— 45 —

Entropy III - Shannon's Theorem

The memory of my father is wrapped up in
white paper, like sandwiches taken for a day at work.

Just as a magician takes towers and rabbits
out of his hat, he drew love from his small body,

and the rivers of his hands
overflowed with good deeds.

Yehuda Amichai, My Father

45.1. Coding: Shannon's Theorem

We are interested in the problem sending messages over a noisy channel. We will assume that the channel noise is "nicely" behaved.

Definition 45.1.1. The input to a ***binary symmetric channel*** with parameter p is a sequence of bits $x_1, x_2, \ldots$, and the output is a sequence of bits $y_1, y_2, \ldots$, such that $\mathbb{P}[x_i = y_i] = 1 - p$ independently for each i.

Translation: Every bit transmitted have the same probability to be flipped by the channel. The question is how much information can we send on the channel with this level of noise. Naturally, a channel would have some capacity constraints (say, at most 4,000 bits per second can be sent on the channel), and the question is how to send the largest amount of information, so that the receiver can recover the original information sent.

Now, its important to realize that noise handling is unavoidable in the real world. Furthermore, there are tradeoffs between channel capacity and noise levels (i.e., we might be able to send considerably more bits on the channel but the probability of flipping (i.e., p) might be much larger). In designing a communication protocol over this channel, we need to figure out where is the optimal choice as far as the amount of information sent.

Definition 45.1.2. A (k,n) ***encoding function*** $\mathtt{Enc} : \{0,1\}^k \to \{0,1\}^n$ takes as input a sequence of k bits and outputs a sequence of n bits. A (k,n) ***decoding function*** $\mathtt{Dec} : \{0,1\}^n \to \{0,1\}^k$ takes as input a sequence of n bits and outputs a sequence of k bits.

Thus, the sender would use the encoding function to send its message, and the decoder would use the received string (with the noise in it), to recover the sent message. Thus, the sender starts with a message with k bits, it blow it up to n bits, using the encoding function, to get some robustness to noise, it send it over the (noisy) channel to the receiver. The receiver, takes the given (noisy) message with n bits, and use the decoding function to recover the original k bits of the message.

Naturally, we would like k to be as large as possible (for a fixed n), so that we can send as much information as possible on the channel. Naturally, there might be some failure probability; that is, the receiver might be unable to recover the original string, or recover an incorrect string.

The following celebrated result of Shannon[45.1] in 1948 states exactly how much information can be sent on such a channel.

Theorem 45.1.3 (Shannon's theorem). *For a binary symmetric channel with parameter $p < 1/2$ and for any constants $\delta, \gamma > 0$, where n is sufficiently large, the following holds:*

(i) For an $k \leq n(1 - \mathbb{H}(p) - \delta)$ there exists (k,n) encoding and decoding functions such that the probability the receiver fails to obtain the correct message is at most γ for every possible k-bit input messages.

(ii) There are no (k,n) encoding and decoding functions with $k \geq n(1 - \mathbb{H}(p) + \delta)$ such that the probability of decoding correctly is at least γ for a k-bit input message chosen uniformly at random.

45.2. Proof of Shannon's theorem

The proof is not hard, but requires some care, and we will break it into parts.

[45.1] Claude Elwood Shannon (April 30, 1916 - February 24, 2001), an American electrical engineer and mathematician, has been called "the father of information theory".

45.2.1. How to encode and decode efficiently

45.2.1.1. The scheme Our scheme would be simple. Pick

$$k \leq n(1 - \mathbb{H}(p) - \delta).$$

For any number $i = 0, \ldots, \widehat{K} = 2^{k+1} - 1$, randomly generate a binary string Y_i made out of n bits, each one chosen independently and uniformly. Let $Y_0, \ldots, Y_{\widehat{K}}$ denote these codewords.

For each of these codewords we will compute the probability that if we send this codeword, the receiver would fail. Let $X_0, \ldots, X_K$, where $K = 2^k - 1$, be the K codewords with the lowest probability of failure. We assign these words to the 2^k messages we need to encode in an arbitrary fashion. Specifically, for $i = 0, \ldots, 2^k - 1$, we encode i as the string X_i.

The decoding of a message w is done by going over all the codewords, and finding all the codewords that are in (Hamming) distance in the range $[p(1-\varepsilon)n, p(1+\varepsilon)n]$ from w. If there is only a single word X_i with this property, we return i as the decoded word. Otherwise, if there are no such word or there is more than one word then the decoder stops and report an error.

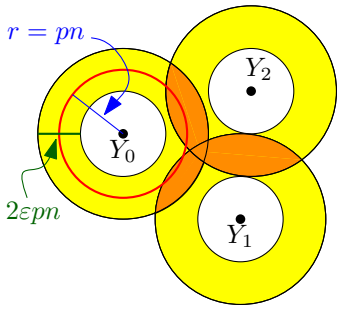


Figure 45.1: The rings of error.

45.2.1.2. The proof **Intuition.** Each code Y_i corresponds to a region that looks like a ring. The "ring" for Y_i is all the strings in Hamming distance between $(1-\varepsilon)r$ and $(1+\varepsilon)r$ from Y_i, where $r = pn$. Clearly, if we transmit a string Y_i, and the receiver gets a string inside the ring of Y_i, it is natural to try to recover the received string to the original code corresponding to Y_i. Naturally, there are two possible bad events here:

(A) The received string is outside the ring of Y_i.

(B) The received string is contained in several rings of different Ys, and it is not clear which one should the receiver decode the string to. These bad regions are depicted as the darker regions in Figure 45.1.

Let $S_i = \mathcal{S}(Y_i)$ be all the binary strings (of length n) such that if the receiver gets this word, it would decipher it to be the original string assigned to Y_i (here are still using the extended set of codewords $Y_0, \ldots, Y_{\widehat{K}}$). Note, that if we remove some codewords from consideration, the set $\mathcal{S}(Y_i)$ just increases in size (i.e., the bad region in the ring of Y_i that is covered multiple times shrinks). Let W_i be the probability that Y_i was sent, but it was not deciphered correctly. Formally, let r denote the received word. We have that

$$W_i = \sum_{r \notin S_i} \mathbb{P}[r \text{ was received when } Y_i \text{ was sent}]. \tag{45.1}$$

To bound this quantity, let $\Delta(x, y)$ denote the Hamming distance between the binary strings x and y. Clearly, if x was sent the probability that y was received is

$$w(x, y) = p^{\Delta(x,y)}(1-p)^{n-\Delta(x,y)}.$$

As such, we have

$$\mathbb{P}[r \text{ received when } Y_i \text{ was sent}] = w(Y_i, r).$$

Definition 45.2.1. Let $\overline{S_{i,r}}$ be an indicator variable which is 1 if $r \notin S_i$. It is one if the receiver gets r, and does not decode it to Y_i (either because of failure, or because r is too close/far from Y_i).

We have that failure probability when sending r is

$$W_i = \sum_{r \notin S_i} \mathbb{P}[r \text{ received when } Y_i \text{ was sent}] = \sum_{r \notin S_i} w(Y_i, r) = \sum_r \overline{S_{i,r}} w(Y_i, r). \tag{45.2}$$

The value of W_i is a random variable over the choice of $Y_0, \ldots, Y_{\widehat{K}}$. As such, its natural to ask what is the expected value of W_i.

Consider the ring

$$\operatorname{ring}(r) = \left\{ x \in \{0,1\}^n \;\middle|\; (1-\varepsilon)np \le \Delta(x,r) \le (1+\varepsilon)np \right\},$$

where $\varepsilon > 0$ is a small enough constant. Observe that $x \in \operatorname{ring}(y)$ if and only if $y \in \operatorname{ring}(x)$. Suppose, that the code word Y_i was sent, and r was received. The decoder returns the original code associated with Y_i, if Y_i is the only codeword that falls inside $\operatorname{ring}(r)$.

Lemma 45.2.2. *Given that Y_i was sent, and r was received and furthermore $r \in \operatorname{ring}(Y_i)$, then the probability of the decoder failing, is*

$$\tau = \mathbb{P}\left[r \notin S_i \;\middle|\; r \in \operatorname{ring}(Y_i) \right] \le \frac{\gamma}{8},$$

where γ is the parameter of Theorem 45.1.3.

Proof: The decoder fails here, only if $\operatorname{ring}(r)$ contains some other codeword Y_j ($j \neq i$) in it. As such,

$$\begin{aligned}
\tau &= \mathbb{P}\left[r \notin S_i \;\middle|\; r \in \operatorname{ring}(Y_i) \right] \\
&\le \mathbb{P}\left[Y_j \in \operatorname{ring}(r), \text{ for any } j \neq i \right] \\
&\le \sum_{j \neq i} \mathbb{P}\left[Y_j \in \operatorname{ring}(r) \right].
\end{aligned}$$

Now, we remind the reader that the Y_js are generated by picking each bit randomly and independently, with probability $1/2$. As such, we have

$$\mathbb{P}\left[Y_j \in \operatorname{ring}(r) \right] = \frac{|\operatorname{ring}(r)|}{|\{0,1\}^n|} = \sum_{m=(1-\varepsilon)np}^{(1+\varepsilon)np} \frac{\binom{n}{m}}{2^n} \le \frac{n}{2^n} \binom{n}{\lfloor (1+\varepsilon)np \rfloor},$$

since $(1+\varepsilon)p < 1/2$ (for ε sufficiently small), and as such the last binomial coefficient in this summation is the largest. By Corollary 45.3.2 (i), we have

$$\mathbb{P}\left[Y_j \in \operatorname{ring}(r) \right] \le \frac{n}{2^n} \binom{n}{\lfloor (1+\varepsilon)np \rfloor} \le \frac{n}{2^n} 2^{n\mathbb{H}\left((1+\varepsilon)p\right)} = n 2^{n\left(\mathbb{H}\left((1+\varepsilon)p\right)-1\right)}.$$

As such, we have

$$\tau = \mathbb{P}\left[r \notin S_i \;\middle|\; r \in \operatorname{ring}(Y_i) \right]$$

$$
\begin{aligned}
&\leq \sum_{j \neq i} \mathbb{P}\big[Y_j \in \operatorname{ring}(r)\big] \\
&\leq \widehat{K}\, \mathbb{P}\big[Y_1 \in \operatorname{ring}(r)\big] \\
&\leq 2^{k+1} n 2^{n\left(\mathbb{H}\left((1+\varepsilon)p\right)-1\right)} \\
&\leq n 2^{n\left(1-\mathbb{H}(p)-\delta\right)+1+n\left(\mathbb{H}\left((1+\varepsilon)p\right)-1\right)} \\
&\leq n 2^{n\left(\mathbb{H}\left((1+\varepsilon)p\right)-\mathbb{H}(p)-\delta\right)+1}
\end{aligned}
$$

since $k \leq n(1-\mathbb{H}(p)-\delta)$. Now, we choose ε to be a small enough constant, so that the quantity $\mathbb{H}((1+\varepsilon)p) - \mathbb{H}(p) - \delta$ is equal to some (absolute) negative (constant), say $-\beta$, where $\beta > 0$ (importantly, δ is fixed, otherwise the claim is of course false). Then, $\tau \leq n 2^{-\beta n+1}$, and choosing n large enough, we can make τ smaller than $\gamma/8$, as desired. As such, we just proved that

$$
\tau = \mathbb{P}\big[r \notin S_i \;\big|\; r \in \operatorname{ring}(Y_i)\big] \leq \frac{\gamma}{8}.
$$



Lemma 45.2.3. *Consider the situation where Y_i is sent, and the received string is r. We have that*

$$
\mathbb{P}\big[r \notin \operatorname{ring}(Y_i)\big] = \sum_{r \notin \operatorname{ring}(Y_i)} w(Y_i, r) \leq \frac{\gamma}{8},
$$

where γ is the parameter of Theorem 45.1.3.

Proof: This quantity, is the probability of sending Y_i when every bit is flipped with probability p, and receiving a string r such that more than $pn + \varepsilon pn$ bits where flipped (or less than $pn - \varepsilon pn$). But this quantity can be bounded using the Chernoff inequality. Indeed, let $Z = \Delta(Y_i, r)$, and observe that $\mathbb{E}[Z] = pn$, and it is the sum of n independent indicator variables. As such

$$
\sum_{r \notin \operatorname{ring}(Y_i)} w(Y_i, r) = \mathbb{P}\big[|Z - \mathbb{E}[Z]| > \varepsilon pn\big] \leq 2 \exp\left(-\frac{\varepsilon^2}{4} pn\right) < \frac{\gamma}{4},
$$

since ε is a constant, and for n sufficiently large.



We remind the reader that $\overline{S_{i,r}}$ is an indicator variable that is one if receiving r (when sending Y_i) is "bad", see Definition 45.2.1. Importantly, this indicator variable also depends on all the other codewords – as they might cause some regions in the ring of Y_i to be covered multiple times.

Lemma 45.2.4. *We have that* $f(Y_i) = \sum_{r \notin \text{ring}(Y_i)} \mathbb{E}\big[\overline{S_{i,r}} w(Y_i, r)\big] \leq \gamma/8$ *(the expectation is over all the choices of the* Y*s excluding* Y_i*).*

Proof: Observe that $\overline{S_{i,r}} w(Y_i, r) \leq w(Y_i, r)$ and for fixed Y_i and r we have that $\mathbb{E}[w(Y_i, r)] = w(Y_i, r)$. As such, we have that

$$\begin{aligned} f(Y_i) &= \sum_{r \notin \text{ring}(Y_i)} \mathbb{E}\Big[\overline{S_{i,r}} w(Y_i, r)\Big] \\ &\leq \sum_{r \notin \text{ring}(Y_i)} \mathbb{E}[w(Y_i, r)] \\ &= \sum_{r \notin \text{ring}(Y_i)} w(Y_i, r) \leq \frac{\gamma}{8}, \end{aligned}$$

by Lemma 45.2.3.

Lemma 45.2.5. *We have that* $g(Y_i) = \sum_{r \in \text{ring}(Y_i)} \mathbb{E}\Big[\overline{S_{i,r}} w(Y_i, r)\Big] \leq \gamma/8$ *(the expectation is over all the choices of the* Y*s excluding* Y_i*).*

Proof: We have that $\overline{S_{i,r}} w(Y_i, r) \leq \overline{S_{i,r}}$, as $0 \leq w(Y_i, r) \leq 1$. As such, we have that

$$\begin{aligned} g(Y_i) &= \sum_{r \in \text{ring}(Y_i)} \mathbb{E}\Big[\overline{S_{i,r}} w(Y_i, r)\Big] \\ &\leq \sum_{r \in \text{ring}(Y_i)} \mathbb{E}\Big[\overline{S_{i,r}}\Big] \\ &= \sum_{r \in \text{ring}(Y_i)} \mathbb{P}[r \notin S_i] \\ &= \sum_{r} \mathbb{P}[r \notin S_i \cap (r \in \text{ring}(Y_i))] \\ &= \sum_{r} \mathbb{P}\big[r \notin S_i \;\big|\; r \in \text{ring}(Y_i)\big]\, \mathbb{P}[r \in \text{ring}(Y_i)] \\ &\leq \sum_{r} \frac{\gamma}{8}\, \mathbb{P}[r \in \text{ring}(Y_i)] \leq \frac{\gamma}{8}, \end{aligned}$$

by Lemma 45.2.2. 

Lemma 45.2.6. *For any i, we have $\mu = \mathbb{E}[W_i] \leq \gamma/4$, where γ is the parameter of Theorem 45.1.3, where W_i is the probability of failure to recover Y_i if it was sent, see Eq. (45.1).*

Proof: We have by Eq. (45.2) that $W_i = \sum_r \overline{S_{i,r}} w(Y_i, r)$. For a fixed value of Y_i, we have by linearity of expectation, that

$$\begin{aligned}
\mathbb{E}[W_i \mid Y_i] &= \mathbb{E}\Big[\sum_r \overline{S_{i,r}} w(Y_i, r) \;\Big|\; Y_i\Big] \\
&= \sum_r \mathbb{E}\Big[\overline{S_{i,r}} w(Y_i, r) \;\Big|\; Y_i\Big] \\
&= \sum_{r \in \operatorname{ring}(Y_i)} \mathbb{E}\Big[\overline{S_{i,r}} w(Y_i, r) \;\Big|\; Y_i\Big] \\
&\quad + \sum_{r \notin \operatorname{ring}(Y_i)} \mathbb{E}\Big[\overline{S_{i,r}} w(Y_i, r) \;\Big|\; Y_i\Big] \\
&= g(Y_i) + f(Y_i) \\
&\leq \frac{\gamma}{8} + \frac{\gamma}{8} = \frac{\gamma}{4},
\end{aligned}$$

by Lemma 45.2.4 and Lemma 45.2.5. Now $\mathbb{E}[W_i] = \mathbb{E}[\mathbb{E}[W_i \mid Y_i]] \leq \mathbb{E}[\gamma/4] \leq \gamma/4$. 

In the following, we need the following trivial (but surprisingly deep) observation.

Observation 45.2.7. *For a random variable X, if $\mathbb{E}[X] \leq \psi$, then there exists an event in the probability space, that assigns X a value $\leq \psi$.*

Lemma 45.2.8. *For the codewords $X_0, \ldots, X_K$, the probability of failure in recovering them when sending them over the noisy channel is at most γ.*

Proof: We just proved that when using $Y_0, \ldots, Y_{\widehat{K}}$, the expected probability of failure when sending Y_i, is $\mathbb{E}[W_i] \leq \gamma/4$, where $\widehat{K} = 2^{k+1} - 1$. As such, the expected total probability of failure is

$$\mathbb{E}\Big[\sum_{i=0}^{\widehat{K}} W_i\Big] = \sum_{i=0}^{\widehat{K}} \mathbb{E}[W_i] \leq \frac{\gamma}{4} 2^{k+1} \leq \gamma 2^k,$$

by Lemma 45.2.6. As such, by Observation 45.2.7, there exist a choice of Y_is, such that

$$\sum_{i=0}^{\widehat{K}} W_i \leq 2^k \gamma.$$

Now, we use a similar argument used in proving Markov's inequality. Indeed, the W_i are always positive, and it can not be that 2^k of them have value larger than γ, because in the summation, we will get that

$$\sum_{i=0}^{\widehat{K}} W_i > 2^k \gamma.$$

Which is a contradiction. As such, there are 2^k codewords with failure probability smaller than γ. We set the 2^k codewords $X_0, \ldots, X_K$ to be these words, where $K = 2^k - 1$. Since we picked only a subset of the codewords for our code, the probability of failure for each codeword shrinks, and is at most γ.

Lemma 45.2.8 concludes the proof of the constructive part of Shannon's theorem.

45.2.2. Lower bound on the message size

We omit the proof of this part. It follows similar argumentation showing that for every ring associated with a codewords it must be that most of it is covered only by this ring (otherwise, there is no hope for recovery). Then an easy packing argument implies the claim.

45.3. From previous lectures

Lemma 45.3.1. *Suppose that nq is integer in the range $[0, n]$. Then* $\frac{2^{n\mathbb{H}(q)}}{n+1} \leq \binom{n}{nq} \leq 2^{n\mathbb{H}(q)}$.

Lemma 45.3.1 can be extended to handle non-integer values of q. This is straightforward, and we omit the easy details.

Corollary 45.3.2. *We have:*

(i) $q \in [0, 1/2] \Rightarrow \binom{n}{\lfloor nq \rfloor} \leq 2^{n\mathbb{H}(q)}$.

(ii) $q \in [1/2, 1]$ $\binom{n}{\lceil nq \rceil} \leq 2^{n\mathbb{H}(q)}$.

(iii) $q \in [1/2, 1] \Rightarrow \frac{2^{n\mathbb{H}(q)}}{n+1} \leq \binom{n}{\lfloor nq \rfloor}$.

(iv) $q \in [0, 1/2] \Rightarrow \frac{2^{n\mathbb{H}(q)}}{n+1} \leq \binom{n}{\lceil nq \rceil}$.

Theorem 45.3.3. *Suppose that the value of a random variable X is chosen uniformly at random from the integers $\{0, \ldots, m-1\}$. Then there is an extraction function for X that outputs on average at least $\lfloor \lg m \rfloor - 1 = \lfloor \mathbb{H}(X) \rfloor - 1$ independent and unbiased bits.*

45.4. Bibliographical Notes

The presentation here follows [MU05, Sec. 9.1-Sec 9.3].

References

[MU05] M. Mitzenmacher and E. Upfal. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005.