channel ← a mathematical model for communication & storage systems

Input alphabet ← 𝒇
Output alphabet ← 𝝋

\[ X \in 𝒇^n \]
\[ Y \in 𝝋^n \]

Two types of channels
→ probabilistic
→ adversarial → later...

probabilistic channel

\[ P(\mathcal{Y} | \mathcal{X}) \rightarrow \text{probability distribution of output, given input.} \]

Example: Binary Symmetric Channel

\[ f = \{0,1\} \quad \text{(BSC)} \]
\[ 𝜏 = \{0,1\} \]

\[ m = n \]
\[ P(\mathcal{Y} | \mathcal{X}) = \left[ \frac{\gamma_i}{\tau_i} \right] \]

\[ = \prod_{i=1}^{n} P(\mathcal{Y} | \mathcal{X}) \]

\[ P(\mathcal{Y} | \mathcal{X}) = \begin{cases} 1 & \text{if } y = x \\ 0 & \text{otherwise} \end{cases} \]

For instance, \( y \neq x \)

\[ P(\mathcal{Y} = 00 | \mathcal{X} = 00) = (1-p)x(1-h) \]
\[ = (1-p)^2 \]

\[ P(\mathcal{Y} = 01 | \mathcal{X} = 00) = (1-p) 0 \]

Alternate view

\[ Y = X + Z \]

\[ Z \rightarrow \text{i.i.d components} \]

\[ P(z_1, p) \]

\[ \rightarrow (\mathcal{Z}) \]

\[ P(z_1, 1) = p \]

\[ P(z_1, 0) = 1-p \]

i.i.d. all operations are modulo 2

Independent assumption

\[ Y_i \text{ is independent of } \mathcal{X} \]

Given \( X \)

Memoryless channels
given \( x_t \)

\[
P_T( y_t = y_c | x_t = x_i, x_2 = x_2, \ldots) = P( y_t = y_c | x_t = x_i) = -P_{y|x}(y_t | x_t)
\]

**Binary erasure channel (BEC)**

\( F = \mathbb{Z}/2, \quad Q = \{0, 1\} \)

\[
P_{y|x} = \prod_{i=1}^{n} P_{y|x_i}(y_i | x_i)
\]

\[
P_{y|x}(y | x) = \begin{cases} 1-b & \text{if } y \neq x \\ b & \text{if } y = x \\ 0 & \text{if } y = \text{erasure} \end{cases}
\]

\![\text{Diagram of Erasure correcting code}]

\( \tilde{u} \in U \)

\( |U| = M \)

An \((n, M)\) error correcting code contains the following:

- A one-to-one mapping from \( U \) to \( F^n \), called encoder
- A one-to-one mapping from \( F^n \) to \( U \) (or from \( F^n \) to \( F^n \) ), called decoder.

Usually, we assume \( m = n \)

\[1/13/2014\]

Formally, an \((n, M)\) code contains a set

\( C \subset F^n \) where \(|C| = M\)

\( \text{encoder mapping from } U \rightarrow C \text{ is implicit} \)
For a given channel, and code $C$, a decoder is a mapping from $\mathbb{F}^m$ to $C$

$$D(x) = x_0,$$  where $x_0 \in C$

Decoding function

Hamming distance

- imposes some geometry on $\mathbb{F}^n$, $\mathbb{F}^m$, etc.

Given a set $F$ and an integer $n$, the Hamming distance between strings $x, y \in F^n$

$$d_H(x, y) = \left\lfloor \frac{1}{2} \left| x \cdot y \right| \right\rfloor$$

where $x = (x_1, \ldots, x_n)$

$$y = (y_1, \ldots, y_n)$$

Example

$$d_H(0010, 0010) = 2.$$  

Hamming weight

For any set $F$ with a reference element $\varnothing$

The Hamming weight of a vector $x \in F^n$ is

$$d_H(x, \varnothing)$$

Rate & min. distance of a code

Rate of a $(n, M)$ code $C$ over alphabet $F$

$$R = \frac{\log M}{n}$$

$$M = |C| \leq |F|^n \Rightarrow \log M \leq n$$

$$\Rightarrow R \leq 1$$

Min. distance of a code $C$ is

$$d_{\min} = \min_{x, y \in C} d_H(x, y)$$

$(n, m, d)$ code has block length $n$ and min. distance $d$.  

Examples
Repetition code over \( F = \{ 0, 1 \} \)

\[ M = 2 \]

arbitrary \( n \).

\( C = \{ 0000 \ldots 0, 1111 \ldots 1 \} \)

Rate \( = \frac{1}{n} \)

\[ \to \frac{n}{n} \text{ as } n \to \infty \]

Min. distance \( = n \)

**Single parity code over \( F = \{ 0, 1 \} \)**

\( M = 2^{n-1} \)

arbitrary \( n \).

\( U = \{ 011 \ldots 1 \}^{n-1} \)

\( \mathcal{W} = \{ u_1, u_2, \ldots, u_n \} \to (u_1, u_2, \ldots, u_n, u_1 \oplus u_2 \oplus \ldots \oplus u_n) \)

Rate \( = \frac{n-1}{n} = 1 - \frac{1}{n} \)

\[ \to 1 \text{ as } n \to \infty \]

**Example**

\( n = 3 \).

\( C = \{ 000, 011, 101, 110 \} \)

Min. distance \( = 2 \)

In fact, in general, for a single parity check code,

\( d_{\text{min}} = 2 \) [does not depend on \( n \)]

\( 00 \ldots 01 \) \{ does not depend \}

\( \frac{1}{n} \)

distances \( = 2 \)

Note that for a single parity code

\( x \in C \) if and only if

\[ x_1 + x_2 + \ldots + x_n \equiv 0 \text{ (mod n) } \]

called a parity check equation

**Decoder for probabilistic channels**

For a channel \( P(y|x) \), the probability of error of a codeword \( x \in C \).

\[ P_{\text{err}}(x_0) = P( D(x_0) \neq x_0 | x = x_0 ) \]

1) Maximal prob of error

\[ P_{\text{err}}(C) = \max_{x_0 \in C} P_{\text{err}}(x_0) \]

2) Average prob of error

\[ \mathbb{E} P_{\text{err}}(x) \]
\[ \text{Perr}(c) = \frac{1}{M} \sum_{c} \text{Perr}(c) \]

**MAP decoder**

\[ \min \text{ pr of error given } P_x(z) \]

\[ D(c) = \arg \max_{c \in C} P_r(y = z^2 / P_x(z)) \]

**ML Decoder**

\[ \min \text{ pr of error } \]

\[ D(c^2) = \arg \max_{c \in C} P_r(y = z^2 / P_x(z)) \]

**BSC(p)**

It can be shown that:

\[ D(c) = \arg \min_{c \in C} d_H(c, z^2) \]

\[ \Rightarrow \text{ML decoder.} \]

\[ D(c^2) \] is the closest codeword.

**Example**

\( b = 3, \) repetition code:

\[ c = \{000, 111\} \]

\[ \rightarrow \]

\[ D(y) = \]

\[ 000 \rightarrow 000, \]

\[ 001 \rightarrow 0000, \]

\[ 010 \rightarrow 000, \]

\[ 011 \rightarrow 111, \]

\[ 100 \rightarrow 000, \]

\[ 101 \rightarrow 111, \]

\[ 110 \rightarrow 111, \]

\[ 111 \rightarrow 111, \]

\[ \text{Perr}(000) = P(y = 110 \mid y = 110) \frac{1}{y^2 + y} \]

\[ \text{Perr}(000) = 3p^3(1-p) + 3p^3 \]

\[ \text{Perr}(000) \leq \frac{1}{16/2014} \]

\[ \text{can be easily verified that } \]

\[ \text{Perr}(000) \leq p \]
As \( n \to \infty \), \( P_{\text{err}} \to 0 \) for repetition code

given a number \( \epsilon \), what is the highest rate \( R \), s.t. it is a \((n,M)\)-code of rate \( R \), with \( P_{\text{err}} \leq \epsilon \).

**Shannon's results for BSC(\( p \))**

1) If \( R < 1 - H(p) \) then (achievable)

   - for every \( \epsilon > 0 \), there exists an \((n,M)\)-code of rate \( R \) s.t.
   - \( P_{\text{err}} \leq \epsilon \), for sufficiently large \( n \).

2) If \( R > 1 - H(p) \), then (infeasible)

   - as \( n \to \infty \), \( P_{\text{err}} \to 1 \)
   - for every sequence of \((n,M)\) codes
   - with rate \( R \).

**Remarks**

1) Shannon's results are non-constructive

   In this course, we will learn how to construct codes

2) Shannon's approach ignored computational complexity,

   E.g. complexity of ML decoding technique BSC

   \( \sim \) order of \( M = |F|^n R \).

   We will learn low-complexity decoding

3) \( H(x) \) is called the binary entropy function.

   ![Entropy function graph]

   *Interpretation of \( H(x) \)*
For $z \in \{0,1\}^n$, let $S(z, t) = \{ \tilde{z} : d_n(\tilde{z}, z) = t \}$

\[ |S(z, t)| = \binom{n}{t} \]

As $n \to \infty$

\[ |S(z, n^n)| \sim 2^n \]

more precisely

\[ |S(z, n^n)| = 2^{n H(t) + o(n)} \]

sublinear

\[ \lim_{n \to \infty} \frac{o(n)}{n} = 0 \]

---

**Adversarial channels**

For every $z$, define a set

\[ \Phi_z \subseteq \Phi^n \]

set of possible output strings

given input was $z$

**Example**

$t$ - error channel

Define

\[ B(z, r) = \{ \tilde{z} \in \Phi^n : d_n(\tilde{z}, z) \leq r \} \]

ball of radius $r$ centered at $z$

$k$ - error channel has

\[ \Phi_z^k = B(z, k) \]