IEEE Information Theory Society

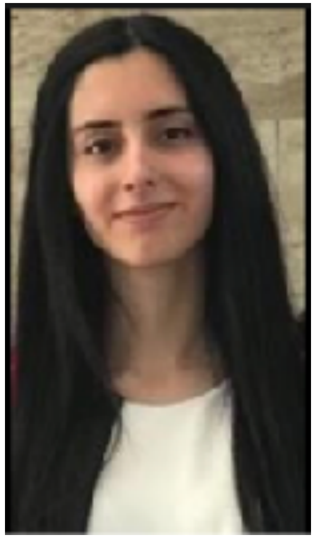
UCLA

Community Aware Group Testing

Christina Fragouli

North America School of Information Theory, August 2022

ARNI
Algorithmic Research in Network Information flow

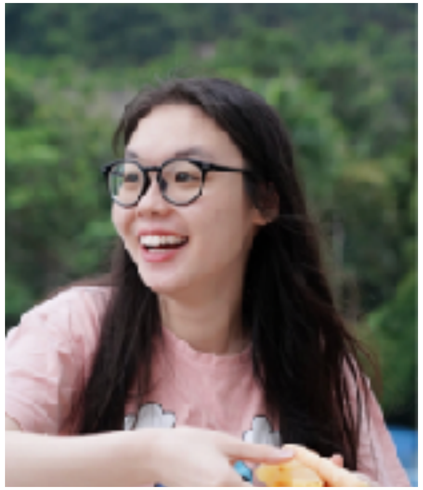


Mine Dogan

Osama Hanna

Merve Karakas

Xinlin Li



Chaorui Yao

UCLA

Thanks to:

Sundar Rajan
Meta (Facebook)

Pavlos Nikolopoulos
EPFL

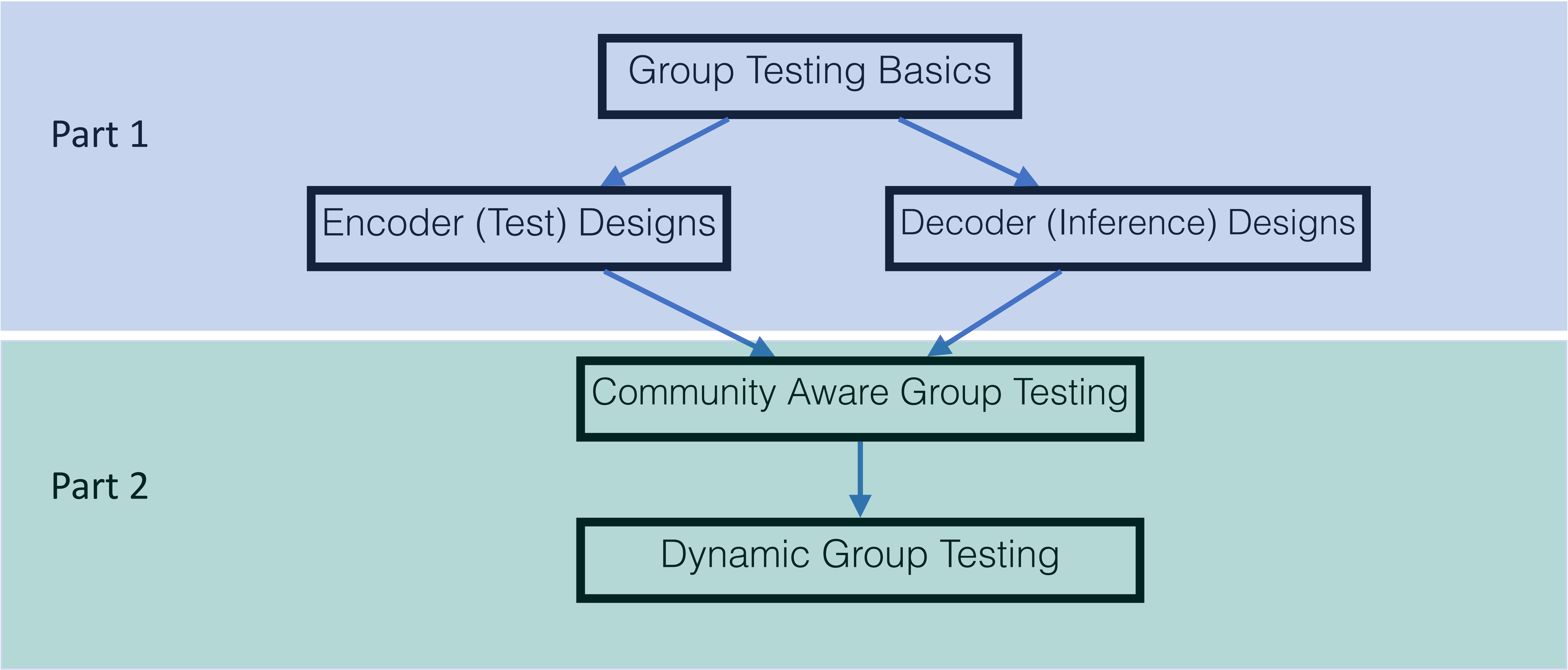
NSF CIF Award 2146828
Group Testing for Epidemics Control

Some references

Disclaimer: this is not an exhaustive coverage of the topic. The goal is to provide an introduction and hopefully convince you that there are many open problems that are interesting to potentially work on.

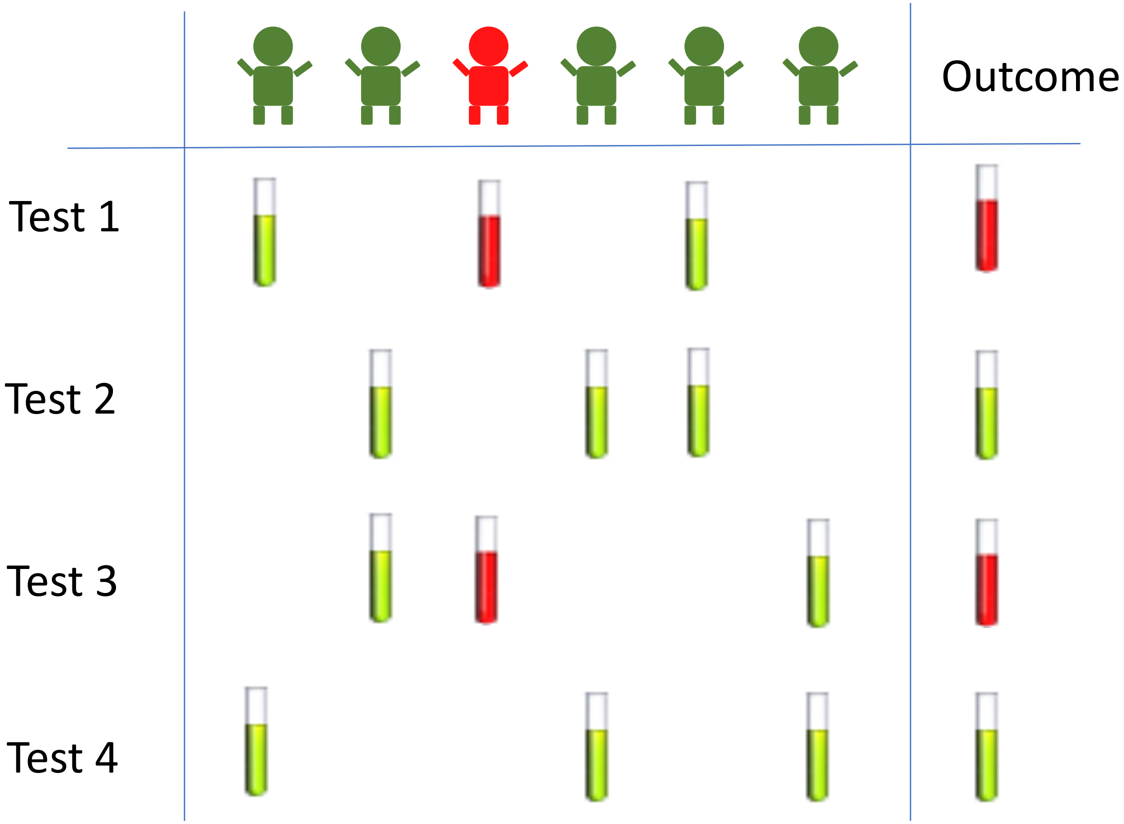
- M. Aldridge, O. Johnson and J. Scarlett Group testing: An Information Theory Perspective, monograph, Now Founrdations and Trends, 2019.
- R. Dorfman, "The Detection of Defective Members of Large Populations," The Annals of Mathematical Statistics, 1943
- P. Nikolopoulos, S. R. Srinivasavaradhan, C. Fragouli and S. N. Diggavi, "Group Testing for Community Infections," in *IEEE BITS the Information Theory Magazine*, vol. 1, no. 1, pp. 57-68, 1 Sept. 2021, doi: 10.1109/MBITS.2021.3126244.
- P. Nikolopoulos, S. R. Srinivasavaradhan, T. Guo, C. Fragouli, S. N. Diggavi, "Group testing for connected communities", In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, PMLR 130:2341-2349, 2021 (ArXiv version June 2020).
- P. Nikolopoulos, S. R. Srinivasavaradhan, T. Guo, C. Fragouli and S. Diggavi, "Group testing for overlapping communities," *ICC 2021 - IEEE International Conference on Communications*, 2021, pp. 1-7, doi: 10.1109/ICC42927.2021.9500791.
- S. R. Srinivasavaradhan, P. Nikolopoulos, C. Fragouli and S. Diggavi, "An entropy reduction approach to continual testing," *2021 IEEE International Symposium on Information Theory (ISIT)*, 2021, pp. 611-616, doi: 10.1109/ISIT45174.2021.9518188.
- S. R. Srinivasavaradhan, P. Nikolopoulos, C. Fragouli and S. Diggavi, "Dynamic group testing to control and monitor disease progression in a population," *arXiv e-prints*,arXiv:2106.10765. June 2021.
- T.Li, C.L.Chan, W.Huang, T.Kaced, and S.Jaggi, "Group testing with prior statistics," in *2014 IEEE International Symposium on Information Theory*, 2014, pp. 2346–2350.
- E. Price and J. Scarlett, A Fast Binary Splitting Approach to Non-Adaptive Group Testing, arXiv:2006.10268v1, 2020.
- T. Li, C. L. Chan, W. Huang, T. Kaced, and S. Jaggi. Group testing with prior statistics. In IEEE International Symposium on Information Theory (ISIT), pages 2346–2350, 2014. doi: 10.1109/ISIT.2014.6875253.
- C. L. Chan, S. Jaggi, V. Saligrama, and S. Agnihotri, "Non-adaptive group testing: Explicit bounds and novel algorithms," in *Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on*. IEEE, 2012, pp. 1837–1841.
- S. Ahn, W. -N. Chen and A. Özgür, "Adaptive Group Testing on Networks with Community Structure," *2021 IEEE International Symposium on Information Theory (ISIT)*, 2021, pp. 1242-1247, doi: 10.1109/ISIT45174.2021.9517888.
- B. Arasli and S. Ulukus, "Group testing with a graph infection spread model," 2021, arXiv:2101.05792.
- J. Zhu, K. Rivera, and D. Baron, "Noisy pooled PCR for virus testing," 2020, arXiv:2004.02689.
- R. Goenka, S.-J. Cao, C.-W. Wong, A. Rajwade, and D. Baron, "Contact tracing enhances the efficiency of COVID-19 group testing," in Proc. IEEE Int. Conf. Acoust., Speech Signal Process., 2021, pp. 8168–8172, doi: 10.1109/ICASSP39728.2021.9414034.
- AC-DC: Amplification Curve Diagnostics for Covid-19 Group Testing, Ryan Gabrys, Srilakshmi Pattabiraman, Vishal Rana, João Ribeiro, Mahdi Cheraghchi, Venkatesan Guruswami, Olgica Milenkovic, ArXiv 2021
- M. Cheraghchi, A. Karbasi, S. Mohajer and V. Saligrama, "Graph-constrained group testing," *2010 IEEE International Symposium on Information Theory*, 2010
- S. Sihag, A. Tajer and U. Mitra, "Adaptive Graph-Constrained Group Testing," in *IEEE Transactions on Signal Processing*, vol. 70, pp. 381-396, 2022, doi: 10.1109/TSP.2021.3137026.

Outline


Part 1
Group Testing Basics
Encoder (Test) Designs
Decoder (Inference) Designs
Part 2
Community Aware Group Testing
Dynamic Group Testing

Group testing

Idea of pooling together multiple diagnostic samples to make medical testing more efficient

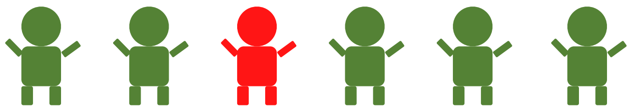


- To find the infected individual, we need: 6 individual tests or 4 pooled tests
- A pooled test is positive, if at least one participating individual is infected

R. Dorfman, "The Detection of Defective Members of Large Populations," The Annals of Mathematical Statistics, 1943

Group testing: notation

$U = [0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0]$



							Outcome
Test 1	1	0	1	0	1	0	1
Test 2	0	1	0	1	1	0	0
Test 3	0	1	1	0	0	1	1
Test 4	1	0	0	1	0	1	0

Test matrix G $\quad$ Y

- Population of N individuals
- $U = (U_1, U_2, \ldots, U_N)$, infection vector $U_i = 1$ if individual i is infected, and 0 otherwise
- $Y = (Y_1, Y_2, \ldots, Y_T)$, the vector of test outcomes
- Test matrix G where $G_{ti} = 1$ if individual i is in test t

$$\begin{bmatrix} Y_1 \\ \vdots \\ Y_T \end{bmatrix} = \begin{bmatrix} G_{11} & \cdots & G_{1N} \\ \vdots & \ddots & \vdots \\ G_{T1} & \cdots & G_{TN} \end{bmatrix} \begin{bmatrix} U_1 \\ \vdots \\ U_N \end{bmatrix}$$

$$Y = GU$$

Matrix multiplication in Boolean algebra

- Infections: combinatorial (k out of N), or probabilistic

$$U_i = \begin{cases} 1 & w.p. \quad p_i \\ 0 & w.p. \quad 1 - p_i \end{cases}$$

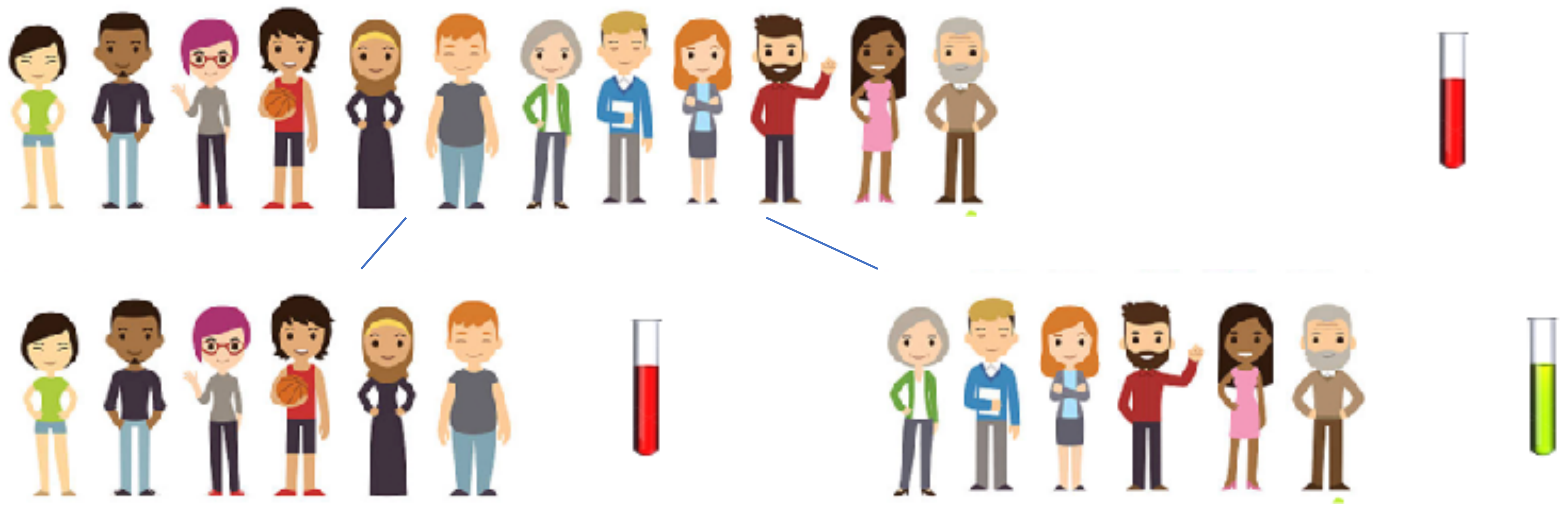
Group testing: needs fewer tests and scales better than individual testing

Adaptive Group Testing

Non-Adaptive Group Testing

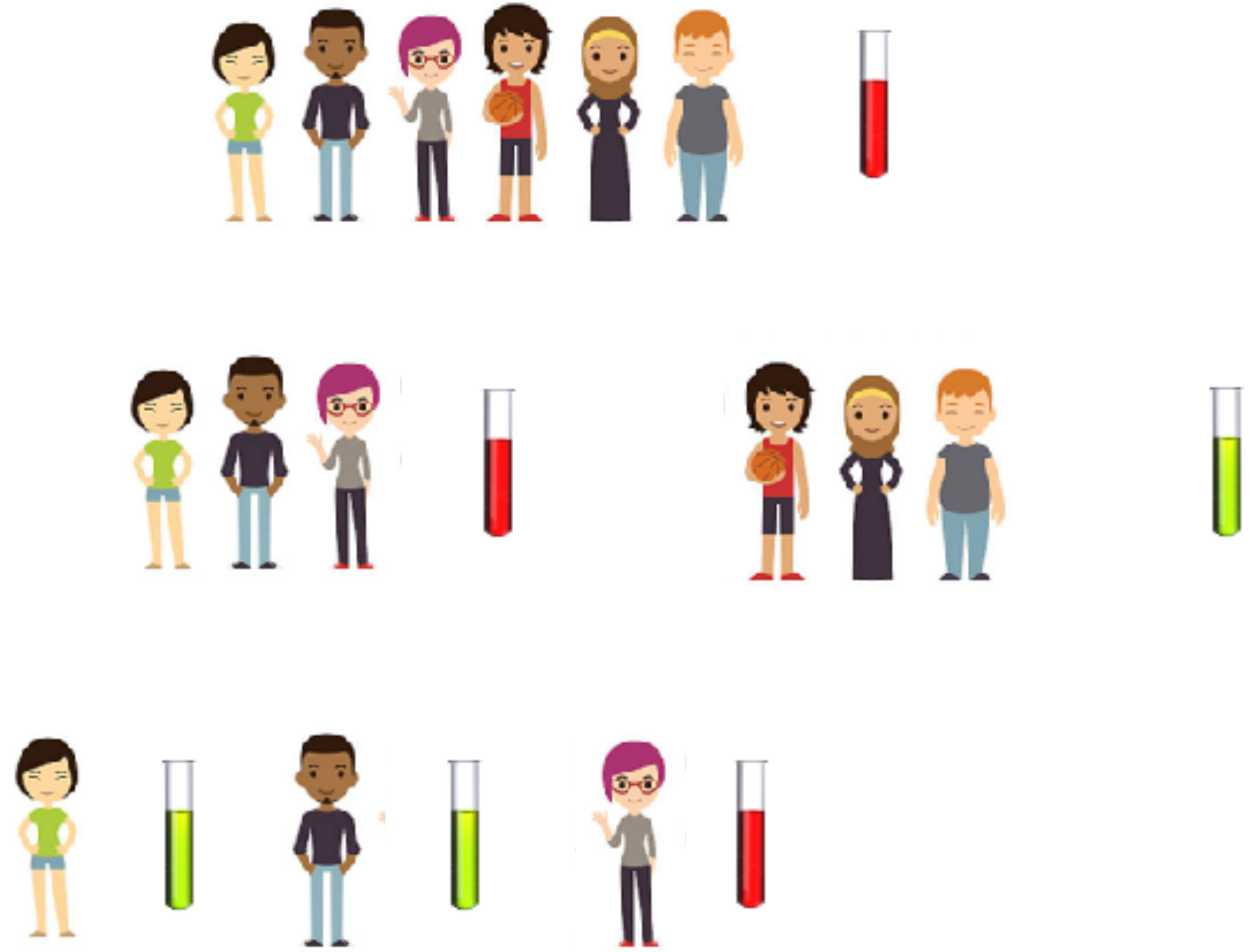
Adaptive Group Testing

Decide what tests to perform next depending on the current test outcome.

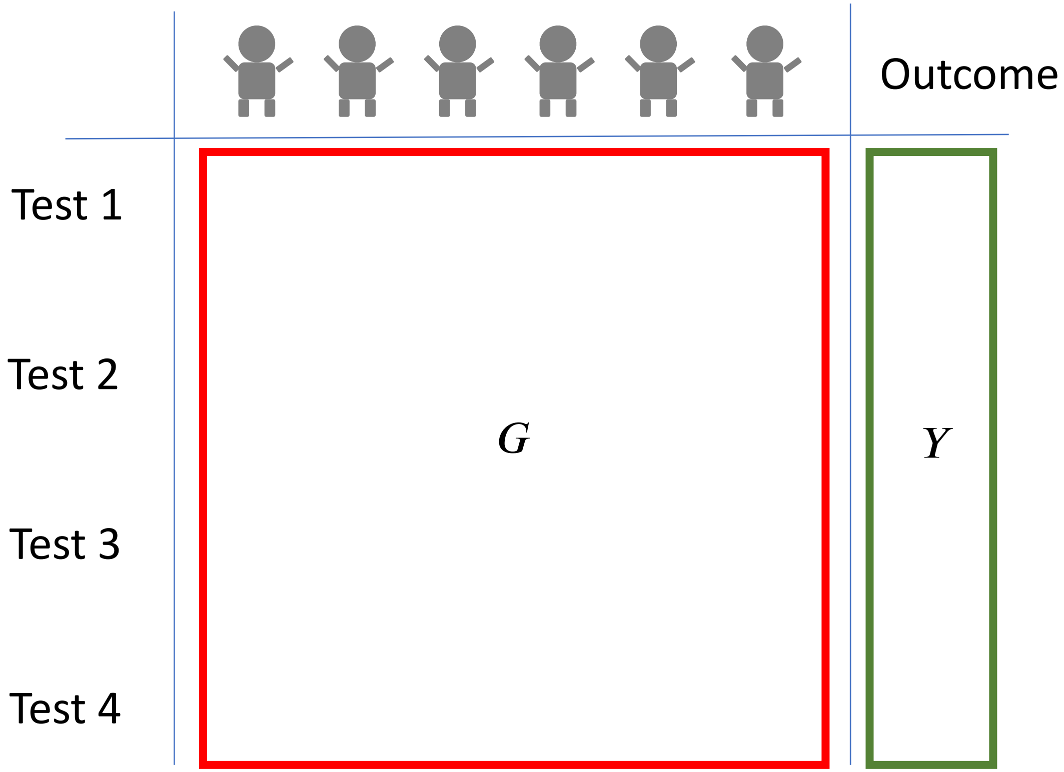


Adaptive Group Testing

Decide what tests to perform next depending on the current test outcome.



Nonadaptive Group Testing



- Goal:

  1) Design test matrix G and 2) an inference (decoding) algorithm such that the probability of error is minimized.

Average probability of error:

$$P(error) = \frac{1}{\binom{N}{K}} \sum_{Y} Pr(\hat{\mathscr{K}} \neq \mathscr{K})$$

$\mathscr{K}$: set of K infected people

$\hat{\mathscr{K}}$: estimated set of K infected people

Lower Bound on Number of Tests Needed

Assume K out of N people are infected uniformly at random. If we want zero error probability, then we need to use a number of tests T such that $T \geq \log_2 \binom{N}{K}$.

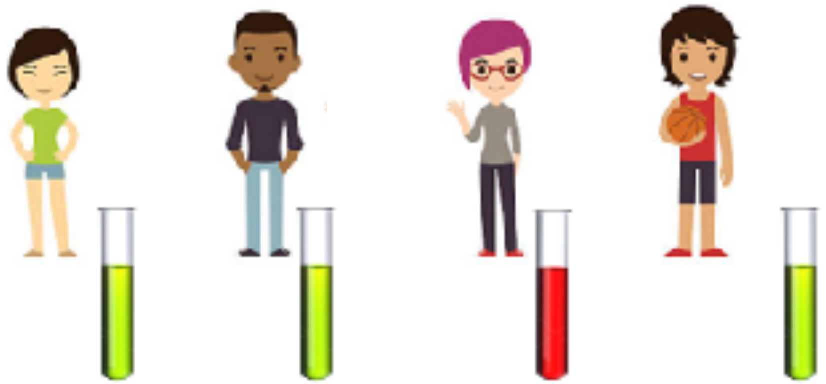
$$\begin{bmatrix} Y_1 \\ \vdots \\ Y_T \end{bmatrix} \qquad 2^T \text{values} \geq \binom{N}{K}$$

$$H(U) = \sum_{i=1}^{N} H(U_i) = \sum_{i=1}^{N} H(p = \frac{K}{N}) = K \log_2(\frac{N}{K}) + O(K)$$

Adaptive Group Testing: Dorfman's procedure

Divide into M subsets
and group test each subset

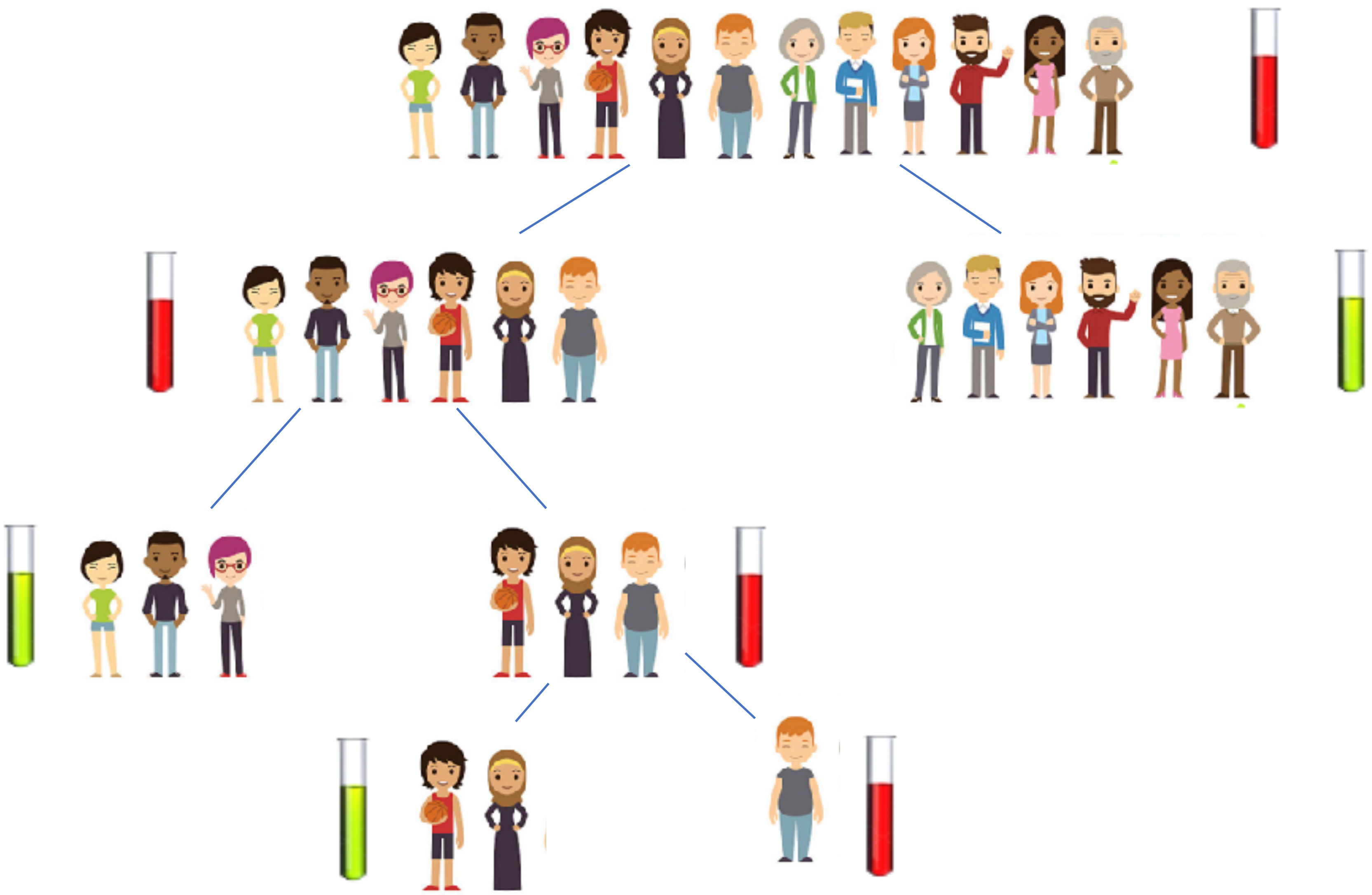
If positive individually test
each person in the subset



$$T(M) = M + K\frac{N}{M}$$

For $M = \sqrt{KN}$, we have that $T(M) = 2\sqrt{KN} << N$

Adaptive Group Testing: Binary splitting



Adaptive Group Testing: Binary splitting

Algorithm 1.1 (Binary splitting)**.** Given a set A:

1. Initialize the algorithm with set A. Perform a single test containing every item in A.
2. If the preceding test is negative, A contains no defective items, and we halt. If the test is positive, continue.
3. If A consists of a single item, then that item is defective, and we halt. Otherwise, pick half of the items in A, and call this set B. Perform a single test of the pool B.
4. If the test is positive, set $A := B$. If the test is negative, set $A := A \setminus B$. Return to Step 3.

Group testing: needs fewer tests and scales better than individual testing

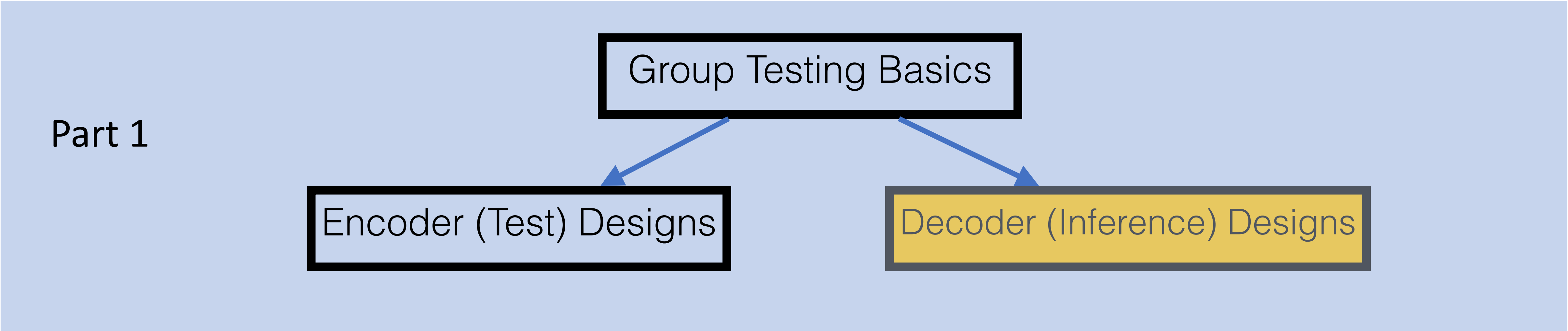
Adaptive Group Testing

order optimal for
$$k = O(N^a), \quad a \in [0,1)$$

Non-Adaptive Group Testing

order optimal for
$$k = O(N^a), \quad a \in [0,0.4)$$

Outline


Part 1
Group Testing Basics
Encoder (Test) Designs
Decoder (Inference) Designs

NonAdaptive Group Testing: Decoding

Assume we are given a test design matrix G and the test outcomes: how do we infer which individuals are positive?

- M. Aldridge, O. Johnson and J. Scarlett Group testing: An Information Theory Perspective, monograph, Now Founrdations and Trends, 2019.

Satisfying Set

Definition 2.1. Consider the noiseless group testing problem with n items, using a test design X and producing outcomes $\mathbf{y}$. A set $\mathcal{L} \subset \{1, 2, \dots, n\}$ is called a *satisfying set* if

- every positive test contains at least one item from $\mathcal{L}$;
- no negative test contains any item from $\mathcal{L}$.

Satisfying Set

?	?	?	?	?	?	?	?	$\mathbf{y}$
1	1	1	1	0	0	0	0	1
0	0	0	0	1	1	1	1	1
1	1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	1
0	0	1	0	1	1	0	1	1
0	0	0	0	1	0	0	0	1

COMP (or DND) Algorithm

Algorithm 2.2. The *COMP* algorithm is defined as follows. We call any item in a negative test *definitely nondefective* (*DND*), and call the remaining items *possibly defective* (*PD*). Then the COMP algorithm outputs $\hat{\mathcal{K}}_{\mathrm{COMP}}$ equalling the set of possible defectives.

DND | PD

1	2	3	4	5	6	7	8	y
1	1	1	1	0	0	0	0	1
0	0	0	0	1	1	1	1	1
1	1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	1
0	0	1	0	1	1	0	1	1
0	0	0	0	1	0	0	0	1

$$\hat{\mathcal{K}} = \{3,4,5,6,7,8\}$$

Definite Defective (DD) Algorithm

Algorithm 2.3. The *definite defectives* (*DD*) algorithm is defined as follows.

1. We say that any item in a negative test is *definitely nondefective* (*DND*), and that any remaining item is a *possible defective* (*PD*).
2. If any PD item is the only PD item in a positive test, we call that item *definitely defective* (DD).
3. The DD algorithm outputs $\hat{\mathcal{K}}_{\mathrm{DD}}$, the set of definitely defective items.

DND		DD		DD	PD			
1	2	3	4	5	6	7	8	**y**
1	1	1	1	0	0	0	0	1
0	0	0	0	1	1	1	1	1
1	1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	1
0	0	1	0	1	1	0	1	1
0	0	0	0	1	0	0	0	1
0	0	0	1	0	1	1	1	1

$$\hat{\mathcal{K}} = \{3,5\}$$

SCOMP Algorithm

Algorithm 2.4. The *SCOMP* algorithm is defined as follows.

1. Initialize $\hat{\mathcal{K}}$ as the estimate $\hat{\mathcal{K}}_{\mathrm{DD}}$ produced by the DD algorithm (Algorithm 2.3), and declare any definitely nondefective items (items appearing in a negative test) to be nondefective. The other possible defectives are not yet declared either way.

2. Any positive test is called *unexplained* if it does not contain any items from $\hat{\mathcal{K}}$. Add to $\hat{\mathcal{K}}$ the possible defective not in $\hat{\mathcal{K}}$ that is in the most unexplained tests, and mark the corresponding tests as no longer unexplained. (Ties may be broken arbitrarily.)

DND	DND	DD		DD				
1	2	3	4	5	6	7	8	y
1	1	1	1	0	0	0	0	1
0	0	0	0	1	1	1	1	1
1	1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	1
0	0	1	0	1	1	0	1	1
0	0	0	0	1	0	0	0	1
0	0	0	1	0	1	1	1	1

Unexplained

$$\hat{\mathcal{K}} = \{3,5,7\}$$

Smallest Satisfying Set = Set Cover Problem

Definition 2.1. Consider the noiseless group testing problem with n items, using a test design X and producing outcomes $\mathbf{y}$. A set $\mathcal{L} \subset \{1, 2, \ldots, n\}$ is called a *satisfying set* if

- every positive test contains at least one item from $\mathcal{L}$;
- no negative test contains any item from $\mathcal{L}$.

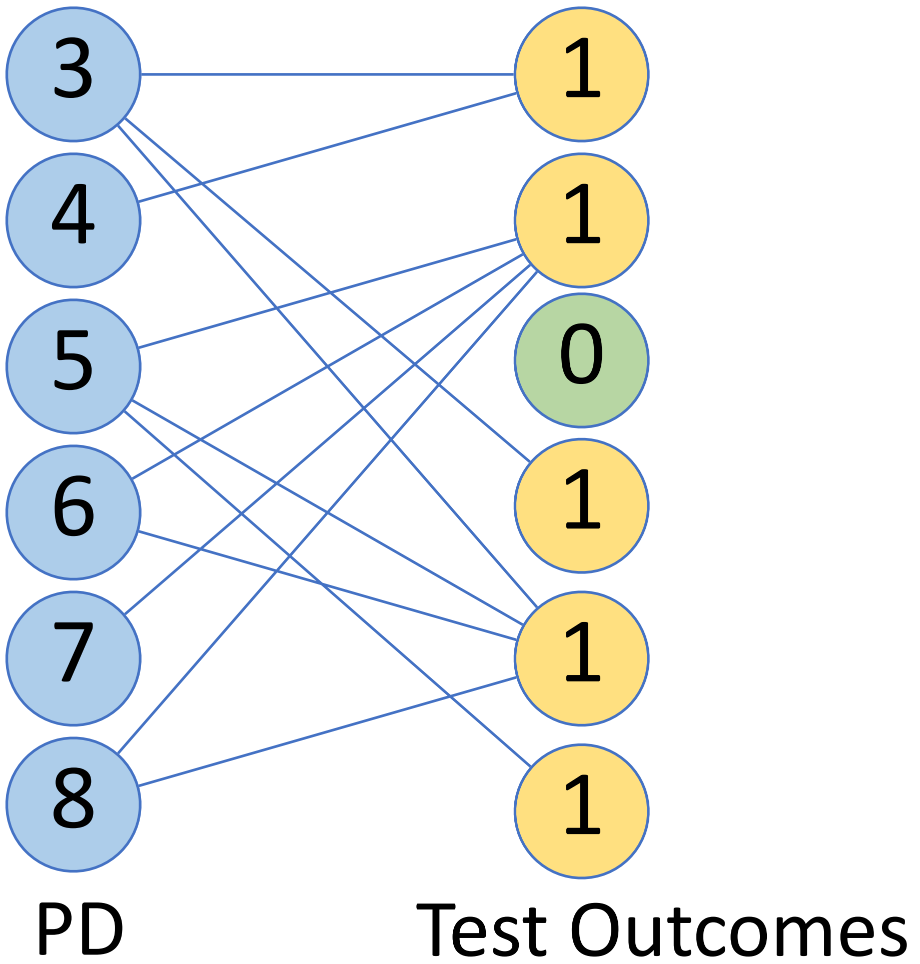
?	?	3	4	5	6	7	8	**y**
1	1	1	1	0	0	0	0	1
0	0	0	0	1	1	1	1	1
1	1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	1
0	0	1	0	1	1	0	1	1
0	0	0	0	1	0	0	0	1

Smallest Satisfying Set = Set Cover Problem

Definition 2.1. Consider the noiseless group testing problem with n items, using a test design X and producing outcomes $\mathbf{y}$. A set $\mathcal{L} \subset \{1, 2, \ldots, n\}$ is called a *satisfying set* if

- every positive test contains at least one item from $\mathcal{L}$;
- no negative test contains any item from $\mathcal{L}$.

?	?	3	4	5	6	7	8	$\mathbf{y}$
1	1	1	1	0	0	0	0	1
0	0	0	0	1	1	1	1	1
1	1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	1
0	0	1	0	1	1	0	1	1
0	0	0	0	1	0	0	0	1

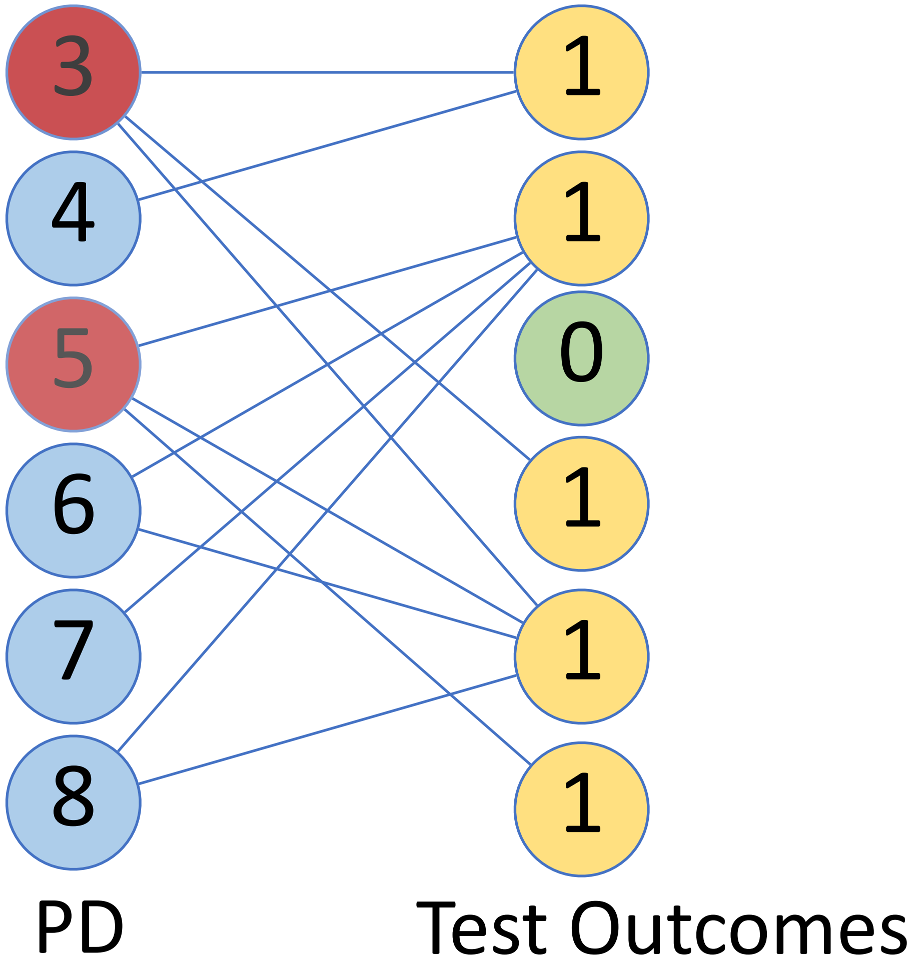


Smallest Satisfying Set = Set Cover Problem

Definition 2.1. Consider the noiseless group testing problem with n items, using a test design X and producing outcomes $\mathbf{y}$. A set $\mathcal{L} \subset \{1, 2, \ldots, n\}$ is called a *satisfying set* if

- every positive test contains at least one item from $\mathcal{L}$;
- no negative test contains any item from $\mathcal{L}$.

?	?	3	4	5	6	7	8	$\mathbf{y}$
1	1	1	1	0	0	0	0	1
0	0	0	0	1	1	1	1	1
1	1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	1
0	0	1	0	1	1	0	1	1
0	0	0	0	1	0	0	0	1

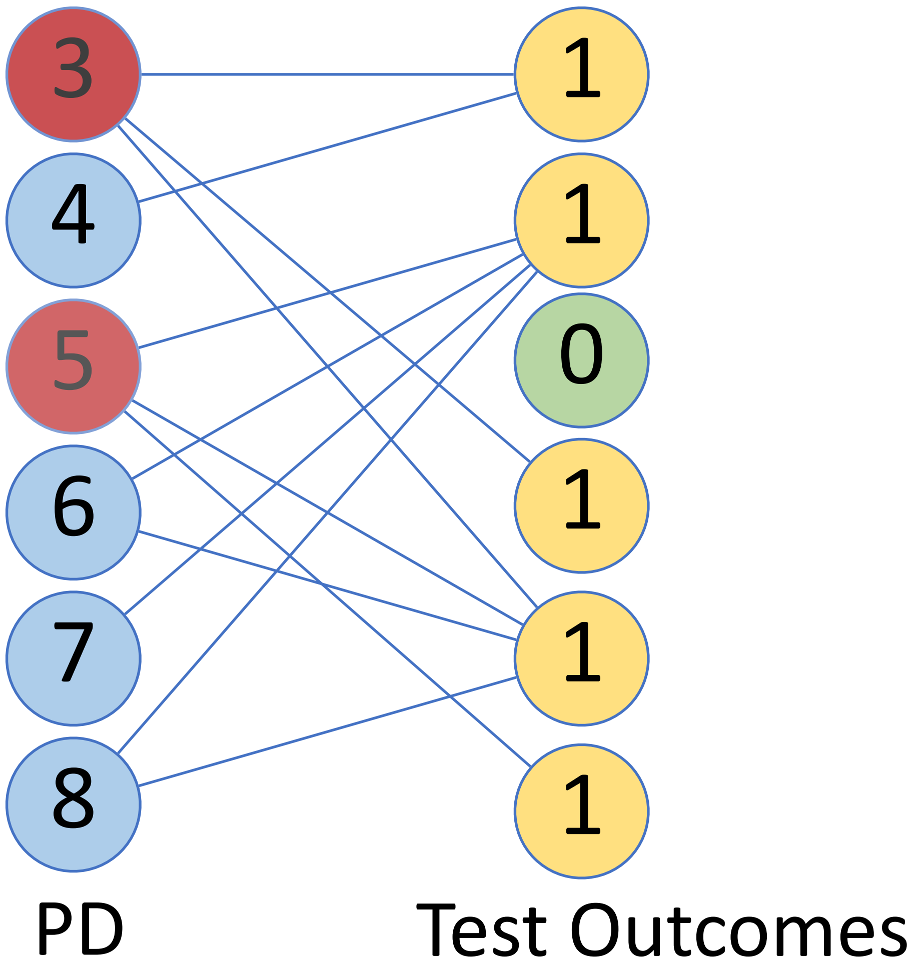


Smallest Satisfying Set = Set Cover Problem

Definition 2.1. Consider the noiseless group testing problem with n items, using a test design X and producing outcomes $\mathbf{y}$. A set $\mathcal{L} \subset \{1, 2, \ldots, n\}$ is called a *satisfying set* if

- every positive test contains at least one item from $\mathcal{L}$;

								Outcome
1	1	1	1	0	0	0	0	Positive
0	0	0	0	1	1	1	1	Positive
1	1	0	0	0	0	0	0	Negative
0	0	1	0	0	0	0	0	Positive
0	0	1	0	1	1	0	0	Positive
0	0	0	0	1	0	0	0	Positive



Integer Programming Decoding

$$\min_{z} \sum_{i=1}^{N} z_i$$

$$\text{subject to} \sum_{i=1}^{N} G_{ti} z_i \geq 1 \text{ when } Y_t = 1$$

$$\sum_{i=1}^{N} G_{ti} z_i = 0 \text{ when } Y_t = 0$$

$$z_i \in \{0,1\}$$

- Population of N individuals
- $U = (U_1, U_2, \ldots, U_N)$, infection vector $U_i = 1$ if individual i is infected, and 0 otherwise
- $Y = (Y_1, Y_2, \ldots, Y_T)$, the vector of test outcomes
- Test matrix G where $G_{ti} = 1$ if individual i is in test t

$$\begin{bmatrix} Y_1 \\ \vdots \\ Y_T \end{bmatrix} = \begin{bmatrix} G_{11} & \cdots & G_{1N} \\ \vdots & \ddots & \vdots \\ G_{T1} & \cdots & G_{TN} \end{bmatrix} \begin{bmatrix} U_1 \\ \vdots \\ U_N \end{bmatrix}$$

Linear Programming Relaxation

$$\min_{z} \sum_{i=1}^{N} z_i$$

$$\text{subject to} \sum_{i=1}^{N} G_{ti} z_i \geq 1 \text{ when } Y_t = 1$$

$$\sum_{i=1}^{N} G_{ti} z_i = 0 \text{ when } Y_t = 0$$

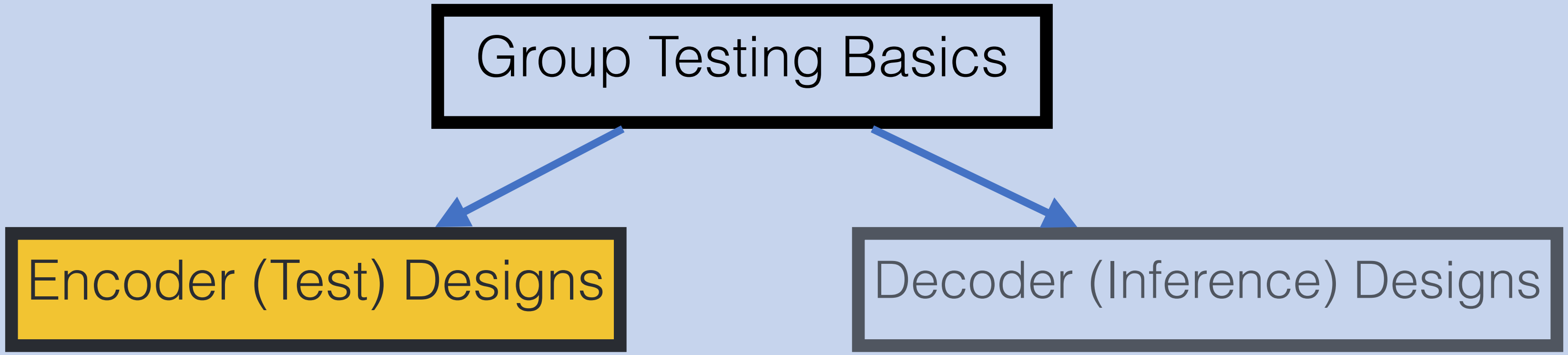
$$0 \leq z_i \leq 1$$

- Population of N individuals
- $U = (U_1, U_2, \ldots, U_N)$, infection vector $U_i = 1$ if individual i is infected, and 0 otherwise
- $Y = (Y_1, Y_2, \ldots, Y_T)$, the vector of test outcomes
- Test matrix G where $G_{ti} = 1$ if individual i is in test t

$$\begin{bmatrix} Y_1 \\ \vdots \\ Y_T \end{bmatrix} = \begin{bmatrix} G_{11} & \cdots & G_{1N} \\ \vdots & \ddots & \vdots \\ G_{T1} & \cdots & G_{TN} \end{bmatrix} \begin{bmatrix} U_1 \\ \vdots \\ U_N \end{bmatrix}$$

Outline

Part 1



NonAdaptive Group Testing: Encoder Designs

Zero Error Deterministic Matrix Designs

A matrix G is called K-separable if, each subset L of K columns, has a distinct support union (for all possible such subsets)

$$\begin{bmatrix} G_{11} & \cdots & G_{1N} \\ \vdots & \ddots & \vdots \\ G_{T1} & \cdots & G_{TN} \end{bmatrix}$$

NonAdaptive Group Testing: Encoder Designs

Zero Error Deterministic Matrix Designs

A matrix G is called K-disjunct if, each subset L of K columns, and for any column i that does not belong in L, the support of column i is not a subset of the support of the set of columns L.

$$\begin{bmatrix} G_{11} & \cdots & G_{1N} \\ \vdots & \ddots & \vdots \\ G_{T1} & \cdots & G_{TN} \end{bmatrix}$$

NonAdaptive Group Testing: Encoder Designs

Zero Error Deterministic Matrix Designs

A matrix G is called K-separable if, each subset L of K columns, has a distinct support union (for all possible such subsets)

A matrix G is called K-disjunct if, each subset L of K columns, and for any column i that does not belong in L, the support of column i is not a subset of the support of the set of columns L.

Construction of such matrices that lead to zero error non-adaptive designs need to use $\min\{\Omega(K^2), N\}$ tests

NonAdaptive Group Testing: Encoder Designs

Bernoulli Matrix Designs

$$\begin{bmatrix} G_{11} & \cdots & G_{1N} \\ \vdots & \ddots & \vdots \\ G_{T1} & \cdots & G_{TN} \end{bmatrix}$$

T x N matrix

$$G_{ij} = \begin{cases} 1 & w.p. \quad \frac{\nu}{K} \\ 0 & w.p. \quad \text{otherwise} \end{cases}$$

where ν is a constant to select and K number of infected people

Every person participates in $T\dfrac{\nu}{K}$ tests and every test contains on average $N\dfrac{\nu}{K}$ people

Probability(test is negative)=$(1-\dfrac{\nu}{K})^K \approx e^{-\nu}$

For $e^{-\nu} = \dfrac{1}{2}$ we get $\nu = \ln 2$

NonAdaptive Group Testing: Encoder Designs

Constant Column Weight (CCW)
Matrix Designs

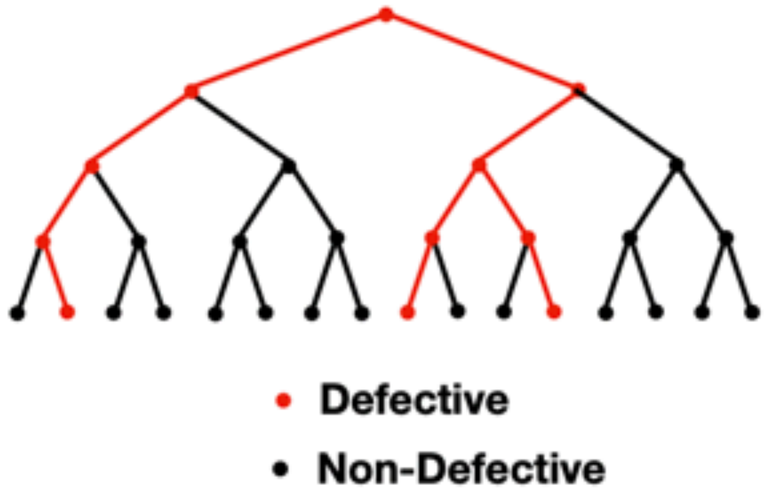
$$\begin{bmatrix} G_{11} & \cdots & G_{1N} \\ \vdots & \ddots & \vdots \\ G_{T1} & \cdots & G_{TN} \end{bmatrix}$$

T x N matrix

Each person is included in a fixed number of L tests (typical choice: $L = \Theta(\frac{T}{K})$)

NonAdaptive Group Testing: Encoder Designs

Mimic Adaptive Test Designs
In this example: K=3 out of N people infected



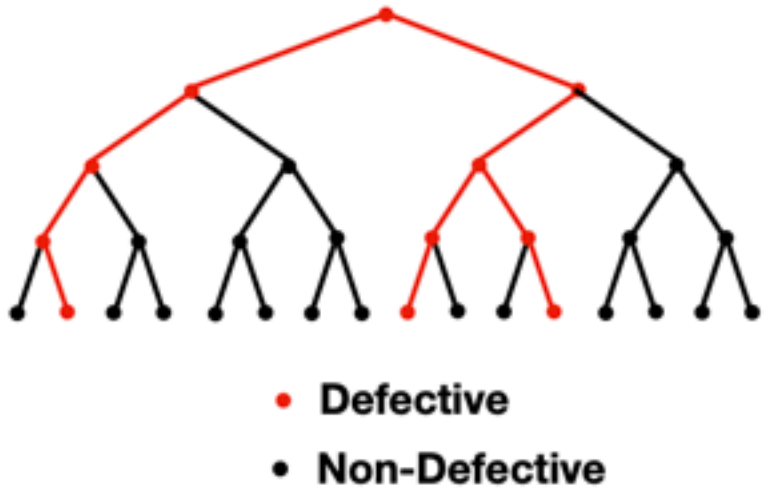
Every level splits the groups in previous levels in half

At each level m we have 2^m groups of size $\frac{N}{2^m}$ each

• E. Price and J. Scarlett, A Fast Binary Splitting Approach to Non-Adaptive Group Testing, arXiv:2006.10268v1, 2020.

NonAdaptive Group Testing: Encoder Designs

Mimic Adaptive Test Designs
In this example: K=3 out of N people infected



Every level splits the groups in previous levels in half

At each level m we have 2^m groups of size $\frac{N}{2^m}$ each

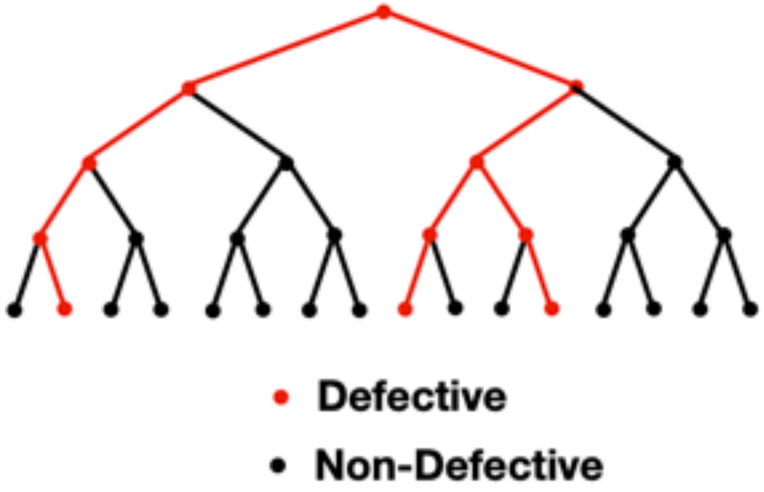
Create a matrix where, for each level m, we have CK tests (C: constant to be decided).
For $j = 1,\ldots,2^m$ place all the members of the group G_j^m in exactly one of the CK tests uniformly at random

- E. Price and J. Scarlett, A Fast Binary Splitting Approach to Non-Adaptive Group Testing, arXiv:2006.10268v1, 2020.

NonAdaptive Group Testing: Encoder Designs

Mimic Adaptive Test Designs
In this example: K=3 out of N people infected

Every level splits the groups
in previous levels in half



- E. Price and J. Scarlett, A Fast Binary Splitting Approach to Non-Adaptive Group Testing, arXiv:2006.10268v1, 2020.

NonAdaptive Group Testing: Non-identical priors

What if people have a different a-priori probability to be infected?

$$U_i = \begin{cases} 1 & w.p. \quad p_i \\ 0 & w.p. \quad 1-p_i \end{cases}$$

- Assumption:

  Independent but non-identical infections

Lower bound:

Let $H(U) = \sum_{i=1}^{N} H(U_i) = \sum_{i=1}^{N} H(p_i)$ then if error probability P_e then $T \geq (1-P_e)H(U)$

- T. Li, C. L. Chan, W. Huang, T. Kaced, and S. Jaggi. Group testing with prior statistics. In IEEE International Symposium on Information Theory (ISIT), pages 2346–2350, 2014. doi: 10.1109/ISIT.2014.6875253.

Proof. The input vector $\mathbf{X}$, noiseless result vector $\mathbf{B}$ and estimated input vector $\mathbf{Y}$ form a Markov chain $\mathbf{X} \to \mathrm{B} \to \mathbf{Y}$. Moreover,

$$H(\mathbf{X}) = H(\mathbf{X}|\mathbf{Y}) + I(\mathbf{X};\mathbf{Y}). \tag{2}$$

Define an error random variable E such that

$$E = \begin{cases} 1, & \text{if } \mathbf{Y} \neq \mathbf{X} \\ 0, & \text{if } \mathbf{Y} = \mathbf{X} \end{cases}.$$

By Fano's inequality, we can bound the conditional entropy as

$$\begin{aligned} H(\mathbf{X}|\mathbf{Y}) &= H(E,\mathbf{X}|\mathbf{Y}) \\ &= H(E|\mathbf{Y}) + \Pr[E=0]\, H(\mathbf{X}|\mathbf{Y}, E=0) + \Pr[E=1]\, H(\mathbf{X}|\mathbf{Y}, E=1) \\ &\leq H(\mathbb{P}_e) + \mathbb{P}_e H(\mathbf{X}). \end{aligned}$$

Also we have $I(\mathbf{X};\mathbf{B}) \leq H(\mathbf{B}) \leq \log_2|\mathbf{B}| = T$ by the data-processing inequality. Hence we obtain that

$$T \geq (1 - \mathbb{P}_e)\, H(\mathbf{X}).$$

- T. Li, C. L. Chan, W. Huang, T. Kaced, and S. Jaggi. Group testing with prior statistics. In IEEE International Symposium on Information Theory (ISIT), pages 2346–2350, 2014. doi: 10.1109/ISIT.2014.6875253.

□

NonAdaptive Group Testing: CCA algorithm

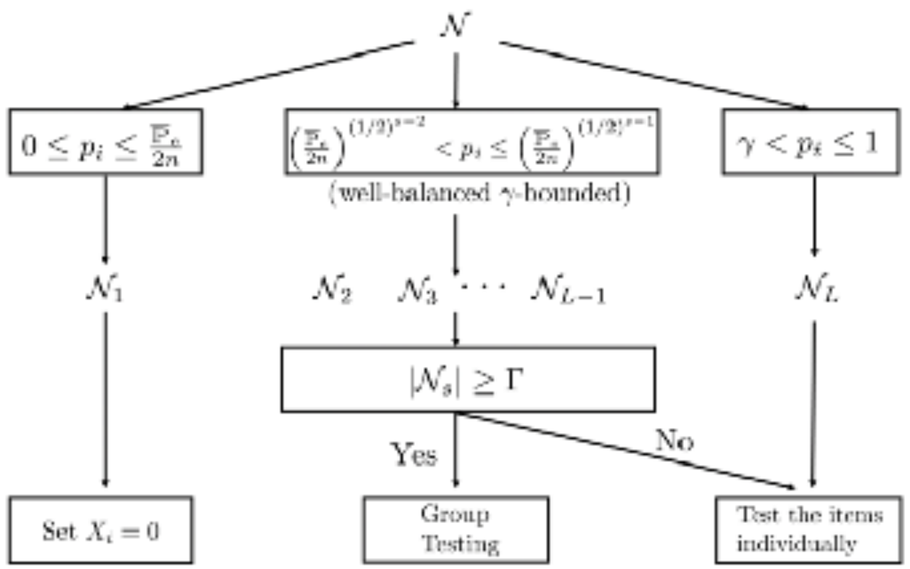
The i-th row of the matrix G is obtained by sampling the probability vector $\hat{p} = (\hat{p}_1, \hat{p}_2, \ldots, \hat{p}_N)$ where $\hat{p}_i = \dfrac{1 - p_i}{N - \mu}$ and $\mu = \sum_{i=1}^{N} p_i \ll N$ for g times with replacement, where g is a constant we optimize.

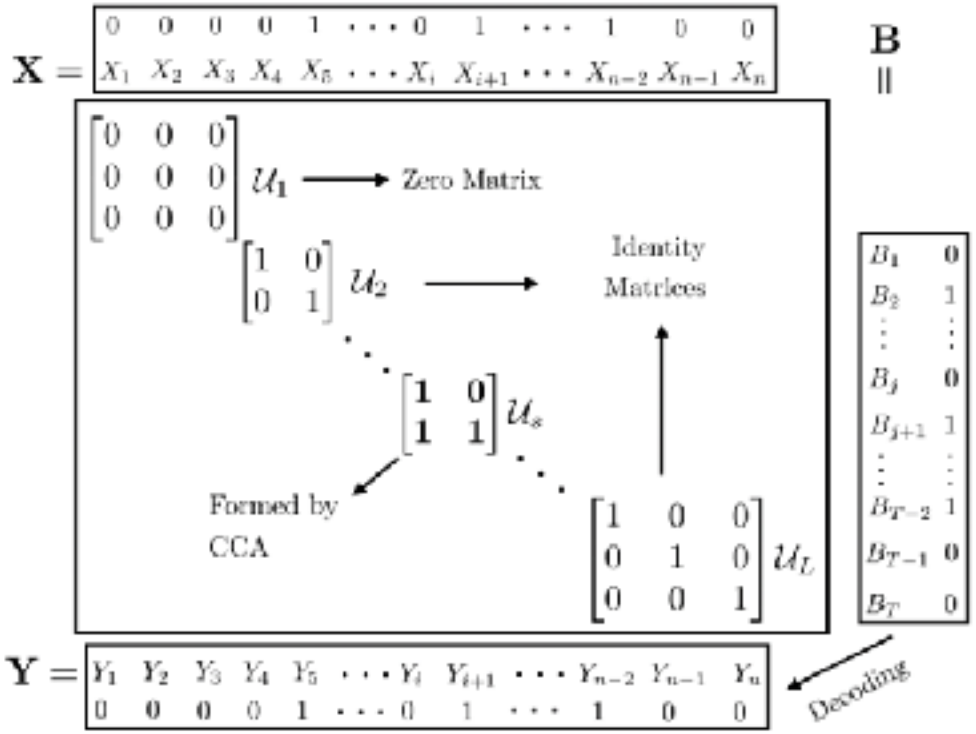
Analysis: algorithm can be viewed as "collecting all the non-defective items in negative tests" (coupon collector)

Testing procedure requires $O(\mu \ln(N))$ tests with high probability

- C. L. Chan, S. Jaggi, V. Saligrama, and S. Agnihotri, "Non-adaptive group testing: Explicit bounds and novel algorithms," in *Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on*. IEEE, 2012, pp. 1837–1841.

NonAdaptive Group Testing: CCA applied in blocks (achieves bound)

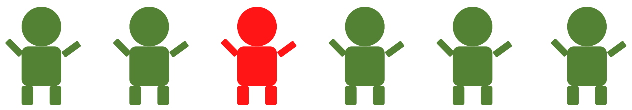




- C. L. Chan, S. Jaggi, V. Saligrama, and S. Agnihotri, "Non-adaptive group testing: Explicit bounds and novel algorithms," in *Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on*. IEEE, 2012, pp. 1837–1841.

Nonadaptive Group Testing as an Optimization Problem

$U = [0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0]$



							Outcome
Test 1	1	0	1	0	1	0	1
Test 2	0	1	0	1	1	0	0
Test 3	0	1	1	0	0	1	1
Test 4	1	0	0	1	0	1	0

Test matrix G

Y

- Population of N individuals
- $U = (U_1, U_2, \ldots, U_N)$, infection vector $U_i = 1$ if individual i is infected, and 0 otherwise
- $Y = (Y_1, Y_2, \ldots, Y_T)$, the vector of test outcomes
- Test matrix G where $G_{ti} = 1$ if individual i is in test t

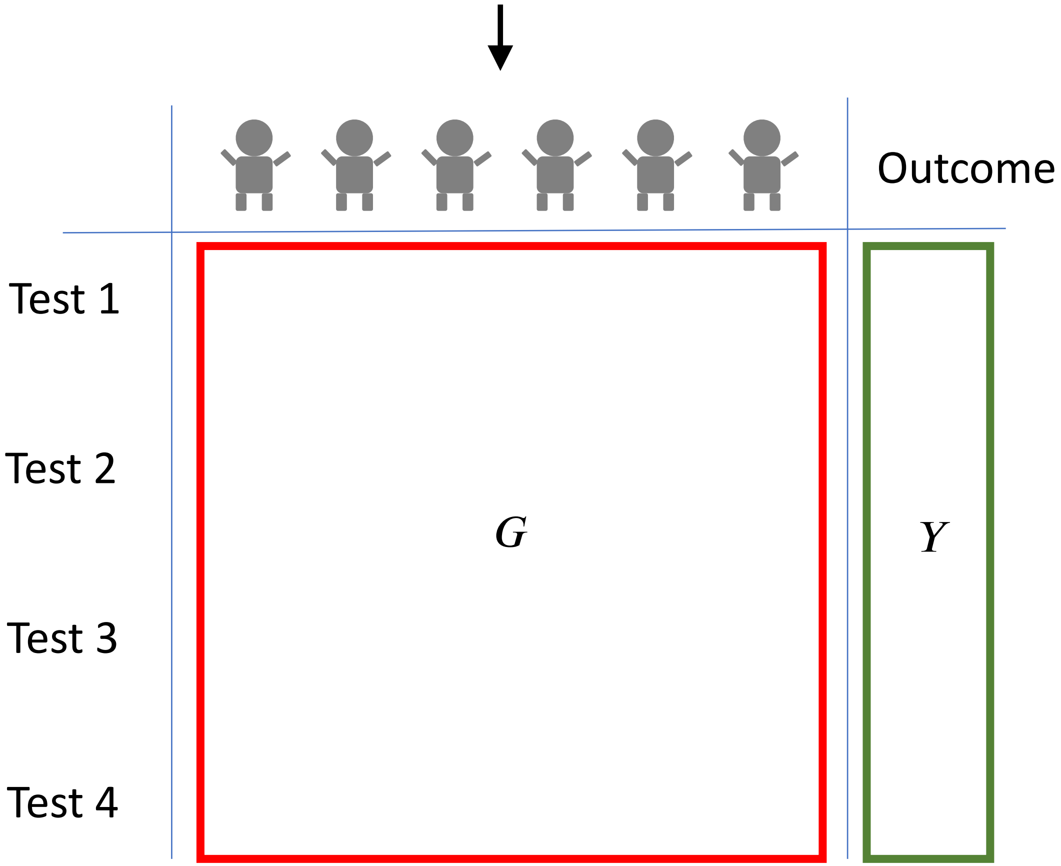
$$\begin{bmatrix} Y_1 \\ \vdots \\ Y_T \end{bmatrix} = \begin{bmatrix} G_{11} & \cdots & G_{1N} \\ \vdots & \ddots & \vdots \\ G_{T1} & \cdots & G_{TN} \end{bmatrix} \begin{bmatrix} U_1 \\ \vdots \\ U_N \end{bmatrix}$$

$$Y = GU$$

Matrix multiplication in Boolean algebra

Optimal test matrix problem

$$U_i = \begin{cases} 1 & w.p. \quad p_i \\ 0 & w.p. \quad 1 - p_i \end{cases}$$



- Assumption:

  Independent but non-identical infections

- Goal:

  Design test matrix G s.t. under the DND decoder, the expected number of misclassifications (false positives) is minimized

Optimal test matrix: optimization formulation

Expected number of false positives under DND decoder:

$$Err(G) = \sum_{i=1}^{N} (1 - p_i)\mathbb{E} \prod_{t \ : i \in \delta_t} \gamma_{t,i}$$

$$\boxed{\underset{G \in \{0,1\}^{T \times N}}{\arg\min} \quad Err(G)}$$

$\gamma_{t,i} = 1$ if test t is positive given that individual i is healthy

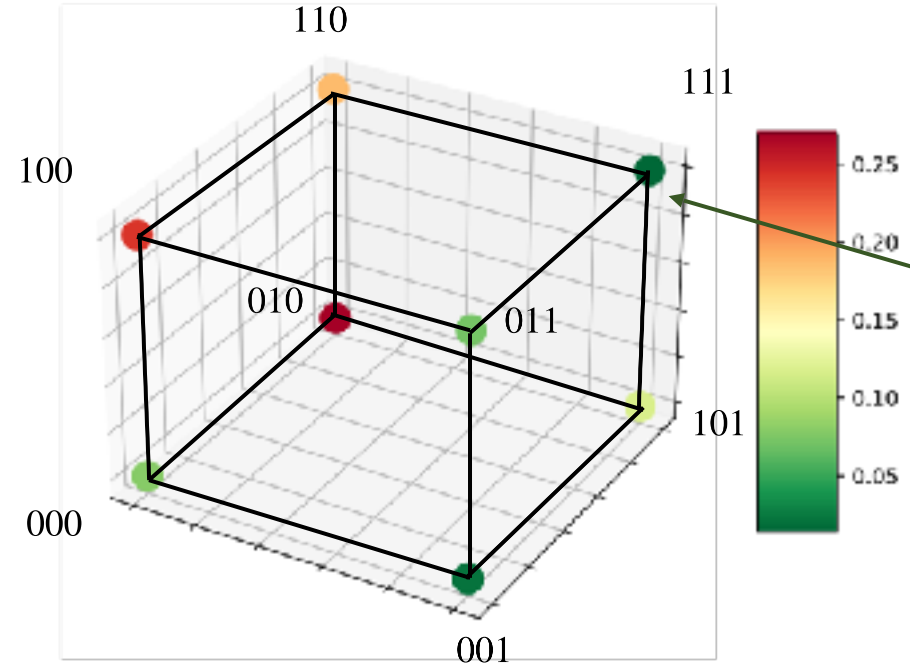
$\delta_t = \{j : G_{tj} = 1\}$: set of individuals who participate in test t

- Prior works: Study how many tests T needed to recover U w.h.p., decoders
- We optimize the test matrix under the constraint of available tests

Detour: combinatorial optimization perspective

$$\underset{G \in \{0,1\}^{T\times N}}{\arg\min} \quad Err(G)$$

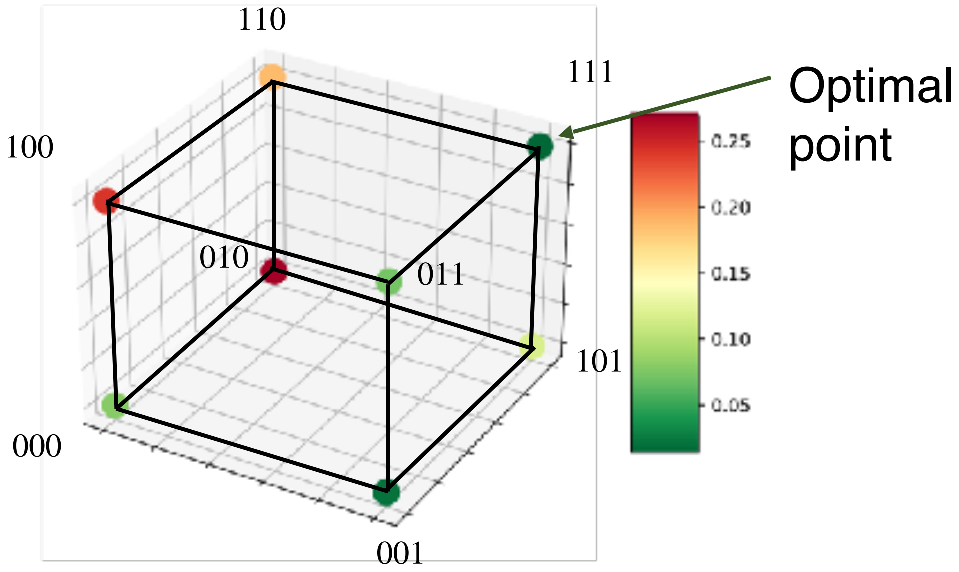
$$\underset{x \in \{0,1\}^{n}}{\operatorname{argmin}} \quad f(x)$$

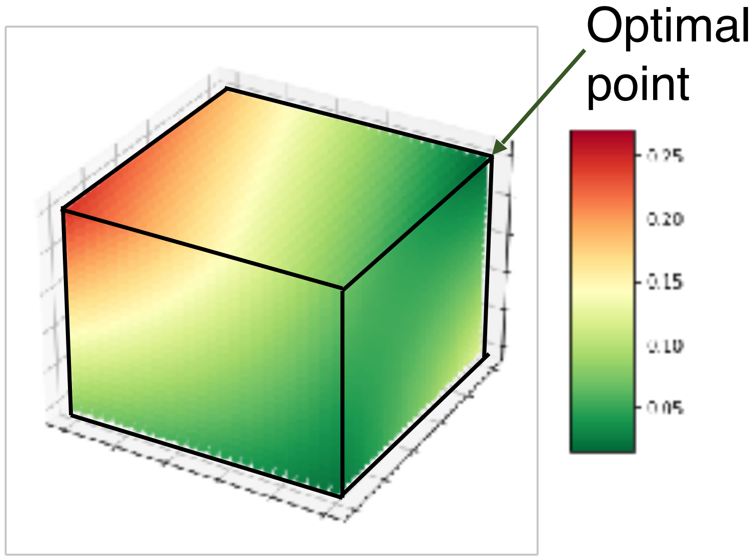


Optimal point

Detour: combinatorial optimization relaxation

$$\underset{x \in \{0,1\}^n}{\operatorname{argmin}} \; f(x) \quad \longleftrightarrow \quad \underset{q \in [0,1]^n}{\operatorname{argmin}} \; g(q)$$





Detour: combinatorial optimization relaxation

$$\underset{x \in \{0,1\}^n}{\operatorname{argmin}} \; f(x) \quad \longleftrightarrow \quad \underset{q \in [0,1]^n}{\operatorname{argmin}} \; g(q)$$

Desirable properties of g

1. $g(q) \geq f^*$ forall $q \in [0,1]^n$: g is at least as large as the minimum value of f
2. $g(x) = f(x)$ forall $x \in \{0,1\}^n$: g is equal to f on the vertices of the hypercube
3. Given a minimizer q^* of g, if one can find $x' \in \{0,1\}^n$ such that $f(x') = g(q^*)$: this implies x' is a minimizer of f

$g^* = f^*$

Detour: combinatorial optimization relaxation

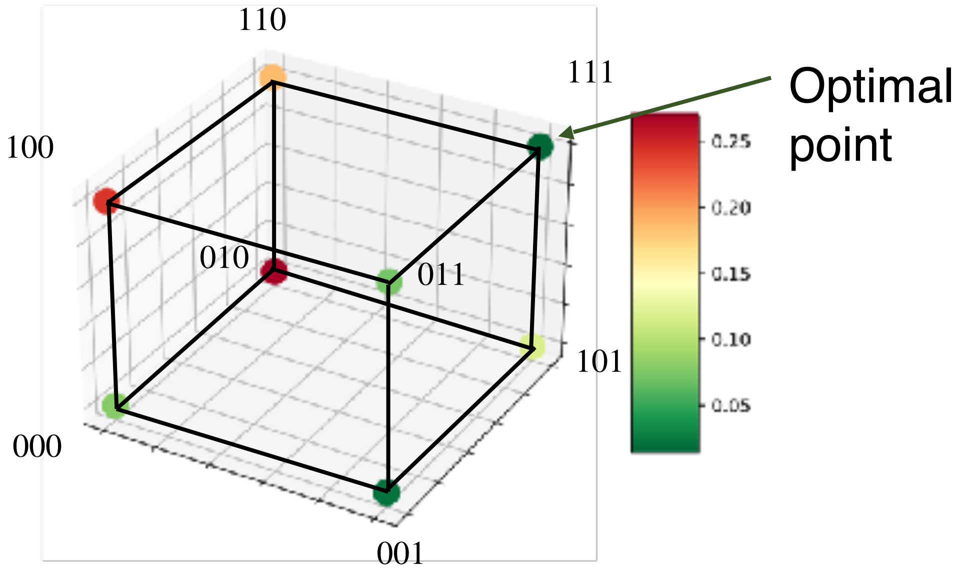
$$\underset{x \in \{0,1\}^n}{\operatorname{argmin}} \; f(x) \qquad \longleftrightarrow \qquad \underset{q \in [0,1]^n}{\operatorname{argmin}} \; g(q)$$

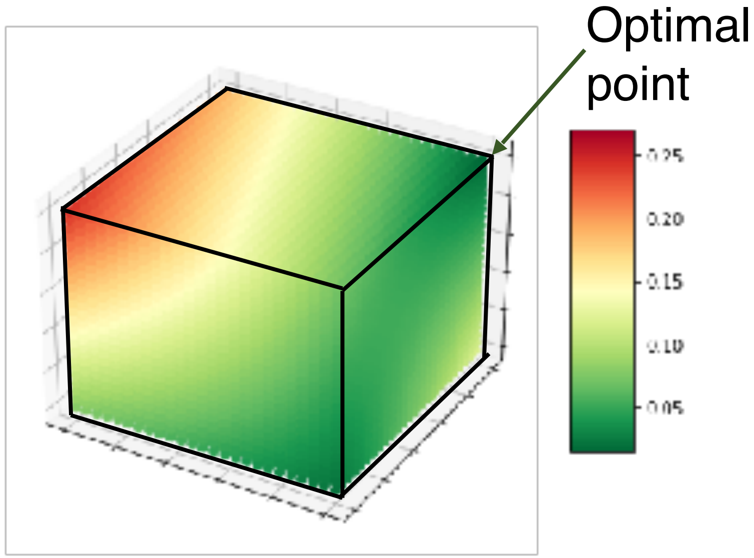
What g has these desirable properties? One choice is to construct g as follows:

- Given $q \in [0,1]^n$, say $q = (q_1, q_2, \ldots, q_n)$
- Sample $Z = (Z_1, Z_2, \ldots, Z_n)$ such that $Z_i \sim Ber(q_i)$: we say that $Z \sim q$
- $g(q) \stackrel{\text{def}}{=} \mathbb{E}_{Z \sim q} f(Z)$

Example

$$\underset{x \in \{0,1\}^n}{\operatorname{argmin}} \; f(x) \quad \Longleftrightarrow \quad \underset{q \in [0,1]^n}{\operatorname{argmin}} \; g(q)$$





$$g\big((q_1, q_2, q_3)\big) = q_1 q_2 q_3 f(111) + q_1 q_2 \left(1 - q_3\right) f(110) + q_1 \left(1 - q_2\right) q_3 f(101) + q_1 \left(1 - q_2\right)\left(1 - q_3\right) f(100)$$
$$+ (1 - q_1) q_2 q_3 f(011) + (1 - q_1) q_2 \left(1 - q_3\right) f(010) + (1 - q_1)\left(1 - q_2\right) q_3 f(001) + (1 - q_1)\left(1 - q_2\right)\left(1 - q_3\right) f(000)$$

Detour: combinatorial optimization relaxation

Theorem: Sufficient to solve P2 to obtain a solution of P1.

$$\underset{x \in \{0,1\}^n}{\operatorname{argmin}} \; f(x) \qquad \text{P1}$$

$$\underset{q \in [0,1]^n}{\operatorname{argmin}} \; g(q) \qquad \text{P2}$$

where $g(q) \stackrel{\text{def}}{=} \mathbb{E}_{Z \sim q}\, f(Z)$

Proof. Satisfies the desirable properties

1. $g(q) \geq f^*$ forall $q \in [0,1]^n$: g is at least as large as the minimum value of f

   Expectation of a r.v. is at least as large as its minimum over the support

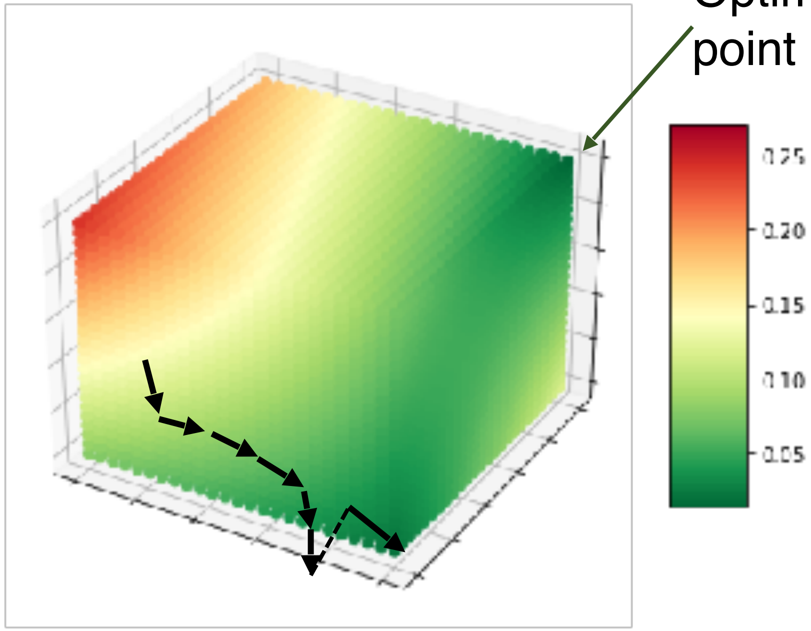
2. $g(x) = f(x)$ forall $x \in \{0,1\}^n$: g is equal to f on the vertices of the hypercube

   At a vertex x, $Z = x$ with probability 1. Thus, $g(x) = f(x)$

3. Given a minimizer q^* of g, sample any $Z \sim q^* \Rightarrow Z$ is a minimizer of f

Relaxed optimization problem

$$\underset{q \in [0,1]^n}{\text{argmin}} \; g(q)$$



Optimal point

Projected gradient descent heuristic

- Start with an initialization for q
- Perform gradient descent until convergence
- In case you go out of the hypercube, project back onto the hypercube and continue
- Finally sample Z from resulting q and estimate that to be a minimizer of f

Cons:

- $g(q)$ and/or its gradient may not be easy to compute
- No guarantee of converging to the global minimum

Optimal test matrix: relaxed optimization formulation

$$\underset{G \in \{0,1\}^{T\times N}}{\arg\min} \quad Err(G)$$

$$Err(G) = \sum_{i=1}^{N} (1 - p_i) \mathbb{E} \prod_{t\ :\ i \in \delta_t} \gamma_{t,i}$$

$$Err(G) = \sum_{i=1}^{N} (1 - p_i) \mathbb{E} \prod_{t=1}^{T} (1 - G_{ti} \prod_{\substack{j=1: \\ j \neq i}}^{N} (1 - G_{tj} U_j))$$

$$\underset{Q \in [0,1]^{T\times N}}{\arg\min} \quad \mathbb{E}_{Z \sim Q} \ Err(Z)$$

$$\underset{Q \in [0,1]^{T\times N}}{\arg\min} \quad \mathbb{E}_{Z \sim Q} \sum_{i=1}^{N} (1 - p_i) \mathbb{E}_U \prod_{t\ :\ i \in \delta_t} \gamma_{t,i}$$

Not clear if this is easy to compute

Optimal test matrix: lower bound

$$\underset{Q \in [0,1]^{T\times N}}{\arg\min} \quad \mathbb{E}_{Z\sim Q} \; Err(Z)$$

Mean-field approximation

$$\geq \prod_{t\ :\ i\in\delta_t} \mathbb{E}_U \, \gamma_{t,i}$$

$$\underset{Q \in [0,1]^{T\times N}}{\arg\min} \quad \mathbb{E}_{Z\sim Q} \sum_{i=1}^{N} (1-p_i)\, \mathbb{E}_U \boxed{\prod_{t\ :\ i\in\delta_t} \gamma_{t,i}}$$

- We show that such an approximation is a lower bound using Fortuin–Kasteleyn–Ginibre (FKG) inequality
- Empirically seen to be a ***very good approximation***

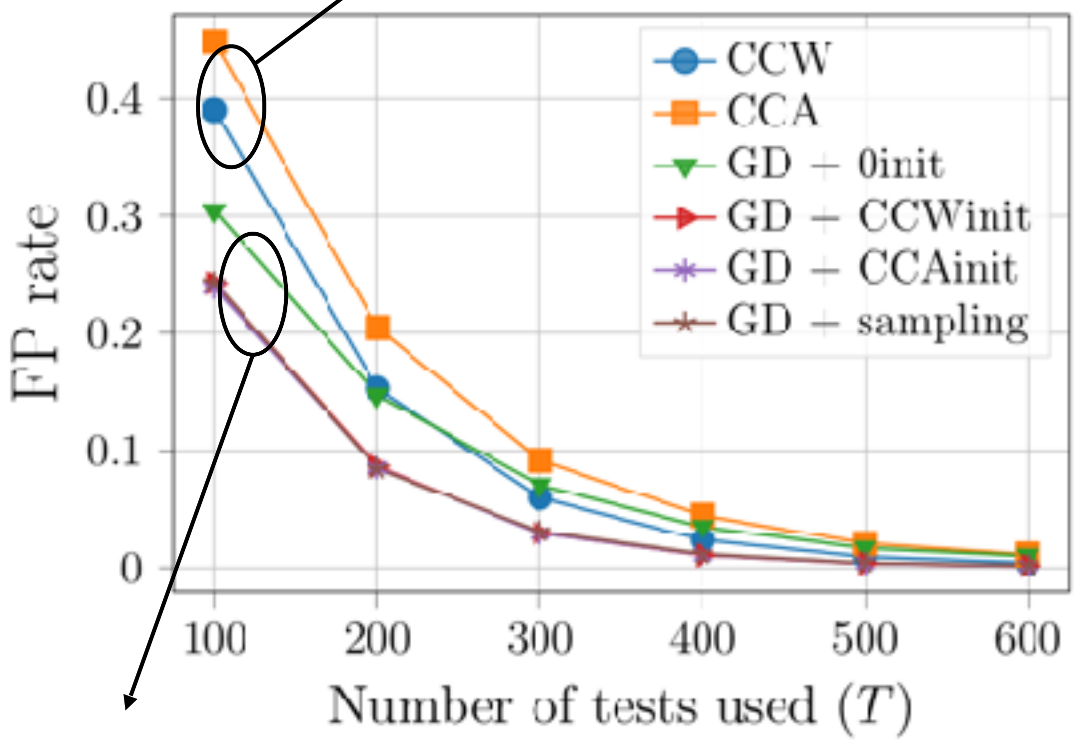
Optimal test matrix: approximate formulation

$$\underset{Q \in [0,1]^{T\times N}}{\arg\min} \quad g(Q) \quad \Leftarrow \quad \underset{Q \in [0,1]^{T\times N}}{\arg\min} \quad g_{LB}(Q)$$

- $g_{LB}(Q)$ and its gradient can be computed in $O(N^2)$ —> GD can be used
- Initialization method:
  - use state-of-the-art test designs to initialize Q
  - refine existing designs
- Alternative:
  - use gradient descent with stochastic reinitialization

Numerical results

Prior test designs



GD based algorithms

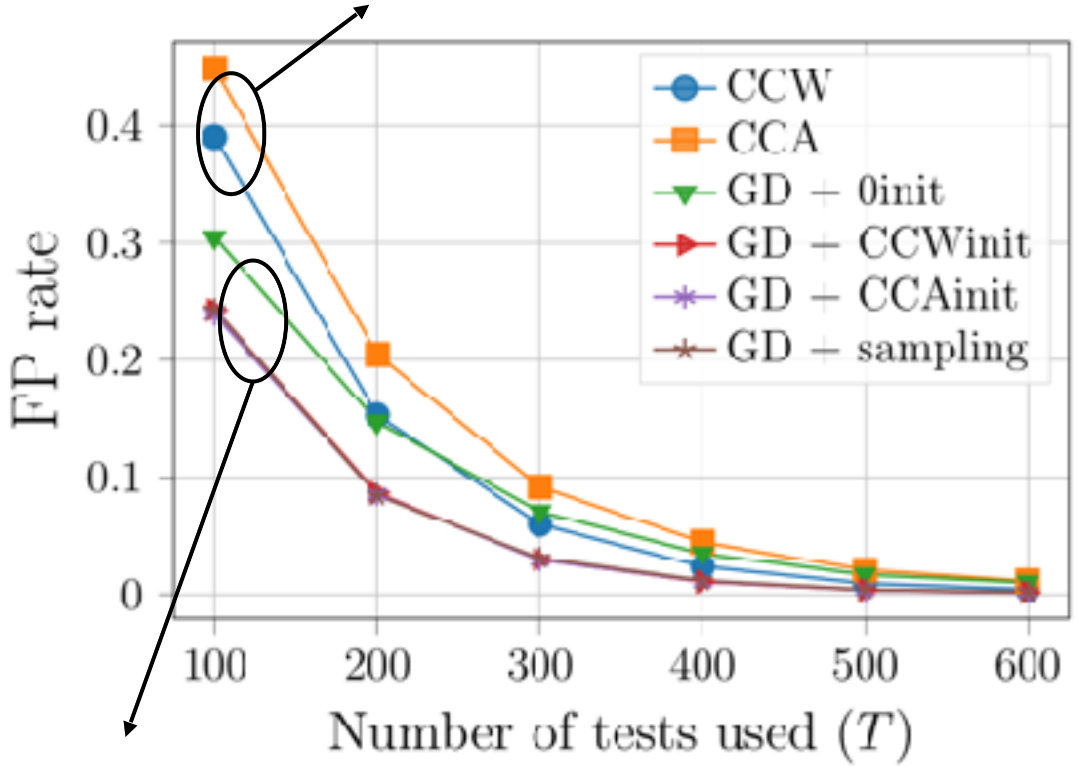
Setup

- Priors sampled from an exponential distribution with mean 0.05, N = 1000
- Metric: average FP rate over 10 instances

Results

- Significant ***benefits*** when the number of tests is limited
- Bonus: similar results with ***DD*** decoder

Prior test designs



GD based algorithms

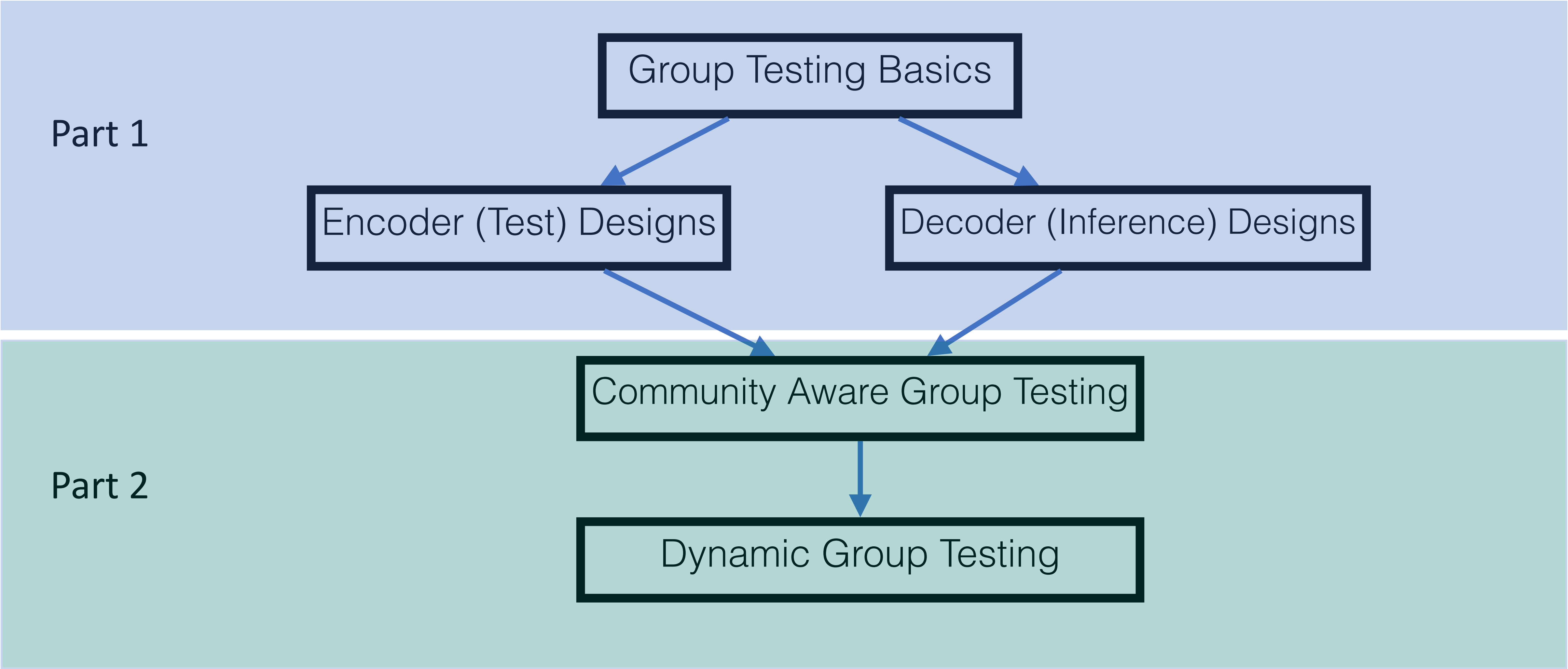
Setup

- Priors sampled from an exponential distribution with mean 0.05, N = 1000
- Metric: average FP rate over 10 instances

Results

- Significant ***benefits*** when the number of tests is limited
- Bonus: similar results with ***DD*** decoder

Outline



Part 1
Group Testing Basics
Encoder (Test) Designs
Decoder (Inference) Designs
Part 2
Community Aware Group Testing
Dynamic Group Testing