TUTORIAL IV

1 Homework 3

1.1 Huffman's algorithm

Huffman's algorithm constructs a prefix code C_{H} given a distribution $(p_1, \ldots, p_m)$ on the symbols $\{1, \ldots, m\}$. The objective of this problem is to show that the expected length $L(C_{\mathrm{H}})$ is minimum among all the prefix codes. Huffman's algorithm constructs a binary tree as follows. The algorithm starts with independent nodes labeled by the elements $1, \ldots, m$ and the corresponding probability. At the beginning, all the nodes or marked unvisited. At each step, we choose the two unvisited nodes u, v with minimum value of p_u, p_v. We create a new node w with an assigned probability $p_w = p_u + p_v$ which is the parent of u and v. w is marked as unvisited and u, v are marked as visited. The step is repeated $m - 1$ times until we have one unvisited node (the root) with an assigned probability 1. To every path from the root to a leaf of the tree, we assign a bitstring where a "left" edge is read as 0 and a "right" edge is read as 1. The obtained tree defines a code in the following way: for any $x \in \{1, \ldots, m\}$, $C_{\mathrm{H}}(x)$ is the bitstring corresponding to the path from the root to x.

1. Show that for any optimal code, it can be transformed to one with the following property: the two longest codewords correspond to the two least likely symbols, and they have the same length and they only differ in the last bit.

2. Conclude that C_{H} achieves the optimal expected length for $(p_1, \ldots, p_m)$.

1.2 Data compression

This problem is to illustrate that some stream codes can use much less than one bit per symbol of the source. Assume the source is composed of symbols in $\mathcal{X}$ and $0 \in \mathcal{X}$ is one of the symbols. To simplify the calculations, you may assume $\mathcal{X} = \{0, 1\}$. For the two types of encodings described below, determine the length of the encoding of the bitstring 0^n (i.e., n times the symbol 0). You can give your result in the form $O(f(n))$ for the smallest possible f.

1. Arithmetic code with the following simple probabilistic model: $P_\ell(x_\ell | x_1 \ldots x_{\ell-1}) = \frac{|\{i \in \{1,\ldots,\ell-1\} : x_i = x_\ell\}| + 1}{\ell - 1 + |\mathcal{X}|}$.

2. There is a widely used family of compression algorithms named after Lempel-Ziv. Unlike all schemes we have seen in class that explicitly use the probabilistic model, these compression algorithms do not make use of any probabilistic model and they are called *universal* for this reason. The basic idea is to use repetitions of sequences of symbols. See http://www-math.mit.edu/ shor/PAM/lempel_ziv_notes.pdf for a description of a simple variant that you are asked to consider.[1]

2 Code for unknown distribution

Recall that we can build a code C that achieve an expected description length $l(C)$ within 1 bit of the lower bound, that is:

$$H(X) \leq l(C) < H(X) + 1$$

This is done using the following choice of word lengths: $l_i = \left\lceil \log \frac{1}{p_i} \right\rceil$. In some case, we don't know the true distribution p, but only have an approximation q, and still want to find a code.

[1] The linked document also shows that if one applies this compression algorithm to an i.i.d. sequence of symbols, the number of used bits per symbol is optimal: it is the entropy of the source.

1. Show that if we use the same choice of word lengths: $l_i = \left\lceil \log \frac{1}{q_i} \right\rceil$, we have:

$$H(p) + D(p||q) \leq E_p(l(C)) < H(p) + D(p||q) + 1$$

3 Channel capacity

1. For a discrete channel $W_{\mathcal{Y}|\mathcal{X}}$ with input alphabet $\mathcal{X}$, output alphabet $\mathcal{Y}$ and probability transition matrix $p(y|x)$, let Let $C(W)$ denote the channel capacity of W. Show that
   (a) $C(W) \geq 0$.
   (b) $C(W) \leq \log_2 |\mathcal{X}|$.
   (c) $C(W) \leq \log_2 |\mathcal{Y}|$.
   (d) $I(X;Y)$ is a continuous concave function of $p(x)$.
2. Given a channel $W_{\mathcal{X}|\mathcal{Y}}$ with transition probabilities $p(y|x)$ and channel capacity $C(W) = \max_{p(x)} I(X;Y)$, suppose you apply a preprocessing step to the output by forming $\tilde{Y} = g(Y)$.
   (a) Does it strictly improve the channel capacity?
   (b) Under what conditions does the capacity not strictly decrease?

4 Jointly typical sequences

The set $A_\epsilon^{(n)}$ of jointly typical sequences $\{(x^n, y^n)\}$ with respect to the distribution $p_{XY}(x, y)$ is given by

$$A_\epsilon^{(n)} = \left\{ (x^n, y^n) \in \mathcal{X}^n \times \mathcal{Y}^n : \left| -\frac{1}{n} \log p_{X^n}(x^n) - H(X) \right| < \epsilon, \right.$$
$$\left| -\frac{1}{n} \log p_{Y^n}(y^n) - H(Y) \right| < \epsilon,$$
$$\left. \left| -\frac{1}{n} \log p_{X^nY^n}(x^n, y^n) - H(X,Y) \right| < \epsilon \right\}$$

where $p_{X^nY^n}(x^n, y^n) = \prod_{i=1}^n p_{XY}(x_i, y_i)$.

1. Let (X^n, Y^n) be sequences of length n drawn i.i.d. according to $p_{X^nY^n}(x^n, y^n) = \prod_{i=1}^n p_{XY}(x_i, y_i)$. Then show that
   (a) $\Pr[(X^n, Y^n) \in A_\epsilon^{(n)}] \to 1$ as $n \to \infty$.
   (b) $|A_\epsilon^{(n)}| \leq 2^{nH(X,Y)+\epsilon}$.
   (c) If $(\tilde{X}^n, \tilde{Y}^n) \sim p_{X^n}(x^n)p_{Y^n}(y^n)$, then $\Pr[(\tilde{X}^n, \tilde{Y}^n) \in A_\epsilon^n] \leq 2^{-n(I(X;Y)-3\epsilon)}$.