2015 Fall - Information Theory

Homework 3 (Due on Oct. 23rd)

Part 1: Analysis of Some Good Codes

Consider the distribution

$$\mathrm{P_X(x)} = \begin{cases} 0.49, & x = x_1 \\ 0.26, & x = x_2 \\ 0.12, & x = x_3 \\ 0.04, & x = x_4 \\ 0.04, & x = x_5 \\ 0.03, & x = x_6 \\ 0.02, & x = x_7 \end{cases}$$

Now

- find a Shannon type code,
- find a Shannon code,
- find a Fano code,
- find the Huffman code.

Part 2: Huffman code

Which of the following assignment cannot possibly be a Huffman code ?

- $\{0, 10, 11\}$
- $\{00, 01, 10, 110\}$
- $\{01, 10\}$

Part 3: 20 question game

Consider a set of n objects, each object can be good $(x_i = 1)$ or defective $(x_i = 0)$ with $P[x_i = 1] = p_i$ and $p_i \geq 1/2$
We are asked to determine the set of all defective objects. We can ask any set of yes/no questions.

- Give a good lower bound on the average number of question required to determine the set of defective objects.
- If the longest set of questions is required by nature's answers our questions, what is the last question we should ask? And what two sets are we distinguishing with this question? Assume a compact (minimum average length) sequence of questions.
- Give an upper bound (within 1 question) on the minimum average number of question required.

Part 4: Be creative

Huffman coding has two drawbacks:

- It requires to know the distribution of the source ahead of time.
- Building the tree is very expensive when we have a large support.

In view of this:

- How would you build a code for source for which we know the ordering of the distribution only, that is we know that $p_1 > p_2 > p_3 ... > p_n$? We don't know the exact value of the probability.
- How would you build a code for a geometric distributed random variable, in which the support is infinite ?

Part 5: Complexity

Consider a uniformly distributed random variable over a support of size n , please generate

- Shannon code
- Fano code
- Huffman code

Now come up with a way to generate an optimal code that has the smallest complexity.

Part 6: Matlab Exercise

download the rat genome from :
http://hgdownload.cse.ucsc.edu/goldenPath/rn6/bigZips/
the file name is : rn6.2bit , size: 716M

Consider sets of 3 bases(6 bits) at the time (The cell reads the DNA code in groups of three bases. Each triplet of bases, also called a codon). Figure out a way to compress this sequence codon by codon. Use whatever method you like: the only thing that matter is the compression rate.

Remark : The file is large and reading and writing into files is a very slow process. If you read and write to often the program will take days to execute!

```
fid = fopen('rn6.2bit');
x = fread(fid,1,'ubit6');
```