Lecture 19: Shannon limit

ENGR 76 lecture notes — June 4, 2024
Ayfer Ozgur, Stanford University

1 Mutual information

Recall that for a discrete random variable X, taking values in alphabet $\mathcal{X}$, e.g., $\mathcal{X} = \{0, 1, 2, 3\}$, the **entropy** of X is defined as:

$$H(X) = \sum_{x \in \mathcal{X}} P(X = x) \log \frac{1}{P(X = x)}.$$

Similarly, for two discrete random variables X and Y, their joint entropy is defined as

$$H(X, Y) = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} P(X = x, Y = y) \log \frac{1}{P(X = x, Y = y)}.$$

We define the mutual information between X and Y as

$$I(X; Y) \triangleq H(X) + H(Y) - H(X, Y).$$

The mutual information represents the reduction in uncertainty when we consider X and Y together rather than considering them individually, and therefore it quantifies the information one variable provides about the other. Note that the expression is symmetric in X and Y, hence the information X provides about Y is the same as the information Y provides about X. We saw before that $H(X, Y) \leq H(X) + H(Y)$, which implies that $I(X; Y) \geq 0$, i.e. mutual information is always non-negative. When X and Y are independent, $H(X, Y) = H(X) + H(Y)$ and therefore $I(X; Y) = 0$, i.e. X and Y do not provide any information about each other. When X and Y are dependent, $H(X, Y) < H(X) + H(Y)$ and hence $I(X; Y) > 0$. The gap is larger when X and Y are more dependent. Thus, the more dependent the random variables X and Y, the larger the mutual information.

2 Shannon's Channel Coding Theorem

We are now ready to state Shannon's channel coding theorem.

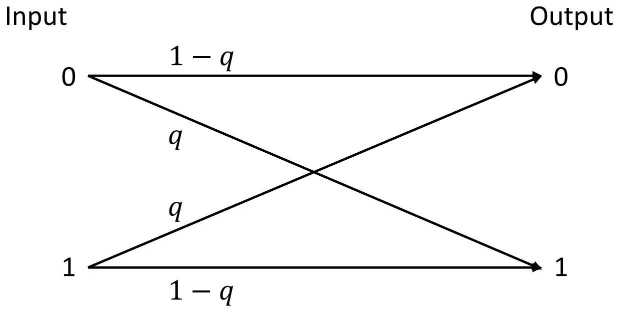
Shannon (1948): For any communication channel (characterized by the conditional distribution of the output Y given the input X, i.e. $P(Y = y \mid X = x)$ for $x \in \mathcal{X}, y \in \mathcal{Y}$), there exists a transmitter and receiver that make it possible to send C bits/channel use with arbitrarily small probability of error. Conversely, if we try to send information at a rate higher than C bits/channel use, errors are inevitable. The threshold C is called the capacity of the channel and can be computed by the following formula:

$$C = \max_{X} I(X; Y) = \max_{p(X)} H(X) + H(Y) - H(Y, X)$$

where the maximization is over all distributions $p(X)$ for the input X. (Note that for any distribution on the input X, we have a joint distribution on (X, Y) induced by the input distribution and the conditional distribution for the channel, hence we can compute $I(X; Y)$.)

Before Shannon's 1948 work, people generally assumed that getting reliable communication, i.e. driving the bit error rate $P_b \to 0$ requires a vanishing rate. But Shannon in his landmark 1948 paper showed that we do not have to sacrifice rate to communicate reliably. He showed that there exists a threshold C associated with every communication channel, its capacity, and we can make the bit error rate P_b as small as we want as long as the the rate is below C. Conversely, he also showed that it is not possible to communicate reliably when the information rate is above C. Moreover, he provided an explicit formula to compute the capacity of any communication channel, which as we will see next is typically not difficult to evaluate.

3 Binary Symmetric Channel



We will next compute the Shannon capacity for the binary symmetric channel. Recall that the binary symmetric channel with parameter q, denoted as $BSC(q)$, is as shown in the figure above. The channel has binary input and binary output (hence the "binary" in its name). For each input, the channel has identical output with probability $1-q$ and flipped output with probability q. Note that the edges in the diagram are marked with the conditional probability of receiving the output symbol given the input symbol being transmitted. The channel is "symmetric" in the sense that the behavior for both inputs (0 or 1) is the same. A non-symmetric channel might have higher probability for a $0 \to 1$ flip than a $1 \to 0$ flip.

In order to evaluate the capacity formula for this channel we need to solve an optimization problem, i.e. characterize the input distribution $p(X)$ that maximizes the mutual information between the input and the output of a $BSC(q)$. It can be shown that this mutual information is maximized when X is uniformly distributed. We state this as a fact without proof.

Fact: $I(X;Y)$ for a $BSC(q)$ is maximized when the input X is uniformly distributed, i.e. $P(X=1) = P(X=0) = 1/2$.

Using this fact we can evaluate the mutual information. Note that when $P(X=1) = P(X=0) = 1/2$,

$$H(X) = 1.$$

Moreover, when $P(X=1) = P(X=0) = 1/2$, Y is also uniformly distributed, i.e. $P(Y=1) = P(Y=0) = 1/2$. Note that the channel treats 0 and 1 symmetrically, therefore when the input bit is equally likely to be 0 or 1, the output bit will also be 0 or 1with equal probability. Therefore,

$$H(Y) = 1.$$

$H(X,Y)$ can be computed by using the following joint probabilities:

$$\begin{aligned}
P(X=0,Y=0) &= P(X=0)P(Y=0|X=0) = \frac{1}{2}(1-p) \\
P(X=0,Y=1) &= P(X=0)P(Y=1|X=0) = \frac{1}{2}p \\
P(X=1,Y=1) &= P(X=1)P(Y=1|X=1) = \frac{1}{2}(1-p) \\
P(X=1,Y=0) &= P(X=1)P(Y=0|X=1) = \frac{1}{2}p
\end{aligned}$$

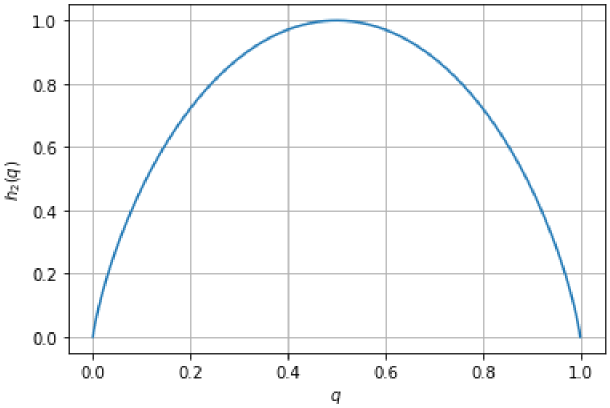
By using the formula for the joint entropy of two random variables, we obtain:

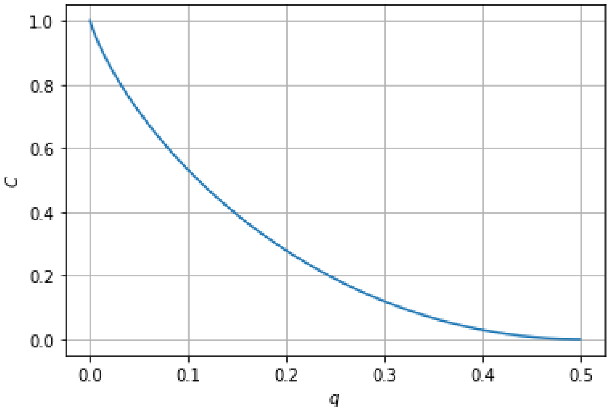
$$H(X,Y) = 1 + q\log_2\frac{1}{q} + (1-q)\log_2\frac{1}{1-q} = 1 + H_2(q)$$

Note that $H_2(q)$ corresponds to the entropy of a binary random variable (Bernoulli) with probability q. Combining the above, we conclude that the capacity of a $BSC(q)$ is given by

$$C = I(X;Y) = H(X) + H(Y) - H(X,Y) = 1 + 1 - (1 + H_2(q)) = 1 - H_2(q).$$

Below we plot $H_2(q)$ and C with varying q.





When $q = 0$, we have a perfect channel and can achieve rate of 1 without needing any coding. When $q = \frac{1}{2}$, we have a completely useless channel since the output of the channel becomes independent of the input, hence the capacity is 0. For points in between, we can communicate reliably at a non-zero rate despite there being a non-negligible bit-flip probability. Much of the research in the field after Shannon's 1948 paper has been focused on building practical codes that achieve capacity.

4 Law of large numbers

We next try to get some insight as to why reliable communication over a noisy channel is at all possible. The key is to leverage the law of large numbers to "eliminate" the randomness in the channel. The law of large numbers formalizes a phenomena we are familiar with from daily experience: the average of the results obtained from a large number of trials should be close to the expected value and tends to become closer to the expected value as more trials are performed. We state this in the context of binary trials, i.e. binary random variables.

Law of Large Numbers (LLN): Let $Z_1, Z_2, \ldots, Z_n$ be a sequence of i.i.d. (independent and identically distributed) random variables such that $Z_i \sim Ber(q)$. This simply means that each Z_i is a binary r.v. with $P(Z_i = 1) = q$ and $P(Z_i = 0) = 1 - q$. The **law of large numbers** (LLN) says that

$$P\left(\left|\frac{1}{n}\sum_{i}^{n} Z_i - q\right| < \delta\right) \xrightarrow{n\to\infty} 1,\ \forall \delta > 0.$$

In words, the average value of the Z_i's is within δ of q with probability approaching 1 for large n, even if we take δ to be arbitrarily small. Equivalently, this means that as n gets large, $\sum_{i=1}^{n} Z_i$ will converge to nq, i.e. the number of 1's in the sequence $Z_1, Z_2, \ldots, Z_n$ will be very close to nq with probability close to 1 (we can make this probability arbitrarily close to 1 by taking n large enough).

What does this say for our communication problem? It suggests that if we use very long codewords of length n, then the number of bit flips introduced by BSC(q) will be very close to nq. This allows us to eliminate the uncertainty about what the channel will do to our transmitted codewords: while we do not know which bits will be flipped by the channel, we know almost surely that the total number of flipped bits will be very close to nq. Hence if we design a code that can correct nq errors (or $n(q+\delta)$ by allowing a small slack δ), we can correct all errors introduced by the channel and achieve reliable communication.

5 Bounding the rate of a reliable code

What is the largest size for a code with codeword length n that can correct nq errors? How many codewords M can such a code have? We can use the sphere packing argument from our previous lectures to bound the size of such a code. We used this argument when we talked about Hamming codes. Recall that we looked at the Hamming balls of radius 1 around each codeword because in that case we were interested in correcting one error. But now, for codewords of length n, we need to look at Hamming balls of radius nq around each codeword, because we expect (by LLN) to receive a sequence that is of Hamming distance nq to the transmitted codeword. Note that the number of sequences with Hamming distance $k = nq$ from a codeword is exactly the same as the number of binary sequences of length n with k ones, which is simply $\binom{n}{k}$. $\binom{n}{k}$ can be approximated with $2^{nH_2(\frac{k}{n})}$, as we verify at the end of this section. Note that in our case $k = nq$, so we have approximately $2^{nH_2(q)}$ sequences in each ball.

Now, successful decoding requires that these balls containing $\binom{n}{k}$ sequences around each codeword to be disjoint, otherwise it is not possible to determine the true codeword from the received sequence. Thus, we can find the maximum number of codewords so that the balls remain disjoint by noting the total number of sequences in all the balls combined cannot exceed the total number of binary sequences of length n.

Thus, we get that the number of codewords M in such a code can be at most $\frac{2^n}{2^{nH_2(q)}} = 2^{n(1-H_2(q))}$. Taking the logarithm, we get that the number of information bits in n channel uses is at most $n(1 - H_2(q))$, which shows that the rate is at most $1 - H_2(q)$. This type of packing argument shows us that we cannot hope to reliably transmit more than $1 - H_2(q)$ information bits per channel use over a binary symmetric channel with bit flip probability q. Note that this was precisely Shannon's capacity for this channel.

To see that $\binom{n}{k} \approx 2^{nH_2(\frac{k}{n})}$, use the Stirling's approximation, which says that $\log n! \approx n \log n$ (you can think of $\log n! = \sum_{i=1}^{n} \log n$ where we have n terms most of which are within a constant factor of n).

We use this to get the following sequence of algebraic manipulations:

$$\begin{aligned}
\log \binom{n}{k} &= \log \frac{n!}{k!(n-k)!} \\
&= \log n! - \log k! - \log(n-k)! \\
&\approx n \log n - k \log k - (n-k) \log(n-k) \\
&= k \log n + (n-k) \log n - k \log k - (n-k) \log(n-k) \\
&= k \log \frac{n}{k} + (n-k) \log \frac{n}{n-k} \\
&= n \left[\frac{k}{n} \log \frac{n}{k} + \left(1 - \frac{k}{n}\right) \log \frac{1}{1 - \frac{k}{n}} \right] \\
&= nH_2\left(\frac{k}{n}\right)
\end{aligned}$$

Thus, we have

$$\binom{n}{k} \approx 2^{nh_2(\frac{k}{n})}.$$

The takeaway message is that the number of binary sequences of length n with fraction q 1's is about $2^{nH_2(q)}$. (This can actually be connected to our results on compression. Any ideas how?)

The above packing argument allows to bound the size M of a code that has codeword length n and can correct nq errors. However, it does tell us that there exists a code with

$$M = \frac{2^n}{2^{nH_2(q)}} = 2^{n(1-H_2(q))}$$

codewords. Shannon showed the existince of such a code with a random construction. He showed that if we generate M binary strings (with M given by the above formula) such that each entry of each string is chosen at random, independently and equally likely to be 0 or 1, then the code consisting of these M binary strings will be able to correct nq errors. However, such a code will have prohibitive encoding and decoding complexity since M is exponential in the codeword length n and we need to choose n large enough to leverage the law of large numbers. Shannon's work has initiated the search for efficiently encodable and decodable codes that achieve the Shannon limit, meaning they can achieve reliable communication at rates close to the capacity of the channel.