Information Theory

Mike Brookes

E4.40, ISE4.51, SO20

Lectures

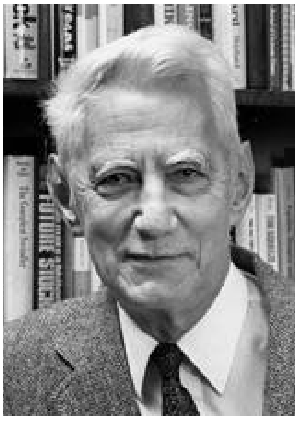
Claude Shannon

- "The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point." (Claude Shannon 1948)

- Channel Coding Theorem:

  It is possible to achieve near perfect communication of information over a noisy channel

- In this course we will:
  - 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.



1916 - 2001

Textbooks

Book of the course:

- *Elements of Information Theory* by T M Cover & J A Thomas, Wiley 2006, 978-0471241959 £30 (Amazon)

Alternative book – a denser but entertaining read that covers most of the course + much else:

- *Information Theory, Inference, and Learning Algorithms*, D MacKay, CUP, 0521642981 £28 or free at http://www.inference.phy.cam.ac.uk/mackay/itila/

Assessment: Exam only – no coursework.

Acknowledgement: Many of the examples and proofs in these notes are taken from the course textbook "Elements of Information Theory" by T M Cover & J A Thomas and/or the lecture notes by Dr L Zheng based on the book.

Notation

- Vectors and Matrices
  - $\mathbf{v}$=vector, $\mathbf{V}$=matrix, ⊙=elementwise product
- Scalar Random Variables
  - x = R.V, x = specific value, $\mathcal{X}$ = alphabet
- Random Column Vector of length N
  - $\mathbf{x}$= R.V, $\mathbf{x}$ = specific value, $\mathcal{X}^N$ = alphabet
  - x_i and x_i are particular vector elements
- Ranges
  - a:b denotes the range a, a+1, …, b

Discrete Random Variables

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

Examples:

$\mathcal{X}$ = [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"

$\mathcal{X}$ = [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 real valued and defined on $\mathcal{X}$ then

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

often write E for E_X

Examples:

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

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

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

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

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

This is the "entropy" of X

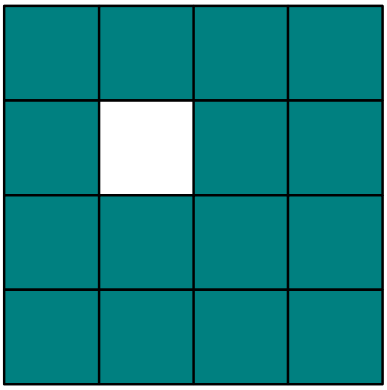
Shannon Information Content

- The Shannon Information Content of an outcome with probability p is $-\log_2 p$
- Example 1: Coin tossing
  - $\mathcal{X}$ = [Heads; Tails], $\mathbf{p}$ = [½; ½], SIC = [1; 1] bits
- Example 2: Is it my birthday ?
  - $\mathcal{X}$ = [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



Guess		Prob	SIC
1.	No	$^{15}/_{16}$	0.093 bits

Entropy

The entropy, $H(x) = E - \log_2(p_x(x)) = -\mathbf{p}_x^T \log_2 \mathbf{p}_x$

- $H(x)$ = 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 $[H(x), H(x)+1)$

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

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

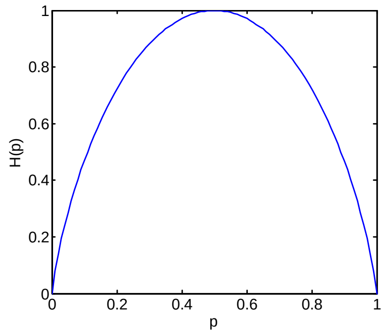
Entropy Examples

(1) Bernoulli Random Variable

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

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

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



(2) Four Coloured Shapes

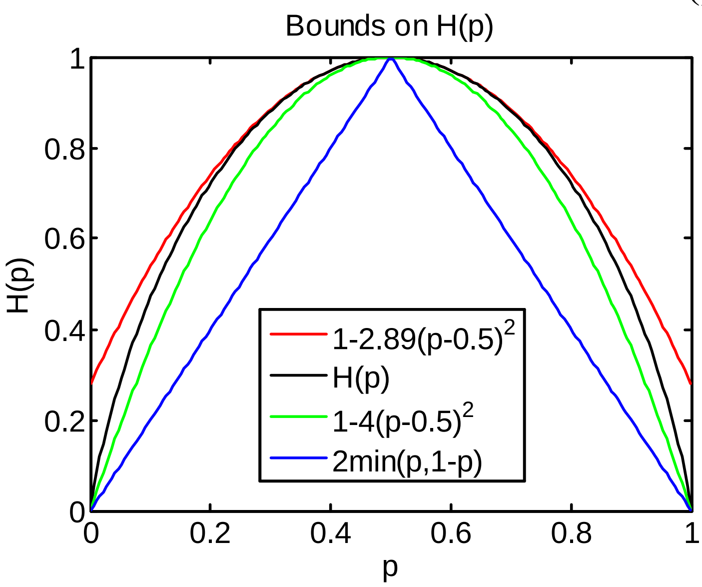
$\mathcal{X}$ = [●; ■; ◆; ✿], $\mathbf{p}_x = [½;\, ¼;\, ^1/_8;\, ^1/_8]$

$$H(x) = H(\mathbf{p}_x) = \sum -\log(p(x))\,p(x)$$
$$= 1\times ½ + 2\times ¼ + 3\times ⅛ + 3\times ⅛ = 1.75 \text{ bits}$$

Bernoulli Entropy Properties

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

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



Quadratic Bounds

$$H(p) \le 1 - 2\log e\,(p-½)^2$$
$$= 1 - 2.89(p-½)^2$$
$$H(p) \ge 1 - 4(p-½)^2$$
$$\ge 2\min(p, 1-p)$$

Proofs in problem sheet

Joint and Conditional Entropy

Joint Entropy: $H(x,y)$

$p(x,y)$	y=0	y=1
x=0	½	¼
x=1	0	¼

$$H(x,y) = E - \log p(x,y)$$
$$= -\tfrac{1}{2}\log\tfrac{1}{2} - \tfrac{1}{4}\log\tfrac{1}{4} - 0\log 0 - \tfrac{1}{4}\log\tfrac{1}{4} = 1.5 \text{ bits}$$

Note: $0 \log 0 = 0$

Conditional Entropy : $H(y \mid x)$

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

Note: rows sum to 1

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

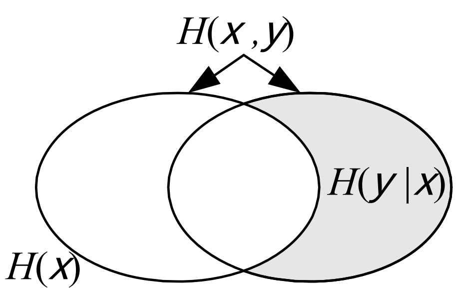
Conditional Entropy – view 1

Additional Entropy:

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

$$p(y \mid x) = p(x,y) \div p(x)$$
$$H(y \mid x) = E - \log p(y \mid x)$$
$$= E\{-\log p(x,y)\} - E\{-\log p(x)\}$$
$$= H(x,y) - H(x) = H(\tfrac{1}{2},\tfrac{1}{4},0,\tfrac{1}{4}) - H(\tfrac{1}{4}) = 0.689 \text{ bits}$$

$H(Y|X)$ is the average <u>additional</u> information in Y when you know X



Conditional Entropy – view 2

Average Row Entropy:

$p(x, y)$	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

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

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

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

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

Chain Rules

- Probabilities

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

- Entropy

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

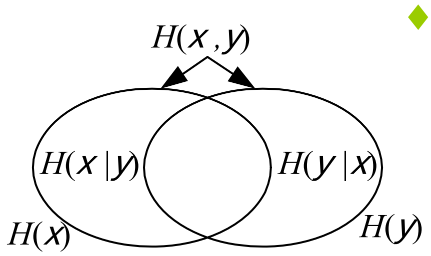
$$H(x_{1:n}) = \sum_{i=1}^{n} H(x_i \mid x_{1:i-1})$$

The log in the definition of entropy converts products of probability into sums of entropy

Summary

- **Entropy:** $H(x) = \sum_{x \in \mathcal{X}} -\log_2(p(x))p(x) = E -\log_2(p_X(x))$
  - Bounded $\quad 0 \le H(x) \le \log|\mathcal{X}|$ ♦
- **Chain Rule:**

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



- **Conditional Entropy:**

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

  - Conditioning reduces entropy $\quad H(y\,|\,x) \le H(y)$ ♦

♦ = inequalities not yet proved

Lecture 2

- 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
- Jensen's Inequality
- Relative Entropy
  - Is a measure of how different two probability mass vectors are
- Information Inequality and its consequences
  - Relative Entropy is always positive
    - Mututal information is positive
    - Uniform bound
    - Conditioning and Correlation reduce entropy

Mutual Information

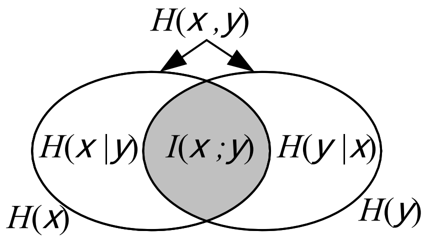
The mutual information is the average amount of information that you get about x from observing the value of y

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

Information in x — Information in x when you already know y

Mutual information is symmetrical

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

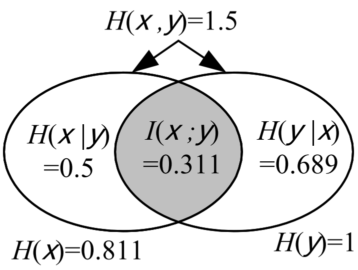


Use ";" to avoid ambiguities between $I(x;y,z)$ and $I(x,y;z)$

Mutual Information Example

$p(x,y)$	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



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

Conditional Mutual Information

Conditional Mutual Information

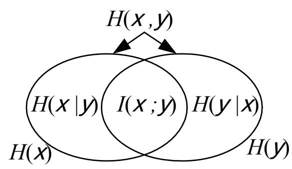
$$I(x;y\,|\,z) = H(x\,|\,z) - H(x\,|\,y,z)$$
$$= H(x\,|\,z) + H(y\,|\,z) - H(x,y\,|\,z)$$

Note: Z conditioning applies to both X and Y

Chain Rule for Mutual Information

$$I(x_1,x_2,x_3;y) = I(x_1;y) + I(x_2;y\,|\,x_1) + I(x_3;y\,|\,x_1,x_2)$$

$$I(x_{1:n};y) = \sum_{i=1}^{n} I(x_i;y\,|\,x_{1:i-1})$$



Review/Preview

- Entropy: $H(x) = \sum_{x\in\mathbf{X}} -\log_2(p(x))p(x) = E - \log_2(p_X(x))$
  - Always positive $H(x) \ge 0$
- 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

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

- 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: $\frac{d^2 f}{dx^2} > 0 \quad \forall x \in (a,b) \qquad \Rightarrow f(x)$ 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 $|\mathcal{X}|$

- $|\mathcal{X}|=1$: $E\, f(X) = f(E\, X) = f(x_1)$
- $|\mathcal{X}|=k$: $E\, f(X) = \sum_{i=1}^{k} p_i f(x_i) = p_k f(x_k) + (1-p_k)\sum_{i=1}^{k-1} \frac{p_i}{1-p_k} f(x_i)$

These sum to 1

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

Assume JI is true for $|\mathcal{X}|=k-1$

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

Can replace by ">" if $f(x)$ is strictly convex unless $p_k \in \{0,1\}$ or $x_k = E(x \mid x \in \{x_{1:k-1}\})$

Jensen's Inequality Example

Mnemonic example:

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

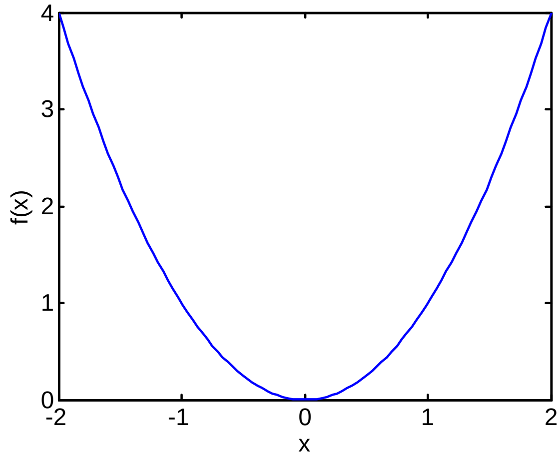
$\mathcal{X} = [-1; +1]$

$\mathbf{p} = [\tfrac{1}{2}; \tfrac{1}{2}]$

$E\, X = 0$

$f(E\, X) = 0$

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



Relative Entropy

Relative Entropy or Kullback-Leibler Divergence between two probability mass vectors **p** and **q**

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

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

$D(\mathbf{p}\|\mathbf{q})$ measures the "distance" between the probability mass functions **p** and **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 **p**, **q** and
- (2) it does not satisfy the triangle inequality.

Relative Entropy Example

$\mathcal{X}$ = [1 2 3 4 5 6]^T

$$\mathbf{p} = \begin{bmatrix} 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \end{bmatrix} \Rightarrow H(\mathbf{p}) = 2.585$$

$$\mathbf{q} = \begin{bmatrix} 1/10 & 1/10 & 1/10 & 1/10 & 1/10 & 1/2 \end{bmatrix} \Rightarrow H(\mathbf{q}) = 2.161$$

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

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

Information Inequality

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

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

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

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

Information Inequality Corollaries

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

$$D(\mathbf{p} \,||\, \mathbf{q}) = E_{\mathbf{p}}\{-\log q(x)\} - H(\mathbf{p}) = \log|\mathcal{X}| - H(\mathbf{p}) \ge 0$$

- Mutual Information is non-negative

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

$$= D(\mathbf{p}_{x,y} \,||\, \mathbf{p}_x \otimes \mathbf{p}_y) \ge 0$$

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

More Corollaries

- Conditioning reduces entropy

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

with equality only if x and y are independent.

- Independence Bound

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

with equality only if all x_i are independent.

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

Conditional Independence Bound

- Conditional Independence Bound

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

- Mutual Information Independence Bound

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

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

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

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.

Summary

- Mutual Information $\quad I(x;y) = H(x) - H(x \mid y) \le H(x)$
- Jensen's Inequality: $\quad f(x)$ convex $\Rightarrow Ef(x) \ge f(Ex)$
- Relative Entropy: $\quad D(\mathbf{p} \,\|\, \mathbf{q}) = E_{\mathbf{p}} \log \dfrac{p(x)}{q(x)} \ge 0$
  - $D(\mathbf{p}\|\mathbf{q}) = 0$ iff $\mathbf{p} \equiv \mathbf{q}$
- Corollaries
  - Uniform Bound: Uniform $\mathbf{p}$ maximizes $H(\mathbf{p})$
  - $I(x\,;\,y) \ge 0 \Rightarrow$ Conditioning reduces entropy
  - Indep bounds: $\quad H(x_{1:n}) \le \sum_{i=1}^{n} H(x_i) \qquad H(x_{1:n} \mid y_{1:n}) \le \sum_{i=1}^{n} H(x_i \mid y_i)$

$$I(x_{1:n}; y_{1:n}) \ge \sum_{i=1}^{n} I(x_i; y_i) \quad \text{if } x_i \text{ or } y_i \text{ are indep}$$

Lecture 3

- Symbol codes
  - uniquely decodable
  - prefix
- Kraft Inequality
- Minimum code length
- Fano Code

Symbol Codes

- Symbol Code: C is a mapping $\mathcal{X} \to \mathcal{D}^+$
  - $\mathcal{D}^+$ = set of all finite length strings from $\mathcal{D}$
  - e.g. $\{E, F, G\} \to \{0,1\}^+ : C(E)=0, C(F)=10, C(G)=11$
- Extension: C^+ is mapping $\mathcal{X}^+ \to \mathcal{D}^+$ formed by concatenating $C(x_i)$ without punctuation
  - e.g. $C^+(EFEEGE) = 01000110$
- Non-singular: $x_1 \neq x_2 \Rightarrow C(x_1) \neq C(x_2)$
- Uniquely Decodable: C^+ is non-singular
  - that is $C^+(x^+)$ is unambiguous

Prefix Codes

- Instantaneous or Prefix Code:
  No codeword is a prefix of another
- Prefix ⇒ Uniquely Decodable ⇒ Non-singular

Examples:

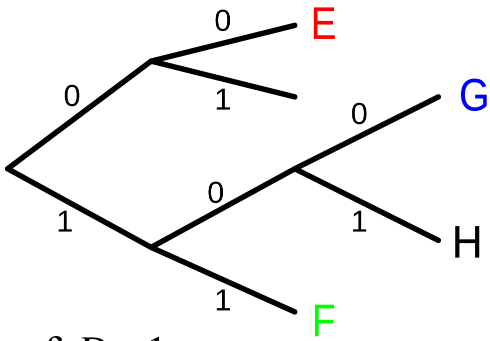
- C(E,F,G,H) = (0, 1, 00, 11) $\overline{P}\,\overline{U}$
- C(E,F) = (0, 101) PU
- C(E,F) = (1, 101) $\overline{P}U$
- C(E,F,G,H) = (00, 01, 10, 11) PU
- C(E,F,G,H) = (0, 01, 011, 111) $\overline{P}U$

Code Tree

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

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

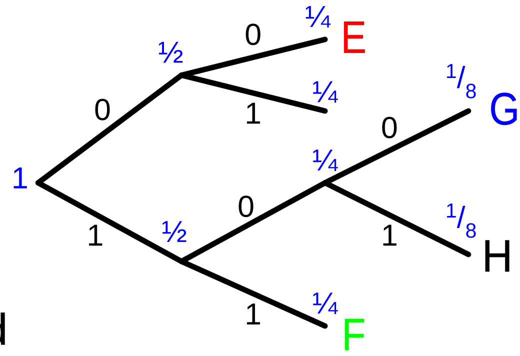
- D branches at each node
- Each node along the path to a leaf is a prefix of the leaf
  ⇒ can't be a leaf itself
- Some leaves may be unused
  all used ⇒ $|\mathcal{X}| - 1$ is a multiple of $D - 1$



$111011000000 \rightarrow$ FHGEE

Kraft Inequality (binary prefix)

- Label each node at depth l with 2^{-l}
- Each node equals the sum of all its leaves
- Codeword lengths:
  $l_1, l_2, \ldots, l_{|\mathcal{X}|} \Rightarrow \sum_{i=1}^{|\mathcal{X}|} 2^{-l_i} \le 1$
- 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

Kraft Inequality

If uniquely decodable C has codeword lengths

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

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

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

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

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

Converse to Kraft Inequality

If $\sum_{i=1}^{|\mathcal{X}|} D^{-l_i} \le 1$ then ∃ a prefix code with codeword lengths $l_1, l_2, \ldots, l_{|\mathcal{X}|}$

Proof:

- Assume $l_i \le l_{i+1}$ and think of codewords as base-D decimals $0.d_1d_2\ldots d_{li}$
- Let codeword $c_k = \sum_{i=1}^{k-1} D^{-l_i}$ with l_k digits
- For any $j < k$ we have $c_k = c_j + \sum_{i=j}^{k-1} D^{-l_i} \ge c_j + D^{-l_j}$
- So c_j cannot be a prefix of c_k because they differ in the first l_j digits.

⇒ non-prefix symbol codes are a waste of time

Kraft Converse Example

Suppose $\mathbf{l} = [2; 2; 3; 3; 3] \Rightarrow \sum_{i=1}^{5|} 2^{-l_i} = 0.875 \le 1$

l_k	$c_k = \sum_{i=1}^{k-1} D^{-l_i}$	Code
2	$0.0 = 0.00_2$	00
2	$0.25 = 0.01_2$	01
3	$0.5 = 0.100_2$	100
3	$0.625 = 0.101_2$	101
3	$0.75 = 0.110_2$	110

Each c_k is obtained by adding 1 to the LSB of the previous row

For code, express c_k in binary and take the first l_k binary places

Minimum Code Length

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

Uniquely decodable code $\Rightarrow L_C \ge H(X)/\log_2 D$

Proof: We define $\mathbf{q}$ by $q(x) = c^{-1}D^{-l(x)}$ where $c = \sum_x D^{-l(x)} \le 1$

$$L_C - H(X)/\log_2 D = E\, l(X) + E \log_D p(X)$$

$$= E\left(-\log_D D^{-l(X)} + \log_D p(X)\right) = E\left(-\log_D cq(X) + \log_D p(X)\right)$$

$$= E\left(\log_D \frac{p(X)}{q(X)}\right) - \log_D c \quad = \log_D 2\left(D(\mathbf{p}\,\|\,\mathbf{q}) - \log c\right) \ge 0$$

with equality only if $c=1$ and $D(\mathbf{p}\|\mathbf{q}) = 0 \Rightarrow \mathbf{p} = \mathbf{q} \Rightarrow l(x) = -\log_D(x)$

Fano Code

Fano Code (also called Shannon-Fano code)

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

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

$H(\mathbf{p}) = 2.81$ bits

$L_{SF} = 2.89$ bits

Always $H(\mathbf{p}) \le L_F \le H(\mathbf{p}) + 1 - 2\min(\mathbf{p})$
$\le H(\mathbf{p}) + 1$

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

Summary

- Kraft Inequality for D-ary codes:
  – any uniquely decodable C has $\sum_{i=1}^{|\mathbf{X}|} D^{-l_i} \le 1$
  – If $\sum_{i=1}^{|\mathbf{X}|} D^{-l_i} \le 1$ then you can create a prefix code
- Uniquely decodable $\Rightarrow L_C \ge H(X)/\log_2 D$
- Fano code
  – Order the probabilities, then repeatedly split in half to form a tree.
  – Intuitively natural but not optimal

Lecture 4

- Optimal Symbol Code
  – Optimality implications
  – Huffman Code
- Optimal Symbol Code lengths
  – Entropy Bound
- Shannon Code

Huffman Code

An Optimal Binary prefix code must satisfy:

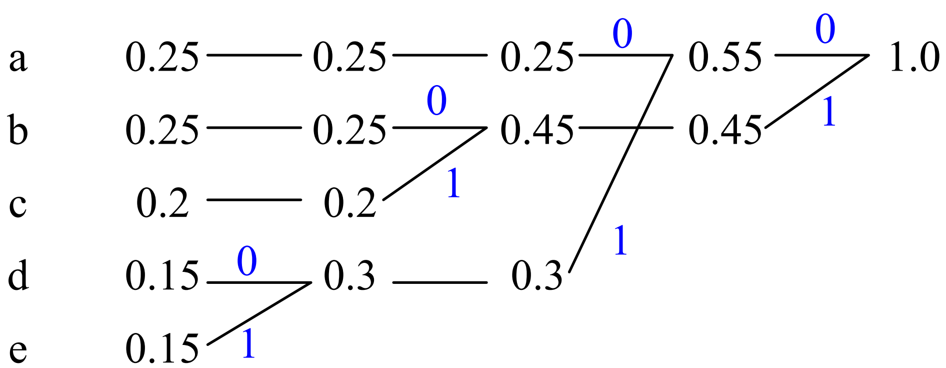
1. $p(x_i) > p(x_j) \Rightarrow l_i \le l_j$ (else swap codewords)
2. The two longest codewords have the same length (else chop a bit off the longer codeword)
3. $\exists$ two longest codewords differing only in the last bit (else chop a bit off all of them)

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

Huffman Code Example

$\boldsymbol{X} = [\text{a, b, c, d, e}],\ p_X = [0.25\ \ 0.25\ \ 0.2\ \ 0.15\ \ 0.15]$



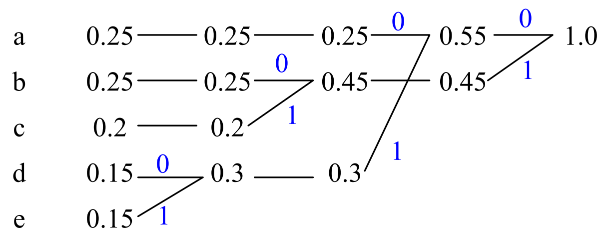
Read diagram backwards for codewords:

$C(\boldsymbol{X}) = [00\ \ 10\ \ 11\ \ 010\ \ 011],\ L_C = 2.3,\ H(\boldsymbol{x}) = 2.286$

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

Huffman Code is Optimal Prefix Code

Huffman traceback gives codes for progressively larger alphabets:



$\mathbf{p}_2$=[0.55 0.45],
$\mathbf{c}_2$=[0 1], L_2=1

$\mathbf{p}_3$=[0.25 0.45 0.3],
$\mathbf{c}_3$=[00 1 01], L_3=1.55

$\mathbf{p}_4$=[0.25 0.25 0.2 0.3],
$\mathbf{c}_4$=[00 10 11 01], L_4=2

$\mathbf{p}_5$=[0.25 0.25 0.2 0.15 0.15],
$\mathbf{c}_5$=[00 10 11 010 011], 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

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

How short are Optimal Codes?

Huffman is optimal but hard to estimate its length.

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

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

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

Simplified version:

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

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

Optimal Codes (non-integer l_i)

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

Use lagrange multiplier:

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

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

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

with these l_i, $E\, l(X) = E - \log_D\big(p(X)\big) = \frac{E - \log_2\big(p(X)\big)}{\log_2 D} = \frac{H(X)}{\log_2 D}$

no uniquely decodable code can do better than this (Kraft inequality)

Shannon Code

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)
- Hence prefix code exists:
  put l_i into ascending order and set

  $$c_k = \sum_{i=1}^{k-1} D^{-l_i} \quad \text{or} \quad c_k = \sum_{i=1}^{k-1} p(x_i)$$

  equally good since $p(x_i) \geq D^{-l_i}$

  to l_i places
- Average length: $\frac{H(X)}{\log_2 D} \leq L_C < \frac{H(X)}{\log_2 D} + 1$ (since we added <1 to optimum values)

Note: since Huffman code is optimal, it also satisfies these limits

Shannon Code Examples

Example 1 (good)

$$\mathbf{p}_x = [0.5 \quad 0.25 \quad 0.125 \quad 0.125]$$
$$-\log_2 \mathbf{p}_x = [1 \quad 2 \quad 3 \quad 3]$$
$$\mathbf{l}_x = \lceil -\log_2 \mathbf{p}_x \rceil = [1 \quad 2 \quad 3 \quad 3]$$
$$L_C = 1.75 \text{ bits}, \ H(x) = 1.75 \text{ bits}$$

Dyadic probabilities

Example 2 (bad)

$$\mathbf{p}_x = [0.99 \quad 0.01]$$
$$-\log_2 \mathbf{p}_x = [0.0145 \quad 6.64]$$
$$\mathbf{l}_x = \lceil -\log_2 \mathbf{p}_x \rceil = [1 \quad 7]$$

(obviously stupid to use 7)

$$L_C = 1.06 \text{ bits}, \ H(x) = 0.08 \text{ bits}$$

We can make $H(x)+1$ bound tighter by encoding longer blocks as a super-symbol

Shannon versus Huffman

Shannon

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

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

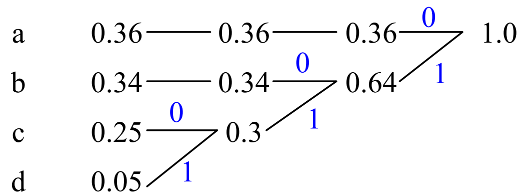
$$\mathbf{l}_S = \lceil -\log_2 \mathbf{p}_x \rceil = [2 \quad 2 \quad 2 \quad 5]$$

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

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

Shannon Competitive Optimality

- $l(x)$ is length of a uniquely decodable code
- $l_S(x) = \lceil -\log p(x) \rceil$ is length of Shannon code

then $p\big(l(x) \le l_S(x) - c\big) \le 2^{1-c}$

Proof: Define $\mathcal{A} = \{x : p(x) < 2^{-l(x)-c+1}\}$ — x with especially short $l(x)$

$$p\big(l(x) \le \lceil -\log p(x) \rceil - c\big) \le p\big(l(x) < -\log p(x) - c + 1\big) = p(x \in \mathcal{A})$$

$$= \sum_{x\in\mathcal{A}} p(x) \le \sum_{x\in\mathcal{A}} \max(p(x) \mid x \in \mathcal{A}) < \sum_{x\in\mathcal{A}} 2^{-l(x)-c+1}$$

$$\le \sum_{x} 2^{-l(x)-c+1} = 2^{-(c-1)} \sum_{x} 2^{-l(x)} \le 2^{-(c-1)}$$

Kraft inequality

(now over all x)

No other symbol code can do much better than Shannon code most of the time

Dyadic Competitive optimality

If $\mathbf{p}$ is dyadic $\Leftrightarrow \log p(x_i)$ is integer, $\forall i$ $\Rightarrow$ Shannon is optimal

then $p\big(l(x) < l_S(x)\big) \le p\big(l(x) > l_S(x)\big)$ with equality iff $l(x) \equiv l_S(x)$

Proof:

- Define $\text{sgn}(x)=\{-1,0,+1\}$ for $\{x<0, x=0, x>0\}$
- Note: $\text{sgn}(i) \le 2^i - 1$ for all integers i — equality iff i=0 or 1

$$p\big(l_S(x) > l(x)\big) - p\big(l_S(x) < l(x)\big) = \sum_x p(x)\,\text{sgn}\big(l_S(x) - l(x)\big)$$

$$\overset{A}{\le} \sum_x p(x)\big(2^{l_S(x)-l(x)} - 1\big) = -1 + \sum_x 2^{-l_S(x)} 2^{l_S(x)-l(x)}$$

$$= -1 + \sum_x 2^{-l(x)} \overset{B}{\le} -1 + 1 = 0$$

sgn() property

dyadic $\Rightarrow p=2^{-l}$

Kraft inequality

Rival code cannot be shorter than Shannon more than half the time.

equality @ A $\Rightarrow l(x) = l_S(x) - \{0,1\}$ but $l(x) < l_S(x)$ would violate Kraft @ B since Shannon has Σ=1

Shannon with wrong distribution

If the real distribution of x is $\mathbf{p}$ but you assign Shannon lengths using the distribution $\mathbf{q}$ what is the penalty ?

Answer: $D(\mathbf{p}||\mathbf{q})$

Proof: $$E\,l(X) = \sum_i p_i \lceil -\log q_i \rceil < \sum_i p_i (1 - \log q_i)$$

$$= \sum_i p_i \left(1 + \log\frac{p_i}{q_i} - \log p_i\right)$$

$$= 1 + D(\mathbf{p}\,||\,\mathbf{q}) + H(\mathbf{p})$$

Therefore

$$H(\mathbf{p}) + D(\mathbf{p}\,||\,\mathbf{q}) \le E\,l(X) < H(\mathbf{p}) + D(\mathbf{p}\,||\,\mathbf{q}) + 1$$

Proof of lower limit is similar but without the 1

If you use the wrong distribution, the penalty is $D(\mathbf{p}||\mathbf{q})$

Summary

- Any uniquely decodable code: $E\,l(x) \ge H_D(x) = \dfrac{H(x)}{\log_2 D}$
- Fano Code: $H_D(x) \le E\,l(x) \le H_D(x) + 1$
  - Intuitively natural top-down design
- Huffman Code:
  - Bottom-up design
  - Optimal $\Rightarrow$ at least as good as Shannon/Fano
- Shannon Code: $l_i = \lceil -\log_D p_i \rceil$ $\quad H_D(x) \le E\,l(x) \le H_D(x) + 1$
  - Close to optimal and easier to prove bounds

Note: Not everyone agrees on the names of Shannon and Fano codes

Lecture 5

- Stochastic Processes
- Entropy Rate
- Markov Processes
- Hidden Markov Processes

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

- Entropy rate estimates the additional entropy per new sample.
- Gives a lower bound on number of code bits per sample.
- If the x_i are not i.i.d. the entropy rate limit may not exist.

Examples:

- x_i i.i.d. random variables: $H(\boldsymbol{X}) = H(x_i)$
- x_i indep, $H(x_i)$ = 0 1 00 11 0000 1111 00000000 … no convergence

Lemma: Limit of Cesàro Mean

$$a_n \to b \quad \Rightarrow \quad \frac{1}{n}\sum_{k=1}^{n} a_k \to b$$

Proof:

- Choose $\varepsilon > 0$ and find N_0 such that $|a_n - b| < \tfrac{1}{2}\varepsilon \quad \forall n > N_0$
- Set $N_1 = 2N_0\varepsilon^{-1}\max(|a_r - b|)$ for $r \in [1, N_0]$
- Then $\forall n > N_1$

$$n^{-1}\sum_{k=1}^{n}|a_k - b| = n^{-1}\sum_{k=1}^{N_0}|a_k - b| + n^{-1}\sum_{k=N_0+1}^{n}|a_k - b|$$

$$\le N_1^{-1}N_0\left(\tfrac{1}{2}N_0^{-1}N_1\varepsilon\right) + n^{-1}n\left(\tfrac{1}{2}\varepsilon\right)$$

$$= \tfrac{1}{2}\varepsilon + \tfrac{1}{2}\varepsilon = \varepsilon$$

The partial means of a_k are called Cesàro Means

Stationary Process

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

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

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

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

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 +ve, decreasing $\Rightarrow$ tends to a limit, say b

Hence from Cesàro Mean lemma:

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

Block Coding

If x_i is a stochastic process

- encode blocks of n symbols
- 1-bit penalty of Shannon/Huffman is now shared between n symbols

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

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

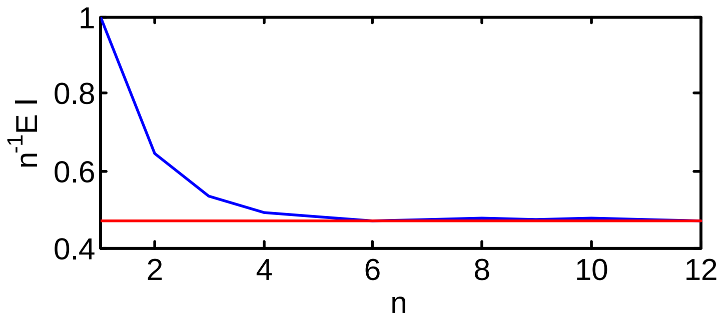
$$n^{-1} H(x_{1:n}) \to H(\boldsymbol{X}) \quad \Rightarrow \quad n^{-1} E\, l(x_{1:n}) \to H(\boldsymbol{X})$$

The extra 1 bit inefficiency becomes insignificant for large blocks

Block Coding Example

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

$H(x_i) = 0.469$



- n=1

sym	A	B
prob	0.9	0.1
code	0	1

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

- 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

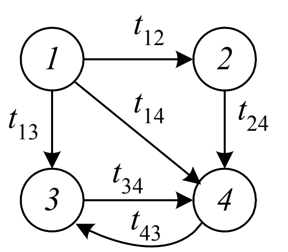
sym	AAA	AAB	...	BBA	BBB
prob	0.729	0.081	...	0.009	0.001
code	0	101	...	10010	10011

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

Markov Process

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
  - 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 a to any b in a finite number of steps

- irreducible iff $(\mathbf{I}+\mathbf{T}^T)^{|\mathcal{X}|-1}$ has no zero entries

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

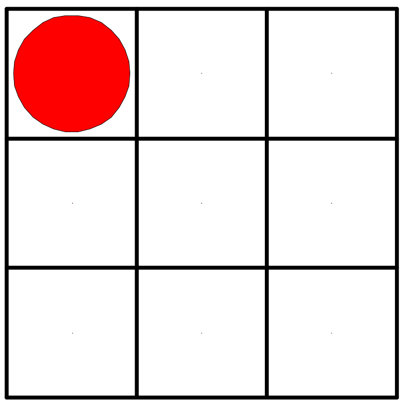
then it has exactly one stationary distribution, $\mathbf{p}_\$$.

– $\mathbf{p}_\$$ 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 $\mathbf{1} = [1 \quad 1 \quad \cdots \quad 1]^T$

Chess Board

H(p1)=0, H(p1 | p0)=0

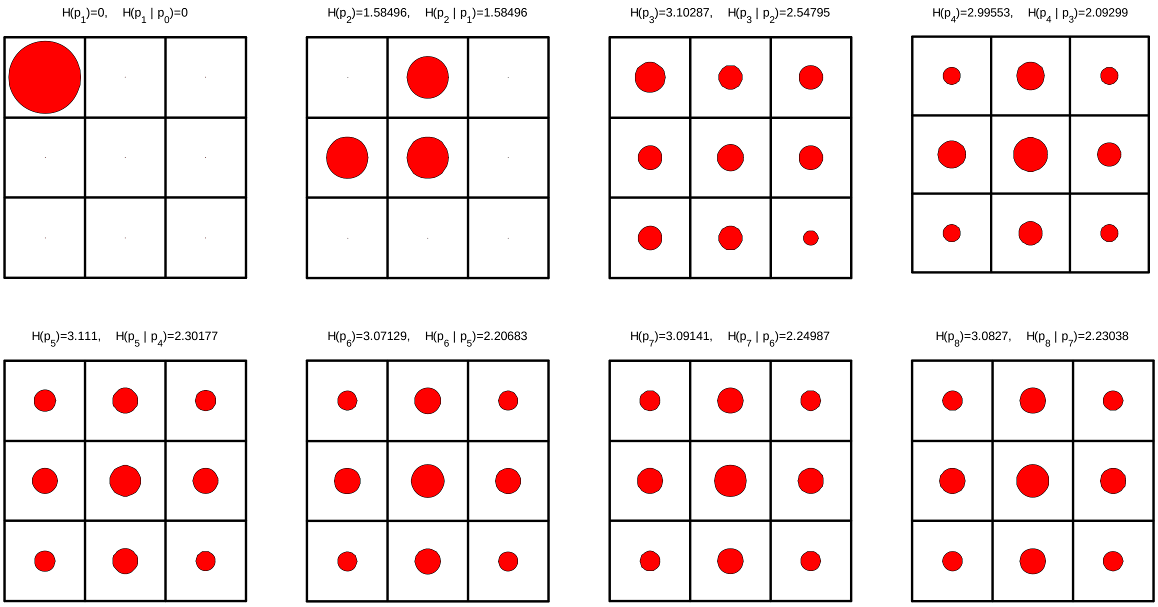


Random Walk

- Move ⇔ ⇕ ⇖ ⇗ ⇙ ⇘ equal prob
- $\mathbf{p}_1 = [1\ 0\ \ldots\ 0]^T$
  - $H(\mathbf{p}_1) = 0$
- $\mathbf{p}_\$ = {}^1/_{40} \times [3\ 5\ 3\ 5\ 8\ 5\ 3\ 5\ 3]^T$
  - $H(\mathbf{p}_\$) = 3.0855$
- $H(\mathcal{X}) = \lim_{n\to\infty} H(x_n \mid x_{n-1}) = \sum_{i,j} - p_{\$,i} t_{i,j} \log(t_{i,j})$
  - $H(\mathcal{X}) = 2.2365$

Time-invariant and $\mathbf{p}_1 = \mathbf{p}_\$ \Rightarrow$ stationary

Chess Board Frames



ALOHA Wireless Example

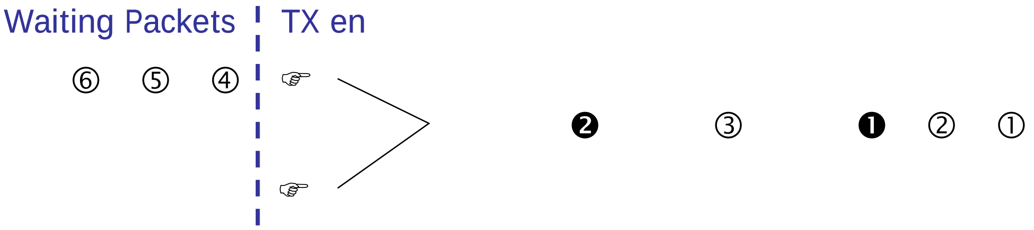
M users share wireless transmission channel

- For each user independently in each timeslot:
  - if its queue is non-empty it transmits with prob q
  - a new packet arrives for transmission with prob p
- If two packets collide, they stay in the queues
- At time t, queue sizes are $\mathbf{x}_t = (n_1, \ldots, n_M)$
  - $\{\mathbf{x}_t\}$ is Markov since $p(\mathbf{x}_t)$ depends only on $\mathbf{x}_{t-1}$

Transmit vector is $\mathbf{y}_t$:
$$p(y_{i,t} = 1) = \begin{cases} 0 & x_{i,t} = 0 \\ q & x_{i,t} > 0 \end{cases}$$

- $\{\mathbf{y}_t\}$ is not Markov since $p(\mathbf{y}_t)$ is determined by $\mathbf{x}_t$ but is not determined by $\mathbf{y}_{t-1}$. $\{\mathbf{y}_t\}$ is called a Hidden Markov Process.

ALOHA example



x:	3	2	1	2	1	0	1	2
	0	1	1	1	1	2	2	1
TX enable, **e**:	1	0	0	1	1	1	1	1
	1	1	0	0	1	1	0	0
y:	1	0	0	1	1	0	1	1
	0	1	0	0	1	1	0	0

$\mathbf{y} = (\mathbf{x}>0)\mathbf{e}$ is a deterministic function of the Markov $[\mathbf{x};\ \mathbf{e}]$

Hidden Markov Process

If $\{x_i\}$ is a stationary Markov process and $y=f(x)$ then $\{y_i\}$ is a stationary Hidden Markov process.

What is entropy rate $H(\mathcal{Y})$?

– Stationarity $\Rightarrow$ $H(y_n \mid y_{1:n-1}) \geq H(\mathcal{Y})$ and $\underset{n\to\infty}{\to} H(\mathcal{Y})$

– Also $H(y_n \mid y_{1:n-1}, x_1) \overset{(1)}{\leq} H(\mathcal{Y})$ and $\underset{n\to\infty}{\overset{(2)}{\to}} H(\mathcal{Y})$

So $H(\mathcal{Y})$ is sandwiched between two quantities which converge to the same value for large n.

Proof of (1) and (2) on next slides

Hidden Markov Process – (1)

Proof (1): $H(y_n \mid y_{1:n-1}, x_1) \le H(\mathcal{Y})$

$$H(y_n \mid y_{1:n-1}, x_1) = H(y_n \mid y_{1:n-1}, x_{-k:1}) \quad \forall k$$ *x* markov

$$= H(y_n \mid y_{1:n-1}, x_{-k:1}, y_{-k:1}) = H(y_n \mid y_{-k:n-1}, x_{-k:1})$$ *y*=*f*(*x*)

$$\le H(y_n \mid y_{-k:n-1}) \quad \forall k$$ conditioning reduces entropy

$$= H(y_{k+n} \mid y_{0:k+n-1}) \underset{k\to\infty}{\to} H(\mathcal{Y})$$ *y* stationary

Just knowing x_1 in addition to $y_{1:n-1}$ reduces the conditional entropy to below the entropy rate.

Hidden Markov Process – (2)

Proof (2): $H(y_n \mid y_{1:n-1}, x_1) \underset{n\to\infty}{\to} H(\mathcal{Y})$

Note that $\sum_{n=1}^{k} I(x_1; y_n \mid y_{1:n-1}) = I(x_1; y_{1:k})$ chain rule

$\le H(x_1)$ defⁿ of I(A;B)

Hence $I(x_1; y_n \mid y_{1:n-1}) \underset{n\to\infty}{\to} 0$ bounded sum of non-negative terms

So $H(y_n \mid y_{1:n-1}, x_1) = H(y_n \mid y_{1:n-1}) - I(x_1; y_n \mid y_{1:n-1})$ defⁿ of I(A;B)

$\underset{n\to\infty}{\to} H(\mathcal{Y}) - 0$

The influence of x_1 on y_n decreases over time.

Summary

- Entropy Rate: $H(\boldsymbol{\mathcal{X}}) = \lim_{n\to\infty} \frac{1}{n} H(x_{1:n})$ if it exists
  - $\{x_i\}$ stationary: $H(\boldsymbol{\mathcal{X}}) = \lim_{n\to\infty} H(x_n \mid x_{1:n-1})$
  - $\{x_i\}$ stationary Markov:

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

  - $y = f(x)$: Hidden Markov:

$$H(y_n \mid y_{1:n-1}, x_1) \le H(\boldsymbol{\mathcal{Y}}) \le H(y_n \mid y_{1:n-1})$$

with both sides tending to $H(\boldsymbol{\mathcal{Y}})$

Lecture 6

- Stream Codes
- Arithmetic Coding
- Lempel-Ziv Coding

Huffman: Good and Bad

Good

– Shortest possible symbol code $\frac{H(x)}{\log_2 D} \le L_S \le \frac{H(x)}{\log_2 D} + 1$

Bad

– Redundancy of up to 1 bit per symbol
- Expensive if $H(x)$ is small
- Less so if you use a block of N symbols
- Redundancy equals zero iff $p(x_i)=2^{-k(i)}\ \forall i$

– Must recompute entire code if any symbol probability changes
- A block of N symbols needs $|\boldsymbol{X}|^N$ pre-calculated probabilities

Arithmetic Coding

- Take all possible blocks of N symbols and sort into lexical order, $\mathbf{x}_r$ for r=1: $|\boldsymbol{X}|^N$
- Calculate cumulative probabilities in binary:

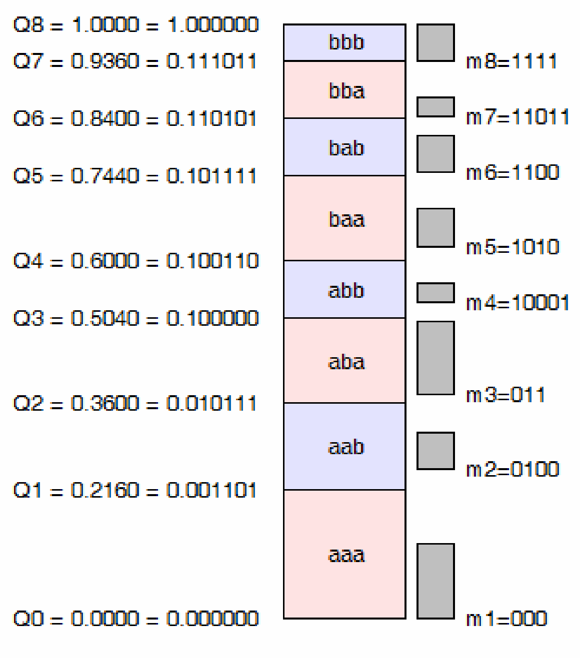
$$Q_r = \sum_{i \le r} p(\mathbf{x}_i),\ Q_0 = 0$$

- To encode $\mathbf{x}_r$, transmit enough binary places to define the interval (Q_{r-1}, Q_r) unambiguously.
- Use first l_r places of $m_r 2^{-l_r}$ where l_r is least integer with

$$Q_{r-1} \le m_r 2^{-l_r} < (m_r+1)2^{-l_r} \le Q_r$$

$\boldsymbol{X}$=[a b], $\mathbf{p}$=[0.6 0.4], N=3

Code



Arithmetic Coding – Code lengths

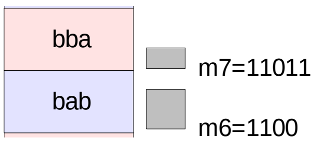
- The interval corresponding to x_r has width $p(x_r) = Q_r - Q_{r-1} \stackrel{\Delta}{=} 2^{-d_r}$
- Define $k_r = \lceil d_r \rceil \Rightarrow d_r \le k_r < d_r + 1 \Rightarrow \tfrac{1}{2} p(x_r) < 2^{-k_r} \le p(x_r)$
- Set $m_r = \lceil 2^{k_r} Q_{r-1} \rceil \Rightarrow Q_{r-1} \le m_r 2^{-k_r}$ Q_{r-1} rounded up to k_r bits
- If $(m_r + 1)2^{-k_r} \le Q_r$ then set $l_r = k_r$; otherwise
  - set $l_r = k_r + 1$ and redefine $m_r = \lceil 2^{l_r} Q_{r-1} \rceil \Rightarrow (m_r - 1)2^{-l_r} < Q_{r-1} \le m_r 2^{-l_r}$
  - now $(m_r + 1)2^{-l_r} = (m_r - 1)2^{-l_r} + 2^{-k_r} < Q_{r-1} + p(x_r) = Q_r$
- We always have $l_r \le k_r + 1 < d_r + 2 = -\log(p_r) + 2$
  - Always within 2 bits of the optimum code for the block (k_r is Shannon len)

$d_{6,7} = -\log p(x_{6,7}) = -\log 0.096 = 3.38$ bits

Q7 = 0.9360 = 0.111011

Q6 = 0.8400 = 0.110101

Q5 = 0.7440 = 0.101111



$k_7 = 4, m_7 = 14, 15 \times 2^{-4} > Q_7$ ✗

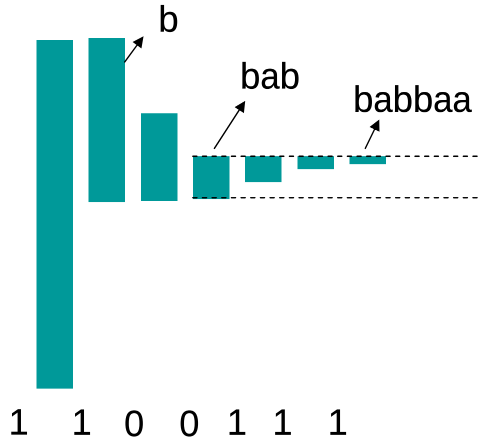
$\Rightarrow l_7 = 5, m_7 = 27, 28 \times 2^{-5} \le Q_7$ ✓

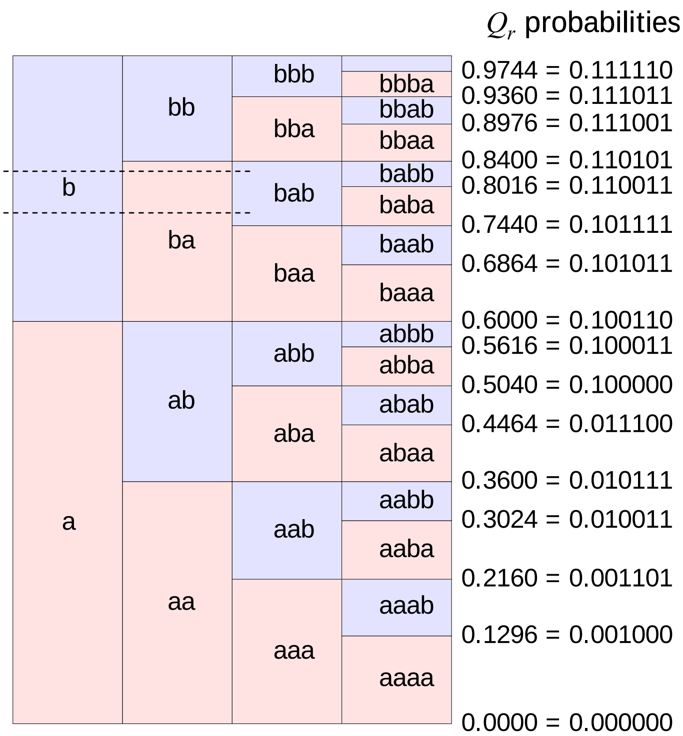
$k_6 = 4, m_6 = 12, 13 \times 2^{-4} < Q_6$ ✓

Arithmetic Coding - Advantages

- Long blocks can be used
  - Symbol blocks are sorted lexically rather than in probability order
  - Receiver can start decoding symbols before the entire code has been received
  - Transmitter and receiver can work out the codes on the fly
    - no need to store entire codebook
- Transmitter and receiver can use identical finite-precision arithmetic
  - rounding errors are the same at transmitter and receiver
  - rounding errors affect code lengths slightly but not transmission accuracy

Arithmetic Coding Receiver





Each additional bit received narrows down the possible interval.

$\boldsymbol{X}$=[a b], **p**=[0.6 0.4]

Arithmetic Coding/Decoding

	Transmitter		Send	Receiver			
Input	Min	Max		Min	Test	Max	Output
	00000000	11111111		00000000	10011001	11111111	
b	10011001	11111111	1		10011001		
a	10011001	11010111					
b	10111110	11010111					
b	11001101	11010111	10		10011001		b
a	11001101	11010011		10011001	11010111	11111111	
a	11001101	11010000					
a	11001101	11001111	011		11010111		a
				10011001	10111110	11010111	b
b	11001110	11001111	1	10111110	11001101	11010111	b
				11001101	11010011	11010111	a
				11001101	11010000	11010011	a
				11001101	11001111	11010000	
…	…	…		…	…	…	…

- Min/Max give the limits of the input or output interval; identical in transmitter and receiver.
- Blue denotes transmitted bits - they are compared with the corresponding bits of the receiver's test value and Red bit show the first difference. Gray identifies unchanged words.

Arithmetic Coding Algorithm

Input Symbols: $\mathcal{X}$ = [a b], $\mathbf{p} = [p \quad q]$

$[min\ ,\ max]$ = Input Probability Range

Note: only keep untransmitted bits of min and max

Coding Algorithm:

Initialize $[min\ ,\ max] = [000\ldots0\ ,\ 111\ldots1]$

For each input symbol, s

If $s{=}a$ then $max{=}min{+}p(max{-}min)$ else $min{=}min{+}p(max{-}min)$

while min and max have the same MSB

transmit MSB and set $min{=}(min{<}{<}1)$ and $max{=}(max{<}{<}1){+}1$

end while

end for

- Decoder is almost identical. Identical rounding errors $\Rightarrow$ no symbol errors.
- Simple to modify algorithm for |X|>2 and/or D>2.
- Need to protect against range underflow when [x y] = [011111…, 100000…].

Adaptive Probabilities

Number of guesses for next letter (a-z, space):

o	r	a	n	g	e	s		a	n	d		l	e	m	o	n	s
17	7	8	4	1	1	2	1	1	5	1	1	1	1	1	1	1	1

We can change the input symbol probabilities based on the context (= the past input sequence)

Example: Bernoulli source with unknown p. Adapt p based on symbol frequencies so far:

$$\mathcal{X} = [\text{a b}], \quad \mathbf{p}_n = [1{-}p_n \quad p_n], \quad p_n = \frac{1 + \underset{1\le i<n}{\text{count}}(x_i = \text{b})}{1+n}$$

Adaptive Arithmetic Coding

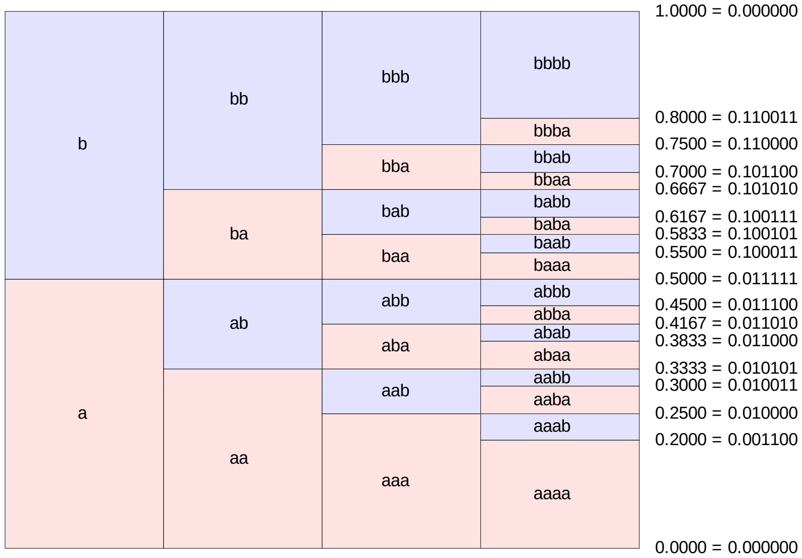
$$p_n = \frac{1 + \operatorname{count}_{1 \le i < n}(x_i = \mathrm{b})}{1 + n}$$

$p_1 = 0.5$

$p_2 = {}^1\!/_3 \text{ or } {}^2\!/_3$

$p_3 = ¼ \text{ or } ½ \text{ or } ¾$

$p_4 = \ldots$



Coder and decoder only need to calculate the probabilities along the path that actually occurs

Lempel-Ziv Coding

Memorize previously occurring substrings in the input data

- parse input into the shortest possible distinct ‘phrases’
- number the phrases starting from 1 (0 is the empty string)

  1011010100010...

  12 3 4 5 6 7
- each phrase consists of a previously occurring phrase (head) followed by an additional 0 or 1 (tail)
- transmit code for head followed by the additional bit for tail

  01001121402010...
- for head use enough bits for the max phrase number so far:

  100011101100001000010...
- decoder constructs an identical dictionary

prefix codes are underlined

Lempel-Ziv Example

Input = 1011010100010010001001010010

Dictionary		Send	Decode
0000	ϕ	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

Improvement

- Each head can only be used twice so at its second use we can:
  - Omit the tail bit
  - Delete head from the dictionary and re-use dictionary entry

LempelZiv Comments

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

- Widely used
  - many versions: compress, gzip, TIFF, LZW, LZ77, ...
  - different dictionary handling, etc
- Excellent compression in practice
  - many files contain repetitive sequences
  - worse than arithmetic coding for text files
- Asymptotically optimum on stationary ergodic source (i.e. achieves entropy rate)
  - $\{X_i\}$ stationary ergodic $\Rightarrow \limsup_{n\to\infty} n^{-1} l(X_{1:n}) \le H(\boldsymbol{X})$ with prob 1
    - Proof: C&T chapter 12.10
  - may only approach this for an enormous file

Summary

- Stream Codes
  - Encoder and decoder operate sequentially
    - no blocking of input symbols required
  - Not forced to send $\geq$1 bit per input symbol
    - can achieve entropy rate even when $H(\boldsymbol{X})<1$
- Require a Perfect Channel
  - A single transmission error causes multiple wrong output symbols
  - Use finite length blocks to limit the damage

Lecture 7

- Markov Chains
- 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

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

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*.

A common special case of a Markov chain is when $z = f(y)$

Markov Chain Symmetry

Iff $x \to y \to z$

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

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

Hence x and z are conditionally independent given y

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

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

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

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

Markov chain property is symmetrical

Data Processing Theorem

If $x \to y \to z$ then $I(x\,;y) \ge I(x\,;\,z)$

– processing y cannot add new information about x

If $x \to y \to z$ then $I(x\,;y) \ge I(x\,;\,y\,|\,z)$

– Knowing z can only decrease the amount x tells you about y

Proof:

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

but $I(x;z\,|\,y) \overset{(a)}{=} 0$

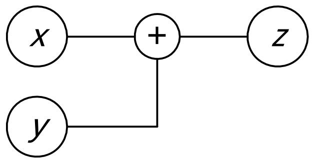
hence $I(x;y) = I(x;z) + I(x;y\,|\,z)$

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

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

Non-Markov: Conditioning can increase I

Noisy Channel: $z=x+y$



- $\mathcal{X}=\mathcal{Y}=[0,1]^T \quad \mathbf{p}_X=\mathbf{p}_Y=[½,½]^T$
- $I(x;y)=0$ since independent
- but $I(x;y\,|\,z)=½$

		Z		
		0	1	2
XY	00	¼		
	01		¼	
	10		¼	
	11			¼

$H(x\,|\,z) = H(y\,|\,z) = H(x, y\,|\,z)$
$= 0\times¼+1\times½+0\times¼ = ½$
since in each case $z\neq1 \Rightarrow H()=0$

$I(x; y\,|\,z) = H(x\,|\,z)+H(y\,|\,z)-H(x, y\,|\,z)$
$= ½+½-½ = ½$

If you know z, then x and y are no longer independent

Long Markov Chains

If $x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4 \rightarrow x_5 \rightarrow x_6$

then Mutual Information increases as you get closer together:

- e.g. $I(x_3, x_4) \geq I(x_2, x_4) \geq I(x_1, x_5) \geq I(x_1, x_6)$

Sufficient Statistics

If pdf of x depends on a parameter θ and you extract a statistic $T(x)$ from your observation,

then $\theta \to x \to T(x) \quad \Rightarrow \quad I(\theta;T(x)) \le I(\theta;x)$

$T(x)$ is sufficient for θ if the stronger condition:

$$\theta \to x \to T(x) \to \theta \quad \Leftrightarrow \quad I(\theta;T(x)) = I(\theta;x)$$

$$\Leftrightarrow \theta \to T(x) \to x \to \theta$$

$$\Leftrightarrow p(x \mid T(x),\theta) = p(x \mid T(x))$$

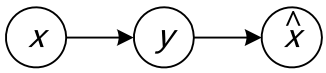
Example: $x_i \sim \text{Bernoulli}(\theta)$,

$$T(x_{1:n}) = \sum_{i=1}^{n} x_i \qquad p\left(X_{1:n} = x_{1:n} \mid \theta, \sum x_i = k\right) = \begin{cases} {}_nC_k^{\,-1} & \text{if } \sum x_i = k \\ 0 & \text{if } \sum x_i \ne k \end{cases}$$

independent of $\theta \Rightarrow$ sufficient

Fano's Inequality

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



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

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

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

Proof: Define a random variable $e = (\hat{x} \ne x)\,?\,1:0$

$$H(e,x \mid y) = H(x \mid y) + H(e \mid x,y) = H(e \mid y) + H(x \mid e,y) \quad \text{chain rule}$$

$$\Rightarrow H(x \mid y) + 0 \le H(e) + H(x \mid e,y) \quad H\ge 0;\ H(e \mid y) \le H(e)$$

$$= H(e) + H(x \mid y, e=0)(1-p_e) + H(x \mid y, e=1)p_e$$

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

Fano's inequality is used whenever you need to show that errors are inevitable

Fano Example

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

$\mathcal{Y} = \{1:2\}$ 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 $\hat{x} = y$

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

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

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

Summary

- Markov: $x \to y \to z \Leftrightarrow p(z\,|\,x, y) = p(z\,|\,y) \Leftrightarrow I(x; z\,|\,y) = 0$
- Data Processing Theorem: if $x \to y \to z$ then
  - $I(x\,;\,y) \ge I(x\,;\,z)$
  - $I(x\,;\,y) \ge I(x\,;\,y\,|\,z)$ can be false if not Markov
- Fano's Inequality: if $x \to y \to \hat{x}$ then

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

weaker but easier to use since independent of p_e

Lecture 8

- Weak Law of Large Numbers
- The Typical Set
  - Size and total probability
- Asymptotic Equipartition Principle

Strong and Weak Typicality

$\mathcal{X}$ = {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

- strongly typical $\Rightarrow$ correct proportions
  aaaabbcd $-\log p(\mathbf{x}) = 14 = 8\times 1.75$
- [weakly] typical $\Rightarrow \log p(\mathbf{x}) = nH(\mathbf{x})$
  aabbbbbb $-\log p(\mathbf{x}) = 14 = 8\times 1.75$
- not typical at all $\Rightarrow \log p(\mathbf{x}) \neq nH(\mathbf{x})$
  dddddddd $-\log p(\mathbf{x}) = 24$

Strongly Typical $\Rightarrow$ Typical

Convergence of Random Numbers

- Convergence

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

Example: $x_n = \pm 2^{-n}, \quad p = [½; ½]$

choose $m = 1 - \log \varepsilon$

- Convergence in probability (weaker than convergence)

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

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

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

Note: y can be a constant or another random variable

Weak law of Large Numbers

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

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

As n increases, Var s_n gets smaller and the values become clustered around the mean

WLLN: $s_n \overset{\text{prob}}{\to} \mu$

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

The “strong law of large numbers” says that convergence is actually almost sure provided that X has finite variance

Proof of WLLN

- Chebyshev's Inequality

$$\operatorname{Var} y = E(y-\mu)^2 = \sum_{y \in \mathcal{Y}} (y-\mu)^2 p(y)$$

$$\geq \sum_{y:|y-\mu|>\varepsilon} (y-\mu)^2 p(y) \geq \sum_{y:|y-\mu|>\varepsilon} \varepsilon^2 p(y) = \varepsilon^2 p(|y-\mu|>\varepsilon)$$

For any choice of ε

- WLLN $\quad s_n = \frac{1}{n}\sum_{i=1}^{n} x_i$ where $E\,x_i = \mu$ and $\operatorname{Var} x_i = \sigma^2$

$$\varepsilon^2 p(|s_n-\mu|>\varepsilon) \leq \operatorname{Var} s_n = \frac{\sigma^2}{n} \xrightarrow[n\to\infty]{} 0$$

Hence $\quad s_n \xrightarrow{\text{prob}} \mu$

Actually true even if $\sigma = \infty$

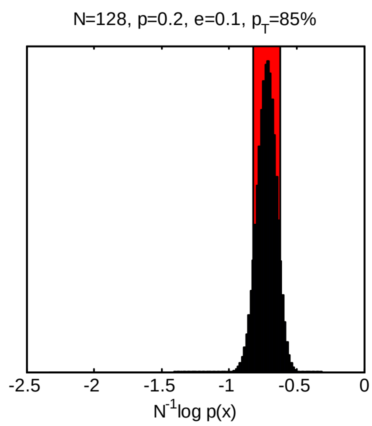
Typical Set

$\mathbf{x}^n$ is the i.i.d. sequence $\{x_i\}$ for $1 \leq i \leq n$

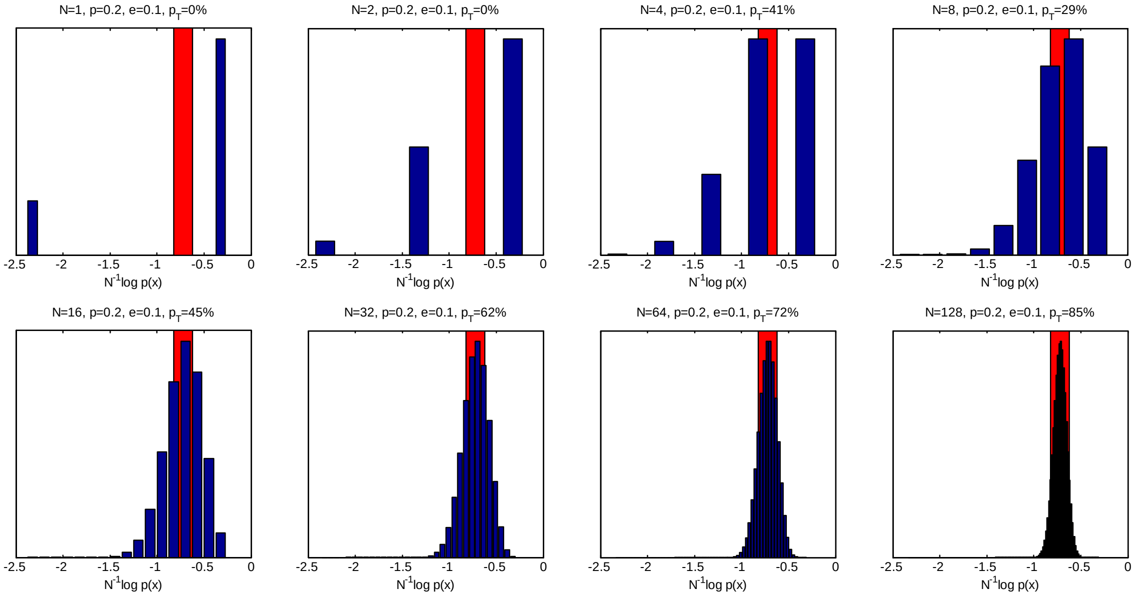
- Prob of a particular sequence is $p(\mathbf{x}) = \prod_{i=1}^{n} p(x_i)$
- $E -\log p(\mathbf{x}) = n\, E -\log p(x_i) = nH(x)$
- Typical set: $T_\varepsilon^{(n)} = \left\{ \mathbf{x} \in \mathcal{X}^n : \left| -n^{-1}\log p(\mathbf{x}) - H(x) \right| < \varepsilon \right\}$

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 $T_{0.1}^{(n)}$



Typical Set Frames



01, 11, 0011, 1101, 00110100, 000010001, 0100100, 0001000010000,
0000100001000000, 0000000010000000

Typical Set: Properties

1. Individual prob: $\mathbf{x} \in T_\varepsilon^{(n)} \Rightarrow \log p(\mathbf{x}) = -nH(X) \pm n\varepsilon$
2. Total prob: $p(\mathbf{x} \in T_\varepsilon^{(n)}) > 1-\varepsilon \text{ for } n > N_\varepsilon$
3. Size: $(1-\varepsilon)2^{n(H(X)-\varepsilon)} \overset{n>N_\varepsilon}{<} \left|T_\varepsilon^{(n)}\right| \le 2^{n(H(X)+\varepsilon)}$

Proof 2: $-n^{-1}\log p(\mathbf{x}) = n^{-1}\sum_{i=1}^{n} -\log p(x_i) \overset{\text{prob}}{\to} E -\log p(x_i) = H(X)$

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

Proof 3a: $\text{f.l.e. } n, \quad 1-\varepsilon < p(\mathbf{x} \in T_\varepsilon^{(n)}) \le \sum_{\mathbf{x}\in T_\varepsilon^{(n)}} 2^{-n(H(X)-\varepsilon)} = 2^{-n(H(X)-\varepsilon)}\left|T_\varepsilon^{(n)}\right|$

Proof 3b: $1 = \sum_{\mathbf{x}} p(\mathbf{x}) \ge \sum_{\mathbf{x}\in T_\varepsilon^{(n)}} p(\mathbf{x}) \ge \sum_{\mathbf{x}\in T_\varepsilon^{(n)}} 2^{-n(H(X)+\varepsilon)} = 2^{-n(H(X)+\varepsilon)}\left|T_\varepsilon^{(n)}\right|$

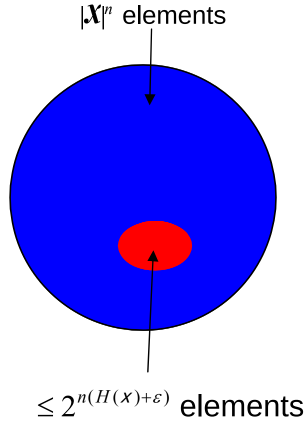
Asymptotic Equipartition Principle

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

Coding consequence

- $\mathbf{x} \in T_\varepsilon^{(n)}$: '0' + at most $1+n(H+\varepsilon)$ bits
- $\mathbf{x} \notin T_\varepsilon^{(n)}$: '1' + at most $1+n\log|\boldsymbol{X}|$ bits
- L = Average code length

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



Source Coding & Data Compression

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

- make a lossless code using only $H(X)+\delta$ bits per symbol on average: $L \le n\left(H + \varepsilon + \varepsilon\log|\boldsymbol{X}| + 2n^{-1}\right)$
- make a code with an error probability < ε using $H(X)+\delta$ bits for each symbol
  - just code T_ε using $n(H+\varepsilon+n^{-1})$ bits and use a random wrong code if $\mathbf{x} \notin T_\varepsilon$

What N_ε ensures that $p\left(\mathbf{x} \in T_\varepsilon^{(n)}\right) > 1-\varepsilon$?

From WLLN, if $\mathrm{Var}\left(-\log p(x_i)\right) = \sigma^2$ then for any n and ε

$$\varepsilon^2 p\left(\left|\frac{1}{n}\sum_{i=1}^{n} -\log p(x_i) - H(X)\right| > \varepsilon\right) \le \frac{\sigma^2}{n} \quad \Rightarrow \quad p\left(\mathbf{x} \notin T_\varepsilon^{(n)}\right) \le \frac{\sigma^2}{n\varepsilon^2}$$ Chebyshev

Choose $N_\varepsilon = \sigma^2 \varepsilon^{-3} \quad \Rightarrow \quad p\left(\mathbf{x} \in T_\varepsilon^{(n)}\right) > 1-\varepsilon$

N_ε increases radidly for small ε

For this choice of N_ε , if $\mathbf{x} \in T_\varepsilon^{(n)}$

$$\log p(\mathbf{x}) = -nH(X) \pm n\varepsilon = -nH(X) \pm \sigma^2 \varepsilon^{-2}$$

So within $T_\varepsilon^{(n)}$, $p(\mathbf{x})$ can vary by a factor of $2^{2\sigma^2\varepsilon^{-2}}$

Within the Typical Set, $p(\mathbf{x})$ can actually vary a great deal when ε is small

Smallest high-probability Set

$T_\varepsilon^{(n)}$ is a small subset of $\mathcal{X}^n$ 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 $\left|S^{(n)}\right| < 2^{n(H(X)-2\varepsilon)}$

$$\begin{aligned} p\left(\mathbf{x} \in S^{(n)}\right) &= p\left(\mathbf{x} \in S^{(n)} \cap T_\varepsilon^{(n)}\right) + p\left(\mathbf{x} \in S^{(n)} \cap \overline{T_\varepsilon^{(n)}}\right) \\ &< \left|S^{(n)}\right| \max_{\mathbf{x} \in T_\varepsilon^{(n)}} p(\mathbf{x}) + p\left(\mathbf{x} \in \overline{T_\varepsilon^{(n)}}\right) \\ &< 2^{n(H-2\varepsilon)} 2^{-n(H-\varepsilon)} + \varepsilon \qquad \text{for } n > N_\varepsilon \\ &= 2^{-n\varepsilon} + \varepsilon < 2\varepsilon \qquad \text{for } n > N_0, \quad 2^{-n\varepsilon} < 2^{\log \varepsilon} = \varepsilon \end{aligned}$$

Answer: No

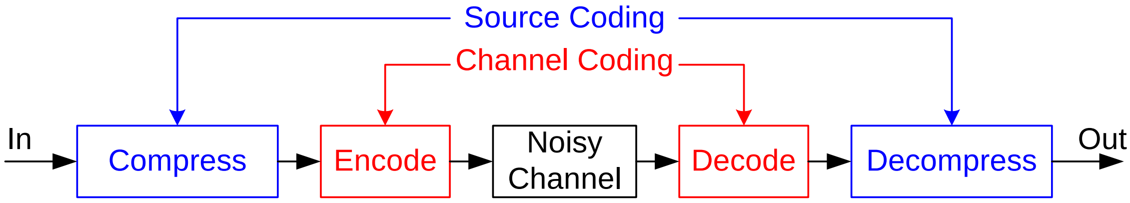
Summary

- Typical Set
  - Individual Prob $\mathbf{x} \in T_\varepsilon^{(n)} \Rightarrow \log p(\mathbf{x}) = -nH(X) \pm n\varepsilon$
  - Total Prob $p(\mathbf{x} \in T_\varepsilon^{(n)}) > 1-\varepsilon \text{ for } n > N_\varepsilon$
  - 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 than $T_\varepsilon^{(n)}$
- Asymptotic Equipartition Principle
  - Almost all event sequences are equally surprising

Lecture 9

- Source and Channel Coding
- Discrete Memoryless Channels
  - Symmetric Channels
  - Channel capacity
    - Binary Symmetric Channel
    - Binary Erasure Channel
    - Asymmetric Channel

Source and Channel Coding



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

Discrete Memoryless Channel

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

- Time-Invariant Transition-Probability Matrix

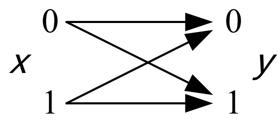
$$\left(\mathbf{Q}_{y|x}\right)_{i,j} = p\left(y = y_j \mid x = x_i\right)$$

  - Hence $\mathbf{p}_y = \mathbf{Q}_{y|x}^T \mathbf{p}_x$
  - $\mathbf{Q}$: each row sum = 1, average column sum = $|\mathcal{X}||\mathcal{Y}|^{-1}$
- Memoryless: $\mathbf{p}(y_n | x_{1:n}, y_{1:n-1}) = \mathbf{p}(y_n | x_n)$
- DMC = Discrete Memoryless Channel

Binary Channels

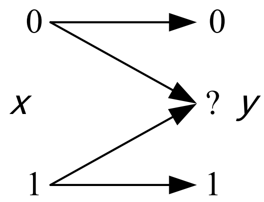
- Binary Symmetric Channel
  - $\mathcal{X}$ = [0 1], $\mathcal{Y}$ = [0 1]

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



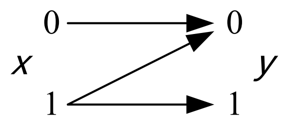
- Binary Erasure Channel
  - $\mathcal{X}$ = [0 1], $\mathcal{Y}$ = [0 ? 1]

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



- Z Channel
  - $\mathcal{X}$ = [0 1], $\mathcal{Y}$ = [0 1]

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



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

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

Weakly Symmetric Channels

Weakly Symmetric:

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

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

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

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

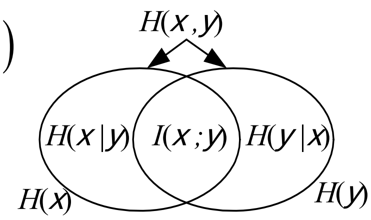
where $\mathbf{Q}_{1,:}$ is the entropy of the first (or any other) row of the $\mathbf{Q}$ matrix

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: $C = \max_{\mathbf{p}_x} I(x;y)$
  - Maximum is over all possible input distributions $\mathbf{p}_x$
  - ∃ 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:

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



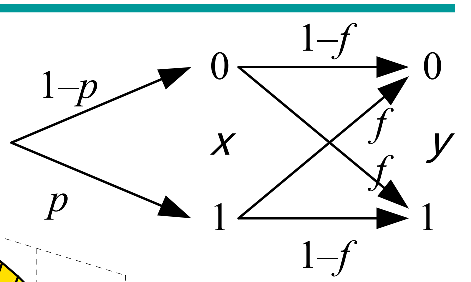
- Capacity for n uses of channel:

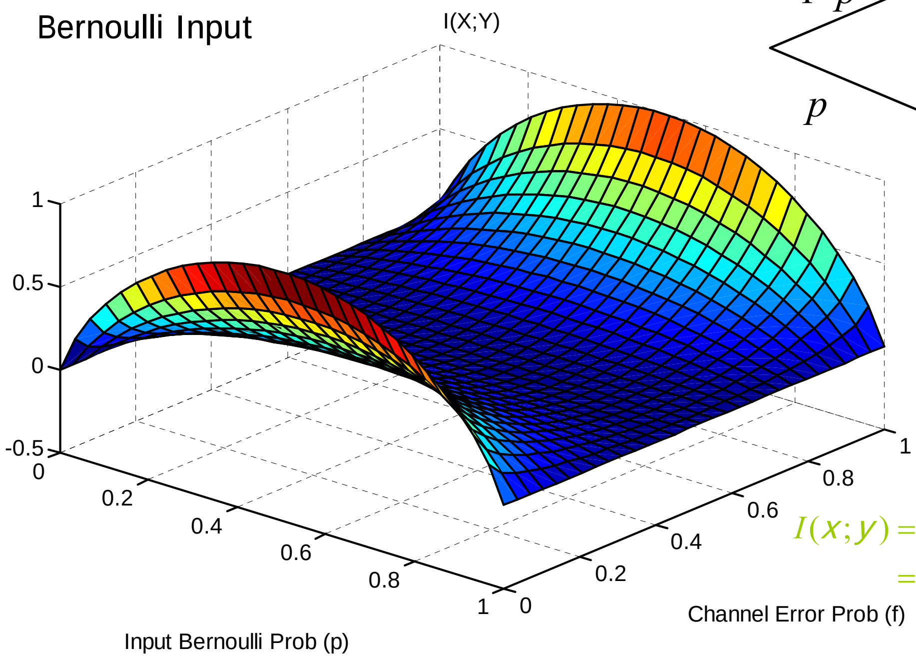
$$C^{(n)} = \frac{1}{n} \max_{\mathbf{p}_{x_{1:n}}} I(x_{1:n}; y_{1:n})$$

♦ = proved in two pages time

Mutual Information Plot

Binary Symmetric Channel
Bernoulli Input





$$I(x;y) = H(y) - H(x \mid y)$$
$$= H(f - 2pf + p) - H(f)$$

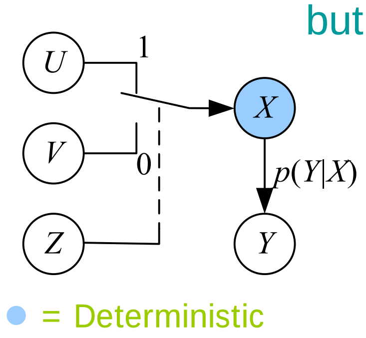
Mutual Information Concave in $\mathbf{p}_X$

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

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

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

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



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

$$\begin{aligned} I(x;y) &\ge I(x;y\,|\,z) \\ &= \lambda I(x;y\,|\,z=1) + (1-\lambda) I(x;y\,|\,z=0) \\ &= \lambda I(u;y) + (1-\lambda) I(v;y) \end{aligned}$$

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

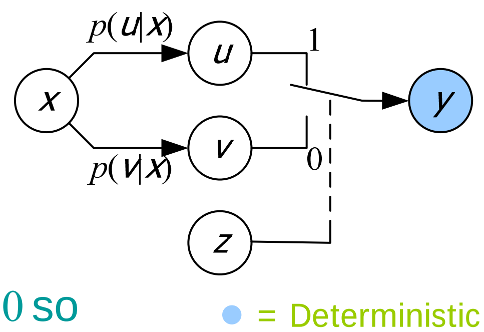
Mutual Information Convex in $\mathbf{p}_{Y|X}$

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

Proof (b) define u, v, x etc:

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

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



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

$$\begin{aligned} I(x;y) &\le I(x;y\,|\,z) \\ &= \lambda I(x;y\,|\,z=1) + (1-\lambda) I(x;y\,|\,z=0) \\ &= \lambda I(x;u) + (1-\lambda) I(x;v) \end{aligned}$$

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 x_i are independent $\Rightarrow$ y_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

Capacity of Symmetric Channel

If channel is weakly symmetric:

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

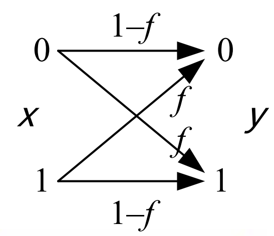
with equality iff input distribution is uniform

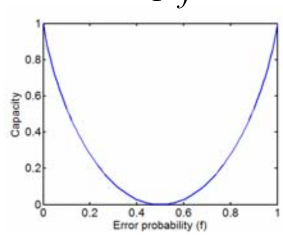
$\therefore$ Information Capacity of a WS channel is $\log|\mathcal{Y}| - H(\mathbf{Q}_{1,:})$

For a binary symmetric channel (BSC):

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

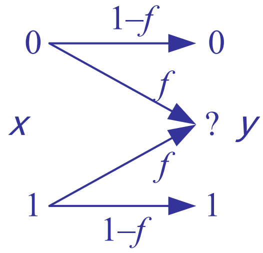
$\therefore$ Information Capacity of a BSC is $1-H(f)$





Binary Erasure Channel (BEC)

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



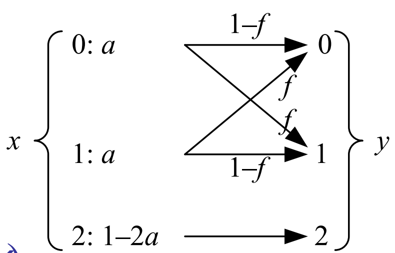
$$I(x;y) = H(x) - H(x \mid y)$$
$$= H(x) - p(y=0)\times 0 - p(y=?)H(x) - p(y=1)\times 0$$
$$= H(x) - H(x)f \qquad H(x \mid y) = 0 \text{ when } y=0 \text{ or } y=1$$
$$= (1-f)H(x)$$
$$\le 1-f$$

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

Asymmetric Channel Capacity

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



$$H(y) = -2a\log a - (1-2a)\log(1-2a)$$
$$H(y \mid x) = 2aH(f) + (1-2a)H(1) = 2aH(f)$$

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

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

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

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

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

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

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

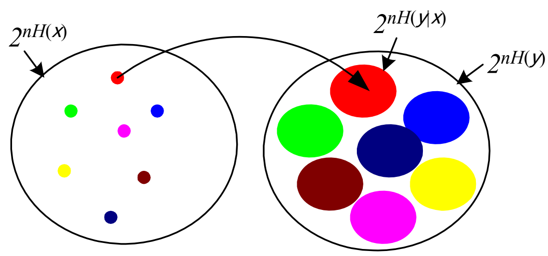
Examples: $f = 0 \Rightarrow H(f) = 0 \Rightarrow a = {}^1/_3 \Rightarrow C = \log 3 = 1.585$ bits/use
$f = ½ \Rightarrow H(f) = 1 \Rightarrow a = ¼ \Rightarrow C = \log 2 = 1$ bits/use

Lecture 10

- Jointly Typical Sets
- Joint AEP
- Channel Coding Theorem
  - Random Coding
  - Jointly typical decoding

Significance of Mutual Information

- Consider blocks of n symbols:



  - 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 $2^{n(H(y)-H(y|x))} = 2^{nI(y;x)}$

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

Jointly Typical Set

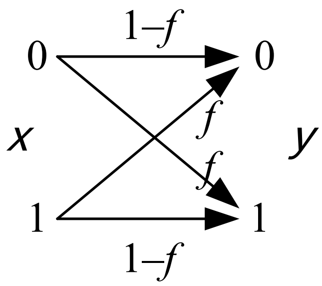
$\mathcal{XY}^n$ is the i.i.d. sequence $\{x_i, y_i\}$ for $1 \le i \le n$

- 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:

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

Jointly Typical Example

Binary Symmetric Channel



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

$\mathbf{p}_Y = (0.65 \quad 0.35)^T, \quad \mathbf{P}_{XY} = \begin{pmatrix} 0.6 & 0.15 \\ 0.05 & 0.2 \end{pmatrix}$

Jointly Typical example (for any ε):

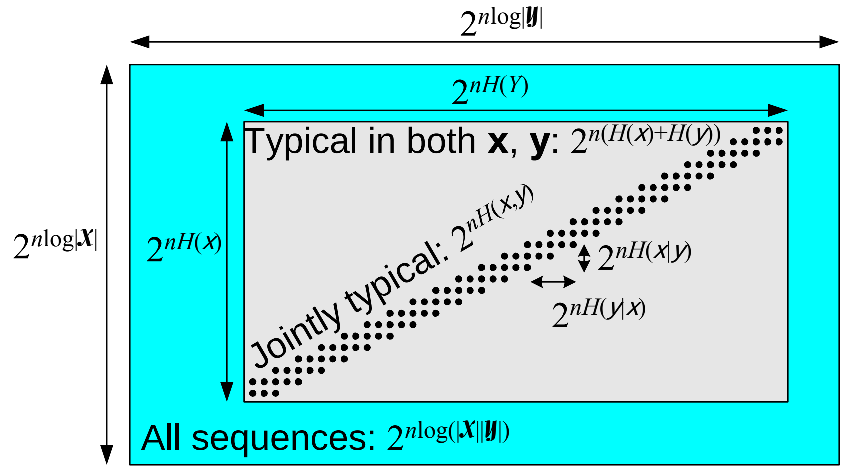
$\mathbf{x}$ = 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

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

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

Joint Typical Set Properties

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

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

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

Joint AEP

If $\mathbf{p}_{x'}=\mathbf{p}_x$ and $\mathbf{p}_{y'}=\mathbf{p}_y$ with x' and y' <u>independent:</u>

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

Proof: $|J| \times (\text{Min Prob}) \le \text{Total Prob} \le |J| \times (\text{Max Prob})$

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

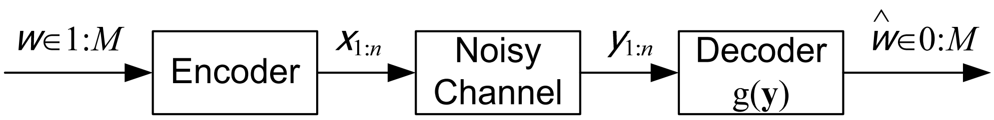
$$p\left(\mathbf{x}',\mathbf{y}'\in J_\varepsilon^{(n)}\right)\le \left|J_\varepsilon^{(n)}\right| \max_{\mathbf{x}',\mathbf{y}'\in J_\varepsilon^{(n)}} p(\mathbf{x}')p(\mathbf{y}')$$

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

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

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

Channel Codes



- Assume Discrete Memoryless Channel with known $\mathbf{Q}_{y|x}$
- An (M,n) code is
  - A fixed set of M codewords $\mathbf{x}(w)\in\mathcal{X}^n$ for $w=1:M$
  - A deterministic decoder $g(\mathbf{y})\in 1:M$
- Error probability $\lambda_w = p(g(\mathbf{y}(w))\ne w)=\sum_{\mathbf{y}\in\mathcal{Y}^n} p(\mathbf{y}\,|\,\mathbf{x}(w))\delta_{g(\mathbf{y})\ne w}$
  - 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

Achievable Code Rates

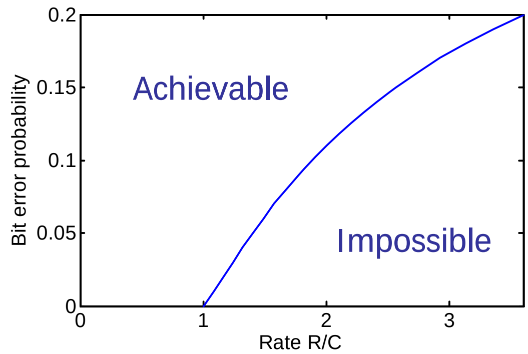
- The rate of an (M,n) code: $R=(\log M)/n$ bits/transmission
- A rate, R, is achievable if
  - $\exists$ a sequence of $\left(\lceil 2^{nR} \rceil, n\right)$ codes for $n=1,2,\ldots$
  - max prob of error $\lambda^{(n)}\to 0$ as $n\to\infty$
  - Note: we will normally write $\left(2^{nR}, n\right)$ to mean $\left(\lceil 2^{nR} \rceil, n\right)$
- The capacity of a DMC is the sup of all achievable rates
- Max error probability for a code is hard to determine
  - Shannon's idea: consider a randomly chosen code
  - show the expected average error probability is small
  - Show this means $\exists$ at least one code with small max error prob
  - Sadly it doesn't tell you how to find 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 $R<C$

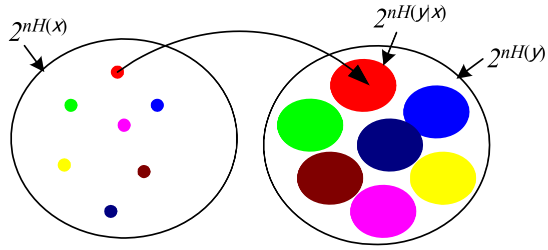


Lecture 11

- Channel Coding Theorem

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 2^{nR} random code vectors $\mathbf{x}(w)$
    - their typical output sequences are unlikely to overlap much.
  - Joint Typical Decoding: A received vector **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

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

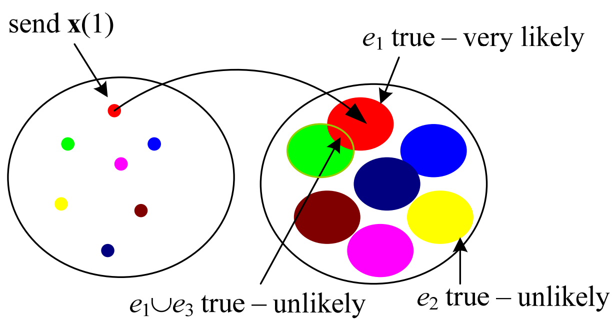
- Choose $\varepsilon \approx$ error prob, joint typicality $\Rightarrow N_\varepsilon$, choose $n>N_\varepsilon$
- Choose $\mathbf{p}_x$ so that $I(x;y)=C$, the information capacity
- Use $\mathbf{p}_x$ to choose a code $\mathcal{C}$ with random $\mathbf{x}(w)\in\mathcal{X}^n$, $w=1:2^{nR}$
  - the receiver knows this code and also the transition matrix $\mathbf{Q}$
- Assume (for now) 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 1 is the decoded message
  - we calculate error probability averaged over all $\mathcal{C}$ and all W

$$p(\mathcal{E}) = \sum_{\mathcal{C}} p(\mathcal{C})2^{-nR}\sum_{w=1}^{2^{nR}}\lambda_w(\mathcal{C}) = 2^{-nR}\sum_{w=1}^{2^{nR}}\sum_{\mathcal{C}} p(\mathcal{C})\lambda_w(\mathcal{C}) \overset{(a)}{=} \sum_{\mathcal{C}} p(\mathcal{C})\lambda_1(\mathcal{C}) = p(\mathcal{E}\,|\,w=1)$$

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

Channel Coding Principle

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



- 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

- We transmit x(1), receive y and decode using joint typicality
- We have an error if either e_1 false or e_w true for $w \ge 2$

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

p(A∪B)≤p(A)+p(B)

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

(1) Joint typicality

(2) Joint AEP

$$\le \varepsilon + 2^{-n(I(X;Y)-R-3\varepsilon)} \le 2\varepsilon \quad \text{for} \quad R < C - 3\varepsilon \text{ and } n > -\frac{\log \varepsilon}{C - R - 3\varepsilon}$$

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

♦ = proved on next page

Code Selection & Expurgation

- Since average of $P_e^{(n)}$ over all codes is $\le 2\varepsilon$ there must be at least one code for which this is true.

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

  K = num of codes

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

  Proof: Assume λ_w are in descending order

$$2\varepsilon \ge M^{-1}\sum_{w=1}^{M}\lambda_w \ge M^{-1}\sum_{w=1}^{½M}\lambda_w \ge M^{-1}\sum_{w=1}^{½M}\lambda_{½M} \ge ½\lambda_{½M}$$

$$\Rightarrow \quad \lambda_{½M} \le 4\varepsilon \quad \Rightarrow \quad \lambda_w \le 4\varepsilon \quad \forall\, w > ½M$$

$M' = ½ \times 2^{nR}$ messages in n channel uses $\Rightarrow \quad R' = n^{-1}\log M' = R - n^{-1}$

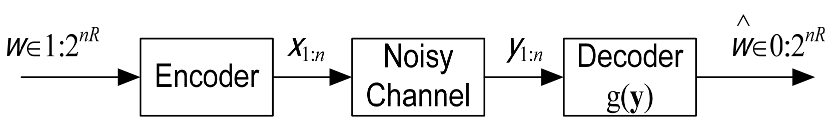
Summary of Procedure

- Given $R'<C$, choose $\varepsilon<\frac{1}{4}(C-R')$ and set $R=R'+\varepsilon \Rightarrow R<C-3\varepsilon$
- Set $n=\max\{N_\varepsilon, -(\log\varepsilon)/(C-R-3\varepsilon), \varepsilon^{-1}\}$ see (a),(b),(c) below
- Find the optimum $\mathbf{p}_X$ so that $I(x\,;y)=C$
- Choosing codewords randomly (using $\mathbf{p}_X$) and using joint typicality as the decoder, construct codes with 2^{nR} codewords (a)
- Since average of $P_e^{(n)}$ over all codes is $\le 2\varepsilon$ there must be at least one code for which this is true. Find it by exhaustive search. (b)
- Throw away the worst half of the codewords. Now the worst codeword has an error prob $\le 4\varepsilon$ with rate $R' = R - n^{-1} > R - \varepsilon$ (c)
- The resultant code transmits at a rate R' with an error probability that can be made as small as desired (but n unnecessarily large).

Note: ε determines both error probability <u>and</u> closeness to capacity

Lecture 12

- Converse of Channel Coding Theorem
  - Cannot achieve R>C
  - Minimum bit-error rate
- Capacity with feedback
  - no gain but simpler encode/decode
- Joint Source-Channel Coding
  - No point for a DMC



Converse of Coding Theorem

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

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

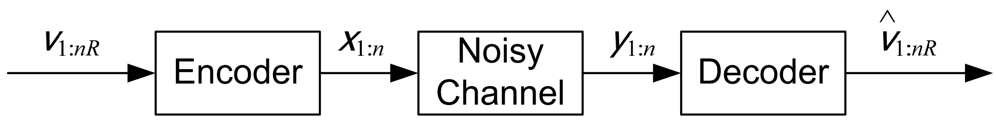
- Hence

$$\begin{aligned} nR = H(w) &= H(w \mid \mathbf{y}) + I(w;\mathbf{y}) && \text{Definition of } I \\ &\le H(w \mid \mathbf{y}) + I(\mathbf{x}(w);\mathbf{y}) && \text{Markov: } w \to \mathbf{x} \to \mathbf{y} \to \hat{w} \\ &\le 1 + nRP_e^{(n)} + I(\mathbf{x};\mathbf{y}) && \text{Fano} \\ &\le 1 + nRP_e^{(n)} + nC && n\text{-use DMC capacity} \end{aligned}$$

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

- Hence for large n, $P_e^{(n)}$ has a lower bound of $(R-C)/R$ if w equiprobable
  - If $R>C$ was achievable for small n, it could be achieved also for large n by concatenation. Hence it cannot be achieved for any n.

Minimum Bit-error Rate



Suppose

- $v_{1:nR}$ is i.i.d. bits with $H(v_i)=1$
- The bit-error rate is $P_b = E_i\{p(v_i \ne \hat{v}_i)\} \stackrel{\Delta}{=} E_i\{p(e_i)\}$

Then

$$nC \stackrel{(a)}{\ge} I(x_{1:n};y_{1:n}) \stackrel{(b)}{\ge} I(v_{1:nR};\hat{v}_{1:nR}) = H(v_{1:nR}) - H(v_{1:nR} \mid \hat{v}_{1:nR})$$

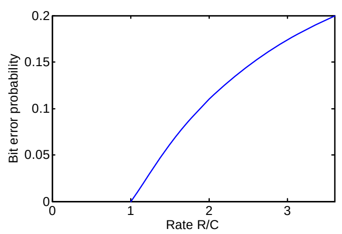
$$= nR - \sum_{i=1}^{nR} H(v_i \mid \hat{v}_{1:nR}, v_{1:i-1}) \stackrel{(c)}{\ge} nR - \sum_{i=1}^{nR} H(v_i \mid \hat{v}_i) = nR\left(1 - E_i\{H(v_i \mid \hat{v}_i)\}\right)$$

$$\stackrel{(d)}{=} nR\left(1 - E_i\{H(e_i \mid \hat{v}_i)\}\right) \stackrel{(c)}{\ge} nR\left(1 - E_i\{H(e_i)\}\right) \stackrel{(e)}{\ge} nR\left(1 - H(E_i P_{b,i})\right) = nR(1 - H(P_b))$$

Hence

$$R \le C(1 - H(P_b))^{-1}$$

$$P_b \ge H^{-1}(1 - C/R)$$



(a) n-use capacity

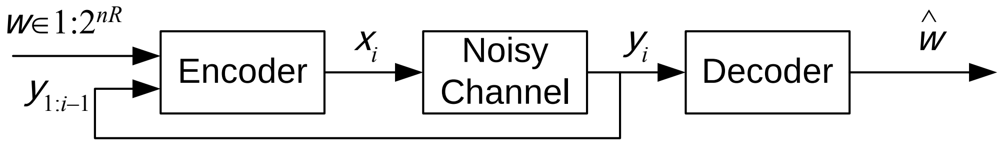
(b) Data processing theorem

(c) Conditioning reduces entropy

(d) $e_i = v_i \oplus \hat{v}_i$

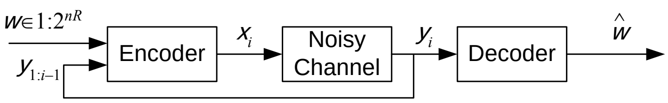
(e) Jensen: $E\,H(x) \le H(E\,x)$

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 $i=1:n$
  - 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

Capacity with Feedback



$$I(w;\mathbf{y}) = H(\mathbf{y}) - H(\mathbf{y}|w)$$
$$= H(\mathbf{y}) - \sum_{i=1}^{n} H(y_i|y_{1:i-1},w)$$
$$= H(\mathbf{y}) - \sum_{i=1}^{n} H(y_i|y_{1:i-1},w,x_i) \quad \text{since } x_i = x_i(w,y_{1:i-1})$$
$$= H(\mathbf{y}) - \sum_{i=1}^{n} H(y_i|x_i) \quad \text{since } y_i \text{ only directly depends on } x_i$$
$$\leq \sum_{i=1}^{n} H(y_i) - \sum_{i=1}^{n} H(y_i|x_i) = \sum_{i=1}^{n} I(x_i;y_i) \leq nC \quad \text{cond reduces ent}$$

Hence

$$nR = H(w) = H(w|\mathbf{y}) + I(w;\mathbf{y}) \leq 1 + nRP_e^{(n)} + nC \quad \text{Fano}$$

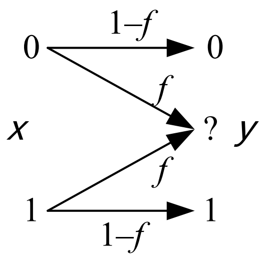
$$\Rightarrow P_e^{(n)} \geq \frac{R - C - n^{-1}}{R}$$

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

Example: BEC with feedback

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

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



- Average number of bits per transmission = $1-f$
- Capacity unchanged but encode/decode algorithm much simpler.

Joint Source-Channel Coding

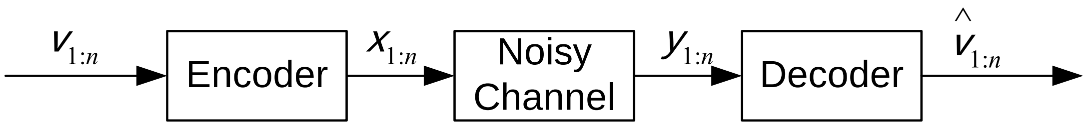
- Assume v_i satisfies AEP and $|\mathcal{V}|<\infty$
  - Examples: i.i.d.; markov; stationary ergodic
- Capacity of DMC channel is C
  - if time-varying: $C = \lim_{n\to\infty} n^{-1} I(\mathbf{x};\mathbf{y})$
- Joint Source Channel Coding Theorem: ♦
  $\exists$ codes with $P_e^{(n)} = P(\hat{v}_{1:n} \neq v_{1:n}) \xrightarrow[n\to\infty]{} 0$ iff $H(\mathcal{V}) < C$
  - errors arise from incorrect (i) encoding of V or (ii) decoding of Y
- Important result: source coding and channel coding might as well be done separately since same capacity

♦ = proved on next page

Source-Channel Proof ($\Leftarrow$)

- For $n>N_\varepsilon$ there are only $2^{n(H(\mathcal{V})+\varepsilon)}$ **v**'s in the typical set: encode using $n(H(\mathcal{V})+\varepsilon)$ bits
  - encoder error < ε
- Transmit with error prob less than ε so long as $H(\mathcal{V})+\varepsilon < C$
- Total error prob $< 2\varepsilon$

Source-Channel Proof ($\Rightarrow$)



Fano's Inequality: $H(\mathbf{v}\,|\,\hat{\mathbf{v}}) \le 1 + P_e^{(n)} n \log|\mathcal{V}|$

$$
\begin{aligned}
H(\mathcal{V}) &\le n^{-1} H(v_{1:n}) && \text{entropy rate of stationary process} \\
&= n^{-1} H(v_{1:n}\,|\,\hat{v}_{1:n}) + n^{-1} I(v_{1:n};\hat{v}_{1:n}) && \text{definition of } I \\
&\le n^{-1}\left(1 + P_e^{(n)} n \log|\mathcal{V}|\right) + n^{-1} I(x_{1:n};y_{1:n}) && \text{Fano + Data Proc Inequ} \\
&\le n^{-1} + P_e^{(n)} \log|\mathcal{V}| + C && \text{Memoryless channel}
\end{aligned}
$$

Let $n \to \infty \;\Rightarrow\; P_e^{(n)} \to 0 \;\Rightarrow\; H(\mathcal{V}) \le C$

Separation Theorem

- For a (time-varying) DMC we can design the source encoder and the channel coder separately and still get optimum performance
- Not true for
  - Correlated Channel and Source
  - Multiple access with correlated sources
    - Multiple sources transmitting to a single receiver
  - Broadcasting channels
    - one source transmitting possibly different information to multiple receivers

Lecture 13

- Continuous Random Variables
- Differential Entropy
  - can be negative
  - not 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: $y = g(x) \Leftrightarrow x = g^{-1}(y)$

$$F_Y(y) = F_X\left(g^{-1}(y)\right) \text{ or } 1 - F_X\left(g^{-1}(y)\right)$$ according to slope of $g(x)$

$$f_Y(y) = \frac{dF_Y(y)}{dy} = f_X\left(g^{-1}(y)\right)\left|\frac{dg^{-1}(y)}{dy}\right| = f_X(x)\left|\frac{dx}{dy}\right| \text{ where } x = g^{-1}(y)$$

- Examples:

Suppose $f_X(x) = 0.5$ for $x \in (0,2) \Rightarrow F_X(x) = 0.5x$

(a) $y = 4x \Rightarrow x = 0.25y \Rightarrow f_Y(y) = 0.5 \times 0.25 = 0.125$ for $y \in (0,8)$

(b) $z = x^4 \Rightarrow x = z^{1/4} \Rightarrow f_Z(z) = 0.5 \times \tfrac{1}{4} z^{-3/4} = 0.125 z^{-3/4}$ for $z \in (0,16)$

Joint Distributions Distributions

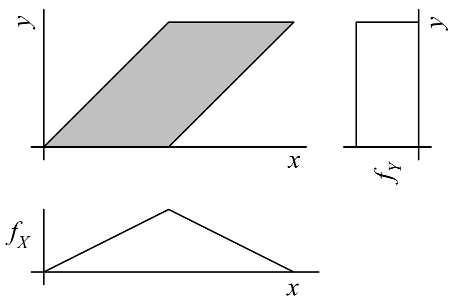
Joint pdf: $f_{X,Y}(x,y)$

Marginal pdf: $f_X(x) = \int_{-\infty}^{\infty} f_{X,Y}(x,y)dy$

Independence: $\Leftrightarrow f_{X,Y}(x,y) = f_X(x) f_Y(y)$

Conditional pdf: $f_{X|Y}(x) = \dfrac{f_{X,Y}(x,y)}{f_Y(y)}$

Example:



$f_{X,Y} = 1$ for $y \in (0,1), x \in (y, y+1)$

$f_{X|Y} = 1$ for $x \in (y, y+1)$

$f_{Y|X} = \dfrac{1}{\min(x, 1-x)}$ for $y \in \left(\max(0, x-1), \min(x,1)\right)$

Quantised Random Variables

- 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
  - $\mathcal{Y} = \{x_i\}$ and $p_y = \{f(x_i)\Delta\}$
  - Scaled,quantised version of $f(x)$ with slightly unevenly spaced x_i
- $$H(y) = -\sum f(x_i)\Delta \log\big(f(x_i)\Delta\big)$$
  $$= -\log\Delta - \sum f(x_i)\log\big(f(x_i)\big)\Delta$$
  $$\underset{\Delta\to 0}{\to} \quad -\log\Delta - \int_{-\infty}^{\infty} f(x)\log f(x)dx = -\log\Delta + h(x)$$
- Differential entropy: $h(x) = -\int_{-\infty}^{\infty} f_x(x)\log f_x(x)dx$

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

- 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 $(b-a) < 1$
- Gaussian Distribution: $X \sim N(\mu, \sigma^2)$
  - $f(x) = \left(2\pi\sigma^2\right)^{-½} \exp\left(-½(x-\mu)^2 \sigma^{-2}\right)$
  - $h(X) = -(\log e)\int_{-\infty}^{\infty} f(x) \ln f(x) dx$

$$= -(\log e)\int_{-\infty}^{\infty} f(x)\left(-½\ln(2\pi\sigma^2) - ½(x-\mu)^2\sigma^{-2}\right)$$

$$= ½(\log e)\left(\ln(2\pi\sigma^2) + \sigma^{-2}E\left((x-\mu)^2\right)\right)$$

$$= ½(\log e)\left(\ln(2\pi\sigma^2) + 1\right) = ½\log\left(2\pi e\sigma^2\right) \cong \log(4.1\sigma) \text{ bits}$$

Multivariate Gaussian

Given mean, $\mathbf{m}$, and symmetric +ve definite covariance matrix $\mathbf{K}$,

$$X_{1:n} \sim \mathbf{N}(\mathbf{m},\mathbf{K}) \quad \Leftrightarrow \quad f(\mathbf{x}) = \left|2\pi\mathbf{K}\right|^{-½} \exp\left(-½(\mathbf{x}-\mathbf{m})^T \mathbf{K}^{-1}(\mathbf{x}-\mathbf{m})\right)$$

$$h(f) = -\left(\log e\right)\int f(\mathbf{x}) \times \left(-½(\mathbf{x}-\mathbf{m})^T\mathbf{K}^{-1}(\mathbf{x}-\mathbf{m}) - ½\ln\left|2\pi\mathbf{K}\right|\right) d\mathbf{x}$$

$$= ½\log(e) \times \left(\ln\left|2\pi\mathbf{K}\right| + E\left((\mathbf{x}-\mathbf{m})^T\mathbf{K}^{-1}(\mathbf{x}-\mathbf{m})\right)\right)$$

$$= ½\log(e) \times \left(\ln\left|2\pi\mathbf{K}\right| + E\,\mathrm{tr}\left((\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^T\mathbf{K}^{-1}\right)\right)$$

$$= ½\log(e) \times \left(\ln\left|2\pi\mathbf{K}\right| + \mathrm{tr}\left(E(\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^T\mathbf{K}^{-1}\right)\right)$$

$$= ½\log(e) \times \left(\ln\left|2\pi\mathbf{K}\right| + \mathrm{tr}\left(\mathbf{K}\mathbf{K}^{-1}\right)\right) = ½\log(e) \times \left(\ln\left|2\pi\mathbf{K}\right| + n\right)$$

$$= ½\log\left(e^n\right) + ½\log\left(\left|2\pi\mathbf{K}\right|\right)$$

$$= ½\log\left(\left|2\pi e\mathbf{K}\right|\right) \quad \text{bits}$$

Other Differential Quantities

Joint Differential Entropy

$$h(\mathbf{x},\mathbf{y}) = -\iint_{x,y} f_{\mathbf{x},\mathbf{y}}(x,y)\log f_{\mathbf{x},\mathbf{y}}(x,y)dxdy = E - \log f_{\mathbf{x},\mathbf{y}}(x,y)$$

Conditional Differential Entropy

$$h(\mathbf{x}\mid\mathbf{y}) = -\iint_{x,y} f_{\mathbf{x},\mathbf{y}}(x,y)\log f_{\mathbf{x},\mathbf{y}}(x\mid y)dxdy = h(\mathbf{x},\mathbf{y}) - h(\mathbf{y})$$

Mutual Information

$$I(\mathbf{x};\mathbf{y}) = \iint_{x,y} f_{\mathbf{x},\mathbf{y}}(x,y)\log \frac{f_{\mathbf{x},\mathbf{y}}(x,y)}{f_{\mathbf{x}}(x)f_{\mathbf{y}}(y)}dxdy = h(\mathbf{x}) + h(\mathbf{y}) - h(\mathbf{x},\mathbf{y})$$

Relative Differential Entropy of two pdf's:

$$D(f\,\|\,g) = \int f(x)\log\frac{f(x)}{g(x)}dx$$

$$= -h_f(\mathbf{x}) - E_f \log g(\mathbf{x})$$

(a) must have $f(x)=0 \Rightarrow g(x)=0$

(b) continuity $\Rightarrow 0\log(0/0)=0$

Differential Entropy Properties

Chain Rules $\quad h(\mathbf{x},\mathbf{y}) = h(\mathbf{x}) + h(\mathbf{y}\mid\mathbf{x}) = h(\mathbf{y}) + h(\mathbf{x}\mid\mathbf{y})$

$$I(\mathbf{x},\mathbf{y};\mathbf{z}) = I(\mathbf{x};\mathbf{z}) + I(\mathbf{y};\mathbf{z}\mid\mathbf{x})$$

Information Inequality: $D(f\,\|\,g) \ge 0$

Proof: Define $S = \{\mathbf{x} : f(\mathbf{x}) > 0\}$

$$-D(f\,\|\,g) = \int_{\mathbf{x}\in S} f(\mathbf{x})\log\frac{g(\mathbf{x})}{f(\mathbf{x})}d\mathbf{x} = E\left(\log\frac{g(\mathbf{x})}{f(\mathbf{x})}\right)$$

$$\le \log\left(E\frac{g(\mathbf{x})}{f(\mathbf{x})}\right) = \log\left(\int_S f(\mathbf{x})\frac{g(\mathbf{x})}{f(\mathbf{x})}d\mathbf{x}\right)$$

Jensen + $\log()$ is concave

$$= \log\left(\int_S g(\mathbf{x})d\mathbf{x}\right) \le \log 1 = 0$$

all the same as for $H()$

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 $f_y(y) = f_x\left(g^{-1}(y)\right)\left|\dfrac{dg^{-1}(y)}{dy}\right|$

$$h(y) = -E\log(f_y(y)) \quad = -E\log\left(f_x(g^{-1}(y))\right) - E\log\left|\frac{dx}{dy}\right|$$

$$= -E\log\left(f_x(x)\right) - E\log\left|\frac{dx}{dy}\right| \quad = h(x) - E\log\left|\frac{dx}{dy}\right|$$

Examples:

- Translation: $y = x + a \quad \Rightarrow \quad dy/dx = 1 \quad \Rightarrow \quad h(y) = h(x)$
- Scaling: $y = cx \quad \Rightarrow \quad dy/dx = c \quad \Rightarrow \quad h(y) = h(x) - \log\left|c^{-1}\right|$
- Vector version: $y_{1:n} = \mathbf{A}x_{1:n} \quad \Rightarrow \quad h(\mathbf{y}) = h(\mathbf{x}) + \log\left|\det(\mathbf{A})\right|$

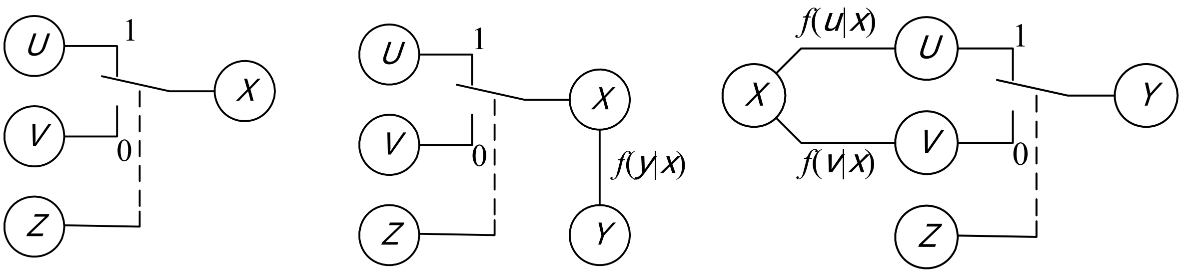
not the same as for $H()$

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: $\mathbf{p}_z = [1-\lambda,\ \lambda]^T$



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

Uniform Distribution Entropy

What distribution over the finite range (a,b) maximizes the entropy ?

Answer: A uniform distribution $u(x)=(b-a)^{-1}$

Proof:

Suppose $f(x)$ is a distribution for $x \in (a,b)$

$$0 \le D(f\,\|\,u) = -h_f(x) - E_f \log u(x) = -h_f(x) + \log(b-a)$$

$$\Rightarrow \quad h_f(x) \le \log(b-a)$$

Maximum Entropy Distribution

What zero-mean distribution maximizes the entropy on $(-\infty, \infty)^n$ for a given covariance matrix $\mathbf{K}$?

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: $0 \le D(f \| \phi) = -h_f(\mathbf{x}) - E_f \log\phi(\mathbf{x})$

$$\Rightarrow h_f(\mathbf{x}) \le -(\log e) E_f\left(-½\ln(|2\pi\mathbf{K}|) - ½\mathbf{x}^T\mathbf{K}^{-1}\mathbf{x}\right)$$

$$= ½(\log e)\left(\ln(|2\pi\mathbf{K}|) + \mathrm{tr}\left(E_f\mathbf{x}\mathbf{x}^T\mathbf{K}^{-1}\right)\right)$$

$$= ½(\log e)\left(\ln(|2\pi\mathbf{K}|) + \mathrm{tr}(\mathbf{I})\right)$$

$$= ½\log(|2\pi e\mathbf{K}|) = h_\phi(\mathbf{x})$$

$E_f\mathbf{x}\mathbf{x}^T = \mathbf{K}$

$\mathrm{tr}(\mathbf{I}) = n = \ln(e^n)$

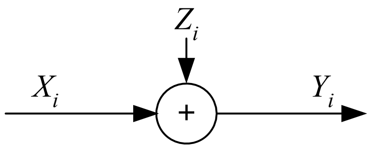
Since translation doesn't affect $h(X)$, we can assume zero-mean w.l.o.g.

Lecture 14

- Discrete-time Gaussian Channel Capacity
  - Sphere packing
- Continuous Typical Set and AEP
- Gaussian Channel Coding Theorem
- Bandlimited Gaussian Channel
  - Shannon Capacity
  - Channel Codes

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

– Define information capacity: $C = \max_{Ex^2 \le P} I(x;y)$

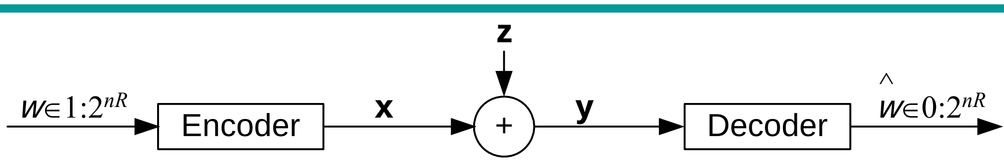
$$I(x;y) = h(y) - h(y \mid x) = h(y) - h(x+z \mid x)$$
$$\overset{(a)}{=} h(y) - h(z \mid x) \quad = h(y) - h(z)$$
$$\le ½ \log 2\pi e(P+N) - ½ \log 2\pi eN$$
$$= ½ \log\left(1 + PN^{-1}\right) \quad = ⅙\left[\tfrac{P+N}{N}\right]_{\text{dB}}$$

(a) Translation independence

Gaussian Limit with equality when $x \sim N(0,P)$

The optimal input is Gaussian & the worst noise is Gaussian

Gaussian Channel Code Rate

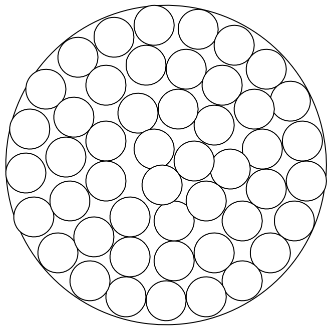


- An (M,n) code for a Gaussian Channel with power constraint is
  - A set of M codewords $\mathbf{x}(w) \in \mathcal{X}^n$ for $w=1{:}M$ with $\mathbf{x}(w)^T\mathbf{x}(w) \le nP \;\; \forall w$
  - A deterministic decoder $g(\mathbf{y}) \in 0{:}M$ where 0 denotes failure
  - Errors: codeword: λ_i max: $\lambda^{(n)}$ (over i) average: $P_e^{(n)}$ (over i)
- Rate R is achievable if $\exists$ seq of $(2^{nR},n)$ codes with $\lambda^{(n)} \xrightarrow[n\to\infty]{} 0$
- Theorem: R achievable iff $R < C = ½ \log\left(1 + PN^{-1}\right)$ ♦

♦ = proved on next pages

Sphere Packing

- Each transmitted $\mathbf{x}_i$ is received as a probabilistic cloud $\mathbf{y}_i$
  - cloud 'radius' = $\sqrt{\mathrm{Var}(\mathbf{y} \mid \mathbf{x})} = \sqrt{nN}$
- 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 $\propto r^n$
- 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 rate is $\frac{1}{2}\log(1+PN^{-1})$

Continuous AEP

Typical Set: Continuous distribution, discrete time i.i.d.

For any ε>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 $f \Leftrightarrow \{\mathbf{x} : f(\mathbf{x})>0\}$

$$f(\mathbf{x}) = \prod_{i=1}^{n} f(x_i) \text{ since } x_i \text{ are independent}$$

$$h(X) = E - \log f(X) = -n^{-1} E \log f(\mathbf{X})$$

Typical Set Properties

1. $p(\mathbf{x} \in T_\varepsilon^{(n)}) > 1-\varepsilon$ for $n > N_\varepsilon$ — Proof: WLLN
2. $(1-\varepsilon)2^{n(h(X)-\varepsilon)} \overset{n>N_\varepsilon}{\le} \mathrm{Vol}(T_\varepsilon^{(n)}) \le 2^{n(h(X)+\varepsilon)}$ — Proof: Integrate max/min prob

   where $\mathrm{Vol}(A) = \int_{\mathbf{x}\in A} d\mathbf{x}$

Continuous AEP Proof

Proof 1: By weak 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: $x_n \xrightarrow{\text{prob}} y \;\Rightarrow\; \forall\varepsilon>0, \exists N_\varepsilon$ such that $\forall n > N_\varepsilon,\quad P(|x_n - y| > \varepsilon) < \varepsilon$

Proof 2a: $1-\varepsilon \le \int_{T_\varepsilon^{(n)}} f(\mathbf{x})d\mathbf{x}$ for $n > N_\varepsilon$ — Property 1

$\le 2^{-n(h(X)-\varepsilon)}\int_{T_\varepsilon^{(n)}} d\mathbf{x} = 2^{-n(h(X)-\varepsilon)}\operatorname{Vol}(T_\varepsilon^{(n)})$ — max $f(x)$ within T

Proof 2b: $1 = \int_{S^n} f(\mathbf{x})d\mathbf{x} \ge \int_{T_\varepsilon^{(n)}} f(\mathbf{x})d\mathbf{x}$

$\ge 2^{-n(h(X)+\varepsilon)}\int_{T_\varepsilon^{(n)}} d\mathbf{x} = 2^{-n(h(X)+\varepsilon)}\operatorname{Vol}(T_\varepsilon^{(n)})$ — min $f(x)$ within T

Jointly Typical Set

Jointly Typical: x_i, y_i i.i.d from $\Re^2$ with $f_{X,Y}(x_i, y_i)$

$$J_\varepsilon^{(n)} = \Big\{\mathbf{x},\mathbf{y} \in \Re^{2n} : \left|-n^{-1}\log f_X(\mathbf{x}) - h(X)\right| < \varepsilon, \left|-n^{-1}\log f_Y(\mathbf{y}) - h(Y)\right| < \varepsilon, \left|-n^{-1}\log f_{X,Y}(\mathbf{x},\mathbf{y}) - h(X,Y)\right| < \varepsilon\Big\}$$

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: $p(\mathbf{x},\mathbf{y} \in J_\varepsilon^{(n)}) > 1-\varepsilon$ for $n > N_\varepsilon$
3. Size: $(1-\varepsilon)2^{n(h(X,Y)-\varepsilon)} \overset{n>N_\varepsilon}{\le} \operatorname{Vol}(J_\varepsilon^{(n)}) \le 2^{n(h(X,Y)+\varepsilon)}$
4. Indep $\mathbf{x}',\mathbf{y}'$: $(1-\varepsilon)2^{-n(I(X;Y)+3\varepsilon)} \overset{n>N_\varepsilon}{\le} p(\mathbf{x}',\mathbf{y}' \in J_\varepsilon^{(n)}) \le 2^{-n(I(X;Y)-3\varepsilon)}$

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 $R < C = \frac{1}{2}\log\left(1 + PN^{-1}\right)$

Proof (⇐):

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 $p\left(\mathbf{x}^T\mathbf{x} > nP\right) \to 0 \;\Rightarrow\; \le \varepsilon$ for $n > M_\varepsilon$

2. $\mathbf{y}$ not J.T. with $\mathbf{x}$ $\quad p(\mathbf{x}, \mathbf{y} \notin J_\varepsilon^{(n)}) < \varepsilon$ for $n > N_\varepsilon$

3. another $\mathbf{x}$ J.T. with $\mathbf{y}$ $\quad \sum_{j=2}^{2^{nR}} p(\mathbf{x}_j, \mathbf{y}_i \in J_\varepsilon^{(n)}) \le (2^{nR} - 1) \times 2^{-n(I(X;Y) - 3\varepsilon)}$

Total Err $P_\varepsilon^{(n)} \le \varepsilon + \varepsilon + 2^{-n\left(I(X;Y) - R - 3\varepsilon\right)} \le 3\varepsilon$ for large n if $R < I(X;Y) - 3\varepsilon$

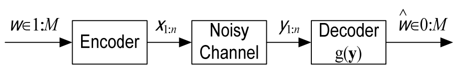
Expurgation: Remove half of codebook*: $\lambda^{(n)} < 6\varepsilon$ now max error

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 (⇒): Assume $P_\varepsilon^{(n)} \to 0$ 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

$$\le \sum_{i=1}^{n} I(x_i; y_i) + 1 + nRP_\varepsilon^{(n)}$$ Z i.i.d + Fano, $|W| = 2^{nR}$

$$\le \sum_{i=1}^{n} \tfrac{1}{2}\log\left(1 + PN^{-1}\right) + 1 + nRP_\varepsilon^{(n)}$$ max Information Capacity

$$R \le \tfrac{1}{2}\log\left(1 + PN^{-1}\right) + n^{-1} + RP_\varepsilon^{(n)} \to \tfrac{1}{2}\log\left(1 + PN^{-1}\right)$$

Bandlimited Channel

- Channel bandlimited to $f \in (-W,W)$ and signal duration T
- Nyquist: Signal is completely defined by $2WT$ samples
- Can represent as a $n=2WT$-dimensional vector space with prolate spheroidal functions as an orthonormal basis
  - 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 $\le PT$
    - Energy constraint per coefficient: $n^{-1}\mathbf{x}^T\mathbf{x}<PT/2WT=\frac{1}{2}W^{-1}P$
- Capacity:

$$C = \tfrac{1}{2}\log\left(1+\tfrac{1}{2}W^{-1}P(\tfrac{1}{2}N_0)^{-1}\right)\times 2W$$
$$= W\log\left(1+N_0^{-1}W^{-1}P\right) \text{ bits/second}$$

Compare discrete time version: $\frac{1}{2}\log(1+PN^{-1})$ bits per channel use

Shannon Capacity

Bandwidth $=W$ Hz, Signal variance $=\frac{1}{2}W^{-1}P$, Noise variance $=\frac{1}{2}N_0$

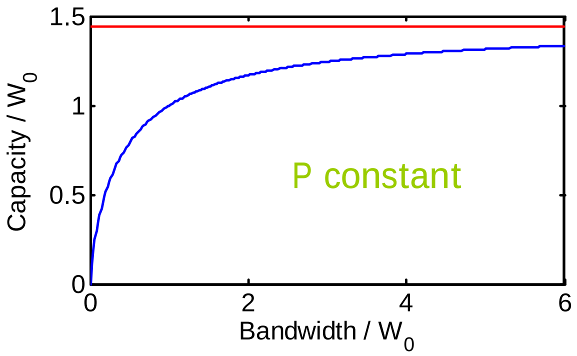
Signal Power $=P$, Noise power $=N_0W$, Min bit energy $=E_b=PC^{-1}$

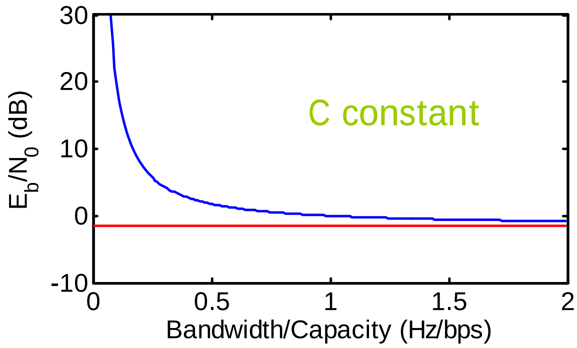
Capacity $=C=W\log\left(1+PN_0^{-1}W^{-1}\right)$ bits per second

Define: $W_0 = PN_0^{-1} \Rightarrow C/W_0 = (W/W_0)\log\left(1+(W/W_0)^{-1}\right) \xrightarrow[W\to\infty]{} \log e$

$$\Rightarrow C^{-1}W_0 = E_bN_0^{-1} \xrightarrow[W\to\infty]{} \ln 2 = -1.6 \text{ dB}$$

- For fixed power, high bandwidth is better – Ultra wideband





Practical Channel Codes

Code Classification:

- Very good: arbitrarily small error up to the capacity
- Good: arbitrarily small error up to less than capacity
- Bad: arbitrarily small error only at zero rate (or never)

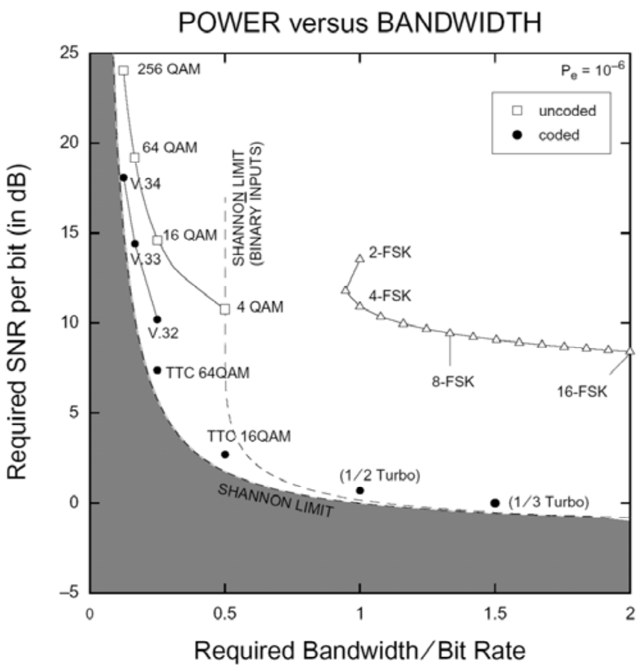
Coding Theorem: Nearly all codes are very good

- but nearly all codes need encode/decode computation $\propto 2^n$

Practical Good Codes:

- Practical: Computation & memory $\propto n^k$ for some k
- Convolution Codes: convolve bit stream with a filter
    - Concatenation, Interleaving, turbo codes (1993)
- Block codes: encode a block at a time
    - Hamming, BCH, Reed-Solomon, LD parity check (1995)

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)

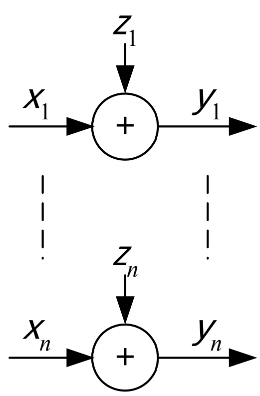
Diagram from "An overview of Communications" by John R Barry, TTC = Japanese Technology Telecommunications Council

Lecture 15

- Parallel Gaussian Channels
  - Waterfilling
- Gaussian Channel with Feedback

Parallel Gaussian Channels

- n gaussian channels (or one channel n times)
  - e.g. digital audio, digital TV, Broadband ADSL
- Noise is independent $z_i \sim N(0,N_i)$
- Average Power constraint $E\mathbf{x}^T\mathbf{x} \le P$
- Information Capacity: $C = \max_{f(\mathbf{x}):E_f\mathbf{x}^T\mathbf{x} \le P} I(\mathbf{x};\mathbf{y})$
- $R<C \Leftrightarrow R$ achievable
  - proof as before
- What is the optimal $f(\mathbf{x})$?



Parallel Gaussian: Max Capacity

Need to find $f(\mathbf{x})$: $C = \max_{f(\mathbf{x}): E_f \mathbf{x}^T\mathbf{x} \le P} I(\mathbf{x};\mathbf{y})$

$$I(\mathbf{x};\mathbf{y}) = h(\mathbf{y}) - h(\mathbf{y}\,|\,\mathbf{x}) = h(\mathbf{y}) - h(\mathbf{z}\,|\,\mathbf{x})$$ Translation invariance

$$= h(\mathbf{y}) - h(\mathbf{z}) = h(\mathbf{y}) - \sum_{i=1}^{n} h(z_i)$$ $\mathbf{x},\mathbf{z}$ indep; Z_i indep

$$\overset{(a)}{\le} \sum_{i=1}^{n} \left(h(y_i) - h(z_i)\right) \overset{(b)}{\le} \sum_{i=1}^{n} \tfrac{1}{2}\log\left(1 + P_i N_i^{-1}\right)$$ (a) indep bound; (b) capacity limit

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 $\sum_{i=1}^{n} \tfrac{1}{2}\log\left(1 + P_i N_i^{-1}\right)$

Parallel Gaussian: Optimal Powers

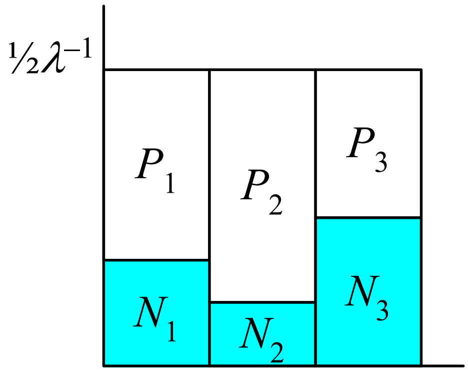
We need to find the P_i that maximise $\log(e)\sum_{i=1}^{n} \tfrac{1}{2}\ln\left(1 + P_i N_i^{-1}\right)$

- subject to power constraint $\sum_{i=1}^{n} P_i = P$
- use Lagrange multiplier

$$J = \sum_{i=1}^{n} \tfrac{1}{2}\ln\left(1 + P_i N_i^{-1}\right) - \lambda \sum_{i=1}^{n} P_i$$

$$\frac{\partial J}{\partial P_i} = \tfrac{1}{2}\left(P_i + N_i\right)^{-1} - \lambda = 0 \quad \Rightarrow \quad P_i + N_i = \tfrac{1}{2}\lambda^{-1}$$

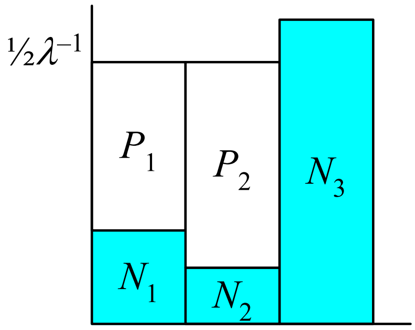
Also $\sum_{i=1}^{n} P_i = P \quad \Rightarrow \quad \lambda = \tfrac{1}{2}n\left(P + \sum_{i=1}^{n} N_i\right)^{-1}$



Water Filling: put most power into least noisy channels to make equal power + noise in each channel

Very Noisy Channels

- Must have $P_j \geq 0 \ \forall i$
- If $\frac{1}{2}\lambda^{-1} < N_j$ then set P_j=0 and recalculate λ



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

$$\nabla J(\mathbf{x}_0) = 0, \quad \mathbf{Ax} + \mathbf{b} = \mathbf{0}, \quad g_i(\mathbf{x}_0) \geq 0, \quad \mu_i \geq 0, \quad \mu_i g_i(\mathbf{x}_0) = 0$$

Correlated Noise

- Suppose $\mathbf{y} = \mathbf{x} + \mathbf{z}$ where $E\,\mathbf{zz}^T = \mathbf{K_z}$ and $E\,\mathbf{xx}^T = \mathbf{K_x}$
- We want to find $\mathbf{K_x}$ to maximize capacity subject to power constraint: $E\sum_{i=1}^{n} x_i^2 \leq nP \quad \Leftrightarrow \quad \mathrm{tr}(\mathbf{K_x}) \leq nP$
  - Find noise eigenvectors: $\mathbf{K}_Z = \mathbf{QDQ}^T$ with $\mathbf{QQ}^T = \mathbf{I}$
  - Now $\mathbf{Q}^T\mathbf{y} = \mathbf{Q}^T\mathbf{x} + \mathbf{Q}^T\mathbf{z} = \mathbf{Q}^T\mathbf{x} + \mathbf{w}$ where $\mathrm{E}\,\mathbf{ww}^T = \mathrm{E}\,\mathbf{Q}^T\mathbf{zz}^T\mathbf{Q} = \mathrm{E}\,\mathbf{Q}^T\mathbf{K}_Z\mathbf{Q} = \mathbf{D}$ is diagonal
    - $\Rightarrow W_i$ are now independent
  - Power constraint is unchanged $\mathrm{tr}(\mathbf{Q}^T\mathbf{K}_X\mathbf{Q}) = \mathrm{tr}(\mathbf{K}_X\mathbf{QQ}^T) = \mathrm{tr}(\mathbf{K}_X)$
  - Choose $\mathbf{Q}^T\mathbf{K}_X\mathbf{Q} = L\mathbf{I} - \mathbf{D}$ where $L = P + n^{-1}\,\mathrm{tr}(\mathbf{D})$

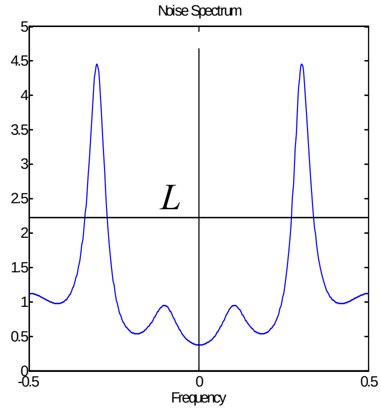
$$\Rightarrow \mathbf{K}_X = \mathbf{Q}(L\mathbf{I} - \mathbf{D})\mathbf{Q}^T$$

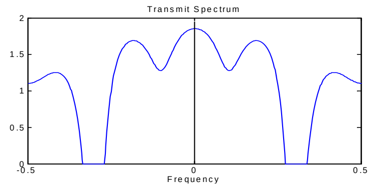
Power Spectrum Water Filling

- If **z** is from a stationary process then $\mathrm{diag}(\mathbf{D}) \underset{n\to\infty}{\to}$ power spectrum
  - To achieve capacity use waterfilling on noise power spectrum

$$P = \int_{-W}^{W} \max\left(L - N(f), 0\right) df$$

$$C = \int_{-W}^{W} \tfrac{1}{2}\log\left(1 + \frac{\max\left(L - N(f), 0\right)}{N(f)}\right) df$$

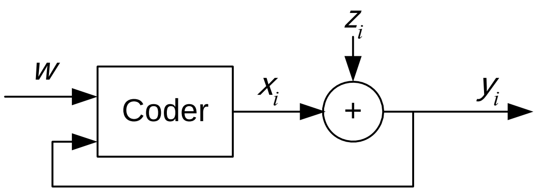




Gaussian Channel + Feedback

Does Feedback add capacity ?

- White noise – No
- Coloured noise – Not much



$x_i = x_i(w, y_{1:i-1})$

$$I(w;\mathbf{y}) = h(\mathbf{y}) - h(\mathbf{y}\,|\,w) = h(\mathbf{y}) - \sum_{i=1}^{n} h(y_i\,|\,w, y_{1:i-1})$$ Chain rule

$$= h(\mathbf{y}) - \sum_{i=1}^{n} h(y_i\,|\,w, y_{1:i-1}, x_{1:i}, z_{1:i-1})$$ $x_i = x_i(w, y_{1:i-1})$, $\mathbf{z} = \mathbf{y} - \mathbf{x}$

$$= h(\mathbf{y}) - \sum_{i=1}^{n} h(z_i\,|\,w, y_{1:i-1}, x_{1:i}, z_{1:i-1})$$ $\mathbf{z} = \mathbf{y} - \mathbf{x}$ and translation invariance

$$= h(\mathbf{y}) - \sum_{i=1}^{n} h(z_i\,|\,z_{1:i-1})$$ z_i depends only on $z_{1:i-1}$

$$= h(\mathbf{y}) - h(\mathbf{z})$$ Chain rule, $h(\mathbf{z}) = \tfrac{1}{2}\log\left|2\pi e \mathbf{K}_\mathbf{z}\right|$ bits

$$= \tfrac{1}{2}\log\frac{|\mathbf{K}_\mathbf{y}|}{|\mathbf{K}_\mathbf{z}|}$$

$\Rightarrow$ 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

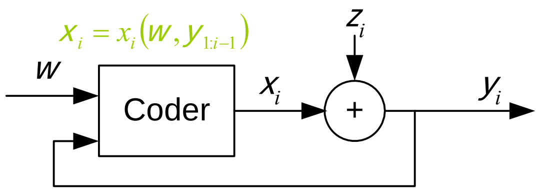
Gaussian Feedback Coder

$\mathbf{x}$ and $\mathbf{z}$ jointly gaussian $\Rightarrow$

$$\mathbf{x} = \mathbf{Bz} + \mathbf{v}(w)$$

where $\mathbf{v}$ is indep of $\mathbf{z}$ and

$\mathbf{B}$ is strictly lower triangular since x_i indep of z_j for $j>i$.



$$\mathbf{y} = \mathbf{x} + \mathbf{z} = (\mathbf{B}+\mathbf{I})\mathbf{z} + \mathbf{v}$$

$$\mathbf{K_y} = E\mathbf{yy}^T = E\left((\mathbf{B}+\mathbf{I})\mathbf{zz}^T(\mathbf{B}+\mathbf{I})^T + \mathbf{vv}^T\right) = (\mathbf{B}+\mathbf{I})\mathbf{K_z}(\mathbf{B}+\mathbf{I})^T + \mathbf{K_v}$$

$$\mathbf{K_x} = E\mathbf{xx}^T = E\left(\mathbf{Bzz}^T\mathbf{B}^T + \mathbf{vv}^T\right) = \mathbf{BK_zB}^T + \mathbf{K_v}$$

Capacity: $$C_{n,FB} = \max_{\mathbf{K_v},\mathbf{B}} \tfrac{1}{2} n^{-1} \frac{|\mathbf{K_y}|}{|\mathbf{K_z}|} = \max_{\mathbf{K_v},\mathbf{B}} \tfrac{1}{2} n^{-1} \log \frac{\left|(\mathbf{B}+\mathbf{I})\mathbf{K_z}(\mathbf{B}+\mathbf{I})^T + \mathbf{K_v}\right|}{|\mathbf{K_z}|}$$

subject to $$\mathbf{K_x} = \operatorname{tr}\left(\mathbf{BK_zB}^T + \mathbf{K_v}\right) \le nP$$

hard to solve ☹

Gaussian Feedback: Toy Example

$$n = 2, \quad P = 2, \quad \mathbf{K_z} = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 & 0 \\ b & 0 \end{pmatrix}$$

$$\mathbf{x} = \mathbf{Bz} + \mathbf{v}$$

Goal: Maximize (w.r.t. $\mathbf{K_v}$ and b)

$$\det(\mathbf{K_Y}) = \det\left((\mathbf{B}+\mathbf{I})\mathbf{K_z}(\mathbf{B}+\mathbf{I})^T + \mathbf{K_v}\right)$$

Subject to:

$\mathbf{K_v}$ must be positive definite

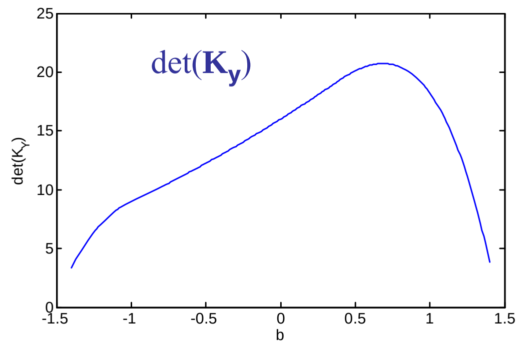
Power constraint : $\operatorname{tr}\left(\mathbf{BK_zB}^T + \mathbf{K_v}\right) \le 4$

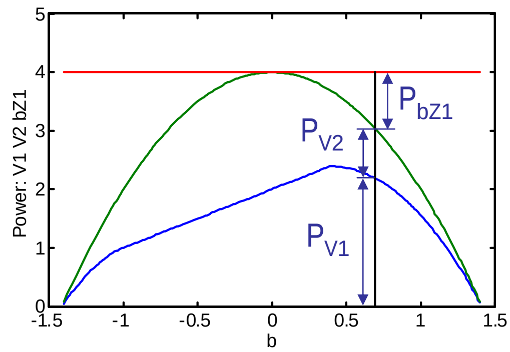
Solution (via numerically search):

b=0: $\det(\mathbf{K_y})=16$ C=0.604 bits

b=0.69: $\det(\mathbf{K_y})=20.7$ C=0.697 bits

Feedback increases C by 16%





$$P_{x1} = P_{v1} \qquad P_{x2} = P_{v2} + P_{bz1}$$

Max Benefit of Feedback: Lemmas

Lemma 1: $\mathbf{K}_{\mathbf{x}+\mathbf{z}}+\mathbf{K}_{\mathbf{x}-\mathbf{z}} = 2(\mathbf{K}_{\mathbf{x}}+\mathbf{K}_{\mathbf{z}})$

$$
\begin{aligned}
\mathbf{K}_{\mathbf{x}+\mathbf{z}}+\mathbf{K}_{\mathbf{x}-\mathbf{z}} &= E(\mathbf{x}+\mathbf{z})(\mathbf{x}+\mathbf{z})^T + E(\mathbf{x}-\mathbf{z})(\mathbf{x}-\mathbf{z})^T \\
&= E\left(\mathbf{x}\mathbf{x}^T+\mathbf{x}\mathbf{z}^T+\mathbf{z}\mathbf{x}^T+\mathbf{z}\mathbf{z}^T \quad + \quad \mathbf{x}\mathbf{x}^T-\mathbf{x}\mathbf{z}^T-\mathbf{z}\mathbf{x}^T+\mathbf{z}\mathbf{z}^T\right) \\
&= E\left(2\mathbf{x}\mathbf{x}^T+2\mathbf{z}\mathbf{z}^T\right) = 2\left(\mathbf{K}_{\mathbf{x}}+\mathbf{K}_{\mathbf{z}}\right)
\end{aligned}
$$

Lemma 2: If $\mathbf{F}$,$\mathbf{G}$ are +ve definite then $\det(\mathbf{F}+\mathbf{G}) \ge \det(\mathbf{F})$

Consider two indep random vectors $\mathbf{f} \sim N(0,\mathbf{F})$, $\mathbf{g} \sim N(0,\mathbf{G})$

$$
\begin{aligned}
\tfrac{1}{2}\log\left((2\pi e)^n \det(\mathbf{F}+\mathbf{G})\right) &= h(\mathbf{f}+\mathbf{g}) \\
&\ge h(\mathbf{f}+\mathbf{g} \mid \mathbf{g}) = h(\mathbf{f} \mid \mathbf{g}) \\
&= h(\mathbf{f}) = \tfrac{1}{2}\log\left((2\pi e)^n \det(\mathbf{F})\right)
\end{aligned}
$$

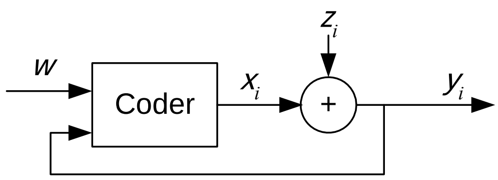
Conditioning reduces $h()$

Translation invariance

$\mathbf{f}$,$\mathbf{g}$ independent

Hence: $\det(2(\mathbf{K}_{\mathbf{x}}+\mathbf{K}_{\mathbf{z}})) = \det(\mathbf{K}_{\mathbf{x}+\mathbf{z}}+\mathbf{K}_{\mathbf{x}-\mathbf{z}}) \ge \det(\mathbf{K}_{\mathbf{x}+\mathbf{z}}) = \det(\mathbf{K}_{\mathbf{y}})$

Maximum Benefit of Feedback



$$
\begin{aligned}
C_{n,FB} &\le \max_{\operatorname{tr}(\mathbf{K}_{\mathbf{x}})\le nP} \tfrac{1}{2}n^{-1}\log\frac{\det(\mathbf{K}_{\mathbf{y}})}{\det(\mathbf{K}_{\mathbf{z}})} \\
&\le \max_{\operatorname{tr}(\mathbf{K}_{\mathbf{x}})\le nP} \tfrac{1}{2}n^{-1}\log\frac{\det\left(2\left(\mathbf{K}_{\mathbf{x}}+\mathbf{K}_{\mathbf{z}}\right)\right)}{\det(\mathbf{K}_{\mathbf{z}})} \\
&= \max_{\operatorname{tr}(\mathbf{K}_{\mathbf{x}})\le nP} \tfrac{1}{2}n^{-1}\log\frac{2^n\det\left(\mathbf{K}_{\mathbf{x}}+\mathbf{K}_{\mathbf{z}}\right)}{\det(\mathbf{K}_{\mathbf{z}})} \\
&= \tfrac{1}{2} + \max_{\operatorname{tr}(\mathbf{K}_{\mathbf{x}})\le nP} \tfrac{1}{2}n^{-1}\log\frac{\det\left(\mathbf{K}_{\mathbf{x}}+\mathbf{K}_{\mathbf{z}}\right)}{\det(\mathbf{K}_{\mathbf{z}})} = \tfrac{1}{2} + C_n
\end{aligned}
$$

bits/transmission

no constraint on $\mathbf{B} \Rightarrow \le$

Lemmas 1 & 2:

det(2(**Kx**+**Kz**)) ≥ det(**Ky**)

$\det(k\mathbf{A}) = k^n\det(\mathbf{A})$

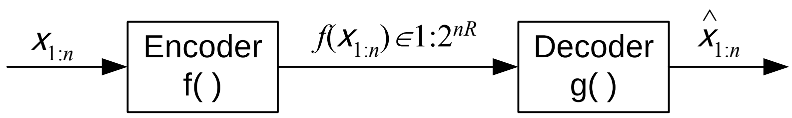
Ky = **Kx**+**Kz** if no feedback

Having feedback adds at most ½ bit per transmission

Lecture 16

- Lossy Coding
- Rate Distortion
  - Bernoulli Scurce
  - Gaussian Source
- Channel/Source Coding Duality

Lossy 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: $d(\mathbf{x},\hat{\mathbf{x}}) \stackrel{\Delta}{=} n^{-1}\sum_{i=1}^{n} d(x_i,\hat{x}_i)$

Distortion of Code $f_n(),g_n()$: $D = E_{\mathbf{x}\in\mathcal{X}^n} d(\mathbf{x},\hat{\mathbf{x}}) = E\; d(\mathbf{x}, g(f(\mathbf{x})))$

Rate distortion pair (R,D) is achievable for source $\mathcal{X}$ if

$\exists$ a sequence $f_n()$ and $g_n()$ such that $\lim_{n\to\infty} E_{\mathbf{x}\in\mathcal{X}^n} d(\mathbf{x}, g_n(f_n(\mathbf{x}))) \leq D$

Rate Distortion Function

Rate Distortion function for $\{x_i\}$ where $p(\mathbf{x})$ is known is

$R(D) = \inf R$ such that (R,D) is achievable

Theorem: $R(D) = \min I(x;\hat{x})$ over all $p(x,\hat{x})$ such that:

(a) $p(x)$ is correct

(b) $E_{x,\hat{x}} d(x,\hat{x}) \le D$

– this expression is the Information Rate Distortion function for X

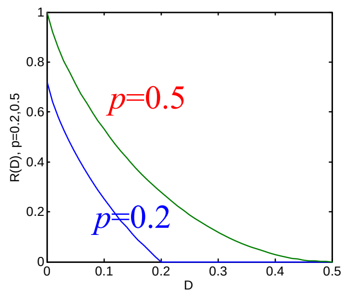
We will prove this next lecture

If $D=0$ then we have $R(D)=I(x\,;\,x)=H(x)$

$R(D)$ bound for Bernoulli Source

Bernoulli: $\mathcal{X} = [0,1]$, $\mathbf{p}_X = [1-p\,,\,p]$ assume $p \le ½$

- Hamming Distance: $d(x,\hat{x}) = x \oplus \hat{x}$
- If $D \ge p$, $R(D)=0$ since we can set $g(\,)\equiv 0$
- For $D < p \le ½$, if $E\,d(x,\hat{x}) \le D$ then



$$
\begin{aligned}
I(x;\hat{x}) &= H(x) - H(x \mid \hat{x}) \\
&= H(p) - H(x \oplus \hat{x} \mid \hat{x}) \\
&\ge H(p) - H(x \oplus \hat{x}) \\
&\ge H(p) - H(D)
\end{aligned}
$$

$\oplus$ is one-to-one

Conditioning reduces entropy

$p(x \oplus \hat{x}) \le D$ and so for $D \le ½$

$H(x \oplus \hat{x}) \le H(D)$ as $H(p)$ monotonic

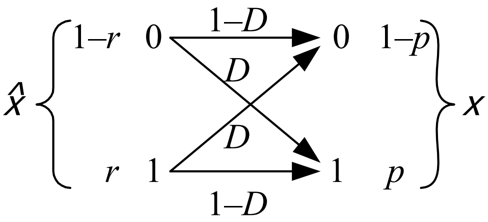
Hence $R(D) \ge H(p) - H(D)$

$R(D)$ 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:



$$r(1-D) + (1-r)D = p$$

$$\Rightarrow r = (p - D)(1 - 2D)^{-1}$$

Now $I(x;\hat{x}) = H(x) - H(x \mid \hat{x}) = H(p) - H(D)$

and $p(x \neq \hat{x}) = D$

Hence $R(D) = H(p) - H(D)$

$R(D)$ 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 $E(x - \hat{x})^2 \leq D$

$$I(x;\hat{x}) = h(x) - h(x \mid \hat{x})$$

$$= ½\log 2\pi e\sigma^2 - h(x - \hat{x} \mid \hat{x})$$ Translation Invariance

$$\geq ½\log 2\pi e\sigma^2 - h(x - \hat{x})$$ Conditioning reduces entropy

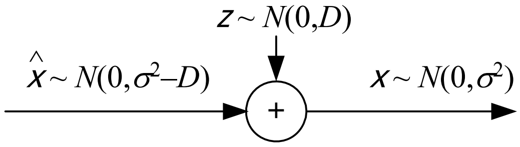
$$\geq ½\log 2\pi e\sigma^2 - ½\log\left(2\pi e \operatorname{Var}(x - \hat{x})\right)$$ Gauss maximizes entropy for given covariance

$$\geq ½\log 2\pi e\sigma^2 - ½\log 2\pi e D$$ require $\operatorname{Var}(x - \hat{x}) \leq E(x - \hat{x})^2 \leq D$

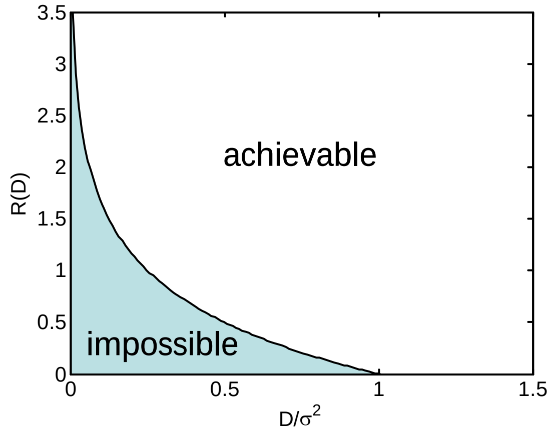
$$I(x;\hat{x}) \geq \max\left(½\log\frac{\sigma^2}{D}, 0\right)$$ $I(x;y)$ always positive

$R(D)$ 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 $D<\sigma^2$



$$I(x;\hat{x}) = h(x) - h(x \mid \hat{x})$$
$$= \tfrac{1}{2}\log 2\pi e\sigma^2 - h(x-\hat{x} \mid \hat{x})$$
$$= \tfrac{1}{2}\log 2\pi e\sigma^2 - h(z \mid \hat{x})$$
$$= \tfrac{1}{2}\log \frac{\sigma^2}{D}$$
$$\Rightarrow \quad D(R) = \left(\frac{\sigma}{2^R}\right)^2$$

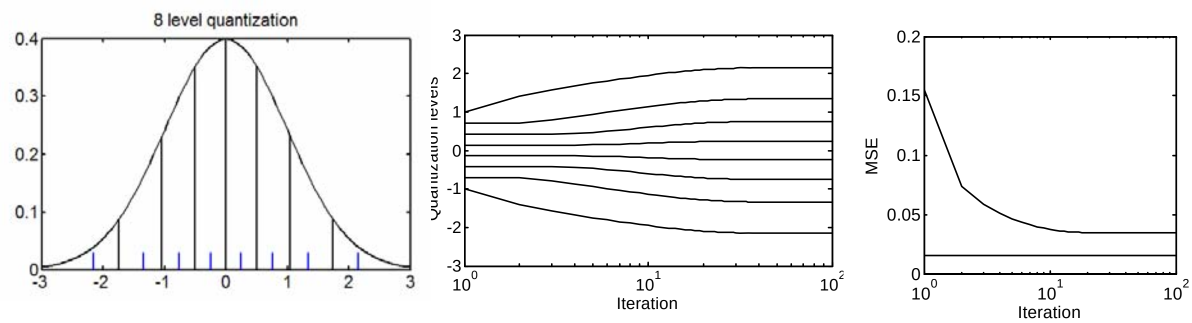


Lloyd Algorithm

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.



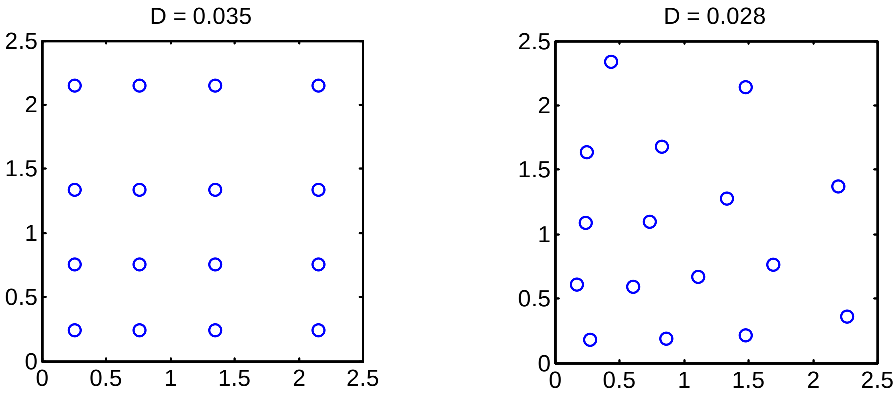
Solid lines are bin boundaries. Initial levels uniform in $[-1,+1]$.

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



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 $d(\mathbf{x},\hat{\mathbf{x}}) = n^{-1}(\mathbf{x}-\hat{\mathbf{x}})^T(\mathbf{x}-\hat{\mathbf{x}}) \le D$

$$I(x_{1:n};\hat{x}_{1:n}) \ge \sum_{i=1}^{n} I(x_i;\hat{x}_i)$$

Mut Info Independence Bound for independent x_i

$$\ge \sum_{i=1}^{n} R(D_i) = \sum_{i=1}^{n} \max\left(\tfrac{1}{2}\log\frac{\sigma_i^2}{D_i},0\right)$$

$R(D)$ for individual Gaussian

We must find the D_i that minimize $\sum_{i=1}^{n} \max\left(\tfrac{1}{2}\log\frac{\sigma_i^2}{D_i},0\right)$

Reverse Waterfilling

Minimize $\sum_{i=1}^{n} \max\left(\tfrac{1}{2}\log\frac{\sigma_i^2}{D_i}, 0\right)$ subject to $\sum_{i=1}^{n} D_i \le nD$

$R_i = \tfrac{1}{2}\log\frac{\sigma_i^2}{D}$

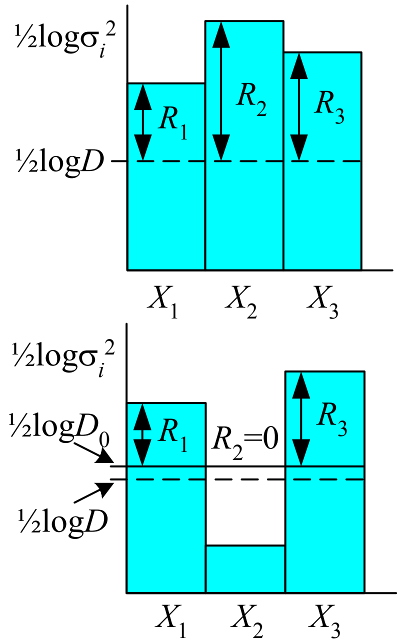
Use a lagrange multiplier:

$$J = \sum_{i=1}^{n} \tfrac{1}{2}\log\frac{\sigma_i^2}{D_i} + \lambda\sum_{i=1}^{n} D_i$$

$$\frac{\partial J}{\partial D_i} = -\tfrac{1}{2}D_i^{-1} + \lambda = 0 \quad\Rightarrow\quad D_i = \tfrac{1}{2}\lambda^{-1} = D_0$$

$$\sum_{i=1}^{n} D_i = nD_0 = nD \quad\Rightarrow\quad D_0 = D$$

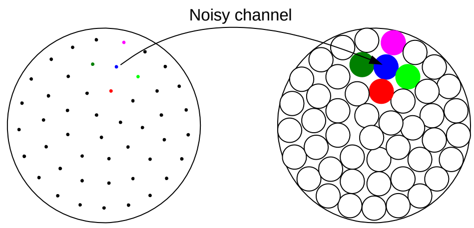
Choose R_i for equal distortion



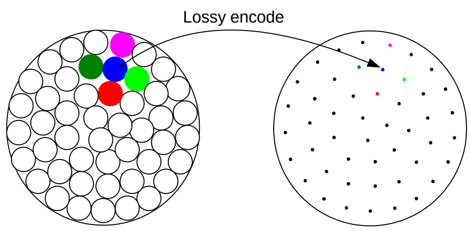
- If $\sigma_i^2 < D$ then set R_i=0 and increase D_0 to maintain the average distortion equal to D
- If x_i are correlated then reverse waterfill the eigenvectors of the correlation matrix

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 fill the sphere.
- Source Coding
  - Find regions that cover the sphere
  - Region size = allowed distortion
  - The minimum number (lowest rate) is when they just don't overlap

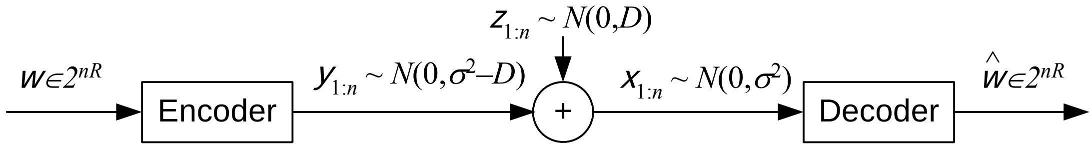


Sphere Packing

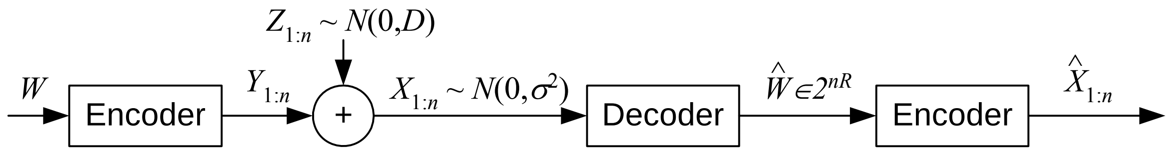


Sphere Covering

Channel Decoder as Source Coder



- 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 $E(x_i - y_i)^2 = D$
- Reverse the roles of encoder and decoder. Since $p\left(w \neq \hat{w}\right) < \varepsilon$, also $p(\hat{x} \neq y) < \varepsilon$ 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

High Dimensional Space

In n dimensions

- "Vol" of unit hypercube: 1
- "Vol" of unit-diameter hypersphere:

$$V_n = \begin{cases} \pi^{\frac{1}{2}n - \frac{1}{2}}\left(\frac{1}{2}n - \frac{1}{2}\right)! / n! & n \text{ odd} \\ \pi^{\frac{1}{2}n} 2^{-n} / (\frac{1}{2}n)! & n \text{ even} \end{cases}$$

- "Area" of unit-diameter hypersphere:

$$A_n = \frac{d}{dr}(2r)^n V^n \Big|_{r=\frac{1}{2}} = 2nV_n$$

- >63% of V_n is in shell $(1 - n^{-1})R \le r \le R$

Proof : $\left(1 - n^{-1}\right)^n \underset{n\to\infty}{\overset{<}{\to}} e^{-1} = 0.3679$

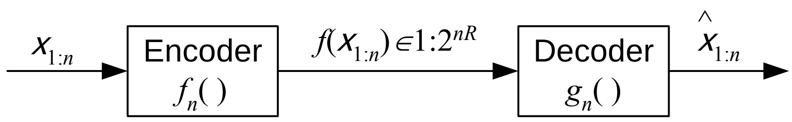
n	V_n	A_n
1	1	2
2	0.79	3.14
3	0.52	3.14
4	0.31	2.47
10	2.5e–3	5e–2
100	1.9e–70	3.7e–68

Most of n-dimensional space is in the corners

Lecture 17

- Rate Distortion Theorem

Review



Rate Distortion function for x whose $p_{\mathbf{x}}(\mathbf{x})$ is known is

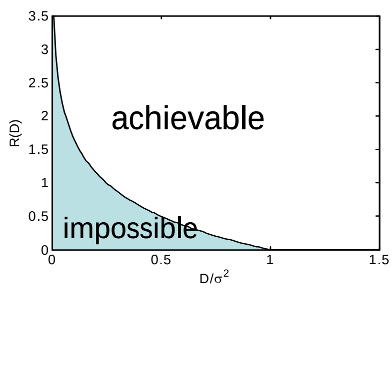
$$R(D) = \inf R \text{ such that } \exists f_n, g_n \text{ with } \lim_{n\to\infty} E_{\mathbf{x}\in\mathcal{X}^n} d(\mathbf{x}, \hat{\mathbf{x}}) \le D$$

Rate Distortion Theorem:

$$R(D) = \min I(x; \hat{x}) \text{ over all } p(\hat{x} \mid x) \text{ such that } E_{x,\hat{x}} d(x, \hat{x}) \le D$$

We will prove this theorem for discrete X and bounded $d(x,y) \le d_{\max}$

$R(D)$ curve depends on your choice of $d(,)$



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 $R_0 \ge R(D) = R\big(E\, d(\mathbf{x};\hat{\mathbf{x}})\big)$

– We show first that $R_0 \ge n^{-1} \sum_i I(x_i;\hat{x}_i)$

– We know that $I(x_i;\hat{x}_i) \ge R\big(E\, d(x_i;\hat{x}_i)\big)$ Def^n of $R(D)$

– and use convexity to show

$$n^{-1} \sum_i R\big(E\, d(x_i;\hat{x}_i)\big) \ge R\big(E\, d(\mathbf{x};\hat{\mathbf{x}})\big) = R(D)$$

We prove convexity first and then the rest

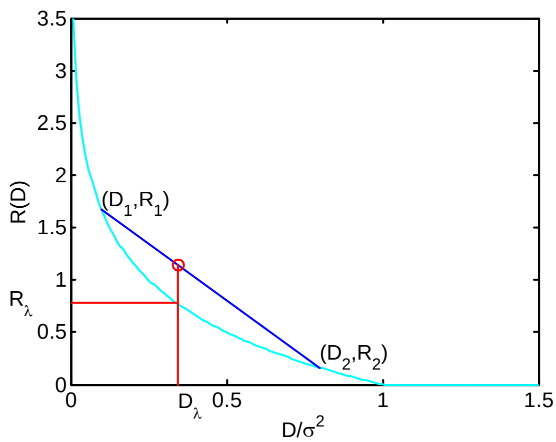
Convexity of $R(D)$

If $p_1(\hat{x}\,|\,x)$ and $p_2(\hat{x}\,|\,x)$ are associated with (D_1, R_1) and (D_2, R_2) on the $R(D)$ curve we define

$$p_\lambda(\hat{x}\,|\,x) = \lambda p_1(\hat{x}\,|\,x) + (1-\lambda) p_2(\hat{x}\,|\,x)$$

Then

$$E_{p_\lambda} d(x,\hat{x}) = \lambda D_1 + (1-\lambda) D_2 = D_\lambda$$



$$R(D_\lambda) \le I_{p_\lambda}(x;\hat{x})$$
$$\le \lambda I_{p_1}(x;\hat{x}) + (1-\lambda) I_{p_2}(x;\hat{x})$$
$$= \lambda R(D_1) + (1-\lambda) R(D_2)$$

$R(D) = \min_{p(\hat{x}|x)} I(x;\hat{x})$

$I(x;\hat{x})$ convex w.r.t. $p(\hat{x}\,|\,x)$

p_1 and p_2 lie on the $R(D)$ curve

Proof that $R \geq R(D)$

$$nR_0 \geq H(\hat{x}_{1:n}) = H(\hat{x}_{1:n}) - H(\hat{x}_{1:n} \mid x_{1:n})$$

Uniform bound; $H(\hat{x} \mid x) = 0$

$$= I(\hat{x}_{1:n}; x_{1:n})$$

Definition of $I(;)$

$$\geq \sum_{i=1}^{n} I(x_i; \hat{x}_i)$$

x_i indep: Mut Inf Independence Bound

$$\geq \sum_{i=1}^{n} R\big(E\, d(x_i; \hat{x}_i)\big) = n \sum_{i=1}^{n} n^{-1} R\big(E\, d(x_i; \hat{x}_i)\big)$$

definition of R

$$\geq nR\left(n^{-1} \sum_{i=1}^{n} E\, d(x_i; \hat{x}_i)\right) = nR\big(E\, d(x_{1:n}; \hat{x}_{1:n})\big)$$

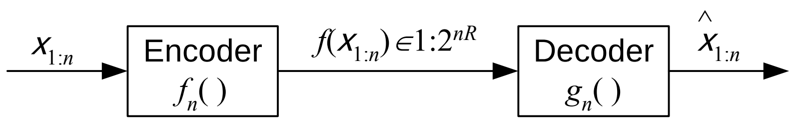
convexity

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 $p(\hat{x} \mid x)$ that gives $I(x; \hat{x}) = R(D)$
- Random Decoder: 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. $\sim p(x,\hat{x})$

$$J_{d,\varepsilon}^{(n)} = \{\mathbf{x}, \hat{\mathbf{x}} \in \mathcal{X}^n \times \hat{\mathcal{X}}^n : \left| -n^{-1} \log p(\mathbf{x}) - H(X) \right| < \varepsilon,$$
$$\left| -n^{-1} \log p(\hat{\mathbf{x}}) - H(\hat{X}) \right| < \varepsilon,$$
$$\left| -n^{-1} \log p(\mathbf{x}, \hat{\mathbf{x}}) - H(X, \hat{X}) \right| < \varepsilon$$
$$\left| d(\mathbf{x}, \hat{\mathbf{x}}) - E\, d(X, \hat{X}) \right| < \varepsilon \}$$

new condition

Properties:

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 def^n of J

$$= p(\hat{\mathbf{x}}) 2^{n(I(X;\hat{X})+3\varepsilon)}$$

def^n of I

Curious but necessary Inequality

Lemma: $u, v \in [0,1], m > 0 \quad \Rightarrow \quad (1-uv)^m \le 1 - u + e^{-vm}$

Proof: $u=0$: $e^{-vm} \ge 0 \quad \Rightarrow \quad (1-0)^m \le 1 - 0 + e^{-vm}$

$u=1$: Define $f(v) = e^{-v} - 1 + v \quad \Rightarrow \quad f'(v) = 1 - e^{-v}$

$f(0) = 0$ and $f'(v) > 0$ for $v > 0 \quad \Rightarrow \quad f(v) \ge 0$ for $v \in [0,1]$

Hence for $v \in [0,1], \quad 0 \le 1 - v \le e^{-v} \quad \Rightarrow (1-v)^m \le e^{-vm}$

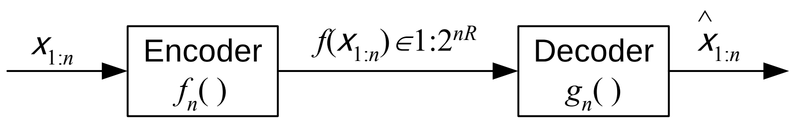
$0<u<1$: Define $g_v(u) = (1-uv)^m$

$\Rightarrow g_v''(x) = m(m-1)v^2(1-uv)^{n-2} \ge 0 \Rightarrow g_v(u)$ convex for $u,v \in [0,1]$

$(1-uv)^m = g_v(u) \le (1-u)g_v(0) + ug_v(1)$ convexity for $u,v \in [0,1]$

$= (1-u)1 + u(1-v)^m \le 1 - u + ue^{-vm} \le 1 - u + e^{-vm}$

Achievability of $R(D)$: preliminaries



- Choose D and find a $p(\hat{x} \mid x)$ such that $I(x;\hat{x}) = R(D); E\, d(x,\hat{x}) \le D$
  Choose $\delta > 0$ and define $\mathbf{p}_{\hat{x}} = \{ \quad p(\hat{x}) = \sum_x p(x)p(\hat{x} \mid x) \quad \}$
- 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 decode 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. Supose the probability of this situation is P_e.

Hence
$$\begin{aligned}\overline{D} &= E_{\mathbf{x},g}\, d(\mathbf{x},\hat{\mathbf{x}}) \\ &\le (1-P_e)(D+\varepsilon) + P_e d_{\max} \\ &\le D + \varepsilon + P_e d_{\max}\end{aligned}$$

We need to show that the expected value of P_e is small

Error Probability

Define the set of valid inputs for code g

$$V(g) = \left\{\mathbf{x} : \exists w \text{ with } (\mathbf{x}, g(w)) \in J_{d,\varepsilon}^{(n)}\right\}$$

We have $P_e = \sum_g p(g) \sum_{\mathbf{x}\notin V(g)} p(\mathbf{x}) = \sum_{\mathbf{x}} p(\mathbf{x}) \sum_{g:\mathbf{x}\notin V(g)} p(g)$

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 is $\left(1 - \sum_{\hat{\mathbf{x}}} p(\hat{\mathbf{x}}) K(\mathbf{x},\hat{\mathbf{x}})\right)^{2^{nR}}$

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 $\mathbf{x},\hat{\mathbf{x}} \in J_{d,\varepsilon}^{(n)} \Rightarrow p(\hat{\mathbf{x}}) \ge p(\hat{\mathbf{x}} \mid \mathbf{x}) 2^{-n\left(I(X;\hat{X})+3\varepsilon\right)}$

$$P_e = \sum_{\mathbf{x}} p(\mathbf{x}) \left(1 - \sum_{\hat{\mathbf{x}}} p(\hat{\mathbf{x}}) K(\mathbf{x},\hat{\mathbf{x}})\right)^{2^{nR}}$$

$$\le \sum_{\mathbf{x}} p(\mathbf{x}) \left(1 - \sum_{\hat{\mathbf{x}}} p(\hat{\mathbf{x}} \mid \mathbf{x}) K(\mathbf{x},\hat{\mathbf{x}}) \quad 2^{-n\left(I(X;\hat{X})+3\varepsilon\right)}\right)^{2^{nR}}$$

Using $(1-uv)^m \le 1 - u + e^{-vm}$

with $u = \sum_{\hat{\mathbf{x}}} p(\hat{\mathbf{x}} \mid \mathbf{x}) K(\mathbf{x},\hat{\mathbf{x}}); \quad v = 2^{-nI(X;\hat{X})-3n\varepsilon}; \quad m = 2^{nR}$

$$\le \sum_{\mathbf{x}} p(\mathbf{x}) \left(1 - \sum_{\hat{\mathbf{x}}} p(\hat{\mathbf{x}} \mid \mathbf{x}) K(\mathbf{x},\hat{\mathbf{x}}) + \exp\left(-2^{-n\left(I(X;\hat{X})+3\varepsilon\right)} 2^{nR}\right)\right)$$

Note : $0 \le u, v \le 1$ as required

Achievability for average code

$$P_e \le \sum_{\mathbf{x}} p(\mathbf{x}) \left(1 - \sum_{\hat{\mathbf{x}}} p(\hat{\mathbf{x}} \mid \mathbf{x}) K(\mathbf{x},\hat{\mathbf{x}}) + \exp\left(-2^{-n\left(I(X;\hat{X})+3\varepsilon\right)} 2^{nR}\right)\right)$$

take out terms not involving $\mathbf{x}$

$$= \left(1 + \exp\left(-2^{-n\left(I(X;\hat{X})+3\varepsilon\right)} 2^{nR}\right)\right) - \sum_{\mathbf{x},\hat{\mathbf{x}}} p(\mathbf{x}) p(\hat{\mathbf{x}} \mid \mathbf{x}) K(\mathbf{x},\hat{\mathbf{x}})$$

$$= 1 - \sum_{\mathbf{x},\hat{\mathbf{x}}} p(\mathbf{x},\hat{\mathbf{x}}) K(\mathbf{x},\hat{\mathbf{x}}) + \exp\left(-2^{n\left(R - I(X;\hat{X}) - 3\varepsilon\right)}\right)$$

$$= P\left\{(\mathbf{x},\hat{\mathbf{x}}) \notin J_{d,\varepsilon}^{(n)}\right\} + \exp\left(-2^{n\left(R - I(X;\hat{X}) - 3\varepsilon\right)}\right)$$

both terms $\to 0$ as $n \to \infty$ provided $nR > I(X,\hat{X}) + 3\varepsilon$

$$\xrightarrow[n\to\infty]{} 0$$

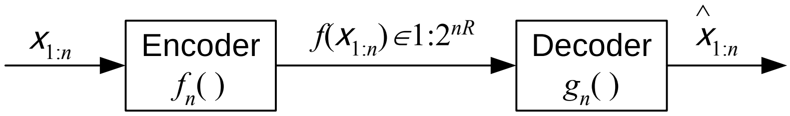
Hence $\forall \delta > 0,\ \overline{D} = E_{\mathbf{x},g}\, d(\mathbf{x},\hat{\mathbf{x}})$ can be made $\le D + \delta$

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:

$\forall \delta > 0, D$ and $R > R(D), \exists f_n, g_n$ with $p(d(\mathbf{x},\hat{\mathbf{x}}) \le D+\delta) \underset{n\to\infty}{\to} 1$

Lecture 18

- Revision Lecture

Summary (1)

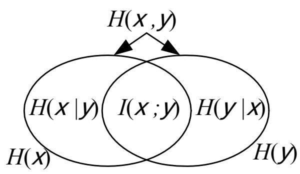
- Entropy: $H(x)=\sum_{x\in\mathcal{X}} p(x)\times -\log_2 p(x) = E-\log_2(p_X(x))$
  - Bounds: $0\le H(x)\le \log|\mathcal{X}|$
  - Conditioning reduces entropy: $H(y\,|\,x)\le H(y)$
  - Chain Rule: $H(x_{1:n})=\sum_{i=1}^{n} H(x_i\,|\,x_{1:i-1})\le \sum_{i=1}^{n} H(x_i)$

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

$$D(\mathbf{p}\,\|\,\mathbf{q}) = E_{\mathbf{p}} \log\big(p(x)/q(x)\big)\ge 0$$

Summary (2)

- Mutual Information:



$$I(y;x)=H(y)-H(y\,|\,x)$$
$$=H(x)+H(y)-H(x,y)=D(\mathbf{p}_{x,y}\,\|\,\mathbf{p}_x\mathbf{p}_y)$$

  - Positive and Symmetrical: $I(x;y)=I(y;x)\ge 0$
  - x, y indep $\Leftrightarrow$ $H(x,y)=H(y)+H(x)\Leftrightarrow I(x;y)=0$
  - Chain Rule: $I(x_{1:n};y)=\sum_{i=1}^{n} I(x_i;y\,|\,x_{1:i-1})$

$$x_i \text{ independent} \quad\Rightarrow\quad I(x_{1:n};y_{1:n})\ge \sum_{i=1}^{n} I(x_i;y_i)$$

$$p(y_i\,|\,x_{1:n};y_{1:i-1})=p(y_i\,|\,x_i) \quad\Rightarrow\quad I(x_{1:n};y_{1:n})\le \sum_{i=1}^{n} I(x_i;y_i)$$

Summary (3)

- Convexity: $f''(x) \ge 0 \Rightarrow f(x)$ convex $\Rightarrow Ef(x) \ge f(Ex)$
  - $H(\mathbf{p})$ 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: $x \to y \to z \Leftrightarrow p(z\,|\,x,y) = p(z\,|\,y) \Leftrightarrow I(x;z\,|\,y) = 0$

  $\Rightarrow I(x\,;y) \ge I(x\,;z)$ and $I(x\,;y) \ge I(x;y\,|\,z)$
- Fano: $x \to y \to \hat{x} \Rightarrow p(\hat{x} \ne x) \ge \dfrac{H(x\,|\,y)-1}{\log(|\mathcal{X}|-1)}$
- Entropy Rate: $H(\mathcal{X}) = \lim_{n\to\infty} n^{-1} H(x_{1:n})$
  - Stationary process $H(\mathcal{X}) \le H(x_n\,|\,x_{1:n-1})$ = as $n \to \infty$
  - Markov Process: $H(\mathcal{X}) = \lim_{n\to\infty} H(x_n\,|\,x_{n-1})$ = if stationary
  - Hidden Markov: $H(y_n\,|\,y_{1:n-1}, x_1) \le H(\mathcal{Y}) \le H(y_n\,|\,y_{1:n-1})$ = as $n \to \infty$

Summary (4)

- Kraft: Uniquely Decodable $\Rightarrow \sum_{i=1}^{|\mathcal{X}|} D^{-l_i} \le 1 \Rightarrow \exists$ prefix code
- Average Length: Uniquely Decodable $\Rightarrow L_C = E\,l(x) \ge H_D(x)$
- Shannon-Fano: Top-down 50% splits. $L_{SF} \le H_D(x)+1$
- Shannon: $l_x = \lceil -\log_D p(x) \rceil$ $L_S \le H_D(x)+1$
- 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
- Arithmetic Coding: $C(x^N) = \sum_{x_i^N < x^N} p(x_i^N)$
  - Long blocks reduce 2-bit overhead
  - Efficient algorithm without calculating all possible probabilities
  - Can have adaptive probabilities

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 $(1-\varepsilon)2^{n(H(X)-\varepsilon)} \overset{n>N_\varepsilon}{<} \left|T_\varepsilon^{(n)}\right| \leq 2^{n(H(X)+\varepsilon)}$
  - 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
- Feedback doesn't increase capacity but simplifies coder
- Joint Source-Channel Coding doesn't increase capacity

Summary (7)

- 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 (8)

- Parallel Gaussian Channels: Total power constraint $\sum P_i = P$
  - White noise: Waterfilling: $P_i = \max(P_0 - N_i, 0)$
  - 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: $R(D) = \max(½\log(\sigma^2 D^{-1}), 0)$
  - Can encode at rate R: random decoder, joint typical encoder
  - Can't encode below rate R: independence bound
- Lloyd Algorithm: iterative optimal vector quantization