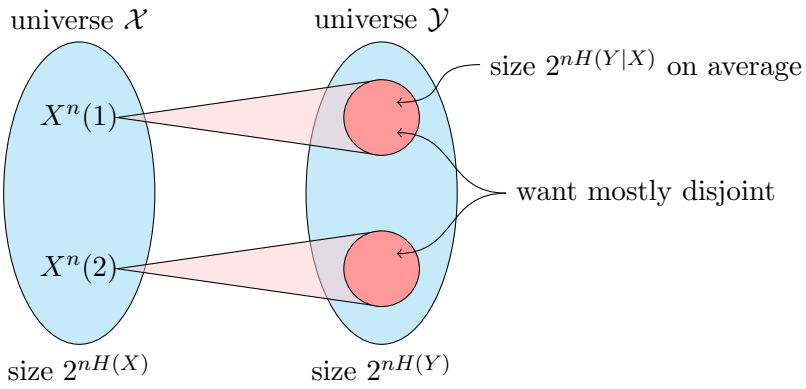
15-859: Information Theory and Applications in TCS **CMU: Spring 2013**

Lecture 8: Channel and source-channel coding theorems; BEC & linear codes

February 7, 2013

Lecturer: Venkatesan Guruswami *Scribe: Dan Stahlke*

1 Intuitive justification for upper bound on channel capacity



Suppose that we are sending codewords from a set X^n into a channel and we receive things from Y^n. The number of different codewords we have available at the input side of the channel is approximately $2^{nH(x)}$, the number of typical sequences. Similarly the number of possibilities at the output is approximately $2^{nH(Y)}$. Each of the input codewords can go to several different outputs, on average there are about $2^{nH(Y|X)}$ possible outputs each input can be sent to by the channel. If we hope for the images of each of the inputs to be mostly disjoint, then the number of inputs times the size of their image under the channel must be less than the size of the channel output. This gives

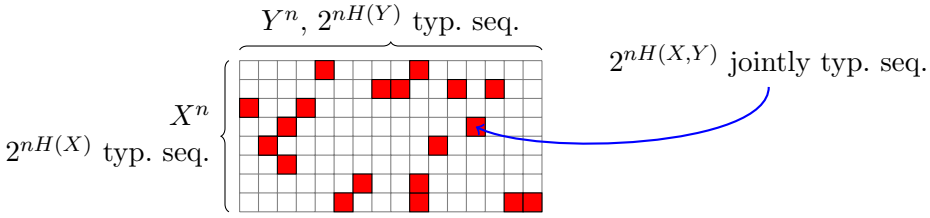
$$\#\text{ codewords} \leq \frac{2^{nH(Y)}}{2^{nH(Y|X)}} = 2^{nI(X;Y)}, \tag{1}$$

therefore the capacity is no greater than $I(X;Y)$ (maximized over all input probability distributions $P(X)$).

2 Achieveability

This is an intuitive explanation of why channel capacity of $C = \max_{P(X)} I(X;Y)$ is achievable. Make a matrix with a row for each typical input sequence X^n (there are $2^{nH(X)}$ of these) and a column for each typical output sequence Y^n (there are $2^{nH(Y)}$ of these). Mark each cell that corresponds to a jointly typical sequence (there are $2^{nH(X,Y)}$ of these).

Suppose we transmit $X^n(1)$ and receive Y^n. Then it is likely that $(X^n(1), Y^n)$ is jointly typical, and so the receiver can determine $X^1(n)$ as long as there is no other possible input $X^n(2)$ such



that $(X^n(2), Y^n)$ is jointly typical. The density of the jointly typical sequences as depicted in the matrix diagram is the number of jointly typical sequences divided by the size of the matrix,

$$\frac{2^{nH(X,Y)}}{2^{nH(X)}2^{nH(Y)}} = 2^{-nI(X;Y)}. \tag{2}$$

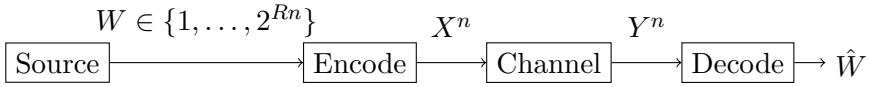
So if the number of possible input codewords is 2^{nR} with $R = I(X;Y) - \epsilon$ then the probability of error is

$$2^{nR}2^{-nI(X;Y)} \leq 2^{-\epsilon n} \to 0 \text{ as } n \to \infty. \tag{3}$$

3 Upper bound on rate

We will show that communication is only possible at rates that don't exceed channel capacity. First of all, it is interesting to note that plotting P_e against rate gives something resembling a step function with P_e jumping rather abruptly from close to 0 (due to the achievability shown last time) to close to 1 (due to what we will show now) as the rate goes above channel capacity.

Suppose a source $W \in \{1, \ldots, 2^{Rn}\}$ is to be sent through a channel:



We will show that if $P_e = \Pr(\hat{W} \neq W) \to 0$ as $n \to \infty$ then $R \leq I(X;Y)$. As a warm-up exercise, first consider the case $P_e = 0$. Take W to be uniform on $\{1, \ldots, 2^{Rn}\}$. Call the decoding operation g. Since we have perfect decoding (i.e. $P_e = 0$), we get $W = \hat{W} = g(Y^n) \implies H(W|Y^n) = 0$, leading to

$$\begin{aligned} nR &= H(W) && (4)\\ &= H(W|Y^n) + I(W;Y^n) && (5)\\ &= I(W;Y^n). && (6) \end{aligned}$$

Now apply the data processing inequality. Since we have the Markov chain $W \to X^n \to Y^n$, we get $I(W;Y^n) \leq I(X^n;Y^n)$. It is tempting at this point to just say $nR \leq I(X^n;Y^n) = nI(X_i;Y_i)$ and call it quits, however mutual information doesn't work that way in general (in this case it does,

but this requires proof!) What we have is

$$\begin{aligned} nR &\leq I(X^n;Y^n) && (7)\\ &= H(Y^n) - H(Y^n|X^n) && (8)\\ &\leq \sum_{i=1}^{n} H(Y_i) - H(Y^n|X^n) && (9)\\ &= \sum_{i=1}^{n} H(Y_i) - \sum_{i=1}^{n} H(Y_i|Y_1,\ldots,Y_{i-1},X^n). \qquad \text{(chain rule)} && (10) \end{aligned}$$

Since the channel is memoryless, Y_i is independent of $X_1,\ldots,X_{i-1}$ and $Y_1,\ldots,Y_{i-1}$ (i.e. Y_i only depends on X_i) so this reduces to

$$\begin{aligned} nR &\leq \sum_{i=1}^{n} H(Y_i) - \sum_{i=1}^{n} H(Y_i|Y_1,\ldots,Y_{i-1},X^n) && (11)\\ &= \sum_{i=1}^{n} H(Y_i) - \sum_{i=1}^{n} H(Y_i|X_i) && (12)\\ &= \sum_{i=1}^{n} I(X_i;Y_i) && (13)\\ &= nI(X_i;Y_i) && (14)\\ &\leq nC. && (15) \end{aligned}$$

Therefore $R \leq C$.

Now for the general case, with $P_e \neq 0$. What changes now is that we no longer have $W = g(Y^n)$ and so $H(W|Y^n) \neq 0$ (but $\hat{W} = g(Y^n)$ still holds). Recycling the previous work from the warm-up exercise gives $nR \leq H(W|Y^n) + nC$. Fano's inequality applies here, giving

$$h(P_e) + P_e \log(2^{nR} - 1) \geq H(W|Y^n). \tag{16}$$

This is a bit too complicated, so simplify it with the bounds $h(P_e) \leq 1$ and $\log(2^{nR} - 1) \leq nR$ to get

$$1 + nRP_e \geq H(W|Y^n). \tag{17}$$

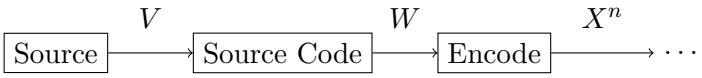
Plugging this into $nR \leq nC + H(W|Y^n)$ gives $R \leq C + 1/n + RP_e$. Then, as $n \to \infty$ and $P_e \to 0$, this becomes $R \leq C + o(1)$.

Comment: If $R < C$ there exists a code with $P_e \leq 2^{-\Theta(R,C)n}$ for $n \to \infty$ where Θ is some constant depending on R and C.

Comment: If $R > C$ then for all coding/decoding schemes, $P_e \to 1$ as $n \to \infty$ (note: $n \to \infty$ is essential since with $n = 1$ a coin toss gives $P_e = 1/2$).

4 Joint source coding (a.k.a. source-channel separation) theorem

Often we want to communicate a source that is not uniformly distributed. Consider a source $\mathcal{V}$ and define $H := H(\mathcal{V})$. We could first compress this source using a source code, and then encode that with a channel encoding:



The source code is possible if $R > H$ and transmission is possible if $R < C$, so this two stage process is possible if $C > H$. However, is it possible to do better by combining the source coding and channel encoding? In fact this would not help, and we will show that $C > H$ is both necessary and sufficient for communication (sufficient is clear because of the two-stage process, necessary is what we will now show).

Theorem 1. *If $V_1, \ldots, V_n$ are i.i.d. samples from $\mathcal{V}$ then there exists a source-channel code with $P_e = \Pr(\hat{V}^n \neq V^n) \to 0$ if and only if $C > H(\mathcal{V})$.*

Proof. By the definition of mutual information, $nH(\mathcal{V}) = H(V^n) = H(V^n|\hat{V}^n) + I(V^n; \hat{V}^n)$.[1] Fano's inequality bounds the first term,

$$H(V^n|\hat{V}^n) \leq h(P_e) + P_e \log(|V^n| - 1) \tag{18}$$
$$\leq 1 + P_e n \log(|V^n|). \tag{19}$$

The data processing inequality applies to the second term, $I(V^n; \hat{V}^n)$, because of the Markov chain $V^n \to X^n \to Y^n \to \hat{V}^n$, giving $I(V^n; \hat{V}^n) \leq I(X^n; Y^n)$. Putting it all together,

$$nH(\mathcal{V}) = H(V^n|\hat{V}^n) + I(V^n; \hat{V}^n) \tag{20}$$
$$\leq [1 + P_e n \log(|V^n|)] + I(X^n; Y^n) \tag{21}$$
$$\leq 1 + P_e n \log(|V^n|) + nC. \tag{22}$$

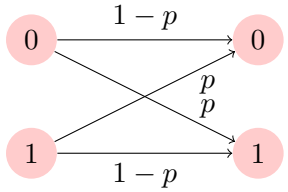
Taking the limit $n \to \infty$ and $P_e \to 0$ then gives $H(\mathcal{V}) \leq C$. □

[1] Cover+Thomas has $nH(\mathcal{V}) \leq H(V^n)$ rather than equality, but I don't understand why.

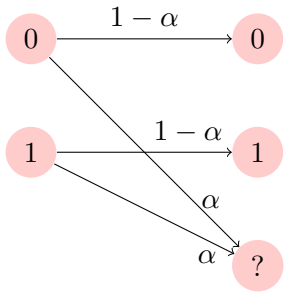
5 Linear codes

The code presented last class is slow, and it takes exponential space to even store the codebook. Linear codes allow for efficient codebook storage and, in some cases, efficient decoding. We will focus on two particular channels:

Binary symmetric channel (BSC_p)
Capacity $C = 1 - h(p)$



Binary erasure channel (BEC_α)
Capacity $C = 1 - \alpha$



We will encode $\{1, \ldots, 2^{Rn}\} \to X^n = \{0,1\}^n$. For simplicity, assume that $k = Rn$ is an integer. This assumption doesn't hurt in the $n \to \infty$ limit. With this, we can say we are encoding $\{1, \ldots, 2^k\} \to X^n = \{0,1\}^n$, in other words, from a k bit string to an n bit string. Think of these bit strings as vectors. The encoding operation will be a linear transformation, $\underline{x} \to G\underline{x}$ where G is an $n \times k$ matrix. Arithmetic is done modulo 2 here, or in other words we are working in the finite field $\mathbb{F}_2$. Such a code is called a *linear code*. Note that G can be stored efficiently. The claim (which we do not prove here) is that for both BEC_α and BSC_p the linear code obtained through choosing a random G will achieve the channel capacity.

How to decode? One option is to use joint typicality. For BSC_p, $(\underline{a}, \underline{b})$ is jointly typical if $\Delta(\underline{a}, \underline{b}) \approx pn$ where Δ is the Hamming distance. So to decode $\underline{y} \in \{0,1\}^n$, just search for $\underline{x} \in \{0,1\}^k$ such that $(p - \epsilon) \leq \Delta(G\underline{x}, \underline{y}) \leq (p + \epsilon)$. Actually, one need not be concerned with picking out vectors that are too similar, so it works just as well to use $\Delta(G\underline{x}, \underline{y}) \leq (p + \epsilon)$. But finding such vectors directly is hard to do.

Another option is maximum likelihood decoding: find the $\underline{x}$ which minimizes $\Delta(G\underline{x}, \underline{y})$. This is also hard to do (in general NP-hard). It always works, but is harder to deal with than joint typicality.

For BEC_α there is an easy decoding strategy. This channel is nice enough to tell us where the errors are (e.g. it sends $01001011 \to 01?0??11$). Just throw out the erased bits to get a reduced vector $\underline{y}_S$, where S is the set of non-erased positions. Also filter the rows of G to get G_S. The remaining bits went through with no error, so we have $\underline{y}_S = G_S\underline{x}$ exactly. If $|S|$ is a bit larger than k (i.e. G_S has a bit more rows than columns) then G_S is likely to be full rank and $\underline{y}_S = G_S\underline{x}$ can be solved for the unique solution of $\underline{x}$ using standard linear algebra techniques.