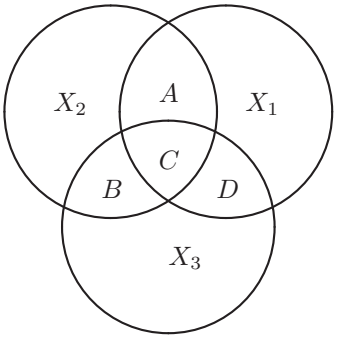
CS177, Homework 4

Due: Wednesday, April 29

This homework has no electronic component. Please write neatly and turn in your homework in class on Wednesday.

Reading

Reading this week covers entropy, mutual information, and coding. Olofsson again does not cover these subjects well; please review your notes and see the online readings posted on the webpage.



Problem 1:

Consider the (7,4) Hamming code discussed in lecture, and shown above. A valid codeword is one which satisfies all the parity check relationships implied by the Venn diagram, i.e., each circle has even parity.

1. Writing your solution as $[ABCDX_1X_2X_3]$, encode $[ABCD] = [0001]$.

2. Is [0110011] a valid codeword? If not, find the closest valid codeword.

3. Show that this code is *linear*, i.e., that for any two codewords V_1, V_2, the binary exclusive or $V_1 \oplus V_2$ is also a valid codeword.

4. The *Hamming distance* between two binary vectors is the number of bits (positions) at which they differ. Show that every codeword has Hamming distance at least 3 from the all-zeros codeword. (Combined with linearity, this shows that all valid codewords are at least distance 3 from each other.)

5. We saw in class that a single parity bit can be used to detect single errors, and that the (7,4) Hamming code can be used to actually correct single errors. Alternatively, the (7,4) Hamming code can be used to detect up to two bit errors in the sequence. Using the Hamming distance, argue why these facts are the case. How many errors can be detected by a code in which each codeword is Hamming distance k apart? How many errors can be corrected?

6. The *repetition code* consists of simply repeating the data, e.g., $[ABCDABCD]$. Why is this code not very good? If, each time we send a bit, we make a mistake with probability $p = .1$, what is the probability of making an undetected error? Compare this to the probability for the (7,4) Huffman code used only for detecting errors (no correction).

Problem 2

Consider building a model for the following fault diagnosis problem. The class variable C represents the health of a disk drive: $C = 1$ means it is operating normally, and $C = 0$ means it is in a failed state. When the drive is running, it continuously monitors itself using a temperature and shock sensor. It records two binary features, X and Y, where each takes values 0 or 1. X indicates whether the drive has been subject to shock (e.g. dropped), and Y is whether the drive has ever been above 70°C. The following is the joint pmf of all three variables:

x	y	c	$p(x, y, c)$
0	0	0	0.1
0	1	0	0.2
1	0	0	0.2
1	1	0	0.1
0	0	1	0.25
0	1	1	0.1
1	0	1	0.05
1	1	1	0.0

In the following problems for the cases where a numerical value is required, provide an equation (or several equations) to show how you calculated the answer and the numerical value of the answer.

1. What is the probability $p(C = 1)$?
2. What is the probability $p(C = 0|X = 1, Y = 0)$?
3. What is the probability $p(X = 0, Y = 0)$?
4. What is the probability $p(C = 0|X = 0)$?
5. Are X and Y independent? Justify your answer.
6. Are X and Y conditionally independent given C? Justify your answer.

Problem 3

Recall that in class we defined the entropy (measured in bits) of a discrete random variable X by

$$H(X) = E_X\left[\log_2 \frac{1}{p(X)}\right] = \sum_x p(x) \log_2 \frac{1}{p(x)}$$

and the mutual information between two discrete random variables X, Y as the change in entropy of X once Y is observed, and showed

$$I(X, Y) = \sum_{x,y} p(x, y) \log_2 \frac{p(x, y)}{p(x)p(y)}$$

Using the joint distribution from Problem 2, please answer the following:

1. Compute $H(X)$ and $H(Y)$, the entropy of the sensor variables. Which variable has more uncertainty?
2. Compute the mutual information between each feature and the drive health, $I(X, C)$ and $I(Y, C)$.
3. Which single diagnostic feature, temperature or shock, would you measure to get the most information about the health of the drive C?
4. Is X independent of C? What would $I(Y, C)$ be if Y and C were independent variables? (Hint: you should be able to answer this question based on the formula for $I(\cdot)$ without having to compute any probabilities.)

Problem 4

Suppose now we wanted to transmit the state of the drive over the network. The drive can be in one of 8 different states $S \in \{s_1, s_2, \ldots, s_8\}$ given by the combination of possible values of X, Y and C. For example, $S = s_1 = (X = 0, Y = 0, C = 0)$. These states are listed in the table in problem 2 along with their probabilities.

1. If we use a fixed length code for S, what is the expected codeword length?
2. Construct a Huffman code for S based on the probabilities given in Problem 2. Derive the code graphically using the technique described in class of building a tree by recursively merging the lowest probability symbols.
3. What is the expected codeword length for your Huffman code?
4. Calculate the entropy of the hard drive state $H(S)$. Briefly comment on the differences between the entropy, the length of a fixed length code and that of the Huffman code.

Problem 5

Suppose you are given six bottles of (the same kind of) wine. You know that precisely one bottle has gone bad (turned to vinegar). From inspection of the exterior, you estimate the probability p_i that bottle i is the bad one as

$$(p_1, \ldots, p_6) = (\frac{8}{23}, \frac{6}{23}, \frac{4}{23}, \frac{2}{23}, \frac{2}{23}, \frac{1}{23}).$$

The bad wine can easily be detected by taste.

1. Suppose you taste the wines one at a time. Choose the order of tastings to minimize the expected number of tastings required to determine the bad wine. (Remember, after you have tasted 5 wines, you don't have to actually taste the 6th.)
   (a) What is the expected number of tastings required?
   (b) Which bottle should you taste first?
2. Knowing all about Huffman coding, you revise your strategy. Instead of one at a time, you decide to try mixing several wines together and tasting the mixture. You proceed, mixing and tasting, until you are sure which wine is bad.
   (a) What is the minimum expected number of tastings required to determine the bad wine?
   (b) What mixture should be tasted first?