

With n bits, it's possible to specify 2^n different numbers, or possibilities (1 bit gives 2, 2 bits give 4, 3 bits give 8, and so on). In the opposite direction, a number N of possibilities is specified by $\log_2 N$ bits of information. For 10 possibilities that would be roughly 3.32 bits of information (i.e., between 3 and 4 since 10 is between 8 and 16). Below, we shall see how to interpret these non-integer values.

We're already familiar with using the binomial coefficient $\binom{N}{m} = N!/m!(N-m)!$, the number of ways to choose m distinct items from a total of N . Further subdividing, there are $\binom{m}{m_1}$ ways to split the m items into m_1 of type 1 and $m_2 = m - m_1$ of type 2, so the total number of ways of choosing the m_1 and m_2 items is $\binom{N}{m} \binom{m}{m_1} = N!/m_1!m_2!m_3!$, where $m_3 = N - (m_1 + m_2)$. Continuing this process leads to the multinomial coefficient,

$$\binom{N}{m_1, m_2, \dots, m_M} = \frac{N!}{m_1!m_2! \cdots m_M!}$$

giving the total of ways of choosing M different types of objects from the original N , with m_i objects of type i , and where $m_1 + m_2 + \dots + m_M = N$. (Another way to see this directly is the same argument as used for the binomial coefficient: there are N choices for first object, $N-1$ for the second, and so on, giving the numerator, but the order in which the m_1 objects of type 1 are chosen doesn't matter, so we divide by $m_1!$, and so on, through $m_N!$.) And for the same reason that $\binom{N}{m}$ is the coefficient of $x^m y^{N-m}$ in the expansion of the binomial $(x+y)^N$, we see that $\binom{N}{m_1, m_2, \dots, m_M}$ is the coefficient of $x_1^{m_1} x_2^{m_2} \cdots x_M^{m_M}$ in the expansion of the multinomial $(x_1 + \dots + x_M)^N$.

Now consider a string consisting of M different types of symbols (e.g., letters of the alphabet), in which the first appears m_1 times, and so on. The number of possible strings of length N is just given by the above multinomial coefficient, and the number of bits of information that can be conveyed by such strings is therefore

$$\log_2 \binom{N}{m_1, m_2, \dots, m_M} = \log_2 \frac{N!}{\prod_i m_i!}.$$

In the limit of N and all the m_i large, we can use Stirling's approximation to write the logarithms as $\log_2 K \approx K \log_2 K - K \log_2 e$, so that the above information content becomes

$$\log_2 \frac{N!}{\prod_i m_i!} \approx N \log_2 N - \sum_i m_i \log_2 m_i = \sum_i m_i (\log_2 N - \log_2 m_i) = - \sum_{i=1}^M m_i \log_2 \frac{m_i}{N}.$$

In terms of the probabilities $p_i = m_i/N$ of occurrence of the i^{th} symbol type, the average # of bits of information content per symbol in this limit is thus

$$\frac{1}{N} \log_2 \frac{N!}{\prod_i m_i!} \approx - \sum_i \frac{m_i}{N} \log_2 \frac{m_i}{N} = - \sum_{i=1}^M p_i \log_2 p_i.$$

In general, for circumstances in which there's a restricted set of alternatives, labelled say $i = 1, \dots, d$, and each of which can be assigned a probability p_i , there is a way to quantify the “information uncertainty” (due to Shannon), in bits of information:

$$H = \sum_{i=1}^d p_i \log_2 \frac{1}{p_i} = - \sum_{i=1}^d p_i \log_2 p_i . \quad (1)$$

For example, in the case of flipping a single fair coin, we would have $p_1 = p_2 = 1/2$, so

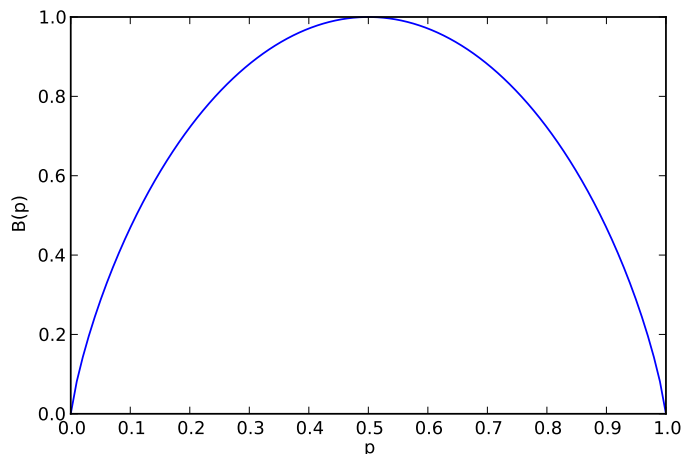
$$H = \frac{1}{2} \log_2 2 + \frac{1}{2} \log_2 2 = \frac{1}{2} + \frac{1}{2} = 1 \text{ bit} ,$$

which is the amount of information conveyed by a single H/T (heads/tails) result. Similarly, the amount of information in 2 fair coin flips ($p_i = 1/4$) is 2 bits, and the amount in 3 fair coin flips ($p_i = 1/8$) is 3 bits.

In general, if there are n equally likely possibilities, then the formula reduces to

$$H = \sum_{i=1}^n p_i \log_2 \frac{1}{p_i} = \sum_{i=1}^n \frac{1}{n} \log_2 n = \log_2 n \text{ bits} ,$$

so the amount of information conveyed by rolling a fair die ($p_i = 1/6$) is $\log_2 6 \approx 2.6$, intermediate between 2 and 3 bits. The information in eqn. (1) is maximized in this equiprobability case: when the p_i are not all equal (but of course still sum to 1), then the information uncertainty is always less than $\log_2 n$. For example, a coin that has a 99% probability of coming up heads has $H = -(99/100) \log_2(99/100) - (1/100) \log_2(1/100) \approx .08$ bits of information, where much less than 1 bit of information is acquired since there was already a large likelihood the result would be heads. The general result for the coin that comes up H with probability $p \in [0, 1]$ is given by $B(p) = -p \log_2 p - (1-p) \log_2(1-p)$, and has maximum of 1 bit for $p = 1/2$:



```
x = np.arange(.001,1,.001)
plot(x, -x*log2(x) - (1-x)*log2(1-x) )
xticks(np.arange(0,1.01,.1))
xlabel('p'), ylabel('B(p)');
savefig('Bp.pdf')
```

The formula (1) has the property that the information is additive when combining independent systems, as in the cases of the multiple coin flips above. This is easy to see in the special case of two systems with m and n equiprobability possibilities, respectively, where the information satisfies $H = \log_2 mn = \log_2 m + \log_2 n$ and is hence the sum of the information uncertainties for the two subsystems (this is ultimately the reason for the logarithm). For example, in the case of flipping a coin and rolling a die, there are 12 possibilities with $p_i = 1/12$ (coin is H, die is 1, etc.) and the total information is $\log_2 12 = \log_2 2 + \log_2 6 \approx 3.6$.

Shannon (1948) showed that eqn.(1) is the unique measure of information (up to overall normalization, which we choose to measure in bits) which has the properties: additive as described above (and independent of the order in which the system is divided into parts), continuous in all the p_i , symmetric in the p_i (i.e., independent of their order), is maximized when all possibilities are equally likely, and increases with that total number of possibilities.

To get more intuition into what it means to have a non-integer number of bits of information, consider a stream of letters $ABAACBABd\dots$ generated by the probability distribution: $p_A = 1/2$, $p_B = 1/4$, $p_c = 1/8$, $p_d = 1/8$. The information content is

$$H = -\frac{1}{2} \log \frac{1}{2} - \frac{1}{4} \log \frac{1}{4} - \frac{1}{8} \log \frac{1}{8} - \frac{1}{8} \log \frac{1}{8} = \frac{1}{2} + \frac{1}{2} + \frac{3}{8} + \frac{3}{8} = 1\frac{3}{4}.$$

In order to transmit the stream of information, naively it would take 2 bits/character to represent each of four possibilities. But can we somehow encode a stream of unequal probability alternatives using fewer bits/character on average? The idea is to reserve the smallest number of bits for the most frequent characters to reduce the average usage. So we represent the most probable A by a single 0, and use 1 as a signal that we need to look at the next bit (known as a “prefix code”). Then we can represent B by 10, and use 11 to signal that we need to look at a third bit, so c can be encoded as 110 and d as 111. The sequence $ABAACBABd$ is then encoded as 0 10 0 0 110 10 0 10 111, and using the rules we can see that the sequence 0100011010010111 can be unambiguously decoded as $ABAACBABd$. What is the average number of bits per character used in this scheme? Since A occurs half the time and needs 1 bit, B occurs 1/4 of the time and needs 2 bits, c and d each occur 1/8 of the time and need 3 bits, on average this means

$$\frac{1}{2} \cdot 1 + \frac{1}{4} \cdot 2 + \frac{1}{8} \cdot 3 + \frac{1}{8} \cdot 3 = 1\frac{3}{4} \text{ bits per character.}$$

In general, the Shannon information gives the number of bits per character for an optimal encoding scheme (if it exists), resulting in the optimal compression ratio.

Shannon also experimented with estimating the bits/character of English text. Since the probabilities of the 26 letters are far from equal, we expect somewhat less than the

maximum $\log_2 26 \approx 4.7$ bits/char. The uncertainty per character was first estimated by giving people a section of text and asking them to guess the next letter (in principle they could employ 1- and higher gram probability distributions of letter co-occurrences to aid them, now this is easier with computers ...). The typical estimates are in the range 1–1.5 bits/char, which is why it's possible to compress text files (using gz, zip, or equivalent).

With the notion of information in hand, there's a related notion of *mutual information* shared by two random variables. Roughly speaking, it quantifies the extent to which they are dependent (so that independent random variables have zero mutual information). It is fun to describe this metaphorically. Consider that the world W consists of a set of states $w \in W$ with some probability distribution $\sum_{w \in W} p(w) = 1$, and associated information uncertainty $H[W] = -\sum_{w \in W} p(w) \log_2 p(w)$. Now imagine there are certain types of data $d \in D$ that we can measure, which as well come with some probability distribution $\sum_{d \in D} p(d) = 1$, and associated information uncertainty $H[D] = -\sum_{d \in D} p(d) \log_2 p(d)$.

Let's ask *on average* how much information we can expect to obtain about the world by making these measurements. After measuring some d , the new probability distribution for the world is $p(w|d) = p(w, d)/p(d)$, i.e., conditioned on having measured d . The new information uncertainty is $H[W|d] = -\sum_{w \in W} p(w|d) \log_2 p(w|d)$, and if lower than $H[W]$ their difference represents the amount of information about the world obtained from having measured d . On average, the expected information uncertainty after measuring is thus $\sum_{d \in D} p(d)H[W|d]$. The mutual information $I[W; D]$ between the world W and data D is defined as the expected information gain from making the measurements: $I[W; D] = H[W] - \sum_{d \in D} p(d)H[W|d]$. From the definitions, we find that it satisfies

$$\begin{aligned} I[W; D] &= H[W] - \sum_{d \in D} p(d)H[W|d] \\ &= -\sum_{w \in W} p(w) \log_2 p(w) + \sum_{d \in D} p(d) \sum_{w \in W} p(w|d) \log_2 p(w|d) \\ &= -\sum_{w \in W, d \in D} p(w, d) \log_2 p(w) + \sum_{w \in W, d \in D} p(w, d) \log_2 p(w, d)/p(d) \\ &= \sum_{w \in W, d \in D} p(w, d) \log_2 \frac{p(w, d)}{p(w)p(d)}. \end{aligned}$$

Perhaps surprisingly, the result is symmetric in W and D : on average we learn the same amount about the world from measuring the data, as we learn about the probability distribution of likely data from knowing about the world. If W and D are independent, i.e., $p(w, d) = p(w)p(d)$ for all $w \in W, d \in D$, then the argument of the logarithm is always 1 and $I[W; D] = 0$. More generally, the mutual information satisfies* $I[W; D] \geq 0$, and vanishes only when the events are independent. Note also: $I[W; D] = H[W] + H[D] - H[W, D]$, where $H[W, D]$ is the information uncertainty of the joint distribution $p(w, d)$.

* This can easily be proved using the inequality $\ln x \leq x - 1$.