## $\int \mathrm{cdj}$

## The elegant geometry behind Rwanda's COVID-19 pooled testing strategy

Due to a strict lockdown and some tech savvy solutions, Rwanda has remarkably been able to keep its COVID-19 prevalence low. Up until Dec. 16, the Rwanda Biomedical Center has reported 7,032 cases in the country. Pennsylvania, which has a similar population of 12.5 million, has reported 481,819 cases. Part of the reason Rwanda has been so successful in preventing the spread of COVID-19 is because of severe action from the government. As reported by the Wall Street Journal, over 70,000 people have been arrested for social distancing infractions. But the country has also developed some innovative methods and technologies for dealing with COVID-19. Researchers at the University of Rwanda, for example, devised a unique geometric method to reduce the number of COVID-19 tests needed. This powerful geometric method, also called pooled testing, is surprisingly elegant, and I'll explain why.

The way pooled testing works is simple. First, each patient is swabbed for COVID-19. These samples are combined into a single batch. If the batch tests negative, then each sample in the batch is COVID-19 negative. If the batch tests positive, each sample is then tested individually for COVID-19.

Cornell, for example, employs pooled testing to screen 93 samples at a time. With the help of robots to automate the pooling process, the College of Veterinary Medicine tests about 6,000 people a day. Despite this impressive result, we can do better. By creatively splitting and recombining batch samples, thousands of dollars can be saved when testing large populations.

To demonstrate how efficient the University of Rwanda's geometric pooled testing strategy is, let's start with a simple example. Suppose we want to test 27 people. If we wanted to test each person individually, we would need 27 tests.

However, with the geometric pooling method, we can reduce the number of tests needed to 9 . First, let's organize the samples onto something that resembles a Rubik's cube. The 3-dimensional cube has a side length of 3 . We cut the cube into three slices in three different directions. In total, we have nine slices to test.


Next, we perform a test on each sample. Three slices happen to test positive.


Unknown


X-Axis Results


Negative Test


Y-Axis Results


Positive Test


Z-Axis Results

Finally, with these nine tests, we can tell which sample is positive by seeing where the positive slices overlap.


X-Axis Cuts


Y-Axis Cuts


Z-Axis Cuts


Using this geometric slicing strategy, we were able to test 27 people with only 9 tests. To curb the spread of COVID-19, a reduction in tests is critical. In Rwanda, where a test can cost a laboratory $\$ 50$, stepping down from 27 tests to 9 can save $\$ 900$.

## Utilizing higher dimensions

Although the Rubik's cube approach is efficient, a limitation of it is that it can only house 27 samples at a time. To increase its efficiency, something creative we can do is place the samples on a 4-dimensional cube - a hypercube. Constructing a hypercube is quite hard to wrap your head around, but l'll try to keep it simple.

Here are several graphs, each in a different dimension.


Source: Wikipedia
Informally, a graph is $n$-dimensional if each vertex has at least $n$ adjacent vertices (neighbors) in $n$ different directions.

For example, in the illustration above, the plane is 2 -dimensional. Each vertex has 2 neighbors in 2 directions: one to the right or left, and one above or below. Similarly, the cube is 3 -dimensional because each of its vertices has 3 neighbors - one in the $x, y$, and $z$ directions.

Therefore, it follows that in the 4D hypercube, each vertex should have 4 neighbors. To illustrate this, we can simply connect the vertices of 2 cubes. Each vertex has not only a neighbor in the $x, y$, and $z$ directions, but also a neighbor in a new $w$ direction.

To illustrate how this principle applies to pooled testing, let us represent the graphs above with our blocks. In the same way a 2D object is a series of 1D objects and a 3D object is a series of 2D objects, a 4D object is a series of 3D objects.

For example, a plane is a combination of 3 lines. The plane is 2D because each block has at least 2 neighbors in two directions: left or right, and front or back.

A cube is a combination of 3 planes stacked on top of each other. Each block now has a neighbor to the left or right, front or back, and above or below.


Therefore, a 4D hypercube should be a combination of 3 cubes next to each other. Here, the $w$ term denotes which cube of the 4D cube a sample is in. To illustrate, this is a 4D cube that has been sliced in the w-axis.

## W-Axis Cuts


$w=0$

$w=1$

$w=2$

Because each cube individually contains 27 samples, the 4D cube will hold 3 times 27 samples, which is equal to 81 samples. Now that we have our 4D hypercube, we can slice it along each axis just like how we would slice the 3D cube. By mixing and matching the samples on the hypercube, we can test each slice individually and see where the positives overlap. To cut the cube in the $x$-axis, we will cut each cube individually along the $x$-axis and group together the slices to form the new rearranged cubes.


The same thing can be done in the $y$-axis and $z$-axis.

## Y-Axis Cuts




$$
y=0
$$


$y=1$

$y=2$

Z-Axis Cuts


## A trickier example

Let's now say that we now want to identify two positives in a group of 81 samples. We place the samples on a 4D hypercube and perform the first round of slice tests. We see that in each axis, two of the three slices test positive.

## X-Axis Cuts


$x=0$

$x=1$

$x=2$

Y-Axis Cuts

$y=0$

$y=1$

$y=2$


## W-Axis Cuts


$w=0$

$w=2$

Notice that in each direction, two out of the three slices test positive. This is different from our 3D example because in that example, only one of three slices tested positive. To determine which samples test positive, another round of testing is needed. The other positive is found by elimination. To find the first positive sample, we take one of the slices and slice it again. For instance, let's take the cube $w=1$

Slices of the cube $w=1$


X-Axis Cuts


Y-Axis Cuts


Z-Axis Cuts

tests positive. With this information, we can update our previous slices by removing the known positive.

X-Axis Cuts


Y-Axis Cuts

$y=0$

$y=1$

$y=2$

Z-Axis Cuts

$z=0$

$z=1$

$z=2$


By seeing where the slices intersect, we see that the remaining positive is at slices $x=0, y=0, z=1$, and $w=2$.


The positive sample on cube $w=2$
In total, we screened 81 samples with 21 tests, a significant improvement!

## Applying this to the real world

Pooled testing has the potential to reduce COVID-19 caseload significantly. Cornell Professor Peter Frazier estimates that if the entire U.S. population was tested in a fourweek period, then the COVID-19 prevalence rate would drop to $0.3 \%$, allowing $90 \%$ of adults to return to work.

To demonstrate how efficient geometric pooled testing, let's use Cornell as an example. On Nov. 13, Cornell saw a spike in COVID-19 cases due to several social gatherings. Out of 5,990 people tested that day, 9 tests came back positive. With Cornell's naive pooling method, the 5,990 samples would first be split up into 65 pools of 93 samples each. Each pool would then be tested for COVID-19. If the pool tests positive, each of the 93 samples is then tested individually to identify the infected individuals. Based on the prevalence rate, about 8.39 out of the 65 pools would test positive. On average,

$$
65+8.39(93)=845
$$

tests would be performed.
Instead of naive pooling, let's now try the geometric pooling method. First, we split the 5,990 samples onto 74 4D hypercubes with length 3 . Each hypercube holds 81 samples.



In the first round of testing, each hypercube is tested as a whole. On average, 8.35 hypercubes test positive.

| des | des | des | -10 | did | den | 1 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| cte | 040 | 100 | den | -10. | dud | 00 |
| -10 | ded | dud | 10 | did | ded |  |
| -10 | ded | 400 | -100 | ded | 104 |  |
| din | 010 | 10 | -100 | -10 | ent |  |
| -10 | -10 | -100 | des | 10 | ded |  |
| dod | dev | 10 | des | dod | dut |  |
| -10 | -100 | 400 | dus | ded | 010 |  |
| -10 | des | ded | did | ded | ded |  |
| ded | dob | -10 | -100 | 00 | dud |  |
| -10 | -100 | -104 | -100 | ded | 400 |  |
| del | des | -10 | did | den | -10 |  |

99.1\% of the time, each positive hypercube only contains 1 positive sample. In this case, 12 tests are performed. The other $0.9 \%$ of the time, the hypercube contains 2 positive samples. Here, 21 tests would be needed. There is also the extreme case that each hypercube contains 3 or more positive samples, but the probability of that occurring is negligible. Taking a weighted average of these probabilities, the expected number of testes required per positive cube is

$$
(99.1 \%)(12)+(0.09 \%)(21)=12.081
$$

tests. Therefore, our total number of tests required is on average,

$$
74+(8.35)(12.081)=175
$$

tests. We see that geometric pooling requires $4.83 \times$ fewer tests than naive pooling!
If pooled testing is so efficient, why aren't more countries adopting it? In reality, pooling is tough. It's easy to imagine how mistakes can be made when sorting pooling by hand. Also, with how sensitive the COVID-19 tests are, pooling may increase the rate of false positives or negatives.

Nonetheless, pooled testing has profound implications in other areas of research such as computer science. For instance, this group testing method can be used to rapidly find a broken light bulb in a series of Christmas lights. In practice, group testing is used to locate incorrect bits in computer programs.

In Rwanda, the researchers are now developing software to automate the geometric pooling process. Although the process is still in its infancy, field trials have begun in the country. In South Africa, a rugby team has also begun to use the testing method to screen its players and staff. With a dramatic expansion in testing, hopefully, we can see an end to this pandemic soon.

## 0 f in in

