**METHODS**

I simulated the effect sample pooling had on prevalence estimates under five different settings for true prevalence, . I started by generating a population of 500,000 individuals and then let each individual have probability of being infected at sampling time. The number of patient samples collected from the population is denoted by , and the number of patient samples that are pooled into a single well is denoted by . The total number of pools are thus , hereby called . The number of positive pools in an experiment is termed . I calculated the estimated prevalence at each parameter combination by replicating the experiment 100,000 times and report here the 2.5% and 97.5% quantiles of the distribution of .

Explored parameter options:

I considered the specificity () of a PCR-based test to be 1.0 but include simulations with the value set to 0.99. Test sensitivity () depends on a range of uncontrollable factors such as virus quantity, sample type, time from sampling, laboratory standard and the skill of personnel [[12]](https://paperpile.com/c/jepAok/xAvu). There have also been reports of it varying with pooling level [[13]](https://paperpile.com/c/jepAok/aXmB). For the purposes of this study, I fixed the sensitivity first at 0.95, then at 0.7, irrespective of the level of pooling. These estimates are rather low, which would suggest that I am somewhat overestimating the uncertainty of . However, since it is possible that tests will be carried out under suboptimal and non-standardized conditions I prefer to err on the side of caution.

A central point of pooled testing is that the number of positive pools, , divided by the total number of pools, , can be used as a proxy to measure the true prevalence when the test sensitivity and specificity is known. Note that the number of positive pools, , can be approximated in infinite populations as a stochastic variable subject to a binomial distribution with parameters and , where the latter is the probability that a single pool will test positive. A positive pool can arise from two different processes: There can be one or more true positive samples in the pool, and they are detected, or there can be no true positive samples in the pool, but the test gives a false positive result. These two possibilities are represented by the first and second part of the following equation [14], respectively:

|  |  |  |
| --- | --- | --- |
|  |  | (1) |

Closer inspection of the above formula reveals something disheartening: When approaches zero, converges towards . Thus, in low-prevalence scenarios, and for typical values of test sensitivity and specificity, most positive test results will be false positives. Nevertheless, with appropriate levels of sample pooling it is possible to get decent estimates of the true prevalence because the probability of having no positive samples in a pool decreases with .

We can modify equation 1 for finite populations by replacing with , with , and then solving for . This gives us the formula of Cowling *et al.*, 1999 [[15]](https://paperpile.com/c/jepAok/co8t), which is used in the following to calculate from a single sample:

|  |  |  |
| --- | --- | --- |
|  |  | (2) |

Note that the formula incorporates the test parameters and thus gives an unbiased estimate of even in low-prevalence settings. In this formula, x is a stochastic variable with a binomial distribution. It depends on the number of truly positive samples in a pool, another stochastic variable with a binomial distribution. As a final layer of complexity, we can take samples from a finite population. For these reasons I will use Monte Carlo simulations to get estimates for rather than evaluating some closed-form mathematical expression.

*An algorithm for patient-level diagnosis*

A crucial objective of testing is to identify which patients have active COVID-19 infections. This information is not readily apparent from pooled tests, and in order to get diagnostic results at the patient level, some samples will need to be retested. The methodologically simplest algorithm is to consider all samples from negative pools as true negatives, but re-test every sample from a positive pool individually. This is also called Dorfman’s method [4]. This strategy is estimated to increase testing capabilities by at least 69% [6]. In this work I use an algorithm that conserves testing resources even more than this, but which might be more difficult to implement in practice: I remove all samples from negative pools, considering them true negatives. All positive pools are split into two equally large sub-pools, and then the process is repeated. Positive patient-level diagnosis is only made from sub-pools of size 1. The algorithm is illustrated in Fig. 1. Note that this is a sub-optimal version of the generalized binary splitting (GBS) algorithm presented in the context of COVID-19 in [16]. My version is sub-optimal in the number of reactions because I am always running a test on both sub-pools when a parent pool has tested positive. It is possible to run an even lower number of reactions by not testing a sub-pool if the other sub-pool from the same parent pool has been run first and tested negative. (The positive result from the parent pool implies that the second sub-pool must be positive.) However, for practical reasons such as the ability to run multiple tests simultaneously and the fact the tests are imperfect, I have used the algorithm in Fig. 1. A thorough discussion on group testing algorithms and their merit in testing for SARS-CoV-2 is available in [7].