EE376A Information Theory Lecture 1 - 01/06/2015 and 01/08/2015

Lecture 1

Lecturer: Tsachy Weissman *Scribe: Aran Nayebi, Clara Fannjiang, Chuanqi Shen*

1 Illustrative Examples of Information Theory

Lossless Compression

Consider a source that emits a sequence of symbols $U_1, U_2, \ldots$ with $U_i \in \{a, b, c\}$. The U_i are i.i.d (independently and identically distributed) according to the probability mass function

$$P(U = a) = 0.7$$
$$P(U = b) = P(U = c) = 0.15$$

Our task is to encode the source sequence into binary bits (1s and 0s). How should we do so?

The naive way is to use two bits to represent each symbol, since there are three possible symbols. For example, we can use 00 to represent a, 01 to represent b and 10 to represent c. This scheme has an expected codeword length of 2 bits per source symbol. Can we do better? One natural improvement is to try to use fewer bits to represent symbols that appear more often. For example, we can use the single bit 0 to represent a since a is the most common symbol, and 10 to represent b and 11 to represent c since they are less common. Note that this code satisfies the prefix condition, meaning no codeword is the prefix of another codeword, which allows us to decode a message without any ambiguity. Thus, if we see the encoded sequence, 001101001101011, we can quickly decode it as follows:

$$\underbrace{0}_{a}\ \underbrace{0}_{a}\ \underbrace{11}_{c}\ \underbrace{0}_{a}\ \underbrace{10}_{b}\ \underbrace{0}_{a}\ \underbrace{11}_{c}\ \underbrace{0}_{a}\ \underbrace{10}_{b}\ \underbrace{11}_{c}$$

If we use this encoding scheme, then $\overline{L}$, which denotes the expected number of bits we use per source symbol, is

$$\overline{L} = 1 \times P(U = a) + 2 \times (P(U = b) + P(U = c)) = 1 \times 0.7 + 2 \times (0.15 + 0.15) = 1.3.$$

This is a significant improvement over our first encoding scheme. But can we do even better? A possible improvement is to encode two values at a time instead of encoding each value individually. For example, the following table shows all the possibilities we can get if we look at 2 values, and their respective probabilities (listed in order of most to least likely pairs). A possible encoding scheme is also given.

source symbols	probability	encoding
aa	0.49	0
ab	0.105	100
ac	0.105	111
ba	0.105	101
ca	0.105	1100
bb	0.0225	110100
bc	0.0225	110101
cb	0.0225	110110
cc	0.0225	110111

Note that this scheme satisfies the two important properties: 1) the prefix condition and 2) more common source symbol pairs have shorter codewords. If we use the above encoding scheme, then the expected number of bits used per source symbol is

$$\overline{L} = 0.5 \times (0.49 \times 1 + 0.105 \times 4 + 0.105 \times 3 \times 3 + 0.0225 \times 6 \times 4) = 1.1975.$$

It can be proven that if we are to encode 2 values at a time, the above encoding scheme achieves the lowest average number of bits per value (*wink* Huffman encoding *wink*).

Generalizing the above idea, we can consider a family of encoding schemes indexed by an integer k. Given an integer k, we can encode k values at a time with a scheme that satisfies the prefix condition and assigns shorter codewords to more common symbols. Under some optimal encoding scheme, it seems reasonable that the expected number of bits per value will decrease as k increases.

We may ask, what is the best we can do? Is there a lower bound on $\overline{L}$? Shannon proved that given any such channel, the best we can do is $H(U)$, which is called the **source entropy** of the channel. By definition, the source entropy is

$$H(U) \triangleq \sum_{u \in U} p(u) \log_2 \frac{1}{p(u)} \tag{1}$$

Thus, Shannon proved the following statement[1],

Theorem 1. *$\forall$ families of schemes, $\overline{L} \geq H(U)$.*

For our example, the lower bound is thus

$$0.7 \times \log_2 \frac{1}{0.7} + 2 \times 0.15 \times \log_2 \frac{1}{0.15} \approx 1.181$$

We will also show an upper bound, namely,

Theorem 2. *$\forall \varepsilon > 0$, $\exists$ family of schemes, such that $\overline{L} \leq H(U) + \varepsilon$.*

Binary Source through Binary Channel

Suppose we have a source that emits a stream of bits $U_1, U_2, \ldots$ The $U_i \in \{0, 1\}$ are i.i.d. Bernoulli random variables with parameter 0.5, or fair coin flips.

We want to transmit the bits U_i through a channel. Suppose the bits that are transmitted are $X_1, X_2, \ldots$ The channel is noisy and flips each bit with probability $q < 1/2$. Therefore, if $Y_1, Y_2, \ldots$ is the sequence of bits that we receive, we have

$$Y_i = X_i \oplus W_i, W_i \sim \mathbf{Ber}(q)$$

where $\oplus$ is the XOR operator.

We want to know how accurate we can be when transmitting the bits. The simplest approach is to let $X_i = U_i$, and to decode the received bit by assuming $Y_i = X_i$. Let p_e be the probability of error per source bit. Then in this case, $p_e = q < 1/2$.

Can we decrease p_e? One approach may be to use repetition encoding, i.e., send each bit k times for some k, and then decode the received bit as the value that appeared most among the k received symbols. For example, if $k = 3$, then p_e is simply the probability that the channel flipped 2 or more of the bits, which is

$$p_e = 3(1 - q)q^2 + q^3 < q.$$

[1]Note that the statements of the theorems here will be informal; they will be made rigorous in later lectures.

However, we need to send 3 times as many bits. To quantify this, we introduce the notion of **bit rate**, which is the ratio of the number of bits sent to the units of channel space used. For this scheme, our bit rate is $\frac{1}{3}$, whereas our bit rate in the previous example was 1.

Generalizing the above example, we see that as we increase k, our error rate p_e will tend to 0, but our bit rate (which is $1/k$) tends to 0 as well. Is there some scheme that has positive bit rate and yet allows us to get reliable communication (error rate tends to 0)? Again, Shannon provides the answer.

Theorem 3. *$\exists R > 0$ and $\exists$ family of schemes with bit rate $> R$ satisfying $p_e \to 0$.*

In fact, we use the theorem to define the **channel capacity** of a channel, which is the largest bit rate that still allows for reliable communication. The channel capacity of a channel with bit-flipping probability q is

$$1 - H(X), X \sim \mathbf{Ber}(q). \tag{2}$$

Moreover, if we let $X \sim \mathbf{Ber}(q)$ and $Y \sim \mathbf{Ber}(p_e)$, we will see that a bit rate r such that

$$r < \frac{1 - H(X)}{1 - H(Y)}, \tag{3}$$

is **achievable**, whereas

$$r > \frac{1 - H(X)}{1 - H(Y)}, \tag{4}$$

is **unachievable**.

Lossy Compression

Suppose we have a source that emits a sequence of values $U_1, U_2, \ldots$, where each U_i is i.i.d. according to $U \sim \mathcal{N}(0, \sigma^2)$. Suppose we want to encode the source using one bit per value. Since we are representing continuous variables with discrete bits, we are employing lossy compression. Can we come up with a scheme that reconstructs the original signal as accurately as possible, based on the bits sent?

Let $B_1, B_2, \ldots$ be the bits sent. One natural scheme is to set

$$B_i = \begin{cases} 1 & \text{if } U_i \geq 0 \\ 0 & \text{if } U_i < 0. \end{cases}$$

After receiving the bits, let $V_1, V_2, \ldots$ be the reconstructed values. The **distortion** of the scheme is defined as

$$D \triangleq \mathbb{E}[(U_i - V_i)^2] \tag{5}$$

The optimal estimation rule for minimum mean squared error is the conditional expectation. Therefore, to minimize distortion, we should reconstruct via $V_i = \mathbb{E}[U_i \mid B_i]$. This results in

$$\begin{aligned} D &= \mathbb{E}[(U_i - V_i)^2] \\ &= Var(U_i \mid B_i) \\ &= 0.5 \times Var(U_i \mid B_i = 1) + 0.5 \times Var(U_i \mid B_i = 0) \text{ (because } U \text{ is symmetric)} \\ &= Var(U_i \mid B_i = 1) \\ &= \mathbb{E}[U_i^2 \mid B_i = 1] - (\mathbb{E}[U_i \mid B_i = 1])^2 \\ &= \sigma^2 \left(1 - \frac{2}{\pi}\right) \\ &\approx 0.363\sigma^2. \end{aligned}$$

We will see in fact that $0.363\sigma^2$ can be improved considerably, as such:

Theorem 4. *Consider a Gaussian memoryless source with mean μ and variance σ^2. $\forall \varepsilon > 0$, $\exists$ family of schemes such that $D \leq \sigma^2/4 + \varepsilon$. Moreover, $\forall$ families of schemes, $D \geq \sigma^2/4$.*

Additive White Gaussian Noise (AWGN) Channel

Suppose we have a source that emits a sequence of bits $U_1, U_2, \ldots, U_N$, where each U_i is i.i.d. according to $U \sim \mathbf{Ber}(\frac{1}{2})$.

However, we can only transmit real numbers $X_1, X_2, \ldots, X_n$. Also, the channel contains some noise. Specifically, if $Y_1, Y_2, \ldots, Y_n$ is the sequence of values we receive, we have

$$Y_i = X_i + N_i, N_i \sim \mathcal{N}(0, \sigma^2)$$

The **rate of transmission** is the ratio $\dfrac{N}{n}$ (which is the ratio of the number of source bits to the number of uses of the channel). We want to develop a scheme so that we can reliably reconstruct U_i from the given Y_i. One way, if we have no usage power constraint, is to make X_i a large positive value if $U_i = 1$ and X_i a large negative value if $U_i = 0$. In this manner, the noise from N_i will be trivial relative to the signal magnitude, and will not impact reconstruction too much. However, suppose there is an additional constraint on the average power of the transmitted signal, such that we require

$$\frac{1}{n} \sum_{i=1}^{n} X_i^2 \leq p,$$

for a given value p. In fact, we will see that

Theorem 5. *If the rate of transmission is $< \frac{1}{2} \log_2 \left(1 + \frac{p}{\sigma^2}\right)$, then $\exists$ family of schemes that communicate reliably. And if the rate of transmission is $> \frac{1}{2} \log_2 \left(1 + \frac{p}{\sigma^2}\right)$, then there is no family of schemes which communicates reliably.*

The ratio $\dfrac{p}{\sigma^2}$ is referred to as the signal-to-noise ratio (SNR).