Lecturers: Hung Q. Ngo and Atri Rudra

Codes

1 Preliminaries

Let Σ be a finite set, $|\Sigma| \geq 2$. We will refer to elements of Σ as *symbols* or *letters*, and Σ as an *alphabet*. A code C over alphabet Σ is a subset of Σ^n, where the positive integer n is called the *length* (or *block length*) and $|C|$ is the *size* of the code. Each member of C is called a *codeword*. In orther words, a codeword is a vector of dimension n, each of whose coordinates is also called a *position*.

The *Hamming distance* between two codewords $\mathbf{c}$ and $\mathbf{c}'$, denoted by $\Delta(\mathbf{c}, \mathbf{c}')$ is the number of positions where $\mathbf{c}$ and $\mathbf{c}'$ are different. The *minimum distance* of a code C, denoted by $\Delta(C)$, is the minimum Hamming distance between two different codewords of C. The *dimension* of a code C on alphabet Σ is defined to be $\dim(C) := \log_{|\Sigma|} |C|$. A code with length n and dimension k on an alphabet of size q is called an $(n, k)_q$-code. An $(n, k)_q$-code with minimum distance Δ is called an $(n, k, \Delta)_q$-code. Sometimes, to emphasize a specific alphabet in use, we use the notations $(n, k)_\Sigma$ and $(n, k, \Delta)_\Sigma$.

Proposition 1.1 (Singleton Bound [3])**.** *For any $(n, k, \Delta)_q$-code, $k \leq n - \Delta + 1$.*

A code achieving equality in the Singleton bound is called a *Maximum distance separable* code, or MDS code. A very widely used MDS code is the celebrated *Reed-Solomon code*, named after its two inventors Irving Reed and Gustave Solomon[1] [2].

Exercise 1. Prove the Singleton bound. (Hint: consider any code C of minimum distance Δ and length n. Let C' be the projection of C on to the first $n - (\Delta - 1)$ coordinates. Show that $|C'| = |C|$ and bound $|C'|$.)

It is often the case[2] that the alphabet Σ is a finite field $\mathbb{F}_q$, because then we are able to take advantage of the underlying (linear) algebraic structures for designing the codes, analyzing its parameters, and discovering good encoding and decoding algorithms. In this case, when C is a linear subspace of $_q^n$, we call C a *linear code*. To emphasize the fact that C is linear, we replace $(n, k)_q$ and $(n, k, \Delta)_q$ by $[n, k]_q$ and $[n, k, \Delta]_q$. Note that the dimension k of the code is now precisely the dimension of the subspace C.

2 Reed-Solomon Codes

Definition 2.1 (Reed-Solomon code)**.** Let $k \leq n \leq q$ be positive integers where q is a prime power. The *Reed-Solomon code* is an $[n, k, n - k + 1]_q$-code (i.e. a linear MDS code) defined as follows. Let

[1] This paper and the likes of Shannons' and Hamming's papers are perfect examples illustrating that we dont have to write humongously long papers to be influential.

[2] See, `http://www.cs.cmu.edu/~venkatg/teaching/codingtheory/notes/algebra-brief-notes.pdf` for a brief introduction to finite fields

$\{\alpha_1, \ldots, \alpha_n\}$ be any n disdinct members of $\mathbb{F}_q$. These are called the *evaluation points* of the code. For each vector $\mathbf{m} = (m_0, \ldots, m_{k-1}) \in \mathbb{F}_q^k$, define a polynomial

$$f_{\mathbf{m}}(x) = \sum_{i=0}^{k-1} m_i x^i$$

which is of degree at most $k-1$. Then, for each $\mathbf{m} \in \mathbb{F}_q^k$ there is a corresponding codeword $\mathrm{RS}(\mathbf{m})$ defined by

$$\mathrm{RS}(\mathbf{m}) = \langle f_{\mathbf{m}}(\alpha_1), \ldots, f_{\mathbf{m}}(\alpha_n) \rangle.$$

Exercise 2. Prove the following.

1. If $\mathbf{m} \neq \mathbf{m}'$, then $\mathrm{RS}(\mathbf{m}) \neq \mathrm{RS}(\mathbf{m}')$. Thus, the RS code defined above has precisely q^k codewords.
2. For any $\mathbf{m}, \mathbf{m}' \in \mathbb{F}_q^k$ and any scalar $a \in \mathbb{F}_q$,

$$\begin{aligned} \mathrm{RS}(\mathbf{m} + \mathbf{m}') &= \mathrm{RS}(\mathbf{m}) + \mathrm{RS}(\mathbf{m}') \\ \mathrm{RS}(a\mathbf{m}) &= a \cdot \mathrm{RS}(\mathbf{m}). \end{aligned}$$

   Thus, the RS code is a linear code. Along with part 1) this means the linear code is of dimension k.
3. Use the fact that any polynomial of degree at most $k-1$ over $\mathbb{F}_q$ has at most $k-1$ roots to show that, for any $\mathbf{m} \neq \mathbf{m}'$ the Hamming distance between $\mathrm{RS}(\mathbf{m})$ and $\mathrm{RS}(\mathbf{m}')$ is at least $n-k+1$.
4. Lastly, consider the distance between the all-zero codeword and the codeword corresponding to the polynomial $\prod_{i=1}^{k-1}(x-\alpha_i)$, prove that the above RS code is an $[n, k, n-k+1]_q$-code.

Note also that,

$$\mathrm{RS}(\mathbf{m}) = (m_0, \ldots, m_{k-1}) \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \alpha_1 & \alpha_2 & \ldots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \ldots & \alpha_n^2 \\ \vdots & \vdots & \ldots & \vdots \\ \alpha_1^{k-1} & \alpha_2^{k-1} & \ldots & \alpha_n^{k-1} \end{pmatrix}$$

This means the symbol at position i of the $\mathbf{m}$th codeword $\mathrm{RS}(\mathbf{m})$ can be computed in time $\mathrm{poly}(q)$. The above matrix is called the $k \times n$ *Vandermonde matrix* which occurs in many contexts in Mathematics and Computer Science.

3 Code concatenation

Let q, n, m, N be integers such that $N \leq q^n$ and $2^m \geq q$. Let C_{out} be a code of length n and size N over an alphabet Σ of size q. Without loss of generality (up to isomorphism) we might as well set $\Sigma = [q]$. Let C_{in} be a binary code (i.e. alphabet $\Sigma = \{0, 1\}$) of length m and size q. A *concatenation* C of C_{out} and C_{in}, denoted by $C = C_{\mathrm{out}} \circ C_{\mathrm{in}}$, is a code C of length mn and size N constructed by replacing each symbol a of a codeword in C_{out} by the ath codeword in C_{in}. Here, we order the codewords in C_{in} in an arbitrary manner. For example, consider the case when $n = q = 3, m = 2$,

$$C_{\mathrm{out}} = \left\{ \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}, \begin{bmatrix} 2 \\ 1 \\ 3 \end{bmatrix}, \begin{bmatrix} 3 \\ 2 \\ 2 \end{bmatrix}, \begin{bmatrix} 3 \\ 2 \\ 3 \end{bmatrix} \right\}, C_{\mathrm{in}} = \left\{ \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right\}.$$

Then,

$$C_{\text{out}} \circ C_{\text{in}} = \left\{ \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \right\}.$$

Abusing notation, we often also state that a code is a matrix which is constructed by putting all codewords of the code as columns of the matrix in an arbitrary order. For example, the matrix $\mathbf{M} = C_{\text{out}} \circ C_{\text{in}}$ above is

$$\mathbf{M} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

In the concatenation $C_{\text{out}} \circ C_{\text{in}}$, C_{out} is called the *outer code* and C_{in} the *inner code*. By instantiating the outer and inner codes with carefully chosen codes, we obtain good group testing matrices. One of the most basic inner codes is the trivial *identity code*, ID_q, which is the binary code of length q and size q whose ith codeword is the ith standard basis vector. The corresponding matrix is the identity matrix of order q.

4 Gilbert-Varshamov Bound

Let $A_q(n, \Delta)$ denote the maximum size of a q-ary code of length n and minimum distance Δ. Determining $A_q(n, \Delta)$ is a major open problem in coding theory. Define

$$\text{Vol}_q(n, \ell) = \sum_{j=0}^{\ell} \binom{n}{j} (q-1)^j$$

to be the "volume" of the Hamming ball of radius ℓ around any codeword, i.e. the number of vectors of distance at most ℓ from a given vector in $\mathbb{F}_q^n$. Gilbert [1] and Varshamov [4] proposed a simple greedy algorithm which constructs a linear code with size at least $q^n/\text{Vol}_q(n, \Delta - 1)$. Actually, Gilbert's algorithm does not produce a linear code; Varshamov's does. However, their algorithms are very similar and achieves similar bounds.

Theorem 4.1 (Gilbert-Varshamov Bound)**.** *The maximum size of a code of length n, alphabet size q, and distance Δ satisfies*

$$A_q(n, \Delta) \geq \frac{q^n}{\text{Vol}_q(n, \Delta - 1)} = \frac{q^n}{\sum_{j=1}^{\Delta-1} \binom{n}{j}(q-1)^j}.$$

There also exists linear codes achieving the bound.

Exercise 3 (Gilbert algorithm)**.** Consider the following algorithm for code construction. Let Σ be an alphabet of size q. Initially let $C = \emptyset$. While there still exists a vector $\mathbf{c} \in \Sigma^n$ which is of distance at least Δ from all the codewords in C, add $\mathbf{c}$ into C. Prove that when the algorithm stops, we obtain a code C whose size is at least $q^n/\text{Vol}_q(n, \Delta - 1)$.

To show that there exist *linear* codes attaining the Gilbert-Varshamov (GV) bound, we use the probabilistic method. In fact, we will prove the asymptotic form of the GV bound for linear codes.

Definition 4.2 (q-ary entropy)**.** Let $q \geq 2$ be an integer. The *q-ary entropy function* $H_q : [0,1] \to \mathbb{R}$ is defined by

$$H_q(\delta) = \delta \log_q \frac{q-1}{\delta} + (1-\delta) \log_q \frac{1}{1-\delta}. \tag{1}$$

When $q = 2$ we drop the subscript q and write the famous (Shannon) *binary entropy function* as

$$H(\delta) = \delta \log \frac{1}{\delta} + (1-\delta) \log \frac{1}{1-\delta}.$$

Occassionally, it might be easier to grasp the q-ary entropy funciton by rewriting (1) as

$$H_q(\delta) = \delta \log_q(q-1) - \delta \log_q \delta - (1-\delta) \log_q(1-\delta).$$

We define $H_q(0) = 0$. The function $H_q(x)$ is continuous in the interval $[0,1]$, is increasing from 0 to $1 - 1/q$, and decreasing from $1 - 1/q$ to 1.

Lemma 4.3. *For any positive integers $n, q \geq 2$ and real number $0 \leq \delta \leq 1 - 1/q$,*

$$q^{n(H_q(\delta) - o(1))} \leq \mathrm{Vol}_q(n, \delta n) \leq q^{nH_q(\delta)}.$$

Proof. We prove the upper-bound first, using the famous Bernstein trick. Without loss of generality, we estimate the volume $\mathrm{Vol}_q(n, \delta n)$ around the all-zero codeword. We can pick uniformly a random word $\mathbf{w} = (w_1, \ldots, w_n) \in \Sigma^n$ by picking each coordinates w_i uniformly and independently from Σ. Let $X_1, \ldots, X_n$ be independent Bernoulli variables with parameter $1 - 1/q$. Let $\mathrm{wt}(\mathbf{x})$ denote the weight of vector $\mathbf{x}$. Then, it is not hard to see that

$$\mathrm{Vol}_q(n, \delta n)/q^n = \mathrm{Prob}[\mathrm{wt}(\mathbf{w}) \leq \delta n] = \mathrm{Prob}[n - \mathrm{wt}(\mathbf{w}) \geq (1-\delta)n] = \mathrm{Prob}\left[\sum_{i=1}^{n} X_i \geq (1-\delta)n\right].$$

Now, let $t \geq 0$ be an arbitrary real number. We have

$$\begin{aligned}
\mathrm{Prob}\left[\sum_{i=1}^{n} X_i \geq (1-\delta)n\right] &\leq \mathrm{Prob}\left[t\sum_{i=1}^{n} X_i \geq t(1-\delta)n\right] \\
&= \mathrm{Prob}\left[e^{t\sum_{i=1}^{n} X_i} \geq e^{t(1-\delta)n}\right] \\
&\leq \frac{\mathrm{E}\left[e^{t\sum_{i=1}^{n} X_i}\right]}{e^{t(1-\delta)n}} \\
&= \frac{\prod_{i=1}^{n} \mathrm{E}\left[e^{tX_i}\right]}{e^{t(1-\delta)n}} \\
&= \frac{\prod_{i=1}^{n} \left((1-1/q)e^t + 1/q\right)}{e^{t(1-\delta)n}} \\
&= \left((1-1/q)e^{\delta t} + (1/q)e^{(1-\delta)t}\right)^n.
\end{aligned}$$

To minimize the right hand side, we can pick $t = \ln \frac{(q-1)(1-\delta)}{\delta}$, which is non-negative because $\delta \leq 1 - 1/q$. Plugging this value of t back into the inequality we conclude that

$$\frac{\mathrm{Vol}_q(n, \delta n)}{q^n} = \text{Prob}\left[\sum_{i=1}^{n} X_i \geq (1-\delta)n\right] \leq \left(\frac{(1-1/q)/\delta}{((q-1)(1-\delta)/\delta)^{1-\delta}}\right)^n = \frac{q^{nH_q(\delta)}}{q^n}.$$

The upper-bound is thus proved. To prove the lower-bound, observe that $\mathrm{Vol}_q(n, \delta n) \geq \binom{n}{\lfloor \delta n \rfloor}(q-1)^{\lfloor \delta n \rfloor}$. For notational simplicity, define $m = \lfloor \delta n \rfloor$ and $p = m/n$. It is not hard to see that the function $f(k) = \binom{n}{k} p^k (1-p)^{n-k}$ is increasing when $1 \leq k \leq m$ and decreasing when $m \leq k \leq n$. Since $\sum_{k=0}^{n} f(k) = 1$, the largest term $f(m)$ is at least $1/(n+1)$ because the sum has $n+1$ terms. Consequently,

$$\mathrm{Vol}_q(n, \delta n) \geq \binom{n}{m}(q-1)^m \geq \frac{1}{n+1}\left(\frac{n}{m}\right)^m \left(\frac{n}{n-m}\right)^{n-m}(q-1)^m.$$

Note that

$$\frac{n-m}{m} = \frac{n}{m} - 1 \geq \frac{1}{\delta} - 1 = \frac{1-\delta}{\delta},$$

and that for sufficiently large n

$$\left(\frac{n-\delta n}{n-m}\right)^n \geq \left(\frac{n-m-1}{n-m}\right)^n = (1 - 1/(n-m))^n \geq (1 - 1/(n-\delta n))^n \geq 1/3^{1-\delta}.$$

Hence, for large n we have

$$\begin{aligned}
& \left(\frac{n}{m}\right)^m \left(\frac{n}{n-m}\right)^{n-m}(q-1)^m \\
= \; & \left(\frac{n}{\delta n}\right)^{\delta n} \left(\frac{n}{n-\delta n}\right)^{n-\delta n}(q-1)^{\delta n} \frac{1}{(q-1)^{\delta n - m}} \left(\frac{n-\delta n}{n-m}\right)^n \left(\frac{n-m}{m}\right)^m \left(\frac{\delta n}{n-\delta n}\right)^{\delta n} \\
\geq \; & \underbrace{\left(\frac{n}{\delta n}\right)^{\delta n} \left(\frac{n}{n-\delta n}\right)^{n-\delta n}(q-1)^m}_{q^{nH_q(\delta)}} \underbrace{\frac{1}{q-1}(1/3)^{1-\delta}\left(\frac{\delta}{1-\delta}\right)^{\delta n - m}}_{q^{-no(1)}}
\end{aligned}$$

□

A central problem in coding theory is to characterize the tradeoff between the distance and the rate of a code. The *relative distance* $\delta(C)$ of a code C of length n is $\Delta(C)/n$. If C has dimension k then its *rate* is defined to be $R(C) = k/n$.

Theorem 4.4 (Asymptotic form of the GV bound)**.** *Let $q \geq 2$ be an integer. For any $0 \leq \delta \leq 1 - 1/q$, there exists an infinite family of q-ary codes with rate $R \geq 1 - H_q(\delta) - o(1)$. In fact, such code exists for all sufficiently large length n.*

We will prove the linear code version of the above bound.

Exercise 4. Show that for a linear code the minimum distance is equal to the minimum weight of a non-zero codeword. (The *weight* of a codeword is the number of non-zero entries in it.)

Exercise 5. For positive integers $k < n$, let $\mathbf{G}$ be a random $k \times n$ matrix chosen by picking each of its entries from $\mathbb{F}_q$ uniformly and independently. Fix a vector $\mathbf{y} \in \mathbb{F}_q^k$. Prove that the vector $\mathbf{yG}$ is a uniformly random vector in $\mathbb{F}_q^n$.

Theorem 4.5 (Linear code version of the asymptotic form of the GV bound)**.** *Let $q \geq 2$ be any prime power. Let $0 \leq \delta \leq 1 - 1/q$, Let $n \geq 2$ be any integer. Then, for any integer $k \leq (1 - H_q(\delta))n$ there exists an $[n, k, \delta n]_q$-code.*

Proof. We want a k-dimensional linear subspace C of $\mathbb{F}_q^n$ where the minimum weight of non-zero codewords is at least $\Delta = \delta n$. The subspace can be generated by a $n \times k$ matrix $\mathbf{G}$ of rank k, called the generator matrix for the code. The columns of $\mathbf{G}$ form a basis for the subspace. We pick a random generator matrix $\mathbf{G}$ and show that it satisfies two properties with positive probability:

(a) $\mathbf{G}$ has full column rank, and

(b) for every non-zero vector $\mathbf{y} \in \mathbb{F}_q^k$, $\mathbf{Gy}$ has weight at least Δ.

Let $\mathrm{wt}(\mathbf{x})$ denote the weight of vector $\mathbf{x}$. Actually, property (b) implies property (a) because if the columns of $\mathbf{G}$ are linearly dependent then there is some non-zero vector $\mathbf{y}$ for which $\mathbf{Gy} = \mathbf{0}$. To pick the random matrix $\mathbf{G}$, we simply pick each of its entry from $\mathbb{F}_q$ uniformly and independently. For any fixed non-zero vector $\mathbf{y} \in \mathbb{F}_q^k$, $\mathbf{Gy}$ is a uniformly random vector in $\mathbb{F}_q^n$. Hence, by Lemma 4.3

$$\text{Prob}[\mathrm{wt}(\mathbf{Gy}) \leq \delta n] = \frac{\text{Vol}_q(n, \delta n)}{q^n} \leq q^{(H_q(\delta)-1)n}.$$

Now, taking a union bound over all non-zero vectors $\mathbf{y} \in \mathbb{F}_q^k$, the probability that $\mathrm{wt}(\mathbf{Gy}) \leq \delta n$ for some $\mathbf{y}$ is at most

$$(q^k - 1)q^{(H_q(\delta)-1)n} < q^{(1-H_q(\delta))n} q^{(H_q(\delta)-1)n} = 1.$$

□

In a later lecture, we shall show how to derandomize the above algorithm.

References

[1] E. N. GILBERT, *A comparison of signalling alphabets*, 31 (1952), pp. 504–522.

[2] I. S. REED AND G. SOLOMON, *Polynomial codes over certain finite fields*, J. Soc. Indust. Appl. Math., 8 (1960), pp. 300–304.

[3] R. C. SINGLETON, *Maximum distance q-nary codes*, IEEE Trans. Information Theory, IT-10 (1964), pp. 116–118.

[4] R. R. VARSHAMOV, *Estimate of the number of signals in error correcting codes*, Dokl. Akad. Nauk. SSSR, (1957).