Information Measure and Entropy

MSDM 5058

Prepared by S.P. Li

Everything, from very little(IT from BIT)

In 1989, the eminent American physicist John Wheeler published an enigmatic essay titled ***Information, Physics, Quantum: The Search for Links***.

Wheeler makes his ambitious project obvious from the start:

This report reviews what quantum mechanics and information theory have to tell us about the age-old question: why exist?

All things physical are information-theoretic in origin.

J. Wheeler, *Information, physics, quantum: The search for links*, in Complexity, Entropy, and the Physics of Information, ed. by W. Zurek (Addison-Wesley, Redwood City, 1990)

What is Information?

The communication or reception of knowledge or intelligence.

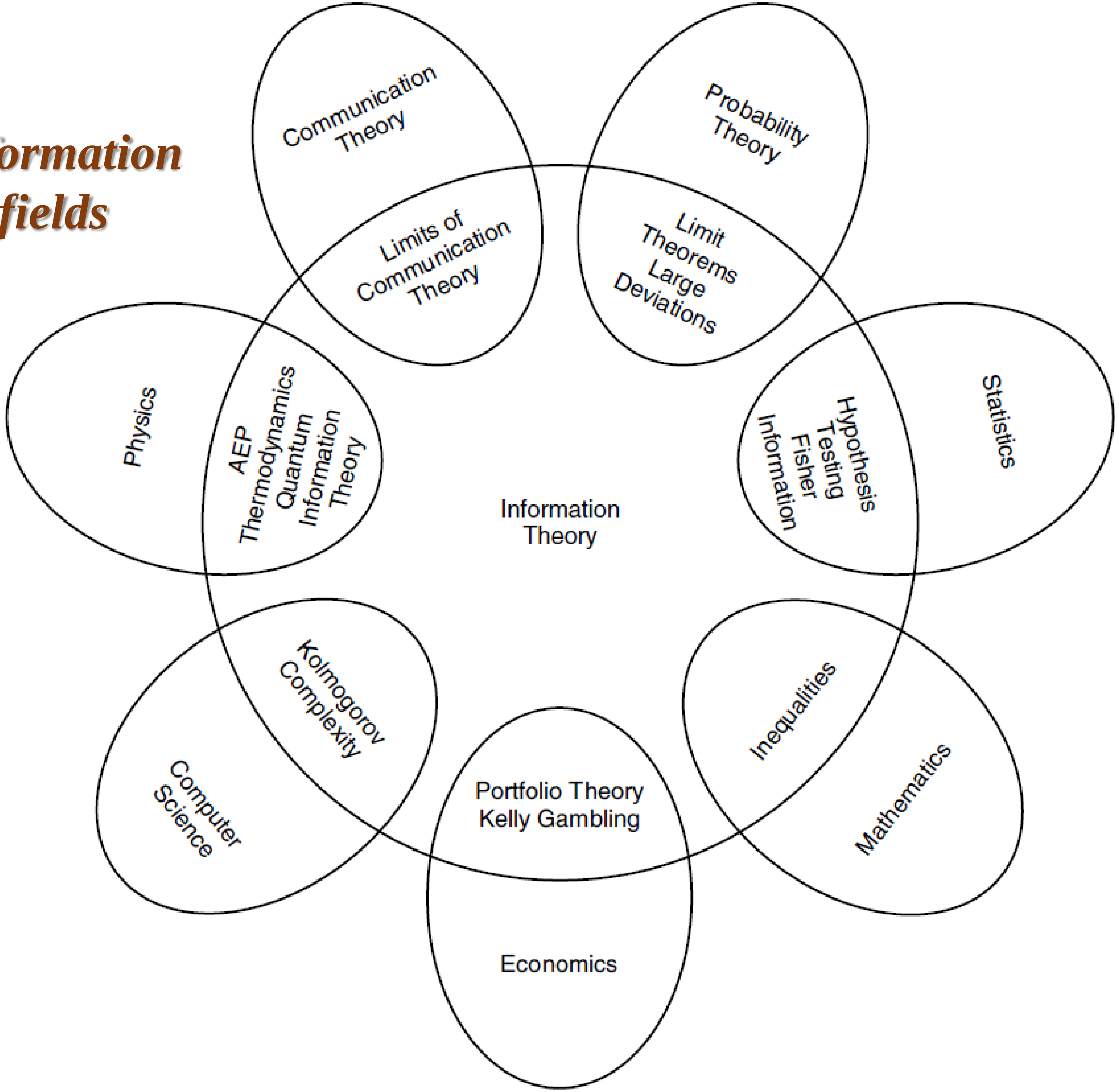
(Merriam-Webster Dictionary)

What is Information Theory?

Information theory is the scientific study of the quantification, storage, and communication of information. The field was fundamentally established by the works of Harry Nyquist and Ralph Hartley in the 1920s, and Claude Shannon in the 1940s. The field is at the intersection of *probability theory*, *statistics*, *computer science*, *statistical mechanics*, *information engineering*, *and electrical engineering*.

(Wikipedia)

Relationship of information theory to other fields


Communication Theory
Probability Theory
Limits of Communication Theory
Limit Theorems Large Deviations
Physics
AEP Thermodynamics Quantum Information Theory
Information Theory
Hypothesis Testing Fisher Information
Statistics
Kolmogorov Complexity
Computer Science
Portfolio Theory Kelly Gambling
Economics
Inequalities
Mathematics

Information theory is concerned with *characterizing message sources* rather than with *characterizing particular messages*. According to Claude Shannon, whose 1948 paper "The Mathematical Theory of Communication" initiated the study of information theory,

> These semantic aspects of communication are irrelevant to the engineering problem. The significant aspect is that the actual message is one *selected from a set of* possible messages. The system must be designed to operate for each possible selection, not just the one which will actually be chosen since this is unknown at the time of design.

Thus, two messages, one of which is heavily loaded with meaning and the other of which is pure nonsense, can be exactly equivalent, from the present viewpoint of Information Theory.

Example:

The *Earth* is larger than the *Moon*.
The *Moon* is larger than the *Earth*.

A Very Short History of Information Theory

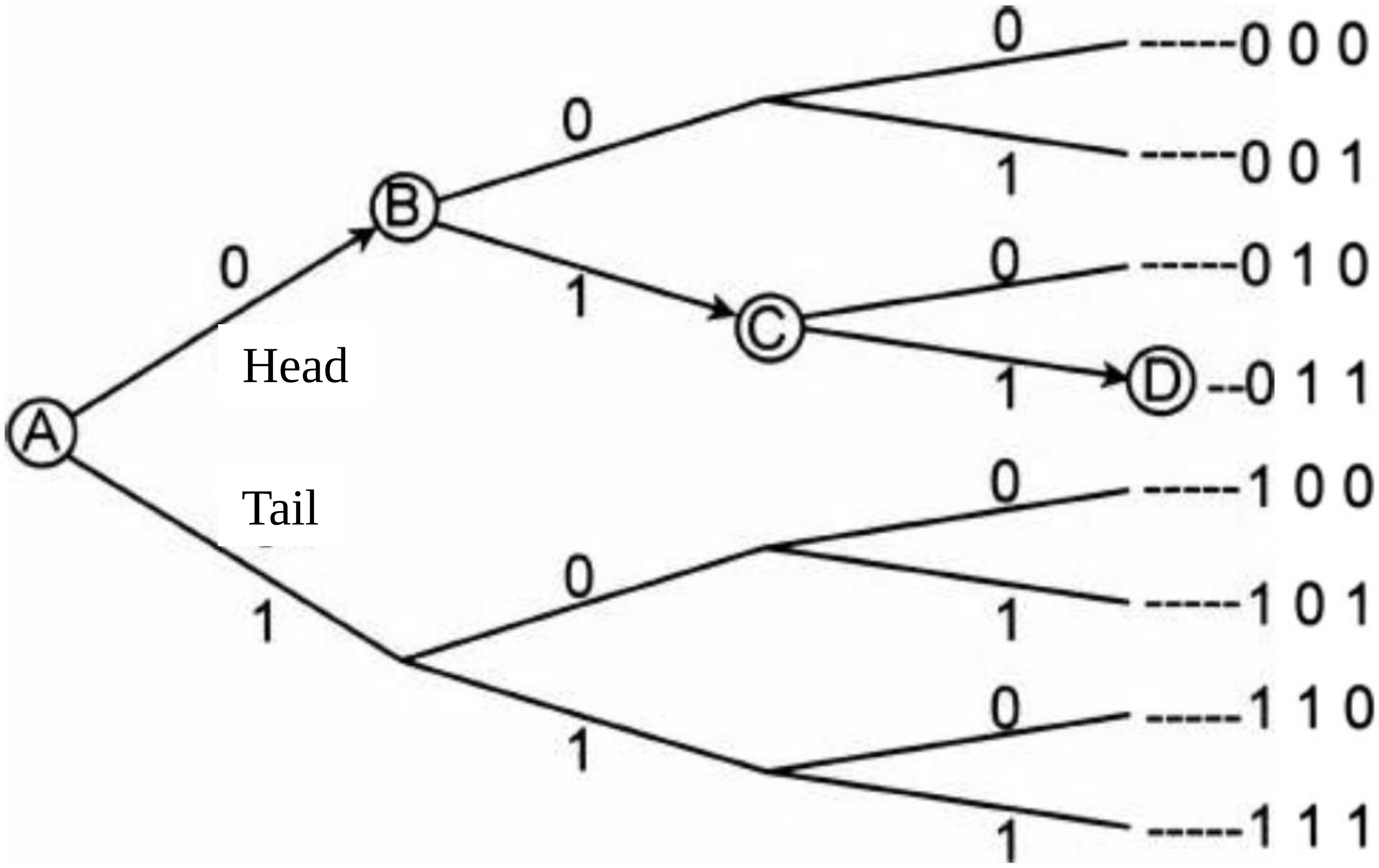
- Boltzmann (1875) and Gibbs (1902) on thermodynamic entropy
- Wiener (1927) on signal processing
- Nyquist (1928) on sampling theory
- Hartley (1928) on information transmission
- Shannon (1948) on a mathematical theory of information

Consider flipping a coin:

- If it is a biased coin, and you always get the ‘head’, what will you get in the next flip?
    - The answer is obviously ‘head’. There is ***no need for extra information*** to know the outcome.

- If it is a fair coin, you need ***extra information*** to know the outcome (not by pure guessing).
    - For example, you ask the question “ Is it a ‘head’? ” By answering ‘yes’ or ‘no’, you know the outcome.

- Next, consider flipping two fair coins. How many questions do you need to ask in order to know the outcome?



What is the outcome of flipping *n* fair coins? One starts by asking " Is the first coin a 'Head'? ", " Is the second coin a 'Head'? ", …, by obtaining the information needed to determine the outcome. Each question is denoted by a 'fork' in the figure above.

We can reformulate this in a more general way if we use *n* to represent the number of forks, and *m* to represent the number of final outcomes. If there are *n* forks, then you have effectively chosen from $m = 2^n$ final outcomes, or in other words, $n = \log_2 m$.

Since the decision at each fork requires one binary digit (0 or 1) of information, *n* forks require *n* binary digits (***bits***) of information, which allow you to choose from 2^n equally probable outcomes.

To summarize:

- One ***bit*** is the amount of information required to choose between two equally probable outcomes (e.g. head/tail), whereas a binary digit is the value of a binary variable, which can adopt one of two possible values, 0 or 1.

- If you have n bits of information, you can choose from $m = 2^n$ equally probable outcomes. Alternatively, if you have to choose between m equally probable outcomes, you need $n = \log_2 m$ bits of information.

Question:

“Can we find a measure of how much ‘choice’ is involved in the selection of the event, or of how much uncertain we are of the outcome?”

(Shannon)

For a mathematical definition of information, Shannon proposed a minimal set of properties:

- ***Continuity:*** The amount of information associated with an outcome increases or decreases continuously as the probability of that outcome changes.
- ***Symmetry:*** The amount of information associated with a sequence of outcomes does not depend on the order in which those outcomes occur.
- ***Maximal Value:*** The amount of information associated with a set of outcomes cannot be increased if those outcomes are already equally probable.
- ***Additive:*** The information associated with a set of outcomes is obtained by adding the information of individual outcomes.

Information Measure and Probability

The information associated with the occurrence of an event

- A rare event will have small probability so its occurrence provides much information. The more certain we know about something, the higher the probability of its occurrence, thus the less information it contains.
- ***Information Entropy (Average Shannon Information)*** is a measure of ***average uncertainty***
- Suppose we are given a coin, and told that when flipped, it lands heads up 90% of the time. When this coin is flipped, we expect it to land heads up, so when it does so we are less surprised than when it lands tails up. The more improbable a particular outcome is, the more surprised we are to observe it. Therefore, one can also define ***Shannon Information (entropy)*** to be a measure of ***surprise***.

Information Measure for Independent Events

Consider two independent events *A* and *B*:

Event *A* occurs with probability *p* ; event *B* with probability *q*, the joint probability of both *A* and *B* will occur is $p \times q$

What is the information associated with this joint independent event of A and B occurring?

Information for Joint event A and B
= information for A + information for B

Logarithm for Information Measure

Information for independent event should be ***additive***

» Logarithm should be used for two reasons:

- Information measure for single event is more if the probability is less, and vice versa
- Information measure for two independent event A and B is the sum of the information measure for each event. Since probability for A and B is multiplied, if A and B are independent, the only way to convert a product into a sum is by taking logarithm

$$\ln\left(\frac{1}{pq}\right) = \ln\left(\frac{1}{p}\right) + \ln\left(\frac{1}{q}\right)$$

Quantifying Uncertainty

Axiom 1: $S(1) = 0$ (i.e., There is no surprise or uncertainty in hearing that an event that is sure to occur has indeed occurred.)

Axiom 2: $S(p)$ is a strictly decreasing function of p; that is, if $p < q$, then $S(p) > S(q)$.

Axiom 3: $S(p)$ is a continuous function of p.

Axiom 4: $S(pq) = S(p) + S(q)$; $0 < p \leq 1,\ 0 < q \leq 1$

Theorem: If $S(.)$ satisfies Axioms 1 through 4, then $S(p) = -C \log_2 p$

We can then define a quantity $H[X]$, known in information theory as the ***entropy*** of the random variable X as

$$H[X] = -\sum_{j=1}^{l} p_j \log p_j$$

==> *If a base 2 logarithm is used, the measure of information is the bit.*

More Examples:

Example 1: A fair coin has two values with equal probability. Its entropy is 1 bit.

Example 2: Imagine throwing M fair coins: the number of all possible outcomes is 2^M. The entropy equals M bits.

Example 3: A fair dice with M faces has entropy $\log_2 M$.

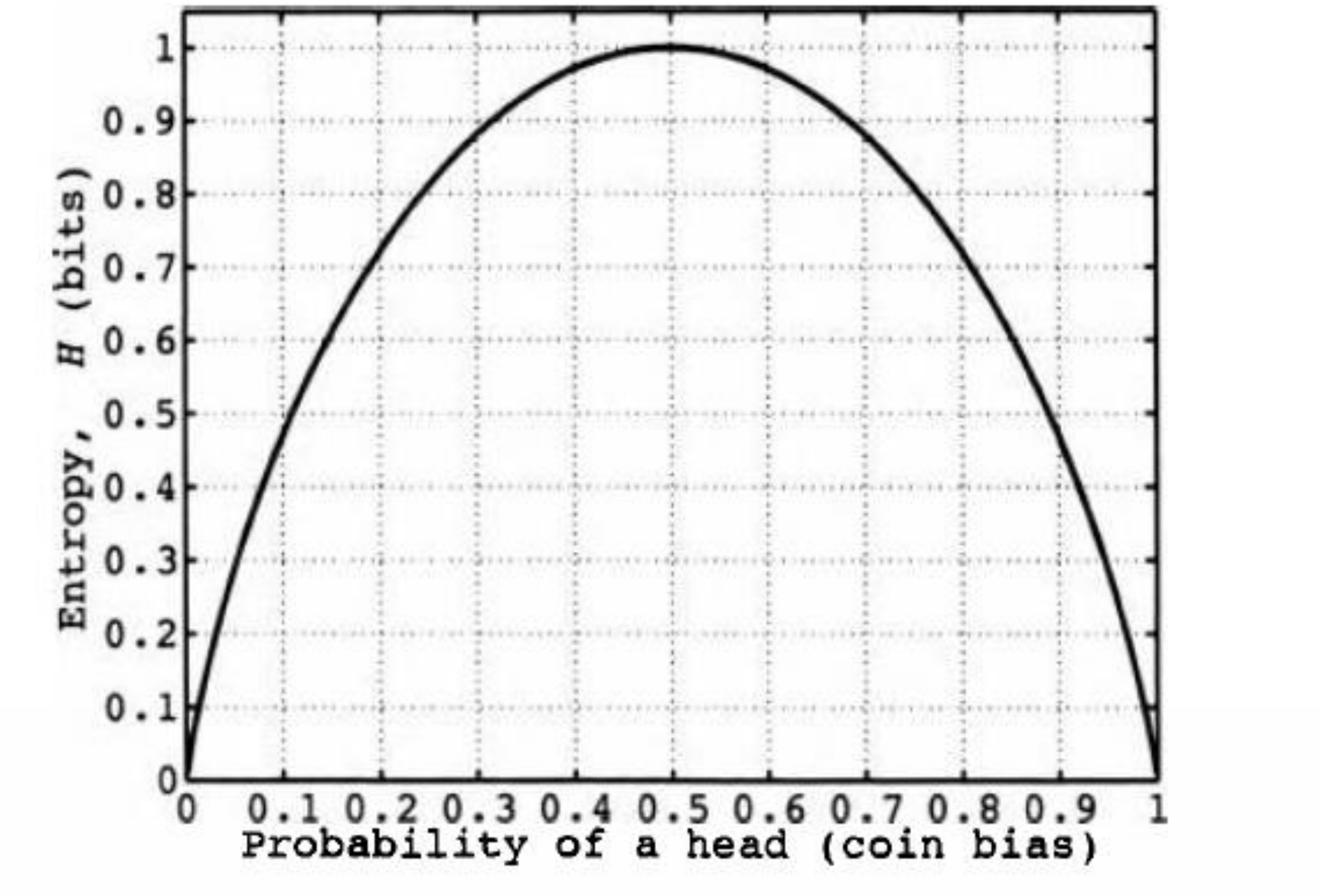
Example 4: *Bernoulli process*.

A Bernoulli random variable X can take values 0, 1 with probabilities $p(0) = q$, $p(1) = 1 - q$. (*Refer to the case of flipping a coin*). Its entropy is

$$H[X] = -q \log_2 q - (1 - q) \log_2(1 - q)$$

which is plotted as a function of q in the next slide. This entropy vanishes when $q = 0$ or 1 because the outcome is certain; it is maximal at $q = ½$, when the uncertainty of the outcome is maximal. (**Exercise: Show that $H[X]$ is concave in q.)

Example 4 (cont’d)
Bernoulli process.



When $q \neq ½$. (Imagine an unfair coin) Let’s choose $q = 0.9$. In this case,

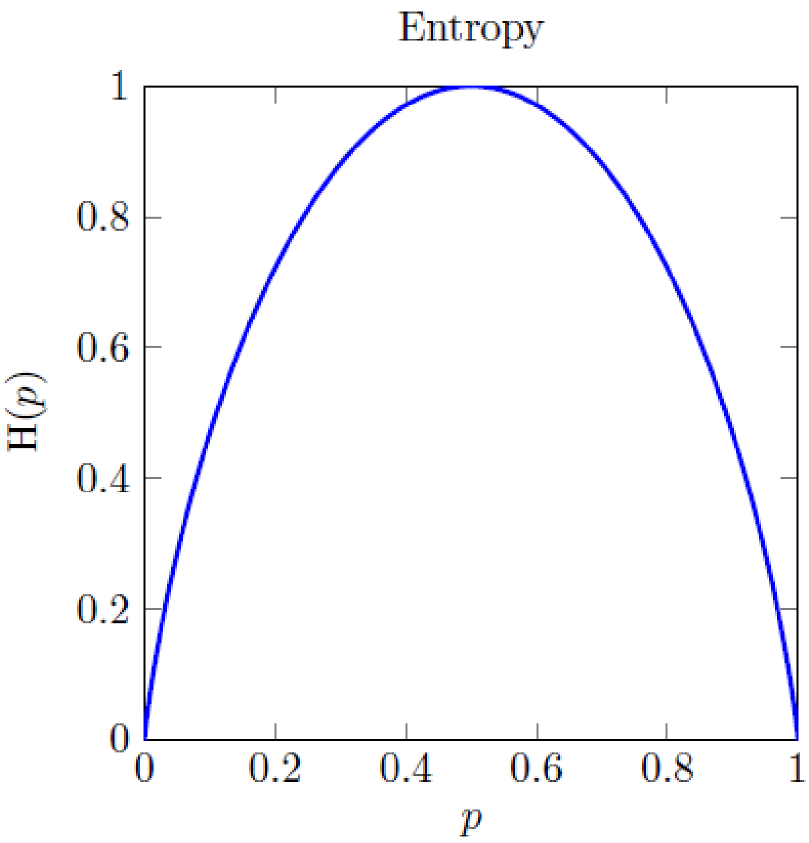
$$H[X] = -0.9 \log_2 0.9 - 0.1 \log_2 0.1 = 0.469 \text{ bit/flip}$$

Since $m = 2^{H(X)}$, we get $m = 1.38$ equally probable outcomes!!!

Quick Review of Information Entropy

$$H[X] = -\sum_{x \in \chi} p(x) \log p(x)$$

$$H[Y] = -\sum_{y \in \Upsilon} p(y) \log p(y)$$



Properties:

- continuous
- symmetric
- $H[p_0, \dots, p_n] \leq H\left[\frac{1}{n}, \dots, \frac{1}{n}\right] = \log n$
- $H[p_0, \dots, p_{n-1}, p_n] = H[p_0, \dots, p_{n-1} + p_n] + (p_{n-1} + p_n) H\left[\frac{p_{n-1}}{p_{n-1}+p_n}, \frac{p_n}{p_{n-1}+p_n}\right]$

Quick Review of Information Entropy (cont'd)

Joint Entropy

Consider a pair of random variables (X,Y) with a joint distribution $p(x,y)$, we can define their joint entropy as

$$H[X,Y] = -\sum_{\substack{x\in\chi \\ y\in\Upsilon}} p(x,y)\log p(x,y)$$

Properties:

- $0 \le H[X,Y]$ with equality iff X, Y is deterministic
- $H[X,Y] = H[X] + H[Y] \leftrightarrow p(x,y) = p(x)p(y)$
- $H[X,Y] \le \sum_{Z=\chi,\Upsilon} \log|Z|$, with equality iff X, Y is uniformly distributed over $\chi \times \Upsilon$

Quick Review of Information Entropy (cont'd)

Joint Entropy (cont'd)

For a set of random variables X_1, …, X_n with a joint distribution $p(x_1, \ldots, x_n)$, we can define their joint entropy as

$$H[X_1, \ldots, X_n] = -\sum_{x_1,\ldots,x_n \in \chi} p(x_1, \ldots, x_n) \log p(x_1, \ldots, x_n)$$

Properties:

- $H[p] \geq 0$
- $H_b[p] = (\log_b a)\, H_a[p]$
- $H[X_1, \ldots, X_n] \leq \sum_{i=1}^{n} H[X_i]$ (i.e., Joint entropy ≤ Sum of Marginal entropies), with equality iff X_i are independent
- $H[p] \leq \log n$ with equality only iff p is uniform on χ
- $H[p]$ is concave in p (**Use *Log Sum Inequality* to prove.)

Quick Review of Information Entropy (cont'd)

Conditional Entropy

$$H[X|Y] = -\sum_{\substack{x\in\chi \\ y\in\Upsilon}} p(x,y)\log p(x|y) = \sum_{y\in\Upsilon} p(y)H[X|Y=y]$$

Properties:

- $0 \leq H[X|Y] \leq H[X]$
- $H[X|Y] = H[X] \leftrightarrow p(x,y) = p(x)p(y)$
- $H[X|Y] = 0 \leftrightarrow \exists f$, such that $X = f(Y)$

** Note that $0 \leq H[X|Y]$ is ***not*** true quantum mechanically!

Quick Review of Information Entropy (cont'd)

Theorem (*Chain Rule*):

$$H[X,Y] = H[X] + H[Y|X] = H[Y] + H[X|Y]$$

Interpretation:

Amount of uncertainty of (*X; Y*)
= Amount of uncertainty of *Y* + Amount of uncertainty of *X* after knowing *Y*

Proof:

$$H[X,Y] = -\sum_{\substack{x\in\chi \\ y\in\Upsilon}} p(x,y)\log p(x,y) = -\sum_{\substack{x\in\chi \\ y\in\Upsilon}} p(x,y)\log[p(x|y)p(y)]$$

$$= -\sum_{\substack{x\in\chi \\ y\in\Upsilon}} p(x,y)\log[p(x|y)] - \sum_{\substack{x\in\chi \\ y\in\Upsilon}} p(x,y)\log[p(y)] = H(Y) + H[X|Y]$$

Quick Review of Information Entropy (cont'd)

Theorem (*Conditioning reduces entropy*):

$$H[X|Y] \leq H[X] , \qquad \text{with equality iff } X \text{ is independent of } Y$$

Interpretation:

The more one learns, the less the uncertainty is. If and only if what you have learned is independent of your target, the amount uncertainty of your target remains the same.

** Note that while $H[X|Y] \leq H[X]$ is always true, it is possible that

$$H[X|Y=y] < H[X] , \qquad \text{or} \qquad H[X|Y=y] > H[X] .$$

Exercise: Construct examples for the above two cases respectively

Quick Review of Information Entropy (cont'd)

Proof: (By definition and Jensen's Inequality)

$$H[X|Y] - H[X] = \sum_{\substack{x\in\chi \\ y\in\Upsilon}} p(x,y)\log\left[\frac{p(x)}{p(x|y)}\right] = \sum_{\substack{x\in\chi \\ y\in\Upsilon}} p(x,y)\log\left[\frac{p(x)p(y)}{p(x,y)}\right]$$

$$\leq \log\left(\sum_{\substack{x\in\chi \\ y\in\Upsilon}} p(x,y)\left[\frac{p(x)p(y)}{p(x,y)}\right]\right) = \log\left(\sum_{\substack{x\in\chi \\ y\in\Upsilon}} p(x)p(y)\right) = \log(1) \ = 0$$

Jensen's Inequality

Jensen's Inequality:

If f is a convex function and X is a random variable, $E[f(X)] \geq f(E[X])$.

Quick Review of Information Entropy (cont'd)

Theorem (*Chain Rule*):

The chain rule can be generalized to more than two random variables as

$$H[X_1, \dots, X_n] = \sum_{i=1}^{n} H[X_1 | X_1, \dots, X_{n-1}]$$

** Proof is left as an exercise for you

Theorem (*Conditioning reduces entropy*):

Conditioning reduces entropy can be generalized to more than two random variables

$$H[X|Y, Z] \leq H[X|Y]$$

** Proof is left as an exercise for you

Quick Review of Information Entropy (cont'd)

Mutual Information

$$I[X\!:Y] = \sum_{\substack{x\in\chi \\ y\in\Upsilon}} p(x,y)\log\frac{p(x,y)}{p(x)p(y)} = H[X] - H[X|Y] = H[Y] - H[Y|X] = H[X] + H[Y] - H[X,Y]$$

Interpretation:

Can be viewed as a measure of dependency between X and Y. If X is determined by Y (highly dependent), $I[X{:}Y]$ is maximized. If X is independent of Y, $I[X{:}Y] = 0$.

Properties:

- $0 \le I[X{:}Y] \le \min\{H[X], H[Y]\}$
- $I[X{:}Y] = I[Y{:}X]$
- Invariant with respect to transformations of X and Y provided the transformations are invertible

** Note: The mutual information between X and itself is equal to its own entropy: $I[X{:}\,X] = H[X]$ since $H[X|X] = 0$. Hence, the entropy is also called "self information" in some literatures.

Quick Review of Information Entropy (cont'd)

Conditional Mutual Information

For a tuple of jointly distributed random variables (X, Y, Z), the mutual information between X and Y, given Z is $I[X{:}Y|Z] = H[X|Z] – H[X|Y,Z]$, we have

$$I[X:Y|Z] = H[X|Z] - H[X|Y,Z] = H[Y|Z] - H[Y|X,Z] = H[X|Z] + H[Y|Z] - H[X,Y|Z]$$

Similarly, we have

- $I[X{:}Y|Z] \geq 0$, with equality iff X, Y are independent given Z, that is $X – Z – Y$ forms a Markov chain
- $I[X{:}Y|Z] \leq H[X|Z]$, with equality iff X is a deterministic function of Y and Z

Quick Review of Information Entropy (cont'd)

Chain Rule for Mutual Information

The chain rule for mutual information with more than two random variables can be defined as follows

$$I[X\!:Y_1,\dots,Y_n] = \sum_{i=1}^{n} I[X\!:Y_i|Y_1,\dots,Y_{i-1}]$$

** The chain rule for mutual information can be proven by definition and the chain rule for entropy and is left as an exercise for you

** Show that $I[X\!:\!Z] \leq I[X\!:Y, Z]$ and $I[X\!:\!Y|Z] \leq I[X\!:Y, Z]$.

Quick Review of Information Entropy (cont'd)

***Data Processing Inequality* (*Theorem*)**

For a Markov chain $X – Y – Z$, that is, $p(x, y, z) = p(x)\, p(y|x)\, p(z|y)$, we have $I[X{:}Y] \geq I[X{:}Z]$.

Interpretation:

The Markov chain $X – Y – Z$ implies that the information of X that Z can provide is already contained in Y. Therefore, the amount of information of X that can be inferred by $Z \leq$ the amount of information of X that can be inferred by Y.

Proof:

Since $X – Y – Z$, we have $I[X{:}Z|Y] = 0$. Thus,

$I[X{:}Y, Z] = I[X{:}\,Y] + I[X{:}Z|Y] = I[X{:}\,Y]$ (since $I[X{:}Z|Y] = 0$)

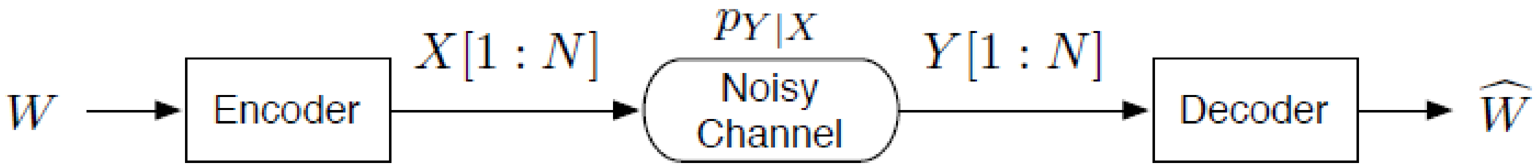
$I[X{:}Y, Z] = I[X{:}\,Z] + I[X{:}Y|Z]$ (Chain Rule)

$\Rightarrow I[X{:}Y] = I[X{:}\,Z] + I[X{:}Y|Z] \geq I[X{:}Z]$ (since $I[X{:}Y|Z] \geq 0$)

Quick Review of Information Entropy (cont'd)

Data Processing Inequality **(*Application*)**

Markov chains are common in communication systems (will be discussed later). For example, in channel coding (without feedback), the message W, the channel input $X^N := X[1 : N]$, the channel output $Y^N := Y[1 : N]$, and the decoded message $\widehat{W}$ form a Markov chain $W - X^N - Y^N - \widehat{W}$,



Data processing inequality is crucial in obtaining the *impossibility* results in information theory.

Quick Review of Information Entropy (cont’d)

Conditioning Reduces Mutual Information?

- Does conditioning reduce mutual information?
- Does conditioning reduce the dependency between two random variables?
- The answer is both “yes” and “no”: sometimes “yes”, and sometimes “no”.

Quick Review of Information Entropy (cont'd)

Conditioning Increases Mutual Information?

Let X and Y be i.i.d. Bernoulli (Ber $(\frac{1}{2})$) random variables, and $Z := X \oplus Y$.

Evaluate $I[X\text{: } Y, Z]$ and show that $I[X\text{:}Y|Z] > I[X\text{: } Y]$.

$$I[X\text{:}Y|Z] = H[X|Z] - H[X|Y,Z] = H[X|Z] - H[X|Y, X \ \oplus Y]$$
$$= H[X|Z] - H[X|Y,X] = H[X|Z] = H[X] = 1$$

Since $I[X\text{: } Y] = 0$, therefore, $1 = I[X\text{: } Y,|Z] > I[X\text{: } Y] = 0$.

Conditioning Decreases Mutual Information?

For a Markov chain X – Y – Z, we have $I[X\text{:}Y] > I[X\text{:}Y|Z]$.

Quick Review of Information Entropy (cont'd)

Convexity and Concavity of Mutual Information

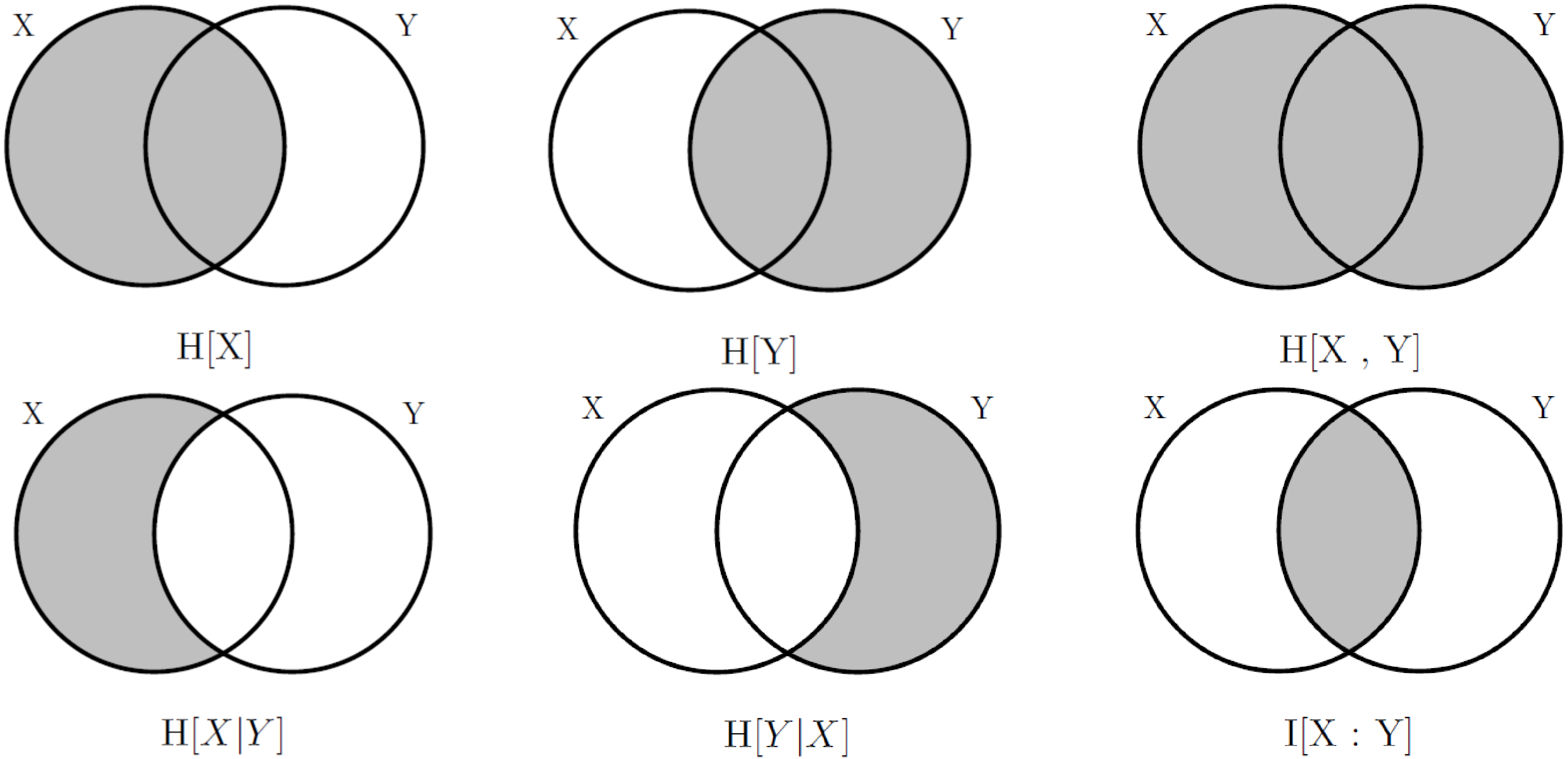
The convexity/concavity properties of mutual information are very useful in computing channel capacity and rate distortion functions.

Let $(X, Y) \sim p(x,y) = p(x)\, p(y|x)$

- With $p(y|x)$ fixed, $I[X{:}Y]$ is a ***concave*** function of $p(x)$ (** Use Log Sum Inequality to prove.)
- With $p(x)$ fixed, $I[X{:}Y]$ is a ***convex*** function of $p(y|x)$ (** Use Log Sum Inequality to prove.)

Quick Review of Information Entropy (cont’d)

Venn diagrams for (a) H(X), (b) H(Y), (c) H[X,Y],(d) H(X|Y), (e) H(Y|X), (f) I(X;Y)



H[X]

H[Y]

H[X , Y]

H[X|Y]

H[Y|X]

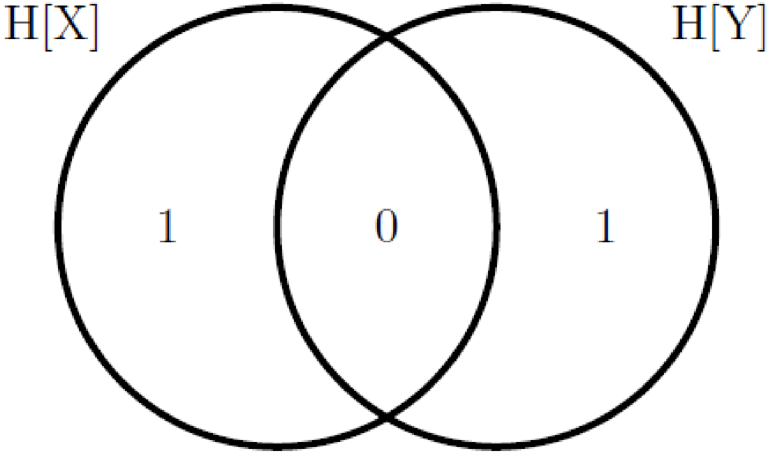
I[X : Y]

Quick Review of Information Entropy (cont'd)

A few examples:

Example 1: Independent

X	Y	p
0	0	¼
0	1	¼
1	0	¼
1	1	¼

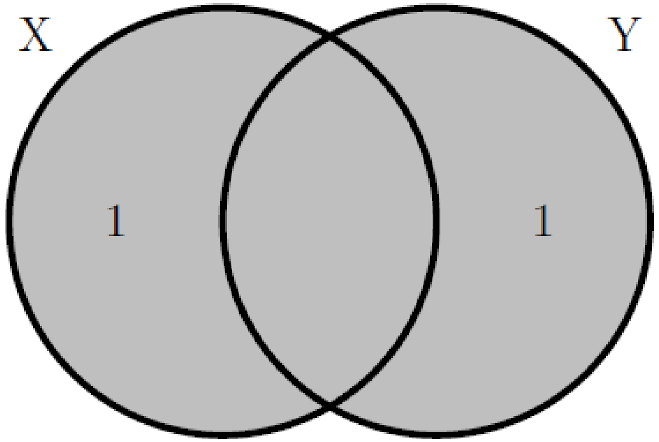


Quick Review of Information Entropy (cont'd)

A few examples:

Example 1: Independent (cont'd)

X	Y	p
0	0	¼
0	1	¼
1	0	¼
1	1	¼



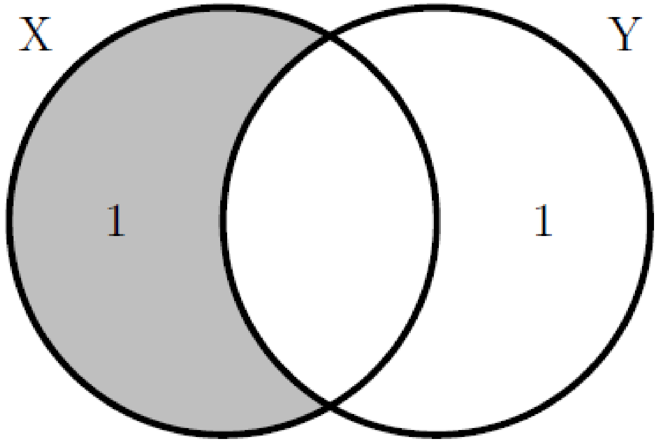
H[X,Y] = 2 bit

Quick Review of Information Entropy (cont'd)

A few examples:

Example 1: Independent (cont'd)

X	Y	p
0	0	¼
0	1	¼
1	0	¼
1	1	¼



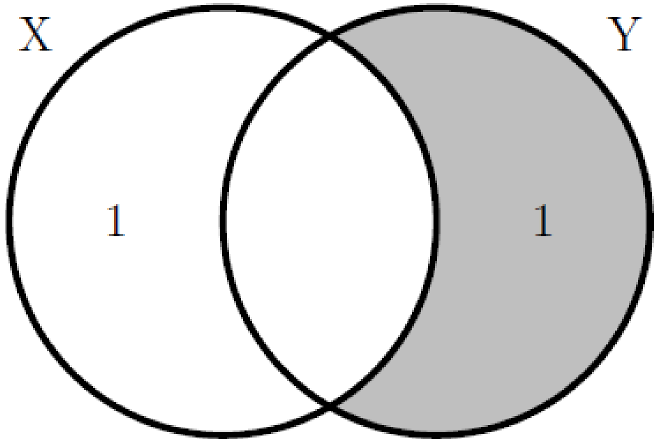
$H[X,Y]$ = 2 bit
$H[X|Y]$ = 1 bit

Quick Review of Information Entropy (cont'd)

A few examples:

Example 1: Independent (cont'd)

X	Y	p
0	0	¼
0	1	¼
1	0	¼
1	1	¼



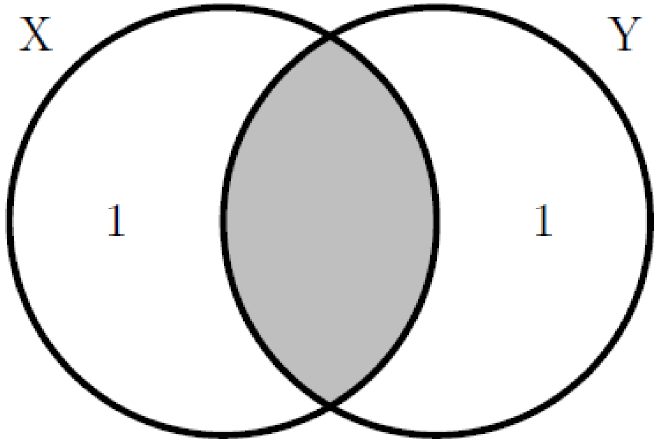
$H[X,Y]$ = 2 bit
$H[X|Y]$ = 1 bit
$H[Y|X]$ = 1 bit

Quick Review of Information Entropy (cont'd)

A few examples:

Example 1: Independent (cont'd)

X	Y	p
0	0	¼
0	1	¼
1	0	¼
1	1	¼



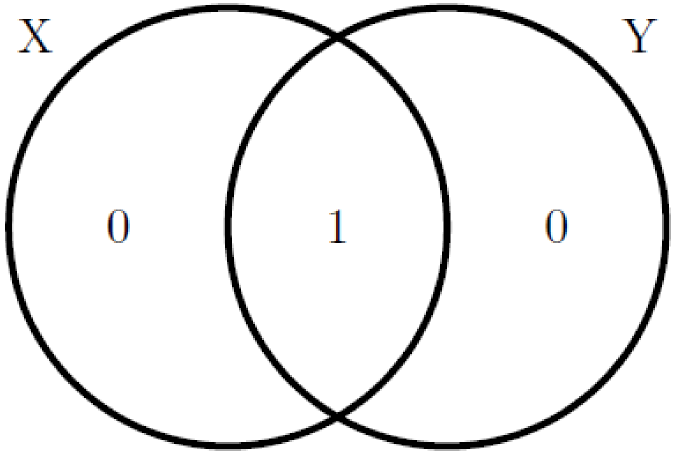
$H[X,Y]$ = 2 bit
$H[X|Y]$ = 1 bit
$H[Y|X]$ = 1 bit
$I[X{:}Y]$ = 0 bit

Quick Review of Information Entropy (cont'd)

A few examples:

Example 2: Redundant

X	Y	p
0	0	½
1	1	½



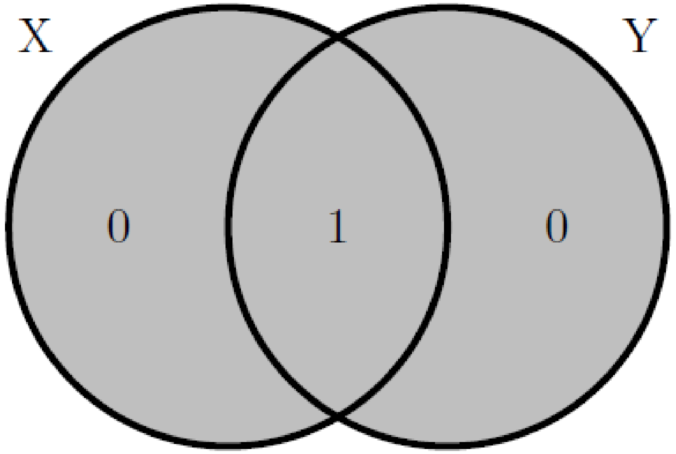
Quick Review of Information Entropy (cont'd)

A few examples:

Example 2: Redundant (cont'd)

X	Y	p
0	0	½
1	1	½

$H[X,Y] = 1$ bit

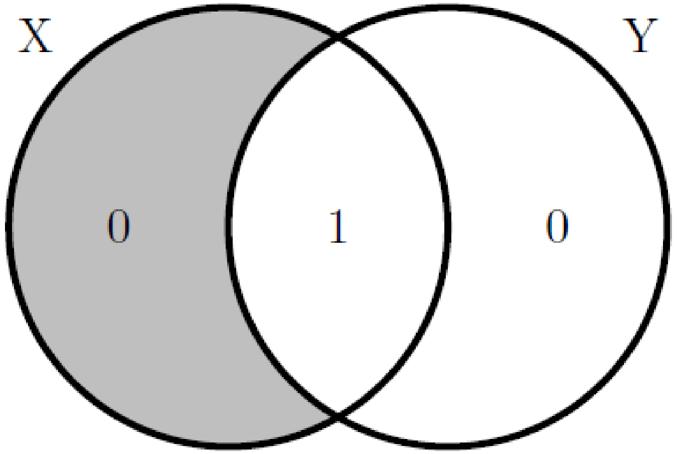


Quick Review of Information Entropy (cont'd)

A few examples:

Example 2: Redundant (cont'd)

X	Y	p
0	0	½
1	1	½



$H[X,Y]$ = 1 bit
$H[X|Y]$ = 0 bit

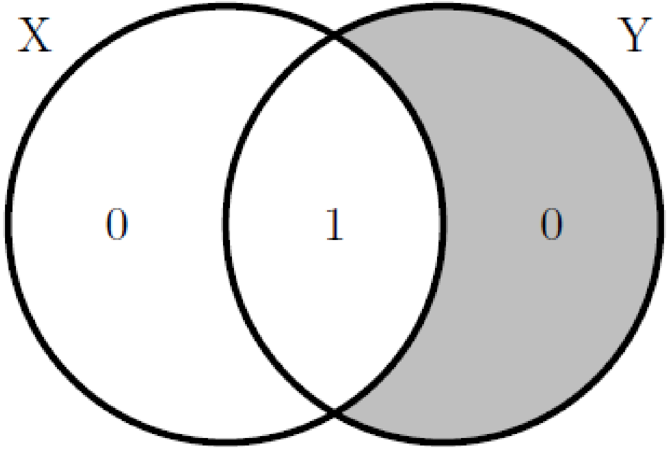
Quick Review of Information Entropy (cont'd)

A few examples:

Example 2: Redundant (cont'd)

X	Y	p
0	0	½
1	1	½

$H[X,Y]$ = 1 bit
$H[X|Y]$ = 0 bit
$H[Y|X]$ = 0 bit



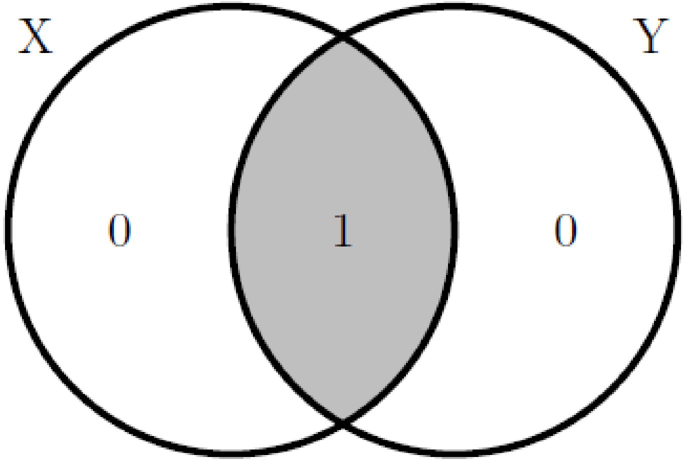
Quick Review of Information Entropy (cont'd)

A few examples:

Example 2: Redundant (cont'd)

X	Y	p
0	0	½
1	1	½

$H[X,Y]$ = 1 bit
$H[X|Y]$ = 0 bit
$H[Y|X]$ = 0 bit
$I[X{:}Y]$ = 1 bit

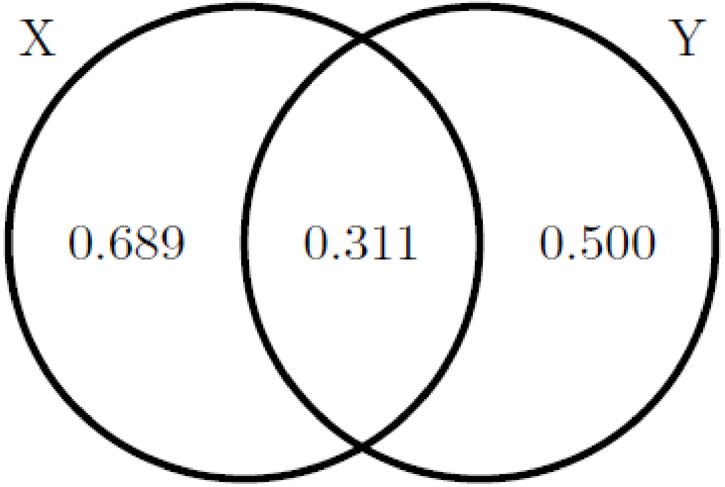


Quick Review of Information Entropy (cont'd)

A few examples:

Example 3: Generic

X	Y	p
0	0	½
1	0	¼
1	1	¼



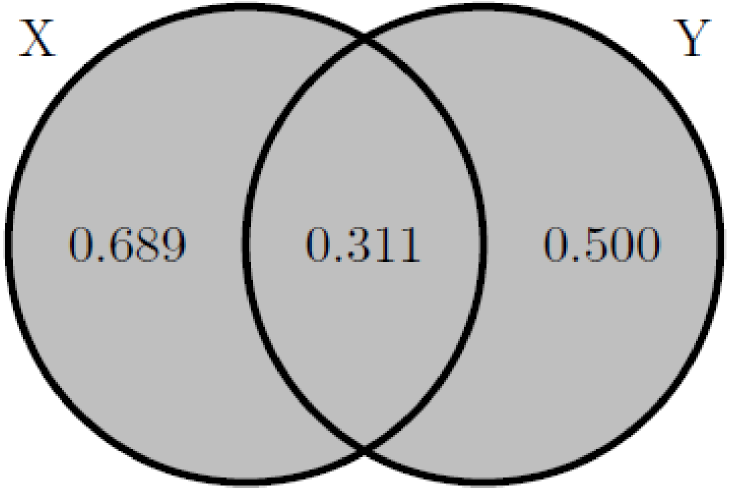
Quick Review of Information Entropy (cont'd)

A few examples:

Example 3: Generic (cont'd)

X	Y	p
0	0	½
1	0	¼
1	1	¼

$H[X,Y] = 1.5$ bit

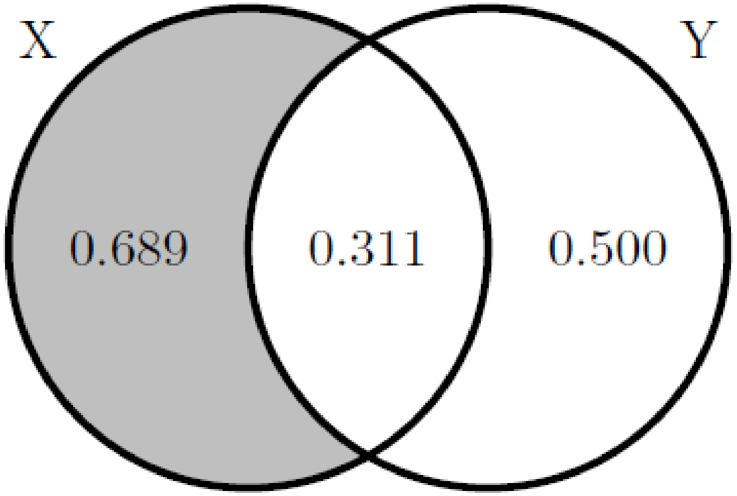


Quick Review of Information Entropy (cont'd)

A few examples:

Example 3: Generic (cont'd)

X	Y	p
0	0	½
1	0	¼
1	1	¼



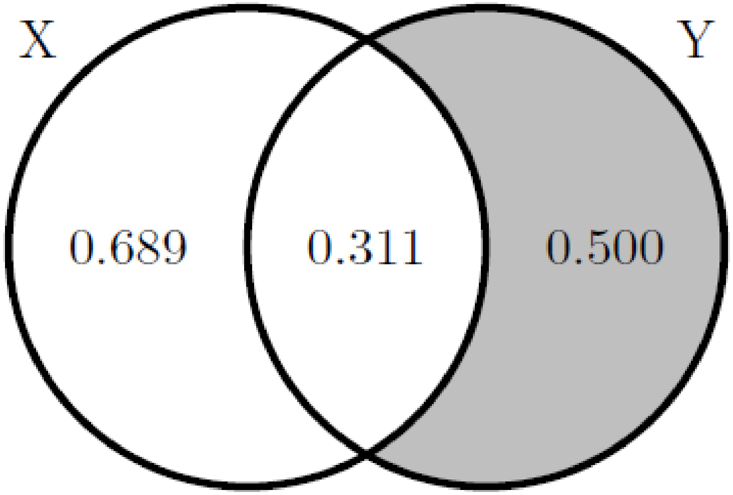
$H[X,Y] = 1.5$ bit
$H[X|Y] = 0.689$ bit

Quick Review of Information Entropy (cont'd)

A few examples:

Example 3: Generic (cont'd)

X	Y	p
0	0	½
1	0	¼
1	1	¼



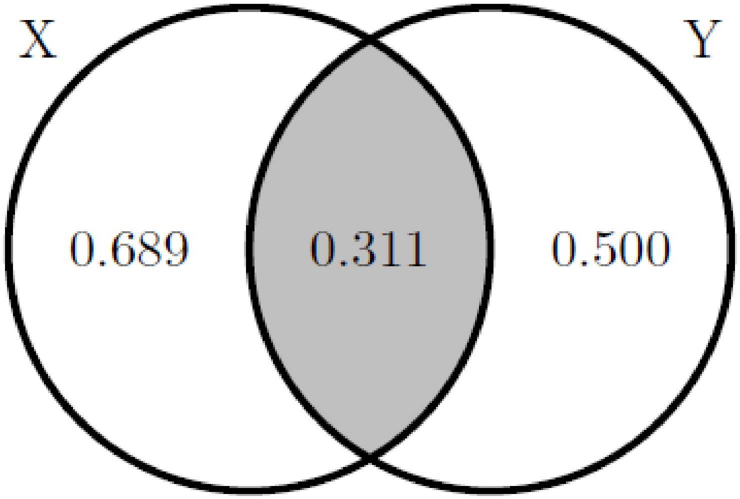
$H[X,Y]$ = 1.5 bit
$H[X|Y]$ = 0.689 bit
$H[Y|X]$ = 0.500 bit

Quick Review of Information Entropy (cont'd)

A few examples:

Example 3: Generic (cont'd)

X	Y	p
0	0	½
1	0	¼
1	1	¼



$H[X,Y]$ = 1.5 bit
$H[X|Y]$ = 0.689 bit
$H[Y|X]$ = 0.500 bit
$I[X{:}Y]$ = 0.311 bit

Entropy in Classical Information Theory

Goal: To construct a measure for the amount of information associated with a message

Historically, Hartley (1928) was the first to quantify the information content of a message source with two parameters: n , the number of characters in the equal-length sequences that compose its messages, and s, the number of equally probable symbols that each character may assume. The s^n distinct messages in Hartley's message source must themselves be equally probable since they are composed of equal numbers of equally probable symbols. Furthermore, the measure of information H of such a message source has the following two properties

- H must be proportional to n. Thus, $H = n\, f(s)$.
- H must be a monotonically increasing function, $H = g(s^n)$ of the number s^n of distinct, equally probable messages the source contains.

This means a message source containing messages with ***twice*** as many characters should contain ***twice*** the information and a message source containing ***more*** messages should contain ***more*** information.

Entropy in Classical Information Theory (cont’d)

The only differentiable, monotonically increasing functions $f(x)$ and $g(x)$ that satisfy the above two properties are

$$f(x) = g(x) = c \ln x \ ,$$

where c is an arbitrary positive constant.

Thus, Hartley defined the information content of a message source containing messages with n characters, each of which may be realized with s equally probable symbols as

$$H = c \ln s^n \ .$$

➡ *The information content of a message source is proportional to the logarithm of the number of distinct, equally probable messages it contains.*

Entropy in Classical Information Theory (cont'd)

Goal: To construct a measure for the amount of information associated with a message

Claude Shannon
(1916-2001)

The amount of information gained from the reception of a message depends on how *likely* it is.

- The *less likely* a message is, the *more* information gained upon its reception.

- Let $\boldsymbol{X} = \{x_1, \ldots, x_l\}$ = set of l messages.

Definition: A probability distribution $\boldsymbol{P} = (p_1, p_2, \ldots, p_l)$ on $\boldsymbol{X}$ is an assignment of a probability $p_j = p(x_j)$ to each message x_j.

- *Recall:* $p_j \geq 0$ and $\sum_{i=1}^{l} p_i = 1$.

Entropy in Classical Information Theory (cont'd)

1. $H(X)$ as Maximum Amount of Message Compression

- Let $X = \{x_1, \ldots, x_\ell\}$ be a set of letters from which we construct the messages.
- Suppose the messages have N letters a piece.
- The probability distribution $P = (p_1, \ldots, p_\ell)$ is now over the letter set.

> *What this means*:
>
> - Each letter x_i has a probability of p_i of occuring in a message.
> - *In other words*: A typical message will contain p_1N occurrences of x_1, p_2N occurrences of x_2, *etc.*

- *Thus*:

$$\left(\begin{array}{l}\textit{The number of distinct}\\ \textit{typical messages}\end{array}\right) = \frac{N!}{(p_1N)!(p_2N)!\cdots(p_\ell N)!}$$

Number of ways to arrange N distinct letters into l bins with capacities p_1N, p_2N, …, p_lN.

- *So*:

$$\log_2\left(\begin{array}{l}\textit{The number of distinct}\\ \textit{typical messages}\end{array}\right) = \log_2\left(\frac{N!}{(p_1N)!(p_2N)!\cdots(p_\ell N)!}\right)$$

Entropy in Classical Information Theory (cont'd)

$$\log_2\left(\frac{N!}{(p_1N)!(p_2N)!\cdots(p_\ell N)!}\right) = \log_2(N!) - \{\log_2((p_1N)!) + \ldots + \log_2((p_\ell N)!)\}$$

$$\approx (N\log_2 N - N) - \{(p_1N\log_2 p_1N - p_1N) + \ldots + (p_\ell N\log_2 p_\ell N - p_\ell N)\}$$

$$= N\{\log_2 N - 1 - p_1\log_2 p_1 - p_1\log_2 N + p_1 - \ldots - p_\ell\log_2 p_\ell - p_\ell\log_2 N + p_\ell\}$$

$$= -N\sum_{j=1}^{\ell} p_j\log_2 p_j$$

$$= NH(X)$$

- *Thus*:

$$\log_2\begin{pmatrix}\textit{The number of distinct}\\ \textit{typical messages}\end{pmatrix} = NH(X)$$

- *So*:

$$\begin{pmatrix}\textit{The number of distinct}\\ \textit{typical messages}\end{pmatrix} = 2^{NH(X)}$$

Entropy in Classical Information Theory (cont'd)

- *So*: There are only $2^{NH(X)}$ typical messages with N letters.
- This means, *at the message level*, we can encode them using only $NH(X)$ bits.

Check: 2 possible messages require 1 bit: 0, 1.
4 possible messages require 2 bits: 00, 01, 10, 11.
etc.

- *Now*: *At the letter level*, how many bits are needed to encode a message of N letters drawn from an ℓ-letter alphabet?

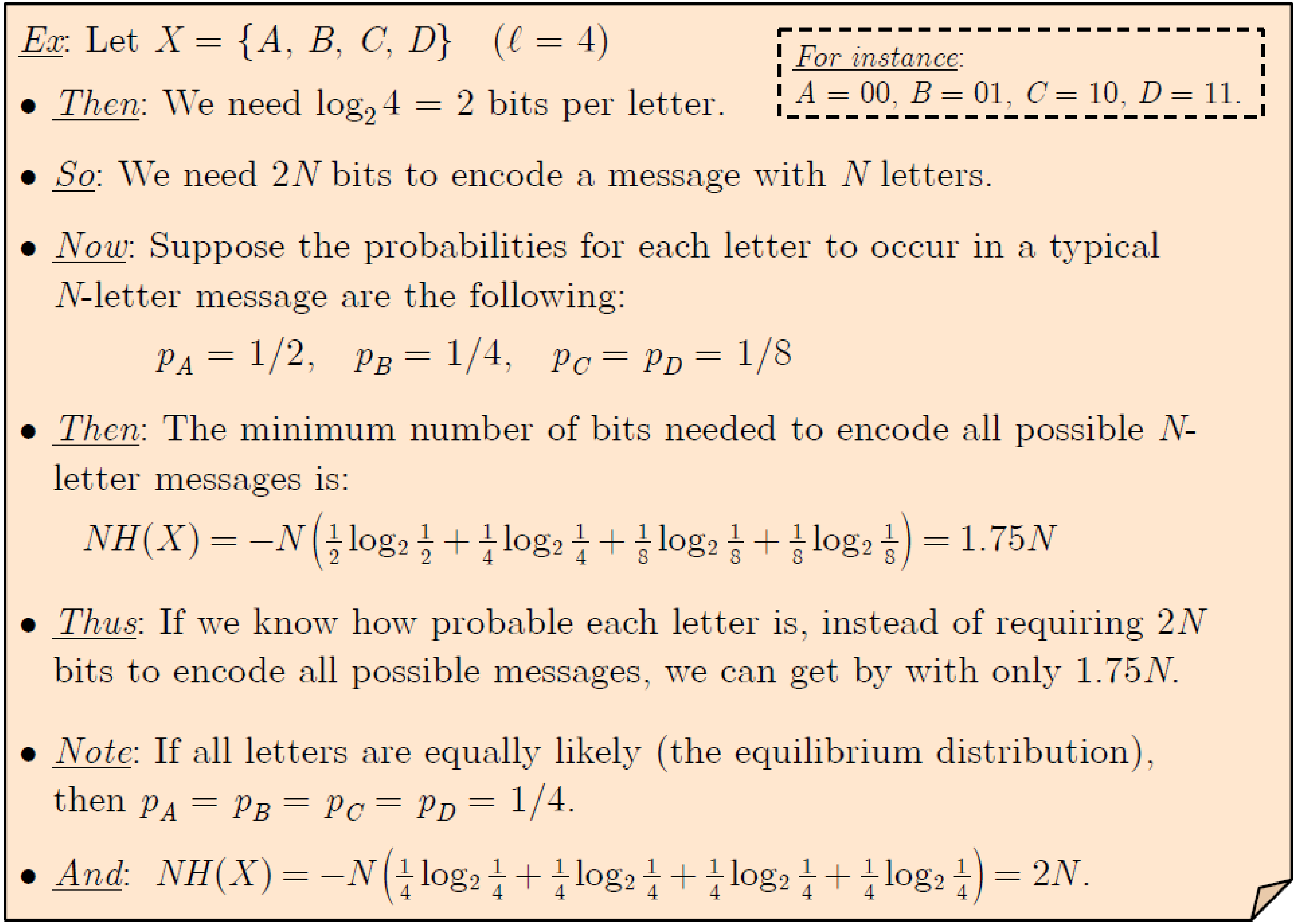
First: How many bits are needed to encode each letter in an ℓ-letter alphabet?

ℓ = #*letters*	x = #bits per letter	
2 letters	1 bit:	0, 1
4 letters	2 bits:	00, 01, 10, 11
8 letters	3 bits:	000, 001, 010, 011, 100, 101, 110, 111

So: $\ell = 2^x$, or $x = \log_2 \ell$

- *Note*: $\log_2 \ell$ bits per letter entails $N \log_2 \ell$ bits for a sequence of N letters.
- *Thus*: *If we know how probable each letter is*, instead of requiring $N \log_2 \ell$ bits to encode our messages, we can get by with only $NH(X)$ bits.
- *So*: $H(X)$ represents the *maximum amount that (typical) messages drawn from a given set of letters can be compressed.*

Entropy in Classical Information Theory (cont'd)



Ex: Let $X = \{A,\ B,\ C,\ D\}$ $\quad (\ell = 4)$

For instance:
$A = 00,\ B = 01,\ C = 10,\ D = 11.$

- *Then*: We need $\log_2 4 = 2$ bits per letter.
- *So*: We need $2N$ bits to encode a message with N letters.
- *Now*: Suppose the probabilities for each letter to occur in a typical N-letter message are the following:

  $$p_A = 1/2, \quad p_B = 1/4, \quad p_C = p_D = 1/8$$

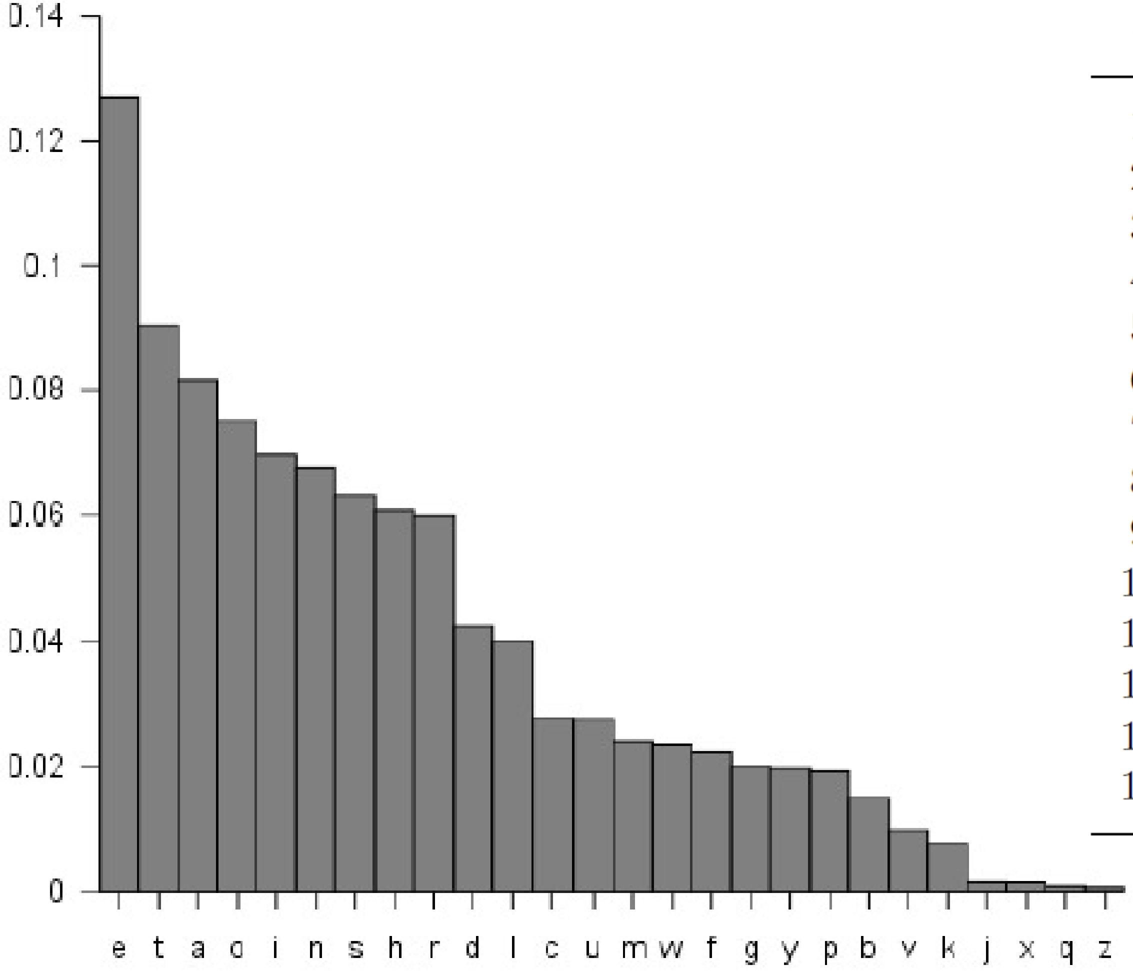
- *Then*: The minimum number of bits needed to encode all possible N-letter messages is:

  $$NH(X) = -N\left(\tfrac{1}{2}\log_2\tfrac{1}{2} + \tfrac{1}{4}\log_2\tfrac{1}{4} + \tfrac{1}{8}\log_2\tfrac{1}{8} + \tfrac{1}{8}\log_2\tfrac{1}{8}\right) = 1.75N$$

- *Thus*: If we know how probable each letter is, instead of requiring $2N$ bits to encode all possible messages, we can get by with only $1.75N$.
- *Note*: If all letters are equally likely (the equilibrium distribution), then $p_A = p_B = p_C = p_D = 1/4$.
- *And*: $NH(X) = -N\left(\tfrac{1}{4}\log_2\tfrac{1}{4} + \tfrac{1}{4}\log_2\tfrac{1}{4} + \tfrac{1}{4}\log_2\tfrac{1}{4} + \tfrac{1}{4}\log_2\tfrac{1}{4}\right) = 2N.$

Example: relative frequencies of letters in English

Suppose you are given 26 letters (27 if we include the space between words) from the English alphabet. The frequency of occurrence of these letters is as shown here.



i	Letter	p_i	$-\log_2 p_i$	i	Letter	p_i	$-\log_2 p_i$
1	a	0.0575	4.1	15	o	0.0689	3.9
2	b	0.0128	6.3	16	p	0.0192	5.7
3	c	0.0263	5.2	17	q	0.0008	10.3
4	d	0.0285	5.1	18	r	0.0508	4.3
5	e	0.0913	3.5	19	s	0.0567	4.1
6	f	0.0173	5.9	20	t	0.0706	3.8
7	g	0.0133	6.2	21	u	0.0334	4.9
8	h	0.0313	5.0	22	v	0.0069	7.2
9	i	0.0599	4.1	23	w	0.0119	6.4
10	j	0.0006	10.7	24	x	0.0073	7.1
11	k	0.0084	6.9	25	y	0.0164	5.9
12	l	0.0335	4.9	26	z	0.0007	10.4
13	m	0.0235	5.4	27	–	0.1928	2.4
14	n	0.0596	4.1				

Example: relative frequencies of letters in English

The entropy for this distribution is

$$H = \sum_{i=1}^{27} p_i \log_2 p_i = 4.138 \text{ bits.}$$

Thus, English has an entropy of about ~ 4.1 bits per letter, but this incorrectly ignores correlations between subsequent letters. Taking these into account results in a much lower value. English text can be compressed to a fraction of its length!

C.E. Shannon, "*Prediction and Entropy of Printed English*", Bell System Technical Journal, Vol.30, issue 1, p.50-64

Entropy in Classical Information Theory (cont'd)

2. H(X) as a Measure of Uncertainty

- Suppose $P = (p_1, \dots, p_l)$ is a probability distribution over a set of values $\{x_1, \dots, x_l\}$ of a random variable X.

  Def. 1. The *expected value* $E[X]$ of X is given by $E[X] = \sum_{j=1}^{l} p_j x_j$.

  Def. 2. The *information gained* if X is measured to have the value x_j is given by $-\log_2 p_j$.

  - *Motivation*: The greater p_j is, the more certain x_j is, and the less information should be associated with it.

- Then the expected value of $-\log_2 p_j$ is just the Shannon information:

$$E\left(-\log_2 p_j\right) = -\sum_{j=1}^{l} p_j \log_2 p_j = H[X]$$

- *What this means*:

  $H[X]$ tells us our expected *information gain* upon measuring X.

Codes in Communications and Information Processing

In communications and information processing, a ***code*** is a system of rules to convert information (e.g., letter, word, sound, image, etc.) into another form for communication through a communication channel or storage in a storage medium.

A ***source code*** *C* for a random variable *X* is a mapping from *X*, the range of *X*, to *D**, the set of finite-length strings of symbols from a *D*-ary alphabet. Let *C*(*x*) denote the codeword corresponding to *x* and let *l*(*x*) denote the length of *C*(*x*). For example, *C*(*red*) = 00, *C*(*blue*) = 11 is a source code for *X* = {*red*, *blue*} with alphabet *D* = {0, 1}.

The ***expected length*** *L*(*C*) of a source code *C*(*x*) for a random variable *X* with probability mass function *p*(*x*) is given by

$$L(C) = \sum_{x\,\in\,\chi} p(x)l(x)\ ,$$

where *l*(*x*) is the length of the codeword associated with *x*.
** Without loss of generality, one can assume that the *D*-ary alphabet is *D* = {0, 1, . . . , *D* − 1}.

Codes in Communications and Information Processing (cont’d)

Example 1: (Cover and Thomas, 2006, p.104)

Let X be a random variable with the following distribution and codeword assignment

$Pr(X = 1) = ½$, codeword $C(1) = 0$
$Pr(X = 2) = ¼$, codeword $C(2) = 10$
$Pr(X = 3) = 1/8$, codeword $C(3) = 110$
$Pr(X = 4) = 1/8$, codeword $C(4) = 111$

The entropy $H[X]$ of X is 1.75 bits, and the expected length of $L(C) = E[l(\mathrm{X})]$ is also 1.75 bits. Any sequence of bits can be uniquely decoded into a sequence of symbols of X. For example, the bit string 0110111100110 is decoded as 134213.

Codes in Communications and Information Processing (cont'd)

Example 2: (Cover and Thomas, 2006, p.104)

Consider another code with the following distribution and codeword assignment

$$Pr(X = 1) = 1/3 \ , \quad \text{codeword } C(1) = 0$$
$$Pr(X = 2) = 1/3 \ , \quad \text{codeword } C(2) = 10$$
$$Pr(X = 3) = 1/3 \ , \quad \text{codeword } C(3) = 11$$

The entropy $H[X]$ of X is log 3 = 1.58 bits and the average length of the encoding is 1.67 bits. Here, $E[l(\mathrm{X})] > H[X]$, but any sequence of bits can still be uniquely decoded into a sequence of symbols of X.

Codes in Communications and Information Processing (cont'd)

Example 3: (Morse Code)

The Morse code is a reasonably efficient code for the English alphabet.

Original Morse code: X contains dots, dashes, and pauses.

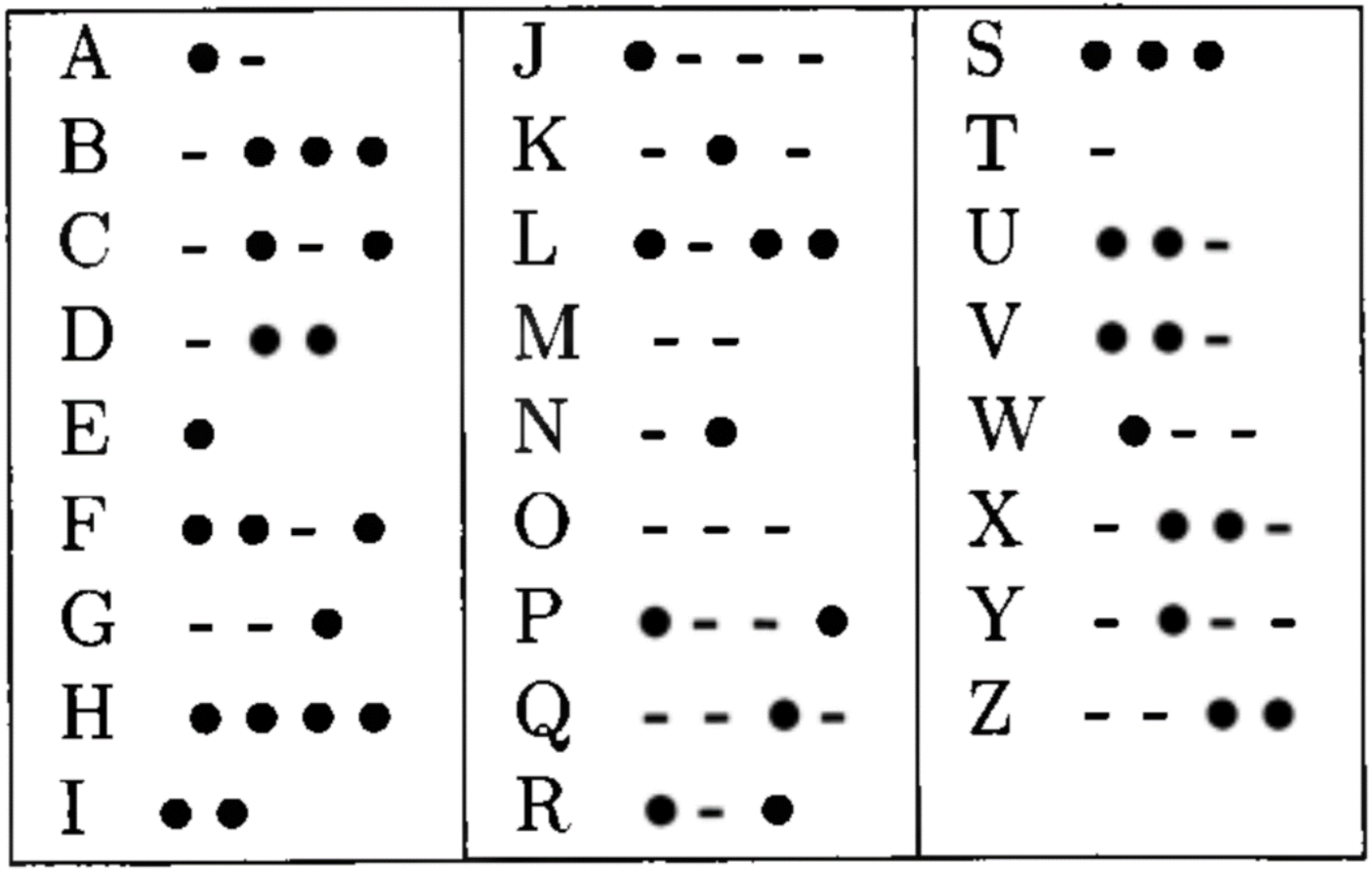
Modern version by A. Vail: X contains Roman letters, numbers and punctuation.

Codewords: strings from dot, dash, short gap, medium gap, long gap.

Design strategy: frequently used letters are assigned short code words for compression. For example, "e" is a single dot, "t" is a single dash. Two-symbol codewords are given to "a", "i", "m" and "n". They got their frequencies from printer's type box.

Codes in Communications and Information Processing (cont’d)

Example 3: (Morse Code) (*cont’d*)



A	● -	J	● - - -	S	● ● ●
B	- ● ● ●	K	- ● -	T	-
C	- ● - ●	L	● - ● ●	U	● ● -
D	- ● ●	M	- -	V	● ● -
E	●	N	- ●	W	● - -
F	● ● - ●	O	- - -	X	- ● ● -
G	- - ●	P	● - - ●	Y	- ● - -
H	● ● ● ●	Q	- - ● -	Z	- - ● ●
I	● ●	R	● - ●		

Morse code. Common letters (e.g. E) have the shortest codewords, whereas rare letters (e.g. Z) have the longest codewords.

Codes in Communications and Information Processing (cont’d)

More Definitions:

A ***code*** is said to be ***nonsingular*** if every element of the range of *X* maps into a different string in *D**. i.e.,

$$x \neq x' \Longrightarrow C(x) \neq C(x').$$

The ***extension*** *C** of a code *C* is the mapping from finite-length strings of χ to finite-length strings of *D*, defined by

$$C(x_1 x_2 \ldots x_n) = C(x_1)C(x_2)\ldots C(x_n),$$

where $C(x_1)C(x_2)\ldots C(x_n)$ indicates concatenation of the corresponding codewords. For example, if $C(x_1)$ = 00 and $C(x_2)$ = 11, then $C(x_1 x_2)$ = 0011.

A ***code*** is called ***uniquely decodable*** if its extension is ***nonsingular***. In other words, any encoded string in a uniquely decodable code has only one possible source string producing it.

Codes in Communications and Information Processing (cont'd)

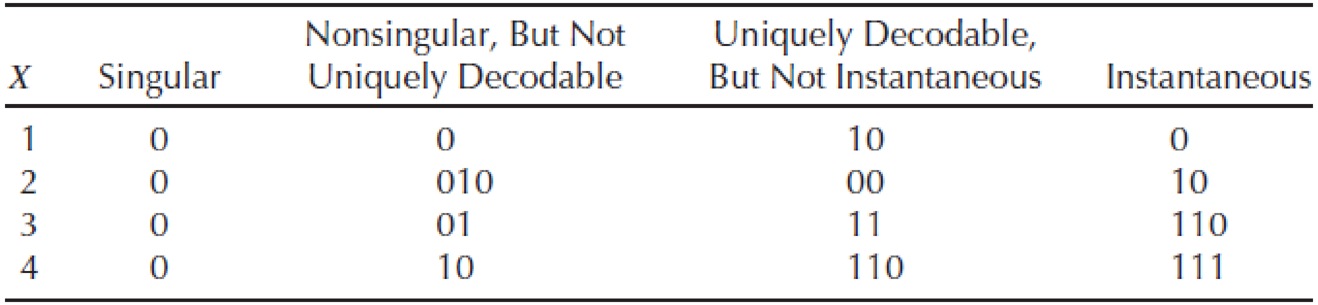
More Definitions: (cont'd)

A ***code*** is called a ***prefix code*** or an ***instantaneous code*** if no codeword is a *prefix* of any other codeword.

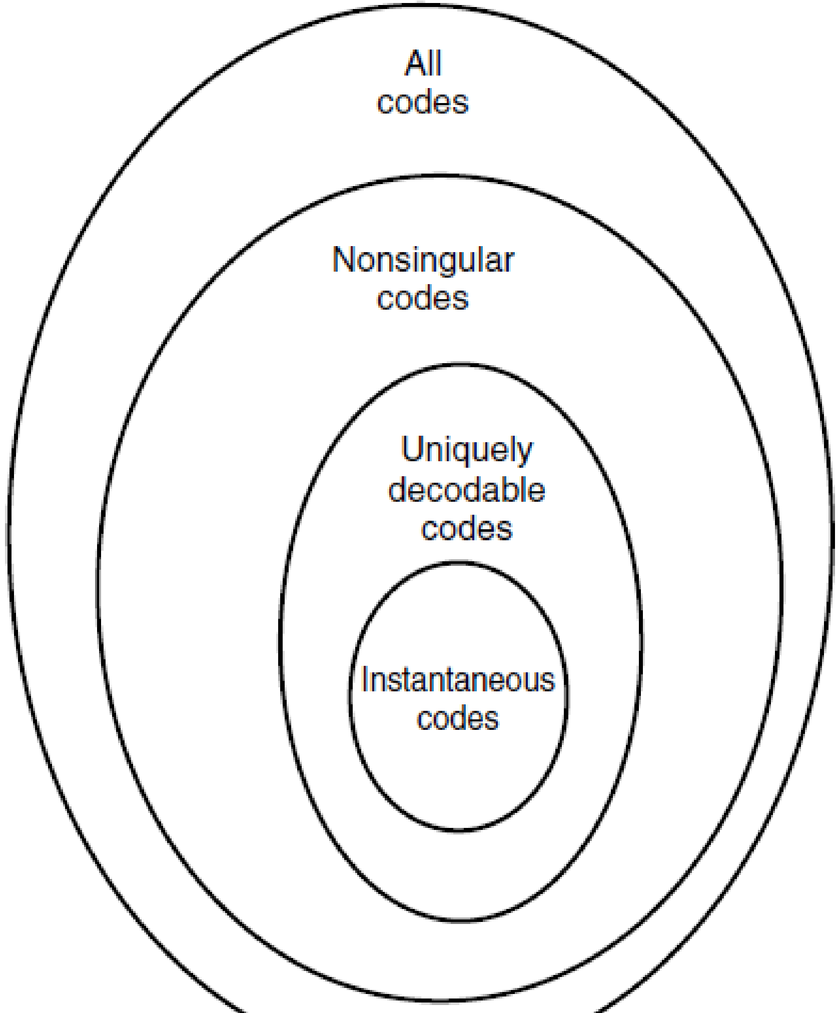
An *instantaneous code* can be decoded without reference to future codewords since the end of a codeword is immediately recognizable. Hence, x_i can be decoded as soon as we come to the end of the codeword corresponding to it. An *instantaneous code* is therefore a *self-punctuating code*, where we can add commas to separate the codewords without looking at later symbols. Refer to *Example 1* above, the bit string 0110111100110 can be parsed as 0,110,111,10,0,110.

Codes in Communications and Information Processing (cont'd)

Classes of codes



X	Singular	Nonsingular, But Not Uniquely Decodable	Uniquely Decodable, But Not Instantaneous	Instantaneous
1	0	0	10	0
2	0	010	00	10
3	0	01	11	110
4	0	10	110	111



Classes of codes

Codes in Communications and Information Processing (cont'd)

We want to construct prefix codes of minimum expected length to describe a given source. It turns out that any set of codeword lengths possible for prefix codes has to satisfy the *Kraft inequality* and that the *Kraft inequality* is a sufficient condition for the existence of a codeword set with the specified set of codeword lengths..

Theorem: Kraft Inequality

For any instantaneous code (prefix code) over an alphabet of size D, the codeword lengths l_1, l_2, . . . , l_m must satisfy the inequality

$$\sum_i D^{-l_i} \leq 1\,.$$

Conversely, given a set of codeword lengths that satisfy this inequality, there exists an *instantaneous code* with these word lengths.

Refer to any textbook on Coding Theory for its **Proof**, e.g., Cover and Thomas, 2006.

Codes in Communications and Information Processing (cont'd)

Finding the prefix code with the minimum expected length (optimal code)

Theorem:

The expected length L of any instantaneous D-ary code for a random variable X is greater than or equal to the entropy $H_D[X]$ i.e.,

$$L \geq H_D[X],$$

with equality iff $D^{-l_i} = p_i$.

Theorem: Bounds on the Optimal Code Length

Let $l_1^*, l_2^*, \ldots, l_m^*$ be optimal codeword lengths for a source distribution $\boldsymbol{p}$ and a D-ary alphabet, and let L^* be the associated expected length of an optimal code (i.e., $L^* = \sum p_i l_i^*$). Then

$$H_D[X] \leq L^* < H_D[X] + 1 .$$

Codes in Communications and Information Processing (cont'd)

Shannon-Fano Codes

Shannon–Fano coding is a name given to two different but related techniques for constructing a prefix code based on a set of symbols and their probabilities (whether estimated or measured).

Shannon–Fano codes are *suboptimal* in the sense that they do not always achieve the lowest possible expected codeword length, as *Huffman codes*. However, they have an expected codeword length within 1 bit of optimal.

Fano's method usually produces encoding with shorter expected lengths than *Shannon's method*, but *Shannon's method* is easier to analyze theoretically.

Codes in Communications and Information Processing (cont'd)

Shannon-Fano Codes

Shannon's Method

Suppose $p(x_1) \geq p(x_2) \geq \cdots \geq p(x_n)$, then Shannon's method chooses a prefix code where a source symbol *i* is given the codeword length

$$L_i = \lceil -\log_2 p(x_i) \rceil.$$

The notation $\lceil \ldots \rceil$ means rounding to the smallest integer that is larger than the value of within the brackets.

Codes in Communications and Information Processing (cont'd)

Shannon-Fano Codes

An Example: (Shannon Code)

Consider a random variable X taking values in the set $\chi = \{a, b, c\}$ with probabilities {0.55, 0.25, 0.2}. The entropy $H[X]$ is 1.439 bits. The length of the codewords are:

$$L(a) = \lceil -\log_2 0.55 \rceil = 1;\ L(b) = \lceil -\log_2 0.25 \rceil = 2;\ L(c) = \lceil -\log_2 0.2 \rceil = 3$$

The codewords are: $C(a) = 0$; $C(b) = 10$; $C(c) = 110$.

The average length is: $0.55 \times 1 + 0.25 \times 2 + 0.2 \times 3 = 1.64$ bits

Codes in Communications and Information Processing (cont'd)

Shannon-Fano Codes

Fano's Method

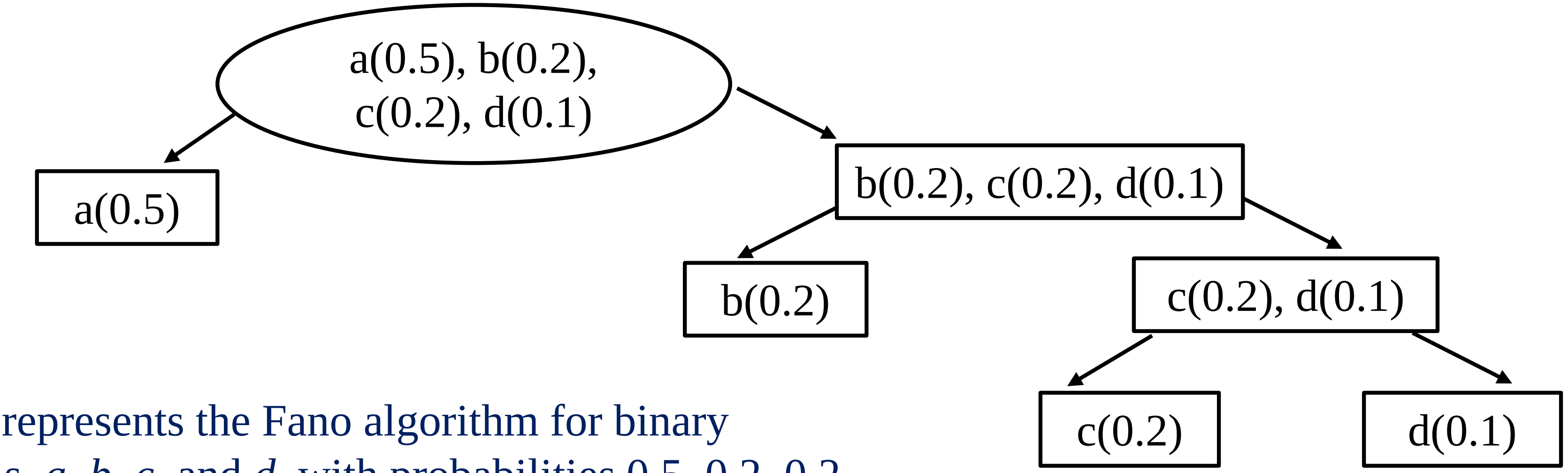
– divides the source symbols into two sets ("0" and "1") with probabilities as close to 1/2 as possible. Those sets are then themselves divided in two, and so on, until each set contains ***only*** one symbol. The codeword for that symbol is the string of "0"s and "1"s that records which half of the divides it fell on.

Codes in Communications and Information Processing (cont'd)

Shannon-Fano Codes

An Example: (Fano Code)

Consider a random variable X taking values in the set $\chi = \{a, b, c, d\}$ with probabilities {0.5, 0.2, 0.2. 0.15}. Put b, c, d into one group. Assign "0" to be the first binary digit of a, and "1" to be the first binary digit of b, c, d. Continue the division on b, c, d by putting b in one group, and c, d in another group. The codewords are: $C(a) = 0$; $C(b) = 10$; $C(c) = 110$, $C(d) = 111$.



A tree graph that represents the Fano algorithm for binary coding of symbols, *a*, *b*, *c*, and *d*, with probabilities 0.5, 0.2, 0.2, and 0.1, respectively.

Codes in Communications and Information Processing (cont’d)

Huffman Codes

While Fano’s method is to create a tree from the root to the leaves, the ***Huffman algorithm*** works in the opposite direction, merging from the leaves to the root, a bottom-up and greedy method.

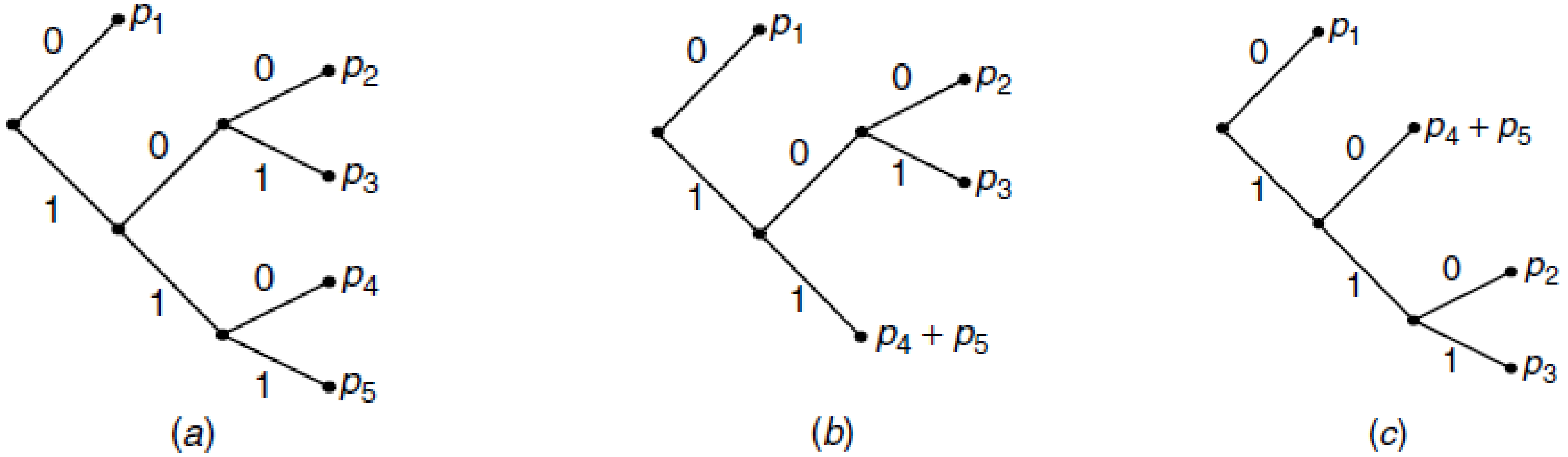
1. Create a leaf node for each symbol and add it to a priority queue, using its frequency of occurrence as the priority

2. While there is more than one node in the queue:
   1) Remove the two nodes of lowest probability or frequency from the queue
   2) Prepend 0 and 1 respectively to any code already assigned to these nodes
   3) Create a new internal node with these two nodes as children and with probability equal to the sum of the two nodes' probabilities
   4) Add the new node to the queue

3. The remaining node is the root node and the tree is complete

Codes in Communications and Information Processing (cont'd)

Huffman Code



Induction step for Huffman coding. Let $p_1 \geq p_2 \geq \cdots \geq p_5$. A canonical optimal code is illustrated in (a). Combining the two lowest probabilities, one gets the code in (b). Rearranging the probabilities in descending order, one gets the canonical code in (c) for m – 1 symbols.

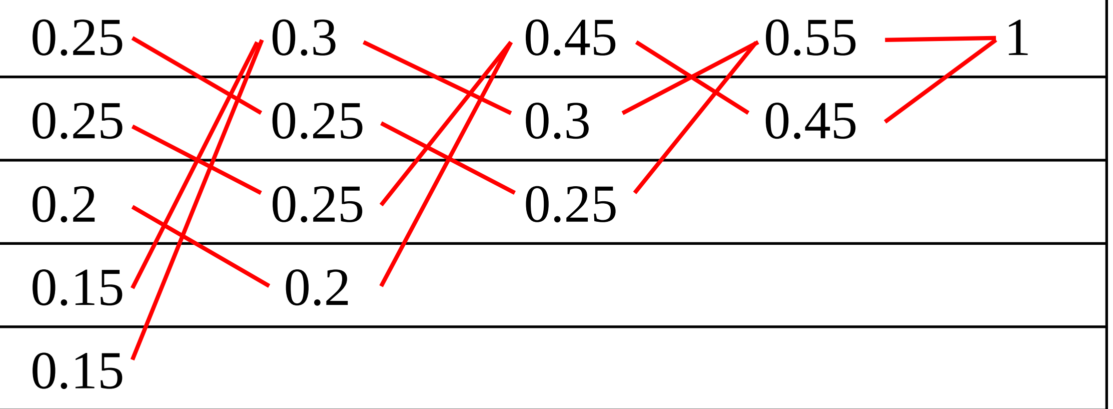
Codes in Communications and Information Processing (cont'd)

Huffman Code

An example:

Consider a random variable X taking values in the set $\chi = \{1, 2, 3, 4, 5\}$ with probabilities $\{0.25, 0.25, 0.2, 0.15, 0.15\}$. One expects the optimal binary code for X to have the longest codewords assigned to the symbols 4 and 5. One can construct a code in which the two longest codewords differ only in the last bit, and combine them into a single source symbol with a single source symbol, with probability 0.30. Continue with this procedure, one has

Codeword Length	**Codeword**	***X***	**Probability**				
2	01	1	0.25	0.3	0.45	0.55	1
2	10	2	0.25	0.25	0.3	0.45	
2	11	3	0.2	0.25	0.25		
3	000	4	0.15	0.2			
3	001	5	0.15				



*****This code has an average length 2.3 bits*****

Codes in Communications and Information Processing (cont'd)

Example: (Comparing Shannon, Fano and Huffman codes)

Consider a random variable X taking values in the set $\chi = \{A, B, C, D, E\}$ with probabilities $\{0.375, 0.2, 0.15, 0.15, 0.125\}$. The entropy $H[X]$ is 2.19 bits. We now use the three different coding methods to obtain the codewords for χ.

Shannon code:

The codeword lengths for A, B, C, D, E are $\lceil -\log_2 0.375 \rceil$, $\lceil -\log_2 0.2 \rceil$, $\lceil -\log_2 0.15 \rceil$, $\lceil -\log_2 0.15 \rceil$, $\lceil -\log_2 0.125 \rceil$, corresponding to 2, 3, 3, 3, 3 respectively. The codeword for χ are

$$C(A) = 00\ ;\ C(B) = 010\ ;\ C(C) = 011\ ;\ C(D) = 100;\ C(E) = 101$$

The average codeword length is: $0.375 \times 2 + (0.2 + 0.15 + 0.15 + 0.125) \times 3 = 2.625$ bits

Codes in Communications and Information Processing (cont'd)

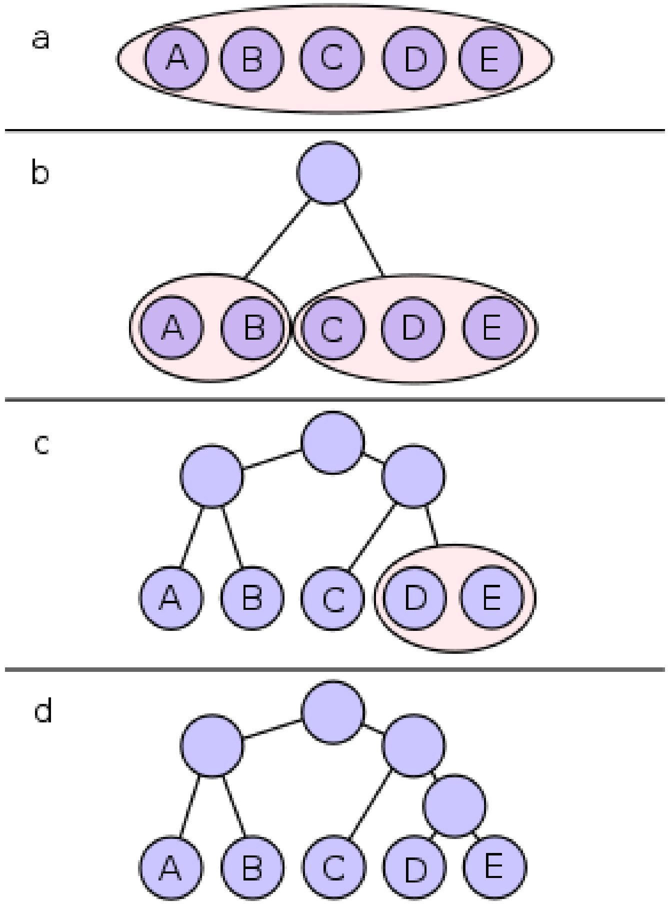
Example: (cont'd)

Fano code:

The final tree after the division procedures is shown on the right.

The codewords of χ are shown in the table below.

Symbol	**A**	**B**	**C**	**D**	**E**
Probabilities	0.375	0.2	0.15	0.15	0.125
First division	0		1		
Second division	0	1	0	1	
Third division				0	1
Codewords	00	01	10	110	111



The average codeword length is: (0.375 + 0.2 + 0.15) × 2 + (0.15 + 0.125) × 3 = 2.275 bits

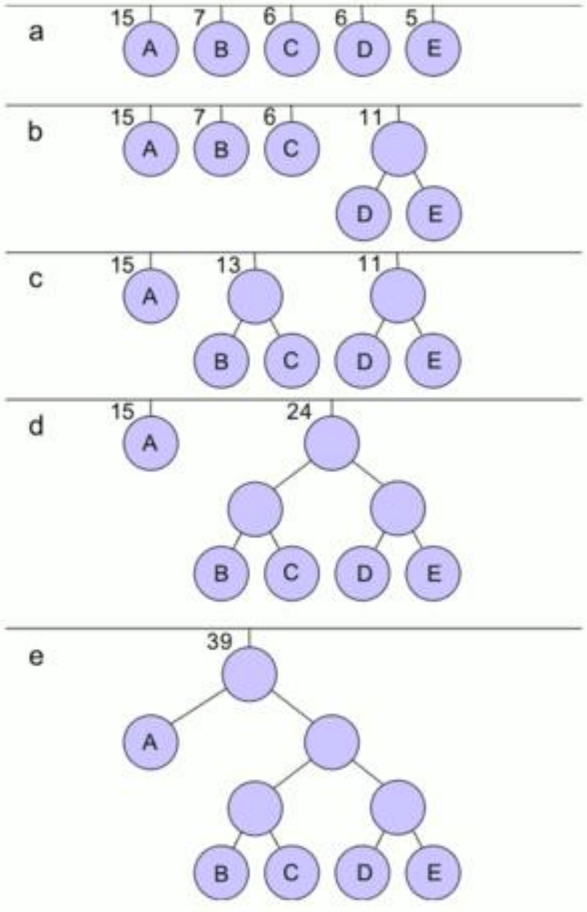
Codes in Communications and Information Processing (cont'd)

Example: (cont'd)

Huffman code:

The final tree after employing the Huffman algorithm is shown on the right.

The codewords of χ are shown in the table below.



Symbol	**A**	**B**	**C**	**D**	**E**
Codewords	0	100	101	110	111

The average codeword length is: $0.375 \times 1 + (0.2 + 0.15 + 0.15 + 0.125) \times 3 = 2.25$ bits

Codes in Communications and Information Processing (cont’d)

Some Remarks on Shannon Code, Fano Code and Huffman Code

Huffman coding is ***optimal***, i.e., if C^* is a Huffman code and C' is any other uniquely decodable code, $L(C^*) \leq L(C')$.

Fano coding is ***suboptimal***. Though not optimal in general, it achieves $L(C) \leq H[X] + 2$.

The Shannon code and the Huffman code differ by less than 1 bit in expected codelength (since both lie between $H[X]$ and $H[X] + 1$. Also, either the Shannon code or the Huffman code can be ***shorter*** for ***individual*** symbols, but the Huffman code is shorter on average.

Codes in Communications and Information Processing (cont'd)

Some Remarks on Shannon Code, Fano Code and Huffman Code

Shannon code reaches ***entropy limit*** for i.i.d. data.

Theorem: Asymptotic Equipartition Property (AEP)

If X_1, X_2, . . . are i.i.d. ~ $p(x)$, then

$$-\frac{1}{n}\log p(X_1, X_2, \dots, X_n) \;\to\; H[X] \quad \text{in probability.}$$

Proof:

Since X_1, X_2, . . . are i.i.d., so are log $p(X_i)$. By the weak law of large numbers,

$$-\frac{1}{n}\log p(X_1, X_2, \dots, X_n) = -\frac{1}{n}\sum_i \log p(X_i) \;\to\; -E[\log p(X)] = H[X].$$

****AEP*** implies that we can represent sequences X^n using $nH[X]$ bits on the average.

Communication Channels

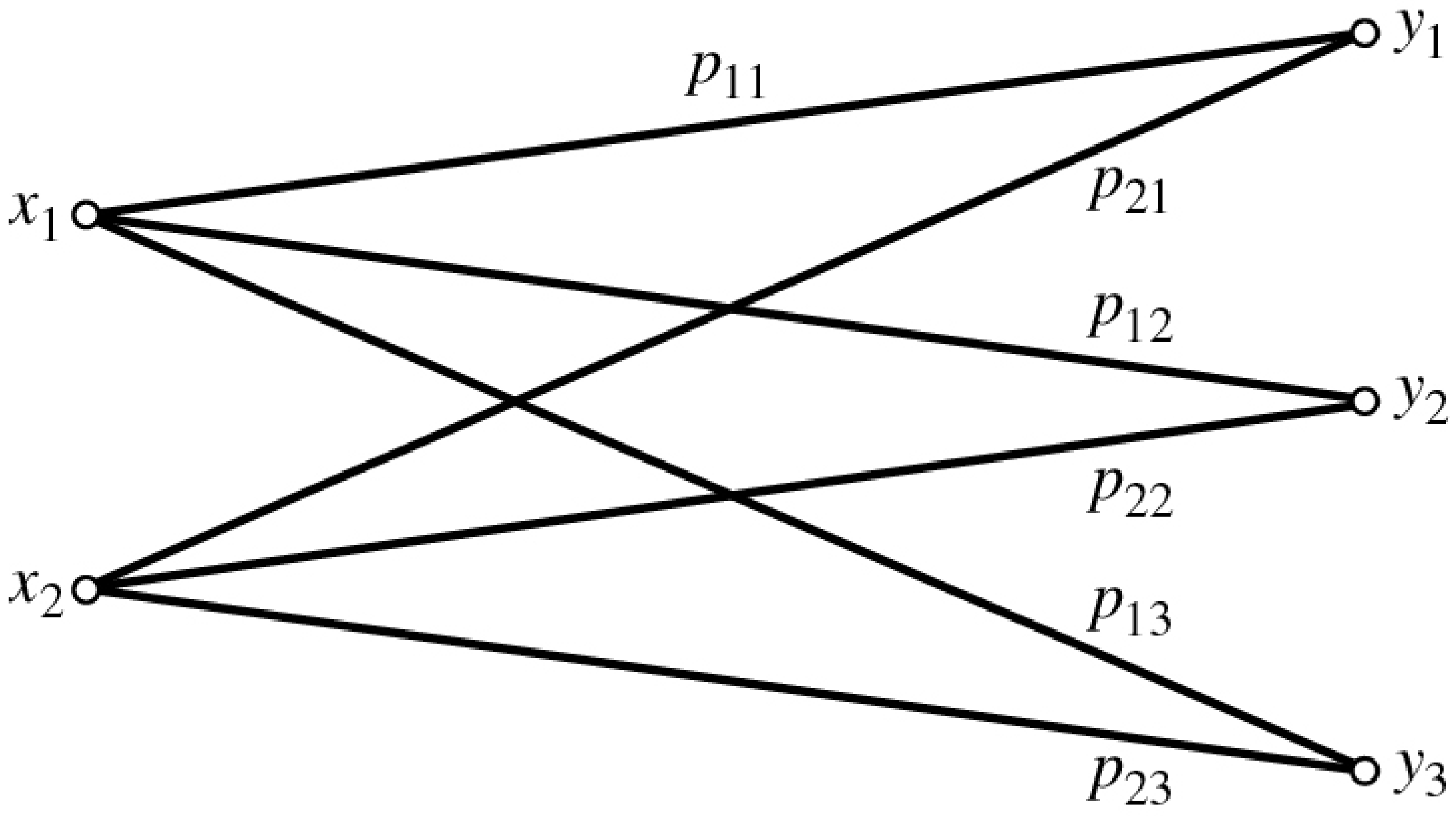
The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point.

(Claude Shannon, 1948)

Shannon's communication theory concerns with point-to-point communications as in telephony and gives limits on coding.

- Source coding: limits on data storage (transmission over noiseless channels)
- Channel coding: limits on data transmission over noisy channels.

Simple Communication Channel



Source Entropy

Entropy of a source is defined as the average information associated with a set of source outputs (i = 1, ..., n) occurring with probabilities p_i.

Source Entropy = average uncertainty of a source

$$S = \left\langle \log\left(\frac{1}{p_i}\right) \right\rangle = \sum_i p_i \log\left(\frac{1}{p_i}\right)$$

Note that entropy function has a maximum: When all source states are equally likely.

Channel: Transition Probability

A channel with n inputs X and m outputs Y is represented by the $n\ m$ transition probabilities

$$p(y_j|x_i), \qquad i = 1, \dots, n\,; j = 1, \dots, m$$

These are the conditional probability of output Y given Input X

There are several kinds of entropy defined for the channel.

$H[X]$ = average uncertainty of the channel input

$H[Y]$ = average uncertainty of the channel output

$H[X|Y]$ = average uncertainty of the channel input given output Y

$H[Y|X]$ = average uncertainty of the channel output given input X

Entropy for a Channel: Mutual Information
Channel Capacity Joint Entropy

Mutual information *between the input and output of a channel is*

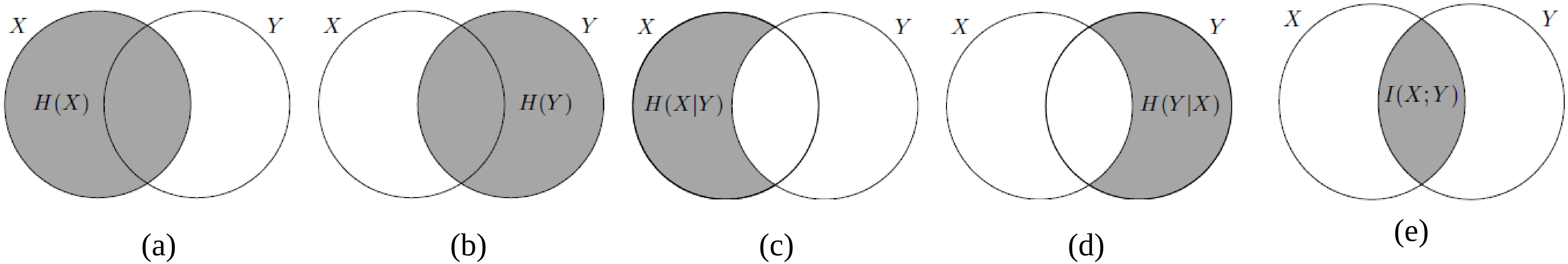
$$I\ [X\colon Y] = H[X] - H[X|Y] = H[Y] - H[Y|X] = I[Y\colon X]$$

The maximum value of mutual information, maximized with respect to the source probabilities, is known as the ***Channel capacity***

Joint Entropy = $H[X,Y]$ = *average uncertainty of the entire communication system*

(defined with joint probability of input and *output)*

Entropy for a Channel: Mutual Information
Channel Capacity Joint Entropy

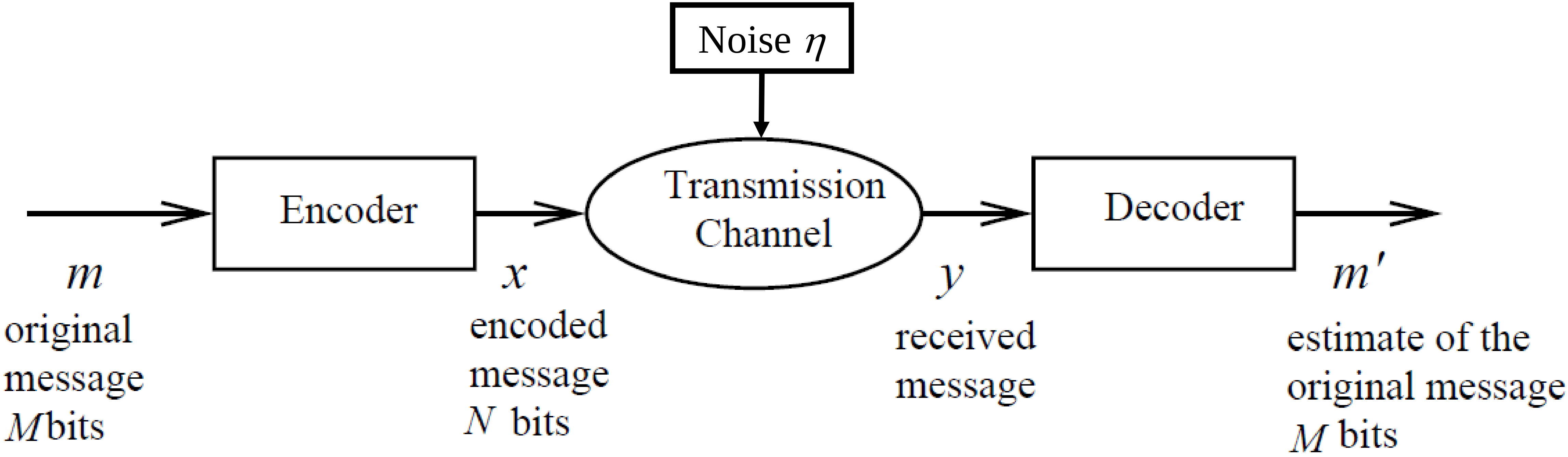


Venn diagrams for (a) $H[X]$, (b) $H[Y]$, (c) $H[X|Y]$, (d) $H[Y|X]$, (e) $I[X{:}Y]$

Mutual Information I[X:Y] between input and output of a channel is given by

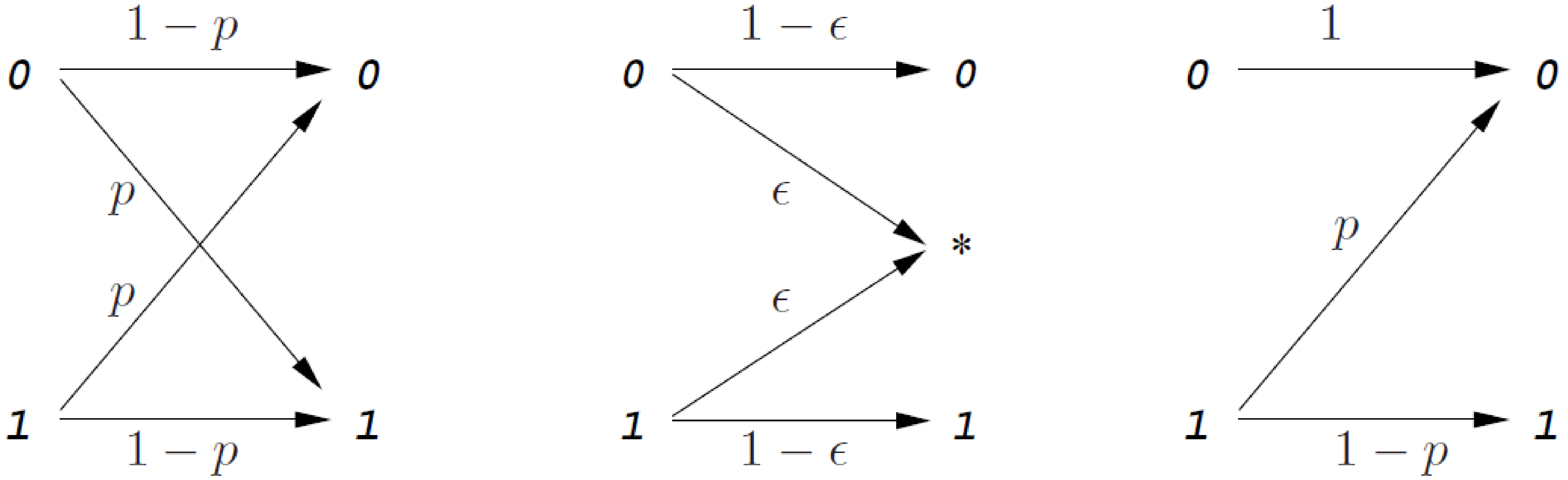
$$I[X{:}Y] = \sum_{x,y} P(x,y)\log\frac{P(x,y)}{P(x)P(y)} = H[X] - H[X|Y] = H[Y] - H[Y|X] = I[Y{:}X]$$

Communication Channels



Typical flowchart of a communication device. A message **s** is encoded as codewords **x** before being transmitted through a channel, which may corrupt the encoded message by adding noise η to produce output $\mathbf{y} = \mathbf{x} + \eta$. A receiver decodes the output **y** to recover inputs **x**, which are then interpreted as a message **s**.

Communication Channels



Three communication channels. Left : the binary symmetric channel. An error in the transmission, in which the output bit is the opposite of the input one, occurs with probability p. Middle: the binary erasure channel. An error in the transmission, signaled by the output $*$, occurs with probability ε. Right : the Z channel. An error occurs with probability p whenever a 1 is transmitted.

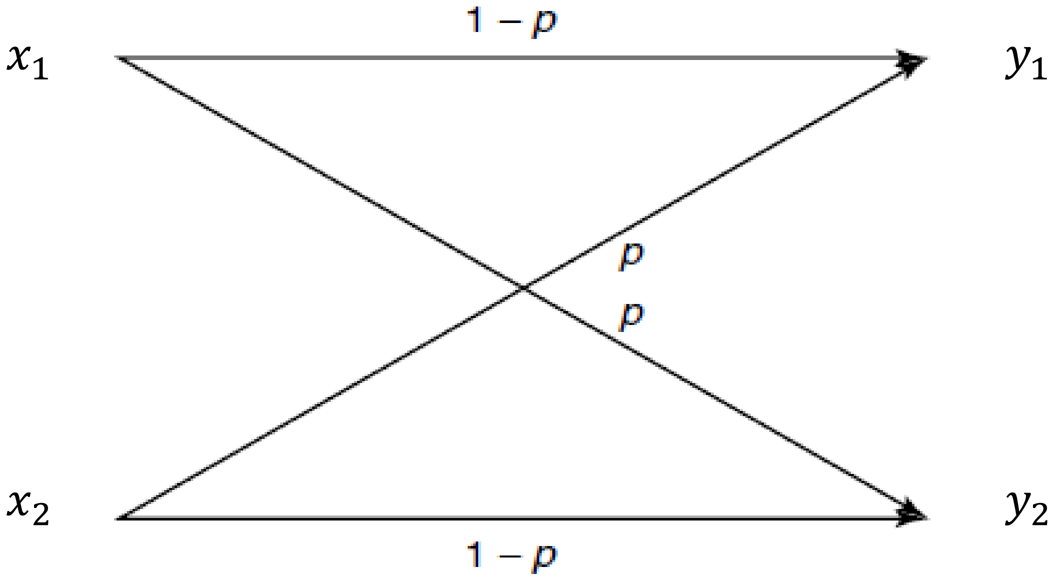
Channel Capacity and Mutual Information

The “Information” Channel Capacity of a discrete memoryless channel as

$$C = \max_{p(x)} I[X\colon Y]$$

where the maximum is taken over all possible input distributions $p(x)$. For a binary symmetric channel,

$$I[X\colon Y] \leq 1 - H[p]$$



C. E. Shannon. “*A mathematical theory of communication*”. Bell Syst. Tech. J., 27:379–423,623–656, 1948.

Calculate Capacity for Binary Channel

For binary source with

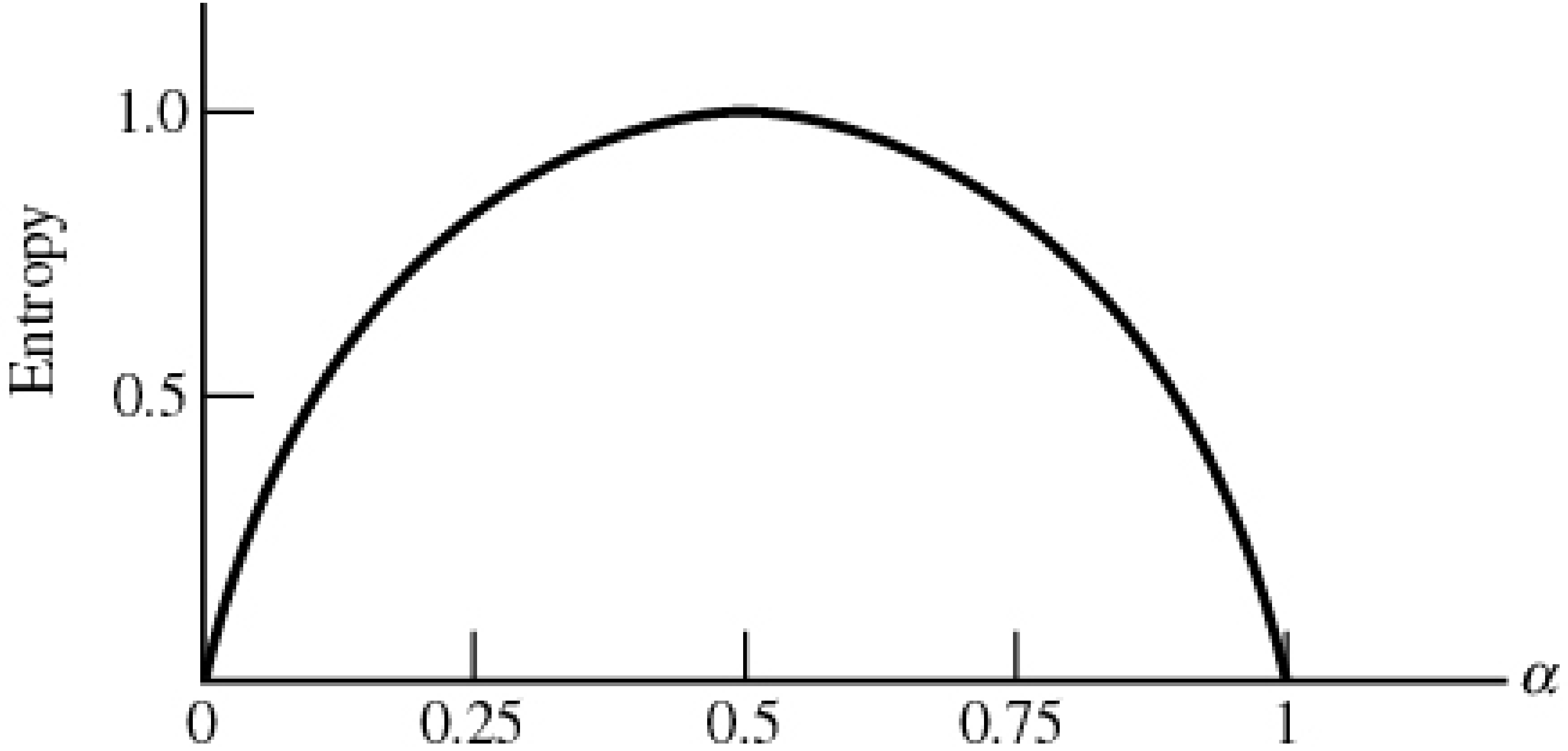
$$p(1) = \alpha \, , \, p(0) = 1 - \alpha$$

the entropy of the source is

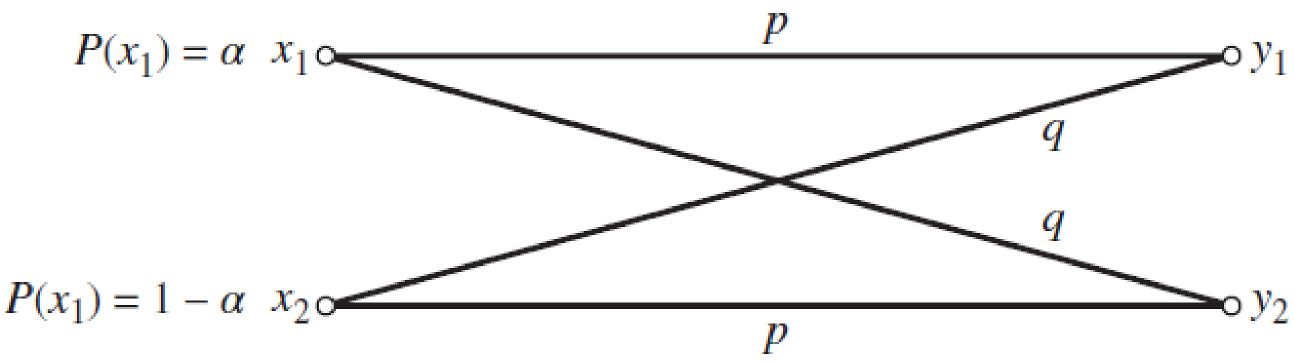
$$H[\alpha] = -\alpha \log_2 \alpha - (1 - \alpha) \log_2 (1 - \alpha)$$

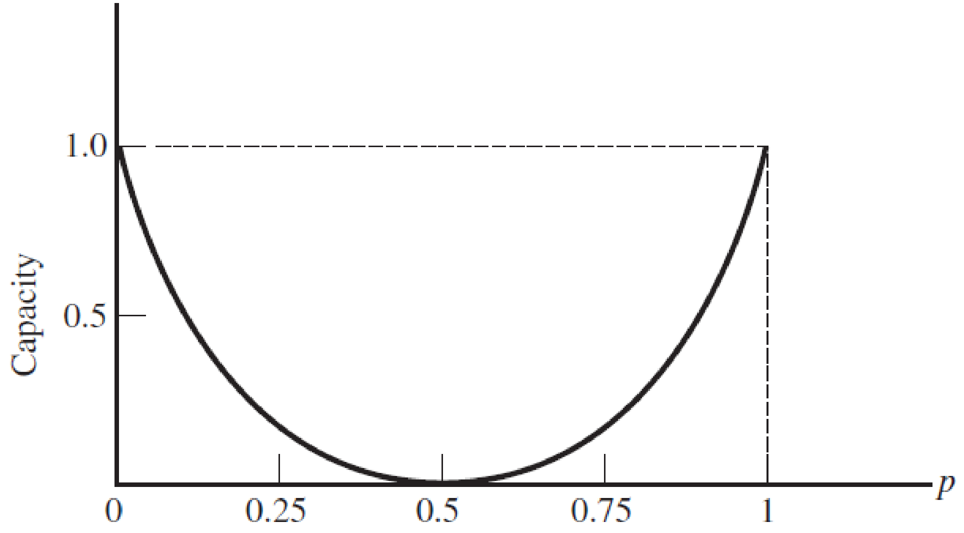
For $p(1) = p(0)$, each symbol is equally likely and our uncertainty is maximum. If the probabilities are different, we are less uncertain as to which symbol appears on the source output.

Entropy of a Binary Source



Capacity of a Binary Symmetric Channel





We determine the capacity by maximizing $I[X{:}Y] = H[Y] - H[Y|X]$, and $C = \max_{p(x)} I[X{:}Y]$

$$H[Y|X] = -\alpha p\log_2 p - (1-\alpha)p\log_2 p - \alpha q\log_2 q - (1-\alpha)q\log_2 q = -p\log_2 p - q\log_2 q$$

where $q = 1 - p$. Since $H[Y]$ is a maximum when each output has a probability of 0.5, in which case $H[Y] = 1$, therefore, $C = 1 + p \log_2 p + q \log_2 q$

Source Information Rate

Information from a source that produces different symbols according to some probability scheme can be described by the entropy $H[X]$ (in bits /symbol) of the source.

Source information rate = R_S is related to the Symbol Rate (symbol/sec) (= r) as

$$R_S = rH[X] \; \textit{in bits}/\mathrm{sec}$$

Shannon's Theorem

C=Channel Capacity (bit/symbol)

S=available Symbol rate from Channel (symbol/bit)

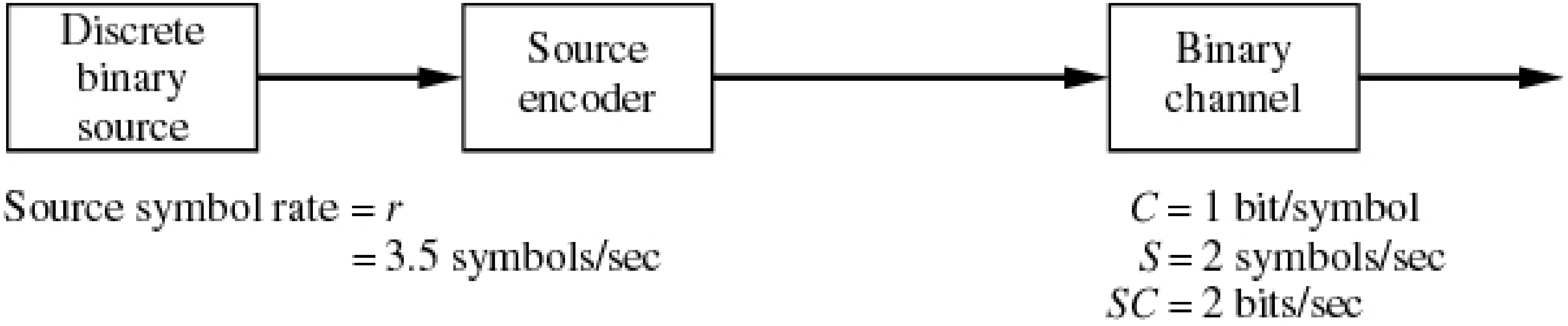
Shannon's noiseless coding theorem states that :

Given a channel and a source that generates information at a rate less than the channel capacity, it is possible to encode the source output so that the source output can be transmitted through the channel.

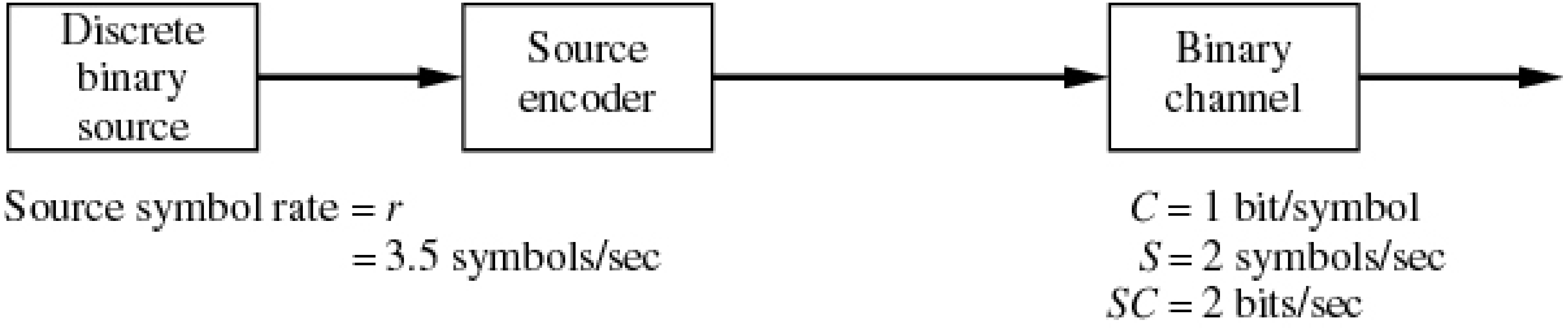
→ Shannon's source coding theorem is essentially about encoding messages into codewords efficiently, which is a form of ***data compression***.

Source Encoding and Decoding

Source encoding *is used to remove redundancy from a source output so that the information per transmitted symbol can be maximized.*



Source Encoding: An example



In the above figure, consider the discrete binary source has two possible outputs A and B with probabilities 0.9 and 0.1 respectively. Here, the source *symbol* rate is greater than the channel capacity, so the source symbols cannot be placed directly into the channel. However, the source entropy is

$$H[X] = -\,0.1\ \log_2 0.1 - 0.9\ \log_2 0.9 = 0.469 \text{ bits/symbol}$$

which corresponds to a source information rate of $R_S = rH[X] = 3.5(0.469) = 1.642$ bps.
If the probability of transmission in the binary channel $p = 1$, the channel capacity is then 1 bit per symbol, which, in this case, is an information rate of 2 bits per second. Therefore, the information rate is less than the channel capacity, so transmission is possible.

How is Shannon Information/Entropy related to other notions of entropy?

Thermodynamic entropy:

$$\Delta S = S_f - S_i =_R \int_i^f \frac{\delta Q_R}{T}$$

Boltzmann entropy:

$$S_B\left[\Gamma_{D_i}\right] = -Nk \sum_{j=1}^{l} p_j \ln\, p_j + constant$$

Gibbs entropy:

$$S_G[\rho] = -k \int_{\Omega} \rho(x,t) \ln\big(\rho(x,t)\big)\, dx$$

Shannon entropy:

$$H[X] = -\sum_{j=1}^{l} p_j \log_2 p_j$$

- Can statistical mechanics be given an information-theoretic foundation?
- Can the Second Law of Thermodynamics be given an information-theoretic foundation?

Principle of Maximum Entropy

An Example: *Burger's Problem*

- A graduate student's daily average meal cost = $2.5
- What is the frequency that each item being ordered?
- $p(B) + p(C) + p(F) + p(T) = 1$
- $\$1p(B) + \$2p(C) + \$3p(F) + \$8p(T) = \$2.5$
- Still cannot determine the frequencies uniquely ……

Berger's Burger

Item	Price	Calories
Burger	$1	1000
Chicken	$2	600
Fish	$3	400
Tofu	$8	200

Principle of Maximum Entropy (cont'd)

Suppose that we are given a set of measurements and we associate with them a set of probabilities,

$$\{p_1, p_2, \ldots, p_n\}$$

Obviously these probabilities are between 0 and 1 and must satisfy the constraint that they sum to unity.

$$\sum_{i=1}^{n} p_i = 1 \qquad (1)$$

References :

1) Entropy optimization principles with applications / J.N. Kapur, H.K. Kesavan Imprint Boston : Academic Press, c1992
2) Maximum-entropy models in science and engineering / J.N. Kapur Imprint New York : Wiley, 1989

Principle of Maximum Entropy (cont'd)

Additional Constraints

However, eq.(1) is very general, and does not tell us much about these probabilities.

We may specify *m* additional constraints on these probabilities,

$$\sum_{i=1}^{n} p_i g_{ri} = a_r; \quad r = 1, \dots, m < n - 1 \qquad (2)$$

For example we may specify the mean or the variance of certain function and *m* in this case is 2.

$$\sum_{i=1}^{n} p_i x_i = \mu\,; \quad \sum_{i=1}^{n} p_i \,(x_i - \mu)^2 = \sigma^2 \qquad (3)$$

Principle of Maximum Entropy (cont'd)

Problem:

There are many solutions that can satisfy these constraints!!!

Now, the question is how can we find the theoretical set of probabilities, $\{p_1, p_2, \ldots, p_n\}$ consistent with (1) and (2) in a most *unbiased* manner?

In our question above, we need a ***guiding principle*** to select the set of probabilities $\{p_1, p_2, \ldots, p_n\}$ since in general there are infinitely many solutions of that satisfy constraints (1) and (2) in our simple example.

Thus, what is the sensible principle to use in the selection among many possible solutions?

Principle of Maximum Entropy (cont'd)

Reduction in Uncertainty

We reduce uncertainty with the help of information given to us and thus we achieve the maximum reduction in uncertainty by using all the information available to us.

In the present case, we have information (1) and (2), but there are still a lot of uncertainty about the set because (1) and (2) cannot determine the set uniquely.

Principle of Maximum Entropy (cont'd)

Scientific Viewpoint

In order not to be biased,
we must therefore take the attitude of
maximally uncertain about
what we do not know.
This is the best and most honest and unbiased thing that we can do.

Principle of Maximum Entropy (cont'd)

How does Entropy enter into this discussion?

We know from our discussion of the measure of information, that entropy is a measure of average information.
The purpose of providing information is to reduce uncertainty.
The fact that we ***want to be least biased*** about the set means that we want to add as little extra information as possible on these sets

$$\{p_1, p_2, \ldots, p_n\}$$

except constraints (1) and (2).

Principle of Maximum Entropy (cont'd)

The Guiding Principle

There are two equivalent interpretations on the following principle:
We like to be as unbiased as possible on the selection of $\{p_1, p_2, \ldots, p_n\}$ *that are all consistent with* (1) *and* (2).

Therefore we should minimize the information we added (*minimizing bias*), *or equivalently, maximizing the information content provided by the set consistent with* (1) *and* (2).

Thus we must maximize the average information contained in this set of probabilities consistent with the constraints (1) *and* (2).

This is called the ***principle of maximum entropy***.

Principle of Maximum Entropy (cont'd)

Edwin Thompson Jaynes

The maximum entropy principle arose in statistical mechanics in the 19th century and has been advocated for use in a broader context by E. T. Jaynes.

- Maximum entropy principle arose in statistical mechanics
- If nothing is known about a distribution except that it belongs to a certain class
- Distribution with the largest entropy should be chosen as the default
- Motivation:
    - Maximizing entropy minimizes the amount of prior information built into the distribution
    - Many physical systems tend to move towards maximal entropy configurations over time

Jaynes, E. T. (1957). "Information Theory and Statistical Mechanics". Physical Review 106 (4): 620–630; "Information Theory and Statistical Mechanics II". Physical Review 108 (2): 171–190.

Principle of Maximum Entropy (cont'd)

Some Examples:

Example 1: Gaussian

If X is continuous and has known first and second moments α_i for i = 1, 2 and $\alpha_2 - \alpha_1^2 > 0$, then the distribution is N(μ, σ^2) with

$$\mu = \alpha_1,\ \sigma^2 = \alpha_2 - \alpha_1^2\ .$$

Example 2: Exponential

If X is positive and continuous and has a known first moment α_1, then X is exponential with mean α_1.

Example 3:

Uniform on a finite set {1, ..., k} is the distribution with no moment constraints.

Principle of Maximum Entropy (cont’d)

Example from Physics

- Temperature of a gas corresponds to the average kinetic energy of the molecules in the gas

$$\sum_i p_i \frac{1}{2} v_i^2 m_i$$

- Distribution of velocities in the gas at a given temperature

- This distribution is the maximum entropy distribution under the temperature constraint: Maxwell-Boltzmann distribution

- Corresponds to the macro-state that has the most micro-states

Principle of Maximum Entropy (cont'd)

Maximum Entropy

The principle of maximum entropy arises quite naturally from the viewpoint of information theory. Its use is widespread over science and engineering. The mathematical question is therefore to find the solution $\{p_1, p_2, \ldots, p_n\}$ *that* ***maximize the entropy subject to the constraints***

$$(1) \qquad \sum_{i=1}^{n} p_i = 1$$

and

$$(2) \qquad \sum_{i=1}^{n} p_i g_{ri} = a_r; \quad r = 1, \ldots, m < n - 1$$

Principle of Maximum Entropy (cont'd)

Formulation

- Maximize entropy

$$H[p] = -\sum_{i=1}^{n} p_i \log p_i$$

- Subject to

$$p_i \geq 0$$

$$\sum_{i=1}^{n} p_i = 1$$

$$\sum_{i=1}^{n} p_i r_{ij} = \alpha_j, \text{for } 1 \leq j \leq m$$

Principle of Maximum Entropy (cont'd)

Maximum Entropy Distribution

- Form Lagrangian

$$J(p) = -\sum_{i=1}^{n} p_i \log p_i + \lambda_0 \left(\sum_{i=1}^{n} p_i - 1 \right) + \sum_{j=1}^{m} \lambda_j \left(\sum_{i=1}^{n} p_i r_{ij} - \alpha_j \right)$$

- Take derivative with respect to p_i: $-1 - \log p_i + \lambda_0 + \sum_{j=1}^{m} \lambda_j r_{ij}$

- Set this to 0, and the solution is maximum entropy distribution

$$p_i^* = \frac{e^{\sum_{j=1}^{m} \lambda_j r_{ij}}}{e^{1-\lambda_0}}$$

$\lambda_0, \lambda_1, \ldots, \lambda_m$ are chosen such that $\sum_{i=1} p_i^* = 1$, and $\sum_{i=1} p_i^* r_{ij} = \alpha_j$.

Principle of Maximum Entropy (cont'd)

Example: Rolling a dice with no constraint

- Let χ = {1, 2, 3, 4, 5, 6}
- No other constraint
- What is the best guess of a distribution if the only requirement it must satisfy $\sum_{i=1}^{6} p_i = 1$?
- The least biased choice is all the outcomes are probably likely
- Maximum entropy distribution is therefore a uniform distribution with

$$p_i = \frac{1}{6}$$

Principle of Maximum Entropy (cont'd)

Now consider rolling a dice with extra constraints

- This example was used by Boltzmann
- Suppose *n* dice are thrown on the table
- The average number of dots showing is $\sum_{i=1}^{6} ip_i = \alpha$, e.g., $\alpha = 4.5$
- What is the proportion of the dice are showing face *i*, *i* = 1, 2, 3, 4, 5, 6?

Principle of Maximum Entropy (cont'd)

Rolling a dice with additional constraints

- Assume n_i dice show face *i*
- There are $\binom{n}{n_1,\ldots,n_6}$ possible configurations
- This is a macro-state indexed by $n_1, \ldots, n_6$ with $\binom{n}{n_1,\ldots,n_6}$ microstates, each having probability $1/6^n$.
- Constraint: $\sum_{i=1}^{6} i n_i = n\alpha$, or $\sum_{i=1}^{6} i p_i = \alpha$
- Using maximum entropy solution, we find

$$p_i^* = \frac{e^{\lambda i}}{\sum_{i=1}^{6} e^{\lambda i}}$$

Principle of Maximum Entropy (cont'd)

Rolling a dice with additional constraints (cont'd)

Form Lagrangian

$$J(p) = -\sum_{i=1}^{n} p_i \log p_i + \lambda_0 \left(\sum_{i=1}^{n} p_i - 1 \right) + \lambda_1 \left(\sum_{i=1}^{n} i p_i - 4.5 \right)$$

Take derivative with respect to p_i: $-1 - \log p_i + \lambda_0 + \lambda_1 i$ and set the derivatives to zero for the condition of maximum.

$$p_i = e^{-1+\lambda_0+\lambda_1 i}$$

Therefore,

$$1 = \sum p_i = e^{-1+\lambda_0} \sum i e^{\lambda_1 i} \; ; \; 4.5 = \sum i p_i = \frac{\sum i e^{\lambda_1 i}}{\sum e^{\lambda_1 i}} = \frac{\sum i x_i}{\sum x_i}$$

where $x_i = e^{\lambda_1 i}$.

Principle of Maximum Entropy (cont'd)

Return to the Burger's Problem:

- ➢ $p^*(B) = e^{\lambda_0 - 1 + \lambda_1}; p^*(C) = e^{\lambda_0 - 1 + 2\lambda_1}; p^*(F) = e^{\lambda_0 - 1 + 3\lambda_1}: p^*(T) = e^{\lambda_0 - 1 + 8\lambda_1}$
- ➢ $p(B) + p(C) + p(F) + p(T) = 1$
- ➢ $\$1p(B) + \$2p(C) + \$3p(F) + \$8p(T) = \$2.5$
- ➢ Solution: $\lambda_0 = 1.2371;\ \lambda_1 = 0.2586$

Item	p^*
Berger	0.3546
Chicken	0.2964
Fish	0.2478
Tofu	0.1011

Berger's Burger

Item	Price	Calories
Burger	\$1	1000
Chicken	\$2	600
Fish	\$3	400
Tofu	\$8	200

Principle of Maximum Entropy (cont'd)

Solving Missing Number problem with Maximum Entropy Method

Consider the following problem for finding a missing number in a given list of n numbers, $x_1, x_2, \ldots, x_n$

Given that x belongs to the same set as $x_1, x_2, \ldots, x_n$ and $a \leq x \leq b$.

We would like to find the missing number x

Different maximum-entropy estimates of the missing value x are obtained by using different methods and by using different measures of entropy.

1) 1st method, the estimate is either a or b or x' where x' is an entropic mean of $x_1, x_2, \ldots, x_n$.
2) 2nd method, the estimate depends on a , b and $\sum_{i=1}^{n} x_i$.
3) 3rd method, it depends on all values $x_1, x_2, \ldots, x_n$.

Principle of Maximum Entropy (cont'd)

Entropy by Size of numbers

Shannon Entropy for proportion distribution of numbers

Let x_1, x_2, …, x_n be n given values and let x be a missing value about which we have no other information except that it belongs to the same set to which x_1, x_2, …, x_n belong. One estimation of x is to choose x so that x_1, x_2, …, x_n and x are as equal among themselves as possible.

We now choose x to maximum a measure of equality of these (n +1) values.
Let's consider that the probability of each value of the occurrence probability given by

$$p_i = \frac{x_i}{T+x}, i = 1, \dots, n \text{ and } p(x) = \frac{x}{T+x} \text{; with } T = \sum_{i=1}^{n} x_i$$

The Shannon entropy gives for the probability (or proportion distribution)

$$H[x] = -\left(\sum_{i=1}^{n} \frac{x_i}{T+x} \ln \frac{x_i}{T+x}\right) - \left(\frac{x}{T+x} \ln \frac{x}{T+x}\right)$$

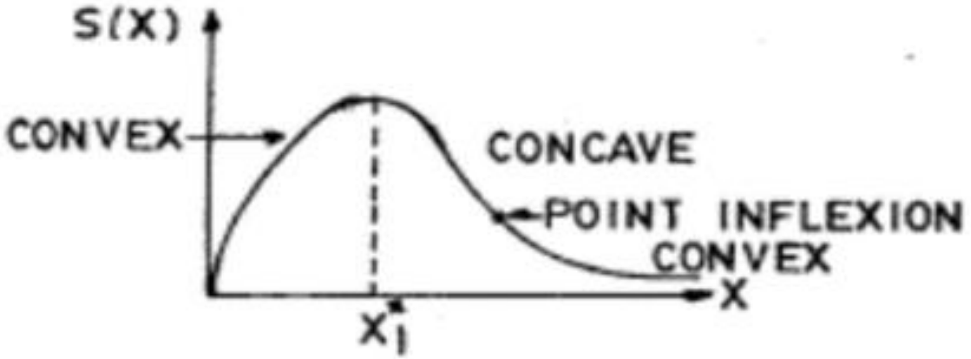
Result: Entropy by Size of numbers

We maximize $H[x]$ subject to $a \le x \le b$, so we take derivative of $H[x]$.

$$\frac{dH}{dx} = \frac{1}{(T+x)^2}\left[\sum_{i=1}^{n} x_i \ln x_i \; - T \ln x\right]$$

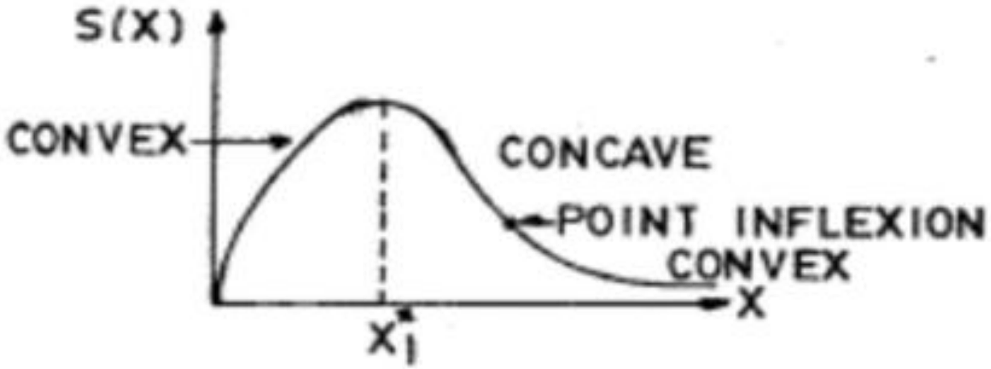
$$\frac{d^2H}{dx^2} = \frac{1}{(T+x)^3}\left[2\left(T \ln x - \sum_{i=1}^{n} x_i \ln x_i\right) - \frac{T(T+x)}{x}\right]$$

When $x = 0$, $dH/dx = \infty$, $d^2H/dx^2 < 0$



Result: Entropy by Size of numbers

As x increases $H[x]$ is a monotonic increasing concave function of x until x reaches the value x_1* where dH/dx becomes zero, and d^2H/dx^2 is still negative.



After that, $H[x]$ begins to decrease and at some stage, d^2H/dx^2 becomes zero, giving rise to a point of inflexion in the $H – x$ curve, and thereafter d^2H/dx^2 is positive and $H[x]$ becomes a convex function.

Ultimately, as $x \rightarrow \infty$, $dH/dx \rightarrow 0$.

Result: Entropy by Size of numbers

The solution of Shannon Entropy for proportion distribution of numbers is to maximize the Shannon entropy with respect to the unknown *x*

By maximizing $H[x]$, we get the estimate called the Shannon entropic mean of $x_1, x_2, \ldots, x_n$ to be

$$X_1' = \left(x_1^{x_1} x_2^{x_2} \ \ldots \ x_n^{x_n}\right)^{1/T}$$

Note that this result depends critically how we measure the probability of the numbers.

If we use other estimate of probability of occurrence of the numbers, then the result can be very different.

This again shows that the quality of estimate of the missing number depends on the input of the probability measure.

Principle of Maximum Entropy (cont'd)

Warning note on using Max Entropy Method

We see the analytic solution of the missing number problem using the method of Maximizing Shannon Entropy, but the solution needs to be checked with boundary values.
Suppose that the known numbers are in the range $a \le x_1, x_2, \ldots, x_n \le b$ and our solution of the missing number x is required to be also in this range $a \le x \le b$, then we must check if the analytic solution

$$x_{MaxEnt} = \left(x_1^{x_1} x_2^{x_2} \ \ldots\ x_n^{x_n}\right)^{1/T} \text{is also in this range } [a, b]$$

If not, then we must go back to the figure and check which is the bigger, $H[a]$ or $H[b]$

The bigger one will give the correct guess provided by the maximum entropy method.

Principle of Maximum Entropy (cont'd)

Summary

- Maximizing entropy minimizes the amount of prior information built into unknown distribution

- Maximum entropy distribution can be found explicitly

$$p_i^* = \frac{e^{\sum_{j=1}^{m} \lambda_j r_{ij}}}{e^{1-\lambda_0}}$$

- Maximum Entropy Principle widely used

Conclusion on the Maximum Entropy Method

From the solution of Shannon Entropy for proportion distribution of numbers, we see that the success of the Maximum entropy method is its simplicity. However, the goodness of the solution depends on the insight of the researcher of the problem on the measurement of probability of occurrence of the event, which in the case of missing number, we use the size distribution of the numbers as a measure, but is that justifiable?
In fact, size distribution is obviously not the only way.

- *For example, in lottery, the size of the numbers in the lottery is not so important for the next number to be guessed.*
- *In fact, one may use the past statistics of the occurrence probability of the number in past lottery ticket to estimate the probability of occurrence in the future lottery tickets, then we can apply the maximization of the Shannon entropy with respect to the unknown x*

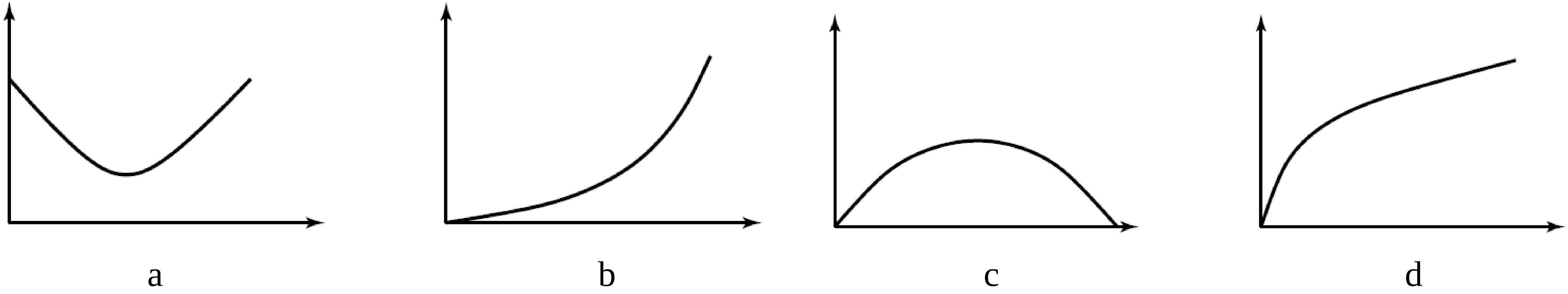
Useful Definitions, Theorems and Inequalities

- A function $f(x)$ is said to be *convex* over an interval (a, b) if for every $x_1, x_2 \in (a, b)$ and $0 \le \lambda \le 1$,

$$f(\lambda x_1 + (1-\lambda)x_2) \le \lambda f(x_1) + (1-\lambda)f(x_2).$$

- A function f is said to be *strictly convex* if equality holds only if $\lambda = 0$ or $\lambda = 1$.
  Examples: x^2, $|x|$, e^x

- A function f is *concave* if $-f$ is convex.
  Examples: *log x*

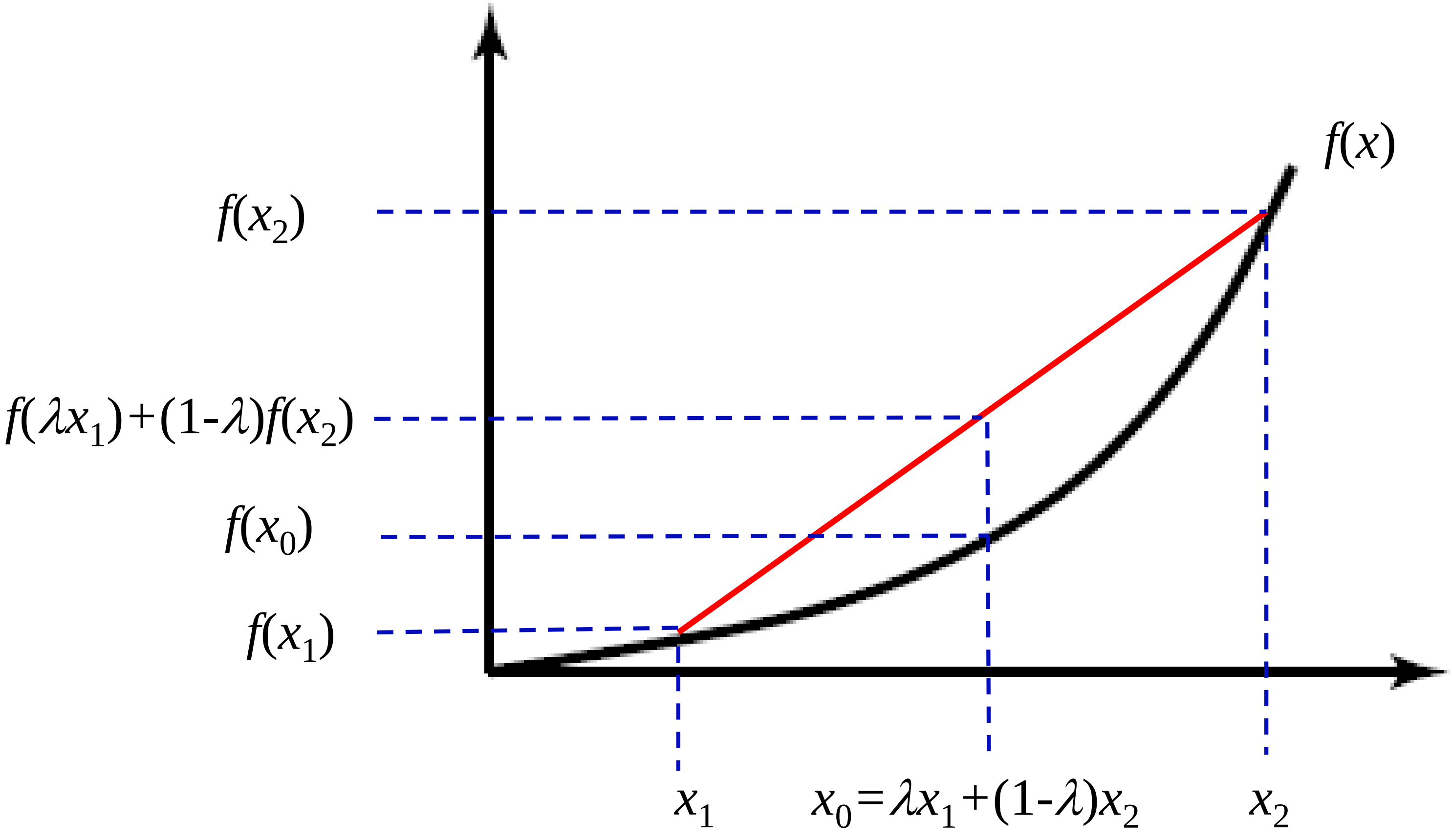


Examples of convex ((a) and (b)) and concave ((c) and (d)) functions

Useful Definitions, Theorems and Inequalities (cont'd)

Theorem

– If the function f has a second derivative that is nonnegative (positive) over an interval, the function is convex (strictly convex) over that interval.


$f(x)$
$f(x_2)$
$f(\lambda x_1)+(1-\lambda)f(x_2)$
$f(x_0)$
$f(x_1)$
x_1
$x_0=\lambda x_1+(1-\lambda)x_2$
x_2

Useful Definitions, Theorems and Inequalities (cont'd)

Theorem (Jensen's Inequality)

– If f is a convex function and X is a random variable,

$$E[f(X)] \geq f(E[X])\ .$$

Moreover, if f is strictly convex, the equality implies that X = E[X] with probability 1 (i.e., X is a constant)

Proof We prove this for discrete distributions by induction on the number of mass points in a distribution. For a two-mass-point distribution, from the definition of convex functions, the inequality becomes

$$p_1 f(x_1) + p_2 f(x_2) \ \geq f(p_1 x_1 + p_2 x_2),$$

Suppose the theorem is true for distributions with k – 1 mass points. Let us write,

$$p_i' = \frac{p_i}{1 - p_k} \text{ for } i = 1, 2, \ldots, k - 1.$$

Useful Definitions, Theorems and Inequalities (cont'd)

Proof (cont'd) We have,

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

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

where the first inequality follows from the induction hypothesis and the second follows from the definition of convexity.

Useful Definitions, Theorems and Inequalities (cont'd)

Theorem (Log Sum inequality)

– For nonnegative numbers, $a_1, a_2, \dots, a_n$ and $b_1, b_2, \dots, b_n$,

$$\sum_{i=1}^{n} a_i \log \frac{a_i}{b_i} \geq \left(\sum_{i=1}^{n} a_i \right) \log \frac{\sum_{i=1}^{n} a_i}{\sum_{i=1}^{n} b_i}$$

With equality if and only if $\frac{a_i}{b_i}$ = constant

Proof Assume with loss of generality, $a_i, b_i > 0$. The function *f*(*t*) = *t* log *t* is strictly convex, since *f''*(*t*) > 0. By Jensen's inequality,

$$\sum p_i f(t_i) \geq f\left(\sum p_i t_i \right)$$

for $p_i \geq 0, \sum_i p_i = 1$. Setting $p_i = \frac{b_i}{\sum_{i=1}^{n} b_i}$ and $t_i = \frac{a_i}{b_i}$, one obtains the log sum inequality,

$$\sum_{i=1}^{n} \frac{a_i}{\sum_{i=1}^{n} b_i} \log \frac{a_i}{b_i} \geq \left(\sum_{i=1}^{n} \frac{a_i}{\sum_{i=1}^{n} b_i} \right) \log \frac{a_i}{\sum_{i=1}^{n} b_i} \quad ,$$

Useful Definitions, Theorems and Inequalities (cont'd)

Asymptotic Equipartition Property

The analog of the *Law of Large Numbers* in Information Theory is the *Asymptotic Equipartition Property* (***AEP***), which is a direct consequence of the weak law of large numbers.

Law of Large Numbers –

For independent, identically distributed (i.i.d.) random variables X_i, $\frac{1}{n}\sum_{i=1}^{n} X_i$ is close to its expectation value, $E[X]$.

AEP – Given that the probability of observing a sequence of independent, identically distributed (i.i.d.) random variables $X_1, X_2, \ldots, X_n$, is $p(X_1, X_2, \ldots, X_n)$. The quantity $\frac{1}{n}\log\frac{1}{p(X_1,X_2,\ldots,X_n)}$ is close to the entropy H. Therefore, the probability assigned to an observed sequence will be close to 2^{-nH}.

As a consequence, ***AEP*** enables us to divide the set of all sequences into two sets, ***the typical set***, where the sample entropy is close to the true entropy, and ***the non-typical set***, which contains the other sequences.

Useful Definitions, Theorems and Inequalities (cont'd)

Asymptotic Equipartition Property

***Theorem* (*AEP*)**

– If the probability of observing a sequence of independent, identically distributed (i.i.d.) random variables $X_1, X_2, \dots, X_n$, is $p(X_1, X_2, \dots, X_n)$, then

$$-\frac{1}{n}\log p(X_1, X_2, \dots, X_n) \to H[X] \text{ in probability.}$$

Proof Since X_i are , identically distributed (i.i.d.) random variables, and functions of i.i.d. random variables are also i.i.d. random variables, therefore, $p(X_1, X_2, \dots, X_n)$ is also an i.i.d. random variable. Hence, by the weak law of large numbers,

$$-\frac{1}{n}\log p(X_1, X_2, \dots, X_n) = -\frac{1}{n}\sum_{i=1}^{n} p(X_i) \to -E[p(X)] = H[X]$$