Wiley StatsRef:
Statistics Reference Online

Group Testing for Identification

By *Christopher R. Bilder*

Keywords: *binary response, infectious disease, pooled testing, R, screening, sensitivity, specificity, testing*

Abstract: Grouping (pooling) items together and testing them as one unit for a binary trait is the process known concurrently as group testing and pooled testing. As long as trait prevalence is small, group testing can lead to significant reductions in the number of tests when compared to testing each item individually. The purpose of this article is to outline commonly used group-testing algorithms and describe how to calculate their corresponding operating characteristics. Focus is on using group testing to identify whether an item has a particular trait rather than estimating the probability an item has this trait.

1 Introduction

Group testing is the process of testing items together as a combined group, rather than individually, to identify those items with a binary trait of interest. One of the largest applications of group testing involves the process of testing blood donations for an infectious disease. In this application, specimens from a set of blood donors are obtained and amalgamated to form a "group" or "pool" to be tested as if it were only one specimen. If the group tests negatively, all individuals within the group are declared disease free. Thus, only one test is needed rather than separate tests for each individual within the group. If the group tests positively, there is at least one individual who is positive for the disease. New subgroups or individuals alone can be retested to determine who is positive and who is negative. By screening all blood donations in this manner, large reductions in the number of tests will occur, when compared to testing each specimen separately, with judiciously chosen group sizes in low disease prevalence settings. These large reductions in the number of tests then lead to large reductions in overall testing costs and time. This is why blood supply organizations worldwide, such as the American Red Cross[1], Canadian Blood Services[2], and the German Red Cross[3], use group testing.

Group testing is widely used in other human infectious disease applications. These applications include checking for antiretroviral treatment failure among HIV-positive individuals[4], screening for chlamydia and gonorrhea[5], and detecting influenza outbreaks[6]. Nonhuman infectious disease applications include disease detection in cattle[7], drug discovery[8], faulty network sensor detection[9], genotyping[10], insect to plant virus transmission[11], milk surveillance[12], and virus monitoring in insects[13]. Group testing has been even suggested as a way to save humanity in science fiction applications[14].

University of Nebraska-Lincoln, Lincoln, NE, USA

This article is © 2019 John Wiley & Sons, Ltd.
DOI: 10.1002/9781118445112.stat08227

No matter the application, it is important to make sure that the number of tests is less than the number of items being tested. For this reason, the expected number of tests needs to be calculated prior to implementation. If the expected number of tests is larger than the number items being tested, group testing is not appropriate to use. However, if the opposite is true, then an application of group testing will likely be successful and can make feasible otherwise infeasible testing applications. Particular aspects of group testing can be controlled during implementation to insure success. The remainder of this article discusses the most important of these aspects. Section 2 presents the two main types of group-testing algorithms. Each of these algorithms has different operating characteristics (i.e., expected number of tests, accuracy) associated with them. Section 3 examines these operating characteristics and shows how they can be used to choose the most efficient group testing implementation. Section 4 provides concluding comments about group testing.

Due to the diversity among group testing applications, we will use the terminology associated with human infectious disease testing throughout our exposition. Thus, group testing will be discussed in the context of testing specimens, such as blood, saliva, or urine, obtained from individuals to determine which are positive or negative for a particular disease, such as HIV, West Nile virus, or chlamydia.

2 Testing Algorithms

Dorfman[15] proposed one of the first uses of group testing as a cost-effective way to screen new US military inductees for syphilis during World War II. He suggested to test amalgamated blood specimens from nonoverlapping groups of individuals. If a group tested negatively, all individuals would be declared negative. If a group tested positively, each individual would be retested separately. This testing algorithm is now often referred to as *Dorfman testing* in honor of its originality. Excellent historical accounts about it and group testing in general are available in Johnson *et al*.[16] and Ding-Zhu and Hwang[17].

Dorfman testing is part of a larger class of testing algorithms known as *hierarchical group testing*. This is because (i) testing is performed over separate stages, (ii) an individual can be part of a test only once per stage, and (iii) whether an individual is retested (stage 2) depends on the test outcome from its original group (stage 1). Thus, Dorfman testing can also be referred to as a two-stage hierarchical group-testing algorithm. More stages are possible with these algorithms. For example, Figure 1 shows a diagram of a three-stage algorithm used in San Francisco for HIV testing[18]. In summary, individuals are tested in

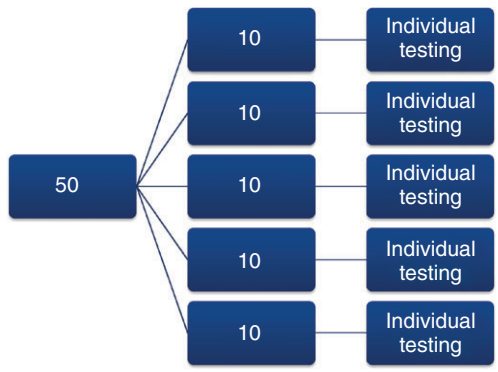

Figure 1. Three-stage hierarchical group testing example. Source: Reproduced by permission of Christopher R. Bilder.

Wiley StatsRef: Statistics Reference Online, © 2014–2019 John Wiley & Sons, Ltd.
This article is © 2019 John Wiley & Sons, Ltd.
DOI: 10.1002/9781118445112.stat08227

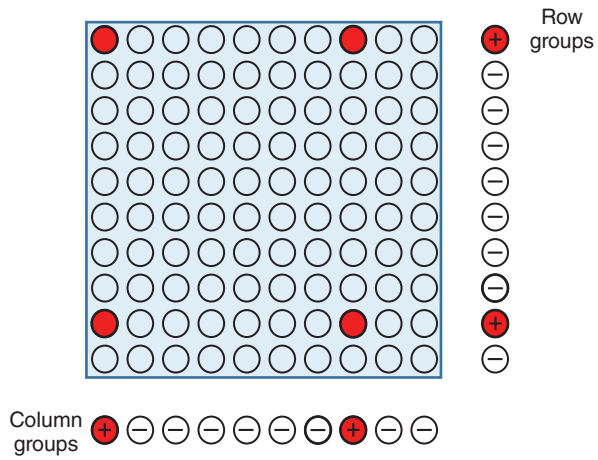

Figure 2. Array testing example. Specimens are pooled within rows and within columns resulting in positive (+) and negative (−) responses. Red-colored wells within the plate indicate where individual testing occurs after initial row and column testing. Source: Reproduced by permission of Christopher R. Bilder.

groups of size 50. If a group tests positively, 5 nonoverlapping subgroups of 10 individuals each are formed in stage 2. If a subgroup tests positively, individual testing is performed upon its members in stage 3. Whenever a group/subgroup tests negatively, all of its members are declared disease free and their testing ends.

Frequently, more than two hierarchical stages can result in a smaller number of tests than using only two. However, implementation of the algorithm becomes more difficult as the number of stages increases. For example, one stage of testing for chlamydia may take an entire workday to complete due to forming groups and assay incubation periods. With respect to human infectious disease testing in general, we have seen at most four stages[19] performed when all original group members can be tested in each stage.

In contrast to hierarchical algorithms, *nonhierarchical group testing* allows for testing the same individuals more than once within a stage. Generally, this type of testing is carried out in array-like structures. Figure 2 provides a diagram of what this looks like relative to a 10×10 microplate. Wells within the plate (represented by circles) contain specimens from separate individuals. Row and column groups are formed from amalgamations of their specimens. These groups are tested simultaneously so that each individual is tested twice in stage 1. Individuals are declared to be negative if their corresponding row and column groups test negatively. Intersections of positive row and column groups indicate where to look for positives. These specimens are retested individually in stage 2 to complete the testing process.

There are a few modifications to the standard array testing algorithm that can be implemented. One modification involves initially testing a *master group* consisting of all specimens within the array[20]. If this group tests negatively, all corresponding individuals are declared to be disease free. If this group tests positively, the testing of row and column groups proceeds as before. Another modification involves arranging specimens either physically or algorithmically into a three-dimensional cubical structure[21, 22]. This leads to groups being formed over rows, columns, and layers, where intersections of positive groups indicate where to look for positive individuals.

Whether testing is performed to find a bacteria in food, an infectious disease in humans, or a binary trait in some other application, testing is often not 100% accurate. Thus, false positives and false negatives may

This article is © 2019 John Wiley & Sons, Ltd.
DOI: 10.1002/9781118445112.stat08227

occur. The possibility of imperfect testing can complicate algorithm implementation. For example, with respect to array testing, a row (column) could test positively without any columns (rows) testing positively and vice versa. A safe approach for handling this particular ambiguity is to simply retest all individuals within the positive row (column)[20].

Interestingly, testing individuals in groups can actually leads to less false positives than through individual testing alone. This is because an individual will always be tested more than once before being declared positive. For example, the hierarchical algorithm given in Figure 1 shows that an individual needs to test positively in two groups and individually once before being declared positive. On the other hand, group testing can result in false negatives more often than with individual testing. Again from Figure 1, an individual could be declared negative from only a negative stage 1 test. For this reason, some group-testing algorithms will retest individuals initially declared to be negative[23]. The potentially higher rate of false negatives can occur due to what is known as the *dilution effect*. Because each individual's specimen becomes a smaller part of the amalgamation as the group size increases, an assay may not be sensitive enough to detect a target that indicates the presence of disease. With respect to infectious disease testing, the dilution effect is more of a concern for older infectious disease testing methods, such as enzyme-linked immunosorbent assays, than newer nucleic acid amplification tests.

3 Operating Characteristics

Section 2 shows that there are two main types of group-testing algorithms, where each can be implemented in a number of different ways. When choosing how to implement an algorithm, the main consideration for a laboratory or other testing body is the number of tests needed for it. For this reason, the expected number of tests is a necessary calculation to make before any application. The accuracy of an algorithm is another important factor to consider as well. An algorithm that results in a small expected number of tests may not be desirable if it has low accuracy. The purpose of this section is to discuss how to calculate these important operating characteristics of a group-testing algorithm. Because derivations can be quite laborious, we sketch out the details for what needs to be found and provide tools from the `binGroup` package of R that automate calculation.

3.1 Hierarchical Testing

Suppose I individuals are to be tested initially as one group through a hierarchical algorithm. Define T as a random variable denoting the number of tests needed to complete the algorithm. Additionally, define G_{sj} as the binary outcome (0 for negative, 1 for positive) from a test for group j at stage s. The number of individuals in group j at stage s is denoted by I_{sj}, and the number of subgroups to be formed if this group tests positively is denoted by m_{sj}. Finally, the number of groups to be tested in stage s is denoted as c_s. Putting this notation into practice, the algorithm depicted in Figure 1 shows $I = I_{11} = 50$ individuals are tested in $c_1 = 1$ group in stage 1, and this test results in the outcome for G_{11}. If the stage 1 group tests positively, $m_{11} = 5$ subgroups are formed. For stage 2, there are $I_{21} = \cdots = I_{25} = 10$ individuals in each of the $c_2 = 5$ subgroups. For each G_{2j}, $j = 1, \ldots, 5$, that results in a positive outcome, there are $m_{2j} = 10$ further tests performed in stage 3. Finally, $I_{3j} = 1$ and $m_{3j} = 0$, $j = 1, \ldots, 50$, because individual testing is performed in stage 3.

To succinctly write out the expected number of tests $E(T)$, we need to introduce the concept of an ancestor group. An ancestor group represents a group that needs to be tested in a previous stage prior to the test of group j in stage s. Define $G_{sj}^{(t)}$ as the ancestor group result for G_{sj} at stage $t \leq s$. Thus, relative to

Wiley StatsRef: Statistics Reference Online, © 2014–2019 John Wiley & Sons, Ltd.
This article is © 2019 John Wiley & Sons, Ltd.
DOI: 10.1002/9781118445112.stat08227

Figure 1, the only ancestor group for G_{21} is $G_{21}^{(1)} \equiv G_{11}$. Also, ancestor groups for G_{31} are $G_{31}^{(2)} \equiv G_{21}$ and $G_{31}^{(1)} \equiv G_{21}^{(1)} \equiv G_{11}$. Notice that all individuals who contribute to the G_{sj} result are a subset of those individuals who contribute to $G_{sj}^{(t)}$. With this ancestor group notation, we can write the expected number of tests for a S-stage hierarchical algorithm as

$$E(T) = 1 + \sum_{s=1}^{S-1} \sum_{j=1}^{c_s} m_{sj} P(G_{sj}^{(1)} = 1, \ldots, G_{sj}^{(s)} = 1) \tag{1}$$

where $G_{sj}^{(s)}$ is equivalent to G_{sj}. The leading 1 in Equation (1) is included for the stage 1 test that will always be performed. The joint probability in Equation (1) corresponds to the testing that occurs in each stage. For example, relative to Figure 1, $m_{21} = 10$ individual tests are performed in stage 3 with probability $P(G_{21}^{(1)} = 1, G_{21}^{(2)} = 1) = P(G_{11} = 1, G_{21} = 1)$. For two-stage hierarchical algorithms, Equation (1) simplifies to only

$$E(T) = 1 + m_{11} P(G_{11} = 1)$$

The joint probability in Equation (1) is a function of the assay sensitivity and specificity and the probabilities that individuals are truly positive for a disease, denoted as p_i for $i = 1, \ldots, I$. Due to the long derivation process and the large expression for the joint probability, we do not provide the expression for it here. Instead, we refer interested readers to Kim *et al.*[20] and Black *et al.*[24] for the expression. The former reference provides the expression only for the special case of when each individual has the same true probability of having a disease ($p_1 = \cdots = p_I$).

3.2 Array Testing

We focus on the standard array testing algorithm in this section. Suppose specimens within an $n \times n$ array are to be tested. Define T again as a random variable denoting the number of tests needed, but now to complete testing for the entire array. The total number of tests for the array is

$$T = 2n + \sum_{i=1}^{n} \sum_{j=1}^{n} T_{ij} \tag{2}$$

where $T_{ij} = 0$ or 1 is the number of retests needed for the specimen in row i and column j after the first stage of testing. The leading $2n$ in Equation (2) is for the number of rows and columns tested. The expected number of tests is the same as Equation (2) but with T_{ij} replaced by $E(T_{ij}) = P(T_{ij} = 1)$, the probability a retest is needed for specimen (i, j).

A retest is needed for specimen (i, j) in three different cases when testing error is possible. To succinctly define these cases, let R_i denote the binary outcome from a group test for row i, and let C_j denote the binary outcome from a group test for column j. A retest needs to be performed on specimen (i, j) when (i) $R_i = 1$ and $C_j = 1$, (ii) $R_i = 1$ and $C_+ = 0$, or (iii) $R_+ = 0$ and $C_j = 1$, where $C_+ = \sum_{j=1}^{n} C_j$ and $R_+ = \sum_{i=1}^{n} R_i$. The latter two cases are needed due to the possibility of testing error. Therefore,

$$P(T_{ij} = 1) = P(R_i = 1, C_j = 1) + P(R_i = 1, C_+ = 0) + P(R_+ = 0, C_j = 1) \tag{3}$$

Each of these probabilities is a function of the assay sensitivity and specificity along with the probabilities that individuals are truly positive, denoted as p_{ij} for $i = 1, \ldots, n$ and $j = 1, \ldots, n$. Kim *et al.*[20] and McMahan *et al.*[25] provide expressions for Equation (3). Again, Kim *et al.*[20] focuses on the case of each individual having the same true probability of disease.

3.3 Accuracy

An ideal group-testing algorithm not only has a small $E(T)$, but also an accuracy as close to perfect as possible. To discuss accuracy in general, define $Y_i = 0$ (1) as a negative (positive) outcome for individual i to be tested in a hierarchical algorithm. Also, define $\widetilde{Y}_i$ as the true binary status of individual i. Note that $P(\widetilde{Y}_i = 1) = p_i$. For nonhierarchical algorithms, the same definitions can be used here and for subsequent discussions by adjusting the context to be for the individual represented by specimen (i, j) in the array.

Two commonly used measures of accuracy are the pooling sensitivity and the pooling specificity. The pooling sensitivity for individual i is the probability that the group-testing algorithm determines that the individual is positive given that the individual is truly positive: $\text{PS}_{e:i} = P(Y_i = 1|\widetilde{Y}_i = 1)$. Similarly, pooling specificity for individual i is the probability that the group-testing algorithm determines that the individual is negative given that the individual is truly negative: $\text{PS}_{p:i} = P(Y_i = 0|\widetilde{Y}_i = 0)$. Predictive probabilities are also used as measures of accuracy. The pooling positive predictive value for individual i is the probability that the individual is truly positive given a positive outcome from a group-testing algorithm: $\text{PPPV}_i = P(\widetilde{Y}_i = 1|Y_i = 1)$. Similarly, the pooling negative predictive value for individual i is the probability that the individual is truly negative given a negative outcome from a group-testing algorithm: $\text{PNPV}_i = P(\widetilde{Y}_i = 0|Y_i = 0)$. Kim *et al.*[20], McMahan *et al.*[25], and Black *et al.*[24] provide expressions for all of these accuracy measures.

The goal for an algorithm is to have each relevant accuracy measure to be as close as possible to 1. At the very least, one wants these probabilities to be as close as possible to the accuracy that could be obtained from individual testing alone.

3.4 Calculations in R

The `binGroup` package in R includes the `hierarchical.desc2()` function that calculates $E(T)$ and accuracy measures for hierarchical algorithms. The function was written for Black *et al.*[24] assuming that sensitivities and specificities are equal for each group tested throughout the testing process. Next are two examples corresponding to the three-stage algorithm used in San Francisco for HIV testing. The overall HIV prevalence among those tested is 0.01752 as given by Sherlock *et al.*[18] Example 1 uses the same three-stage hierarchical algorithm as shown in Figure 1 with $p_1 = \cdots = p_{50} = 0.01752$ and sensitivity and specificity equal to 0.99. Example 2 uses a three-stage hierarchical algorithm again, but with second stage subgroup sizes of $I_{21} = 18, I_{22} = 9, I_{23} = 7, I_{24} = 5, I_{25} = 4, I_{26} = 4, I_{27} = 4$, and $I_{28} = 3$ and $p_1, \ldots, p_{50}$ chosen to be the expected value of order statistics from a beta(0.5, 28.04) distribution. Note that random variables from this beta distribution have an expected value of 0.01752.

```
> # Example 1
> p.vec1 <- rep(x = 0.01752, times = 50)
> hier1 <- hierarchical.desc2(p = p.vec1, I2 = c(10, 10, 10, 10, 10),
    se = 0.99, sp = 0.99)
> hier1$ET  # E(T)
[1] 12.07698
> hier1$individual.testerror[1,]  # Accuracy for first individual
        p  pse.vec   psp.vec  pppv.vec  pnpv.vec
1 0.01752 0.970299 0.9985153 0.9209763 0.9994699
```


Wiley StatsRef: Statistics Reference Online, © 2014–2019 John Wiley & Sons, Ltd.
This article is © 2019 John Wiley & Sons, Ltd.
DOI: 10.1002/9781118445112.stat08227

```
> # Example 2
> p.vec2 <- beta.dist(p = 0.01752, alpha = 0.5, grp.sz = 50)
> hier2 <- hierarchical.desc2(p = p.vec2, I2 = c(18, 9, 7, 5, 4, 4, 3),
    se = 0.99, sp = 0.99, order.p = TRUE)
> hier2$ET  # E(T)
[1] 9.306209
> hier2$individual.testerror[1,]  # Accuracy for first individual
             p  pse.vec   psp.vec    pppv.vec pnpv.vec
1 3.557914e-07 0.970299 0.9997398 0.001324909        1

> (hier1$ET - hier2$ET)/hier1$ET  # Proportion reduction in E(T)
[1] 0.2294259
```

The expected number of tests per individual is $12.08/50 = 0.24$ and $9.31/50 = 0.19$ for examples 1 and 2, respectively. Thus, group testing would reduce the number of tests on average by more than 75% when compared to individual testing. The pooling sensitivity (`pse.vec`) for both algorithms with respect to the first individual is 0.97, which is less than the 0.99 that would be obtained from individual testing. Additional confirmatory testing could potentially increase this pooling sensitivity. The pooling specificity (`psp.vec`) for this individual is greater than the 0.99 that would be obtained from individual testing. Note that the pooling positive predictive value for the second example is very low because this individual has an extremely low probability of being positive ($p_1 = 3.6 \times 10^{-7}$).

When comparing the two examples, example 2 has 22.9% less expected number of tests than example 1. This second example uses a strategy known as *informative group testing*[26] to be more efficient. In general, informative group testing exploits differences in the probabilities of disease by arranging individuals in an optimal sense at each stage. For this example, the methods of Black *et al.*[24] are implemented by ordering individuals using their probabilities of disease (`order.p = TRUE` argument) and sequentially putting them into optimally sized groups in stage 2. Thus, those individuals with the smallest (largest) probabilities are put into the first (last) group. Of course, one does not know if individuals tested for HIV in San Francisco have the probabilities of disease used for the second example. In actual application, individual risk factors and clinical observations can be used in a regression model to estimate these probabilities of disease.

The `Array.Measures()` function from `binGroup` calculates $E(T)$ and accuracy measures for array testing. The function was written for McMahan *et al.*[25] assuming that sensitivities and specificities are equal for the row/column group tests and additional individual retests. Next are two additional examples demonstrating this function using a 10×10 array size and the San Francisco HIV testing setting as motivation. Example 3 assumes that each individual has an equal probability of having the disease. Example 4 assumes that each individual has a probability that is the expected value of order statistics from the same beta(0.5, 28.04) distribution as in example 2. These individuals are arranged in the array using the *gradient pattern* proposed by McMahan *et al.*[25] for informative group testing.

```
> # Example 3
> p.mat1 <- matrix(data = 0.01752, ncol = 10, nrow = 10)
> array1 <- Array.Measures(p = p.mat1, se = 0.99, sp = 0.99)
> array1$T  # E(T)
[1] 24.431
```


This article is © 2019 John Wiley & Sons, Ltd.
DOI: 10.1002/9781118445112.stat08227

```
> c(array1$PSe[1,1], array1$PSp[1,1], array1$PPV[1,1], array1$NPV[1,1])
    # Accuracy for individual (1,1)
[1] 0.9740124 0.9997244 0.9843829 0.9995367

> # Example 4
> p.vec3 <- beta.dist(p = 0.01752, alpha = 0.5, grp.sz = 100)
> p.mat2 <- Informative.array.prob(prob.vec = p.vec3, nr = 10, nc = 10,
                                   method = "gd")
> array2 <- Array.Measures(p = p.mat2, se = 0.99, sp = 0.99)
> array2$T  # E(T)
[1] 23.71078
> p.mat2[1,1]  # p_11
[1] 0.1287002
> c(array2$PSe[1,1], array2$PSp[1,1], array2$PPV[1,1], array2$NPV[1,1])
    # Accuracy for individual (1,1)
[1] 0.9759013 0.9993788 0.9957092 0.9964508

> (array1$T - array2$T)/array1$T
[1] 0.02947999
```

The expected number of tests per individual is approximately 0.24 for both examples, representing a significant reduction in tests again when compared to individual testing. When comparing the two examples directly, example 4 has 2.9% less expected number of tests than example 3. McMahan *et al.*[25] show that reductions in tests for informative group testing with arrays can be much more significant in higher prevalence situations. The pooling sensitivity and specificity values show similar trends, when compared to individual testing alone, as was seen for the hierarchical algorithms.

The `OTC()` function from `binGroup` can be used to find the testing configuration (number of stages, group sizes) that minimizes the expected number of tests per individual[27]. This allows a laboratory to determine the *optimal testing configuration* for a particular situation. For the San Francisco HIV testing example, we use the `OTC()` function by searching three-stage hierarchical algorithms (`algorithm` ="`D3`") with initial group sizes of $I = 4, \ldots, 60$ and by assuming each individual has a probability of disease equal to 0.01752.

```
# Example 5
> save.config <- OTC(algorithm = "D3", p = 0.01752, Se = 0.99, Sp = 0.99,
                     group.sz = 4:60, obj.fn = "ET")

<output edited>

> save.config$opt.ET$OTC  # Optimal testing configuration
$Stage1
[1] 16
$Stage2
[1] 4 4 4 4
> save.config$opt.ET$ET  # E(T)
[1] 3.10546
```


Wiley StatsRef: Statistics Reference Online, © 2014–2019 John Wiley & Sons, Ltd.
This article is © 2019 John Wiley & Sons, Ltd.
DOI: 10.1002/9781118445112.stat08227

The expected number of tests per individual is $3.1055/16 = 0.1941$ when a testing configuration with $I_{11} = 16$ and $I_{21} = \cdots = I_{24} = 4$ is used. In comparison to example 1, there is a 19.6% reduction in the expected number of tests per individual by using the optimal testing configuration.

4 Conclusion

Group testing is used in a vast number of applications from infectious disease testing to genetics to quality control. The key elements that tie all of these applications together is the search for a binary trait when there is a low overall prevalence for it. To insure that an application of group testing is successful, the expected number of tests needs to be calculated and related to the total number of items to be tested. This expectation can also be used to find the optimal testing configuration for group testing.

As mentioned in Section 3.4, the true probabilities of disease (p_i for hierarchical algorithms and p_{ij} for array testing) will be unknown in actual application. Most often, estimates from past experience are used so that an estimate of $E(T)$ can be found. Because group testing is used in large testing situations, such as screening high volumes of clinical specimens for infectious diseases, there is usually a plethora of past data available to provide good estimates. Alternatively, uncertainity in these true probabilities can be incorporated into a Bayesian context. Limited statistical research has been performed for this scenario. For example, Malinovsky and Albert[28] proposed using uniform or Jeffreys' priors for an overall disease prevalence p and minimizing the Bayes risk to find an optimal group size. However, this work is only for two-stage hierarchical testing without the possibility of testing error.

Our article examines group testing for one disease of interest. Multiple-disease tests, known as *multiplex assays*, are becoming more common for infectious disease testing. For example, the Aptima Combo 2 Assay tests for chlamydia and gonorrhea simultaneously rather than performing separate tests for each disease. These multiplex assays allow laboratories to save time and costs beyond using group testing alone. Tebbs *et al.*[29] and Hou *et al.*[30] were the first papers to examine the combination of multiplex assays and group testing in a statistical context. These papers were awarded the American Statistical Association's Outstanding Statistical Application Award in 2014 and 2018, respectively, for their work. Following these papers, Bilder *et al.*[31] proposed the use of informative group testing with multiplex assays.

Group testing is used frequently to estimate the probability an individual has a particular disease. Research on estimation was originally performed in the context of each individual having the same true probability of disease[32–34]. For example, understanding the overall prevalence of a disease is an important goal of public health studies. More recently, research on estimation has developed a wide variety of parametric, nonparametric, and Bayesian regression techniques to estimate the probability an individual has a disease given a set of subject-specific factors[35–37]. For example, understanding what subject-specific factors affect the probability of disease may also be an important goal of public health studies. Remarkably, whether estimation is done for an overall prevalence or subject-specific probabilities, one can have more precise estimators when using group testing rather than individual testing[38, 39]. Thus, despite a smaller sample size as represented by the number of tests, group testing leads to smaller standard errors than individual testing.

Group testing is also used for estimating probabilities when identifying specific items with a trait is not even a goal. For example, multiple vector transfer design experiments are often performed by plant pathologists to determine the transmission rate of a virus from an insect vector (an insect carrying a pathogen) to a plant[40, 41]. In this type of experiment, separate groups of insects are transferred to individually enclosed plants. These plants are observed for a period of time to determine if they become infected. Thus, a group response denotes if a plant has a disease. Knowing which insects are responsible within a group is not of interest.

This article is © 2019 John Wiley & Sons, Ltd.
DOI: 10.1002/9781118445112.stat08227

Outside of statistics, *combinatorial group testing* has similar goals as described in this article but relies primarily on nonstatistical approaches to search for items with a binary trait. For example, these approaches may assume that a particular number of positive items exist and then search for them with tree-based methods. General references for this area include Ding-Zhu and Hwang[17, 42].

Acknowledgment

This research was supported in part by Grant R01 AI121351 from the National Institutes of Health.

Related articles

Statistics in Drug Discovery; **Dorfman-Type Screening Procedures**.

References

[1] American Red Cross (2018) *Infectious Disease Testing*, *https://www.redcrossblood.org/biomedical-services/blood-diagnostic-testing/blood-testing.html* (retrieved 25 October 2018).
[2] O'Brien, S., Yi, Q., Fan, W. et al. (2012) Current incidence and residual risk of HIV, HBV and HCV at Canadian Blood Services. *Vox Sang.*, **103**, 83–86.
[3] Schmidt, M., Pichl, L., Jork, C. et al. (2010) Blood donor screening with cobas s 201/cobas Taqscreen MPX under routine conditions at German Red Cross institutes. *Vox Sang.*, **98**, 37–46.
[4] Kim, S., Kim, H., Kim, H. et al. (2014) Pooled nucleic acid testing to identify antiretroviral treatment failure during HIV infection in Seoul, South Korea. *Scand. J. Infect. Dis.*, **46**, 136–140.
[5] Papp, J., Schachter, J., Gaydos, C., and Van Der Pol, B. (2014) Recommendations for the laboratory-based detection of Chlamydia trachomatis and Neisseria gonorrhoeae, Tech. Rep. RR-02. Centers for Disease Control and Prevention.
[6] Hourfar, M., Themann, A., Eickmann, M., et al. (2007) Blood screening for influenza. *Emerg. Infect. Dis.*, **13**, 1081–1083.
[7] Nebraska Veterinary Diagnostic Center (2016) General Policies and Fee Schedule, *http://vbms.unl.edu/VDC/Information/VDCFeeSchedule.pdf* (retrieved 25 October 2018).
[8] Salzer, E., Nixon, E., Drewes, G., et al. (2016) Screening pools of compounds against multiple endogenously expressed targets in a chemoproteomics binding assay. *J. Lab. Autom.*, **21**, 133–142.
[9] Lo, C., Bai, Y., Liu, M., and Lynch, J. (2013) *Efficient sensor fault detection using combinatorial group testing 2013. IEEE International Conference on Distributed Computing in Sensor Systems,* pp. 199–206.
[10] Chi, X., Lou, X., Yang, M., and Shu, Q. (2009) An optimal DNA pooling strategy for progressive fine mapping. *Genetica*, **135**, 267.
[11] Gildow, F., Shah, D., Sackett, W. et al. (2008) Transmission efficiency of Cucumber mosaic virus by aphids associated with virus epidemics in snap bean. *Phytopathology*, **98**, 1233–1241.
[12] Græsbøll, K., Andresen, L., Halasa, T., and Toft, N. (2017) Opportunities and challenges when pooling milk samples using ELISA. *Prev. Vet. Med.*, **139**, 93–98.
[13] Khan, S., Chowdhury, P., Choudhury, P., and Dutta, P. (2017) Detection of West Nile virus in six mosquito species in synchrony with seroconversion among sentinel chickens in India. *Parasites Vectors*, **10**, 13.
[14] Bilder, C. (2009) Human or Cylon? Group testing on Battlestar Galactica. *Chance*, **22**, 46–50.
[15] Dorfman, R. (1943) The detection of defective members of large populations. *Ann. Math. Stat.*, **14**, 436–440.
[16] Johnson, N., Kotz, S., and Wu, X. (1991) *Inspection Errors for Attributes in Quality Control*, CRC Press, London.
[17] Ding-Zhu, D. and Hwang, F. (2000) *Combinatorial Group Testing and its Applications*, World Scientific, Singapore.
[18] Sherlock, M., Zetola, N., and Klausner, J. (2007) Routine detection of acute HIV infection through RNA pooling: Survey of current practice in the United States. *Sex. Trans. Dis.*, **34**, 314–316.
[19] Quinn, T., Brookmeyer, R., Kline, R. et al. (2000) Feasibility of pooling sera for HIV-1 viral RNA to diagnose acute primary HIV-1 infection and estimate HIV incidence. *AIDS*, **14**, 2751–2757.
[20] Kim, H., Hudgens, M., Dreyfuss, J. et al. (2007) Comparison of group testing algorithms for case identification in the presence of test error. *Biometrics*, **63**, 1152–1163.

Wiley StatsRef: Statistics Reference Online, © 2014–2019 John Wiley & Sons, Ltd.
This article is © 2019 John Wiley & Sons, Ltd.
DOI: 10.1002/9781118445112.stat08227

[21] Kim, H. and Hudgens, M. (2009) Three-dimensional array-based group testing algorithms. *Biometrics*, **65**, 903–910.
[22] Martin, E., Salaru, G., Mohammed, D. et al. (2013) Finding those at risk: Acute HIV infection in Newark, NJ. *J. Clin. Virol.*, **58**, e24–e28.
[23] Litvak, E., Tu, X., and Pagano, M. (1994) Screening for the presence of a disease by pooling sera samples. *J. Am. Stat. Assoc.*, **89**, 424–434.
[24] Black, M., Bilder, C., and Tebbs, J. (2015) Optimal retesting configurations for hierarchical group testing. *J. Royal Stat. Soc. C: Appl. Stat.*, **64**, 693–710.
[25] McMahan, C., Tebbs, J., and Bilder, C. (2012) Two-dimensional informative array testing. *Biometrics*, **68**, 793–804.
[26] Bilder, C., Tebbs, J., and Chen, P. (2010) Informative retesting. *J. Am. Stat. Assoc.*, **105**, 942–955.
[27] Hitt, B., Bilder, C., Tebbs, J., and McMahan, C. (2018) The optimal group size controversy for group testing: Much ado about nothing? Tech. rep., Department of Statistics, University of Nebraska-Lincoln, *http://www.chrisbilder.com/grouptesting* (retreived 25 October 2018).
[28] Malinovsky, Y. and Albert, P. (2015) A note on the minimax solution for the two-stage group testing problem. *Am. Stat.*, **69**, 45–52.
[29] Tebbs, J., McMahan, C., and Bilder, C. (2013) Two-stage hierarchical group testing for multiple infections with application to the Infertility Prevention Project. *Biometrics*, **69**, 1064–1073.
[30] Hou, P., Tebbs, J., Bilder, C., and McMahan, C. (2017) Hierarchical group testing for multiple infections. *Biometrics*, **73**, 656–665.
[31] Bilder, C., Tebbs, J., and McMahan, C. (2018) Informative group testing for multiplex assays. *Biometrics*. doi:10.1111/biom.12988.
[32] Thompson, K. (1962) Estimation of the proportion of vectors in a natural population of insects. *Biometrics*, **18**, 568–578.
[33] Bilder, C. and Tebbs, J. (2005) Empirical Bayesian estimation of the disease transmission probability in multiple-vector-transfer designs. *Biometrical J.*, **47**, 502–516.
[34] Hepworth, G. and Watson, R. (2009) Debiased estimation of proportions in group testing. *J. Royal Stat. Soc. C Appl. Stat.*, **58**, 105–121.
[35] Xie, M. (2001) Regression analysis of group testing samples. *Stat. Med.*, **20**, 1957–1969.
[36] Wang, D., McMahan, C., Gallagher, C., and Kulasekera, K. (2013) Semiparametric group testing regression models. *Biometrika*, **101**, 587–598.
[37] McMahan, C., Tebbs, J., Hanson, T., and Bilder, C. (2017) Bayesian regression for group testing data. *Biometrics*, **73**, 1443–1452.
[38] Liu, A., Liu, C., Zhang, Z., and Albert, P. (2012) Optimality of group testing in the presence of misclassification. *Biometrika*, **99**, 245–251.
[39] Zhang, B., Bilder, C., and Tebbs, J. (2013) Group testing regression model estimation when case identification is a goal. *Biometrical J.*, **55**, 173–189.
[40] Swallow, W. (1985) Group testing for estimating infection rates and probabilities of disease transmission. *Phytopathology*, **75**, 882–889.
[41] Tebbs, J. and Bilder, C. (2004) Confidence interval procedures for the probability of disease transmission in multiple-vector-transfer designs. *J. Agri. Biol. Environ. Stat.*, **9**, 75–90.
[42] Ding-Zhu, D. and Hwang, F. (2006) *Pooling Designs and Nonadaptive Group Testing: Important Tools for DNA Sequencing*, World Scientific, Singapore.

This article is © 2019 John Wiley & Sons, Ltd.
DOI: 10.1002/9781118445112.stat08227