Shannon's noiseless coding theorem

We are working with messages written in an alphabet of symbols $x_1, \ldots, x_n$ which occur with probabilities $p_1, \ldots, p_n$. We have defined the *entropy* E of this set of probabilities to be

$$E = -\sum_{i=1}^{n} p_i \log_2 p_i.$$

These pages give a proof of an important special case of Shannon's theorem, which holds for any uniquely decipherable code. We will prove it for *prefix codes*, which are defined as follows:

Definition: A (binary) *prefix code* is an assignment of binary strings (strings of 0s and 1s, "code words") to symbols in the source alphabet so that no code word occurs as the beginning of another code word.

Note that message written in a prefix code can be unambiguously decoded by "slicing off" code words as they occur.

Theorem: For any binary prefix code encoding $x_1, \ldots, x_n$ the average length of a word must be greater than E. More explicitly, setting ℓ_i as the length of the code word for x_i,

$$\sum_{i=1}^{n} p_i \ell_i \geq E.$$

Our proof of this theorem will involve two lemmas.

Lemma 1: (Gibbs' inequality). Suppose $p_1, \ldots, p_n$ is a *probability distribution* (i.e. each $p_i \geq 0$ and $\sum_i p_i = 1$). Then for any other probability distribution $q_1, \ldots, q_n$ with the same number of elements,

$$\sum_{i=1}^{n} p_i \log_2 p_i \geq \sum_{i=1}^{n} p_i \log_2 q_i.$$

(Note: the inequality is usually stated

$$-\sum_{i=1}^{n} p_i \log_2 p_i \leq -\sum_{i=1}^{n} p_i \log_2 q_i.$$

Our formulation avoids many minus signs, even though the numbers involved are both negative. For a heuristic motivation consider the case where all the p_i are equal. Each $p_i = \frac{1}{n}$ and

$$\sum_{i=1}^{n} p_i \log_2 p_i = \sum_{i=1}^{n} \log_2 p_i = \log_2(p_1 \cdot p_2 \cdot \ldots \cdot p_n).$$

Substituting $q_1, \ldots q_n$ for $p_1, \ldots p_n$ gives $\log_2(q_1 \cdot q_2 \cdot \ldots \cdot q_n)$. And we know that when the sum of the side-lengths is fixed, the maximum volume of a rectangular solid is obtained when all the sides are equal.)

Proof: (from http : //en.wikipedia.org/wiki/Gibbs%27_inequality)

Since $\log_2 p_i = \frac{\ln p_i}{\ln 2}$ and $\ln 2 > 0$ it is enough to prove the inequality with $\log_2$ replaced by ln wherever it occurs.

We use the following property of the natural logarithm:

$$\ln x \leq x - 1 \text{ for all } x > 0, \text{ and } \ln x = x - 1 \text{ only when } x = 1.$$

In order to avoid zero denominators in the following calculation, we set $I = \{i | p_i > 0\}$, the set of indices for which p_i is non-zero. Then we write

$$\sum_{i \in I} p_i \ln \frac{q_i}{p_i} \leq \sum_{i \in I} p_i (\frac{q_i}{p_i} - 1) = \sum_{i \in I} q_i + \sum_{i \in I} p_i = \sum_{i \in I} q_i + 1 \leq 0.$$

Since $\ln \frac{q_i}{p_i} = \ln q_i - \ln p_i$, this chain of inequalities gives

$$\sum_{i \in I} p_i \ln q_i \leq \sum_{i \in I} p_i \ln p_i.$$

Now $\sum_{i \in I} p_i \ln p_i = \sum_{i=1}^{n} p_i \ln p_i$ since the new terms all have $p_i = 0$; and $\sum_{i \in I} p_i \ln q_i \geq \sum_{i=1}^{n} p_i \ln q_i$ since new terms are ≤ 0. I.e.

$$\sum_{i=1}^{n} p_i \ln q_i \leq \sum_{i \in I} p_i \ln q_i \leq \sum_{i \in I} p_i \ln p_i = \sum_{i=1}^{n} p_i \ln p_i$$

yielding Gibbs' inequality.

Lemma 2: (Kraft's inequality for binary prefix codes) Let $x_1, \ldots, x_n$ be the symbols in our alphabet, and suppose we have encoded them as binary words using a prefix code. Let $\ell_1, \ldots, \ell_n$ be the lengths of the words corresponding to $x_1, \ldots, x_n$. Then

$$\sum_{i=1}^{n} 2^{-\ell_i} \leq 1.$$

Proof: Note that a (binary) prefix code can always be represented as a binary tree: as a word is read, the tree branches right or left according as the next bit is 0 or 1. Each word occurs at the end of a unique "branch."

Set $L = \max_i \ell_i$. Then the tree corresponding to our prefix code can be extended to a tree where every branch has length L, and there are 2^L branches. A code word of length ℓ_i corresponds to pruning off from this tree all the possible extensions of the corresponding branch. There are $2^{L-\ell_i}$ of these. The total number of deleted branches is then $\sum_i 2^{L-\ell_i}$; since this sum must be smaller than the total number of branches, we have

$$\sum_{i=1}^{n} 2^{L-\ell_i} = 2^L \sum_{i=1}^{n} 2^{-\ell_i} \leq 2^L,$$

so $\sum_{i=1}^{n} 2^{-\ell_i} \leq 1$, Kraft's inequality.

Proof of Shannon's theorem: Take $x_1, \ldots, x_n$ and $p_1, \ldots, p_n$ as in the statement, suppose the x_i have been encoded in a binary prefix code, and let ℓ_i be the length of the code word for x_i. Then by Kraft's inequality $\sum_i 2^{-\ell_i} \leq 1$. Call this number $1/C$, so that $C2^{-\ell_1}, \ldots, C2^{-\ell_n}$ is a probability distribution, and can play the role of $\{q_i\}$ in Gibbs' inequality, which then tells us

$$\sum_{i=1}^{n} p_i \log_2 p_i \geq \sum_{i=1}^{n} p_i \log_2(C2^{-\ell_i}) = \sum_{i=1}^{n} p_i(\log_2 C - \ell_i) = \log_2 C - \sum_{i=1}^{n} p_i \ell_i.$$

Now put back the minus signs and remember that since $1/C \leq 1$ we have $C \geq 1$ and $\log_2 C \geq 0$. We get

$$\sum_{i=1}^{n} p_i \ell_i \geq -\sum_{i=1}^{n} p_i \log_2 p_i + \log_2 C \geq -\sum_{i=1}^{n} p_i \log_2 p_i,$$

as required.