

## 1 The Asymptotic Equipartition Property

Many of the results in information theory have to do with encoding long sequences of symbols  $X_1, \dots, X_n$ . In what follows we assume for simplicity that these sets are generated IID. We saw an instance of that in block loss-less coding. The reason why this is a useful setting is that such long sequences are similar to each other. For example if we sample IID a biased coin ( $[0.1, 0.9]$ ) then all long sequences will have roughly this proportion of heads and tails. Intuitively it seems obvious that all these long sequences will have the same probability. On the other hand, a sequence like  $0, 0, 0, 0, 0, 0, 0, \dots$  could be drawn but will have very low probability compared to sequences with the above “typical” sequences.

Formally, the above intuition is captured in a result known as the Asymptotic Equipartition Property. We begin by looking at the probability of long sequences (assume we use  $\log 2$  throughout).

$$\frac{1}{n} \log p(x_1, \dots, x_n) = \frac{1}{n} \sum_i \log p(x_i) = \frac{1}{n} \sum_x n_x \log p(x) \rightarrow \sum_x p(x) \log p(x) = -H(X) \quad (1)$$

In other words we have that  $\log p(x_1, \dots, x_n) \rightarrow 2^{-nH(X)}$ . As noted above, clearly there are sequences for which this is not the probability. The AEP theorem tells us that with probability approaching one we will draw a sequence such that  $\log p(x_1, \dots, x_n) \approx 2^{-nH(X)}$ . Such sequences are called typical sequences. Since these sequences are equiprobable and contain all the probability mass, the number of such sequences is approximately  $2^{nH(X)}$ . Note that there are  $2^{n \log_2 |X|} = |X|^n$  sequences overall, so there are generally far fewer typical sequences than possible ones (when are there no non-typical sequences?).

This can be used as a simple argument for the block coding result we showed last time (assume we are in the binary code case for convenience). For large  $n$  we encode each of the typical sequences with  $nH(X)$  bits. Thus we have a unique codeword per typical sequence and there will be no loss in encoding them. The other non-typical sequences will appear with probability approaching zero and thus we will almost never see them. (note this is a bit different from the setting we had before, since we allow errors in decoding here, but these turn out to be probability zero events).

## 2 The Data Processing Inequality

Begin by recalling useful facts about entropy and information:

- Conditioning reduces entropy  $H(X|Y) \leq H(X)$  (a consequence of the non-negativity of information).

- Entropy and information chain rules.
- If  $X$  is a deterministic function of  $Y$  then  $H(X|Y) = 0$ .

Suppose we have a variable  $X$ . We then apply (a possible stochastic) function  $f$  to it to generate a variable  $Y = f(X)$ , and then  $Z = g(Y)$ . Our intuition tells us that  $Z$  cannot provide more information about  $X$  than  $Y$  can, or in other words:

$$I(X; Z) \leq I(Y; Z) \quad (2)$$

This is indeed true. To state it more formally, we need some definitions:

**Definition:** We say that three variables  $X, Y, Z$  form a Markov chain  $X \rightarrow Y \rightarrow Z$  if their joint probability can be written as:

$$p(x, y, z) = p(x)p(y|x)p(z|y) \quad (3)$$

Note this implies that

$$p(x, z|y) = \frac{p(x, y, z)}{p(y)} = \frac{p(x, y)p(z|y)}{p(y)} = p(x|y)p(z|y) \quad (4)$$

Thus  $X$  and  $Z$  are conditionally independent given  $Y$  and  $I(X; Z|Y) = 0$ .

**Data Processing Inequality:** If  $X \rightarrow Y \rightarrow Z$  is a Markov chain, then  $I(X; Z) \leq I(Y; Z)$  with equality iff  $I(X; Y|Z) = 0$ .

**Proof:** Recall the chain rule for information:

$$I(X_1, \dots, X_n; Y) = \sum_{i=1}^n I(X_i; Y | X_{i-1}, \dots, X_1) \quad (5)$$

Expand  $I(X; Y, Z)$  in two ways using the chain rule for information.

$$\begin{aligned} I(X; Y, Z) &= I(X; Z) + I(X; Y|Z) \\ I(X; Y, Z) &= I(X; Y) + I(X; Z|Y) = I(X; Y) \end{aligned}$$

We have:

$$I(X; Z) + I(X; Y|Z) = I(X; Y) \quad (6)$$

An important special case is when  $Z = g(Y)$ , so that  $X \rightarrow Y \rightarrow Z$  is a Markov chain and the inequality holds.

The data processing inequality gives us a simple method to lower bound the information between variable  $X$  and  $Y$ . Assume we cannot calculate the joint probability  $p(X, Y)$  since for example  $Y$  is some huge cardinality variable (e.g., the neural response). Assume that  $X$  is a relatively small variable (e.g., one of eight directions an animal moves to). Here are two examples of how we might use the inequality:

- Quantization - Lets say  $Y$  is a spike train. We can choose to represent it via the total number of spikes it has  $Z = f(Y)$ . Lets say the train can have at most 10 spikes (or if it has more, we clip  $Z$  to 10). We can get a pretty reliable estimate of  $p(X, Z)$ , since both  $X, Z$  are small. We can then use  $I(X; Z)$  as a lower bound on  $I(Y; Z)$ .
- Classification - We can use our favorite classifier to build a function  $\hat{X} = f(Y)$  that tries to predict  $X$ . Then  $I(X; \hat{X})$  is a lower bound on the information.

### 3 Relation between information and classification error

Suppose we have a variable  $Y$  and a variable  $X$ . One way of quantifying their dependence is to measure the error we would get by predicting  $Y$  from  $X$  via a MAP predictor (maximizing  $p(y|x)$ ). This error is referred to as the Bayes error. Denote it by  $e_B$ . How is  $e_B$  related to mutual information and entropy? Turns out we can both upper and lower bound  $e_B$  via the mutual information.

**Claim:** The Bayes error is upper bounded as follows (Raviv and Heller 70):

$$e_B \leq \frac{1}{2}H(Y|X) \quad (7)$$

**Claim:** The Bayes error satisfies (Fano):

$$h(e_B) + e_B \log(|Y| - 1) \geq H(Y|X) \quad (8)$$

When  $|Y| = 2$  this yields  $h(e_B) \geq H(Y|X)$ .

**Proof:** Define the variable

$$E = \begin{cases} 1 & \hat{Y} \neq Y \\ 0 & \hat{Y} = Y \end{cases} \quad (9)$$

Note that  $p(E = 1) = e_B$ . Then write  $H(E, Y|X)$  in two different ways using the entropy chain rule:

$$\begin{aligned} H(E, Y|X) &= H(Y|X) + H(E|X, Y) = H(Y|X) \\ H(E, Y|X) &= H(E|X) + H(Y|E, X) \end{aligned}$$

First, note that  $H(E|X) \leq H(E) = h(e_B)$ . Also:

$$\begin{aligned} H(Y|E, X) &= p(E = 0)H(Y|X, E = 0) + p(E = 1)H(Y|X, E = 1) \\ &\leq e_B \log(|Y| - 1) \end{aligned}$$

Putting it together we have:

$$H(Y|X) \leq h(e_B) + e_B \log(|Y| - 1) \quad (10)$$

## 4 Channel Coding

One of the key problems in communication is how to transfer information from one point to another. Information is propagated physically via some quantity that can travel the medium between the sender and receiver (e.g., sound waves, radio waves, hard disk etc). However, in many cases we cannot guarantee error-less transmission. For example:

- Speech might be degraded because of external noise, telephone line limitations etc
- Cellphone transmission might experience interference
- The firing of neurons is not reproducible
- Hard disk failures

The key formal object in these cases is a channel, which takes as input symbols of type  $X$  and output symbols of type  $Y$ . The channel is characterized by the distribution  $p(y|x)$ , which will typically be non-deterministic. In what follows we will assume that the channel is memory-less, meaning the probability of an output  $y^n$  given an input  $x^n$  is:

$$p(y^n|x^n) = \prod_{i=1}^n p(y_i|x_i) \quad (11)$$

Some examples of channels are:

- Binary symmetric channel.  $\mathcal{X}, \mathcal{Y} = \{0, 1\}$ . And  $p(y|x) = \mathcal{I}[x = y](1 - \epsilon) + \mathcal{I}[x \neq y]\epsilon$ .
- Binary erasure channel.  $\mathcal{Y} = \{0, 1, e\}$  where  $e$  denotes that the input turned into a value which we cannot decode as either zero or one. The matrix  $p(y|x)$  is:

$$\begin{bmatrix} 1 - \epsilon & 0 & \epsilon \\ 0 & 1 - \epsilon & \epsilon \end{bmatrix} \quad (12)$$

Formally, a communication problem has several components:

- A set of messages  $w \in \{1, \dots, M\}$  we want to send.
- A channel over which we can transmit. The input to the channel is  $\mathcal{X}$  and its output is  $\mathcal{Y}$ .
- The probabilistic description of the channel  $p(y|x)$ .

We wish to transmit messages  $\mathcal{M}$  such that the receiver on the other end of the channel knows what we sent. Thus we are interested in a coding scheme made up of the following:

**Definition:** An  $(M, n)$  code is comprised of:

- A set of messages  $w \in \{1, \dots, M\}$  we want to send.
- A encoding function  $f : \{1, \dots, M\} \rightarrow \mathcal{X}^n$ . That is, a mapping between the messages and code-words of  $n$  symbols in  $\mathcal{X}$ .
- A decoding function  $g : Y^n \rightarrow \{1, \dots, M\}$ . That is a mapping between the output of the channel and the original set of messages.
- Noisy typewriter: each letter (e.g.,  $a$ ) is mapped uniformly to itself or the next letter (e.g.  $b$ ). And  $z$  can be mapped to  $z, a$  with 0.5.

**Definition:** The rate of an  $(M, n)$  code is:

$$R = \frac{\log_2 M}{n} \quad (13)$$

The rate is measured in bits per symbol. The idea is this: suppose we had  $M = 8$ . In a noise-less binary channel we could have encoded  $M$  with three bits (but not less). If our code manages to encode these messages with  $n$  symbols, it has  $\frac{3}{n}$  bits per symbol. Note that if we have a error-free code then necessarily:

$$\frac{\log_2 M}{n} \leq \frac{\log_2 |\mathcal{X}|^n}{n} = \log_2 |\mathcal{X}| \quad (14)$$

Specifically, in a binary input channel the rate is always less than one.

The communication works by first choosing a message, then encoding it via  $f$ , sending it over the channel, obtaining an output  $y^n$  and decoding it via  $g$ . The key problem of communication in a noisy channel is how to transmit as many messages ( $M$ ) as possible by using as few channel transmissions as possible ( $n$ ), where we would like the messages to be decoded without error.

It's pretty clear we generally cannot have error-less transmission by using a finite  $n$ . For example, say  $M = 2$  and a BSC. A  $(2, 1)$  would result in erroneous decoding because we will not be able to correct the flips made by the channel. A natural choice is to use a  $(2, n)$  code where each inputs get either the sequence of all zeros or all ones, and we decode by majority. As  $n \rightarrow \infty$  this will result in zero error. However, we have payed a high price for having zero error.

**Definition:** For a given channel, a rate  $R$  is said to be achievable if there exists a sequence of  $(\lceil 2^{nR} \rceil, n)$  codes such that the probability of decoding any message incorrectly tends to zero.

The repetition code above has rate zero since the number of messages does not grow with  $M$ . Shannon's striking theorem tells us that non-zero rates are achievable, and goes further to characterize the maximum achievable rate.

**Definition:** The capacity of a channel is defined as:

$$C = \max_{p(x)} I(X; Y) = \max_{p(x)} \sum_{x,y} p(y|x)p(x) \log \frac{p(y|x)}{\sum_x p(y|x)p(x)} \quad (15)$$

**Shannon's channel coding theorem:** All rates  $R$  that satisfy  $R < C$  are achievable, and a rate is achievable only if  $R \leq C$ .

Here's a rough intuition for why this is true: Say you have a code  $f(w) \in X^n$  for a message  $w$ . Denote it by  $x^n(i)$ . If we broadcast this word, the output of the channel will be distributed as  $p(y^n|x^n(i))$ . The AEP tells us (very roughly...) that asymptotically there will be about  $2^{nH(Y|X)}$  typical sequences in this set. We need to partition the set  $2^{nH(Y)}$  into disjoint sets of size  $2^{nH(Y|X)}$ , and the maximum number of these is  $2^{nI(X;Y)}$  implying that the maximum rate is the maximum value of  $I(X; Y)$ .

The actual proof proceeds by fixing  $p(x)$  and choosing a code as an IID set of  $n$  symbols from it (for each of the  $2^{nR}$  words). Turns out that at least one of these codes will have a small probability of error.

**Example: Noisy typewriter.** Let us first calculate its capacity.

$$I(X; Y) = H(Y) - H(Y|X) = H(Y) - 1 \quad (16)$$

The maximum value for  $H(Y)$  is  $\log_2 26$ , which we will get by choosing  $p(x) = \frac{1}{26}$  implying  $p(y) = \frac{1}{26}$ . Thus we have:

$$\max_{p(x)} I(X; Y) = \log_2 26 - 1 = \log_2 13 \quad (17)$$

Lets see that we can indeed achieve this rate for errorless transmission. First, note that we can encode  $M = 13$  messages with one symbol without error. Simply encode  $f(1) = a, f(2) = c, \dots, \dots, f(13) = y$ . And decode  $g(b, a) = a, g(c, d) = c, \dots$ . Similarly we can encode  $M = 13^n$  messages with  $n$  symbols without error yielding a rate:

$$R = \frac{\log_2 13^n}{n} = \log_2 13 \quad (18)$$

The theorem tells us we can't do any better.

**Example: BSC.**

$$H(Y|X) = p_0 h(\epsilon) + p_1 h(\epsilon) \quad (19)$$

$$I(X; Y) = H(Y) - H(Y|X) = H(Y) - h(\epsilon) \leq 1 - h(\epsilon) \quad (20)$$

Equality is achieved with uniform  $p(x)$  (yielding uniform  $p(y)$ ).

## 5 Proof that $R > C$ is unachievable

We first need the following lemma.

**Lemma:** Say we have a channel  $p(y|x)$  and we send a sequence  $X_1, \dots, X_n$  through it. Then  $I(X_1, \dots, X_n; Y_1, \dots, Y_n) \leq nC$  where  $C$  is the capacity of the channel.

**Proof:**

$$\begin{aligned}
 I(X^n; Y^n) &= H(Y^n) - H(Y^n|X^n) \\
 &= H(Y^n) - \sum_i H(Y_i|Y_1, \dots, Y_{i-1}, X^n) \\
 &= H(Y^n) - \sum_i H(Y_i|X_i) \\
 &= \sum_i H(Y_i|Y_1, \dots, Y_{i-1}) - \sum_i H(Y_i|X_i) \\
 &\leq \sum_i H(Y_i) - \sum_i H(Y_i|X_i) = \sum_i I(X_i; Y_i) \leq nC
 \end{aligned}$$

We can proceed to prove the claim. Denote by  $W$  the send message. There are  $2^{nR}$  possible values for  $W$  (since we are assuming we are sending with a rate  $R$ ).  $W$  is encoded into a sequence  $X^n(W)$  and sent over the channel to produce  $Y^n$ . Finally we decode the received message into a variable  $\hat{W}$ . We are interested in the best possible error of  $\hat{W}$  w.r.t.  $W$ . This is the Bayes error  $e_B$ . Using Fano's inequality we know that:

$$h(e_B) + e_B \log(|W| - 1) \geq H(W|\hat{W}) \quad (21)$$

Which we can further bound via:

$$1 + e_B nR \geq h(e_B) + e_B \log(|W| - 1) \geq H(W|\hat{W}) \quad (22)$$

Now we use:

$$\begin{aligned}
 H(W|\hat{W}) &= H(W) - I(W; \hat{W}) \\
 &= nR - I(W; \hat{W}) \\
 &\geq nR - I(X^n(W); Y^n) \\
 &\geq nR - nC = n(R - C)
 \end{aligned}$$

Putting it together we have:

$$1 + e_B nR \geq n(R - C) \quad (23)$$

So:

$$e_B \geq \frac{R - C}{R} - \frac{1}{n} = 1 - \frac{C}{R} - \frac{1}{n} \quad (24)$$

This implies that when  $R > C$  as  $n \rightarrow \infty$  the error is strictly greater than zero (exercise: why can't we get zero error for finite  $n$ ?).