Problem Set 4

Information and Coding Theory

March 14, 2021 CLAYTON SEITZ

Problem 0.1. *More on linear codes*

A linear code is a subspace $C \subseteq \mathbb{F}_2^n$. The distance of such a code $\Delta(C)$ can be written as a minimization of the weight over all $x \in C$.

Solution.

$$\begin{aligned}\Delta(C) &= \min_{x_1, x_2 \in C} \Delta(x_1, x_2)\\ &= \min_{x_1, x_2 \in C} \Delta(0, x_2 - x_1)\\ &= \min_{x \in C \setminus \{0^n\}} \mathrm{wt}(x)\end{aligned}$$

Since we have required that C is linear i.e. $\forall x_1, x_2 \in C$ we have that $x_2 - x_1 \in C$ (C is closed under addition).

Now, we consider the general Hamming code $C : \mathbb{F}_2^k \to \mathbb{F}_2^n$ which maps messages w of length k to $\mathbb{F}_2^n$ via the generator matrix $G \in \mathbb{F}_2^{n \times k}$. We also define the parity check matrix $H \in \mathbb{F}_2^{r \times n}$ which has the property that for any encoded message x we have $Hx = 0$. Note this also means that the columns of G form a basis for the null space of H.

Since we are defining our code to be the null space of $H \in \mathbb{F}_2^{r \times n}$, the dimension of the code is given by

$$\begin{aligned}\dim(\mathrm{null}(H)) &= n - \mathrm{rank}(H)\\ &= n - r\\ &= 2^r - 1 - r\end{aligned}$$

Also, since H has r columns, the block length of a general Hamming code is $2^r - 1$. Finally, we can show that the distance of such a code is 3 by writing the Hamming bound and assuming we can correct $t = 1$ errors:

$$
\begin{aligned}
|C| &\leq \frac{2^n}{|B(0,t)|} \\
&= \frac{2^n}{|B(0,1)|} \\
&= 2^{n-r}
\end{aligned}
$$

where 2^{n-r} is indeed the size of the code according to $\dim(\text{null}(H))$ calculated above.

Finally, we will define the dual code $C^\perp$ to be the code with generator matrix H^T and parity check matrix G^T. As stated above, the generator matrix G is a matrix with columns equal to the basis vectors of the null space of H i.e. $HG = 0$. This is equivalent to saying that the columns of H^T form the basis of the null space of G^T:

$$HG = 0 \iff G^T H^T = 0$$

Therefore H^T can be viewed as the generator matrix and G^T the parity check matrix for the dual code $C^\perp$. We can repeat our calculations of dimension, block length, and minimum distance for the dual code $C^\perp$:

$$
\begin{aligned}
\dim(\text{null}(G^T)) &= n - \text{rank}(G^T) \\
&= n - k \\
&= 2^r - 1 - k
\end{aligned}
$$

and has block length $2^n - 1$ and minimum distance 3 by the same packing argument made above except the size of the dual code $|C^\perp| = 2^{n-k}$.

■

Problem 0.2. *Good distance codes from linear compression*

We would like to prove that an arbitrary compression algorithm Com : $\mathbb{F}_2^n \to \mathbb{F}_2^m$ implemented by the matrix $H \sim \mathbb{F}_2^{m\times n}$ yields a good compression scheme for a sequence $Z \sim (\text{Bern}(p))^n$ which means that there exists a decompression algorithm Decom : $\mathbb{F}_2^m \to \mathbb{F}_2^n$ with the following error bound

$$\mathop{\mathbf{Pr}}_{Z\sim(\text{Bern}(p))^n}[\text{Decom}(HZ) \neq Z] \leq 2^{-t}$$

Solution.

To prove such a bound, we can first rewrite the LHS of the above. Notice that since $m < n$ (compression) we cannot have a Decom : $\mathbb{F}_2^m \to \mathbb{F}_2^n$ that is a bijection. Thus, we have to define Decom in such a way that we minimize the probability that the wrong Z is recovered after decompression while accounting for the distribution over the input: $Z \sim (\text{Bern}(p))^n$.

Given that $\text{im}(H) = \mathbb{F}_2^m$, we need to capture this "lack of bijectivity" of our compression scheme. To do this, we consider the probability that our Decom decompresses $w \in \mathbb{F}_2^m$ to a draw from the input distribution.

$$\mathop{\mathbf{Pr}}_{Z\sim(\text{Bern}(p))^n}[\text{Decom}(HZ) \neq Z] = 1 - \mathop{\mathbf{Pr}}_{Z\sim(\text{Bern}(p))^n}[\text{Decom}(HZ) = Z]$$

where we can write the term on the right hand side as

$$\mathop{\mathbf{Pr}}_{Z\sim(\text{Bern}(p))^n}[\text{Decom}(HZ) = Z] = \sum_{w\in\mathbb{F}_q^m} \mathop{\mathbf{Pr}}_{Z\sim(\text{Bern}(p))^n}[\text{Decom}(w) = Z]$$

Now, let's say there is a set of elements in $\mathbb{F}_2^n$ that map to the same element in $\mathbb{F}_2^m$: $S = \left\{z \in \mathbb{F}_q^n | Hz = w\right\}$. Since we have said that $p < \frac{1}{2}$, the decompression algorithm that picks the z with minimal weight i.e.

$$\text{Decom}(w) := \mathop{\text{argmin}}_{x\,\in\,S}\{\text{wt}(x)\}$$

maximizes $\mathop{\mathbf{Pr}}_{Z\sim(\text{Bern}(p))^n}[\text{Decom}(w) = Z]$. This can be seen if we let $z^* \in S$ be the element with minimal weight

$$\mathop{\mathbf{Pr}}_{z\,\in\,S}[\text{wt}(z) \geq \text{wt}(z^*)] = \sum_{t>\text{wt}(z^*)} \binom{n}{k} \cdot p^k(1-p)^{n-k} \to 0$$

as $p \to 0$. Finally, we can show that a code $C \subseteq \mathbb{F}_q^n$ which is defined as

$$C = \left\{x \in \mathbb{F}_q^n | Hx = 0\right\}$$

has distance that satisfies

$$\delta \geq \frac{t}{\log(1/p)}$$

To see why this inequality is true, consider $z \in \mathbb{F}_q^n$ and $z' \in \mathbb{F}_q^n$ where $Hz = Hz'$ i.e. both z and z' map to the same codeword $x \in C$. Now if

$$\begin{aligned} \dim(C) &= n - \text{rank}(H) \\ &= n - m \end{aligned}$$

and z and z' each define a linear combination of $n - m$ column vectors of H. Such a linear combination may result in the same $x \in C$. In the worst case, such a collision can occur by flipping a single bit which occurs with probability p since $Z \sim (\text{Bern}(p))^n$. The number of ways we can perform such a flip is 2^δ so we have

$$\mathop{\mathbf{Pr}}_{Z \sim (\text{Bern}(p))^n} [\text{Decom}(HZ) \neq Z] = 2^\delta \cdot p \leq 2^{-t}$$

Therefore, $\delta \geq \frac{t}{\log(1/p)}$.

■

Problem 0.3. *Mixing polynomials*

Solution.

We are given two sequences of values b_n and c_n which are the result of evaluating polynomials $f_1(x)$ and $f_2(x)$ at points a_n. However, for some values of n, b_n and c_n are swapped during transmission.

For any particular a_n, the sum $f_1(a_n) + f_2(a_n) = b_n + c_n$ and the product $f_1(a_n) \cdot f_2(a_n) = b_n \cdot c_n$ do not change upon swapping b_n and c_n. Let y be the sequence of values received, then we can write a bi-variate polynomial

$$
\begin{aligned}
h(x,y) &= y^2 - y(f_1(x) + f_2(x)) + f_1(x) \cdot f_2(x) \\
&= (y - f_1(x))(y - f_2(x))
\end{aligned}
$$

If we can perform such a factorization of $h(x,y)$ then we can descramble $f_1(x)$ and $f_2(x)$.

In the second case, we are given a value β_n at each point in the domain a_n but we don't know whether β_n came from $f_1(x)$ or $f_2(x)$. However, we are given the guarantee that the number of points coming from $f_1(x)$ satisfies $\frac{n}{3} \leq n_1 \leq \frac{2n}{3}$ and the points coming from $f_2(x)$ satisfies $\frac{n}{3} \leq n_2 \leq \frac{2n}{3}$ where $n = n_1 + n_2$.

We can recast this problem by thinking of the points that came from one polynomial, say $f_1(x)$, as "errors" and define an error polynomial e that is zero when $y \neq f_1(x)$. Then, we can use the general Reed-Solomon decoding scheme to solve for $f_2(x)$. Once we have $f_2(x)$, finding $f_1(x)$ is straightforward: use Lagrange interpolation again on the difference between $f_2(x)$ and y.

Recall that Lagrange interpolation requires that $k \leq n - t$ where n is the total number of points, t the number of errors, and k is the degree of the polynomial to interpolate. We can still apply Lagrange interpolation here since $k \leq \frac{n}{3}$ because $t = \frac{2n}{3}$ is the maximum number of "errors" which must be true since we are certain that $k < \frac{n}{6}$.

$\blacksquare$