Introduction to Information Theory
CS 446/546

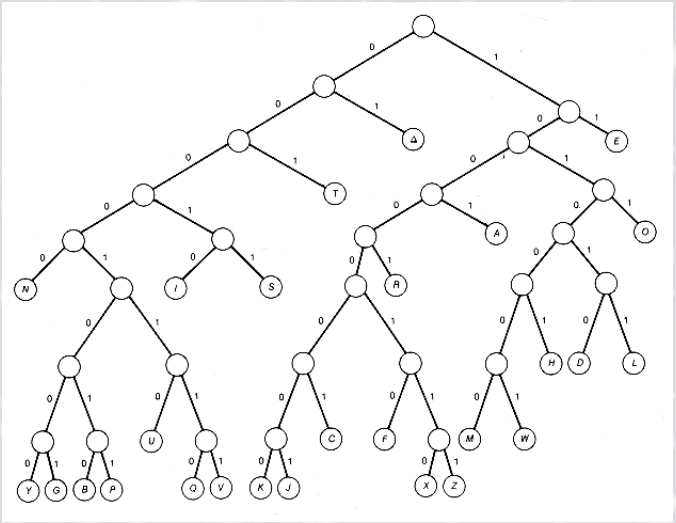
Outline

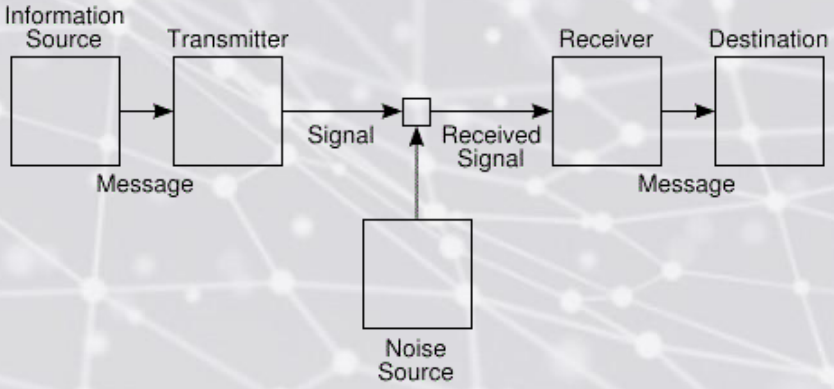
- Overview / Algorithmic Information Theory / Holographic Universe (?!)
- Entropy
- The Source Coding Theorem
- Kullback-Leibler Divergence, Information Inequality, Mutual Information
- Noisy Channel Theorem

Overview

- **Information Theory** deals with compact representations of data (i.e. **data compression/source coding**), as well as with transmitting and storing data robustly (i.e. **error correction/channel coding**).

- How does this relate to Machine Learning?

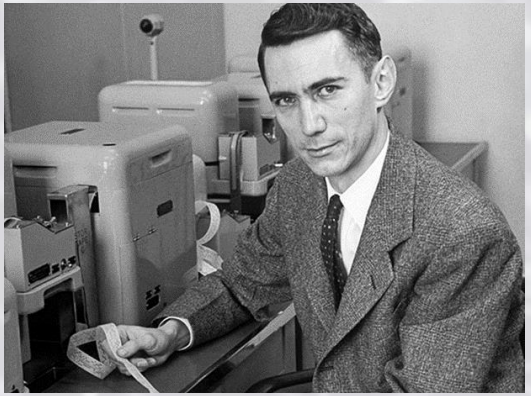




Overview

- How does this relate to Machine Learning?

- In his seminal paper "A Mathematical Theory of Communication" (1948), Shannon claimed that "the semantic aspects of communication are irrelevant to the *engineering problem*."



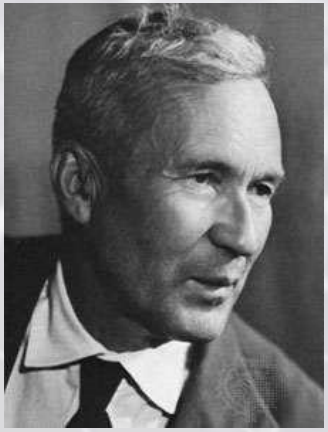
Shannon

- For Machine Learning applications, Information Theory provides us with a rich analytical scheme that differs from many conventional ML approaches – in particular these informational notions provide a **data and domain-agnostic framework** for ML applications.

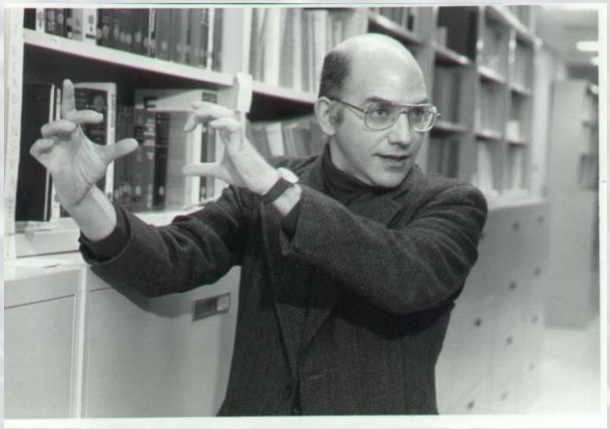
- For example, the concepts of "independence", non-informative priors, noise, etc., can all be articulated – in some sense *more generically* – in terms of information theory.

Overview: (aside) Algorithmic Information Theory

- Information Theory is in fact vital, more broadly, to foundational issues in computer science.



Kolmogorov



Chaitin

- In the field of algorithmic information theory (Kolmogorov, Chaitin), problems pertaining to complexity and computability are framed in terms of *information*.
- Informally, the information content of a string is *equivalent* to the most-compressed possible representation of that string (in this way an 5,000 page encyclopedia contains less information than a 5,000 page "random" string).

Overview: (aside) Algorithmic Information Theory

Definition of Kolmogorov Complexity:

If a description **d(s)** of a string *s* is of minimal length (i.e. it uses the fewest bits), it is called a minimal description of *s*. Thus, the length d(s) is the Kolmogorov complexity of s, written K(s).

$$K(s)=|d(s)|$$

Overview: (aside) Algorithmic Information Theory

Definition of Kolmogorov Complexity:

If a description **d(s)** of a string *s* is of minimal length (i.e. it uses the fewest bits), it is called a minimal description of *s*. Thus, the length d(s) is the Kolmogorov complexity of s, written K(s).

$$K(s) = |d(s)|$$

$K(s_1) << K(s_2)$

Example 1: abababababababababababababababab

Example 2: 4c1j5b2p0cv4w1x8rx2y39umgw5q85s7

Overview: (aside) Algorithmic Information Theory

Definition of Kolmogorov Complexity:

If a description **d(s)** of a string *s* is of minimal length (i.e. it uses the fewest bits), it is called a minimal description of *s*. Thus, the length d(s) is the Kolmogorov complexity of s, written K(s).

$$K(s) = |d(s)|$$

$K(s_1) << K(s_2)$

Example 1: abababababababababababababababab

Example 2: 4c1j5b2p0cv4w1x8rx2y39umgw5q85s7

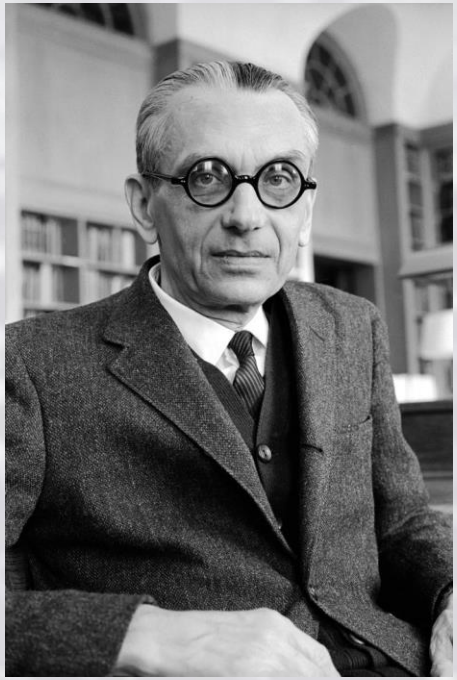
(*) Consequence: Most string are *complex*!

Overview: (aside) Algorithmic Information Theory

Definition of Kolmogorov Complexity:

$$K(s) = |d(s)|$$

(*) Consequence: Most string are *complex*!



Gödel

(*) A few interesting results:

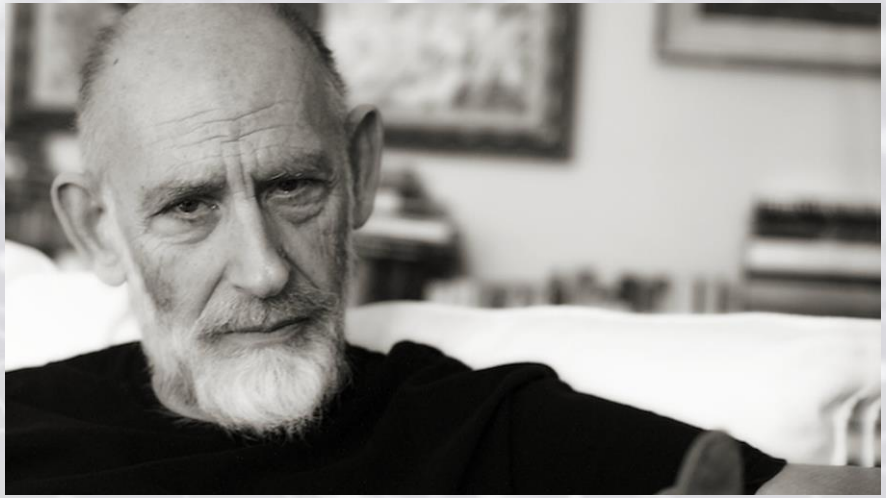
(i) There exist strings of arbitrarily large Kolmogorov complexity.

(ii) K is not a computable function (but one can compute upperbounds).

(iii) *Chaitin's Incompleteness Theorem* (using Gödelization): One can't in general prove that a specific string is complex.

Aside: Is the Basis of the Universe Information?

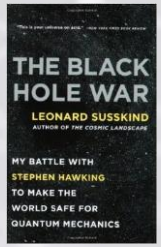
- The so-called "Black Hole Wars" arose from a debate between, in principle, Leonard Susskind and Stephen Hawking regarding the nature of information in black holes.



- While Hawking argued that information is lost in black holes, Susskind asserted that this would violate the law of the conservation of information. The debate spurred the "holographic principle" which postulates that in lieu of being lost, information is in fact preserved and stored on the boundary of a given system.

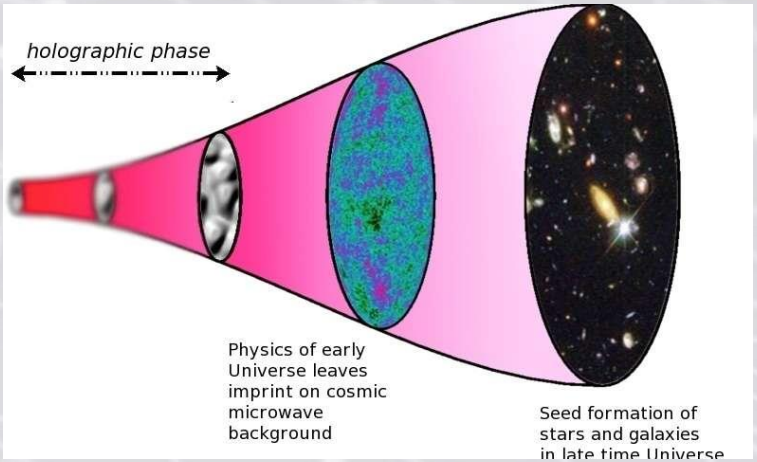
Susskind: "On the world as hologram":

https://www.youtube.com/watch?v=2DIl3Hfh9tY



Aside: Is the Basis of the Universe Information?

- A study in 2017 revealed substantial evidence that **we live in a holographic universe**.
- In this view, we might be caught inside a giant hologram; the cosmos is a projection, much like a 3D simulation.

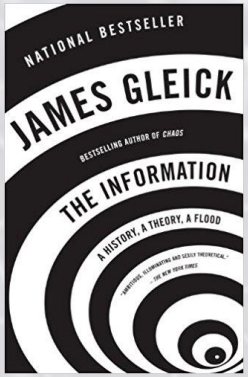


- If the nature of reality is in fact reducible to information itself, that implies a conscious mind on the receiving end, to interpret and comprehend it.

https://journals.aps.org/prl/abstract/10.1103/PhysRevLett.118.041301

Aside: Is the Basis of the Universe Information?

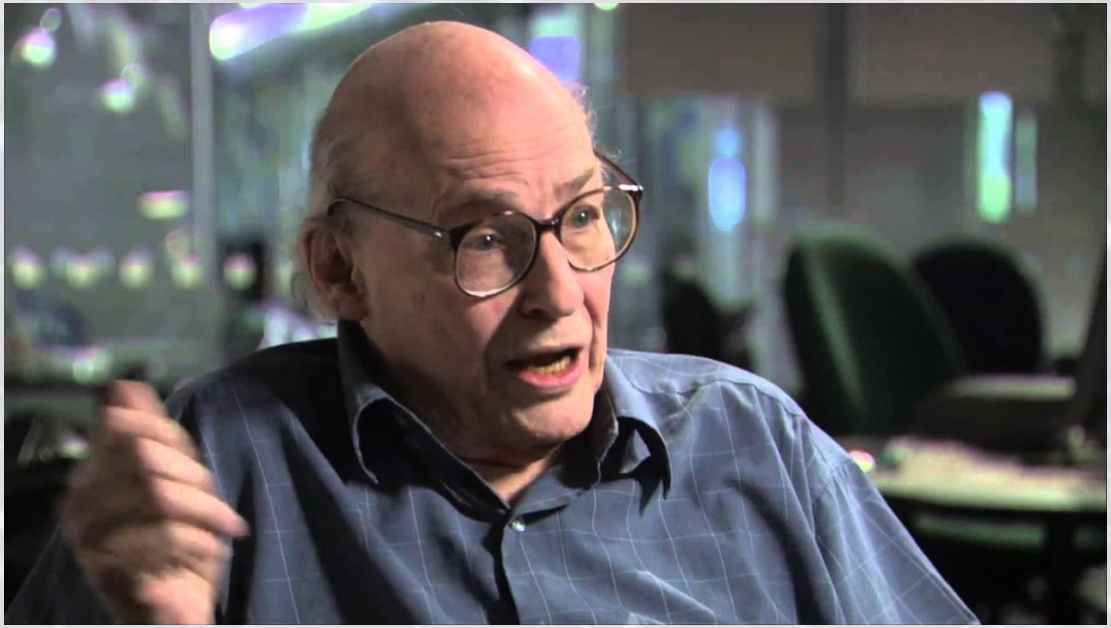
- A. Wheeler believed in a *participatory universe*, where consciousness holds a central role.
- It is possible that information theory may in the future help bridge the gap between general relativity and quantum mechanics, or aid in our understanding of dark matter.



"The universe is a physical system that contains and processes information in a systematic fashion and that can do everything a computer can do.“ – Seth Lloyd, MIT

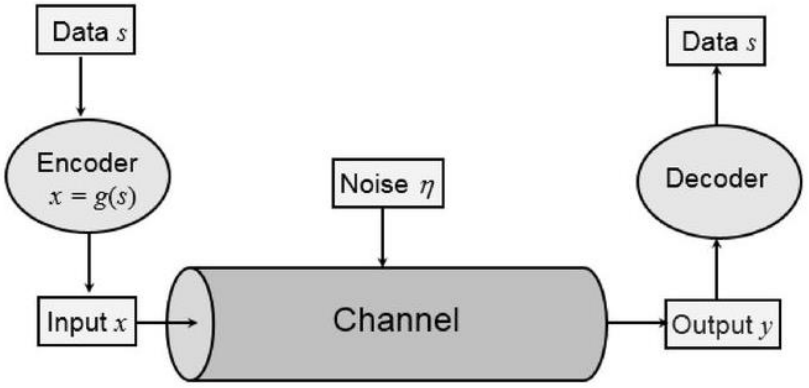
Aside: Is the Basis of the Universe Information?

- Just for fun…here is a short conversation with Minsky on the question of whether information is a basic building block of reality.



https://www.closertotruth.com/series/information-fundamental

Overview: Information Theory

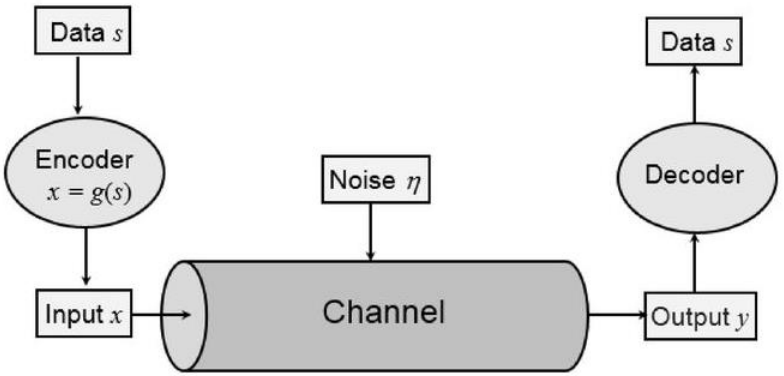


Source generates message s – sequence of symbols
s is encoded into channel input codewords $x = g(s)$
Channel input output x => output y decoded to s

• A source generates messages, $\mathbf{s}=(s_1,\ldots,s_k)$; a communication channel is used to transmit data from its input to its output; if the data are transmitted without error, then they have been successfully *communicated*.

• Before being transmitted, each message **s** is transformed by an encoder: $\mathbf{x}=g(\mathbf{s})$, which renders a sequence of codewords: $\mathbf{x}=(x_1,\ldots,x_n)$, where each codeword is the value of a random variable which can adopt any of *m* different values from a codebook.

Overview: Information Theory

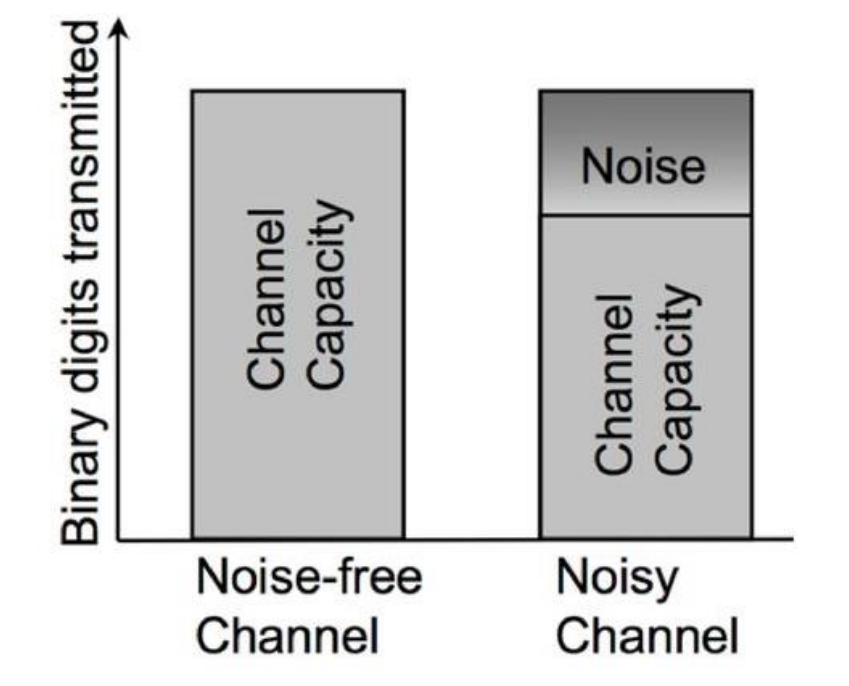


Source generates message s – sequence of symbols
s is encoded into channel input codewords $x = g(s)$
Channel input output x => output y decoded to s

• Typically, the encoded version g(**s**) of the original message **s** is a compression of the original message (e.g., remove natural redundancies). If the compression allows the original message to be decoded perfectly, then we say the compression is *lossless* (otherwise it is *lossy*).

• In order to ensure that the encoded message can withstand the effects of a noisy communication channel, some redundancy may be added to the codewords before they are transmitted.

Overview: Information Theory



• *Channel capacity* is the **maximum amount of information** which can be communicated from a channel's input to its output.

• The capacity (units of information/time) of a noiseless channel is numerically equal to the rate at which it transmits binary digits, whereas the capacity of a noisy channel is less than this.

• For example: For an alphabet of α symbols (α=2 is binary), if a noiseless channel transmits data at a fixed rate of n symbol/sec, then it transmits information at a maximum rate/channel capacity of $(n \log \alpha)$ bits/sec, or n bits/sec for binary data.

Entropy

Historically, **Shannon's desiderata** for defining information mathematically included (4) basic sets of properties:

(1) Continuity: (information associated with an outcome should increase/decrease smoothly as the probability of the outcome changes).

(2) Symmetry: The amount of information associated with a sequence of outcomes doesn't depend on the order of those outcomes.

(3) Maximal Value: The amount of information associated with a set of outcomes cannot be increased if those outcomes are already equally probable.

(4) Additive: The information associated with a set of outcomes is obtained by adding the information of individual outcomes.

Entropy

Motivating Example:

Suppose we are given a *biased coin* that, we are told, lands heads up 90% of the time.

If we wanted to quantify our "surprise" after a flip, we could consider the expression $1/p(x)$, where $p(x)$ is the probability of that particular outcome. In this way, our surprise associated with the outcome value x increases as the probability of x decreases.

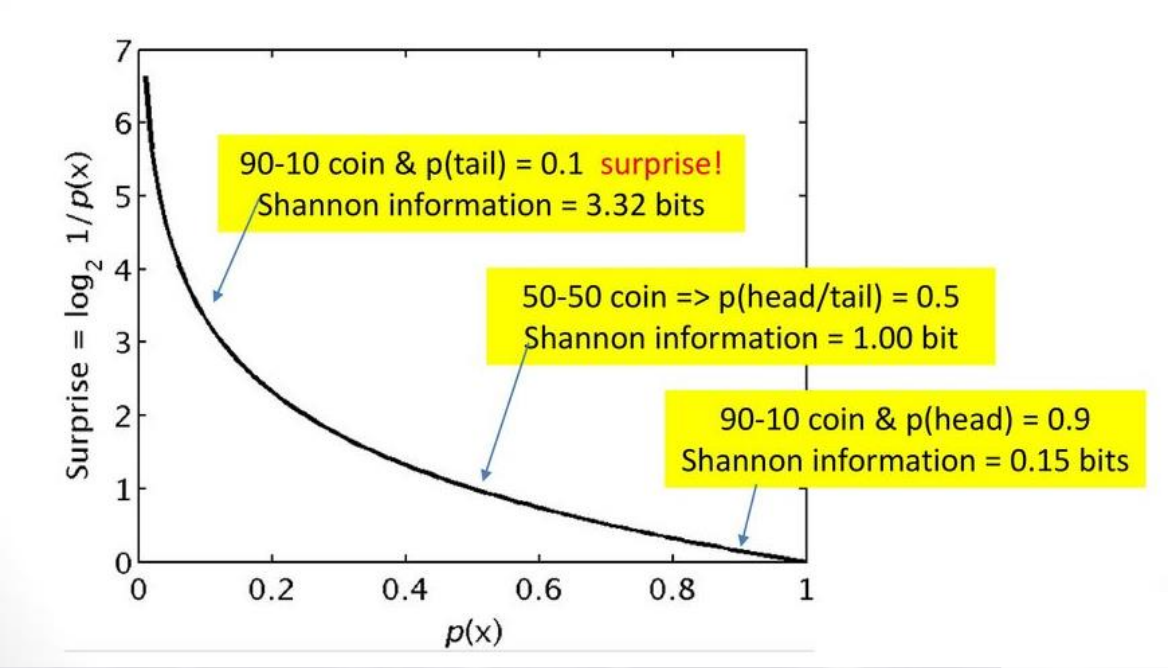
Entropy

Motivating Example:

Suppose we are given a *biased coin* that, we are told, lands heads up 90% of the time.

If we wanted to quantify our "surprise" after a flip, we could consider the expression 1/p(x), where p(x) is the probability of that particular outcome. In this way, our surprise associated with the outcome value x increases as the probability of x decreases.

In order to obey additivity (see previous slide), one can define surprise as: **log(1/p(x))** – this is known as the **Shannon information of *x***.



Entropy

Motivating Example:

Suppose we are given a *biased coin* that, we are told, lands heads up 90% of the time.

If we wanted to quantify our "surprise" after a flip, we could consider the expression 1/p(x), where p(x) is the probability of that particular outcome. In this way, our surprise associated with the outcome value x increases as the probability of x decreases.

In order to obey additivity (see previous slide), one can define surprise as: **log(1/p(x))** – this is known as the **Shannon information of *x***.

Lastly, if we want to compute the average surprise of a random variable X with associated probability distribution p(x), which quantity should we compute?

Entropy

Motivating Example:

Suppose we are given a *biased coin* that, we are told, lands heads up 90% of the time.

If we wanted to quantify our "surprise" after a flip, we could consider the expression 1/p(x), where p(x) is the probability of that particular outcome. In this way, our surprise associated with the outcome value x increases as the probability of x decreases.

In order to obey additivity (see previous slide), one can define surprise as: **log(1/p(x))** – this is known as the **Shannon information of *x***.

Lastly, if we want to compute the average surprise of a (discrete over K states) random variable X with associated probability distribution p(x), which quantity should we compute?

$$E\left[\log\left(1/X\right)\right] = \sum_{k=1}^{K} p(X=k)\log\left(1/p(X=k)\right) = -\sum_{i=1}^{K} p(X=k)\log\left(p(X=k)\right)$$

Entropy

• The **entropy** of a random variable X with distribution *p*, denoted by *H*(X) or sometimes *H*(p) is a measure of surprise/uncertainty. In particular, for a discrete random variable with *K* states, it is defined:

$$H(X) = -\sum_{k=1}^{K} p(X=k)\log_2\left(p(X=k)\right)$$

Usually we use log base 2, in which case the units are called *bits*.

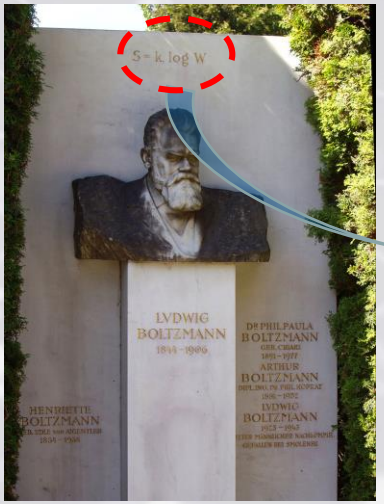
Entropy

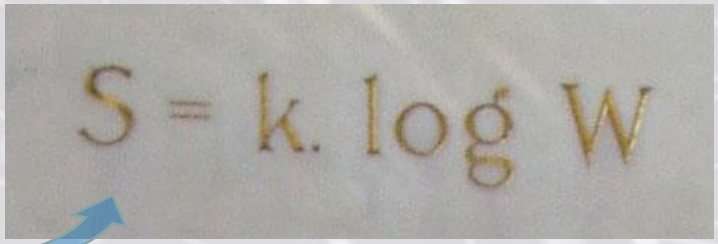
• The **entropy** of a random variable X with distribution *p*, denoted by *H*(X) or sometimes *H*(p) is a measure of surprise/uncertainty. In particular, for a discrete random variable with *K* states, it is defined:

$$H(X) = -\sum_{k=1}^{K} p(X=k)\log_2\left(p(X=k)\right)$$

Usually we use log base 2, in which case the units are called *bits*.

(*) Generally, entropy refers to disorder or uncertainty, and the definition of entropy used in information theory is directly analogous to the definition used in statistical thermodynamics.





Boltzman entropy formula (~1872), as found on his gravestone.

Entropy

• The **entropy** of a random variable X with distribution *p*, denoted by *H*(X) or sometimes *H*(p) is a measure of surprise/uncertainty. In particular, for a discrete random variable with *K* states, it is defined:

$$H(X) = -\sum_{k=1}^{K} p(X=k)\log_2\left(p(X=k)\right)$$

Usually we use log base 2, in which case the units are called *bits*.

For example: For a fair coin, for X ϵ {H,T}, p(H)=p(T), we have that H(X)=1 bit (you should confirm this).

This means the "average Shannon information" for a fair coin is 1 bit of information (that is to say: one binary outcome with equiprobable outcomes has a Shannon entropy of 1 bit).

Entropy

• The **entropy** of a random variable X with distribution *p*, denoted by *H*(X) or sometimes *H*(p) is a measure of surprise/uncertainty. In particular, for a discrete random variable with *K* states, it is defined:

$$H(X) = -\sum_{k=1}^{K} p(X=k)\log_2\left(p(X=k)\right)$$

Usually we use log base 2, in which case the units are called *bits*.

For example: For a fair coin, for X ϵ {H,T}, p(H)=p(T), we have that H(X)=1 bit (you should confirm this).

This means the "average Shannon information" for a fair coin is 1 bit of information (that is to say: one binary outcome with equiprobable outcomes has a Shannon entropy of 1 bit).

Compare this result with the previous biased coin: X ϵ {H,T}, p(H)=0.9, p(T)=0.1, we have that H(X)=.469 bits (you should also confirm this).

(*) **Main idea**: the average uncertainty of the biased coin is less than that of an unbiased coin.

Entropy

$$H(X) = -\sum_{k=1}^{K} p(X = k) \log_2 \left(p(X = k) \right)$$

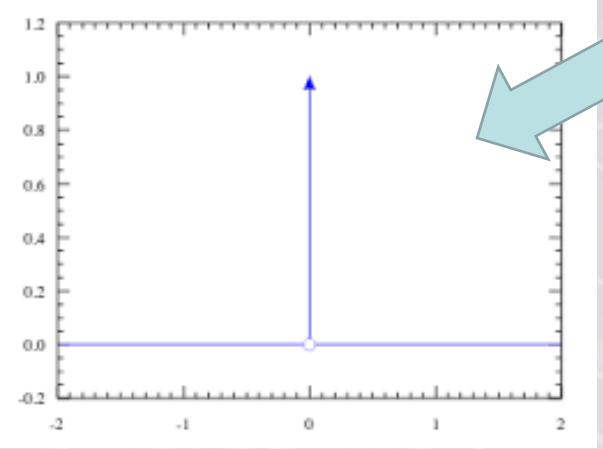
Q: When would entropy equal zero?

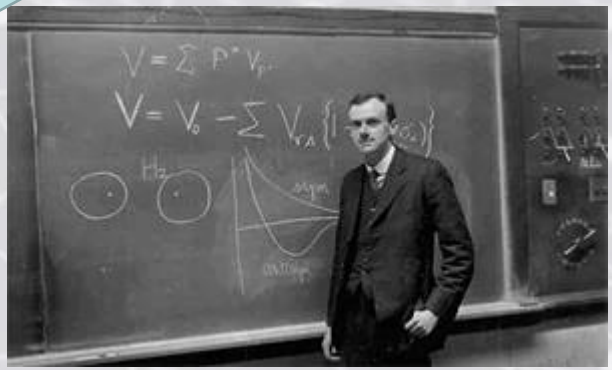
Entropy

$$H(X) = -\sum_{k=1}^{K} p(X = k) \log_2 \left(p(X = k) \right)$$

Q: When would entropy equal zero?

A: For a *deterministic event* (i.e. a probability distribution for which one outcome has probability 1 and all others have probability zero – this is known as *Dirac/delta distribution*) – intuitively: there is no "uncertainty" in this case.





Dirac

"Mathematical why": log(1) = 0, and $\lim_{x \to 0^+} x \log x = 0$ (prove this).

Entropy

$$H(X) = -\sum_{k=1}^{K} p(X = k) \log_2 \left(p(X = k) \right)$$

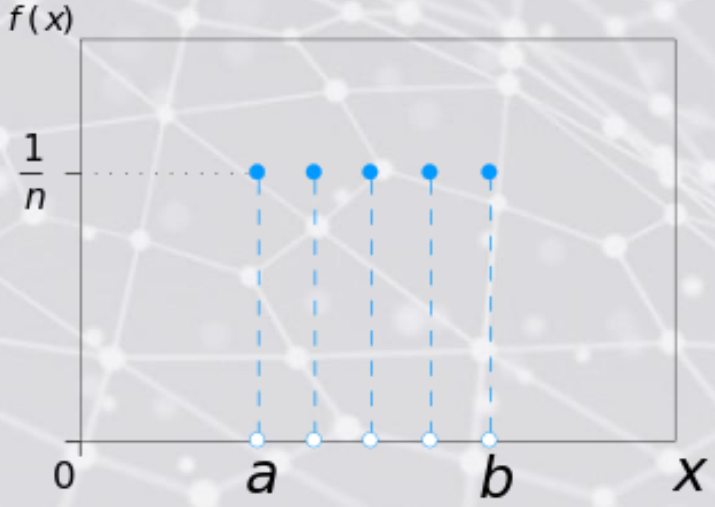
Q: When is entropy maximal?

Entropy

$$H(X) = -\sum_{k=1}^{K} p(X=k)\log_2\left(p(X=k)\right)$$

Q: When is entropy maximal?

A: When uncertainty is maximal, i.e. for the uniform distribution, p(x)=1/K. We'll prove this result shortly, but it should be intuitively clear.



Entropy

$$H(X) = -\sum_{k=1}^{K} p(X = k) \log_2 \left(p(X = k) \right)$$

• For the special case of (*Bernoulli*) binary random variables X ϵ {0,1}, we can write p(X = 1) = θ and p(X = 0) = 1- θ. Hence the entropy becomes:

$$H(X) = -\left[p(X = 1) \log_2 p(X = 1) + p(X = 0) \log_2 p(X = 0) \right]$$

Entropy

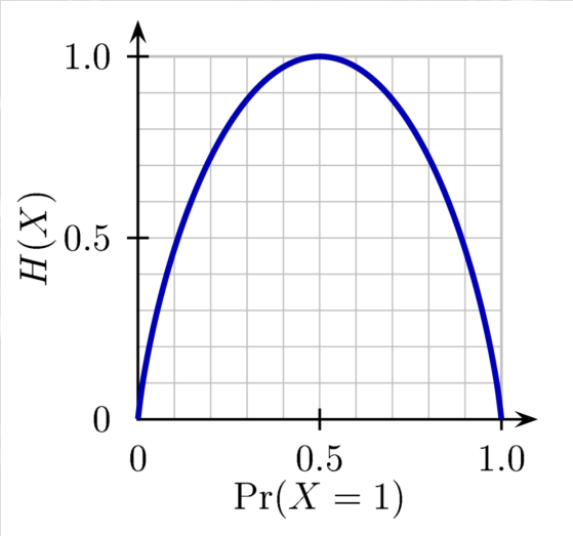
$$H(X) = -\sum_{k=1}^{K} p(X=k)\log_2\left(p(X=k)\right)$$

• For the special case of (*Bernoulli*) binary random variables X ϵ {0,1}, we can write p(X = 1) = θ and p(X = 0) = 1- θ. Hence the entropy becomes:

$$H(X) = -\left[p(X=1)\log_2 p(X=1) + p(X=0)\log_2 p(X=0)\right]$$

$$= -\left[\theta\log_2\theta + (1-\theta)\log_2(1-\theta)\right]$$

• This is called the **binary entropy function**, and is also written H(θ). The plot is shown; note that the maximum value of 1 coincides with the value θ = 0.5 (i.e., when the distribution is uniform).



Source Coding Theorem

• Shannon's *source coding theorem* guarantees that for any message there exists an encoding of symbols such that each channel input of C binary digits can convey, on average, close to C bits of information.

This encoding process yields inputs with a specific distribution p(X), which determines H(X), and therefore how much information each input carries.

The **capacity** of a discrete, noiseless channel is defined as the maximum number of bits it can communicate:

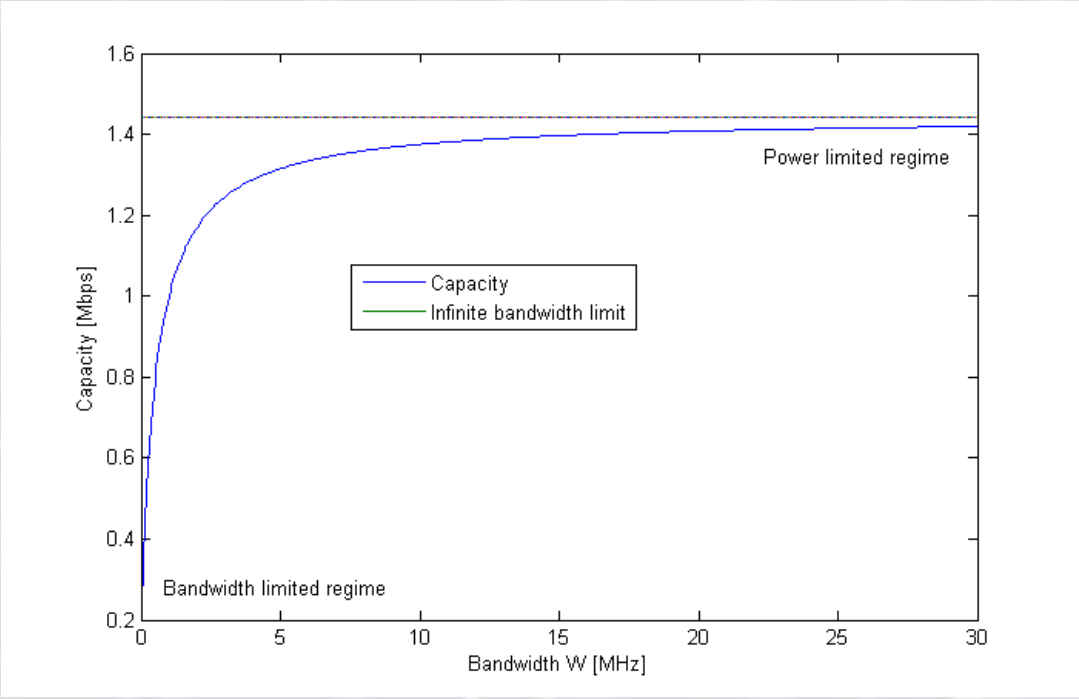
$$capacity = \max_{p(X)} H(X) \ \ bits/s$$

(*) In other words, the channel capacity is achieved by the distribution p(X) that makes H(X) as large as possible (i.e. the uniform distribution).

Source Coding Theorem

- The *source coding theorem* states that for a discrete, noiseless channel:

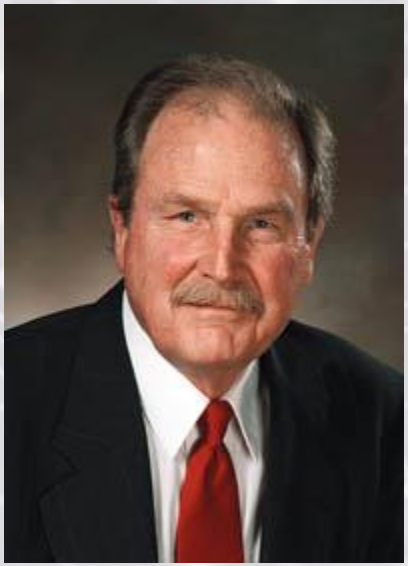
Let a source have entropy H (bits per symbol) and a channel have capacity C (bits per second). Then it is possible to encode the output of the source in such a way as to transmit at the average rate C/H – ε (symbols per second for arbitrarily small ε). It is not possible to transmit at an average rate greater than C/H (symbols per second).



Huffman Coding

• **Huffman coding** (1952) is a classic, greedy method for efficiently encoding symbols into a corresponding set of codewords. The algorithm results in an optimal *prefix code* (meaning no codewords share prefixes). The Huffman tree associated with an encoding is consequently a binary tree, with the property that leaves equate to codewords).

• Huffman coding can be regarded as an *entropy encoding method*: the basic idea is that more common symbols are generally represented using fewer bits, while less common symbols use more bits, on average. Huffman coding is used, among other applications, with JPEG and MPEG compression schemes.



Huffman

Huffman Coding

• **Huffman coding** (1952) is a classic, lossless greedy method for efficiently encoding symbols into a corresponding set of codewords. The algorithm results in an optimal *prefix code* (meaning no codewords share prefixes). The Huffman tree associated with an encoding is consequently a binary tree, with the property that leaves equate to codewords).

• Huffman coding can be regarded as an *entropy encoding method*: the basic idea is that more common symbols are generally represented using fewer bits, while less common symbols use more bits, on average.

The algorithm works recursively as follows:

Repeatedly join two nodes with the smallest probabilities to form a new node with the sum of the probabilities just joined. Assign a 0 to one branch and a 1 to the other branch.

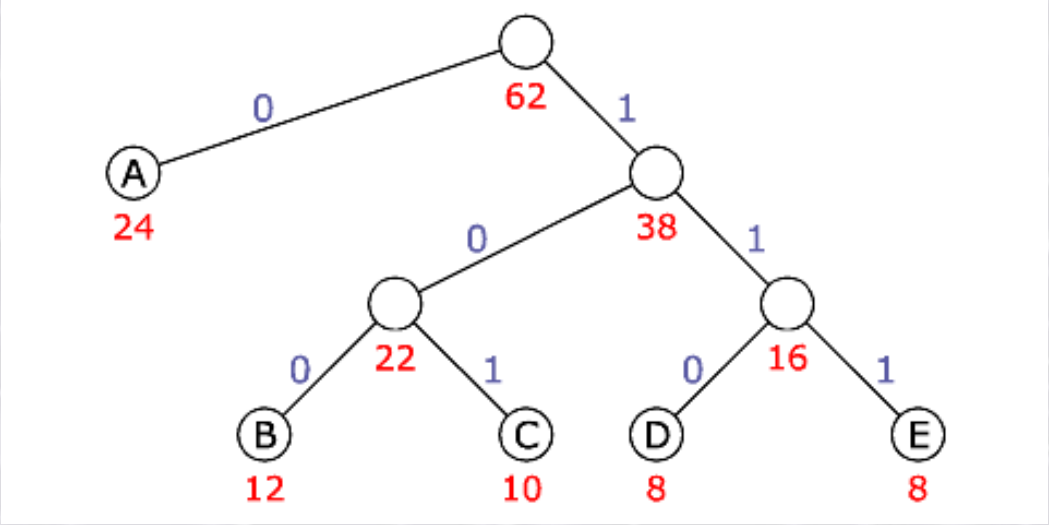
Rinse and repeat…

Huffman Coding

The algorithm works recursively as follows:

Repeatedly join two nodes with the smallest probabilities to form a new node with the sum of the probabilities just joined. Assign a 0 to one branch and a 1 to the other branch.

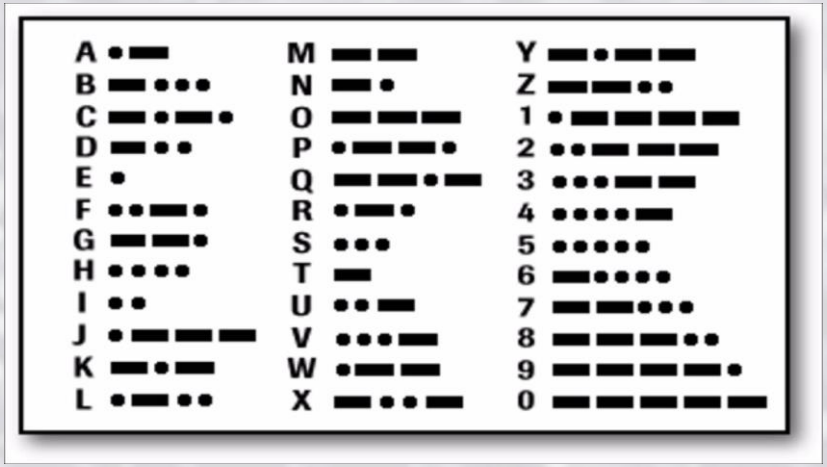
Demo: https://people.ok.ubc.ca/ylucet/DS/Huffman.html



Huffman Coding

Symbol	Frequency	Huffman Code
[space]	67962112	111
e	37907119	010
t	28691274	1101
a	24373121	1011
o	23215532	1001
i	21820970	1000
n	21402466	0111
s	19059775	0011
h	18058207	0010
r	17897352	0001
l	11730498	10101
d	10805580	01101
c	8982417	00001
u	8022379	00000
f	7486889	110011
m	7391366	110010
w	6505294	110001
y	5910495	101001
p	5719422	101000
g	5143059	011001
b	4762938	011000
v	2835696	1100000
k	1720909	11000011
x	562732	110000100
j	474021	1100001011
q	297237	11000010101
z	93172	11000010100

compare



Morse code

Define L(X) the *coding efficiency* (also known as ABL/average bits per letter) the product of the probability of each symbol x from an alphabet S and its code length (|c(x)|), summed over all symbols:

$$L(X) = \sum_{x \in S} p(x)\,|c(x)|$$

(*) A property of Huffman codes (from the Shannon source coding theorem) is that:

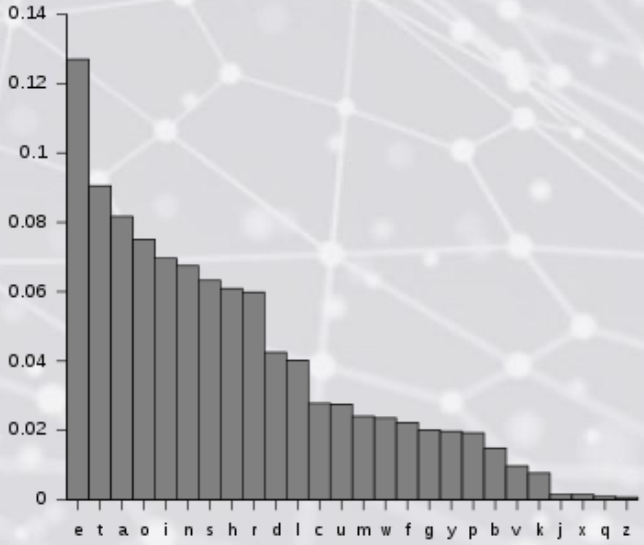
$$H(X) \le L(X) < H(X) + 1$$

Example: The Entropy of the English Language

• Let's consider the problem of computing the entropy of the English language (naturally this quantity can provide a useful bound for a multitude of source coding applications).

• If we take account of the relative frequency of each letter *x*, then we effectively consider a block of letters of length N =1 (we include space as the 27th letter). We'll call this our *first order estimate* of the entropy of English (H):

$$G_1 = \sum_{i=1}^{m=27} p(x_i) \log \frac{1}{p(x_i)} \approx 4.08 \ bits / letter$$



Example: The Entropy of the English Language

• Using a block length of N=2 effectively takes account of the dependencies between adjacent letters (there are $729=27^2$ such distinct pairs; denote $B_k=[x_i,y_j]$).

The second order estimate G_2 of H is:

$$G_2 = \frac{1}{2}\sum_{k=1}^{729} p(B_k)\log\frac{1}{p(B_k)} \approx 3.32 \ bits/letter$$

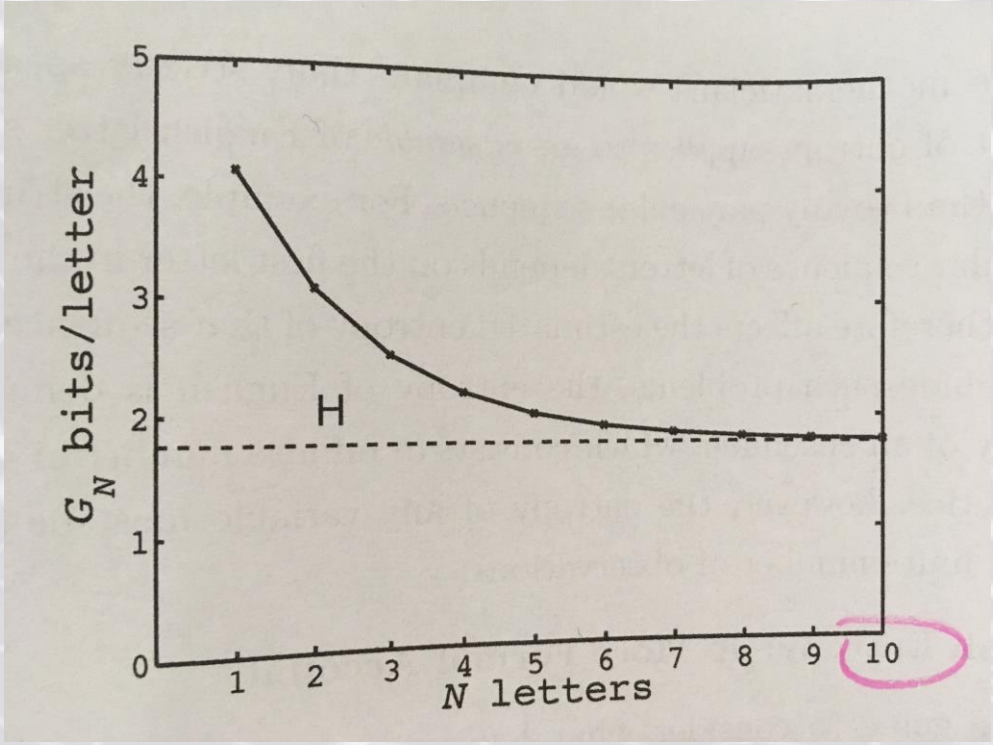
• Similarly, we can consider longer range dependencies if we use blocks of N = 3 letters (for 19,683 distinct letter triplets).

$$G_3 = \frac{1}{3}\sum_{k=1}^{19,683} p(B_k)\log\frac{1}{p(B_k)} \approx 2.73 \ bits/letter$$

Example: The Entropy of the English Language

• In principle, the process of calculating G_N for larger values of N (where N is the block length) renders the block probabilities p(Bk) nearly independent.

In practice, as N increases, the estimated entropy of English converges to a value of about G_N=1/8 bits/letter.



(*) If the entropy of English is H = 1.8 bits/letter, then Shannon's source coding theorem guarantees that we should be able to communicate letters using just over 1.8 binary digits per letter.

Joint Distributions

• Entropy can be defined analogously for a *joint distribution*:

$$H(X,Y) = \sum_i \sum_j p(X=i, Y=j) \log \frac{1}{p(X=i, Y=j)}$$

• H(X,Y) is commonly expressed in units *bits per pair*; the joint entropy H(X,Y) is the average amount of Shannon information of each pair of values, where this average is taken over all possible pairs.

• Just as entropy of a single variable can be considered a measure of uncertainty/non-uniformity, so the entropy of a joint distribution is also a measure of uncertainty/non-uniformity. If all possible pairs of values are equally probable, then this defines a uniform, *maximum entropy distribution*.

Joint Distributions

$$H(X,Y) = \sum_i \sum_j p(X=i, Y=j) \log \frac{1}{p(X=i, Y=j)}$$

• Recall that if random variables X and Y are *statistically independent,* then knowing the value of X provides no information about Y and *vice versa*. In particular, the joint distribution factors for independent variables, viz., p(X,Y)=p(X)p(Y).

(*) If X and Y are independent, then the entropy of the joint distribution p(X,Y) is equal to the summed entropies of the marginal distributions, namely:

$$H(X,Y) = H(X) + H(Y) \text{ when } X \perp Y$$

Pf. $$H(X,Y) = E\left[\log \frac{1}{p(x,y)}\right]$$

Joint Distributions

$$H(X,Y)=\sum_i\sum_j p(X=i,Y=j)\log\frac{1}{p(X=i,Y=j)}$$

• Recall that if random variables X and Y are *statistically independent,* then knowing the value of X provides no information about Y and *vice versa*. In particular, the joint distribution factors for independent variables, viz., p(X,Y)=p(X)p(Y).

(*) If X and Y are independent, then the entropy of the joint distribution p(X,Y) is equal to the summed entropies of the marginal distributions, namely:

$$H(X,Y)=H(X)+H(Y) \text{ when } X\perp Y$$

Pf.

$$H(X,Y)=E\left[\log\frac{1}{p(x,y)}\right]=E\left[\log\frac{1}{p(x)p(y)}\right]=E\left[\log\frac{1}{p(x)}+\log\frac{1}{p(y)}\right]$$

Joint Distributions

$$H(X,Y)=\sum_i\sum_j p(X=i,Y=j)\log\frac{1}{p(X=i,Y=j)}$$

• Recall that if random variables X and Y are *statistically independent,* then knowing the value of X provides no information about Y and *vice versa*. In particular, the joint distribution factors for independent variables, viz., p(X,Y)=p(X)p(Y).

(*) If X and Y are independent, then the entropy of the joint distribution p(X,Y) is equal to the summed entropies of the marginal distributions, namely:

$$H(X,Y)=H(X)+H(Y) \text{ when } X\perp Y$$

Pf.

$$H(X,Y)=E\left[\log\frac{1}{p(x,y)}\right]=E\left[\log\frac{1}{p(x)p(y)}\right]=E\left[\log\frac{1}{p(x)}+\log\frac{1}{p(y)}\right]=E\left[\frac{1}{p(x)}\right]+E\left[\frac{1}{p(y)}\right]=H(X)+H(Y)$$

Why?

Joint Distributions

$$H(X,Y) = H(X) + H(Y) \text{ when } X \perp Y$$

Example: Consider X, Y the values of two unbiased 6-sided dice after a roll.

H(X,Y) = log 36 $\approx$ 5.17 *bits per outcome pair*

H(X) = H(Y) = log 6 $\approx$ 2.59 *bits per outcome pair*.

Hence, H(X,Y) = H(X) + H(Y).

KL Divergence

• One way to measure the *dissimilarity* of two probability distributions, *p* and *q*, is known as the **Kullback-Leilber divergence** (KL Divergence) or *relative entropy*:

$$KL\big(p(X)\,\|\,q(X)\big)=\sum_{k=1}^{K}p(X=k)\log\frac{p(X=k)}{q(X=k)}$$

KL Divergence

• One way to measure the *dissimilarity* of two probability distributions, p and q, is known as the **Kullback-Leilber divergence** (KL Divergence) or *relative entropy*:

$$KL\big(p(X)\|q(X)\big)=\sum_{k=1}^{K}p(X=k)\log\frac{p(X=k)}{q(X=k)}$$

This can be rewritten as:

$$KL\big(p(X)\|q(X)\big)=\sum_{k=1}^{K}p(X=k)\log p(X=k)-\sum_{k=1}^{K}p(X=k)\log q(X=k)=-H(p)+H(p,q)$$

Where $H(p,q)=\sum_k P(x=k)\log q(X=k)$; this expression is known as **cross entropy**.

(*) KL divergence can be interpreted as the average number of extra bits needed to encode the data – due to the fact that we used distribution q to encode the data instead of the true distribution p; note that KL divergence is *not symmetric*!

(*) The "extra number of bits" interpretation should make it clear that $KL(p||q)\geq 0$.

KL Divergence: Information Inequality

$$KL\big(p(X)\|q(X)\big)=\sum_{k=1}^{K}p(X=k)\log\frac{p(X=k)}{q(X=k)}$$

(*) KL divergence can be interpreted as the average number of extra bits needed to encode the data – due to the fact that we used distribution q to encode the data instead of the true distribution p. The "extra number of bits" interpretation should make it clear that $KL(p||q) \geq 0$.

Theorem (*Information inequality*): $\mathrm{KL}(p||q) \geq 0 \; and \; KL = 0 \; iff \; p = q$.

KL Divergence: Information Inequality

$$KL\big(p(X)\|q(X)\big)=\sum_{k=1}^{K}p(X=k)\log\frac{p(X=k)}{q(X=k)}$$

(*) KL divergence can be interpreted as the average number of extra bits needed to encode the data – due to the fact that we used distribution q to encode the data instead of the true distribution p. The "extra number of bits" interpretation should make it clear that $KL(p||q) \geq 0$.

Theorem (*Information inequality*): $\mathrm{KL}(p||q) \geq 0$ *and* $KL = 0$ *iff* $p = q$.

Pf.

$$-KL\big(p\|q\big)=-\sum_{k}p\big(X=k\big)\log\frac{p\big(X=k\big)}{q\big(X=k\big)}$$

KL Divergence: Information Inequality

$$KL\big(p(X)\,\|\,q(X)\big)=\sum_{k=1}^{K}p(X=k)\log\frac{p(X=k)}{q(X=k)}$$

(*) KL divergence can be interpreted as the average number of extra bits needed to encode the data – due to the fact that we used distribution q to encode the data instead of the true distribution p. The "extra number of bits" interpretation should make it clear that $KL(p||q) \geq 0$.

Theorem (*Information inequality*): $\mathrm{KL}(p||q) \geq 0$ *and* $KL = 0$ *iff* $p = q$.

Pf.

$$-KL\big(p\,\|\,q\big)=-\sum_{k}p(X=k)\log\frac{p(X=k)}{q(X=k)}=\sum_{k}p(X=k)\log\frac{q(X=k)}{p(X=k)}$$

Why?

KL Divergence: Information Inequality

$$KL\big(p(X)\,\|\,q(X)\big)=\sum_{k=1}^{K}p(X=k)\log\frac{p(X=k)}{q(X=k)}$$

(*) KL divergence can be interpreted as the average number of extra bits needed to encode the data – due to the fact that we used distribution q to encode the data instead of the true distribution p. The "extra number of bits" interpretation should make it clear that $KL(p||q) \geq 0$.

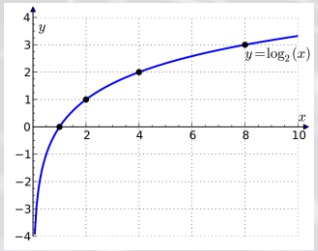
Theorem (*Information inequality*): $\mathrm{KL}(p||q) \geq 0$ *and* $KL = 0$ *iff* $p = q$.

Pf.

$$-KL\big(p\,\|\,q\big)=-\sum_{k}p(X=k)\log\frac{p(X=k)}{q(X=k)}=\sum_{k}p(X=k)\log\frac{q(X=k)}{p(X=k)}$$

$$\leq\log\sum_{k}p(X=k)\frac{q(X=k)}{p(X=k)}$$

This follows because log is a **convex function**, i.e. $\log(x) \leq x$



KL Divergence: Information Inequality

$$KL\big(p(X)\,\|\,q(X)\big)=\sum_{k=1}^{K}p(X=k)\log\frac{p(X=k)}{q(X=k)}$$

(*) KL divergence can be interpreted as the average number of extra bits needed to encode the data – due to the fact that we used distribution q to encode the data instead of the true distribution p. The "extra number of bits" interpretation should make it clear that $KL(p||q)\geq 0$.

Theorem (*Information inequality*): $\mathrm{KL}(p||q)\geq 0$ *and* $KL=0$ *iff* $p=q$.

Pf.

$$-KL\big(p\,\|\,q\big)=-\sum_{k}p(X=k)\log\frac{p(X=k)}{q(X=k)}=\sum_{k}p(X=k)\log\frac{q(X=k)}{p(X=k)}$$

$$\leq\log\sum_{k}p(X=k)\frac{q(X=k)}{p(X=k)}=\log\sum_{k}q(X=k)=\log 1=0$$

Why?

In summary: $\mathrm{KL}(p||q)\geq 0$, as was to be shown.

KL Divergence: Information Inequality

(*) One important consequence of the information inequality (that we alluded to previously) is that the discrete distribution with maximum entropy is the uniform distribution. More precisely, $H(X) \leq \log |K|$, where K is the number of states for the random variable X, with equality holding iff p(x) is uniform.

KL Divergence: Information Inequality

(*) One important consequence of the information inequality (that we alluded to previously) is that the discrete distribution with maximum entropy is the uniform distribution. More precisely, $H(X) \leq \log |K|$, where K is the number of states for the random variable X, with equality holding iff p(x) is uniform.

Pf. Consider any generic discrete probability distribution p(x), and let u(x)=1/K, the uniform distribution on K states.

$$0 \leq KL(p \| u)$$

Why?

KL Divergence: Information Inequality

(*) One important consequence of the information inequality (that we alluded to previously) is that the discrete distribution with maximum entropy is the uniform distribution. More precisely, $H(X) \leq \log |K|$, where K is the number of states for the random variable X, with equality holding iff p(x) is uniform.

Pf. Consider any generic discrete probability distribution p(x), and let u(x)=1/K, the uniform distribution on K states.

$$0 \leq KL(p \| u) = \sum_k p(X=k) \log \frac{p(X=k)}{u(X=k)} = \sum_k p(X=k) \log p(X=k) - \sum_k p(X=k) \log u(X=k)$$

KL Divergence: Information Inequality

(*) One important consequence of the information inequality (that we alluded to previously) is that the discrete distribution with maximum entropy is the uniform distribution. More precisely, $H(X) \leq \log |K|$, where K is the number of states for the random variable X, with equality holding iff p(x) is uniform.

Pf. Consider any generic discrete probability distribution p(x), and let u(x)=1/K, the uniform distribution on K states.

$$0 \leq KL(p \| u) = \sum_k p(X=k) \log \frac{p(X=k)}{u(X=k)} = \sum_k p(X=k) \log p(X=k) - \sum_k p(X=k) \log u(X=k)$$

$$= -H(X) - \sum_k p(X=k) \log u(X=k)$$

KL Divergence: Information Inequality

(*) One important consequence of the information inequality (that we alluded to previously) is that the discrete distribution with maximum entropy is the uniform distribution. More precisely, $H(X) \leq \log K$, where K is the number of states for the random variable X, with equality holding iff p(x) is uniform.

Pf. Consider any generic discrete probability distribution p(x), and let u(x)=1/K, the uniform distribution on K states.

$$0 \leq KL(p \| u) = \sum_k p(X=k) \log \frac{p(X=k)}{u(X=k)} = \sum_k p(X=k) \log p(X=k) - \sum_k p(X=k) \log u(X=k)$$

$$= -H(X) - \sum_k p(X=k) \log u(X=k) = -H(X) - \sum_k p(X=k) \log(1/K)$$

Why?

KL Divergence: Information Inequality

(*) One important consequence of the information inequality (that we alluded to previously) is that the discrete distribution with maximum entropy is the uniform distribution. More precisely, $H(X) \leq \log K$, where K is the number of states for the random variable X, with equality holding iff p(x) is uniform.

Pf. Consider any generic discrete probability distribution p(x), and let u(x)=1/K, the uniform distribution on K states.

$$0 \leq KL(p \| u) = \sum_k p(X=k) \log \frac{p(X=k)}{u(X=k)} = \sum_k p(X=k) \log p(X=k) - \sum_k p(X=k) \log u(X=k)$$

$$= -H(X) - \sum_k p(X=k) \log u(X=k) = -H(X) - \sum_k p(X=k) \log(1/K)$$

$$= -H(X) - (\log 1/K) \sum_k p(X=k)$$

Why?

KL Divergence: Information Inequality

(*) One important consequence of the information inequality (that we alluded to previously) is that the discrete distribution with maximum entropy is the uniform distribution. More precisely, $H(X) \leq \log K$, where K is the number of states for the random variable X, with equality holding iff p(x) is uniform.

Pf. Consider any generic discrete probability distribution p(x), and let u(x)=1/K, the uniform distribution on K states.

$$0 \leq KL(p \| u) = \sum_k p(X=k) \log \frac{p(X=k)}{u(X=k)} = \sum_k p(X=k) \log p(X=k) - \sum_k p(X=k) \log u(X=k)$$

$$= -H(X) - \sum_k p(X=k) \log u(X=k) = -H(X) - \sum_k p(X=k) \log(1/K)$$

$$= -H(X) - (\log 1/K) \sum_k p(X=k) = -H(X) + (\log K) \sum_k p(X=k)$$

Why?

KL Divergence: Information Inequality

(*) One important consequence of the information inequality (that we alluded to previously) is that the discrete distribution with maximum entropy is the uniform distribution. More precisely, $H(X) \leq \log K$, where K is the number of states for the random variable X, with equality holding iff p(x) is uniform.

Pf. Consider any generic discrete probability distribution p(x), and let u(x)=1/K, the uniform distribution on K states.

$$0 \leq KL(p \| u) = \sum_k p(X=k) \log \frac{p(X=k)}{u(X=k)} = \sum_k p(X=k) \log p(X=k) - \sum_k p(X=k) \log u(X=k)$$

$$= -H(X) - \sum_k p(X=k) \log u(X=k) = -H(X) - \sum_k p(X=k) \log(1/K)$$

$$= -H(X) - (\log 1/K) \sum_k p(X=k) = -H(X) + (\log K) \sum_k p(X=k) = -H(X) + \log K$$

Why?

KL Divergence: Information Inequality

(*) One important consequence of the information inequality (that we alluded to previously) is that the discrete distribution with maximum entropy is the uniform distribution. More precisely, H(X) ≤ log K, where K is the number of states for the random variable X, with equality holding iff p(x) is uniform.

Pf. Consider any generic discrete probability distribution p(x), and let u(x)=1/K, the uniform distribution on K states.

$$0 \le KL(p\|u) = \sum_k p(X=k)\log\frac{p(X=k)}{u(X=k)} = \sum_k p(X=k)\log p(X=k) - \sum_k p(X=k)\log u(X=k)$$

$$= -H(X) - \sum_k p(X=k)\log u(X=k) = -H(X) - \sum_k p(X=k)\log(1/K)$$

$$= -H(X) - (\log 1/K)\sum_k p(X=k) = -H(X) + (\log K)\sum_k p(X=k) = -H(X) + \log K$$

(*) **In summary**: H(X) ≤ log K, where K is the number of states for the random variable X, with equality holding iff p(x) is uniform; thus the discrete distribution with maximum entropy is the uniform distribution, as was to be shown.

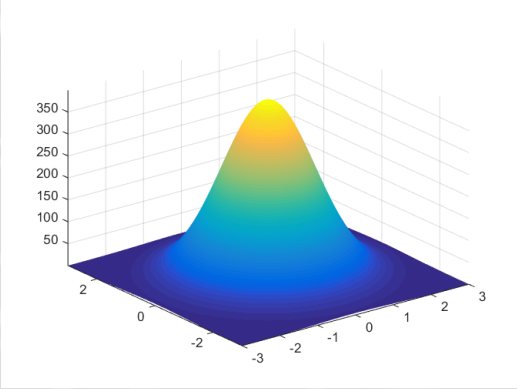
KL Divergence: Information Inequality

• We've demonstrated, using the information inequality, that the discrete distribution with maximum entropy is the uniform distribution.

• This is a formulation of Laplace's **Principle of Insufficient Reason** (PIR) which argues in favor of using uniform distributions when there are no other reasons to favor one distribution over another.

• In Bayesian learning, we typically want a distribution that satisfies certain constraints but is otherwise as least-commital as possible (for example we might prefer to choose priors with maximum entropy).

(*) Among all real-valued distributions with a specified variance (i.e. second moment) the Gaussian distribution has maximum entropy.



Mutual Information

• Consider two random variables, X and Y. Suppose we want to know how much knowing one variable tells us about the other. We could compute a quantity such a correlation, however, this is a very limited measure of dependence.

• A more general approach is to determine how similar the joint distribution p(X,Y) is to the factored distribution p(X)p(Y). This leads to the definition of **mutual information**:

$$I(X,Y) = KL\big(p(X,Y) \,\|\, p(X)p(Y)\big) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

(*) It follows that $I(X,Y) \geq 0$ with equality iff $p(X,Y) = p(X)p(Y)$, meaning that mutual information equals zero iff X and Y are independent (Note that, by contrast, correlation between X and Y can be zero, even when X and Y are *dependent*).

Mutual Information

$$I(X,Y) = KL\left(p(X,Y) \,\|\, p(X)p(Y)\right) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

(*) It follows that I(X,Y) $\geq$ 0 with equality iff p(X,Y) = p(X)p(Y), meaning that mutual information equals zero iff X and Y are independent (Note that, by contrast, correlation between X and Y can be zero, even when X and Y are *dependent*).

• Intuitively, mutual information measures the information that X and Y share: It measures how much knowing one of these variables reduces uncertainty about the other.

For example, if X and Y are independent, then knowing X does not give any information about Y and vice versa, so their mutual information is zero.

At the other extreme, if X is a deterministic function of Y and Y is a deterministic function of X then all information conveyed by X is shared with Y: knowing X determines the value of Y and vice versa. As a result, in this case the mutual information is the same as the uncertainty contained in Y (or X) alone, namely the entropy of Y (or X).

Mutual Information

$$I(X,Y) = KL\left(p(X,Y) \,\|\, p(X)p(Y)\right) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

(*) One can show that mutual information (MI) is equivalent to:

$$I(X,Y) = H(X) - H(X \mid Y) = H(Y) - H(Y \mid X)$$

Where H(Y | X) is the **conditional entropy** of Y given X, which is the average uncertainty in the value of Y after X is observed (H(X | Y) is, similarly, the average uncertainty in value of X after Y is observed).

Mutual Information

$$I(X,Y) = KL\left(p(X,Y) \,\|\, p(X)p(Y)\right) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

(*) One can show that mutual information (MI) is equivalent to:

$$I(X,Y) = H(X) - H(X \mid Y) = H(Y) - H(Y \mid X)$$

Where $H(Y \mid X)$ is the **conditional entropy** of Y given X, which is the average uncertainty in the value of Y after X is observed ($H(X \mid Y)$ is, similarly, the average uncertainty in value of X after Y is observed).

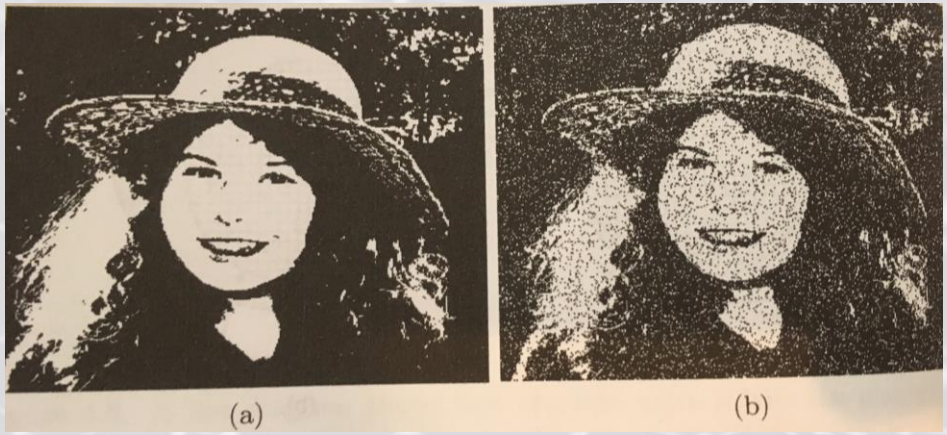
Consequently, MI between X and Y can be interpreted as the reduction in uncertainty about X after observing Y, or by symmetry, the reduction in uncertainty about Y after observing X.

(*) One can show that I(X,Y)=H(X)+H(Y)-H(X,Y)

Mutual Information: Example of Coding Efficiency

• Consider an example of a binary image X (shown on the left) transmitted using a noisy channel, so that each pixel value has a 10% probability of being flipped resulting in distorted image Y (on right).

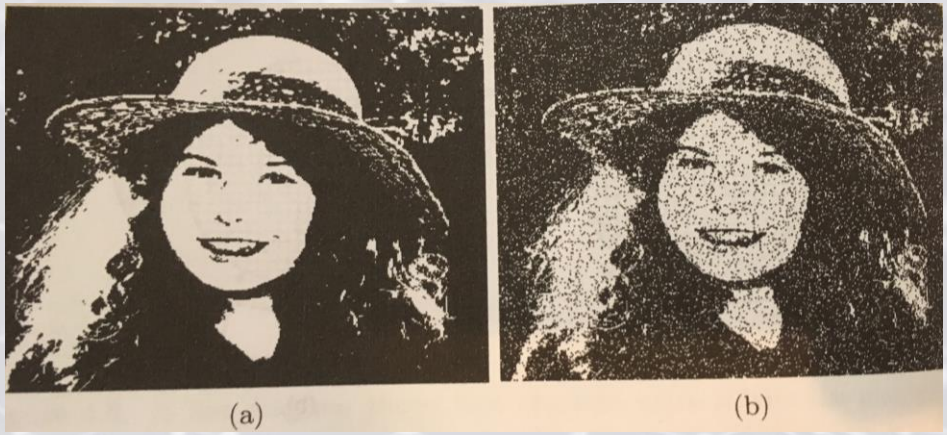
Fist, we'll compute I(X,Y) and then quantify the transmission efficiency; recall that I(X,Y)=H(X)+H(Y)-H(X,Y).



Mutual Information: Example of Coding Efficiency

• Consider an example of a binary image X (shown on the left) transmitted using a noisy channel, so that each pixel value has a 10% probability of being flipped resulting in distorted image Y (on right).

Fist, we'll compute I(X,Y) and then quantify the transmission efficiency; recall that I(X,Y)=H(X)+H(Y)-H(X,Y).

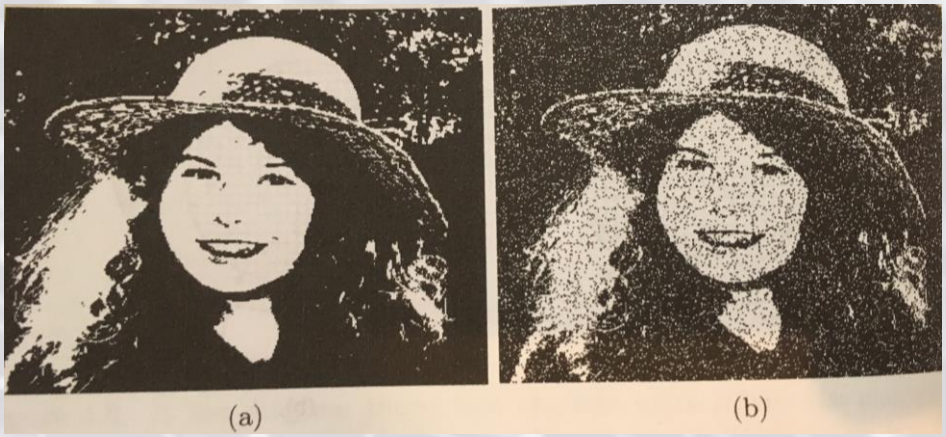


(*) In the original image, 0.724 of the pixels are black (0) and 0.276 are white (1), so the entropy is:

H(X)=p(0)log(1/p(0))+p(1)log(1/p(1))=0.851 bits/pixel. (in truth this is an overestimate of the entropy, because we ignored adjacency relationships).

Mutual Information: Example of Coding Efficiency

Fist, we'll compute I(X,Y) and then quantify the transmission efficiency; recall that I(X,Y)=H(X)+H(Y)-H(X,Y).



(*) In the original image, 0.724 of the pixels are black (0) and 0.276 are white (1), so the entropy is:

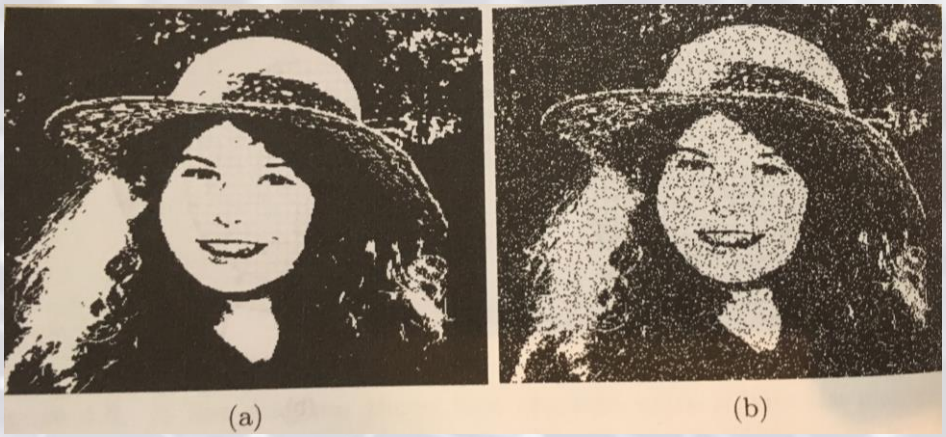
H(X)=p(0)log(1/p(0))+p(1)log(1/p(1))=0.851 bits/pixel. (in truth this is an overestimate of the entropy, because we ignored adjacency relationships).

(*) In the corrupted image, a proportion 0.679 of the pixels are black and 0.322 are white (again, we ignore correlations between neighboring pixel values).

H(Y)= p(0)log(1/p(0))+p(1)log(1/p(1))=0.906 bits/pixel.

Mutual Information: Example of Coding Efficiency

Fist, we'll compute I(X,Y) and then quantify the transmission efficiency; recall that I(X,Y)=H(X)+H(Y)-H(X,Y).



H(X)= 0.851 bits/pixel.

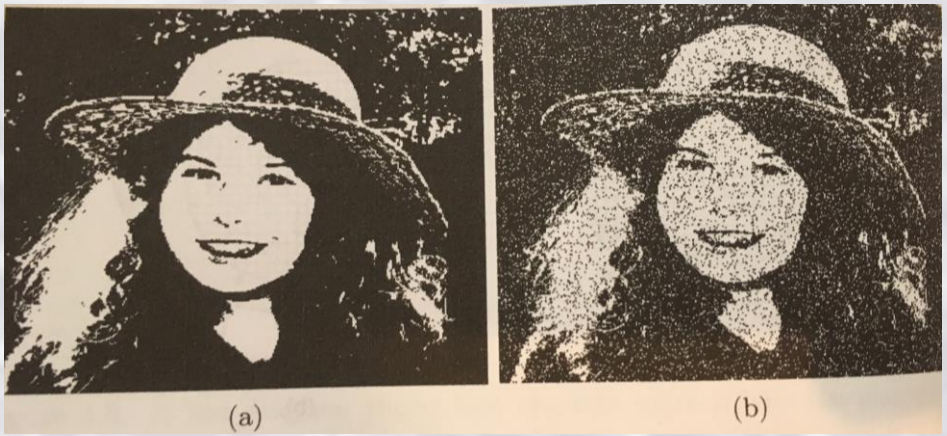
H(Y)= 0.906 bits/pixel.

Computing the joint entropy: H(X,Y)=p(0,0)log(1/p(0,0))+…+p(1,1)log(1/p(1,1))=1.32 bits/pixel.

I(X,Y)=H(X)+H(Y)-H(X,Y)=0.851+0.906-1.32=0.436 bits.

This means that each value of the output Y reduces our uncertainty about the corresponding value of the input X by about half a bit.

Mutual Information: Example of Coding Efficiency



H(X)= 0.851 bits/pixel; H(Y)= 0.906 bits/pixel; H(X,Y)=1.32 bits/pixel.

I(X,Y)=H(X)+H(Y)-H(X,Y)=0.851+0.906-1.32=0.436 bits.

This means that each value of the output Y reduces our uncertainty about the corresponding value of the input X by about half a bit.

Lastly, one can quantify the *transmission efficiency* by computing the ratio:

I(X,Y)/H(Y) = 0.436/0.906 = **0.481**.

(*) This implies that almost half of the entropy of the output depends on the input, and the remainder is due to noise in the channel.

Shannon's Noisy Channel Coding Theorem

• The most general definition of channel capacity for any channel is:

$$capacity = \max_{p(X)} I(X,Y) \ bits$$

This states that the channel capacity is achieved by the distribution p(X) which makes the mutual information I(X,Y) between the input and output as large as possible.

• Using conditional entropy, as previously stated, we can rewrite this equation as:

$$capacity = \max_{p(X)} H(X) - H(X \mid Y) \ bits$$

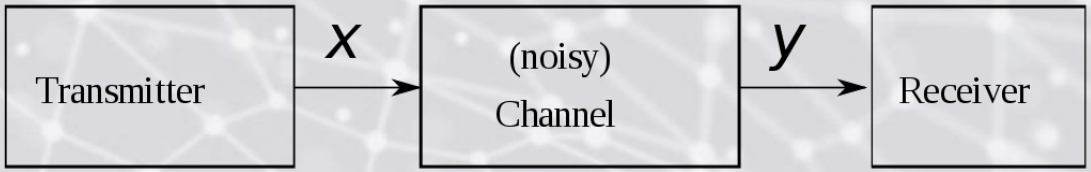
Note that if there is no noise in the channel, then H(X|Y)=0 (why?), in which case the channel capacity formula which reduces to the definition of channel capacity provided earlier for the case with no noise.

Shannon's Noisy Channel Coding Theorem

• The most general definition of channel capacity for any channel is:

$$capacity = \max_{p(X)} I(X,Y) \ bits$$

This states that the channel capacity is achieved by the distribution p(X) which makes the mutual information I(X,Y) between the input and output as large as possible.



(*) Shannon's noisy channel coding theorem states (paraphrasing): it is possible to use a communication channel to communicate information with a low error rate, at a rate arbitrarily close to the channel capacity, but it is not possible to communication information at a rate greater than the channel capacity.

Fin