Information Theory SV1

Question sources: Matthew Ireland's SV worksheet, 23/24 Examples Sheet

Question 1: What is the maximum possible entropy H of an alphabet consisting of N different letters? In such a maximum entropy alphabet, what is the probability of its most likely letter? What is the probability of its least likely letter?

Question 2: If discrete symbols from an alphabet S having entropy $H(S)$ are encoded into blocks of length n symbols, we derive a new alphabet of symbol blocks S^n. If the occurrence of symbols is independent, derive the entropy $H(S^n)$ of this new alphabet.

Question 3: Why are fixed-length codes inefficient for alphabets whose letters are not equiprobable? Discuss this in relation to Morse Code.

Question 4: A fair coin is secretly flipped until the first head occurs. Let X denote the number of flips required. The flipper will truthfully answer any "yes-no" questions about his experiment, and we wish to discover thereby the value of X as efficiently as possible.

(a) What is the most efficient possible sequence of such questions?

(b) On average, how many questions should we need to ask?

(c) Relate the sequence of questions to the bits in a uniquely decodable prefix code for X.

Question 5: Is it possible to construct a prefix code in which the codewords have the following lengths: 1, 2, 3, 3, 4, 4?

Question 6: Implement an algorithm (in your chosen programming language) that takes a list of codeword lengths and either provides a prefix code with these lengths or reports that it is impossible.

Question 7: Construct an ensemble where the difference between the entropy and the expected length of the Huffman code is as large as you can make it.

Question 8: Is it possible for two people to use the same set of alphabet probabilities to produce Huffman codes with a differing set of lengths?

Question 9: Consider the following symbol encoding scheme:

(a) Each input symbol is associated with a probability of occurrence and an initially blank codeword.

(b) Partition the set of input symbols into two subsets, where the sum of the probabilities in each subset is as close to equal as possible.

(c) Append a 0 to every codeword in the first subset and a 1 to every codeword in the second subset.

(d) Recurse on each subset until there is only one symbol left in each partition.

Does this produce instantaneous codes? How does it compare to Huffman coding? (Note: full analysis of the average codeword length is non-trivial and not required, but you may attempt it.)

Mackay ex 4.9   While some people, when they first encounter the weighing problem with 12 balls and the three-outcome balance, think that weighing six balls against six balls is a good first weighing, others say "no, weighing six against six conveys no information at all". Explain to the second group why they are both right and wrong. Compute the information gained about which is the odd ball, and the information gained about which is the odd ball and whether it is heavy or light.

Question 10: You are tasked with investigating a funky random number generator, which generates integers i, where $1 \leq i \leq n$. The true distribution $(p_1, p_2, \ldots, p_n)$ is unknown, but the average μ is known. In order to perform inference and estimate the posterior distribution using Bayes' theorem, we need a prior distribution.

(a) Show, using the method of Lagrange multipliers, that for the maximum entropy prior, p_i can be written as Cr^i for some C and r.

(b) Use normalisation to find C.

(c) Use the known average of the distribution to obtain the following equation:
$$nr^{n+1} - (n+1)r^n + 1 = \mu(r^n - 1)(r - 1)$$