GT Lecture 7 Nov. 16

Why 0-pool detection can be applied to find d defectives by using a d-disjunct matrix? $\widehat{M}$

(•) Suppose that $C_1, C_2, \cdots, C_d$ are defectives. Now, for each good item C_{d+1}, $C_{d+1} \not\subseteq \bigcup_{i=1}^{d} C_i$, there exists a test ($i^{th}$ row) such that $M(i, d+1) = 1$ and $M(i,j) = 0$ for all $j = 1, 2, \cdots, d$. Hence, the outcome of the test (i^{th} row) is "0", and thus C_{d+1} is a good item determined by the 0-pool.

Since each good item can be found, those d defectives are determined. //

Non-adaptive algorithms for error-model

Due to the possibility of errors in experiments, a d-disjunct matrix or d-separable matrix may not be able to find all d defectives or find the wrong d defectives. Therefore, it takes more effort in designing a matrix in which we can still find all defectives by allowing certain number of errors occurred in experiments.

First, we need some knowledge of Coding Theory.

Basic Idea in Coding Theory

(To be used in GT with errors occurred in outcomes! (Non-adaptive algorithm))

Definition (q-ary codes)

A q-ary code $\mathbb{C}$ of length n is a subset of $(\mathbb{Z}_q)^n$, i.e., $\mathbb{C}$ is a set of vectors (codewords) selected from $(\mathbb{Z}_q)^n$. Clearly, $(\mathbb{Z}_q)^n$ contains q^n codewords.

In general, $\mathbb{Z}_q$ is replaced by $GF(q)$ a finite field of order q and thus q is chosen to be a prime power. (If $q=2$, then we have a binary code.)

Definition (Distance)

The distance between two codewords $\vec{x} = (x_0, x_1, \cdots, x_{n-1})$ and $\vec{y} = (y_0, y_1, \cdots, y_{n-1})$,

$d(\vec{x}, \vec{y}) = |\{ i \mid x_i \neq y_i,\ i = 0, 1, 2, \cdots, n-1\}|$ i.e., the number of coordinates which are distinct following the order.

Definition (Distance of a code)

Let $\mathbb{C} = \{\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_m\}$ be a code. Then the distance of $\mathbb{C}$,

$d(\mathbb{C}) \underset{def}{=} \min\{ d(\vec{x}_i, \vec{x}_j) \mid 1 \le i < j \le m\}$.

For example, $\mathbb{C} = \{ (1,1,0,1,0,0,0), (0,1,1,0,1,0,0), (0,0,1,1,0,1,0), (0,0,0,1,1,0,1), (1,0,0,0,1,1,0), (0,1,0,0,0,1,1), (1,0,1,0,0,0,1) \}$. $d(\mathbb{C}) = 4$.

(o) <u>Fact</u> If C is code (binary) of distance d, then C can detect $d-1$ (up to) errors and correct $\lfloor \frac{d-1}{2} \rfloor$ (up to) errors.

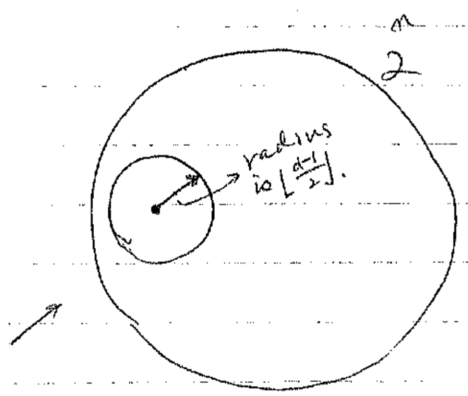
<u>The fundamental problem of coding theory</u>

Let $A_q(n,d)$ denote the maximum number of codewords in a q-ary code of length n with code distance d. (If $q=2$, we use $A(n,d)$ for convenience.)

<u>Example</u> $A(7,3) = 16$.

<u>Fact</u> (Sphere packing bound)

$$A_q(n,d) \le q^n \Big/ \sum_{i=0}^{\lfloor \frac{d-1}{2} \rfloor} \binom{n}{i} \cdot (q-1)^i .$$



$$A(n,d) \le 2^n \Big/ \binom{n}{0} + \binom{n}{1} + \cdots + \binom{n}{\lfloor \frac{d-1}{2} \rfloor} .$$

<u>Example</u> $A(7,3) \le 2^7 \Big/ \binom{7}{0} + \binom{7}{1} = 2^4$.

$A(7,3) \ge 16$

16 codewords

0000000	1111111
1011000	0100111
0101100	1010011
0010110	1101001
0001011	1110100
1000101	0111010
1100010	0011101
0110001	1001110

Outcome Vector : Can be recognized as a set, V
(in error-models)

Denote by $t_0^V(C) = |C \setminus V|$ where C is a column.

e.g.

$$
\begin{array}{c|cccccc|c}
 & 1 & ② & ③ & 4 & 5 & 6 & V \\
 & & \text{positives} & & & & & \\
\hline
t_1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\
t_2 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\
t_3 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\
t_4 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\
\hline
 & C_1 & C_2 & C_3 & C_4 & C_5 & C_6 &
\end{array}
$$

$t_1^V(C) = |C \cap V|$

$\{1,2\}$
$\{1,3\}$
$\{1,4\}$
$\{2,3\}$
$\{2,6\}$
$\{3,4\}$

(*) $V = C_2 \cup C_3$ (without errors), $|V| = 1$, $|V| > 1$ (with errors)

Note that $\underbrace{t_0^V(C) = |C \setminus V|, \ t_1^V(C) = |C \cap V|.}$

$|C \setminus V| > 0 \Rightarrow C$ is a negative item $\Leftarrow$ (X)

$t_0^V(C_1) = 1$	$t_1^V(C_1) = 1$
$t_0^V(C_2) = 0$	$t_1^V(C_2) = 2$
$t_0^V(C_3) = 0$	$t_1^V(C_3) = 2$
$t_0^V(C_4) = 1$	$t_1^V(C_4) = 1$
$t_0^V(C_5) = 1$	$t_1^V(C_5) = 1$
$t_0^V(C_6) = 0$	$t_1^V(C_6) = 2$

↑

Negatives
C_1, C_4, C_5

Two positives: $\begin{cases} ② ③ \text{ or} \\ ② ⑥ \text{ or} \\ ③ ⑥. \end{cases}$

(•) 如果是 2-disjunct 就不会留下 C_2, C_3, C_6 等待判断。

With one error,

$$V = \left\{ \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} + \vec{e} \;\middle|\; \vec{e} \in \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \right\} \right\} \quad (\langle \mathbb{Z}_2, + \rangle).$$

For example, $\begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$. Hence, we may conclude that "6" is positive.

(*) A stronger disjunct matrix is needed in order to determine the positives.

(**) From Coding Theory point of view, we need to make sure all the outcome vectors are with "distance" from each other. By the idea metioned able, if we would like to correct one error spot, two vectors must be at least of distance "3".

(○) With stronger properties, finding such matrices are getting harder, especially if we still ask for less tests.

Item $\to$ clone

Definition ($(d;z)$-disjunct matrix).

A matrix is said to be $(d;z)$-disjunct if for any $d+1$ columns $C_0, C_1, \dots, C_d$,

$$\left| C_0 \setminus \bigcup_{i=1}^{d} C_i \right| \geq z.$$

Note $(d;1)$-disjunct $\approx$ d-disjunct.

(*) There exist at least z rows in each of which C_0 has a "1"-entry and every C_i, $1 \le i \le d$, has a "0"-entry.

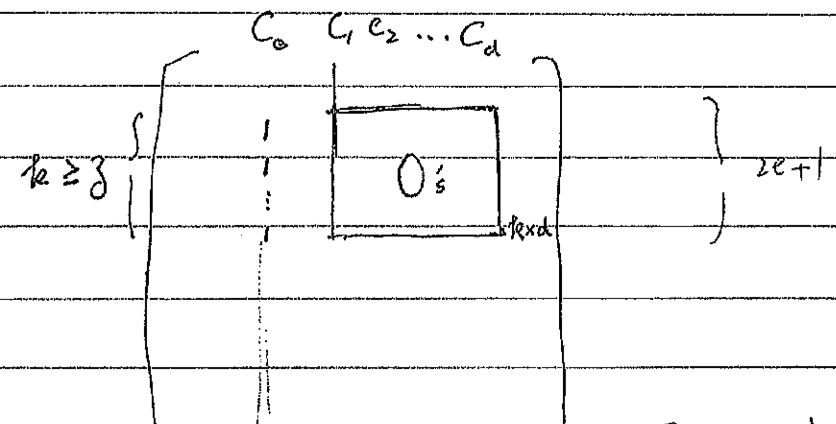


Figure 1

Theorem A $(d; \underset{2e+1}{\overset{\|}{z}})$-disjunct matrix can be applied to identify up-to-d positives with at most e errors.

Proof. We remark that the errors occur on the conclusion of tests, i.e., the outcome vector (as a codeword) has at most e errors. (Due the fact that errors are possible, we shall have a set of possible outcomes instead of just one outcome vector.) Now, assume the number of errors is e. For each negative clone $\bar{C}$, by definition, there exist at least $2e+1$ rows intersecting $\bar{C}$ but none of D (the set of positives). This implies that for each clone $\bar{C}$ or C^+,

$t_0^V(C^-) \geq (2e+1) - e$, and $t_0^V(C^+) \leq e$.

說明：$\left(t_0^V(C^*) =_{def} |C^* \setminus V| \right) \Rightarrow$ See 5′ for examples.

由於 Outcome vector 是所有 Positive clones 的联集；如果用 $(d; 2e+1)$-disjunct 矩陣來测试的话，对于所有 positive clone C^+，$\boxed{t_1^V(C^+) \geq 2e+1}$（没有任何 Errors 產生的情况）。這可以從上頁 Figure 1 看出来：把 C_0 當成 C(positive)，$|V| \geq 2e+1$。

如果把 positives 都放入 $\{C_1, C_2, \cdots, C_d\}$ 而 $\underset{C^-}{\overset{\shortparallel}{C_0}}$ 为 negative，则 $t_0^V(C^-) \geq 2e+1$。

於是在有 e 个錯误的情况下 $t_0^V(C^-) \geq e+1$；

另外 $\underset{|C^+ \setminus V| = |\phi|}{\overset{}{t_0^V(C^+)}} = 0$（没有錯误），在有 e 个錯误的情况下，$t_0^V(C^+) \leq e$。

上述的说明也提供了 Linear Time Decoding Algorithm (on $|N|$)

$\Uparrow$

每个 clone 都测试一次。

(*) How to construct $(d; z)$-disjunct matrices?

(•) We can use a d-disjunct matrix to identify up-to-d positive clones. (Respectively, use a $(d; 2e+1)$-disjunct matrix for error-models with at most e errors.)

<u>Theorem</u> If a set of n clones which contains at most d positive then a d-disjunct matrix can be applied to identify the positives. Moreover, the decoding algorithm is of linear complexity.

<u>Proof.</u> The proof follows by evaluating $t_0^V(C)$ for each clone C. (V is the outcome vector.) We remark here that
$t_0^V(C) = 0$ if C is positive and $t_0^V(C) \geq 1$ if C is negative.
$(t_0^V(C) = |C \setminus V|.)$

∎

(•) If there are at most e errors, then $t_0^V(C^+) \leq e$ and $t_0^V(C^-) \geq e+1$.

(*) <u>Algorithm (Clone Selection (N, V, D, e))</u> possible # of errors

1. for each clone $C \in N$, $(D \leftarrow \emptyset)$
2. compute $t_0^V(C)$
3. if $t_0^V(C) \leq e$
4. then $D \leftarrow D \cup \{C\}$
5. return

$$\boxed{V = \bigcup_{C \text{ positive}} C}$$

(*) 上面定理用 Clone Selection $(N, V, D, 0)$.

(•) <u>Key Point</u> If C is positive, then $t_0^V(C) = 0$; and
(V is the outcome vector.)
C is negative, then $t_0^V(C) \geq 1$.

(Note)

If errors occur, then an adaptive algorithm is getting much complicate than using a non-adaptive algorithm. The reason is very simple, since we have no control on the occurrence of errors. In a word, it may occur in an early stage, so does in later stages.

(*) 如果我们可以确定錯误的產生是来自"过多"的 Items 一起测試，则總 Tests 的個數就不会太多；換句话说，我们可以在每一次测試的 pool 放入不超过可能產生的量，也可以解決這個問題。

(**) 如果是以 Game 的角度來看這個问題，那麼適当引進新的"Adaptive algorithms"是可以獲得較好的结果。

口頭報告的順序 (書面報告繳交截止日期 Jan. 18, 2022.)

1. A study of fake coins problem

   Dec. 7, 呂欣芳 ; Dec. 21, 賴冠宇

2. Determine $M(2, n)$

   Dec. 7, 陳建弘 ; Dec. 21, 郭傑森

3. Generalized binary splitting algorithm

   Dec. 7, 官梓甸 ; Dec. 28, 賴淳偉

4. A study of half-size induced subgraph of a bipartite graph

   Dec. 14, 楊珈禎 ; Dec. 28, 曾永杰

5. Use probabilistic method to construct disjunct matrices

   Dec. 14, 黃伯丞 ; Dec. 28, 楊子儀

6. Find the maximum size of a bipartite graph which is C_4-free

   Dec. 14, 陳彥德 ; Jan. 4, 陳惟彥

7. How to find a hidden subgraph of a complete graph

   Dec. 21, 楊子賢 ; Jan. 4, 溫佳甫