10-704: Information Processing and Learning **Spring 2012**

Homework 4: Solution

Lecturer: Aarti Singh

Note: *The TA graciously thank Rafael Stern for providing most of these solutions*

4.1 Problem 1

4.1.1 (a)

Recall that a tree is uniquely identified by the set of its leaves. Every node in the tree can be obtained through a binary search, which we represent by a sequence of 0's and 1's. For example, 01 represents starting at the whole interval, dividing it in two, going to the left split, dividing the left split in two and going to the right split of the current interval. Hence, the set of all leaves completely determines the tree and every leaf can be encoded by its corresponding binary search.

Let $|\tau|$ denote the number of leaves of the tree. We can encode this number as in $HW3.2.b$. Similarly, we can encode every binary search sending every direction twice and appending a 01 in the end. For example, 01 would become 001101. Finally, encode the tree by sending the encoded version of $|\tau|$ and then sending the encoded version of each leaf in any order. This is a prefix code by the same argument as in $HW3.2.d$. In the worst case scenario, this encoding scheme uses $\gamma + 2^{\gamma}(2\gamma + 2)$ bits.

Next, for each leaf, we need to encode the probability of a data point being in it. Order the leaves according to the lexicographic order in their binary search representations. Observe that the empirical probabilities can assume at most $(n + 1)$ values. Hence, there are at most $(n + 1)^{|\tau|}$ combinations of possible empirical probabilities for the ordered leaves. Hence, we can encode the observed empirical probability with $|\tau| \log(n+1)$ bits.

4.1.2 (b)

Let $l_q(x^n)$ denote the log-likelihood as an argument of the model q. The description length of model q is given by:

$$|\tau_q| + 2^{|\tau_q|}(2|\tau_q| + 2) + |\tau| \log(n + 1) - l_q(x^n)$$

The two stage optimization procedure consists of:

1. For each tree, choose the leaf probabilities which maximize the Shannon information of the data. These probabilities maximize $l_q(x^n)$ and are the maximum likelihood estimates. That is, they correspond to the empirical probabilities.

2. Choose the tree (with the leaf probabilities selected in step 1) which minimizes the description length.

4.1.3 (c)

First, observe that the encoding of the tree does not grow with n and, thus, is negligeble for sufficiently large n. Hence, for large n we are interested in minimizing:

$$|\tau| \log(n+1) - l_q(x^n)$$

An efficient way of searching for this minimizer is starting with a complete tree and prunning the pair of leaves which provide the least information until prunning such leaves increases the description length.

4.2 Problem 2

4.2.1 (a)

Let X be an encoding. Consider the process of inputting X to channel 1 and define X_1 as the output of the channel. Consider passing X_1 as an input to channel 2 and define the final output as $X_{1,2}$. Also consider the process of inputting X to the channel with capacity C and define the output as X_3. Observe that X_3 and $X_{1,2}$ are identically distributed conditionally on X. Hence, $I(X;X_3) = I(X;X_{1,2})$. Next, by the data processing inequality, for any distribution on X, $I(X;X_{1,2}) \leq I(X;X_1)$. Hence,

$$C = \max_{F_X(x)} \{I(X;X_3)\} = \max_{F_X(x)} \{I(X;X_{1,2})\} \leq \max_{F_X(x)} \{I(X;X_1)\} = C_1$$

Also using the data processing inequality, observe that $I(X;X_{1,2}) \leq I(X_1;X_{1,2})$. Hence,

$$C = \max_{F_X(x)} \{I(X;X_3)\} = \max_{F_X(x)} \{I(X;X_{1,2})\} \leq \max_{P_1 F_X(x)} \{I(X_1;X_{1,2})\} \leq \max_{F_{X_1}(x_1)} \{I(X_1;X_{1,2})\} = C_2$$

Conclude that $C \leq \min\{C_1, C_2\}$.

4.2.2 (b)

Let X and Y be, respectively, the input and output of this channel. Call $P(X=1) = p$. Recall that $I(X;Y) = H(Y) - H(Y|X)$. Observe that:

$$H(Y|X) = (1-p)H(0) + pH(0.5) = p$$

and

$$H(Y) = -P(Y=0)\log(P(Y=0)) - P(Y=1)\log(P(Y=1)) =$$

$$-(1-0.5p)\log(1-0.5p) - 0.5p\log(0.5p)$$

Hence, $I(X;Y) = -(1-0.5p)\log(1-0.5p) - 0.5p\log(0.5p) - p$.

$$\frac{dI(X;Y)}{dp} = 0.5\log(1-0.5p) - 0.5\log(0.5p) - 1$$

Setting $\frac{dI(X;Y)}{dp} = 0$:

$$\frac{1-0.5\hat{p}}{0.5\hat{p}} = 2$$

$$1 - 0.5\hat{p} = 2\hat{p}$$

$$\hat{p} = \frac{2}{5}$$

Since $p \in [0,1]$ and $I(X;Y) = 0$ for $p = 0$ and $p = 1$, conclude by Weierstrass's Theorem that $\hat{p}$ maximizes the mutual information.

4.3 Problem 3

4.3.1 (a)

4.3.1.1 (i)

We use the following fact from Statistics: Let $X = (X_1, ..., X_n) \sim N(0, \sigma^2 I_p)$, then we have that $\mathbb{E}\|X\|^2 = \sigma^2 n$ and $\|X\|^2$ is sharply concentrated around $\sigma^2 n$.

The expectation can be computed in a straightforward manner; the proof of the concentration uses a Gaussian tail bound and can be found in intermediate level mathematical statistics textbooks (or Google)

We have then that $\|X\|$ is about $\sigma\sqrt{n}$ with high probability.

4.3.1.2 (ii)

The volume of a n-sphere of radius R is cR^n. Hence, the volume of the sphere the sequence is expected to lie in is $c(\sqrt{n\sigma^2})^n$. Similarly, the volume of each sphere corresponding to a codeword is $c(\sqrt{nD})^n$. Thus, at least $\left(\frac{\sigma^2}{D}\right)^{n0.5}$ codewords are required to cover the sphere the sequence lie in.

4.3.1.3 (iii)

Thus, at least $\frac{n}{2}\log(\frac{\sigma^2}{D})$ bits are required to describe the sequence to distortion D and the average usage of bits per element of the sequence is $\frac{1}{2}\log(\frac{\sigma^2}{D})$.

4.3.2 (b)

Suppose we have K Gaussian sources to compress with variances σ_k^2 for $k = 1, ..., K$. Then the optimization problem becomes:

$$\min_{D_1,\ldots,D_K} \sum_{k=1}^{K} \log\left(\frac{\sigma_k^2}{D_k}\right)$$
$$s.t. \sum_{k=1}^{K} D_k \leq D$$
$$D_k \leq \sigma_k^2$$

Taking the Lagrangian, we get

$$\mathcal{L}(D_k, \lambda, \mu_k) = \sum_{k=1}^{K} \log\left(\frac{\sigma_k^2}{D_k}\right) + \lambda(\sum_{k=1}^{K} D_k - D) + \sum_{k=1}^{K} \mu_k(D_k - \sigma_k^2)$$

Taking the derivative with respect to D_k, we get that

$$\frac{\partial L}{\partial D_k} = -\frac{1}{D_k} + \lambda + \mu_k$$

By KKT conditions then, we know the following:

- $D_k = \frac{1}{\lambda + \mu_k}$
- if $D_k < \sigma_k^2$, then $\mu_k = 0$
- if $\sum_k D_k < D$, then $\lambda = 0$

Hence, we can conclude that, for all k such that $D_k < \sigma_k^2$, D_k is constant.

To interprete this solution as "reverse water-filling", note that if σ_k^2 is too small, then we set $D_k = \sigma_k^2$, otherwise, we set D_k as all constant.