Week 8

1 Achievability in channel coding theorem for BSC

For any $\epsilon > 0$, there exists a sequence of codes $\mathscr{C}$(one for each x) with rate $R : 1 - H_2(p) - \epsilon$ and P(error)→ 0 as $x \to \infty$.

Proof: We will use a 'random' code. Entire argument is for a fixed value of n. We will assume n is large (so that number of flips is np).

We will pick a code $\mathscr{C}_n$ of rate $R = 1 - H_2(p) - \epsilon$. So we want $|\mathscr{C}_n| = 2^{nR}$. (such that nR is an integer)

1.1 Random code generation

- Pick each codeword in $\mathscr{C}_n$ from $\{0, 1\}^n$ uniformly at random. Then P(codeword is specific n-length sequence)= $\frac{1}{2}^n$
- Repeat this process 2^{nR} times.

We get,

$$|\mathscr{C}_n| = 2^{nR}$$

Now we want to prove,

$$P(\underline{\hat{c}} \neq \underline{c}) \leq 2^{-n\delta} \qquad \forall \qquad \underline{c} \in \mathscr{C}_n, \delta > 0$$

For getting a handle on the probability of error, we have to first define the decoding function (i.e. how is estimate $\underline{\hat{c}}$ calculated from a particular recieved vector).

Let the decoding function be denoted by,

$$D : \{0, 1\}^n \to \mathscr{C}_n$$

For any $y \in \{0, 1\}^n$,

$$D(y) := \text{argmax}_{\underline{c}' \in \mathscr{C}_n} P(y|\underline{c}')$$

D(y) is the estimate $\underline{\hat{c}}$ of the code when recieved vector is y. This is called the "Maximum likelihood decoding rule".

To show that $P(\underline{\hat{c}} \neq \underline{c})$ is small, we will show that $P(D(y) \neq \underline{c})$ is small where y is the random vector when $\underline{c}$ is transmitted.

For the above decoder, we want to show $P(\underline{\hat{c}} \neq \underline{c}) \leq 2^{-n\delta}$, $\delta > 0$.

For any specific $\underline{c}$, we want to find an upper limit for $P(\underline{\hat{c}} \neq \underline{c})$. This occurs when $\underline{c}$ is not the closest codeword to y.

By law of large numbers,

$$d_H(y, \underline{c}) \approx np$$

$$B(y,np) = \{\underline{\mathrm{x}} \in \{0,1\}^n : d_H(\underline{\mathrm{x}},\underline{\mathrm{y}}) \leq np\}$$

If there are other codewords within this ball B(y,np), the decoder can make an error. Else it will not make an error.

$$\begin{aligned} P(\hat{\underline{\mathrm{c}}} \neq \underline{\mathrm{c}}) &\leq P(\underline{\mathrm{c}}' \in B(y,np) : \hat{\underline{\mathrm{c}}} \neq \underline{\mathrm{c}}) \\ &\leq \frac{|B(y,np)|}{2^n} \\ &\leq \frac{\sum_{i=0}^{np}\binom{n}{i}}{2^n} = \frac{\binom{n}{np} + \sum_{i=0}^{np-1}\binom{n}{i}}{2^n} \\ &\approx \frac{2^{nH_2(p)}}{2^n} = 2^{-n(1-H_2(p))} \end{aligned}$$

As n grows large, this value will be dominated by the first term of value $\binom{n}{np}$.

$$\Rightarrow P(\hat{\underline{\mathrm{c}}} \neq \underline{\mathrm{c}}) \leq 2^{-n(1-H_2(p))} \tag{1}$$

(we assume p $<$ 0.5, otherwise we can change the channel to BSC(p') where p'= 1-p)

We would like to show this result (1) for all codewords rather than specific codewords. We want,

$$P\left(\bigcup_{\underline{\mathrm{c}} \in \mathscr{C}} (\hat{\underline{\mathrm{c}}} \neq \underline{\mathrm{c}})\right) \leq 2^{-n\delta} \qquad \delta > 0$$

This is known as the union bound.

1.2 Union bound

We know,

$$\begin{aligned} P\left(\bigcup_{\underline{\mathrm{c}} \in \mathscr{C}_n} (\hat{\underline{\mathrm{c}}} \neq \underline{\mathrm{c}})\right) &\leq \sum_{\underline{\mathrm{c}} \in \mathscr{C}_n} P(\hat{\underline{\mathrm{c}}} \neq \underline{\mathrm{c}}) \\ &\leq P\left(\bigcup_{\underline{\mathrm{c}} \in \mathscr{C}} (\hat{\underline{\mathrm{c}}} \neq \underline{\mathrm{c}})\right) \leq \sum_{\underline{\mathrm{c}} \in \mathscr{C}_n} 2^{-n(1-H_2(p))} \\ &\leq 2^{nR}2^{-n(1-H_2(p))} \\ &\leq 2^{-n(1-H_2(p)-R)} \\ &\leq 2^{-n\varepsilon} \end{aligned}$$

$$\Rightarrow P(\hat{\underline{\mathrm{c}}} \neq \underline{\mathrm{c}}) \leq 2^{-n\epsilon} \quad \forall \underline{\mathrm{c}} \in \mathscr{C}_n$$

Hence proved.

In practice using random codes and minimum distance/likelihood decoder (MDD/MLD) for BSC is very complex (complexity of decoder/encoder is very high). Hence, we use structured codes which have low encoding/decoding performance, mainly linear codes.

2 Linear codes

The random code construction is not really useful for implementation as:

1. we could end up with a bad code due to the random construction.

2. Encoding and decoding complexity is very large.

So we want codes which are good in rate and P(error) and also have reasonable encoding/decoding complexity. An important class of codes having above properties are linear codes.

We will look at some simple examples of liner codes for binary channel with worst-case/bounded error model. Construction of codes which are useful for implementation is dealt with in coding theory.

2.1 Worst case/bounded error model for binary channel

Let t, n be some integers such that t $\leq$ n.

We input some $x \in \{0,1\}^n$ to a binary channel and we get $y \in \{0,1\}^n$. $d_H(y,x) \leq t \Rightarrow$ There are atmost t positions where recieved vector y is different from transmitted vector x.

For this channel, we want to design a code $\mathscr{C} \subseteq \{0,1\}^n$ (such that all upto t errors are corrected).

2.1.1 Example

Now, suppose $= \{0,1\}^n$, t=1.

Let us construct a situation in which the decoder will surely make an error in decoding.

Suppose $\underline{c} = (1, \cdots, 1) \in \mathscr{C}$ was transmitted.

Suppose $y = (0, 1, \cdots, 1)$, $y \in$. (we will assume min Hamming distance decoder $\hat{\underline{c}} = argmin_{\mathscr{C}} d_H(y, \underline{c})$)

Hence iff $\underline{c} = y$,

$$min_{\underline{c} \in \mathscr{C}} d_H(y, \underline{c}) = 0$$

$\Rightarrow$ Decoding error has happened as $\hat{\underline{c}} \neq \underline{c}$ (estimate and transmitted are not same).

So correcting any $t \geq 1$ error requires us to pick proper subsets of $\{0,1\}^n$.

But we also want to pick large subsets of $\{0,1\}^n$ as the code because we want to maximize $R = \frac{\log |\mathscr{C}|}{n}$ bits/channel use.

But picking large $\mathscr{C}$, codewords are closer in Hamming distance, which means that it is more likely to create decoding errors.

2.2 Lemma

Let $\mathscr{C} \subseteq \{0,1\}^n$ be chosen. Define,

$$d_{min}(\mathscr{C}) = min_{c,c' \in \mathscr{C} \& c' \neq c} d_H(c, c')$$

$\mathscr{C}$ can be correct upto t errors if and only if $d_{min}(\mathscr{C}) \geq 2t+1$.

Proof:

If part: Given: $\mathscr{C}$ can correct any t errors.

To prove: $d_{min}(\mathscr{C}) \geq 2t+1$.

Given statement implies any for any c, c' $\in \mathscr{C}$.

$$B_t(c) = \text{Hamming ball of radius t} := \{x \in \{0,1\}^n : d_H(x,c) \leq t\}$$

Then,

$$B_t(c) \bigcap B_t(c') = \phi$$

$$\Rightarrow d_H(c,c') > 2t \qquad \forall \qquad c,c' \in \mathscr{C} \quad c \neq c'$$

This can be proved by contradiction.

2.3 Terminology

- Size of code = $|\mathscr{C}|$.
- Length of code (Block length) = n.

Lemma above relates the error correcting capability of the code with the minimum distance, minimum disctance calculation has nothing to do with the channel.

Suppose code has minimum distance of d, then it can be used on a channel for correcting upto $\lfloor \frac{d-1}{2} \rfloor$.

This says that code design can be theoretically done independent done of the channel and it's performance can be tested based on it's minimum distance.

2.4 Hamming Bound

Hamming bound is the upper bound on the size of code based on a given minimum distance.

Lemma: Let $\mathscr{C}$ be any code with $d_{min}(\mathscr{C}) = d$.

Then,

$$|\mathscr{C}| \leq \frac{2^n}{\sum_{i=0}^{t} \binom{n}{i}} \qquad t = \lfloor \frac{d-1}{2} \rfloor$$

Proof follows as we can pick atmost one codeword per ball.

2.5 Linear codes (over $\mathbb{F}_2$)

$$\mathbb{F}_2 \to (\{0,1\}, +, \cdot)$$

$$+ \to XOR$$

Definition: A linear code over $\mathbb{F}_2$ of length n is subset $\mathscr{C} \subseteq {\mathbb{F}_2}^n$ and also a subspace of the vector space ${\mathbb{F}_2}^n$.

$$\Rightarrow \forall a, b \in \mathbb{F}_2 \quad c_1, c_2 \in \mathscr{C}$$

$$ac_1 + bc_2 \in \mathscr{C}$$

Since only non-trivial values of a, b above are a=1 and b=1.

$\Leftrightarrow \mathscr{C}$ is a subspace of ${\mathbb{F}_2}^n$ iff $\forall c_1, c_2 \in \mathscr{C}$, we have $c_1 + c_2 \in \mathscr{C}$.