ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE
School of Computer and Communication Sciences

Handout 15 | Information Theory and Coding
Midterm exam | Oct. 31, 2017

3 problems, 85 points
165 minutes
1 sheet (2 pages) of notes allowed.

Good Luck!

PLEASE WRITE YOUR NAME ON EACH SHEET OF YOUR ANSWERS.

PLEASE WRITE THE SOLUTION OF EACH PROBLEM ON A SEPARATE SHEET.

Problem 1. (25 points) Suppose X,Y and Z are random variables.

(a) (5 pts) Show that $H(X) + H(Y) + H(Z) \geq \frac{1}{2}\big[H(XY) + H(YZ) + H(ZX)\big]$.

(b) (5 pts) Show that $H(XY) + H(YZ) \geq H(XYZ) + H(Y)$.

(c) (5 pts) Show that

$$2\big[H(XY) + H(YZ) + H(ZX)\big] \geq 3H(XYZ) + H(X) + H(Y) + H(Z).$$

(d) (5 pts) Show that $H(XY) + H(YZ) + H(ZX) \geq 2H(XYZ)$.

(e) (5 pts) Suppose n points in three dimensions are arranged so that their their projections to the xy, yz and zx planes give n_{xy}, n_{yz} and n_{zx} points. Clearly $n_{xy} \leq n$, $n_{yz} \leq n$, $n_{zx} \leq n$. Use part (d) show that

$$n_{xy} n_{yz} n_{zx} \geq n^2.$$

Problem 2. (30 points) Consider the distribution Q on the positive integers $\{1, 2, \dots\}$ with

$$Q(u) = (1-2p)p^{\lfloor \log_2(u) \rfloor}, \quad u = 1, 2, \dots,$$

i.e.,

$$\begin{aligned} Q(1) &= (1-2p), \\ Q(2) = Q(3) &= (1-2p)p, \\ Q(4) = Q(5) = Q(6) = Q(7) &= (1-2p)p^2, \\ &\cdots\cdots \\ Q(2^j) = \cdots = Q(2^{j+1}-1) &= (1-2p)p^j, \quad (j = 0, 1, \dots). \end{aligned}$$

Suppose the random variable V has distribution Q.

(a) (5 pts) Find $H(V)$. [Hint: $\sum_{j=0}^{\infty} x^j = 1/(1-x)$, $\sum_{j=1}^{\infty} jx^j = x/(1-x)^2$.]

(b) (5 pts) Find $L(V) := E[\lfloor \log_2 V \rfloor]$.

(c) (5 pts) Show that

$$\begin{aligned} H(V) &= L(V) + L(V)\log_2(1 + 1/L(V)) + \log_2(1 + L(V)) && \text{(c1)} \\ &\le L(V) + \log_2(1 + L(V)) + \log_2 e. && \text{(c2)} \end{aligned}$$

(d) (5 pts) With V as above, show that if U is a random variable taking values in $\{1, 2, \dots\}$ for which $L(U) = L(V)$, then

$$\begin{aligned} H(U) &\le \sum_i \Pr(U = i) \log_2 \frac{1}{Q(i)} && \text{(d1)} \\ &= \sum_i \Pr(V = i) \log_2 \frac{1}{Q(i)} && \text{(d2)} \\ &= H(V). \end{aligned}$$

Order the set of binary strings in increasing length (λ, 0, 1, 00, 01, ...), with λ denoting the null string. Note that $\lfloor \log_2 i \rfloor$ is the length of the i'th string in this list.

(e) (5 pts) Suppose U is a random variable taking values in $\{1, 2, \dots\}$ with distribution P, and suppose $P(1) \ge P(2) \ge \dots$. Find the non-singular code $\mathcal{C}$ with smallest possible $E[\text{length}(\mathcal{C}(U))]$, and express length $\mathcal{C}(u)$ in terms of $\log_2 u$.

(f) (5 pts) For any random variable U taking values in $\{1, 2 \dots\}$, let

$$L^* = \min_{\mathcal{C}\text{: non-singular}} E[\text{length}(\mathcal{C}(U))].$$

Show that $H(U) \le L^* + \log(1 + L^*) + \log_2 e$.

PROBLEM 3. (30 points) Consider the following variation on the Lempel-Ziv algorithm to encode an infinite sequence $u_1u_2\dots$ from an alphabet $\mathcal{U}$.

1. Set the dictionary $\mathcal{D} = \mathcal{U}$. Denote the dictionary entries as $d(0), \dots, d(s-1)$, with $s = |\mathcal{U}|$ being the size of the dictionary. Set $i = 0$ (the number of input letters read so far).

2. Find the largest l such that $w = u_{i+1}\dots u_{i+l}$ is in $\mathcal{D}$.

3. With $0 \le j < s$ denoting the index of w in $\mathcal{D}$, output the $\lceil \log_2 s \rceil$ bit binary representation of j.

4. Add the word wu_{i+l+1} to $\mathcal{D}$, i.e., set $d(s) = wu_{i+l+1}$, and increment s by 1. Increment i by l. Goto step 2.

For example, with $\mathcal{U} = \{\texttt{a}, \texttt{b}\}$, the input string `abbbbaaab...` will lead to the execution steps

$\mathcal{D}$ at 2	w	output at 3	added-word at 4
a b	a	0	ab
a b ab	b	01	bb
a b ab bb	bb	11	bbb
a b ab bb bbb	b	001	ba
a b ab bb bbb ba	a	000	aa
a b ab bb bbb ba aa	aa	110	aab

(a) (5 pts) Can the decoder reconstruct the input sequence $u_1u_2\dots$ from the output of the algorithm? If so, how? (The crucial difficulty is that the description of w in step 3 does not determine the word added to the dictionary in step 4.)

(b) (5 pts) The algorithm parses the sequence $u_1u_2\dots$ into a sequence of words $w_1w_2\dots$, (the w's found in step 2). Show that a word w can appear at most $|\mathcal{U}|$ times in the parsing.

(c) (5 pts) Suppose $u^n = u_1\dots u_n$ is parsed into $m(u^n)$ words $w_1\dots w_m$ by the algorithm. Show that for any $k \ge 1$

$$n \ge k[m(u^n) - F(k)],$$

where $F(k) = |\mathcal{U}|\sum_{i=1}^{k-1}|\mathcal{U}|^i$.

(d) (5 pts) Show that $\lim_{n\to\infty} m(u^n)/n = 0$.

(e) (5 pts) Show that after reading u^n the algorithm outputs fewer than $m(u^n)\lceil \log_2[|\mathcal{U}| + m(u^n)]\rceil$ bits.

Let $L(m, k)$ denote the minimum possible total length of a collection of m binary strings where no string appears more than k times.

(f) (5 pts) Show that if u^n is fed to an information lossless finite state machine with s states, then the machine outputs at least $L(m(u^n), s^2|\mathcal{U}|)$ bits.