COMPSCI 690T Coding Theory and Applications	Mar 22, 2017
Lecture 14	
Instructor: Arya Mazumdar	*Scribe: Names Redacted*

1 Group testing

The goal is to identify t defective elements among n of them by doing m tests. Each test reveals if there is a defective element in a set of elements. Here we consider the case where $t << n$.

1.1 Adaptive

The adaptive case is not interesting, since the problem is solved, i.e. the lower bound and upper bound are tight. The lower bound can be obtained by considering how many bits are needed to describe the selection. There are $\binom{n}{t}$ possible answers and each test reveals only one bit of information. Let m be the number of test needed, we have

$$m \geq \log_2 \binom{n}{t} \geq t \log_2 \frac{n}{t}$$

For the upper bound, the naive algorithm which does binary search for t times requires $m = O(t \log n)$. A more sophisticated method can achieve $O(t \log \frac{n}{t})$. But it does not matter too much for small t.

1.2 Non-adaptive

For the non-adaptive case, the testing scheme can be represented by a $m \times n$ binary matrix A, where $A_{i,j}$ is the indicator of whether element j is included in the test i. Let x be a length n vector which indicates whether an element is defective (1) or not (0). The test result, which is a length m binary vector can be computed by thresholding $A \wedge x$. It is the same as computing the Boolean OR of the columns in A corresponding to the support of x.

Definition 1 (t-separable) *A $m \times n$ binary matrix A is t-separable if the union of up to t columns of A are all distinct.*

Proposition 2 *If the testing matrix A is t-separable, then the tests can identify up to t defective element.*

This follows directly from the definition of t-separable.

Although a t-separable testing matrix is enough for identifying t defects, a t-disjunct matrix allows more efficient algorithm for identification.

Definition 3 (t-disjunct) *A $m \times n$ binary matrix A is t-disjunct if the union of any t columns does not contain other columns.*

Proposition 4 *t-disjunct $\Rightarrow$ t-separable.*

Proof (By Contradiction) Suppose a matrix is *t-disjunct* but not *t-separable*, there exist $S_1, S_2 \in [n]$, such that $|S_1|, |S_2| \leq t$, $S_1 \cap S_2 = \emptyset$, and the union ("OR" sum) of columns in S_1 equals the union of columns in S_2. Then every column in S_2 is contained in the union of columns in S_1. So it conflicts with the *t-disjunct* assumption. ■

Proposition 5 *t-separable $\Rightarrow$ $(t-1)$-disjunct.*

Proof (By Contradiction) Suppose a matrix is *t-separable* but not *(t-1)-disjunct*, there exist a set $S \in [n]$, $|S| \leq t-1$, and another column $i \notin S$, so that i is contained in the union of columns if S. If we set $S_2 = S_1 \cup \{i\}$, the union of columns in S_1 and that in S_2 are equal, conflicting with the *t-separable* assumption. ■

Algorithm 1 Decoding Algorithm

Input: A, y

Output: The non-zero positions of x

```
Initialize D = [n]
for i from 1 to m do
    if y_i == 0 then
        D = D\supp(A[i])
return D
```

1.3 Efficient Decoding Algorithm

Suppose $y = A \wedge x$ is the test result, we want to recover x based on y. The decoding algorithm is illustrated in **Algorithm 1**. We use $\mathrm{supp}(\cdot)$ to denote support, the set of indices of all non-zeros entries of a vector.

Correctness: The decoding algorithm is guaranteed to recover the original x on if there are at most t 1s in x and the matrix A is *t-disjunct*.

Proof Since the algorithm only removes elements from D when $y_i = 0$, it will not leave out any defective element. It suffices to show that all the normal elements are removed. Fix a set of t columns (defects), and consider another column i. There exist a row where i is the only "one" element among these $t+1$ columns, since otherwise the union of those t columns will contain i. This is a test with i and without any defective elements. So as long as the $\mathrm{supp}(x) \leq t$, every non-defective element will be removed. ■

Now the only problem is to construct a *t-disjunct* matrix of size $m \times n$. The lower bound of m is $\Omega((t^2/\log t)\log n)$. In the following section, we will give an $m = O(t^2 \log n)$ construction method.

1.4 Random Construction of Testing Matrix A

In this part, we will construct a random testing matrix A and show that with high probability it is *t-disjunct*.

Specifically, we generate a matrix of size $m \times n$, of which each entry is 1 with probability p and 0 with probability $(1-p)$. To make the matrix *t-disjunct*, for any $t+1$ columns and any column within, there should be a row that has 1 in that column and 0 in others. Now will we compute the probability that such condition is not satisfied.

Fix $t+1$ columns and a column within, for one row the above condition does not hold with probability $1 - p(1-p)^t$. The probability that for all the rows such condition does not hold is $(1-p(1-p)^t)^m$. Consider that there are $(t+1)$ columns within, and there are $\binom{n}{t+1}$ choices for such $(t+1)$ columns, by union bound the probability that the matrix is not *t-disjunct* is bounded as

$$P_e \leq \binom{n}{t+1}(t+1)(1-p(1-p)^t)^m$$

We want to minimize the upper bound. The maximum value of $p(1-p)^t$ is $t^t/(t+1)^{t+1}$, achieved at $p = 1/t+1$. So we choose $p = 1/(t+1)$. Also with the inequality $\binom{n}{t+1} \leq (ne)^{t+1}/(t+1)^t$, the probability can be bounded by

$$P_e \leq \frac{(ne)^{t+1}}{(t+1)^t}(1 - \frac{t^t}{(t+1)^{t+1}})^m$$

Consider that $1/e < (t/(t+1))^t < 1/2$, we can further derived

$$P_e \leq \frac{(ne)^{t+1}}{(t+1)^t}(1-\frac{1}{(t+1)e})^m \leq \frac{(ne)^{t+1}}{(t+1)^t}\exp(-\frac{m}{e(t+1)})$$

The last inequality is due to the fact that $\forall x \in (-1,0) \log(1+x) < x$. Suppose the right part of the inequality is smaller than α, where α is a small probability. Then we have

$$-\frac{1}{e(t+1)}m < \log\alpha - \log\frac{(ne)^{t+1}}{(t+1)^t}$$
$$\Rightarrow m > e(t+1)\left((t+1)\log\frac{ne}{t+1} + \log(t+1) - \log\alpha\right)$$

So if we set $m = \Omega(t^2 \log \frac{n}{t})$, w.h.p the *t-disjunct* condition will be satisfied.

1.5 Explicit construction

An explicit construction of t-disjunct matrix of $m = O(t^2\log^2(n))$ exists by using RS code.

Consider a matrix whose columns are codewords of a code $\mathcal{C}$ with constant weight w and minimum distance d. Let c_i, c_j be two columns (codewords) and λ be the number of overlapping weights. We have

$$\lambda \leq w - \frac{d}{2},$$

since otherwise the distance between the two codewords, i.e. the non overlapping bits, will be $w-\lambda+w-\lambda<d$.

Now we claim that if $t\lambda < w$, then the matrix is t-disjunct. This is because, for any t columns set S, the overlap between a column in S and another column $i \notin S$ is no greater than λ. So the overlap between S and i cannot be greater than w, which means i cannot be contained in the union of S.

Setting $t = w/\lambda$, we have

$$t = \frac{w}{\lambda} \geq \frac{w}{w-\frac{d}{2}}$$