CS 229r Essential Coding Theory March 30, 2020

Lecture 16

Instructor: Madhu Sudan *Scribe: Juspreet Singh Sandhu*

1 Today

1.1 Admin

- Grading: Will count 4 best psets out of 6
- PS-4 due Friday; PS-5 due April-13
- PS-6: Optional

1.2 Polar Codes

- Motivation: Efficient Coding & Decoding in the Gap to Capacity (ε).
- Construction:
  1. Reduction to Linear Compression
  2. Polarizing Transformation + Information-Theoretic basis

2 Motivation

Recall the familiar $BSC(p)$ channel and what we know about it via Shannon:

1. $\Pr[\hat{m} = m] \geq 1 - o(1)$ - We can, w.h.p., decode a message with errors introduced by the channel (that look Gaussian as $n \to \infty$).
2. We can do this with $R \to 1 - H(p)$. More concretely, for rate $R = (1 - H(p) - \varepsilon)$, we can achieve error-probability $\approx \exp(-\varepsilon' n)$. The $(E, D)_{k,n}$ maps that achieve this are proved to exist by the Probabilistic Method (non-constructively).
3. In *Pset-3*, we show that we can, in fact, encode and decode in $poly(n)$ time using code concatenation: $C_{out} = RS[n, (1-\varepsilon)n]_n$, C_{in} is a random binary code of rate R over $\log(n)$ bits.

Unfortunately, while the encoding and decoding schemes are polynomial in n, they are exponential in $\frac{1}{\varepsilon}$ - We end up requiring $\approx n^2 2^{\frac{1}{\varepsilon^2}}$ time since the blocks are of size $\frac{1}{\varepsilon^2}$. The problem is, therefore, formulated only in terms of ε.

3 Problem

The problem, informally stated, asks for Encoding & Decoding maps for a coding scheme that have rate ε close to the Capacity of the $BSC(p)$-channel and also run in time $poly(\frac{1}{\varepsilon})$. More formally:

$\forall \varepsilon > 0$, determine k, n that satisfy $\frac{k}{n} \geq 1 - H(p) - \varepsilon$ and $\exists\ E_k : \{0,1\}^k \to \{0,1\}^n$, $D_n : \{0,1\}^n \to \{0,1\}^k$, such that:

- $\Pr_{m,BSC}[D_n(BSC(E_k(m))) \neq m] \geq 1 - o(1)$

- Time(E_k), Time(D_n) $= poly(\frac{1}{\varepsilon})$

The problem has a Shannon-like flavor, so we average over all messages (in addition to the channel) - This is not strictly necessary, as Linear Codes will achieve this Rate averaged over just the channels (and deal with adversarial messages).

4 History

1. The question about the existence of these (E, D)-schemes was initially raised by *[Luby, Mitzenmacher, Shokrollahi, Spielman].*
2. In 2008, *[Arikan]* proposed polar codes (although, not as an answer to the open-question previously posed).
3. In 2013, *[Guruswami, Xia]* & *[Hassani, Alishai, Urbanki]* resolved the open question using the aforementioned construction of polar codes.

5 Construction

We begin to cover the construction of Polar Codes. As we will prove in the coming lectures, this construction will turn out to have an efficient $poly(\frac{1}{\varepsilon})$ randomized algorithm that will help decode the received codewords (w.h.p.).

We fist show that our problem as stated can be reduced to the problem of finding a 'good' linear-compression (via the cosntruction of a 'good' parity-check matrix). We then describe the construction proposed by Arikan, and show why it 'polarizes' entropy by shifting it around. We end with a few claims that we shall prove in the subsequent lectures.

5.1 Reduction to Linear Compression

We let:

- $Z = Z_1..Z_n$ denote a string of n *i.i.d.* drawn Ber(p) random variables.
- $X = X_1..X_k$ be the message we want to transmit.
- (G, H) represent the generator matrices of a linear code.

We say that the generator matrix G is a generator of a good code $\iff$ H (Parity-Check matrix) is a 'good' compressor of Ber$(p)^{\otimes n}$. We will often be sloppy here in differentiating between H and H^T w.r.t. our previous lectures. For this lecture, $H \in \mathbb{F}_2^{m\times n}$. We now define what a 'good' compressor is.

Definition 1 (Good Compressor)**.** A compression scheme $C : \mathbb{F}_2^n \to \mathbb{F}_2^m$ is a good compressor (for Bern(p)$^{\mathrm{n}}$) if

$$C \text{ is linear, i.e., } \exists H \in \mathbb{F}_2^{n\times m} \text{ s.t. } C(Z) = ZH. \tag{1}$$

$$m \leq (H(p) + \varepsilon)n \tag{2}$$

$$\exists \text{ decompression algorithm D, st, } \Pr[D(C(Z)) \neq Z] = o(1) \tag{3}$$

$$\text{Time}(D) = \text{poly}(\frac{1}{\varepsilon}) \tag{4}$$

Given a "good compressor" as above, it is easy to see that the code generated by G (or parity-checked by H) is easy to decode on the binary symmetric channel. The transmission leads to the following sequence:

$$X \xrightarrow{\text{Encoding}} XG \xrightarrow{\text{Channel}} XG + Z = Y,$$

where $Z \sim \mathrm{Ber(p)^n}$. Given Y the decoding algorithm outputs $Y - D(YH)$ which equals the transmitted codeword XG w.h.p. as shown below:

$$Y - D(YH) = Y - D(XGH + ZH) = Y - D(ZH) =_{\text{w.h.p.}} Y - Z = GX,$$

where the first and last equality use the definition of Y, the second equality uses $GH = 0$ and the third (w.h.p.) equality from from the property of being a "good compressor". Note further that D runs in $poly(\frac{1}{\varepsilon})$ if H is a good compressor. This shows that finding a good compressor suffices to solve our decoding problem. Actually it turns out the problems are equivalent and we leave this as an exercise.

Exercise 2. *Prove that finding a good compressor H is equivalent to solving the original problem with a good linear code (G, H).*

If it were not for the fact that we expect the compression to be linear, satisfying the other conditions of Definition 1 are quite easy!

Exercise 3. *Demonstrate a non-linear compression scheme $C : \mathbb{F}_2^n \to \mathbb{F}_2^m$ that satisfies all properties of Definition 1 except (1).*

5.2 Polarizing Transformation

Having reduced the original problem to that of finding a good compression scheme with a Linear Code, we now show *Arikan's* original construction and give some information-theoretic intuition for why the construction shifts entropy around by concentrating it very strongly on a subset of bits (and extremely weakly on the remaining ones).
Idea: Take 2 bits; act on the first with a linear operator → Entropy of 1st bit increases; Entropy of 2nd bit (conditioned on the 1st) decreases.

Definition 4 (Polarizing Map)**.** Given $\mathrm{u} \sim \mathrm{Ber(p)}$, $\mathrm{v} \sim \mathrm{Ber(q)}$, $(\mathrm{u}, \mathrm{v}) \mapsto (\mathrm{u} \oplus \mathrm{v}, \mathrm{v})$

This map is clearly linear and invertible, so what have we gained? The answer, as suggested in the inuition, relies on reasoning about the entropy of the 2nd bit conditioned on the 1st, and the entropy of $u \oplus v$ vs. that of u or v.
We begin by recalling a few elementary definitios and properties:

Definition 5 (Entropy)**.** Given a random variable X with a probability distribution $\Pr[X = i] = p_i$, it has entropy:

$$H(X) = \mathop{\mathbb{E}}_{p_i}[\log(\frac{1}{p_i})] \tag{5}$$

The entropy captures how efficiently n *i.i.d.* copies of a source X can be compressed. In fact, we can state that $H(X_1, .., X_n) \approx n(H(X) \pm \varepsilon)$, where $X_1, .., X_n \sim_{r,i.i.d.} X$.

Definition 6 (Conditional Entropy)**.** Given 2 random variables X and Y with a joint distribution p_{XY} and marginals p_X, p_Y, we define conditional entropy as:

$$H(X|Y) = \mathop{\mathbb{E}}_{p_y}[H(X|Y = y)] \tag{6}$$

The conditional entropy may be considered as the reduction in uncertainty of X given knowledge of Y. Alternatively, it is the average uncertainty of the distribution of X weighted by the likelihood of observing particular instances of Y.

Lemma 7 (Chain Rule)**.** *Given random variables X and Y:*

$$H(X,Y) = H(X) + H(Y|X) = H(Y) + H(X|Y) \tag{7}$$

Lemma 8 (Permutations preserve Entropy)**.** *Given a permutation $\pi : im(X) \to im(X)$:*

$$H(X) = H(\pi(X)) \tag{8}$$

We are now in a position to understand how, if u, v are not equally-likely (or either being deterministic), $u \oplus v$ has higher entropy than either u or v. It is best to prove this as an exercise:

Exercise 9. *Given $u \sim Ber(p)$, $v \sim Ber(q)$, $p, q \neq 0, \frac{1}{2}, 1$, $H(u \oplus v) > H(u), H(v)$*

Hint: Start by understanding how $u \oplus v$ is paramterized by its own Bernoulli parameter; Ignore the edge cases.
To build further intuition for this, we see that if $p, q = .01$ (or some small constant), then $\Pr[u = 1|u \oplus v = 1] = \frac{1}{2}$, whereas $\Pr[u = 1|u \oplus v = 0] \approx 10^{-4}$. However, note that the former is a rarer event than the latter. As a result, on expectation, u will be less random than $u \oplus v$.
We now claim that we have sufficient ammunition to make our second claim:

Theorem 10 (The 2nd bit loses conditional entropy)**.** *Given u, v as drawn previously:*

$$H(v|u \oplus v) < H(v) \tag{9}$$

Proof. Note that $H(u, v) = H(u \oplus v, v)$. This follows from using the chain-rule (7) and then an application of 8. By the chain-rule (again):

$$H(v|u \oplus v) = H(u \oplus v, v) - H(u \oplus v) = H(u, v) - H(u \oplus v) \tag{10}$$

Note from 9 that $H(u \oplus v) > H(u)$. Applying this into the equation above yields:

$$H(v|u \oplus v) < H(u, v) - H(u) = H(v|u) = H(v) \tag{11}$$

QED □

We now notice that between 9 and 10 we have shown that our process is polarizing via our intuition. The idea now is to apply this process repetitively on an input string of n bits till we have an output string where every bit has an entropy that either very close to 1 or very close to 0. This can be visualized in **Figure-5.2** We label the output bits of the procedure on the right as $W = W_1..W_n$. The input bits on the left can be thought of as $Z = Z_1..Z_n$. We now ask the following question:

Question 11. *What is $H(W_i|W_{<i})$?*

As we shall see, for the required number of coordinates, the answer will be that this conditional entropy will be very close to 1 or 0 (in keeping with our intuition about polarization).

6 Conclusion & Future

We end by definting a set of coordinates of high entropy and stating that the compression of Z can be viewed as the restriction of the output W to this state, and then stating two claims that we shall prove in the next few lectures.

Definition 12 (High-Entropy Set)**.**

$$S = \{i | H(W_i|W_{<i}) \to 1\} \tag{12}$$

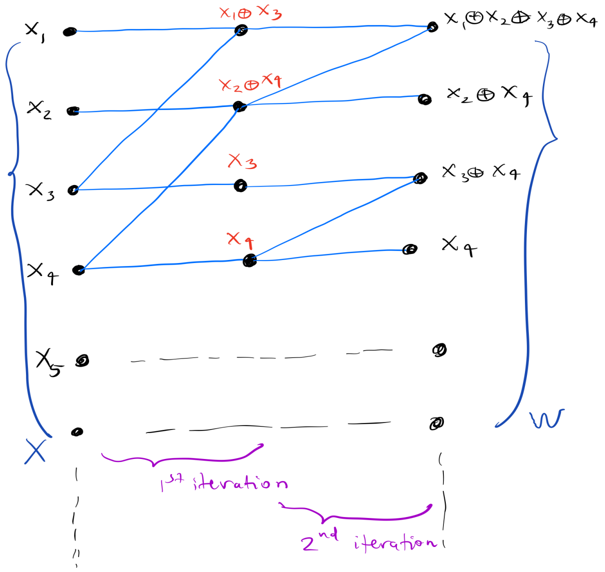

Figure 1: Polarization process

We will prove the following 2 claims in the coming lectures:

Lemma 13 (Polarization compresses near-optimally)**.**

$$|S| \approx n(H(p) \pm \varepsilon) \tag{13}$$

Lemma 14 (Polarization allows for efficient recovery of noise)**.** *Given $W|_S$ (the restriction of W to high-entropy coordinates), we can compute W. Given W, we can compute Z efficiently (w.h.p.).*

Notice that Lemma-13 is stating that the polarization process achieves the first property required of a 'good' compressor. Lemma-14 is far deeper, and says that provided we identify the subset of high-entropy coordinates in the polarized output W, we can efficiently recover (w.h.p.) its remaining coordinates - This can follow by inserting an erasure to be corrected by the deocding algorithm (D). The last task that remains is to recover Z after having constructed W (in $poly(\frac{1}{\varepsilon})$ time). At this point, it is also a good exercise to prove that the repetitive application of the polarizaton process is linear.