BRIDGEi2i
INFORMATION→INSIGHT→IMPACT

Whitepaper

Pooled Sample Pandemic Testing: Optimal Strategy Determination using BRIDGEi2i Optimizer™

Author:
Joydeep Dam
Vice President - AI & Innovation Labs

Background

One of the key strategies to arrest the spread of COVID-19, as proposed by many experts, is to test a significantly higher number of cases. By conducting more tests, authorities will be able to get a clearer picture of the actual level of the infection in the population, identify potential sources of further infections, locate virus hotspots, and gather a lot more useful information. These can be enormously useful in designing actionable strategies to contain the pandemic and eventually eliminating the virus.

However, there is a major weakness in this plan. The number of testing kits is very low compared to the size of the population. Thus, testing kits are an extremely scarce resource. Any testing has to be considered very thoroughly and is usually done only when there are enough external symptoms visible to suggest the presence of the virus.
One major side effect of this is that asymptomatic carriers remain untested, but it's precisely why the undetected cases pose a higher risk.

A unique and innovative solution that has been considered and proposed to tackle this problem is that of "pooled testing."
The concept behind this is simple – if a large number of subjects can be tested together, and the test result turns out to be negative, then the entire group's infection status can be determined to be negative using only one test. The actual designing and implementation of an efficient strategy for pooled testing can, however, be more complex.

History

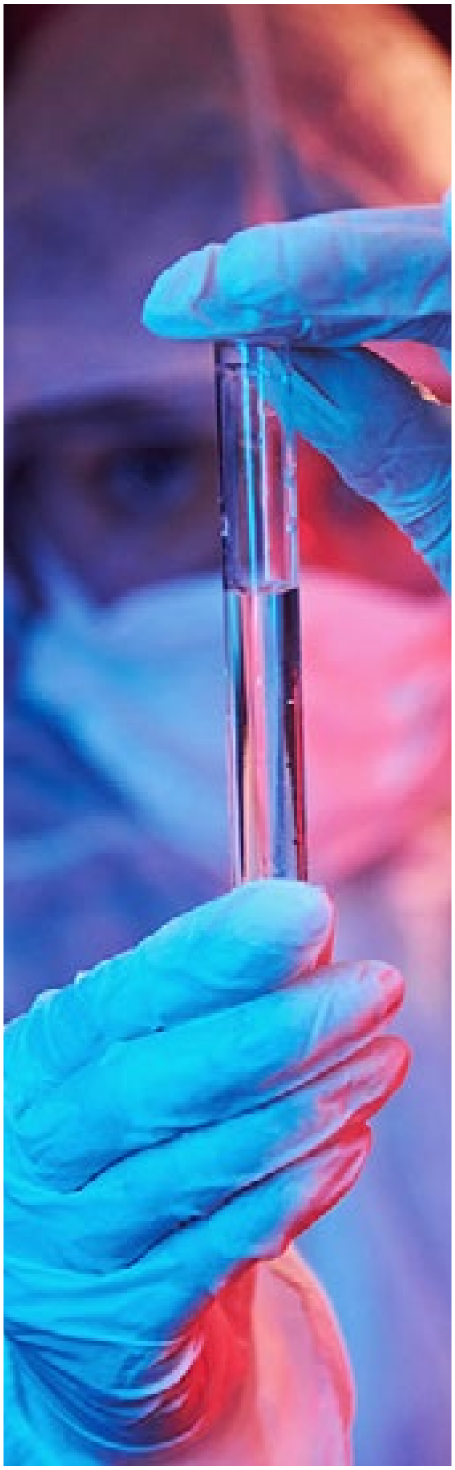
The concept of pooled testing (or group testing as it is also known) has been around for quite some time, and there is a rich and deep set of literature covering various different aspects of it.

The concept of group testing was proposed by Robert Dorfman for the first time in 1943, during the second world war. Dorfman's report published in the Annals of Mathematical Statistics focused on the probabilistic problem, to use the idea of pooled testing to reduce the expected number of tests required to detect all syphilitic men in a given pool of soldiers. The method was simple: group the soldiers into smaller pools of a given size, identify the positive groups and test every individual in the positive groups to find which individuals were infected. Dorfman tabulated the optimum group sizes for this trategy against the prevalence rate of defectiveness in the population. Such tables are fairly simple to derive using elementary probabilistic computations, as we will see later in the paper.

For the initial few years, the approach towards grouptesting was primarily probabilistic. Later on, other approaches such as combinatorial group testing and non-adaptive testing also drew interest, and these techniques have been studied in a lot of depth. As a result of all these studies, a lot of theory is known today about different ways of conducting pooled testing and their relative advantages.

The famous riddle of an evil king trying to identify which bottle of wine is poisoned from a set of a thousand wine bottles using eight slaves is one example of such a test strategy.

Elementary analysis



The core idea behind pooled testing is simple. If a condition is rare, most tests will be negative. Therefore, a group of samples combined together will often result in a negative outcome. Due to this fact, often many negative instances may be identified using a single test, thereby saving a lot of time and resource spent in testing each instance separately. However, there will be additional costs of re-testing each individual sample within a group if a group turns out to be positive. The expectation is that these costs will be offset by the savings in negative groups.

The most basic strategy of group testing then becomes something like this:

a. Determine the size of the group.

b. Form groups of individuals of the determined group size

c. Test every group as one combined entity

d. If the group test outcome is negative – infer that all the group members are negative

e. If the group test outcome is positive, then test every indi vidual in that group to ascertain their status.

Determining the group size is critical for obtaining the maximum benefit from this testing scheme. On the one hand, there may be a desire to increase the group size, in an effort to make sure that we can determine the status of a higher number of individuals with one test. On the other hand, increasing group size means the probability of one (or more) positive individuals being included is higher, thereby negating any advantage of the pooled testing.

Intuitively – it can be seen that there will be some optimal size for which the pooled testing strategy should give maximum benefits. It can also be seen that the optimal group should be a function of the proportion of positives in the population.

To determine the optimal group size, we can consider the following set-up:

- The Proportion of positives in a population is considered to be some constant number. Let us call this p.
- Let us consider a pooled testing strategy with group size = n.
- Then the expected number of tests required for the group = E(T), T being the number of tests required, which can be determined using elementary probabilistic techniques.
- The n for which the expected number of tests per individual = E(T)/n is minimized can then be thought of as the optimized test with the maximum benefit.

Table 3.1 This shows for a specific value of p (0.01) the expected number of tests required for different group sizes, the expected number of tests required for the group and per individual and the optimal size of the group.

	p	0.01	
Group size	**Tests reqd**	**Test per person**	**Gradient**
6	1.3511	0.2252	0.0183
7	1.4755	0.2108	0.0110
8	1.6180	0.2023	0.0064
9	1.7783	0.1976	0.0032
10	1.9562	0.1956	0.0009
11	**2.1513**	**0.1956**	**-0.0007**
12	2.3634	0.1969	-0.0020
13	2.5922	0.1994	-0.0029
14	2.8376	0.2027	-0.0036
15	3.0991	0.2066	-0.0042
16	3.3767	0.2110	-0.0047

Table 3.1: Optimal group size determination when p=0.01.

p	Optimized group size	Expected no. of tests	Tests per person	Gradient
0.01	11	2.1513	0.1956	-0.00073
0.02	8	2.1939	0.2742	-0.00156
0.03	6	2.0022	0.3337	0.00241
0.07	4	2.0078	0.5019	0.00821
0.001	32	2.0083	0.0628	0.00001
0.002	23	2.0350	0.0885	-0.00002

Table 3.2: Optimized group sizes for various values of p

Table 3.2 This shows the optimized groups sizes for different values of p as well as the expected number of tests required for the optimized group size and the expected number of test per individual that boils down to.

One of the nicer aspects of the simple strategies is that they are analytically tractable and the optimized set-ups may be determined using probability and calculus.

Deeper strategies

With some consideration, it can be seen that there is a huge scope of bettering the pooled testing strategy that we saw earlier. A few general classes of betterment that may be considered are:

a. Multidimensional pooling: The pooling strategy we considered above can be thought of as one-dimensional pooling. Can we do better than that by considering higher dimensional pooling?

b. Multistage pooling: In the above strategy we pooled once and then tested all units of groups that came positive. Can we do better than that by pooling for a second time also? And then a third time? How many stages of pooling do we attempt and how do we determine the optimal strategy (No. of stages, No. of groups in each stage)?

c. Scoring of test units: Instead of using a population-level incidence rate, can we do better optimization if some model can score the test units for their individual probability of being positive (using available and known attributes) and then use those scores to come with a smarter scoring approach that can provide more benefits?

The simple pooling approach that was section 3 above was analytically tractable, and hence the optimal strategies could be determined using closed-form approaches. These strategies are either not analytically tractable at all, or even if it is possible to handle them analytically, the result will become too cumbersome to be of any real use.

The best way to analyze these strategies and find the optimized set-up would be to use the powerful sequential optimization techniques of BRIDGEi2i Optimizer™.

However, before going there, it would be better to understand these deeper techniques and strategies a little more clearly

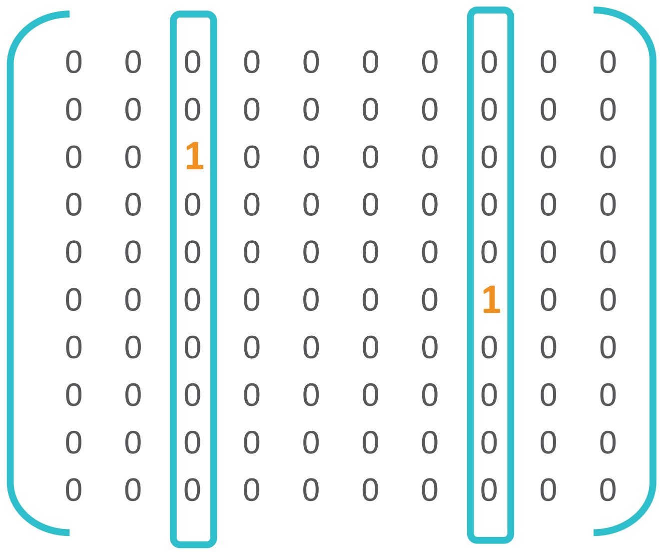
a. Multidimensional pooling

Instead of pooling test units along one dimension and then testing the groups, pool them along multiple dimensions and then test those groups along different dimensions. The intersection of the positive groups along different dimensions will then help in zeroing down on the positive units faster.

Consider the following example, where two units out of a hundred units are positive.

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

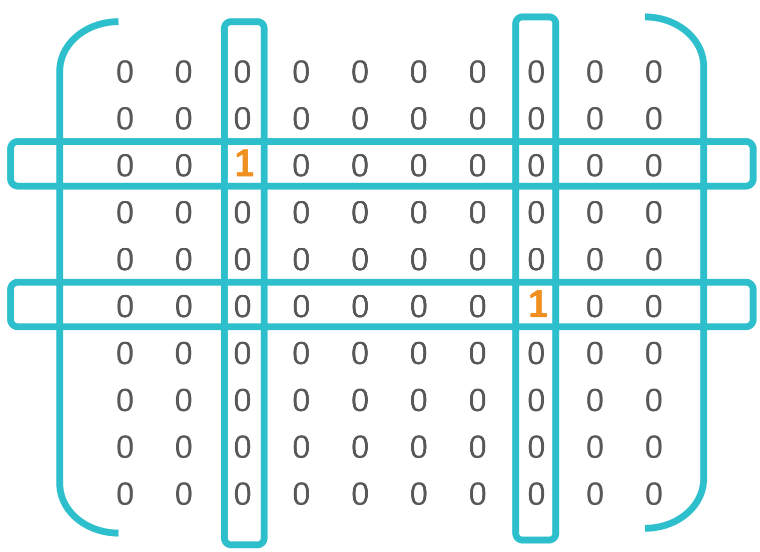
If we do a one-dimensional pooling the situation would look something like below:



1. We made 10 groups.
2. For each positive group 10 further tests will be required
   - 10 tests required for the first level filtering
   - 10+10 = 20 more tests required for the second level tests to ascertain the status of all the test units.

Hence a total of 30 tests required to ascertain the status of all the 100 test units.

However, if we do a two-dimensional pooling, the situation would look something like below:



1. Make 10 groups along one dimension and 10 groups along the second dimension.
2. The intersection of the positive groups need to be tested further.
   - 20 tests required for the first level filtering.
   - 4 more tests required for the second level tests to ascertain the status of all the test units.

Hence a total of 24 tests required to ascertain the status of all the 100 test units.

Hence, we can see that the two-dimensional pooling is more efficient than one-dimensional pooling. It's interesting to note that the multidimensional tests need not be restricted to two dimensions. We can extend the concept and create higher dimensional tests (3,4,5,6,...) and it will be important to see for which dimension the structure may be optimized to provide the maximum benefit from a single test.

b. Multistage pooling

The strategy explores possibilities of increasing testing efficiency by combining test units in multiple different stages. The primary approach can be summarized as:

I. Determine the number of pooling stages

II. Make the first stage pools.

III. Discard the negative pools. Retains positive pools for further testing.

IV. Make next stage pools

V. Repeat steps II to IV till the number of pooling stages are the same as that determined in step I.

VI. Test all units in the positive groups individually.

Thus, every multistage pooled testing strategy may be represented by a tree structure with a certain number of levels (which is the same as number of stages in the test) and the number of branches in each level (which is the same as the number of groups at any specific stage).

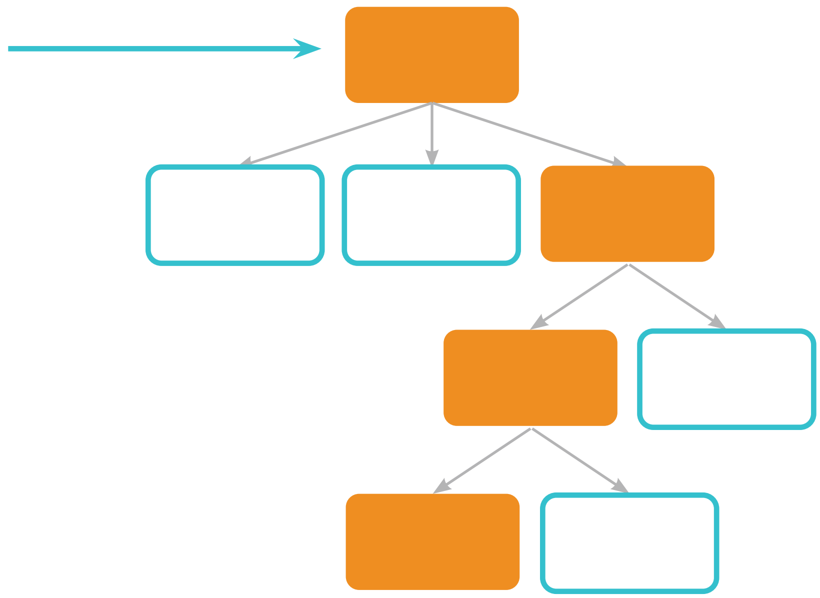
Thus a {3,2,2,4} tree/multistage pooling would have three branching/grouping stages and would look something like below:

Stage 0: First level population of positive group

Stage 1: Three stage 1 sub-groups of which one is positive

Stage 2: Two stage 2 sub-groups of which one is positive

Stage 3: Two stage 3 sub-groups of which one is positive. All the units in the stage 3 sub-group are to be tested individually and are of size 4.



The question that remains is how do we figure out the optimal multistage pooling strategy, especially in view of the analytical intractability of the general class of such multistage pooled tests.

In the next section we shall see how BRIDGEi2i Optimizer™ techniques can be used to determine optimized strategies in complex situations like this.

c. Scoring test units

Another strategy that may help us is if we have some idea about the individual propensity of each test unit to test positive. Saraniti[1] mentions a technique of increasing efficiency of tests by segregating the units in high likelihood cases and low likelihood cases. We can improve on that by knowing the propensity of each unit to test positive and then determining the pools using the additional information available through the scoring models.

Each of the classes of strategies mentioned above, either by themselves or in conjunction with the others can significantly improve the efficiency of the tests and help in covering a large chunk of the population using a relatively small number of tests. However, the layered structures, as well as the analytical intractability, means that it becomes very difficult to study these tests and come up with an optimized set-up of these tests that may be applied in real-life scenarios.

In the next section, we see how we can use BRIDGEi2i Optimizer™ powerful sequential optimization tools to study the class of strategies described in section 4.2 (multistage pooling) and come up with optimal set-ups that provide significant enhancement in testing efficiency.

Optimization in Multistage Pooling

To determine the optimized multistage pooled testing strategies, we use BRIDGEi2i Optimizer™ to generate the distribution of the number of tests required under various population assumptions and testing strategies.
For the purpose of illustration, we consider a population with p = 0.002 (0.2%).

The following are the generated distributions for a few specific strategies:

10,10.10 – 2 stage strategy
Average test per unit = 0.0472

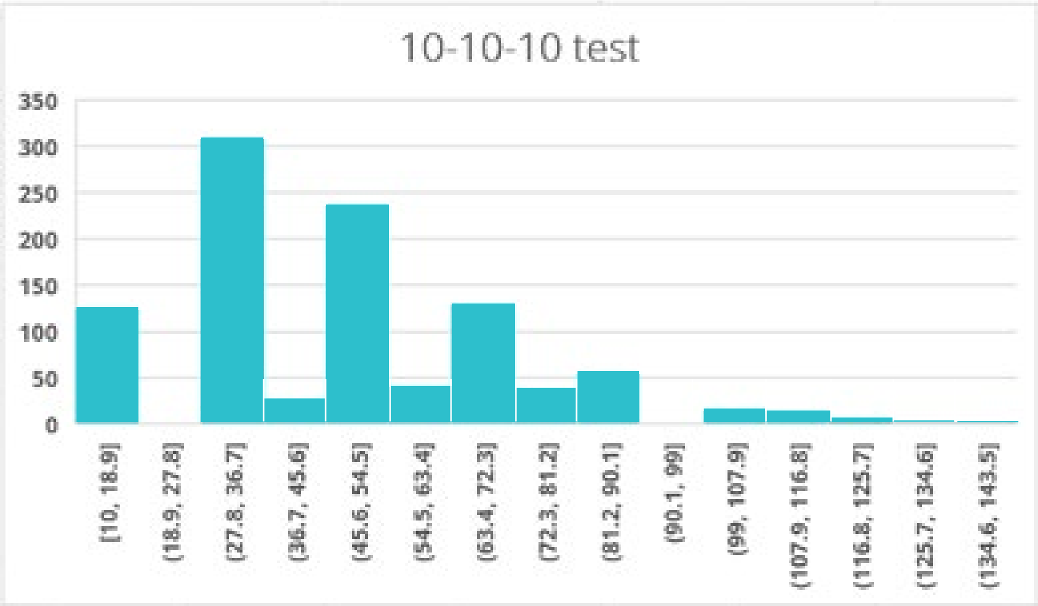


Fig 5.1: Distribution of number of tests required to confirm the status of all test units using a multistage 10-10-10 testing

6,7,6,8 – 3 stage strategy
Average number of tests per individual: 0.03999

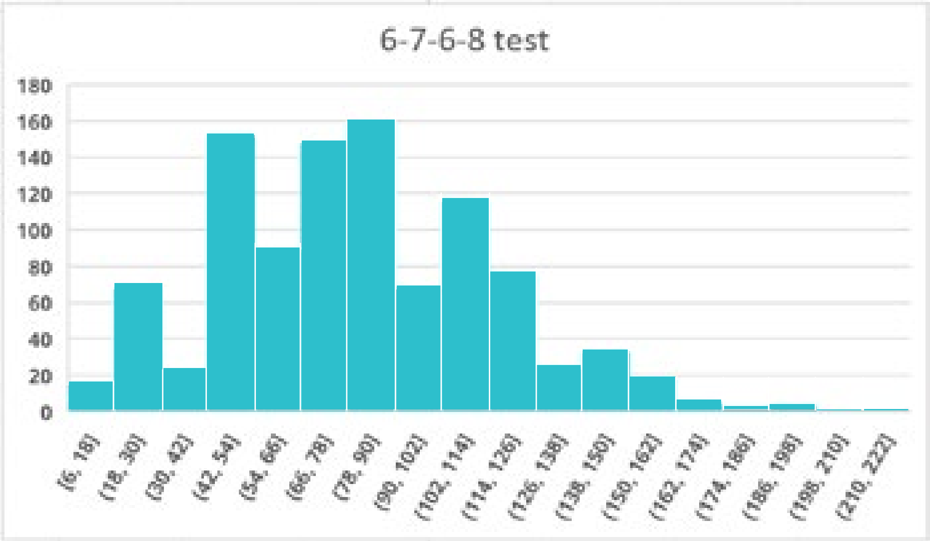


Fig 5.2: Distribution of number of tests required to confirm the status of all test units using a multistage 6-7-6-8 testing

3,3,3,3,3 – 4 stage strategy
Average number of tests per individual: 0.03599

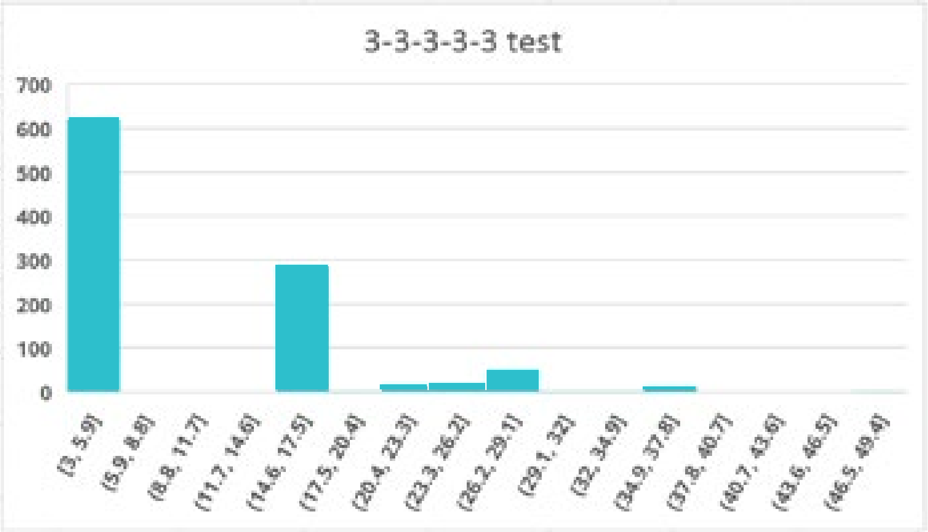


Fig 5.3: Distribution of number of tests required to confirm the status of all test units using a multistage 3-3-3-3-3 testing

To find the optimal strategy, various different stages and branching were considered. The following table summarizes the results over the balanced trees:

2 stage tests		3 stage tests		4 stage tests		5 stage tests	
Test	#Test per unit	Test	#Test per unit	Test	#Test per unit	Test	#Test per unit
6,6,6	0.05206	3,3,3,3	0.05378	2,2,2,2,2	0.07737	2,2,2,2,2,2	0.10044
7,7,7	0.04761	4,4,4,4	0.38563	3,3,3,3,3	0.03599	3,3,3,3,3,3	0.03237
8,8,8	0.04748	5,5,5,5	0.0368	4,4,4,4,4	0.03225	4,4,4,4,4,4	0.03441
9,9,9	0.04222	6,6,6,6	0.03836	5,5,5,5,5	0.03671	5,5,5,5,5,5	0.03606
10,10,10	0.0472	7,7,7,7	0.04068	6,6,6,6,6	0.03815		
11,11,11	0.05007			7,7,7,7,7	0.04062		

Table 5.1: Number of tests required per testing unit using various different multistage testing strategies.

It can be seen that the best results are obtained in a 4 stage test designed as a (4,4,4,4,4) test. For this optimized test, a mere 0.03225 tests are required per individual on average. That translates to more than 31 individuals state being confirmed by a single test.

It may be recalled from Table 3.2 that the best single stage test involves a group size of 23 and is able to get a best possible result of 0.0885 tests per individual. Thus – the best multistage test can deliver a performance in excess of 2.7 times that of the best single stage test.

Also – it may be noted that the optimization has been performed only on the class of tests described in section 4.2. This may be combined to the strategies mentioned in sections 4.1 and 4.3 to produce even more optimized test set-ups.

General structure

Determining the optimal test structure among all these different possible options can be quite a daunting task. However, as illustrated by the above example, if the optimization is done properly, the result can be superior to other alternatives by a large factor. Hence it is worth considering the structured approaches through which such optimization may be achieved. The general approach for solving any such problem can be through a combination of the strategy classes discussed in sections 4.1, 4.2 and 4.3. Different approaches may suit different population with varying inherent characteristics.

The broad steps may be characterized as below:

Step 1: Develop an ML model to score the population for the propensity of being positive for the attribute being tested

Step 2: Develop a second ML model that uses the population characteristics, score distribution, etc. to specify the optimized attributes of multi-stage and multi-dimensional pooled tests (like optimal No. of stages and branching, optimal No of dimensions, etc.)

Step 3: Search around the optimized parameters to look for any possible improvement.

Conclusion

The concept of pooled testing can be applied in multiple different areas, from clinical trials to industrial quality control to network security, among others. In all these places, the techniques discussed above can be very useful in real testing scenarios.

Also, the entire problem of optimizing pooled testing is a smaller set of applications of a broader class of problems of sequential optimization with a wide range of applications. The BRIDGEi2i Optimizer™, in general, uses a host of powerful proprietary tools to find solutions for such problems.

References

[1] Saraniti BA (2006) Optimal pooled testing. Health care manage sci. 9:143–149

- Thai MT (2012) Group testing theory in network security, Springer
- Du D-Z, Hwang FK (2006) Pooling designs and nonadaptive group testing. World Scientific, Singapore
- Du D-Z, Hwang FK (2000) Combinatorial group testing and its applications. World Scientific, Singapore

About BRIDGEi2i

BRIDGEi2i is a trusted partner for enabling AI for Digital Enterprises by leveraging Data Engineering, Advanced Analytics, proprietary AI accelerators and Consulting expertise. BRIDGEi2i enables businesses to contextualize data, generate actionable insights from complex business problems, and make data-driven decisions across pan-enterprise processes to create sustainable business impact.

For more details contact us: **enquiries@bridgei2i.com**

India Office

Umiya Business Bay, Tower 2, 2nd Floor,
Cessna Business Park, Kadubeesanahalli,
Outer Ring Road, Bangalore-560103
Phone: +91-80-67422100

U.S. Office

42808 Christy St., Suite 226,
Fremont, CA, 94538,
Phone: +1-650-666-0005

@BRIDGEi2i /bridgei2i /company/bridgei2i @bridgei2i

www.BRIDGEi2i.com