Probabilistic Constructions of Pooling Designs

Hung-Lin Fu
Deparment of Applied Mathematics
National Chiao Tung University
Hsin Chu, Taiwan

Group Testing

- Robert Dorfman's paper in 1943 introduced the field of (Combinatorial) Group Testing. The motivation arose during the Second World War when the United States Public Health Service and the Selective service embarked upon a large scale project. The objective was to weed out all syphilitic (梅毒) men called up for induction. However, syphilis testing back then was expensive and testing every soldier individually would have been very cost heavy and inefficient.

Formal Definitions

- Consider a set N of n items consisting of at most d *positive* (used to be called defective) items with the others being *negative* (used to be called good) items.
- A group test, sometimes called a *pool*, can be applied to an arbitrary set S of items with two possible outcomes; *negative:* all items in S are negative; *positive:* at least one positive item in S, not knowing which or how many.

Adaptive (Sequential) and Non-adaptive

- There are two basic algorithms.
- The first one (*adaptive or sequential*): You ask the second question (query) after knowing the answer of the first one and continue …. That is, the previous knowledge will be used later.
- The second one (*non-adaptive*): You can ask all the questions (queries) at the same time.

Deterministic Algorithms

- Adaptive algorithm
- Non-adaptive algorithm
- k-stage algorithm

The most popular one is a 2-stage algorithm in which *we use non-adaptive algorithm in the first stage and then in the second stage we test the left "suspected" items one by one.*

Non-adaptive Algorithm

We can use a matrix A to describe a non-adaptive algorithm.

Items are indexed by columns and the tests are indexed by rows. Therefore, the A(i, j) = 1 if the item j is included in the *pool* i (for test), and 0 otherwise.

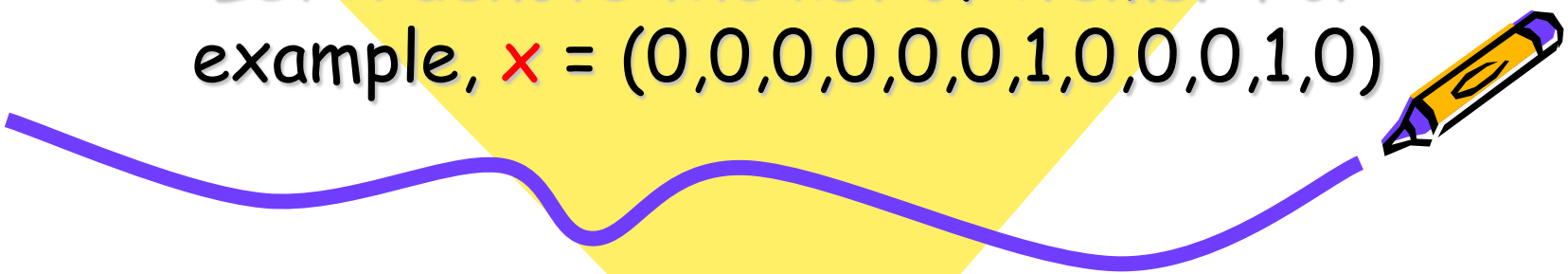
Incidence Matrix

Blanks are zeros

1			1			1			1		
1				1			1			1	
1					1			1			1
	1		1					1		1	
	1			1		1					1
	1				1		1		1		
		1	1				1				1
		1		1				1	1		
		1			1	1				1	

Implementation

Let x denote the list of items. For example, x = (0,0,0,0,0,0,1,0,0,0,1,0)



Outcome Vectors

- The vector **y** is an *outcome vector* which is corresponding to an input **x**.
- So, $\mathbf{y}^t$ = (1,1,0,1,1,0,0,0,1,0,0,0).
- If *A* is 1-1, then we can decode **x** long as we know its outcome vector.
- In order to get the job done with lower decoding complexity, extra properties for *A* is needed.

Can we find positives from the above matrix?

- Yes, we can if the number of positives is not too many, say at most 2, by running the 9 tests simultaneously corresponding to rows.
- The reason is that the union of (at most) 2 columns can not contain any other distinct column. (?)

Decoding Idea

- Since there are 12 items and the number of positives is at most 2, we have 1 + 12 + 66 possible inputs. (?)
- There are 2^9 possible distinct outcome vectors.
- It seems that we can use a matrix with less rows. The question is "how to construct such a matrix".

Set Notation (Design)

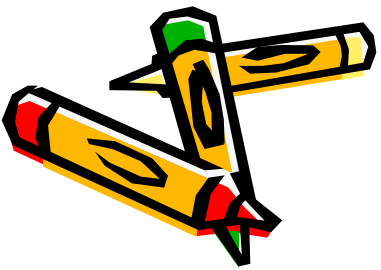
- Let $M = [m_{i,j}]$ be a txn matrix mentioned above. Then we can use n sets (ordered) S_i's to represent the matrix where

  $S_k = \{i : m_{i,k} = 1, i = 1, 2, \ldots, t\}, k = 1, 2, \ldots, n.$

- The following sets represent the (0,1)-matrix of the last 9x12 matrix:

  {1,2,3}, {4,5,6}, {7,8,9}, {1,4,7}, {2,5,8}, {3,6,9}, {1,5,9}, ({2,6,7}, {3,4,8}, {1,6,8}, {2,4,9}, {3,5,7}.

  (Projective plane of order 3.)



d-separable and d-disjunct matrices

- A matrix is *d-separable* if $\cup D \neq \cup D'$ for any two distinct d-sets D and D' (columns), i.e. no two unions of d columns are the same.
- A matrix is *d-disjunct* if no column is contained in the union of any other d columns.

Important Facts

- A d-disjunct matrix is also a d-separable matrix.
- *A d-separable matrix can be applied to find exactly d positives.*

  Proof. The union of d columns corresponds to distinct *outcome* vector.
- d-disjunct matrices have a simple decoding algorithm, namely, a column is positive if and only if it does not appear in a negative row. (The above matrix is 2-disjunct.)

Various Models

- *Errors* occurred.
- There are *inhibitors*.
- There is a *threshold*.
- Defective items are sets: *complex* model.
- *Competitive* model: the number of defectives is unknown.
- The defectives are *mutually obscuring*.

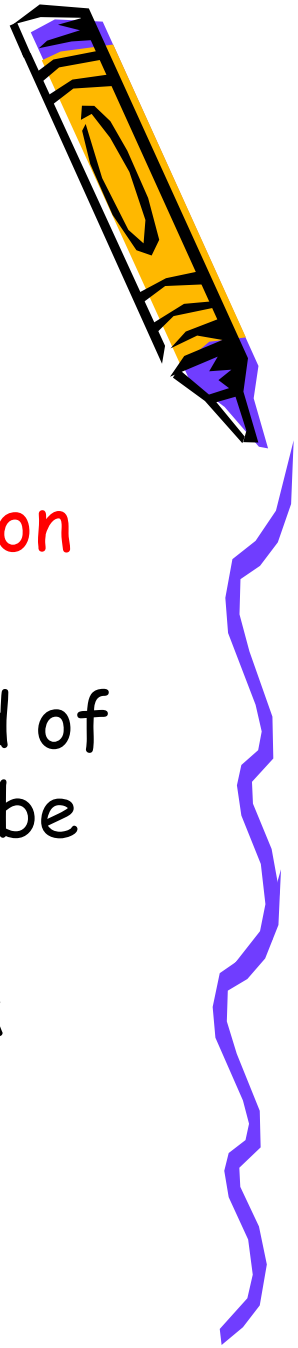
Disjunct Matrices

- Disjunct matrices can be considered as special designs or special codes.
- Extra constrains are needed in order to handle various models in group testing.
- We are aiming at finding all the defectives with comparatively easier decoding algorithms.
- Of course, also minimize the number of tests.

(d; z)-disjunct

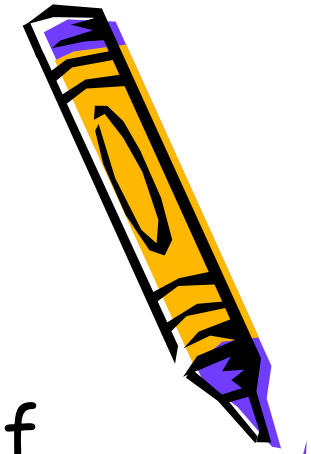
- Let A be a $t \times n$ (0, 1)-matrix in which the rows are indexed by $[1, t]$ and the columns are indexed with the supports of columns. (Columns are subsets of $[1, t]$.)
- A is (d; z)-disjunct if for any d+1 columns $C_0, C_1, \ldots, C_d$, $|C_0/(C_1 \cup C_2 \cup \ldots \cup C_d)| \geq z$.
- If $z \geq 2e + 1$, then this matrix can be applied to find d defectives with e errors in outcome vectors.

(d, r]-disjunct



- A is (d, r]-disjunct if the union of any d columns does not contain the intersection of any other r columns.
- If the defectives are r-subsets instead of items (complex model), this matrix can be utilized to find d defective complexes.
- In case of errors occurred, we define a
- (d, r; z]-disjunct matrix accordingly.

(d, h)-disjunct



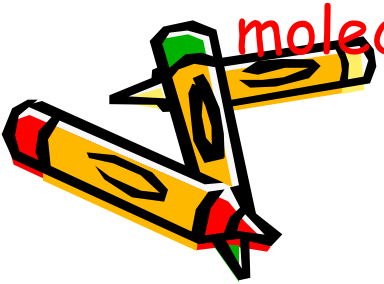
- A is called (d, h)-disjunct if the union of any d columns does not contain the union of the other h columns.
- A can be applied to find d defectives in an inhibitor model with h inhibitors.
- We need extra effort to handle this model with errors!

(d, s out of r]-disjunct

- *A* is (d, s out of r]-disjunct if for any d columns and any other r columns, there exists a row (index) in which none of the d columns appear and at least s of r columns do.
- If s = 1, then *A* is (d, r)-disjunct and if s = r, then *A* is (d, r]-disjunct.

(k, m, n)-selector

- For $m \leq k$, *A is a (k, m, n)-selector* if any $t \times k$ submatrix of *A* contains at least m rows of the identity matrix I_k.
- *A (k, m, n)-selector* corresponds to well-known combinatorial objects such as superimposed code and *k*-selectors.
- Superimposed codes and *k*-selectors are very basic combinatorial structures and find application in a variety of areas such as cryptography, data security, computational molecular biology etc.



Construction of disjunct-matrices

- There are known constructions, deterministic and probabilistic.
- Here, we shall use the well-known Lovász Local Lemma to construct such matrices.
- We shall present one of them and the others are similar in some sense.

Lovász Local Lemma

- (Symmetric Case)

Let $A_1, A_2, \ldots, A_m$ be events in an arbitrary probability space. Suppose that each event A_i is mutually independent of a set of other events A_j but at most μ of them, and that $Pr(A_i)$ is at most $0 < p < 1$ for all $1 \leq I \leq m$.

If $e \cdot p \cdot (\mu + 1) \leq 1$, then the product of m probabilities $Pr(B_i)$ is larger than 0 where B_i is the complement of the event A_i.

(d, r]-disjunct Case

- Let t(n, d, r] denote the minimum number of rows for a (d, r]-disjunct matrix with n columns.
- For convenience of "typing", we shall use ((n, d)) to denote the combination number: n chooses d, i. e. $n!/d!\cdot(n-d)!$, and (([n], k)) to denote all k-subsets of {1, 2, …, n}.
- Notice that $-\ln(1-x) \geq x$ for $0 \leq x < 1$.

Upper Bound

- $t(n, d, r] \leq (1 + d/r)^r(1 + r/d)^d \cdot$
  $\{1 + \ln[((n, d))((n-d, r)) - ((n-d-r, d))\cdot$
  $((n-2d-r, r))]\}.$

Proof. (Sketch)

Let M be a $t \times n$ (0, 1)-matrix with $Pr(m_{i,j} = 1) = p$ and $Pr(m_{i,j} = 0) = 1 - p$. Note that all entries are chosen independently.

- Let $D \in (([n], d))$, $R \in (([n], r))$, $D \cap R = \emptyset$ and $A_{D,R}$ be the event such that $\cap R \subseteq \cup D$. Hence, the complement of $A_{D,R}$ says that $\cap R$ is not contained in $\cup D$.
- $Pr(A_{D,R}) = (1 - p^r(1-p)^d)^t$.
- $(\mu + 1) = ((n, d)) \cdot ((n-d, r)) - ((n-d-r, d)) \cdot ((n-2d-r, r))$.
- Hence, $e \cdot (1 - p^r(1-p)^d)^t \cdot [((n, d)) \cdot ((n-d, r)) - ((n-d-r, d)) \cdot ((n-2d-r, r))] \leq 1$ is required to make sure that the intersection of the complements of all events $A_{D,R}$ does have nonnegative probability and hence we have a (d, r]-disjunct matrix.

- Solving the inequality, we have

  $t \geq (1 + \ln[((n, d)) \cdot ((n-d, r)) - ((n-d-r, d)) \cdot ((n-2d-r, r))] / (-\ln((1 - p^r(1-p)^d))$. … (*)

  By the fact $-\ln(1-x) \geq x$ for $0 \leq x < 1$, if

  $t \geq (1 + \ln[((n, d)) \cdot ((n-d, r)) - ((n-d-r, d)) \cdot ((n-2d-r, r))] / p^r(1-p)^d$, then (*) holds. …(**)

- Let $p = r/(d+r)$. Then the RHS of (**) has the minimum value. Hence, as long as we have this number of tests, a (d, r]-disjunct matrix does exist. This implies that $t(n, d, r] \leq$ RHS of (**).

References

- A new construction of 3-bar-*separable* matrices via improved decoding of Macula construction (with F. K. Hwang), **Discrete Optim.**, 5(2008), 700-704.
- An upper bound of the number of tests in pooling designs for the *error-tolerant complex model* (with Hong-Bin. Chen and F. K. Hwang), **Optimization Letters**, 2(2008), no. 3, 425-431.
- The minimum number of e-vertex-cover among *hypergraphs* with e edges of given rank (with F. H. Chang, F. K. Hwang and B. C. Lin), **Discrete Applied Math.**, 157(2009), 164-169.
- Non-adaptive algorithms for *threshold* group testing (with Hong-Bin Chen), **Discrete Applied Math.**, 157(2009), 1581-1585.

Continued

- Identification and classification problem on pooling designs for *inhibitor* models (with Huilan Chang and Hong-Bin Chen), **J. Computational Biology**, 17(2010), No. 7, 927-941.
- Reconstruction of *hidden graphs and threshold* group testing (with Huilan Chang, Hong-Bin Chen and Chih-Huai Shih), **J. Combin. Optimization**, 22(2011), no. 2, 270 – 281.
- Group testing with multiple *mutually-obscuring positives* (with Hong-Bin Chen), **Lecture Notes Computer Science,** 7777(2013), 557 – 568.
- *Threshold* group testing on *inhibitor* model (Huilan Chang and Chih-Huai Shih), **J. Computational Biology**, Vol. 20, No. 6, 2013, 1 – 7.

Continued

- New bound on *2-bar-separable* codes of length 2 (with Minquan Cheng, J. Jiang, Yuan-Hsun Lo and Ying Miao), **Des. Code Cryptography**, to appear.
- Learning a *hidden graph* (with Huilan Chang and Chie-Huai Shih), under review of Optimization Letters.
- Kuo-An Yu, Applications of the *Lovász Local Lemma* to pooling designs, M.S. thesis, 2007, National Chiao Tung University.