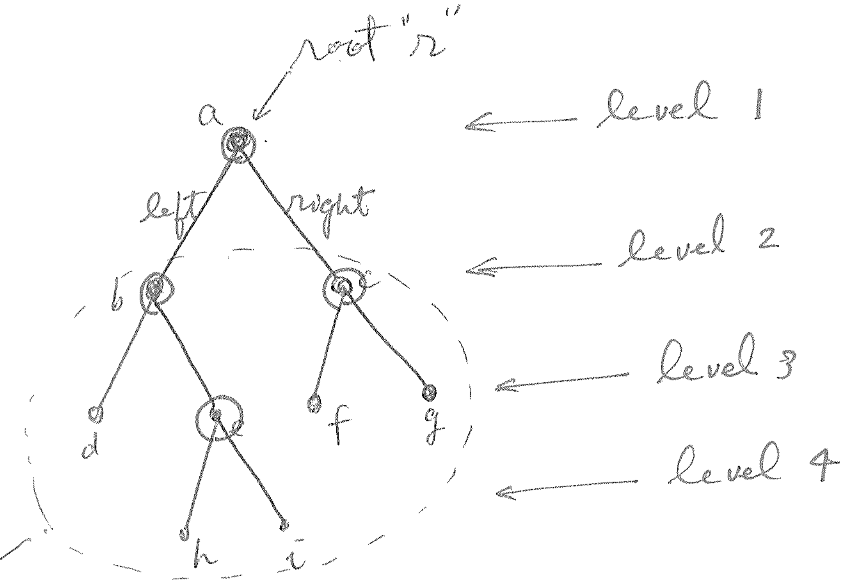
General Sequential Algorithms

1. The Binary Tree Representation



- All nodes are decendants of r. (d,e,h,i 是 b 的 decendants)

- All nodes having a given node as a decendant are ancestors (a,b 为 e 的 ancestors)

- Two immediate decendants are children (h,i 为 e 的 children)

- Two children of the same parent are siblings

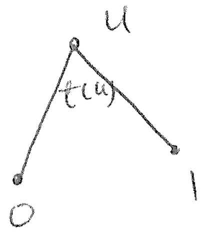
  d,e 为 siblings.

- The immediate ancestor is called a parent (c 为 f, g 的 parent)

- Nodes which have no decendants are leaves and the other nodes are internal nodes

Let S denote the sample space, i.e., the set of items to be tested. Then a group testing algorithm T for S can be represented by a binary tree, also denoted by T, by the following rules:

(i) Every internal node u is associated with a test $t(u)$; its two links associated with the two outcomes of $t(u)$, which are 0 (negative) and 1 (positive).
(left) (right)



(Note: The <u>test history</u> $H(u)$ of a node u is the set of tests and outcomes associated with the nodes and links on the path of u.)
(r to u)

(ii) Each node u is also associated with an event $S(u)$ which consists of all the members of S consistent with $H(u)$. $|S(v)| \le 1$ for each leaf v.

Let $p(s)$ denote the path of the leaf v associated with the sample point s and let $|p(s)|$ denote its length.

- $M_T(S) = \max_{s} |p(s)|$

✓ • $\sum_{s \in S} 2^{-|p(s)|} = 1$ for every reasonable algorithm.

- $M(d,n) \le M(d,n+1)$

  $(M(1,64) < M(1,65))$

Let $M(m;d,n)$ denote the minimum number of tests necessary to identify the d defectives among n items when a particular subset of m items is known to be contaminated.

- $M(m;d,n) \ge 1 + M(d-1, n-1)$ for $m \ge 2$ and $0<d<n$,

  (Theorem)

Proof. It suffices to consider $m=2$ since

$$M(m;d,n) \ge M(2;d,n).$$

Let T be an algorithm for the $(2;d,n)$ problem, and let $M_T(2;d,n) = k$, i.e., k is the maximum length of a leaf in T (binary tree representation). Let I_1 and I_2 be the two items in the contaminated group.

$$\vdots$$

- $M(d,n) \le n-1$.

- $M(n-1,n) = n-1$

(*) <u>Remark</u>

If d is quite large compare to n, then individual testing algorithm is the best!

$$M(n-1,n) = M(n;\, n-1,n)$$
$$\ge 1 + M(n-2,n-1)$$
$$\ge \cdots \ge n-1 + M(0,1) = n-1.$$

$\Uparrow$

why?

Individual testing algorithm shows that $M(n-1,n) \le n-1$.

在找到d了
以前已經有d-1個

(0) $M(d,n) \ge M(d-1,n)$

- $M(d,n) \ge 1 + M(d-1,n-1) \ge M(d-1,n)$ for $0<d<n$.

$$M(d-1,n) \le M_T(d-1,n) \quad (d\ge 2)$$
$$= 1 + \max\{M(d-1,n-1),\ M(d-2,n-1)\}.$$

任選一個 test

$$= 1 + M(d-1,n-1).$$

- Suppose that $n-d>1$. Then

$$M(d,n) = n-1 \Rightarrow M(d,n-1) = n-2.$$

- $M(d,n) = M(d-1,n) \Rightarrow M(d,n) = n-1$.

(*) • Suppose $M(d,n) < n-1$. Then
$M(d,n) \geq 2l + M(d-l, n-l)$ for
$0 < l \leq d < n$.

- $M(d,n) \geq \min\{n-1, 2l + \lceil \log_2 \binom{n-l}{d-l} \rceil\}$ for
$0 < l \leq d < n$.

(**) • $M(d,n) + 1 \geq M(\bar{d}, n) \geq M(d, n+1)$

<u>Theorem</u> For each algorithm T of (d,n) problem, there exists an algorithm T' for the $(\bar{d},n)$ problem such that

$$M_T(d,n) + 1 \geq M_{T'}(\bar{d},n).$$

The structure of GT

3, 2, 2016 — 3, 3

①

Consider a set of six items containing exactly two defectives.

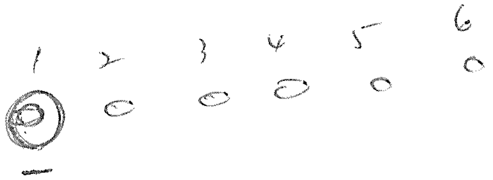
Informatic lower bound :
(Information)

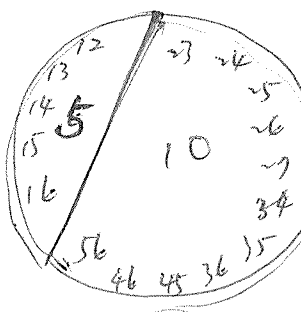
$$\left\lceil \log_2 \binom{6}{2} \right\rceil = \lceil \log_2 15 \rceil = 4 .$$

In general,

$$\left\lceil \log_2 \binom{n}{d} \right\rceil \approx d \lceil \log_2 n \rceil - ?$$

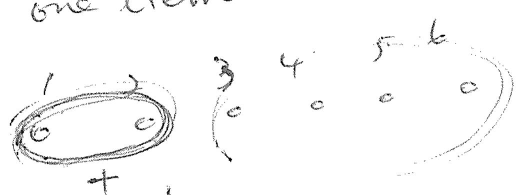
But, how to find the two defectives in 4 tests ?

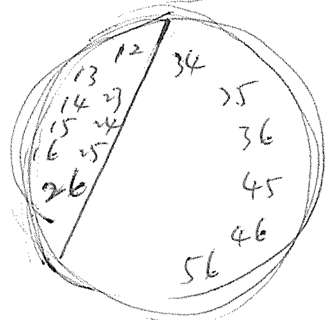




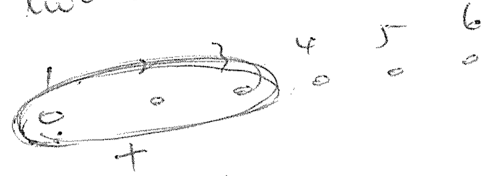
都需要至少5次 (worst case).

① Test one item

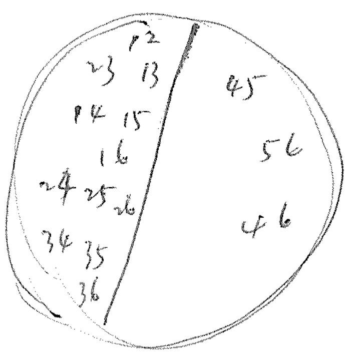




② Test two items



③ Test three items



④ Test 4 items

+

(*) 對於 defectives 的個數 d 大於 1 的時候，
Splitting algorithm 並不是把 $\binom{n}{d}$ 個子集合
一次分成兩半；如上頁所示，要怎麼分跟
Group Test 有關；例如 (12)(3456) 就把
$\binom{6}{2}$ 個集合分成 9, 6 的兩組而不是我們希望的
"一半，一半"。（Binary splitting algorithm）

Generalized binary splitting algorithm [Hwang, 1972]
JASA 67, 605–608.
A method for detecting all defective members in a population by group testing.

Algorithm (GBS)

Step 1. If $n \le 2d-2$, test the n items individually.
If $n \ge 2d-1$, set $l = n-d+1$. Define

$$\alpha = \lfloor \log(\frac{l}{d}) \rfloor.$$

Step 2. If $n > 2d-2$, test a group of size 2^α which is about $\frac{l}{d}$. If the outcome is negative, the 2^α items are good. Set $n := n-2^\alpha$ and go to Step 1. If the outcome is positive, use binary splitting algorithm to identify one defective and an unspecified number of "x" good items. Set $n := n-1-x$ and $d := d-1$. Go to Step 1.

Theorem (Hwang)

$$M_{GBS}(d,n) = \begin{cases} n & \text{for } n-1 \le 2d-2 \\ (\alpha+2)d+p-1 & \text{for } n \ge 2d-1 \end{cases}$$

one algorithm

where $p<d$ is a non-negative integer uniquely defined in $l = 2^\alpha d + 2^\alpha p + \theta$, $0 \le \theta < 2^\alpha$.

Proof is omitted.

Example

$d=2$, $n=100$

$$M_{GBS}(2,100) = \begin{cases} \\ \underbrace{(\alpha+2)}_{14} d + \underbrace{p}_{1} - 1 \end{cases}$$

$l = 100-2+1 = 99$

$\alpha = \left\lfloor \log \frac{99}{2} \right\rfloor = 5$

$l = 2^\alpha d + 2^\alpha p + \theta$

$99 = \underbrace{32 \cdot 2}_{64} + \underbrace{32 \cdot p}_{32} + \underbrace{\theta}_{3}$

$p=1$

The most important contribution of this theorem.

① For $\frac{n}{d}$ large.

$$M_{GBS}(d,n) \longrightarrow d\log\left(\frac{n}{d}\right) \qquad (?)$$

Proof.

$$M_{GBS}(d,n) = (\alpha+2)d + p - 1 = \alpha d + 2d + p - 1$$

$$\alpha = \left\lfloor \log \frac{n-d+1}{d} \right\rfloor$$

$$= \left\lfloor \log\left(\frac{n}{d} - \frac{d-1}{d}\right) \right\rfloor$$

$$\approx \left\lfloor \log \frac{n}{d} \right\rfloor \text{ if } \frac{n}{d} \text{ is large.}$$

$$M_{GBS}(d,n) \approx d \cdot \left\lfloor \log \frac{n}{d} \right\rfloor \qquad 2d+p-1$$

相对较小,

显然 $d\log\frac{n}{d}$ 小于 $d\lceil \log n \rceil$

Corollary $M_{GBS}(d,n) - \lceil \log \binom{n}{d} \rceil \le d-1$ for $d \ge 2$.

Proof.

$$M_{GBS}(d,n) = \alpha d + 2d + p - 1$$
$$= \boxed{\alpha d + d + p} + d - 1.$$

It suffices to show that $\alpha d + d + p \le \lceil \log \binom{n}{d} \rceil$.

$$\binom{n}{d} = \binom{l+d-1}{d} = \frac{(l+d-1)(l+d-2)\cdots(l)}{d(d-1)\cdots 1}$$
$$\ge \frac{l^d}{d!} = \frac{(2^\alpha d + 2^\alpha p + \theta)^d}{d!}$$
$$\ge \frac{(2^\alpha (d+p))^d}{d!} = \frac{\left(2^\alpha \cdot d \cdot \left(1+\frac{p}{d}\right)\right)^d}{d!}$$
$$= \boxed{\frac{d^d}{d!}} \cdot 2^{\alpha d} \cdot \left(1+\frac{p}{d}\right)^d$$
$$\ge 2^{d-1} \cdot 2^{\alpha d} \cdot 2^p$$
$$= 2^{\alpha d + d + p - 1}$$

$\boxed{\left(1+\frac{p}{d}\right)^d \ge 2^p}$

$d \ge 2$, $p < d$.

(令 $f(x) = \left(1+\frac{p}{x}\right)^x$, $f'(x) > 0$.)

$$\frac{k^k}{k!} \ge 2^{(k-1)}.$$

$$\frac{(k+1)^{k+1}}{(k+1)!} = \frac{\cancel{(k+1)}(k+1)^k}{\cancel{(k+1)}\cdot k!} \qquad \frac{(k+1)^k}{k^k} = \left(\frac{k+1}{k}\right)^k$$
$$= \left(1+\frac{1}{k}\right)^k = \left(1^k + k\cdot\frac{1}{k} + \ldots\right)$$
$$\ge 2$$

$p=0$, true

⑥

To prove $\left(1+\frac{p}{d}\right)^d \geq 2^p$, it suffices to show the function

$$f(x)=\left(1+\frac{p}{x}\right)^x \text{ is increasing on } x.$$

$$f'(x)=\left(1+\frac{p}{x}\right)' \cdot \ln\left(1+\frac{p}{x}\right)\cdot\left(1+\frac{p}{x}\right)^x$$

$$=\underline{\left(1-\frac{p}{x^2}\right)}\,\underline{\left(1+\frac{p}{x}\right)^x\left(1+\frac{p}{x}\right)^x}$$

For $x \geq 2$

$f'(x)>0$.

$$\binom{n}{d} \geq 2^{\alpha d+d+p-1}$$

$$\Rightarrow \log_2\binom{n}{d} \geq \alpha d+d+p-1$$

$$\Rightarrow \left\lceil \log_2\binom{n}{d}\right\rceil \geq \alpha d+d+p. \qquad \blacksquare$$

Remark

上述的演算法需要先计算 α，而且它随着 n,d 的改变而变动；$n=2^\alpha$，所以算 α 是 constant time.

總共的時間 $O(d\log\frac{n}{d})$. ← 一修演算法喔

隨機演算法 (Probabilistic Group Testing)

Sobel and Groll [1959, Bell System Tech. J., 1179–1252]
Group Testing to eliminate efficiently all defectives in a binomial sample

(Nested class) Algorithm

1. There is no restriction on the test group until a group is tested to be contaminated (positive). Mark this group the "current contaminated group" (C.C.G) & denote it by C.

2. The next group must on a group, ~~say~~ G, $G \subsetneq C$. If G is positive, then replaces C as the current contaminated group. Otherwise, items in G are good and $C \setminus G$ replaces C as C.C.G.

3. If the C.C.G. is of size one, identify the item in the group as defective. Test any group of unidentified items, if any!

(*) (GBS is also in nested class.)

Discussion (Recurssive Conditions)

Let $H(d,n)$ denote the number of tests that in a mini-max nested algorithm and $F(m;d,n)$ denote the same except for the existence of a c.c.G. of size m.

$$H(d,n) = \min_{1\le m<n} \max\{H(d,n-m), F(m;d,n)\}$$

$$F(m;d,n) = \min_{1\le k<m} \max\{F(m-k;d,n-k), F(k;d,n)\}$$

with boundary conditions

$$H(d,d) = H(0,n) = 0.$$

$$F(1,d,n) = H(d-1,n-1).$$

(※) 有三个参数 d, n, m；暴力法大约需要 $O(dn^3)$ 的时间。(高起来)

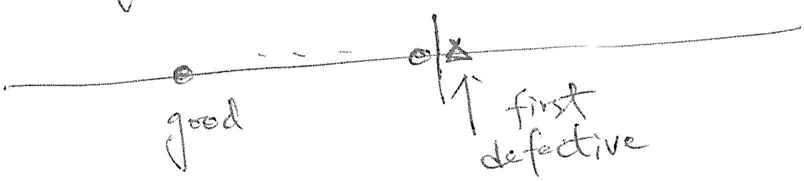
详细的分析可以降低它的复杂度。

Line algorithm

Define a line algorithm as one which orders the unclassified items linearly and always tests a group at the top of the order.

(*) A line algorithm identifies the items in order except

1. A good item may be identified together with a seq. of items up to the first defective after it.



2. When only one unidentified defective is left, then the order of identification is from both ends towards the defective (this is because once a contaminated group is identified, all the other items ~~one~~ can be deduced to be good).

(Fact) Every nested algorithm can be implemented as a line algorithm.

$O(d^2(\log(\frac{n}{d}))^2)$

有用的性质 ((d,n)-problem).

(*) There exists an algorithm which can identify a defective among n items in at most $k \underset{\text{def}}{=} \lceil \log_2 n \rceil$ tests. Furthermore, if $\lceil \log_2 n \rceil$ tests are actually used, then at least $2^k - n$ good items are also identified.

Proof. By induction on k. $k=1 \Rightarrow n=2$ ✓

$k=2$ $n=3$ or 4	$n=3$	o o	o

1. $n-2^{k-1} > 2^{k-2}$. $(2n-2^k > 2^{k-1})$

Test a group of $n-2^{k-1}$ items.

negative

$n-2^{k-1} \geq 2^k - n$ good items are identified

Use $k-1$ tests to find a defective in 2^{k-1} items left.

positive

By induction at least $2^{k-1} - (n-2^{k-1}) = 2^k - n$ good items are identified.

Again

2. $n - 2^{k-1} \le 2^{k-2}$.

Test a group of 2^{k-2} items.

negative

2^{k-2} good items are already identified and

$n - 2^{k-1} \le 2^{k-2}$ ✓

$\quad\;\; \vee|$

$2^k - n$

By induction another $2^{k-1} - (n - 2^{k-2})$ good items will be identified in the remaining $n - 2^{k-2}$ items along with a defective if $k-1$ more tests are used.

positive

a defective can be identified in the 2^{k-2} items in $k-2$ tests and a total of $k-1$ tests are used.

$\Downarrow$ How many good items are identified?