code. Here, r will be the number of message bits and n the length of the codeword.

Clearly $0 < \frac{r}{n} \leq 1$.

> All other things being equal a higher rate code is preferable to a lower rate code - it is more efficient

Basic set-up for error correcting codes



Remark Errors can effect the extra bits added - not just the message bits!

We shall now define the concepts needed that will enable us to decode received vectors.

Let $\underline{x}, \underline{y} \in \mathbb{Z}_2^n$. The Hamming distance between $\underline{x}$ and $\underline{y}$, $d(\underline{x}, \underline{y})$, is the no. of places where $\underline{x}$ and $\underline{y}$ differ.

Example $00111, 11001 \in \mathbb{Z}_2^5$ and

$d(00111, 11001) = 4$

Lemma 1

(i) $d(\underline{x}, \underline{y}) = 0 \iff \underline{x} = \underline{y}$.

(ii) $d(\underline{x}, \underline{y}) = d(\underline{y}, \underline{x})$ for all $\underline{x}, \underline{y} \in \mathbb{Z}_2^n$.

(iii) Triangle inequality

$$d(\underline{x}, \underline{y}) \leq d(\underline{x}, \underline{z}) + d(\underline{z}, \underline{y})$$

for all $\underline{x}, \underline{y}, \underline{z} \in \mathbb{Z}_2^n$.

Remark Functions satisfying these three conditions are called metrics.

The proofs of (i) and (ii) are immediate.

The proof of (iii) follows from the observation that $d(\underline{x}, \underline{y})$ is the minimum no. of bits needed to change $\underline{x}$ to $\underline{y}$.

The following is the idea we shall apply in order to use codes to correct errors.

Nearest neighbour decoding principle

Let $C \subseteq \mathbb{Z}_2^n$ be a code.

Let $\underline{x} \in C$ be the transmitted codeword.

Let $\underline{y}$ be the received vector.

Decode $\underline{y}$ to $\underline{x}'$ where

(1) $\underline{x}' \in C$

(2) $d(\underline{y}, \underline{x}') \leq d(\underline{y}, \underline{x}'')$ where $\underline{x}'' \in C$.

i.e. "decode to the nearest codeword".

In a well designed code, we expect $\underline{x}' = \underline{x}$.

The error correcting capabilities of a code depend on a parameter that we define now.

Let $C \subseteq \mathbb{Z}_2^n$ be a code.

The minimum distance d of C is the no.

$$\min \{ d(\underline{x}, \underline{y}) : \underline{x}, \underline{y} \in C, \underline{x} \neq \underline{y} \}$$

Theorem 2 Let $C \subseteq \mathbb{Z}_2^n$ be a code with min. distance d. Then C can correct at most

$$t = \left\lfloor \frac{d-1}{2} \right\rfloor \quad \left(t \leq \frac{d-1}{2}\right)$$

errors.

Proof Let $\underline{x} \in C$ be a codeword. Let $\underline{y}$ be the received vector. We suppose that $d(\underline{x}, \underline{y}) \leq t$ i.e. no more than t errors have occurred.

Let $\underline{x}' \in C$, $\underline{x}' \neq \underline{x}$, $d(\underline{x}', \underline{y}) \leq t$ — (*)

Then

$$d(\underline{x}, \underline{x}') \leq d(\underline{x}, \underline{y}) + d(\underline{y}, \underline{x}')$$
$$\leq t + t = 2t \leq d-1 < d$$

But this contradicts the fact that the min. distance is d.

It follows that (*) cannot happen.

Thus if no more than t errors have occurred we shall correctly decode $\underline{y}$ to $\underline{x}$. ■

The goal of the theory of error-correcting codes is to design "good" codes. What do we mean by saying that a code $C \subseteq \mathbb{Z}_2^n$ is good?

(i) $|C|$ should be as large as possible so that we can encode many messages.

(ii) The min. distance of C should be as large as possible so that we can correct as many errors as possible.

But goals (i) and (ii) are clearly pulling in different directions — intuitively, more code words means a greater density of codewords means min. distances become smaller.
Thus designing good codes is an optimization problem.

To help us design good codes, we shall need to make more assumptions about them – at the moment they are merely subsets.

A code $C \subseteq \mathbb{Z}_2^n$ is said to be <u>linear</u> if it is a subspace of the vector space $\mathbb{Z}_2^n$.
This translates into the following two conditions in the case of binary codes:

(i) $\underline{0} \in C$.

(ii) $\underline{u}, \underline{v} \in C \Rightarrow \underline{u} + \underline{v} \in C$.

In $\mathbb{Z}_2^n$, $-\underline{u} = \underline{u}$. It follows that C is a subgroup of the group $\mathbb{Z}_2^n$. Linear codes are sometimes called <u>group codes</u>.

Not all codes are linear. However, linear codes are much easier to work with so we shall now show.

Let $\underline{x} \in \mathbb{Z}_2^n$. We define the weight of $\underline{x}$, $w(\underline{x})$, to be the no. of non-zero entries in $\underline{x}$.

Lemma 3 If $\underline{x}, \underline{y} \in \mathbb{Z}_2^n$ then

$$w(\underline{x}-\underline{y}) = d(\underline{x}, \underline{y}).$$

Proof The vector $\underline{x}-\underline{y}$ has non-zero entries precisely in the places where $\underline{x}$ and $\underline{y}$ differ. ∎

Theorem 4 Let C be a linear code. Then the min. distance d of C is the smallest non-zero weight of all codewords in C.

Proof Let $d = d(\underline{x}, \underline{y})$ where $\underline{x}, \underline{y} \in C$, $\underline{x} \neq \underline{y}$. Since C is linear $\underline{x}+\underline{y} \in C$ and $w(\underline{x}+\underline{y}) = w(\underline{x}-\underline{y}) = d(\underline{x}, \underline{y}) = d$. Thus C contains a codeword of weight d. Observe that $w(\underline{z}) = d(\underline{z}, \underline{0})$. If $\underline{z} \in C$, $\underline{z} \neq \underline{0}$, $w(\underline{z}) < d$ then $d(\underline{z}, \underline{0}) < d$ which contradicts the fact that d is the min. distance. ∎

Thus once we have shown that a code is linear determining its error-correction capabilities is much less onerous than in the non-linear case. For the remaining lectures, all of our codes will be linear.