COMPSCI 690T Coding Theory and Applications Feb 1, 2017

Lecture 4

Instructor: Arya Mazumdar *Scribe: Names Redacted*

1 Sphere Packing Bound

In the previous lecture we have seen what the singleton bound is. In this lecture we will start with the Sphere Packing bound which is tighter than the singleton bound and as we will see that even though the Hamming codes do not satisfy the singleton bound, they satisfy the Sphere packing bound with equality proving that they are actually optimal. Now, we can easily see that the number of strings of length n is 2^n. Now we will consider the codewords which are at least a distance of d from each other. The volume of the set of codewords is denoted by $\mathcal{A}(n,d)$. Now, if we consider a ball around a particular codeword x which is defined by

$$S_{n,\lfloor\frac{d-1}{2}\rfloor},= \{y : d(x,y) \leq \lfloor\tfrac{d-1}{2}\rfloor\}$$

each of these balls around the codewords is supposed to be disjoint (otherwise there must exist a string of length n which will be at a distance of $\lfloor\frac{d-1}{2}\rfloor$ from 2 codewords at the same time and hence by triangle inequality the distance between these 2 codewords will be less than d which will be a contradiction). Hence the sum of the volumes of these spheres must be less than the volume of the entire space. Now the volume of $S_{n,t} = \sum_{i=0}^{t}\binom{n}{i}$ Hence we must have

$$\begin{aligned}\mathcal{A}(n,d)|S_{n,\lfloor\frac{d-1}{2}\rfloor}| &\leq 2^n \\ \mathcal{A}(n,d) &\leq \frac{2^n}{\sum_{i=0}^{\lfloor\frac{d-1}{2}\rfloor}\binom{n}{i}}\end{aligned}$$

Now let us show that the Hamming code satisfies the sphere packing bound with equality. As we have already seen, we can write the family of Hamming codes $[n,k,d]$ in the form $[2^m-1, 2^m-m-1, 3]$ for string length m. From the above bound we have

$$\begin{aligned}\mathcal{A}(n,3) &\leq \frac{2^n}{n+1} \\ &= \frac{2^{2^m-1}}{2^m-1+1} \\ &= 2^{2^m-1-m}\end{aligned}$$

But since $k = 2^m-1-m$ which is the dimension of the code, hence the actual volume must be 2^{2^m-m+1} which is the expression we obtained from the above bound. Hence the Hamming codes are optimal.

2 Bounds on codes

Definition *An asymptotically good code is a code family whose minimum distance increases linearly with the blocklength and it has asymptotically a positive rate.*

Define,

$$R(\delta) = \lim_{n\to\infty}\log_2\frac{\mathcal{A}(n,\delta n)}{n}$$

with $0 \leq \delta \leq 1$.

We can easily observe that the Hamming codes are NOT asymptotically good since they have a relative distance δ going to 0 whereas the rate R goes to 1. This fact can be easily verified from the following 2 identities:-

$$\lim_{m\to\infty} \frac{3}{2^m - 1} = 0$$
$$\lim_{m\to\infty} \frac{2^m - 1 - m}{2^m - 1} = 1$$

Now let us understand what the Singleton bound and the Sphere Packing bound mean for the asymptotically good codes.

2.1 Singleton Bound

From the singleton bound we have

$$A(n, d) \leq 2^{n-d+1}$$

Hence we have that

$$\begin{aligned} R(\delta) &\leq \lim_{n\to\infty} \frac{n - \delta n + 1}{n} \\ R(\delta) &\leq 1 - \delta \end{aligned}$$

2.2 Sphere-packing Bound

From the sphere packing bound we have

$$\begin{aligned} R(\delta) &\leq \frac{n - \log \sum_{i=1}^{\lfloor \frac{d-1}{2} \rfloor} \binom{n}{i}}{n} \\ &\leq 1 - \frac{\log \sum_{i=1}^{\lfloor \frac{d-1}{2} \rfloor} \binom{n}{i}}{n} \\ &\leq 1 - \frac{\log \binom{n}{\lfloor \frac{d-1}{2} \rfloor}}{n} \end{aligned}$$

The last step is because we have $\binom{n}{t} \leq \sum_{i=0}^{t} \binom{n}{t} \leq (t+1)\binom{n}{t}$ and $\lim_{t\to\infty} \frac{\log t + 1}{t} = 0$. Now we can use Stirling's formula ($n! \sim (\frac{n}{e})^n$) to simplify the above expression. Using Stirling's expression we can simplify and write $\binom{n}{t} = \frac{n^n}{t^t (n-t)^{n-t}}$. Now substituting $t = \tau n$ and taking log on both sides we will have

$$\begin{aligned} \log \binom{n}{\tau n} &= n \log n - \tau n \log \tau n - (n - \tau n) log(n - \tau n) \\ &= n \log n - \tau n \log n - \tau n \log \tau - (n - \tau n) \log(1 - \tau) - (n - \tau n) \log n \end{aligned}$$

Hence we must have that

$$\frac{1}{n} \log \binom{n}{\tau n} = -\tau \log \tau - (1 - \tau) \log(1 - \tau) = h(\tau)$$

where h is the binary entropy function. Substituting the above result by using $d = \delta n$ and hence $\tau = \frac{\delta}{2}$ in the above calculations for rate, we have that

$$R(\delta) \leq 1 - h(\tfrac{\delta}{2})$$

2.3 Gilbert-Varshamov (GV) Bound

We will end this section by showing an achievability proof due to Gilbert known as the Gilbert-Varshamov construction. Gilbert constructed an asymptotically good code by a greedy approach in which a took a random string and discarded all the neighbor strings which are at a distance of $d-1$ or less from the selected string. From the remaining strings again select one randomly and discard the neighboring strings in the ball of radius $d-1$. Continue this process until we can no longer select any string. So the entire set of strings selected will form the codewords whose minimum distances is guranteed to be d by the construction process. For every codeword the volume of strings discarded is $\sum_{i=0}^{d-1} \binom{n}{i}$. Hence at most this many number of new strings are discarded for every codeword. Since the volume of the entire space is 2^n, hence the process to terminate, the volume of the codewords can be lower bounded by the following

$$\mathcal{A}(n,d) \geq \frac{2^n}{\sum_{i=0}^{d-1} \binom{n}{i}}.$$

This is called the Gilbert Varshamov bound. Using Stirling's approximation like before we have that

$$R(\delta) \geq 1 - h(\delta)$$

. However we must note that the construction process is computationally infeasible since we have to search in a space exponential in the length of the string in every iteration . Also we should note there does not exist any known result about the hardness of construction of an asymptotically good code and the fact that there does not exist any explicit construction of a code so far that is able to satisfy the GV bound.

3 Plotkin bound

Gilbert Varshamov bound gives us a lower bound on the rate when $\delta \leq 1/2$. Although it is widely believed that Gilbert-Varshamov bound is tight, we do not know of any efficient (sub-exponential time) construction of a code which satisfies it. The Sphere Packing bound gives us an upper bound when $\delta < 1$. Thus, there is a gap between the Gilbert-Varshamov bound and the sphere packing bound for every δ for which the bounds are defined. The Plotkin bound makes the sphere packing bound tighter for $\delta = 0.5$ and matches with the GV bound at that point.

We derive it by first proving that

$$A(n,d) \leq \frac{2d}{2d-n} \text{ for all } d > n/2$$

Consider a set of code words $C = \{c_1, c_2, \ldots, c_m\}$ where $m = A(n,d)$ with the minimum distance d. We arrange these code words in a matrix like fashion - with m rows and n columns. Let $\lambda_i =$ number of 1's in ith *column*.
Consider sum of all pairwise distances between two code words in C.

$$\sum_{c_1,c_2,c_1 \neq c_2} d(c_1,c_2) = \sum_{i=1}^{n} \lambda_i(m-\lambda_i)$$

It is easy to see that the expression on right hand side is also equal to the sum of all pairwise distances between two code words. The idea is to sum up the contribution of each column in the pairwise distance sum. For an i^{th} column, we know that there are λ_i 1's and $m-\lambda_i$ 0's. So, each column will contribute to a distance of $\lambda_i(m-\lambda_i)$.
As the minimum distance is d and there are $\binom{m}{2}$ pairs which contribute to pairwise distance sum,

$$\binom{m}{2} d \quad \leq \quad \sum_{i=1}^{n} \lambda_i(m-\lambda_i)$$

Now As $\lambda(m-\lambda)$ is maximized at $\lambda = \frac{m}{2}$ hence

$$\begin{aligned}
\binom{m}{2} d &\leq \sum_{i=1}^{n} \frac{m}{2}(m - \frac{m}{2}) \\
&= \frac{m^2 n}{4} \\
d &\leq \frac{mn}{2(m-1)} \\
m &\leq \frac{2d}{2d-n}
\end{aligned}$$

As $m > 0$, we have $d > n/2$. Therefore, $A(n,d) \leq \frac{2d}{2d-n}$ when $d > n/2$.

$$R(\delta) = \lim_{n\to\infty} \log_2 \frac{\mathcal{A}(n, \delta n)}{n}$$
$$R(\delta) = \lim_{n\to\infty} \log_2 \frac{2\delta n}{n(2\delta n - n)}$$
$$R(\delta) = \lim_{n\to\infty} \log_2 \frac{2\delta}{n(2\delta - 1)}$$
$$R(\delta) = 0$$

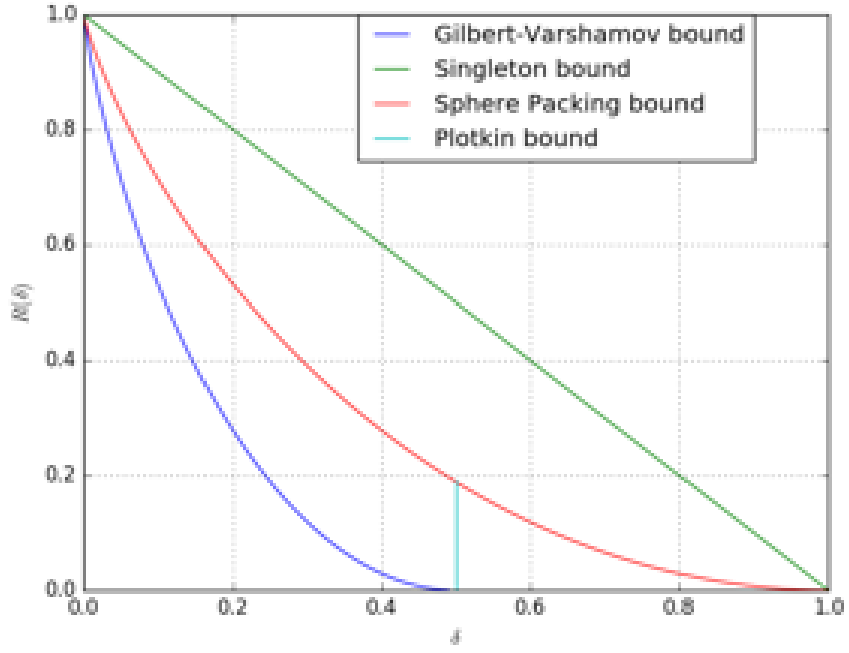

Figure 1: coding theory bounds

The above graph is plot of all the bounds that we have done so far. As we can easily observe there is a large gap at all values of δ except the endpoints between the upper and the lower bounds which remains unresolved so far. After using the plotkin bound the sphere packing bound makes a sudden drop to 0 which creates a sharp point. Hence in the next section we will study the Johnson bound and the Elias-Basslygo bound which gives a smooth bound that matches the lower bound at the endpoints

4 Johnson bound

We consider another type of bound called Johnson Bound for codes with constant weight. Recall, that weight of a code word is the number of 1's in the code word.

$$\forall x \in C, wt(x) = w \text{ (a fixed constant)}$$

Let $A(n, d, w)$ denote the maximum possible size of a code with parameters n, d and w. We apply similar arguments as that of Plotkin bound to derive Johnson bound. Let C be a set of code words with minimum distance d such that $|C| = m$ and $\forall x \in C : wt(x) = w$. Let $\lambda_i =$ number of 1s in i^{th} column . Hence we can easily observe that Total number of 1's in a matrix when counting row-wise or column-wise is the same and hence $\sum_{i=1}^{n} \lambda_i = mw$

$$\begin{aligned} \binom{m}{2} d &\leq \sum_{i=1}^{n} \lambda_i (m - \lambda_i) \\ &= m \sum_{i=1}^{n} \lambda_i - \sum_{i=1}^{n} \lambda_i^2 \end{aligned}$$

Now we know from Cauchy Schwartz inequality or the weighted AM inequality that

$$\sum_{i=1}^{n} \lambda_i^2 \geq \frac{(\sum_{i=1}^{n} \lambda_i)^2}{n}$$

Hence we must have

$$\begin{aligned} \binom{m}{2} d &\leq m(mw) - \frac{(mw)^2}{n} \\ \frac{m(m-1)d}{2} &\leq m^2 w - \frac{m^2 w^2}{n} \\ \frac{m-1}{m} &\leq \frac{2w}{d} - \frac{2w^2}{nd} \\ m &\leq \frac{1}{1 - \frac{2w}{d}(1 - \frac{w}{n})} \end{aligned}$$

Therefore the Johnson Bound is given by the following expression :

$$A(n, d, w) \leq \frac{1}{1 - \frac{2w}{d}(1 - \frac{w}{n})} = \frac{nd}{nd - 2wn + 2w^2}.$$