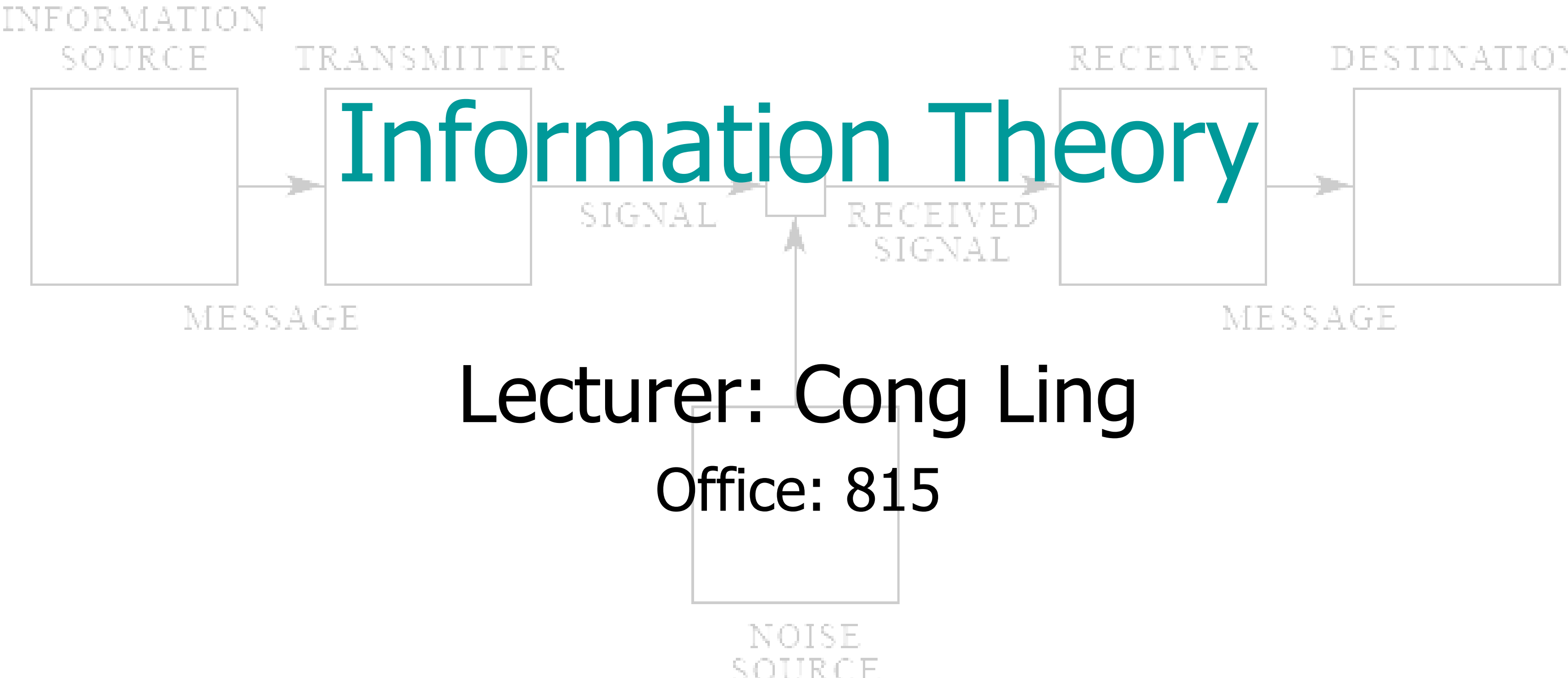
Information Theory



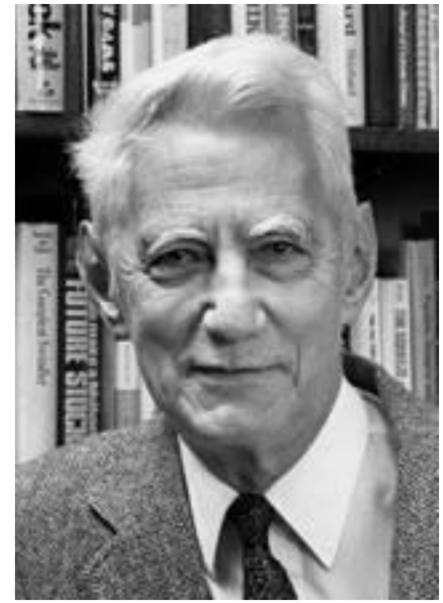
Lecturer: Cong Ling

Office: 815

Lectures

Claude Shannon

- C. E. Shannon, "A mathematical theory of communication," Bell System Technical Journal, 1948.
- Two fundamental questions in communication theory:
- Ultimate limit on data compression
  - entropy
- Ultimate transmission rate of communication
  - channel capacity
- Almost all important topics in information theory were initiated by Shannon

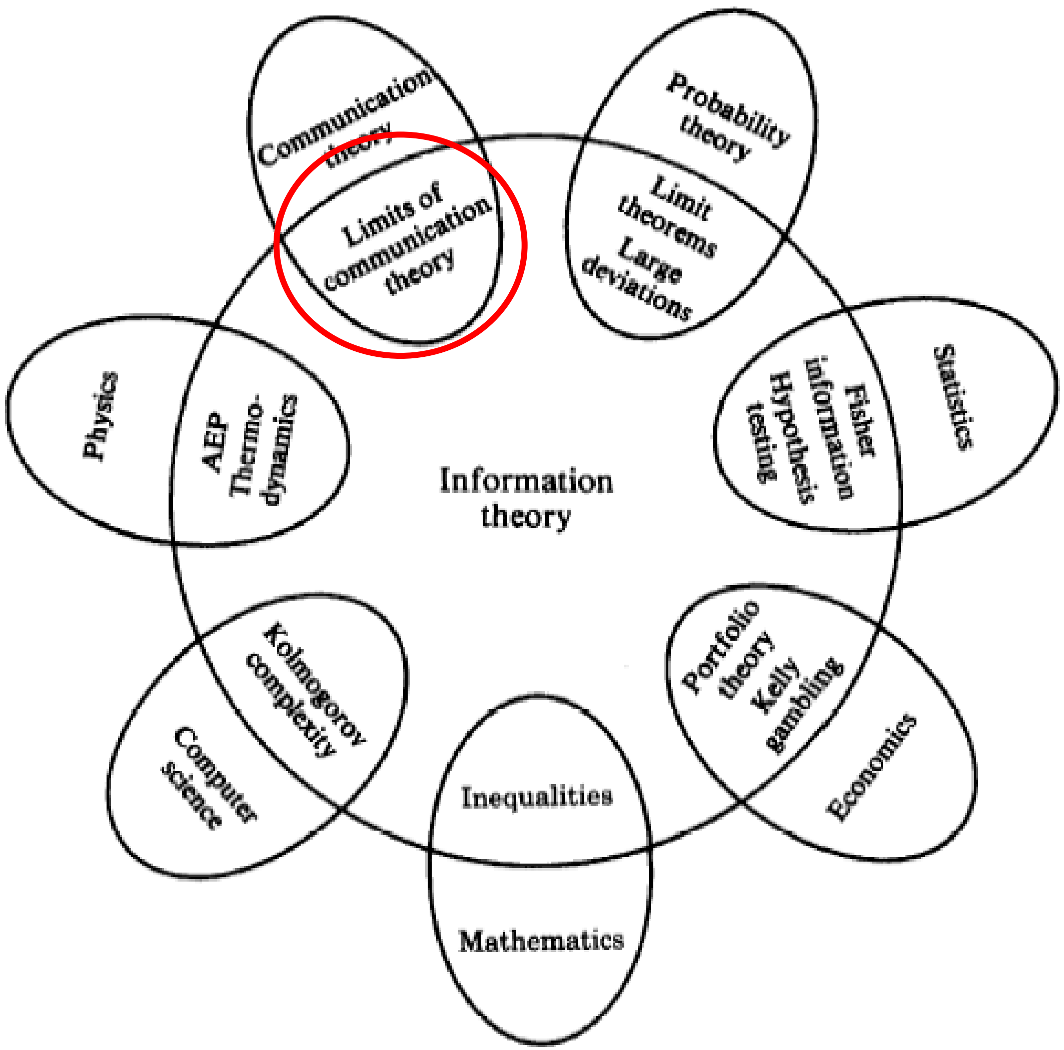


1916 - 2001

Origin of Information Theory

- Common wisdom in 1940s:
  - It is impossible to send information error-free at a positive rate
  - Error control by using retransmission: rate $\to 0$ if error-free
- Still in use today
  - ARQ (automatic repeat request) in TCP/IP computer networking
- Shannon showed reliable communication is possible for all rates below channel capacity
- As long as source entropy is less than channel capacity, asymptotically error-free communication can be achieved
- And anything can be represented in bits
  - Rise of digital information technology

Relationship to Other Fields



Course Objectives

- In this course we will (focus on communication theory):
  - Define what we mean by information.
  - Show how we can compress the information in a source to its theoretically minimum value and show the tradeoff between data compression and distortion.
  - Prove the channel coding theorem and derive the information capacity of different channels.
  - Generalize from point-to-point to network information theory.

Relevance to Practice

- Information theory suggests means of achieving ultimate limits of communication
  - Unfortunately, these theoretically optimum schemes are computationally impractical
  - So some say “little info, much theory” (wrong)
- Today, information theory offers useful guidelines to design of communication systems
  - Polar code (achieves channel capacity)
  - CDMA (has a higher capacity than FDMA/TDMA)
  - Channel-coding approach to source coding (duality)
  - Network coding (goes beyond routing)

Books/Reading

Book of the course:

- *Elements of Information Theory* by T M Cover & J A Thomas, Wiley, £39 for 2nd ed. 2006, or £14 for 1st ed. 1991 (Amazon)

Free references

- *Information Theory and Network Coding* by R. W. Yeung, Springer http://iest2.ie.cuhk.edu.hk/~whyeung/book2/

- *Information Theory, Inference, and Learning Algorithms* by D MacKay, Cambridge University Press http://www.inference.phy.cam.ac.uk/mackay/itila/

- *Lecture Notes on Network Information Theory* by A. E. Gamal and Y.-H. Kim, (Book is published by Cambridge University Press) http://arxiv.org/abs/1001.3404

- C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal,* Vol. 27, pp. 379–423, 623–656, July, October, 1948.

Other Information

- Course webpage: http://www.commsp.ee.ic.ac.uk/~cling
- Assessment: Exam only – no coursework.
- Students are encouraged to do the problems in problem sheets.
- Background knowledge
  - Mathematics
  - Elementary probability
- Needs intellectual maturity
  - Doing problems is not enough; spend some time thinking

Notation

- Vectors and matrices
  - $\mathbf{v}$=vector, $\mathbf{V}$=matrix
- Scalar random variables
  - x = R.V, x = specific value, X = alphabet
- Random column vector of length N
  - $\mathbf{\mathsf{x}}$ = R.V, $\mathbf{x}$ = specific value, X^N = alphabet
  - x_i and x_i are particular vector elements
- Ranges
  - a:b denotes the range a, a+1, …, b
- Cardinality
  - |X| = the number of elements in set X

Discrete Random Variables

- A random variable x takes a value x from the alphabet X with probability $p_X(x)$. The vector of probabilities is $\mathbf{p}_X$.

Examples:

X = [1;2;3;4;5;6], $\mathbf{p}_X = [{}^1/_6; {}^1/_6; {}^1/_6; {}^1/_6; {}^1/_6; {}^1/_6]$

$\mathbf{p}_X$ is a "probability mass vector"

"english text"

X = [a; b;…, y; z; <space>]

$\mathbf{p}_X$= [0.058; 0.013; …; 0.016; 0.0007; 0.193]

Note: we normally drop the subscript from p_X if unambiguous

Expected Values

- If $g(x)$ is a function defined on X then

$$E_X\, g(X) = \sum_{x \in \mathrm{X}} p(x)\, g(x)$$

often write E for E_X

Examples:

X = [1;2;3;4;5;6], $\mathbf{p}_X$ = [$^1/_6$; $^1/_6$; $^1/_6$; $^1/_6$; $^1/_6$; $^1/_6$]

$$E\, X = 3.5 = \mu$$

$\sigma^2 = E(x^2) - (E(x))^2 = 15.17 - 12.25 = 2.92$

$$E\, X^2 = 15.17 = \sigma^2 + \mu^2$$

$$E\, \sin(0.1X) = 0.338$$

$-\log_2(\frac{1}{6}) = 2.58$

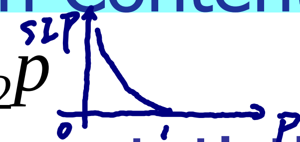
$$E - \log_2(p(X)) = 2.58$$

This is the "entropy" of X

Shannon Information Content

SIP: the amount of info. associated with an event with probability P.

- The Shannon Information Content of an outcome with probability p is $-\log_2 p$



smaller probability
↓
more information.
uncommon cases.

- Shannon's contribution – a statistical view
  - Messages, noisy channels are random
  - Pre-Shannon era: deterministic approach (Fourier…)

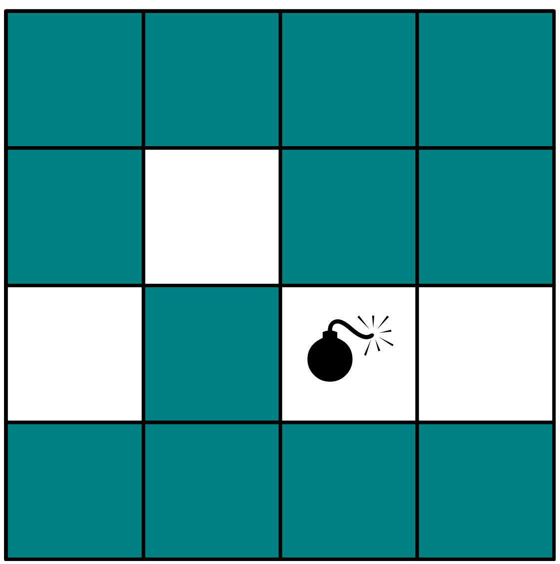
- Example 1: Coin tossing
  - X = [Head; Tail], $\mathbf{p} = [½; ½]$, SIC $= [1; 1]$ bits
- Example 2: Is it my birthday ?
  - X = [No; Yes], $\mathbf{p} = [{}^{364}/_{365}; {}^{1}/_{365}]$, SIC $= [0.004; 8.512]$ bits

Unlikely outcomes give more information

Minesweeper

- Where is the bomb ?
- 16 possibilities – needs 4 bits to specify

$4 = \log_2 16$



Guess		Prob	SIC
1.	No	$^{15}/_{16}$	0.093 bits
2.	No	$^{14}/_{15}$	0.100 bits
3.	No	$^{13}/_{14}$	0.107 bits
4.	Yes	$^{1}/_{13}$	3.700 bits
		Total	4.000 bits

$\text{SIC} = -\log_2 p$

$H'(X) = \log_2 |X| \geq H(X)$

'=' apply at uniform distribution.

entropy: a measurement of uncertainty; bits numbers required to describe a r.v..; expectation of information = transmission cost or verification

Entropy

$$H(x) = E - \log_2(p_x(x)) = -\sum_{x \in X} p_x(x) \log_2 p_x(x)$$

- $H(x)$ = the average Shannon Information Content of x
- $H(x)$ = the average information gained by knowing its value
- the average number of "yes-no" questions needed to find x is in the range $[H(X), H(X)+1)$
- $H(x)$ = the amount of uncertainty before we know its value

We use $\log(x) \equiv \log_2(x)$ and measure $H(X)$ in bits

- if you use $\log_e$ it is measured in nats
- 1 nat $= \log_2(e)$ bits $= 1.44$ bits

- $\log_2(x) = \dfrac{\ln(x)}{\ln(2)}$ $\qquad \dfrac{d \log_2 x}{dx} = \dfrac{\log_2 e}{x}$

$H(X)$ depends only on the probability vector $\mathbf{p}_X$ not on the alphabet X, so we can write $H(\mathbf{p}_X)$

$H(x) = -(1-p)\log(1-p) - p\log p$

$H'(x) = \log(1-p) + 1 - \log p - 1 = \log(1-p) - \log p$

$H'(x) = 0 \Rightarrow p = \frac{1}{2}$

$H(x) = -\frac{1}{2}\log\frac{1}{2} - \frac{1}{2}\log\frac{1}{2} = 1$

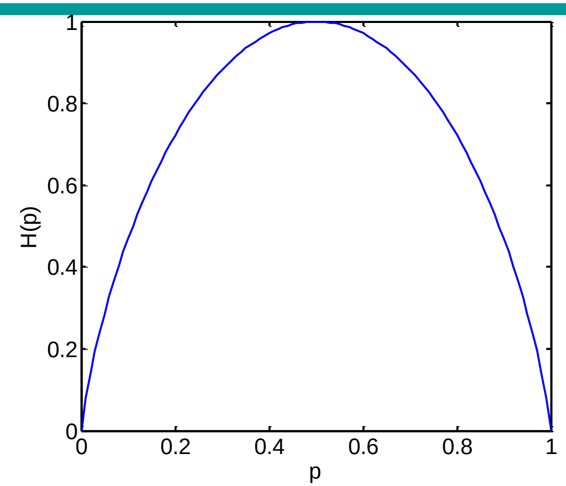
Entropy Examples

(1) Bernoulli Random Variable

X = [0;1], $\mathbf{p}_X = [1-p;\, p]$

$$H(x) = -(1-p)\log(1-p) - p\log p$$

Very common – we write $H(p)$ to mean $H([1-p;\, p])$.



$$H(p) = -(1-p)\log(1-p) - p\log p$$

$$H'(p) = \log(1-p) - \log p$$

$$H''(p) = -p^{-1}(1-p)^{-1}\log e$$

Maximum is when p=1/2

(2) Four Coloured Shapes

X = [●; ■; ◆; ❀], $\mathbf{p}_X = [½;\, ¼;\, ^1/_8;\, ^1/_8]$

$$H(x) = H(\mathbf{p}_x) = \sum -\log(p(x))\,p(x)$$

$$= 1 \times ½ + 2 \times ¼ + 3 \times ⅛ + 3 \times ⅛ = 1.75 \text{ bits}$$

entropy $\begin{cases} \text{monotonic non-decreasing} \\ \text{additive} \\ \text{non-negative} \end{cases}$ $\Rightarrow H(X) = -c \sum_{x \in X} p(x) \log p(x)$ $(c > 0 \to c = 1)$

Comments on Entropy

- Entropy plays a central role in information theory
- Origin in thermodynamics
  - $S = k \ln \Omega$, k: Boltzman's constant, Ω: number of microstates
  - The second law: entropy of an isolated system is non-decreasing
- Shannon entropy
  - Agrees with intuition: additive, monotonic, continuous
  - Logarithmic measure could be derived from an axiomatic approach (Shannon 1948)

Lecture 2

- Joint and Conditional Entropy
  - Chain rule
- Mutual Information
  - If x and y are correlated, their mutual information is the average information that y gives about x
    - E.g. Communication Channel: x transmitted but y received
    - It is the amount of information transmitted through the channel
- Jensen's Inequality

Joint and Conditional Entropy

Joint Entropy: $H(x,y)$

$H(x,y) = E -\log P(x,y)$

$= -\sum_{x,y} P(x,y) \log P(x,y)$

$= \frac{1}{2} \times 1 + \frac{1}{4} \times 2 + \frac{1}{4} \times 2 = 1.5$ bits

$p(x,y)$	y=0	y=1
x=0	½	¼
x=1	0	¼

$$H(x,y) = E - \log p(x,y)$$

$$= -½ \log ½ - ¼ \log ¼ - 0 \log 0 - ¼ \log ¼ = 1.5 \text{ bits}$$

Note: $0 \log 0 = 0$

$0 \log 0 = \lim_{x \to 0} x \log x = \lim_{x \to 0} \frac{\log x}{\frac{1}{x}} = \lim_{x \to 0} \frac{\frac{1}{x}}{-\frac{1}{x^2}} = \lim_{x \to 0} (-x) = 0$

Conditional Entropy: $H(y|x)$

$p(y\|x)$	y=0	y=1
x=0	$^2/_3$	$^1/_3$
x=1	0	1

$H(y|x) = E -\log P(y|x)$

$= -\sum_{x,y} P(x,y) \log P(y|x)$

$= -\frac{1}{2} \times \log \frac{2}{3} - \frac{1}{4} \times \log \frac{1}{3} - \frac{1}{4} \times \log 1$

$= 0.689$ bits

$$H(y|x) = E - \log p(y|x)$$

$$= -\sum_{x,y} p(x,y) \log p(y|x)$$

$$= -½ \log ⅔ - ¼ \log ⅓ - 0 \log 0 - ¼ \log 1 = 0.689 \text{ bits}$$

Note: rows sum to 1

Conditional Entropy – View 1

$$P(y|x) = \frac{P(x,y)}{P(x)}$$

$$H(y|x) = E - \log P(y|x) = E - \log \frac{P(x,y)}{P(x)} = E - \log P(x,y) - E - \log P(x)$$

$$= \sum_x \sum_y P(x,y) \log P(x) = \sum_x P(x) \log P(x)$$

$p(x,y)$	y=0	y=1	$p(x)$
x=0	½	¼	¾
x=1	0	¼	¼

Additional Entropy:

$$p(y \mid x) = p(x,y) \div p(x)$$

$$H(y \mid x) = E - \log p(y \mid x) = -\sum_{x,y} P(x,y) \log P(x,y) + \sum_{x,y} P(x,y) \log P(x)$$

$$= E\{-\log p(x,y)\} - E\{-\log p(x)\} = H(x,y) - H(x)$$

$$= H(x,y) - H(x) = H(\tfrac{1}{2}, \tfrac{1}{4}, 0, \tfrac{1}{4}) - H(\tfrac{3}{4}, \tfrac{1}{4}) = 0.689 \text{ bits}$$

remaining uncertainty

H(*Y*|*X*) is the average additional information in *Y* when you know *X*

$$H(x,y) = -\sum_{x,y} P(x,y) \log P(x,y)$$

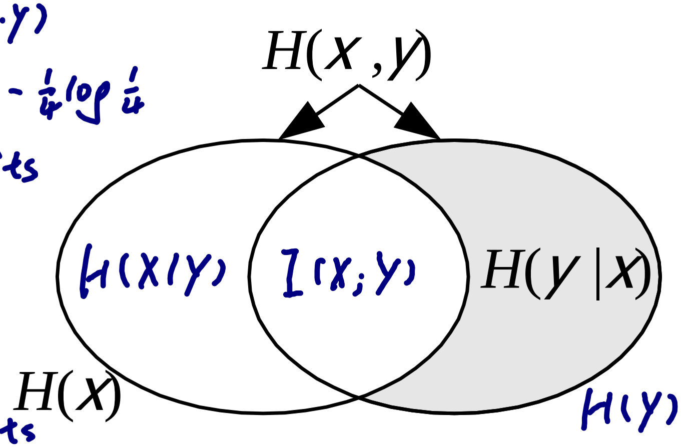
$$= -\tfrac{1}{2}\log\tfrac{1}{2} - \tfrac{1}{4}\log\tfrac{1}{4} - \tfrac{1}{4}\log\tfrac{1}{4}$$

$$= \tfrac{1}{2} + \tfrac{1}{2} + \tfrac{1}{2} = 1.5 \text{ bits}$$

$$H(x) = -\sum_x P(x) \log P(x)$$

$$= -\tfrac{3}{4}\log\tfrac{3}{4} - \tfrac{1}{4}\log\tfrac{1}{4}$$

$$= 0.311 + 0.5 = 0.811 \text{ bits}$$



$$H(y|x) = H(x,y) - H(x) = 0.689 \text{ bits}$$

Conditional Entropy – View 2

$p(x, y)$	y=0	y=1	$H(y \mid x{=}x)$	$p(x)$
x=0	½	¼	H(1/3)	¾
x=1	0	¼	H(1)	¼

Average Row Entropy:

$$H(Y|X) = -\sum_{x,y} P(x,y) \log P(y|x)$$
$$= -\sum_{x,y} P(y|x) P(x) \log P(y|x)$$
$$= -\sum_x P(x) \sum_y P(y|x) \log P(y|x) = \sum_x P(x) H(Y|X=x)$$

Conditional entropy $H(Y|X)$

remaining uncertainty of Y known X

$H(Y|X) = H(X,Y) - H(X)$

weighted average row entropy

$H(Y|X) = \sum_x P(x) H(Y|X=x)$

$$H(y \mid x) = E - \log p(y \mid x) = \sum_{x,y} -p(x,y) \log p(y \mid x)$$

$$= \sum_{x,y} -p(x)\, p(y \mid x) \log p(y \mid x) = \sum_{x \in X} p(x) \sum_{y \in Y} -p(y \mid x) \log p(y \mid x)$$

$$= \sum_{x \in X} p(x) H\left(y \mid x = x\right) = \tfrac{3}{4} \times H(\tfrac{1}{3}) + \tfrac{1}{4} \times H(0) = 0.689 \text{ bits}$$

$$H(Y|X=0) = -\sum_y P(y|X=0) \log P(y|X=0) = -\tfrac{2}{3}\log\tfrac{2}{3} - \tfrac{1}{3}\log\tfrac{1}{3} = 0.918 \text{ bits}$$

$$H(Y|X=1) = -\sum_y P(y|X=1) \log P(y|X=1) = -1\log 1 = 0 \text{ bits}$$

Take a weighted average of the entropy of each row using $p(x)$ as weight

$$\therefore H(Y|X) = \sum_x P(x) H(Y|X=x) = \tfrac{3}{4} \times 0.918 + \tfrac{1}{4} \times 0 = 0.689 \text{ bits}$$

$$H(x,y,z) = -\sum_{x,y,z} P(x,y,z)\log P(x,y,z) = -\sum_{x,y,z} P(x,y,z)\log P(z|x,y)P(y|x)P(x)$$

$$= -\sum_{x,y,z} P(x,y,z)\log P(z|x,y) - \sum_{x,y,z} P(x,y,z)\log P(y|x) - \sum_{x,y,z} P(x,y,z)\log P(x)$$

$$= -\sum_{z} P(z)\log P(z|x,y) - \sum_{y} P(y)\log P(y|x) - \sum_{x} P(x)\log P(x)$$

$$= H(z|x,y) + H(y|x) + H(x)$$

Chain Rules

- Probabilities

$$p(x,y,z) = p(z \mid x,y)\, p(y \mid x)\, p(x)$$

- Entropy

$$\boxed{\begin{aligned} H(x,y,z) &= H(z \mid x,y) + H(y \mid x) + H(x) \\ H(x_{1:n}) &= \sum_{i=1}^{n} H(x_i \mid x_{1:i-1}) \end{aligned}}$$

The log in the definition of entropy converts <u>products</u> of probability into <u>sums</u> of entropy

Mutual information: the reduction of uncertainty in X given Y.

↕

Conditional entropy: the remaining uncertainty of X given Y.

Mutual Information

Mutual information is the average amount of information that you get about x from observing the value of y

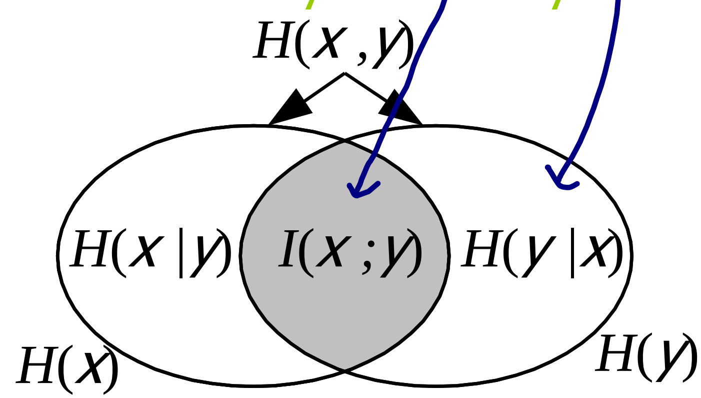
- Or the reduction in the uncertainty of x due to knowledge of y

$$\boxed{I(x;y) = H(x) - H(x \mid y) = H(x) + H(y) - H(x,y)}$$

target

Information in x

Information in x when you already know y



Mutual information is symmetrical

$$I(x;y) = I(y;x)$$

Use ";" to avoid ambiguities between $I(x;y,z)$ and $I(x,y;z)$

Mutual Information Example

$H(x) = -\sum_x P(x)\log P(x) = -\frac{3}{4}\log\frac{3}{4} - \frac{1}{4}\log\frac{1}{4} = 0.811$ bits

$H(y) = -\sum_y P(y)\log P(y) = -\frac{1}{2}\log\frac{1}{2} - \frac{1}{2}\log\frac{1}{2} = 1$ bit

$H(x,y) = -\sum_{x,y} P(x,y)\log P(x,y) = -\frac{1}{2}\log\frac{1}{2} - \frac{1}{4}\log\frac{1}{4} - \frac{1}{4}\log\frac{1}{4} = 1.5$ bits

H(X,Y)

H(X|Y) I(X;Y) H(Y|X)

H(X) H(Y)

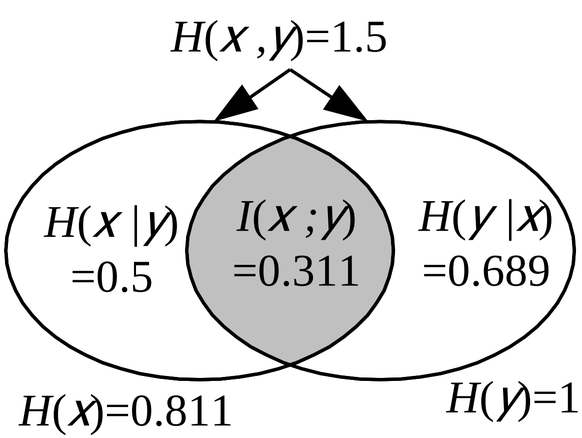
$H(X|Y) = H(X,Y) - H(Y) = 0.5$ bits

$H(Y|X) = H(X,Y) - H(X) = 0.689$ bits

$I(X;Y) = H(X) + H(Y) - H(X,Y) = 0.311$ bits

$p(x,y)$	$y=0$	$y=1$
$x=0$	½	¼
$x=1$	0	¼

- If you try to guess y you have a 50% chance of being correct.
- However, what if you know x?
  - Best guess: choose $y = x$
  - If $x=0$ ($p=0.75$) then 66% correct prob
  - If $x=1$ ($p=0.25$) then 100% correct prob
  - Overall 75% correct probability



$$I(x;y) = H(x) - H(x \mid y)$$
$$= H(x) + H(y) - H(x,y)$$
$$H(x) = 0.811, \quad H(y) = 1, \quad H(x,y) = 1.5$$
$$I(x;y) = 0.311$$

$I(X;Y) = H(X) - H(X|Y)$

$I(X_1, X_2 \cdots X_n; Y) = H(X_1, X_2 \cdots X_n) - H(X_1, X_2 \cdots X_n | Y)$

Conditional Mutual Information

$= \sum_{i=1}^{n} H(X_i | X_{1:i-1}) - \sum_{i=1}^{n} H(X_i | X_{1:i-1}, Y)$

$= \sum_{i=1}^{n} I(X_i ; Y | X_{1:i-1})$

Conditional Mutual Information

$I(X_1, X_2; Y) = I(X_1; Y) + I(X_2; Y | X_1)$

$= H(X_1) - H(X_1|Y) + H(X_2|X_1) - H(X_2|X_1, Y)$

$= H(X_1, X_2) - H(X_1, X_2 | Y)$

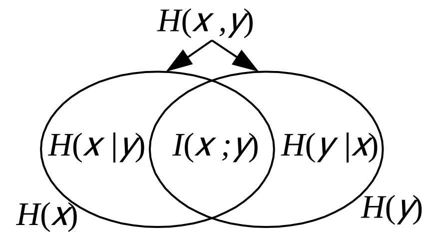
$$\boxed{\begin{aligned} I(x;y\,|\,z) &= H(x\,|\,z) - H(x\,|\,y,z) \\ &= H(x\,|\,z) + H(y\,|\,z) - H(x,y\,|\,z) \end{aligned}}$$

<u>Note</u>: Z conditioning applies to <u>both</u> X and Y

Chain Rule for Mutual Information

$$\boxed{\begin{aligned} & I(x_1, x_2, x_3; y) = I(x_1; y) + I(x_2; y\,|\,x_1) + I(x_3; y\,|\,x_1, x_2) \\ & I(x_{1:n}; y) = \sum_{i=1}^{n} I(x_i; y\,|\,x_{1:i-1}) \end{aligned}}$$

Review/Preview



- **Entropy:** $H(x) = \sum_{x \in \mathrm{X}} -\log_2(p(x))\,p(x) = E - \log_2(p_X(x))$
  - Positive and bounded $0 \le H(x) \le \log|\mathrm{X}|$ ♦
    fixed pattern   uniform distribution
- **Chain Rule:** $H(x,y) = H(x) + H(y\,|\,x) \le H(x) + H(y)$ ♦
  - Conditioning reduces entropy $H(y\,|\,x) \le H(y)$ ♦
- **Mutual Information:**

$$I(y;x) = H(y) - H(y\,|\,x) = H(x) + H(y) - H(x,y)$$

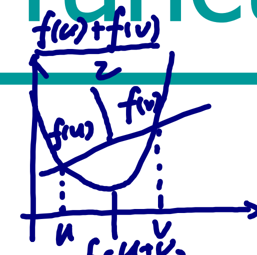
  - Positive and Symmetrical $I(x;y) = I(y;x) \ge 0$ ♦
  - x and y independent $\Leftrightarrow H(x,y) = H(y) + H(x)$ ♦

$$\Leftrightarrow I(x;y) = 0$$

♦ = inequalities not yet proved

Convex & Concave functions

$f(x)$ is strictly convex over (a,b) if



definition

$$f(\lambda u+(1-\lambda)v)<\lambda f(u)+(1-\lambda)f(v) \quad \forall u\neq v\in(a,b),\ 0<\lambda<1$$

- every chord of $f(x)$ lies above $f(x)$
- $f(x)$ is concave $\Leftrightarrow -f(x)$ is convex

Concave is like this

- Examples
  - Strictly Convex: $x^2, x^4, e^x, x\log x\ [x\geq 0]$
  - Strictly Concave: $\log x, \sqrt{x}\quad [x\geq 0]$
  - Convex and Concave: x
  - Test: $\dfrac{d^2 f}{dx^2}>0 \quad \forall x\in(a,b) \qquad \Rightarrow f(x)$ is strictly convex

"convex" (not strictly) uses "≤" in definition and "≥" in test

Jensen's Inequality

Jensen's Inequality: (a) $f(x)$ convex $\Rightarrow Ef(X) \geq f(EX)$

(b) $f(x)$ strictly convex $\Rightarrow Ef(x) > f(Ex)$ unless x constant

Proof by induction on |X|

These sum to 1

– |X|=1: $E\ f(x) = f(E\ x) = f(x_1)$

– |X|=k: $E\ f(x) = \sum_{i=1}^{k} p_i f(x_i) = p_k f(x_k) + (1 - p_k) \sum_{i=1}^{k-1} \frac{p_i}{1 - p_k} f(x_i)$

convex: $Ef(x) \geq f(Ex)$

$$\geq p_k f(x_k) + (1 - p_k) f\left(\sum_{i=1}^{k-1} \frac{p_i}{1 - p_k} x_i \right)$$

Assume JI is true for |X|=k–1

definition: $f(\lambda u + (1-\lambda) v) \leq \lambda f(u) + (1-\lambda) f(v)$

$$\geq f\left(p_k x_k + (1 - p_k) \sum_{i=1}^{k-1} \frac{p_i}{1 - p_k} x_i \right) = f(E\ x)$$

Follows from the definition of convexity for two-mass-point distribution

Jensen's Inequality Example

Mnemonic example:

$f(x) = x^2$: strictly convex

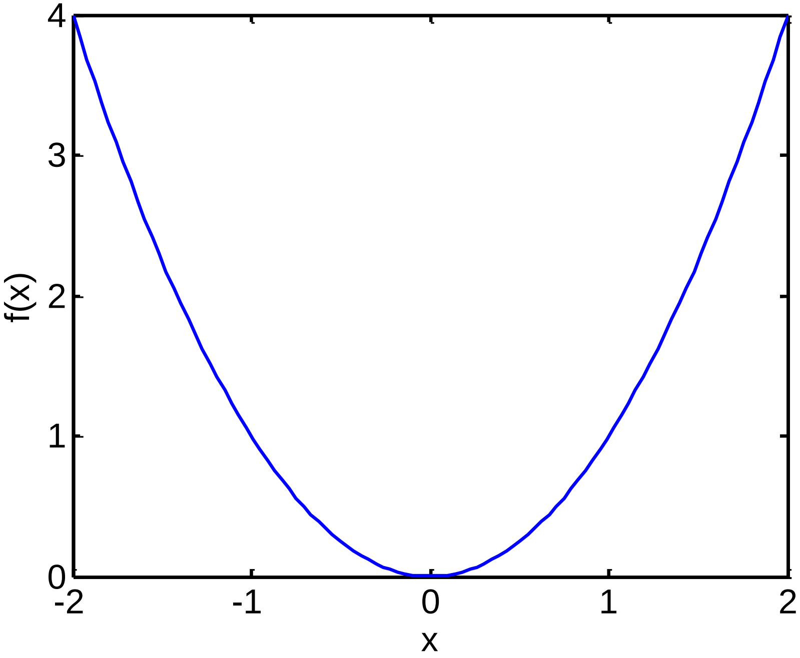
X = $[-1; +1]$

$\mathbf{p} = [½; ½]$

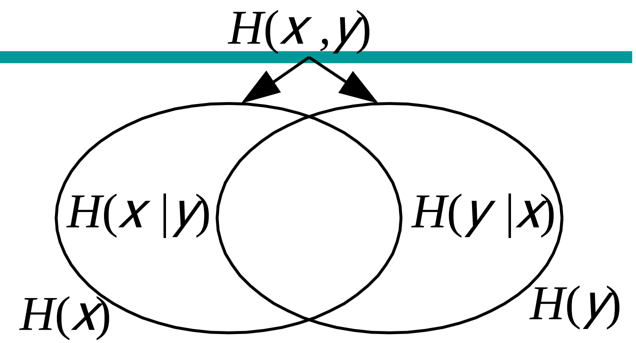
$E\, X = 0$

$f(E\, X) = 0$

$E\, f(X) = 1 > f(E\, X)$



Summary



- Chain Rule:

$$H(x,y) = H(y\,|\,x) + H(x)$$

- Conditional Entropy:

$$H(y\,|\,x) = H(x,y) - H(x) = \sum_{x \in X} p(x) H\left(y\,|\,x\right)$$

  – Conditioning reduces entropy

$$H(y\,|\,x) \le H(y) \quad ♦$$

- Mutual Information $I(x;y) = H(x) - H(x\,|\,y) \le H(x)$ ♦
  – In communications, mutual information is the amount of information transmitted through a noisy channel

- Jensen's Inequality $f(x)$ convex $\Rightarrow Ef(x) \ge f(Ex)$

♦ = inequalities not yet proved

Lecture 3

- Relative Entropy
  - A measure of how different two probability mass vectors are
- Information Inequality and its consequences
  - Relative Entropy is always positive
    - Mutual information is positive
    - Uniform bound
    - Conditioning and correlation reduce entropy
- Stochastic Processes
  - Entropy Rate
  - Markov Processes

entropy $H(X) = -\sum_x P(x)\log_2 P(x) = \sum_x P(x)\log_2 \frac{1}{P(x)}$: a measurement of uncertainty

cross entropy $H(p,q) = -\sum_x p(x)\log_2 q(x) = \sum_x p(x)\log_2 \frac{1}{q(x)}$: the cost of selected scheme to eliminate the system uncertainty.

relative entropy $D(p\|q) = \sum_x p(x)\log_2 \frac{p(x)}{q(x)}$: the cost difference of schemes corresponding to different distributions

→ the invalidity of the system when the real distribution is $p(x)$ and the assumption is $q(x)$

Relative Entropy

Relative Entropy or Kullback-Leibler Divergence between two probability mass vectors $\mathbf{p}$ and $\mathbf{q}$

$$D(\mathbf{p}\,\|\,\mathbf{q}) = \sum_{x\in \mathrm{X}} p(x)\log\frac{p(x)}{q(x)} = E_{\mathbf{p}}\log\frac{p(X)}{q(X)} = E_{\mathbf{p}}\left(-\log q(X)\right) - H(X)$$

where $E_{\mathbf{p}}$ denotes an expectation performed using probabilities $\mathbf{p}$

$D(p\|q) \geq 0$: the assumed distribution $q(x)$ cannot be more accurate than the real case.

$D(\mathbf{p}\|\mathbf{q})$ measures the "distance" between the probability mass functions $\mathbf{p}$ and $\mathbf{q}$.

We must have p_i=0 whenever q_i=0 else $D(\mathbf{p}\|\mathbf{q})=\infty$

Beware: $D(\mathbf{p}\|\mathbf{q})$ is not a true distance because:

- (1) it is asymmetric between $\mathbf{p}$, $\mathbf{q}$ and
- (2) it does not satisfy the triangle inequality.

Relative Entropy Example

$D(p\|q) = E_p \log_2 \frac{p(x)}{q(x)} = E_p \log_2 \frac{1}{q(x)} - H(p(x))$

X = [1 2 3 4 5 6]^T

$H(p) = -\sum_x p(x) \log_2 p(x) = 2.585$. $H(q) = -\sum q(x) \log_2 q(x) = 2.161$

$D(p\|q) = E_p \log_2 \frac{1}{q(x)} - H(p(x)) = 2.935 - 2.585 = 0.35$

$D(q\|p) = E_q \log_2 \frac{1}{p(x)} - H(q(x)) = 2.585 - 2.161 = 0.424$

$$\mathbf{p} = \left[\tfrac{1}{6} \quad \tfrac{1}{6} \quad \tfrac{1}{6} \quad \tfrac{1}{6} \quad \tfrac{1}{6} \quad \tfrac{1}{6}\right] \Rightarrow \quad H(\mathbf{p}) = 2.585$$

$$\mathbf{q} = \left[\tfrac{1}{10} \quad \tfrac{1}{10} \quad \tfrac{1}{10} \quad \tfrac{1}{10} \quad \tfrac{1}{10} \quad \tfrac{1}{2}\right] \Rightarrow \quad H(\mathbf{q}) = 2.161$$

$$D(\mathbf{p}\,\|\,\mathbf{q}) = E_{\mathbf{p}}(-\log q_x) - H(\mathbf{p}) = 2.935 - 2.585 = 0.35$$

$$D(\mathbf{q}\,\|\,\mathbf{p}) = E_{\mathbf{q}}(-\log p_x) - H(\mathbf{q}) = 2.585 - 2.161 = 0.424$$

Information Inequality

The assumed distribution q(x) cannot be more accurate than the real case.

Information (Gibbs') Inequality: $D(\mathbf{p}\,\|\,\mathbf{q}) \geq 0$

- Define $\mathsf{A} = \{x : p(x) > 0\} \subseteq \mathsf{X}$
- Proof $-D(\mathbf{p}\,\|\,\mathbf{q}) = -\sum_{x\in\mathsf{A}} p(x)\log\frac{p(x)}{q(x)} = \sum_{x\in\mathsf{A}} p(x)\log\frac{q(x)}{p(x)}$

log: concave function
$Ef(x) \le f(Ex)$

Jensen's inequality

$$\le \log\left(\sum_{x\in\mathsf{A}} p(x)\frac{q(x)}{p(x)}\right) = \log\left(\sum_{x\in\mathsf{A}} q(x)\right) \le \log\left(\sum_{x\in\mathsf{X}} q(x)\right) = \log 1 = 0$$

If $D(\mathbf{p}\|\mathbf{q})=0$: Since $\log(\,)$ is strictly concave we have equality in the proof only if $q(x)/p(x)$, the argument of $\log$, equals a constant.

But $\sum_{x\in\mathsf{X}} p(x) = \sum_{x\in\mathsf{X}} q(x) = 1$ so the constant must be 1 and $\mathbf{p} \equiv \mathbf{q}$

Information Inequality Corollaries

- Uniform distribution has highest entropy
  - Set $\mathbf{q} = [|X|^{-1}, \ldots, |X|^{-1}]^T$ giving $H(\mathbf{q})=\log|X|$ bits

$$D(\mathbf{p} \parallel \mathbf{q}) = E_{\mathbf{p}}\{-\log q(x)\} - H(\mathbf{p}) = \log|X| - H(\mathbf{p}) \geq 0$$

- Mutual Information is non-negative

$D(P \| Q) = E_P \log_2 \frac{P(x)}{Q(x)}$

$$I(y;x) = H(x) + H(y) - H(x,y) = E \log \frac{p(x,y)}{p(x)p(y)}$$

$I(X;Y) = E \log_2 \frac{P(x,y)}{P(x)P(y)} = D(P(x,y) \| P(x)P(y))$

$$= D(p(x,y) \,\|\, p(x)p(y)) \geq 0$$

with equality only if $p(x,y) \equiv p(x)p(y) \Leftrightarrow x$ and y are independent.

More Corollaries

- Conditioning reduces entropy

$$0 \le I(x;y) = H(y) - H(y \mid x) \quad \Rightarrow \quad H(y \mid x) \le H(y)$$

with equality only if x and y are independent.

- Independence Bound

$$H(x_{1:n}) = \sum_{i=1}^{n} H(x_i \mid x_{1:i-1}) \le \sum_{i=1}^{n} H(x_i)$$

with equality only if all x_i are independent.

E.g.: If all x_i are identical $H(x_{1:n}) = H(x_1)$

independence bound $H(x_{1:n}) = \sum_{i=1}^{n} H(x_i | x_{1:i-1}) \le \sum_{i=1}^{n} H(x_i)$

conditional independence bound $H(x_{1:n}|y_{1:n}) = \sum_{i=1}^{n} H(x_i | x_{1:i-1}, y_{1:n}) \le \sum_{i=1}^{n} H(x_i|y_i)$

mutual information independence bound $I(x_{1:n}|y_{1:n}) = H(x_{1:n}) - H(x_{1:n}|y_{1:n})$

$\ge \sum_{i=1}^{n} H(x_i) - \sum_{i=1}^{n} H(x_i|y_i) = \sum_{i=1}^{n} I(x_i;y_i)$

Conditional Independence Bound

- Conditional Independence Bound

$$H(x_{1:n} | y_{1:n}) = \sum_{i=1}^{n} H(x_i | x_{1:i-1}, y_{1:n}) \le \sum_{i=1}^{n} H(x_i | y_i)$$

- Mutual Information Independence Bound

If all x_i are independent or, by symmetry, if all y_i are independent:

$$I(x_{1:n}; y_{1:n}) = H(x_{1:n}) - H(x_{1:n} | y_{1:n})$$

$$\ge \sum_{i=1}^{n} H(x_i) - \sum_{i=1}^{n} H(x_i | y_i) = \sum_{i=1}^{n} I(x_i; y_i)$$

E.g.: If $n=2$ with x_i i.i.d. Bernoulli ($p=0.5$) and $y_1=x_2$ and $y_2=x_1$, then $I(x_i;y_i)=0$ but $I(x_{1:2}; y_{1:2}) = 2$ bits.

Stochastic Process

Stochastic Process $\{x_i\} = x_1, x_2, \ldots$

Entropy: $H(\{x_i\}) = H(x_1) + H(x_2 \mid x_1) + \ldots \overset{\text{often}}{=} \infty$

Entropy Rate: $\boxed{H(\mathrm{X}) = \lim_{n \to \infty} \frac{1}{n} H(x_{1:n})}$ if limit exists

the increasing rate of entropy w.r.t. n.

- Entropy rate estimates the additional entropy per new sample.
- Gives a lower bound on number of code bits per sample.

Examples:

- Typewriter with m equally likely letters each time: $H(\mathrm{X}) = \log m$
- x_i i.i.d. random variables: $H(\mathrm{X}) = H(x_i)$

Stationary Process

Stochastic Process $\{x_i\}$ is stationary iff

$$p(x_{1:n} = a_{1:n}) = p(x_{k+(1:n)} = a_{1:n}) \quad \forall k, n, a_i \in \mathcal{X}$$

If $\{x_i\}$ is stationary then $H(\mathcal{X})$ exists and

$$\boxed{H(\mathcal{X}) = \lim_{n\to\infty} \frac{1}{n} H(x_{1:n}) = \lim_{n\to\infty} H(x_n \mid x_{1:n-1})}$$

Proof:

$$0 \le H(x_n \mid x_{1:n-1}) \overset{(a)}{\le} H(x_n \mid x_{2:n-1}) \overset{(b)}{=} H(x_{n-1} \mid x_{1:n-2})$$

(a) conditioning reduces entropy, (b) stationarity

Hence $H(x_n|x_{1:n-1})$ is positive, decreasing $\Rightarrow$ tends to a limit, say b

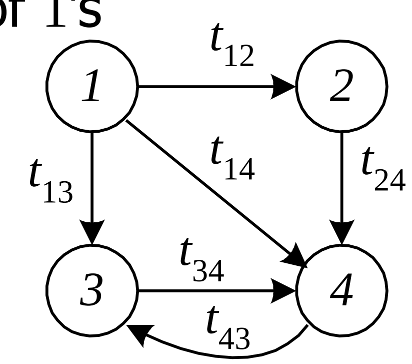
Hence

$$H(x_k \mid x_{1:k-1}) \to b \quad \Rightarrow \quad \frac{1}{n} H(x_{1:n}) = \frac{1}{n}\sum_{k=1}^{n} H(x_k \mid x_{1:k-1}) \to b = H(\mathcal{X})$$

Markov Process (Chain)

Discrete-valued stochastic process $\{x_i\}$ is

- Independent iff $p(x_n|x_{0:n-1})=p(x_n)$
- Markov iff $p(x_n|x_{0:n-1})=p(x_n|x_{n-1})$
  - time-invariant iff $p(x_n=b|x_{n-1}=a) = p_{ab}$ indep of n
  - States
  - Transition matrix: $\mathbf{T} = \{t_{ab}\}$
    - Rows sum to 1: $\mathbf{T1} = \mathbf{1}$ where $\mathbf{1}$ is a vector of 1's
    - $\mathbf{p}_n = \mathbf{T}^T\mathbf{p}_{n-1}$
    - Stationary distribution: $\mathbf{p}_\$ = \mathbf{T}^T\mathbf{p}_\$$



Independent Stochastic Process is easiest to deal with, Markov is next easiest

Stationary Markov Process

If a Markov process is

a) irreducible: you can go from any state a to any b in a finite number of steps

b) aperiodic: $\forall$ state a, the possible times to go from a to a have highest common factor = 1

then it has exactly one stationary distribution, $\mathbf{p}_\$$.

– $\mathbf{p}_\$$ is the eigenvector of $\mathbf{T}^T$ with $\lambda = 1$: $\mathbf{T}^T \mathbf{p}_\$ = \mathbf{p}_\$$
  $\mathbf{T}^n \underset{n\to\infty}{\rightarrow} \mathbf{1}\mathbf{p}_\$^T$ where $\mathbf{1} = [1 \quad 1 \quad \cdots \quad 1]^T$

– Initial distribution becomes irrelevant (asymptotically stationary) $(\mathbf{T}^T)^n \mathbf{p}_0 = \mathbf{p}_\$ \mathbf{1}^T \mathbf{p}_0 = \mathbf{p}_\$, \quad \forall \mathbf{p}_0$

Chess Board

$H(X) = \lim_{n\to\infty} H(X_n|X_{n-1})$

$= \lim_{n\to\infty} -\sum P(X_n, X_{n-1}) \log_2 P(X_n|X_{n-1})$

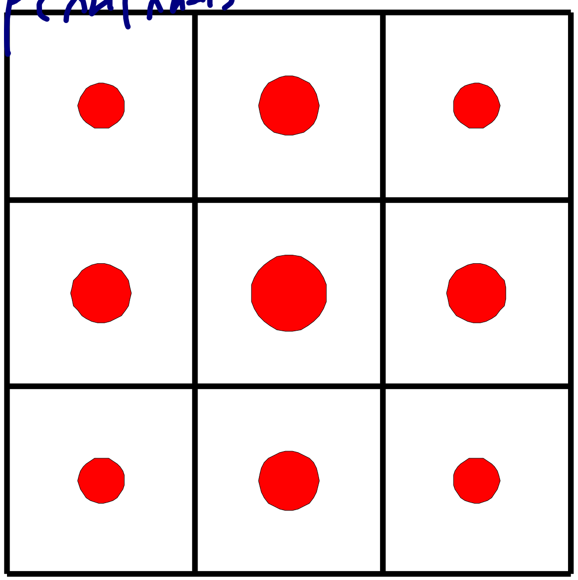
Random Walk

$= -\sum P(X_{n-1}) P(X_n|X_{n-1}) \log_2 P(X_n|X_{n-1})$

$= -\sum P_{\$,i}\, t_{ij} \log_2(t_{ij})$

$P(X_n|X_{n-1}) = \dfrac{P(X_n, X_{n-1})}{P(X_{n-1})}$

H(p_8)=3.0827, H(p_8 | p_7)=2.23038



- Move ⬄ ⇳ ⬁ ⬀ ⬃ ⬂ equal prob
- $\mathbf{p}_1 = [1\ 0\ \ldots\ 0]^T$
  - $H(\mathbf{p}_1) = 0$
- $\mathbf{p}_\$ = {}^1/_{40} \times [3\ 5\ 3\ 5\ 8\ 5\ 3\ 5\ 3]^T$
  - $H(\mathbf{p}_\$) = 3.0855$
- $H(X) = \lim_{n\to\infty} H(\mathcal{X}_n \mid \mathcal{X}_{n-1})$

$$= \lim_{n\to\infty} \sum -p(x_n, x_{n-1}) \log p(x_n \mid x_{n-1}) = \sum_{i,j} -p_{\$,i} t_{i,j} \log(t_{i,j}) = 2.2365$$

Summary

- Relative Entropy: $D(\mathbf{p}\,||\,\mathbf{q}) = E_{\mathbf{p}} \log \dfrac{p(x)}{q(x)} \geq 0$
  - $D(\mathbf{p}||\mathbf{q}) = 0$ iff $\mathbf{p} \equiv \mathbf{q}$
- Corollaries
  - Uniform Bound: Uniform $\mathbf{p}$ maximizes $H(\mathbf{p})$
  - $I(x\,;\,y) \geq 0 \Rightarrow$ Conditioning reduces entropy
  - Indep bounds: $H(x_{1:n}) \leq \sum_{i=1}^{n} H(x_i)$ $\quad H(x_{1:n} \mid y_{1:n}) \leq \sum_{i=1}^{n} H(x_i \mid y_i)$

    $I(x_{1:n}; y_{1:n}) \geq \sum_{i=1}^{n} I(x_i; y_i)$ if x_i or y_i are indep
- Entropy Rate of stochastic process:
  - $\{x_i\}$ stationary: $H(\mathrm{X}) = \lim_{n\to\infty} H(x_n \mid x_{1:n-1})$
  - $\{x_i\}$ stationary Markov:

$$H(\mathrm{X}) = H(x_n \mid x_{n-1}) = \sum_{i,j} -p_{\$,i} t_{i,j} \log(t_{i,j})$$

Lecture 4

- Source Coding Theorem
  - n i.i.d. random variables each with entropy H(X) can be compressed into more than nH(X) bits as n tends to infinity
- Instantaneous Codes
  - Symbol-by-symbol coding
  - Uniquely decodable
- Kraft Inequality
  - Constraint on the code length
- Optimal Symbol Code lengths
  - Entropy Bound

Source Coding

- Source Code: C is a mapping $X \rightarrow D^+$
  - X a random variable of the message
  - D^+ = set of all finite length strings from D
  - D is often binary

an extension contains all codewords derived from the basis set.

  - e.g. {E, F, G} $\rightarrow \{0,1\}^+$: C(E)=0, C(F)=10, C(G)=11
- Extension: C^+ is mapping $X^+ \rightarrow D^+$ formed by concatenating $C(x_i)$ without punctuation
  - e.g. C^+(EFEEGE) =01000110

Desired Properties

- Non-singular: $x_1 \neq x_2 \Rightarrow C(x_1) \neq C(x_2)$
  - Unambiguous description of a single letter of X
- Uniquely Decodable: C^+ is non-singular
  extended message
  - The sequence $C^+(x^+)$ is unambiguous
  - A stronger condition
  - Any encoded string has only one possible source string producing it
  - However, one may have to examine the entire encoded string to determine even the first source symbol
  - One could use punctuation between two codewords but inefficient

Instantaneous Codes

- Instantaneous (or Prefix) Code
  - No codeword is a prefix of another
  - Can be decoded instantaneously without reference to future codewords
- Instantaneous ⇒ Uniquely Decodable ⇒ Non-singular

Examples:

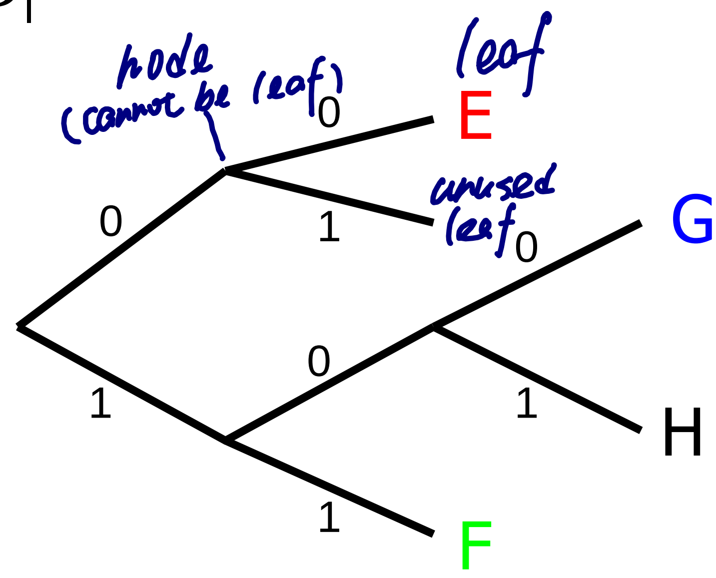
- $C(E,F,G,H) = (0, 1, 00, 11)$ u ✓ I ✗ $\quad \bar{I}\bar{U}$
- $C(E,F) = (0, 101)$ u ✓ I ✓ $\quad IU$
- $C(E,F) = (1, 101)$ u ✓ I ✗ $\quad \bar{I}U$
- $C(E,F,G,H) = (00, 01, 10, 11)$ u ✓ I ✓ $\quad IU$
- $C(E,F,G,H) = (0, 01, 011, 111)$ u ✓ I ✗ $\quad \bar{I}U$

Code Tree

Instantaneous code: C(E,F,G,H) = (00, 11, 100, 101)

Form a D-ary tree where $D = |\mathrm{D}|$

- D branches at each node
- Each codeword is a leaf
- Each node along the path to a leaf is a prefix of the leaf
  - $\Rightarrow$ can't be a leaf itself
- Some leaves may be unused



$11101100000 \rightarrow$ FHGEE

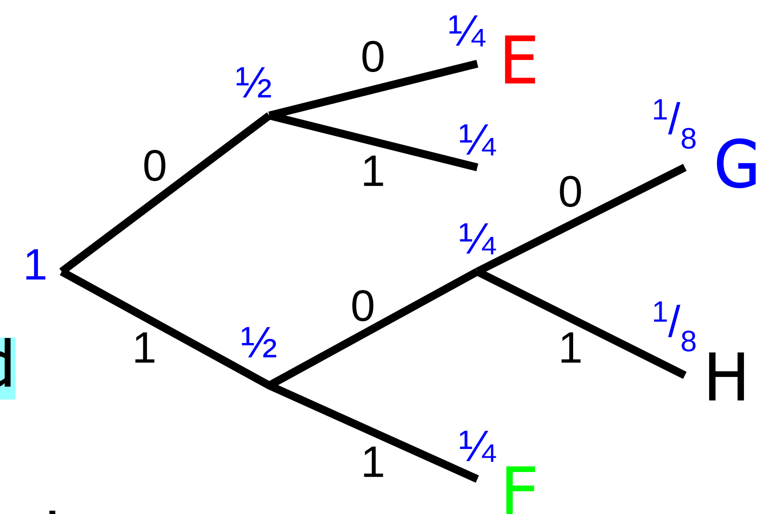
Kraft inequality (for instantaneous codes)

$\sum_{i=1}^{|X|} 2^{-l_i} \le 1$ (budget)

Kraft Inequality (instantaneous codes)

code cost $\propto \frac{1}{\text{code length}}$

- Limit on codeword lengths of instantaneous codes
  - Not all codewords can be too short
- Codeword lengths $l_1, l_2, \ldots, l_{|X|} \Rightarrow$ $\boxed{\sum_{i=1}^{|X|} 2^{-l_i} \le 1}$
- Label each node at depth l with 2^{-l}
- Each node equals the sum of all its leaves
- Equality iff all leaves are utilised
- Total code budget = 1
  Code 00 uses up ¼ of the budget
  Code 100 uses up $^1/_8$ of the budget



Same argument works with D-ary tree

McMillan inequality (for uniquely decodable codewords)

$\sum_{i=1}^{|X|} D^{-l_i} \le 1$ (same with Kraft ineq.)

McMillan Inequality (uniquely decodable codes)

$S^N = \left(\sum_{i=1}^{|X|} D^{-l_i}\right)^N = \sum_{i_1=1}^{|X|}\sum_{i_2=1}^{|X|}\cdots\sum_{i_N=1}^{|X|} D^{-(l_{i_1}+l_{i_2}+\cdots+l_{i_N})} = \sum_{x\in X} D^{-\text{length}(C^+(x))}$

If uniquely decodable C has codeword lengths

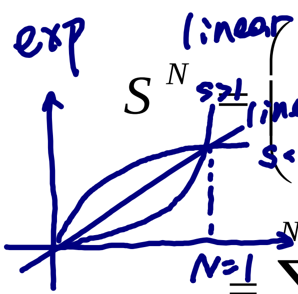
$l_1, l_2, \ldots, l_{|X|}$, then $\sum_{i=1}^{|X|} D^{-l_i} \le 1$ The same

every codebook

length sum

Proof: Let $S = \sum_{i=1}^{|X|} D^{-l_i}$ and $\boxed{M = \max l_i}$ then for ~~any N~~ $N \ge 1$,

$S^N \le NM$ for $N \ge 1 \Rightarrow S \le 1$.



$$S^N = \left(\sum_{i=1}^{|X|} D^{-l_i}\right)^N = \sum_{i_1=1}^{|X|}\sum_{i_2=1}^{|X|}\cdots\sum_{i_N=1}^{|X|} D^{-\left(l_{i_1}+l_{i_2}+\ldots+l_{i_N}\right)} = \sum_{\mathbf{x}\in X^N} D^{-\text{length}\{C^+(\mathbf{x})\}}$$

sum over all possible codeword length

Sum over all sequences of length N

re-order sum by total length

$$= \sum_{l=1}^{NM} D^{-l}\,|\,\mathbf{x} : l = \text{length}\{C^+(\mathbf{x})\}\,| \le \sum_{l=1}^{NM} D^{-l} D^{l} = \sum_{l=1}^{NM} 1 = NM$$

max number of distinct sequences of length l

If $S > 1$ then $S^N > NM$ for some N. Hence $S \le 1$.

Implication: uniquely decodable codes doesn't offer further reduction of codeword lengths than instantaneous codes

McMillan Inequality (uniquely decodable codes)

If uniquely decodable C has codeword lengths

$l_1, l_2, \ldots, l_{|\mathrm{X}|}$, then $\sum_{i=1}^{|\mathrm{X}|} D^{-l_i} \le 1$ The same

Proof: Let $S = \sum_{i=1}^{|\mathrm{X}|} D^{-l_i}$ and $M = \max l_i$ then for any N,

$$S^N = \left(\sum_{i=1}^{|\mathrm{X}|} D^{-l_i} \right)^N = \sum_{i_1=1}^{|\mathrm{X}|} \sum_{i_2=1}^{|\mathrm{X}|} \cdots \sum_{i_N=1}^{|\mathrm{X}|} D^{-\left(l_{i1}+l_{i_2}+\ldots+l_{i_N}\right)} = \sum_{\mathbf{x} \in \mathrm{X}^N} D^{-\mathrm{length}\{C^+(\mathbf{x})\}}$$

$$= \sum_{l=1}^{NM} D^{-l} \,|\, \mathbf{x} : l = \mathrm{length}\{C^+(\mathbf{x})\} \,| \le \sum_{l=1}^{NM} D^{-l} D^{l} = \sum_{l=1}^{NM} 1 = NM$$

If $S > 1$ then $S^N > NM$ for some N. Hence $S \le 1$.

Implication: uniquely decodable codes doesn't offer further reduction of codeword lengths than instantaneous codes

How Short are Optimal Codes?

If $l(x) = \text{length}(C(x))$ then C is optimal if
$L = E\, l(x)$ is as small as possible.

We want to minimize $\sum_{x \in X} p(x) l(x)$ subject to

1. $\sum_{x \in X} D^{-l(x)} \leq 1$
2. all the $l(x)$ are integers

Simplified version:

Ignore condition 2 and assume condition 1 is satisfied with equality.

optimistical mistakes

less restrictive so lengths may be shorter than actually possible $\Rightarrow$ lower bound

Optimal Codes (non-integer l_i)

- Minimize $\sum_{i=1}^{|X|} p(x_i) l_i$ subject to $\sum_{i=1}^{|X|} D^{-l_i} = 1$

optimum at bound

$D^{-l_i} = \frac{p(x_i)}{\lambda \ln D}$

$\underbrace{\sum_{i=1}^{|X|} D^{-l_i}}_{} = \underbrace{\sum_{i=1}^{|X|} p(x_i)}_{} \cdot \frac{1}{\lambda \ln D} \Rightarrow \lambda = \frac{1}{\ln D}$

Use Lagrange multiplier:

Define $J = \sum_{i=1}^{|X|} p(x_i) l_i + \lambda \sum_{i=1}^{|X|} D^{-l_i}$ and set $\frac{\partial J}{\partial l_i} = 0$

$\frac{\partial D^x}{dx} = \ln D \cdot D^x$

$\therefore D^{-l_i} = p(x_i)$

$\therefore l_i = -\log_D p(x_i) = H_D(X)$

$$\frac{\partial J}{\partial l_i} = p(x_i) - \lambda \ln(D) D^{-l_i} = 0 \quad \Rightarrow \quad D^{-l_i} = p(x_i) / \lambda \ln(D)$$

also $\sum_{i=1}^{|X|} D^{-l_i} = 1 \quad \Rightarrow \quad \lambda = 1/\ln(D) \quad \Rightarrow \quad l_i = -\log_D(p(x_i))$

Information content = codeword length

$$E\, l(X) = E - \log_D(p(X)) = \frac{E - \log_2(p(X))}{\log_2 D} = \frac{H(X)}{\log_2 D} = H_D(X)$$

entropy of x base D.

no uniquely decodable code can do better than this

Bounds on Optimal Code Length

Round up optimal code lengths: $l_i = \lceil -\log_D p(x_i) \rceil$

- l_i are bound to satisfy the Kraft Inequality (since the optimum lengths do) $\sum D^{-l_i} \le 1$? $\lceil l_i \rceil$ ✓ $\lfloor l_i \rfloor$ × $\lfloor l_i \rceil$ ×
- For this choice, $-\log_D(p(x_i)) \le l_i \le -\log_D(p(x_i)) + 1$
- Average shortest length:

$$\boxed{H_D(x) \le L^* < H_D(x) + 1}$$

(since we added <1 to optimum values)

- We can do better by encoding blocks of n symbols

normalise the length

$$n^{-1} H_D(x_{1:n}) \le n^{-1} E\, l(x_{1:n}) \le n^{-1} H_D(x_{1:n}) + n^{-1}$$

$n \uparrow \Rightarrow$ tighter bound $\Rightarrow$ avg. symbol length $\to$ entropy.

- If entropy rate of x_i exists ($\Leftarrow x_i$ is stationary process)

$$n^{-1} H_D(x_{1:n}) \to H_D(\mathrm{X}) \quad \Rightarrow \quad n^{-1} E\, l(x_{1:n}) \to H_D(\mathrm{X})$$

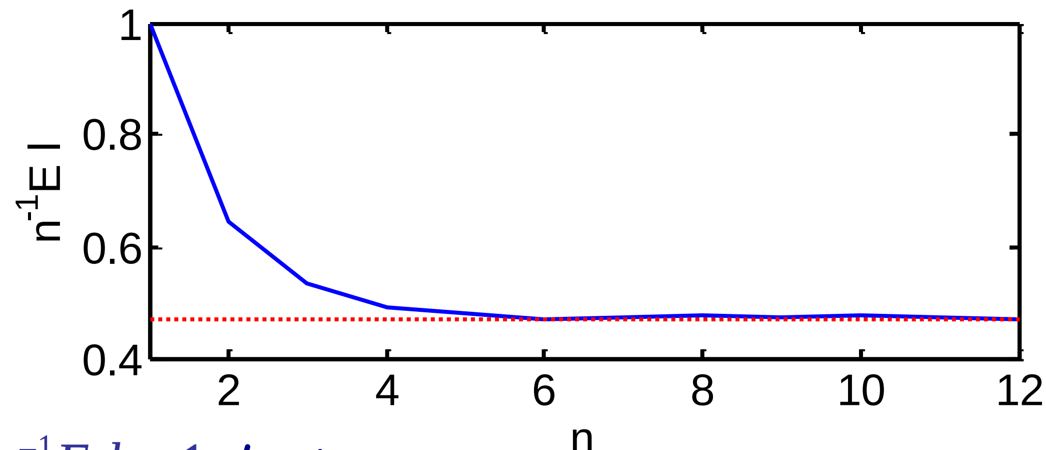
Also known as source coding theorem

Block Coding Example

X = [A;B], $\mathbf{p}_x = [0.9; 0.1]$

$H(x_i) = 0.469$

Huffman coding:



$n^{-1}El = \frac{0.9\times1+0.1\times1}{1} = 1$

- n=1

sym	A	B
prob	0.9	0.1
code	0	1

$n^{-1}E\,l = 1$ block length n

$n^{-1}El = \frac{0.81\times1+0.09\times2+0.1\times3}{2} = 0.645$

⇒ avg. symbol length → entropy

$n^{-1}El$ (longer block. less uncertainty)

- n=2

sym	AA	AB	BA	BB
prob	0.81	0.09	0.09	0.01
code	0	11	100	101

$n^{-1}E\,l = 0.645$

- n=3

sym	AAA	AAB	…	BBA	BBB
prob	0.729	0.081	…	0.009	0.001
code	0	101	…	10010	10011

$n^{-1}E\,l = 0.583$

The extra 1 bit inefficiency becomes insignificant for large blocks

Summary

- McMillan Inequality for D-ary codes:
  - any uniquely decodable C has $\sum_{i=1}^{|\mathsf{X}|} D^{-l_i} \leq 1$
- Any uniquely decodable code:

$$E\, l(x) \geq H_D(x)$$

- Source coding theorem
  - Symbol-by-symbol encoding

$$H_D(x) \leq E\, l(x) \leq H_D(x) + 1$$

  - Block encoding $n^{-1} E\, l(x_{1:n}) \rightarrow H_D(\mathsf{X})$

Lecture 5

- Source Coding Algorithms
- Huffman Coding
- Lempel-Ziv Coding

Huffman Code

An optimal binary instantaneous code must satisfy:

1. $p(x_i) > p(x_j) \Rightarrow l_i \le l_j$ (else swap codewords)
2. The two longest codewords have the same length (else chop a bit off the longer codeword)
3. $\exists$ two longest codewords differing only in the last bit (else chop a bit off all of them)

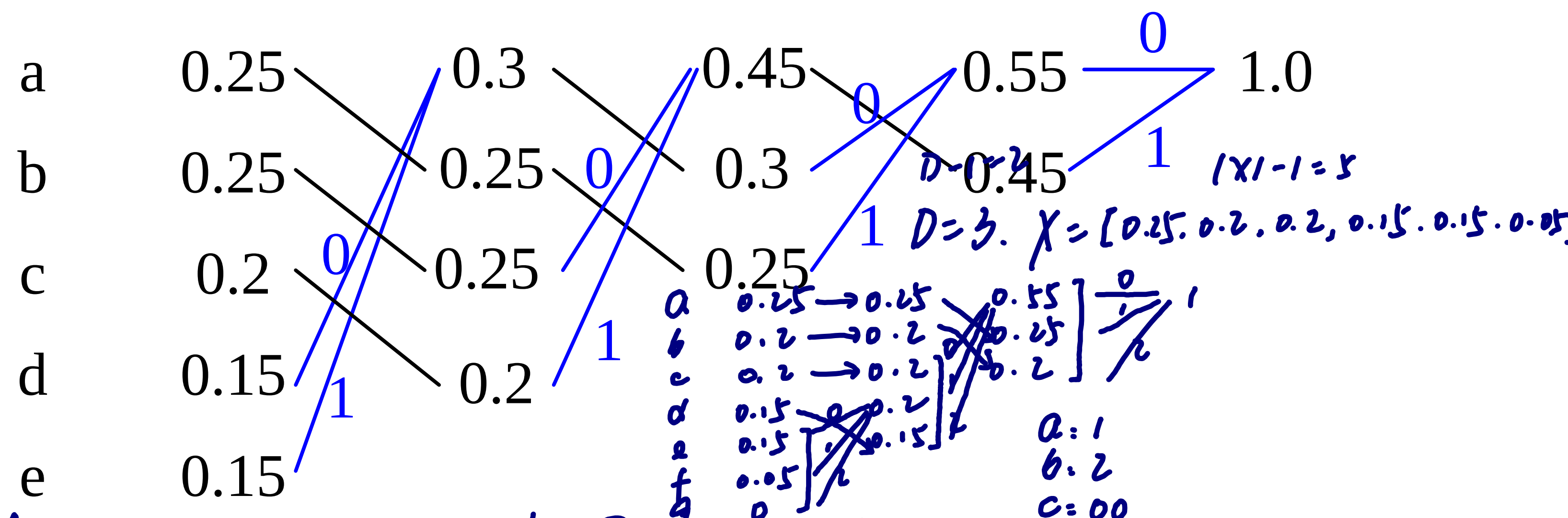
Huffman Code construction

1. Take the two smallest $p(x_i)$ and assign each a different last bit. Then merge into a single symbol.
2. Repeat step 1 until only one symbol remains

Used in JPEG, MP3…

Huffman Code Example

X = [a, b, c, d, e], p_X = [0.25 0.25 0.2 0.15 0.15]



a: 1
b: 2
c: 00
d: 02
e: 010
f: 011
g: 012 (X)

① go from symbol to the end ② read backwards

Read diagram backwards for codewords:

C(X) = [01 10 11 000 001], L = 2.3, H(X) = 2.286

For D-ary code, first add extra zero-probability symbols until |X|–1 is a multiple of D–1 and then group D symbols at a time

Huffman Code is Optimal Instantaneous Code

Huffman traceback gives codes for progressively larger alphabets:

a 0.55 0 1 a:0
b 0.45 1 b:1

$L_2 = 0.55 + 0.45 = 1$

0.55 —0— 1.0
0.45 —1—

$\mathbf{p}_2$=[0.55 0.45],

$\mathbf{c}_2$=[0 1], L_2=1

a 0.45, b 0.3 0, c 0.25 1 → 0.55 0, 0.45 1 → 1 a: 1 b: 00 c: 01

$L_3 = 0.45 \times 1 + 0.55 \times 2 = 1.55$

$\mathbf{p}_3$=[0.45 0.3 0.25],

$\mathbf{c}_3$=[1 00 01], L_3=1.55

a 0.3, b 0.25, c 0.25 0, d 0.2 1 → 0.45, 0.3 0, 0.25 1 → 0.55 0, 0.45 1 → 1 a: 00 b: 01 c: 10 d: 11

$L_4 = 2$

$\mathbf{p}_4$=[0.3 0.25 0.25 0.2],

$\mathbf{c}_4$=[00 01 10 11], L_4=2

a 0.25, b 0.25, c 0.2, d 0.15 0, e 0.15 1 → 0.3, 0.25, 0.25 0, 0.2 1 → 0.45, 0.3 0, 0.25 1 → 0.55 0, 0.45 1 → 1 a: 01 b: 10 c: 11 d: 000 e: 001

$\mathbf{p}_5$=[0.25 0.25 0.2 0.15 0.15],

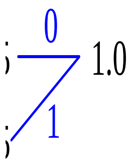
$\mathbf{c}_5$=[01 10 11 000 001], L_5=2.3

$L_5 = 2 \times 0.7 + 3 \times 0.3 = 2.3$

We want to show that all these codes are optimal including C_5

Huffman Code is Optimal Instantaneous Code

Huffman traceback gives codes for progressively larger alphabets:



$\mathbf{p}_2$=[0.55 0.45],

$\mathbf{c}_2$=[0 1], L_2=1

$\mathbf{p}_3$=[0.45 0.3 0.25],

$\mathbf{c}_3$=[1 00 01], L_3=1.55

$\mathbf{p}_4$=[0.3 0.25 0.25 0.2],

$\mathbf{c}_4$=[00 01 10 11], L_4=2

$\mathbf{p}_5$=[0.25 0.25 0.2 0.15 0.15],

$\mathbf{c}_5$=[01 10 11 000 001], L_5=2.3

We want to show that all these codes are optimal including C_5

Huffman Optimality Proof

Suppose one of these codes is sub-optimal:

- $\exists\ m>2$ with $\mathbf{c}_m$ the first sub-optimal code (note $\mathbf{c}_2$ is definitely optimal)
- An optimal $\mathbf{c}'_m$ must have $L_{C'm} < L_{Cm}$
- Rearrange the symbols with longest codes in $\mathbf{c}'_m$ so the two lowest probs p_i and p_j differ only in the last digit (doesen't change optimality)
- Merge x_i and x_j to create a new code $\mathbf{c}'_{m-1}$ as in Huffman procedure
- $L_{C'm-1} = L_{C'm} - p_i - p_j$ since identical except 1 bit shorter with prob $p_i + p_j$
- But also $L_{Cm-1} = L_{Cm} - p_i - p_j$ hence $L_{C'm-1} < L_{Cm-1}$ which contradicts assumption that $\mathbf{c}_m$ is the first sub-optimal code

Hence, Huffman coding satisfies $H_D(x) \le L < H_D(x)+1$

Note: Huffman is just one out of many possible optimal codes

Shannon-Fano Code

Fano code

Fano: split

1. Put probabilities in decreasing order
2. Split as close to 50-50 as possible; repeat with each half

source of information loss

largest probability: shortest. all zeros

a	0.20	00
b	0.19	010
c	0.17	011
d	0.15	100
e	0.14	101
f	0.06	110
g	0.05	1110
h	0.04	1111

smallest probability. longest. all ones.

$H(\boldsymbol{x}) = 2.81$ bits

$L_{SF} = 2.89$ bits

Not necessarily optimal: the best code for this $\mathbf{p}$ actually has $L = 2.85$ bits

Shannon versus Huffman

Shannon

$$F_i = \sum_{k=1}^{i-1} p(x_k), \qquad p(x_1) \geq p(x_2) \geq \cdots \geq p(x_m)$$

encoding: round the number $F_i \in [0,1]$ to $\lceil -\log p(x_i) \rceil$ bits

Shannon: $\lceil -\log p(x_i) \rceil$

$$H_D(x) \leq L_{SF} \leq H_D(x) + 1 \qquad \text{(excercise)}$$

$$\mathbf{p}_x = [0.36 \quad 0.34 \quad 0.25 \quad 0.05] \Rightarrow H(x) = 1.78 \text{ bits}$$

$$-\log_2 \mathbf{p}_x = [1.47 \quad 1.56 \quad 2 \quad 4.32]$$

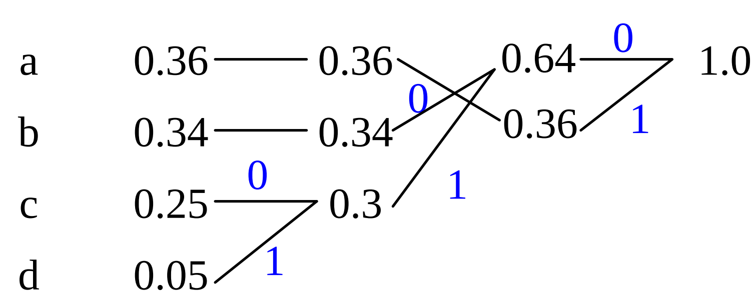
$$\mathbf{l}_S = \lceil -\log_2 \mathbf{p}_x \rceil = [2 \quad 2 \quad 2 \quad 5]$$

$$L_S = 2.15 \text{ bits}$$

Huffman

$$\mathbf{l}_H = [1 \quad 2 \quad 3 \quad 3]$$

$$L_H = 1.94 \text{ bits}$$



Individual codewords may be longer in Huffman than Shannon but not the average

Issues with Huffman Coding

- Requires the probability distribution of the source
  - Must recompute entire code if any symbol probability changes
    - A block of N symbols needs $|X|^N$ pre-calculated probabilities
- For many practical applications, however, the underlying probability distribution is unknown
  - Estimate the distribution
    - Arithmetic coding: extension of Shannon-Fano coding; can deal with large block lengths
  - Without the distribution
    - Universal coding: Lempel-Ziv coding

Universal Coding

- Does not depend on the distribution of the source
- Compression of an individual sequence
- Run length coding
  - Runs of data are stored (e.g., in fax machines)

    Example: WWWWWWWWWBBWWWWWWWBBBBBBWW

    white black

    9W2B7W6B2W
- Lempel-Ziv coding encode strings into phrases
  - Generalization that takes advantage of runs of strings of characters (such as WWWWWWWWWBB)
  - Adaptive dictionary compression algorithms
  - Asymptotically optimum: achieves the entropy rate for any stationary ergodic source

Lempel-Ziv Coding (LZ78)

Memorize previously occurring substrings in the input data

- parse input into the shortest possible distinct 'phrases', i.e., each phrase is the shortest phrase not seen earlier

phrases	#	codewords (head location + tail)
A	1	0A
B	2	0B
AA	3	1A
BA	4	2A
BAB	5	4B
BB	6	2B
AB	7	1B

- number the phrases starting from 1 (0 is the empty string)

  ABAABABABBBAB… new phrase = old phrase (head) + tail

  12 3 4 5 6 7

  Look up a dictionary

- each phrase consists of a previously occurring phrase (head) followed by an additional A or B (tail)
- encoding: give location of head followed by the additional symbol for tail

  0A0B1A2A4B2B1B…

- decoder uses an identical dictionary

locations are underlined

dict	msg
0001	1
0010	0
0011	11
0100	01
0101	010
0110	00
0111	10
1000	0100
1001	01001

Lempel-Ziv Example

Input = 1011010100010010001001010010

Dictionary		Send	Decode
0000	φ	(0)1	1
0001	1	00	0
0010	0	011	11
0011	11	101	01
0100	01	1000	010
0101	010	0100	00
0110	00	0010	10
0111	10	1010	0100
1000	0100	10001	01001
1001	01001	10010	010010

location

No need to always send 4 bits

Remark:

- No need to send the dictionary (imagine zip and unzip!)
- Can be reconstructed
- Need to send 0's in 01, 010 and 001 to avoid ambiguity (i.e., instantaneous code)

Lempel-Ziv Comments

Dictionary D contains K entries $D(0), \ldots, D(K-1)$. We need to send $M=\text{ceil}(\log K)$ bits to specify a dictionary entry. Initially $K=1$, $D(0)= \phi$ = null string and $M=\text{ceil}(\log K) = 0$ bits.

Input	Action
1	"1" $\notin D$ so send "1" and set $D(1)$="1". Now $K=2 \Rightarrow M=1$.
0	"0" $\notin D$ so split it up as "ϕ"+"0" and send location "0" (since $D(0)= \phi$) followed by "0". Then set $D(2)$="0" making $K=3 \Rightarrow M=2$.
1	"1" $\in D$ so don't send anything yet – just read the next input bit.
1	"11" $\notin D$ so split it up as "1" + "1" and send location "01" (since $D(1)$= "1" and $M=2$) followed by "1". Then set $D(3)$="11" making $K=4 \Rightarrow M=2$.
0	"0" $\in D$ so don't send anything yet – just read the next input bit.
1	"01" $\notin D$ so split it up as "0" + "1" and send location "10" (since $D(2)$= "0" and $M=2$) followed by "1". Then set $D(4)$="01" making $K=5 \Rightarrow M=3$.
0	"0" $\in D$ so don't send anything yet – just read the next input bit.
1	"01" $\in D$ so don't send anything yet – just read the next input bit.
0	"010" $\notin D$ so split it up as "01" + "0" and send location "100" (since $D(4)$= "01" and $M=3$) followed by "0". Then set $D(5)$="010" making $K=6 \Rightarrow M=3$.

So far we have sent 1000111011000 where dictionary entry numbers are in red.

Lempel-Ziv Properties

- Simple to implement
- Widely used because of its speed and efficiency
  - applications: compress, gzip, GIF, TIFF, modem …
  - variations: LZW (considering last character of the current phrase as part of the next phrase, used in Adobe Acrobat), LZ77 (sliding window)
  - different dictionary handling, etc
- Excellent compression in practice
  - many files contain repetitive sequences
  - worse than arithmetic coding for text files

Asymptotic Optimality

- Asymptotically optimum for stationary ergodic source (i.e. achieves entropy rate)
- Let $c(n)$ denote the number of phrases for a sequence of length n
- Compressed sequence consists of $c(n)$ pairs (location, last bit)
- Needs $\boxed{c(n)[\log c(n)+1]}$ bits in total
- $\{X_i\}$ stationary ergodic $\Rightarrow$

$$\limsup_{n\to\infty} n^{-1} l(X_{1:n}) = \limsup_{n\to\infty} \frac{c(n)[\log c(n)+1]}{n} \le H(\mathrm{X}) \text{ with probability } 1$$

  - Proof: C&T chapter 12.10
  - may only approach this for an enormous file

Summary

- Huffman Coding: $H_D(x) \le E\, l(x) \le H_D(x) + 1$
  - Bottom-up design
  - Optimal $\Rightarrow$ shortest average length
- Shannon-Fano Coding: $H_D(x) \le E\, l(x) \le H_D(x) + 1$
  - Intuitively natural top-down design
- Lempel-Ziv Coding
  - Does not require probability distribution
  - Asymptotically optimum for stationary ergodic source (i.e. achieves entropy rate)

Lecture 6

- Markov Chains
  - Have a special meaning
  - Not to be confused with the standard definition of Markov chains (which are sequences of discrete random variables)
- Data Processing Theorem
  - You can't create information from nothing
- Fano's Inequality
  - Lower bound for error in estimating X from Y

Markov Chains

If we have three random variables: x, y, z

$$p(x, y, z) = p(z \mid x, y)\, p(y \mid x)\, p(x)$$

they form a Markov chain $x \to y \to z$ if

$$p(z \mid x, y) = p(z \mid y) \Leftrightarrow p(x, y, z) = p(z \mid y)\, p(y \mid x)\, p(x)$$

y, agent. translator.

A Markov chain $x \to y \to z$ means that

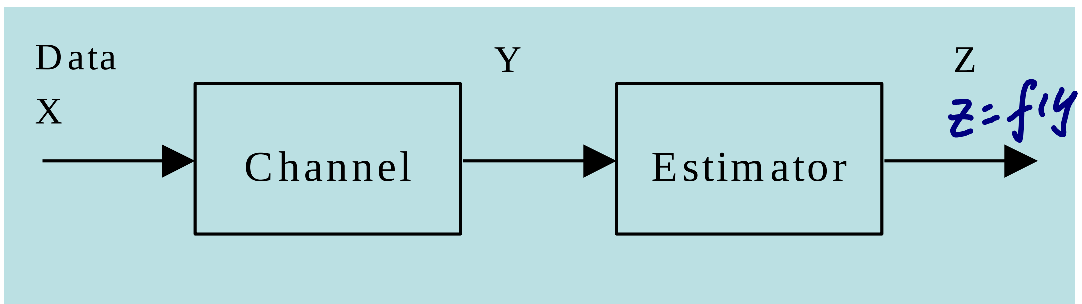
- the only way that x affects z is through the value of y
- if you already know y, then observing x gives you no additional information about z, i.e. $I(x; z \mid y) = 0 \Leftrightarrow H(z \mid y) = H(z \mid x, y)$
- if you know y, then observing z gives you no additional information about x.

$H(x \mid y) = H(x \mid y, z)$

$I(x; z \mid y) = H(z \mid y) - H(z \mid x, y)$

Data Processing

- Estimate $z = f(y)$, where f is a function
- A special case of a Markov chain $x \rightarrow y \rightarrow f(y)$



$z = f(y)$: estimation

- Does processing of y increase the information that y contains about x? No. even less!

Markov Chain Symmetry

$x \to y \to z$ $\Rightarrow$ $z \to y \to x$
MC MC

If $x \to y \to z$

$p(z|x.y)$
$||$

$$p(x,z\,|\,y) = \frac{p(x,y,z)}{p(y)} \overset{(a)}{=} \frac{p(x,y)\,p(z\,|\,y)}{p(y)} = p(x\,|\,y)\,p(z\,|\,y)$$

given y. x.z are independent.

(a) $p(z\,|\,x,y) = p(z\,|\,y)$

Hence x and z are conditionally independent given y

$p(z|y) = \frac{p(y.z)}{p(y)}$

Also $x \to y \to z$ iff $z \to y \to x$ since

$p(x|y)p(z|y) = p(x.z|y)$

$$p(x\,|\,y) = p(x\,|\,y)\frac{p(z\,|\,y)\,p(y)}{p(y,z)} \overset{(a)}{=} \frac{p(x,z\,|\,y)\,p(y)}{p(y,z)} = \frac{p(x,y,z)}{p(y,z)}$$

$$= p(x\,|\,y,z)$$

(a) $p(x,z\,|\,y) = p(x\,|\,y)\,p(z\,|\,y)$

Conditionally indep.

given y, z does not provide additional infomation

Markov chain property is symmetrical

Data Processing Theorem

If $x \rightarrow y \rightarrow z$ then $\boxed{I(x\,;y) \geq I(x\,;\,z)}$ extract useful info do not lose mutual info

(even decrease)

– processing y cannot add new information about x

If $x \rightarrow y \rightarrow z$ then $\boxed{I(x\,;y) \geq I(x\,;\,y \mid z)}$

– Knowing z does not increase the amount y tells you about x

(even less)

Proof:

Apply chain rule in different ways

$$I(x\,;y,z) = I(x\,;y) + \underbrace{I(x\,;z \mid y)}_{=0} = I(x\,;z) + I(x\,;y \mid z)$$

(a)

$x \to y$: observation

$y \to z$: data processing

but $\quad I(x\,;z \mid y) = 0$

hence $\quad I(x\,;y) = I(x\,;z) + I(x\,;y \mid z)$

so $\quad I(x\,;y) \geq I(x\,;z)$ and $\quad I(x\,;y) \geq I(x\,;y \mid z)$

(a) $I(x\,;z)=0$ iff x and z are independent; Markov $\Rightarrow p(x,z \mid y)=p(x \mid y)p(z \mid y)$

So Why Processing?

- One can not create information by manipulating the data
- But no information is lost if equality holds
- Sufficient statistic
  - z contains all the information in y about x
  - Preserves mutual information $I(x;y) = I(x;z)$
- The estimator should be designed in a way such that it outputs sufficient statistics
- Can the estimation be arbitrarily accurate?

- what if $|X| \to \inf$?

Fano's Inequality

P_e has a lower bound.

If we estimate x from y, what is $p_e = p(\hat{x} \neq x)$?

$$x \to y \xrightarrow{H(P_e)} \hat{x}$$

$$H(x \mid y) \le H(p_e) + p_e \log |\mathcal{X}|$$

$H(P_e) = -P_e \log P_e - (1-P_e)\log(1-P_e) \le 1$

$= 1$ when uniform dist.

$$\Rightarrow p_e \ge \frac{(H(x \mid y) - H(p_e))}{\log |\mathcal{X}|} \overset{(a)}{\ge} \frac{(H(x \mid y) - 1)}{\log |\mathcal{X}|}$$

(a) the second form is weaker but easier to use

$P_e \sim H(X|Y)$

Proof: Define a random variable $e = \begin{cases} 1 & \hat{x} \neq x \ (P_e) \\ 0 & \hat{x} = x \ (1-P_e) \end{cases}$

$$H(e, x \mid \hat{x}) = H(x \mid \hat{x}) + \underline{H(e \mid x, \hat{x})} = H(e \mid \hat{x}) + H(x \mid e, \hat{x})$$ chain rule

no entropy if $x, \hat{x}$ are known

$$\Rightarrow H(x \mid \hat{x}) + 0 \le H(e) + H(x \mid e, \hat{x})$$ remove restraint $H \ge 0$; $H(e|y) \le H(e)$

$$= H(e) + \underline{H(x \mid \hat{x}, e=0)}(1 - p_e) + \underline{H(x \mid \hat{x}, e=1)}\, p_e$$

no entropy if no error at all ($x = \hat{x}$)

$x \neq \hat{x}$. still uncertainty $= \log(|X|-1)$ (-1: at least 1 error)

$$\le H(p_e) + 0 \times (1 - p_e) + p_e \log |\mathcal{X}|$$ $H(e) = H(p_e)$

$e \to x$

$$\underline{H(x \mid y) \le H(x \mid \hat{x}) \text{ since } I(x; \hat{x}) \le I(x; y)}$$ Markov chain

Implications

$P_e \geq \frac{H(Y|X) - H(P_e)}{\log(|X|-1)} \geq \frac{H(Y|X)-1}{\log|X|}$

- Zero probability of error $p_e = 0 \Rightarrow H(x \mid y) = 0$
- Low probability of error if $H(x|y)$ is small
- If $H(x|y)$ is large then the probability of error is high
- Could be slightly strengthened to

$$H(x \mid y) \leq H(p_e) + p_e \log(|\mathrm{X}| - 1)$$

- Fano's inequality is used whenever you need to show that errors are inevitable
  - E.g., Converse to channel coding theorem

MAP (maximum a posteriori prob)

$x \to y \to \hat{x}$

$\hat{x} = \arg\max_x P(x|y) = \begin{cases} 1, & y=1 \\ 2, & y=2 \end{cases}$

x\y	1	2
1	0.3	0.05
2	0.05	0.3
3	0.05	0.05
4	0.05	0.05
5	0.05	0.05

(select largest prob. in col.)

choose $\hat{x} = y$

$P_e = 1 - 0.6 = 0.4$

Fano Example

X = {1:5}, $\mathbf{p}_X = [0.35, 0.35, 0.1, 0.1, 0.1]^T$

Y = {1:2} if $x \le 2$ then $y = x$ with probability 6/7 while if $x > 2$ then $y = 1$ or 2 with equal prob.

Our best strategy is to guess $\hat{x} = y \quad (x \to y \to \hat{x})$

- $\mathbf{p}_{x|y=1} = [0.6, 0.1, 0.1, 0.1, 0.1]^T$
- actual error prob: $p_e = 0.4$

$H(y|x) = -\sum_{x,y} P(x,y)\log_2 P(y|x) = -0.3\log_2\frac{6}{7} \times 2 - 0.05\log_2\frac{1}{7} \times 2 - 0.05\log_2\frac{1}{2} \times 6 = 0.714$ bits

$H(x|y) = -\sum_{x,y} P(x,y)\log_2 P(x|y) = -0.3\log_2 0.6 \times 2 - 0.05\log_2 0.1 \times 8 = 1.771$ bits

Fano bound: $p_e \ge \dfrac{H(x|y) - 1}{\log(|X| - 1)} = \dfrac{1.771 - 1}{\log(4)} = 0.3855$ (exercise)

Main use: to show when error free transmission is impossible since $p_e > 0$

Summary

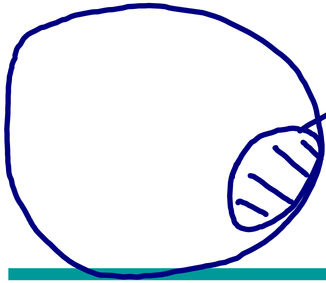
- Markov: $x \to y \to z \Leftrightarrow p(z \mid x, y) = p(z \mid y) \Leftrightarrow I(x; z \mid y) = 0$
- Data Processing Theorem: if $x \to y \to z$ then
  - $I(x\,;\,y) \ge I(x\,;\,z),\ I(y\,;\,z) \ge I(x\,;\,z)$
  - $I(x\,;\,y) \ge I(x\,;\,y \mid z)$ can be false if not Markov
  - Long Markov chains: If $x_1 \to x_2 \to x_3 \to x_4 \to x_5 \to x_6,$ then Mutual Information increases as you get closer together:
    - e.g. $I(x_3, x_4) \ge I(x_2, x_4) \ge I(x_1, x_5) \ge I(x_1, x_6)$
- Fano's Inequality: if $x \to y \to \hat{x}$ then

$$p_e \ge \frac{H(x \mid y) - H(p_e)}{\log(|\mathcal{X}| - 1)} \ge \frac{H(x \mid y) - 1}{\log(|\mathcal{X}| - 1)} \ge \frac{H(x \mid y) - 1}{\log |\mathcal{X}|}$$

weaker but easier to use since independent of p_e

Lecture 7

- Law of Large Numbers
  - Sample mean is close to expected value
- Asymptotic Equipartition Principle (AEP)
  - $-\log P(x_1, x_2, \ldots, x_n)/n$ is close to entropy H
- The Typical Set
  - Probability of each sequence close to 2^{-nH}
  - Size ($\sim 2^{nH}$) and total probability (~ 1)
- The Atypical Set
  - Unimportant and could be ignored



Typicality: Example

typical: $\log p(x) = nH(x)$

not typical: $\log p(x) \neq nH(p)$

X = {a, b, c, d}, $\mathbf{p} = [0.5\ 0.25\ 0.125\ 0.125]$

$-\log \mathbf{p} = [1\ 2\ 3\ 3] \Rightarrow H(\mathbf{p}) = 1.75$ bits

Sample eight i.i.d. values

- typical $\Rightarrow$ correct proportions

  adbabaac $-\log p(\mathbf{x}) = 14 = 8\times1.75 = nH(\mathbf{x})$

- not typical $\Rightarrow \log p(\mathbf{x}) \neq nH(\mathbf{x})$

  dddddddd $-\log p(\mathbf{x}) = 24$

Convergence of Random Variables

- Convergence

$$x_n \xrightarrow[n\to\infty]{} y \quad \Rightarrow \quad \forall \varepsilon > 0, \exists m \text{ such that } \quad \forall n > m, \underline{|x_n - y| < \varepsilon}$$

Example: $x_n = \pm 2^{-n}, \quad y = 0$

choose $m = -\log \varepsilon$

- Convergence in probability (weaker than convergence)

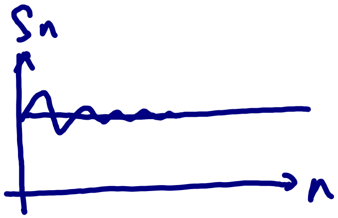
$$x_n \xrightarrow{\text{prob}} y \quad \Rightarrow \quad \forall \varepsilon > 0, \quad \underline{P\left(|x_n - y| > \varepsilon\right) \to 0}$$

Example: $x_n \in \{0;1\}, \quad p = [1 - n^{-1}; n^{-1}]$

for any small ε , $p(|x_n| > \varepsilon) = n^{-1} \xrightarrow{n\to\infty} 0$

so $x_n \xrightarrow{\text{prob}} 0$ (but $x_n \not\to 0$)

Note: y can be a constant or another random variable



Law of Large Numbers

Given i.i.d. $\{x_i\}$, sample mean $s_n = \frac{1}{n}\sum_{i=1}^{n} x_i$

– $E\ s_n = E\ x = \mu \qquad \mathrm{Var}\ s_n = n^{-1}\,\mathrm{Var}\ x = n^{-1}\sigma^2$

As n increases, $\mathrm{Var}\ s_n$ gets smaller and the values become clustered around the mean

LLN: $$s_n \xrightarrow{\text{prob}} \mu$$

$$\Leftrightarrow \quad \forall \varepsilon > 0, \quad P(|s_n - \mu| > \varepsilon) \xrightarrow[n\to\infty]{} 0$$

The expected value of a random variable is equal to the long-term average when sampling repeatedly.

Asymptotic Equipartition Principle

- $\mathbf{x}$ is the i.i.d. sequence $\{x_i\}$ for $1 \le i \le n$
  - Prob of a particular sequence is $p(\mathbf{x}) = \prod_{i=1}^{n} p(x_i)$
  - Average $E - \log p(\mathbf{x}) = n\, E - \log p(x_i) = nH(x)$
- AEP: average SIC of a certain content

$$-\frac{1}{n}\log p(\mathbf{x}) \xrightarrow{\text{prob}} H(x)$$

deterministic

- Proof:

$$-\frac{1}{n}\log p(\mathbf{x}) = -\frac{1}{n}\sum_{i=1}^{n} \log p(x_i)$$

long-term average

n↑

mean

law of large numbers $\xrightarrow{\text{prob}} E - \log p(x_i) = H(x)$

N=1   log 0.2
      log 0.8

Typical Set

Typical set (for finite n)

$|-\frac{1}{n}\log P(x) - H(X)| \le \varepsilon$

$|-\log P(x) - nH(X)| \le n\varepsilon$

$\log P(x) = nH(X) \pm n\varepsilon$

size   n: { sequence length
          # i.i.d.

$$T_\varepsilon^{(n)} = \left\{\mathbf{x} \in \mathcal{X}^n : \left| -n^{-1}\log\ p(\mathbf{x}) - H(X)\right| < \varepsilon\right\}$$

← deviation

typical set: average SIC close to the entropy divided by n.

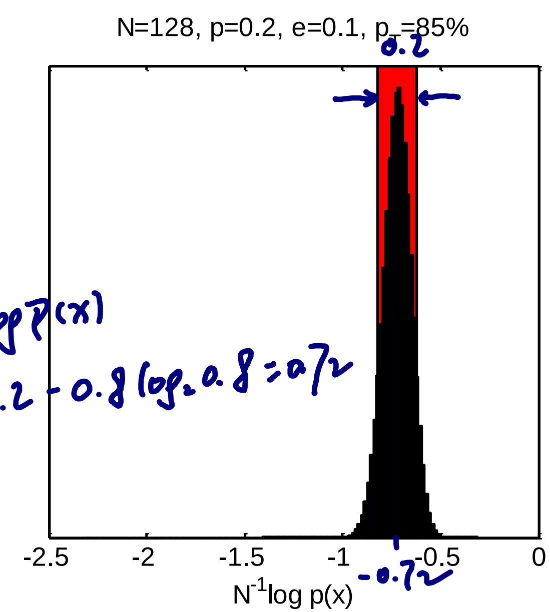
$n \nearrow \Rightarrow$ typicality $\nearrow$

Example:

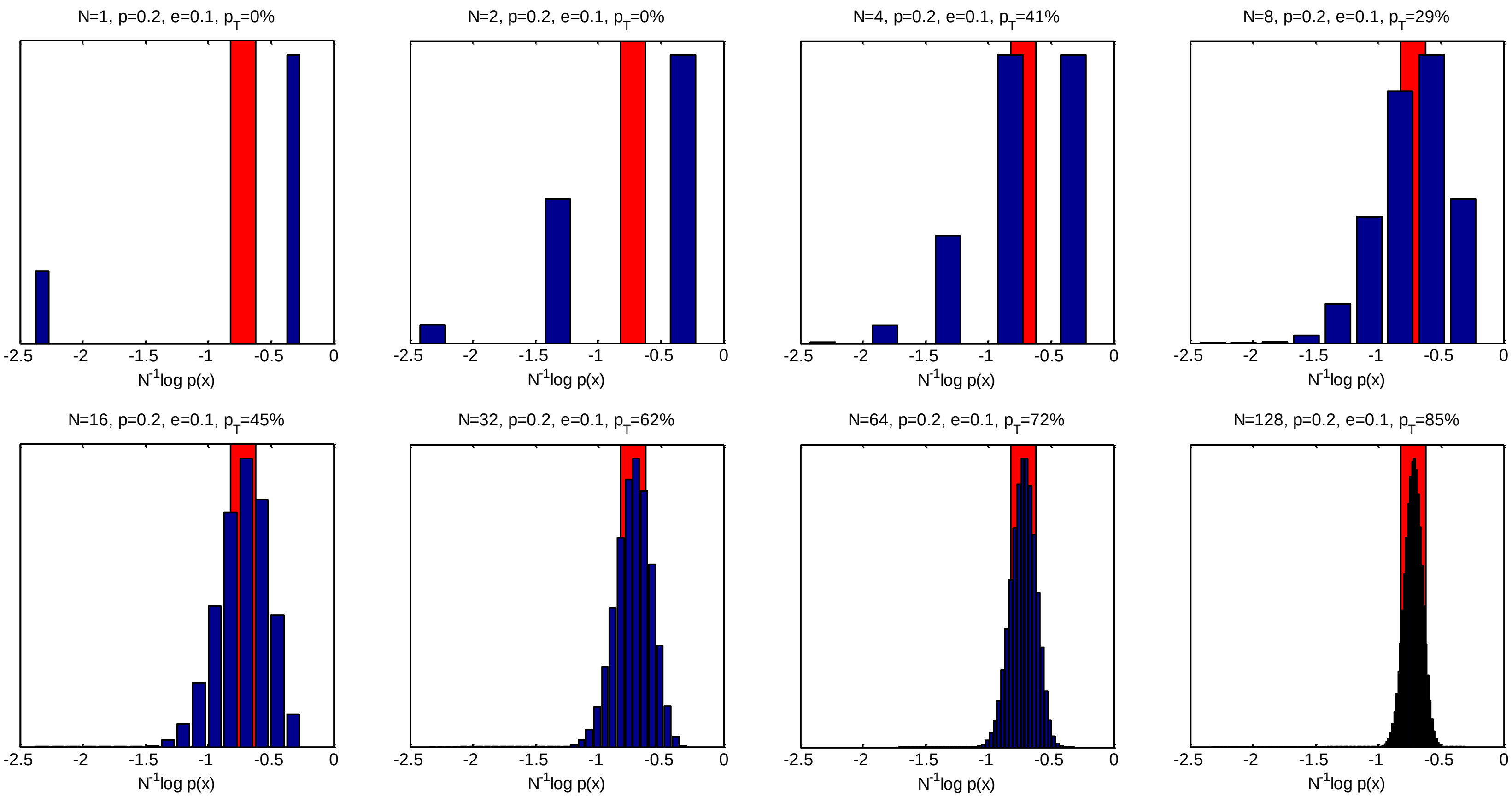
- x_i Bernoulli with $p(x_i=1)=p$
- *e.g.* $p([0\ 1\ 1\ 0\ 0\ 0])=p^2(1-p)^4$
- For $p=0.2$, $H(X)=0.72$ bits
- Red bar shows $T_{0.1}^{(n)}$

$H(X) = -\sum_x P(x)\log P(x)$

$= -0.2\log_2 0.2 - 0.8\log_2 0.8 = 0.72$



Typical Set Frames



00100000000000, 0010100, 00, 0100000010000,
0000100001000000, 0000000010000000

Typical Set: Properties

$$-\frac{1}{n}\log P(\bar{x}) = \frac{1}{n}\sum_{i=1}^{n} -\log P(x_i) \xrightarrow{\text{prob}} E(-\log P(x_i)) = H(X)$$

$$\forall n > N_\varepsilon,\ P\left[\left|-\frac{1}{n}\log P(\bar{x}) - H(X)\right| > \varepsilon\right] < \varepsilon \Rightarrow P(\bar{x} \in T_\varepsilon^{(n)}) > 1-\varepsilon \text{ for } n > N_\varepsilon$$

$$\underset{1-\varepsilon}{\overset{\vee}{P(\bar{x} \in T_\varepsilon^{(n)})}} \xrightarrow[P(x) \le 2^{-nH(X)+n\varepsilon}]{\log P(x) \le -nH(X)+n\varepsilon} \le \sum_{\bar{x} \in T_\varepsilon^{(n)}} 2^{-n(H(X)-\varepsilon)} = 2^{-n(H(X)-\varepsilon)} |T_\varepsilon^{(n)}|$$

$$-nH(X)+n\varepsilon \ge \log P(x) \ge -nH(X)-n\varepsilon$$

$$\mathbf{x} \in T_\varepsilon^{(n)} \Rightarrow \log\ p(\mathbf{x}) = -nH(X) \pm n\varepsilon$$

$$\underset{1}{\overset{\wedge}{P(\bar{x} \in T_\varepsilon^{(n)})}} \xrightarrow[P(x) \ge 2^{-nH(X)-n\varepsilon}]{\log P(x) \ge -nH(X)-n\varepsilon} \ge \sum_{\bar{x} \in T_\varepsilon^{(n)}} 2^{-n(H(X)+\varepsilon)} = 2^{-n(H(X)+\varepsilon)} \cdot |T_\varepsilon^{(n)}|$$

$$\boxed{p(\mathbf{x} \in T_\varepsilon^{(n)}) > 1-\varepsilon \quad \text{for} \quad n > N_\varepsilon}$$

$$\boxed{(1-\varepsilon)2^{n(H(X)-\varepsilon)} \overset{n>N_\varepsilon}{<} \left|T_\varepsilon^{(n)}\right| \le 2^{n(H(X)+\varepsilon)}}$$

$$\therefore 2^{n(H(X)+\varepsilon)} \ge |T_\varepsilon^{(n)}| > (1-\varepsilon)2^{n(H(X)-\varepsilon)} \quad (\text{for } n > N_\varepsilon)$$

size $\approx 2^{nH(X)}$ # numbers in source coding

Proof 2:

$$-n^{-1}\log\ p(\mathbf{x}) = n^{-1}\sum_{i=1}^{n} -\log\ p(x_i) \xrightarrow{\text{prob}} E -\log\ p(x_i) = H(X)$$

Hence $\forall \varepsilon > 0\ \exists N_\varepsilon$ s.t. $\forall n > N_\varepsilon \quad p\left(\left|-n^{-1}\log\ p(\mathbf{x}) - H(X)\right| > \varepsilon\right) < \varepsilon$

Proof 3a:

$$\text{f.l.e. } n, \quad 1-\varepsilon < p(\mathbf{x} \in T_\varepsilon^{(n)}) \overset{\log P(x) \le -nH(X)+n\varepsilon}{\le} \sum_{\mathbf{x} \in T_\varepsilon^{(n)}} 2^{-n(H(X)-\varepsilon)} = 2^{-n(H(X)-\varepsilon)} \overset{\text{size}}{\left|T_\varepsilon^{(n)}\right|}$$

Proof 3b:

$$1 = \sum_{\mathbf{x}} p(\mathbf{x}) \ge \sum_{\mathbf{x} \in T_\varepsilon^{(n)}} p(\mathbf{x}) \overset{\log P(x) \ge -nH(X)-n\varepsilon}{\ge} \sum_{\mathbf{x} \in T_\varepsilon^{(n)}} 2^{-n(H(X)+\varepsilon)} = 2^{-n(H(X)+\varepsilon)} \left|T_\varepsilon^{(n)}\right|$$

Consequence

- for any ε and for $n > N_\varepsilon$
  "Almost all events are almost equally surprising"

- $p(\mathbf{x} \in T_\varepsilon^{(n)}) > 1 - \varepsilon$ and $\log p(\mathbf{x}) = -nH(X) \pm n\varepsilon$

$2^{n(H(X)+\varepsilon)} \geq |T_\varepsilon^{(n)}| > (1-\varepsilon)\, 2^{n(H(X)-\varepsilon)}$

Coding consequence

$\geq \lceil n(H+\varepsilon) \rceil$

- $\mathbf{x} \in T_\varepsilon^{(n)}$: '0' + at most 1+$n(H+\varepsilon)$ bits
  typical
- $\mathbf{x} \notin T_\varepsilon^{(n)}$: '1' + at most 1+$n\log|X|$ bits
  not typical   small prob.   (uniform)
- L = Average code length

$$\leq p(\mathbf{x} \in T_\varepsilon^{(n)})[2 + n(H+\varepsilon)]$$

$2 + n(H+\varepsilon)$

$+\ \varepsilon[2 + n\log|X|]$

$$+\, p(\mathbf{x} \notin T_\varepsilon^{(n)})[2 + n\log|X|]$$

$= 2 + nH + n\varepsilon + 2\varepsilon + n\varepsilon\log|X|$

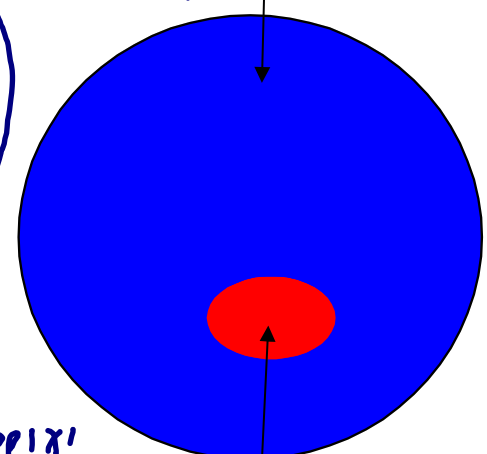
$1 \cdot$ 1st term $+ \varepsilon \cdot$ 2nd term

$= n\left(H + \varepsilon + \varepsilon\log|X| + \frac{2}{n}(1+\varepsilon)\right)$

$$\leq n(H+\varepsilon) + \varepsilon\left(n\log|X|\right) + 2\varepsilon + 2$$

$= n(H+\varepsilon')$

$$= n\left(H + \varepsilon + \varepsilon\log|X| + 2(\varepsilon+2)n^{-1}\right) = n(H+\varepsilon')$$



Source Coding & Data Compression

For any choice of $\varepsilon > 0$, we can, by choosing block size, n, large enough, do the following:

$n(H+\varepsilon') \approx L \Rightarrow H+\varepsilon' \approx \frac{L}{n}$ (expectation of typical codelength $\approx nH$)

- make a lossless code using only $H(X)+\varepsilon$ bits per symbol on average:

typical avg. code length

$$\frac{L}{n} \leq \underbrace{H + \varepsilon}_{\text{fixed}}$$

block size – n (# i.i.d bits)

n small: lossy (not typical)
n large: complex

express X^n by nH bits.

- The coding is one-to-one and decodable
  - However impractical due to exponential complexity
- Typical sequences have short descriptions of length $\approx nH$
  - Another proof of source coding theorem (Shannon's original proof)
- However, encoding/decoding complexity is exponential in n

Smallest high-probability Set

$T_\varepsilon^{(n)}$ is a small subset of X^n containing most of the probability mass. Can you get even smaller ?

For any $0 < \varepsilon < 1$, choose $N_0 = -\varepsilon^{-1}\log \varepsilon$, then for any $n > \max(N_0, N_\varepsilon)$ and any subset $S^{(n)}$ satisfying $\left|S^{(n)}\right| < 2^{n(H(X)-2\varepsilon)}$

$$p\left(\mathbf{x} \in S^{(n)}\right) = p\left(\mathbf{x} \in S^{(n)} \cap T_\varepsilon^{(n)}\right) + p\left(\mathbf{x} \in S^{(n)} \cap \overline{T_\varepsilon^{(n)}}\right)$$

$$< \left|S^{(n)}\right| \max_{\mathbf{x} \in T_\varepsilon^{(n)}} p(\mathbf{x}) + p\left(\mathbf{x} \in \overline{T_\varepsilon^{(n)}}\right)$$

$$< 2^{n(H-2\varepsilon)} 2^{-n(H-\varepsilon)} + \varepsilon \qquad \text{for } n > N_\varepsilon$$

$$= 2^{-n\varepsilon} + \varepsilon < 2\varepsilon \qquad \text{for } n > N_{0,} \quad 2^{-n\varepsilon} < 2^{\log \varepsilon} = \varepsilon$$

Answer: No

Summary

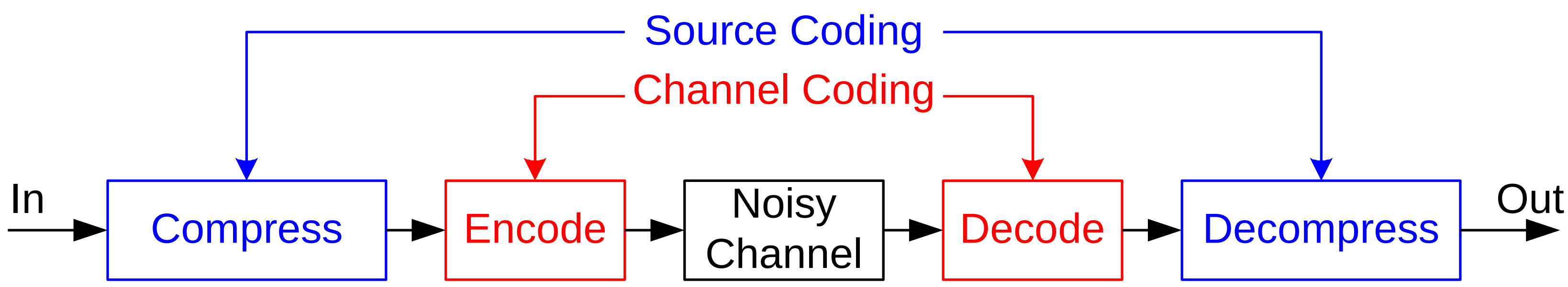
- Typical Set
  - Individual Prob $\mathbf{x} \in T_{\varepsilon}^{(n)} \Rightarrow \log\ p(\mathbf{x}) = -nH(x) \pm n\varepsilon$
  - Total Prob $p(\mathbf{x} \in T_{\varepsilon}^{(n)}) > 1-\varepsilon \quad \text{for} \quad n > N_{\varepsilon}$
  - Size $(1-\varepsilon)2^{n(H(x)-\varepsilon)} \overset{n>N_{\varepsilon}}{<} \left|T_{\varepsilon}^{(n)}\right| \leq 2^{n(H(x)+\varepsilon)}$
- No other high probability set can be much smaller than $T_{\varepsilon}^{(n)}$
- Asymptotic Equipartition Principle
  - Almost all event sequences are equally surprising
- Can be used to prove source coding theorem

Lecture 8

- Channel Coding
- Channel Capacity
  - The highest rate in bits per channel use that can be transmitted reliably ♦
  - The maximum mutual information
- Discrete Memoryless Channels
  - Symmetric Channels
  - Channel capacity
    - Binary Symmetric Channel
    - Binary Erasure Channel
    - Asymmetric Channel

♦ = proved in channel coding theorem

Model of Digital Communication

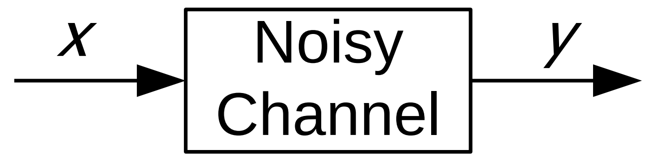


- Source Coding
  - Compresses the data to remove redundancy
- Channel Coding
  - Adds redundancy/structure to protect against channel errors

Discrete Memoryless Channel

(i/o discrete)

- Input: $x \in$ X, Output $y \in$ Y



- Time-Invariant Transition-Probability Matrix

$$\left(\mathbf{Q}_{y|x}\right)_{i,j} = p\left(y = y_j \mid x = x_i\right)$$

  – Hence $\mathbf{p}_y = \mathbf{Q}_{y|x}^T \mathbf{p}_x$

  – $\mathbf{Q}$: each row sum = 1, average column sum = $|X||Y|^{-1}$

- Memoryless: $\mathbf{p}(y_n | x_{1:n}, y_{1:n-1}) = \mathbf{p}(y_n | x_n)$

current output
$\updownarrow$
current input

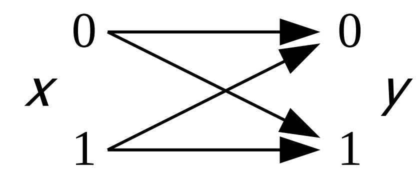
- DMC = Discrete Memoryless Channel

Binary Channels

- Binary Symmetric Channel
  (BSC)
  – X = [0 1], Y = [0 1]

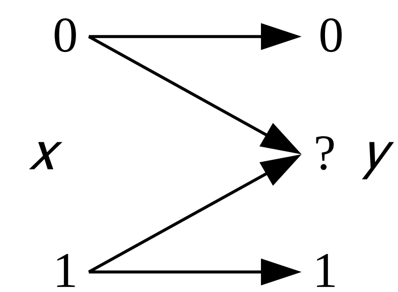
$$\begin{pmatrix} 1-f & f \\ f & 1-f \end{pmatrix} S$$

f : error prob.
1-f : correct prob.



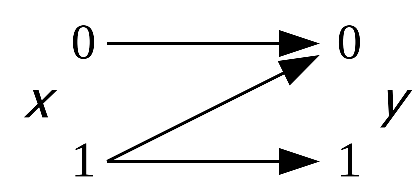
- Binary Erasure Channel
  (BEC)
  – X = [0 1], Y = [0 ? 1]

$$\begin{array}{c|ccc} x \backslash y & 0 & ? & 1 \\ \hline 0 & & & \\ 1 & & & \end{array}\quad \begin{pmatrix} 1-f & f & 0 \\ 0 & f & 1-f \end{pmatrix}$$



- Z Channel
  – X = [0 1], Y = [0 1]

$$\begin{array}{c|cc} x \backslash y & 0 & 1 \\ \hline 0 & & \\ 1 & & \end{array}\quad \begin{pmatrix} 1 & 0 \\ f & 1-f \end{pmatrix}$$



Symmetric: rows are permutations of each other; columns are permutations of each other

$\Sigma = \frac{2}{3}$ (same sum)

Weakly Symmetric: rows are permutations of each other; columns have the same sum

WS $\begin{pmatrix} \frac{1}{3} & \frac{1}{6} & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & \frac{1}{6} \end{pmatrix}$ $\Sigma = 1$ (def.)

Capacity $\begin{cases} S \\ WS \end{cases}$

Weakly Symmetric Channels

x\Y

Weakly Symmetric: Q =

1. All columns of $\mathbf{Q}$ have the same sum = $|\mathsf{X}||\mathsf{Y}|^{-1}$
   - If x is uniform (i.e. $p(x) = |\mathsf{X}|^{-1}$) then y is uniform

Column

$$p(y) = \sum_{x \in X} p(y \mid x)\, p(x) = |\mathsf{X}|^{-1} \sum_{x \in X} p(y \mid x) = |\mathsf{X}|^{-1} \times |\mathsf{X}||\mathsf{Y}|^{-1} = |\mathsf{Y}|^{-1}$$

2. All rows are permutations of each other
   - Each row of Q has the same entropy so

$$H(y \mid x) = \sum_{x \in \mathsf{X}} p(x) H(y \mid x = x) = H(\mathbf{Q}_{1,:}) \sum_{x \in \mathsf{X}} p(x) = H(\mathbf{Q}_{1,:})$$

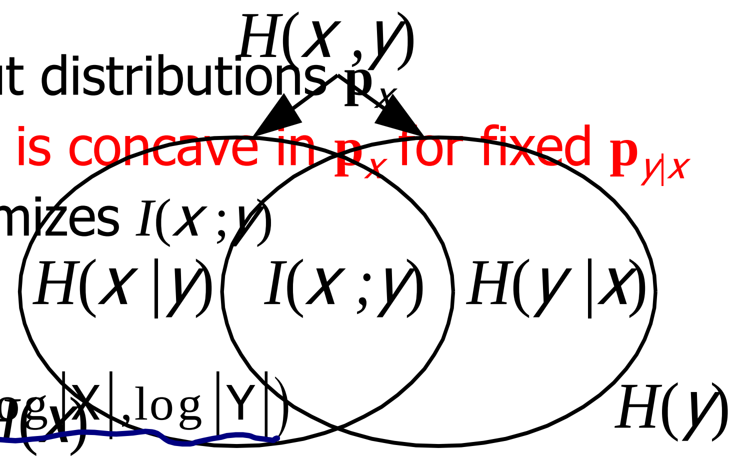
where $\mathbf{Q}_{1,:}$ is the entropy of the first (or any other) row of the $\mathbf{Q}$ matrix

Symmetric: 1. All rows are permutations of each other
2. All columns are permutations of each other
Symmetric $\Rightarrow$ weakly symmetric

Channel Capacity

- Capacity of a DMC channel: $\boxed{C = \max_{\mathbf{p}_x} I(x;y)}$
  - Mutual information (not entropy itself) is what could be transmitted through the channel
  - Maximum is over all possible input distributions $\mathbf{p}_x$ ♦
  - $\exists$ only one maximum since $I(x;y)$ is concave in $\mathbf{p}_x$ for fixed $\mathbf{p}_{y|x}$
  - We want to find the $\mathbf{p}_x$ that maximizes $I(x;y)$
  - Limits on C:

$$0 \le C \le \min\left(H(x), H(y)\right) \le \min\left(\log|X|, \log|Y|\right)$$

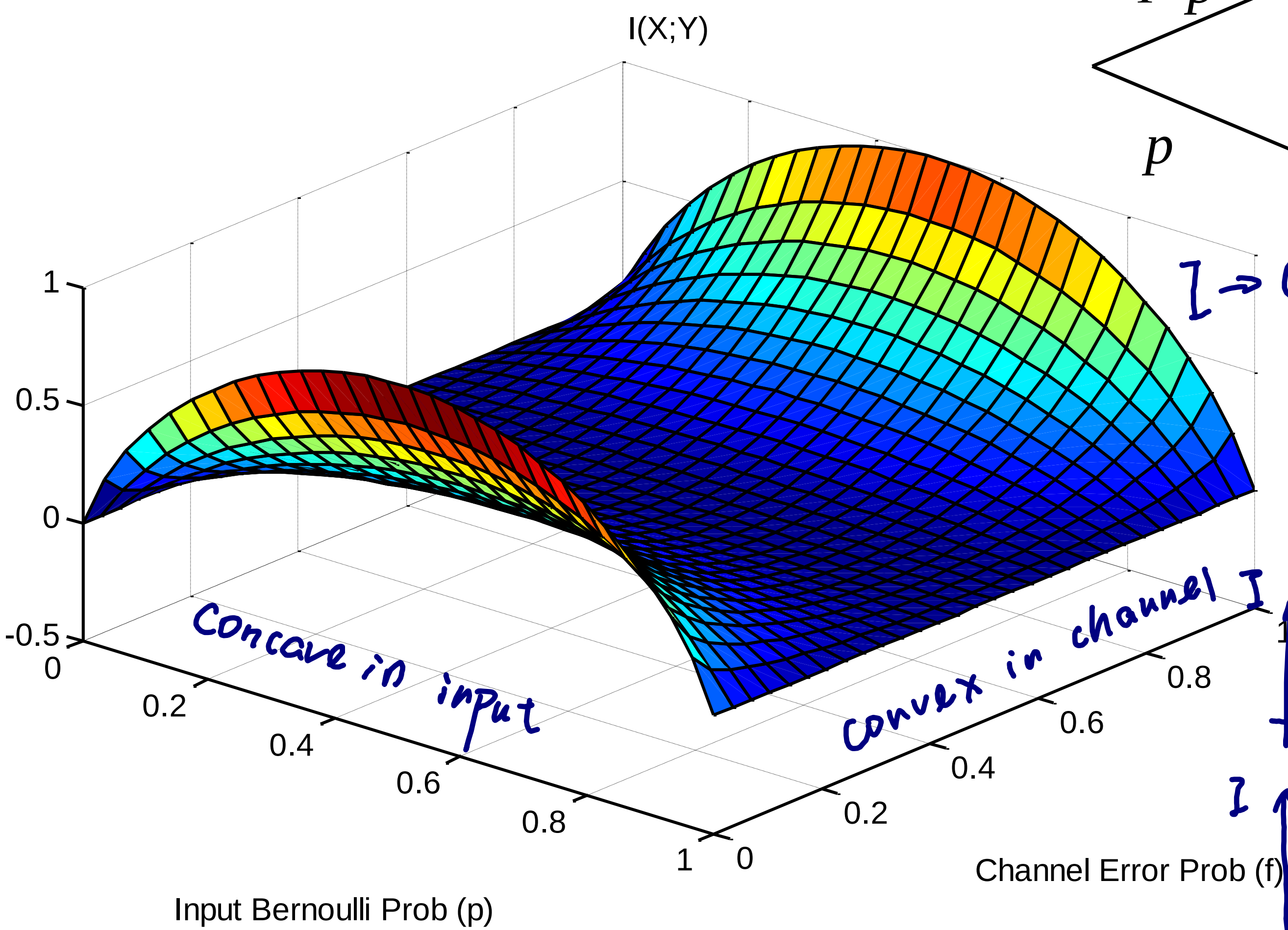


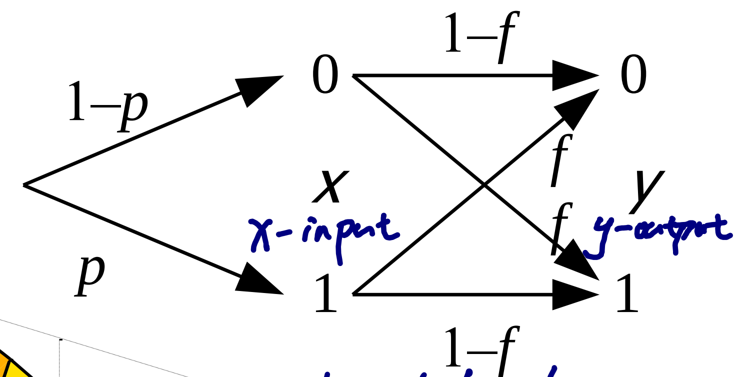
- Capacity for n uses of channel:

$$C^{(n)} = \frac{1}{n} \max_{\mathbf{p}_{x_{1:n}}} I(x_{1:n}; y_{1:n})$$

♦ = proved in two pages time

Mutual Information Plot





$I \to C$ | channel fixed, input uniform

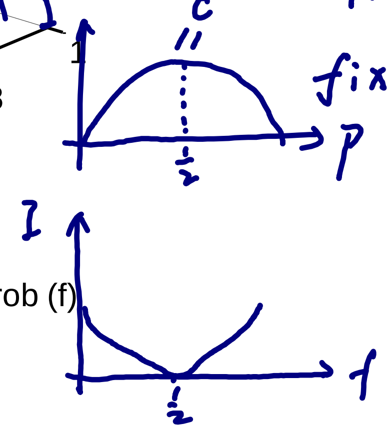
$H(Y|X)$

$pf+p)-H(f)$

$I(X;Y) = H(f-2pf+p) - H(f)$

$= C$

fix f



$I(x,z;y) = I(x;y) + I(z;y|x) = I(z;y) + I(x;y|z)$

$I(z;y|x) = H(y|x) - H(y|x,z) \overset{\text{fixed } P_{y|x}}{=} 0$

Mutual Information Concave in $\mathbf{p}_X$

$\therefore I(x;y) \geq I(x;y|z)$

$= \lambda I(x;y|z=1) + (1-\lambda) I(x;y|z=0)$

$= \lambda I(u;y) + (1-\lambda) I(v;y)$

Mutual Information $I(x;y)$ is concave in $\mathbf{p}_X$ for fixed $\mathbf{p}_{y|x}$

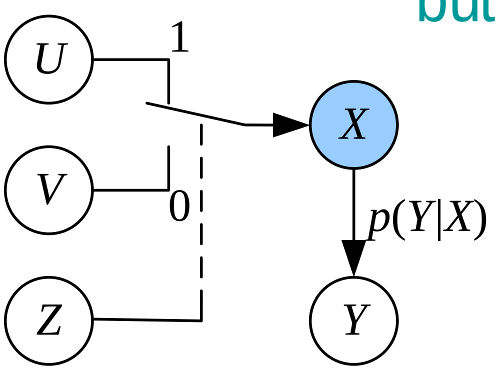
(input)

Proof: Let *u* and *v* have prob mass vectors $\mathbf{p_u}$ and $\mathbf{p_v}$

x: input
y: output

- Define *z*: bernoulli random variable with $p(1) = \lambda$
- Let $x = u$ if $z=1$ and $x=v$ if $z=0$ $\Rightarrow$ $\mathbf{p}_X = \lambda\,\mathbf{p_u} + (1-\lambda)\,\mathbf{p_v}$

$$I(x,z;y) = I(x;y) + I(z;y\,|\,x) = I(z;y) + I(x;y\,|\,z)$$



but $I(z;y\,|\,x) = H(y\,|\,x) - H(y\,|\,x,z) = 0$ so

$$I(x;y) \geq I(x;y\,|\,z)$$

$$= \lambda I(x;y\,|\,z=1) + (1-\lambda) I(x;y\,|\,z=0)$$

$$= \lambda I(u;y) + (1-\lambda) I(v;y)$$

Special Case: $y=x \Rightarrow I(x;x)=H(x)$ is concave in $\mathbf{p}_X$

$I(X;Y,Z) = I(X;Y) + I(X;Y|Z) = I(X;Z) + I(X;Z|Y)$

$I(X;Z) \overset{\text{fixed } P_X}{=} 0$

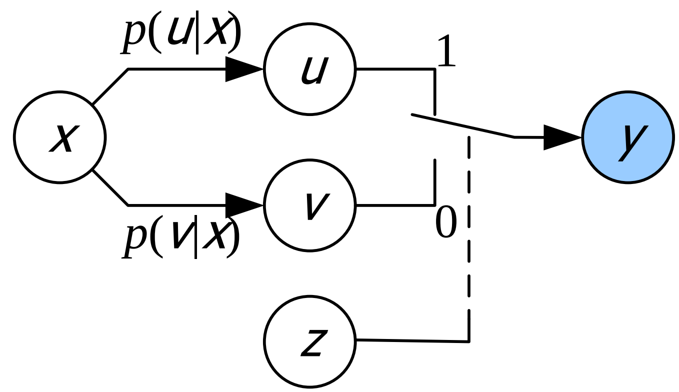
Mutual Information Convex in $\mathbf{p}_{Y|X}$

$I(X;Y) \le I(X;Z|Y)$

$= \lambda I(X;Y|Z=1) + (1-\lambda) I(X;Y|Z=0) = \lambda I(X;U) + (1-\lambda) I(X;V)$

Mutual Information $I(x\,;y)$ is convex in $\mathbf{p}_{y|x}$ for fixed $\mathbf{p}_x$
(channel)

Proof: define u, v, x etc:



– $$\mathbf{p}_{y|x} = \lambda \mathbf{p}_{u|x} + (1-\lambda)\mathbf{p}_{v|x}$$

$$I(x;y,z) = I(x;y\,|\,z) + I(x;z)$$

$$= I(x;y) + I(x;z\,|\,y)$$

but $I(x;z) = 0$ and $I(x;z\,|\,y) \ge 0$ so

$$I(x;y) \le I(x;y\,|\,z)$$

$$= \lambda I(x;y\,|\,z=1) + (1-\lambda) I(x;y\,|\,z=0)$$

$$= \lambda I(x;u) + (1-\lambda) I(x;v)$$

n-use Channel Capacity

For Discrete Memoryless Channel:

$$I(x_{1:n};y_{1:n}) = H(y_{1:n}) - H(y_{1:n} \mid x_{1:n})$$

$$= \sum_{i=1}^{n} H(y_i \mid y_{1:i-1}) - \sum_{i=1}^{n} H(y_i \mid x_i)$$

Chain; Memoryless

$$\le \sum_{i=1}^{n} H(y_i) - \sum_{i=1}^{n} H(y_i \mid x_i) = \sum_{i=1}^{n} I(x_i;y_i)$$

Conditioning Reduces Entropy

with equality if y_i are independent $\Rightarrow$ x_i are independent

We can maximize $I(\mathbf{x};\mathbf{y})$ by maximizing each $I(x_i;y_i)$ independently and taking x_i to be i.i.d.

- We will concentrate on maximizing $I(x; y)$ for a single channel use
- The elements of X_i are not necessarily i.i.d.

BSC $\begin{pmatrix} 1-f & f \\ f & 1-f \end{pmatrix}$ $I(X;Y) = H(Y) - H(Y|X)$

$= H(Y) - H(Q_{1,:}) \le \log|Y| - H(Q_{1,:})$

Capacity of Symmetric Channel

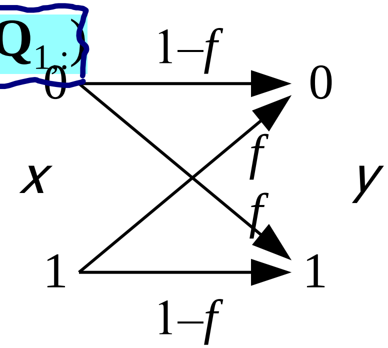
$I(X;Y) \le \log_2 2 - H(f) = \underbrace{1-H(f)}_{C}$

If channel is weakly symmetric: $I \to C$ $\begin{cases} \text{channel fixed} \\ \text{input uniform} \end{cases}$

$$I(x;y) = H(y) - \underbrace{H(y\,|\,x)}_{\text{row entropy}} = H(y) - \underline{H(\mathbf{Q}_{1,:})} \le \log|Y| - H(\mathbf{Q}_{1,:})$$

with equality iff input distribution is uniform

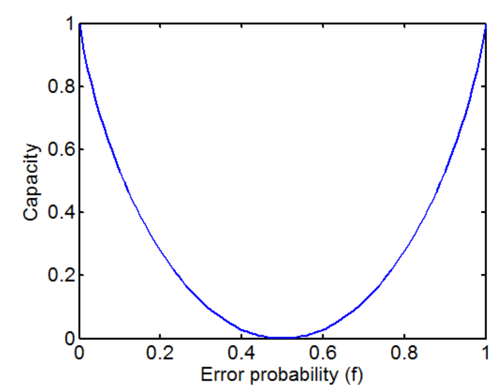
∴ Information Capacity of a WS channel is $\boxed{C = \log|Y| - H(\mathbf{Q}_{1,:})}$



For a binary symmetric channel (BSC): $\begin{pmatrix} 1-f & f \\ f & 1-f \end{pmatrix}$ $H(f)$

- $|Y| = 2$
- $H(\mathbf{Q}_{1,:}) = H(f)$
- $I(x;y) \le 1 - H(f)$

∴ Information Capacity of a BSC is $1 - H(f)$



BEC

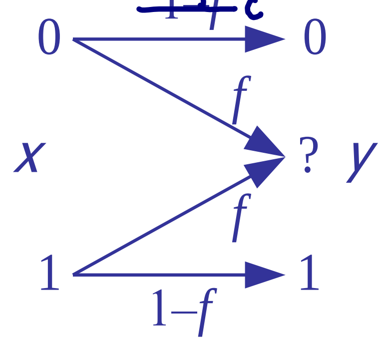
$x \backslash y$	0	?	1
0	$1-f$	f	0
1	0	f	$1-f$

$$I(x;y) = H(x) - H(x|y)$$
$$= H(x) - P(y=0)H(x|y=0) - P(y=?)H(x|y=?) - P(y=1)H(x|y=1)$$

Binary Erasure Channel (BEC)

$$= H(x) - P(y=0)\cdot 0 - P(y=?)H(x) - P(y=1)\cdot 0 = H(x)(1-f)$$
$$\le 1-f$$

$$\begin{array}{c|ccc} x \backslash y & 0 & ? & 1 \\ \hline 0 & & & \\ 1 & & & \end{array} \quad \begin{pmatrix} 1-f & f & 0 \; (H=0) \\ \underset{(H=0)}{0} & f & 1-f \end{pmatrix}$$



$$I(x;y) = H(x) - H(x \mid y)$$
$$= H(x) - p(y=0)\times 0 - p(y=?)H(x) - p(y=1)\times 0$$
$$= H(x) - H(x)f$$
$$= (1-f)H(x)$$
$$\le 1-f$$
$$C = 1-f$$

$H(x|y) = 0$ when $y=0$ or $y=1$

mutual info ⇒ capacity $\begin{cases} \le \\ = \text{ when uniform distribution} \end{cases}$

since max value of $H(x) = 1$
with equality when x is uniform

since a fraction f of the bits are lost, the capacity is only $1-f$ and this is achieved when x is uniform

$I(X;Y) = H(Y) - H(Y|X)$

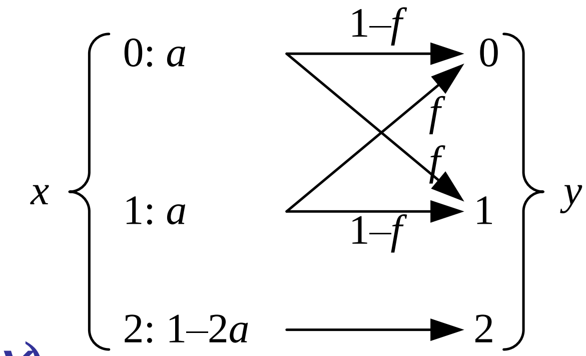
$H(Y) = 2\cdot(-a\log a) - (1-2a)\log(1-2a)$

$H(Y|X) = 2aH(f) + (1-2a)\cdot 0$

$$\begin{array}{cc|ccc} & x\backslash y & 0 & 1 & 2 \\ \hline a & 0 & 1-f & f & 0 \\ a & 1 & f & 1-f & 0 \\ 1-2a & 2 & 0 & 0 & 1 \end{array}$$

Asymmetric Channel Capacity

Let $\mathbf{p}_x = [a \;\; a \;\; 1-2a]^T \Rightarrow \mathbf{p}_y = \mathbf{Q}^T\mathbf{p}_x = \mathbf{p}_x$



$$H(y) = -2a\log a - (1-2a)\log(1-2a)$$

$$H(y\,|\,x) = 2aH(f) + (1-2a)H(1) = 2aH(f)$$

To find C, maximize $I(x;y) = H(y) - H(y\,|x)$

$$I = -2a\log a - (1-2a)\log(1-2a) - 2aH(f)$$

$$\frac{dI}{da} = -2\log e - 2\log a + 2\log e + 2\log(1-2a) - 2H(f) = 0$$

$$\mathbf{Q} = \begin{pmatrix} 1-f & f & 0 \\ f & 1-f & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\log\frac{1-2a}{a} = \log\left(a^{-1}-2\right) = H(f) \quad\Rightarrow\quad a = \left(2+2^{H(f)}\right)^{-1}$$

Note:
$d(\log x) = x^{-1}\log e$

$$\Rightarrow \quad C = -2a\log\left(a2^{H(f)}\right) - (1-2a)\log(1-2a) \;= -\log(1-2a)$$

Examples: $f = 0 \Rightarrow \mathrm{H}(f) = 0 \Rightarrow a = {}^1/_3 \Rightarrow C = \log 3 = 1.585$ bits/use
$f = ½ \Rightarrow \mathrm{H}(f) = 1 \Rightarrow a = ¼ \Rightarrow C = \log 2 = 1$ bits/use

Summary

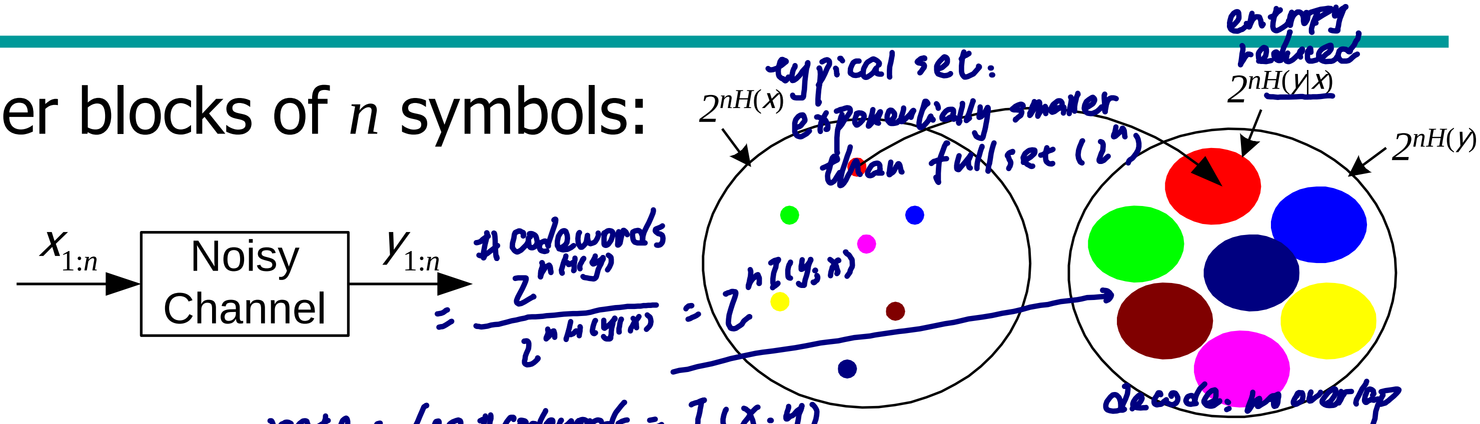
- Given the channel, mutual information is concave in input distribution
- Channel capacity $C = \max_{\mathbf{p}_x} I(x;y)$
  - The maximum exists and is unique
- DMC capacity
  - Weakly symmetric channel: $\log|Y| - H(\mathbf{Q}_{1,:})$
  - BSC: $1 - H(f)$
  - BEC: $1 - f$
  - In general it very hard to obtain closed-form; numerical method using convex optimization instead

Lecture 9

- Jointly Typical Sets
- Joint AEP
- Channel Coding Theorem
  - Ultimate limit on information transmission is channel capacity
  - The central and most successful story of information theory
  - Random Coding
  - Jointly typical decoding

Intuition on the Ultimate Limit

- Consider blocks of n symbols:



rate $= \frac{\log \# \text{codewords}}{\text{codeword length}} = I(x;y)$

  - For large n, an average input sequence $x_{1:n}$ corresponds to about $2^{nH(y|x)}$ typical output sequences
  - There are a total of $2^{nH(y)}$ typical output sequences
  - For nearly error free transmission, we select a number of input sequences whose corresponding sets of output sequences hardly overlap
  - The maximum number of distinct sets of output sequences is $2^{n(H(y)-H(y|x))} = 2^{nI(y\,;x)}$
  - One can send $I(y;x)$ bits per channel use

for large n can transmit at any rate < C with negligible errors

Jointly Typical Set

x,y is the i.i.d. sequence $\{x_i, y_i\}$ for $1 \le i \le n$

- Prob of a particular sequence is $p(\mathbf{x},\mathbf{y}) = \prod_{i=1}^{N} p(x_i, y_i)$
- $E - \log\ p(\mathbf{x},\mathbf{y}) = n\ E - \log\ p(x_i, y_i) = nH(X, Y)$
- Jointly Typical set: $\begin{cases} x \text{ typical} \\ y \text{ typical} \\ x.y \text{ typical} \end{cases}$

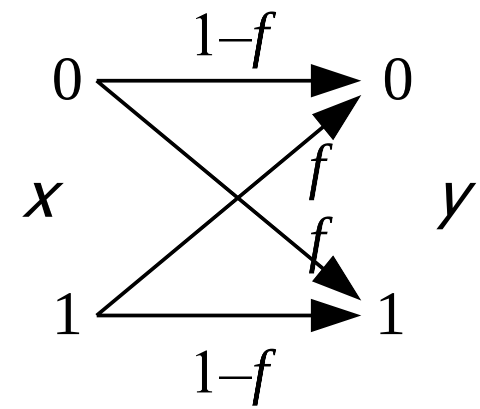
$$J_\varepsilon^{(n)} = \Big\{ \mathbf{x},\mathbf{y} \in \mathrm{XY}^n : \left| -n^{-1} \log p(\mathbf{x}) - H(X) \right| < \varepsilon,$$

$$\left| -n^{-1} \log p(\mathbf{y}) - H(Y) \right| < \varepsilon,$$

$$\left| -n^{-1} \log p(\mathbf{x},\mathbf{y}) - H(X, Y) \right| < \varepsilon \Big\}$$

Jointly Typical Example

Binary Symmetric Channel



$$f = 0.2, \quad \mathbf{p}_x = (0.75 \quad 0.25)^T$$

$$\mathbf{p}_y = (0.65 \quad 0.35)^T, \quad \mathbf{P}_{xy} = \begin{array}{c|c} x \backslash y & \begin{array}{cc} 0 \quad\quad\quad & \quad\quad\quad 1 \end{array} \\ \begin{array}{c} 0 \\ \\ 1 \end{array} & \begin{pmatrix} 0.6 & 0.15 \\ \scriptstyle 0.25\times0.8 & \scriptstyle 0.75\times0.2 \\ 0.05 & 0.2 \\ \scriptstyle 0.25\times0.2 & \scriptstyle 0.25\times0.8 \end{pmatrix} \end{array}$$

Jointly typical example (for any ε):

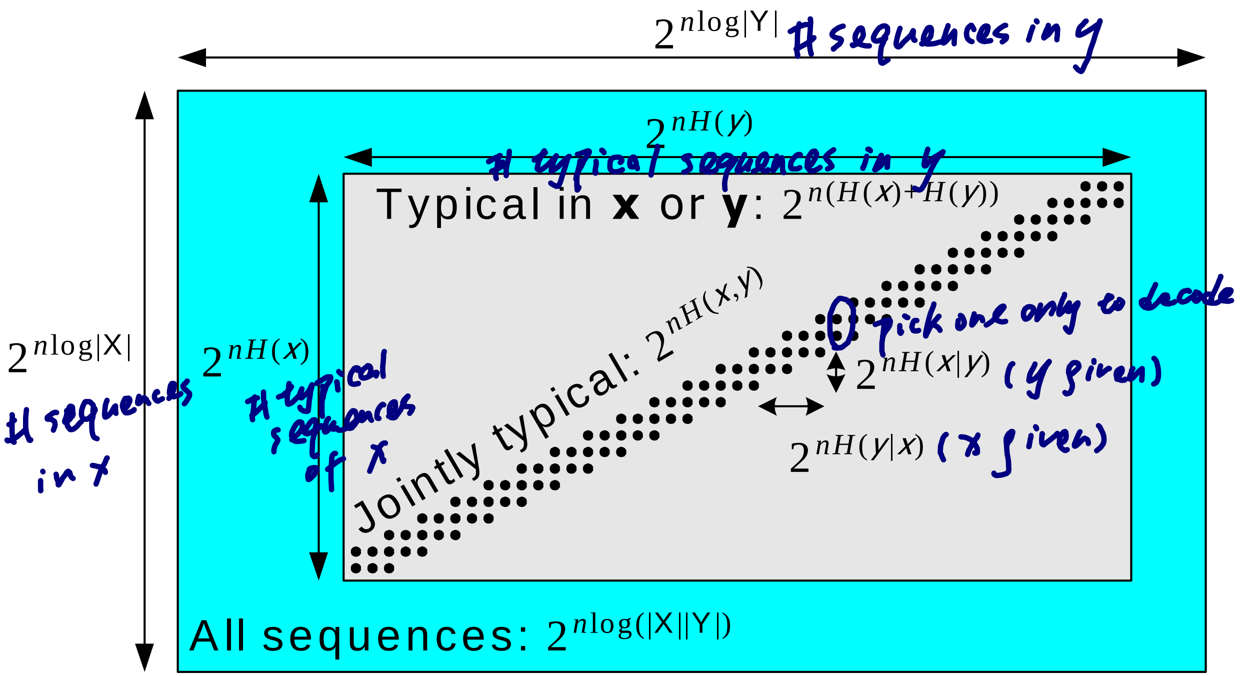
$\mathbf{x}$ = 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

$\mathbf{y}$ = 1 1 1 1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0

jointly typical set: all combinations of *x* and *y* have exactly the right frequencies

Jointly Typical Diagram

Each point defines both an **x** sequence and a **y** sequence



Dots represent jointly typical pairs (**x**,**y**)

Inner rectangle represents pairs that are typical in **x** or **y** but not necessarily jointly typical

- There are about $2^{nH(x)}$ typical **x**'s in all
- Each typical **y** is jointly typical with about $2^{nH(x|y)}$ of these typical **x**'s
- The jointly typical pairs are a fraction $2^{-nI(x;y)}$ of the inner rectangle
- Channel Code: choose **x**'s whose J.T. **y**'s don't overlap; use J.T. for decoding
- There are $2^{nI(x;y)}$ such codewords **x**'s

Joint Typical Set Properties

1. Indiv Prob: $\mathbf{x},\mathbf{y} \in J_\varepsilon^{(n)} \quad \Rightarrow \quad \log p(\mathbf{x},\mathbf{y}) = -\underline{n}H(x,y) \pm \underline{n}\varepsilon$
2. Total Prob: $p\left(\mathbf{x},\mathbf{y} \in J_\varepsilon^{(n)}\right) > 1-\varepsilon$ for $n > N_\varepsilon$
3. Size: $(1-\varepsilon)2^{n(H(x,y)-\varepsilon)} \overset{n>N_\varepsilon}{<} \left|J_\varepsilon^{(n)}\right| \le 2^{n(H(x,y)+\varepsilon)}$

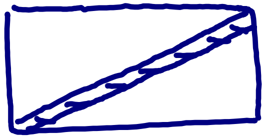
Proof 2: (use weak law of large numbers)

Choose N_1 such that $\forall n > N_1, \quad p\left(\left|-n^{-1}\log p(\mathbf{x}) - H(x)\right| > \varepsilon\right) < \frac{\varepsilon}{3}$

Similarly choose N_2, N_3 for other conditions and set $N_\varepsilon = \max(N_1, N_2, N_3)$

Proof 3: $1-\varepsilon < \sum_{\mathbf{x},\mathbf{y}\in J_\varepsilon^{(n)}} p(\mathbf{x},\mathbf{y}) \le \left|J_\varepsilon^{(n)}\right| \max_{\mathbf{x},\mathbf{y}\in J_\varepsilon^{(n)}} p(\mathbf{x},\mathbf{y}) = \left|J_\varepsilon^{(n)}\right| 2^{-n(H(x,y)-\varepsilon)}$ $\quad n>N_\varepsilon$

$$1 \ge \sum_{\mathbf{x},\mathbf{y}\in J_\varepsilon^{(n)}} p(\mathbf{x},\mathbf{y}) \ge \left|J_\varepsilon^{(n)}\right| \min_{\mathbf{x},\mathbf{y}\in J_\varepsilon^{(n)}} p(\mathbf{x},\mathbf{y}) = \left|J_\varepsilon^{(n)}\right| 2^{-n(H(x,y)+\varepsilon)} \qquad \forall n$$



Properties

4. If $\mathbf{p}_{x'}=\mathbf{p}_x$ and $\mathbf{p}_{y'}=\mathbf{p}_y$ with x' and y' independent:

$$(1-\varepsilon)2^{-n(I(x,y)+3\varepsilon)} \le p\left(\mathbf{x}',\mathbf{y}'\in J_\varepsilon^{(n)}\right) \le 2^{-n(I(x,y)-3\varepsilon)} \text{ for } n > N_\varepsilon$$

Proof: |J| × (Min Prob) ≤ Total Prob ≤ |J| × (Max Prob)

$$p\left(\mathbf{x}',\mathbf{y}'\in J_\varepsilon^{(n)}\right) = \sum_{\mathbf{x}',\mathbf{y}'\in J_\varepsilon^{(n)}} p(\mathbf{x}',\mathbf{y}') = \sum_{\mathbf{x}',\mathbf{y}'\in J_\varepsilon^{(n)}} p(\mathbf{x}')p(\mathbf{y}')$$

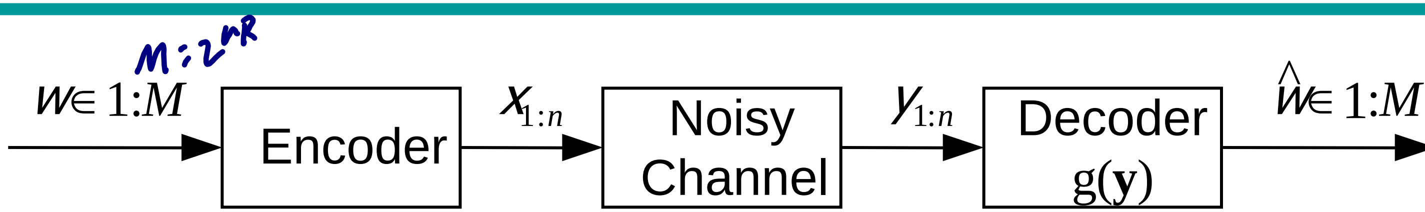
$$p\left(\mathbf{x}',\mathbf{y}'\in J_\varepsilon^{(n)}\right) \le \left|J_\varepsilon^{(n)}\right| \max_{\mathbf{x}',\mathbf{y}'\in J_\varepsilon^{(n)}} p(\mathbf{x}')p(\mathbf{y}')$$

$$\le 2^{n(H(x,y)+\varepsilon)}2^{-n(H(x)-\varepsilon)}2^{-n(H(y)-\varepsilon)} = 2^{-n(I(x;y)-3\varepsilon)}$$

$$p\left(\mathbf{x}',\mathbf{y}'\in J_\varepsilon^{(n)}\right) \ge \left|J_\varepsilon^{(n)}\right| \min_{\mathbf{x}',\mathbf{y}'\in J_\varepsilon^{(n)}} p(\mathbf{x}')p(\mathbf{y}')$$

$$\ge (1-\varepsilon)2^{-n(I(x;y)+3\varepsilon)} \quad \text{for} \quad n > N_\varepsilon$$

Channel Coding



- Assume Discrete Memoryless Channel with known $\mathbf{Q}_{y|x}$

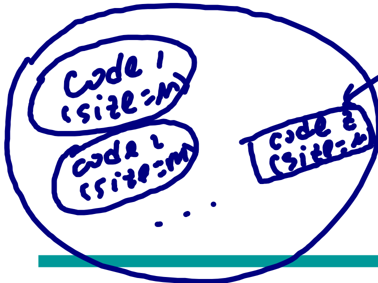
  # codewords　　code length

- An (M, n) code is　　$M = 2^{nR}$
  - A fixed set of M codewords $\mathbf{x}(w) \in \mathcal{X}^n$ for $w=1:M$
  - A deterministic decoder $g(\mathbf{y}) \in 1:M$

  normalised

- The rate of an (M,n) code: $R=(\log M)/n$ bits/transmission
- Error probability $\lambda_w = p\left(g\left(\mathbf{y}(w)\right) \neq w\right) = \sum_{\mathbf{y} \in \mathcal{Y}^n} p\left(\mathbf{y} \mid \mathbf{x}(w)\right) \delta_{g(\mathbf{y}) \neq w}$
  - Maximum Error Probability $\lambda^{(n)} = \max_{1 \le w \le M} \lambda_w$
  - Average Error probability $P_e^{(n)} = \frac{1}{M} \sum_{w=1}^{M} \lambda_w$

$\delta_C = 1$ if C is true or 0 if it is false



small error prob.

Shannon's ideas

- Channel coding theorem: the basic theorem of information theory
  - Proved in his original 1948 paper
- How do you correct all errors?
- Shannon's ideas
  - Allowing arbitrarily small but nonzero error probability
  - Using the channel many times in succession so that AEP holds (by sending successive bits

    AEP: $-\frac{1}{n}\log P(X_1, X_2 \ldots X_n) \to H(X)$
  - Consider a randomly chosen code and show the expected average error probability is small
    - Use the idea of typical sequences
    - Show this means ∃ at least one code with small max error prob
    - Sadly it doesn't tell you how to construct the code

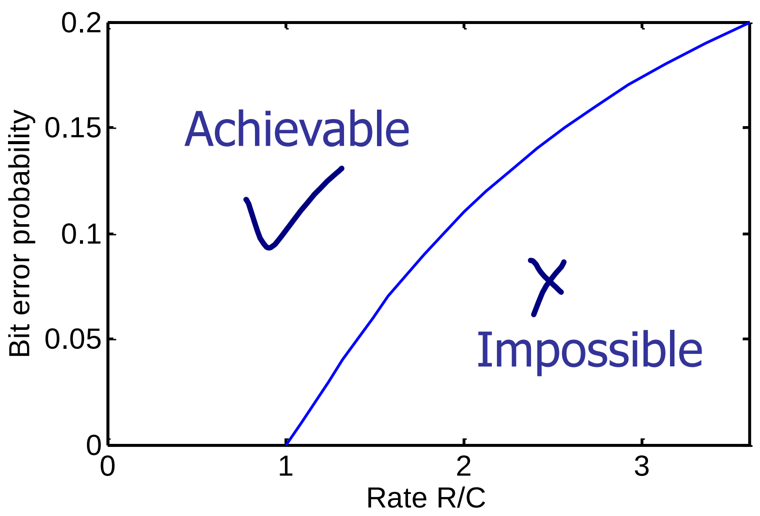
Channel Coding Theorem

- A rate R is achievable if $R<C$ and not achievable if $R>C$
  - If $R<C$, $\exists$ a sequence of $(2^{nR},n)$ codes with max prob of error $\lambda^{(n)}\to 0$ as $n\to\infty$
  - Any sequence of $(2^{nR},n)$ codes with max prob of error $\lambda^{(n)}\to 0$ as $n\to\infty$ must have $R\le C$

A very counterintuitive result:

Despite channel errors you can get arbitrarily low bit error rates provided that $R<C$

don't need to sacrify data rate when $R<C$.



Summary

- Jointly typical set

$$-\log p(\mathbf{x},\mathbf{y}) = nH(X,Y) \pm n\varepsilon$$

$$p\left(\mathbf{x},\mathbf{y} \in J_\varepsilon^{(n)}\right) > 1-\varepsilon$$

$$\left|J_\varepsilon^{(n)}\right| \le 2^{n\left(H(X,Y)+\varepsilon\right)}$$

$$(1-\varepsilon)2^{-n\left(I(X,Y)+3\varepsilon\right)} \le p\left(\mathbf{x}',\mathbf{y}' \in J_\varepsilon^{(n)}\right) \le 2^{-n\left(I(X,Y)-3\varepsilon\right)}$$

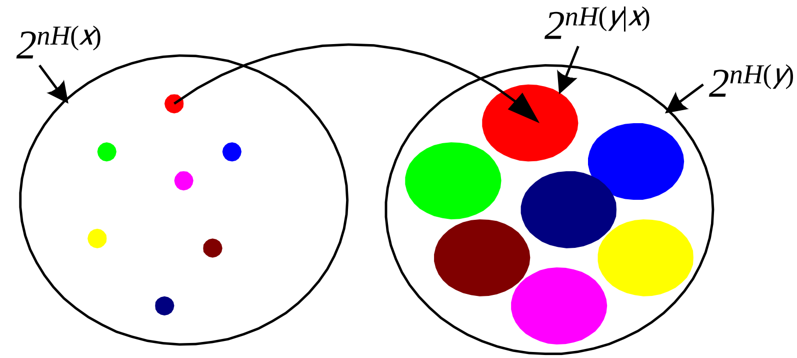
- Machinery to prove channel coding theorem

Lecture 10

- Channel Coding Theorem
  - Proof
    - Using joint typicality
    - Arguably the simplest one among many possible ways
    - Limitation: does not reveal $P_e \sim e^{-nE(R)}$
  - Converse (next lecture)

Channel Coding Principle

- Consider blocks of n symbols:



  - An average input sequence $x_{1:n}$ corresponds to about $2^{nH(y|x)}$ typical output sequences
  - Random Codes: Choose $M = 2^{nR}$ ($R \le I(x;y)$) random codewords $\mathbf{x}(w)$
    - their typical output sequences are unlikely to overlap much.
  - Joint Typical Decoding: A received vector $\mathbf{y}$ is very likely to be in the typical output set of the transmitted $\mathbf{x}(w)$ and no others. Decode as this w.

Channel Coding Theorem: for large n, can transmit at any rate $R < C$ with negligible errors

$$\begin{array}{c|cccc} w \backslash n & 1 & 2 & \cdots & n \\ \hline w=1 \;\; x(1) & x(1,1) & x(1,2) & \cdots & x(1,n) \\ w=2 \;\; x(2) & x(2,1) & x(2,2) & \cdots & x(2,n) \\ \vdots & & & & \\ w=M \;\; x(M) & x(M,1) & x(M,2) & \cdots & x(M,n) \end{array}$$

$M = 2^{nR}$

$x(\cdot,j) \sim P_x$

code: $M \times n$ i.i.d.

↳ completely random → can generate many!

Random $(2^{nR},n)$ Code

- Choose $\varepsilon \approx$ error prob, joint typicality $\Rightarrow N_\varepsilon$, choose $n>N_\varepsilon$
- Choose $\mathbf{p}_X$ so that $I(X;Y)=C$, the information capacity
- Use $\mathbf{p}_X$ to choose a code C with random $\mathbf{x}(w)\in X^n$, $w=1:2^{nR}$
  - the receiver knows this code and also the transition matrix $\mathbf{Q}$
- Assume the message $W\in 1:2^{nR}$ is uniformly distributed
- If received value is $\mathbf{y}$; decode the message by seeing how many $\mathbf{x}(w)$'s are jointly typical with $\mathbf{y}$
  - if $\mathbf{x}(k)$ is the only one then k is the decoded message
  - if there are 0 or ≥ 2 possible k's then declare an error message 0
  - we calculate error probability averaged over all C and all W

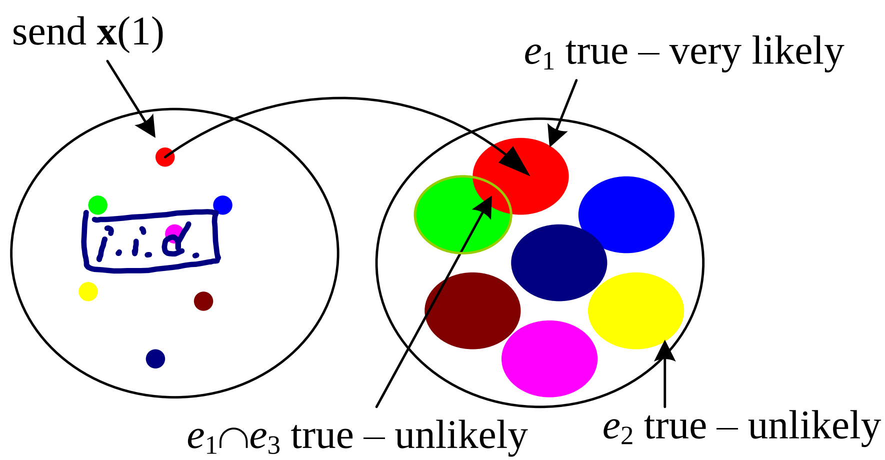
$$p(E) = \underbrace{\sum_{C} p(C) 2^{-nR} \underbrace{\sum_{w=1}^{2^{nR}} \lambda_w(C)}_{\text{all codewords for certain code}}}_{\text{ensemble average over all codes}} = 2^{-nR} \sum_{w=1}^{2^{nR}} \sum_{C} p(C)\lambda_w(C) \overset{(a)}{=} \sum_{C} p(C)\lambda_1(C) = \underbrace{p(E \mid w=1)}_{\text{error prob. of sending 1st codeword}}$$

w. i.i.d.

(a) since error averaged over all possible codes is independent of w

Decoding Errors

- Assume we transmit $\mathbf{x}(1)$ and receive $\mathbf{y}$
- Define the J.T. events $e_w = \left\{(\mathbf{x}(w), \mathbf{y}) \in J_\varepsilon^{(n)}\right\}$ for $w \in 1:2^{nR}$



- Decode using joint typicality
- We have an error if either e_1 false or e_w true for $w \geq 2$
- The $\mathbf{x}(w)$ for $w \neq 1$ are independent of $\mathbf{x}(1)$ and hence also independent of $\mathbf{y}$. So $p(e_w \text{ true}) < 2^{-n(I(X,Y)-3\varepsilon)}$ for any $w \neq 1$

Joint AEP

Error Probability for Random Code

$\varepsilon + 2^{-n\overbrace{(C-R-3\varepsilon)}^{>0}}$

$\leq \varepsilon + 2^{+\frac{\log \varepsilon}{C-R-3\varepsilon}(C-R-3\varepsilon)}$

$= \varepsilon + 2^{+\log \varepsilon}$

$= 2\varepsilon$

- Upper bound

Union bound: $P(A \cup B) \leq P(A) + P(B)$

$p(A \cup B) \leq p(A) + p(B)$

$$p(E) = p(E \mid W = 1) = p\left(\overline{e_1} \cup e_2 \cup e_3 \cup \cdots \cup e_{2^{nR}}\right) \leq p\left(\overline{e_1}\right) + \sum_{w=2}^{2^{nR}} p(e_w)$$

$$\leq \varepsilon + \sum_{i=2}^{2^{nR}} 2^{-n(I(X;Y)-3\varepsilon)} < \varepsilon + 2^{nR} 2^{-n(I(X;Y)-3\varepsilon)}$$

(1) Joint typicality

(2) Joint AEP

$$\leq \varepsilon + 2^{-n(C-R-3\varepsilon)} \leq 2\varepsilon \quad \text{for } R < C - 3\varepsilon \text{ and } n > -\frac{\log \varepsilon}{C-R-3\varepsilon}$$

$P(E) \leq 2\varepsilon$: the error prob. can be arbitrarily small if

$R < C - 3\varepsilon$: rate < capacity

$n > -\frac{\log \varepsilon}{C-R-3\varepsilon}$: n large enough

we have chosen $p(x)$ such that $I(X;Y) = C$

- Since average of $P(E)$ over all codes is $\leq 2\varepsilon$ there must be at least one code for which this is true: this code has $2^{-nR}\sum_{w} \lambda_w \leq 2\varepsilon$ ♦
- Now throw away the worst half of the codewords; the remaining ones must all have $\boxed{\lambda_w \leq 4\varepsilon}$. The resultant code has rate $R - n^{-1} \cong R$. ♦

♦ = proved on next page

$$2\varepsilon \geq \frac{1}{M}\sum_{w}\lambda_w = \frac{1}{M}\Big(\underbrace{\sum_{w=1}^{M/2}\lambda_w}_{\text{bad}(v)} + \underbrace{\sum_{w=\frac{M}{2}+1}^{M}\lambda_w}_{\text{good}(x)}\Big) \geq \frac{1}{M}\sum_{w=1}^{M/2}\lambda_w \geq \frac{1}{M}\frac{M}{2}\lambda_{\frac{M}{2}} = \frac{1}{2}\lambda_{\frac{M}{2}}$$

$$\Rightarrow \lambda_{\frac{M}{2}} \leq 4\varepsilon.$$

rate: $R' = \frac{\log \frac{M}{2}}{n} = \frac{\log M - \log 2}{n} = \frac{\log M}{n} - \frac{\log 2}{n} = R - \frac{1}{n} \xrightarrow{n\to\infty} R$

Code Selection & Expurgation

- Since average of $P(E)$ over all codes is $\leq 2\varepsilon$ there must be at least one code for which this is true.

  Proof:
  $$2\varepsilon \geq K^{-1}\sum_{i=1}^{K} P_{e,i}^{(n)} \geq K^{-1}\sum_{i=1}^{K} \min_i \left(P_{e,i}^{(n)}\right) = \min_i \left(P_{e,i}^{(n)}\right)$$

  K = num of codes

- Expurgation: Throw away the worst half of the codewords; the remaining ones must all have $\lambda_w \leq 4\varepsilon$.

  code i size=M

  throw away bad half

  sort by error prob.
  $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_M$

  Proof: Assume λ_w are in descending order

  $$2\varepsilon \geq M^{-1}\sum_{w=1}^{M}\lambda_w \geq M^{-1}\sum_{w=1}^{\frac{1}{2}M}\lambda_w \geq M^{-1}\sum_{w=1}^{\frac{1}{2}M}\lambda_{\frac{1}{2}M} \geq \tfrac{1}{2}\lambda_{\frac{1}{2}M}$$

  $$\Rightarrow \quad \lambda_{\frac{1}{2}M} \leq 4\varepsilon \quad \Rightarrow \quad \lambda_w \leq 4\varepsilon \quad \forall\ w > \tfrac{1}{2}M$$

$M' = \frac{1}{2}\times 2^{nR}$ messages in n channel uses $\Rightarrow$ $R' = n^{-1}\log M' = R - n^{-1}$

Summary of Procedure

- For any $R<C-3\varepsilon$ set $n = \max\left\{N_{\varepsilon}, -(\log \varepsilon)/(C-R-3\varepsilon), \varepsilon^{-1}\right\}$ sufficient large

see (a),(b),(c) below

- Find the optimum $\mathbf{p}_X$ so that $I(x\,;\,y) = C$
- Choosing codewords randomly (using $\mathbf{p}_X$) to construct codes with 2^{nR} codewords and using joint typicality as the decoder $M=2^{nR}$ (a)
- Since average of $P(E)$ over all codes is $\le 2\varepsilon$ there must be at least one code for which this is true. (b)
- Throw away the worst half of the codewords. Now the worst codeword has an error prob $\le 4\varepsilon$ with rate $= R - n^{-1} > R - \varepsilon$ (c)
- The resultant code transmits at a rate as close to C as desired with an error probability that can be made as small as desired (but n unnecessarily large).

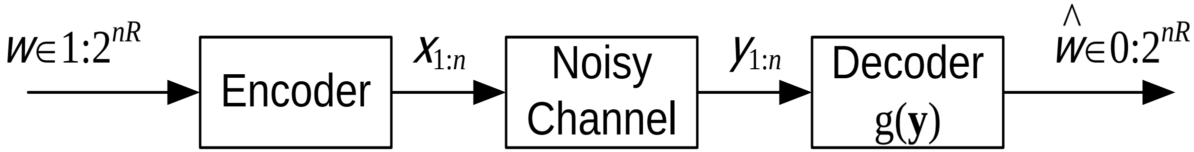
Note: ε determines both error probability and closeness to capacity

Remarks

- Random coding is a powerful method of proof, not a method of signaling
- Picking randomly will give a good code
- But n has to be large (AEP)
- Without a structure, it is difficult to encode/decode
  - Table lookup requires exponential size
- Channel coding theorem does not provide a practical coding scheme
- Folk theorem (but outdated now):
  - Almost all codes are good, except those we can think of

Lecture 11

- Converse of Channel Coding Theorem
  - Cannot achieve R>C
- Capacity with feedback
  - No gain for DMC but simpler encoding/ decoding
- Joint Source-Channel Coding
  - No point for a DMC



$W_{1:nR}$ i.i.d. $\Rightarrow H(W_i)=1 \Rightarrow H(\underline{W})=nR$ (by summation)

Converse of Coding Theorem

- Fano's Inequality: if $P_e^{(n)}$ is error prob when estimating w from **y**,

$$H(W \mid \mathbf{y}) \le 1 + P_e^{(n)} \underbrace{\log|W|}_{H(W)} = 1 + \underbrace{nR}_{H(W)} P_e^{(n)}$$

- Hence $\quad nR = H(W) = H(W \mid \mathbf{y}) + I(W;\mathbf{y})$ Definition of I

$nR \le 1 + nRP_e^{(n)} + nC$

$P_e^{(n)} \ge \dfrac{R-C-\frac{1}{n}}{R}$

$\xrightarrow{n\to\infty} 1-\dfrac{C}{R} \begin{cases} >0 \text{ if } R>C \\ \le 0 \text{ (can be 0) if } R \le C \end{cases}$

$W \to X(W)$

$$\le H(W \mid \mathbf{y}) + I(\mathbf{x}(W);\mathbf{y})$$ Markov : $W \to \mathbf{x} \to \mathbf{y} \to \hat{W}$

$$\le 1 + nRP_e^{(n)} + I(\mathbf{x};\mathbf{y})$$ Fano

cannot achieve zero error rate!

$$\le 1 + nRP_e^{(n)} + nC$$ n-use DMC capacity

lower bound of P_e

$$\Rightarrow P_e^{(n)} \ge \frac{R-C-n^{-1}}{R} \underset{n\to\infty}{\to} 1-\frac{C}{R} > 0 \text{ if } R > C$$

- For large (hence for all) n, $P_e^{(n)}$ has a lower bound of $(R-C)/R$ if w equiprobable
  - If achievable for small n, it could be achieved also for large n by concatenation.

Minimum Bit-Error Rate

Suppose

- $w_{1:nR}$ is i.i.d. bits with $H(w_i)=1$ $H(\underline{w})=nR$
- The bit-error rate is $P_b = E_i\left\{p(w_i \neq \hat{w}_i)\right\} \stackrel{\Delta}{=} E_i\left\{p(e_i)\right\}$

Then

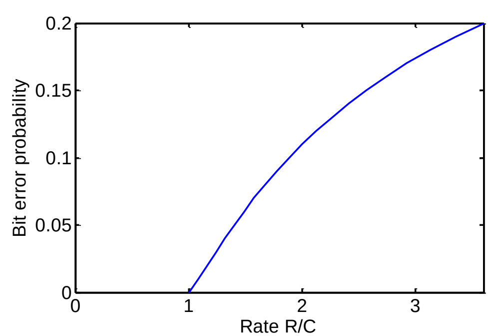
$$nC \stackrel{(a)}{\geq} I(x_{1:n};y_{1:n}) \stackrel{(b)}{\geq} I(w_{1:nR};\hat{w}_{1:nR}) = H(w_{1:nR}) - H(w_{1:nR}\,|\,\hat{w}_{1:nR})$$

$$= nR - \sum_{i=1}^{nR} H(w_i\,|\,\hat{w}_{1:nR}, w_{1:i-1}) \stackrel{(c)}{\geq} nR - \sum_{i=1}^{nR} H(w_i\,|\,\hat{w}_i) = nR\left(1 - E_i\left\{H(w_i\,|\,\hat{w}_i)\right\}\right)$$

$$\stackrel{(d)}{=} nR\left(1 - E_i\left\{H(e_i\,|\,\hat{w}_i)\right\}\right) \stackrel{(c)}{\geq} nR\left(1 - E_i\left\{H(e_i)\right\}\right) \stackrel{(e)}{\geq} nR\left(1 - H(E_i\, P(e_i))\right) = nR\left(1 - H(P_b)\right)$$

Hence

$$\boxed{\begin{aligned} R &\leq C\left(1 - H(P_b)\right)^{-1} \\ P_b &\geq H^{-1}(1 - C/R) \end{aligned}}$$

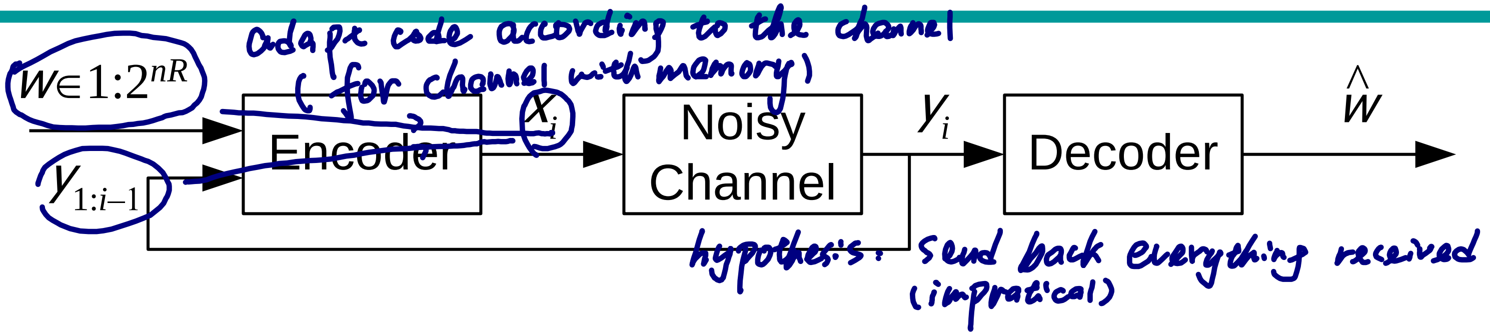


(a) n-use capacity
(b) Data processing theorem
(c) Conditioning reduces entropy
(d) $e_i = w_i \oplus \hat{w}_i$
(e) Jensen: $E\,H(x) \leq H(E\,x)$

Coding Theory and Practice

- Construction for good codes
  - Ever since Shannon founded information theory
  - Practical: Computation & memory $\propto n^k$ for some k
- Repetition code: rate → 0
- Block codes: encode a block at a time
  - Hamming code: correct one error
  - Reed-Solomon code, BCH code: multiple errors (1950s)
- Convolutional code: convolve bit stream with a filter
- Concatenated code: RS + convolutional
- Capacity-approaching codes:
  - Turbo code: combination of two interleaved convolutional codes (1993)
  - Low-density parity-check (LDPC) code (1960)
  - Dream has come true for some channels today

Channel with Feedback



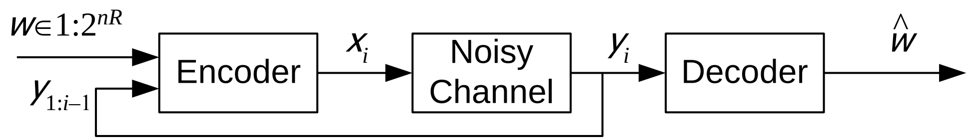
- Assume error-free feedback: does it increase capacity ?
- A $(2^{nR}, n)$ feedback code is
  - A sequence of mappings $x_i = x_i(w, y_{1:i-1})$ for $i=1:n$ depends
  - A decoding function $\hat{w} = g(y_{1:n})$

- A rate R is achievable if ∃ a sequence of $(2^{nR}, n)$ feedback codes such that $P_e^{(n)} = P(\hat{w} \neq w) \xrightarrow[n\to\infty]{} 0$
- Feedback capacity, $C_{FB} \geq C$, is the sup of achievable rates

intuition: $C_{FB} \geq C$ if abandon all feedbacks

Fano's ineq. ⇒ prove impossible

Feedback Doesn't Increase Capacity



$$I(W;\mathbf{y}) = H(\mathbf{y}) - H(\mathbf{y} \mid W)$$

$$= H(\mathbf{y}) - \sum_{i=1}^{n} H(y_i \mid \underline{y_{1:i-1}, W})$$

$x_i \sim \{ w, y_{1:n-1}$

↓ no new conditions

$$= H(\mathbf{y}) - \sum_{i=1}^{n} H(y_i \mid y_{1:i-1}, W, x_i) \quad \text{since } x_i = x_i(w, y_{1:i-1})$$

↓ y_i only depends on x_i

$$= H(\mathbf{y}) - \sum_{i=1}^{n} H(y_i \mid x_i) \quad \text{since } y_i \text{ only directly depends on } x_i$$

$H(y) = \sum_{i=1}^{n} H(y_i \mid y_{1:i-1}) \le \sum_{i=1}^{n} H(y_i)$

$$\le \sum_{i=1}^{n} H(y_i) - \sum_{i=1}^{n} H(y_i \mid x_i) = \sum_{i=1}^{n} I(x_i; y_i) \le nC$$

cond reduces ent

DMC

$H(W \mid y) \le 1 + nRP_e^{(n)}$

Hence

$$nR = H(W) = \underline{H(W \mid \mathbf{y})} + I(W;\mathbf{y}) \le \underline{1 + nRP_e^{(n)}} + nC \quad \text{Fano}$$

$$\Rightarrow P_e^{(n)} \ge \frac{R - C - n^{-1}}{R}$$

➔ Any rate > C is unachievable

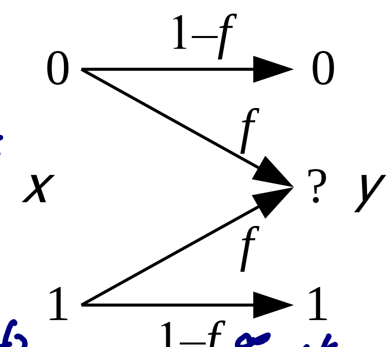
The DMC does not benefit from feedback: $C_{FB} = C$

$E\{k\} = (1-f) + 2(f - f^2) + 3(f^2 - f^3) + \cdots + k(f^{n-1} - f)$
$= 1 + f + f^2 + \cdots = \frac{1}{1-f}$

Example: BEC with feedback

k: # transmissions required to recover one bit.

$k=1 \quad P_1 = 1-f$
$k=2 \quad P_2 = f(1-f) = f - f^2$
$k=3 \quad P_3 = f^2(1-f) = f^2 - f^3$
$k \quad P_k = f^{k-1}(1-f) = f^{k-1} - f^k$



- Capacity is $1-f$
- Encode algorithm
  - If $y_i = ?$, tell the sender to retransmit bit i
  - Average number of transmissions per bit:

$$1 + f + f^2 + \cdots = \frac{1}{1-f}$$

capacity: number of successful transmission bits per transmission

$E\{k\} = \sum_{k=1}^{\infty} k f^{k-1}(1-f)$
$h(f) = H'(f)$
$h(f) = \sum_{k=1}^{\infty} k f^{k-1}$

$H(f) = \sum_{k=1}^{\infty} f^k$
$= \sum_{k=0}^{\infty} f^k - 1$
$= \frac{1}{1-f} - 1 = \frac{f}{1-f}$

- Average number of successfully recovered bits per transmission $= 1-f$
  - Capacity is achieved!

$H'(f) = \frac{1-f+f}{(1-f)^2} = \frac{1}{(1-f)^2}$

$E\{k\} = \frac{1}{(1-f)^2} \cdot (1-f) = \frac{1}{1-f}$

- Capacity unchanged but encoding/decoding algorithm much simpler.

Joint Source-Channel Coding

$w_{1:n} \rightarrow$ Encoder $\xrightarrow{x_{1:n}}$ Noisy Channel $\xrightarrow{y_{1:n}}$ Decoder $\rightarrow \hat{w}_{1:n}$

- Assume w_i satisfies AEP and $|W|<\infty$
  - Examples: i.i.d.; Markov; stationary ergodic
- Capacity of DMC channel is C
  - if time-varying: $C = \lim_{n \to \infty} n^{-1} I(\mathbf{x};\mathbf{y})$

source code channel code

$H(W) < R < C$

- Joint Source-Channel Coding Theorem: ♦

  $\exists$ codes with $P_e^{(n)} = P(\hat{w}_{1:n} \neq w_{1:n}) \xrightarrow[n \to \infty]{} 0$ iff $H(W) < C$
  - errors arise from two reasons
    - Incorrect encoding of **w** (source coding)
    - Incorrect decoding of **y** (channel coding)

♦ = proved on next page

Source-Channel Proof ($\Leftarrow$)

- Achievability is proved by using two-stage encoding
  - Source coding
  - Channel coding
- For $n > N_\varepsilon$ there are only $2^{n(H(\mathrm{W})+\varepsilon)}$ **w**'s in the typical set: encode using $n(H(W)+\varepsilon)$ bits
  - encoder error < ε
- Transmit with error prob less than ε so long as $H(W)+\varepsilon < C$
- Total error prob $< 2\varepsilon$

Source-Channel Proof ($\Rightarrow$)

Fano's Inequality: $H(\mathbf{w} \mid \hat{\mathbf{w}}) \le 1 + P_e^{(n)} n \log |\mathrm{W}|$

$$H(W) \le n^{-1} H(w_{1:n})$$ entropy rate of stationary process

$$= n^{-1} H(w_{1:n} \mid \hat{w}_{1:n}) + n^{-1} I(w_{1:n}; \hat{w}_{1:n})$$ definition of I

$$\le n^{-1}\left(1 + P_e^{(n)} n \log |\mathrm{W}|\right) + n^{-1} I(x_{1:n}; y_{1:n})$$ Fano + Data Proc Inequ

$$\le n^{-1} + P_e^{(n)} \log |\mathrm{W}| + C$$ Memoryless channel

Let $n \to \infty \quad \Rightarrow \quad P_e^{(n)} \to 0 \quad \Rightarrow \quad H(W) \le C$

Separation Theorem

- Important result: source coding and channel coding might as well be done separately since same capacity
  - Joint design is more difficult (for linear channels)
- Practical implication: for a DMC we can design the source encoder and the channel coder separately
  - Source coding: efficient compression
  - Channel coding: powerful error-correction codes
- Not necessarily true for
  - Correlated channels
  - Multiuser channels
- Joint source-channel coding: still an area of research
  - Redundancy in human languages helps in a noisy environment

Summary

- Converse to channel coding theorem
  - Proved using Fano's inequality
  - Capacity is a clear dividing point:
    - If $R < C$, error prob. → 0
    - Otherwise, error prob. → 1
- Feedback doesn't increase the capacity of DMC
  - May increase the capacity of memory channels (e.g., ARQ in TCP/IP)
- Source-channel separation theorem for DMC and stationary sources

Lecture 12

- Polar codes
  - Channel polarization
  - How to construct polar codes
  - Encoding and decoding
- Polar source coding
- Extension

About Polar Codes

- Provably capacity-achieving
- Encoding complexity $O(N \log N)$
- Successive decoding complexity $O(N \log N)$
- Probability of error $\approx 2^{-\sqrt{N}}$
- Main idea: channel polarization

What Is Channel Polarization?

- Normal channel
- Extreme channel

Polarization

Useless channel

Perfect channel

sometimes cute,
sometimes lazy,
hard to manage

Channel Polarization

- Among all channels, there are two classes which are easy to communicate optimally
  - The perfect channels

    the output Y determines the input X
  - The useless channels

    Y is independent of X
- Polarization is a technique to convert noisy channels to a mixture of extreme channels
  - The process is information-conserving

$F_2 \otimes F_2 = \begin{bmatrix} F_2 & 0 \\ F_2 & F_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$ $F_8 = F_2 \otimes F_2 \otimes F_2 = F_2 \otimes F_4$

Generator Matrix

- Generator Matrix

$$\mathbf{F}_N = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}^{\otimes n}, N = 2^n$$

$\otimes n$ denotes the n-fold Kronecker product.

- Example

$$\mathbf{F}_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}, \mathbf{F}_4 = \begin{bmatrix} \mathbf{F}_2 & 0 \\ \mathbf{F}_2 & \mathbf{F}_2 \end{bmatrix} \text{ and so on.}$$

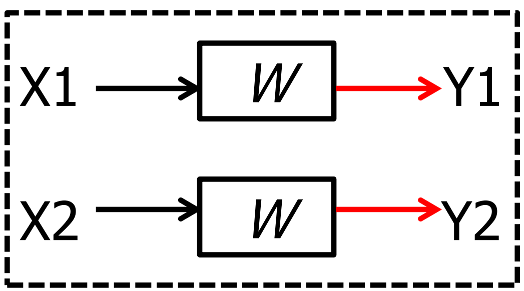
- Encoding

Let $\mathbf{u}$ be the length-N input to the encoder, then $\mathbf{x} = \mathbf{u}\mathbf{F}_N$ is the codeword.

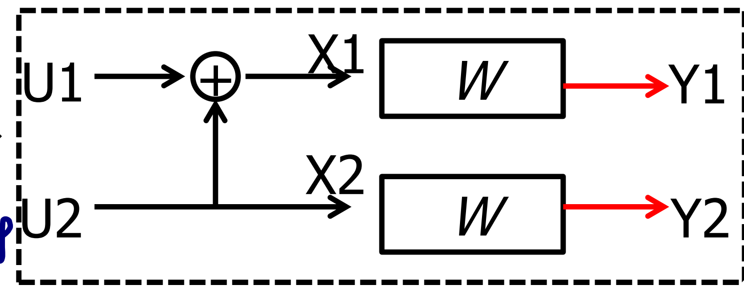
Channel Combining and Splitting

$\underline{x} = \underline{u} F_N$ $\quad [x_1\ x_2] = [u_1\ u_2] \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} = [u_1+u_2\ \ u_2]$

- Basic operation ($N = 2$)

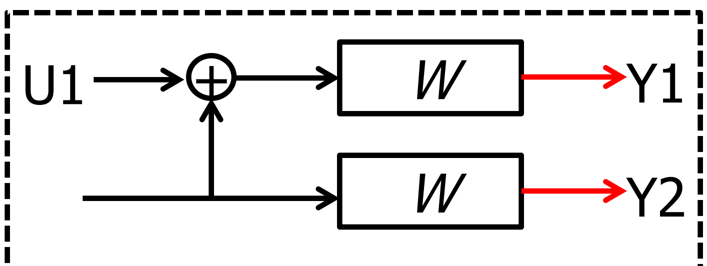


channel combining

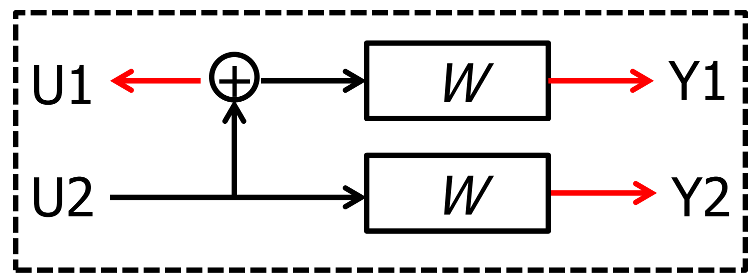


(X1, X2)=(U1, U2)$\mathbf{F}_2$

Channel splitting



channel splitting



W^-: U1→(Y1, Y2)

W^+: U2→(Y1, Y2, U1)

$(y_1, y_2) =$ ① $(u_1+u_2,\ u_2)$ $(1-p)^2$

② $(?,\ u_2)$ $p(1-p)$

③ $(u_1+u_2,\ ?)$ $p(1-p)$

④ $(?,\ ?)$ p^2

W^-

decode u_1? ①

W^-: erasure prob. $1-(1-p)^2 = 2p-p^2 \overset{p<1}{>} p$ (gets worse!)

What Happens?

- Suppose W is a BEC(p), i.e., Y=X with probability $1-p$, and Y=? (erasure) with probability p.

(y_1, y_2) ① $(u_1+u_2,\ u_2)$ $(1-p)^2$

② $(?,\ u_2)$ $p(1-p)$

③ $(u_1+u_2,\ ?)$ $p(1-p)$

④ $(?,\ ?)$ p^2

W^+

u_1 is known

decode u_2 (assume u_1 known)? ① ② ③

W^+: erasure prob. $p^2 \overset{p<1}{<} p$ (gets better!)

  - W^- has input U1 and output (Y1, Y2)=(U1+U2, U2) or (?, U2) or (U1+U2, ?) or (?, ?).
  - W^- is a BEC($2p - p^2$)

$C(W^-) = 1-(2p-p^2) = 1-2p+p^2 = (1-p)^2$

$C(W^+) = 1-p^2$

  - W^+ has input U2 and output (Y1, Y2, U1)=(U1+U2, U2, U1) or (?, U2, U1) or (U1+U2, ?, U1) or (?, ?, U1).

$C(W^+) - C(W^-) = 1-p^2-(1-2p+p^2) = -2p^2+2p = 2p(1-p) \geq 0$

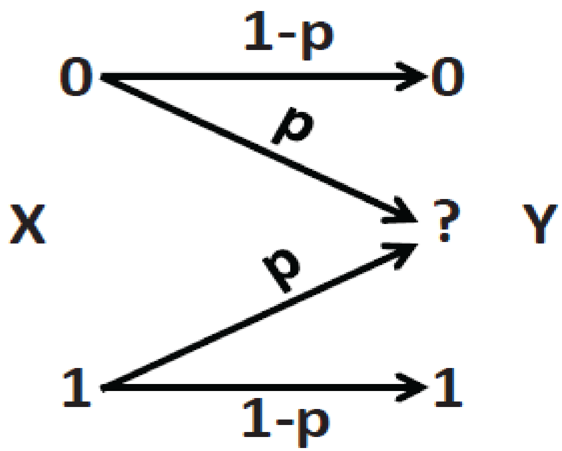
$C(W^-)+C(W^+) = 2(1-p) = 2C(W)$

  - W^+ is a BEC(p^2)

- W^- is worse than W, and W^+ is better (recall capacity $C(W)=1-p$).
  - $C(W^-) + C(W^+) = 2C(W)$
  - $C(W^-) \leq C(W) \leq C(W^+)$

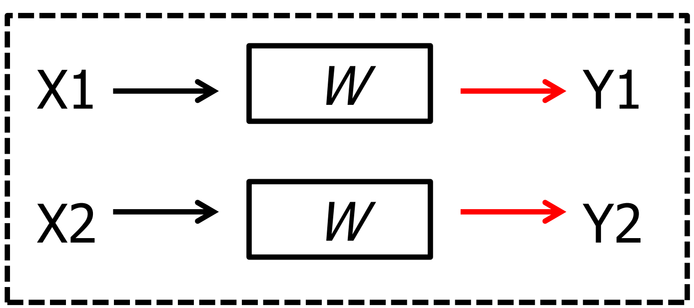
Example: BEC(0.5)

- *W* is a BEC with erasure probability p = 0.5.



$$\begin{array}{cc} & Y \\ X & \boxed{\begin{matrix} 1-p & p & 0 \\ 0 & p & 1-p \end{matrix}} \end{array}$$

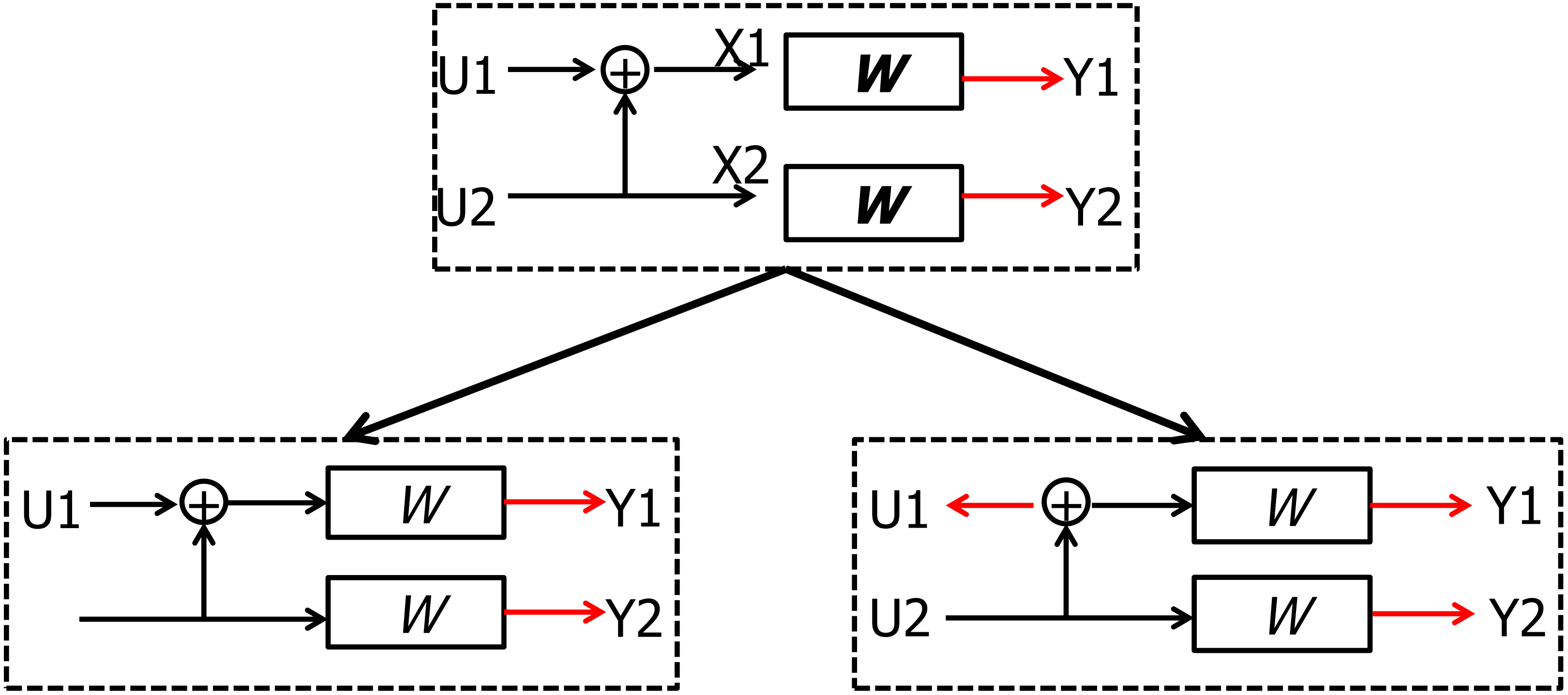
- If we use two copies of *W* separately



$$2C(W) = 2 \times 0.5 = 1$$

Example: BEC(0.5)

- Channel combining and splitting



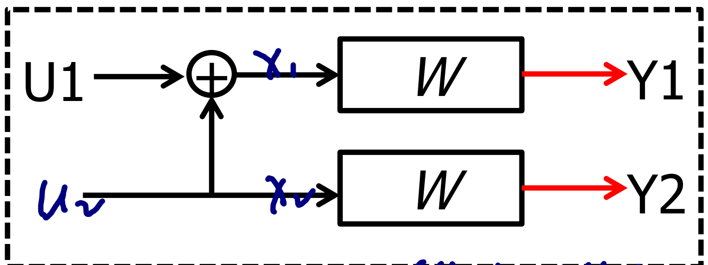
Example: BEC(0.5)

$[x_1\ x_2] = [u_1\ u_2]\begin{bmatrix}1 & 0\\ 1 & 1\end{bmatrix}$

$x_1 = u_1 + u_2 \rightarrow y_1$

$x_2 = u_2 \rightarrow y_2$

- Channel W^-



$$C(W^-) = 4 \times \frac{1}{16} \log 2 = 0.25$$

Transitional probabilities

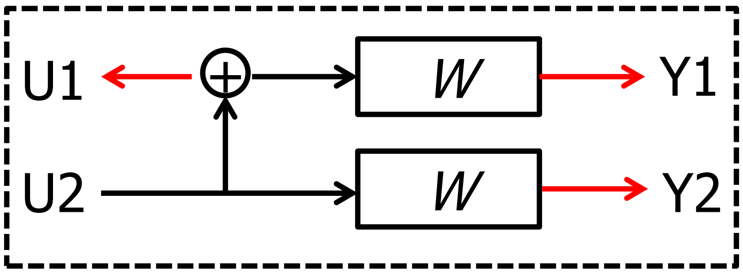
(Y1,Y2)

	$\begin{cases}u_1+u_2=0\\u_2=0\end{cases} \Rightarrow \begin{cases}u_1=0\\u_2=0\end{cases}$	$\begin{cases}u_1+u_2=0\\u_2=?\end{cases} \Rightarrow \begin{cases}u_1=0\\u_2=0\end{cases}$	$\begin{cases}u_1+u_2=0\\u_2=1\end{cases}$ ✗	$\begin{cases}u_1+u_2=?\\u_2=0\end{cases} \Rightarrow \begin{cases}u_1=?\\u_2=0\end{cases}$	$\begin{cases}u_1+u_2=?\\u_2=?\end{cases} \Rightarrow \begin{cases}u_1=?\\u_2=?\end{cases}$	$\begin{cases}u_1+u_2=?\\u_2=1\end{cases} \Rightarrow \begin{cases}u_1=?\\u_2=1\end{cases}$	$\begin{cases}u_1+u_2=1\\u_2=0\end{cases} \Rightarrow \begin{cases}u_1=1\\u_2=0\end{cases}$	$\begin{cases}u_1+u_2=1\\u_2=?\end{cases} \Rightarrow \begin{cases}u_1=?\\u_2=?\end{cases}$	$\begin{cases}u_1+u_2=1\\u_2=1\end{cases} \Rightarrow \begin{cases}u_1=?\\u_2=1\end{cases}$
	00	0?	01	?0	??	?1	10	1?	11
U1 (sent) 0 (u_2 ↓ ch)	1/8 $\frac{1}{2}\times\frac{1}{4}$	1/8 $\frac{1}{2}\times\frac{1}{4}$	0	1/8 $\frac{1}{2}\times\frac{1}{4}$	1/4 $1\times\frac{1}{4}$	1/8 $\frac{1}{2}\times\frac{1}{4}$	0	1/8 $\frac{1}{2}\times\frac{1}{4}$	1/8 $\frac{1}{2}\times\frac{1}{4}$
1	0	1/8	1/8	1/8	1/4	1/8	1/8	1/8	0

only 1 possible input given output ⇒ valid transmission

Example: BEC(0.5)

- Channel W^+



$\frac{1}{8}\log_2\frac{1}{4}$

$$C(W^+) = 12 \times \frac{1}{16} \log 2 = 0.75$$

$$C(W^-) + C(W^+) = 2C(W)$$

$$C(W^-) < C(W) < C(W^+)$$

(Y1,Y2,U1) Transitional probabilities

U2

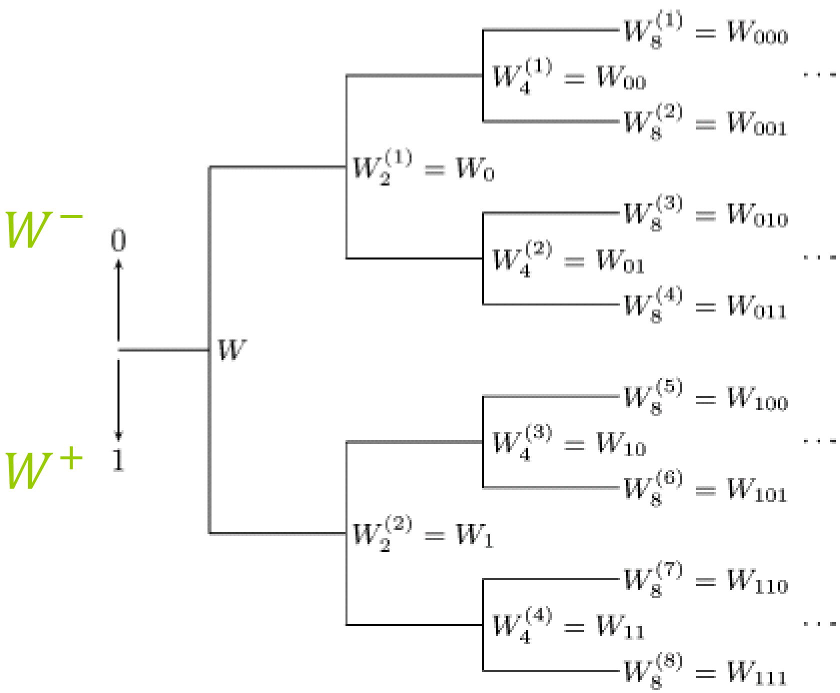
	000	0?0	010	?00	??0	?10	100	1?0	110
0	1/8	1/8	0	1/8	1/8	0	0	0	0
1	0	0	0	0	1/8	1/8	0	1/8	1/8

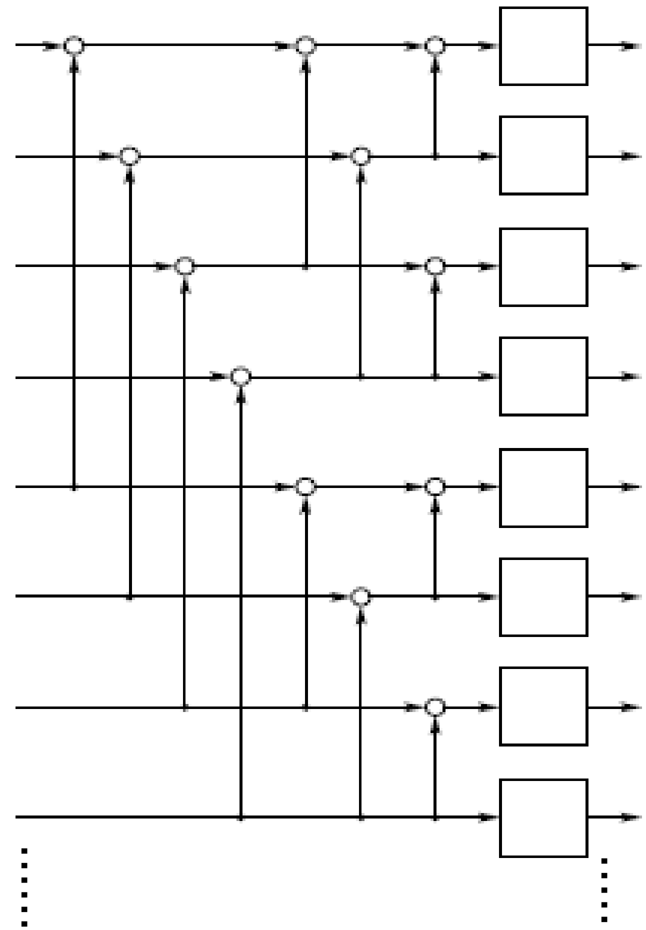
impossible confused!

	001	0?1	011	?01	??1	?11	101	1?1	111
0	0	0	0	1/8	1/8	0	1/8	1/8	0
1	0	1/8	1/8	0	1/8	1/8	0	0	0

More Polarization

- Repeating this, we obtain *N* ‘bit channels’ at the *n*-th step.
- More conveniently, this process can be described as a binary tree.
  - –Note how the ‘bit channels’ $W_{b_1 b_2 \cdots b_n}$ are labelled in the tree.





Martingale

- Now pick a 'bit channel' uniformly at random on the *n*-th level of the tree, which is equivalent to a random traverse on the tree, namely, at each step the r.v b_i takes the value of 0 or 1 with equal probability.
- We claim capacity C_n at the *n*-th step is a martingale.
- Proof: By information-preserving

$$E[C_{n+1}|b_1,\ldots,b_n] = \frac{1}{2}C\left(W_{b_1b_2\ldots b_n0}\right) + \frac{1}{2}C\left(W_{b_1b_2\ldots b_n1}\right)$$
$$= C\left(W_{b_1b_2\ldots b_n}\right) = C_n$$

- By the martingale convergence theorem, C_n converges to a random variable C_∞ such that $E[C_\infty] = E[C_0] = C_0 = C(W)$.
- In fact, the limit $C_\infty = 0$ or 1 is a binary random variable (these are the fixed points of the polar transform).

Review of Martingales

- Let $\{X_n, n \geq 0\}$ be a random process. If
$$E[X_{n+1} | X_n, \ldots, X_1, X_0] = X_n$$
then $\{X_n\}$ is referred to as a martingale.
- Martingale convergence theorem: Let $\{X_n, n \geq 0\}$ be a martingale with finite means. Then there exists a random variable X_∞ such that
$$X_n \to X_\infty \quad \text{almost surely}$$
as $n \to \infty$.

How to construct polar codes

- To achieve $C(W)$, we need to identify the indices of those bit channels (branches in tree) with capacity ≈ 1.
- For BEC, this can be computed recursively

$$C\left(W_{b_1 b_2 \ldots b_n 0}\right) = C\left(W_{b_1 b_2 \ldots b_n}\right)^2$$

$$C\left(W_{b_1 b_2 \ldots b_n 1}\right) = 2C\left(W_{b_1 b_2 \ldots b_n}\right) - C\left(W_{b_1 b_2 \ldots b_n}\right)^2$$

- For other types of channels, it is difficult to obtain closed-form formulas. So numerical computation is often used.

perfect useless

F_N

$R = \frac{NC(W)}{N} = C(W)$

Polarization Speed

- For any positive real number $\beta < 0.5$,

$$\lim_{n\to\infty} \frac{1}{N} \# \left\{(b_1 \cdots b_n): C\left(W_{b_1 b_2 \dots b_n}\right) \geq 1 - 2^{-N^\beta}\right\}$$

the fraction of good channels equals the capacity.

$$= C(W).$$

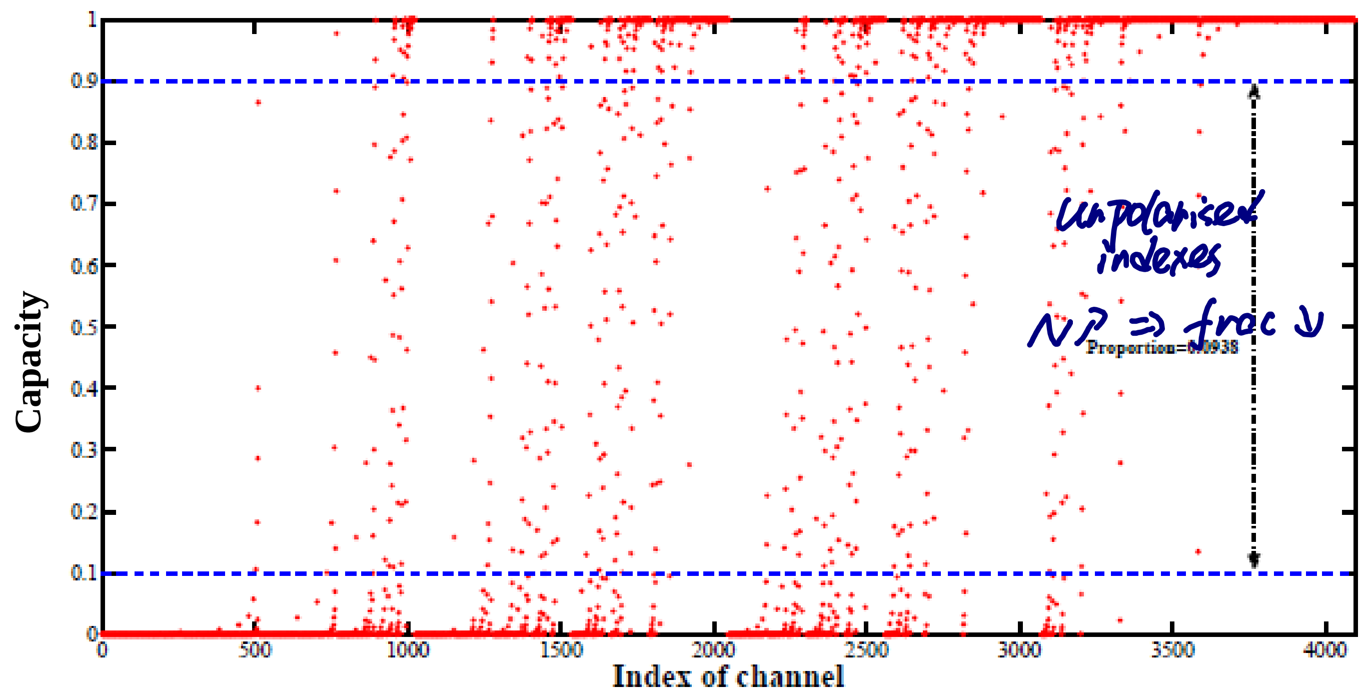
$$\lim_{n\to\infty} \frac{1}{N} \# \left\{(b_1 \cdots b_n): C\left(W_{b_1 b_2 \dots b_n}\right) < 1 - 2^{-N^\beta}\right\}$$

$$= 1 - C(W).$$

- The above statements do not hold for $\beta > 0.5$.
- Thus, the polarization speed is roughly $2^{-\sqrt{N}}$.

Convergence

- The portion of almost prefect bit channels is $C(W)$, meaning that the capacity is achieved.
- Example: capacities for $N = 2^{12}$ for BEC(0.5)



Encoding

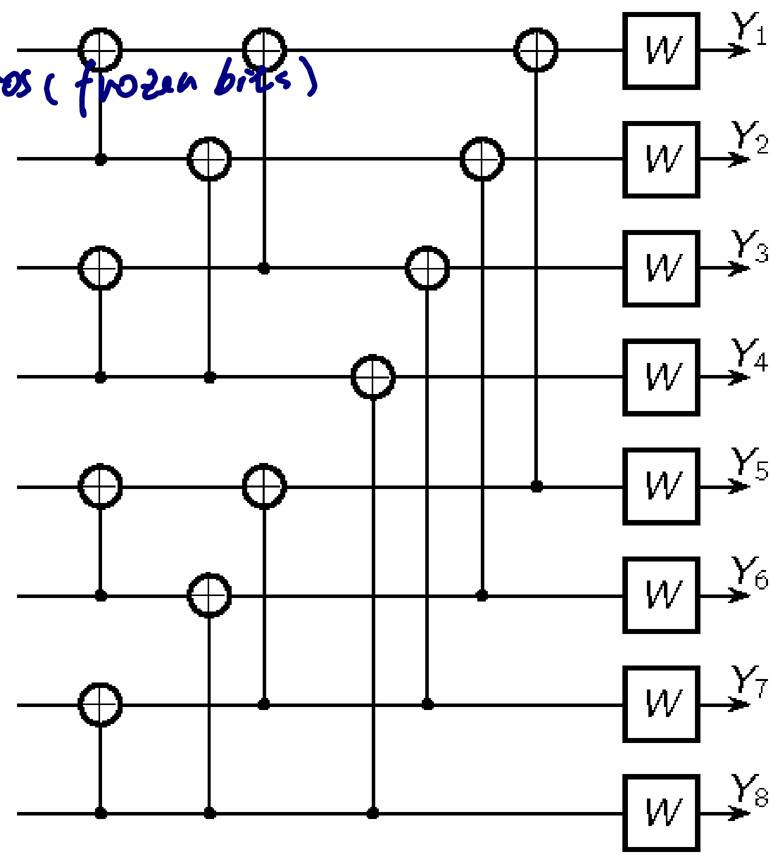
- Given $N = 2^n$, calculate $C\left(W_{b_1 b_2 \ldots b_n}\right)$ for all synthetic bit channels.
- Given rate $R < 1$ and $K = NR$, sort $C\left(W_{b_1 b_2 \ldots b_n}\right)$ in descending order and define the union of the indices of the first K elements as the information set Ω.
- Choose the information bits u^{Ω} and freeze u^{Ω^c} to be all-zero. Obtain the codeword $\mathbf{x} = \left(u^{\Omega}, u^{\Omega^c}\right) \cdot \mathbf{F}_N$.

Construction Example

- W is a BEC(0.5), $N = 8$, R=0.5.

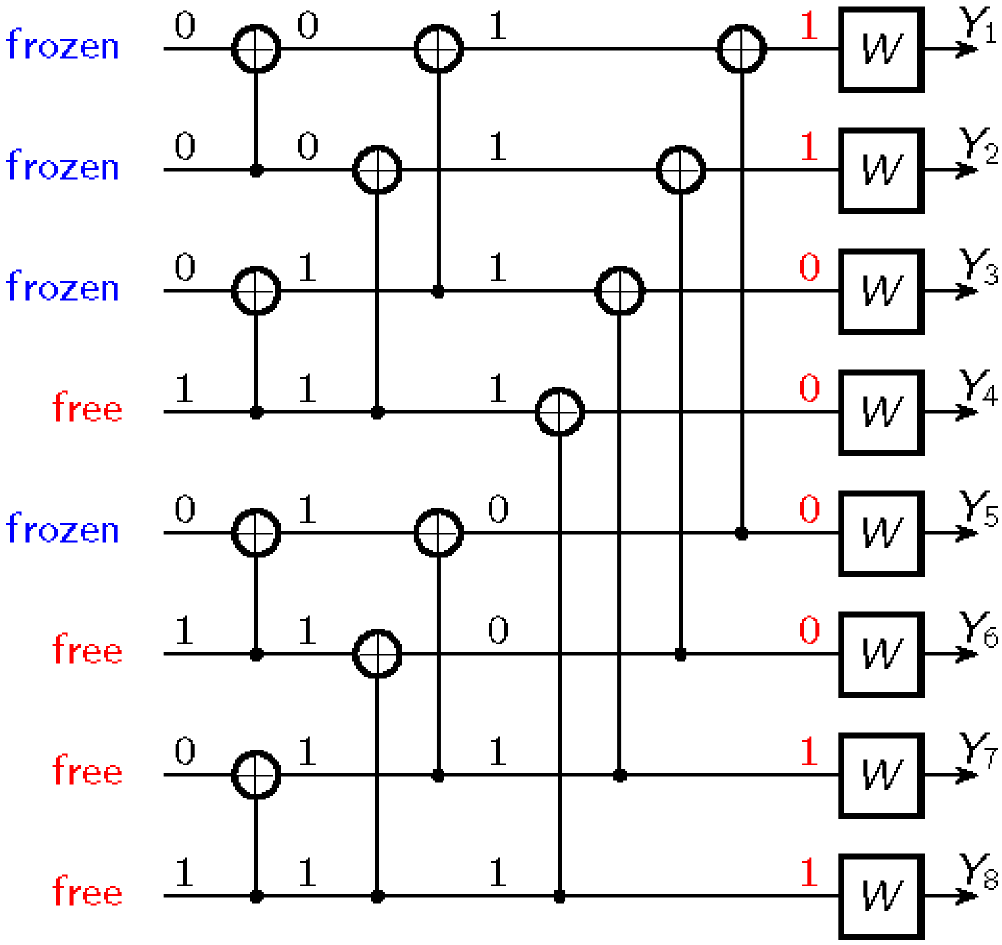
$C(W_i)$	Rank		
0.0039	8	frozen	0
0.1211	7	frozen	0
0.1914	6	frozen	0
0.6836	4	data	U_4
0.3164	5	frozen	0
0.8086	3	data	U_6
0.8789	2	data	U_7
0.9961	1	data	U_8

all zeros (frozen bits)



Construction Example

- W is a BEC(0.5), $N = 8$, R=0.5.



Encoding complexity $O(N \log N)$

Successive decoding

- For the decoding we need to compute the likelihood ratio for $u_i, i = (b_1 \cdots b_n)$

$$LR(u_i) = \frac{W_{b_1 b_2 \dots b_n}(\cdot\,|1)}{W_{b_1 b_2 \dots b_n}(\cdot\,|0)}$$

  If $i \in \Omega$, $\hat{u}_i = 1$ if $LR(u_i) > 1$; otherwise, $\hat{u}_i = 0$.

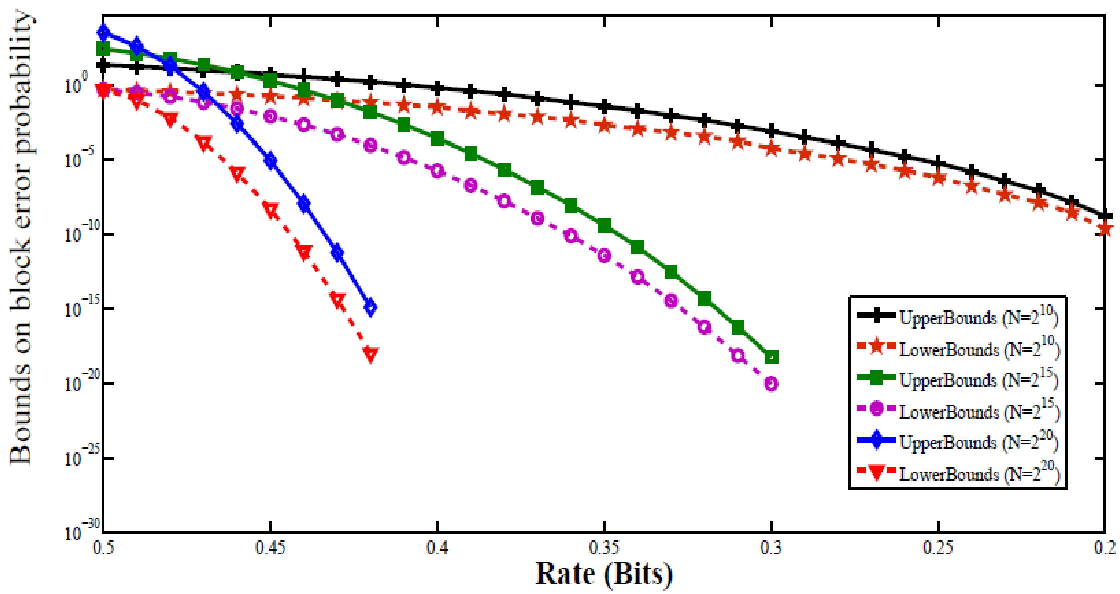
- Similar to $C\left(W_{b_1 b_2 \dots b_n}\right)$, $LR(u_i)$ can also be calculated recursively.
- For more details of the decoding, see

E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. on Information Theory,*, vol. 55, no. 7, pp. 3051–3073, 2009.

Probability of Error

- For a polar code with block length N and rate $R < C(W)$, the error probability under the successive cancellation decoding is given by

$$P_e \leq O(2^{-N^{\beta}}) \qquad \beta < 0.5$$



Error Bound of the SC decoding for BEC(0.5)

Polar code
source coding
channel coding

Polar Source Coding

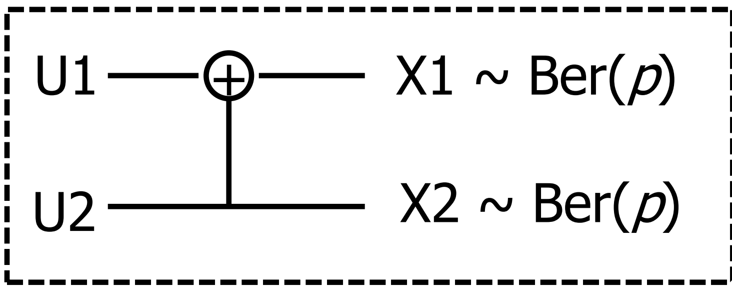
- Let x be a random variable generated by a Bernoulli source Ber(p), i.e.,
$$\Pr(x=0)=p \text{ and } \Pr(x=1)=1-p.$$
- The entropy (in bits) of x is
$$H(x) = -p\log_2 p - (1-p)\log_2(1-p)$$
- If $H(x) = 0$, i.e., $p = 0$ or 1, x is a constant, no need for compression.
- If $H(x) = 1$, i.e., $p = 0.5$, x is totally random, we cannot do any compression.
- In other cases, can the polarization technique be used to achieve rate $H(x)$?

Source Polarization

- Similar idea applies to source coding:

  general sources $\xrightarrow{\text{polarization}}$ extreme sources

- Basic source polarization



$$(\mathrm{U1}, \mathrm{U2}) = (\mathrm{X1}, \mathrm{X2})\mathbf{F}_2$$

$$H(\mathrm{U1}) + H(\mathrm{U2}|\mathrm{U1}) = H(\mathrm{U1}, \mathrm{U2}) = H(\mathrm{X1}, \mathrm{X2}) = 2H(x)$$

$$H(\mathrm{U1}) \geq H(x) \geq H(\mathrm{U2}|\mathrm{U1})$$

The process is entropy-conserving, but we obtain two new sources with higher and lower entropy than the original one.

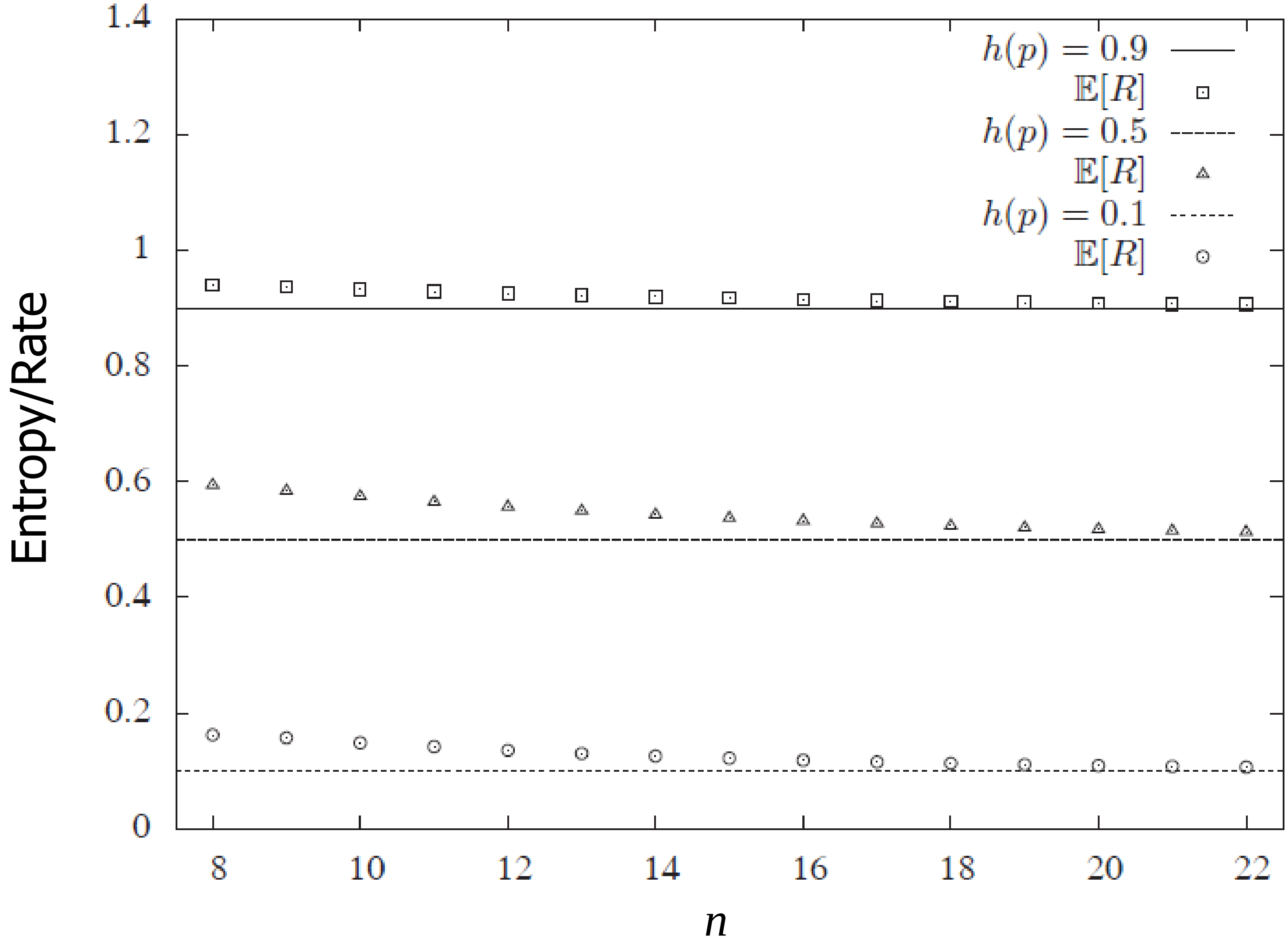
- Example: when $p = 0.11$, $H(x) = 0.5$, $H(\mathrm{U1}) = 0.713$, $H(\mathrm{U2}|\mathrm{U1}) = 0.287$.

Source Coding

- Keep polarizing by increasing N, the entropy of the synthetic sources tends to 0 or 1.
- Again, by the property of the martingale, the proportion of those sources with entropy close to 1 is close to $H(x)$.
- Source coding is realized by recording the indexes with entropy close to 1, while the rest bits can be recovered with high probability because their associated entropy is almost 0.
- For me details, see

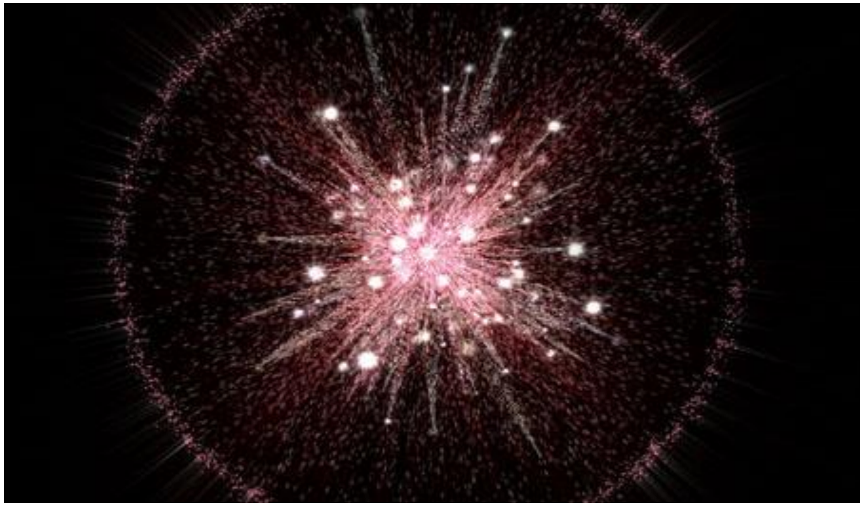
  E. Arikan, "Source polarization," *IEEE* ISIT 2010, pp. 899-903.

Performance



Extensions

- Polar codes also achieve capacity of other types of channels (discrete or continuous).
- Achieve entropy bound of other types of sources (lossless or lossy).
- Quantum polar codes, network information theory…



Big bang in information theory

Lecture 13

- Continuous Random Variables
- Differential Entropy
  - can be negative
  - not really a measure of the information in x
  - coordinate-dependent
- Maximum entropy distributions
  - Uniform over a finite range
  - Gaussian if a constant variance

Continuous Random Variables

Changing Variables

- pdf: $f_X(x)$ CDF: $F_X(x) = \int_{-\infty}^{x} f_X(t)\,dt$
- For $g(x)$ monotonic: $y = g(x) \Leftrightarrow x = g^{-1}(y)$

$$F_Y(y) = F_X\left(g^{-1}(y)\right) \quad \text{or} \quad 1 - F_X\left(g^{-1}(y)\right)$$ according to slope of $g(x)$

$$f_Y(y) = \frac{dF_Y(y)}{dy} = f_X\left(g^{-1}(y)\right)\left|\frac{dg^{-1}(y)}{dy}\right| = f_X(x)\left|\frac{dx}{dy}\right|$$ where $x = g^{-1}(y)$

- Examples:

$y = 4x \Rightarrow x = \frac{y}{4}$. $\left|\frac{dx}{dy}\right| = \frac{1}{4}$

$f_Y(y) = f_X(x)\cdot\left|\frac{dx}{dy}\right| = 0.5 \times \frac{1}{4} = \frac{1}{8}$ $y \in (0,8)$

$z = x^4 \Rightarrow x = z^{\frac{1}{4}}$ $\left|\frac{dx}{dz}\right| = \frac{1}{4} z^{-\frac{3}{4}}$

Suppose $f_X(x) = 0.5$ for $x \in (0,2) \Rightarrow F_X(x) = 0.5x$

$f_Z(z) = f_X(x)\cdot\left|\frac{dx}{dz}\right| = 0.5 \times \frac{1}{4} z^{-\frac{3}{4}} = \frac{1}{8} z^{-\frac{3}{4}}$ $z \in (0,16)$

(a) $y = 4x \Rightarrow x = 0.25y \Rightarrow f_Y(y) = 0.5 \times 0.25 = 0.125$ for $y \in (0,8)$

(b) $z = x^4 \Rightarrow x = z^{1/4} \Rightarrow f_Z(z) = 0.5 \times \frac{1}{4} z^{-3/4} = 0.125\, z^{-3/4}$ for $z \in (0,16)$

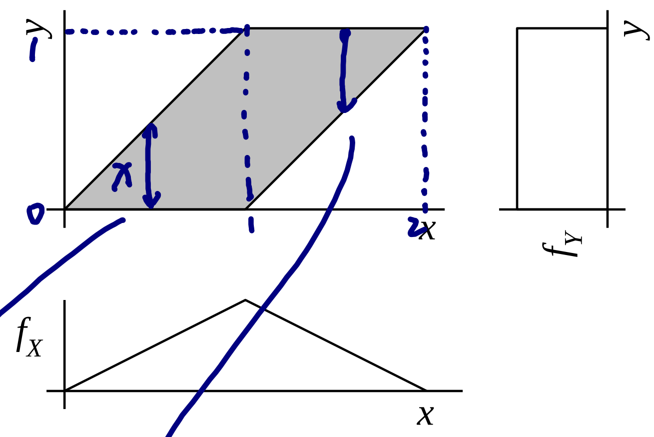
Joint Distributions

Joint pdf: $f_{X,Y}(x,y)$

Marginal pdf: $f_X(x) = \int_{-\infty}^{\infty} f_{X,Y}(x,y)dy$

Independence: $\Leftrightarrow \quad f_{X,Y}(x,y) = f_X(x) f_Y(y)$

Conditional pdf: $f_{X|Y}(x) = \dfrac{f_{X,Y}(x,y)}{f_Y(y)}$



Example:

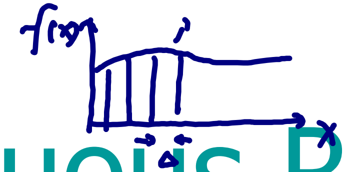
$$f_{X,Y} = 1 \text{ for } y \in (0,1), x \in (y, y+1)$$

$$f_{X|Y} = 1 \text{ for } x \in (y, y+1)$$

$$f_{Y|X} = \frac{1}{\min(x, 1-x)} \text{ for } y \in \left(\max(0, x-1), \min(x,1)\right)$$

$\int f(x)dx = \lim_{\Delta\to 0} \sum_i f(x_i)\Delta = 1$ $P_i = f(x_i)\Delta$

the prob. of x in slot i.



Entropy of Continuous R.V.

$H = -\sum_i P_i \log P_i = -\sum_i f(x_i)\Delta \log[f(x_i)\Delta] = -\sum_i f(x_i)\Delta[\log f(x_i) + \log\Delta]$

$= -\sum_i f(x_i)\Delta \log f(x_i) - \underbrace{\sum_i f(x_i)\Delta}_{1} \log\Delta \xrightarrow{\Delta\to 0} -\log\Delta - \underbrace{\int_{-\infty}^{\infty} f(x)\log f(x)dx}_{h(x)}$

- Given a continuous pdf $f(x)$, we divide the range of x into bins of width Δ
  - For each i, $\exists x_i$ with $f(x_i)\Delta = \int_{i\Delta}^{(i+1)\Delta} f(x)dx$ mean value theorem
- Define a discrete random variable Y
  - Y = $\{x_i\}$ and $p_Y = \{f(x_i)\Delta\}$
  - Scaled, quantised version of $f(x)$ with slightly unevenly spaced x_i
- $$H(y) = -\sum f(x_i)\Delta \log\left(f(x_i)\Delta\right)$$
  $$= -\log\Delta - \sum f(x_i)\log\left(f(x_i)\right)\Delta$$
  $$\xrightarrow[\Delta\to 0]{} -\log\Delta - \int_{-\infty}^{\infty} f(x)\log f(x)dx = -\log\Delta + h(x)$$
- Differential entropy: $\boxed{h(x) = -\int_{-\infty}^{\infty} f_x(x)\log f_x(x)dx}$

  - Similar to entropy of discrete r.v. but there are differences

Differential Entropy

Differential Entropy: $h(x) \stackrel{\Delta}{=} -\int_{-\infty}^{\infty} f_x(x) \log f_x(x) dx = E - \log f_x(x)$

Bad News:

- $h(x)$ does not give the amount of information in x
- $h(x)$ is not necessarily positive
- $h(x)$ changes with a change of coordinate system

Good News:

- $h_1(x) - h_2(x)$ does compare the uncertainty of two continuous random variables provided they are quantised to the same precision
- Relative Entropy and Mutual Information still work fine
- If the range of x is normalized to 1 and then x is quantised to n bits, the entropy of the resultant discrete random variable is approximately $h(x)+n$

Differential Entropy Examples

$h(X) = -\int_{-\infty}^{+\infty} f_X(x)\log f_X(x)\,dx = -\int_a^b \frac{1}{b-a}\log\frac{1}{b-a}\,dx = \log(b-a)$

- Uniform Distribution: $X \sim U(a,b)$
  - $f(x) = (b-a)^{-1}$ for $x \in (a,b)$ and $f(x) = 0$ elsewhere
  - $h(X) = -\int_a^b (b-a)^{-1}\log(b-a)^{-1}\,dx = \log(b-a)$
  - Note that $h(X) < 0$ if $(b-a) < 1$

$h(X) = -\int_{-\infty}^{+\infty} f_X(x)\log f_X(x)\,dx = -\int_{-\infty}^{+\infty} f(x)\log_e f(x)\,dx \cdot \log e = -\log e \int f(x)\left(\ln\frac{1}{\sqrt{2\pi}\sigma} - \frac{(x-\mu)^2}{2\sigma^2}\right)dx$

- Gaussian Distribution: $X \sim N(\mu,\sigma^2)$
  - $f(x) = \left(2\pi\sigma^2\right)^{-1/2}\exp\left(-\frac{1}{2}(x-\mu)^2\sigma^{-2}\right)$

$= -\frac{1}{2}\log e\int f(x)[-\ln 2\pi\sigma^2 - (x-\mu)^2\sigma^{-2}]\,dx = \frac{1}{2}\log e\left[\underbrace{\int f(x)\,dx}_{1}\cdot\ln 2\pi\sigma^2 + \underbrace{\int f(x)(x-\mu)^2\sigma^{-2}\,dx}_{\sigma^{-2}\underbrace{E[(x-\mu)^2]}_{\sigma^2}}\right]$

  - $h(X) = -(\log e)\int_{-\infty}^{\infty} f(x)\ln f(x)\,dx$

$f(x) = \frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$

$= \frac{1}{2}\log e\left(\ln(2\pi\sigma^2) + 1\right)$

$$= -\frac{1}{2}(\log e)\int_{-\infty}^{\infty} f(x)\left(-\ln(2\pi\sigma^2) - (x-\mu)^2\sigma^{-2}\right)$$

$= \frac{1}{2}[\log e\cdot\ln(2\pi\sigma^2) + \log e]$

$$= \frac{1}{2}(\log e)\left(\ln(2\pi\sigma^2) + \sigma^{-2}E\left((x-\mu)^2\right)\right)$$

$= \frac{1}{2}\left[\log e\cdot\frac{\log(2\pi\sigma^2)}{\log e} + \log e\right] = \frac{1}{2}\log(2\pi e\sigma^2)$

$$= \frac{1}{2}(\log e)\left(\ln(2\pi\sigma^2) + 1\right) = \tfrac{1}{2}\log\left(2\pi e\sigma^2\right) \cong \log(4.1\sigma) \text{ bits}$$

$\log_x y = \frac{\log_e y}{\log_e x}$

$$\underset{1\times n}{(x-m)^T}\,\underset{n\times n}{K^{-1}}\,\underset{n\times 1}{(x-m)} = tr\left((x-m)^T K^{-1}(x-m)\right) = tr\left((x-m)(x-m)^T K^{-1}\right)$$

Multivariate Gaussian

Const → tr(matrix)

Given mean, $\mathbf{m}$, and symmetric positive definite covariance matrix $\mathbf{K}$,

$$X_{1:n} \sim \mathbf{N}(\mathbf{m},\mathbf{K}) \quad \Leftrightarrow \quad f(\mathbf{x}) = \overset{\text{det}}{\left|2\pi\mathbf{K}\right|}^{-1/2} exp\left(-\frac{1}{2}\underset{1\times n}{(\mathbf{x}-\mathbf{m})^T}\,\underset{n\times n}{\mathbf{K}^{-1}}\,\underset{n\times 1}{(\mathbf{x}-\mathbf{m})}\right)$$

$$h(f) = -(\log e)\int f(\mathbf{x})\times\left(-\frac{1}{2}(\mathbf{x}-\mathbf{m})^T\mathbf{K}^{-1}(\mathbf{x}-\mathbf{m}) - \tfrac{1}{2}\ln\left|2\pi\mathbf{K}\right|\right)d\mathbf{x}$$

$$= \tfrac{1}{2}\log(e)\times\left(\ln\left|2\pi\mathbf{K}\right| + E\left((\mathbf{x}-\mathbf{m})^T\mathbf{K}^{-1}(\mathbf{x}-\mathbf{m})\right)\right)$$

$$= \tfrac{1}{2}\log(e)\times\left(\ln\left|2\pi\mathbf{K}\right| + E\,\mathrm{tr}\left((\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^T\mathbf{K}^{-1}\right)\right) \qquad \mathrm{tr}(\mathbf{AB}) = \mathrm{tr}(\mathbf{BA})$$

?

⇓ trace is a sum

$$= \tfrac{1}{2}\log(e)\times\left(\ln\left|2\pi\mathbf{K}\right| + \mathrm{tr}\left(\underbrace{E(\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^T}_{\triangleq K}\mathbf{K}^{-1}\right)\right) \qquad E_f\mathbf{x}\mathbf{x}^T = \mathbf{K}$$

$$= \tfrac{1}{2}\log(e)\times\left(\ln\left|2\pi\mathbf{K}\right| + \mathrm{tr}\left(\mathbf{K}\mathbf{K}^{-1}\right)\right) \quad = \tfrac{1}{2}\log(e)\times\left(\ln\left|2\pi\mathbf{K}\right| + n\right)$$

$$= \tfrac{1}{2}\log\left(e^n\right) + \tfrac{1}{2}\log\left(\left|\underset{n\times n}{2\pi\mathbf{K}}\right|\right) \qquad \mathrm{tr}(\mathbf{I}) = n = \ln(e^n)$$

can be negative if $|K|$ is small

$$= \tfrac{1}{2}\log\left(\left|2\pi e\mathbf{K}\right|\right) = \tfrac{1}{2}\log\left((2\pi e)^n\left|\mathbf{K}\right|\right) \text{ bits}$$

$\det(aA) = a^n \det(A)$

Other Differential Quantities

Joint Differential Entropy

$$h(x,y) = -\iint_{x,y} f_{x,y}(x,y) \log f_{x,y}(x,y)\,dxdy \;=\; E - \log f_{x,y}(x,y)$$

Conditional Differential Entropy

$$h(x \mid y) = -\iint_{x,y} f_{x,y}(x,y) \log f_{x,y}(x \mid y)\,dxdy \;=\; h(x,y) - h(y)$$

Mutual Information

$$I(x;y) = \iint_{x,y} f_{x,y}(x,y) \log \frac{f_{x,y}(x,y)}{f_x(x)\,f_y(y)}\,dxdy \;=\; h(x) + h(y) - h(x,y)$$

Relative Differential Entropy of two pdf's:

$$D(f \,\|\, g) = \int f(x) \log \frac{f(x)}{g(x)}\,dx$$

$$= -h_f(x) - E_f \log g(x)$$

(a) must have $f(x)=0 \Rightarrow g(x)=0$

(b) continuity $\Rightarrow 0 \log(0/0) = 0$

Differential Entropy Properties

Chain Rules

$$h(x,y) = h(x) + h(y \mid x) = h(y) + h(x \mid y)$$

$$I(x,y;z) = I(x;z) + I(y;z \mid x)$$

Information Inequality: $D(f \,\|\, g) \geq 0$

Proof: Define $S = \{\mathbf{x} : f(\mathbf{x}) > 0\}$ $D(f\|g) = -\int f(x)\log g(x)dx - h(x)$

$$-D(f \,\|\, g) = \int_{\mathbf{x} \in S} f(\mathbf{x}) \log \frac{g(\mathbf{x})}{f(\mathbf{x})} d\mathbf{x} = E_f\left(\log \frac{g(\mathbf{x})}{f(\mathbf{x})}\right) = -\int f(x)\log \frac{g(x)}{f(x)}dx$$

$$\leq \log\left(E\,\frac{g(\mathbf{x})}{f(\mathbf{x})}\right) = \log\left(\int_S f(\mathbf{x}) \frac{g(\mathbf{x})}{f(\mathbf{x})} d\mathbf{x}\right)$$

Jensen + $\log()$ is concave

$$= \log\left(\int_S g(\mathbf{x}) d\mathbf{x}\right) \leq \log 1 = 0$$

all the same as for discrete r.v. $H()$

Information Inequality Corollaries

Mutual Information ≥ 0

$$I(x;y) = D(f_{x,y} \,\|\, f_x f_y) \geq 0$$

Conditioning reduces Entropy

$$h(x) - h(x \mid y) = I(x;y) \geq 0$$

Independence Bound

$$h(x_{1:n}) = \sum_{i=1}^{n} h(x_i \mid x_{1:i-1}) \leq \sum_{i=1}^{n} h(x_i)$$

all the same as for $H()$

Change of Variable

Change Variable: $y = g(x)$

from earlier $f_y(y) = f_x\left(g^{-1}(y)\right)\left|\dfrac{dg^{-1}(y)}{dy}\right|$

$$f_y(y) = f_x(g^{-1}(y))\left|\frac{dg^{-1}(y)}{dy}\right|$$
$$h(y) = h(x) + E\log\left|\frac{dy}{dx}\right|$$

$$h(y) = -E\log(f_y(y)) = -E\log\left(f_x(g^{-1}(y))\right) - E\log\left|\frac{dx}{dy}\right|$$

$$= -E\log\left(f_x(x)\right) - E\log\left|\frac{dx}{dy}\right| = h(x) + E\log\left|\frac{dy}{dx}\right|$$

Examples:

- Translation: $y = x + a \quad\Rightarrow\quad dy/dx = 1 \quad\Rightarrow\quad h(y) = h(x)$
- Scaling: $y = cx \quad\Rightarrow\quad dy/dx = c \quad\Rightarrow\quad h(y) = h(x) + \log|c|$
- Vector version: $y_{1:n} = \mathbf{A}x_{1:n} \quad\Rightarrow\quad h(\mathbf{y}) = h(\mathbf{x}) + \log|\det(\mathbf{A})|$

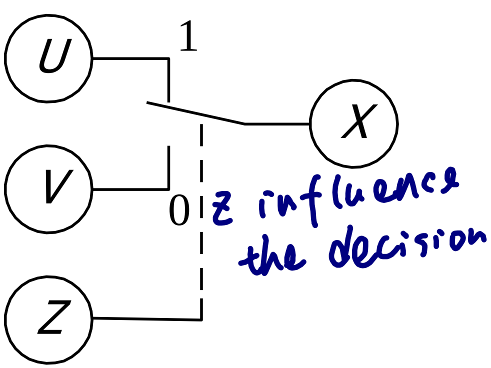
not the same as for $H()$

Concavity & Convexity

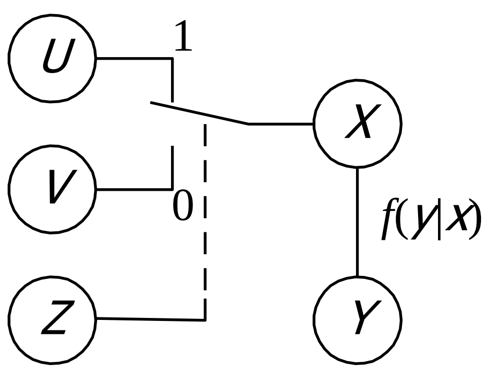
- Differential Entropy:
  - $h(x)$ is a concave function of $f_x(x)$ $\Rightarrow \exists$ a maximum
- Mutual Information:
  - $I(x\,;\,y)$ is a concave function of $f_x(x)$ for fixed $f_{y|x}(y)$
  - $I(x\,;\,y)$ is a convex function of $f_{y|x}(y)$ for fixed $f_x(x)$

Proofs:

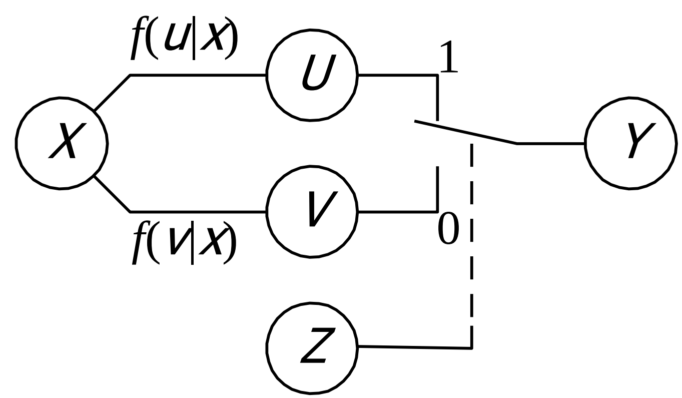
Exactly the same as for the discrete case: $\mathbf{p}_z = [1-\lambda,\ \lambda]^T$



$H(x) \geq H(x\,|\,z)$



$I(x\,;y) \geq I(x\,;y\,|z)$



$I(x\,;y) \leq I(x\,;y\,|z)$

Uniform Distribution Entropy

What distribution over the finite range (a,b) maximizes the entropy ?

$D(f\|u)=\int_{-\infty}^{+\infty} f(x)\log\frac{f(x)}{u(x)}dx$

$=-h(x)-\int_{-\infty}^{+\infty} f(x)\log u(x)dx$

$=-h(x)+\log(b-a)\geq 0$

$\therefore h(x)\leq\log(b-a)$

Answer: A uniform distribution $u(x)=(b-a)^{-1}$

Proof:

Suppose $f(x)$ is a distribution for $x\in(a,b)$

$$0\leq D(f\|u) = -h_f(x)-E_f\log u(x)$$

$$= -h_f(x)+\log(b-a)$$

$$\Rightarrow \boxed{h_f(x)\leq\log(b-a)}$$

$E_f \log(\phi(x)) = \log e \int f(x)$

Maximum Entropy Distribution

What zero-mean distribution maximizes the entropy on $(-\infty, \infty)^n$ for a given covariance matrix $\mathbf{K}$? (known covariance)

Answer: A multivariate Gaussian $\phi(\mathbf{x}) = |2\pi\mathbf{K}|^{-½} \exp\left(-½\,\mathbf{x}^T\mathbf{K}^{-1}\mathbf{x}\right)$

Proof: $0 \le D(f \,\|\, \phi) = -h_f(\mathbf{x}) - E_f \log \phi(\mathbf{x})$

$$\Rightarrow h_f(\mathbf{x}) \le -(\log e) E_f\left(-½ \ln\left(|2\pi\mathbf{K}|\right) - ½\,\mathbf{x}^T\mathbf{K}^{-1}\mathbf{x}\right)$$

$$h_f(\underline{x}) \le \tfrac{1}{2}\log|2\pi e\underline{K}| \quad = ½(\log e)\left(\ln\left(|2\pi\mathbf{K}|\right) + \operatorname{tr}\left(E_f\,\mathbf{x}\mathbf{x}^T\mathbf{K}^{-1}\right)\right)$$

$$= ½(\log e)\left(\ln\left(|2\pi\mathbf{K}|\right) + \operatorname{tr}(\mathbf{I})\right) \qquad E_f\mathbf{x}\mathbf{x}^T = \mathbf{K}$$

$$= ½\log\left(|2\pi e\mathbf{K}|\right) = h_\phi(\mathbf{x}) \qquad \operatorname{tr}(\mathbf{I}) = n = \ln(e^n)$$

Since translation doesn't affect $h(X)$, we can assume zero-mean w.l.o.g.

Summary

- Differential Entropy: $h(x) = -\int_{-\infty}^{\infty} f_x(x) \log f_x(x) dx$
  - Not necessarily positive
  - $h(x+a) = h(x)$, $h(ax) = h(x) + \log|a|$
- Many properties are formally the same
  - $h(x|y) \le h(x)$
  - $I(x; y) = h(x) + h(y) - h(x, y) \ge 0$, $D(f||g)=E \log(f/g) \ge 0$
  - $h(x)$ concave in $f_x(x)$; $I(x; y)$ concave in $f_x(x)$
- Bounds:
  - Finite range: Uniform distribution has max: $h(x) = \log(b-a)$

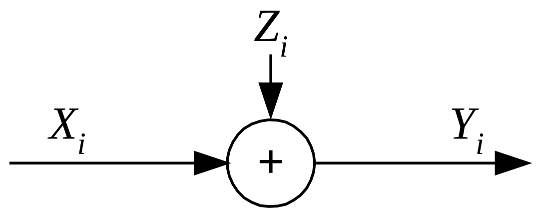
    (translation ~ mean)
  - Fixed Covariance: Gaussian has max: $h(x) = ½\log((2\pi e)^n|\mathbf{K}|)$

Lecture 14

- Discrete-Time Gaussian Channel Capacity
- Continuous Typical Set and AEP
- Gaussian Channel Coding Theorem
- Bandlimited Gaussian Channel
  - Shannon Capacity

Capacity of Gaussian Channel

Discrete-time channel: $y_i = x_i + z_i$



- Zero-mean Gaussian i.i.d. $z_i \sim N(0,N)$
- Average power constraint $n^{-1}\sum_{i=1}^{n} x_i^2 \le P$

$$Ey^2 = E(x+z)^2 = Ex^2 + 2E(x)E(z) + Ez^2 \le P + N$$

X,Z indep and $EZ=0$

Information Capacity

$h(x) = \frac{1}{2}\log((2\pi e)^n |K|)$

- Define information capacity: $C = \max_{Ex^2 \le P} I(x;y)$

$$I(x;y) = h(y) - h(y \mid x) = h(y) - h(x+z \mid x)$$

$E(y^2) \le P+N$

$$\overset{(a)}{=} h(y) - h(z \mid x) = h(y) - h(z)$$

(a) Translation independence

Power (var.) given, entropy is bounded by Gaussian distribution.

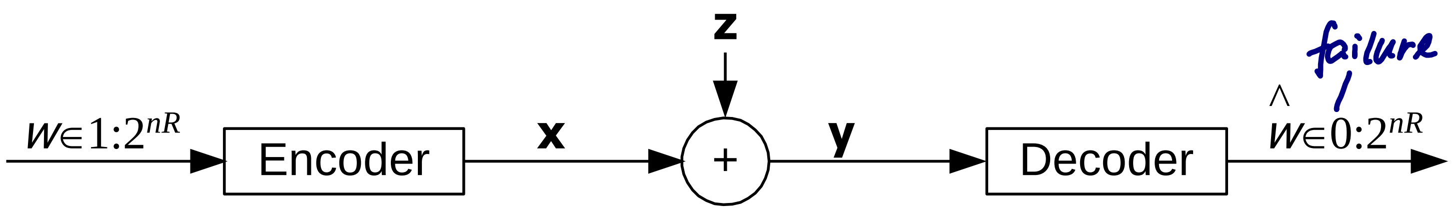
$$\le \tfrac{1}{2}\log 2\pi e(P+N) - \tfrac{1}{2}\log 2\pi e N$$

Gaussian Limit with equality when $x \sim N(0,P)$

$$= \frac{1}{2}\log\left(1 + \frac{P}{N}\right)$$

The optimal input is Gaussian

Achievability



- An (M,n) code for a Gaussian Channel with power constraint is
  - A set of M codewords $\mathbf{x}(w) \in \mathrm{X}^n$ for $w=1{:}M$ with $\mathbf{x}(w)^T\mathbf{x}(w) \le nP \quad \forall w$
  - A deterministic decoder $g(\mathbf{y}) \in 0{:}M$ where 0 denotes failure
  - Errors: codeword : λ_i $\quad$ $\max_i$: $\lambda^{(n)}$ $\quad$ $\operatorname{average}_i$: $P_e^{(n)}$
- Rate R is achievable if $\exists$ seq of $(2^{nR},n)$ codes with $\lambda^{(n)} \xrightarrow[n\to\infty]{} 0$
- Theorem: R achievable iff $R < C = \tfrac{1}{2}\log\left(1 + PN^{-1}\right)$ ♦

♦ = proved on next pages

$\#: \frac{\sqrt{n(P+N)}^{\,n}}{\sqrt{nN}^{\,n}} = \left(1+\frac{P}{N}\right)^{\frac{n}{2}}$

Argument by Sphere Packing

$R = \frac{1}{n}\log\# = \frac{1}{n}\cdot\frac{n}{2}\log\left(1+\frac{P}{N}\right) = \frac{1}{2}\log\left(1+\frac{P}{N}\right)$

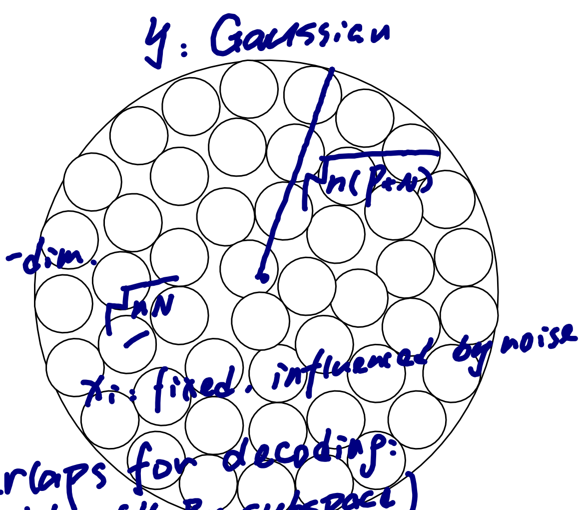
- Each transmitted $\mathbf{x}_i$ is received as a probabilistic cloud $\mathbf{y}_i$
  - cloud 'radius' = $\sqrt{\mathrm{Var}(\mathbf{y}\mid\mathbf{x})} = \sqrt{nN}$

Law of large numbers

- Energy of $\mathbf{y}_i$ constrained to $n(P+N)$ so clouds must fit into a hypersphere of radius $\sqrt{n(P+N)}$
- Volume of hypersphere $\propto r^{n}$ hypersphere: sphere in n-dim.
- Max number of non-overlapping clouds:

$$\frac{(nP+nN)^{\frac{1}{2}n}}{(nN)^{\frac{1}{2}n}} = 2^{\frac{1}{2}n\log\left(1+PN^{-1}\right)}$$

- Max achievable rate is $\frac{1}{2}\log(1+P/N)$



no overlaps for decoding:

$\# = \frac{\mathrm{Vol}(y: \text{Rx subspace})}{\mathrm{Vol}(x: \text{under noise})} = \frac{(\sqrt{n(P+N)})^n}{(\sqrt{nN})^n} = \left(1+\frac{P}{N}\right)^{\frac{n}{2}}$

$R = \frac{1}{n}\log\# = \frac{1}{n}\cdot\frac{n}{2}\cdot\log\left(1+\frac{P}{N}\right)$

Continuous AEP

Typical Set: Continuous distribution, discrete time i.i.d.

For any $\varepsilon>0$ and any n, the typical set with respect to $f(\mathbf{x})$ is

$$T_\varepsilon^{(n)} = \left\{ \mathbf{x} \in S^n : \left| -n^{-1} \log f(\mathbf{x}) - h(X) \right| \le \varepsilon \right\}$$

where S is the support of $f \Leftrightarrow \{\mathbf{x} : f(\mathbf{x})>0\}$

$$f(\mathbf{x}) = \prod_{i=1}^{n} f(x_i) \quad \text{since } x_i \text{ are independent}$$

$$h(X) = E - \log f(X) = -n^{-1} E \log f(\mathbf{x})$$

Typical Set Properties

1. $p(\mathbf{x} \in T_\varepsilon^{(n)}) > 1 - \varepsilon$ for $n > N_\varepsilon$ typicality — Proof: LLN
2. $(1-\varepsilon)2^{n(h(X)-\varepsilon)} \overset{n>N_\varepsilon}{\le} \mathrm{Vol}(T_\varepsilon^{(n)}) \le 2^{n(h(X)+\varepsilon)}$ size $\approx 2^{nh(x)}$ — Proof: Integrate max/min prob

where $\mathrm{Vol}(A) = \int_{\mathbf{x}\in A} d\mathbf{x}$

Continuous AEP Proof

Proof 1: By law of large numbers

$$-n^{-1}\log f(x_{1:n}) = -n^{-1}\sum_{i=1}^{n}\log f(x_i) \xrightarrow{prob} E - \log f(x) = h(x)$$

Reminder: $x_n \xrightarrow{\text{prob}} y \quad \Rightarrow \quad \forall \varepsilon > 0, \exists N_\varepsilon$ such that $\forall n > N_\varepsilon, \quad P\left(|x_n - y| > \varepsilon\right) < \varepsilon$

Proof 2a:

$$1-\varepsilon \le \int_{T_\varepsilon^{(n)}} f(\mathbf{x})d\mathbf{x} \qquad \text{for } n > N_\varepsilon$$

Property 1

$$\le 2^{-n(h(X)-\varepsilon)} \int_{T_\varepsilon^{(n)}} d\mathbf{x} = 2^{-n(h(X)-\varepsilon)}\,\mathrm{Vol}\left(T_\varepsilon^{(n)}\right)$$

max $f(x)$ within T

Proof 2b:

$$1 = \int_{S^n} f(\mathbf{x})d\mathbf{x} \ge \int_{T_\varepsilon^{(n)}} f(\mathbf{x})d\mathbf{x}$$

$$\ge 2^{-n(h(X)+\varepsilon)} \int_{T_\varepsilon^{(n)}} d\mathbf{x} = 2^{-n(h(X)+\varepsilon)}\,\mathrm{Vol}\left(T_\varepsilon^{(n)}\right)$$

min $f(x)$ within T

Jointly Typical Set

Jointly Typical: x_i, y_i i.i.d from $\Re^2$ with $f_{x,y}(x_i, y_i)$

jointly typical:
x typical
y typical
(x, y) typical

$$J_\varepsilon^{(n)} = \left\{ \mathbf{x}, \mathbf{y} \in \Re^{2n} : \left| -n^{-1} \log f_X(\mathbf{x}) - h(x) \right| < \varepsilon, \right.$$
$$\left| -n^{-1} \log f_Y(\mathbf{y}) - h(y) \right| < \varepsilon,$$
$$\left. \left| -n^{-1} \log f_{X,Y}(\mathbf{x}, \mathbf{y}) - h(x, y) \right| < \varepsilon \right\}$$

Properties:

1. Indiv p.d.: $\mathbf{x}, \mathbf{y} \in J_\varepsilon^{(n)} \Rightarrow \log f_{x,y}(\mathbf{x}, \mathbf{y}) = -nh(x, y) \pm n\varepsilon$
2. Total Prob: $p\left(\mathbf{x}, \mathbf{y} \in J_\varepsilon^{(n)}\right) > 1 - \varepsilon$ for $n > N_\varepsilon$
3. Size: $(1-\varepsilon) 2^{n(h(x,y)-\varepsilon)} \overset{n>N_\varepsilon}{\leq} \operatorname{Vol}\left(J_\varepsilon^{(n)}\right) \leq 2^{n(h(x,y)+\varepsilon)}$
4. Indep $\mathbf{x}', \mathbf{y}'$: $(1-\varepsilon) 2^{-n(I(x;y)+3\varepsilon)} \overset{n>N_\varepsilon}{\leq} p\left(\mathbf{x}', \mathbf{y}' \in J_\varepsilon^{(n)}\right) \leq 2^{-n(I(x;y)-3\varepsilon)}$

Proof of 4.: Integrate max/min $f(\mathbf{x}', \mathbf{y}') = f(\mathbf{x}') f(\mathbf{y}')$, then use known bounds on $\operatorname{Vol}(J)$

Gaussian Channel Coding Theorem

R is achievable iff $R < C = ½ \log\left(1 + PN^{-1}\right)$

Proof ($\Leftarrow$):

Choose $\varepsilon > 0$

Random codebook: $\mathbf{x}_w \in \Re^n$ for $w = 1 : 2^{nR}$ where x_w are i.i.d. $\sim N(0, P - \varepsilon)$

Use Joint typicality decoding

Errors: 1. Power too big $p\left(\mathbf{x}^T\mathbf{x} > nP\right) \to 0 \Rightarrow \leq \varepsilon$ for $n > M_\varepsilon$

2. $\mathbf{y}$ not J.T. with $\mathbf{x}$ $p(\mathbf{x}, \mathbf{y} \notin J_\varepsilon^{(n)}) < \varepsilon$ for $n > N_\varepsilon$

3. another $\mathbf{x}$ J.T. with $\mathbf{y}$ $\sum_{j=2}^{2^{nR}} p(\mathbf{x}_j, \mathbf{y}_i \in J_\varepsilon^{(n)}) \leq (2^{nR} - 1) \times 2^{-n(I(X;Y) - 3\varepsilon)}$

Total Err $P_\varepsilon^{(n)} \leq \varepsilon + \varepsilon + 2^{-n\left(I(X;Y) - R - 3\varepsilon\right)} \leq 3\varepsilon$ for large n if $R < I(X;Y) - 3\varepsilon$

Expurgation: Remove half of codebook*: $\lambda^{(n)} < 6\varepsilon$ now max error

We have constructed a code achieving rate $R - n^{-1}$

*:Worst codebook half includes $\mathbf{x}_i$: $\mathbf{x}_i^T\mathbf{x}_i > nP \Rightarrow \lambda_i = 1$

Gaussian Channel Coding Theorem

Proof ($\Rightarrow$): Assume $P_e^{(n)} \to 0$ and $n^{-1}\mathbf{x}^T\mathbf{x} < P$ for each $\mathbf{x}(w)$

$$nR = H(w) = I(w;y_{1:n}) + H(w \mid y_{1:n})$$

$$\le I(x_{1:n};y_{1:n}) + H(w \mid y_{1:n})$$ Data Proc Inequal

$$= h(y_{1:n}) - h(y_{1:n} \mid x_{1:n}) + H(w \mid y_{1:n})$$

$$\le \sum_{i=1}^{n} h(y_i) - h(z_{1:n}) + H(w \mid y_{1:n})$$ Indep Bound + Translation

$$\le \sum_{i=1}^{n} I(x_i;y_i) + 1 + nRP_e^{(n)}$$ Z i.i.d + Fano, $|W|=2^{nR}$

$$\le \sum_{i=1}^{n} \tfrac{1}{2}\log\left(1+PN^{-1}\right) + 1 + nRP_e^{(n)}$$ max Information Capacity

$$R \le \tfrac{1}{2}\log\left(1+PN^{-1}\right) + n^{-1} + RP_e^{(n)} \quad \to \tfrac{1}{2}\log\left(1+PN^{-1}\right)$$

Bandlimited Channel

- Channel bandlimited to $f \in (-W, W)$ and signal duration T
  - Not exactly
  - Most energy in the bandwidth, most energy in the interval
- Nyquist: Signal is defined by $2WT$ samples
  - white noise with double-sided p.s.d. $\frac{1}{2}N_0$ becomes i.i.d gaussian $N(0, \frac{1}{2}N_0)$ added to each coefficient
  - Signal power constraint = $P \Rightarrow$ Signal energy $\leq PT$
    - Energy constraint per coefficient: $n^{-1}\mathbf{x}^T\mathbf{x} < PT/2WT = \frac{1}{2}W^{-1}P$
- Capacity: $C' = \frac{1}{2}\log\left(1+\frac{P'}{N'}\right) = \frac{1}{2}\log\left(1+\frac{\frac{P}{2W}}{\frac{N_0}{2}}\right) = \frac{1}{2}\log\left(1+\frac{P}{N_0W}\right)$

$P' = \frac{P}{2W}$

$N' = \frac{N_0}{2}$

$$C = \frac{1}{2}\log\left(1+\frac{1/2 \cdot P/W}{N_0/2}\right) \times \frac{2WT}{T} = W\log\left(1+\frac{P}{WN_0}\right) \text{ bits/second}$$

$$C = \frac{\#\text{bits}}{T} = \frac{2WT \cdot \frac{1}{2}\log\left(1+\frac{P}{N_0W}\right)}{T} = W\log\left(1+\frac{P}{WN_0}\right)$$

- More precisely, it can be represented in a vector space of about $n=2WT$ dimensions with prolate spheroidal functions as an orthonormal basis

Compare discrete time version: $\frac{1}{2}\log(1+PN^{-1})$ bits per channel use

Limit of Infinite Bandwidth

$$C = W \log\left(1 + \frac{P}{WN_0}\right) \text{ bits/second}$$

$$C \underset{W\to\infty}{\to} \frac{P}{N_0}\log e$$

$$\lim_{W\to\infty} C = W \log\Big(1+\frac{P}{WN_0}\Big)^{\boxed{\frac{WN_0}{P}}} \cdot \frac{P}{WN_0}$$

$= 1$

$$= \log e \cdot \frac{P}{N_0}$$

Minimum signal to noise ratio (SNR)

$$\frac{E_b}{N_0} = \frac{PT_b}{N_0} = \frac{P/C}{N_0} \underset{W\to\infty}{\to} \ln 2 = -1.6 \text{ dB}$$

required for reliable transmission

Given capacity, trade-off between P and W

$$R \le W\log\Big(1+\frac{P}{WN_0}\Big)$$

- Increase P, decrease W

$$r \le \log\Big(1+\frac{P}{WN_0}\Big) = \log\Big(1+\frac{E_bR}{N_0W}\Big) = \log\Big(1+r\frac{E_b}{N_0}\Big)$$

- Increase W, decrease P

$$2^r \le 1 + r\frac{E_b}{N_0}$$

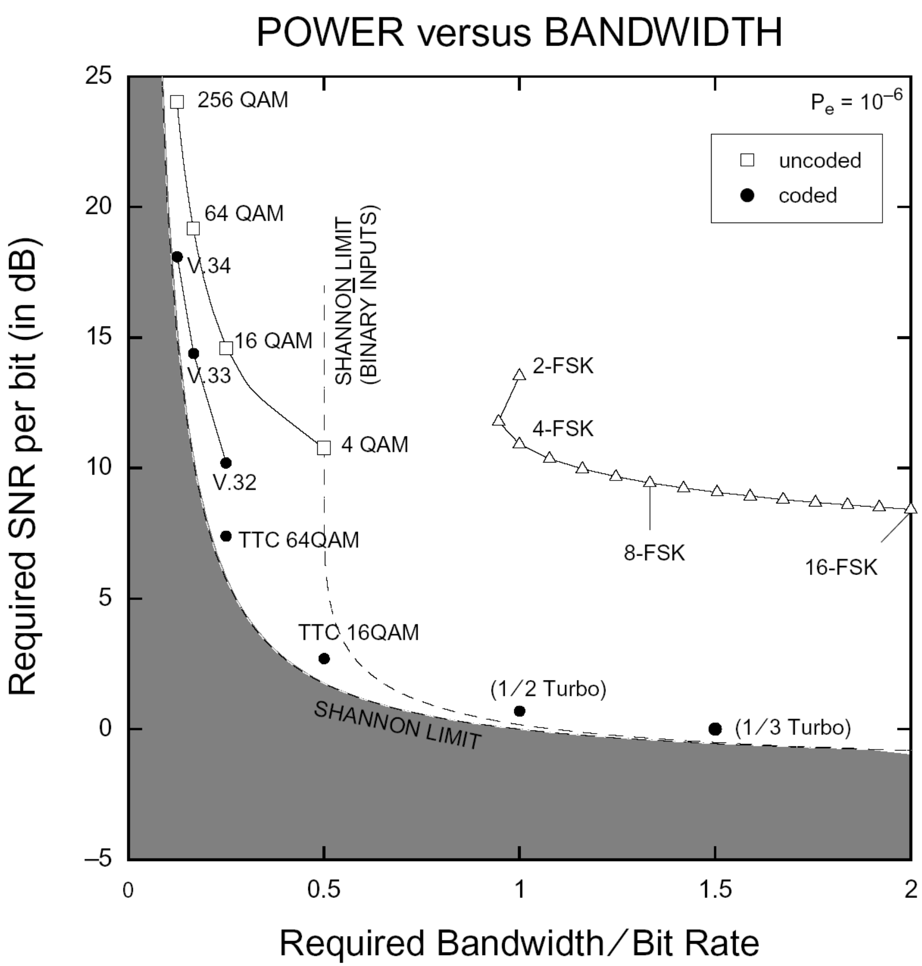
  - spread spectrum

$$\frac{E_b}{N_0} \ge \frac{2^r-1}{r} \ge \lim_{r\to0}\frac{2^r-1}{r} = \lim_{r\to0}\frac{e^{\ln 2\cdot r}-1}{r}$$

  - ultra wideband

$$\underline{\underline{\text{der.}}}\ \frac{\ln 2\, e^{\ln 2\cdot r}}{1} = \ln 2 = -1.6\text{ dB}$$

Channel Code Performance



- Power Limited
  - High bandwidth
  - Spacecraft, Pagers
  - Use QPSK/4-QAM
  - Block/Convolution Codes
- Bandwidth Limited
  - Modems, DVB, Mobile phones
  - 16-QAM to 256-QAM
  - Convolution Codes
- Value of 1 dB for space
  - Better range, lifetime, weight, bit rate
  - $80 M (1999)

Diagram from “An overview of Communications” by John R Barry

$$\lim_{W\to\infty} C = \lim_{W\to\infty} W \log\left(1+\frac{P}{WN_0}\right)$$

$$= \lim_{W\to\infty} \frac{\log\left(1+\frac{P}{WN_0}\right)}{\frac{1}{W}}$$

$$= \lim_{W\to\infty} \frac{\frac{1}{1+\frac{P}{WN_0}}\left(-\frac{1}{W^2}\right)\frac{P}{N_0}\log e}{-\frac{1}{W^2}} = \frac{P}{N_0}$$

Summary

- Gaussian channel capacity

discrete time

$$C = \frac{1}{2}\log\left(1+\frac{P}{N}\right) \quad \text{bits/transmission}$$

- Proved by using continuous AEP (or sphere packing)
- Bandlimited channel

band limited continuous time

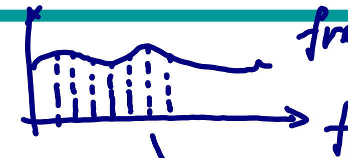
$$C = W \log\left(1+\frac{P}{WN_0}\right) \quad \text{bits/second}$$

  – Minimum SNR = –1.6 dB as W $\to\infty$

Lecture 15

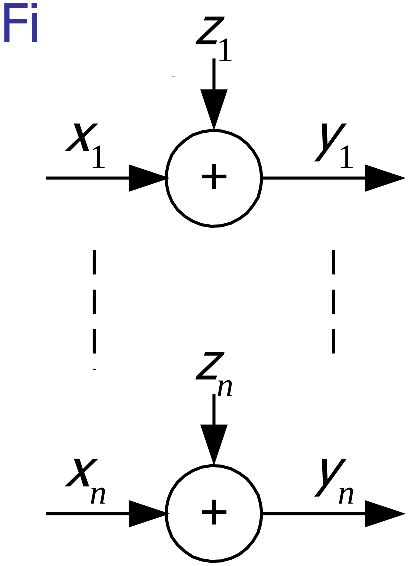
- Parallel Gaussian Channels
  - Waterfilling
- Gaussian Channel with Feedback
  - Memoryless: no gain
  - Memory: at most ½ bits/transmission

Parallel Gaussian Channels



- n independent Gaussian channels
  - A model for nonwhite noise wideband channel where each component represents a different frequency

    subbands: almost flat
  - e.g. digital audio, digital TV, Broadband ADSL, WiFi (multicarrier/OFDM) DMT
- Noise is independent $z_i \sim N(0,N_i)$
- Average Power constraint $E\mathbf{x}^T\mathbf{x} \le nP$
- Information Capacity: $C = \max\limits_{f(\mathbf{x}):E_f \mathbf{x}^T\mathbf{x} \le nP} I(\mathbf{x};\mathbf{y})$
- $R<C \Leftrightarrow R$ achievable
  - proof as before
- What is the optimal $f(\mathbf{x})$?



Parallel Gaussian: Max Capacity

Need to find $f(\mathbf{x})$: $C = \max_{f(\mathbf{x}):E_f \mathbf{x}^T\mathbf{x} \le nP} I(\mathbf{x};\mathbf{y})$

$$I(\mathbf{x};\mathbf{y}) = h(\mathbf{y}) - h(\mathbf{y}\,|\,\mathbf{x}) = h(\mathbf{y}) - h(\mathbf{z}\,|\,\mathbf{x})$$

Translation invariance

$$= h(\mathbf{y}) - h(\mathbf{z}) = h(\mathbf{y}) - \sum_{i=1}^{n} h(z_i)$$

joint entropy = sum of entropy

known variance:

$\mathbf{x},\mathbf{z}$ indep; Z_i indep

Gaussian y to maximise capacity

$$\overset{(a)}{\le} \sum_{i=1}^{n} \left(h(y_i) - h(z_i)\right) \overset{(b)}{\le} \sum_{i=1}^{n} ½ \log\left(1 + P_i N_i^{-1}\right)$$

- fixed channel entropy

(a) indep bound;
(b) capacity limit

= Gaussian input

Equality when: (a) y_i indep $\Rightarrow$ x_i indep; (b) $x_i \sim N(0, P_i)$

We need to find the P_i that maximise $\sum_{i=1}^{n} ½ \log\left(1 + P_i N_i^{-1}\right)$

Parallel Gaussian: Optimal Powers

We need to find the P_i that maximise $\log(e)\sum_{i=1}^{n} \frac{1}{2}\ln\left(1+P_i N_i^{-1}\right)$

- subject to power constraint $\sum_{i=1}^{n} P_i = nP$
- use Lagrange multiplier

$$J = \sum_{i=1}^{n} \frac{1}{2}\ln\left(1+P_i N_i^{-1}\right) - \lambda \sum_{i=1}^{n} P_i$$

$$\frac{\partial J}{\partial P_i} = \frac{1}{2}\left(P_i + N_i\right)^{-1} - \lambda = 0 \quad \Rightarrow \quad P_i + N_i = v$$

Also $\sum_{i=1}^{n} P_i = nP \quad \Rightarrow \quad v = P + n^{-1}\sum_{i=1}^{n} N_i$

$$\frac{\partial J}{\partial P_i} = \frac{1}{2}\,\frac{\frac{1}{N_i}}{1+\frac{P_i}{N_i}} - \lambda$$

$$\frac{1}{2}\,\frac{1}{N_i+P_i} = \lambda$$

$$P_i + N_i = \frac{1}{2\lambda} \triangleq v$$

$$\therefore P_i = v - N_i = \Big(\underbrace{P + \frac{\sum N_i}{n}}_{\text{avg}}\Big) - N_i$$

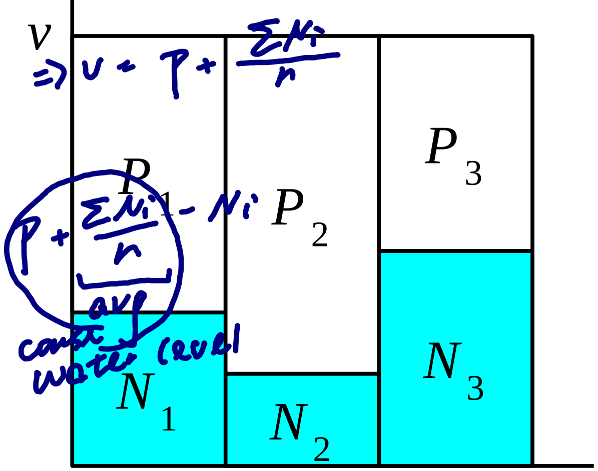
const water level

$P_i = v - N_i$

$\sum P_i = \sum (v - N_i)$

$= nv - \sum N_i = nP$

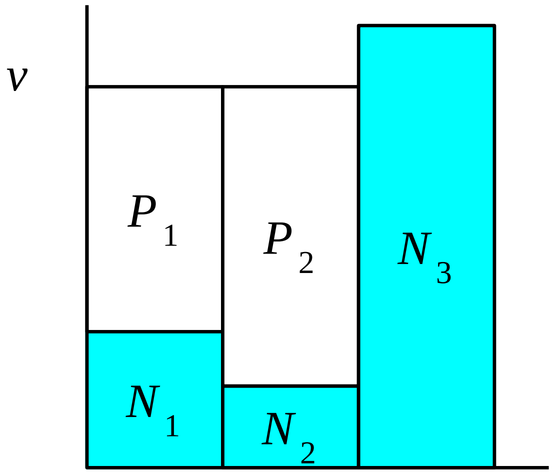
$\Rightarrow v = P + \frac{\sum N_i}{n}$



Water Filling: put most power into least noisy channels to make equal power + noise in each channel

Very Noisy Channels

- What if water is not enough?
- Must have $P_i \geq 0 \ \forall i$
- If $v < N_i$ then set $P_i = 0$ and recalculate v (i.e., $P_i = \max(v - N_i, 0)$)



Kuhn-Tucker Conditions:

(not examinable)

- Max $f(\mathbf{x})$ subject to $\mathbf{Ax}+\mathbf{b}=\mathbf{0}$ and $g_i(\mathbf{x}) \geq 0$ for $i \in 1:M$ with f, g_i concave
- set $J(\mathbf{x}) = f(\mathbf{x}) - \sum_{i=1}^{M} \mu_i g_i(\mathbf{x}) - \boldsymbol{\lambda}^T \mathbf{Ax}$
- Solution $\mathbf{x}_0$, λ, μ_i iff

$$\nabla J(\mathbf{x}_0) = 0, \quad \mathbf{Ax} + \mathbf{b} = \mathbf{0}, \quad g_i(\mathbf{x}_0) \geq 0, \quad \mu_i \geq 0, \quad \mu_i g_i(\mathbf{x}_0) = 0$$

Colored Gaussian Noise

- Suppose $\mathbf{y} = \mathbf{x} + \mathbf{z}$ where $E\,\mathbf{z}\mathbf{z}^T = \mathbf{K_Z}$ and $E\,\mathbf{x}\mathbf{x}^T = \mathbf{K_X}$
- We want to find $\mathbf{K_X}$ to maximize capacity subject to power constraint: $E\sum_{i=1}^{n} x_i^2 \le nP \quad \Leftrightarrow \quad \operatorname{tr}(\mathbf{K_X}) \le nP$
  - Find noise eigenvectors: $\mathbf{K}_Z = \mathbf{Q}\Lambda\mathbf{Q}^T$ with $\mathbf{Q}\mathbf{Q}^T = \mathbf{I}$
  - Now $\mathbf{Q}^T\mathbf{y} = \mathbf{Q}^T\mathbf{x} + \mathbf{Q}^T\mathbf{z} = \mathbf{Q}^T\mathbf{x} + \mathbf{w}$

    where $\mathrm{E}\ \mathbf{w}\mathbf{w}^T = \mathrm{E}\ \mathbf{Q}^T\mathbf{z}\mathbf{z}^T\mathbf{Q} = \mathrm{E}\ \mathbf{Q}^T\mathbf{K}_Z\mathbf{Q} = \mathbf{\Lambda}$ is diagonal
    - $\Rightarrow W_i$ are now independent (so previous result on P.G.C. applies)
  - Power constraint is unchanged $\operatorname{tr}\left(\mathbf{Q}^T\mathbf{K}_X\mathbf{Q}\right) = \operatorname{tr}\left(\mathbf{K}_X\mathbf{Q}\mathbf{Q}^T\right) = \operatorname{tr}\left(\mathbf{K}_X\right)$
  - Use water-filling and indep. messages $\mathbf{Q}^T\mathbf{K}_X\mathbf{Q} + \mathbf{\Lambda} = v\mathbf{I}$
  - Choose $\mathbf{Q}^T\mathbf{K}_X\mathbf{Q} = v\mathbf{I} - \mathbf{\Lambda}$ where $v = P + n^{-1}\operatorname{tr}(\mathbf{\Lambda})$

$$\Rightarrow \quad \mathbf{K}_X = \mathbf{Q}\left(v\mathbf{I} - \mathbf{\Lambda}\right)\mathbf{Q}^T$$

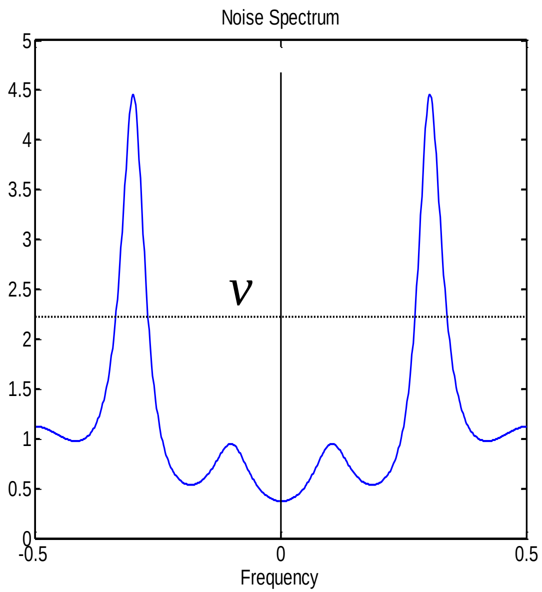
Power Spectrum Water Filling

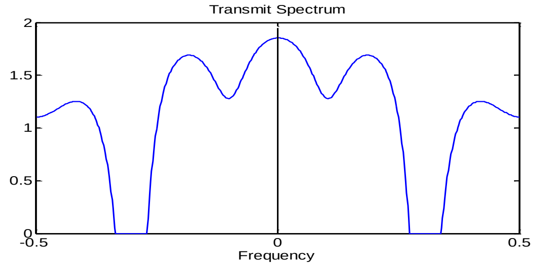
- If **z** is from a stationary process then $\mathrm{diag}(\Lambda) \underset{n\to\infty}{\rightarrow}$ power spectrum $N(f)$
  - To achieve capacity use waterfilling on noise power spectrum

$$P = \int_{-W}^{W} \max\left(v - N(f), 0\right) df$$

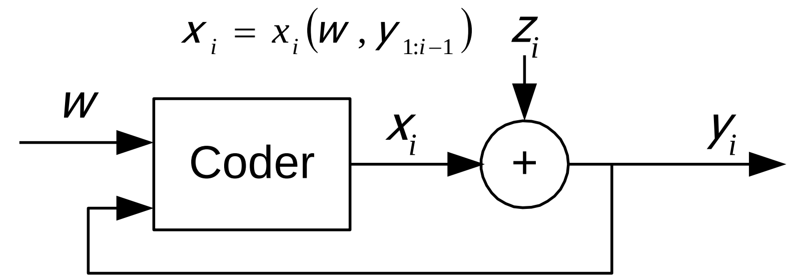
$$C = \int_{-W}^{W} \tfrac{1}{2} \log\left(1 + \frac{\max\left(v - N(f), 0\right)}{N(f)}\right) df$$

  - Waterfilling on spectral domain





Gaussian Channel + Feedback



Does Feedback add capacity ?

- White noise (& DMC) – No
- Coloured noise – Not much

$$I(w;\mathbf{y}) = h(\mathbf{y}) - h(\mathbf{y}\,|\,w) = h(\mathbf{y}) - \sum_{i=1}^{n} h(y_i\,|\,w, y_{1:i-1})$$ Chain rule

$\begin{cases} w \\ y \end{cases} \Rightarrow x \,.\quad \begin{cases} y \\ x \end{cases} \Rightarrow z$

$$= h(\mathbf{y}) - \sum_{i=1}^{n} h(y_i\,|\,w, y_{1:i-1}, x_{1:i}, z_{1:i-1})$$ $x_i = x_i(w, y_{1:i-1})$, $\mathbf{z} = \mathbf{y} - \mathbf{x}$

$$= h(\mathbf{y}) - \sum_{i=1}^{n} h(z_i\,|\,w, y_{1:i-1}, x_{1:i}, z_{1:i-1})$$ $\mathbf{z} = \mathbf{y} - \mathbf{x}$ and translation invariance

$z_i \sim z_{1:i-1}$

$$= h(\mathbf{y}) - \sum_{i=1}^{n} h(z_i\,|\,z_{1:i-1})$$ $\mathbf{z}$ may be colored; z_i depends only on $z_{1:i-1}$

max $= \frac{1}{2}\log|2\pi e K_z|$

fixed

$$= h(\mathbf{y}) - h(\mathbf{z})$$ Chain rule, $h(\mathbf{z}) = \frac{1}{2}\log\left(\left|2\pi e\mathbf{K}_{\mathbf{z}}\right|\right)$ bits

known var.

$$\le \frac{1}{2}\log\frac{|\mathbf{K}_{\mathbf{y}}|}{|\mathbf{K}_{\mathbf{z}}|}$$

$\Rightarrow$ maximize $I(w;\mathbf{y})$ by maximizing $h(\mathbf{y}) \Rightarrow \mathbf{y}$ gaussian

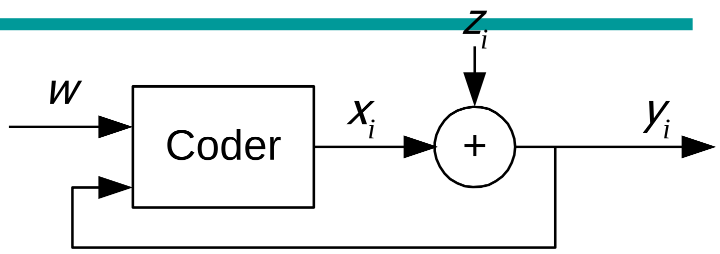
$\Rightarrow$ we can take $\mathbf{z}$ and $\mathbf{x} = \mathbf{y} - \mathbf{z}$ jointly gaussian

$y = x + z$

$K_y = K_x + K_z$ if no feedback

$\neq$ if with feedback

Maximum Benefit of Feedback



$$C_{n,FB} = \max_{\mathrm{tr}(\mathbf{K_x}) \le nP} \tfrac{1}{2} n^{-1} \log \frac{|\mathbf{K_y}|}{|\mathbf{K_z}|}$$

$$\le \max_{\mathrm{tr}(\mathbf{K_x}) \le nP} \tfrac{1}{2} n^{-1} \log \frac{|2(\mathbf{K_x} + \mathbf{K_z})|}{|\mathbf{K_z}|}$$

Lemmas 1 & 2:

$|2(\mathbf{Kx}+\mathbf{Kz})| \ge |\mathbf{Ky}|$

$$= \max_{\mathrm{tr}(\mathbf{K_x}) \le nP} \tfrac{1}{2} n^{-1} \log \frac{2^n |\mathbf{K_x} + \mathbf{K_z}|}{|\mathbf{K_z}|}$$

$|k\mathbf{A}| = k^n |(\mathbf{A}|$

$$= \tfrac{1}{2} + \underbrace{\max_{\mathrm{tr}(\mathbf{K_x}) \le nP} \tfrac{1}{2} n^{-1} \log \frac{\overbrace{|\mathbf{K_x} + \mathbf{K_z}|}^{= K_y \text{ (w/o feedback)}}}{|\mathbf{K_z}|}}_{\text{capacity w/o feedback}} = \tfrac{1}{2} + C_n \text{ bits / transmission}$$

Ky = **Kx**+**Kz** if no feedback

C_n: capacity without feedback

Having feedback adds at most ½ bit per transmission for colored Gaussian noise channels

Max Benefit of Feedback: Lemmas

Lemma 1: $\mathbf{K}_{\mathbf{x}+\mathbf{z}}+\mathbf{K}_{\mathbf{x}-\mathbf{z}} = 2(\mathbf{K}_{\mathbf{x}}+\mathbf{K}_{\mathbf{z}})$

$$\begin{aligned}
\mathbf{K}_{\mathbf{x}+\mathbf{z}} + \mathbf{K}_{\mathbf{x}-\mathbf{z}} &= E(\mathbf{x}+\mathbf{z})(\mathbf{x}+\mathbf{z})^T + E(\mathbf{x}-\mathbf{z})(\mathbf{x}-\mathbf{z})^T \\
&= E\left(\mathbf{xx}^T + \mathbf{xz}^T + \mathbf{zx}^T + \mathbf{zz}^T \quad + \quad \mathbf{xx}^T - \mathbf{xz}^T - \mathbf{zx}^T + \mathbf{zz}^T\right) \\
&= E\left(2\mathbf{xx}^T + 2\mathbf{zz}^T\right) = 2(\mathbf{K}_{\mathbf{x}} + \mathbf{K}_{\mathbf{z}})
\end{aligned}$$

Lemma 2: If $\mathbf{F},\mathbf{G}$ are positive definite then $|\mathbf{F}+\mathbf{G}| \geq |\mathbf{F}|$

Consider two indep random vectors $\mathbf{f}\sim N(0,\mathbf{F})$, $\mathbf{g}\sim N(0,\mathbf{G})$

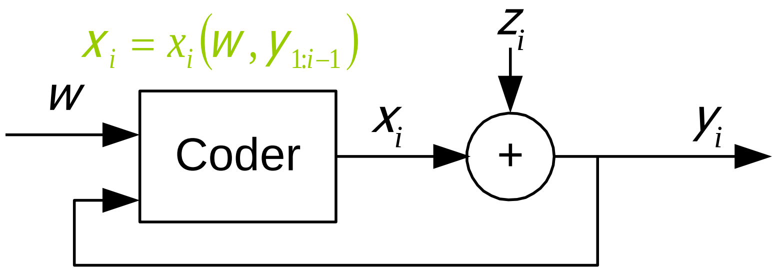
$$\begin{aligned}
½\log\left((2\pi e)^n |\mathbf{F}+\mathbf{G}|\right) &= h(\mathbf{f}+\mathbf{g}) \\
&\geq h(\mathbf{f}+\mathbf{g} \mid \mathbf{g}) = h(\mathbf{f} \mid \mathbf{g}) \\
&= h(\mathbf{f}) = ½\log\left((2\pi e)^n |\mathbf{F}|\right)
\end{aligned}$$

Conditioning reduces $h()$
Translation invariance

$\mathbf{f},\mathbf{g}$ independent

Hence: $|2(\mathbf{K}_{\mathbf{x}}+\mathbf{K}_{\mathbf{z}})| = |\mathbf{K}_{\mathbf{x}+\mathbf{z}}+\mathbf{K}_{\mathbf{x}-\mathbf{z}}| \geq |\mathbf{K}_{\mathbf{x}+\mathbf{z}}| = |\mathbf{K}_{\mathbf{y}}|$

Gaussian Feedback Coder



$\mathbf{x}$ and $\mathbf{z}$ jointly gaussian $\Rightarrow$

$$\mathbf{x} = \mathbf{B}\mathbf{z} + \mathbf{v}(W)$$

where $\mathbf{v}$ is indep of $\mathbf{z}$ and

$\mathbf{B}$ is strictly lower triangular since x_i indep of z_j for $j>i$.

$$\mathbf{y} = \mathbf{x} + \mathbf{z} = (\mathbf{B} + \mathbf{I})\mathbf{z} + \mathbf{v}$$

$$\mathbf{K}_{\mathbf{y}} = E\mathbf{y}\mathbf{y}^T = E\left((\mathbf{B} + \mathbf{I})\mathbf{z}\mathbf{z}^T(\mathbf{B} + \mathbf{I})^T + \mathbf{v}\mathbf{v}^T\right) = (\mathbf{B} + \mathbf{I})\mathbf{K}_{\mathbf{z}}(\mathbf{B} + \mathbf{I})^T + \mathbf{K}_{\mathbf{v}}$$

$$\mathbf{K}_{\mathbf{x}} = E\mathbf{x}\mathbf{x}^T = E\left(\mathbf{B}\mathbf{z}\mathbf{z}^T\mathbf{B}^T + \mathbf{v}\mathbf{v}^T\right) = \mathbf{B}\mathbf{K}_{\mathbf{z}}\mathbf{B}^T + \mathbf{K}_{\mathbf{v}}$$

Capacity: $$C_{n,FB} = \max_{\mathbf{K}_{\mathbf{v}},\mathbf{B}} \tfrac{1}{2} n^{-1} \frac{|\mathbf{K}_{\mathbf{y}}|}{|\mathbf{K}_{\mathbf{z}}|} = \max_{\mathbf{K}_{\mathbf{v}},\mathbf{B}} \tfrac{1}{2} n^{-1} \log \frac{\left|(\mathbf{B} + \mathbf{I})\mathbf{K}_{\mathbf{z}}(\mathbf{B} + \mathbf{I})^T + \mathbf{K}_{\mathbf{v}}\right|}{|\mathbf{K}_{\mathbf{z}}|}$$

subject to $$\mathbf{K}_{\mathbf{x}} = \operatorname{tr}\left(\mathbf{B}\mathbf{K}_{\mathbf{z}}\mathbf{B}^T + \mathbf{K}_{\mathbf{v}}\right) \le nP$$

hard to solve ☹

Optimization can be done numerically

Gaussian Feedback: Toy Example

$$n = 2, \quad P = 2, \quad \mathbf{K_z} = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 & 0 \\ b & 0 \end{pmatrix}$$

$$\mathbf{x} = \mathbf{B}\mathbf{z} + \mathbf{v} \Rightarrow x_1 = v_1,\ x_2 = bz_1 + v_2$$

Goal: Maximize (w.r.t. $\mathbf{K_v}$ and b)

$$|\mathbf{K_Y}| = |(\mathbf{B}+\mathbf{I})\mathbf{K_z}(\mathbf{B}+\mathbf{I})^T + \mathbf{K_v}|$$

Subject to:

$\mathbf{K_v}$ must be positive definite

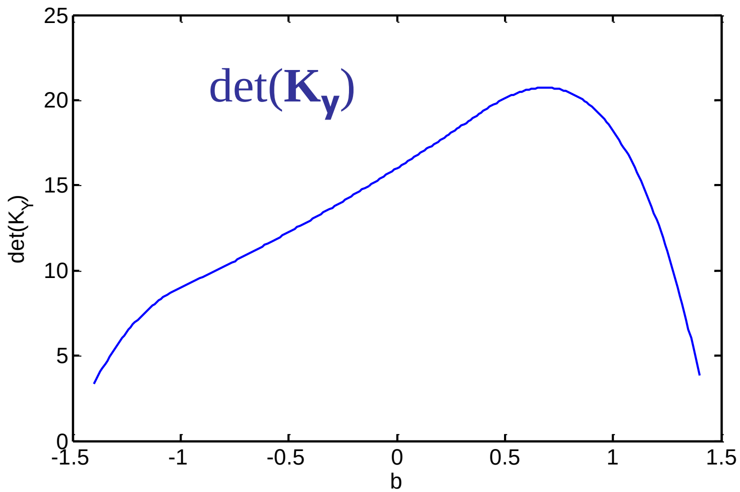
Power constraint : $\mathrm{tr}\left(\mathbf{B}\mathbf{K_z}\mathbf{B}^T + \mathbf{K_v}\right) \le 4$

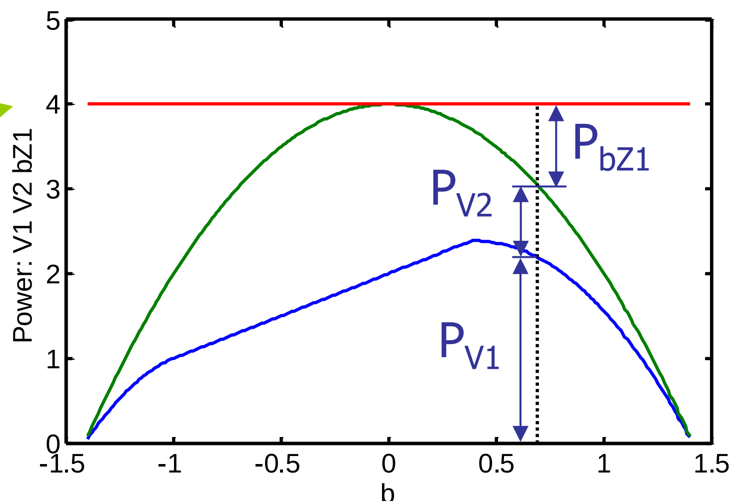
Solution (via numerically search):

b=0: $|\mathbf{K_y}|=16$ $C=0.604$ bits

b=0.69: $|\mathbf{K_y}|=20.7$ $C=0.697$ bits

Feedback increases C by 16%





$$P_{x1} = P_{v1} \qquad P_{x2} = P_{v2} + P_{bz1}$$

Summary

- Water-filling for parallel Gaussian channel

$$C = \sum_{i=1}^{n} \frac{1}{2} \log\left(1 + \frac{(v - N_i)^+}{N_i}\right)$$

$$x^+ = \max(x, 0)$$

$$\sum (v - N_i)^+ = nP$$

- Colored Gaussian noise

$$C = \sum_{i=1}^{n} \frac{1}{2} \log\left(1 + \frac{(v - \lambda_i)^+}{\lambda_i}\right)$$

λ_i eigenvealues of $\mathbf{K}_{\mathbf{z}}$

$$\sum (v - \lambda_i)^+ = nP$$

- Continuous Gaussian channel

$$C = \int_{-W}^{W} ½ \log\left(1 + \frac{\left(v - N(f)\right)^+}{N(f)}\right) df$$

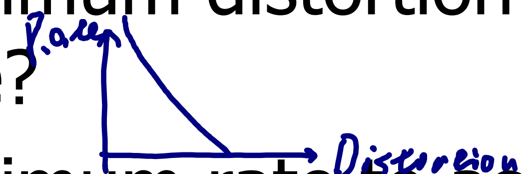
- Feedback bound

$$C_{n,FB} \leq C_n + \frac{1}{2}$$

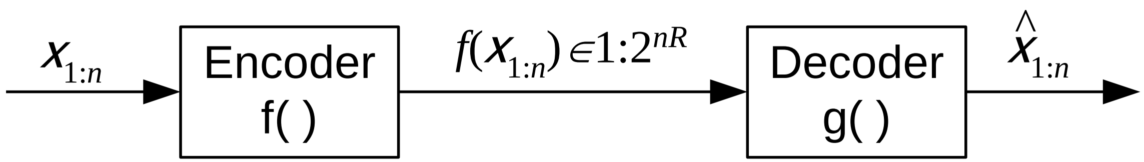
(= holds for coloured Gaussian noise)

Lecture 16

- Lossy Source Coding
  - For both discrete and continuous sources
  - Bernoulli Source, Gaussian Source
- Rate Distortion Theory
  - What is the minimum distortion achievable at a particular rate?
  - What is the minimum rate to achieve a particular distortion?
- Channel/Source Coding Duality



Lossy Source Coding



Distortion function: $d(x, \hat{x}) \geq 0$

- examples: (i) $d_S(x, \hat{x}) = (x - \hat{x})^2$ (ii) $d_H(x, \hat{x}) = \begin{cases} 0 & x = \hat{x} \\ 1 & x \neq \hat{x} \end{cases}$
- sequences: $d(\mathbf{x}, \hat{\mathbf{x}}) \overset{\Delta}{=} n^{-1} \sum_{i=1}^{n} d(x_i, \hat{x}_i)$

average distortion

f( ): encoding
g( ): decoding
$\hat{x} = g(f(x))$

Distortion of Code $f_n(\,)$, $g_n(\,)$: $D = E_{\mathbf{x} \in X^n} d\left(\mathbf{x}, \hat{\mathbf{x}}\right) = E\ d\left(\mathbf{x}, g(f(\mathbf{x}))\right)$

Rate distortion pair (R,D) is achievable for source X if

$\exists$ a sequence $f_n(\,)$ and $g_n(\,)$ such that $\lim_{n \to \infty} E_{\mathbf{x} \in X^n} d(\mathbf{x}, g_n(f_n(\mathbf{x}))) \leq D$

Rate Distortion Function

Rate Distortion function for $\{x_i\}$ with pdf $p(\mathbf{x})$ is defined as

$$R(D) = \min\{R\} \text{ such that } (R,D) \text{ is achievable}$$

Theorem: $\boxed{R(D) = \min I(X;\hat{X})}$ over all $p(x,\hat{x})$ such that :

$\boxed{R(D) = \min I(X;\hat{X})}$

$D=0 \Rightarrow X=\hat{X} \Rightarrow R(D)=I(X;X)=H(X)$

$D=1 \Rightarrow X=\,? \Rightarrow R(D)=I(X;\,?)=0$

rate distortion function $R(D)$:

(a) $p(x)$ is correct

(b) $E_{x,\hat{x}}\, d(x,\hat{x}) \le D$

– this expression is the Rate Distortion function for X

. the rate R required to describe source information with the expectation of distortion function smaller than or equal to D.

Proof is not examinable

Lossless coding: If $D=0$ then we have $R(D) = I(x;x) = H(x)$

◆ $p(x,\hat{x}) = p(x)q(\hat{x}\,|\,x)$

$R(D)$ bound for Bernoulli Source

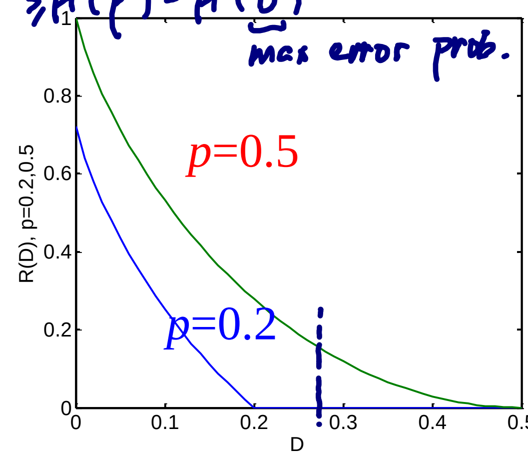
$I(X, \hat{X}) = H(X) - H(X|\hat{X})$

$R(D) = \min I(X, \hat{X})$ $= H(p) - H(X \oplus \hat{X} | \hat{X}) \geq H(p) - H(\underbrace{X \oplus \hat{X}}_{\text{error}})$

Bernoulli: X = $[0,1]$, $\mathbf{p}_X = [1-p, p]$ assume $p \leq ½$

$\geq H(p) - H(\underbrace{D}_{\text{max error prob.}})$

- Hamming Distance: $d(x, \hat{x}) = x \oplus \hat{x}$
- If $D \geq p$, $R(D)=0$ since we can set $g(\ ) \equiv 0$
- For $D < p \leq ½$, if $E\ d(x, \hat{x}) \leq D$ then



$$I(x; \hat{x}) = H(x) - H(x | \hat{x})$$

$$= H(p) - H(x \oplus \hat{x} | \hat{x})$$

$$\geq H(p) - H(\underbrace{x \oplus \hat{x}}_{e})$$

$$\geq H(p) - H(D)$$

$\oplus$ is one-to-one

Conditioning reduces entropy

Prob.$(x \oplus \hat{x} = 1) \leq D$ for $D \leq ½$

$H(x \oplus \hat{x}) \leq H(D)$ as $H(p)$ monotonic

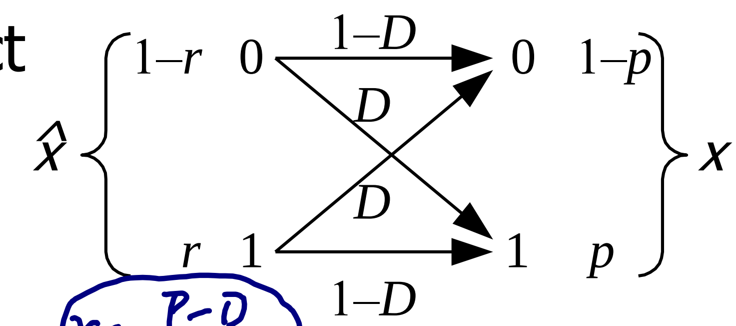
Hence $\boxed{R(D) \geq H(p) - H(D)}$

$R(D)$ for Bernoulli source

We know optimum satisfies $R(D) \geq H(p) - H(D)$

- We show we can find a $p(\hat{x}, x)$ that attains this.
- Peculiarly, we consider a channel with $\hat{x}$ as the input and error probability D

Now choose r to give x the correct probabilities:



$$r(1-D) + (1-r)D = p$$

$$\Rightarrow r = (p-D)(1-2D)^{-1}, \qquad D \leq p$$

$r = \frac{p-D}{1-2D}$

Now $I(x;\hat{x}) = H(x) - H(x \mid \hat{x}) = H(p) - H(D)$

and $p(x \neq \hat{x}) = D \quad \Rightarrow \quad \text{distortion} \leq D$

Hence $R(D) = H(p) - H(D)$

If $D \geq p$ or $D \geq 1-p$, we can achieve $R(D)$=0 trivially.

$R(D)$ bound for Gaussian Source

- Assume $X \sim N(0,\sigma^2)$ and $d(x,\hat{x}) = (x-\hat{x})^2$
- Want to minimize $I(X;\hat{X})$ subject to $E(X-\hat{X})^2 \le D$

$$I(X;\hat{X}) = h(X) - h(X\,|\,\hat{X})$$

$$= \tfrac{1}{2}\log 2\pi e\sigma^2 - h(X-\hat{X}\,|\,\hat{X})$$ Translation Invariance

$$\ge \tfrac{1}{2}\log 2\pi e\sigma^2 - h(X-\hat{X})$$ Conditioning reduces entropy

$$\ge \tfrac{1}{2}\log 2\pi e\sigma^2 - \tfrac{1}{2}\log\left(2\pi e\,\mathrm{Var}\left(X-\hat{X}\right)\right)$$ Gauss maximizes entropy for given covariance

$$\ge \tfrac{1}{2}\log 2\pi e\sigma^2 - \tfrac{1}{2}\log 2\pi eD$$ require $\mathrm{Var}\left(X-\hat{X}\right) \le E(X-\hat{X})^2 \le D$

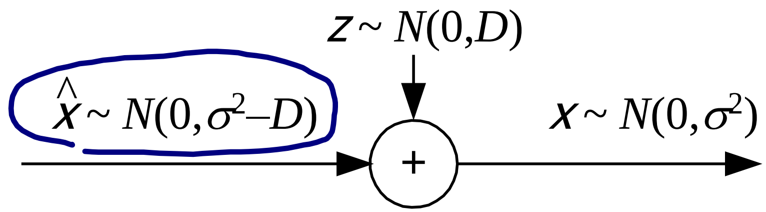
$$\boxed{I(X;\hat{X}) \ge \max\left(\tfrac{1}{2}\log\frac{\sigma^2}{D}, 0\right)}$$

$I(X;Y)$ always positive

$R(D) = \begin{cases} \left(\frac{1}{2}\log\frac{\sigma^2}{D}\right)^+, & \text{Gaussian (square dist)} \\ H(P) - H(D). & \text{Bernoulli (Hamming dist.)} \end{cases}$

$R(D)$ for Gaussian Source

To show that we can find a $p(\hat{x}, x)$ that achieves the bound, we construct a test channel that introduces distortion $D < \sigma^2$



$R(D) = \left(\frac{1}{2}\log\frac{\sigma^2}{D}\right)^+$

$$I(X;\hat{X}) = h(X) - h(X \mid \hat{X})$$

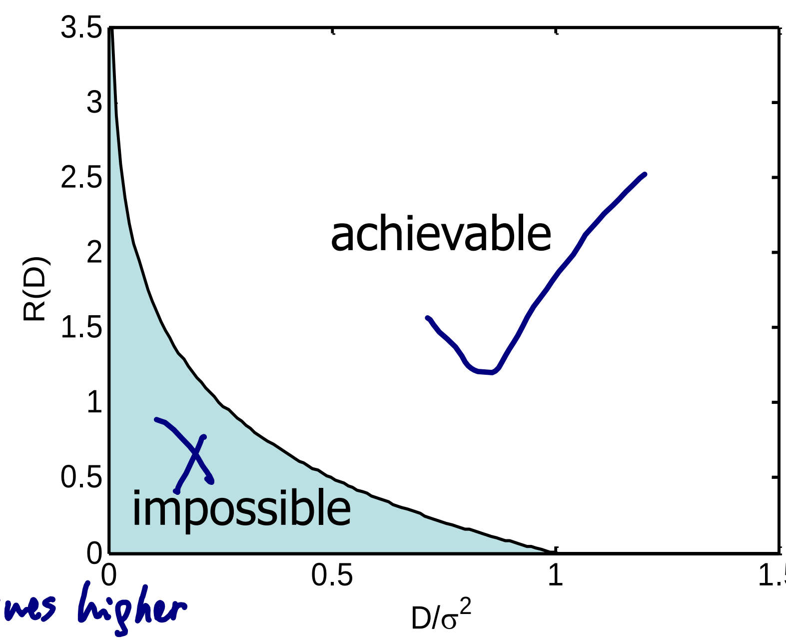
$D(R) = \frac{\sigma^2}{2^{2R}}$

$$= \tfrac{1}{2}\log 2\pi e\sigma^2 - h(X - \hat{X} \mid \hat{X})$$

$$= \tfrac{1}{2}\log 2\pi e\sigma^2 - h(Z \mid \hat{X})$$

$$= \tfrac{1}{2}\log\frac{\sigma^2}{D}$$

$$\Rightarrow R(D) = \max\{\tfrac{1}{2}\log\frac{\sigma^2}{D}, 0\}$$



PCM: $\frac{16}{3}$ times higher

$$\Rightarrow D(R) = \frac{\sigma^2}{2^{2R}} \qquad \text{cf. PCM} \quad D(R) = \frac{m_p^2/3}{2^{2R}} \overset{m_p = 4\sigma}{=} \frac{16/3 \cdot \sigma^2}{2^{2R}}$$

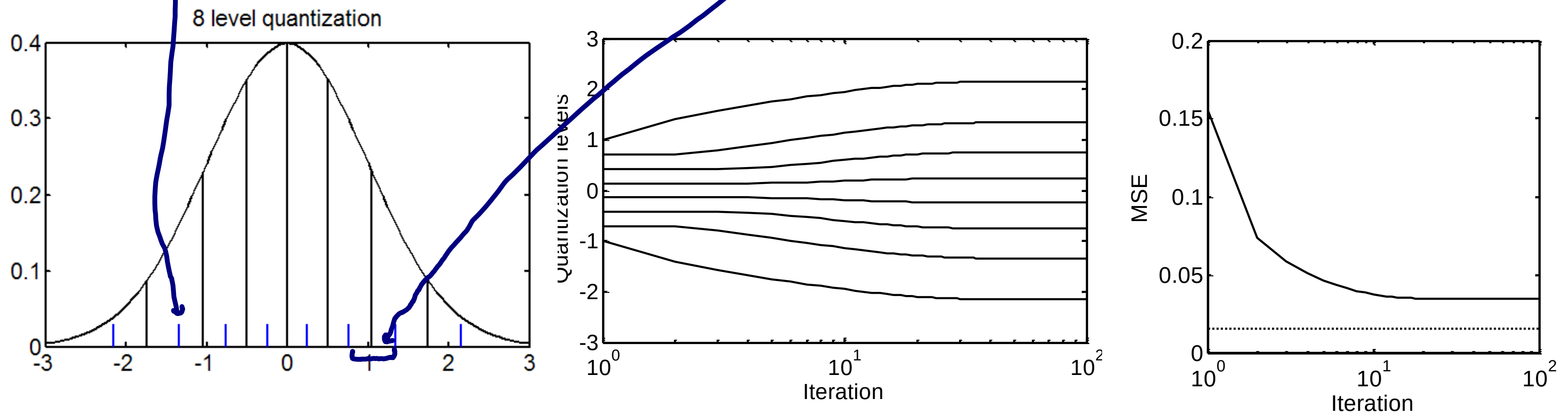
Lloyd Algorithm

Problem: Find optimum quantization levels for Gaussian pdf

a. Bin boundaries are midway between quantization levels

b. Each quantization level equals the mean value of its own bin

Lloyd algorithm: Pick random quantization levels then apply conditions (a) and (b) in turn until convergence.



Solid lines are bin boundaries. Initial levels uniform in $[-1,+1]$.

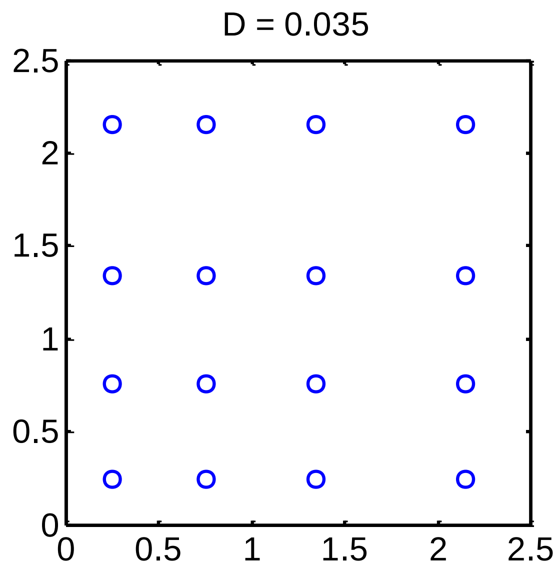
Best mean sq error for 8 levels = $0.0345\sigma^2$. Predicted $D(R) = (\sigma/8)^2 = 0.0156\sigma^2$

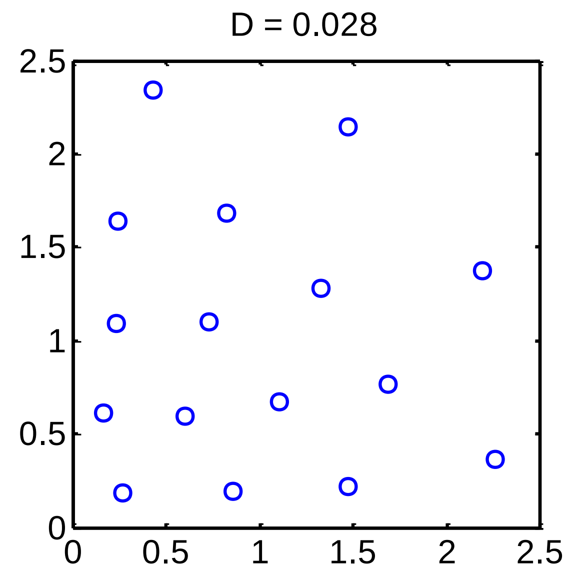
Vector Quantization

To get $D(R)$, you have to quantize many values together

- True even if the values are independent

Scalar quant.





Vector quant.

Two gaussian variables: one quadrant only shown

- Independent quantization puts dense levels in low prob areas
- Vector quantization is better (even more so if correlated)

Multiple Gaussian Variables

- Assume $x_{1:n}$ are independent gaussian sources with different variances. How should we apportion the available total distortion between the sources?
- Assume $x_i \sim N(0,\sigma_i^2)$ and $d(\mathbf{x},\hat{\mathbf{x}}) = n^{-1}\left(\mathbf{x}-\hat{\mathbf{x}}\right)^T\left(\mathbf{x}-\hat{\mathbf{x}}\right) \le D$

$$I(x_{1:n};\hat{x}_{1:n}) \ge \sum_{i=1}^{n} I(x_i;\hat{x}_i)$$

Mut Info Independence Bound for independent x_i

$$\ge \sum_{i=1}^{n} R(D_i) = \sum_{i=1}^{n} \max\left(½\log\frac{\sigma_i^2}{D_i},0\right)$$

$R(D)$ for individual Gaussian

We must find the D_i that minimize

$$\sum_{i=1}^{n} \max\left(½\log\frac{\sigma_i^2}{D_i},0\right)$$

$$\Rightarrow D_i = \begin{cases} D_0 & \text{if } D_0 < \sigma_i^2 \\ \sigma_i^2 & \text{otherwise} \end{cases}$$

such that $n^{-1}\sum_{i=1}^{n} D_i = D$

$I(X_{1:n};\hat{X}_{1:n}) \ge \sum_{i=1}^{n}\left(\frac{1}{2}\log\frac{\sigma_i^2}{D_i}\right)^+$

$\xRightarrow{\min I} D_i = \begin{cases} D_0, & \text{if } D_0 < \sigma_i^2 \\ \sigma_i^2, & \text{otherwise} \end{cases}$

do not describe r.v. whose variance $\le D_0$ by bits

only describe r.v. whose variance $> D_0$

Reverse Water-filling

Minimize $\sum_{i=1}^{n}\max\left(\tfrac{1}{2}\log\frac{\sigma_i^2}{D_i},0\right)$ subject to $\sum_{i=1}^{n} D_i \le nD$

$R_i = \tfrac{1}{2}\log\frac{\sigma_i^2}{D}$

Use a Lagrange multiplier: $\frac{\partial J}{\partial D_i} = \frac{1}{2}\frac{-\frac{\sigma_i^2}{D_i^2}}{\frac{\sigma_i^2}{D_i}} + \lambda = -\frac{1}{2}\frac{1}{D_i} + \lambda$

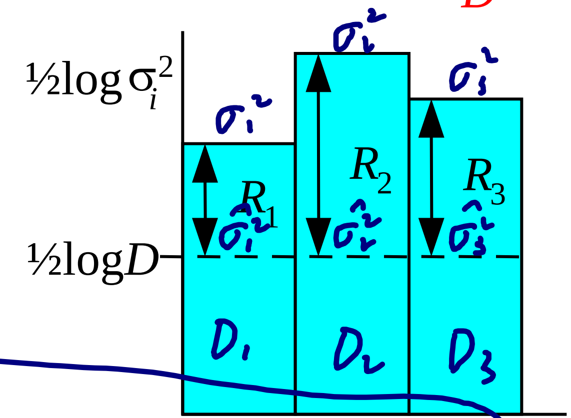
$$J = \sum_{i=1}^{n}\tfrac{1}{2}\log\frac{\sigma_i^2}{D_i} + \lambda\sum_{i=1}^{n} D_i$$

$$\frac{\partial J}{\partial D_i} = -\tfrac{1}{2}D_i^{-1} + \lambda = 0 \quad\Rightarrow\quad D_i = \tfrac{1}{2}\lambda^{-1} = D_0$$

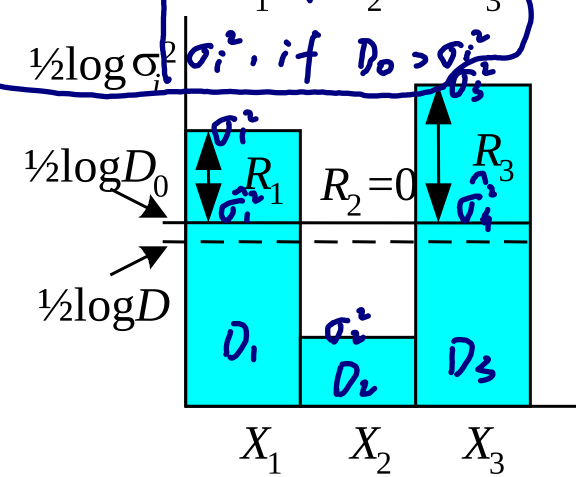
$D_i = \frac{1}{2\lambda} = D_0$

$$\sum_{i=1}^{n} D_i = nD_0 = nD \quad\Rightarrow\quad D_0 = D$$

Choose R_i for equal distortion



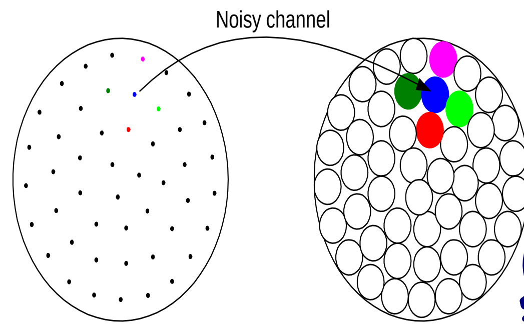
$\sigma_i^2 - \hat{\sigma}_i^2 = D_i = \begin{cases} D_0, & \text{if } D_0 < \sigma_i^2 \\ \sigma_i^2, & \text{if } D_0 > \sigma_i^2 \end{cases}$



- If $\sigma_i^2 < D$ then set $R_i = 0$ (meaning $D_i = \sigma_i^2$) and increase D_0 to maintain the average distortion equal to D

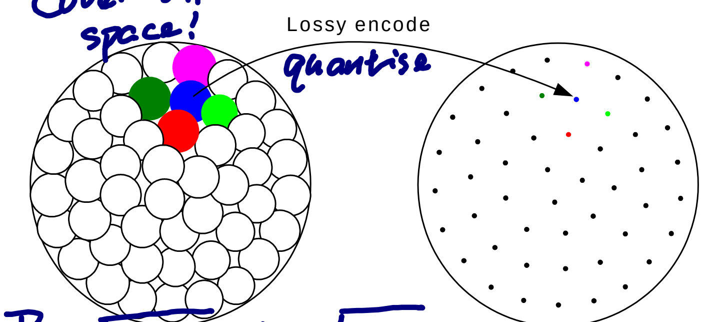
Channel/Source Coding Duality

- Channel Coding
  - Find codes separated enough to give non-overlapping output images.
  - Image size = channel noise
  - The maximum number (highest rate) is when the images just don't overlap (some gap).
- Source Coding
  - Find regions that cover the sphere
  - Region size = allowed distortion
  - The minimum number (lowest rate) is when they just fill the sphere (with no gap).



leave space.

Sphere Packing



$R=\sqrt{n\sigma^2}$ $\quad r=\sqrt{nD}$

Sphere Covering

$$\# = \frac{(n\sigma^2)^{\frac{n}{2}}}{(nD)^{\frac{n}{2}}} = \left(\frac{\sigma^2}{D}\right)^{\frac{n}{2}} = 2^{nR}$$

$$R = \frac{1}{n}\log\left(\frac{\sigma^2}{D}\right)^{\frac{n}{2}} = \frac{1}{2}\log\frac{\sigma^2}{D}$$

Gaussian Channel/Source

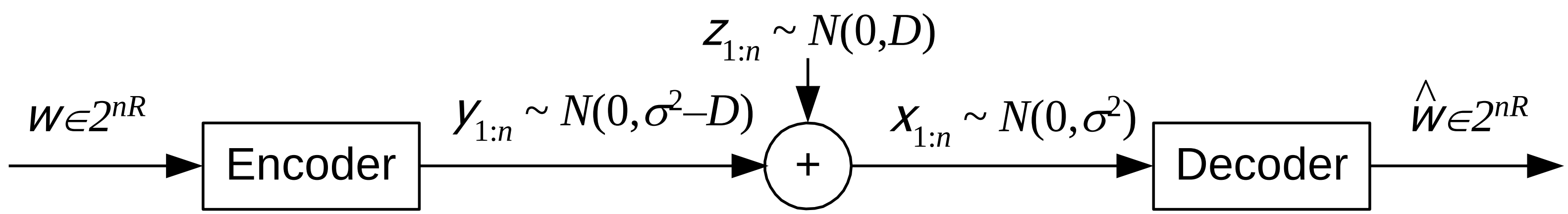
- Capacity of Gaussian channel (n: length)
  - Radius of big sphere $\sqrt{n(P+N)}$
  - Radius of small spheres $\sqrt{nN}$
  - Capacity $2^{nC} = \dfrac{\sqrt{n(P+N)}^{\,n}}{\sqrt{nN}^{\,n}} = \left(\dfrac{P+N}{N}\right)^{n/2}$

Maximum number of small spheres packed in the big sphere

- Rate distortion for Gaussian source
  - Variance $\sigma^2 \rightarrow$ radius of big sphere $\sqrt{n\sigma^2}$
  - Radius of small spheres $\sqrt{nD}$ for distortion D
  - Rate $2^{nR(D)} = \left(\dfrac{\sigma^2}{D}\right)^{n/2}$ = #

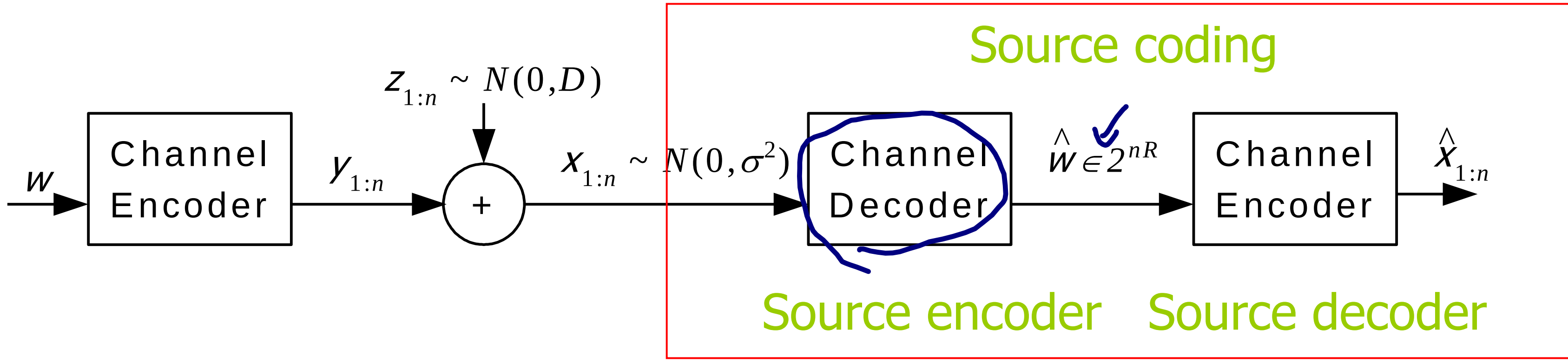
Minimum number of small spheres to cover the big sphere

Channel Decoder as Source Encoder



- For $R \cong C = \frac{1}{2}\log\left(1+\left(\sigma^2 - D\right)D^{-1}\right)$, we can find a channel encoder/decoder so that $p(\hat{w} \neq w) < \varepsilon$ and $E(x_i - y_i)^2 = D$
- Now reverse the roles of encoder and decoder. Since

$$p(\hat{x} \neq y) = p\left(w \neq \hat{w}\right) < \varepsilon \text{ and } E(x_i - \hat{x}_i)^2 \cong E(x_i - y_i)^2 = D$$



We have encoded x at rate $R = \frac{1}{2}\log(\sigma^2 D^{-1})$ with distortion D!

Summary

- Lossy source coding: tradeoff between rate and distortion
- Rate distortion function

$$R(D) = \min_{\mathbf{p}_{\hat{x}|x}\ s.t.\ Ed(x,\hat{x}) \le D} I(x;\hat{x})$$

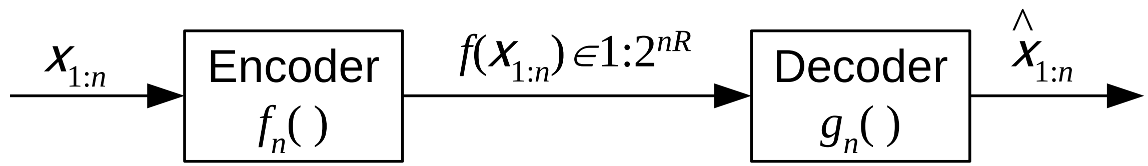
- Bernoulli source: $R(D) = (H(p) - H(D))^{+}$
- Gaussian source (reverse waterfilling): $$R(D) = \left(\frac{1}{2}\log\frac{\sigma^2}{D}\right)^{+}$$
- Duality: channel decoding (encoding) $\Leftrightarrow$ source encoding (decoding)

Nothing But Proof

- Proof of Rate Distortion Theorem
  - Converse: if the rate is less than R(D), then distortion of any code is higher than D
  - Achievability: if the rate is higher than R(D), then there exists a rate-R code which achieves distortion D

Quite technical!

Review



Rate Distortion function for x whose $p_{\mathbf{x}}(\mathbf{x})$ is known is

$$R(D) = \inf R \ \text{ such that } \ \exists f_n, g_n \ \text{ with } \ \lim_{n\to\infty} E_{\mathbf{x}\in X^n} d(\mathbf{x}, \hat{\mathbf{x}}) \le D$$

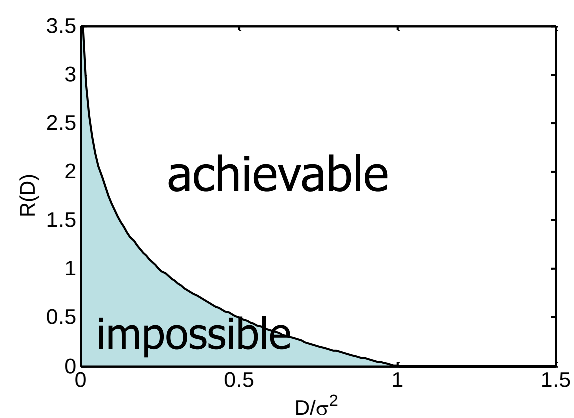
Rate Distortion Theorem:

$$R(D) = \min I(x;\hat{x}) \ \text{ over all } \ p(\hat{x}\,|\,x) \ \text{ such that } \ E_{x,\hat{x}} d(x,\hat{x}) \le D$$

We will prove this theorem for discrete X and bounded $d(x,y) \le d_{\max}$

$R(D)$ curve depends on your choice of $d(,)$

Decreasing and convex



Converse: Rate Distortion Bound

Suppose we have found an encoder and decoder at rate R_0 with expected distortion D for independent x_i (worst case)

We want to prove that $R_0 \geq R(D) = R\big(E\, d(\mathbf{x};\hat{\mathbf{x}})\big)$

- We show first that $R_0 \geq n^{-1}\sum_i I(x_i;\hat{x}_i)$
- We know that $I(x_i;\hat{x}_i) \geq R\big(E\ d(x_i;\hat{x}_i)\big)$ Def^n of $R(D)$
- and use convexity of $R(D)$ to show

$$n^{-1}\sum_i R\Big(E\ d(x_i;\hat{x}_i)\Big) \geq R\left(n^{-1}\sum_i E\ d(x_i;\hat{x}_i)\right) = R\Big(E\ d(\mathbf{x};\hat{\mathbf{x}})\Big) = R(D)$$

We prove convexity first and then the rest

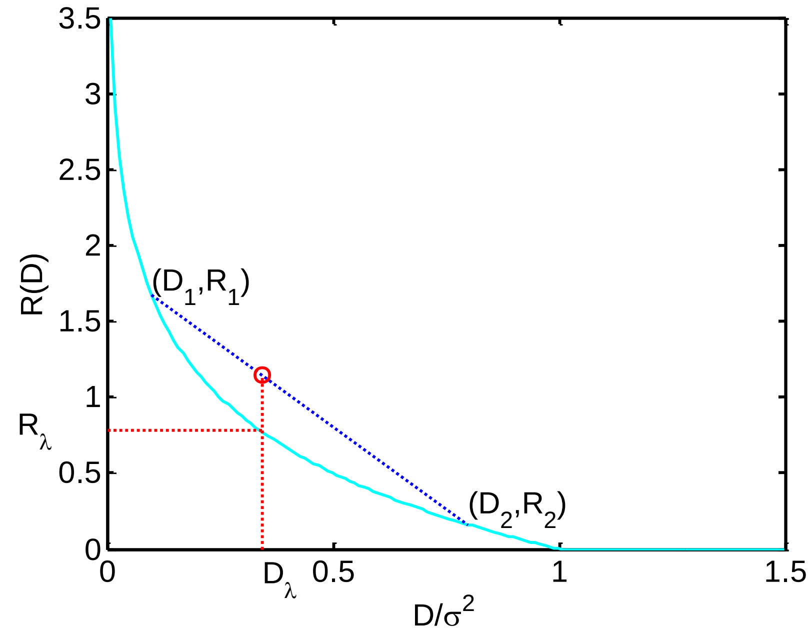
Convexity of $R(D)$

If $p_1(\hat{x} \mid x)$ and $p_2(\hat{x} \mid x)$ are associated with (D_1, R_1) and (D_2, R_2) on the $R(D)$ curve we define

$$p_\lambda(\hat{x} \mid x) = \lambda p_1(\hat{x} \mid x) + (1-\lambda) p_2(\hat{x} \mid x)$$

Then

$$E_{p_\lambda} d(x, \hat{x}) = \lambda D_1 + (1-\lambda) D_2 = D_\lambda$$



$$\begin{aligned} R(D_\lambda) &\le I_{p_\lambda}(X; \hat{X}) \\ &\le \lambda I_{p_1}(X; \hat{X}) + (1-\lambda) I_{p_2}(X; \hat{X}) \\ &= \lambda R(D_1) + (1-\lambda) R(D_2) \end{aligned}$$

$$R(D) = \min_{p(\hat{X}|X)} I(X; \hat{X})$$

$I(X; \hat{X})$ convex w.r.t. $p(\hat{x} \mid x)$

p_1 and p_2 lie on the $R(D)$ curve

Proof that $R \geq R(D)$

$$nR_0 \geq H(\hat{x}_{1:n}) \geq H(\hat{x}_{1:n}) - H(\hat{x}_{1:n} \mid x_{1:n})$$

Uniform bound; $H(\hat{x} \mid x) \geq 0$

$$= I(\hat{x}_{1:n}; x_{1:n})$$

Definition of $I(;)$

$$\geq \sum_{i=1}^{n} I(x_i; \hat{x}_i)$$

x_i indep: Mut Inf Independence Bound

$$\geq \sum_{i=1}^{n} R\left(E\, d(x_i; \hat{x}_i)\right) = n \sum_{i=1}^{n} n^{-1} R\left(E\, d(x_i; \hat{x}_i)\right)$$

definition of R

$$\geq nR\left(n^{-1} \sum_{i=1}^{n} E\, d(x_i; \hat{x}_i)\right) = nR\left(E\, d(x_{1:n}; \hat{x}_{1:n})\right)$$

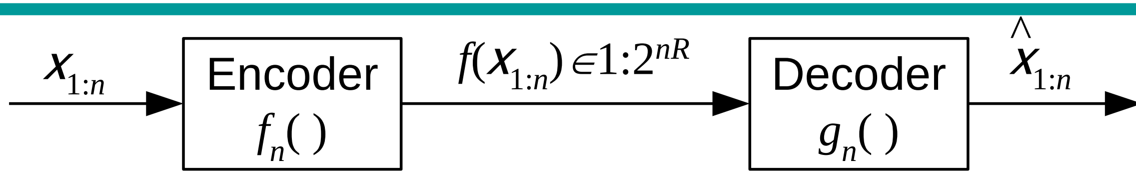
convexity

defn of vector $d(\,)$

$$\geq nR(D)$$

original assumption that $E(d) \leq D$ and $R(D)$ monotonically decreasing

Rate Distortion Achievability



We want to show that for any D, we can find an encoder and decoder that compresses $x_{1:n}$ to $nR(D)$ bits.

- $\mathbf{p}_X$ is given
- Assume we know the $p(\hat{x}\,|\,x)$ that gives $I(x;\hat{x}) = R(D)$
- Random codebook: Choose 2^{nR} random $\hat{x}_i \sim \mathbf{p}_{\hat{x}}$
  - There must be at least one code that is as good as the average
- Encoder: Use joint typicality to design
  - We show that there is almost always a suitable codeword

First define the typical set we will use, then prove two preliminary results.

Distortion Typical Set

Distortion Typical: $(x_i, \hat{x}_i) \in \mathcal{X} \times \hat{\mathcal{X}}$ drawn i.i.d. $\sim p(x, \hat{x})$

$$
\begin{aligned}
J_{d,\varepsilon}^{(n)} = \Big\{ \mathbf{x}, \hat{\mathbf{x}} \in \mathcal{X}^n \times \hat{\mathcal{X}}^n : & \left| -n^{-1} \log p(\mathbf{x}) - H(X) \right| < \varepsilon, \\
& \left| -n^{-1} \log p(\hat{\mathbf{x}}) - H(\hat{X}) \right| < \varepsilon, \\
& \left| -n^{-1} \log p(\mathbf{x}, \hat{\mathbf{x}}) - H(X, \hat{X}) \right| < \varepsilon \\
& \left| d(\mathbf{x}, \hat{\mathbf{x}}) - E\, d(X, \hat{X}) \right| < \varepsilon \Big\}
\end{aligned}
$$

new condition

Properties of Typical Set:

1. Indiv p.d.: $\mathbf{x}, \hat{\mathbf{x}} \in J_{d,\varepsilon}^{(n)} \Rightarrow \log p(\mathbf{x}, \hat{\mathbf{x}}) = -nH(X, \hat{X}) \pm n\varepsilon$
2. Total Prob: $p\left(\mathbf{x}, \hat{\mathbf{x}} \in J_{d,\varepsilon}^{(n)}\right) > 1 - \varepsilon$ for $n > N_\varepsilon$

weak law of large numbers; $d(X_i, \hat{X}_i)$ are i.i.d.

Conditional Probability Bound

Lemma: $\mathbf{x},\hat{\mathbf{x}} \in J_{d,\varepsilon}^{(n)} \Rightarrow p(\hat{\mathbf{x}}) \geq p(\hat{\mathbf{x}} \mid \mathbf{x})2^{-n(I(X;\hat{X})+3\varepsilon)}$

Proof:

$$p(\hat{\mathbf{x}} \mid \mathbf{x}) = \frac{p(\hat{\mathbf{x}},\mathbf{x})}{p(\mathbf{x})}$$

$$= p(\hat{\mathbf{x}})\frac{p(\hat{\mathbf{x}},\mathbf{x})}{p(\hat{\mathbf{x}})p(\mathbf{x})}$$

take max of top and min of bottom

$$\leq p(\hat{\mathbf{x}})\frac{2^{-n(H(X,\hat{X})-\varepsilon)}}{2^{-n(H(X)+\varepsilon)}2^{-n(H(\hat{X})+\varepsilon)}}$$

bounds from defn of J

$$= p(\hat{\mathbf{x}})2^{n(I(X;\hat{X})+3\varepsilon)}$$

defn of I

Curious but Necessary Inequality

Lemma: $u, v \in [0,1], m > 0 \quad \Rightarrow \quad (1-uv)^m \leq 1-u+e^{-vm}$

Proof: u=0: $e^{-vm} \geq 0 \quad \Rightarrow \quad (1-0)^m \leq 1-0+e^{-vm}$

u=1: Define $f(v) = e^{-v} - 1 + v \quad \Rightarrow \quad f'(v) = 1 - e^{-v}$

$f(0) = 0$ and $f'(v) > 0$ for $v > 0 \quad \Rightarrow \quad f(v) \geq 0$ for $v \in [0,1]$

Hence for $v \in [0,1]$, $\quad 0 \leq 1 - v \leq e^{-v} \quad \Rightarrow (1-v)^m \leq e^{-vm}$

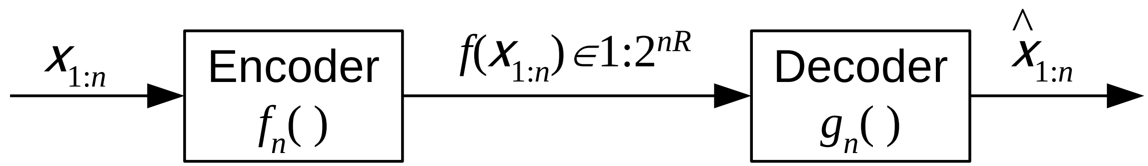
0<u<1: Define $g_v(u) = (1-uv)^m$

$\Rightarrow g_v''(x) = m(m-1)v^2(1-uv)^{n-2} \geq 0 \Rightarrow g_v(u)$ convex for $u,v \in [0,1]$

$(1-uv)^m = g_v(u) \leq (1-u)g_v(0) + ug_v(1)$ convexity for $u,v \in [0,1]$

$= (1-u)1 + u(1-v)^m \leq 1-u+ue^{-vm} \leq 1-u+e^{-vm}$

Achievability of $R(D)$: preliminaries



- Choose D and find a $p(\hat{x}\,|\,x)$ such that $I(x;\hat{x}) = R(D);\ E\,d(x,\hat{x}) \le D$

  Choose $\delta > 0$ and define $\mathbf{p}_{\hat{x}} = \{\ p(\hat{x}) = \sum_x p(x)\,p(\hat{x}\,|\,x)\ \}$

- **Decoder:** For each $w \in 1:2^{nR}$ choose $g_n(w) = \hat{\mathbf{x}}_w$ drawn i.i.d. $\sim \mathbf{p}_{\hat{x}}^{\,n}$
- **Encoder:** $f_n(\mathbf{x}) = \min w$ such that $(\mathbf{x},\hat{\mathbf{x}}_w) \in J_{d,\varepsilon}^{(n)}$ else 1 if no such w
- **Expected Distortion:** $\overline{D} = E_{\mathbf{x},g}\, d(\mathbf{x},\hat{\mathbf{x}})$
  - over all input vectors $\mathbf{x}$ and all random decoding functions, g
  - for large n we show $\overline{D} = D + \delta$ so there must be one good code

Expected Distortion

We can divide the input vectors $\mathbf{x}$ into two categories:

a) if $\exists w$ such that $(\mathbf{x}, \hat{\mathbf{x}}_w) \in J_{d,\varepsilon}^{(n)}$ then $d(\mathbf{x}, \hat{\mathbf{x}}_w) < D + \varepsilon$

since $E\ d(\mathbf{x}, \hat{\mathbf{x}}) \le D$

b) if no such w exists we must have $d(\mathbf{x}, \hat{\mathbf{x}}_w) < d_{\max}$ since we are assuming that $d(,)$ is bounded. Suppose the probability of this situation is P_e.

Hence $\overline{D} = E_{\mathbf{x},g}\ d(\mathbf{x}, \hat{\mathbf{x}})$

$$\le (1 - P_e)(D + \varepsilon) + P_e d_{\max}$$

$$\le D + \varepsilon + P_e d_{\max}$$

We need to show that the expected value of P_e is small

Error Probability

Define the set of valid inputs for (random) code g

$$V(g) = \left\{ \mathbf{x} : \exists w \text{ with } (\mathbf{x}, g(w)) \in J_{d,\varepsilon}^{(n)} \right\}$$

We have $P_e = \sum_g p(g) \sum_{\mathbf{x} \notin V(g)} p(\mathbf{x}) = \sum_{\mathbf{x}} p(\mathbf{x}) \sum_{g : \mathbf{x} \notin V(g)} p(g)$ Change the order

Define $K(\mathbf{x}, \hat{\mathbf{x}}) = 1$ if $(\mathbf{x}, \hat{\mathbf{x}}) \in J_{d,\varepsilon}^{(n)}$ else 0

Prob that a random $\hat{\mathbf{x}}$ does not match $\mathbf{x}$ is $1 - \sum_{\hat{\mathbf{x}}} p(\hat{\mathbf{x}}) K(\mathbf{x}, \hat{\mathbf{x}})$

Prob that an entire code does not match $\mathbf{x}$ is $\left(1 - \sum_{\hat{\mathbf{x}}} p(\hat{\mathbf{x}}) K(\mathbf{x}, \hat{\mathbf{x}})\right)^{2^{nR}}$

Codewords are i.i.d.

Hence $P_e = \sum_{\mathbf{x}} p(\mathbf{x}) \left(1 - \sum_{\hat{\mathbf{x}}} p(\hat{\mathbf{x}}) K(\mathbf{x}, \hat{\mathbf{x}})\right)^{2^{nR}}$

Achievability for Average Code

Since $\mathbf{x},\hat{\mathbf{x}} \in J_{d,\varepsilon}^{(n)} \Rightarrow p(\hat{\mathbf{x}}) \geq p(\hat{\mathbf{x}} \mid \mathbf{x}) 2^{-n\left(I(X;\hat{X})+3\varepsilon\right)}$

$$P_e = \sum_{\mathbf{x}} p(\mathbf{x}) \left(1 - \sum_{\hat{\mathbf{x}}} p(\hat{\mathbf{x}}) K(\mathbf{x},\hat{\mathbf{x}}) \right)^{2^{nR}}$$

$$\leq \sum_{\mathbf{x}} p(\mathbf{x}) \left(1 - \sum_{\hat{\mathbf{x}}} p(\hat{\mathbf{x}} \mid \mathbf{x}) K(\mathbf{x},\hat{\mathbf{x}}) \cdot 2^{-n\left(I(X;\hat{X})+3\varepsilon\right)} \right)^{2^{nR}}$$

Using $(1-uv)^m \leq 1-u+e^{-vm}$

with $u = \sum_{\hat{\mathbf{x}}} p(\hat{\mathbf{x}} \mid \mathbf{x}) K(\mathbf{x},\hat{\mathbf{x}});\quad v = 2^{-nI(X;\hat{X})-3n\varepsilon};\quad m = 2^{nR}$

$$\leq \sum_{\mathbf{x}} p(\mathbf{x}) \left(1 - \sum_{\hat{\mathbf{x}}} p(\hat{\mathbf{x}} \mid \mathbf{x}) K(\mathbf{x},\hat{\mathbf{x}}) + \exp\left(-2^{-n\left(I(X;\hat{X})+3\varepsilon\right)} 2^{nR}\right) \right)$$

Note: $0 \leq u,v \leq 1$ as required

Achievability for Average Code

$$P_e \le \sum_{\mathbf{x}} p(\mathbf{x}) \left(1 - \sum_{\hat{\mathbf{x}}} p(\hat{\mathbf{x}} \mid \mathbf{x}) K(\mathbf{x}, \hat{\mathbf{x}}) + \exp\left(- 2^{-n\left(I(X;\hat{X})+3\varepsilon\right)} 2^{nR} \right) \right)$$

$$= 1 - \sum_{\mathbf{x},\hat{\mathbf{x}}} p(\mathbf{x}, \hat{\mathbf{x}}) K(\mathbf{x}, \hat{\mathbf{x}}) + \exp\left(- 2^{n\left(R - I(X;\hat{X}) - 3\varepsilon\right)} \right)$$

Mutual information does not involve particular $\mathbf{x}$

$$= P\left\{ (\mathbf{x}, \hat{\mathbf{x}}) \notin J_{d,\varepsilon}^{(n)} \right\} + \exp\left(- 2^{n\left(R - I(X;\hat{X}) - 3\varepsilon\right)} \right)$$

$$\xrightarrow[n \to \infty]{} 0$$

since both terms $\to 0$ as $n \to \infty$ provided $nR > I(X, \hat{X}) + 3\varepsilon$

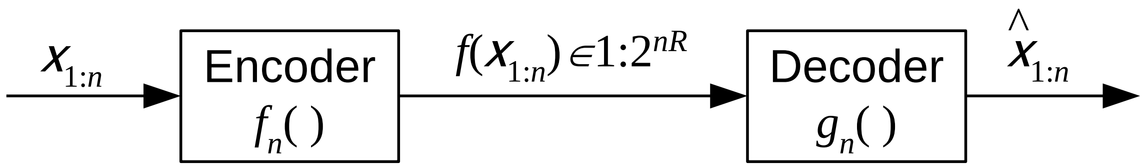
Hence $\forall \delta > 0$, $\overline{D} = E_{\mathbf{x},g}\, d(\mathbf{x}, \hat{\mathbf{x}})$ can be made $\le D + \delta$

Achievability

Since $\forall\delta>0,\ \overline{D}=E_{\mathbf{x},g}\ d(\mathbf{x},\hat{\mathbf{x}})$ can be made $\le D+\delta$

there must be at least one g with $E_{\mathbf{x}}\ d(\mathbf{x},\hat{\mathbf{x}})\le D+\delta$

Hence (R,D) is achievable for any $R>R(D)$



that is $\lim_{n\to\infty} E_{X_{1:n}}(\mathbf{x},\hat{\mathbf{x}})\le D$

In fact a stronger result is true (proof in C&T 10.6):

$\forall\delta>0, D$ and $R>R(D), \exists f_n, g_n$ with $p(d(\mathbf{x},\hat{\mathbf{x}})\le D+\delta)\xrightarrow[n\to\infty]{}1$

Lecture 17

- Introduction to network information theory
- Multiple access
- Distributed source coding

Network Information Theory

- System with many senders and receivers
- New elements: interference, cooperation, competition, relay, feedback…
- Problem: decide whether or not the sources can be transmitted over the channel
  - Distributed source coding
  - Distributed communication
  - The general problem has not yet been solved, so we consider various special cases
- Results are presented without proof (can be done using mutual information, joint AEP)

Implications to Network Design

- Examples of large information networks
  - Computer networks
  - Satellite networks
  - Telephone networks
- A complete theory of network communications would have wide implications for the design of communication and computer networks
- Examples
  - CDMA (code-division multiple access): mobile phone network
  - Network coding: significant capacity gain compared to routing-based networks

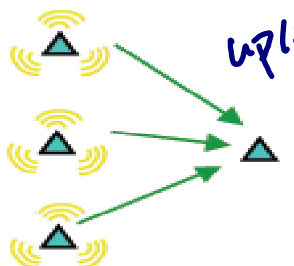
Network Models Considered

- Multi-access channel
- Broadcast channel
- Distributed source coding
- Relay channel
- Interference channel
- Two-way channel
- General communication network

State of the Art

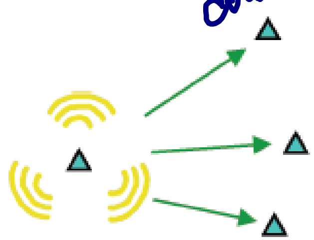
- Triumphs
  - Multi-access channel

uplink: many to one

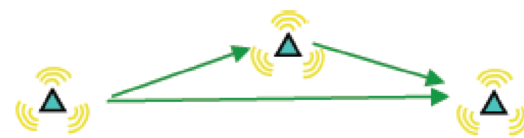


  - Gaussian broadcast channel

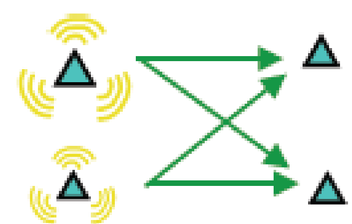
downlink: one to many



- Unknowns
  - The simplest relay channel



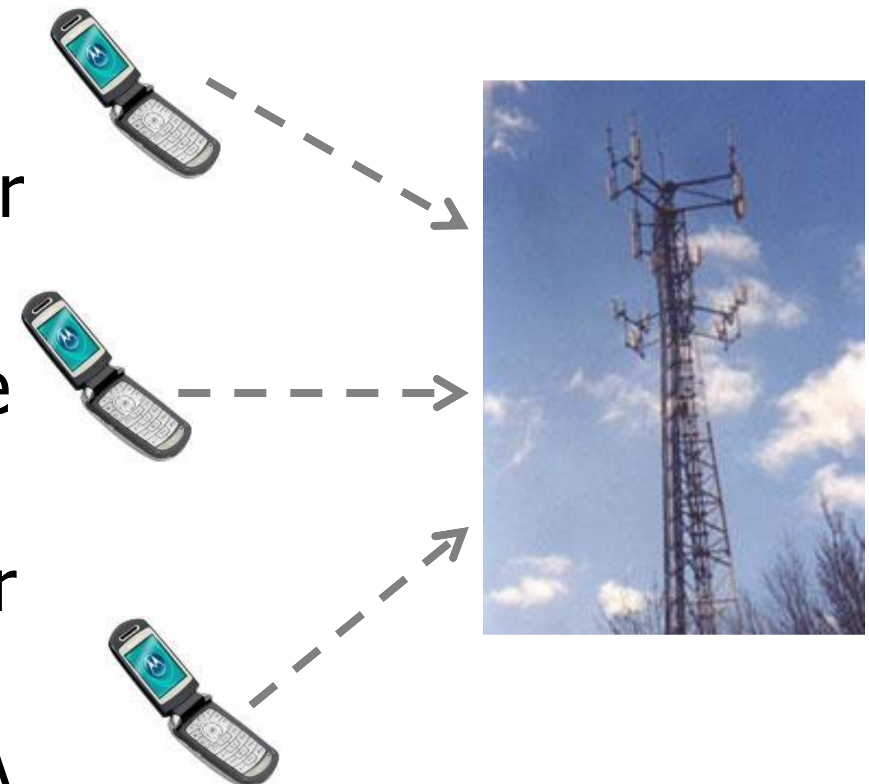
  - The simplest interference channel



Reminder: Networks being built (ad hoc networks, sensor networks) are much more complicated

Multi-Access Channel

- Example: many users communicate with a common base station over a common channel
- What rates are achievable simultaneously?
- Best understood multiuser channel
- Very successful: 3G CDMA mobile phone networks



Capacity Region

- Capacity of single-user Gaussian channel

$$C = \frac{1}{2}\log\left(1 + \frac{P}{N}\right) = C\left(\frac{P}{N}\right)$$

- Gaussian multi-access channel with m users

$$Y = \sum_{i=1}^{m} X_i + Z$$

X_i has equal power P
noise Z has variance N

- Capacity region

$$R_i < C\left(\frac{P}{N}\right)$$

$$R_i + R_j < C\left(\frac{2P}{N}\right)$$

$$R_i + R_j + R_k < C\left(\frac{3P}{N}\right)$$

$$\vdots$$

$$\sum_{i=1}^{m} R_i < C\left(\frac{mP}{N}\right)$$

R_i: rate for user i

Transmission: independent and simultaneous (i.i.d. Gaussian codebooks)

Decoding: joint decoding, look for m codewords whose sum is closest to Y

The last inequality dominates when all rates are the same

The sum rate goes to ∞ with m

Two-User Channel

SIC:

1. decode user 2 (treating 1 as noise) $Y = X_2 + X_1 + N$ ($X_1 + N$ treat as noise)

$R_2 < C\left(\frac{P_2}{P_1+N}\right)$

2. subtract out user 2, then decode user 1 $Y' = Y - X_1 = X_1 + N$

$R_1 < C\left(\frac{P_1}{N}\right)$

$C\left(\frac{P_2}{P_1+N}\right) + C\left(\frac{P_1}{N}\right) = C\left(\frac{P_1+P_2}{N}\right)$

$\frac{1}{2}\log\left(1+\frac{P_2}{P_1+N}\right) + \frac{1}{2}\log\left(1+\frac{P_1}{N}\right) = \frac{1}{2}\log\left(1+\frac{P_1+P_2}{N}\right)$

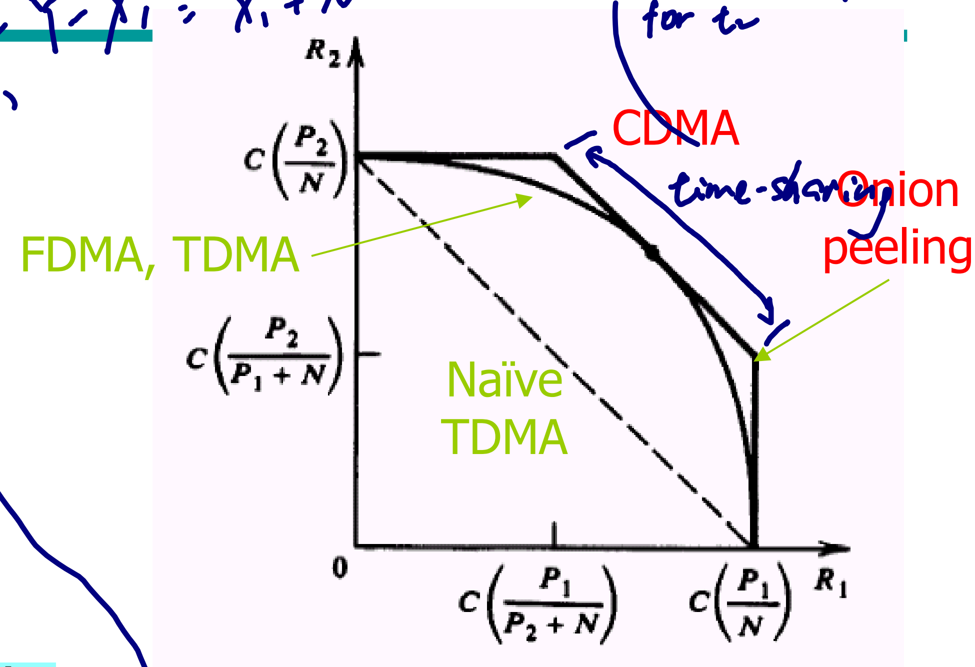
$\frac{1}{2}\log\left(\frac{P_1+P_2+N}{P_1+N}\cdot\frac{P_1+N}{N}\right) = \frac{P_1+P_2+N}{N}$

- Capacity region

$$R_1 < C\left(\frac{P_1}{N}\right)$$

$$R_2 < C\left(\frac{P_2}{N}\right)$$

$$R_1 + R_1 < C\left(\frac{P_1+P_2}{N}\right)$$



  - Corresponds to CDMA
  - Surprising fact: sum rate = rate achieved by a single sender with power P_1+P_2
- Achieves a higher sum rate than treating interference as noise, i.e.,

$$C\left(\frac{P_1}{P_2+N}\right) + C\left(\frac{P_2}{P_1+N}\right)$$

Onion Peeling

- Interpretation of corner point: onion-peeling
  - First stage: decoder user 2, considering user 1 as noise
  - Second stage: subtract out user 2, decoder user 1
- In fact, it can achieve the entire capacity region
  - Any rate-pairs between two corner points achievable by time-sharing
- Its technical term is successive interference cancelation (SIC)
  - Removes the need for joint decoding
  - Uses a sequence of single-user decoders
- SIC is implemented in the uplink of CDMA 2000 EV-DO (evolution-data optimized)
  - Increases throughput by about 65%

1G : FDMA
2G : TDMA
3G : CDMA
4G : OFDMA

Comparison with TDMA and FDMA

- FDMA (frequency-division multiple access)

$$R_1 = W_1 \log\left(1 + \frac{P_1}{N_0 W_1}\right)$$

$$R_2 = W_2 \log\left(1 + \frac{P_2}{N_0 W_2}\right)$$

Total bandwidth $W = W_1 + W_2$

Varying W_1 and W_2 tracing out the curve in the figure

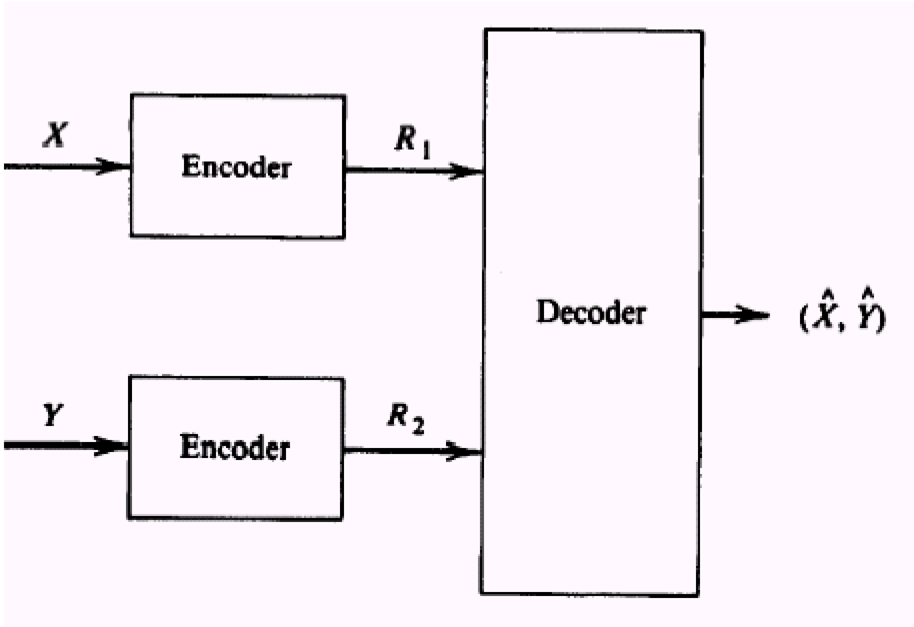
- TDMA (time-division multiple access)
  - Each user is allotted a time slot, transmits and other users remain silent
  - Naïve TDMA: dashed line
  - Can do better while still maintaining the same average power constraint; the same capacity region as FDMA
- CDMA capacity region is larger
  - But needs a more complex decoder

Distributed Source Coding

- Associate with nodes are sources that are generally dependent
- How do we take advantage of the dependence to reduce the amount of information transmitted?
- Consider the special case where channels are noiseless and without interference
- Finding the set of rates associate with each source such that all required sources can be decoded at destination
- Data compression dual to multi-access channel

Two-User Distributed Source Coding

- X and Y are correlated
- But the encoders cannot communicate; have to encode independently
- A single source: $R > H(X)$
- Two sources: $R > H(X,Y)$ if encoding together



- What if encoding separately?
  - Of course one can do $R > H(X) + H(Y)$
  - Surprisingly, $R = H(X,Y)$ is sufficient (Slepian-Wolf coding, 1973)
  - Sadly, the coding scheme was not practical (again)

Slepian-Wolf Coding

lossless

/ : Slepian-Wolf joint coding
\ : individual coding

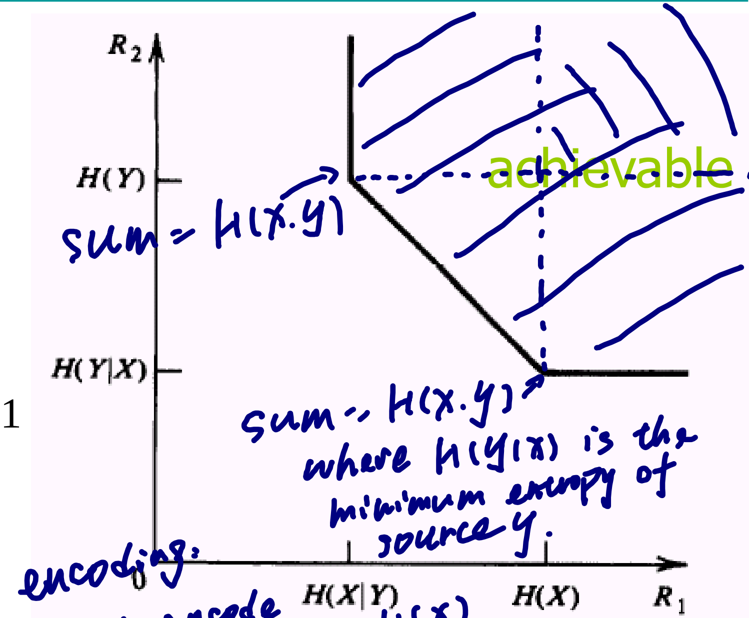
- Achievable rate region

$$R_1 \geq H(X \mid Y)$$

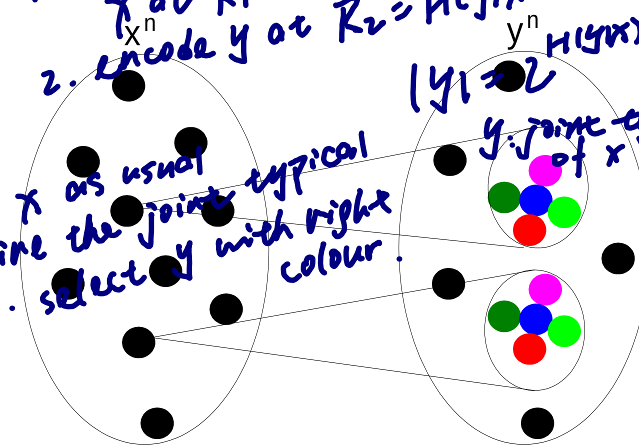
$$R_2 \geq H(Y \mid X)$$

$$R_1 + R_2 \geq H(X, Y)$$

- Core idea: joint typicality
- Interpretation of corner point R_1 $H(X)$, $R_2 = H(Y|X)$
  - X can encode as usual
  - Associate with each x^n is a jointly typical fan (however Y doesn't know)
  - Y sends the color (thus compression)
  - Decoder uses the color to determine the point in jointly typical fan associated with x^n
- Straight line: achieved by time-sharing



encoding:
1. encode X at $R_1 = H(X)$
2. encode Y at $R_2 = H(Y|X)$

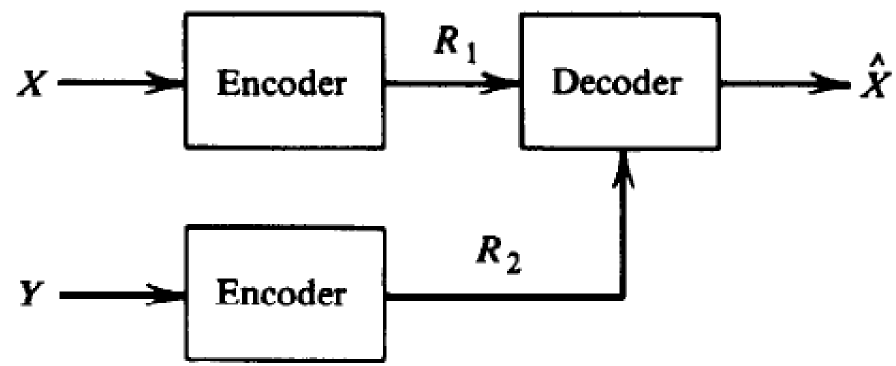


decoding:
1. decode X as usual
2. examine the joint typical fan. select Y with right colour.

Wyner-Ziv Coding

lossy

- Distributed source coding with side information
- Y is encoded at rate R_2
- Only X to be recovered
- How many bits R_1 are required?



- If $R_2 = H(Y)$, then $R_1 = H(X|Y)$ by Slepian-Wolf coding
- In general

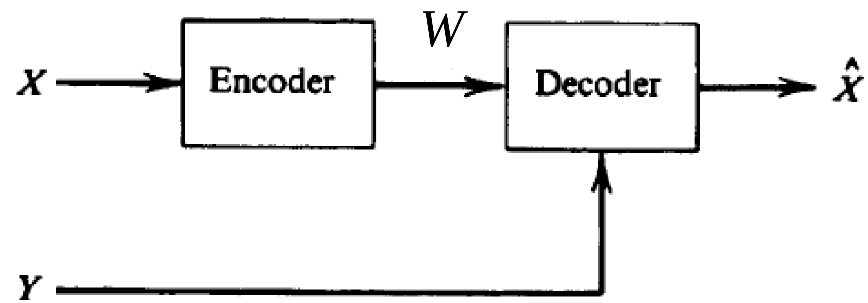
$$\boxed{\begin{aligned} R_1 &\geq H(X|U) \\ R_2 &\geq I(Y;U) \end{aligned}}$$

$R_1 + R_2 = H(X|U) + I(Y;U)$
$= H(X|U) + H(Y) + H(U) - H(Y,U)$
$= H(X,U) + H(Y) - H(Y,U)$

where U is an auxiliary random variable (can be thought of as approximate version of Y)

Rate-Distortion

- Given Y, what is the rate-distortion to describe X?

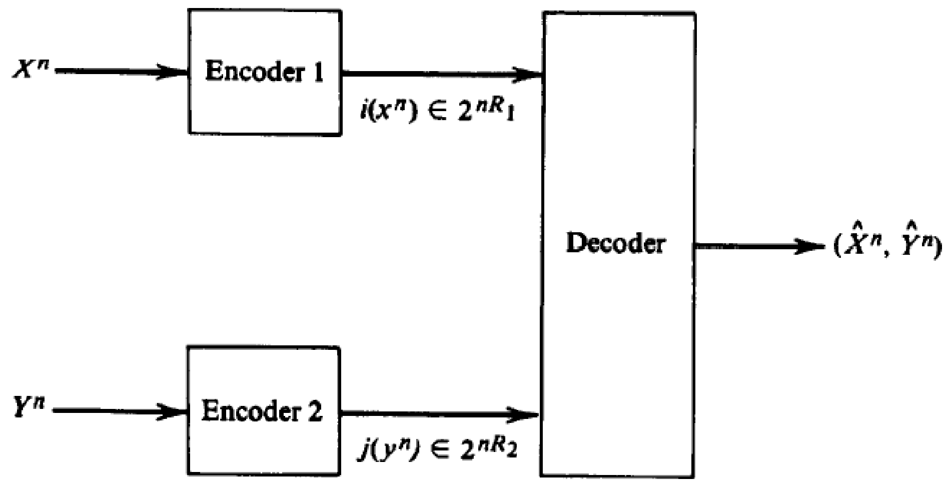


$$R_Y(D) = \min_{p(w|x)} \min_{f} \{I(X;W) - I(Y;W)\}$$

over all decoding functions $f : Y \times W \to \hat{X}$

and all $p(w \mid x)$ such that $E_{x,w,y} d(x, \hat{x}) \leq D$

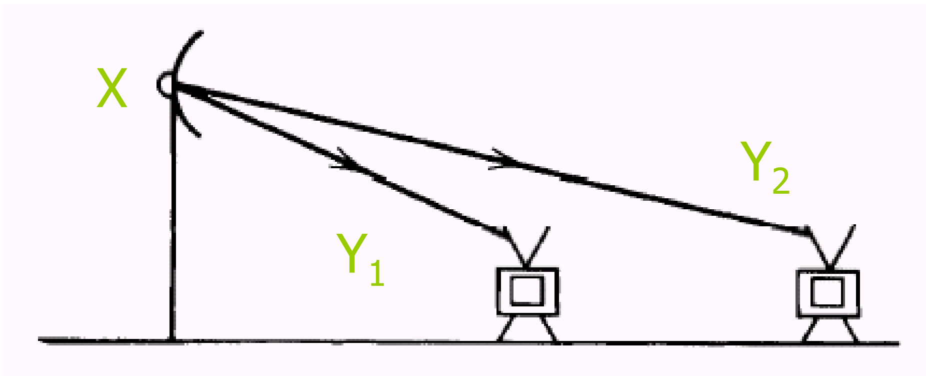
- The general problem of rate-distortion for correlated sources remains unsolved



Lecture 18

- Network information theory – II
  - Broadcast
  - Relay
  - Interference channel
  - Two-way channel
  - Comments on general communication networks

Broadcast Channel



- One-to-many: HDTV station sending different information simultaneously to many TV receivers over a common channel; lecturer in classroom
- What are the achievable rates for all different receivers?
- How does the sender encode information meant for different signals in a common signal?
- Only partial answers are known.

Two-User Broadcast Channel

- Consider a memoryless broadcast channel with one encoder and two decoders
- Independent messages at rate R_1 and R_2
- Degraded broadcast channel: $p(y_1, y_2|x) = p(y_1|x)\, p(y_2| y_1)$
  - Meaning $X \to Y_1 \to Y_2$ (Markov chain)
  - Y_2 is a degraded version of Y_1 (receiver 1 is better)
- Capacity region of degraded broadcast channel

$$R_2 \le I(U;Y_2)$$
$$R_1 \le I(X;Y_1|U)$$

U: $C\left(\frac{P_2}{P_1+N}\right)$?

U is an auxiliary random variable

decoded by the bad receiver

Scalar Gaussian Broadcast Channel

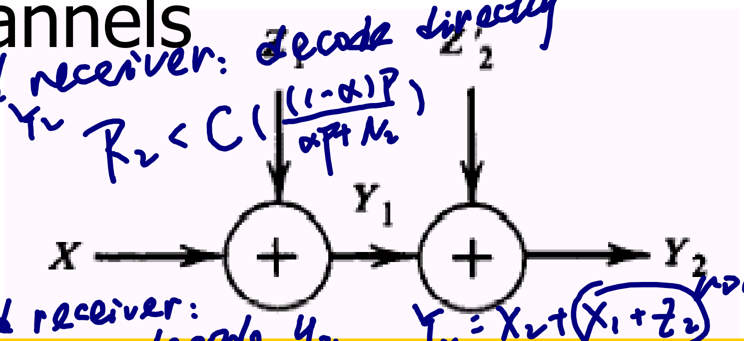
- All scalar Gaussian broadcast channels belong to the class of degraded channels

$$Y_1 = X + Z_1$$
$$Y_2 = X + Z_2$$

Assume variance
$N_1 < N_2$

bad receiver: decode directly Y_2

$R_2 < C\left(\frac{(1-\alpha)P}{\alpha P + N_2}\right)$



good receiver:
1. decode Y_2
2. subtract. decode Y_1.

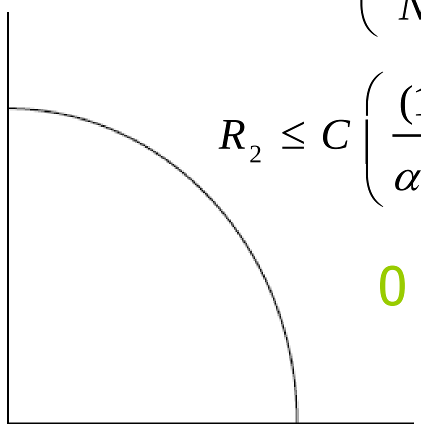
$Y_2 = X_2 + (X_1 + Z_2)$ noise
$Y_1 = X_1 + X_2 + Z_1$
$Y_1' = Y_1 - X_2 = X_1 + Z_1$

- Capacity region

$$R_1 \le C\left(\frac{\alpha P}{N_1}\right)$$

$$R_2 \le C\left(\frac{(1-\alpha)P}{\alpha P + N_2}\right)$$

$0 \le \alpha \le 1$



Coding Strategy

Encoding: one codebook with power αP at rate R_1, another with power $(1-\alpha)P$ at rate R_2, send the sum of two codewords

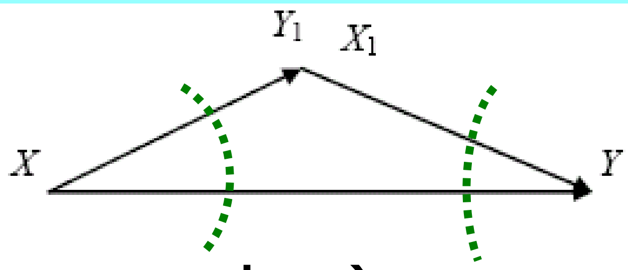
αP ~

$R_1 < C\left(\frac{\alpha P}{N_1}\right)$

Decoding: Bad receiver Y_2 treats Y_1 as noise; good receiver Y_1 first decode Y_2, subtract it out, then decode his own message

Relay Channel

- One source, one destination. one or more intermediate relays
- Example: one relay



  - A broadcast channel (X to Y and Y_1)
  - A multi-access channel (X and X_1 to Y)
  - Capacity is unknown! Upper bound:

max flow depends on the bottleneck of system pipes.

$$C \le \sup_{p(x,x_1)} \min\{ \underbrace{I(X, X_1; Y)}_{\text{MAC}}, \underbrace{I(X; Y, Y_1 \mid X_1)}_{\text{BC}} \}$$

  - Max-flow min-cut interpretation
    - First term: maximum rate from X and X_1 to Y
    - Second term: maximum rate from X to Y and Y_1

Degraded Relay Channel

- In general, the max-flow min-cut bound cannot be achieved
- Reason
  - Interference
  - What for the relay to forward?
  - How to forward?
- Capacity is known for degraded relay channel (i.e, Y is a degradation of Y_1, or relay is better than receiver), i.e., the upper bound is achieved

$$C = \sup_{p(x,x_1)} \min\{I(X, X_1; Y), I(X; Y, Y_1 \mid X_1)\}$$

Gaussian Relay Channel

- Channel model

$$Y_1 = X + Z_1 \qquad \text{Variance}(Z_1) = N_1$$

$$Y = X + Z_1 + X_1 + Z_2 \qquad \text{Variance}(Z_2) = N_2$$

- Encoding at relay: $X_{1i} = f_i(Y_{11}, Y_{12}, \cdots, Y_{1i-1})$
- Capacity

correlation

$$C = \max_{0 \le \alpha \le 1} \min \left\{ C\left(\frac{P + P_1 + 2\sqrt{(1-\alpha)PP_1}}{N_1 + N_2} \right), C\left(\frac{\alpha P}{N_1} \right) \right\}$$

X has power P

X1 has power P1

  - If

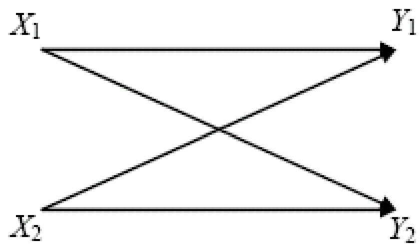
relay – desitinati on SNR $\frac{P_1}{N_2} \ge \frac{P}{N_1}$ source – relay SNR

then $C = C(P/N_1)$ (capacity from source to relay can be achieved; exercise)

  - Rate $C = C(P/(N_1 + N_2))$ without relay is increased by the relay to $C = C(P/N_1)$

Interference Channel

- Two senders, two receivers, with crosstalk



  - Y_1 listens to X_1 and doesn't care what X_2 speaks or what Y_2 hears
  - Similarly with X_2 and Y_2
- Neither a broadcast channel nor a multiaccess channel
- This channel has not been solved
  - Capacity is known to within one bit (Etkin, Tse, Wang 2008)
  - A promising technique — interference alignment (Camdambe, Jafar 2008)

Symmetric Interference Channel

strong interf. 1. decode interf
2. cancel interf

- Model

$$Y_1 = X_1 + aX_2 + Z_1 \qquad \text{equal power } P$$

$$Y_2 = X_2 + aX_1 + Z_2 \qquad \mathrm{Var}(Z_1) = \mathrm{Var}(Z_2) = N$$

- Capacity has been derived in the strong interference case ($a \geq 1$) (Han, Kobayashi, 1981)
  - Very strong interference ($a^2 \geq 1 + P/N$) is equivalent to no interference whatsoever
- Symmetric capacity (for each user $R_1 = R_2$)

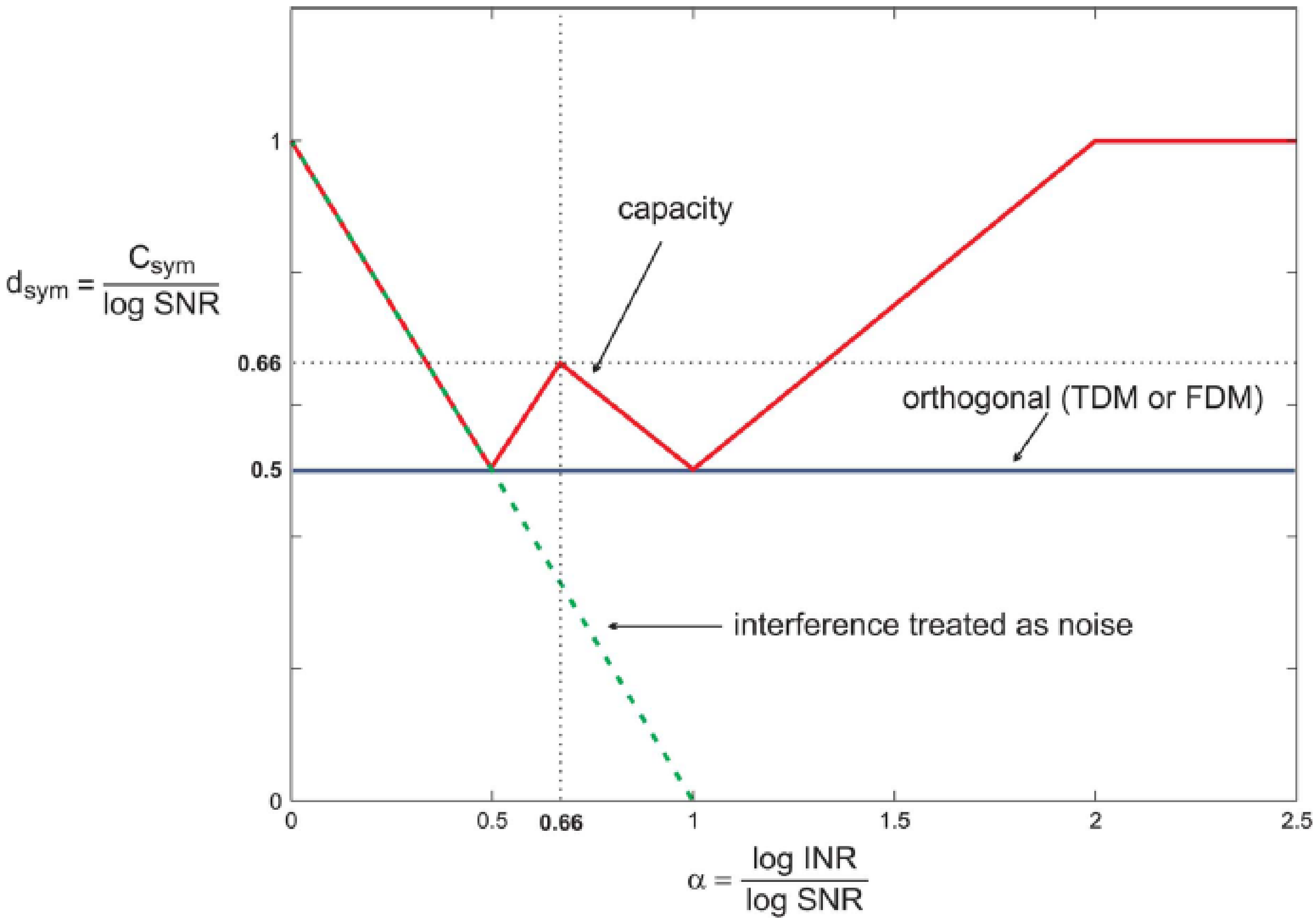
$$d_{\mathrm{sym}} = \begin{cases} 1-\alpha, & 0 \leq \alpha < \frac{1}{2} \\ \alpha, & \frac{1}{2} \leq \alpha < \frac{2}{3} \\ 1-\frac{\alpha}{2}, & \frac{2}{3} < \alpha \leq 1 \\ \frac{\alpha}{2}, & 1 \leq \alpha < 2 \\ 1, & \alpha \geq 2. \end{cases}$$

$$d_{\mathrm{sym}}(\alpha) := \lim_{\mathsf{SNR},\mathsf{INR}\to\infty;\ \frac{\log \mathsf{INR}}{\log \mathsf{SNR}}=\alpha} \frac{C_{\mathrm{sym}}(\mathsf{INR},\mathsf{SNR})}{C_{\mathrm{awgn}}(\mathsf{SNR})},$$

SNR = P/N

INR = a^2P/N

Capacity



$d_{sym} = \frac{C_{sym}}{\log SNR}$
1
0.66
0.5
0
0
0.5
0.66
1
1.5
2
2.5
capacity
orthogonal (TDM or FDM)
interference treated as noise
$\alpha = \frac{\log INR}{\log SNR}$

receiver 1 (to decode user 1)

1. decode user 2 (treat user 1 as noise)

$C(\frac{a^2P}{P+N}) > C(\frac{P}{N})$

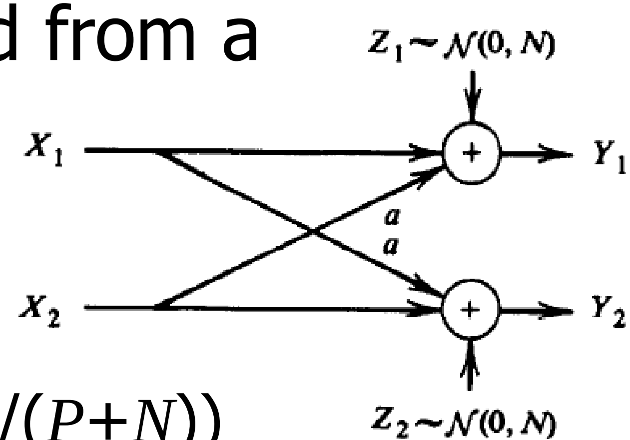
2. subtract user 2, then decode user 1

$y_1 = x_1 + \cancel{a x_2} + z_1$

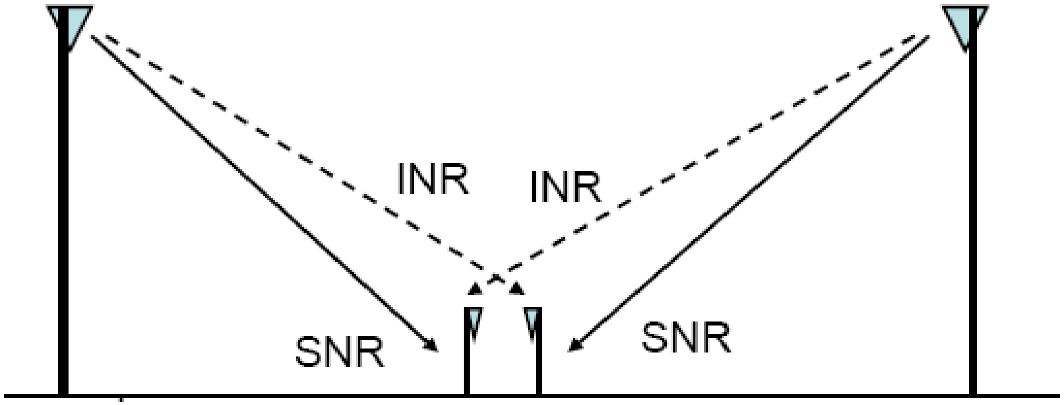
$R_1 < C(\frac{P}{N})$

Very strong interference = no interference

- Each sender has power P and rate $C(P/N)$
- Independently sends a codeword from a Gaussian codebook
- Consider receiver 1
  - Treats sender 1 as interference
  - Can decode sender 2 at rate $C(a^2P/(P+N))$
  - If $C(a^2P/(P+N)) > C(P/N)$, i.e.,
    rate 2 → 1 > rate 2 → 2 (crosslink is better)
    he can perfectly decode sender 2
  - Subtracting it from received signal, he sees a clean channel with capacity $C(P/N)$



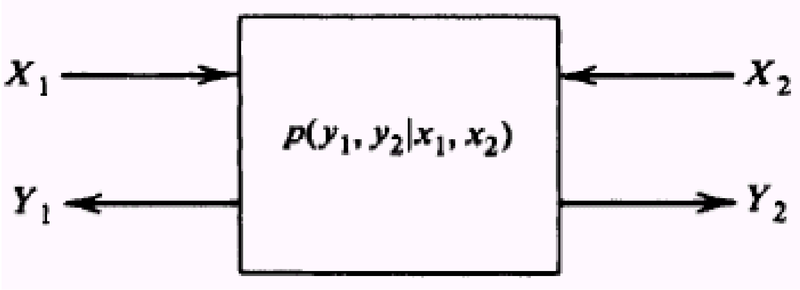
An Example



- Two cell-edge users (bottleneck of the cellular network)
- No exchange of data between the base stations or between the mobiles
- Traditional approaches
  - Orthogonalizing the two links (reuse ½)
  - Universal frequency reuse and treating interference as noise
- Higher capacity can be achieved by advanced interference management

Two-Way Channel

- Similar to interference channel, but in both directions (Shannon 1961)



- Feedback
  - Sender 1 can use previously received symbols from sender 2, and vice versa
  - They can cooperate with each other
- Gaussian channel:
  - Capacity region is known (not the case in general)
  - Decompose into two independent channels

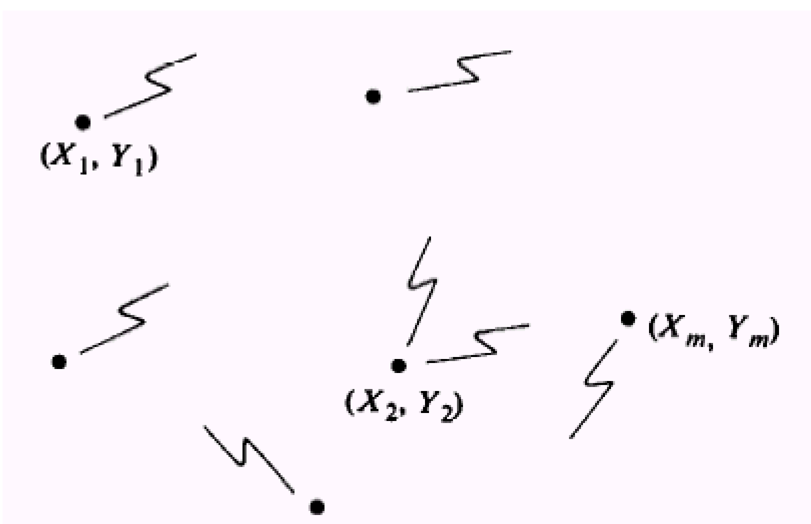
$$R_1 < C\left(\frac{P_1}{N_1}\right)$$

$$R_2 < C\left(\frac{P_2}{N_2}\right)$$

Coding strategy: Sender 1 sends a codeword; so does sender 2. Receiver 2 receives a sum but he can subtract out his own thus having an interference-free channel from sender 1.

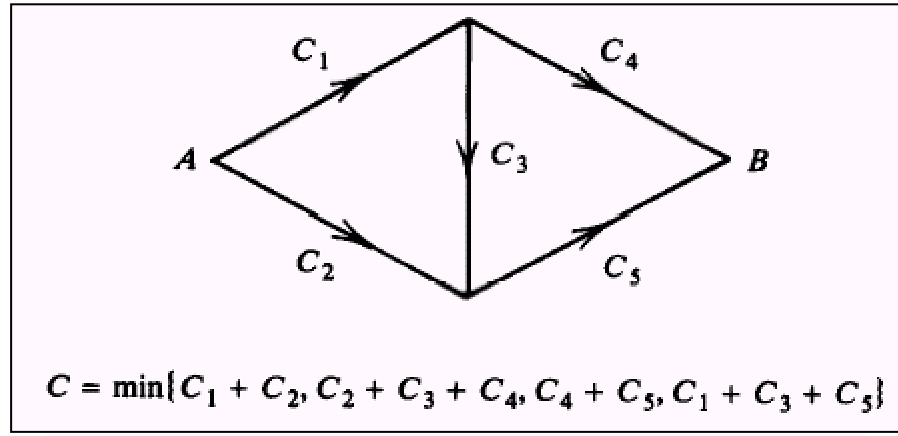
General Communication Network

- Many nodes trying to communicate with each other
- Allows computation at each node using it own message and all past received symbols
- All the models we have considered are special cases
- A comprehensive theory of network information flow is yet to be found



Capacity Bound for a Network

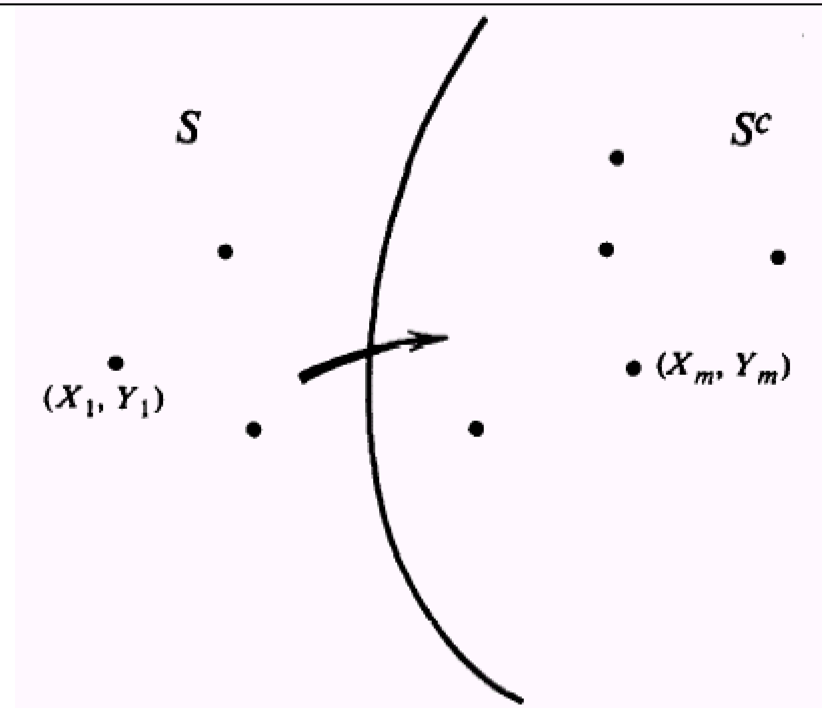
- Max-flow min-cut
  - Minimizing the maximum flow across cut sets yields an upper bound on the capacity of a network



- Outer bound on capacity region

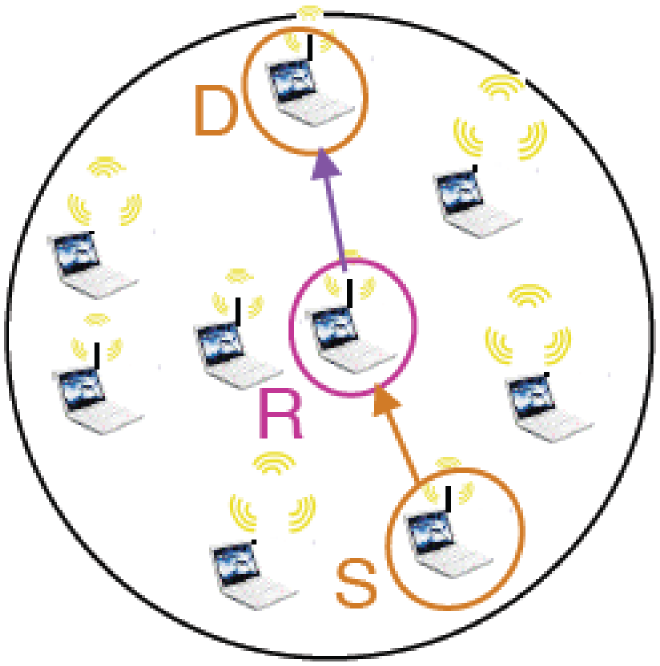
$$\sum_{i\in S,\, j\in S^c} R^{(i,j)} \le I(X^{(S)}; Y^{(S^c)} \mid X^{(S^c)})$$

  - Not achievable in general



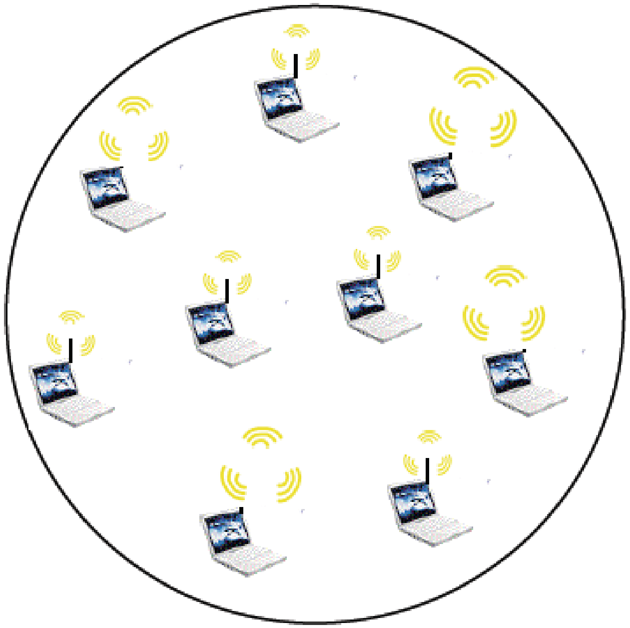
Questions to Answer

- Why multi-hop relay? Why decode and forward? Why treat interference as noise?
- Source-channel separation? Feedback?
- What is really the best way to operate wireless networks?
- What are the ultimate limits to information transfer over wireless networks?



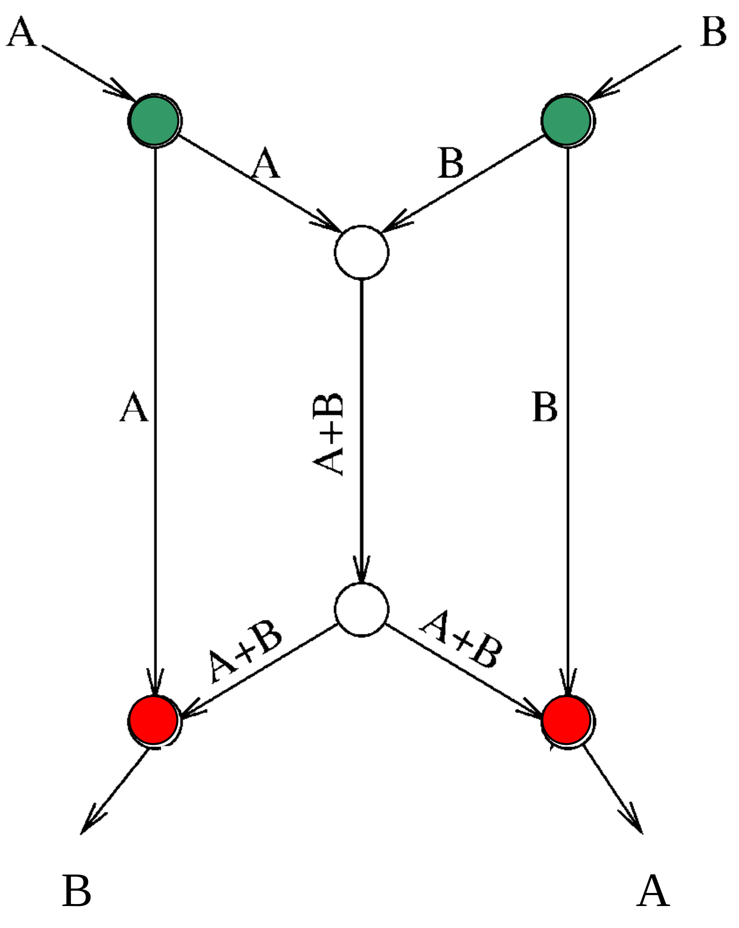
Scaling Law for Wireless Networks

- High signal attenuation: (transport) capacity is O(n) bit-meter/sec for a planar network with n nodes (Xie-Kumar'04)
- Low attenuation: capacity can grow superlinearly
- Requires cooperation between nodes
- Multi-hop relay is suboptimal but order optimal



Network Coding

- Routing: store and forward (as in Internet)
- Network coding: recompute and redistribute
- Given the network topology, coding can increase capacity (Ahlswede, Cai, Li, Yeung, 2000)
  - Doubled capacity for butterfly network
- Active area of research



Butterfly Network

Lecture 19

- Revision Lecture

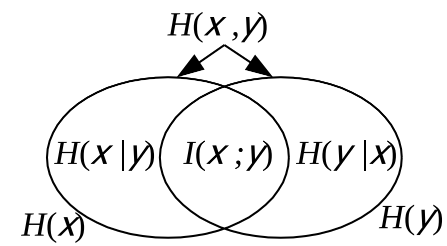
Summary (1)

- Entropy: $$H(x) = \sum_{x \in \mathcal{X}} p(x) \times -\log_2 p(x) = E - \log_2 (p_X(x))$$
  - Bounds: $$0 \leq H(x) \leq \log |\mathcal{X}|$$
  - Conditioning reduces entropy: $$H(y \mid x) \leq H(y)$$
  - Chain Rule: $$H(x_{1:n}) = \sum_{i=1}^{n} H(x_i \mid x_{1:i-1}) \leq \sum_{i=1}^{n} H(x_i)$$
    $$H(x_{1:n} \mid y_{1:n}) \leq \sum_{i=1}^{n} H(x_i \mid y_i)$$
- Relative Entropy:

$$D(\mathbf{p} \parallel \mathbf{q}) = E_{\mathbf{p}} \log \left(p(x) / q(x) \right) \geq 0$$

Summary (2)

- Mutual Information:



$$I(y;x) = H(y) - H(y|x)$$
$$= H(x) + H(y) - H(x,y) = D\left(\mathbf{p}_{x,y} \parallel \mathbf{p}_x \mathbf{p}_y\right)$$

  - Positive and Symmetrical: $I(x;y) = I(y;x) \geq 0$
  - x, y indep $\Leftrightarrow$ $H(x,y) = H(y) + H(x) \Leftrightarrow I(x;y) = 0$
  - Chain Rule: $I(x_{1:n};y) = \sum_{i=1}^{n} I(x_i;y \mid x_{1:i-1})$

$$x_i \text{ independent} \quad \Rightarrow \quad I(x_{1:n};y_{1:n}) \geq \sum_{i=1}^{n} I(x_i;y_i)$$

$$p(y_i \mid x_{1:n};y_{1:i-1}) = p(y_i \mid x_i) \quad \Rightarrow \quad I(x_{1:n};y_{1:n}) \leq \sum_{i=1}^{n} I(x_i;y_i)$$

n-use DMC capacity

Summary (3)

- Convexity: $f''(x) \geq 0 \Rightarrow f(x)$ convex $\Rightarrow Ef(x) \geq f(Ex)$
  - $H(\mathbf{p})$ concave in $\mathbf{p}$
  - $I(x\,;\,y)$ concave in $\mathbf{p}_x$ for fixed $\mathbf{p}_{y|x}$
  - $I(x\,;\,y)$ convex in $\mathbf{p}_{y|x}$ for fixed $\mathbf{p}_x$
- Markov: $x \to y \to z \Leftrightarrow p(z \mid x, y) = p(z \mid y) \Leftrightarrow I(x; z \mid y) = 0$

  $\Rightarrow I(x\,;\,y) \geq I(x\,;\,z)$ and $I(x\,;\,y) \geq I(x; y \mid z)$
- Fano: $x \to y \to \hat{x} \Rightarrow p(\hat{x} \neq x) \geq \dfrac{H(x \mid y) - 1}{\log(|\mathcal{X}| - 1)}$
- Entropy Rate:
  - Stationary process

    $$H(\mathcal{X}) = \lim_{n\to\infty} n^{-1} H(x_{1:n})$$

    $$H(\mathcal{X}) = \lim_{n\to\infty} H(x_n \mid x_{1:n-1})$$
  - Markov Process:

    $$H(\mathcal{X}) = \lim_{n\to\infty} H(x_n \mid x_{n-1})$$ if stationary

Summary (4)

- Kraft: Uniquely Decodable $\Rightarrow \sum_{i=1}^{|X|} D^{-l_i} \le 1 \Rightarrow \exists$ instant code
- Average Length: Uniquely Decodable $\Rightarrow \quad L_C = E\, l(x) \ge H_D(x)$
- Shannon-Fano: Top-down 50% splits. $L_{SF} \le H_D(x)+1$
- Huffman: Bottom-up design. Optimal. $L_H \le H_D(x) + 1$
  - Designing with wrong probabilities, $\mathbf{q} \Rightarrow$ penalty of $D(\mathbf{p}||\mathbf{q})$
  - Long blocks disperse the 1-bit overhead
- Lempel-Ziv Coding:
  - Does not depend on source distribution
  - Efficient algorithm widely used
  - Approaches entropy rate for stationary ergodic sources

Summary (5)

- Typical Set
  - Individual Prob $\mathbf{x} \in T_\varepsilon^{(n)} \Rightarrow \log\ p(\mathbf{x}) = -nH(X) \pm n\varepsilon$
  - Total Prob $p(\mathbf{x} \in T_\varepsilon^{(n)}) > 1 - \varepsilon \text{ for } n > N_\varepsilon$
  - Size $(1-\varepsilon)2^{n(H(X)-\varepsilon)} \overset{n>N_\varepsilon}{<} \left|T_\varepsilon^{(n)}\right| \le 2^{n(H(X)+\varepsilon)}$
  - No other high probability set can be much smaller
- Asymptotic Equipartition Principle
  - Almost all event sequences are equally surprising

Summary (6)

- DMC Channel Capacity: $C = \max_{\mathbf{p}_x} I(x;y)$
- Coding Theorem
  - Can achieve capacity: random codewords, joint typical decoding
  - Cannot beat capacity: Fano inequality
- Feedback doesn't increase capacity of DMC but could simplify coding/decoding
- Joint Source-Channel Coding doesn't increase capacity of DMC

Summary (7)

- Polar codes are low-complexity codes directly built from information theory.
- Their constructions are aided by the polarization phenomenon.
- For channel coding, polar codes achieve channel capacity.
- For source coding, polar codes achieve the entropy bound.
- And much more.

Summary (8)

- Differential Entropy: $h(x) = E - \log f_x(x)$
  - Not necessarily positive
  - $h(x+a) = h(x)$, $h(ax) = h(x) + \log|a|$, $h(x|y) \le h(x)$
  - $I(x; y) = h(x) + h(y) - h(x, y) \ge 0$, $D(f||g)=E \log(f/g) \ge 0$
- Bounds:
  - Finite range: Uniform distribution has max: $h(x) = \log(b-a)$
  - Fixed Covariance: Gaussian has max: $h(x) = ½\log((2\pi e)^n|K|)$
- Gaussian Channel
  - Discrete Time: $C=½\log(1+PN^{-1})$
  - Bandlimited: $C=W \log(1+PN_0^{-1}W^{-1})$
    - For constant C: $E_b N_0^{-1} = PC^{-1}N_0^{-1} = (W/C)\left(2^{(W/C)^{-1}} - 1\right) \underset{W \to \infty}{\to} \ln 2 = -1.6$ dB
  - Feedback: Adds at most ½ bit for coloured noise

Summary (9)

- Parallel Gaussian Channels: Total power constraint $\sum P_i = nP$
  - White noise: Waterfilling: $P_i = \max(v - N_i, 0)$
  - Correlated noise: Waterfill on noise eigenvectors
- Rate Distortion: $R(D) = \min_{\mathbf{p}_{\hat{x}|x}\, s.t.\, Ed(x,\hat{x}) \le D} I(x;\hat{x})$
  - Bernoulli Source with Hamming d: $R(D) = \max(H(\mathbf{p}_x) - H(D), 0)$
  - Gaussian Source with mean square d: $R(D) = \max(\frac{1}{2}\log(\sigma^2 D^{-1}), 0)$
  - Can encode at rate R: random decoder, joint typical encoder
  - Can't encode below rate R: independence bound

Summary (10)

- Gaussian multiple access channel

$$R_1 < C\left(\frac{P_1}{N}\right), \qquad R_2 < C\left(\frac{P_2}{N}\right)$$

$$R_1 + R_1 < C\left(\frac{P_1 + P_2}{N}\right), \qquad C(x) = \frac{1}{2}\log(1+x)$$

- Distributed source coding
  - Slepian-Wolf coding

$$R_1 \geq H(X \mid Y), \quad R_2 \geq H(Y \mid X)$$

$$R_1 + R_2 \geq H(X, Y)$$

- Scalar Gaussian broadcast channel

$$R_1 \leq C\left(\frac{\alpha P}{N_1}\right), \qquad R_2 \leq C\left(\frac{(1-\alpha)P}{\alpha P + N_2}\right), \qquad 0 \leq \alpha \leq 1$$

- Gaussian Relay channel

$$C = \max_{0 \leq \alpha \leq 1} \min \left\{ C\left(\frac{P + P_1 + 2\sqrt{(1-\alpha)PP_1}}{N_1 + N_2}\right), C\left(\frac{\alpha P}{N_1}\right) \right\}$$

Summary (11)

- Interference channel
  - Strong interference = no interference
- Gaussian two-way channel
  - Decompose into two independent channels
- General communication network
  - Max-flow min-cut theorem
  - Not achievable in general
  - But achievable for multiple access channel and Gaussian relay channel