Information Theory

Chapter I: Discrete information sources

Block diagram of a communication system

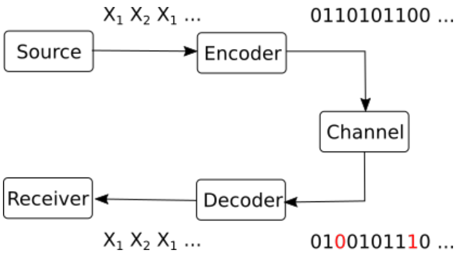


Figure 1: Block diagram of a communication system

- Source: creates information messages
- Encoder: converts messages into symbols for transmission (i.e bits)
- Channel: delivers the symbols, introduces errors
- Decoder: detects/corrects the errors, rebuilds the information messages

What is information?

Example:

- Consider the sentence: "your favorite football team lost the last match"
- Does this message carry information? How, why, how much?
- Consider the following facts:
  - the message carries information only when you don't already know the result
  - if you already known the result, the message is useless (brings no information)
  - if the result was to be expected, there is little information. If the result is highly unusual, there is more information in this message (think betting)

Information and events

- We define the notion of **information** for a **probabilistic event**
  - the happening of a probabilistic event = creation of information
- Information brought by an event depends on the **probability** of the event
- Rule of thumb: if you can guess something most of the times, it has little information
- Questions:
  - does a sure event ($p = 1$) bring any information?
  - does an almost sure event (e.g. $P = 0.9999$) bring little or much information?
  - does a rare event (e.g. $P = 0.0001$) bring a little or much information?

Information

- The information attached to a particular event (known as "message") s_i is rigorously defined as:

$$\boxed{i(s_i) = -\log_2(\underbrace{p(s_i)}_{\le 1})}_{\le 0} \; [\text{bits}]$$

$p = 1 \Rightarrow -\log_2(1) = 0$

- Properties:

$p \in [0,1] \Rightarrow p \le 1 \Rightarrow \log_2(p) \le 0 \Rightarrow -\log_2(p) \ge 0$

  - $i(s_i) \geq 0$
  - lower probability (rare events) means higher information
  - higher probability (frequent events) means lower information
  - a certain event brings no information: $-\log(1) = 0$
  - an event with probability 0 brings infinite information (but it never happens. . .)
  - for two independent events, their information gets added

$$i(p(s_i) \cdot p(s_j)) = i(s_i) + i(s_j)$$

$P(\underbrace{\text{Dice 1} = 3 \text{ AND Dice 2} = 6}_{D_1 \text{ AND } D_2}) = \frac{1}{36} = \frac{1}{6} \cdot \frac{1}{6}$

$= P(\underbrace{\text{Dice}_1 = 3}_{D_1}) \cdot P(\underbrace{\text{Dice}_2 = 6}_{D_2})$

$i(D_1 \cap D_2) = -\log_2(P(D_1 \cap D_2)) =$

$= -\log_2(P(D_1) \cdot P(D_2))$

$= \underbrace{-\log_2(P(D_1))}_{i(D_1)} + \underbrace{-\log_2(P(D_2))}_{i(D_2)}$

The choice of logarithm

- Any base of logarithm can be used in the definition.
- Usual convention: use binary logarithm $\log_2()$
- In this case, the information $i(s_i)$ is measured in **bits**
- If using natural logarithm *ln*(), it is measured in *nats*.
- Logarithm bases can be converted to/from one another:

$$\log_b(x) = \frac{\log_a(x)}{\log_a(b)} = ct$$

- Information defined using different logarithms differ only in scaling:

$$i_b(s_i) = \frac{i_a(s_i)}{\log_a(b)}$$

Information source

- A probabilistic event is always part of a set of multiple events (options)
  - e.g: a football team can win/lose/draw a match (3 possible events)
  - each event has a certain probability. All probabilities are known beforehand
  - at a given time, only one of the events can happen
- An **information source** = the set of all events together with their probabilities
- One event is called a **message**
- Each message carries the information that **it** happened, the quantity of information is dependent on its probability

$$S : \begin{array}{ccc} s_1 & s_2 & s_3 \end{array}$$
$$\begin{pmatrix} \text{lose} & \text{win} & \text{draw} \\ 0.99 & 0.005 & 0.005 \end{pmatrix}$$

Sequence of messages

- An information source creates a **sequence of messages**
  - e.g. like throwing a coin or a dice several times in a row
- The probabilities of the messages are known and fixed
- Each time, a new message is randomly selected according to the probabilities

$s_1 \quad s_2 \quad s_1 \quad s_3 \quad s_1 \quad s_2 \quad s_1 \quad \ldots\ldots$

Discrete memoryless source

- A **discrete memoryless source** (DMS) is an information source which produces a sequence of **independent** messages
  - i.e. the choice of a message at one time does not depend on the previous messages
- Each message has a fixed probability. The set of probabilities is the **distribution** of the source:

$$S : \begin{pmatrix} s_1 & s_2 & s_3 \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \end{pmatrix}$$

$\frac{1}{3}$ 3

3

6

Discrete memoryless source

$$S : \begin{pmatrix} s_1 & s_2 & s_3 \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \end{pmatrix}$$

- Terminology:
  - Discrete: it can take a value from a discrete set (“alphabet”)
  - Complete: $\sum p(s_i) = 1$
  - Memoryless: succesive values are independent of previous values (e.g. successive throws of a coin)
- A message from a DMS is also called a **random variable** in probabilistics.

Examples

- A coin is a discrete memoryless source (DMS) with two messages:

$$S : \begin{pmatrix} heads & tails \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

- A dice is a discrete memoryless source (DMS) with six messages:

$$S : \begin{array}{c} \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 \end{matrix} \\ \begin{pmatrix} s_1 & s_2 & s_3 & s_4 & s_5 & s_6 \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \end{pmatrix} \end{array}$$

- Playing the lottery can be modeled as DMS:

$$S : \begin{array}{c} \begin{matrix} \text{lose} & \text{win} \end{matrix} \\ \begin{pmatrix} s_1 & s_2 \\ 0.9999 & 0.0001 \end{pmatrix} \end{array}$$

Examples

- An extreme type of DMS containing the certain event:

$$S : \begin{pmatrix} s_1 & s_2 \\ 1 & 0 \end{pmatrix}$$

- Receiving an unknown *bit* (0 or 1) with equal probabilities:

$$S : \begin{pmatrix} 0 & 1 \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

Sequence of messages from DMS

- A DMS produces a sequence of messages by randomly selecting a message every time, with the same fixed probabilities
  - throwing a dice several times in a row you can get a sequence $4, 2, 3, 2, 1, 6, 1, 5, 4, 5$
- If the sequence is very long (has N messages, N very large), each message s_i appears approximately $p(s_i) * N$ times in the sequence
  - gets more precise as $N \to \infty$

Entropy of a DMS

$$S: \begin{pmatrix} s_1 & s_2 & s_3 \\ 0.9 & 0.01 & 0.09 \end{pmatrix}$$

$s_1\ s_1\ s_2\ s_2\ s_3\ s_1\ s_1\ s_2\ s_1\ s_1$

- We usually don't care about a single message. We are interested in long sequences of messages (think millions of bits of data)
- We are interested in the *average* information of a message from a DMS
- Definition: the **entropy** of a DMS source S is **the average information of a message**:

$$\boxed{H(S) = \sum_k p(s_k) \underbrace{i(s_k)}_{-\log_2(p(s_k))} = -\sum_k p(s_k) \log_2(p_k)}$$

where $p(s_k)$ is the probability of message k

$$= p(s_1) \cdot i(s_1) + p(s_2) \cdot i(s_2) + p(s_3) \cdot i(s_3) + \dots$$

$x_1, x_2, \dots x_n$

$$\overline{X}_{average} = \sum_i p(x_i) \cdot x_i$$

$\cancel{G_x}\quad G_e \quad G_s$

$$G_{average} = 0.6 \cdot G_e + 0.4 \cdot G_s$$

Entropy of a DMS

- Since information of a message is measured in bits, entropy is measured in **bits** (or **bits / message**, to indicate it is an average value)
- Entropies using information defined with different logarithms differ only in scaling:

$$H_b(S) = \frac{H_a(S)}{\log_a(b)}$$

Examples

$$S : \begin{pmatrix} s_1 & s_2 \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix} \Rightarrow H(S) = -\frac{1}{2}\log_2\left(\frac{1}{2}\right) - \frac{1}{2}\log_2\frac{1}{2} = 1 \text{ bit}$$

$$S : \begin{pmatrix} s_1 & \cdots & s_6 \\ \frac{1}{6} & \cdots & \frac{1}{6} \end{pmatrix} \Rightarrow H(S) = \left(-\frac{1}{6}\log\frac{1}{6}\right)\cdot 6 = \log(6) = 2.58 \text{ bits}$$

- Coin: $H(S) = 1 bit/message$
- Dice: $H(S) = \log(6) bits/message$
- Lottery: $H(S) = -0.9999\log(0.9999) - 0.0001\log(0.0001)$
- Receiving 1 bit: $H(S) = 1 bit/message$ (hence the name!)

Interpretation of the entropy

All the following interpretations of entropy are true:

- H(S) is the *average uncertainty* of the source S
- H(S) is the *average information* of the messages from source S
- A long sequence of N messages from S has total information $\approx N \cdot H(S)$
- H(S) is the minimum number of bits (0,1) required to uniquely represent an average message from source S

Coin

$$S_1 : \begin{pmatrix} s_1 & s_2 \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix} \Rightarrow H(S) = 1$$

Dice

$$S_2 : \begin{pmatrix} s_1 & s_2 & s_3 & s_4 & s_5 & s_6 \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \end{pmatrix} \Rightarrow H(S) = 2.58$$

2.58 bits

Properties of entropy

We prove the following **properties of entropy**:

1. $H(S) \geq 0$ (non-negative)

   $H(S) = -\sum \underbrace{p(s_i)}_{\geq 0} \cdot \underbrace{\log(\underbrace{p(s_i)}_{\leq 1})}_{\leq 0} \geq 0$

   Proof: via definition

2. $H(S)$ is maximum when all n messages have equal probability $\frac{1}{n}$. The maximum value is $\max H(S) = \log(n)$

   Proof: only for the case of 2 messages, use derivative in definition

3. *Diversification* of the source always increases the entropy

   Proof: compare entropies in both cases

2). $S: \begin{pmatrix} s_1 & s_2 \\ p & 1-p \end{pmatrix}$

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

Find max:

$\frac{\partial H(S)}{\partial p} = 0 \iff$

$-\log(p) - \cancel{p} \cdot \frac{1}{\cancel{p} \cdot \ln 2} - (-1) \cdot \log(1-p) - \cancel{(1-p)} \cdot \frac{1 \cdot (-1)}{\cancel{(1-p)} \ln 2} = 0$

$\iff -\log p - \cancel{\frac{1}{\ln 2}} + \log(1-p) + \cancel{\frac{1}{\ln 2}} = 0$

$\iff \log(1-p) = \log p \Rightarrow \boxed{p = 1-p = \frac{1}{2}}$

The entropy of a binary source

- Consider a general DMS with two messages:

$$S : \begin{pmatrix} s_1 & s_2 \\ p & 1-p \end{pmatrix}$$

~~0.6~~ ~~0.4~~

- It's entropy is:

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

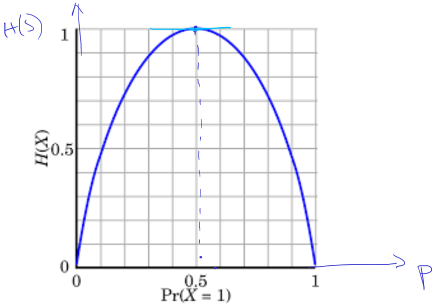


Figure 2: Entropy of a binary source

$\log a - \log b = \log \frac{a}{b}$

3).

$$S_1 : \begin{pmatrix} s_1 & s_2 & \dots & s_n \\ p(s_1) & p(s_2) & \dots & p(s_n) \end{pmatrix}$$

$$S_2 : \begin{pmatrix} s_{1a} & s_{1b} & s_2 & \dots & s_n \\ p(s_{1a}) & p(s_{1b}) & p(s_2) & \dots & p(s_n) \end{pmatrix}$$

$$H(S_2) \geq H(S_1)$$

$$-p(s_{1a}) \cdot \log p(s_{1a}) - p(s_{1b}) \cdot \log p(s_{1b}) - \cancel{p(s_2) \cdot \log(p(s_2))} - \dots$$

$$\geq -p(s_1) \log(p(s_1)) - \cancel{p(s_2) \cdot \log(p(s_2))} - \dots$$

$(=)$

$p(s_{1a}) = p_1$

$p(s_{1b}) = p_2$

$p(s_1) = p_1 + p_2$

$$-p_1 \log p_1 - p_2 \log p_2 \geq -(p_1+p_2) \cdot \log(p_1+p_2)$$

$$-p_1 \log p_1 - p_2 \log p_2 + p_1 \cdot \log(p_1+p_2) + p_2 \log(p_1+p_2) \geq 0$$

$$\underbrace{p_1}_{\geq 0} \cdot \underbrace{\log \underbrace{\frac{p_1+p_2}{p_1}}_{\geq 1}}_{\geq 0} + \underbrace{p_2}_{\geq 0} \cdot \underbrace{\log \underbrace{\frac{p_1+p_2}{p_2}}_{\geq 1}}_{\geq 0} \geq 0$$

q.e.d.

Example - Game

$$N: \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & \frac{1}{8} & \frac{1}{8} \end{pmatrix}$$

$$\Rightarrow H(N) = -\frac{1}{\cancel{8}} \cdot \log\left(\frac{1}{8}\right) \cdot \cancel{8} = -\log\frac{1}{8}$$

$$= \log_2 8 = \log_2 2^3 = \underline{3 \text{ bits}}$$

Game: I think of a number between 1 and 8. You have to guess it by asking yes/no questions.

- How much uncertainty does the problem have?
- How is the best way to ask questions? Why?
- What if the questions are not asked in the best way?
- On average, what is the number of questions required to find the number?

1) Is it >4 ?

$$\text{Answer}: \begin{pmatrix} \text{Yes} & \text{No} \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

$$H(\text{Answer}) = -\frac{1}{\cancel{2}} \log\frac{1}{2} \cdot \cancel{2} = \underline{1 \text{ bit}}$$

1) Is it number 1?

$$\text{Answer}: \begin{pmatrix} \text{Yes} & \text{No} \\ \frac{1}{8} & \frac{7}{8} \end{pmatrix} \Rightarrow H(\text{Answer}) = 0.54 \text{ bit}$$

Example - Game v2

- Suppose I choose a number according to the following distribution:

$$S : \begin{pmatrix} s_1 & s_2 & s_3 & s_4 \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{8} & \frac{1}{8} \end{pmatrix}$$

$H(S) = -\frac{1}{2}\log\frac{1}{2} - \frac{1}{4}\log\frac{1}{4} - 2\cdot\frac{1}{8}\log\frac{1}{8} = \frac{1}{2} + \frac{1}{2} + \frac{3}{4} = \frac{7}{4} = 1.75b$

Questions: 1). Is it 1? Answer: $\begin{pmatrix} Yes & No \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}$

Is it 2? Answer: $\begin{pmatrix} Yes & No \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}$

Is it 3? Answer: $\begin{pmatrix} Yes & No \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}$

"Optimal decision tree"

Number of questions needed on average:

$Q = \frac{1}{2}\cdot 1 + \frac{1}{4}\cdot 2 + \frac{1}{8}\cdot 3 + \frac{1}{8}\cdot 3 = 1.75$ questions

  - On average, what is the number of questions required to find the number?
  - What questions would you ask?

- What if the distribution is:

$$S : \begin{pmatrix} s_1 & s_2 & s_3 & s_4 \\ 0.14 & 0.29 & 0.4 & 0.17 \end{pmatrix}$$

Optimal decision tree:

1) Is it 3? 40% yes → 3; 60% No → Is it 2? Yes → 2; No → Is it 1? → 1, 4

0.14, 0.29, 0.17

- In general:
  - What distribution makes guessing the number the most difficult?
  - What distribution makes guessing the number the easiest?

Efficiency and redundancy

2nd choice:

$H = 7.96$

$H_{max} = 8$

$S : \begin{pmatrix} s_1 & s_2 & s_3 \\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \end{pmatrix} \rightarrow H(S)$

$S_{max} : \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix} \rightarrow H_{max}(S) = \log_2(3)$

- Efficiency of a DMS:

$$\eta = \frac{H(S)}{H_{max}} = \frac{H(S)}{\log(n)}$$

$= 90\%$

$= 99\%$

Is it 1 or 2?

Yes → Is it 1? (Yes → 1; No → 2)

No → Is it 3? (Yes → 3; No → 4)

Number of questions needed?

2

2

- Absolute redundancy of a DMS:

$$R = H_{max} - H(S)$$

- Relative redundancy of a DMS:

$$\rho = \frac{H_{max} - H(S)}{H_{max}} = 1 - \eta$$

$\eta = 90\% = 0.9$

$\rho = 10\% = 0.1$

Information flow of a DMS

- Suppose that message s_i takes time t_i to be transmitted via some channel.
- Definition: the **information flow** of a DMS S is the average information transmitted per unit of time:

$$H_\tau(S) = \frac{H(S)}{\overline{t}} \quad = \frac{\text{entropy}}{\text{average transm. duration}}$$

where $\overline{t}$ is the average duration of transmitting a message:

$$\overline{t} = \sum_i p_i t_i$$

- Measured in **bps** (bits per second) 100 kb/s
- Important for data communication

Distance between distributions

- How to measure how similar / how different are two distributions?
  - must have the same number of messages
  - example: $p(s_1), \ldots p(s_n)$ and $q(s_1), \ldots q(s_n)$
- **Definition**: the **Kullback–Leibler distance** of two distributions P and Q is

$$D_{KL}(P||Q) = \sum_i p(s_i) \log\left(\frac{p(s_i)}{q(s_i)}\right)$$

- It is a way to measure the **distance (difference)** between two distributions
- Also known as *relative entropy*, or the Kullback-Leibler *divergence*

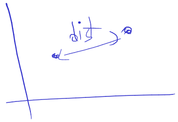
$$S_1 : \begin{pmatrix} a & b & \ldots & z \\ 0.06 & 0.01 & & 0.003 \end{pmatrix}$$

$$S_2 : \begin{pmatrix} a & b & & z \\ 0.065 & 0.008 & & 0.0031 \end{pmatrix}$$

KL distance

$$D_{KL}(S_1, S_2) = 0.06 \cdot \log_2 \frac{0.06}{0.065} + 0.01 \cdot \log_2 \frac{0.01}{0.008} + \ldots + 0.003 \cdot \log_2 \frac{0.003}{0.0031}$$

Properties of Kullback-Leibler distance



- Properties:
  - $D_{KL}(P||Q)$ is always ≥ 0, and is equal to 0 only when P and Q are the same
  - the higher $D_{KL}(P||Q)$ is, the more different the distributions are
  - it is **not commutative**: $D_{KL}(P||Q) \neq D_{KL}(Q||P)$
- Example: at whiteboard
- Example usage: classification systems (cross-entropy loss)

Extended DMS

$S \rightarrow S^m$

- Definition: the **n-th order extension** of a DMS S, S^n is a source which has as messages all the combinations of n messages of S:

$$\sigma_i = \underbrace{s_j s_k \ldots s_l}_{n}$$

- If S has k messages, S^n has k^n messages
- Since S is DMS, probabilities multiply:

$$p(\sigma_i) = p(s_j) \cdot p(s_k) \cdot \ldots \cdot p(s_l)$$

Extended DMS - Example

- Examples:

$$S : \begin{pmatrix} s_1 & s_2 \\ \frac{1}{4} & \frac{3}{4} \end{pmatrix}$$

$$S^2 : \begin{pmatrix} \sigma_1 = s_1 s_1 & \sigma_2 = s_1 s_2 & \sigma_3 = s_2 s_1 & \sigma_4 = s_2 s_2 \\ \frac{1}{16} & \frac{3}{16} & \frac{3}{16} & \frac{9}{16} \end{pmatrix}$$

$$S^3 : \begin{pmatrix} s_1 s_1 s_1 & s_1 s_1 s_2 & s_1 s_2 s_1 & s_1 s_2 s_2 & s_2 s_1 s_1 & s_2 s_1 s_2 & s_2 s_2 s_1 & s_2 s_2 s_2 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \end{pmatrix}$$

$\frac{1}{4} \cdot \frac{3}{4} \cdot \frac{1}{4}$

Extended DMS - Another example

- Long sequence of binary messages:

$$0100110011100\ldots$$

- Can be grouped in bits, half-bytes, bytes, 16-bit words, 32-bit long words, and so on
- Can be considered:
  - N messages from a binary source (with 1 bit), or
  - N/2 messages from a source with 4 messages (with 2 bits)...
  - etc

16 messages from $S : \begin{pmatrix} 0 & 1 \\ p_0 & p_1 \end{pmatrix}$

4 messages from $S^4 : \begin{pmatrix} 0000 & 0001 & \ldots & 1111 \\ p_{\ldots} & p & & \end{pmatrix}$

16 bits = 2 messages, each 8 bit long

$16 \times H(S) = 2 \cdot H(S^8)$

Property of DMS

b

bbbbbbbb

Proof:

$$S: \begin{pmatrix} s_1 & \dots & s_N \\ p_1 & \dots & p_N \end{pmatrix}$$

$$S^n: \begin{pmatrix} \sigma_1 & \dots & \sigma_M \end{pmatrix}$$

$$\sigma = \underbrace{s_k s_l \dots s_j}_{n}$$

$$p(\sigma) = p(s_k)\cdot p(s_l)\cdots p(s_j)$$

$$H(S^n) = -\sum_i p(\sigma_i)\cdot \log(p(\sigma_i))$$

$$= -\underbrace{\sum_{k_1}\sum_{k_2}\dots\sum_{k_n}}_{n} \underbrace{p(s_{k_1})\cdots p(s_{k_n})}_{p(\sigma_i)} \cdot \log(p(s_{k_1})\cdots p(s_{k_n}))$$

$$= -\sum_{k_1}\sum_{k_2}\dots\sum_{k_n} \underbrace{p(s_{k_1})\cdots p(s_{k_n})}_{p(\sigma_i)} \cdot \left(\log(p(s_{k_1})) + \dots + \log(p(s_{k_n}))\right)$$

- Theorem: The entropy of a n-th order extension is n times larger than the entropy of the original DMS

$$H(S^n) = nH(S)$$

- Interpretation: grouping messages from a long sequence in blocks of n does not change total information (e.g. groups of 8 bits = 1 byte)

$$= -\sum_{k_1}\sum_{k_2}\dots\sum_{k_n} \underbrace{p(s_{k_1})\cdots p(s_{k_n})}_{n} \cdot \log(p(s_{k_1})) - \sum_{k_1}\sum_{k_2}\dots\sum_{k_n} \underbrace{p(s_{k_1})\cdots p(s_{k_n})}_{p(\sigma_i)} \cdot \log(p(s_{k_2})) - \dots - \sum_{k_1}\sum_{k_2}\dots\sum_{k_n} \underbrace{p(s_{k_1})\cdots p(s_{k_n})}_{p(\sigma_i)} \cdot \log(p(s_{k_n})) =$$

$$-\underbrace{\sum_{k_1} p(s_{k_1})\cdot \log p(s_{k_1})}_{H(S)} \cdot \underbrace{\sum_{k_2} p(s_{k_2})}_{1} \cdots \underbrace{\sum_{k_n} p(s_{k_n})}_{1} \; - \underbrace{\sum_{k_1} p(s_{k_1})}_{1} \cdot \underbrace{\sum_{k_2} p(s_{k_2})\cdot \log p(s_{k_2})}_{H(S)} \cdot \underbrace{\sum_{k_3} p(s_{k_3})}_{1} \cdots \underbrace{\sum_{k_n} p(s_{k_n})}_{1}$$

same

n times the same thing

$$= n \cdot H(S)$$

An example [memoryless is not enough]

- The distribution (frequencies) of letters in English:

letter	probability	letter	probability
A	.082	N	.067
B	.015	O	.075
C	.028	P	.019
D	.043	Q	.001
E	.127	R	.060
F	.022	S	.063
G	.020	T	.091
H	.061	U	.028
I	.070	V	.010
J	.002	W	.023
K	.008	X	.001
L	.040	Y	.020
M	.024	Z	.001

DMS :
memoryless

$S : \begin{pmatrix} A & \cdots & Z \\ 0.082 & & 0.001 \end{pmatrix}$

wha?
t
l

- Text from a memoryless source with these probabilities:

OCRO HLI RGWR NMIELWIS EU LL NBNESEBYA TH EEI
ALHENHTTPA OOBTTVA NAH BRL

(taken from Elements of Information Theory, Cover, Thomas)

- What's wrong? **Memoryless**

Sources with memory

$$
\begin{array}{c|cccc}
 & s_1 & s_2 & s_3 & s_4 \\
\hline
S_1 & 0.4 & 0.3 & 0.2 & 0.1 \\
S_2 & 0.33 & 0.37 & 0.15 & 0.15 \\
S_3 & 0.2 & 0.35 & 0.41 & 0.04 \\
S_4 & 0.1 & 0.2 & 0.3 & 0.4
\end{array}
$$

- **Definition**: A source has **memory of order** m if the probability of a message depends on the last m messages.
- The last m messages = the **state** of the source (notation S_i).
- A source with n messages and memory m => has n^m states in all.
- For every state, messages can have a different set of probabilities. Notation: $p(s_i|S_k)$ = *"probability of s_i in state S_k"*.
- Also known as *Markov sources*.

$s_i \mid S_k$ "if" "conditioned"

"The source S is in state S_3" means that the last 2 messages were $s_1 s_3$

Example:

A source with 3 messages (s_1, s_2, s_3)

memory order 2: 3^2 states

States:

$S_1 = s_1 s_1$

$S_2 = s_1 s_2$

$S_3 = s_1 s_3$

$S_4 = s_2 s_1$

$S_5 = s_2 s_2$

$S_6 = s_2 s_3$

$S_7 = s_3 s_1$

$S_8 = s_3 s_2$

$S_9 = s_3 s_3$

9 states $= 3^2$

Example

- A source with $n = 4$ messages and memory $m = 1$
  - if last message was s_1, choose next message with distribution

source is in state 1

$$\text{State 1: } S_1 : \begin{pmatrix} s_1 & s_2 & s_3 & s_4 \\ 0.4 & 0.3 & 0.2 & 0.1 \end{pmatrix} \quad P(s_3|S_1) \Rightarrow H(S_1)$$

  - if last message was s_2, choose next message with distribution

source is in state 2

$$\text{State 2 } S_2 : \begin{pmatrix} s_1 & s_2 & s_3 & s_4 \\ 0.33 & 0.37 & 0.15 & 0.15 \end{pmatrix} \quad p(s_3|S_2) \Rightarrow H(S_2)$$

  - if last message was s_3, choose next message with distribution

$$\text{State 3: } S_3 : \begin{pmatrix} s_1 & s_2 & s_3 & s_4 \\ 0.2 & 0.35 & 0.41 & 0.04 \end{pmatrix} \quad p(s_4|S_3) = P(S_4|S_3) = \text{Source transitions moves from } S_3 \text{ to } S_4 \Rightarrow H(S_3)$$

  - if last message was s_4, choose next message with distribution

$$\text{State 4: } S_4 : \begin{pmatrix} s_1 & s_2 & s_3 & s_4 \\ 0.1 & 0.2 & 0.3 & 0.4 \end{pmatrix} \quad P(s_1|S_4) \Rightarrow H(S_4)$$

$\underline{s_1}\ \underline{?}$ $(m=1)$

$\underline{s_2}\ \underline{s_2}\ \underline{?}$ $(m=2)$

$\underline{s_3}\ \underline{s_4}$

old state S_3 — new state S_4

Transitions

- When a new message is provided, the source **transitions** to a new state:

$$\dots \underbrace{s_i s_j s_k}_{\text{old state } = S_1} s_l$$

$$\dots s_i \underbrace{s_j s_k s_l}_{\text{new state } = S_2}$$

$$P(s_l \mid S_1) = P(\text{to transition from } S_1 \text{ to state } S_2)$$

$$= P(S_2 \mid S_1)$$

- The message probabilities = the probabilities of transitions from some state S_u to another state S_v

Transition matrix

- The transition probabilities are organized in a **transition matrix** $[T]$

$$[T] = \begin{bmatrix} p_{11} & \overset{P(S_2|S_1)}{p_{12}} & \cdots & p_{1N} \\ p_{21} & p_{22} & \cdots & p_{2N} \\ \cdots & \cdots & \cdots & \cdots \\ p_{N1} & p_{N2} & \cdots & p_{NN} \end{bmatrix}$$

$$T = \begin{array}{c|c} & S_1 \quad S_2 \quad \cdots \quad S_j \quad \cdots \quad S_n \\ \hline \begin{array}{c} S_1 \\ S_2 \\ S_i \\ \vdots \\ S_n \end{array} & \begin{bmatrix} & & & & \\ & & & & \\ & & & \boxed{p_{ij}} & \\ & & & & \\ & & & & \end{bmatrix} \end{array}$$

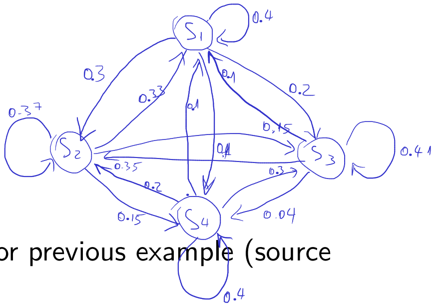
- p_{ij} is the transition probability from state S_i to state S_j
- N is the total number of states

$N =$

$$T = \begin{bmatrix} \\ \\ \\ \end{bmatrix}$$

Graphical representation

$$T = \begin{array}{c} \\ S_1 \\ S_2 \\ S_3 \\ S_4 \end{array} \begin{array}{c} \begin{array}{cccc} S_1 & S_2 & S_3 & S_4 \end{array} \\ \begin{bmatrix} 0.4 & 0.3 & 0.2 & 0.1 \\ 0.33 & 0.37 & 0.15 & 0.15 \\ 0.2 & 0.35 & 0.41 & 0.04 \\ 0.1 & 0.2 & 0.3 & 0.4 \end{bmatrix} \end{array}$$



Generating a sequence of messages = a random walk on the graph

At whiteboard: draw states and transitions for previous example (source with $n = 4$ messages and memory $m = 1$)

$s_1, \dots, s_4$

DMS :

Entropy of sources with memory

$$T = \begin{array}{c|cccc} & S_1 & S_2 & S_3 & S_4 \\ \hline S_1 & 0.4 & 0.3 & 0.2 & 0.1 \\ S_2 & 0.33 & 0.37 & 0.15 & 0.15 \\ S_3 & 0.2 & 0.35 & 0.41 & 0.04 \\ S_4 & 0.1 & 0.2 & 0.3 & 0.4 \end{array}$$

$\sum = 1 \Rightarrow H(S_1)$

$\sum = 1 \Rightarrow H(S_2)$

$\Rightarrow H(S_3)$

$\Rightarrow H(S_4)$

- What entropy does s source with memory have?
- Each state S_k has a different distribution –> each state has a different entropy $H(S_k)$

$$H(S_k) = -\sum_i p(s_i|S_k) \cdot \log(p(s_i|S_k))$$

$H(S_1) =$

$H(S_2) =$

$\vdots$

$H(S_N) =$

- Global entropy (of the sources) = average entropy of the states

$$H(S) = \sum_k \underset{?}{p_k} H(S_k)$$

where p_k = probability that the source is in state S_k

- (i.e. after a very long sequence of messages, the fraction of time when the source was in state S_k)

Ergodic sources

- How to find out the weights p_k?
- They are known as the **stationary probabilities**
- p_k = probability that the source is in state S_k, after running for a very long time
  - (i.e. after a very long sequence of messages, the fraction of time when the source was in state S_k)
- We need to answer the following question:

If we know the state S_k at time n, what will be the state at time $n+1$?

Ergodic sources

- Let $p_i^{(n)}$ = the probability that source S is in state S_i at time n.
- In what state will it be at time $n + 1$? (after one more message)
  - i.e. what are the probabilities of the states at time $n + 1$?
- Just multiply with T

$$[p_1^{(n)}, p_2^{(n)}, ..., p_N^{(n)}] \cdot [T] = [p_1^{(n+1)}, p_2^{(n+1)}, ..., p_N^{(n+1)}]$$

- After one more message:

$$[p_1^{(n)}, p_2^{(n)}, ..., p_N^{(n)}] \cdot [T] \cdot [T] = [p_1^{(n+2)}, p_2^{(n+2)}, ..., p_N^{(n+2)}]$$

- For every new moment of time, one more multiplication with T

every new message

At time 0:

$$\begin{array}{cccc} S_1 & S_2 & S_3 & S_4 \\ \end{array}$$
$$\left[\begin{array}{cccc} 1 & 0 & 0 & 0 \\ p_1^{(0)} & p_2^{(0)} & p_3^{(0)} & p_4^{(0)} \end{array}\right] \cdot [T] =$$

At time 1:

$$\begin{array}{cccc} S_1 & S_2 & S_3 & S_4 \\ \end{array}$$
$$\left[\begin{array}{cccc} 0.4 & 0.3 & 0.2 & 0.1 \end{array}\right]$$

Ergodic sources

- In general, starting from time 0, after n messages the probabilities that the source is in a certain state are:

$$[p_1^{(0)}, p_2^{(0)}, ..., p_N^{(0)}] \cdot [T]^n = [p_1^{(n)}, p_2^{(n)}, ..., p_N^{(n)}]$$

$1 \quad 0 \quad \ldots \quad 0$

$0 \quad 0 \quad \ldots \quad 1$

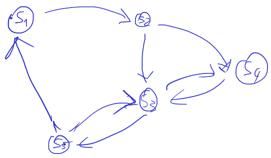
$\frac{1}{N} \quad \frac{1}{N} \quad \ldots \quad \frac{1}{N}$

Ergodicity

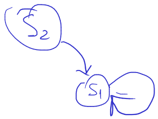
- A source (with memory) is called **ergodic** if every state can be reached from every state, in a finite number of steps.
- Property of ergodic sources:
  - After many messages, the probabilities of the states *become stationary* (converge to some fixed values), irrespective of the initial probabilities (no matter what state the source started from initially)

$$\lim_{n\to\infty}[p_1^{(n)}, p_2^{(n)}, ...p_N^{(n)}] = [p_1, p_2, ...p_N]$$

Ergodic source



Non-ergodic:



Finding the stationary probabilties

- How to find the value of the stationary probabilities?
- When n is very large, after n messages and after $n+1$ messages the probabilities are the same:

$$[p_1^{(n)}, p_2^{(n)}, \ldots p_N^{(n)}] \cdot [T] = [p_1^{(n+1)}, p_2^{(n+1)}, \ldots p_N^{(n+1)}]$$

$$[p_1 \; p_2 \; p_3 \; p_4] \cdot \begin{bmatrix} 0.4 & 0.3 & 0.2 & 0.1 \\ 0.33 & 0.37 & 0.15 & 0.15 \\ 0.2 & 0.35 & 0.41 & 0.04 \\ 0.1 & 0.2 & 0.3 & 0.4 \end{bmatrix}^{T} = [p_1 \; p_2 \; p_3 \; p_4]$$

- This is an equation system in matrix form
- One line should be removed (linear combination), and replaced with:

$$p_1 + p_2 + \ldots + p_N = 1$$

$$\begin{cases} 0.4p_1 + 0.33p_2 + 0.2p_3 + 0.1p_4 = p_1 \\ 0.3p_1 + 0.37p_2 + 0.35p_3 + 0.2p_4 = p_2 \\ 0.2p_1 + 0.15p_2 + 0.41p_3 + 0.3p_4 = p_3 \\ \cancel{0.1p_1 + 0.15p_2 + 0.04p_3 + 0.4p_4 = p_4} \\ p_1 + p_2 + p_3 + p_4 = 1 \end{cases}$$

- Solve the resulting system of equations, find values of p_k

Entropy of ergodic sources with memory

- The entropy of an ergodic source with memory is

$$H(S) = \sum_k p_k \underline{H(S_k)} = -\sum_k p_k \underbrace{\sum_i p(s_i|S_k) \cdot \log(p(s_i|S_k)))}_{H(S_k)}$$

Exercise

1. Consider a discrete source with memory, with the graphical representation given below. The states are defined as follows: $S_1 : s_1 s_1$, $S_2 : s_1 s_2$, $S_3 : s_2 s_1$, $S_4 : s_2 s_2$.

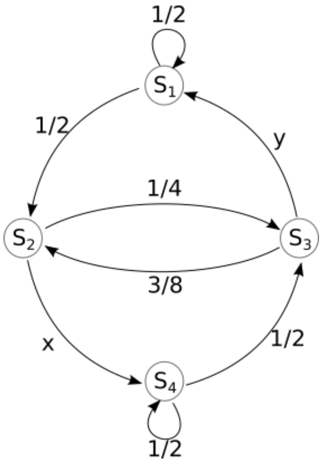


Figure 3: Graphical representation of the source

Exercise (continued)

Questions:

a. What are the values of x and y?
b. Write the transition matrix $[T]$;
c. Compute the entropy in state S_4;
d. Compute the global entropy of the source;
e. What are the memory order, m, and the number of messages of the source, n?
f. If the source is initially in state S_2, in what states and with what probabilities will the source be after 2 messages?

Example English text as sources with memory

(taken from Elements of Information Theory, Cover, Thomas)

- Memoryless source, equal probabilities:

  XFOML RXKHRJFFJUJ ZLPWCFWKCYJ
  FFJEYVKCQSGXYD QPAAMKBZAACIBZLHJQD

  $S: \begin{pmatrix} a & \ldots & z & \sqcup \\ \frac{1}{27} & \ldots & \frac{1}{27} & \frac{1}{27} \end{pmatrix}$

  xzabzfg

- Memoryless source, probabilities of each letter as in English:

  OCRO HLI RGWR NMIELWIS EU LL NBNESEBYA TH EEI
  ALHENHTTPA OOBTTVA NAH BRL

  $S: \begin{pmatrix} a & \ldots & z & \sqcup \\ & & & \end{pmatrix}$

- Source with memory $m = 1$, frequency of pairs as in English:

  ON IE ANTSOUTINYS ARE T INCTORE ST BE S DEAMY
  ACHIN D ILONASIVE TUCOOWE AT TEASONARE FUSO
  TIZIN ANDY TOBE SEACE CTISBE

  TH
  EA

  $\begin{array}{c|c} & a \quad b \quad \ldots \quad z \\ \hline a & \\ b & \\ \vdots & \\ z & \end{array}$

Example English text as sources with memory

e th ?

- Source with memory $m = 2$, frequency of triplets as in English:

IN NO IST LAT WHEY CRATICT FROURE BERS GROCID PONDENOME OF DEMONSTURES OF THE REPTAGIN IS REGOACTIONA OF CRE

- Source with memory $m = 3$, frequency of 4-plets as in English:

THE GENERATED JOB PROVIDUAL BETTER TRAND THE DISPLAYED CODE, ABOVERY UPONDULTS WELL THE CODERST IN THESTICAL IT DO HOCK BOTHE MERG. (INSTATES CONS ERATION. NEVER ANY OF PUBLE AND TO THEORY. EVENTIAL CALLEGAND TO ELAST BENERATED IN WITH PIES AS IS WITH THE)

NEV ?

i 95%

V E H ?

DISPL A

26^3 combinations =

$T = 17576 \times 17576$

Example application

VEHI?LE

C

- Suppose we receive a text with random missing letters
- We need to fill the blanks with the appropriate letters
- How?
  - build a model: source with memory of some order
  - fill the missing letter with the most likely letter given by the model

Chapter summary

- Information of a message: $i(s_k) = -\log_2(p(s_k))$
- Entropy of a memoryless source:
  $H(S) = \sum_k p_k i(s_k) = -\sum_k p_k \log_2(p_k)$
- Properties of entropy:
  1. $H(S) \geq 0$
  2. Is maximum when all messages have equal probability ($H_{max}(S) = \log(n)$)
  3. *Diversfication* of the source always increases the entropy
- Sources with memory: definition, transitions
- Stationary probabilities of ergodic sources with memory:
  $[p_1, p_2, ...p_N] \cdot [T] = [p_1, p_2, ...p_N]$, $\sum_i p_i = 1$.
- Entropy of sources with memory:

$$H(S) = \sum_k p_k H(S_k) = -\sum_k p_k \sum_i p(s_i|S_k) \cdot \log(p(s_i|S_k))$$