COMPSCI 690T Coding Theory and Applications Jan 23, 25 2017

Lecture 1+2

Instructor: Arya Mazumdar *Scribe: Arya Mazumdar*

1 Logistics

Please see the slides under Logistics.

2 Error-correction

To correct errors, one must introduce redundancy. Assume, k-bit sequences are being mapped to $n-k$ bit sequences. Therefore the redundancy of the code is $n-k$. The rate of the code is defined to be

$$R \equiv \frac{k}{n}.$$

Example 1: Consider the simple encoding scheme:

data	codeword
0	000
1	111

In this case $k = 1, n = 3$. Hence, rate $R = \frac{1}{3}$. Now to decode, a simple decoding scheme would be to map any a vector of length 3 to 0 if it has majority of 0's and to 1 if its has a majority of 1's (this is also the nearest neighbor decoding as explained below).

The set of the codewords of length n is collectively called the *code*. A code $\mathcal{C} \subseteq \{0,1\}^n$. In the above example, $\mathcal{C} = \{000, 111\}$.

To summarize, a code $\mathcal{C}$ is a subset of n-length strings, $\mathcal{C} \subset \{0,1\}^n$. The n-length strings are called codewords. To encode k bits each k-bit string is mapped to an n-bit string, via a bijective mapping.

To encode maximum amount of information we need to find a code as large as possible. Since the mapping from the message to codewords is not that important, it is not necessary that the size of the code be a power of 2. Although in reality $\lfloor \log_2 |\mathcal{C}| \rfloor$ bits can be encoded to $\mathcal{C}$, the, the rate of the code is defined to be,

$$R(\mathcal{C}) = \frac{\log_2 |\mathcal{C}|}{n}.$$

There are two types of error that can be introduced by a medium.

2.1 Adversarial error

In this case, an adversary, that has access to the codewords and the decoding algorithm, maliciously introduces errors such that two codewords are confused. Usually when a decent random model for errors is unavailable (such as a scratch in a Compact Disk), this adversarial noise model is used.

A code is called t-error correcting, if the decoding algorithm is successful even when an adversary flips t bits.

2.1.1 Hamming distance

The Hamming distance between two vectors $x, y \in \{0,1\}^n$ is defined to be the number of coordinates they do not match. In other words,

$$d(x,y) = w(x+y \mod 2)$$

where $x + y \mod 2$ is the coordinate-wise summation modulo 2, or Boolean XOR, and $w(x)$ is the number of nonzero entries in x (or ℓ_0 norm).

The *minimum distance* or simply *distance* of a code $\mathcal{C}$ is the minimum pairwise Hamming distance between any two different codewords:

$$d(\mathcal{C}) = \min_{x,y \in \mathcal{C}: x \neq y} d(x, y).$$

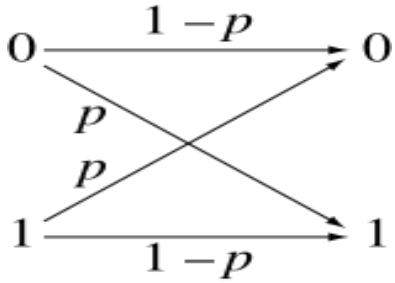
The *nearest neighbor* decoding, or *minimum distance* decoding of a code, refers to decoding to the nearest (in the sense of Hamming distance) codeword to the received noisy codeword.

Theorem 1 *Any t-error correcting code must have distance at least $2t + 1$.*

Note, the code of Example 1 has minimum distance 3, hence corrects any one error.

2.2 Random error

In this case, errors are introduced randomly and independently, with some probability. For example, take the popular model of *binary symmetric channel*, where each bit is flipped with probability p.



Let us see, how the code of Example 1 performs for this channel. An error can occur when there are more than two bit-flips. Thus the probability of error is,

$$P_e = \binom{3}{2} p^2(1-p) + p^3.$$

We have,

$$P_e < p \iff p < 1/2$$

Thus the probability of error is reduced if $p \in [0, 1/2)$. We can further reduce the probability of error by increasing the number of repeat bits. Let $P_e^{(n)}$ be the probability of error, when we repeat it n times.

Exercise: Find out the expression for $P_e^{(n)}$ and show that

$$\lim_{n \to \infty} P_e^{(n)} \to 0.$$

However, note that, in the above coding scheme, as $P_e^{(n)} \to 0$, the rate $R = \frac{1}{n} \to 0$ as well.

To everyone's surprise, Shannon in 1948 showed that there exists a coding scheme for which probability of error goes to zero as long as rate $R < 1 - h(p)$, where $h(p)$ is the binary entropy function $h(p) = -p \log_2 p - (1-p) \log_2 1 - p$. This is one of the earliest use if probabilistic methods.

2.2.1 Optimum decoding for BSC

The optimum decoding over a random channel is given by the maximum a-posteriori decoding or MAP decoding. By definition, MAP decoding that takes the received vector $y \in \{0,1\}^n$ as input, is following:

$$\arg\max_{c \in \mathcal{C}} P(x \text{ sent} \mid y).$$

Let us just manipulate this decoder to have,

$$\begin{aligned}\arg\max_{c \in \mathcal{C}} P(x \mid y) &= \arg\max_{c \in \mathcal{C}} P(y \mid x)\frac{P(x)}{P(y)} \\ &= \arg\max_{c \in \mathcal{C}} P(y \mid x),\end{aligned}$$

where we have assumed $P(x)$ is equal for all $x \in \mathcal{C}$ (all codewords are equally likely to be sent). This is called the Maximum Likelihood (ML) decoder. Further, since for a binary symmetric channel $P(y \mid x)$ depends on the number of bits in which x and y differ,

$$\begin{aligned}\arg\max_{c \in \mathcal{C}} P(y \mid x) &= \arg\max_{c \in \mathcal{C}} p^{d(x,y)}(1-p)^{n-d(x,y)} \\ &= \arg\max_{c \in \mathcal{C}} \left(\frac{p}{1-p}\right)^{d(x,y)} \\ &= \arg\min_{c \in \mathcal{C}} d(x,y),\end{aligned}$$

where the last equation follows since for $p < 1/2$, $p/(1-p) < 1$. Hence the optimal decoding is the nearest codeword decoding.

3 Erasures

There is another popular model of noise, called erasures. If you have an erasure, then the bit is not corrupted, it just goes missing. Erasure is a popular model for storage and networks. Just like in the case of errors, erasures can follow either an adversarial model or a random model.

3.1 Adversarial erasures

Theorem 2 *A code corrects $d-1$ erasures, if and only if it has distance at least d.*

Proof Since any two codewords are distance d apart, they can still be differentiated after $d-1$ erasures, as there will still be at least a coordinate where they differ. ■

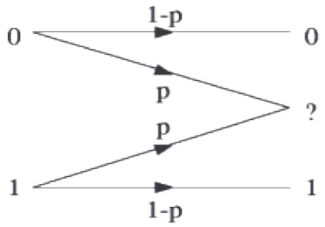
As a result, a t-error correcting code $\iff$ distance of the code is at least $2t+1$ $\iff$ the code corrects $2t$ erasures.

Surprisingly, this two-times-error correctability rule holds for random models as well.

3.2 Random erasures

A general random model for erasures is the *binary erasure channel,* (BEC) where each bit is erased randomly and independently with probability p.

Theorem 3 *If a code $\mathcal{C}$ achieves a small probability of error ϵ over a BSC with parameter p, the code will achieve probability of error at most ϵ over a BEC with parameter $2p$ (with a different decoder).*



Proof Consider the concatenation of two channels back to back below.

We concatenate BEC($2p$) with a channel with ternary input $\{0, 1, ?\}$ and binary output $\{0, 1\}$, such that with probability 1 the inputs $\{0, 1\}$ remain the same, and with uniform probability ? goes to $\{0, 1\}$. The overall channel forms a BSC with parameter p.

To verify this, see that the overall input-output relation is given by,

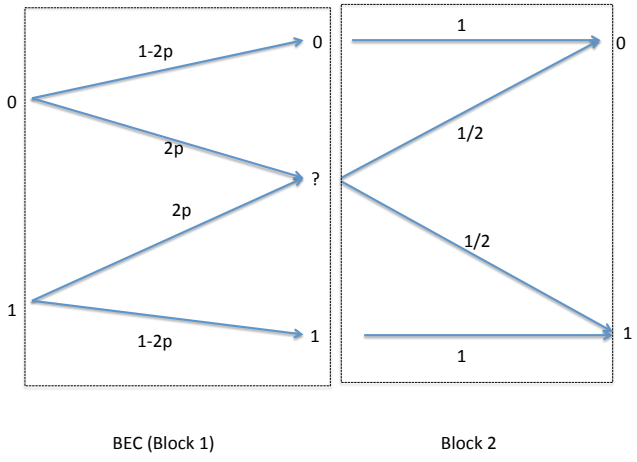
$$P(1 \text{ received} \mid 0 \text{ sent}) = 2p \cdot \frac{1}{2} = p;$$

and

$$P(0 \text{ received} \mid 0 \text{ sent}) = (1 - 2p) \cdot 1 + 2p \cdot \frac{1}{2} = 1 - p.$$

Therefore the overall channel is BSC with parameter p.

If a code can recover correctly from the end of the second block, it must be able to recover from the output of first block.



■

4 Singleton bound: bounds on codes

The larger the minimum distance of a code, greater the number of errors that can be corrected. Therefore, we want to choose point in $\{0, 1\}^n$ to minimize the minimum distance between the codewords.

The maximum possible size (number of codewords) of a code of length n and minimum distance d is denoted as $A(n, d)$. There is no good estimate of this quantity. For $0 \leq \delta \leq 1$, define the asymptotic optimal rate as,

$$R^*(\delta) \equiv \lim_{n\to\infty} \frac{\log_2 A(n, \delta n)}{n}.$$

It is not even known whether this limit exists or not. It is a central problem of combinatorics.

We will show one bound on $A(n, d)$ called the Singleton bound.

Theorem 4 (Singleton Bound)

$$A(n, d) \leq 2^{n-d+1}.$$

Proof Suppose $\mathcal{C}$ has length n and distance d. Remove the last $d-1$ coordinates from each codewords of $\mathcal{C}$ to obtain $\mathcal{C}'$, a code with length $n - (d - 1) = n - d + 1$. Since $\mathcal{C}$ had distance d, we must have $|\mathcal{C}| = \mathcal{C}'$. But $|\mathcal{C}'| \leq 2^{n-d+1}$. ■

5 Recap

Look up the definitions of a Field and a Vector Space. In $\{0, 1\}$, the summation operation here is always modulo 2, which will be used without mention. The linear operations below are over the finite field in context.

6 Linear codes

Introducing structure in a code allows us to find codewords with ease. Also, encoding becomes easy. Linear codes allow us to form codewords using less computational capacity. Linear codes are codes that form a subspace of the field $\{0, 1\}^n$. A subspace of a field is a subset which is closed under addition (mod two for $\{0, 1\}^n$) i.e.

$$x, y \in \mathcal{C} \implies x + y \in \mathcal{C}.$$

Since linear codes are a subspace, the codes can be represented by the basis of the subspace (in $\{0, 1\}^n$). Therefore forming a matrix $G_{k\times n}$ from the basis vectors we have,

$$\mathcal{C} = \{x^T G, x \in \{0, 1\}^k\}$$

Here G is called the generator matrix of $\mathcal{C}$. The size of the code is $M = 2^k$. Thus a linear code can be written in terms of its length n, dimension k, and minimum distance d as $[n, k, d]$.

An $[n, k, d]$ linear code is a subspace of a vector space and also an $(n, 2^k, d)$ code. It is given by a $k \times n$ *generator matrix*.

[6,3,3] code

Suppose, a generator matrix G is given by:

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

The encoding is given by just multiplying with the generator matrix:

$$\begin{aligned}
000 &\to 000000\\
001 &\to 001111\\
010 &\to 010110\\
011 &\to 011001\\
100 &\to 100101\\
101 &\to 101010\\
110 &\to 110011\\
111 &\to 111100
\end{aligned}$$

Note that, the parameters of the code ($[6, 3, 3]$) remain same if we change the order of labeling. In error-correcting codes it does not matter what the particular map between the messages and codewords is, the only geometry that matters is that of the codewords.

Our objective is to construct codes of arbitrary length that correct any given number of errors with simple encoding and decoding algorithms.

Repetition code

Generator matrix is given by:

$$G = [11 \dots 1].$$

Single parity check code/ even weight code

$$G = \begin{pmatrix}
1 & 1 & 0 & 0 & \cdot & \cdot & \cdot & 0\\
1 & 0 & 1 & 0 & \cdot & \cdot & \cdot & 0\\
1 & 0 & 0 & 1 & \cdot & \cdot & \cdot & 0\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot\\
1 & 0 & 0 & 0 & \cdot & \cdot & 0 & 1
\end{pmatrix}$$

Theorem 5 *The minimum distance of a linear code $\mathcal{C}$ is the minimum weight of any vector (excluding 0 codeword) in $\mathcal{C}$.*

Proof

$$\begin{aligned}
\min_{x,y\in\mathcal{C}:x\neq y} d(x,y) &= \min_{x,y\in\mathcal{C}:x\neq y} w(x+y)\\
&= \min_{z\in\mathcal{C}:z\neq 0} w(z),
\end{aligned}$$

since $x + y$ must also belong to $\mathcal{C}$. ■