CHAPTER 6

Shannon entropy

This chapter is a digression in **information theory**. This is a fascinating subject, which arose once the notion of information got precise and *quantifyable*. From a physical point of view, information theory has nothing to do with physics. However, the concept of Shanon entropy shares some intuition with Boltzmann's, and some of the mathematics developed in information theory turns out to have relevance in statistical mechanics. This chapter contains very basic material on information theory.

1. Quantifying information

Computers have popularized the notion of *bit*, a unit of information that takes two values, 0 or 1. We introduce the **information size** $H_0(A)$ of a set A as the number of bits that is necessary to encode each element of A separately, i.e.

$$H_0(A) = \log_2 |A|. \tag{6.1}$$

This quantity has a unit, the bit. If we have two sets A and B, then

$$H_0(A \times B) = H_0(A) + H_0(B). \tag{6.2}$$

This justifies the logarithm. The information size of a set is not necessarily integer. If we need to encode the elements of A, the number of necessary bits is $\lceil H_0(A) \rceil$ rather than $H_0(A)$; but this is irrelevant for the theory.

The ideas above are natural and anybody might have invented the concept of information size. The next notion, the *information gain*, is intuitive but needed a genius to define and quantify it.

Suppose you need to uncover a certain English word of five letters. You manage to obtain one letter, namely an e. This is useful information, but the letter e is common in English, so this provides little information. If, on the other hand, the letter that you discover is j (the least common in English), the search has been more narrowed and you have obtained more information. The information gain quantifies this heuristics.

We need to introduce a relevant formalism. Let $\boldsymbol{A} = (A, p)$ be a discrete probability space. That is, $A = \{a_1, \dots, a_n\}$ is a finite set, and each element has probability p_i. (The σ-algebra is the set of all subsets of A.) The information gain $G(B|A)$ measures the gain obtained by the knowledge that the outcome belongs to the set $B \subset A$. We denote $p(B) = \sum_{i \in B} p_i$.

Definition 6.1. *The* **information gain** *is*

$$G(B|A) = \log_2 \frac{1}{p(B)} = -\log_2 p(B).$$

The information gain is positive, and it satisfies the following additivity property. Let $B \subset C \subset A$. The gain for knowing that the outcome is in C is $G(C|A) = -\log_2 p(C)$. The gain for knowing that it is in B, after knowing that it is in C, is

$$G(B|C) = -\log_2 p(B|C) = -\log \frac{p(B)}{p(C)}. \tag{6.3}$$

It follows that $G(B|A) = G(C|A) + G(B|C)$, as it should be.

The unit for the information gain is the bit. We gain 1 bit if $p(B) = \frac{1}{2}$.

2. Shannon entropy

It is named after Shannon[1], although its origin goes back to Pauli and von Neumann.

Definition 6.2. *The* **Shannon entropy** *of* $\boldsymbol{A}$ *is*

$$\boxed{H(\boldsymbol{A}) = -\sum_{i=1}^{n} p_i \log_2 p_i.}$$

The extension to continuum probability spaces is not straightforward and we do not discuss it here.

Proposition 6.1. $H(\boldsymbol{A}) \leqslant \log_2 n$, *with equality iff* $p(a_i) = 1/n$ *for all* i.

Proof. Since $-\log_2$ is convex, we have from Jensen's inequality

$$-\log_2 \Bigl(\underbrace{\sum_{i=1}^{n} \frac{1}{p_i} p_i}_{n} \Bigr) \leqslant \sum_{i=1}^{n} \Bigl(-\log_2 \frac{1}{p_i} \Bigr) p_i = -H(\boldsymbol{A}). \tag{6.4}$$

Since $-\log_2$ is strictly convex, the inequality above is strict unless $p_1 = \dots = p_n$. □

Proposition 6.2. *H is strictly concave with respect to p. That is, writing $H(p)$ instead of $H(\boldsymbol{A} = (A,p))$, we have*

$$H(\alpha p^{(1)} + (1-\alpha) p^{(2)}) \geqslant \alpha H(p^{(1)}) + (1-\alpha) H(p^{(2)}).$$

Proof. The space of probabilities on A is the convex set

$$P = \{(p_1, \dots, p_n) : 0 \leqslant p_i \leqslant 1, \sum p_i = 1\}. \tag{6.5}$$

(It is actually a simplex.) Given p in the interior of P, let $q = (q_1, \dots, q_n)$ such that $\sum q_i = 0$, and such that $p + \lambda q \in P$ provided λ is a number small enough. Then

$$H(p + \lambda q) = -\sum_{i=1}^{n} (p_i + \lambda q_i) \log_2 (p_i + \lambda q_i). \tag{6.6}$$

The derivatives with respect to λ are

$$\frac{\mathrm{d}H}{\mathrm{d}\lambda} = -\sum_{i=1}^{n} q_i \log_2 (p_i + \lambda q_i), \qquad \frac{\mathrm{d}^2 H}{\mathrm{d}\lambda^2} = -\frac{1}{\log 2} \sum_{i=1}^{n} \frac{q_i^2}{p_1 + \lambda q_i}. \tag{6.7}$$

The latter is strictly negative. □

[1] The American Claude Shannon (1916–2001) wrote *A mathematical theory of communication* in 1948, an article that created information theory.

Definition 6.3. *The* **relative entropy** *of the probability p with respect to the probability q is*

$$H(p|q) = \sum_{i=1}^{n} p_i \log_2 \frac{p_i}{q_i}.$$

The definition is not symmetric under exchanges of p and q, $H(p|q) \neq H(q|p)$ unless $p = q$.

Proposition 6.3.

$$H(p|q) \geqslant 0,$$

with equality iff $p = q$.

Proof. By Jensen,

$$H(p|q) = -\sum_i p_i \log_2 \frac{q_i}{p_i} \geqslant -\log_2 \Bigl(\sum_i p_i \frac{q_i}{p_i} \Bigr) = 0. \tag{6.8}$$

□

3. Relation with Boltzmann entropy

Statistical mechanics provides three probability measures on the phase space, the microcanonical, canonical, and grand-canonical measures. We now study the Shannon entropy of these measures; as it turns, it coincides with the thermodynamic entropy. For simplicity we consider the Ising model in the lattice gas interpretation, but the present discussion is clearly more general. Recall that the domain $D \subset \mathbb{Z}^d$ is discrete, and that the state space is $\Omega = \{0,1\}^D$. The Hamiltonian $\Omega \to \mathbb{R}$ involves a sum over nearest neighbours.

Microcanonical ensemble. The probability is uniform over all states with energy U and number of particles N:

$$p_{\rm micro}(\omega) = \begin{cases} \frac{1}{X(U,D,N)} & \text{if } H(\omega) = U \text{ and } N(\omega) = N, \\ 0 & \text{otherwise.} \end{cases}$$

Then $H(p_{\rm micro}) = \log_2 X(U,D,N) = \frac{1}{k_{\rm B} \log 2} S(U,D,N)$.

Canonical ensemble. We now have the Gibbs factor.

$$p_{\rm can}(\omega) = \begin{cases} \frac{{\rm e}^{-\beta H(\omega)}}{Y(\beta,D,N)} & \text{if } N(\omega) = N, \\ 0 & \text{otherwise.} \end{cases}$$

One easily obtains

$$H(p_{\rm can}) = \frac{1}{\log 2} \bigl[\beta \langle H(\omega) \rangle + \log Y(\beta,D,N) \bigr].$$

The average of the Hamiltonian is equal to the thermodynamic energy $U(\beta,D,N)$. The logarithm of Y is equal to $-\beta F(\beta,D,N)$. Recall that the free energy is related to the energy and the entropy by $U = F - TS$. With $\beta = 1/k_{\rm B}T$, we see that $H(p_{\rm can}) = \frac{1}{k_{\rm B} \log 2} S(\beta,D,N)$.

Grand-canonical ensemble. The probability $p_{\mathrm{gd-can}}$ involves now the chemical potential. As is checked in the exercises, we have $H(p_{\mathrm{gd-can}}) = \frac{1}{k_{\mathrm{B}} \log 2} S(\beta, D, \mu)$.

A thorough rigorous discussion of the relations between Shannon and thermodynamic entropies, and of the theory of large deviations can be found in [Pfister, 2002][2]

4. Shannon theorem

A basic problem in information theory deals with encoding large quantities of information. We start with a finite set A, that can denote the 26 letters from the Latin alphabet, or the 128 ASCII symbols, or a larger set of words. We consider a file that contains $N|A|$ symbols with N large. How many bits are required so that the file can be encoded without loss of information? The answer is given by the information size, $H_0(A^N) = NH_0(A)$.

The question becomes more interesting, and the answer more surprising, if we allow an error δ. We now seek to encode only files that fall in a set $B \subset A$, such that $p(B) \geqslant 1 - \delta$. If a file turns out to be in $A \setminus B$, then we lose the information. The information size is given by $H_\delta(A)$, where

$$H_\delta(A) = \inf_{\substack{B \subset A \\ p(B) \geqslant 1-\delta}} \log_2 |B|. \tag{6.9}$$

Notice that $\lim_{\delta \to 0} H_\delta(A) = H_0(A)$.

The occurrence of probabilities may be confusing, so a discussion is needed. In the real world, information is selected for a purpose and its content is well-chosen. But we modelize the problem of information transmission using probabilities; the process can be described as follows. A string of N characters is selected at random using some probability. One wants to encode this string, to send it (or to store it), and to decode it. We assume that no error is committed during these operations, except that the string may lie outside the set of codable strings. This modelization is behind all compression algorithms. Strange as it may seem, an MP3 file of *Manon Lescaut* has been compressed as if the music was the the result of random composing by Puccini, random performing by the orchestra, and random singing by the soli!

So we want to say something about $H_\delta(\boldsymbol{A}^N)$ (the probability of $(a_1, \dots, a_N) \in A^N$ is $\prod p(a_i)$, i.e. we assume independence). Notice that $H_\delta(\boldsymbol{A}) \leqslant H_0(A)$, and also that $H(\boldsymbol{A}) \leqslant H_0(A)$, with equality iff elements of A come with equal probability. We have $H_0(A^N) = NH_0(A)$, but $H_\delta(\boldsymbol{A}^N)$ is smaller than $NH_\delta(\boldsymbol{A})$ in general.

Theorem I (Shannon source coding theorem). *For any $\delta > 0$,*

$$\lim_{N \to \infty} \frac{1}{N} H_\delta(\boldsymbol{A}^N) = H(\boldsymbol{A}).$$

The theorem says that if we allow for a tiny error, and if our message is large (depending on the error), the number of required bits is roughly $NH(\boldsymbol{A})$. Notice that the limit in the theorem is a true limit, not a lim inf or a lim sup. Thus Shannon entropy gives the optimal compression rate, that can be approached but not improved.

[2] Ch.-É. Pfister, *Thermodynamical aspects of classical lattice systems*, in *In and out of equilibrium: Physics with a probability flavor*, Progr. Probab. 51, Birkhäuser (2002)

Proof. It is based on the (weak) law of large numbers. Consider the random variable $-\log_2 p(a)$. The law of large numbers states that, for any $\varepsilon > 0$,

$$\lim_{N\to\infty} \operatorname{Prob}\Bigl(\Bigl\{(a_1,\dots,a_N) : \Bigl| \underbrace{-\frac{1}{N}\sum_{i=1}^{N} \log_2 p(a_i)}_{-\frac{1}{N}\log_2 p(a_1,\dots,a_N)} - \underbrace{E\bigl(-\log_2 p(a)\bigr)}_{H(\boldsymbol{A})} \Bigr| > \varepsilon\Bigr\}\Bigr) = 0. \tag{6.10}$$

There exists therefore a set $A_{N,\varepsilon} \subset A^N$ such that $\lim_N p(A_{N,\varepsilon}) = 1$, and such that any $(a_1,\dots,a_N) \in A_{N,\varepsilon}$ satisfies

$$2^{-N(H(\boldsymbol{A})+\varepsilon)} \leqslant p(a_1,\dots,a_N) \leqslant 2^{-N(H(\boldsymbol{A})-\varepsilon)}. \tag{6.11}$$

The number of elements of $A_{N,\varepsilon}$ is easily estimated:

$$1 \geqslant p(A_{N,\varepsilon}) \geqslant |A_{N,\varepsilon}|\, 2^{-N(H(\boldsymbol{A})+\varepsilon)}, \tag{6.12}$$

so that $|A_{N,\varepsilon}| \leqslant 2^{N(H(\boldsymbol{A})+\varepsilon)}$. For any $\delta > 0$, we can choose N large enough so that $p(A_{N,\varepsilon}) > 1-\delta$. Then

$$H_\delta(\boldsymbol{A}^N) \leqslant \log_2 |A_{N,\varepsilon}| \leqslant N(H(\boldsymbol{A})+\varepsilon). \tag{6.13}$$

It follows that

$$\limsup_{N\to\infty} \frac{1}{N} H_\delta(\boldsymbol{A}^N) \leqslant H(\boldsymbol{A}). \tag{6.14}$$

For the lower bound, let $B_{N,\delta}$ be the minimizer for H_δ; that is, $p(B_{N,\delta}) \geqslant 1-\delta$, and

$$H_\delta(\boldsymbol{A}^N) = \log_2 |B_{N,\delta}| \geqslant \log_2 \bigl|B_{N,\delta} \cap A_{N,\varepsilon}\bigr|. \tag{6.15}$$

We need a lower bound for the latter term.

$$1-\delta \leqslant \underbrace{p\bigl(B_{N,\delta} \cap A_{N,\varepsilon}\bigr)}_{\leqslant |B_{N,\delta}\cap A_{N,\varepsilon}| 2^{-N(H(\boldsymbol{A})-\varepsilon)}} + \underbrace{p\bigl(B_{N,\delta} \cap A^{\mathrm{c}}_{N,\varepsilon}\bigr)}_{\leqslant \delta \text{ if } N \text{ large}}. \tag{6.16}$$

Then

$$\bigl|B_{N,\delta} \cap A_{N,\varepsilon}\bigr| \geqslant (1-2\delta)\, 2^{N(H(\boldsymbol{A})-\varepsilon)}. \tag{6.17}$$

We obtain

$$\frac{1}{N} H_\delta(\boldsymbol{A}^N) \geqslant \frac{1}{N}\log_2(1-2\delta) + H(\boldsymbol{A}) - \varepsilon. \tag{6.18}$$

This gives the desired bound for the lim inf, and Shannon's theorem follows. □

It is instructive to quantify the ε's and δ's of the proof. Invoking the central limit theorem instead of the law of large numbers, we get that

$$p(A_{N,\varepsilon}) \approx 2\sigma^2 \int_{\sqrt{N}\varepsilon}^{\infty} \mathrm{e}^{-\frac{1}{2}t^2}\, \mathrm{d}t \approx \mathrm{e}^{-N\varepsilon^2} \approx \delta.$$

(σ^2 is the variance of the random variable $-\log_2 p(a)$.) This shows that $\varepsilon \approx N^{-\frac{1}{2}}$ and $\delta \approx \mathrm{e}^{-N}$. It is surprising that δ can be so tiny, and yet makes such a difference!

5. Codes and compression algorithms

A code is a mapping from a "string" (a finite sequence of letters) to a finite sequence of binary numbers. The coding of a string is sequential, i.e. letters are coded one by one, independently of one another. We need some notation. A binary sequence, or sequence of bits, is written $b = b_1 b_2 \dots b_m$, where each $b_i = 0, 1$. The *concatenation* of two binary sequences b and b', of respective lengths m and n, is the sequence

$$bb' = b_1 \dots b_m b'_1 \dots b'_n.$$

Of course, the length of bb' is $m+n$. Let $\mathbb{B}$ be the set of all finite binary sequences, of arbitrary length. We want to encode a string $(a_1, \dots, a_n)$ of elements of an "alphabet" A.

Definition 6.4. *A* **code** *is a map c from $\cup_{N \geqslant 1} A^N$ to $\mathbb{B}$, which satisfies the following sequential property:*

$$c(a_1, \dots, a_n) = c(a_1) \dots c(a_n).$$

A code is **uniquely decodable** *if the mapping is injective (one-to-one).*

Some codes map any letter $a \in A$ to a binary sequence with fixed length. For instance, ASCII characters use 7 bits for any letter. But compression algorithms use variable length codes. A special class of variable length codes are **prefix codes**. These are uniquely decodable codes where a sequence of binary numbers can be decoded sequentially. That is, one reads the binary numbers from the left, one by one, until one recognizes the code of a letter. One extracts the letter, and reads the next binary numbers until the next letter is identified. These notions are best understood with the help of examples.

Let $A = \{a_1, a_2, a_3, a_4\}$. The code $c^{(1)}$ is undecodable; $c^{(2)}$ is uniquely decodable but is not a prefix code; and $c^{(3)}$ and $c^{(4)}$ are prefix codes.

$$\begin{array}{ll|ll|ll|ll}
c^{(1)}: & a_1 \mapsto 0 & c^{(2)}: & a_1 \mapsto 0 & c^{(3)}: & a_1 \mapsto 00 & c^{(4)}: & a_1 \mapsto 0 \\
 & a_2 \mapsto 1 & & a_2 \mapsto 01 & & a_2 \mapsto 01 & & a_2 \mapsto 10 \\
 & a_3 \mapsto 00 & & a_3 \mapsto 011 & & a_3 \mapsto 10 & & a_3 \mapsto 110 \\
 & a_4 \mapsto 11 & & a_4 \mapsto 0111 & & a_4 \mapsto 11 & & a_4 \mapsto 111
\end{array}$$

Any fixed length, uniquely decodable code is also prefix. Prefix codes can be represented by trees, see Fig. 6.1.

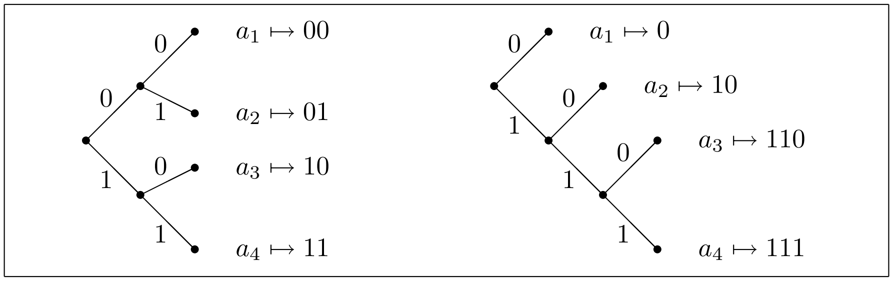

Figure 6.1. Tree representations for the prefix codes $c^{(3)}$ and $c^{(4)}$.

The goal of compression algorithms[3] is to encode strings with the smallest sequence of binary numbers. For $a \in A$, let $\ell(a)$ be the length of the sequence $c(a)$. If each letter a occurs with (independent) probability $p(a)$, the expectation for the length of one letter is

$$L(\boldsymbol{A}, c) = \sum_{a \in A} p(a)\ell(a). \tag{6.19}$$

Thus the goal is to find the code that minimizes $\mathbb{E}(\ell)$. It is instructive to consider the following two situations with $A = \{a_1, a_2, a_3, a_4\}$, and the two prefix codes above.

1. If $p_1 = p_2 = p_3 = p_4 = \frac{1}{4}$, the expected length for $c^{(3)}$ is 2 bits, and for $c^{(4)}$ it is 2.25 bits. The first code is better. Notice that $H(\boldsymbol{A}) = 2$ bits.
2. If $p_1 = \frac{1}{2}$, $p_2 = \frac{1}{4}$, $p_3 = p_4 = \frac{1}{8}$, the expected length for $c^{(3)}$ is still 2 bits, but it is now 1.75 bits for $c^{(4)}$. The second code is better. Notice that $H(\boldsymbol{A}) = 1.75$ bits.

As we will see, the codes are optimal for these two situations.

The idea is that *frequent letters should be coded with smaller length.* This clearly comes at the expense of other letters, that will need longer strings in order for the code to be decodable. There is a bound on minimally achievable lengths, that does not involve probabilities. It is known as **Kraft inequality**.

Proposition 6.4 (Kraft inequality).

- *Any one-to-one code on A satisfies*

$$\sum_{a \in A} 2^{-\ell(a)} \leqslant 1.$$

- *Given $\{\ell(a) : a \in A\}$ satisfying the inequality above, there corresponds a prefix code.*

Proof. We need to prove the first statement for any decodable code, not necessarily a prefix code. We start with

$$\begin{aligned} \Big(\sum_{a \in A} 2^{-\ell(a)} \Big)^N &= \sum_{a_1, \dots, a_N} 2^{-\ell(a_1) - \dots - \ell(a_N)} \\ &= \sum_{L = N\ell_{\min}}^{N\ell_{\max}} 2^{-L} \# \big\{ (a_1, \dots, a_N) : \ell\big(c(a_1, \dots, a_N)\big) = L \big\}. \end{aligned} \tag{6.20}$$

We set $\ell_{\min} = \min_{a \in A} \ell(a)$, and similarly for $\ell_{\max}$. Since the code is one-to-one, the number $\#\{\cdot\}$ above is no more than 2^L. Then

$$\Big(\sum_{a \in A} 2^{-\ell(a)} \Big)^N \leqslant \sum_{L = N\ell_{\min}}^{N\ell_{\max}} 2^{-L} 2^L = N(\ell_{\max} - \ell_{\min} + 1). \tag{6.21}$$

Since the right side grows like N, the left side cannot grow exponentially with N; it must be less or equal to 1.

The second claim can be proved e.g. by suggesting an explicit construction for the prefix code, given lengths that satisfy Kraft inequality. It is left as an exercise. □

[3] The word "algorithm" derives from Abu Ja'far Muhammad ibn Musa Al-Khwarizmi, who was born in Baghdad around 780, and who died around 850.

Shannon entropy again appears as a limit for data compression.

Theorem II (Limit to compression). *For any alphabet A, and any probability p on A, the optimal prefix code c satisfies*

$$H(\boldsymbol{A}) \leqslant L(\boldsymbol{A}, c) \leqslant H(\boldsymbol{A}) + 1.$$

Proof. For the lower bound, consider a code c with lengths $\ell_i = \ell(c(a_i))$. Define $q_i = \frac{2^{-\ell_i}}{z}$, with $z = \sum_j 2^{-\ell_j}$. We have

$$L(\boldsymbol{A}, c) = \sum_{i=1}^{n} p_i \ell_i = -\sum_{i=1}^{n} p_i \log_2 q_i - \log_2 z \geqslant -\sum_{i=1}^{n} p_i \log_2 p_i = H(\boldsymbol{A}). \tag{6.22}$$

The inequality holds because of positivity of the relative entropy, Proposition 6.3, and because $z \leqslant 1$ (Kraft inequality).

For the upper bound, define $\ell_i = \lceil -\log_2 p_i \rceil$ (the integer immediately bigger than $-\log_2 p_i$). Then

$$\sum_{i=1}^{n} 2^{-\ell_i} \leqslant \sum_{i=1}^{n} p_i = 1. \tag{6.23}$$

This shows that Kraft inequality is verified, so there exists a prefix code c with these lengths. The expected length is easily estimated,

$$L(\boldsymbol{A}, c) = \sum_{i=1}^{n} p_i \lceil -\log_2 p_i \rceil \leqslant \sum_{i=1}^{n} p_i (-\log_2 p_i + 1) = H(\boldsymbol{A}) + 1. \tag{6.24}$$

□

Exercise 6.1. State and prove Jensen's inequality.

Exercise 6.2. Check that Shannon's entropy of the grand-canonical probability is equal to the corresponding entropy. Consider a discrete statistical mechanics model such as the lattice gas with nearest-neighbour interactions (Ising model).

Exercise 6.3. Recall the definitions for the codes $c^{(1)}$, $c^{(2)}$, $c^{(3)}$, and $c^{(4)}$. Explain why $c^{(1)}$ is undecodable; $c^{(2)}$ is uniquely decodable but not prefix; $c^{(3)}$ and $c^{(4)}$ are prefix codes.

Exercise 6.4. Given lengths $\{\ell(a)\}$ satisfying Kraft inequality, show the existence of a corresponding prefix code.