

# Information Theory

Mike Brookes  
E4.40, ISE4.51, SO20

## Lectures

### Entropy Properties

- 1 Entropy - 6
- 2 Mutual Information - 19

### Lossless Coding

- 3 Symbol Codes - 30
- 4 Optimal Codes - 41
- 5 Stochastic Processes - 55
- 6 Stream Codes - 68

### Channel Capacity

- 7 Markov Chains - 83
- 8 Typical Sets - 93
- 9 Channel Capacity - 105
- 10 Joint Typicality - 118

- 11 Coding Theorem - 128
- 12 Separation Theorem - 135

### Continuous Variables

- 13 Differential Entropy - 145
- 14 Gaussian Channel - 159
- 15 Parallel Channels - 172

### Lossy Coding

- 16 Lossy Coding - 184
- 17 Rate Distortion Bound - 198

### Revision

- 18 Revision - 212
- 19
- 20

## 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,  $\odot$ =elementwise product

### Scalar Random Variables

- $X = \mathcal{R.V.}$ ,  $x$  = specific value,  $\mathcal{X}$  = alphabet

### Random Column Vector of length $N$

- $\mathbf{x} = \mathcal{R.V.}$ ,  $\mathbf{x}$  = specific value,  $\mathcal{X}^N$  = alphabet
- $x_i$  and  $x_i$  are particular vector elements

### Ranges

- $a:b$  denotes the range  $a, a+1, \dots, b$

## Discrete Random Variables

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

Examples:

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

$\mathbf{p}_{\mathcal{X}}$  is a "probability mass vector"

"english text"

$$\mathcal{X} = [a; b; \dots; y; z; \text{<space>}]$$

$$\mathbf{p}_{\mathcal{X}} = [0.058; 0.013; \dots; 0.016; 0.0007; 0.193]$$

Note: we normally drop the subscript from  $p_{\mathcal{X}}$  if unambiguous

## Expected Values

- If  $g(x)$  is real valued and defined on  $\mathcal{X}$  then

$$E_x g(x) = \sum_{x \in \mathcal{X}} p(x)g(x) \quad \text{often write } E \text{ for } E_x$$

Examples:



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

$$E x = 3.5 = \mu$$

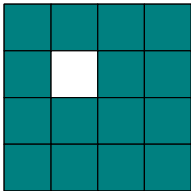
$$E x^2 = 15.17 = \sigma^2 + \mu^2$$

$$E \sin(0.1x) = 0.338$$

$$E -\log_2(p(x)) = 2.58 \quad \leftarrow \text{This is the "entropy" of } \mathcal{X}$$

## Minesweeper

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



Guess	Prob	SIC
1. No	$15/16$	0.093 bits

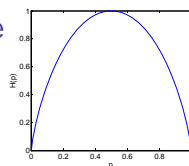
## Entropy Examples

### (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

$$\mathcal{X} = [\bullet; \blacksquare; \blacklozenge; \bullet], \mathbf{p}_x = [1/2; 1/4; 1/8; 1/8]$$

$$H(x) = H(\mathbf{p}_x) = -\sum \log(p(x))p(x) = 1 \times 1/2 + 2 \times 1/4 + 3 \times 1/8 + 3 \times 1/8 = 1.75 \text{ bits}$$

## 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

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

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

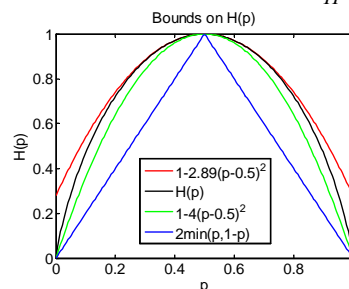
## Bernoulli Entropy Properties

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

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

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

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



### Quadratic Bounds

$$H(p) \leq 1 - 2\log e(p - 1/2)^2 = 1 - 2.89(p - 1/2)^2$$

$$H(p) \geq 1 - 4(p - 1/2)^2 \geq 2\min(p, 1-p)$$

Proofs in problem sheet

## Joint and Conditional Entropy

### Joint Entropy: $H(X, Y)$

$$H(X, Y) = E - \log p(X, Y)$$

$$= -\frac{1}{2} \log \frac{1}{2} - \frac{1}{4} \log \frac{1}{4} - 0 \log 0 - \frac{1}{4} \log \frac{1}{4} = 1.5 \text{ bits} \quad \text{Note: } 0 \log 0 = 0$$

### Conditional Entropy : $H(Y | X)$

$$H(Y | X) = E - \log p(Y | X)$$

$$= -\frac{1}{2} \log \frac{2}{3} - \frac{1}{4} \log \frac{1}{3} - 0 \log 0 - \frac{1}{4} \log 1 = 0.689 \text{ bits}$$

$p(x, y)$	$y=0$	$y=1$
$x=0$	$\frac{1}{2}$	$\frac{1}{4}$
$x=1$	0	$\frac{1}{4}$

$p(y   x)$	$y=0$	$y=1$
$x=0$	$\frac{2}{3}$	$\frac{1}{3}$
$x=1$	0	1

Note: rows sum to 1

## Conditional Entropy – view 1

### Additional Entropy:

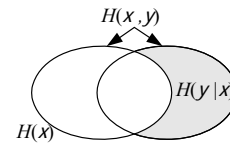
$$p(Y | X) = p(X, Y) \div p(X)$$

$$H(Y | X) = E - \log p(Y | X)$$

$$= E \{-\log p(X, Y)\} - E \{-\log p(X)\}$$

$$= H(X, Y) - H(X) = H(\frac{1}{2}, \frac{1}{4}, 0, \frac{1}{4}) - H(\frac{1}{4}) = 0.689 \text{ bits}$$

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



## Conditional Entropy – view 2

### Average Row Entropy:

$p(x, y)$	$y=0$	$y=1$	$H(Y   X=x)$	$p(x)$
$x=0$	$\frac{1}{2}$	$\frac{1}{4}$	$H(1/3)$	$\frac{3}{4}$
$x=1$	0	$\frac{1}{4}$	$H(1)$	$\frac{1}{4}$

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

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

$$= \sum_{x \in \mathcal{X}} p(x) H(Y | X = x) = \frac{3}{4} \times H(\frac{1}{3}) + \frac{1}{4} \times H(1) = 0.689 \text{ bits}$$

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

## Chain Rules

### • Probabilities

$$p(X, Y, Z) = p(Z | X, Y) p(Y | X) p(X)$$

### • Entropy

$$H(X, Y, Z) = H(Z | X, Y) + H(Y | X) + H(X)$$

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

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  $0 \leq H(X) \leq \log |\mathcal{X}|$

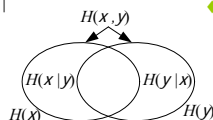
### • Chain Rule:

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

### • Conditional Entropy:

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

– Conditioning reduces entropy  $H(Y | X) \leq H(Y)$



♦ = 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
  - Mutual 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  $y$

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

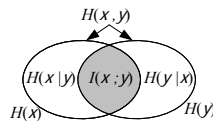
Information in  $x$

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

Mutual information is **symmetrical**

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

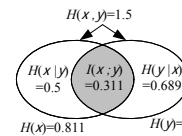
Use ";" to avoid ambiguities between  $I(x, y, z)$  and  $I(x, y; z)$



## Mutual Information Example

$p(x, y)$	$y=0$	$y=1$
$x=0$	$\frac{1}{2}$	$\frac{1}{4}$
$x=1$	0	$\frac{1}{4}$

- 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



$$\begin{aligned} I(x; y) &= H(x) - H(x | y) \\ &= H(x) + H(y) - H(x, y) \\ H(x) &= 0.811, \quad H(y) = 1, \quad H(x, y) = 1.5 \\ I(x; y) &= 0.311 \end{aligned}$$

## Conditional Mutual Information

### Conditional Mutual Information

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

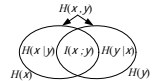
**Note:**  $Z$  conditioning applies to **both**  $X$  and  $Y$

### 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:**  $H(x) = \sum_{x \in \mathcal{X}} -\log_2(p(x))p(x) = E[-\log_2(p_X(x))]$ 
  - Always positive  $H(x) \geq 0$
- Chain Rule:**  $H(x, y) = H(x) + H(y | x) \leq H(x) + H(y)$ 
  - Conditioning reduces entropy  $H(y | x) \leq H(y)$
- Mutual Information:**
  - $I(y; x) = H(y) - H(y | x) = H(x) + H(y) - H(x, y)$
  - Positive and Symmetrical  $I(x; y) = I(y; x) \geq 0$
  - $x$  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:  $\log x, \sqrt{x} [x \geq 0]$
- Convex and Concave:  $x$

- Test:  $\frac{d^2 f}{dx^2} > 0 \quad \forall x \in (a, b) \Rightarrow f(x)$  is strictly convex

"convex" (not strictly) uses " $\leq$ " in definition and " $\geq$ " in test

## Jensen's Inequality

**Jensen's Inequality:** (a)  $f(x)$  convex  $\Rightarrow E f(x) \geq f(E x)$

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

**Proof by induction on  $|X|$**

$$\begin{aligned} - |X|=1: & E f(x) = f(E x) = f(x_1) \\ - |X|=k: & E f(x) = \sum_{i=1}^k p_i f(x_i) = p_k f(x_k) + (1 - p_k) \sum_{i=1}^{k-1} \frac{p_i}{1 - p_k} f(x_i) \\ & \geq p_k f(x_k) + (1 - p_k) f\left(\sum_{i=1}^{k-1} \frac{p_i}{1 - p_k} x_i\right) \quad \text{Assume JI is true for } |X|=k-1 \\ & \geq f\left(p_k x_k + (1 - p_k) \sum_{i=1}^{k-1} \frac{p_i}{1 - p_k} x_i\right) = f(E x) \end{aligned}$$

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

## Jensen's Inequality Example

Mnemonic example:

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

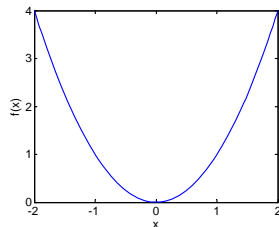
$\mathbf{X} = [-1; +1]$

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

$E \mathbf{x} = 0$

$f(E \mathbf{x}) = 0$

$E f(\mathbf{x}) = 1 > f(E \mathbf{x})$



## Relative Entropy Example



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

$\mathbf{p} = [1/6 \ 1/6 \ 1/6 \ 1/6 \ 1/6 \ 1/6] \Rightarrow H(\mathbf{p}) = 2.585$

$\mathbf{q} = [1/10 \ 1/10 \ 1/10 \ 1/10 \ 1/10 \ 1/2] \Rightarrow H(\mathbf{q}) = 2.161$

$D(\mathbf{p} \parallel \mathbf{q}) = E_{\mathbf{p}}(-\log q_{\mathbf{x}}) - H(\mathbf{p}) = 2.935 - 2.585 = 0.35$

$D(\mathbf{q} \parallel \mathbf{p}) = E_{\mathbf{q}}(-\log p_{\mathbf{x}}) - H(\mathbf{q}) = 2.585 - 2.161 = 0.424$

## Information Inequality Corollaries

- Uniform distribution has highest entropy

– Set  $\mathbf{q} = [|\mathbf{X}|^{-1}, \dots, |\mathbf{X}|^{-1}]^T$  giving  $H(\mathbf{q}) = \log |\mathbf{X}|$  bits

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

- Mutual Information is non-negative

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

$$= D(\mathbf{p}_{\mathbf{x}, \mathbf{y}} \parallel \mathbf{p}_{\mathbf{x}} \otimes \mathbf{p}_{\mathbf{y}}) \geq 0$$

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

## Relative Entropy

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

$$D(\mathbf{p} \parallel \mathbf{q}) = \sum_{\mathbf{x} \in \mathbf{X}} p(\mathbf{x}) \log \frac{p(\mathbf{x})}{q(\mathbf{x})} = E_{\mathbf{p}} \log \frac{p(\mathbf{x})}{q(\mathbf{x})} = E_{\mathbf{p}} (-\log q(\mathbf{x})) - H(\mathbf{x})$$

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

$D(\mathbf{p} \parallel \mathbf{q})$  measures the “distance” between the probability mass functions  $\mathbf{p}$  and  $\mathbf{q}$ .

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

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

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

## Information Inequality

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

- Define  $\mathbf{A} = \{\mathbf{x} : p(\mathbf{x}) > 0\} \subseteq \mathbf{X}$

$$\begin{aligned} \text{• Proof } -D(\mathbf{p} \parallel \mathbf{q}) &= -\sum_{\mathbf{x} \in \mathbf{A}} p(\mathbf{x}) \log \frac{p(\mathbf{x})}{q(\mathbf{x})} = \sum_{\mathbf{x} \in \mathbf{A}} p(\mathbf{x}) \log \frac{q(\mathbf{x})}{p(\mathbf{x})} \\ &\leq \log \left( \sum_{\mathbf{x} \in \mathbf{A}} p(\mathbf{x}) \frac{q(\mathbf{x})}{p(\mathbf{x})} \right) = \log \left( \sum_{\mathbf{x} \in \mathbf{A}} q(\mathbf{x}) \right) \leq \log \left( \sum_{\mathbf{x} \in \mathbf{X}} q(\mathbf{x}) \right) = \log 1 = 0 \end{aligned}$$

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

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

## More Corollaries

- Conditioning reduces entropy

$$0 \leq I(\mathbf{x}; \mathbf{y}) = H(\mathbf{y}) - H(\mathbf{y} | \mathbf{x}) \Rightarrow H(\mathbf{y} | \mathbf{x}) \leq H(\mathbf{y})$$

with equality only if  $\mathbf{x}$  and  $\mathbf{y}$  are independent.

- Independence Bound

$$H(\mathbf{x}_{1:n}) = \sum_{i=1}^n H(\mathbf{x}_i | \mathbf{x}_{1:i-1}) \leq \sum_{i=1}^n H(\mathbf{x}_i)$$

with equality only if all  $\mathbf{x}_i$  are independent.

E.g.: If all  $\mathbf{x}_i$  are identical  $H(\mathbf{x}_{1:n}) = H(\mathbf{x}_1)$

## Conditional Independence Bound

### Conditional Independence Bound

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

### Mutual Information Independence Bound

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

$$\begin{aligned} I(x_{1:n}; y_{1:n}) &= H(x_{1:n}) - H(x_{1:n} | y_{1:n}) \\ &\geq \sum_{i=1}^n H(x_i) - \sum_{i=1}^n H(x_i | y_i) = \sum_{i=1}^n I(x_i; y_i) \end{aligned}$$

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**  $I(x; y) = H(x) - H(x | y) \leq H(x)$
  - **Jensen's Inequality:**  $f(x)$  convex  $\Rightarrow E f(x) \geq f(E x)$
  - **Relative Entropy:**
    - $D(p \| q) = 0$  iff  $p = q$   $D(p \| q) = E_p \log \frac{p(x)}{q(x)} \geq 0$
  - **Corollaries**
    - Uniform Bound: Uniform  $p$  maximizes  $H(p)$
    - $I(x; y) \geq 0 \Rightarrow$  Conditioning reduces entropy
    - Indep bounds:  $H(x_{1:n}) \leq \sum_{i=1}^n H(x_i)$   $H(x_{1:n} | y_{1:n}) \leq \sum_{i=1}^n H(x_i | y_i)$
- $$I(x_{1:n}; y_{1:n}) \geq \sum_{i=1}^n I(x_i; y_i) \quad \text{if } x_i \text{ or } y_i \text{ are indep}$$

## Lecture 3

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

## Symbol Codes

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

Examples:

- $C(E, F, G, H) = (0, 1, 00, 11)$
- $C(E, F) = (0, 101)$
- $C(E, F) = (1, 101)$
- $C(E, F, G, H) = (00, 01, 10, 11)$
- $C(E, F, G, H) = (0, 01, 011, 111)$

$\overline{P}U$

$PU$

$\overline{P}U$

$PU$

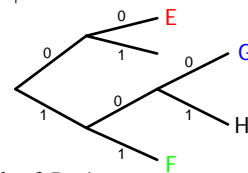
$\overline{P}U$

## Code Tree

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

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

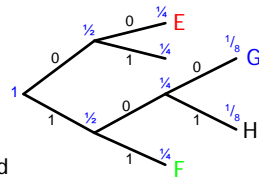
- $D$  branches at each node
- Each node along the path to a leaf is a prefix of the leaf
  - $\Rightarrow$  can't be a leaf itself
- Some leaves may be unused
  - all used  $\Rightarrow |\mathcal{X}| - 1$  is a multiple of  $D - 1$



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:  $l_1, l_2, \dots, l_{|X|} \Rightarrow \sum_{i=1}^{|X|} 2^{-l_i} \leq 1$
- Equality iff all leaves are utilised
- Total code budget = 1  
Code 00 uses up  $1/4$  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  $l_1, l_2, \dots, l_{|X|}$ , then  $\sum_{i=1}^{|X|} D^{-l_i} \leq 1$

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

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

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

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

## Converse to Kraft Inequality

If  $\sum_{i=1}^{|X|} D^{-l_i} \leq 1$  then  $\exists$  a prefix code with codeword lengths  $l_1, l_2, \dots, l_{|X|}$

**Proof:**

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

$\Rightarrow$  non-prefix symbol codes are a waste of time

## Kraft Converse Example

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

$l_k$	$c_k = \sum_{i=1}^{k-1} D^{-l_i}$	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  $\Rightarrow L_C \geq H(X)/\log_2 D$

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

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

$$= E (-\log_D D^{-l(x)} + \log_D p(x)) = E (-\log_D c q(x) + \log_D p(x))$$

$$= E \left( \log_D \frac{p(x)}{q(x)} \right) - \log_D c = \log_D 2(D(\mathbf{p} \parallel \mathbf{q}) - \log c) \geq 0$$

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

## Fano Code

Fano Code (also called Shannon-Fano code)

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

a	0.20	0	00
b	0.19	0	010
c	0.17	1	011
d	0.15	0	100
e	0.14	0	101
f	0.06	1	110
g	0.05	1	1110
h	0.04	1	1111

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

$L_{SF} = 2.89$  bits

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

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  $\sum_{i=1}^{|X|} D^{-l_i} \leq 1$
  - If  $\sum_{i=1}^{|X|} D^{-l_i} \leq 1$  then you can create a prefix code
- Uniquely decodable  $\Rightarrow L_C \geq H(X) \log_2 D$
- 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

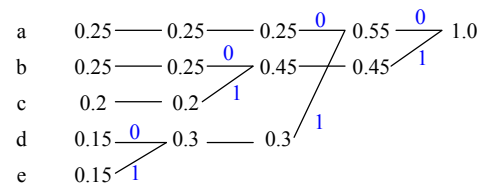
An Optimal Binary prefix code must satisfy:

1.  $p(x_i) > p(x_j) \Rightarrow l_i \leq l_j$  (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

$$\mathbf{X}=[a, b, c, d, e], p_x=[0.25 \ 0.25 \ 0.2 \ 0.15 \ 0.15]$$


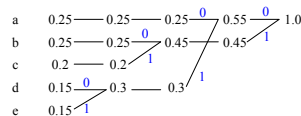
Read diagram **backwards** for codewords:

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

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

## Huffman Code is Optimal 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  $L_{C_m} < L_{C_m}$
- 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 (doesn'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  $L_{C_{m-1}} = L_{C_m} - p_i - p_j$  hence  $L_{C_{m-1}} < L_{C_{m-1}}$  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  $\sum_{x \in X} p(x) l(x)$  subject to

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

Simplified version:

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

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

## Optimal Codes (non-integer $l_i$ )

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

Use lagrange multiplier:

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

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

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

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

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

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

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)

$$\mathbf{p}_x = [0.99 \quad 0.01]$$

$$-\log_2 \mathbf{p}_x = [0.0145 \quad 6.64]$$

$$\mathbf{l}_x = \lceil -\log_2 \mathbf{p}_x \rceil = [1 \quad 7] \quad (\text{obviously stupid to use 7})$$

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

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

## Shannon versus Huffman

Shannon

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

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

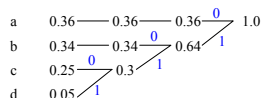
$$\mathbf{l}_s = \lceil -\log_2 \mathbf{p}_x \rceil = [2 \quad 2 \quad 2 \quad 5]$$

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

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  $p(l(x) \leq l_s(x) - c) \leq 2^{1-c}$

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

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

$$= \sum_{x \in \mathbf{A}} p(x) \leq \sum_{x \in \mathbf{A}} \max(p(x) | x \in \mathbf{A}) < \sum_{x \in \mathbf{A}} 2^{-l(x)-c+1}$$

$$\leq \sum_x 2^{-l(x)-c+1} = 2^{-(c-1)} \sum_x 2^{-l(x)} \leq 2^{-(c-1)} \quad \text{Kraft inequality}$$

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  $p(l(x) < l_s(x)) \leq p(l(x) > l_s(x))$  with equality iff  $l(x) = l_s(x)$

**Proof:**

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

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

$$\stackrel{A}{\leq} \sum_x p(x) (2^{l_s(x) - l(x)} - 1) = -1 + \sum_x 2^{-l_s(x)} 2^{l_s(x) - l(x)} \quad \text{sgn() property dyadic} \Rightarrow p = 2^{-l}$$

$$= -1 + \sum_x 2^{-l(x)} \stackrel{B}{\leq} -1 + 1 = 0 \quad \text{Kraft inequality}$$

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  $\Sigma = 1$

## Summary

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

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

## Stochastic process

Stochastic Process  $\{X_i\} = X_1, X_2, \dots$

**Entropy:**  $H(\{X_i\}) = H(X_1) + H(X_2 | X_1) + \dots = \infty$  often

**Entropy Rate:**  $H(\mathbf{X}) = \lim_{n \rightarrow \infty} \frac{1}{n} H(X_{1:n})$  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:  $H(\mathbf{X}) = H(x_i)$
- $x_i$  indep,  $H(x_i) = 0$  1 00 11 0000 1111 00000000 ... no convergence

## Lemma: Limit of Cesàro Mean

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

**Proof:**

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

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

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

The partial means of  $a_k$  are called Cesàro Means

## Stationary Process

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

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

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

$$H(\mathbf{X}) = \lim_{n \rightarrow \infty} \frac{1}{n} H(x_{1:n}) = \lim_{n \rightarrow \infty} H(x_n | x_{1:n-1})$$

**Proof:**  $0 \leq H(x_n | x_{1:n-1}) \leq H(x_n | x_{2:n-1}) = H(x_{n-1} | x_{1:n-2})$   
 (a) conditioning reduces entropy, (b) stationarity

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

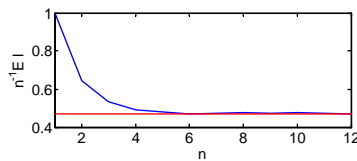
Hence from Cesàro Mean lemma:

$$H(x_k | x_{1:k-1}) \rightarrow b \Rightarrow \frac{1}{n} H(x_{1:n}) = \frac{1}{n} \sum_{k=1}^n H(x_k | x_{1:k-1}) \rightarrow b = H(\mathbf{X})$$

## Block Coding Example

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

$$H(x_i) = 0.469$$



- $n=1$ 

sym	A	B
prob	0.9	0.1
code	0	1

 $n^{-1} E I = 1$
- $n=2$ 

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

 $n^{-1} E I = 0.645$
- $n=3$ 

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

 $n^{-1} E I = 0.583$

## 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

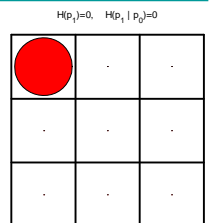
then it has exactly one stationary distribution,  $\mathbf{p}_s$ .

- $\mathbf{p}_s$  is the eigenvector of  $\mathbf{T}^T$  with  $\lambda = 1$ :  $\mathbf{T}^T \mathbf{p}_s = \mathbf{p}_s$
- $\mathbf{T}^n \rightarrow \mathbf{1} \mathbf{p}_s^T$  where  $\mathbf{1} = [1 \ 1 \ \dots \ 1]^T$

## Chess Board

Random Walk

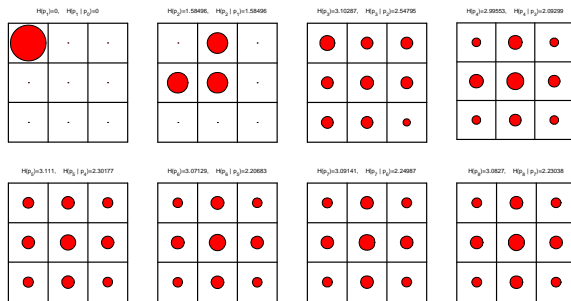
- Move  $\Leftrightarrow \Updownarrow \Leftarrow \Rightarrow$  equal prob
- $\mathbf{p}_1 = [1 \ 0 \ \dots \ 0]^T$ 
  - $H(\mathbf{p}_1) = 0$
- $\mathbf{p}_s = \frac{1}{40} \times [3 \ 5 \ 3 \ 5 \ 8 \ 5 \ 3 \ 5 \ 3]^T$ 
  - $H(\mathbf{p}_s) = 3.0855$



- $H(\mathbf{X}) = \lim_{n \rightarrow \infty} H(x_n | x_{n-1}) = \sum_{i,j} -p_{s,i} t_{i,j} \log(t_{i,j})$ 
  - $H(\mathbf{X}) = 2.2365$

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

## Chess Board Frames



## ALOHA example

Waiting Packets | TX en



<b>x:</b>	3	2	1	2	1	0	1	2
	0	1	1	1	1	2	2	1
<b>TX enable, e:</b>	1	0	0	1	1	1	1	1
	1	1	0	0	1	1	0	0
<b>y:</b>	1	0	0	1	1	0	1	1
	0	1	0	0	1	1	0	0

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

## Hidden Markov Process – (1)

Proof (1):  $H(y_n | y_{1:n-1}, x_1) \leq H(y)$

$$\begin{aligned}
 H(y_n | y_{1:n-1}, x_1) &= H(y_n | y_{1:n-1}, x_{-k:1}) \quad \forall k \quad \text{x markov} \\
 &= H(y_n | y_{1:n-1}, x_{-k:1}, y_{-k:1}) = H(y_n | y_{-k:n-1}, x_{-k:1}) \quad y=f(x) \\
 &\leq H(y_n | y_{-k:n-1}) \quad \forall k \quad \text{conditioning reduces entropy} \\
 &= H(y_{k+n} | y_{0:k+n-1}) \xrightarrow{k \rightarrow \infty} H(y) \quad y \text{ stationary}
 \end{aligned}$$

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

## 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  $\mathbf{x}_t = (n_1, \dots, n_M)$ 
  - $\{\mathbf{x}_t\}$  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.

## Hidden Markov Process

If  $\{x_t\}$  is a stationary Markov process and  $y_t = f(x_t)$  then  $\{y_t\}$  is a stationary Hidden Markov process.

What is entropy rate  $H(y)$ ?

- Stationarity  $\Rightarrow H(y_n | y_{1:n-1}) \geq H(y)$  and  $\xrightarrow{n \rightarrow \infty} H(y)$
- Also  $H(y_n | y_{1:n-1}, x_1) \leq H(y)$  and  $\xrightarrow[n \rightarrow \infty]{(1)} H(y)$

So  $H(y)$  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):  $H(y_n | y_{1:n-1}, x_1) \leq H(y)$

$$\begin{aligned}
 H(y_n | y_{1:n-1}, x_1) &= H(y_n | y_{1:n-1}, x_{-k:1}) \quad \forall k \quad \text{x markov} \\
 &= H(y_n | y_{1:n-1}, x_{-k:1}, y_{-k:1}) = H(y_n | y_{-k:n-1}, x_{-k:1}) \quad y=f(x) \\
 &\leq H(y_n | y_{-k:n-1}) \quad \forall k \quad \text{conditioning reduces entropy} \\
 &= H(y_{k+n} | y_{0:k+n-1}) \xrightarrow{k \rightarrow \infty} H(y) \quad y \text{ stationary}
 \end{aligned}$$

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):  $H(y_n | y_{1:n-1}, x_1) \xrightarrow{n \rightarrow \infty} H(y)$

Note that  $\sum_{n=1}^k I(x_1; y_n | y_{1:n-1}) = I(x_1; y_{1:k}) \leq H(x_1)$  chain rule  
def<sup>n</sup> of  $I(A;B)$

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

So  $H(y_n | y_{1:n-1}, x_1) = H(y_n | y_{1:n-1}) - I(x_1; y_n | y_{1:n-1}) \xrightarrow{n \rightarrow \infty} H(y) - 0$  def<sup>n</sup> of  $I(A;B)$

The influence of  $x_1$  on  $y_n$  decreases over time.

## Summary

- Entropy Rate:  $H(\mathbf{X}) = \lim_{n \rightarrow \infty} \frac{1}{n} H(\mathbf{X}_{1:n})$  if it exists
  - $\{x_i\}$  stationary:  $H(\mathbf{X}) = \lim_{n \rightarrow \infty} H(x_n | x_{1:n-1})$
  - $\{x_i\}$  stationary Markov:
 
$$H(\mathbf{X}) = H(x_n | x_{n-1}) = \sum_{i,j} -p_{s,i} t_{i,j} \log(t_{i,j})$$
  - $y=f(x)$ : Hidden Markov:
 
$$H(y_n | y_{1:n-1}, x_1) \leq H(\mathbf{Y}) \leq H(y_n | y_{1:n-1})$$
 with both sides tending to  $H(\mathbf{Y})$

## Huffman: Good and Bad

### Good

- Shortest possible symbol code  $\frac{H(x)}{\log_2 D} \leq L_s \leq \frac{H(x)}{\log_2 D} + 1$

### 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  $p(x_i) = 2^{-k(i)} \quad \forall i$
- Must recompute entire code if any symbol probability changes
  - A block of  $N$  symbols needs  $|\mathbf{X}|^N$  pre-calculated probabilities

## Arithmetic Coding – Code lengths

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

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

$$Q_7 = 0.9360 = 0.111011$$

$$Q_6 = 0.8400 = 0.110101$$

$$Q_5 = 0.7440 = 0.101111$$

bba	m7=11011
bab	m6=1100

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

$$\Rightarrow l_7 = 5, m_7 = 27, 28 \times 2^{-5} \leq Q_7 \quad \checkmark$$

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

## Lecture 6

- Stream Codes
- Arithmetic Coding
- Lempel-Ziv Coding

## Arithmetic Coding

- Take all possible blocks of  $N$  symbols and sort into lexical order,  $\mathbf{x}_r$  for  $r=1: |\mathbf{X}|^N$
- Calculate cumulative probabilities in binary:
 
$$Q_r = \sum_{i \leq r} p(\mathbf{x}_i), Q_0 = 0$$
- 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  $Q_{r-1} \leq m_r 2^{-l_r} < (m_r + 1)2^{-l_r} \leq Q_r$

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

Code

$$Q_8 = 1.0000 = 1.0000000$$

$$Q_7 = 0.9360 = 0.111011$$

$$Q_6 = 0.8400 = 0.110101$$

$$Q_5 = 0.7440 = 0.101111$$

$$Q_4 = 0.6000 = 0.100110$$

$$Q_3 = 0.5040 = 0.100000$$

$$Q_2 = 0.3600 = 0.010111$$

$$Q_1 = 0.2160 = 0.001101$$

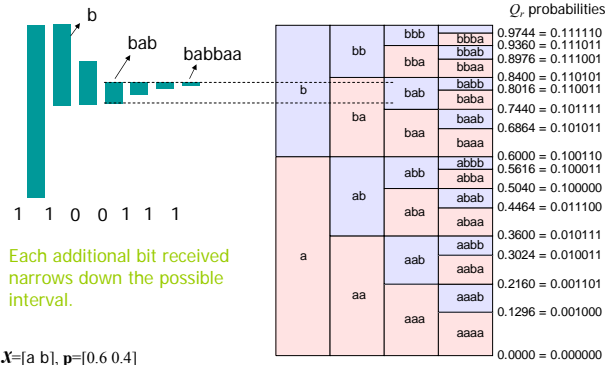
$$Q_0 = 0.0000 = 0.0000000$$

bbb	m8=1111
bba	m7=11011
bab	m6=1100
baa	m5=1010
abb	m4=10001
aba	m3=011
aab	m2=0100
aaa	m1=000

## 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



## Arithmetic Coding/Decoding

Input	Transmitter Min	Transmitter Max	Send	Receiver Min	Receiver Test	Receiver 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 digits.

## Arithmetic Coding Algorithm

Input Symbols:  $X = [a\ b], p = [p\ q]$

$[min, max]$  = Input Probability Range

Note: only keep untransmitted bits of  $min$  and  $max$

Coding Algorithm:

Initialize  $[min, max] = [000...0, 111...1]$

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:

$$X = [a\ b], \quad p_n = [1-p_n \quad p_n], \quad p_n = \frac{1 + \text{count}(x_i = b)}{1 + n}$$

## Adaptive Arithmetic Coding

$$p_n = \frac{1 + \text{count}(x_i = b)}{1 + n}$$

$$p_1 = 0.5$$

$$p_2 = 1/3 \text{ or } 2/3$$

$$p_3 = 1/4 \text{ or } 1/2 \text{ or } 3/4$$

$$p_4 = \dots$$

				1.0000 = 0.000000
b	bb	bbb	bbbb	0.8000 = 0.110011
			bbba	0.7500 = 0.110000
			bbab	0.7000 = 0.101100
			bbaa	0.6667 = 0.101010
	ba	baab	0.6167 = 0.100111	
		baaa	0.5833 = 0.100101	
		baab	0.5500 = 0.100011	
		baaa	0.5000 = 0.011111	
ab	abbb	0.4500 = 0.011100		
	abba	0.4167 = 0.011010		
	abaa	0.3833 = 0.011000		
	abaa	0.3333 = 0.010101		
a	aabb	0.3000 = 0.010011		
	aaba	0.2500 = 0.010000		
	aaab	0.2000 = 0.001100		
	aaaa	0.0000 = 0.000000		

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
- for head use enough bits for the max phrase number so far:
- decoder constructs an identical dictionary

prefix codes are underlined

## Lempel-Ziv Example

Input = 1011010100010010001001010010

Dictionary	Send	Decode
0000 $\phi$	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  $D(0), \dots, D(K-1)$ . We need to send  $M = \lceil \log K \rceil$  bits to specify a dictionary entry. Initially  $K=1$ ,  $D(0) = \phi$  = null string and  $M = \lceil \log K \rceil = 0$  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 " $\phi$ " + "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 100011011000 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  $\Rightarrow \limsup_{n \rightarrow \infty} n^{-1} l(X_{1:n}) \leq H(\mathbf{X})$  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  $H(\mathbf{X}) < 1$
- 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:  $x, y, z$

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

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

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

A Markov chain  $x \rightarrow y \rightarrow z$  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.  $I(x; z | y) = 0 \Leftrightarrow H(z | y) = H(z | x, y)$
- 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  $x \rightarrow y \rightarrow z$

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

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

Hence  $x$  and  $z$  are conditionally independent given  $y$

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

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

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

Markov chain property is symmetrical

## Data Processing Theorem

If  $x \rightarrow y \rightarrow z$  then  $I(x; y) \geq I(x; z)$

– processing  $y$  cannot add new information about  $x$

If  $x \rightarrow y \rightarrow z$  then  $I(x; y) \geq I(x; y | z)$

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

Proof:

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

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

$$\text{hence } I(x; y) = I(x; z) + I(x; y | z)$$

$$\text{so } I(x; y) \geq I(x; z) \text{ and } I(x; y) \geq I(x; y | z)$$

(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:  $z = x + y$

–  $\mathbf{X} = \mathbf{y} = [0, 1]^T$   $\mathbf{p}_X = \mathbf{p}_Y = [1/2, 1/2]^T$

–  $I(x; y) = 0$  since independent

– but  $I(x; y | z) = 1/2$

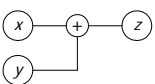
$$H(x | z) = H(y | z) = H(x, y | z)$$

$$= 0 \times 1/4 + 1 \times 1/2 + 0 \times 1/4 = 1/2$$

since in each case  $z \neq 1 \Rightarrow H() = 0$

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

$$= 1/2 + 1/2 - 0 = 1$$



		Z		
		0	1	2
XY	00	1/4		
	01		1/4	
	10		1/4	
	11			1/4

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  $\theta$  and you extract a statistic  $T(x)$  from your observation,

then  $\theta \rightarrow x \rightarrow T(x) \Rightarrow I(\theta; T(x)) \leq I(\theta; x)$

$T(x)$  is **sufficient** for  $\theta$  if the stronger condition:

$$\theta \rightarrow x \rightarrow T(x) \rightarrow \theta \Leftrightarrow I(\theta; T(x)) = I(\theta; x)$$

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

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

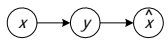
Example:  $x_i \sim \text{Bernoulli}(\theta)$ ,

$$T(x_{1:n}) = \sum_{i=1}^n x_i \quad p(X_{1:n} = x_{1:n} | \theta, \sum x_i = k) = \begin{cases} \binom{n}{k} \theta^k (1-\theta)^{n-k} & \text{if } \sum x_i = k \\ 0 & \text{if } \sum x_i \neq k \end{cases}$$

independent of  $\theta \Rightarrow$  sufficient

## Fano's Inequality

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

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


$$\Rightarrow p_e \geq \frac{(H(x | y) - H(p_e))^{(a)}}{\log(|\mathcal{X}| - 1)} \geq \frac{(H(x | y) - 1)}{\log(|\mathcal{X}| - 1)}$$

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

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

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

$$\Rightarrow H(x | y) + 0 \leq H(e) + H(x | e, y) \quad H \geq 0; H(e | y) \leq H(e)$$

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

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

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



## Fano Example

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

$\mathbf{Y} = \{1:2\}$  if  $x \leq 2$  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: 
$$p_e \geq \frac{H(X|Y) - 1}{\log(|\mathbf{X}| - 1)} = \frac{1.771 - 1}{\log(4)} = 0.3855$$

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

## Summary

- **Markov:**  $x \rightarrow y \rightarrow z \Leftrightarrow p(z|x, y) = p(z|y) \Leftrightarrow I(X; Z|Y) = 0$
- **Data Processing Theorem:** if  $x \rightarrow y \rightarrow z$  then
  - $I(X; Y) \geq I(X; Z)$
  - $I(X; Y) \geq I(X; Y|Z)$  can be false if not Markov

- **Fano's Inequality:** if  $x \rightarrow y \rightarrow \hat{x}$  then

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

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

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

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

Sample eight i.i.d. values

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

Strongly Typical  $\Rightarrow$  Typical

## Convergence of Random Numbers

- **Convergence**

$$X_n \xrightarrow{n \rightarrow \infty} Y \Rightarrow \forall \varepsilon > 0, \exists m \text{ such that } \forall n > m, |X_n - Y| < \varepsilon$$

Example:  $X_n = \pm 2^{-n}$ ,  $p = [1/2; 1/2]$

choose  $m = 1 - \log \varepsilon$

- **Convergence in probability (weaker than convergence)**

$$X_n \xrightarrow{\text{prob}} Y \Rightarrow \forall \varepsilon > 0, P(|X_n - Y| > \varepsilon) \rightarrow 0$$

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

$$\text{for any small } \varepsilon, P(|X_n| > \varepsilon) = n^{-1} \xrightarrow{n \rightarrow \infty} 0$$

Note:  $y$  can be a constant or another random variable

## Weak law of Large Numbers

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

$$- E S_n = E X = \mu \quad \text{Var } S_n = n^{-1} \text{Var } X = n^{-1} \sigma^2$$

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

**WLLN:**  $S_n \xrightarrow{\text{prob}} \mu$

$$\Leftrightarrow \forall \varepsilon > 0, P(|S_n - \mu| > \varepsilon) \xrightarrow{n \rightarrow \infty} 0$$

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

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

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

For any choice of  $\varepsilon$

- WLLN

$$S_n = \frac{1}{n} \sum_{i=1}^n x_i \text{ where } E x_i = \mu \text{ and } \text{Var } x_i = \sigma^2$$

$$\varepsilon^2 p(\underbrace{S_n - \mu}_{\text{prob}} > \varepsilon) \leq \text{Var } S_n = \frac{\sigma^2}{n} \xrightarrow{n \rightarrow \infty} 0$$

Hence  $S_n \rightarrow \mu$

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  $p(\mathbf{x}) = \prod_{i=1}^n p(x_i)$

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

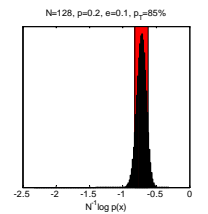
Example:

- $x_i$  Bernoulli with  $p(x_i=1)=p$

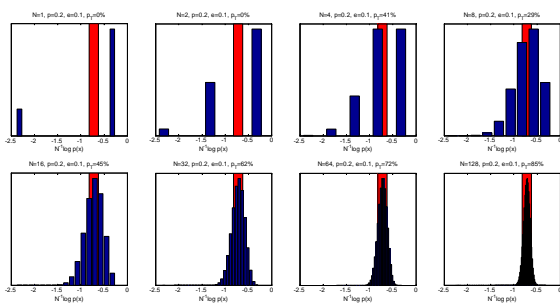
- e.g.  $p([0 \ 1 \ 1 \ 0 \ 0 \ 0])=p^2(1-p)^4$

- For  $p=0.2$ ,  $H(X)=0.72$  bits

- Red bar shows  $T_{0.1}^{(n)}$



## Typical Set Frames

[illegible]

## Typical Set: Properties

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

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

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

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

$$\text{Proof 3b: } 1 = \sum_{\mathbf{x}} p(\mathbf{x}) \geq \sum_{\mathbf{x} \in T^{(n)}_{\epsilon}} p(\mathbf{x}) \geq \sum_{\mathbf{x} \in T^{(n)}_{\epsilon}} 2^{-n(H(\mathbf{x}) + \epsilon)} = 2^{-n(H(\mathbf{x}) + \epsilon)} |T^{(n)}_{\epsilon}|$$

## Asymptotic Equipartition Principle

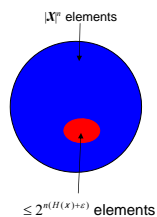
- for any  $\varepsilon$  and for  $n > N_\varepsilon$   
 “Almost all events are almost equally surprising”

- $p(\mathbf{X} \in T_\varepsilon^{(n)}) > 1 - \varepsilon$  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  $1+n\log|\mathbf{X}|$  bits
- $L$  = Average code length
 
$$\leq 2+n(H+\varepsilon)+\varepsilon(n\log|\mathbf{X}|)$$

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



## 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: 
$$L \leq n(H + \varepsilon + \varepsilon \log |\mathbf{X}| + 2n^{-1})$$
- make a code with an error probability  $< \varepsilon$  using  $H(x) + \delta$  bits for each symbol
  - just code  $T_\varepsilon$  using  $n(H + \varepsilon + n^{-1})$  bits and use a random wrong code if  $\mathbf{x} \notin T_\varepsilon$

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

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

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

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

$N_\varepsilon$  increases rapidly for small  $\varepsilon$

For this choice of  $N_\varepsilon$ , 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  $\varepsilon$  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  $|S^{(n)}| < 2^{n(H(X) - 2\varepsilon)}$

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

Answer: No

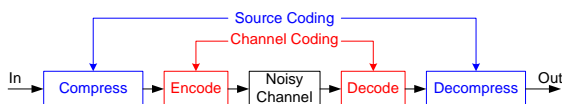
## Summary

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

## Lecture 9

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

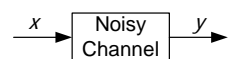
## Source and Channel Coding



- 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

$$(Q_{y|x})_{i,j} = p(y = y_j | x = x_i)$$

$$\text{Hence } \mathbf{p}_y = Q_{y|x}^T \mathbf{p}_x$$

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

- 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**  $\begin{pmatrix} 1-f & f \\ f & 1-f \end{pmatrix}$   $x \begin{matrix} 0 \rightarrow 0 \\ 1 \rightarrow 1 \end{matrix} y$   
 -  $\mathbf{X} = [0 \ 1]$ ,  $\mathbf{Y} = [0 \ 1]$
- **Binary Erasure Channel**  $\begin{pmatrix} 1-f & f & 0 \\ 0 & f & 1-f \end{pmatrix}$   $x \begin{matrix} 0 \rightarrow 0 \\ 1 \rightarrow ? \end{matrix} y$   
 -  $\mathbf{X} = [0 \ 1]$ ,  $\mathbf{Y} = [0 \ ? \ 1]$
- **Z Channel**  $\begin{pmatrix} 1 & 0 \\ f & 1-f \end{pmatrix}$   $x \begin{matrix} 0 \rightarrow 0 \\ 1 \rightarrow 1 \end{matrix} y$   
 -  $\mathbf{X} = [0 \ 1]$ ,  $\mathbf{Y} = [0 \ 1]$

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  $= |\mathbf{X}| |\mathbf{Y}|^{-1}$   
 - If  $x$  is uniform (i.e.  $p(x) = |\mathbf{X}|^{-1}$ ) then  $y$  is uniform  

$$p(y) = \sum_{x \in \mathbf{X}} p(y|x) p(x) = |\mathbf{X}|^{-1} \sum_{x \in \mathbf{X}} p(y|x) = |\mathbf{X}|^{-1} \times |\mathbf{X}| |\mathbf{Y}|^{-1} = |\mathbf{Y}|^{-1}$$
2. All rows are permutations of each other  
 - Each row of  $\mathbf{Q}$  has the same entropy so  

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

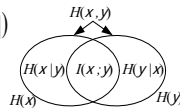
## Channel Capacity

- **Capacity** of a DMC channel:  $C = \max_{\mathbf{p}_x} I(x;y)$   
 - Maximum is over all possible input distributions  $\mathbf{p}_x$   
 -  $\exists$  only one maximum since  $I(x;y)$  is concave in  $\mathbf{p}_x$  for fixed  $\mathbf{p}_{y|x}$   $\blacklozenge$   
 - We want to find the  $\mathbf{p}_x$  that maximizes  $I(x;y)$   
 - Limits on  $C$ :  

$$0 \leq C \leq \min(H(x), H(y)) \leq \min(\log |\mathbf{X}|, \log |\mathbf{Y}|)$$

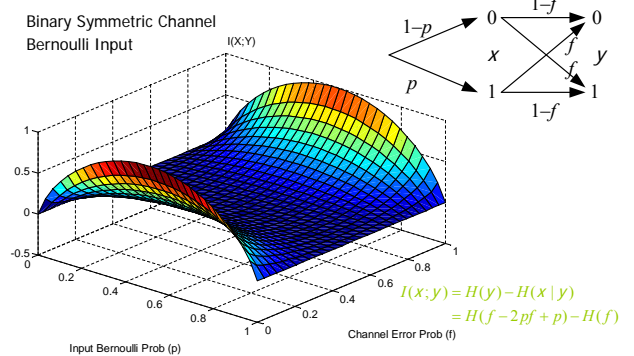
- **Capacity** for  $n$  uses of channel:

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



$\blacklozenge$  = proved in two pages time

## Mutual Information Plot



## 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

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

$\bullet$  = Deterministic

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:

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

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

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

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

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

$\bullet$  = Deterministic

## $n$ -use Channel Capacity

For Discrete Memoryless Channel:

$$\begin{aligned}
 I(X_{1:n}; Y_{1:n}) &= H(Y_{1:n}) - H(Y_{1:n} | X_{1:n}) \\
 &= \sum_{i=1}^n H(Y_i | Y_{1:i-1}) - \sum_{i=1}^n H(Y_i | X_i) \quad \text{Chain; Memoryless} \\
 &\leq \sum_{i=1}^n H(Y_i) - \sum_{i=1}^n H(Y_i | X_i) = \sum_{i=1}^n I(X_i; Y_i) \quad \text{Conditioning Reduces Entropy}
 \end{aligned}$$

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

## Binary Erasure Channel (BEC)

$$\begin{aligned}
 I(x; y) &= H(x) - H(x | y) \\
 &= H(x) - p(y=0) \times 0 - p(y=? ) H(x) - p(y=1) \times 0 \\
 &= H(x) - H(x) f \quad H(x|y) = 0 \text{ when } y=0 \text{ or } y=1 \\
 &= (1-f) H(x) \\
 &\leq 1-f \quad \text{since max value of } H(x) = 1 \text{ with equality when } x \text{ is uniform}
 \end{aligned}$$

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

## Lecture 10

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

## Capacity of Symmetric Channel

If channel is weakly symmetric:

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

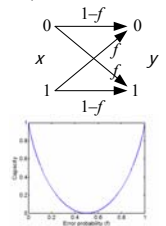
with equality iff input distribution is uniform

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

For a binary symmetric channel (BSC):

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

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



## Asymmetric Channel Capacity

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

$$\begin{aligned}
 H(y) &= -2a \log a - (1-2a) \log(1-2a) \\
 H(y|x) &= 2aH(f) + (1-2a)H(1) = 2aH(f)
 \end{aligned}$$

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

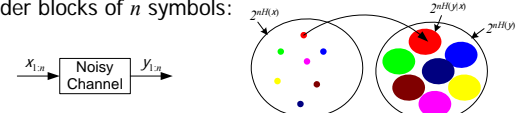
$$\begin{aligned}
 I &= -2a \log a - (1-2a) \log(1-2a) - 2aH(f) \\
 \frac{dI}{da} &= -2 \log e - 2 \log a + 2 \log e + 2 \log(1-2a) - 2H(f) = 0 \\
 \log \frac{1-2a}{a} &= \log(a^{-1} - 2) = H(f) \Rightarrow a = (2 + 2^{H(f)})^{-1} \\
 \Rightarrow C &= -2a \log(a 2^{H(f)}) - (1-2a) \log(1-2a) = -\log(1-2a)
 \end{aligned}$$

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  
 $f=1/2 \Rightarrow H(f)=1 \Rightarrow a=1/4 \Rightarrow C=\log 2 = 1$  bits/use

## Significance of Mutual Information

- Consider blocks of  $n$  symbols:



- An average input sequence  $x_{1:n}$  corresponds to about  $2^{nH(X)}$  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  $2^{n(H(Y) - H(Y|X))} = 2^{nI(Y; X)}$

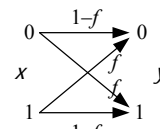
Channel Coding Theorem: for large  $n$  can transmit at any rate  $< C$  with negligible errors

## Jointly Typical Set

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

## Jointly Typical Example

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



all combinations of  $x$  and  $y$  have exactly the right frequencies

## Jointly Typical Diagram

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

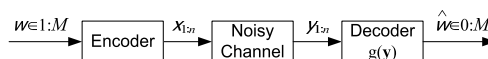
## Joint Typical Set Properties

$$\begin{aligned} \text{Proof 3: } 1 - \varepsilon &< \sum_{\mathbf{x}, \mathbf{y} \in J_\varepsilon^{(n)}} p(\mathbf{x}, \mathbf{y}) \leq \left| J_\varepsilon^{(n)} \right| \max_{\mathbf{x}, \mathbf{y} \in J_\varepsilon^{(n)}} p(\mathbf{x}, \mathbf{y}) = \left| J_\varepsilon^{(n)} \right| 2^{-n(H(\mathbf{x}, \mathbf{y}) - \varepsilon)} \quad n > N_\varepsilon \\ 1 &\geq \sum_{\mathbf{x}, \mathbf{y} \in J_\varepsilon^{(n)}} p(\mathbf{x}, \mathbf{y}) \geq \left| J_\varepsilon^{(n)} \right| \min_{\mathbf{x}, \mathbf{y} \in J_\varepsilon^{(n)}} p(\mathbf{x}, \mathbf{y}) = \left| J_\varepsilon^{(n)} \right| 2^{-n(H(\mathbf{x}, \mathbf{y}) + \varepsilon)} \quad \forall n \end{aligned}$$

## Joint AEP

$$\geq (1-\varepsilon)2^{-n(I(X;Y)+3\varepsilon)} \quad \text{for } n > N_\varepsilon$$

## Channel Codes

 $\delta_C = 1$  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  $(\lceil 2^{nR} \rceil, n)$  codes for  $n=1, 2, \dots$
  - max prob of error  $\lambda^{(n)} \rightarrow 0$  as  $n \rightarrow \infty$
  - Note:** we will normally write  $(2^{nR}, n)$  to mean  $(\lceil 2^{nR} \rceil, n)$
- 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

## Lecture 11

- Channel Coding Theorem

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

- Choose  $\epsilon \approx$  error prob, joint typicality  $\Rightarrow N_\epsilon$ , choose  $n > N_\epsilon$
- Choose  $\mathbf{p}_x$  so that  $I(x; y) = C$ , the information capacity
- Use  $\mathbf{p}_x$  to **choose a code**  $\mathbf{c}$  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  $\geq 2$  possible  $k$ 's then 1 is the decoded message
  - we calculate error probability averaged over all  $\mathbf{c}$  and all  $W$

$$p(\mathbf{E}) = \sum_{\mathbf{c}} p(\mathbf{c}) 2^{-nR} \sum_{w=1}^{2^{nR}} \lambda_w(\mathbf{c}) = 2^{-nR} \sum_{w=1}^{2^{nR}} \sum_{\mathbf{c}} p(\mathbf{c}) \lambda_w(\mathbf{c}) \stackrel{(a)}{=} \sum_{\mathbf{c}} p(\mathbf{c}) \lambda_1(\mathbf{c}) = p(\mathbf{E} | W=1)$$

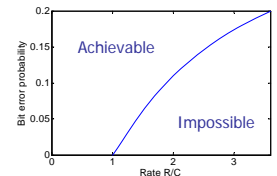
(a) since error averaged over all possible codes is independent of  $w$

## 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  $\lambda^{(n)} \rightarrow 0$  as  $n \rightarrow \infty$
  - Any sequence of  $(2^{nR}, n)$  codes with max prob of error  $\lambda^{(n)} \rightarrow 0$  as  $n \rightarrow \infty$  must have  $R \leq C$

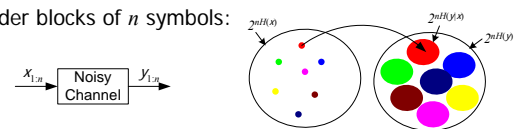
A very counterintuitive result:

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



## Channel Coding Principle

- Consider blocks of  $n$  symbols:

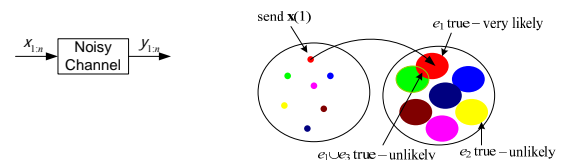


- An average input sequence  $x_{1:n}$  corresponds to about  $2^{nH(Y|X)}$  typical output sequences
- Random Codes:** Choose  $2^{nR}$  random code vectors  $\mathbf{x}(w)$ 
  - their typical output sequences are unlikely to overlap much.
- Joint Typical Decoding:** A received vector  $\mathbf{y}$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

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

## Channel Coding Principle

- Assume we transmit  $\mathbf{x}(1)$  and receive  $\mathbf{y}$
- Define the events  $e_w = \{\mathbf{x}(w), \mathbf{y} \in J_\epsilon^{(n)}\}$  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  $p(e_w \text{ true}) < 2^{-n(I(x,y) - 3\epsilon)}$  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  $w \geq 2$

$$p(\mathbf{E} | W=1) = p(\bar{e}_1 \cup e_2 \cup e_3 \cup \dots \cup e_{2^{nR}}) \leq p(\bar{e}_1) + \sum_{w=2}^{2^{nR}} p(e_w) \quad p(A \cup B) \leq p(A) + p(B)$$

$$\leq \varepsilon + \sum_{i=2}^{2^{nR}} 2^{-n(I(x;y) - 3\varepsilon)} = \varepsilon + 2^{nR} 2^{-n(I(x;y) - 3\varepsilon)} \quad (1) \text{ Joint typicality}$$

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

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

## Code Selection & Expurgation

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

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

$K = \text{num of codes}$

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

Proof: Assume  $\lambda_w$  are in descending order

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

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

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

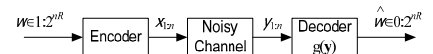
## Summary of Procedure

- 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  $\leq 2\varepsilon$  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  $\leq 4\varepsilon$  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:  $\varepsilon$  determines both error probability and 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

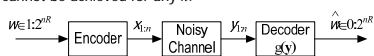
- Fano's Inequality:** If  $P_e^{(n)}$  is error prob when estimating  $w$  from  $\mathbf{y}$ ,  $H(w | \mathbf{y}) \leq 1 + P_e^{(n)} \log |\mathcal{W}| = 1 + nRP_e^{(n)}$
- Hence  $nR = H(w) = H(w | \mathbf{y}) + I(w; \mathbf{y})$  Definition of  $I$ 

$$\leq H(w | \mathbf{y}) + I(\mathbf{x}(w); \mathbf{y}) \quad \text{Markov: } w \rightarrow \mathbf{x} \rightarrow \mathbf{y} \rightarrow \hat{w}$$

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

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

$$\Rightarrow P_e^{(n)} \geq \frac{R - C - n^{-1}}{R} \xrightarrow{n \rightarrow \infty} \frac{R - C}{R}$$
- 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  $P_b = E\{p(v_i \neq \hat{v}_i)\} = E\{p(e_i)\}$

Then  $nC \geq I(x_{1:n}; y_{1:n}) \geq I(v_{1:nR}; \hat{v}_{1:nR}) = H(v_{1:nR}) - H(v_{1:nR} | \hat{v}_{1:nR})$

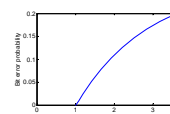
$$= nR - \sum_{i=1}^{nR} H(v_i | \hat{v}_{1:nR}, v_{1:nR}) \stackrel{(b)}{\geq} nR - \sum_{i=1}^{nR} H(v_i | \hat{v}_i) = nR(1 - E\{H(v_i | \hat{v}_i)\})$$

$$\stackrel{(d)}{=} nR(1 - E\{H(e_i | \hat{v}_i)\}) \stackrel{(c)}{\geq} nR(1 - E\{H(e_i)\}) \stackrel{(a)}{\geq} nR(1 - H(P_b)) = nR(1 - H(P_b))$$

Hence

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

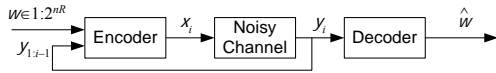
$$P_b \geq H^{-1}(1 - C/R)$$



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



## 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  $\exists$  a sequence of  $(2^{nR}, n)$  feedback codes such that  $P_e^{(n)} = P(\hat{w} \neq w) \xrightarrow{n \rightarrow \infty} 0$
- Feedback capacity**,  $C_{FB} \geq C$ , is the sup of achievable rates

## Capacity with Feedback

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

Hence

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

Fano

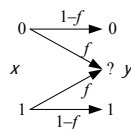
$$\Rightarrow P_e^{(n)} \geq \frac{R - C - n^{-1}}{R}$$

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:

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



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

## Joint Source-Channel Coding



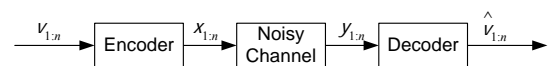
- Assume  $v_i$  satisfies AEP and  $|\mathbf{V}| < \infty$ 
  - Examples: i.i.d.; markov; stationary ergodic
- Capacity of DMC channel is  $C$ 
  - if time-varying:  $C = \lim_{n \rightarrow \infty} n^{-1} I(\mathbf{x}; \mathbf{y})$
- Joint Source Channel Coding Theorem:**
  - $\exists$  codes with  $P_e^{(n)} = P(\hat{v}_{1:n} \neq v_{1:n}) \xrightarrow{n \rightarrow \infty} 0$  iff  $H(\mathbf{V}) < C$ 
    - 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  $2^{n(H(\mathbf{V}) + \varepsilon)}$   $\mathbf{v}$ 's in the typical set: encode using  $n(H(\mathbf{V}) + \varepsilon)$  bits
  - encoder error  $< \varepsilon$
- Transmit with error prob less than  $\varepsilon$  so long as  $H(\mathbf{V}) + \varepsilon < C$
- Total error prob  $< 2\varepsilon$

## Source-Channel Proof ( $\Rightarrow$ )



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

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

Let  $n \rightarrow \infty \Rightarrow P_e^{(n)} \rightarrow 0 \Rightarrow H(\mathbf{V}) \leq C$

## 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:  $y = g(x) \Leftrightarrow x = g^{-1}(y)$ 

$$F_y(y) = F_x(g^{-1}(y)) \text{ or } 1 - F_x(g^{-1}(y)) \quad \text{according to slope of } g(x)$$

$$f_y(y) = \frac{dF_y(y)}{dy} = f_x(g^{-1}(y)) \left| \frac{dg^{-1}(y)}{dy} \right| = f_x(x) \left| \frac{dx}{dy} \right| \text{ where } x = g^{-1}(y)$$
- 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)  $z = x^4 \Rightarrow x = z^{1/4} \Rightarrow f_z(z) = 0.5 \times \frac{1}{4} z^{-3/4} = 0.125 z^{-3/4}$  for  $z \in (0, 16)$

## Joint Distributions

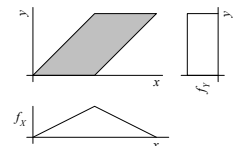
Joint pdf:  $f_{x,y}(x, y)$   
 Marginal pdf:  $f_x(x) = \int_{-\infty}^{\infty} f_{x,y}(x, y) dy$   
 Independence:  $\Leftrightarrow f_{x,y}(x, y) = f_x(x) f_y(y)$   
 Conditional pdf:  $f_{x|y}(x) = \frac{f_{x,y}(x, y)}{f_y(y)}$

### Example:

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

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

$$f_{y|x} = \frac{1}{\min(x, 1-x)} \text{ for } y \in (\max(0, x-1), \min(x, 1))$$



## Quantised Random Variables

- Given a continuous pdf  $f(x)$ , we divide the range of  $x$  into bins of width  $\Delta$ 
  - 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  $x_i$
- $H(Y) = -\sum_i f(x_i)\Delta \log(f(x_i)\Delta)$ 

$$= -\log \Delta - \sum_i f(x_i) \log(f(x_i)\Delta)$$

$$\xrightarrow{\Delta \rightarrow 0} -\log \Delta - \int_{-\infty}^{\infty} f(x) \log f(x) dx = -\log \Delta + h(x)$$
- Differential entropy:  $h(x) = -\int_{-\infty}^{\infty} f_x(x) \log f_x(x) dx$

## Differential Entropy

Differential Entropy:  $h(x) = -\int_{-\infty}^{\infty} f_x(x) \log f_x(x) dx = E - \log f_x(x)$

### 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:**  $X \sim N(\mu, \sigma^2)$ 
  - $f(x) = (2\pi\sigma^2)^{-1/2} \exp(-1/2(x-\mu)^2/\sigma^2)$
  - $h(X) = -(\log e) \int_{-\infty}^{\infty} f(x) \ln f(x) dx$ 

$$= -(\log e) \int_{-\infty}^{\infty} f(x) \left( -1/2 \ln(2\pi\sigma^2) - 1/2(x-\mu)^2/\sigma^2 \right)$$

$$= 1/2(\log e) (\ln(2\pi\sigma^2) + \sigma^{-2} E((x-\mu)^2))$$

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

## Multivariate Gaussian

Given mean,  $\mathbf{m}$ , and symmetric +ve definite covariance matrix  $\mathbf{K}$ ,  
 $X_{1:n} \sim \mathbf{N}(\mathbf{m}, \mathbf{K}) \Leftrightarrow f(\mathbf{x}) = |2\pi\mathbf{K}|^{-1/2} \exp(-1/2(\mathbf{x}-\mathbf{m})^T \mathbf{K}^{-1}(\mathbf{x}-\mathbf{m}))$

$$\begin{aligned} h(f) &= -(\log e) \int f(\mathbf{x}) \times \left( -1/2(\mathbf{x}-\mathbf{m})^T \mathbf{K}^{-1}(\mathbf{x}-\mathbf{m}) - 1/2 \ln|2\pi\mathbf{K}| \right) d\mathbf{x} \\ &= 1/2 \log(e) \times (\ln|2\pi\mathbf{K}| + E((\mathbf{x}-\mathbf{m})^T \mathbf{K}^{-1}(\mathbf{x}-\mathbf{m}))) \\ &= 1/2 \log(e) \times (\ln|2\pi\mathbf{K}| + E \text{tr}((\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^T \mathbf{K}^{-1}))) \\ &= 1/2 \log(e) \times (\ln|2\pi\mathbf{K}| + \text{tr}(E(\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^T \mathbf{K}^{-1}))) \\ &= 1/2 \log(e) \times (\ln|2\pi\mathbf{K}| + \text{tr}(\mathbf{K}\mathbf{K}^{-1})) = 1/2 \log(e) \times (\ln|2\pi\mathbf{K}| + n) \\ &= 1/2 \log(e^n) + 1/2 \log(|2\pi\mathbf{K}|) \\ &= 1/2 \log(2\pi e \mathbf{K}) \text{ bits} \end{aligned}$$

## Other Differential Quantities

### Joint Differential Entropy

$$h(X, Y) = -\iint_{x,y} f_{x,y}(x, y) \log f_{x,y}(x, y) dx dy = E - \log f_{x,y}(x, y)$$

### Conditional Differential Entropy

$$h(X|Y) = -\iint_{x,y} f_{x,y}(x, y) \log f_{x,y}(x|y) dx dy = h(X, Y) - h(Y)$$

### Mutual Information

$$I(X; Y) = \iint_{x,y} f_{x,y}(x, y) \log \frac{f_{x,y}(x, y)}{f_x(x)f_y(y)} dx dy = h(X) + h(Y) - h(X, Y)$$

### Relative Differential Entropy of two pdf's:

$$\begin{aligned} D(f \| g) &= \int f(x) \log \frac{f(x)}{g(x)} dx && \text{(a) must have } f(x)=0 \Rightarrow g(x)=0 \\ &= -h_f(X) - E_f \log g(X) && \text{(b) continuity } \Rightarrow 0 \log(0/0) = 0 \end{aligned}$$

## Differential Entropy Properties

**Chain Rules**  $h(X, Y) = h(X) + h(Y|X) = h(Y) + h(X|Y)$   
 $I(X, Y; Z) = I(X; Z) + I(Y; Z|X)$

**Information Inequality:**  $D(f \| g) \geq 0$

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

$$\begin{aligned} -D(f \| g) &= \int_{x \in S} f(x) \log \frac{g(x)}{f(x)} dx = E \left( \log \frac{g(\mathbf{X})}{f(\mathbf{X})} \right) \\ &\leq \log \left( E \frac{g(\mathbf{X})}{f(\mathbf{X})} \right) = \log \left( \int_S f(x) \frac{g(x)}{f(x)} dx \right) \quad \text{Jensen + log() is concave} \\ &= \log \left( \int_S g(x) dx \right) \leq \log 1 = 0 \end{aligned}$$

all the same as for  $H()$

## Information Inequality Corollaries

**Mutual Information  $\geq 0$**

$$I(X; Y) = D(f_{x,y} \| f_x f_y) \geq 0$$

**Conditioning reduces Entropy**

$$h(X) - h(X|Y) = I(X; Y) \geq 0$$

**Independence Bound**

$$h(X_{1:n}) = \sum_{i=1}^n h(X_i | X_{1:i-1}) \leq \sum_{i=1}^n h(X_i)$$

all the same as for  $H()$

## Change of Variable

**Change Variable:**  $Y = g(X)$

from earlier  $f_Y(y) = f_X(g^{-1}(y)) \left| \frac{dg^{-1}(y)}{dy} \right|$

$$\begin{aligned} h(Y) &= -E \log(f_Y(Y)) = -E \log(f_X(g^{-1}(Y))) - E \log \left| \frac{dx}{dy} \right| \\ &= -E \log(f_X(X)) - E \log \left| \frac{dx}{dy} \right| = h(X) - E \log \left| \frac{dx}{dy} \right| \end{aligned}$$

**Examples:**

- Translation:  $Y = X + a \Rightarrow dy/dx = 1 \Rightarrow h(Y) = h(X)$
- Scaling:  $Y = cX \Rightarrow dy/dx = c \Rightarrow h(Y) = h(X) - \log|c^{-1}|$
- Vector version:  $Y_{1:n} = \mathbf{A}X_{1:n} \Rightarrow h(\mathbf{Y}) = h(\mathbf{X}) + \log|\det(\mathbf{A})|$

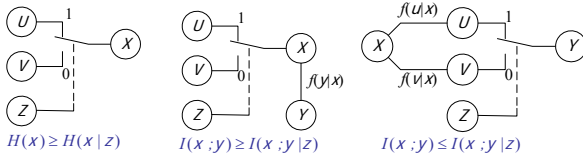
not the same as for  $H()$

## Concavity & Convexity

- **Differential Entropy:**
  - $h(x)$  is a **concave** function of  $f_X(x) \Rightarrow \exists$  a maximum
- **Mutual Information:**
  - $I(x; y)$  is a **concave** function of  $f_X(x)$  for fixed  $f_{Y|X}(y)$
  - $I(x; y)$  is a **convex** function of  $f_{Y|X}(y)$  for fixed  $f_X(x)$

**Proofs:**

Exactly the same as for the **discrete case**:  $\mathbf{p}_Z = [1-\lambda, \lambda]^T$



## 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  $\phi(\mathbf{x}) = |\mathbf{K}|^{-1/2} \exp(-\frac{1}{2}\mathbf{x}^T \mathbf{K}^{-1} \mathbf{x})$

**Proof:**  $0 \leq D(f \parallel \phi) = -h_f(\mathbf{x}) - E_f \log \phi(\mathbf{x})$

$$\begin{aligned} \Rightarrow h_f(\mathbf{x}) &\leq -(\log e) E_f \left( -\frac{1}{2} \ln(2\pi\mathbf{K}) - \frac{1}{2} \mathbf{x}^T \mathbf{K}^{-1} \mathbf{x} \right) \\ &= \frac{1}{2} (\log e) \left( \ln(2\pi\mathbf{K}) + \text{tr}(E_f \mathbf{x} \mathbf{x}^T \mathbf{K}^{-1}) \right) \\ &= \frac{1}{2} (\log e) \left( \ln(2\pi\mathbf{K}) + \text{tr}(\mathbf{I}) \right) \quad E_f \mathbf{x} \mathbf{x}^T = \mathbf{K} \\ &= \frac{1}{2} \log(2\pi e \mathbf{K}) = h_\phi(\mathbf{x}) \quad \text{tr}(\mathbf{I}) = n = \ln(e^n) \end{aligned}$$

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

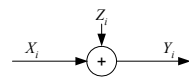
## Capacity of Gaussian Channel

**Discrete-time channel:**  $y_i = x_i + z_i$

– Zero-mean Gaussian i.i.d.  $z_i \sim N(0, N)$

– Average power constraint  $n^{-1} \sum_{i=1}^n x_i^2 \leq P$

$$E y^2 = E(x+z)^2 = E x^2 + 2E(x)E(z) + E z^2 \leq P + N \quad X, Z \text{ indep and } E Z = 0$$



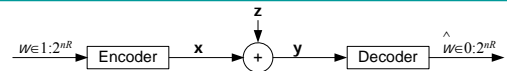
**Information Capacity**

– Define **information capacity**:  $C = \max_{E X^2 \leq P} I(X; Y)$

$$\begin{aligned} I(x; y) &= h(y) - h(y|x) = h(y) - h(x+z|x) \\ &\stackrel{(a)}{=} h(y) - h(z|x) = h(y) - h(z) \quad (a) \text{ Translation independence} \\ &\leq \frac{1}{2} \log 2\pi e(P+N) - \frac{1}{2} \log 2\pi e N \quad \text{Gaussian Limit with equality when } x \sim N(0, P) \\ &= \frac{1}{2} \log(1 + PN^{-1}) = \frac{1}{2} \log \left[ \frac{P+N}{N} \right]_{\text{dB}} \end{aligned}$$

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  $\mathbf{x}(w) \in \mathbb{R}^n$  for  $w=1:M$  with  $\mathbf{x}(w)^T \mathbf{x}(w) \leq nP \quad \forall w$
  - A deterministic decoder  $g(\mathbf{y}) \in 0:M$  where 0 denotes failure
  - **Errors:** codeword:  $\lambda_i$      $\max_i: \lambda_i^{(n)}$     average:  $P_e^{(n)}$
- Rate  $R$  is **achievable** if  $\exists$  seq of  $(2^{nR}, n)$  codes with  $\lambda^{(n)} \xrightarrow{n \rightarrow \infty} 0$
- **Theorem:**  $R$  achievable iff  $R < C = \frac{1}{2} \log(1 + PN^{-1})$  ♦

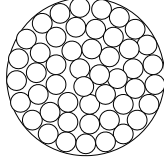
♦ = proved on next pages

## Sphere Packing

- Each transmitted  $\mathbf{x}_i$  is received as a probabilistic cloud  $\mathbf{y}_i$ 
  - cloud 'radius' =  $\sqrt{\text{Var}(\mathbf{y}|\mathbf{x})} = \sqrt{nN}$
- 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:

$$\frac{(nP + nN)^{1/2n}}{(nN)^{1/2n}} = 2^{\frac{1}{2}n \log(1 + PN^{-1})}$$

- Max rate is  $\frac{1}{2} \log(1 + PN^{-1})$



## Continuous AEP Proof

**Proof 1:** By weak law of large numbers

$$-n^{-1} \log f(\mathbf{x}_{1:n}) = -n^{-1} \sum_{i=1}^n \log f(\mathbf{x}_i) \xrightarrow{\text{prob}} E - \log f(\mathbf{x}) = h(\mathbf{x})$$

Reminder:  $\mathbf{x}_n \rightarrow \mathbf{y} \xrightarrow{\text{prob}} \forall \varepsilon > 0, \exists N_\varepsilon$  such that  $\forall n > N_\varepsilon, P(\|\mathbf{x}_n - \mathbf{y}\| > \varepsilon) < \varepsilon$

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

$$\leq 2^{-n(h(\mathbf{x}) - \varepsilon)} \int_{T_\varepsilon^{(n)}} d\mathbf{x} = 2^{-n(h(\mathbf{x}) - \varepsilon)} \text{Vol}(T_\varepsilon^{(n)})$$

max  $f(\mathbf{x})$  within  $T$

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

$$\geq 2^{-n(h(\mathbf{x}) + \varepsilon)} \int_{T_\varepsilon^{(n)}} d\mathbf{x} = 2^{-n(h(\mathbf{x}) + \varepsilon)} \text{Vol}(T_\varepsilon^{(n)})$$

min  $f(\mathbf{x})$  within  $T$

## Gaussian Channel Coding Theorem

$R$  is achievable iff  $R < C = \frac{1}{2} \log(1 + PN^{-1})$

**Proof ( $\Leftarrow$ ):**

Choose  $\varepsilon > 0$

**Random codebook:**  $\mathbf{x}_w \in \mathcal{R}^n$  for  $w = 1: 2^{nR}$  where  $\mathbf{x}_w$  are i.i.d.  $\sim \mathcal{N}(0, P - \varepsilon)$

Use **Joint typicality decoding**

- Errors:**
- Power too big  $p(\mathbf{x}^T \mathbf{x} > nP) \rightarrow 0 \Rightarrow \leq \varepsilon$  for  $n > M_\varepsilon$
  - $\mathbf{y}$  not J.T. with  $\mathbf{x}$   $p(\mathbf{x}, \mathbf{y} \notin J_\varepsilon^{(n)}) < \varepsilon$  for  $n > N_\varepsilon$
  - another  $\mathbf{x}$  J.T. with  $\mathbf{y}$   $\sum_{j=2}^{2^{nR}} p(\mathbf{x}_j, \mathbf{y}_i \in J_\varepsilon^{(n)}) \leq (2^{nR} - 1) \times 2^{-n(I(\mathbf{x}; \mathbf{y}) - 3\varepsilon)}$

**Total Err**  $P_\varepsilon^{(n)} \leq \varepsilon + \varepsilon + 2^{-n(I(\mathbf{x}; \mathbf{y}) - R - 3\varepsilon)} \leq 3\varepsilon$  for large  $n$  if  $R < I(\mathbf{x}; \mathbf{y}) - 3\varepsilon$

**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  $\mathbf{x}_i$ ;  $\mathbf{x}_i^T \mathbf{x}_i > nP \Rightarrow \lambda_i = 1$

## Continuous AEP

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

For any  $\varepsilon > 0$  and any  $n$ , the **typical set** with respect to  $f(\mathbf{x})$  is

$$T_\varepsilon^{(n)} = \left\{ \mathbf{x} \in \mathcal{S}^n : \left| -n^{-1} \log f(\mathbf{x}) - h(\mathbf{x}) \right| \leq \varepsilon \right\}$$

where  $\mathcal{S}$  is the **support** of  $f \Leftrightarrow \{\mathbf{x} : f(\mathbf{x}) > 0\}$

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

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

**Typical Set Properties**

- $p(\mathbf{x} \in T_\varepsilon^{(n)}) > 1 - \varepsilon$  for  $n > N_\varepsilon$
- $(1 - \varepsilon) 2^{n(h(\mathbf{x}) - \varepsilon)} \leq \text{Vol}(T_\varepsilon^{(n)}) \leq 2^{n(h(\mathbf{x}) + \varepsilon)}$

Proof: WLLN

Proof: Integrate max/min prob

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

## Jointly Typical Set

**Jointly Typical:**  $x_i, y_i$  i.i.d from  $\mathcal{R}^2$  with  $f_{x,y}(x_i, y_i)$

$$J_\varepsilon^{(n)} = \left\{ \mathbf{x}, \mathbf{y} \in \mathcal{R}^{2n} : \begin{aligned} & \left| -n^{-1} \log f_{\mathbf{x}}(\mathbf{x}) - h(\mathbf{x}) \right| < \varepsilon, \\ & \left| -n^{-1} \log f_{\mathbf{y}}(\mathbf{y}) - h(\mathbf{y}) \right| < \varepsilon, \\ & \left| -n^{-1} \log f_{\mathbf{x}, \mathbf{y}}(\mathbf{x}, \mathbf{y}) - h(\mathbf{x}, \mathbf{y}) \right| < \varepsilon \end{aligned} \right\}$$

**Properties:**

- Indiv p.d.:  $\mathbf{x}, \mathbf{y} \in J_\varepsilon^{(n)} \Rightarrow \log f_{\mathbf{x}, \mathbf{y}}(\mathbf{x}, \mathbf{y}) = -nh(\mathbf{x}, \mathbf{y}) \pm n\varepsilon$
- Total Prob:  $p(\mathbf{x}, \mathbf{y} \in J_\varepsilon^{(n)}) > 1 - \varepsilon$  for  $n > N_\varepsilon$
- Size:  $(1 - \varepsilon) 2^{n(h(\mathbf{x}, \mathbf{y}) - \varepsilon)} \leq \text{Vol}(J_\varepsilon^{(n)}) \leq 2^{n(h(\mathbf{x}, \mathbf{y}) + \varepsilon)}$
- Indep  $\mathbf{x}', \mathbf{y}'$ :  $(1 - \varepsilon) 2^{-n(I(\mathbf{x}, \mathbf{y}) + 3\varepsilon)} \leq p(\mathbf{x}', \mathbf{y}' \in J_\varepsilon^{(n)}) \leq 2^{-n(I(\mathbf{x}, \mathbf{y}) - 3\varepsilon)}$

Proof of 4.: Integrate max/min  $f(\mathbf{x}', \mathbf{y}') = f(\mathbf{x}')f(\mathbf{y}')$ , then use known bounds on  $\text{Vol}(J)$

## Gaussian Channel Coding Theorem

**Proof ( $\Rightarrow$ ):** Assume  $P_\varepsilon^{(n)} \rightarrow 0$  and  $n^{-1} \mathbf{x}^T \mathbf{x} < P$  for each  $\mathbf{x}(w)$

$$\begin{aligned} nR &= H(\mathbf{w}) = I(\mathbf{w}; \mathbf{y}_{1:n}) + H(\mathbf{w} | \mathbf{y}_{1:n}) \xrightarrow{\text{We 12.4}} \xrightarrow{\text{Encoder}} \xrightarrow{\mathbf{y}_{1:n}} \xrightarrow{\text{Noisy Channel}} \xrightarrow{\mathbf{y}_{1:n}} \xrightarrow{\text{Decoder}} \xrightarrow{\hat{\mathbf{w}}_{1:n}} \\ &\leq I(\mathbf{x}_{1:n}; \mathbf{y}_{1:n}) + H(\mathbf{w} | \mathbf{y}_{1:n}) \\ &= h(\mathbf{y}_{1:n}) - h(\mathbf{y}_{1:n} | \mathbf{x}_{1:n}) + H(\mathbf{w} | \mathbf{y}_{1:n}) \\ &\leq \sum_{i=1}^n h(y_i) - h(\mathbf{z}_{1:n}) + H(\mathbf{w} | \mathbf{y}_{1:n}) \quad \text{Data Proc Inequal} \\ &\leq \sum_{i=1}^n I(x_i; y_i) + 1 + nR P_\varepsilon^{(n)} \quad \text{Indep Bound + Translation} \\ &\leq \sum_{i=1}^n \frac{1}{2} \log(1 + PN^{-1}) + 1 + nR P_\varepsilon^{(n)} \quad \text{Z i.i.d + Fano, } |\mathbf{w}|^2 = 2^{nR} \\ &\leq \frac{1}{2} \log(1 + PN^{-1}) + 1 + nR P_\varepsilon^{(n)} \quad \text{max Information Capacity} \\ R &\leq \frac{1}{2} \log(1 + PN^{-1}) + n^{-1} + R P_\varepsilon^{(n)} \rightarrow \frac{1}{2} \log(1 + PN^{-1}) \end{aligned}$$

## 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  $\leq PT$ 
    - Energy constraint per coefficient:  $n^{-1}\mathbf{x}^T\mathbf{x} < PT/2WT = \frac{1}{2}W^{-1}P$
- Capacity**:  $C = \frac{1}{2} \log \left( 1 + \frac{1}{2}W^{-1}P \left( \frac{1}{2}N_0 \right)^{-1} \right) \times 2W$   
 $= W \log \left( 1 + N_0^{-1}W^{-1}P \right)$  bits/second

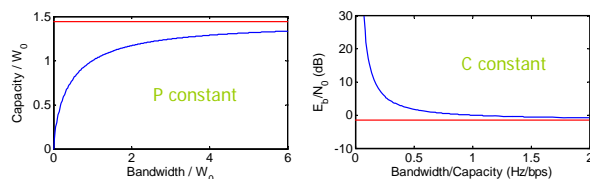
Compare discrete time version:  $\frac{1}{2} \log(1 + PN^{-1})$  bits per channel use

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

Define:  $W_0 = PN_0^{-1} \Rightarrow C/W_0 = (W/W_0) \log(1 + (W/W_0)^{-1}) \xrightarrow{W \rightarrow \infty} \log e$   
 $\Rightarrow C^{-1}W_0 = E_b N_0^{-1} \xrightarrow{W \rightarrow \infty} \ln 2 = -1.6 \text{ dB}$

- 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)

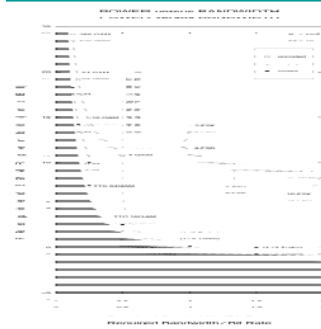
### Coding Theorem: Nearly all codes are very good

- 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)

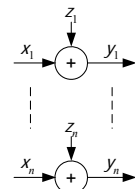
Diagram from "An overview of Communications" by John R Barry, TTC = Japanese Technology Telecommunications Council

## 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  $E \mathbf{x}^T \mathbf{x} \leq P$
- Information Capacity:  $C = \max_{f(\mathbf{x}): E \mathbf{x}^T \mathbf{x} \leq P} I(\mathbf{x}; \mathbf{y})$
- $R < C \Leftrightarrow R$  achievable
  - proof as before
- What is the optimal  $\underline{f}(\mathbf{x})$ ?



## Parallel Gaussian: Max Capacity

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

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

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

$$\stackrel{(a)}{\leq} \sum_{i=1}^n (h(y_i) - h(z_i)) \stackrel{(b)}{\leq} \sum_{i=1}^n \frac{1}{2} \log(1 + P_i N_i^{-1})$$

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

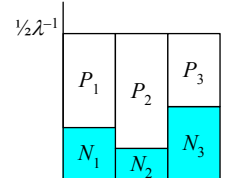
## Parallel Gaussian: Optimal Powers

We need to find the  $P_i$  that maximise  $\log(e) \sum_{i=1}^n \frac{1}{2} \log(1 + P_i N_i^{-1})$   
 - subject to power constraint  $\sum_{i=1}^n P_i = P$   
 - use Lagrange multiplier

$$J = \sum_{i=1}^n \frac{1}{2} \log(1 + P_i N_i^{-1}) - \lambda \sum_{i=1}^n P_i$$

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

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



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

## Very Noisy Channels

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

Kuhn Tucker Conditions:

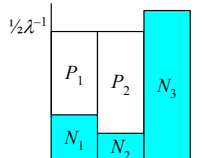
(not examinable)

- 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  $J(\mathbf{x}) = f(\mathbf{x}) - \sum_{i=1}^M \mu_i g_i(\mathbf{x}) - \lambda^T \mathbf{Ax}$

- Solution  $\mathbf{x}_0, \lambda, \mu_i$  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  $E \mathbf{z} \mathbf{z}^T = \mathbf{K}_Z$  and  $E \mathbf{x} \mathbf{x}^T = \mathbf{K}_X$
- We want to find  $\mathbf{K}_X$  to maximize capacity subject to power constraint:  $E \sum_{i=1}^n x_i^2 \leq nP \Leftrightarrow \text{tr}(\mathbf{K}_X) \leq nP$

- Find noise eigenvectors:  $\mathbf{K}_Z = \mathbf{Q} \mathbf{D} \mathbf{Q}^T$  with  $\mathbf{Q} \mathbf{Q}^T = \mathbf{I}$

- Now  $\mathbf{Q}^T \mathbf{y} = \mathbf{Q}^T \mathbf{x} + \mathbf{Q}^T \mathbf{z} = \mathbf{Q}^T \mathbf{x} + \mathbf{w}$   
 where  $E \mathbf{w} \mathbf{w}^T = E \mathbf{Q}^T \mathbf{z} \mathbf{z}^T \mathbf{Q} = E \mathbf{Q}^T \mathbf{K}_Z \mathbf{Q} = \mathbf{D}$  is diagonal  
 •  $\Rightarrow W_i$  are now independent

- Power constraint is unchanged  $\text{tr}(\mathbf{Q}^T \mathbf{K}_X \mathbf{Q}) = \text{tr}(\mathbf{K}_X \mathbf{Q} \mathbf{Q}^T) = \text{tr}(\mathbf{K}_X)$

- Choose  $\mathbf{Q}^T \mathbf{K}_X \mathbf{Q} = \mathbf{L} \mathbf{I} - \mathbf{D}$  where  $L = P + n^{-1} \text{tr}(\mathbf{D})$

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

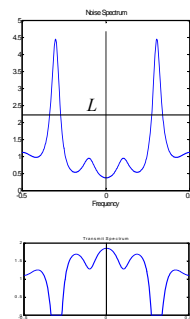
## Power Spectrum Water Filling

- If  $\mathbf{z}$  is from a stationary process then  $\text{diag}(\mathbf{D}) \xrightarrow{n \rightarrow \infty}$  power spectrum

- To achieve capacity use waterfilling on noise power spectrum

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

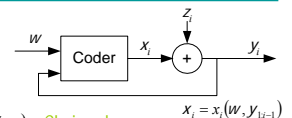
$$C = \int_{-W}^W \frac{1}{2} \log \left( 1 + \frac{\max(L - N(f), 0)}{N(f)} \right) df$$



## Gaussian Channel + Feedback

Does Feedback add capacity?

- White noise - No
- Coloured noise - Not much



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

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

$$x_i = x_i(w, y_{1:i-1}), \quad \mathbf{z} = \mathbf{y} - \mathbf{x}$$

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

$$\mathbf{z} = \mathbf{y} - \mathbf{x} \text{ and translation invariance}$$

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

$$z_i \text{ depends only on } z_{1:i-1}$$

$$= h(\mathbf{y}) - h(\mathbf{z})$$

$$\text{Chain rule, } h(\mathbf{z}) = \frac{1}{2} \log(2\pi e \mathbf{K}_Z) \text{ bits}$$

$$= \frac{1}{2} \log \frac{|\mathbf{K}_Y|}{|\mathbf{K}_Z|}$$

$$\Rightarrow \text{maximize } I(w; \mathbf{y}) \text{ by maximizing } h(\mathbf{y}) \Rightarrow \mathbf{y} \text{ gaussian}$$

$$\Rightarrow \text{we can take } \mathbf{z} \text{ and } \mathbf{x} = \mathbf{y} - \mathbf{z} \text{ jointly gaussian}$$

## Gaussian Feedback Code

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

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

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

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

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

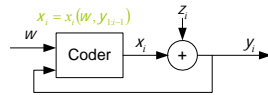
$$\mathbf{K}_y = E\mathbf{y}\mathbf{y}^T = E((\mathbf{B} + \mathbf{I})\mathbf{z}\mathbf{z}^T(\mathbf{B} + \mathbf{I})^T + \mathbf{v}\mathbf{v}^T) = (\mathbf{B} + \mathbf{I})\mathbf{K}_z(\mathbf{B} + \mathbf{I})^T + \mathbf{K}_v$$

$$\mathbf{K}_x = E\mathbf{x}\mathbf{x}^T = E(\mathbf{B}\mathbf{z}\mathbf{z}^T\mathbf{B}^T + \mathbf{v}\mathbf{v}^T) = \mathbf{B}\mathbf{K}_z\mathbf{B}^T + \mathbf{K}_v$$

$$\text{Capacity: } C_{n,FB} = \max_{\mathbf{K}_v, \mathbf{B}} \frac{1}{2} n^{-1} \log \frac{|\mathbf{K}_y|}{|\mathbf{K}_z|} = \max_{\mathbf{K}_v, \mathbf{B}} \frac{1}{2} n^{-1} \log \frac{|(\mathbf{B} + \mathbf{I})\mathbf{K}_z(\mathbf{B} + \mathbf{I})^T + \mathbf{K}_v|}{|\mathbf{K}_z|}$$

$$\text{subject to } \mathbf{K}_x = \text{tr}(\mathbf{B}\mathbf{K}_z\mathbf{B}^T + \mathbf{K}_v) \leq nP$$

hard to solve ☹



## Max Benefit of Feedback: Lemmas

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

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

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

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

$$\begin{aligned} \frac{1}{2} \log((2\pi)^n \det(\mathbf{F} + \mathbf{G})) &= h(\mathbf{f} + \mathbf{g}) \\ &\geq h(\mathbf{f} + \mathbf{g} | \mathbf{g}) = h(\mathbf{f} | \mathbf{g}) \quad \text{Conditioning reduces } h() \\ &= h(\mathbf{f}) \quad \text{Translation invariance} \\ &= h(\mathbf{f}) = \frac{1}{2} \log((2\pi)^n \det(\mathbf{F})) \quad \mathbf{f}, \mathbf{g} \text{ independent} \end{aligned}$$

$$\text{Hence: } \det(2(\mathbf{K}_x + \mathbf{K}_z)) = \det(\mathbf{K}_{\mathbf{x}+\mathbf{z}} + \mathbf{K}_{\mathbf{x}-\mathbf{z}}) \geq \det(\mathbf{K}_{\mathbf{x}+\mathbf{z}}) = \det(\mathbf{K}_y)$$

## Lecture 16

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

## Gaussian Feedback: Toy Example

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

$$\mathbf{x} = \mathbf{B}\mathbf{z} + \mathbf{v}$$

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

$$\det(\mathbf{K}_y) = \det((\mathbf{B} + \mathbf{I})\mathbf{K}_z(\mathbf{B} + \mathbf{I})^T + \mathbf{K}_v)$$

Subject to:

$\mathbf{K}_v$  must be positive definite

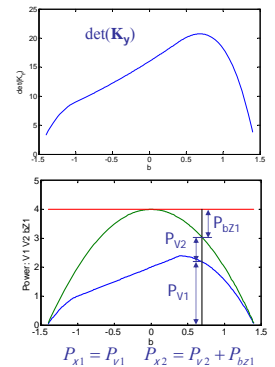
$$\text{Power constraint: } \text{tr}(\mathbf{B}\mathbf{K}_z\mathbf{B}^T + \mathbf{K}_v) \leq 4$$

Solution (via numerically search):

$$b=0: \quad \det(\mathbf{K}_y)=16 \quad C=0.604 \text{ bits}$$

$$b=0.69: \quad \det(\mathbf{K}_y)=20.7 \quad C=0.697 \text{ bits}$$

Feedback increases C by 16%



## Maximum Benefit of Feedback

$$\begin{aligned} C_{n,FB} &\leq \max_{\text{tr}(\mathbf{K}_x) \leq nP} \frac{1}{2} n^{-1} \log \frac{\det(\mathbf{K}_y)}{\det(\mathbf{K}_z)} \\ &\leq \max_{\text{tr}(\mathbf{K}_x) \leq nP} \frac{1}{2} n^{-1} \log \frac{\det(2(\mathbf{K}_x + \mathbf{K}_z))}{\det(\mathbf{K}_z)} \\ &= \max_{\text{tr}(\mathbf{K}_x) \leq nP} \frac{1}{2} n^{-1} \log \frac{2^n \det(\mathbf{K}_x + \mathbf{K}_z)}{\det(\mathbf{K}_z)} \\ &= \frac{1}{2} + \max_{\text{tr}(\mathbf{K}_x) \leq nP} \frac{1}{2} n^{-1} \log \frac{\det(\mathbf{K}_x + \mathbf{K}_z)}{\det(\mathbf{K}_z)} = \frac{1}{2} + C_n \text{ bits/transmission} \end{aligned}$$

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

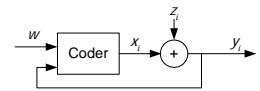
Lemmas 1 & 2:

$$\det(2(\mathbf{K}_x + \mathbf{K}_z)) \geq \det(\mathbf{K}_y)$$

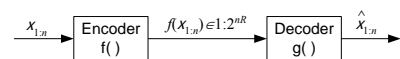
$$\det(k\mathbf{A}) = k^n \det(\mathbf{A})$$

$$\mathbf{K}_y = \mathbf{K}_x + \mathbf{K}_z \text{ if no feedback}$$

Having feedback adds at most  $\frac{1}{2}$  bit per transmission



## Lossy Coding



Distortion function:  $d(x, \hat{x}) \geq 0$

$$\text{examples: (i) } d_S(x, \hat{x}) = (x - \hat{x})^2 \quad \text{(ii) } d_H(x, \hat{x}) = \begin{cases} 0 & x = \hat{x} \\ 1 & x \neq \hat{x} \end{cases}$$

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

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

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

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



## 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)  $E_{x, \hat{x}} d(x, \hat{x}) \leq D$

– this expression is the Information Rate Distortion function for  $X$

We will prove this next lecture

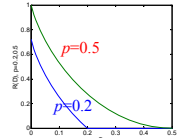
If  $D=0$  then we have  $R(D)=I(x; \hat{x})=H(x)$

## $R(D)$ bound for Bernoulli Source

Bernoulli:  $\mathbf{X} = [0, 1]$ ,  $\mathbf{p}_X = [1-p, p]$  assume  $p \leq 1/2$

- Hamming Distance:  $d(x, \hat{x}) = x \oplus \hat{x}$
- If  $D \geq p$ ,  $R(D)=0$  since we can set  $g(\cdot) \equiv 0$
- For  $D < p \leq 1/2$ , if  $E d(x, \hat{x}) \leq D$  then

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



$\oplus$  is one-to-one

Conditioning reduces entropy

$p(x \oplus \hat{x}) \leq D$  and so for  $D \leq 1/2$   
 $H(x \oplus \hat{x}) \leq H(D)$  as  $H(p)$  monotonic

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

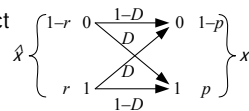
## $R(D)$ for Bernoulli source

We know optimum satisfies  $R(D) \geq H(p) - H(D)$

- We show we can find a  $p(\hat{x}, x)$  that attains this.
- Peculiarly, we consider a channel with  $\hat{x}$  as the input and error probability  $D$

Now choose  $r$  to give  $x$  the correct probabilities:

$$\begin{aligned} r(1-D) + (1-r)D &= p \\ \Rightarrow r &= (p-D)/(1-2D)^{-1} \end{aligned}$$



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

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

$$\begin{aligned} I(x; \hat{x}) &= h(x) - h(x | \hat{x}) \\ &= \frac{1}{2} \log 2\pi e \sigma^2 - h(x - \hat{x} | \hat{x}) \\ &\geq \frac{1}{2} \log 2\pi e \sigma^2 - h(x - \hat{x}) \\ &\geq \frac{1}{2} \log 2\pi e \sigma^2 - \frac{1}{2} \log (2\pi e \text{Var}(x - \hat{x})) \\ &\geq \frac{1}{2} \log 2\pi e \sigma^2 - \frac{1}{2} \log 2\pi e D \end{aligned}$$

Translation Invariance

Conditioning reduces entropy

Gauss maximizes entropy for given covariance

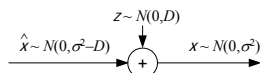
require  $\text{Var}(x - \hat{x}) \leq E(x - \hat{x})^2 \leq D$

$$I(x; \hat{x}) \geq \max \left( \frac{1}{2} \log \frac{\sigma^2}{D}, 0 \right)$$

$I(x; y)$  always positive

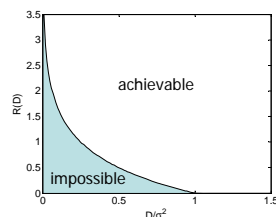
## $R(D)$ for Gaussian Source

To show that we can find a  $p(\hat{x}, x)$  that achieves the bound, we construct a test channel that introduces distortion  $D < \sigma^2$



$$\begin{aligned} I(x; \hat{x}) &= h(x) - h(x | \hat{x}) \\ &= \frac{1}{2} \log 2\pi e \sigma^2 - h(x - \hat{x} | \hat{x}) \\ &= \frac{1}{2} \log 2\pi e \sigma^2 - h(z | \hat{x}) \\ &= \frac{1}{2} \log \frac{\sigma^2}{D} \end{aligned}$$

$$\Rightarrow D(R) = \left( \frac{\sigma^2}{2^R} \right)^2$$

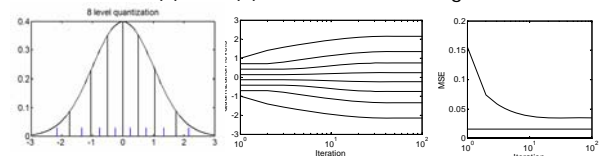


## Lloyd Algorithm

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.

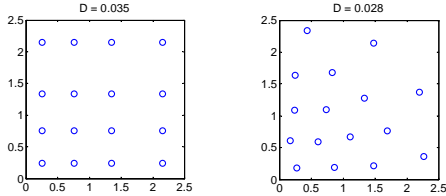


Solid lines are bin boundaries. Initial levels uniform in  $[-1, +1]$ .

Best mean sq error for 8 levels =  $0.0345 \sigma^2$ . Predicted  $D(R) = (\sigma^2/8)^2 = 0.0156 \sigma^2$

## 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)

## Reverse Waterfilling

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

Use a lagrange multiplier:

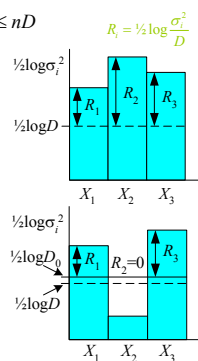
$$J = \sum_{i=1}^n \frac{1}{2} \log \frac{\sigma_i^2}{D_i} + \lambda \sum_{i=1}^n D_i$$

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

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

Choose  $R_i$  for equal distortion

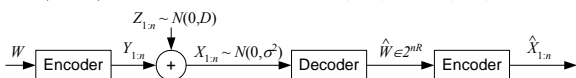
- If  $\sigma_i^2 < D$  then set  $R_i=0$  and increase  $D_0$  to maintain the average distortion equal to  $D$
- If  $x_i$  are correlated then reverse waterfill the eigenvectors of the correlation matrix



## Channel Decoder as Source Coder



- For  $R \equiv C = \frac{1}{2} \log(1 + (\sigma^2 - D)D^{-1})$ , 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  $p(W \neq \hat{W}) < \varepsilon$ , also  $p(\hat{X} \neq Y) < \varepsilon$  and  $E(X_i - \hat{X}_i)^2 \equiv E(X_i - Y_i)^2 = D$



We have encoded  $x$  at rate  $R = \frac{1}{2} \log(\sigma^2 D^{-1})$  with distortion  $D$

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

$$I(x_{1:n}; \hat{x}_{1:n}) \geq \sum_{i=1}^n I(x_i; \hat{x}_i) \quad \text{Mut Info Independence Bound for independent } x_i$$

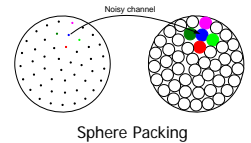
$$\geq \sum_{i=1}^n R(D_i) = \sum_{i=1}^n \max\left(\frac{1}{2} \log \frac{\sigma_i^2}{D_i}, 0\right) \quad R(D) \text{ for individual Gaussian}$$

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

## Channel/Source Coding Duality

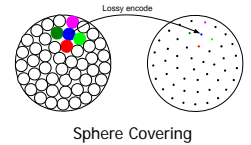
### Channel Coding

- 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



## High Dimensional Space

### In $n$ dimensions

- "Vol" of unit hypercube: 1
- "Vol" of unit-diameter hypersphere:  $V_n = \begin{cases} \pi^{\frac{1}{2}n - \frac{1}{2}} (\frac{1}{2}n - \frac{1}{2})! / n! & n \text{ odd} \\ \pi^{\frac{1}{2}n} 2^{-n} / (\frac{1}{2}n)! & n \text{ even} \end{cases}$

- "Area" of unit-diameter hypersphere:

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

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

$$\text{Proof: } (1 - n^{-1})^n \xrightarrow{n \rightarrow \infty} e^{-1} = 0.3679$$

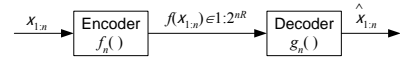
$n$	$V_n$	$A_n$
1	1	2
2	0.79	3.14
3	0.52	3.14
4	0.31	2.47
10	2.5e-3	5e-2
100	1.9e-70	3.7e-68

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

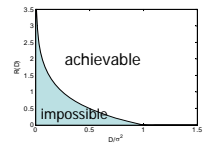
$$R(D) = \inf R \text{ such that } \exists f_n, g_n \text{ with } \lim_{n \rightarrow \infty} E_{\mathbf{x} \in \mathbf{x}} d(\mathbf{x}, \hat{\mathbf{x}}) \leq D$$

Rate Distortion Theorem:

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

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

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



## Rate Distortion Bound

Suppose we have found an encoder and decoder at rate  $R_0$  with expected distortion  $D$  for independent  $x_i$  (worst case)

We want to prove that  $R_0 \geq R(D) = R(E d(x; \hat{x}))$

- We show first that  $R_0 \geq n^{-1} \sum I(x_i; \hat{x}_i)$
- We know that  $I(x_i; \hat{x}_i) \geq R(E d(x_i; \hat{x}_i))$  Def'n of  $R(D)$
- and use convexity to show

$$n^{-1} \sum R(E d(x_i; \hat{x}_i)) \geq R(E d(x; \hat{x})) = R(D)$$

We prove convexity first and then the rest

## Convexity of $R(D)$

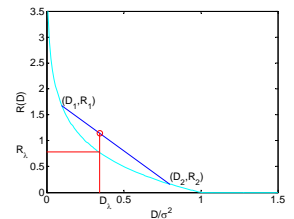
If  $p_1(\hat{x}|x)$  and  $p_2(\hat{x}|x)$  are associated with  $(D_1, R_1)$  and  $(D_2, R_2)$

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

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



$$R(D_\lambda) \leq I_{p_\lambda}(x; \hat{x})$$

$$\leq \lambda I_{p_1}(x; \hat{x}) + (1-\lambda) I_{p_2}(x; \hat{x})$$

$$= \lambda R(D_1) + (1-\lambda) R(D_2)$$

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

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

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

## Proof that $R \geq R(D)$

$$nR_0 \geq H(\hat{x}_{1:n}) = H(\hat{x}_{1:n}) - H(\hat{x}_{1:n} | x_{1:n}) \quad \text{Uniform bound; } H(\hat{x} | x) = 0$$

$$= I(\hat{x}_{1:n}; x_{1:n}) \quad \text{Definition of } I(\cdot)$$

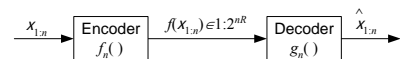
$$\geq \sum_{i=1}^n I(x_i; \hat{x}_i) \quad \text{X indep: Mut Inf Independence Bound}$$

$$\geq \sum_{i=1}^n R(E d(x_i; \hat{x}_i)) = n \sum_{i=1}^n R(E d(x_i; \hat{x}_i)) \quad \text{definition of } R$$

$$\geq nR \left( n^{-1} \sum_{i=1}^n E d(x_i; \hat{x}_i) \right) = nR(E d(x_{1:n}; \hat{x}_{1:n})) \quad \text{convexity defn of vector } d(\cdot)$$

$$\geq nR(D) \quad \text{original assumption that } E(d) \leq D \text{ and } R(D) \text{ monotonically decreasing}$$

## 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.

- $p_X$  is given
- Assume we know the  $p(\hat{x}|x)$  that gives  $I(x; \hat{x}) = R(D)$
- **Random Decoder:** Choose  $2^{nR}$  random  $\hat{x}_i \sim p_{\hat{x}}$ 
  - 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:**  $(x_i, \hat{x}_i) \in \mathbf{X} \times \hat{\mathbf{X}}$  drawn i.i.d.  $\sim p(x, \hat{x})$

$$J_{d,\varepsilon}^{(n)} = \left\{ \mathbf{x}, \hat{\mathbf{x}} \in \mathbf{X}^n \times \hat{\mathbf{X}}^n : \begin{aligned} & \left| -n^{-1} \log p(\mathbf{x}) - H(\mathbf{x}) \right| < \varepsilon, \\ & \left| -n^{-1} \log p(\hat{\mathbf{x}}) - H(\hat{\mathbf{x}}) \right| < \varepsilon, \\ & \left| -n^{-1} \log p(\mathbf{x}, \hat{\mathbf{x}}) - H(\mathbf{x}, \hat{\mathbf{x}}) \right| < \varepsilon \\ & \left| d(\mathbf{x}, \hat{\mathbf{x}}) - E d(\mathbf{x}, \hat{\mathbf{x}}) \right| < \varepsilon \end{aligned} \right\} \quad \text{new condition}$$

**Properties:**

1. Indiv p.d.:  $\mathbf{x}, \hat{\mathbf{x}} \in J_{d,\varepsilon}^{(n)} \Rightarrow \log p(\mathbf{x}, \hat{\mathbf{x}}) = -nH(\mathbf{x}, \hat{\mathbf{x}}) \pm n\varepsilon$
2. Total Prob:  $p(\mathbf{x}, \hat{\mathbf{x}} \in J_{d,\varepsilon}^{(n)}) > 1 - \varepsilon$  for  $n > N_\varepsilon$   
weak law of large numbers;  $d(x_i, \hat{x}_i)$  are i.i.d.

## Conditional Probability Bound

**Lemma:**  $\mathbf{x}, \hat{\mathbf{x}} \in J_{d,\varepsilon}^{(n)} \Rightarrow p(\hat{\mathbf{x}}) \geq p(\hat{\mathbf{x}} | \mathbf{x}) 2^{-n(I(\mathbf{x}; \hat{\mathbf{x}}) + 3\varepsilon)}$

**Proof:**

$$\begin{aligned} p(\hat{\mathbf{x}} | \mathbf{x}) &= \frac{p(\mathbf{x}, \hat{\mathbf{x}})}{p(\mathbf{x})} \\ &= p(\hat{\mathbf{x}}) \frac{p(\mathbf{x}, \hat{\mathbf{x}})}{p(\hat{\mathbf{x}})p(\mathbf{x})} \quad \text{take max of top and min of bottom} \\ &\leq p(\hat{\mathbf{x}}) \frac{2^{-n(H(\mathbf{x}, \hat{\mathbf{x}}) - \varepsilon)}}{2^{-n(H(\mathbf{x}) + \varepsilon)} 2^{-n(H(\hat{\mathbf{x}}) + \varepsilon)}} \quad \text{bounds from defn of } J \\ &= p(\hat{\mathbf{x}}) 2^{n(I(\mathbf{x}; \hat{\mathbf{x}}) + 3\varepsilon)} \quad \text{defn of } I \end{aligned}$$

## Curious but necessary Inequality

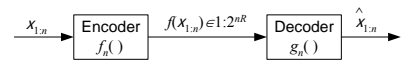
**Lemma:**  $u, v \in [0, 1], m > 0 \Rightarrow (1 - uv)^m \leq 1 - u + e^{-vm}$

**Proof:**  $u=0$ :  $e^{-vm} \geq 0 \Rightarrow (1 - 0)^m \leq 1 - 0 + e^{-vm}$

$u=1$ : Define  $f(v) = e^{-v} - 1 + v \Rightarrow f'(v) = 1 - e^{-v}$   
 $f(0) = 0$  and  $f'(v) > 0$  for  $v > 0 \Rightarrow f(v) \geq 0$  for  $v \in [0, 1]$   
Hence for  $v \in [0, 1]$ ,  $0 \leq 1 - v \leq e^{-v} \Rightarrow (1 - v)^m \leq e^{-vm}$

$0 < u < 1$ : Define  $g_v(u) = (1 - uv)^m$   
 $\Rightarrow g_v''(u) = m(m-1)v^2(1 - uv)^{m-2} \geq 0 \Rightarrow g_v(u)$  convex for  $u, v \in [0, 1]$   
 $(1 - uv)^m = g_v(u) \leq (1 - u)g_v(0) + ug_v(1)$  convexity for  $u, v \in [0, 1]$   
 $= (1 - u)1 + u(1 - v)^m \leq 1 - u + ue^{-vm} \leq 1 - u + e^{-vm}$

## Achievability of $R(D)$ : preliminaries



- Choose  $D$  and find a  $p(\hat{x} | x)$  such that  $I(x; \hat{x}) = R(D)$ ;  $E d(x, \hat{x}) \leq D$   
Choose  $\delta > 0$  and define  $\mathbf{p}_{\hat{x}} = \{ p(\hat{x}) = \sum_x p(x) p(\hat{x} | x) \}$
- **Decoder:** For each  $w \in 1:2^{nR}$  choose  $g_n(w) = \hat{\mathbf{x}}_w$  drawn i.i.d.  $\sim \mathbf{p}_{\hat{x}}^n$
- **Encoder:**  $f_n(\mathbf{x}) = \min_w$  such that  $(\mathbf{x}, \hat{\mathbf{x}}_w) \in J_{d,\varepsilon}^{(n)}$  else 1 if no such  $w$
- **Expected Distortion:**  $\bar{D} = E_{\mathbf{x}, g} d(\mathbf{x}, \hat{\mathbf{x}})$ 
  - over all input vectors  $\mathbf{x}$  and all random decode functions,  $g$
  - for large  $n$  we show  $\bar{D} = D + \delta$  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  $E d(\mathbf{x}, \hat{\mathbf{x}}) \leq D$
- b) if no such  $w$  exists we must have  $d(\mathbf{x}, \hat{\mathbf{x}}_w) < d_{\max}$   
since we are assuming that  $d(\cdot)$  is bounded. Suppose the probability of this situation is  $P_e$ .  
Hence  $\bar{D} = E_{\mathbf{x}, g} d(\mathbf{x}, \hat{\mathbf{x}})$   

$$\leq (1 - P_e)(D + \varepsilon) + P_e d_{\max}$$

$$\leq D + \varepsilon + P_e d_{\max}$$

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

## Error Probability

Define the set of valid inputs for code  $g$

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

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

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

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

## Achievability for average code

Since  $\mathbf{x}, \hat{\mathbf{x}} \in J_{d,\varepsilon}^{(n)} \Rightarrow p(\hat{\mathbf{x}}) \geq p(\hat{\mathbf{x}} | \mathbf{x}) 2^{-n(I(\mathbf{x}; \hat{\mathbf{x}}) + 3\varepsilon)}$

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

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

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

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

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

## Achievability for average code

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

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

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

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

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

both terms  $\rightarrow 0$  as  $n \rightarrow \infty$  provided  $nR > I(\mathbf{x}; \hat{\mathbf{x}}) + 3\varepsilon$

$$\xrightarrow{n \rightarrow \infty} 0$$

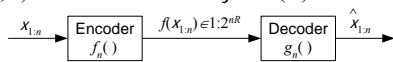
Hence  $\forall \delta > 0, \bar{D} = E_{\mathbf{x}, \mathbf{g}} d(\mathbf{x}, \hat{\mathbf{x}})$  can be made  $\leq D + \delta$

## Achievability

Since  $\forall \delta > 0, \bar{D} = E_{\mathbf{x}, \mathbf{g}} d(\mathbf{x}, \hat{\mathbf{x}})$  can be made  $\leq D + \delta$

there must be at least one  $\mathbf{g}$  with  $E_{\mathbf{x}} d(\mathbf{x}, \hat{\mathbf{x}}) \leq D + \delta$

Hence  $(R, D)$  is achievable for any  $R > R(D)$



that is  $\lim_{n \rightarrow \infty} E_{X_{1:n}}(\mathbf{x}, \hat{\mathbf{x}}) \leq D$

In fact a stronger result is true:

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

## Lecture 18

- Revision Lecture

## Summary (1)

- Entropy:  $H(X) = \sum_{x \in \mathcal{X}} p(x) (-\log_2 p(x)) = E - \log_2(p_X(x))$ 
  - Bounds:  $0 \leq H(X) \leq \log |\mathcal{X}|$
  - Conditioning reduces entropy:  $H(Y | X) \leq H(Y)$
  - Chain Rule:  $H(X_{1:n}) = \sum_{i=1}^n H(X_i | X_{1:i-1}) \leq \sum_{i=1}^n H(X_i)$

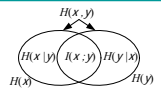
$$H(X_{1:n} | Y_{1:n}) \leq \sum_{i=1}^n H(X_i | Y_i)$$

- Relative Entropy:

$$D(\mathbf{p} \parallel \mathbf{q}) = E_{\mathbf{p}} \log(p(X) / q(X)) \geq 0$$

## Summary (2)

- Mutual Information:



$$I(Y; X) = H(Y) - H(Y | X) \\ = H(X) + H(Y) - H(X, Y) = D(\mathbf{p}_{X,Y} \parallel \mathbf{p}_X \mathbf{p}_Y)$$

– Positive and Symmetrical:  $I(X; Y) = I(Y; X) \geq 0$

–  $X, Y$  indep  $\Leftrightarrow H(X, Y) = H(X) + H(Y) \Leftrightarrow I(X; Y) = 0$

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

$$X_i \text{ independent} \Rightarrow I(X_{1:n}; Y_{1:n}) \geq \sum_{i=1}^n I(X_i; Y_i)$$

$$p(Y_i | X_{1:n}; Y_{1:i-1}) = p(Y_i | X_i) \Rightarrow I(X_{1:n}; Y_{1:n}) \leq \sum_{i=1}^n I(X_i; Y_i)$$

## Summary (3)

- **Convexity:**  $f''(x) \geq 0 \Rightarrow f(x)$  convex  $\Rightarrow E f(X) \geq f(E X)$ 
  - $H(\mathbf{p})$  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:**  $X \rightarrow Y \rightarrow Z \Leftrightarrow p(z|x, y) = p(z|y) \Leftrightarrow I(X; Z|Y) = 0$   
 $\Rightarrow I(X; Y) \geq I(X; Z)$  and  $I(X; Y) \geq I(X, Y|Z)$
- **Fano:**  $X \rightarrow Y \rightarrow \hat{X} \Rightarrow p(\hat{X} \neq X) \geq \frac{H(X|Y) - 1}{\log(|\mathcal{X}| - 1)}$
- **Entropy Rate:**  $H(\mathbf{X}) = \lim_{n \rightarrow \infty} n^{-1} H(X_{1:n})$ 
  - Stationary process:  $H(\mathbf{X}) \leq H(X_n | X_{1:n-1}) \xrightarrow{\text{a.s.}} \infty$
  - Markov Process:  $H(\mathbf{X}) = \lim_{n \rightarrow \infty} H(X_n | X_{n-1}) \xrightarrow{\text{if stationary}}$
  - Hidden Markov:  $H(Y_n | Y_{1:n-1}, X_1) \leq H(\mathbf{Y}) \leq H(Y_n | Y_{1:n-1}) \xrightarrow{\text{a.s.}} \infty$

## Summary (5)

- **Typical Set**
  - Individual Prob:  $\mathbf{x} \in T_\epsilon^{(n)} \Rightarrow \log p(\mathbf{x}) = -nH(\mathbf{X}) \pm n\epsilon$
  - Total Prob:  $p(\mathbf{x} \in T_\epsilon^{(n)}) > 1 - \epsilon$  for  $n > N_\epsilon$
  - Size:  $(1 - \epsilon)2^{n(H(\mathbf{X}) - \epsilon)} < |T_\epsilon^{(n)}| \leq 2^{n(H(\mathbf{X}) + \epsilon)}$
  - No other high probability set can be much smaller
- **Asymptotic Equipartition Principle**
  - Almost all event sequences are equally surprising

## Summary (7)

- **Differential Entropy:**  $h(X) = E - \log f_X(X)$ 
  - Not necessarily positive
  - $h(X+a) = h(X)$ ,  $h(aX) = h(X) + \log|a|$ ,  $h(X|Y) \leq h(X)$
  - $I(X, Y) = h(X) + h(Y) - h(X, Y) \geq 0$ ,  $D(f||g) = E \log(f/g) \geq 0$
- **Bounds:**
  - **Finite range:** Uniform distribution has max:  $h(X) = \log(b-a)$
  - **Fixed Covariance:** Gaussian has max:  $h(X) = \frac{1}{2} \log((2\pi e)^n |K|)$
- **Gaussian Channel**
  - **Discrete Time:**  $C = \frac{1}{2} \log(1 + PN^{-1})$
  - **Bandlimited:**  $C = W \log(1 + PN_0^{-1} W^{-1})$ 
    - For constant C:  $E_b N_0^{-1} = PC^{-1} N_0^{-1} = (W/C) \left( 2^{(W/C)^{-1}} - 1 \right) \xrightarrow{W \rightarrow \infty} \ln 2 = -1.6 \text{ dB}$
  - **Feedback:** Adds at most  $\frac{1}{2}$  bit for coloured noise

## Summary (4)

- **Kraft:** Uniquely Decodable  $\Rightarrow \sum_{i=1}^{|\mathcal{X}|} D^{-l_i} \leq 1 \Rightarrow \exists$  prefix code
- **Average Length:** Uniquely Decodable  $\Rightarrow L_C = E l(X) \geq H_D(X)$
- **Shannon-Fano:** Top-down 50% splits.  $L_{SF} \leq H_D(X) + 1$
- **Shannon:**  $l_x = \lceil -\log_D p(x) \rceil$   $L_S \leq H_D(X) + 1$
- **Huffman:** Bottom-up design. Optimal.  $L_H \leq H_D(X) + 1$ 
  - 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 (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 (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:**  $R(D) = \min_{\mathbf{p}_{\hat{X}|X} \text{ s.t. } E d(X, \hat{X}) \leq D} I(X; \hat{X})$ 
  - **Bernoulli Source** with Hamming  $d$ :  $R(D) = \max(H(\mathbf{p}_X) - H(D), 0)$
  - **Gaussian Source** with mean square  $d$ :  $R(D) = \max(\frac{1}{2} \log(\sigma^2/D), 0)$
  - Can encode at rate  $R$ : random decoder, joint typical encoder
  - Can't encode below rate  $R$ : independence bound
- **Lloyd Algorithm:** iterative optimal vector quantization