

CS258 Information Theory Homework 1

Zhou Litao 518030910407 F1803016

March 5, 2020

Exercise 1 (Coin flips) A fair coin is flipped until the first head occurs. Let X denote the number of flips required.

1. Find the entropy $H(X)$ in bits. The following expressions may be useful:

$$\sum_{n=0}^{\infty} r^n = \frac{1}{1-r}, \quad \sum_{n=0}^{\infty} nr^n = \frac{r}{(1-r)^2} \quad (1)$$

2. A random variable X is drawn according to this distribution. Find an “efficient” sequence of yes-no questions of the form, “Is X contained in the set S ?” Compare $H(X)$ to the expected number of questions required to determine X .

Solution.

1. The distribution of X is

$$P(X = k) = \left(\frac{1}{2}\right)^k, k = 1, 2, \dots$$

Hence the entropy in bits is

$$\begin{aligned} H(X) &= - \sum_{n=1}^{\infty} \left(\frac{1}{2}\right)^n \log_2 \left(\frac{1}{2}\right)^n \\ &= \sum_{n=1}^{\infty} n \left(\frac{1}{2}\right)^n = 2. \end{aligned} \quad (2)$$

2. Since the probability of X to be a number x will decrease as x grows, an “efficient” way of asking questions about X will be like ‘Is X contained in the set S_i ?’ where $S_i = \{1, 2, \dots, i\}$ for $i \in \mathbf{R}$.

If the answer is yes, then the question is done by concluding X is the current i . Otherwise, the next question will be asked. Let Y denote the number required to determine X . Clearly, $Y = X$.

Thus the expected number of questions is

$$E(Y) = \sum_{k=1}^{\infty} k \left(\frac{1}{2}\right)^k = 2.$$

The result is equal to $H(X)$.

□

Exercise 2 (Zero conditional entropy) Show that if $H(Y|X) = 0$, then Y is a function of X [i.e., for all x with $p(x) > 0$, there is only one possible value of y with $p(x, y) > 0$].

Proof. By condition we have

$$H(Y|X) = \sum_x p(x) H(Y|X = x) = 0$$

Note that $p(x) > 0$, thus for any x , we have $H(Y|X = x) = 0$.

It follows that when x is determined, the distribution of Y is a single value.

That is to say, Y is a function of X .

□

Exercise 3 (Coin weighing) Suppose that one has n coins, among which there may or may not be one counterfeit coin. If there is a counterfeit coin, it may be either heavier or lighter than the other coins. The coins are to be weighed by a balance.

1. Find an upper bound on the number of coins n so that k weighings will find the counterfeit coin (if any) and correctly declare it to be heavier or lighter.
2. (Difficult) What is the coin-weighing strategy for $k = 3$ weighings and 12 coins?

Solution.

1. For a single weighing, there are three cases of results, namely left, right or equal. Therefore k weighings can be represented by at most 3^k situations.

For n coins, there are at most $2n + 1$ different cases, namely, one of the coins is heavier or lighter, or there exist no counterfeit coins.

Theoretically, if we can find an injection from $2n + 1$ cases to 3^k test results, we can determine the counterfeit directly from the information given by k pieces of facts. Hence, a necessary condition for this problem will be

$$2n + 1 \leq 3^k.$$

It follows that $\frac{3^k - 1}{2}$ is an upper bound of n .

2. In this part we should construct a mapping from $2 \times 12 + 1 = 25$ cases to 3^3 test results. A natural idea is to use the ternary number.

Let $\{-1, 0, 1\}$ be the test result for every weighing, where -1 indicates the left side is heavier, and 1 indicating the heavier right side. Then the test result can be represented a ternary number.

In order to maximize the information this ternary number represents, we should arrange the order of every weighing carefully. Here we propose several principles to follow.

- The numbers of coins on both side of the weighing should be equal. Otherwise no extra information will be gained from this weighing.
- All coins should be weighed at least once. Otherwise no information about the coin will be gained from three weighings.
- The weighing arrangement for every coin should be unique. That is to say, if coin A is weighed three times, no other coins should be weighed also three times. Otherwise we can't distinguish between these two coins, indicating that the information we gain from the three weighings is not sufficient.

It follows naturally from the third rule to construct a ternary mapping from digits $\{-1, 0, 1\}$ to 12 coins in the following table, where -1 represents that the coin is weighed on the left side, 1 represents the coin should be weighed on the right side.

Coin	1	2	3	4	5	6	7	8	9	10	11	12
3^0	1	-1	0	1	-1	0	1	-1	0	1	-1	0
3^1	0	1	1	1	-1	-1	-1	0	0	0	1	1
3^2	0	0	0	0	1	1	1	1	1	1	1	1

Table 1: First Try in Building Weighing Strategy

In this way, a unique weighing strategy is generated for each coin. However, this schedule violates the first rule. In the second weighing, for example, there are two extra coins on the right. To address this problem, we can simply reverse some of the coins.

Coin	1	2	3	4	5	6	7	8	9	10	11	12
3^0	1	-1	0	1	-1	0	-1	-1	0	1	1	0
3^1	0	1	1	1	-1	-1	1	0	0	0	-1	-1
3^2	0	0	0	0	1	1	-1	1	-1	1	-1	-1

Table 2: Actual Weighing Strategy

Now we can build the weighing roadmap.

- For every weighing, put the -1 -labelled coins on the left and 1 -labelled to the right.
- Record the weighing result as -1 to be leftward-sloping, 0 to be balanced and 1 to be rightward-sloping.
- Combine the number and check the table 2. $(0, 0, 0)$ indicates that there are no counterfeits. If the vector matches the i -th column, then coin i is heavier. Or if the component-wise negation of the vector matches the i -th column, then coin i is lighter.

□