

Information Theory and Statistics

Mark Hansen and Bin Yu

June 1, 2005, Graybill Conference, Fort Collins

Outline

Morning: Entropy

- Lecture 1: History of entropy, codes and probability distributions, entropy as coding limit
- Lecture 2: Entropy properties and examples, maximum entropy principle, and estimation of entropy
- Lab Session
 - Decoding Braille text
 - Compressing Lawrence's Sons and Lovers by designing your own code
 - Plotting entropy function
 - Coding by symbols vs in blocks and corresponding entropy rates
 - Reproducing sequence logo for donor data
 - Which stimulus drives neuron cells better in song birds?

Entropy: physics origin

1850: Idea of entropy by Clausius, but not the term

1865: Entropy first appeared. First and second laws of Thermodynamics by Clausius using entropy

$$\Delta S = \Delta Q/T,$$

1877: Boltzmann quantifies entropy of an equilibrium thermodynamic system as

$$S = K \log W,$$

S - energy, K - Boltzmann constant, W - number of microstates in the system.

1870's: Gibbs gives a general entropy expression for a thermodynamic system:

$$H = - \sum_j p_j \log p_j,$$

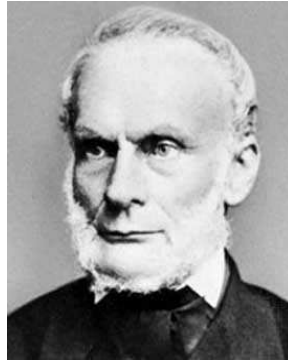
where p_j is the probability that the system is at microstate j .

If $p_j = 1/W$, then the two definitions agree.

Rudolf Julius Emmanuel Clausius

Born: 2 Jan 1822 in Koslin, Prussia (now Koszalin, Poland)

Died: 24 Aug 1888 in Bonn, Germany



Clausius was a theoretical physicist who played an important role in establishing theoretical physics as a discipline. His most famous paper was read to the Berlin Academy on 18 February 1850 and published in *Annalen der Physik* in the same year. This paper marks the foundation of the modern thermodynamics. In his paper of 1865 Clausius stated the First and Second laws of thermodynamics in the following form.

1. The energy of the universe is constant.
2. The entropy of the universe tends to a maximum.

Ludwig Boltzmann

Born: 20 Feb 1844 in Vienna, Austria

Died: 5 Oct 1906 in Duino (near Trieste), Austria (now Italy)

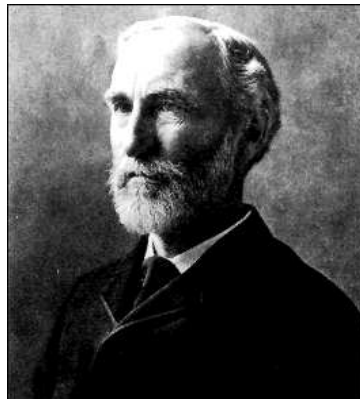


A theoretical physicist at Vienna (and Graz), Boltzmann's fame is based on his invention of statistical mechanics. This he did independently of Willard Gibbs. Their theories connected the properties and behaviour of atoms and molecules with the large scale properties and behaviour of the substances of which they were the building blocks.

Willard Gibbs

Born: 11 Feb 1839 in New Haven, Connecticut, USA

Died: 28 April 1903 in New Haven, Connecticut, USA

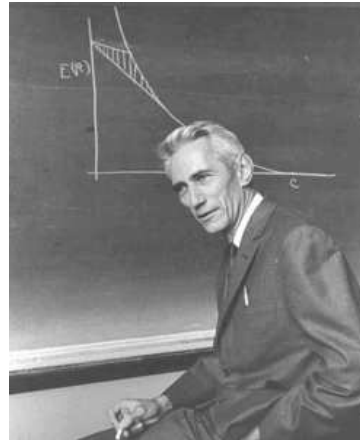


A Europe-trained mathematical physicist at Yale College. His work on statistical mechanics provided a mathematical framework for quantum theory and for Maxwell's theories. His last publication, *Elementary Principles in Statistical Mechanics*, beautifully lays a firm foundation for statistical mechanics.

Claude Elwood Shannon's entropy in communication theory

Born in Gaylord, Michigan, on April 30, 1916

Died on Feb. 26, 2001.



Shannon is considered as the founding father of electronic communications age. His work on technical and engineering problems within the communications industry laid the groundwork for both the computer industry and telecommunications.

Shannon (1948): A Mathematical Theory of Communication

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

In Shannon's theory, a message is a random draw from a probability distribution on messages and entropy gives the data compression (source coding) limit.

Shannon's information theory

Shannon's entropy measures "information" content in a message, but this "information" is not the meaningful information. It is simply the uncertainty in the message just as Boltzmann-Gibbs entropy measures the disorder in a thermodynamic system.

Shannon's theory concerns with point-to-point communications as in telephony and gives limits on coding. It consists of:

1. Source coding: limits on data storage (transmission over noiseless channels) (our text files are stored in binary format or through a binary code on our computers)
2. Channel coding: limits on data transmission over noisy channels.

What is a code?

Given a discrete *alphabet* \mathcal{X} of message symbols, a binary *code* is a mapping from *symbols* in \mathcal{X} to a set of *codewords* of binary strings.

\mathcal{X} could be the Roman letters, numbers, and other writing symbols.

Example: $\mathcal{X} = \{a, b, c\}$. Here is one binary code:

$$\begin{aligned} a &\rightarrow 00 \\ b &\rightarrow 01 \\ c &\rightarrow 10 \end{aligned} \tag{1}$$

$aabacbcbaa \rightarrow 00000100100110010000$

$bcccbabcca \rightarrow 01101010010001101000.$

Each requiring 20 bits (binary digits by Tukey).

What is decoding?

Decoding involves splitting the encoded string into segments or sub-strings of 0's and 1's, and then performing a table lookup to see which symbol is associated with each segment.

Another code:

$$\begin{array}{l} a \rightarrow 0 \\ b \rightarrow 10 \\ c \rightarrow 11 \end{array} \quad (2)$$

$aabacbcbaa \rightarrow 001001110111000$

$bcccbabcca \rightarrow 101111111001011110.$

This is a *prefix* code; that is, no codeword is the prefix of another.

This property means that encoded messages are *uniquely decodable*, even if we don't include separation commas between codewords.

Examples of real codes: ASCII and Unicode

a. **ASCII**- the American Standard Code for Information Interchange:

7 bit binary code for the alphabet set of Roman letters, digits, mathematical symbols, and punctuations.

Used for storing and transmitting English documents.

Each symbol is mapped into a digit between $0 - 127 = 2^7 - 1$ and then this number can be coded with 7 bits.

b. **Unicode** is a 16-bit code that assigns a unique number to every character or symbol in use, from Bengali to Braille. $2^{16} = 65,000$ codewords. They are fixed length codes.

Examples of real codes: Morse code

Original Morse code: \mathcal{X} contains dots, dashes, and pauses.

Modern version by A. Vail: \mathcal{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.

Examples of real codes: Braille






























































L. Braille (1829): 6-bit code allows 63 possible codes (flat or raised dot).

26 are used to encode Roman letters, the rest to encode common words (and, for, of, the, with) and common two letter combinations (ch, gh, sh, th, wh, ed, er, ou, ow).



Unified English Braille Code (UEBC), 1992: unite separate Braille codes for mathematics, scientific notation and computer symbols.

THE ALPHABET ONE FEELS WITH THE FINGERS TO READ

 A 1	 B 2	 C 3	 D 4	 E 5	 F 6	 G 7	 H 8	 I 9	 J 0
 K	 L	 M	 N	 O	 P	 Q	 R	 S	 T
 U	 V	 X	 Y	 Z	 and	 for	 of	 the	 with
 ch	 gh	 sh	 th	 wh	 ed	 er	 ou	 ow	 w
 .	 ;	 :	 .	 en	 !	 ()	 "?	 in	 ""
 st	 ing	 or	 ar	 .	 -				
 general accent sign	 used for two-called contractions		 italic sign; decimal point	 letter sign	 capital sign				

WHEN THE FIRST 10 LETTERS ARE PRECEDED BY THE NUMERIC INDICATOR (1-0) THE SIGNS HAVE NUMBER VALUE.

Direct use of frequent structures leads to better compression.

Codes and Probability Distributions

Given a binary code on \mathcal{X} , the *length function* L maps symbols in \mathcal{X} to the length of their codeword in bits. Using the code in (2), we have $L(a) = 1$ and $L(b) = L(c) = 2$.

In general, there is a correspondence between the length function of a prefix code and the quantity $-\log_2 Q$ for a probability distribution Q defined on \mathcal{X} .

Kraft's inequality:

For any binary prefix code, the code length function L must satisfy the inequality

$$\sum_{x \in \mathcal{X}} 2^{-L(x)} \leq 1. \quad (3)$$

onversely, given a set of codeword lengths that satisfy this inequality, there exists a prefix binary code with these code lengths.

Proof: Use the one-one mapping of a binary prefix code and a binary tree with codewords only on the end-nodes.

Prefix codes and probability distributions are equivalent

Using the Kraft inequality, we can take any length function L and construct a distribution as follows

$$Q(x) = \frac{2^{-L(x)}}{\sum_{x \in \mathcal{X}} 2^{-L(x)}} \quad \text{for any } x \in \mathcal{X}. \quad (4)$$

Conversely, for any distribution Q on \mathcal{X} and any $x \in \mathcal{X}$, we can find a prefix code with length function

$$L(x) = \lceil -\log Q(x) \rceil,$$

the smallest integer greater than or equal to $-\log Q(x)$, because

$$L(x) \geq -\log Q(x), \text{ hence } 2^{-L(x)} \leq Q(x),$$

and it follows that

$$\sum 2^{-L(x)} \leq \sum Q(x) = 1.$$

Making frequency counting precise (Shannon, 1948)

Suppose our messages are constructed by randomly selecting elements of \mathcal{X} according to a distribution P . Then, the *expected length* of a code is given by

$$L = \sum_{x \in \mathcal{X}} P(x)L(x). \quad (5)$$

Gibbs' inequality:

Suppose the elements of \mathcal{X} are generated according to a probability distribution P . For any prefix code on \mathcal{X} with length function $L(\cdot)$, the expected code length L is bounded below

$$L \geq - \sum_{x \in \mathcal{X}} P(x) \log_2 P(x) = H(P) \quad (6)$$

where equality holds if and only if $L = -\log_2 P$.

Proof of Gibbs' inequality

By Kraft's inequality, $C_L = \sum_x 2^{-L(x)} \leq 1$.

Then

$$Q(x) = 2^{-L(x)} / C_L.$$

is a probability distribution.

Since $E_P L = E_P[-\log(Q(X)) - \log C_L] \geq E_P[-\log(Q(X))]$,

$$EP_L - H(P) = -E_P \log \frac{Q(X)}{P(X)} \geq^* -\log E_P \frac{Q(X)}{P(X)} = -\log 1 = 0.$$

\geq^* holds because of Jensen's inequality applied to the convex function \log :
for $Y = Q(X)/P(X)$

$$(E \log Y) \leq \log EY, \quad \text{which implies } -(E \log Y) \geq -\log EY.$$

Formal Definition of Entropy

Given a probability function P defined on a discrete alphabet \mathcal{X} , we define the entropy $H(P)$ to be

$$H(P) = - \sum_{x \in \mathcal{X}} P(x) \log P(x) = -E_P \log P(X). \quad (7)$$

The logarithm in this expression is usually in base 2, and the units of entropy are referred to as *bits* (coined by Tukey).

Coding algorithms when P is known

Given P on $\mathcal{X} = \{x_1, \dots, x_k\}$, let code length function be

$$L^*(x) = \lceil -\log P(x) \rceil.$$

$$\sum_{x \in \mathcal{X}} 2^{-\lceil -\log P(x) \rceil} \leq \sum_{x \in \mathcal{X}} 2^{\log P(x)} = \sum_{x \in \mathcal{X}} P(x) = 1. \quad (8)$$

Therefore, by Kraft's inequality L^* corresponds to prefix code. Since the ceiling operator introduces an error of at most one bit,

$$H(P) \leq EL^* \leq H(P) + 1 \quad (9)$$

from Gibbs' inequality.

Shannon code with known P

Suppose $P(x_1) \geq P(x_2) \geq \dots \geq P(x_k)$, and let

$$F_i = \sum_{j=1}^{i-1} P(x_j),$$

which is the CDF function at x_{i-1} .

Shannon code:

$x_i \rightarrow F_i$ rounded to $\lceil -\log P(x_i) \rceil$ bits.

Obviously this code has code length $L^*(x_i) = \lceil -\log P(x_i) \rceil$.

Shannon code: example

Let P be a distribution on $\{a, b, c\}$ with probability $11/20, 1/4, 1/5$. Its entropy $H(P) = 1.439$ bits. Then

$$F_1 = 0, F_2 = 11/20, F_3 = 4/5,$$

$$\lceil -\log P_1 \rceil = \lceil 0.86 \rceil = 1; \lceil -\log P_2 \rceil = \lceil 2 \rceil = 2; \lceil -\log P_3 \rceil = \lceil 2.3 \rceil = 3$$

Rounding F_i 's to the right number of bits:

$$\text{round}(F_1) = 0, \text{round}(F_2) = 10, \text{round}(F_3) = 110,$$

because

$$F_1 = 0, F_2 = 11/20 = 1/2 + 1/20; F_3 = 4/5 = 1/2 + 1/4 + 1/20.$$

Expected code length $L = 11/20 + 2 \times 1/4 + 3 \times 1/5 = 8/5 = 1.6$ bits.

which is within 1 bit of the entropy 1.439 bits.

Huffman code with known P

Shannon's code achieves entropy when it is applied to n-blocks of messages. The optimal coding question remained open for a finite alphabet. Huffman(1952) solved this problem by designing the now so-called Huffman code, which constructs in a bottom-up and greedy fashion.

"Growing" the coding tree from the end-nodes:

for a P with $P(a) = 1/2$, and $P(b) = P(c) = 1/4$

- find the two elements b and c with the smallest probabilities and connect them with leaves 0 and 1 to form the intermediate node bc and assign it probability the sum of $P(b)$ and $P(c)$.
- iterate the process until no element is left.

Huffman code is OPTIMAL because it gives the prefix code with the smallest expected code length.

Arithmetic code with known P

Both Shannon and Huffman codes require sorting, expensive for large alphabets (large blocks) and not so easy to update when the block size changes.

Shannon-Fano-Elias is a modification on Shannon code:

Let

$$\bar{F}(x_i) = \sum_{j=1}^{i-1} P(x_j) + \frac{1}{2}P(x_i).$$

Map:

$x_i \rightarrow \bar{F}(x_i)$ rounded to $\lceil -\log P(x_i) \rceil + 1$ bits.

This code is easily updated when more symbols come along. It acquired a new name *Arithmetic Code*, when applied to blocks.

Shannon code reaches entropy limit for iid data almost surely

Asymptotic Equal Partition (AEP)

Suppose X_1, \dots, X_n iid, then average code length for the Shannon code on n -tuples goes to the entropy almost surely.

$$\frac{1}{n} L^*(X_1, \dots, X_n) \rightarrow H(P).$$

Proof:

$$L^*(X_1, \dots, X_n)/n = \log P(X^n)/n + O(1/n).$$

LLN applied to $\log P(X^n)/n = \sum \log P(X_i)/n$.

n-block codes achieve entropy rate!

Shannon-McMillian-Breiman theorem: AEP for stationary and ergodic processes.

Uniform distribution in n-tuple space

In the space of n-tuple sequences $\{1, \dots, k\}^n$,

$$P(X^n) \approx 2^{-nH(P)}$$

for "most" sequences.

This gives an approximate uniform distribution and its entropy is the log of the number of possibilities, that is,

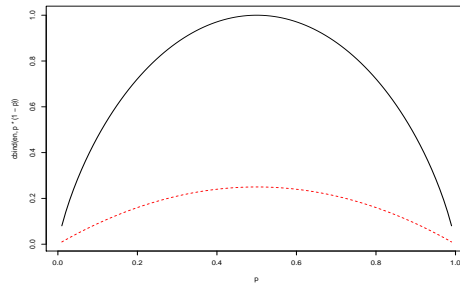
$$\log 1/2^{-nH(P)} = nH(P).$$

So Boltzman's definition of entropy in the uniform case can be seen the general one in this sense.

Examples of entropy

If X is Bernoulli(p), then

$$H(p) = -p \log p - (1 - p) \log(1 - p), \quad \text{while } V(x) = p(1 - p).$$



Both are maximized at $p = 1/2$. And $H(p)$ is not too flat near the maximum at $p = 1/2$:

$$H''(p) = -\frac{1}{p(1-p)} = -4 \quad \text{at } p = 1/2.$$

Examples of entropy

Binomial (n, p)

$$H(X) = nH(p)$$

Poisson (μ) :

$$H(X) = \mu + \log \mu + E \log X!$$

Geometric (p) :

$$H(X) = H(p)/p$$

Multinomial $(n; p_1, \dots, p_k)$:

$$n \sum_{j=1}^k -p_j \log p_j$$

Properties of Entropy

P is a distribution on $\{x_1, \dots, x_k\}$.

1. $H(P) \geq 0$

2. $H_b(P) = (\log_b a)H_a(P)$

3. $H(X|Y) \leq H(X)$ with equality only if X and Y are independent, where

$$H(X|Y) = \sum_y P(Y = y)H(X|Y = y).$$

4. $H(X_1, \dots, X_n) \leq \sum_{i=1}^n H(X_i)$ with equality only if the X_i are independent.

5. $H(P) \leq \log k$ with equality only if P is uniform on \mathcal{X} .

6. $H(P)$ is concave in P .

A sketchy proof:

1. Obvious because $-\log P(x) \geq 0$.
2. Obvious.
3. Intuitively obvious because conditioning reduces uncertainty.

$$H(X) - H(X|Y) = I(X, Y) = E \log(P(X, Y)/P(X)P(Y)) \geq 0$$

by Gibbs' inequality

4. Use (3) and chain rule:

$$H(X_1, \dots, X_n) = \sum_{i=1}^n H(X_i | X_{i-1}, \dots, X_1).$$

5. because $H(P)$ is maximized when $P(x) = 1/K$. That is, the solution of $\max H(P)$ subject to $\sum_x P(x) = 1$ is $P(x) = 1/K$ – uniform is the maxent distribution with no constraints.
6. it follows from the convexity of \log or the concavity of $-\log$.

Entropy rate for stationary sequences

For a stationary ergodic sequence X_1, \dots, X_n, \dots , we can define its entropy rate as

$$H(X_1, \dots, X_n, \dots) = \lim_{n \rightarrow \infty} H(X_1, \dots, X_n)/n,$$

which is equivalent to

$$\lim_{n \rightarrow \infty} H(X_n | X_{n-1}, \dots, X_1).$$

Entropy rate of a Markov sequence

For a stationary and ergodic Markov chain with stationary distribution π_i and transition matrix p_{ij} ,

The entropy rate is

$$\sum_i \pi_i \sum_j [-p_{ij} \log p_{ij}].$$

For a two-state Markov chain with a transition Matrix

$$\begin{pmatrix} 1 - p_1 & p_1 \\ p_2 & 1 - p_2 \end{pmatrix}$$

and stationary distribution

$$\pi_1 = \frac{p_2}{p_1 + p_2}; \pi_2 = \frac{p_1}{p_1 + p_2}.$$

The entropy of X_n is

$$H(X_n) = H\left(\frac{p_1}{p_1 + p_2}\right).$$

However, the entropy rate of the sequence is LOWER due to the dependence and it is

$$H(X_2|X_1) = \frac{p_2}{p_1 + p_2} H(p_1) + \frac{p_1}{p_1 + p_2} H(p_2)$$

For low flip rates $p_1 = 0.1$, $p_2 = 0.02$, the marginal entropy is

$$H(1/3) = 0.92(\text{bits}),$$

while the entropy rate of the sequence is

$$\frac{2}{3} \times H(0.02) + \frac{1}{3} \times H(0.01) = 0.14/3 + 2 \times 0.08/3 = 0.12(\text{bits}).$$

Entropy estimation in a bioinformatics example

Motifs: chromosome regions with specific biological structural significance or function. Usually short: 6-20 base pairs. Examples: splice sites, transcription factor binding sites, translation initiation sites, enhancers, silencers.

The table below is a weight matrix learned from 15,155 mamalian donor sites (exon and intron junctions) from the SpliceDB database. Entries are frequencies of bases at each position.

Base	-3	-2	-1	0	+1	+2	+3	+4	+5
A	33	61	10	0	0	53	71	7	16
C	37	13	3	0	0	3	8	6	16
G	18	12	80	100	0	42	12	81	22
T	12	14	7	0	100	2	9	6	46

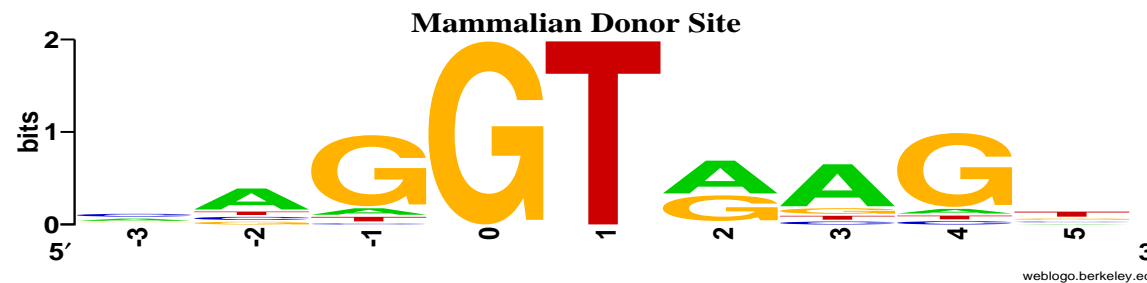
Sequence logo

A graphical method to display patterns in a set of aligned sequences:

- Height of stack at each position is the "information" content from the frequencies:

$$\text{max. entropy} - \text{estimated entropy} = 2 - \text{estimated entropy}$$

- Letters (A, T, G, C) are arranged in decreasing order of frequency whose heights are proportional to the frequencies.



The entropy estimate is the plug-in estimate.

Thanks to Xiaoyue Zhao and Terry Speed for providing the data.

Plug-in estimate of entropy

Given an iid sequence X_1, \dots, X_n with probabilities p_1, \dots, p_k on $\{1, \dots, k\}$,

$N_j = \sum_i I(X_i = j)$ for $j = 1, \dots, k$ are multinomial, and MLE of p 's are

$$\hat{p}_j = N_j/n.$$

Then the plug-in MLE of $H(X)$ is

$$\hat{H} = H(\hat{p}_1, \dots, \hat{p}_k) = - \sum_{j=1}^k \frac{N_j}{n} \log \frac{N_j}{n}.$$

Downward bias of plug-in estimate of entropy

Miller (1954) showed that

$$H - \hat{H} = \sum_j \frac{N_j}{n} \log \frac{N_j}{np_j} + \sum_j \left\{ \frac{N_j}{n} - p_j \right\} \log p_j$$

Bias

$$E(H - \hat{H}) = E\left(\sum_j \frac{N_j}{n} \log \frac{N_j}{np_j}\right),$$

because the second term has expectation zero.

$2 \sum_j N_j \log \frac{N_j}{np_j}$ has an approximate χ_{k-1}^2 distribution; hence

$$E(H - \hat{H}) \approx (k - 1)/(2n) + O(1/n^2).$$

The $1/n^2$ term is actually

$$\left(\sum \frac{1}{p_j} - 1\right)/(12n^2).$$

Limiting Distributions of \hat{H}

From Miller's expansion, we can easily see that when X is NOT uniform,

$$\sqrt{n}(\hat{H} - H) \rightarrow N(0, \sigma_H^2),$$

where $\sigma_H^2 = -\sum_{j \neq j'} p_j p_{j'} \log p_j \log p_{j'} + \sum_j p_j (1 - p_j) (\log p_j)^2$.

When X is uniform, a faster convergence rate holds:

$$n(\hat{H} - H) \rightarrow \frac{1}{2} \chi_{k-1}^2.$$

Differential entropy

If X is a continuous random variable with probability density function $f(x)$, then

$$H(X) = \int f(x) \log f(x) dx = E(-\log f(X))$$

If one discretize X with precision $\delta > 0$ into a discrete variable X_δ ,

$$H(X_\delta) \approx H(X) - \log \delta.$$

Hence differential entropy could be negative.

For Gaussian $N(\mu, \sigma^2)$,

$$H(X) = \frac{1}{2} \log(2e\pi\sigma^2).$$

For exponential (λ)

$$H(X) = \log[e/\lambda].$$

Downward bias of MLE plug in of differential entropy

Suppose $f(x^n, \theta)$ is a parametric n -tuple density function of a stationary and ergodic sequence, $\hat{\theta}$ is the MLE of θ .

Then the MLE plug-in estimate of differential entropy rate

$$\hat{H}_n(f) = -\log f(x^n, \hat{\theta})/n,$$

underestimates $H(f_n) = E[-\log f(X^n, \theta)/n]$ under regularity conditions and the expected bias is

$$d/n,$$

where d is the dimension of θ , because

$$-\log f(x^n, \theta) + \log f(x^n, \hat{\theta}) \approx \frac{1}{2}\chi_d^2.$$

Jaynes' Maximum Entropy (Maxent) Principle



In his famous 1957 paper ("information theory and statistical mechanics"), Ed. T. Jaynes wrote:

Information theory provides a constructive criterion for setting up probability distributions on the basis of partial knowledge, and leads to a type of statistical inference which is called the maximum entropy estimate. It is least biased estimate possible on the given information; i.e., it is maximally noncommittal with regard to missing information.

That is to say, when characterizing some unknown events with a statistical model, we should always choose the one that has Maximum Entropy.

Applications: computer vision, spatial physics, natural language processing.

Examples of Maxent distributions

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 maxent distribution is $N(\mu, \sigma^2)$ 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 $\{1, \dots, k\}$ is the maxent distribution with no moment constraints.

Maxent continued

Example 4 (Boltzmann) The maxent distribution on a finite set $\{1, \dots, k\}$ with a first moment constraint $\alpha_1 > 0$ is

$$p_j = e^{\lambda j} / \sum_{j=1}^k e^{\lambda j}.$$

That is, the most probable "macrostate" (probability distribution) is (p_1, \dots, p_k) as above, provided that

$$\sum_j j p_j = \alpha_1$$

is fixed.

Maxent distributions are in the exponential family

Maxent problem:

Maximize the entropy of f over all probability density functions such that

- $f(x) \geq 0$
- $\int f(x)dx = 1$
- $\int f(x)T_i(x)dx = \alpha_i$ for $i = 1, \dots, m$

The maxent solution takes the form

$$f(x) = e^{\lambda_0 - 1 + \sum_{i=1}^m \lambda_i T_i(x)},$$

where the λ 's are chosen to satisfy the constraints.

A derivation through calculus

Let

$$J(f) = - \int f \log f + \lambda_0 \int f + \sum_i \lambda_i \int f T_i.$$

Differentiate with respect to $f(x)$:

$$\frac{\partial J}{\partial f(x)} = -\log f(x) - 1 + \lambda_0 + \sum_i \lambda_i T_i(x).$$

Setting this to zero, we get

$$f(x) = e^{\lambda_0 - 1 + \sum_{i=1}^m \lambda_i T_i(x)},$$

where the λ 's are chosen to satisfy the constraints.

Strong et al's method of entropy estimation in neuroscience

For a window size T , take non-overlapping windows and estimate the joint probabilities of T -tuples and plug in these empirical joint probabilities to get entropy estimate \hat{H}_T for window size T . Stationarity is implicitly assumed.

For a sequence of size n with enough mixing, one could generalize Miller's result to show that the bias of Strong et al's estimate is of order

$$O(2^T/n),$$

which follows that when $T = O(\log n)$ the bias is of order $O(1)$.

Summary of Morning Lectures

- Entropy history in physics
- Shannon's entropy and codes
- Kraft's inequality connects prefix codes with probability distributions
- Entropy gives the coding limit through block codes
- Properties and examples of entropy
- Maxent principle
- Estimation of entropy – plug in estimate under-estimate entropy

Lab Problem I: Decoding Braille

Using the code book given, decode the Braille text into English.

a	b	c	d	e	f	g	h	i	j

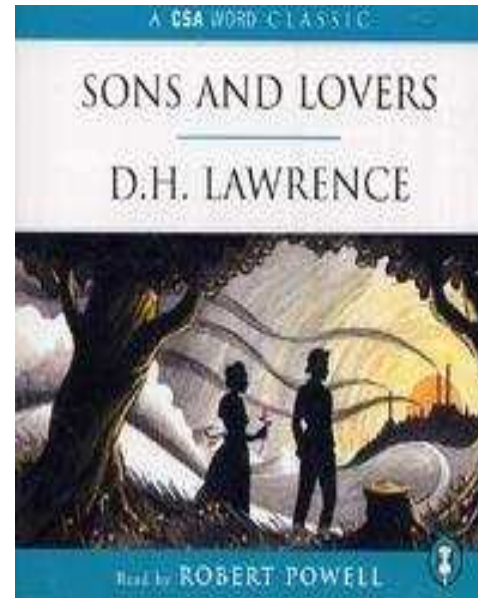
k	l	m	n	o	p	q	r	s	t

u	v	w	x	y	z

Text to be decoded:

u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j

Lab Problem II: Compressing letters in "Sons and Lovers"



Lab Problem II: Compressing letters in "Sons and Lovers"

Letter percentages (%) in decreasing order

e	t	a	h	o	i	s	n	r
12.95	8.59	7.84	7.59	7.21	6.67	6.52	6.44	5.59
d	l	w	u	m	y	g	f	c
4.75	4.47	2.85	2.72	2.68	2.17	2.13	1.99	1.94
b	p	k	v	x	j	q	z	
1.39	1.39	1.07	0.76	0.10	0.08	0.08	0.03	

Design a Shannon Code from these percentages and calculate the average code length per letter and compare with the estimated entropy rate.

Lab Problem III: compare Huffman code on symbols with Huffman code on 3-tuples (blocks)

Given an iid binary message source with probability 0.1, 0.9, design the Huffman code on $\{0, 1\}$ and the Huffman code on $\{0, 1\}^3$.

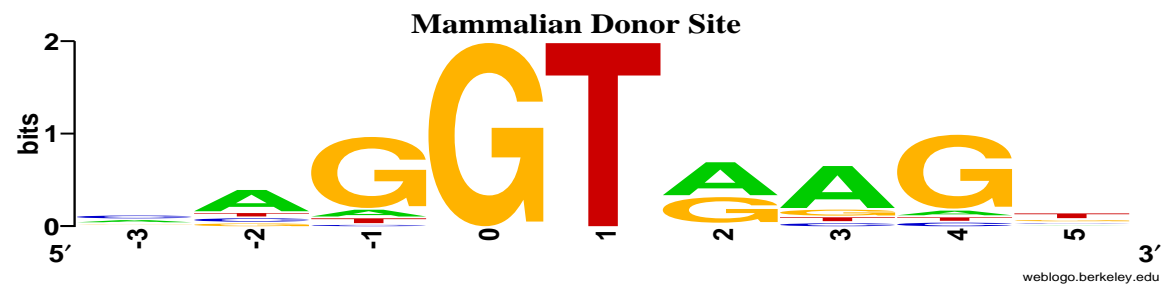
Calculate the average code lengths and compare with the entropy rates.

Lab Problem VI: entropy function plotting

Make a 3-dim plot of the entropy function of a trinomial distribution (p_1, p_2, p_3) as a function of the first two probabilities.

Lab Problem V: Reproducing the logo plot

Go to weblogo.berkeley.edu and read the information at the website and upload file "donor9_15155.q" to reproduce:



Lab Problem VI: natural vs synthetic stimuli to song birds

Spike trains (0-1 sequence) are recorded on a single neuron in the auditory pathway of a song bird while sound stimuli are played to the bird.

Sequence 1 ("natural" file):

Natural stimulus: bird song.

Sequence 2 ("synthetic" file):

Synthetic stimulus: man-made songs matching some power-spectrum characteristics of a bird song.

Use the Strong et al method to estimate the entropy rates of these two sequences and what hypothesis do you want to put forward based on these estimates?

In R, type `source("strong.r")` and `r.x=entropyrate(x,maxwordlen=L)` gives the entropy rates for $T=1, \dots, L$ estimated from x .

Data provided by F. Theunissen Lab, UC Berkeley