Homework Set #2

1. Construct a source on alphabet \{A, B, C, D, E\} such that the average codeword length of the Shannon–Fano code is strictly larger than that of the Huffman code.

2. Consider a source on alphabet \{A, B, C, D\} with probabilities $p_A \geq p_B \geq p_C \geq p_D$. Find the sufficient and necessary condition that the binary Huffman code is \{00, 01, 10, 11\}.

3. In the lecture notes, we defined a uniquely decodable code as one such that every distinct concatenation of $n$ codewords is distinct for every $n$; in other words, if

$$x_1 \cdots x_n \neq y_1 \cdots y_n,$$

then

$$C(x_1) \cdots C(x_n) \neq C(y_1) \cdots C(y_n)$$

for every $n$. Show that the following more general statement holds for a uniquely decodable code. If

$$x_1 \cdots x_m \neq y_1 \cdots y_n,$$

then

$$C(x_1) \cdots C(x_m) \neq C(y_1) \cdots C(y_n)$$

for every $m, n$.

4. Consider an i.i.d. source with alphabet \{A, B, C, D\} and probabilities \{0.1, 0.2, 0.4, 0.3\}.

(a) Compute the entropy of the source base 2, 3, and 4.

(b) Find a binary Huffman code for encoding 2 source symbols at a time. Compute the average number of binary code symbols per source symbol and compare it to the entropy (with the appropriate base).

(c) Repeat part (b) for ternary codes. Here the code alphabet is \{0, 1, 2\}.

(d) Repeat part (b) for quaternary codes. Here the code alphabet is \{0, 1, 2, 3\}.
Programming Assignment

You can use any programming language you prefer (MATLAB, Python, C/C++, or Julia, for example). Write down your code as clearly as possible and add suitable comments. Turn in the hard copy of answers to each problem and submit your code to ece154ucsd@gmail.com with the exact subject ECE 154C (HW2).

1. Write a program for a function \texttt{daryHuffman(pmf, d)} that takes a probability vector \texttt{pmf} and the size of alphabet \texttt{d} as inputs, and outputs a \texttt{d}-ary Huffman code. For example,

\[
\text{["0", "1", "20", "21", "22"] = daryHuffman([0.6, 0.2, 0.1, 0.05, 0.05], 3)}
\]

(a) Run the program for \texttt{pmf} = \{0.6, 0.3, 0.1\} with \texttt{d} = 3 and compute the average length of the resulting Huffman code.

(b) Consider block lengths \texttt{n} = 2, 3, 5. For each block length, find the ternary Huffman code and the average length per symbol.

2. Consider a binary source \{\texttt{A, B}\}, and suppose we have a probability vector \texttt{src.pmf} on this set. Write a program for a function \texttt{Tunstall(src.pmf, num.phrases)} that takes the probability vector \texttt{src.pmf} and the desired number of phrases \texttt{num.phrases} as inputs and outputs the resulting phrases of Tunstall coding and the probability vector of it. For example,

\[
\text{["B", "AB", "AAA", "AAB"], [0.1, 0.09, 0.729, 0.081] = Tunstall([0.9, 0.1], 4)},
\]

and

\[
\text{["B", "AB", "AAB", "AAAA", "AAAB"], [0.2, 0.16, 0.128, 0.4096, 0.1024] = Tunstall([0.8, 0.2], 5)}.
\]

(a) For \texttt{src.pmf} = \{0.9, 0.1\}, find the length of codewords per average number of symbols for \texttt{num.phrases} = 4, 8, 16, \ldots, \texttt{210} = 1024.

(b) For \texttt{src.pmf} = \{\texttt{p}, 1 - \texttt{p}\}, plot the length of codewords per average number of symbols for \texttt{num.phrases} = 1024 as a function of \texttt{p}. The plot should be evaluated at 100 values of \texttt{p}.

(c) Compare the result in part (b) with the Huffman code rate in Q4 of Homework Set #1.