1 Linear Codes

We are beginning to close the gap betwene upper and lower bounds, expressed in terms of normalied rate R and distance δ.
Question: *What is the best availiable rate?*

Proposition 1. *Gilbert's Bound. There exists a code C of distance d in $\{0,1\}^n$ satisfying $|C| \geq \frac{2^n}{Vol(n,d-1)}$. This implies there exists a code with $R(C) \geq 1 - H(\delta) - o(1)$.*

We achieved this with a greedy algorithm and a volumetric bound. How can we improve on this?

Theorem 2. *Varshamov Bound. There exists a code C of distance d in $\{0,1\}^n$ satisfying $|C| \geq \frac{2^n}{Vol(n,d-2)+1}$.*

We begin this proof by developing the structure of linear codes. A linear code can be specified by a partity check matrix. If we fix some $H \in \mathbb{F}^{n \times m}$, we can define the code C generated by H as:

$$C = \{y \in \mathbb{F}^n : y \cdot H = 0\}$$

Claim 3. *For a code generated by H we have $|C| \geq 2^{n-m}$.*

Proof. We know from Problem Set 0 there exists $G \in \mathbb{F}^{n-m \times n}$ such that $GH = 0$ and G has full rank. We then have that the rows of G are linearly independent and all such combinations lie in the kernel of H. □

Since we define the rate as $\frac{|C|}{2^n}$, smaller m relative to n improves our performance.
Question: *How do we know the distance given H?*

Claim 4. *The code checked by H has distance d if and only if no subset of $d-1$ rows are linearly dependent.*

Proof. Since we are over $\mathbb{F}_2$, linear dependence means there at most $d-1$ rows such that their sum is 0. Fixing H, suppose we have linear dependent rows with indices ℓ_i for $i \in [d-1]$. We can then construct the codeword $v = \sum_i^{d-1} e_{\ell_i}$. This is equivalent to creating a codeword with a 1 in each index of a linearly dependent row. But we then have that $v \cdot H = 0$ from the defintion, but $d(v, 0) = d-1$ so the code cannot have distance d.
For the opposite direction, if we have that $x, y \in C$, we have that $(x-y) \cdot H = 0$, which implies that $\Delta(x-y) \geq d$ or else we would have some linearly dependent subset. □

Now that we have this, how do we construct such an H?
Algorithm: We will build H recursively, and at each step avoid inserting new rows where the distance from the existing rows is too small. We can visualize this as walking down a $m \times 2^m - 1$ matrix, where the ith row is i in binary, and we add this row to H if it cannot be written as the sum of $d-2$ previous rows.

1. Initialize $S = \{0,1\}^m - \{0\}$, $H = \emptyset$
2. While $\exists x \in S$, set $H \leftarrow H \cup x$. If we denote $R_x = \{s \in S : \exists h_i, ...d_k \in H, k \leq d-3,$ such that $s = x + \sum_i h_i\}$, set $S \leftarrow S \setminus \{x\} \cup R_x$.

Eventually we will terminate. When this occurs, for every row we will have thrown it out or added to H. For each addition to the parity check matrix, we throw out $\sum_i^{d-3}\binom{n}{i}$ additional vectors by out construction, which is $\text{Vol}(n, d-2)$. Therefore when we terminate:

$$\text{Vol}(n, d-2) \geq 2^m - 1$$

We can then combine this with the bound $|C| \geq \frac{2^n}{2^m}$ which gives:

$$|C| \geq \frac{2^n}{2^m} \geq \frac{2^n}{\text{Vol}(n, d-2) + 1}$$

As desired.

Exercise 5. *Give an algorithm to actually perform this construction and analyze its runtime. What data structure gives the best performance for this operation?*

2 Dual Codes

Suppose we have some code C with parity check matrix H and generator G, so $GH = 0$. We can create the dual code with generator matrix H^T and parity check matrix G^T. What is this code? We have that $(C^\perp)^\perp = C$.
Remark We are currently unaware of any region of the $R > 0, \delta > 0$ parameter space that is achievable, but not by linear codes.

Exercise 6. *(hard). Either prove this is true for all parameters or prove a counterexample.*

3 More Bounds

We return to analyzing the parameter space for R and δ. There is a blank region of space with positive rate to the right of $\delta = 1/2$, where all codes must disagree on more than $1/2$ of coordinates. Can we construct one?

For intutition, suppose we are working over $\mathbb{F}_2$ and want to achieve $\delta = 2/3$. WLOG we can set the first codeword to 0. The second codeword must have $> 2/3$ entires 1. For the third word, we are stuck. It must have $> 2/3$ ones from the first constraint, but then it would overlap on $> 1/3$ of entries with the second. Unfortunately, this argument is ad-hoc, so how do we get more general bounds?

3.1 Moving to $\mathbb{R}$

So far, we have had to deal with the geometry of Hamming distance. Can we pass to an easier space to analyze?

Definition 7. *Define $f : \{0,1\}^n \to \mathbb{R}^n$ by $f(x) = (-1)^x$.*

This maps the binary code into $\{-1,1\}^n$. We can then use our exsting inner product on $\mathbb{R}^n$. For $x, y \in \mathbb{R}^n$ we have:

$$\langle x, y\rangle = \sum_i^n x_i y_i$$

Given this inner product and f, we can see that for x, y where $x = f(a), y = f(b)$ for $a \neq b$:

1. $\langle x, x\rangle = n$
2. $\langle x, y\rangle = n - 2\Delta(x, y)$

So we the inner product and Hamming distance are related. Using this, we can prove a new, stronger bound:

3.2 Plotkin Bound

Theorem 8. *Plotkin Bound. If code C has $\Delta(C) \geq n/2$, then $|C| \leq 2n$*

Remark This is extremely bad. We desire codes of exponential size, this is linear. In addition, this is tight as $n \to \infty$.

To prove this, we state and prove a lemma:

Lemma 9. *If $u_1, ...u_m \in \mathbb{R}^n$ satisfy:*

1. $\langle u_i, u_j \rangle \leq 0$ *for* $i \neq j$
2. $\langle u_i, u_i \rangle = 0$.

Then $m \leq 2n$.

Note that this is tight because we can create $2n$ vectors $\pm e_i$, where it is easy to verify they satisfy the desired properties. Note that when we recast this using f, it is easy to see how this implies the theorem. If all codewords x_i disagree on $\geq 1/2$ of indices, we must have $\langle f(x_i), f(x_j) \rangle \leq 0$ for $i \neq j$. We can simply divide each vector by $\frac{1}{n}$ to ensure norms of 1 instead of n.

Proof. Fix $u_1, ...u_m$ as in the statement. We can choose some v at random. With probability one it has nonzero product with all u_i, so there are at least $m/2$ vectors such that $\langle u_i, v \rangle > 0$. We now resitrct our attention solely to these vectors, which will be denoted v_i.

If $m > n$ (the only parameter regime we need to prove) we know that v_i are linearly dependent. Therefore we can write $\sum_i^k \lambda_i v_i = \sum_{j=k+1}^n \lambda_j v_j$ where all $\lambda \geq 0$. We will now consider the inner product:

$$\langle \sum_i^k \lambda_i v_i, \sum_{j=k+1}^n \lambda_j v_j \rangle$$

- This is nonzero. If we set $w = \sum_i^k \lambda_i v_i$, we can write this as $\langle w, w \rangle > 0$. However, note that we can use bileratity and expand the product as:

  $$\langle w, w \rangle = \sum_{i \neq j} \lambda_i \lambda_j \langle v_i, v_j \rangle = 0$$

  So we have a contradiction.

- This is 0. This implies $\sum_i^k \lambda_i v_i = 0$ yet some $\lambda_i > 0$. We then have:

  $$0 = \langle \sum_i^k \lambda_i v_i, v \rangle = \sum_i^k \lambda_i \langle v_i, v \rangle$$

  But since some $\lambda_i > 0$ this is > 0 by the condition on v, so we again have a contradction.

Therefore we cannot have $\frac{m}{2} \geq n \implies m \leq 2n$, so we are done. □

3.3 Extending to Higher Bases

We can easily adapt this to higher bases by creating new functions $f_q\{0,1\}^n \to \mathbb{R}^{d_q}$. For $q = 3$, we can map into $\mathbb{R}^2$, with the three vectors having equal, negative inner product.

Exercise 10. *Construct such a map, and ues it to prove for a q-ary code of positive rate:*

$$\delta < 1 - \frac{1}{q}$$

We know have a slightly strange picture. There is a barrier for $\delta \geq 1/2$, but there is a concave section with codes with slightly lower distance but *much* higher rate. We can address this with a proofd that we will sketch here.

Lemma 11. *Given a (n, k, d) code, we can create both $(n-1, k-1, d)$ and $(n-1, k, d-1)$ codes.*

Proof.

1. To move to $(n-1, k, d-1)$, we use **puncturing**. Delete the final bit. This decreases the distance between any two codewords by at most 1.
2. To move to $(n-1, k-1, d)$, we use **restruction**. If we look at the final bit of codewords, there is one value that appears at least $\frac{1}{q}$ of the codes. But note that we can then only consider these codes, which already have distance despite agreeing on the final coordinate, and leave the final coordinate implicit.

□

Exercise 12. *Given this, show by a diagonal walk argument that $R + 2\delta \leq 1$.*