QI Lecture 7

Classical Information

What is Information?

In 1948, when Claude Shannon first created our modern mathematical description of information, he did so by first framing the problem.

"The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point. [...] The significant aspect is that the actual message is one *selected from a set* of possible messages. The system must be designed to operate for each possible selection, not just the one which will actually be chosen since this is unknown at the time of design."

Shannon then appeals to our intuition to justify using logarithms to define information.

"One feels, for example, that two punched cards should have twice the capacity of one for information storage..."

Shannon used H to describe information and took as his first premise that the amount of information that can be communicated by selecting an item from a set with n items total obeys

$$H \propto \log(n)$$

This corresponds to our intuition. If there are n possible states for one punch card and it can store $\log(n)$ information, then there are n^2 possible states for two punch cards and the two combined can store $\log\left(n^2\right) = 2\log(n)$ information.

Setting the base of the logarithm defines the units of the information. In particular, $H = \log_2(n)$ defines information in units of "bits". Using base e, the natural log, the units are "nats" and using base π the units are "slices". Nats are typically used for "differential entropy" where the state space is continuous, such as when talking about how much information can be sent using radiowaves within some limited band. The one time that base π is used is when making that "slices" joke. You won't see it again.

If every member of an "alphabet" with n "letters" is equally likely to be sent, then our job is already done. But consider the following strings:

$$SKFJGHIUYSDRTFUDGVEBRJKGFHJNOHFGUDYGIUEGHBFA$$

$$AAAAAAAAAAAAAAAAAAAAAAAAACAAAAAAAAAAAAAAAAA$$

The first string contains more information than the second, because the second can be compressed. Think about how you would read these off to someone over the phone; for the first you'd need to laboriously list off every letter and for the second you'd probably say "26 A's, a C, and then 19 more A's".

We think of the English alphabet as having 26 letters, but in effect it has fewer. For example, Q, X, and Z are pretty rare. We might naively estimate that the "entropy rate" of written English should be about $H = \log_2(26) \approx 4.7$ bits. However, that's the rate for a completely random string of equally-likely letters. Experimentally, by asking people to predict the next letter in a (grammatically correct) sentence, we find that the entropy rate of written English is in the neighborhood of 1 bit per letter (Shannon estimated 0.6-1.3). So we not only need to worry about the size of the alphabet, n, but also the probability of each letter.

A "**probability distribution**", $p(X)$, of a "**random variable**", X, is a function which specifies the probability of each possible value, x_j, of X being selected. We write $p(X = x_j) = p_j$ or just $p(x_j) = p_j$.

We define the Shannon Entropy as a function of probability distributions and have a few different ways of writing the same situation:

$$H[X] = H[\{p_1, p_2, \ldots, p_n\}] = H[\{p_j\}_j]$$

So far we know that if $p_1 = p_2 = \ldots = p_n = \frac{1}{n}$, then

$$\boxed{H\left[\left\{\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n}\right\}\right] = \log_2(n)}$$

"The Number of Yes/No Questions"

Another way to think about information measured in bits, is to ask yourself "what is the minimum number of yes/no questions I need to ask to specify a random variable, *on average*?"

Example By considering an alphabet $\{A, B, C, D\}$ with a probability distribution $p(A) = p(B) = p(C) = p(D) = \frac{1}{4}$. Find $H\left[\left\{\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right\}\right]$ by "asking yes/no questions" about which letter has been selected.

A terrible way to do this is to ask "Is it A?", "Is it B?", and "Is it C?" If all the answers are "no", then clearly D has been selected. The average number of questions in this case is

$$\frac{1}{4}1 + \frac{1}{4}2 + \frac{1}{2}3 = \frac{9}{4} = 2.25$$

But when we ask "Is it A?", we expect the answer to be "no" with probability $\frac{3}{4}$. A more efficient way to search this alphabet is to make the yes and no equally probable; to partition the alphabet in half.

A better series of questions is "Is it A or B?" followed by (depending on the answer) "Is it A?" or "Is it B?". The number of questions is always exactly two.

But here's something to notice: after the first question is asked, the entropy of the remaining possibilities drops from $H\left[\left\{\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right\}\right]$ to $H\left[\left\{\frac{1}{2}, \frac{1}{2}\right\}\right]$. That is to say, we go from one of four possibilities to one of two. Since the entropy is the average number of questions

$$H\left[\left\{\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right\}\right] = 1 + H\left[\left\{\frac{1}{2}, \frac{1}{2}\right\}\right] = 1 + 1 = 2$$

This corresponds with exactly what we'd expect

$$H\left[\left\{\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right\}\right] = \log_2(4) = 2$$

■

Example Assume the following probability distribution on the letters A-F.

$item$	A	B	C	D	E	F
$p(item)$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{8}$

Once again, the most efficient way to ask questions is to have yes's and no's that are equally likely. There are a few ways to do this, but we'll use "is it A or D?" as the first question. After that first question there's a $\frac{1}{2}$ probability of the remaining choices being A or D and $\frac{1}{2}$ of the remaining choices being B, C, E, or F. Notice what this does to the entropy:

$$H\left[\left\{\frac{1}{4},\frac{1}{8},\frac{1}{8},\frac{1}{4},\frac{1}{8},\frac{1}{8}\right\}\right] = \underbrace{H\left[\left\{\frac{1}{2},\frac{1}{2}\right\}\right]}_{\text{Q: "A or D?"}} + \underbrace{\frac{1}{2}H\left[\left\{\frac{1}{2},\frac{1}{2}\right\}\right]}_{A\text{:"}yes\text{"}} + \underbrace{\frac{1}{2}H\left[\left\{\frac{1}{4},\frac{1}{4},\frac{1}{4},\frac{1}{4}\right\}\right]}_{A\text{:"}no\text{"}} = 1 + \frac{1}{2}\cdot 1 + \frac{1}{2}\cdot 2$$

■

We can generalize this. Partition the distribution into n distinct sets, such that the probability of selecting any item from the jth set is q_j and the probability of selecting only the kth item in the jth set is $q_j p_{jk}$. We ask one round of questions to figure out which set the selection is in, then ask a second round of questions (dependent on the result from the first round) to determine the particular item in the set.[1] The average number of yes/no questions is then

$$H\left[\{q_1p_{11},\ldots,q_1p_{1n_1},q_2p_{21},\ldots,q_2p_{2n_2},\ldots\}\right] = \underbrace{H\left[\{q_1,q_2,\ldots,q_n\}\right]}_{\text{1st round of questions}} + \underbrace{\sum_k q_k H\left[\{p_{k1},p_{k2},\ldots,p_{kn_k}\}\right]}_{\text{2nd round of questions}}$$

This is the key to finding a general equation or the Shannon entropy.

Assume that the probabilities in a finite distribution, $\{p_1,p_2,\ldots,p_n\}$, are all rational valued. Since we can approximate any number to arbitrary precision using rational numbers, this isn't a terribly disruptive assumption.

It follows that for an appropriate choice of m, we can partition m items into n groups such that $m = \sum_j m_j$ and $p_j = \frac{m_j}{m} = \frac{m_j}{\sum_j m_j}$. If the probability of selecting any one of the m items is equal, then p_j is the probability of selecting any item from the jth partition.

$$H\left[\left\{\frac{1}{m},\frac{1}{m},\ldots,\frac{1}{m}\right\}\right] = H\left[\left\{\frac{m_1}{m},\frac{m_2}{m},\ldots,\frac{m_n}{m}\right\}\right] + \sum_k \frac{m_k}{m} H\left[\left\{\frac{1}{m_k},\frac{1}{m_k},\ldots,\frac{1}{m_k}\right\}\right]$$

$$\log(m) = H\left[\left\{\frac{m_1}{m},\frac{m_2}{m},\ldots,\frac{m_n}{m}\right\}\right] + \sum_k \frac{m_k}{m}\log(m_k)$$

$$\sum_k \frac{m_k}{m}\log(m) = H\left[\left\{\frac{m_1}{m},\frac{m_2}{m},\ldots,\frac{m_n}{m}\right\}\right] + \sum_k \frac{m_k}{m}\log(m_k)$$

$$0 = H\left[\left\{\frac{m_1}{m},\frac{m_2}{m},\ldots,\frac{m_n}{m}\right\}\right] + \sum_k \frac{m_k}{m}\left(\log(m_k) - \log(m)\right)$$

[1] This can be extended ad nauseam, with sets within sets within sets, but what we're really interested in is this equation.

$$H\left[\left\{\frac{m_1}{m}, \frac{m_2}{m}, \ldots, \frac{m_n}{m}\right\}\right] = -\sum_k \frac{m_k}{m} \log\left(\frac{m_k}{m}\right)$$

Finally, substituting the probabilities back in, $p_j = \frac{m_j}{m}$, we get the famous equation for Shannon Entropy!

$$\boxed{H\left[\{p_1, p_2, \ldots, p_n\}\right] = -\sum_k p_k \log\left(p_k\right)}$$

Since $\log(x) < 0$ for $x \in (0, 1)$, that negative actually ensures that the entropy is positive. The bounds on the Shannon entropy are

$$\boxed{0 \leq H\left[\{p_1, p_2, \ldots, p_n\}\right] \leq \log(n)}$$

The lower bound, zero, occurs for probability distributions that are completely unbalanced, where a single letter has probability 1. The upper bound, $\log(n)$, occurs for uniform probability distributions, where every letter is equally likely. This is a good rule in general; the greatest amount of information is communicated when every letter is equally likely and the greatest amount is learned when every answer to a question is equally likely.

The entropy rate, or the average number of bits per letter, gives us a limit on data compression. A string of bits, seemingly random and equally probable, has an entropy rate of 1 bit per character. The byte (8 bits) was created initially to store a single character of text but (as can be demonstrated experimentally or with computers and lots of reading material) the entropy rate of written English is actually closer to 1 bit per character. That implies the possibility of an at most 8-fold compression.

Shannon entropy tells us nothing about what that encoding scheme should be; it only gives us a limit on how good it might be.

Less Than a Bit

A 0/1 can contain less than one bit of information if the probability of each isn't $\frac{1}{2}$. For example, "11111111111111..." doesn't communicate much.

There are two things to notice here. First, entropy is maximized for $p = \frac{1}{2}$.

$$0 = \frac{dH}{dp} = \frac{1}{\ln(2)}\left[-\ln(p) - 1 + \ln(1-p) + 1\right] \quad \Rightarrow \quad \ln(p) = \ln(1-p) \quad \Rightarrow \quad p = \frac{1}{2}$$

Second, we'll use the standard that

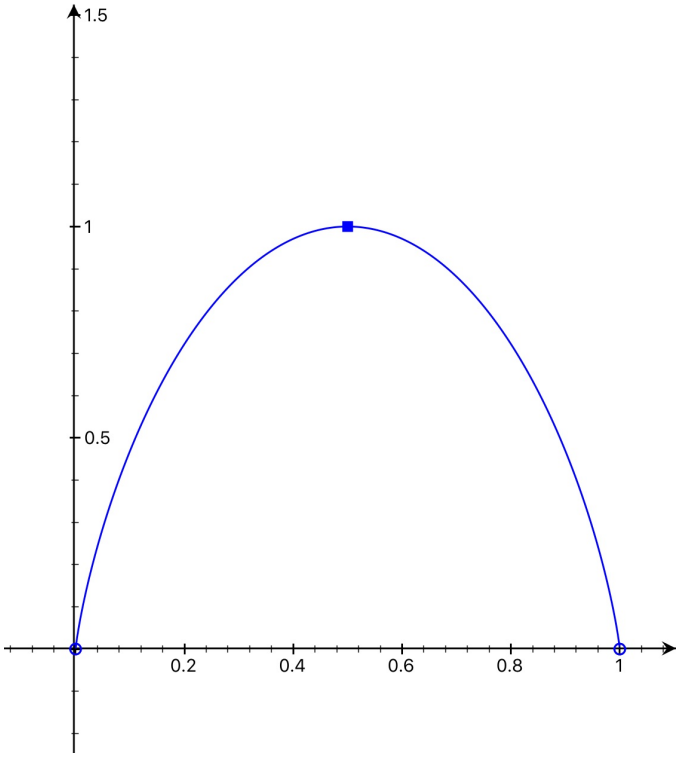

Figure 1: $H\left[\{p, 1-p\}\right] = -p\log_2(p) - (1-p)\log_2(1-p)$

$$0\log(0) = 0$$

because

$$\lim_{p\to 0} p\log_b(p) = \frac{1}{\ln(b)}\lim_{p\to 0} p\ln(p) = \frac{1}{\ln(b)}\lim_{p\to 0}\frac{\ln(p)}{\frac{1}{p}} = \frac{1}{\ln(b)}\lim_{p\to 0}\frac{\frac{1}{p}}{-\frac{1}{p^2}} = \frac{-1}{\ln(b)}\lim_{p\to 0} p = 0$$

Now, it may bother you that a string of 1's and 0's can carry less than 1 bit of information per digit. After all, regardless of the probabilities involved, it takes 1 bit to describe a 0-or-1. The key to compressing data is to break your bits into larger blocks and use "codewords" of different lengths, so that short code words correspond to likely sequences and long codewords apply to unlikely sequences. For example, if $p(0) = 0.99$ and $p(1) = 0.01$, then a Huffman code[2] could use "1" to represent "0000000000" and "01" to

[2]One example of an algorithm for creating optimal or nearly optimal "codebooks"..

represent "0000000001", and some very long codeword to represent the extremely unlikely "1111111111".[3]

The string of compressed data uses its symbols approximately equally often, so in the case of bits you have an entropy of 1. This means that a string of bits with an entropy of 0.1 bits per digit can be compressed by a factor of ten. Typically, the larger the blocks that are considered, the closer the encoding comes to being optimal.

Conditional and Mutual Information

We describe both parties using random variables, X for the sender and Y for the receiver. The probability distributions of X and Y are $p(x)$ and $p(y)$, the joint distribution is $p(x,y)$, and the conditional probability distribution is $p(y|x) = p(x)p(x,y)$.

The "**conditional entropy**"

$$H[Y|X] = -\sum_{x,y} p(x,y)\log(p(y|x))$$

is the entropy left in Y when you know what X is. For example, if you have two coins that always land the same way when flipped, the entropy of either is $H[X] = H[Y] = 1\,bit$. But if they're always the same, then $p(y|x) = 0, 1$ and therefore $H[Y|X] = 0$. That is, if you know the value of X, and Y and X are always the same, then there's no entropy remaining in Y.

The "**mutual information**" can be written in a few mathematically equivalent forms

$$I[X;Y] = H[Y] - H[Y|X] = -\sum_{y} p(y)\log(p(y)) + \sum_{x,y} p(x,y)\log(p(y|x))$$

$$I[X;Y] = H[X] - H[X|Y] = -\sum_{x} p(x)\log(p(x)) + \sum_{x,y} p(x,y)\log(p(x|y))$$

$$I[X;Y] = D\left(p(x,y)\|p(x)p(y)\right) = \sum_{x,y} p(x,y)\log\left(\frac{p(x,y)}{p(x)p(y)}\right)$$

is the amount of information shared by two random variables. In the "always the same side up" example from a moment ago, $I[X;Y] = H[Y] - H[Y|X] = 1 - 0 = 1$ because each coin flip produces one bit of information and they share that same bit. However, if they're completely independent (like actual coins), then Y has no dependence on X and therefore $I[X;Y] = H[Y] - H[Y|X] = H[Y] - H[Y] = 0$.

[3] There's a (probably apocryphal) story about engineers at Bell Labs, shortly after Huffman codes were invented, shouting "One!" at each other rather than "F*&$ you!", since it was the most efficient encoding of their conversations.

The third equation, $I[X;Y] = D\left(p(x,y) \| p(x)p(y)\right)$, describes the mutual information as the "relative entropy" (something we won't cover here) between the joint probability distribution, $p(x,y)$, and the distribution that assumes the two variables are independent, $p(x)p(y)$.

Channel Capacity

Successful communication means that two parties will share the same information after the communication event. A good way to describe that is in terms of the mutual information shared by the two parties. A "channel" is described by the conditional probability, $p(y|x)$, since we're interested in the probability of X sending a letter and Y receiving that letter.[4]

The "**channel capacity**"

$$C \equiv \max_{p(x)} I[X;Y] = \max_{p(x)} H[Y] - H[Y|X]$$

is the maximum possible value of the mutual information. The conditional probabilities are defined by the channel that's being used, but the probability distribution over X, $p(x)$, can be modified. The channel capacity is the maximum amount of information that can be transmitted over a channel with each use. Typically, this is coupled with some kind of encoding scheme optimized for the channel. Fortunately, information is a "resource" in the sense that the engineers worried about transmitting information don't need to worry about what that information represents or how it was encoded previously; they just need to send "information".

Theorem (The Channel Coding Theorem)**.** *If the channel capacity is C, then any information rate less than or equal to C is achievable using "codewords" of length $\frac{n}{C}$ to encode strings of characters of length n. Moreover, the probability of error approaches zero as $n \to \infty$.*

In application, n doesn't need to be terribly large. So for example, if $C = 0.5\,bits$, then we could encode (for example) "010101 → 110010110001". The codeword is longer to absorb the inevitable errors. For example, "110010100001" and "110010110011" (which have a single bit-flip error each) would both be assumed to be equivalent to the original codeword and would be decoded as "010101". As n gets bigger, the law of large numbers kicks in pretty quickly and the probability of one codeword being mistaken for another becomes vanishingly small.

[4] The mutual information is a little more sophisticated than that. For example, when using a substitution cipher none of the received letters are the same as the sent letters and yet that has no impact on the calculation of the mutual information.

Example First, we'll calculate the channel capacity of Alice, in a quiet room, clearly and slowly saying "zero" or "one" to Bob, who is listening carefully. Alice's random variable is X and Bob's is Y.

Alice's probability distribution is

$$\begin{cases} p(x=1) &= a \\ p(x=0) &= b \end{cases}$$

These are the probabilities that she will say either 0 or 1. The conditional probability on this very clean channel is

$$\begin{cases} p(y=1|x=1) &= 1 \\ p(y=1|x=0) &= 0 \\ p(y=0|x=1) &= 0 \\ p(y=0|x=0) &= 1 \end{cases}$$

and the joint probabilities are

$$\begin{cases} p(x=1,y=1) &= p(x=1)\,p(y=1|x=1) &= a \\ p(x=0,y=1) &= p(x=0)\,p(y=1|x=0) &= 0 \\ p(x=1,y=0) &= p(x=1)\,p(y=0|x=1) &= 0 \\ p(x=0,y=0) &= p(x=0)\,p(y=0|x=0) &= b \end{cases}$$

and finally the probability distribution for Bob is the marginal probability of Y

$$\begin{cases} p(y=1) &= p(x=1,y=1)+p(x=0,y=1) &= a \\ p(y=0) &= p(x=1,y=0)+p(x=0,y=0) &= b \end{cases}$$

We see immediately that $H[Y] = H[\{a,b\}]$ and that the conditional probability is

$$\begin{aligned} & H[Y|X] \\ = \;& -\textstyle\sum_{x,y=0}^{1} p(x,y)\log(p(y|x)) \\ = \;& -p(0,0)\log(p(0|0)) - p(0,1)\log(p(1|0)) - p(1,0)\log(p(0|1)) - p(1,1)\log(p(1|1)) \\ = \;& 0+0+0+0 \end{aligned}$$

Here we used the "$0\log(0)=0$" standard.

This result makes sense, because there is no randomness in Bob's random variable given Alice's. If you've been listening to Alice, then there's nothing new to learn from reading what Bob writes down. Therefore

$$C = \max_{p(x)} H[Y] - H[Y|X] = \max_{p(x)} H\left[\{a, b\}\right] = H\left[\left\{\frac{1}{2}, \frac{1}{2}\right\}\right] = 1$$

Since $a = b = \frac{1}{2}$ is the known maximum for $H\left[\{a, b\}\right]$.

■

Example The denizens of a castle are under siege and their only hope is to communicate with friendly forces beyond the besieging army. They've long since run out of paper (things are not going well), but they still have a supply of messenger birds. Every hour they attempt to communicate one bit of information to their friends by either releasing a bird (1) or not (0).

To complicate things, there are archers turning their 1's into 0's half the time, thus adding noise to their communication channel. How much information can be communicated per bird on average?

The answer is the "channel capacity", which is the maximum value of the mutual information between the castle, random variable X, and the friendly army, system Y. It's up to X to choose a probability distribution that will maximize the mutual information.

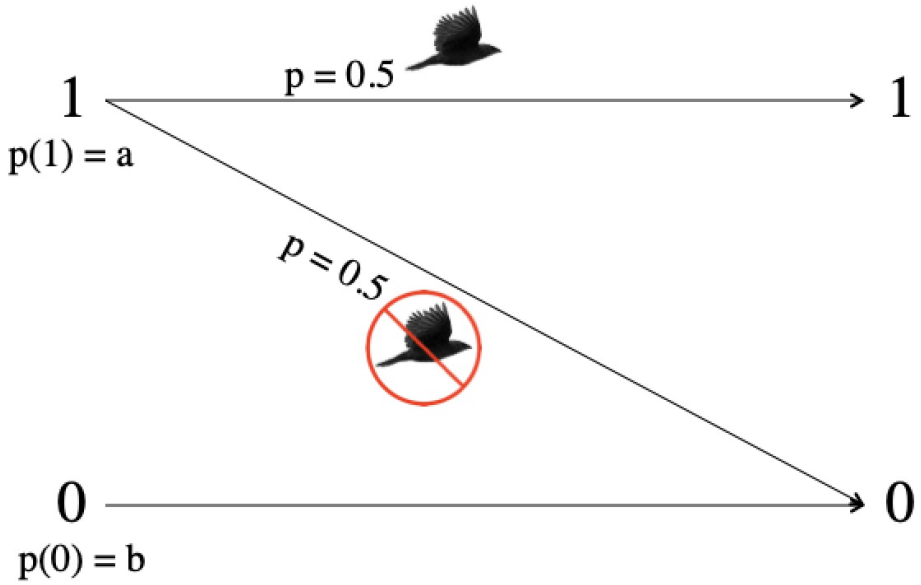

Figure 2: With probability a a bird is released, with probability $b = 1 - a$ it is not, and there is a $\frac{1}{2}$ chance that when a bird is released it will be shot down.

The probability distribution over X, the sending party, is

$$\begin{cases} p(x=1) &= a \\ p(x=0) &= b \end{cases}$$

The conditional probability over this anti-avian channel is

$$\begin{cases} p(y=1|x=1) &= \frac{1}{2} \\ p(y=1|x=0) &= 0 \\ p(y=0|x=1) &= \frac{1}{2} \\ p(y=0|x=0) &= 1 \end{cases}$$

The joint distribution is

$$\begin{cases} p(x=1,y=1) &= p(x=1)\,p(y=1|x=1) &= \frac{a}{2} \\ p(x=0,y=1) &= p(x=0)\,p(y=1|x=0) &= 0 \\ p(x=1,y=0) &= p(x=1)\,p(y=0|x=1) &= \frac{a}{2} \\ p(x=0,y=0) &= p(x=0)\,p(y=0|x=0) &= b \end{cases}$$

and therefore the probability distribution over Y, the receiving party, is

$$\begin{cases} p(y=1) &= p(x=1,y=1)+p(x=0,y=1) &= \frac{a}{2} \\ p(y=0) &= p(x=1,y=0)+p(x=0,y=0) &= b+\frac{a}{2} \end{cases}$$

First, the entropy of the friendly army's random variable is

$$H[Y] = H\left[\left\{\frac{a}{2}, b+\frac{a}{2}\right\}\right] = H\left[\left\{\frac{a}{2}, 1-\frac{a}{2}\right\}\right] = -\frac{a}{2}\log\left(\frac{a}{2}\right) - \left(1-\frac{a}{2}\right)\log\left(1-\frac{a}{2}\right)$$

where we used the fact that $b = 1-a$. The conditional entropy is

$$\begin{aligned} & H[Y|X] \\ = \; & -\textstyle\sum_{x,y=0}^{1} p(x,y)\log(p(y|x)) \\ = \; & -p(0,0)\log(p(0|0)) - p(0,1)\log(p(1|0)) - p(1,0)\log(p(0|1)) - p(1,1)\log(p(1|1)) \\ = \; & 0+0-\tfrac{a}{2}\log\left(\tfrac{1}{2}\right) - \tfrac{a}{2}\log\left(\tfrac{1}{2}\right) \\ = \; & a\log(2) \\ = \; & a \end{aligned}$$

and therefore

$$I(X;Y) = -\frac{a}{2}\log\left(\frac{a}{2}\right) - \left(1-\frac{a}{2}\right)\log\left(1-\frac{a}{2}\right) - a$$

To find the maximum of this we take the derivative, setting it equal to zero, and remembering that the log is base 2

$$\begin{aligned}
0 &= \tfrac{dI(X;Y)}{da} \\
0 &= -\tfrac{1}{2}\log\left(\tfrac{a}{2}\right) - \tfrac{1}{2} + \tfrac{1}{2}\log\left(1-\tfrac{a}{2}\right) + \tfrac{1}{2} - 1 \\
1 &= -\tfrac{1}{2}\log\left(\tfrac{a}{2}\right) + \tfrac{1}{2}\log\left(1-\tfrac{a}{2}\right) \\
2 &= -\log\left(\tfrac{a}{2}\right) + \log\left(1-\tfrac{a}{2}\right) \\
2 &= \log\left(\tfrac{1-\frac{a}{2}}{\frac{a}{2}}\right) \\
2 &= \log\left(\tfrac{2}{a} - 1\right) \\
2^2 &= \tfrac{2}{a} - 1 \\
4+1 &= \tfrac{2}{a} \\
a &= \tfrac{2}{5}
\end{aligned}$$

This strikes a balance between minimizing the noise, which is only a problem when $x = 1$, and maximizing the information, which is greatest when $a = b = \frac{1}{2}$. Plugging this maximum, $a = \frac{2}{5}$, into the mutual information gives us the channel capacity

$$C = -\frac{1}{5}\log\left(\frac{1}{5}\right) - \frac{4}{5}\log\left(\frac{4}{5}\right) - \frac{2}{5} \approx 0.322$$

So those arches are really slowing down communication. Instead of communicating 1 bit per hour, the people in the castle are reduced to at most a little less than a third of a bit per hour.

Using a block length of $n = 10$ bits (for example), their codewords should be $\lfloor \frac{n}{C} \rfloor = \lfloor \frac{10}{0.322} \rfloor = 31\,bits$ long and should use the optimal probability distribution, $p(0) = \frac{3}{5}$ and $p(1) = \frac{2}{5}$. For example, one particular block of ten bits could be encoded as

$$1000101010 \longrightarrow 0110000011010010000010000100001$$

■

Exercises

#1) Entropy

State all answers in bits.

a) $H\left[\left\{\frac{1}{3}, \frac{1}{6}, 0, \frac{1}{2}\right\}\right]$ =?

b) How much information can be stored on a 3-digit padlock?

c) Let X be the random variable that represents the sum of two six-sided dice. What is $H[X]$?

#2) Counterfeit

In a bag of N coins there is a single counterfeit. All of the coins look the same and weigh exactly the same, with the exception of the counterfeit which is either lighter or heavier. As Master of Coin it's your job to find the counterfeit, but you only have access to a balance scale and you're short on time.

Figure 3: A balance scale doesn't tell you how much something weighs, only which side is heavier.

a) How many different responses can a balance scale give you and what are they?

b) To find the counterfeit, how many possibilities do you need to be able to distinguish?

c) In theory, what is the minimum number of measurements you'll need in order to find the counterfeit?

d) (optional) Figure out how to find a single either-heavier-or-lighter coin from a set of 12 while using the scale only 3 times.

#3) The Symmetric Channel

In the symmetric channel there is a chance of α that no message is received at all. So X can be 0 or 1, while Y can be 0, 1, or e for "error". The conditional probabilities here are:

$$\begin{cases} p(y=1|x=1) = 1-\alpha \\ p(y=1|x=0) = 0 \\ \hline p(y=0|x=1) = 0 \\ p(y=0|x=0) = 1-\alpha \\ \hline p(y=e|x=1) = \alpha \\ p(y=e|x=0) = \alpha \end{cases}$$

What is the channel capacity of the symmetric channel?

#4) Unconditional

Show that if $H[Y|X] = 0$, then Y is a one-to-one function of X. That is, show that for all x such that $p(x) \neq 0$, there is only one possible value of y with $p(x,y) > 0$.

This shows that 1-1 codes, like a substitution cipher (A=3, B=1, C=15, ...), don't change the information content.