**Computer Science CS134 (Fall 2020)**

*Daniel Aalberts, Duane Bailey, & Molly Feldman*

Laboratory 10

*More Efficient Covid Testing*

**Objective:** Learn how to conduct simulations of random events.

In this lab [1], we will be exploring ways to improve the cost-effectiveness of covid PCR[2] testing. PCR tests duplicate the part of a viral RNA ⇀ bookended between two "primers" to make ⇌. (This sort of copying from a template is similar to how our DNA is copied when cells divide.) In each PCR round, the double strands are separated and become templates to make another copy (each ⇀ makes a ↽, and each ↽ makes a ⇀). This amplification process produces an exponential growth that allows detection of whether a particular sequence was present in the sample.

| round | ⇀ | ↽ |
|---|---|---|
| 0 | 1 | 0 |
| 1 | 1 | 1 |
| 2 | 2 | 2 |
| 3 | 4 | 4 |
| 4 | 8 | 8 |
| n | $2^{n-1}$ | $2^{n-1}$ |

There are typically 30 or 40 rounds of amplification so 1 virus molecule can create billions of this fragment, which is sufficient to detect. PCR is exquisitely sensitive, so there are very few False Negatives (samples with virus that do not produce a signal) and very few False Positives (samples with no virus that produce a signal). The reason people are not PCR tested post-infection is that with such a sensitive test, sub-infectious levels of virus (while the patient's immune system eliminates the last remnants) can still be detected.

Although PCR is very effective, there is a cost and efficiency tradeoff; we can run one test per sample, but that can be expensive. On the other hand, we can combine multiple samples into a single test, but knowing which sample is positive can be challenging.

In this lab, we will investigate three possible testing protocols with different ways of handling this tradeoff. To understand these methods, we need to consider two values: the probability p, representing the fraction of positive tests, and the size of a batch S, representing the number of samples (one sample represents *one person*) in a given test batch. For any given batch, the number of positive tests will be some integer close to the expected number, p*S.

"Diagnostic testing" is performed when there is reason to suspect an individual is infected, p=0.03=3% in Berkshire County. "Screening testing" is performed even when there is no known exposure, p=0.0002=0.02% at Williams in Fall 2020.

Here are the three possible testing protocols that we will compare:

1. Basic. One sample per PCR test.

   For example, if p=1% on 384 samples, then about 4 tests are positive (infectious people) and most tests are negative (healthy people).

[1] Inspired by https://www.nytimes.com/2020/08/21/health/fast-coronavirus-testing-israel.html

[2] PCR=Polymerase Chain Reaction. https://www.youtube.com/watch?v=fkUDu042xic

2. The Dorfman Method:

   (a) Pool together part of s samples into one vial for PCR testing. In round one, the number of vials v=S/s.

   (b) Each uninfected vial clears s healthy individuals; but for any vial that tests positive, each of those s samples must be retested to find the infected individuals.

   For example, S=384 people could be tested with s=16 samples in each of 24 vials, or s=8 samples in 48 vials, or s=4 samples in 96 vials.

   Some fraction of these samples will need to be retested. We will conduct a simulation to compare the expected expense for these different s values as a function of the test positivity rate p.

   Other logistical considerations with the Dorfman method: samples must be retained until the first round of testing is completed. Because it requires a second round of testing, the Dorfman method takes more time to identify the sick individuals.

3. The P-Best method of Shental *et al.*[3]:

   (a) For each of the 384 samples, robots pipette a bit into that sample's set of v=6 of 48 vials.

   (b) In the end, there are n=48 samples in each of the 48 vials, but the mapping of samples to vials creates minimal sample-to-sample fingerprint overlap.

   If only one individual in the batch has the virus, v vials will be positive, and the patient with the matching pattern can be identified. There will be minimal overlap with the patterns of the other individuals. If two individuals in the batch have virus, then $\leq$2v vials will be positive.

**Getting Started.** Clone the lab resources from the gitlab repository, as usual:

```
git clone https://evolene.cs.williams.edu/cs134-labs/22xyz3/lab10.git ~/cs134/lab10
```

where your CS username replaces 22xyz3. You will the find Python files `basic.py`, `dorfman.py`, `pbest.py`, and the data file `pooling384-48-by-sample.txt`.

**Required tasks (8 points for Basic+Dorfman, 2 points for PBest).**

We will be making plots of number of PCR tests performed (which is proportional to the expense) to test 384 individuals with the three different methods. We will also estimate the number of False Positives.

1. We begin by simulating the basic PCR testing approach.

   (a) Complete `basic(p,samples=384)` to use `random` to simulate whether each of the 384 samples is positive or negative, and to return the total number of positives for probability p.

   (b) Complete `simBasic(p, samples=384, numTrials)` to repeat the `basic` simulation over 1000 batches to get reasonable statistics. This function should return a list with the total positive tests from each batch to make a histogram plot to compare these three probabilities.

[3] `https://advances.sciencemag.org/content/6/37/eabc5961.full`

(c) Run the code to make a histogram plot of the number of infected individuals observed in a given run of 384 samples, with test-positivity rates of p of 0.02%, 0.2%, 2%. (With p=0.02= 2%, the mean number is about 7.68, so in a given batch, you should get an integer near that value.) Answer the questions in the docstring at the top of the file.

2. With the Dorfman method, we will look at three variants (s=16 samples in each of v=24 vials, s=8 samples in v=48 vials, or s=4 samples in v=96 vials).

   (a) Complete `nDorfman(p,s,v)` that simulates a batch with probability p, and computes how many total PCR tests are required for that batch. Recall that for a given batch, the first-round vials with infected samples will need samples to be retested in the second-round.

   (b) Repeat the simulations for 1000 batches for `pvals = [0.02*(2**i) for i in range(-5,3)]` and `(s,v) in [(16,24), (8,48), (4,96)]` to find the mean number of PCR tests for each. Plot this data, with a line for each of the three `(s,v)` versions with p on the x-axis. Which `(s,v)` combination is most cost effective? Where would Basic and P-Best costs be on the plot?

   (c) Make a histogram of the number of tests for `(16,24)` and `(4,96)` at `p=0.02`. What do you observe about the means and the distribution widths? Answer docstring questions.

3. In Shental's P-Best protocol with 384 samples, 48 vials, and v=6 vials per person there are $384 \cdot 383/2$ pairs of individuals.

   (a) Complete `overlaps(pools)` that evaluates how many pairs of individuals have zero vials in common? only one vial that overlaps? two vials? three vials? in a P-Best pool design file (e.g. `pooling384-48-by-sample.txt`). Put answers in the docstring.

   (b) With P-Best protocol, the issue is whether the infected samples can be correctly identified. We will plot performance for `p in pvals = [0.001*i for i in range(0,16+1)]`.

   Complete `pBest(p, pools)` that simulates one random batch. Samples are randomly assigned as infectious with probability p. For now, you may assume that if there is any infected sample in a vial that it will give a positive test result. Retain information about which random samples were sick and which vials give positive PCR results. Your function should return a tuple `(TP, FP, FN, TN)`. Here:

   `TP` = True Positives = number of samples test sick and actually sick

   `FP` = False Positives = number of samples test sick and actually healthy

   `FN` = False Negatives = number of samples test healthy and actually sick

   `TN` = True Negatives = number of samples test healthy and actually healthy

   (c) Complete `simulate(p, pools, numTrials)` that simulates `numTrials=100` random batches. If desired, increase `numTrials`.

   (d) The `False Positive Rate = FPR = FP/(TP+FP+FN+TN)` is the probability of false positives. Plot `FPR` as a function of p for the Shental pool design.

   At about what p are the average number of True Positives and False Positives equal?

   Thought question: For high test probabilities the FPR becomes unreasonable with the original Shental design. What happens to the `FPR` if you halve the number of samples in each batch from 384 to 192? (You would double the number of vials from 48 to 96 to still test 384 samples.)