Math 5251 Noisy coding (Chap. 4)

* Good time to watch the 3Blue1Brown video on our syllabus!

Now we worry less about minimizing length of codewords based on the input/source alphabet $\Sigma_{in} = \{x_1, x_2, \dots, x_m\}$ (with probabilities $p_1, p_2, \dots, p_m$)

and focus more on dealing with random noise that corrupts the x_i's into output alphabet $\Sigma_{out} = \{y_1, y_2, \dots, y_n\}$ with certain conditional probabilities

$$p_{ij} := P(y_j \text{ is received} \mid x_i \text{ is sent})$$

read the bar as "given that"

Called a discrete memoryless channel C

QUICK CONDITIONAL PROBABILITY REVIEW

$\Omega = \{\omega_1, \omega_2, \ldots, \omega_m\}$ = (finite) sample space

(so probs $p_i = P(\omega_i)$, $p_i \in [0,1]$, $\sum_i p_i = 1$)

Any subset $A \subset \Omega$ is called an event

has a probability $P(A) := \sum_{\omega_i \in A} P(\omega_i)$

For events $A, B \subset \Omega$,

– they're called independent if $P(A \cap B) = P(A) \cdot P(B)$

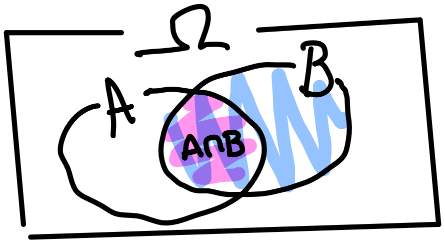
"A and B"

– the conditional probability

$$P(A|B) \underset{\text{DEF'N}}{:=} \frac{P(A \cap B)}{P(B)}$$

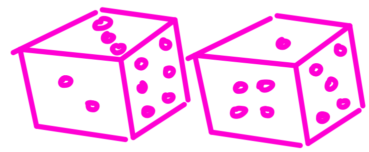
"A given B"

so $P(A \cap B) = P(A|B) \cdot P(B)$



(easy)
EXERCISE: If $P(B) \neq 0$,
A, B independent
$\Leftrightarrow P(A|B) = P(A)$

EXAMPLE $\Omega = \{\text{rolls } \overset{\omega_{ij}=}{(i,j)} \text{ of 2 fair 6-sided dice}\}$



all $P(\omega_{ij}) = \frac{1}{6^2} = \frac{1}{36}$

(uniform distribution / sample space)

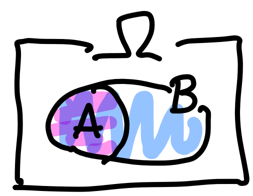
TOTAL:

(1,1)	(1,2)	(1,3)	(1,4)	(1,5)	(1,6)
(2,1)	(2,2)	(2,3)	(2,4)	(2,5)	(2,6)
(3,1)	(3,2)	(3,3)	(3,4)	(3,5)	(3,6)
(4,1)	(4,2)	(4,3)	(4,4)	(4,5)	(4,6)
(5,1)	(5,2)	(5,3)	(5,4)	(5,5)	(5,6)
(6,1)	(6,2)	(6,3)	(6,4)	(6,5)	(6,6)

2 3 4 5 6 7 8 9 10 11 12

$A = \{\text{rolling a total of } 7\}$ $\quad P(A) = \frac{6}{36} = \frac{1}{6}$

$B = \{\text{rolling an odd total}\}$ $\quad P(B) = \frac{2+4+6+4+2}{36} = \frac{1}{2}$



$P(A \cap B) = P(A) = \frac{1}{6}$

since 7 is odd

$$P(A|B) = \frac{P(A\cap B)}{P(B)} = \frac{\frac{1}{6}}{\frac{1}{2}} = \frac{1}{3}$$

$$P(B|A) = \frac{P(A\cap B)}{P(A)} = \frac{\frac{1}{6}}{\frac{1}{6}} = 1$$

EXAMPLES of discrete memoryless channels C

(1) The **binary symmetric channel (BSC)**
with **error probability p**

(most important for us; imagine image bits 0, 1 sent from Mars)

$\Sigma_{in} = \{0,1\} = \Sigma_{out}$

$$\Sigma_{in} \left\{ \begin{tikzcd} 0 \arrow[r, "p_{00}=1-p"] \arrow[dr, "p_{01}=p"'] & 0 \\ 1 \arrow[r, "p_{11}=1-p"'] \arrow[ur, "p_{10}=p"] & 1 \end{tikzcd} \right.$$

Define the Markov ~~transition~~ matrix $M := (p_{ij})$ where

where $p_{ij} := P\left(y_j \text{ received} \,\middle|\, x_i \text{ sent}\right)$

---

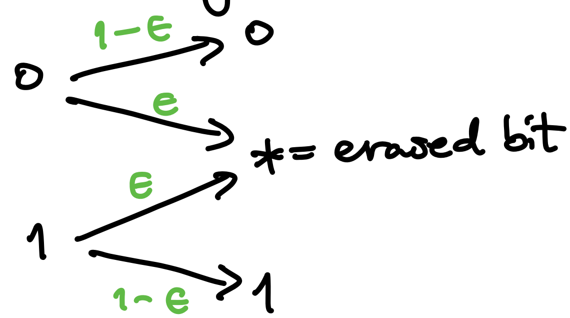
e.g. for BSC,

$$M = \begin{array}{cc} & \overbrace{\begin{array}{cc} 0 & 1 \end{array}}^{\Sigma_{out}} \\ \Sigma_{in} = \left\{ \begin{array}{c} 0 \\ 1 \end{array} \right. & \begin{bmatrix} p_{00} & p_{01} \\ p_{10} & p_{11} \end{bmatrix} \end{array} = \begin{bmatrix} 1-p & p \\ p & 1-p \end{bmatrix}$$

(2) The binary erasure channel with erasure probability ϵ

(imagine bits on a storage device scratched out)

$\Sigma_{in} = \{0,1\}$



$\Sigma_{out} = \{0,1,*\}$

$$M = \begin{array}{cc} & \overbrace{\begin{array}{ccc} 0 & \epsilon & 1 \end{array}}^{\Sigma_{out}} \\ \Sigma_{in} \left\{ \begin{array}{c} 0 \\ 1 \end{array} \right. & \begin{bmatrix} 1-\epsilon & \epsilon & 0 \\ 0 & \epsilon & 1-\epsilon \end{bmatrix} \end{array}$$

Necessarily the rows of M all sum to 1, i.e.

$$\forall i = 1, \dots, m \quad \sum_{j=1}^{n} p_{ij} = \sum_{j=1}^{n} P(y_j \text{ received} \mid x_i \text{ sent}) = 1$$

Why?

Such M are called stochastic matrices.

($p_{ij} \in [0,1]$, $\sum p_{ij} = 1 \;\; \forall$ rows i)

Parity checks (§4.2)

Assuming errors occur independently for each transmitted letter of Σ_{in} (memoryless assumption) one has a calculable chance of transmission error in a longer string, and can try to mitigate it by adding a parity check bit:

means "even/odd-ness"

send

$$b_1 b_2 \cdots b_\ell \text{ as } \begin{cases} b_1 b_2 \cdots b_\ell 0 & \text{if } \sum_i b_i \equiv 0 \bmod 2 \text{ (even)} \\ b_1 b_2 \cdots b_\ell 1 & \text{if } \sum_i b_i \equiv 1 \bmod 2 \text{ (odd)} \end{cases}$$

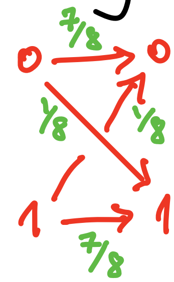
e.g.

$$0100 \xmapsto{\text{sent as}} 01001$$
$$0101 \longmapsto 01010$$

Allows some detection of errors, but no correction, similar to ISBN # error-detection from 1st day

EXAMPLE

Assume we are sending strings from $\{0,1\}^*$ of length 5 through a BSC with error probability $p = 1/8$. What's the probability of an undetected error if



(a) we use no parity check bit ?

(b) we do use a parity check bit, so sending them as strings of length 6 ?

01101 $\longmapsto$ 011011

(a) Each bit $b_1 b_2 b_3 b_4 b_5$ has an equal probability of error, all undetected, so

$$P\left(\begin{matrix}\text{undetected error}\\ \text{with no parity check}\end{matrix}\right) = 1 - P(\text{no errors})$$

$$= 1 - P\left(\begin{matrix}\text{no error}\\ \text{in } b_1\end{matrix}\right) P\left(\begin{matrix}\text{no error}\\ \text{in } b_2\end{matrix}\right) \cdots P\left(\begin{matrix}\text{no error}\\ \text{in } b_5\end{matrix}\right)$$

multiply because BSC is memoryless; errors independent

$$= 1 - \left(1 - \tfrac{1}{8}\right)^5$$

$$\approx 0.4871$$ pretty high!

(b) With 6^{th} parity check bit added, an error is **detected** if exactly 1 bit, or 3 bits, or 5 bits are corrupted, and **undetected** if it's 2, 4, or 6 bits.

So $P\left(\begin{array}{l}\text{undetected error}\\ \text{with parity check bit}\end{array}\right)$

$$= P\left(\begin{array}{c}\text{exactly}\\ \text{2 errors}\end{array} \text{ OR } \begin{array}{c}\text{exactly}\\ \text{4 errors}\end{array} \text{ OR } \begin{array}{c}\text{exactly}\\ \text{6 errors}\end{array}\right)$$

$$= P(\text{2 errors}) + P(\text{4 errors}) + P(\text{6 errors})$$

Why?

$$= \binom{6}{2}\left(\frac{1}{8}\right)^2\left(\frac{7}{8}\right)^4 + \binom{6}{4}\left(\frac{1}{8}\right)^4\left(\frac{7}{8}\right)^2 + \binom{6}{6}\left(\frac{1}{8}\right)^6\left(\frac{7}{8}\right)^0$$

$0\,1\,\cancel{1}\,0\,\cancel{1}\,1$ $\quad$ $\cancel{0}\,\cancel{1}\,\cancel{1}\,0\,1\,\cancel{1}$ $\quad$ $\cancel{0}\,\cancel{1}\,\cancel{1}\,\cancel{0}\,\cancel{1}\,\cancel{1}$

$0\,1\,0\,0\,0\,1$ $\quad$ $1\,0\,0\,0\,1\,0$ $\quad$ $1\,0\,0\,1\,0\,0$

pick the 2 positions from 6 choices for the error locations

≈ 0.1402

much improved!

RECALL **binomial coefficients**

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} = \begin{array}{l}\text{\# of choices}\\ \text{of } k \text{ elements}\\ \text{from } \{1,2,\ldots,n\}\end{array}$$

$$\begin{array}{c} 1 \\ 1 \quad 1 \\ 1 \quad 2 \quad 1 \\ 1 \quad 3 \quad 3 \quad 1 \\ 1 \quad 4 \quad 6 \quad 4 \quad 1 \end{array}$$

$$(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}$$

Note that by adding parity check bits or other redundancy, we are reducing the efficiency of our transmission.

DEF'N: Given a set $\mathcal{C} \subset \{0,1\}^*$ of codewords to send, if the maximum length of the codewords in $\mathcal{C}$ is ℓ, then the the (binary) **rate** of $\mathcal{C}$ is

$$\text{rate}(\mathcal{C}) := \frac{\log_2(|\mathcal{C}|)}{\ell}$$

EXAMPLES

(1) Adding a 6^{th} parity check bit to binary words of length 5 gives a code

$$\mathcal{C} = \left\{ (b_1, b_2, \ldots, b_5, b_6) : \sum_{i=1}^{6} b_i \equiv 0 \bmod 2 \right\}$$

of size $|\mathcal{C}| = 2^5$ and max length 6

so $\text{rate}(\mathcal{C}) = \dfrac{\log_2(2^5)}{6} = \dfrac{5}{6}$

(2) Repeating each string twice before sending

$01101 \longmapsto 01101|01101$

(called a repetition code)

gives a code $\mathcal{C}$ with $|\mathcal{C}| = 2^5$

max length $\ell = 10$

so $\text{rate}(\mathcal{C}) = \dfrac{\log_2(2^5)}{10} = \dfrac{5}{10} = \dfrac{1}{2}$

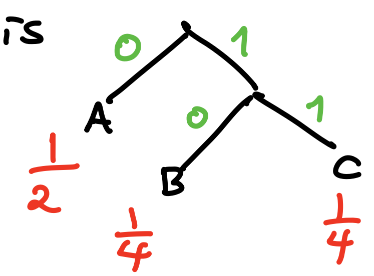
---

(3) Ehrenborg's parlor trick from 1st day conveyed a codeword from $\mathcal{C} = \{0,1,2,\ldots,15\}$ using 7 YES/NO questions = 7 bits $b_1 b_2 \cdots b_7$.

So $\text{rate}(\mathcal{C}) = \dfrac{\log_2(|\mathcal{C}|)}{7} = \dfrac{\log_2(16)}{7} = \dfrac{4}{7}$

---

(4) A Huffman code like this (with no parity checks)



has $\mathcal{C} = \{0, 10, 11\}$

with $|\mathcal{C}| = 3$

max length $\ell = 2$

so $\text{rate}(\mathcal{C}) = \dfrac{\log_2(3)}{2} \approx 0.8$

REMARK:

Why did we divide by ℓ in $\text{rate}(\mathcal{C}) = \frac{\log_2(|\mathcal{C}|)}{\ell}$?

Note that for any u.d. encoding $W \xrightarrow{f} \{0,1\}^*$ where the codewords $\mathcal{C} = \text{image}(f)$ have max length ℓ one will have $\text{rate}(\mathcal{C}) \leq 1$ by this calculation (similar to EXERCISE 3.02):

$$1 \geq \sum_{i=1}^{|\mathcal{C}|} \frac{1}{2^{\ell_i}} \geq \sum_{i=1}^{|\mathcal{C}|} \frac{1}{2^{\ell}} = \frac{|\mathcal{C}|}{2^{\ell}}$$

since max length is ℓ

$$\Rightarrow \quad |\mathcal{C}| \leq 2^{\ell}$$

$$\Rightarrow \quad \text{rate}(\mathcal{C}) = \frac{\log_2(|\mathcal{C}|)}{\ell} \leq \frac{\ell}{\ell} = 1$$