Summer 2021

# Regression Methods for Group Testing Data

Michael Stutz

# Regression Methods for Group Testing Data

by

Michael Stutz

Bachelor of Science 2010
Coastal Carolina University

Master of Science 2017
University of South Carolina

---

Submitted in Partial Fulfillment of the Requirements

for the Degree of Doctor of Philosophy in

Statistics

College of Arts and Sciences

University of South Carolina

2021

Accepted by:

Joshua M. Tebbs, Major Professor

Dewei Wang, Committee Member

Xianzheng Huang, Committee Member

Minsuk Shin, Committee Member

Stella Self, Committee Member

Tracey L. Weldon, Interim Vice Provost and Dean of the Graduate School

# Abstract

Group testing is an efficient method of disease screening, whereby individual specimens (e.g., blood, urine, etc.) are pooled together and tested as a whole for the presence of disease. A common goal is to use data arising from these testing protocols to better understand the relationship between disease status and potential risk factors (e.g., age, symptom status, etc.). Numerous statistical methodologies have been developed for this purpose, most of which are built within the framework of a generalized linear model. Recent authors have suggested the inadequacy of such regression methods to capture the true functional relationships when nonlinear effects are present. In this dissertation, we develop new parametric and nonparametric regression methods for group testing data using the expectation-maximization algorithm. Our methods can be implemented with any group testing algorithm and have the flexibility to seamlessly account for both linear and nonlinear covariate effects. In addition, our methods are the first within the group testing literature to integrate machine learning techniques. A growing number of assays have the ability to detect multiple diseases simultaneously. One such assay is the Aptima Combo 2 Assay (AC2A), which is able to simultaneously test for the presence of chlamydia and gonorrhea. With this as our motivating example, we generalize our regression methods to allow for a bivariate response. We use simulation to demonstrate the estimation performance of our algorithms and provide a real data application of our methods using disease screening data obtained from the University of Iowa.

# Table of Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

Group testing, or pooled testing, first proposed by Dorfman (1943) to screen World War II soldiers for syphilis, is a process whereby individual specimens (e.g., blood, urine, etc.) are pooled together and tested as a whole for the presence of a characteristic of interest, such as disease. By testing specimens in pools, fewer tests are needed to resolve the individual statuses, as multiple specimens are resolved simultaneously when a pool tests negatively. This often confers substantial cost savings when compared to individual testing alone, which has been group testing's primary motivation since its inception (Peeling et al., 1998). However, the recent COVID-19 pandemic highlights an overlooked but invaluable utility of group testing. If testing supplies become scarce, such as in a humanitarian crisis situation, group testing allows more individuals to be screened for life-threatening diseases with the same set of limited resources. Regarding COVID-19, FDA commissioner Stephen Hahn stated "sample pooling becomes especially important as infection rates decline and we begin testing larger portions of the population." Although group testing applications can be found in numerous disciplines including molecular biology (Farach et al., 1997) and genetics (Gastwirth, 2000), its applications in epidemiology are ubiquitous (Kleinman et al., 2005; Lewis, Lockary, and Kobic, 2012; Krajden et al., 2014). Infectious disease applications arise naturally due to the inherent ease of specimen pooling. In addition, the efficiency conferred by group testing is greatest when the characteristic of interest is rare, as is the case with most sexually transmitted diseases.

## 1.1 Common Group Testing Algorithms

Group testing algorithms come in many forms, from the most basic master pool testing, to highly complex hierarchical and array-based methods. In this section, we describe some of the most common algorithms seen in practice.

Master pool testing consists of combining specimens into non-overlapping (master) pools, such that each specimen belongs to a single pool. All specimens within a given pool are then declared either disease-positive or disease-negative based upon the test result of the pool. Master pool testing can be quite efficacious, but only when model estimation is of central interest, as the individual statuses are not resolved. Dorfman testing, as the name suggests, is attributed to Dorfman's seminal work. It is a two-stage hierarchical procedure and is the most common group testing procedure used in practice (McMahan, Tebbs, and Bilder, 2012a). The first stage consists of master pool testing. If a pool tests positively, a separate aliquot of each specimen that contributed to that pool is then individually tested, while if a pool tests negatively, each specimen within that pool is declared negative without further retesting. Figure 1.1 illustrates the use of Dorfman testing on an example data set of nine specimens.

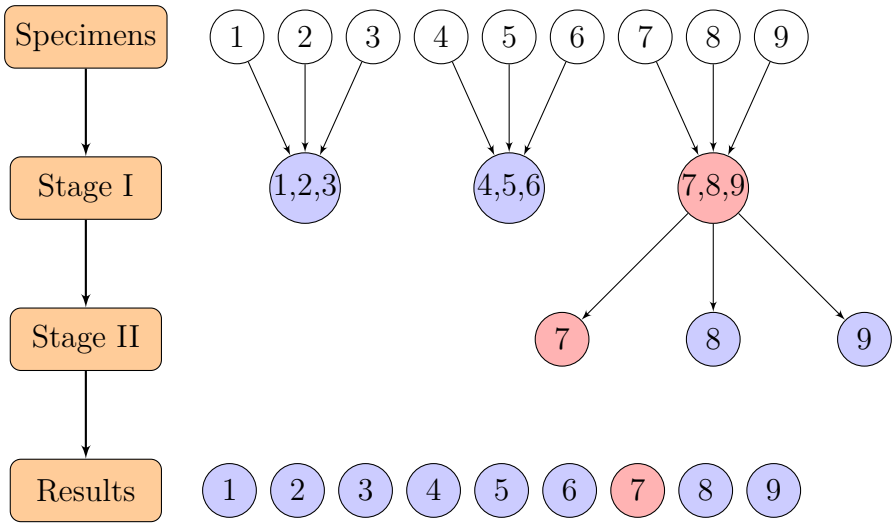


Figure 1.1: Dorfman testing on an example data set of nine specimens. A blue circle indicates a negative test result, while a red circle indicates a positive test result.

Array testing is most commonly a two-stage procedure in which specimens are placed into a square array, with each column and each row of the array representing a pool. The first stage consists of testing each pool, after which, individual tests are performed following the convention of Kim et al. (2007). That is, if a column pool and a row pool both test positively, a separate aliquot of the specimen which contributed to both pools is individually tested. In addition, if a column pool tests positively, but all row pools test negatively, each specimen that contributed to the column pool is individually tested. Likewise, if a row pool tests positively, but all column pools test negatively, each specimen that contributed to the row pool is individually tested. All other specimens are declared negative during the first stage of testing. Figure 1.2 illustrates the use of array testing on an example data set of nine specimens.

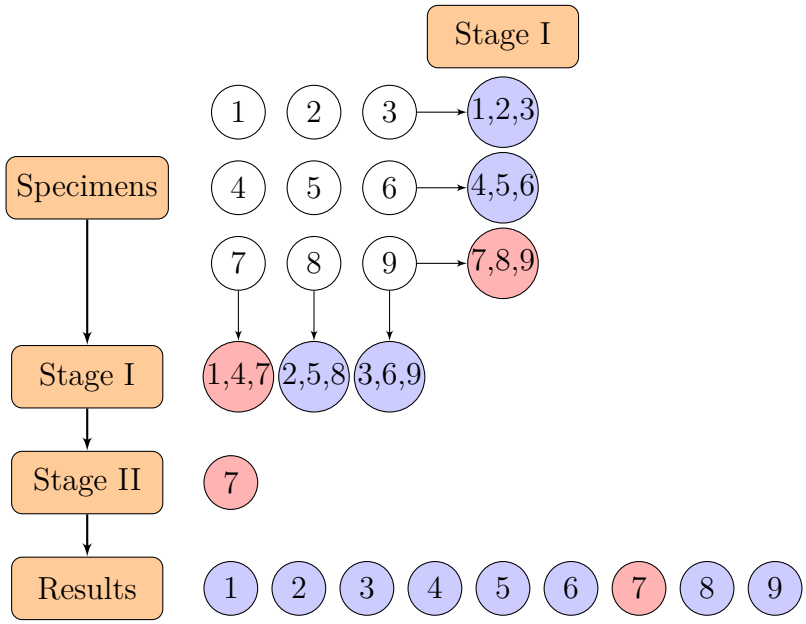


Figure 1.2: Array testing on an example data set of nine specimens. A blue circle indicates a negative test result, while a red circle indicates a positive test result.

## 1.2 Literature Review

To date, the focus of most statistical research on group testing has been either estimation or case identification. In infectious disease applications, estimation concerns

one's ability to estimate the population prevalence of a disease (i.e., the proportion of individuals within a population who have the disease), as well as population-level regression models that relate covariate information (e.g., age, symptom status, etc.) to disease status. Statistical research on case identification, or classification, examines the ability of a group testing algorithm to correctly classify individuals as being either positive or negative for a disease, with a preponderance of the literature examining the performance of hierarchical and array-based pooling algorithms (Kim et al., 2007). The primary focus of this dissertation is model estimation.

The so called "estimation problem" in group testing has historically dealt with estimating disease prevalence. The first research in this area was performed by Thompson (1962), who was interested in estimating the proportion of virus transmitting vectors within a natural population of insects. Upon exposure to a vector, a plant will display symptoms of infection. Thus, if a plant were exposed to $k$ insects, it would have probability $(1-p)^k$ of remaining non-infected, where $p$ (the population prevalence) is the proportion of virus transmitting insects within the population. By repeating this process for $N$ plants, one is able to estimate $p$. Thompson derived the maximum likelihood estimator (MLE) of $p$ as well as its asymptotic distribution, assuming both an error-free response and that the vector-statuses of the insects are independent and identically distributed. Sobel and Elashoff (1975) proposed an alternative approach which takes into account retesting information. Swallow (1985) showed that the MLE of $p$ in Thompson (1962) is biased, and Burrows (1987) proposed an estimator of $p$ which is superior to the MLE in terms of bias and mean square error. Hughes-Oliver and Swallow (1994) developed an adaptive method to estimate $p$ which uses a priori knowledge of $p$ to determine optimal pool sizes. Their method is adaptive in that the pool sizes vary from stage to stage, with the pool size for a given stage being determined based on information from the previous stage. As a worst case scenario, individual testing is performed, provided the a priori upper bound for $p$ is not less

than the true value of $p$. Hung and Swallow (1999) developed models to allow for imperfect testing and non-independence of specimens. More recently, Nguyen, Bish, and Aprahamian (2018) studied a sequential procedure for estimating disease prevalence that uses continuous testing outcomes.

Chaubey and Li (1995) were among the first to approach the estimation problem from a Bayesian prospective, developing two different Bayes estimators of $p$. The first estimator specified a two-parameter beta prior distribution for $p$, while the second estimator specified a prior distribution for $1-(1-p)^k$, the proportion of pools containing an infected specimen. However, both estimators are highly influenced by the choice of hyperparameters. Tebbs, Bilder, and Moser (2003) proposed a parametric empirical Bayes estimator which eliminated the hyperparameter subjectively in the work of Chaubey and Li (1995). Bilder and Tebbs (2005) developed empirical Bayesian approaches for estimating the disease transmission probability in multiple-vector-transfer designs and demonstrated the superiority of their methods when compared to other frequentist and Bayesian approaches.

Farrington (1992) proposed modeling individual-specific disease probabilities based on available covariate information. In this seminal work, Farrington used a generalized linear model with a complementary log-log link function to model the individual disease probabilities, assuming error-free testing and that the individuals within each pool have identical covariate values. The later assumption was also made in the Bayesian approach of Tu, Kowalski, and Jia (1999). Vansteelandt, Goetghebeur, and Verstraeten (2000) generalized the work of Farrington (1992) by allowing for any link function to be used, for testing error, and for heterogeneous pooling (i.e., not requiring the covariate composition within each pool to be identical, or even similar). Huang (2009) extended the work of Vansteelandt et al. (2000) by allowing for covariate measurement error. Chen, Tebbs, and Bilder (2009) were the first to propose a mixed effects model within the group testing setting. Bilder and Tebbs

(2009) explored the effects of pooling composition on parameter estimation. McMahan, Tebbs, and Bilder (2013) proposed incorporating external biomarker information into the modeling process.

The regression methods of the 20th century all share a common restrictive feature, that only the initial (master) pool responses are used for estimation. Unless one is only interested in estimating disease prevalence, group testing protocols involve multiple stages of testing, the information from which is ignored by these methods (Gastwirth and Johnson, 1994; Krajden et al., 2014). Xie (2001) developed an EM algorithm for group testing data capable of incorporating this additional information into the modeling process, and is credited as the first to do so. Since this seminal work, many others have forwarded this area of research. Zhang, Bilder, and Tebbs (2013a) implemented Xie's (2001) EM algorithm to examine the efficiency of numerous group testing protocols and showed that some may even yield more efficient estimates than individual testing when diagnostic tests are imperfect. Wang et al. (2014) proposed a semi-parametric approach to incorporate decoding information (i.e., information gained from retesting positive pools). McMahan et al. (2017) proposed a general Bayesian regression framework for modeling group testing data. Gregory, Wang, and McMahan (2019) developed an adaptive elastic net estimator capable of both model estimation and variable selection. Delaigle, Huang, and Lei (2019) studied model estimation in the presence of missing covariates.

In recent years, there has been a burgeoning interest in semiparametric and nonparametric modeling extensions, including the works of Delaigle and Meister (2011), Delaigle and Hall (2012; 2015), Wang, Zhou, and Kulasekera (2013), Delaigle, Hall, and Wishart (2014), Wang et al. (2014), and Liu et al. (2020). These works, unlike those previously mentioned, do not assume the relationship between disease status and potential risk factors is linear in nature; thus, providing a more flexible framework for assessing an individual's risk for disease. However, the majority of these

papers continue to make the rigid assumptions of their predecessors (e.g., requiring master pool testing), which render them inapplicable in most settings. The Bayesian generalized additive model of Liu et al. (2020) provides a compromise between the familiarity and interpretability of a linear model and flexibility of a nonparametric model, allowing some effects to remain linear while others are modeled with smooth functions.

Many researchers have shifted to developing methods capable of handling data arising from the use of multiplex assays, or assays with the ability to detect multiple diseases simultaneously. Zhang et al. (2013b) were among the first to develop regression methods for multiple-disease group testing data, using a generalized estimating equations approach. However, their estimation procedure can only incorporate information from single-stage master pool testing. Tebbs, McMahan, and Bilder (2013) use the expectation-maximization (EM) algorithm to simultaneously estimate the prevalence of multiple diseases, while allowing for testing error. Warasi et al. (2016) developed a Bayesian approach which can estimate the prevalence of multiple diseases as well as the accuracies of the testing assay. Haber and Malinovsky (2017) proposed random walk designs for selecting pool sizes and applied their methods to estimate the prevalence of two Australian crop diseases. Li, Liu, and Xiong (2017) proposed the use of the D-optimal criterion for estimating the prevalence of correlated diseases. Hyun, Gastwirth, and Graubard (2018) developed a method for estimating the prevalence of two traits from complex survey data. Lin, Wang, and Zheng (2019) developed a method for regression analysis and variable selection for multiple-infection group testing data which is able to incorporate retesting information.

## 1.3 Contribution

We first develop a generalized linear model (GLM) which is suitable for data that arise from any group testing methodology (i.e., regardless of the testing algorithm per-

formed, covariate composition within the pools, or number and/or types of diagnostic assays used). This model can best be seen as a generalization of the expectation-maximization (EM) algorithm of Xie (2001), in that it allows for unknown assay accuracy probabilities. This extension enables the algorithm to be applied in more realistic settings, where the effect of pooling dilution may not be fully understood. In addition, this model provides the algorithmic framework which we used to integrate *boosting*. Boosting is a powerful machine learning technique commonly used for regression and classification (Schapire, 1990; Freund, 1995; Freund and Schapire, 1997). Although previous researchers have combined the EM and boosting algorithms to create latent variable models (see, e.g., Yasui et al., 2004; Ward et al., 2009), similar methods have surprisingly never been developed within the group testing literature.

We develop two boosting algorithms; the first is a boosted version of our generalized linear model (BGLM). Although the GLM and BGLM provide similar estimates, boosting can drastically reduce the computation time. Even with as few as twenty covariates this reduction can be a hundredfold. In fact, when the dimensionality of the predictor space becomes large (e.g., 20, etc.), traditional maximization algorithms often become numerically unstable and fail to converge altogether. Unfortunately, boosting does not remove the potentially restrictive linearity assumptions of the GLM. This motivates our second boosting algorithm, a nonparametric boosted regression trees model (BTM). By changing the base learner of the boosting algorithm to a regression tree, we can drastically improve estimation performance in cases when a parametric model fit is in question. In addition, even under linearity conditions, the performance of the BTM is comparable to that of the linear models.

Many modern assays have the capability to screen for multiple diseases simultaneously. One such assay is the Aptima Combo 2 Assay (AC2A), which is able to simultaneously test for the presence of chlamydia and gonorrhea. With this as our

motivating example, we generalize each of the aforementioned regression methods to allow for a bivariate response.

Lastly, we develop a flexible generalized additive model (GAM) that uses the expectation-maximization algorithm to resolve the latency of the true disease statuses. This approach allows linear effects to be identified and retain their interpretability, while more complex relationships are modeled with smooth functions. In addition, it offers many benefits over the Bayesian approach of Liu et al. (2020).

## 1.4 Structure of the Dissertation

The remainder of the dissertation is organized as follows: In Chapter 2, we discuss our generalized linear model for group testing data. We define notation and describe modeling assumptions in Section 2.2 and discuss the methodology in Section 2.3. The chapter ends with a brief discussion. In Chapter 3, we discuss our boosting methods for group testing data. We define notation and describe modeling assumptions in Section 3.2; we discuss the methodology in Section 3.3; we use simulation to examine the estimation performances of our GLM, BGLM, and BTM in Section 3.4; we fit our models to chlamydia screening data obtained from the State Hygienic Laboratory at the University of Iowa in Section 3.5, and conclude the chapter with a brief discussion. In Chapter 4, we extend the methods presented in the previous chapters to allow for multiple-disease group testing data, and in Chapter 5, we discuss our generalized additive model, with the structure of each of these chapters being similar to that of Chapter 3.

# Chapter 2

# Generalized Linear Modeling for Group Testing Data

## 2.1 Introduction

In this chapter, we develop a generalized linear model (GLM) which is suitable for data that arise from any group testing protocol. This model is not only a contribution in its own right, but provides the foundation upon which further algorithms are developed; see Chapters 3, 4, and 5.

## 2.2 Preliminaries

Suppose we have a sample of $N$ individuals that have been tested for a binary characteristic of interest, such as disease status. Let $J$ be the number of pools used to test these individuals. For $j = 1, 2, \ldots, J$, let $\mathcal{P}_j$ be the set of individuals in the $j$th pool, where we require $\cup_j \mathcal{P}_j = \{1, 2, \ldots, N\}$. Individual testing can be seen as a special case of group testing in which $|\mathcal{P}_j| = 1 \ \forall j$, where $j \in \{1, 2, \ldots, N\}$. Allowing for imperfect assays, let $Z_j$ be the observed test result of the $j$th pool and $\widetilde{Z}_j \equiv I(\sum_{i \in \mathcal{P}_j} \widetilde{Y}_i > 0)$ be its latent true status, where $I(\cdot)$ is the indicator function and $\widetilde{Y}_i$ is the true status of the $i$th individual; thus, a pool is truly positive if and only if it contains at least one truly positive individual. Further, let $S_{e(l)} = \text{pr}(Z_j = 1 | \widetilde{Z}_j = 1)$ and $S_{p(l)} = \text{pr}(Z_j = 0 | \widetilde{Z}_j = 0)$ be the sensitivity and specificity of the $l$th assay respectively, where $j \in \mathcal{M}(l)$, and $\mathcal{M}(l)$ is the set of pools such that the $l$th assay was used. We do not require these assay accuracy probabilities to be known and they

may be estimated along with the other model parameters. Additionally, this notation allows the accuracy probabilities of an assay to vary based upon the pool size.

We assume the observed testing outcomes are conditionally independent given the individual true statuses, and the conditional distribution $\mathbf{Z}|\widetilde{\mathbf{Y}}$ does not depend on the covariates, where $\widetilde{\mathbf{Y}} = (\widetilde{Y}_1, \widetilde{Y}_2, \ldots, \widetilde{Y}_N)'$ denotes the latent individual statuses and $\mathbf{Z} = (Z_1, Z_2, \ldots, Z_J)'$ denotes the testing outcomes. Additionally, we assume the true statuses follow independent Bernoulli distributions with success probabilities $P_i \equiv \text{pr}(\widetilde{Y}_i = 1|\mathbf{x}_i, \boldsymbol{\beta}) = H^{-1}(\mathbf{x}_i'\boldsymbol{\beta})$, where $H(\cdot)$ is a monotone differentiable link function, $\mathbf{x}_i = (1, x_{i1}, x_{i2}, \ldots, x_{ir})'$ is the $(r+1)$-dimensional vector of covariates for the $i$th individual, and $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_r)'$ is the $(r+1)$-dimensional vector of regression parameters.

## 2.3 Methodology

As with a standard GLM, parameter estimates can be obtained via maximum-likelihood estimation. Using the notation established in the previous section, the complete log-likelihood function for any group testing protocol can be written as

$$l(\boldsymbol{\theta}|\mathbf{Z}, \widetilde{\mathbf{Y}}, \mathbf{X}) = \sum_{l=1}^{L} \sum_{j \in \mathcal{M}(l)} \{\widetilde{Z}_j \log[S_{e(l)}^{Z_j}(1-S_{e(l)})^{1-Z_j}] + (1-\widetilde{Z}_j)\log[(1-S_{p(l)})^{Z_j} S_{p(l)}^{1-Z_j}]\} + \sum_{i=1}^{N} \{\widetilde{Y}_i \log P_i + (1-\widetilde{Y}_i)\log(1-P_i)\}, \quad (2.1)$$

where $\boldsymbol{\theta} = (\mathbf{S}_e', \mathbf{S}_p', \mathbf{P}')'$, $\mathbf{S}_e = (S_{e(1)}, S_{e(2)}, \ldots, S_{e(L)})'$, $\mathbf{S}_p = (S_{p(1)}, S_{p(2)}, \ldots, S_{p(L)})'$, $\mathbf{P} = (P_1, P_2, \ldots, P_N)'$, $\mathbf{X} = (\mathbf{x}_1\ \mathbf{x}_2\ \ldots\ \mathbf{x}_N)'$, and $L$ is the number of sets of assay accuracy probabilities. Due to the latency of the true disease statuses, direct maximization of the log-likelihood function is not possible. However, the EM algorithm allows one to calculate maximum-likelihood estimates from incomplete or missing data (Dempster et al., 1977). To begin, we calculate the conditional expectation of

the log-likelihood function in equation (2.1), which is given by

$$
\begin{aligned}
Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) &= \mathbb{E}[l(\boldsymbol{\theta}|\mathbf{Z},\widetilde{\mathbf{Y}},\mathbf{X})|\mathbf{Z},\mathbf{X},\boldsymbol{\theta}^{(t)}] = \\
&\sum_{l=1}^{L}\sum_{j\in\mathcal{M}(l)}\Bigg\{\underbrace{\text{pr}\left(\sum_{i\in\mathcal{P}_j}\widetilde{Y}_i>0\middle|\mathbf{Z},\mathbf{X},\boldsymbol{\theta}^{(t)}\right)}_{C_j}\times\log[S_{e(l)}^{Z_j}(1-S_{e(l)})^{1-Z_j}] \\
&\quad+\left[1-\text{pr}\left(\sum_{i\in\mathcal{P}_j}\widetilde{Y}_i>0\middle|\mathbf{Z},\mathbf{X},\boldsymbol{\theta}^{(t)}\right)\right]\times\log[(1-S_{p(l)})^{Z_j}S_{p(l)}^{1-Z_j}]\Bigg\} \\
&+\sum_{i=1}^{N}\{\text{pr}(\widetilde{Y}_i=1|\mathbf{Z},\mathbf{x}_i,\boldsymbol{\theta}^{(t)})\log P_i+[1-\text{pr}(\widetilde{Y}_i=1|\mathbf{Z},\mathbf{x}_i,\boldsymbol{\theta}^{(t)})]\log(1-P_i)\}, \quad (2.2)
\end{aligned}
$$

where the superscript $t$ denotes an estimate from the $t$th iteration of the algorithm. To complete the E-step, we must first derive expressions for $\text{pr}(\widetilde{Y}_i=1|\cdot)$ and $C_j$, where $C_j$ is the conditional probability of the $j$th pool being truly positive. Unfortunately, for most group testing protocols these expectations are intractable. Therefore, we stochastically approximate these probabilities by repeatedly sampling from the conditional distributions of the individual true statuses using Markov Chain Monte Carlo. We have $\widetilde{Y}_i|\mathbf{Z},\widetilde{\mathbf{Y}}_{-i},\mathbf{X},\boldsymbol{\theta}$ follows a Bernoulli distribution with success probability $p_{i1}^*/(p_{i0}^*+p_{i1}^*)$, where

$$
\begin{aligned}
p_{i1}^* &= P_i\prod_{j\in\mathcal{A}_i}S_{ej}^{Z_j}(1-S_{ej})^{1-Z_j}, \\
p_{i0}^* &= (1-P_i)\prod_{j\in\mathcal{A}_i}[S_{ej}^{Z_j}(1-S_{ej})^{1-Z_j}]^{I(\Sigma_{i'\in\mathcal{P}_{ij}}\widetilde{Y}_{i'}>0)}\times[(1-S_{pj})^{Z_j}S_{pj}^{1-Z_j}]^{I(\Sigma_{i'\in\mathcal{P}_{ij}}\widetilde{Y}_{i'}=0)}, \quad (2.3)
\end{aligned}
$$

$\widetilde{\mathbf{Y}}_{-i}=\{\widetilde{Y}_1,\ldots,\widetilde{Y}_{i-1},\widetilde{Y}_{i+1},\ldots,\widetilde{Y}_N\}$, $\mathcal{A}_i=\{j:i\in\mathcal{P}_j\}$, $S_{ej}$ and $S_{pj}$ are the sensitivity and specificity respectively of the assay used to test the $j$th pool, and $\mathcal{P}_{ij}=\{i'\in\mathcal{P}_j:i'\neq i\}$. The estimate of $\text{pr}(\widetilde{Y}_i=1|\cdot)$ is given by the proportion of the imputed statuses of the $i$th individual which are positive. Similarly, the estimate of $C_j$ is given by the proportion of the imputed statuses of the $j$th pool which are positive.

The M-step consists of maximizing the conditional expectation in equation (2.2) with respect to $\boldsymbol{\theta}$. Note, the second line of equation (2.2) is a function of only $\mathbf{S}_e$, the third line is a function of only $\mathbf{S}_p$, and the final line is a function of only $\mathbf{P}$. As

such, we may separately consider the maximizations of $\mathbf{S}_e$, $\mathbf{S}_p$, and $\mathbf{P}$. Maximizing $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ directly with respect to $\mathbf{S}_e$ and $\mathbf{S}_p$ yields the following closed-form solutions:

$$S_{e(l)}^{(t+1)} = \frac{\sum_{j\in\mathcal{M}(l)} C_j^* Z_j}{\sum_{j\in\mathcal{M}(l)} C_j^*} \qquad S_{p(l)}^{(t+1)} = \frac{\sum_{j\in\mathcal{M}(l)}(1-C_j^*)(1-Z_j)}{\sum_{j\in\mathcal{M}(l)}(1-C_j^*)}, \tag{2.4}$$

for $l \in \{1, 2, \ldots, L\}$, where $C_j^*$ is the stochastic approximation of $C_j$.

To complete the M-step, it still remains to maximize $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ with respect to $\mathbf{P}$. For ease of explication, we assume $H(\cdot)$ is the logit link, that is, $P(\cdot)$ has the functional form given by $P_i = \left(1 + e^{-\mathbf{x}_i'\beta}\right)^{-1}$. Thus, we may proceed by maximizing $Q(\cdot)$ with respect to $\boldsymbol{\beta}$. As the double-summation in equation (2.2) is not a function of $\boldsymbol{\beta}$, we may rewrite the equation as

$$\begin{aligned} Q_\beta(\boldsymbol{\beta}|\boldsymbol{\theta}^{(t)}) = -\sum_{i=1}^{N} \mathrm{pr}(\widetilde{Y}_i = 1|\mathbf{Z}, \mathbf{x}_i, \boldsymbol{\theta}^{(t)}) \log(1 + e^{-\mathbf{x}_i'\beta}) \\ + [1 - \mathrm{pr}(\widetilde{Y}_i = 1|\mathbf{Z}, \mathbf{x}_i, \boldsymbol{\theta}^{(t)})] \log(1 + e^{\mathbf{x}_i'\beta}) + c, \end{aligned}$$

where $c$ is constant with respect to $\boldsymbol{\beta}$. If $\mathrm{pr}(\widetilde{Y}_i = 1|\cdot)$ is intractable, it is approximated according to equation (2.3). The estimate obtained from fitting a logistic regression model to the observed responses can be used as the initial estimate $\boldsymbol{\beta}^{(0)}$. The estimate is then updated via the recursion $\boldsymbol{\beta}^{(t+1)} = \arg\max_\beta Q_\beta(\boldsymbol{\beta}|\boldsymbol{\theta}^{(t)})$, where the maximization is performed using the Newton-Raphson algorithm.

The entire process is repeated until $|Q(\boldsymbol{\theta}^{(t+1)}|\boldsymbol{\theta}^{(t)}) - Q(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t-1)})| < \epsilon$, for some small $\epsilon > 0$. The vector of maximum likelihood estimates is given by $(\widehat{\mathbf{S}}_e', \widehat{\mathbf{S}}_p', \widehat{\boldsymbol{\beta}}')' \equiv (\mathbf{S}_e^{(T)'}, \mathbf{S}_p^{(T)'}, \boldsymbol{\beta}^{(T)'})'$, where $T$ is the iteration upon which the algorithm converges. By invariance, the maximum likelihood estimate of $P_i$ is given by $\widehat{P}_i = (1 + e^{-\mathbf{x}_i'\widehat{\beta}})^{-1}$.

## 2.4 Discussion

In this chapter, we used the framework of the expectation-maximization algorithm to develop a generalized linear model which is applicable for data arising from any group testing protocol, and which can be updated on a continual basis using new screening

results. This chapter lays the foundation which we continue to build upon in the coming chapters. Simulation and data analysis are deferred to the next chapter, where the estimation performance of our GLM is compared to that of boosting alternatives.

# Chapter 3

# Boosting Methods for Group Testing Data

## 3.1 Introduction

In this chapter, we apply boosting to the algorithmic framework presented in Chapter 2. This is done in two ways: the first is a straightforward extension of our generalized linear model, while the second is a nonparametric approach which uses a regression tree base learner.

## 3.2 Preliminaries

The preliminary notation and assumptions of this chapter are consistent with those established in the previous chapter; see Section 2.2 for details.

### 3.2.1 Machine Learning Concepts

Before proceeding with the methodology, we briefly describe a few machine learning concepts used within the algorithms of this chapter; namely, boosting, regression trees, and gradient boosting.

#### Boosting

Boosting is a machine learning algorithm developed in the 1990's for regression and classification problems (Schapire, 1990; Freund, 1995; Freund and Schapire, 1997). Friedman, Hastie, and Tibshirani (2000) were the first to illuminate the inner-workings of the boosting algorithm by drawing parallels to well-known statistical

methodologies. Similar to other regression techniques, boosting is inappropriate for data in which the response variable of interest is latent (e.g., due to testing error). Boosting produces a prediction model in the form of an ensemble of *weak* models or learners, often decision trees (Friedman et al., 2001). It is a form of stage-wise additive modeling, at each stage building the model in such a way as to focus on correcting the shortcomings of the model from the previous stage, with model deficiencies being quantified via a loss function. The model is fit by minimizing the expected loss over a set of training data. To do this, we begin by initializing the model at $\mathbf{F}_0$, a vector of constants of length $N$, where $N$ is the sample size. Boosting being a *greedy* algorithm will then attempt local optimization at each iteration. For $m = 1, 2, \ldots, M$, we have

$$\mathbf{F}_m = \mathbf{F}_{m-1} + \arg\min_{\mathbf{b}_m \in \mathcal{B}} \sum_{i=1}^{N} \mathcal{L}(Y_i, F_{i(m-1)} + \omega_{im} b_{im}),$$

where $\mathbf{F} \equiv (F_1, F_2, \ldots, F_N)'$, $\mathbf{b} \equiv (b_1, b_2, \ldots, b_N)'$, $F_i \equiv F(\mathbf{x}_i)$, $b_i \equiv b(\mathbf{x}_i)$, $\mathbf{x}_i$ is the vector of covariates for the $i$th individual, $\mathcal{B}$ is a class of weak learners or basis functions, $\mathcal{L}$ is the loss function, $Y_i$ is the response variable for the $i$th individual, $\omega_i$ is the weight of the basis function for the $i$th individual, and $M$ is the number of iterations. $M$ is often chosen using predictive measures such as cross validation.

### Regression Trees

The base learner used in a boosting algorithm is often chosen to be a decision tree, or more specifically, a regression tree. Trees use recursive binary partitioning to separate the predictor space, also known as the input or feature space, into disjoint regions. This is done greedily, at each step choosing the variable $x_{\cdot p}$ and the split value $c_{x_{\cdot p}}$ that best partitions the space, where $x_{ip}$ denotes the $i$th individual's value for the $p$th predictor variable. The method of least squares is used to determine these optimal values. At each step, $x_{\cdot p}$ and $c_{x_{\cdot p}}$ are given by

$$\arg\min_{x_{\cdot p}, c_{x_{\cdot p}}} \left[ \sum_{i_1^*} (y_i - \bar{y}_1)^2 + \sum_{i_2^*} (y_i - \bar{y}_2)^2 \right],$$

where $i_1^* = \{i : x_{ip} \leq c_{x \cdot p}\}$, $i_2^* = \{i : x_{ip} > c_{x \cdot p}\}$, and $\bar{y}_d$ is the mean of $\{y_i : i \in i_d^*\}$, for $d \in \{1, 2\}$.

Once a tree has been fully grown it will have $K$ terminal nodes or leaves, where $K$ is the number of distinct regions in the predictor space chosen by the model. To prevent overfitting, the optimal number of leaves is often chosen using cost-complexity pruning; however, the number of leaves is usually fixed in a boosted tree model. The predicted value for each observation in a node is equal to the average response value of observations in that node.

### Gradient Boosting

Gradient boosting is a derivative of boosting that incorporates gradient descent, an iterative optimization algorithm similar to Newton-Raphson. The goal is again to minimize a loss function $\mathcal{L}(\cdot)$ using a set of training data. For pedagogical purposes, we assume the base learner is a regression tree. We begin by initializing the model at $\mathbf{F}_0$, a vector of constants of length $N$. For $i = 1, \ldots, N$, we compute the pseudo-residual or negative gradient of the $i$th individual $G_{im} \equiv \left[\partial \mathcal{L}\left(Y_i, F_{i(m-1)}\right) \Big/ \partial F_{i(m-1)}\right]$. We then fit a $K$-node regression tree to $\{G_{im}, \mathbf{x}_i\}_{i=1}^N$, viewing $G_{\cdot m}$ as the response variable. This yields a set of disjoint regions $\{R_{km}\}_{k=1}^K$ that form a partition of the predictor space. For computational simplicity, the number of terminal nodes is the same for each iteration. Often four to eight nodes are chosen, although two nodes may be sufficient or even superior in some applications (Friedman et al., 2000). Allowing for more terminal nodes allows for more interaction between the covariates, although this does not necessarily result in a better model. Optimal step sizes $\gamma_{km}$ are found separately for each of the $K$ regions. For $k \in \{1, ..., K\}$, we have $\gamma_{km} = \arg\min_\gamma \sum_{x_i \in R_{km}} \mathcal{L}(Y_i, F_{i(m-1)} + \gamma)$, where $\gamma$ is a constant. The model is then updated as $F_{im} = F_{i(m-1)} + \tau \sum_{k=1}^K \gamma_{km} \cdot I(\mathbf{x}_i \in R_{km})$, where $\tau$ is the learning rate, a tuning parameter used to aid in convergence.

Stochastic gradient boosting, proposed by Friedman (1999), not only improves the accuracy of the gradient boosting algorithm, but also reduces its computation time. At each iteration, the regression tree is fit to a random subsample of $\{G_{im}, \mathbf{x}_i\}_{i=1}^N$ instead of to the set itself, where the sampling is performed without replacement. Typically, the size of the subsample is chosen using standard predictive measures.

## 3.3 Methodology

### 3.3.1 Boosted Generalized Linear Model

The concept of boosting (i.e., stage-wise additive modeling) can be applied to our generalized linear model (GLM) for group testing data. To allow for a direct comparison of the models, we assume the base learner is logistic regression; as such, the parametric assumptions of our GLM are retained. This alternative approach may accelerate the convergence of the maximum likelihood estimates depending on the group testing methodology to which it is applied.

Our goal is to maximize $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ with respect to $\boldsymbol{\theta}$, where $Q(\cdot)$ is given by equation (2.2). As with our GLM, $\mathbf{S}_e^{(t+1)}$ and $\mathbf{S}_p^{(t+1)}$ are updated according to equation (2.4). To maximize $Q(\cdot)$ with respect to $\mathbf{P}$, we model the log-odds of disease $\mathbf{F} = (F_1, F_2, \ldots, F_N)$, where $F_i = \log[P_i/(1-P_i)]$. We begin by reparamaterizing $Q(\cdot)$ as

$$Q_{\alpha,\beta}(\alpha, \beta|\boldsymbol{\theta}^{(t)}) = -\sum_{i=1}^{N} \text{pr}(\widetilde{Y}_i = 1|\mathbf{Z}, \mathbf{x}_i, \boldsymbol{\theta}^{(t)}) \log(1 + e^{-F_{i(m-1)} - \alpha - \beta x_{ip}}) + [1 - \text{pr}(\widetilde{Y}_i = 1|\mathbf{Z}, \mathbf{x}_i, \boldsymbol{\theta}^{(t)})] \log(1 + e^{F_{i(m-1)} + \alpha + \beta x_{ip}}) + c, \quad (3.1)$$

where $F_{i0} = F_i^{(t)}$ and $c$ is constant with respect to $(\alpha, \beta)'$; if $\text{pr}(\widetilde{Y}_i = 1|\cdot)$ is intractable, it is approximated according to equation (2.3). To annotate, the superscript $t$ reflects the cyclical process of the EM algorithm, while the subscript $m$ reflects the iterative process within each M-step.

The estimate obtained from fitting a logistic regression model to the observed responses can be used as the initial estimate $\mathbf{F}^{(0)}$. At each M-step, we iteratively

select the predictor variable $x_{\cdot p}$, $p \in \{1, 2, \ldots, r\}$, that results in the largest value of $Q_{\alpha,\beta}(\widehat{\alpha}_p, \widehat{\beta}_p|\boldsymbol{\theta}^{(t)})$, where $(\widehat{\alpha}_p, \widehat{\beta}_p)'$ are the values of $(\alpha, \beta)'$ that maximize $Q_{\alpha,\beta}(\cdot)$. Let $x_{\cdot p^*}$ denote the selected predictor variable, and $(\widehat{\alpha}_{p^*}, \widehat{\beta}_{p^*})'$ denote the corresponding maximizers. The model is then updated as $F_{im} = F_{i(m-1)} + \tau(\widehat{\alpha}_{p^*} + \widehat{\beta}_{p^*} x_{\cdot p^*})$, where $\tau$ is the learning rate. The updated log-odds of the $i$th individual is given by $F_i^{(t+1)} = F_{iM}$, where $M$ is the iteration upon which $||\mathbf{F}_m - \mathbf{F}_{(m-1)}||$ is sufficiently small.

The entire process is repeated until $|Q(\boldsymbol{\theta}^{(t+1)}|\boldsymbol{\theta}^{(t)}) - Q(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t-1)})| < \epsilon$, for some small $\epsilon > 0$. The estimate of $(\mathbf{S}_e', \mathbf{S}_p')'$ is given by $(\widehat{\mathbf{S}}_e', \widehat{\mathbf{S}}_p')' \equiv (\mathbf{S}_e^{(T)'}, \mathbf{S}_p^{(T)'})'$, and the estimate of $\boldsymbol{\beta}$ is given by the simple linear regression of $\widehat{\mathbf{F}} \equiv \mathbf{F}^{(T)}$ on $\mathbf{X}$, where $T$ is the iteration upon which the algorithm converges. Lastly, the estimate of $P_i$ is given by $\widehat{P}_i = (1 + e^{-\widehat{F}_i})^{-1}$. A summary is provided in the description entitled "Algorithm 1" in the Appendix.

### 3.3.2 Boosted Regression Trees Model

In this section, we develop a flexible, nonparametric alternative to our generalized linear models (i.e., GLM and BGLM). The proposed model is a boosting algorithm with a regression tree base learner. More specifically, it is a form of stochastic gradient boosted regression trees model. Being a nonparametric approach, we no longer place any assumptions on the parametric form of $P(\cdot)$. Instead, we directly model the log-odds of disease $\mathbf{F} = (F_1, F_2, \ldots, F_N)$, where $F_i = \log[P_i/(1 - P_i)]$.

Our goal is again to maximize $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ with respect to $\boldsymbol{\theta}$, where $Q(\cdot)$ is given by equation (2.2). Note, this is equivalent to viewing $-Q(\cdot)$ as our loss function. As before, we update $\mathbf{S}_e^{(t+1)}$ and $\mathbf{S}_p^{(t+1)}$ according to equation (2.4). However, the maximization of $Q(\cdot)$ with respect to $\mathbf{P}$ is quite different than it was for our previous models. We begin by initializing the model at $\mathbf{F}^{(0)} = \mathbf{0}$. Then, at each iteration, for $i = 1, \ldots, N$, we compute the negative gradients $G_{im} \equiv [\partial Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})/\partial F_i]$, which are

given by

$$G_{im} = \text{pr}(\widetilde{Y}_i = 1|\mathbf{Z}, \mathbf{x}_i, \boldsymbol{\theta}^{(t)}) - \frac{1}{1 + e^{-F_{i(m-1)}}}, \tag{3.2}$$

where $F_{i0} = 0$; if $\text{pr}(\widetilde{Y}_i = 1|\cdot)$ is intractable, it is approximated according to equation (2.3). Interestingly, the negative gradients are able to retain their residual-like structure despite the complex nature of group testing data.

We then fit a $K$-node regression tree to a random subsample of $\{G_{im}, \mathbf{x}_i\}_{i=1}^{N}$ of size $\eta N$, where $\eta$, the desired proportion of individuals to be included in the subsample, is often chosen between 0.40 and 0.80 (Friedman, 2002). For computational simplicity, the number of terminal nodes chosen is the same for each iteration. Trees use recursive binary partitioning to separate the predictor space into a set of disjoint regions $\{R_{km}\}_{k=1}^{K}$, where the method of least squares is used to determine the partition. Optimal step sizes are then found separately for each of these $K$ regions. A one-step Newton-Raphson algorithm yields

$$\gamma_{km} = \left( \sum_{i:\mathbf{x}_i \in R_{km}} G_{im} \right) \Bigg/ \left( \sum_{i:\mathbf{x}_i \in R_{km}} \frac{1}{1 + e^{-F_{i(m-1)}}} \times \frac{1}{1 + e^{F_{i(m-1)}}} \right)$$

as the step size for the $k$th region. The model is then updated as $F_{im} = F_{i(m-1)} + \tau \sum_{k=1}^{K} \gamma_{km} \times I(\mathbf{x}_i \in R_{km})$, where $\tau$ is the learning rate; setting $0 < \tau < 1$ may aid in convergence if the step sizes are too large.

Traditionally, the updated log-odds of the $i$th individual would be given by $F_i^{(t+1)} = F_{iM}$, where $M$ is the iteration upon which further partitioning no longer leads to a significant reduction in the least squares. However, if one sets the complexity parameter ($cp$) to 0, the algorithm will continue splitting regardless of the reduction in the least squares. An alternative update for log-odds of the $i$th individual could then be given by the average of $F_{iM}, F_{i(M+1)}, \ldots, F_{iM^*}$, where $M^*$ is an arbitrarily chosen stopping iteration. This latter approach can be shown to markedly improve the estimation performance of the algorithm.

The entire process is repeated until the empirical sampling distributions become stationary. The vector of parameter estimates $(\widehat{\mathbf{S}}_e', \widehat{\mathbf{S}}_p', \widehat{\mathbf{F}}')'$ is given by the vector

of means of these stationary distributions, and the estimate of $P_i$ is given by $\widehat{P}_i = (1+e^{-\widehat{F}_i})^{-1}$. A summary is provided in the description entitled "Algorithm 2" in the Appendix.

### 3.4 Simulation

In this section, we use simulation to demonstrate the estimation performances of our generalized linear model (GLM), boosted generalized linear model (BGLM), and boosted regression trees model (BTM). We consider two testing scenarios: Dorfman testing (DT) and array testing (AT); see Section 1.1 for details. Observations are simulated from the following population models:

$$\text{M1} : \text{logit}(P_i) = -3 + 2x_{i1} - x_{i2} \qquad \text{M2} : \text{probit}(P_i) = -2.7 + g_1(x_{i3}) + g_2(x_{i4}),$$

where $x_{i1} \sim \mathcal{N}(0,1)$, $x_{i2} \sim$ Bernoulli(0.5), $x_{i3}, x_{i4} \sim \mathcal{U}(-3,3)$,

$$g_1(x) = \exp\{-[1.2^2 I(x>0) + 1.2^{-2} I(x<0)]x^2/5\}, \text{ and}$$
$$g_2(x) = 1.5\exp\{-(x+1.5)^2\} + 0.7\exp\{-(x-1.5)^2\}.$$

The functions $g_1(\cdot)$ and $g_2(\cdot)$ were chosen to represent common nonlinear patterns, being unimodal and bimodal respectively. Both population models yield a prevalence of approximately 10%. Under each population model, we simulated 500 data sets for both DT and AT, with each data set containing $N = 5{,}000$ individuals. For $i = 1, 2, \ldots, N$, the true status of the $i$th individual is simulated according to a Bernoulli($P_i$) distribution, where $P_i$ is given by the corresponding population model. Each master pool consists of specimens from five randomly selected individuals. The observed status of the $j$th pool is simulated according to a Bernoulli$[S_{ej}\widetilde{Z}_j + (1 - S_{pj})(1-\widetilde{Z}_j)]$ distribution; to capture a mild dilution effect, we consider the use of a single screening assay with accuracies $S_{e(1)} = 0.95$ and $S_{p(1)} = 0.99$ for pools, and $S_{e(2)} = S_{p(2)} = 0.98$ for individual specimens. These assay accuracy probabilities are not assumed to be known for the model building process.

For our BTM, the tuning parameters were set as follows: $K = 2$, $\eta = 0.6$, $\tau = 1$, and $cp = 0$. In addition, a minimum split criterion was employed to help ensure the stability of the algorithm; as such, a split was only considered if at least 5% (250) of observations fell within each of the two terminal nodes. Because of the algorithm's design, specifically its stochastic nature and the choice of $cp$, setting $K > 2$ is unlikely to provide substantial improvement in estimation performance, even if the covariate space is large. The learning rate $\tau$ is set to 1 by default, and need only be modified in the presence of convergence difficulties. In practice, choosing optimal values for these parameters can be done using tuning methods such as cross validation. The iteration lengths, $T$ and $M$, serve only to ensure the convergence of the algorithm (i.e., that the distributions are stationary and that the log-odds are stable) and can be chosen by trial-and-error if need be.

In total, we consider three simulations. The first simulates data under population model M1. This simulation is designed to compare the performances of all three algorithms under optimal linearity conditions. The second simulation also generates data from population model M1, but is designed to highlight the computational efficiency of the BGLM. To do this, the GLM and BGLM are both fit with the addition of 18 random-noise covariates, each following a standard normal distribution. The final simulation generates data from population model M2. This simulation is designed to showcase the performance of the nonparametric BTM. All simulations were performed on a high-powered computing cluster.

### 3.4.1 Simulation Results

Table 3.1 displays the empirical bias and standard error estimates for the first simulation under population model M1. The estimates from all three algorithms exhibit similar variabilities and little to no bias, despite the BTM making no assumptions about the functional form of $P(\cdot)$. The GLM and BGLM have nearly identical com-

Table 3.1: Simulation study comparing the estimation performances of our generalized linear model (GLM), boosted generalized linear model (BGLM), and boosted regression trees model (BTM) under population model M1. The average bias (Bias) and estimated standard error (ESE) from 500 data sets is shown. The sample size for each data set is 5000. Dorfman testing and array testing use master pools of size five.

| Dorman Testing | GLM | | BGLM | | BTM | |
|---|---|---|---|---|---|---|
| Parameter | Bias | ESE | Bias | ESE | Bias | ESE |
| $\beta_0 = -3$ | 0.00 | 0.13 | 0.00 | 0.13 | - | - |
| $\beta_1 = 2$ | 0.02 | 0.11 | 0.02 | 0.11 | - | - |
| $\beta_2 = -1$ | −0.01 | 0.14 | −0.01 | 0.14 | - | - |
| $S_{e(1)} = 0.95$ | −0.01 | 0.03 | −0.01 | 0.03 | 0.01 | 0.04 |
| $S_{e(2)} = 0.98$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{p(1)} = 0.99$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{p(2)} = 0.98$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |
| Array Testing | GLM | | BGLM | | BTM | |
| Parameter | Bias | ESE | Bias | ESE | Bias | ESE |
| $\beta_0 = -3$ | −0.01 | 0.12 | −0.01 | 0.12 | - | - |
| $\beta_1 = 2$ | 0.01 | 0.10 | 0.01 | 0.10 | - | - |
| $\beta_2 = -1$ | 0.00 | 0.13 | 0.00 | 0.13 | - | - |
| $S_{e(1)} = 0.95$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{e(2)} = 0.98$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{p(1)} = 0.99$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{p(2)} = 0.98$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |

putation times, averaging 30 to 50 seconds per data set. However, the BTM is significantly slower, averaging 20 to 28 minutes per data set.

Table 3.2 displays the empirical bias and standard error estimates for the second simulation under population model M1. The estimates were again similar for the GLM and BGLM, and were minimally affected by the addition of the random-noise variables. However, the computation times were decidedly impacted. For DT, the GLM had an average runtime of 30 minutes, while the BGLM had an average runtime of only two minutes. For AT, the GLM averaged 2.5 hours per data set, while the BGLM averaged less than two minutes. In fact, a number of simulated data sets had

Table 3.2: Higher-dimensional simulation study comparing the estimation performances of our generalized linear model (GLM) and boosted generalized linear model (BGLM) under population model M1. The average bias (Bias) and estimated standard error (ESE) from 500 data sets is shown. For the 18 random noise variables, Bias represents the average absolute bias and ESE represents the average ESE. The sample size for each data set is 5000. Dorfman testing and array testing use master pools of size five.

| Dorman Testing | GLM | | BGLM | |
|---|---|---|---|---|
| Parameter | Bias | ESE | Bias | ESE |
| $\beta_0 = -3$ | $-0.05$ | 0.13 | $-0.05$ | 0.13 |
| $\beta_1 = 2$ | 0.04 | 0.11 | 0.04 | 0.11 |
| $\beta_2 = -1$ | $-0.02$ | 0.14 | $-0.02$ | 0.14 |
| $\beta_3 : \beta_{20} = 0$ | 0.00 | 0.06 | 0.00 | 0.06 |
| $S_{e(1)} = 0.95$ | 0.00 | 0.04 | $-0.01$ | 0.04 |
| $S_{e(2)} = 0.98$ | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{p(1)} = 0.99$ | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{p(2)} = 0.98$ | 0.00 | 0.01 | 0.00 | 0.01 |
| Array Testing | GLM | | BGLM | |
| Parameter | Bias | ESE | Bias | ESE |
| $\beta_0 = -3$ | $-0.05$ | 0.12 | $-0.05$ | 0.12 |
| $\beta_1 = 2$ | 0.03 | 0.09 | 0.03 | 0.09 |
| $\beta_2 = -1$ | $-0.02$ | 0.14 | $-0.02$ | 0.14 |
| $\beta_3 : \beta_{20} = 0$ | 0.00 | 0.06 | 0.00 | 0.06 |
| $S_{e(1)} = 0.95$ | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{e(2)} = 0.98$ | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{p(1)} = 0.99$ | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{p(2)} = 0.98$ | 0.00 | 0.01 | 0.00 | 0.01 |

to be rerun, as the GLM failed to converge. Further increasing the dimensionality of the predictor space should only heighten the disparity between the models.

Table 3.3 displays the empirical bias and standard error estimates for the simulation under population model M2. Comparing to the first simulation setting (see Table 3.1), the only noticeable change is the inflated bias and variability of the pool sensitivity estimates for all three models. Aside from this, the models all displayed similar variabilities and little to no bias. However, the assay accuracy estimates do not capture the inadequacy of the parametric models to estimate an individual's prob-

Table 3.3: Simulation study comparing the estimation performances of our generalized linear model (GLM), boosted generalized linear model (BGLM), and boosted regression trees model (BTM) under population model M2. The average bias (Bias) and estimated standard error (ESE) from 500 data sets is shown. The sample size for each data set is 5000. Dorfman testing and array testing use master pools of size five.

| Dorman Testing | GLM | | BGLM | | BTM | |
|---|---|---|---|---|---|---|
| Parameter | Bias | ESE | Bias | ESE | Bias | ESE |
| $S_{e(1)} = 0.95$ | −0.04 | 0.06 | −0.04 | 0.06 | −0.04 | 0.08 |
| $S_{e(2)} = 0.98$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.02 |
| $S_{p(1)} = 0.99$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{p(2)} = 0.98$ | 0.01 | 0.01 | 0.01 | 0.01 | 0.01 | 0.01 |
| Array Testing | GLM | | BGLM | | BTM | |
| Parameter | Bias | ESE | Bias | ESE | Bias | ESE |
| $S_{e(1)} = 0.95$ | 0.00 | 0.02 | 0.00 | 0.02 | 0.00 | 0.01 |
| $S_{e(2)} = 0.98$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{p(1)} = 0.99$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{p(2)} = 0.98$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |

ability of disease. Figure 3.1 displays the predicted probability of disease against the true probability of disease for a sample data set simulated under population model M2. On the left side are the estimates of the GLM (the GLM and BGLM provide nearly identical estimates) and on the right side are the estimates of the BTM. We clearly see from this figure that a parametric approach is not appropriate for this data, while the BTM estimates the disease probabilities quite well. The GLM and BGLM had an average runtime of approximately two minutes per data set, while the BTM averaged 25 to 30 minutes per data set.

## 3.5 Data Application

Each business day, the State Hygienic Laboratory (SHL) at the University of Iowa receives urine specimens and endocervical swab specimens from testing clinics throughout the state. The SHL tests these specimens for the presence of infection (e.g.,

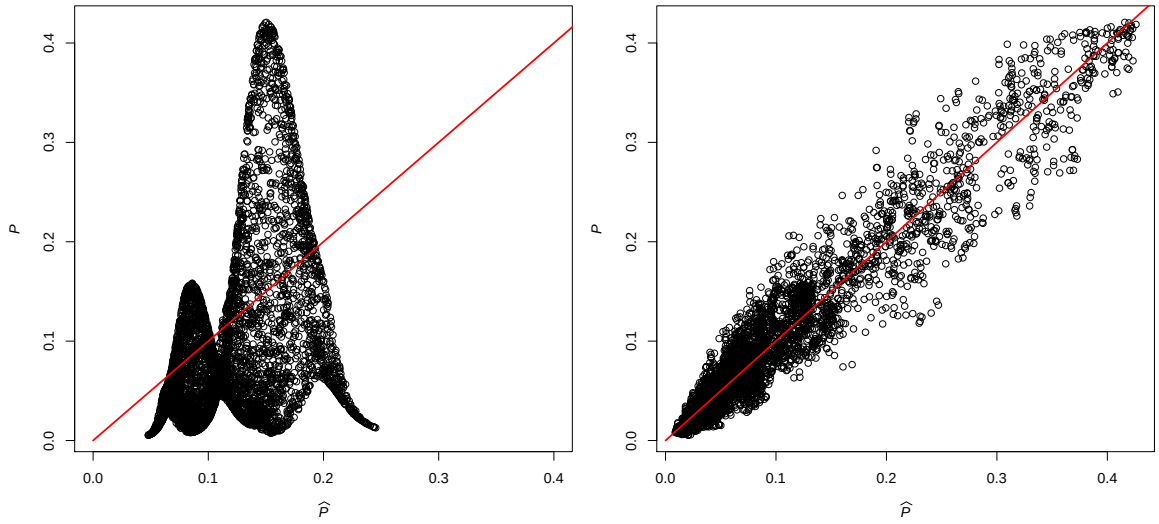


Figure 3.1: Model predicted probability of disease ($\widehat{P}$) against true probability of disease ($P$) for a sample data set simulated under population model M2. Left: GLM. Right: BTM.

gonorrhea, chlamydia, etc.) using the Aptima Combo 2 Assay (AC2A), a second generation nucleic acid amplification test (Gen-Probe, San Diego). Urine specimens are individually tested, while swab specimens are tested either using individual testing or Dorfman testing.

These data, collected in 2014, contain the observed chlamydia status, the observed gonorrhea status, and the covariate information for 13,862 female subjects (4,315 individual urine specimens, 417 individual swab specimens, 2,273 swab master pools of size 4, 12 swab master pools of size 3, and one swab master pool of size 2). In this section, chlamydia status is the sole response variable of interest. The covariates, which are observed on each individual, include age ($x_1$), whether the individual is Caucasian ($x_2 = 1$), whether the individual reports having a new sexual partner within the last 90 days ($x_3 = 1$), whether the individual reports having multiple partners within the last 90 days ($x_4 = 1$), whether the individual reports having a

partner with an STD within the last year ($x_5 = 1$), and whether the individual shows symptoms of infection ($x_6 = 1$).

We fit each of our models to the data, allowing the accuracies of the AC2A to depend upon both pool size (pooled vs unpooled) and specimen type (swab vs urine). This yields three sets of assay accuracy probabilities which our algorithms estimate along with the other model parameters. The tuning parameters of the BTM were set as follows: $K = 2$, $\eta = 0.6$, $\tau = 1$, $cp = 0$, and $minsplit = 0.05$. The model fitting process for the GLM, BGLM, and BTM took 35, 20, and 100 minutes respectively.

Table 3.4 displays the model parameter estimates for the Iowa chlamydia data. Standard error estimates (ESE) were calculated using 500 bootstrap samples; a modified bootstrap procedure was used to preserve the proportions of the key features of the data (i.e., positive pools and resulting retests, negative pools, individual swab specimens, and urine specimens). Louis' method (1982) is a standard approach for estimating standard errors in a GLM when using the EM algorithm. However, this method was not pursued as our primary focus is on the boosting algorithms.

The direction of each covariate effect is consistent with the findings of previous epidemiological studies of chlamydial infection (see, e.g., Navarro et al., 2002; Einwalter et al., 2005). All three models provide similar assay accuracy estimates, aside from the sensitivity estimate for urine specimens which is markedly higher for the BTM. However, the precision of all three urine sensitivity estimates is questionable due to their large standard errors. This lack of precision can be attributed to the testing structure chosen by the lab, which does not lend itself to precise estimation of the sensitivity of urine specimens or pooled swab specimens.

## 3.6 Discussion

In this chapter, we developed the first algorithms within the group testing literature to integrate machine learning techniques. We used simulation to compare the esti-

Table 3.4: Iowa chlamydia data parameter estimates for our generalized linear model (GLM), boosted generalized linear model (BGLM), and boosted regression trees model (BTM).

| | | GLM | | BGLM | | BTM | |
|---|---|---|---|---|---|---|---|
| Parameter | Description | Est. | ESE | Est. | ESE | Est. | ESE |
| $\beta_0$ | | $-0.803$ | 0.190 | $-0.823$ | 0.211 | - | - |
| $\beta_1$ | Age | $-0.072$ | 0.008 | $-0.071$ | 0.008 | - | - |
| $\beta_2$ | Race | $-0.353$ | 0.082 | $-0.349$ | 0.086 | - | - |
| $\beta_3$ | New partner | 0.275 | 0.071 | 0.273 | 0.067 | - | - |
| $\beta_4$ | Multiple partners | 0.333 | 0.092 | 0.332 | 0.094 | - | - |
| $\beta_5$ | Contact with STD | 1.408 | 0.120 | 1.399 | 0.117 | - | - |
| $\beta_6$ | Symptoms | 0.292 | 0.085 | 0.287 | 0.080 | - | - |
| $S_{e(1)}$ | Swab pool | 0.939 | 0.031 | 0.938 | 0.062 | 0.919 | 0.077 |
| $S_{e(2)}$ | Swab individual | 0.999 | 0.001 | 0.999 | 0.001 | 0.999 | 0.001 |
| $S_{e(3)}$ | Urine individual | 0.854 | 0.067 | 0.886 | 0.093 | 0.994 | 0.084 |
| $S_{p(1)}$ | Swab pool | 0.999 | 0.001 | 0.999 | 0.001 | 0.999 | 0.001 |
| $S_{p(2)}$ | Swab individual | 0.976 | 0.006 | 0.976 | 0.008 | 0.978 | 0.009 |
| $S_{p(3)}$ | Urine individual | 0.987 | 0.007 | 0.990 | 0.007 | 0.999 | 0.004 |

mation performances of our generalized linear model, boosted regression trees model, and boosted generalized linear model. In addition, we estimated population-level regression models for chlamydia using data obtained from the State Hygienic Laboratory in Iowa. The methods presented in this chapter are only applicable for group testing protocols which screen for a single disease. Extending our methods for use with multiplex assays is the topic of the next chapter.

# Chapter 4

# Regression Methods for Multiple-Disease Group Testing Data

## 4.1 Introduction

In this chapter, we extend our work in Chapters 2 and 3 to accommodate group testing protocols which simultaneously test for the presence of two diseases. The primary motivation behind this extension is the dual screening for chlamydia and gonorrhea provided by assays such as the Aptima 2 combo assay.

## 4.2 Preliminaries

Let $N$ be the number of individuals screened for disease, and $J$ be the number of pools used during the screening process. For $j = 1, 2, \ldots, J$, let $\mathcal{P}_j$ be the set of individuals in the $j$th pool, where we require $\cup_j \mathcal{P}_j = \{1, 2, \ldots, N\}$. Let $Z_{j1}$ and $Z_{j2}$ be the observed test results of the $j$th pool for the first and second diseases respectively, and $\widetilde{Z}_{j1} \equiv I(\sum_{i \in \mathcal{P}_j} \widetilde{Y}_{i1} > 0)$ and $\widetilde{Z}_{j2} \equiv I(\sum_{i \in \mathcal{P}_j} \widetilde{Y}_{i2} > 0)$ be the corresponding latent true statuses, where $I(\cdot)$ is the indicator function, $\widetilde{Y}_{i1}$ is the true status of the $i$th individual for the first disease, and $\widetilde{Y}_{i2}$ is the true status of the $i$th individual for the second disease. Further, let $S_{e(l)1} = \text{pr}(Z_{j1} = 1 | \widetilde{Z}_{j1} = 1)$ and $S_{e(l)2} = \text{pr}(Z_{j2} = 1 | \widetilde{Z}_{j2} = 1)$ be the sensitivities of the $l$th assay for the first and second diseases respectively, and $S_{p(l)1} = \text{pr}(Z_{j1} = 0 | \widetilde{Z}_{j1} = 0)$ and $S_{p(l)2} = \text{pr}(Z_{j2} = 0 | \widetilde{Z}_{j2} = 0)$ be its specificities, where $j \in \mathcal{M}(l)$, and $\mathcal{M}(l)$ is the set of pools such that the $l$th assay was used. We do not require these assay accuracy probabilities to be known and they may be

estimated along with the other model parameters. Additionally, this notation allows the accuracy probabilities of an assay to vary based upon the pool size.

We assume the observed testing outcomes $\mathbf{Z}$ are conditionally independent given the individual true statuses $\widetilde{\mathbf{Y}}$, and the conditional distribution $\mathbf{Z}|\widetilde{\mathbf{Y}}$ does not depend on the covariates, where $\mathbf{Z} = (\mathbf{Z}_1\ \mathbf{Z}_2)'$, $\widetilde{\mathbf{Y}} = (\widetilde{\mathbf{Y}}_1\ \widetilde{\mathbf{Y}}_2)'$, $\mathbf{Z}_1 = (Z_{11}, Z_{21}, \ldots, Z_{J1})'$, $\mathbf{Z}_2 = (Z_{12}, Z_{22}, \ldots, Z_{J2})'$, $\widetilde{\mathbf{Y}}_1 = (\widetilde{Y}_{11}, \widetilde{Y}_{21}, \ldots, \widetilde{Y}_{N1})'$, and $\widetilde{\mathbf{Y}}_2 = (\widetilde{Y}_{12}, \widetilde{Y}_{22}, \ldots, \widetilde{Y}_{N2})'$. Additionally, we assume that the assay accuracies for one disease are independent of the true status of the other disease.

With respect to their true disease statuses, we can imagine individuals falling into one of four categories: individuals who are positive for both diseases ($\widetilde{Y}_{i(1)} = 1$), individuals who are positive for the first disease but not the second ($\widetilde{Y}_{i(2)} = 1$), individuals who are positive for the second disease but not the first ($\widetilde{Y}_{i(3)} = 1$), and lastly, individuals who are negative for both diseases ($\widetilde{Y}_{i(4)} = 1$); the parentheses around the second subscript are intended to differentiate this categorical response from the binary responses $\widetilde{Y}_{i1}$ and $\widetilde{Y}_{i2}$. For our generalized linear models, we further assume the true statuses follow independent categorical distributions with probabilities $P_{id} \equiv \text{pr}(\widetilde{Y}_{i(d)} = 1|\mathbf{x}_i, \boldsymbol{\beta}) = H^{-1}(\mathbf{x}_i'\boldsymbol{\beta}_d)$, for $d = 1, 2, 3$, where $H(\cdot)$ is a monotone differentiable link function, $\mathbf{x}_i = (1, x_{i1}, x_{i2}, \ldots, x_{ir})'$ is the $(r+1)$-dimensional vector of covariates for the $i$th individual, $\boldsymbol{\beta}_d = (\beta_{0d}, \beta_{1d}, \ldots, \beta_{rd})'$ is an $(r+1)$-dimensional vector of regression parameters which enables the comparison of the $d$th category to the fourth (reference) category, and $\boldsymbol{\beta} = (\boldsymbol{\beta}_1', \boldsymbol{\beta}_2', \boldsymbol{\beta}_3')'$. The probability of an individual falling into the fourth category is given by $P_{i4} = 1 - P_{i1} - P_{i2} - P_{i3}$.

## 4.3 Methodology

### 4.3.1 Generalized Linear Model

The complete log-likelihood function for any group testing protocol in which individuals are simultaneously tested for the presence of two diseases can be written

as

$$
\begin{aligned}
l\left(\boldsymbol{\theta}\middle|\mathbf{Z},\widetilde{\mathbf{Y}},\mathbf{X}\right) = \sum_{l=1}^{L}\sum_{j\in\mathcal{M}(l)}\Bigg\{ & \widetilde{Z}_{j1}\widetilde{Z}_{j2}\log\Big[(S_{e(l)1}S_{e(l)2})^{Z_{j1}Z_{j2}}[S_{e(l)1}(1-S_{e(l)2})]^{Z_{j1}(1-Z_{j2})} \\
& \times[(1-S_{e(l)1})S_{e(l)2}]^{(1-Z_{j1})Z_{j2}}[(1-S_{e(l)1})(1-S_{e(l)2})]^{(1-Z_{j1})(1-Z_{j2})}\Big] \\
& + \widetilde{Z}_{j1}(1-\widetilde{Z}_{j2})\log\Big[[S_{e(l)1}(1-S_{p(l)2})]^{Z_{j1}Z_{j2}}(S_{e(l)1}S_{p(l)2})^{Z_{j1}(1-Z_{j2})} \\
& \times[(1-S_{e(l)1})(1-S_{p(l)2})]^{(1-Z_{j1})Z_{j2}}[(1-S_{e(l)1})S_{p(l)2}]^{(1-Z_{j1})(1-Z_{j2})}\Big] \\
& + (1-\widetilde{Z}_{j1})\widetilde{Z}_{j2}\log\Big[[(1-S_{p(l)1})S_{e(l)2}]^{Z_{j1}Z_{j2}}[(1-S_{p(l)1})(1-S_{e(l)2})]^{Z_{j1}(1-Z_{j2})} \\
& \times(S_{p(l)1}S_{e(l)2})^{(1-Z_{j1})Z_{j2}}[S_{p(l)1}(1-S_{e(l)2})]^{(1-Z_{j1})(1-Z_{j2})}\Big] \\
& + (1-\widetilde{Z}_{j1})(1-\widetilde{Z}_{j2})\log\Big[[(1-S_{p(l)1})(1-S_{p(l)2})]^{Z_{j1}Z_{j2}}[(1-S_{p(l)1})S_{p(l)2}]^{Z_{j1}(1-Z_{j2})} \\
& \times[S_{p(l)1}(1-S_{p(l)2})]^{(1-Z_{j1})Z_{j2}}(S_{p(l)1}S_{p(l)2})^{(1-Z_{j1})(1-Z_{j2})}\Big]\Bigg\} \\
+\sum_{i=1}^{N}\Big\{ & \widetilde{Y}_{i1}\widetilde{Y}_{i2}\log P_{i1}+\widetilde{Y}_{i1}(1-\widetilde{Y}_{i2})\log P_{i2}+(1-\widetilde{Y}_{i1})\widetilde{Y}_{i2}\log P_{i3}+(1-\widetilde{Y}_{i1})(1-\widetilde{Y}_{i2})\log P_{i4}\Big\},
\end{aligned}
\tag{4.1}
$$

where $\boldsymbol{\theta} = (\mathbf{S}_e', \mathbf{S}_p', \mathbf{P}')'$, $\mathbf{S}_e = (\mathbf{S}_{e1}', \mathbf{S}_{e2}')'$, $\mathbf{S}_{e1} = (S_{e(1)1}, S_{e(2)1}, \ldots, S_{e(L)1})'$, $\mathbf{S}_{e2} = (S_{e(1)2}, S_{e(2)2}, \ldots, S_{e(L)2})'$, $\mathbf{S}_p = (\mathbf{S}_{p1}', \mathbf{S}_{p2}')'$, $\mathbf{S}_{p1} = (S_{p(1)1}, S_{p(2)1}, \ldots, S_{p(L)1})'$, $\mathbf{S}_{p2} = (S_{p(1)2}, S_{p(2)2}, \ldots, S_{p(L)2})'$, $\mathbf{P} = (\mathbf{P}_1', \mathbf{P}_2', \mathbf{P}_3')'$, $\mathbf{P}_1 = (P_{11}, P_{21}, \ldots, P_{N1})'$, $\mathbf{P}_2 = (P_{12}, P_{22}, \ldots, P_{N2})'$, $\mathbf{P}_3 = (P_{13}, P_{23}, \ldots, P_{N3})'$, $\mathbf{X} = (\mathbf{x}_1\ \mathbf{x}_2\ \ldots\ \mathbf{x}_N)'$, and $L$ is the number of sets of assay accuracy probabilities. As in Chapter 2, we use the EM algorithm to calculate maximum-likelihood estimates due to the latency of the response variables.

To begin, we calculate the conditional expectation of the log-likelihood function in equation (4.1), which is given by

$$Q\left(\boldsymbol{\theta}\middle|\boldsymbol{\theta}^{(t)}\right) = \mathbb{E}\left[l\left(\boldsymbol{\theta}\middle|\mathbf{Z},\widetilde{\mathbf{Y}},\mathbf{X}\right)\middle|\mathbf{Z},\mathbf{X},\boldsymbol{\theta}^{(t)}\right] =$$

$$\sum_{l=1}^{L}\sum_{j\in\mathcal{M}(l)}\Bigg\{\underbrace{\text{pr}\left(\sum_{i\in\mathcal{P}_j}\widetilde{Y}_{i1}>0\bigcap\sum_{i\in\mathcal{P}_j}\widetilde{Y}_{i2}>0\middle|\mathbf{Z},\mathbf{X},\boldsymbol{\theta}^{(t)}\right)}_{C_{j1}}\log\Big[\left(S_{e(l)1}S_{e(l)2}\right)^{Z_{j1}Z_{j2}}$$

$$\times\left[S_{e(l)1}(1-S_{e(l)2})\right]^{Z_{j1}(1-Z_{j2})}\left[(1-S_{e(l)1})S_{e(l)2}\right]^{(1-Z_{j1})Z_{j2}}\left[(1-S_{e(l)1})(1-S_{e(l)2})\right]^{(1-Z_{j1})(1-Z_{j2})}\Big]$$

$$+\underbrace{\text{pr}\left(\sum_{i\in\mathcal{P}_j}\widetilde{Y}_{i1}>0\bigcap\sum_{i\in\mathcal{P}_j}\widetilde{Y}_{i2}=0\middle|\mathbf{Z},\mathbf{X},\boldsymbol{\theta}^{(t)}\right)}_{C_{j2}}\log\Big[\left[S_{e(l)1}(1-S_{p(l)2})\right]^{Z_{j1}Z_{j2}}$$

$$\times\left(S_{e(l)1}S_{p(l)2}\right)^{Z_{j1}(1-Z_{j2})}\left[(1-S_{e(l)1})(1-S_{p(l)2})\right]^{(1-Z_{j1})Z_{j2}}\left[(1-S_{e(l)1})S_{p(l)2}\right]^{(1-Z_{j1})(1-Z_{j2})}\Big]$$

$$+\underbrace{\text{pr}\left(\sum_{i\in\mathcal{P}_j}\widetilde{Y}_{i1}=0\bigcap\sum_{i\in\mathcal{P}_j}\widetilde{Y}_{i2}>0\middle|\mathbf{Z},\mathbf{X},\boldsymbol{\theta}^{(t)}\right)}_{C_{j3}}\log\Big[\left[(1-S_{p(l)1})S_{e(l)2}\right]^{Z_{j1}Z_{j2}}$$

$$\times\left[(1-S_{p(l)1})(1-S_{e(l)2})\right]^{Z_{j1}(1-Z_{j2})}\left(S_{p(l)1}S_{e(l)2}\right)^{(1-Z_{j1})Z_{j2}}\left[S_{p(l)1}(1-S_{e(l)2})\right]^{(1-Z_{j1})(1-Z_{j2})}\Big]$$

$$+\underbrace{\text{pr}\left(\sum_{i\in\mathcal{P}_j}\widetilde{Y}_{i1}=0\bigcap\sum_{i\in\mathcal{P}_j}\widetilde{Y}_{i2}=0\middle|\mathbf{Z},\mathbf{X},\boldsymbol{\theta}^{(t)}\right)}_{C_{j4}}\log\Big[\left[(1-S_{p(l)1})(1-S_{p(l)2})\right]^{Z_{j1}Z_{j2}}$$

$$\times\left[(1-S_{p(l)1})S_{p(l)2}\right]^{Z_{j1}(1-Z_{j2})}\left[S_{p(l)1}(1-S_{p(l)2})\right]^{(1-Z_{j1})Z_{j2}}\left(S_{p(l)1}S_{p(l)2}\right)^{(1-Z_{j1})(1-Z_{j2})}\Big]\Bigg\}$$

$$+\sum_{i=1}^{N}\Bigg\{\underbrace{\text{pr}\left(\widetilde{Y}_{i1}=1\bigcap\widetilde{Y}_{i2}=1\middle|\mathbf{Z},\mathbf{x}_i,\boldsymbol{\theta}^{(t)}\right)}_{C_{i1}}\log P_{i1}+\underbrace{\text{pr}\left(\widetilde{Y}_{i1}=1\bigcap\widetilde{Y}_{i2}=0\middle|\mathbf{Z},\mathbf{x}_i,\boldsymbol{\theta}^{(t)}\right)}_{C_{i2}}$$

$$\times\log P_{i2}+\underbrace{\text{pr}\left(\widetilde{Y}_{i1}=0\bigcap\widetilde{Y}_{i2}=1\middle|\mathbf{Z},\mathbf{x}_i,\boldsymbol{\theta}^{(t)}\right)}_{C_{i3}}\log P_{i3}$$

$$+\underbrace{\text{pr}\left(\widetilde{Y}_{i1}=0\bigcap\widetilde{Y}_{i2}=0\middle|\mathbf{Z},\mathbf{x}_i,\boldsymbol{\theta}^{(t)}\right)}_{C_{i4}}\log P_{i4}\Bigg\}, \quad (4.2)$$

where the superscript $t$ denotes an estimate from the $t$th iteration of the algorithm. To complete the E-step, we must first derive expressions for the $C_i$'s and the $C_j$'s, where, for $d = 1, 2, 3, 4$, $C_{id}$ is the conditional probability of the $i$th individual falling into the $d$th category, and similarly, $C_{jd}$ is the conditional probability of the $j$th pool

falling into the $d$th category. Unfortunately, for most group testing protocols these expectations are intractable. Therefore, we stochastically approximate these probabilities by repeatedly sampling from the conditional distributions of the individual true statuses using Markov Chain Monte Carlo. We have $\widetilde{Y}_{i1}, \widetilde{Y}_{i2} | \mathbf{Z}, \widetilde{\mathbf{Y}}_{-i}, \mathbf{X}, \boldsymbol{\theta}$ follows a categorical distribution with success probabilities $p^*_{i1}/p^*_{i\cdot}$, $p^*_{i2}/p^*_{i\cdot}$, $p^*_{i3}/p^*_{i\cdot}$, and $p^*_{i4}/p^*_{i\cdot}$, where

$$p^*_{i1} = P_{i1} \prod_{j \in \mathcal{A}_i} (S_{ej1}S_{ej2})^{Z_{j1}Z_{j2}} [S_{ej1}(1-S_{ej2})]^{Z_{j1}(1-Z_{j2})} \times [(1-S_{ej1})S_{ej2}]^{(1-Z_{j1})Z_{j2}} [(1-S_{ej1})(1-S_{ej2})]^{(1-Z_{j1})(1-Z_{j2})},$$

$$p^*_{i2} = P_{i2} \prod_{j \in \mathcal{A}_i} \Bigg[ (S_{ej1}S_{ej2})^{Z_{j1}Z_{j2}} [S_{ej1}(1-S_{ej2})]^{Z_{j1}(1-Z_{j2})} \times [(1-S_{ej1})S_{ej2}]^{(1-Z_{j1})Z_{j2}} [(1-S_{ej1})(1-S_{ej2})]^{(1-Z_{j1})(1-Z_{j2})} \Bigg]^{I(\Sigma_{i' \in \mathcal{P}_{ij}} \widetilde{Y}_{i'2} > 0)} \times \Bigg[ [S_{e(l)1}(1-S_{p(l)2})]^{Z_{j1}Z_{j2}} (S_{ej1}S_{pj2})^{Z_{j1}(1-Z_{j2})} \times [(1-S_{ej1})(1-S_{pj2})]^{(1-Z_{j1})Z_{j2}} [(1-S_{ej1})S_{pj2}]^{(1-Z_{j1})(1-Z_{j2})} \Bigg]^{I(\Sigma_{i' \in \mathcal{P}_{ij}} \widetilde{Y}_{i'2} = 0)},$$

$$p^*_{i3} = P_{i3} \prod_{j \in \mathcal{A}_i} \Bigg[ (S_{ej1}S_{ej2})^{Z_{j1}Z_{j2}} [S_{ej1}(1-S_{ej2})]^{Z_{j1}(1-Z_{j2})} \times [(1-S_{ej1})S_{ej2}]^{(1-Z_{j1})Z_{j2}} [(1-S_{ej1})(1-S_{ej2})]^{(1-Z_{j1})(1-Z_{j2})} \Bigg]^{I(\Sigma_{i' \in \mathcal{P}_{ij}} \widetilde{Y}_{i'1} > 0)} \times \Bigg[ [(1-S_{pj1})S_{ej2}]^{Z_{j1}Z_{j2}} [(1-S_{pj1})(1-S_{ej2})]^{Z_{j1}(1-Z_{j2})} \times (S_{pj1}S_{ej2})^{(1-Z_{j1})Z_{j2}} [S_{pj1}(1-S_{ej2})]^{(1-Z_{j1})(1-Z_{j2})} \Bigg]^{I(\Sigma_{i' \in \mathcal{P}_{ij}} \widetilde{Y}_{i'1} = 0)},$$

$$
\begin{aligned}
p_{i4}^{*} = P_{i4} \prod_{j \in \mathcal{A}_i} \Big[ & (S_{ej1}S_{ej2})^{Z_{j1}Z_{j2}} [S_{ej1}(1-S_{ej2})]^{Z_{j1}(1-Z_{j2})} \\
& \times [(1-S_{ej1})S_{ej2}]^{(1-Z_{j1})Z_{j2}} [(1-S_{ej1})(1-S_{ej2})]^{(1-Z_{j1})(1-Z_{j2})} \Big]^{I(\Sigma_{i' \in \mathcal{P}_{ij}} \widetilde{Y}_{i'1} > 0 \bigcap \Sigma_{i' \in \mathcal{P}_{ij}} \widetilde{Y}_{i'2} > 0)} \\
& \times \Big[ [S_{ej1}(1-S_{pj2})]^{Z_{j1}Z_{j2}} (S_{ej1}S_{pj2})^{Z_{j1}(1-Z_{j2})} \\
& \times [(1-S_{ej1})(1-S_{pj2})]^{(1-Z_{j1})Z_{j2}} [(1-S_{ej1})S_{pj2}]^{(1-Z_{j1})(1-Z_{j2})} \Big]^{I(\Sigma_{i' \in \mathcal{P}_{ij}} \widetilde{Y}_{i'1} > 0 \bigcap \Sigma_{i' \in \mathcal{P}_{ij}} \widetilde{Y}_{i'2} = 0)} \\
& \times \Big[ [(1-S_{pj1})S_{ej2}]^{Z_{j1}Z_{j2}} [(1-S_{pj1})(1-S_{ej2})]^{Z_{j1}(1-Z_{j2})} \\
& \times (S_{pj1}S_{ej2})^{(1-Z_{j1})Z_{j2}} [S_{pj1}(1-S_{ej2})]^{(1-Z_{j1})(1-Z_{j2})} \Big]^{I(\Sigma_{i' \in \mathcal{P}_{ij}} \widetilde{Y}_{i'1} = 0 \bigcap \Sigma_{i' \in \mathcal{P}_{ij}} \widetilde{Y}_{i'2} > 0)} \\
& \times \Big[ [(1-S_{pj1})(1-S_{pj2})]^{Z_{j1}Z_{j2}} [(1-S_{pj1})S_{pj2}]^{Z_{j1}(1-Z_{j2})} \\
& \times [S_{pj1}(1-S_{pj2})]^{(1-Z_{j1})Z_{j2}} (S_{pj1}S_{pj2})^{(1-Z_{j1})(1-Z_{j2})} \Big]^{I(\Sigma_{i' \in \mathcal{P}_{ij}} \widetilde{Y}_{i'1} = 0 \bigcap \Sigma_{i' \in \mathcal{P}_{ij}} \widetilde{Y}_{i'2} = 0)}, \quad (4.3)
\end{aligned}
$$

$p_{i\cdot}^{*} = p_{i1}^{*} + p_{i2}^{*} + p_{i3}^{*} + p_{i4}^{*}$, $\widetilde{\mathbf{Y}}_{-i} = \{\widetilde{Y}_{11}, \widetilde{Y}_{12}, \ldots, \widetilde{Y}_{(i-1)1}, \widetilde{Y}_{(i-1)2}, \widetilde{Y}_{(i+1)1}, \widetilde{Y}_{(i+1)2}, \ldots, \widetilde{Y}_{N1}, \widetilde{Y}_{N2}\}$, $\mathcal{A}_i = \{j : i \in \mathcal{P}_j\}$, $S_{ej1}$ and $S_{ej2}$ are the sensitivities for the first and second disease respectively of the assay used to test the $j$th pool, $S_{pj1}$ and $S_{pj1}$ are the specificities for the first and second disease respectively of the assay used to test the $j$th pool, and $\mathcal{P}_{ij} = \{i' \in \mathcal{P}_j : i' \neq i\}$. The estimate of $C_{id}$ is given by the proportion of the imputed statuses of the $i$th individual which fall into the $d$th category. Similarly, the estimate of $C_{jd}$ is given by the proportion of the imputed statuses of the $j$th pool which fall into the $d$th category.

The M-step consists of maximizing the conditional expectation in equation (4.2) with respect to $\boldsymbol{\theta}$. Maximizing $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ directly with respect to $\mathbf{S}_e$ and $\mathbf{S}_p$ yields the following closed-form solutions:

$$
\begin{aligned}
S_{e(l)1}^{(t+1)} &= \frac{\sum_{j \in \mathcal{M}(l)} (C_{j1}^{*} + C_{j2}^{*}) Z_{j1}}{\sum_{j \in \mathcal{M}(l)} C_{j1}^{*} + C_{j2}^{*}} \qquad S_{p(l)1}^{(t+1)} = \frac{\sum_{j \in \mathcal{M}(l)} (C_{j3}^{*} + C_{j4}^{*})(1 - Z_{j1})}{\sum_{j \in \mathcal{M}(l)} C_{j3}^{*} + C_{j4}^{*}} \\
S_{e(l)2}^{(t+1)} &= \frac{\sum_{j \in \mathcal{M}(l)} (C_{j1}^{*} + C_{j3}^{*}) Z_{j2}}{\sum_{j \in \mathcal{M}(l)} C_{j1}^{*} + C_{j3}^{*}} \qquad S_{p(l)2}^{(t+1)} = \frac{\sum_{j \in \mathcal{M}(l)} (C_{j2}^{*} + C_{j4}^{*})(1 - Z_{j2})}{\sum_{j \in \mathcal{M}(l)} C_{j2}^{*} + C_{j4}^{*}}, \quad (4.4)
\end{aligned}
$$

for $l \in \{1, 2, \ldots, L\}$, where $C_{jd}^{*}$ is the stochastic approximation of $C_{jd}$, for $d = 1, 2, 3, 4$.

To complete the M-step, it still remains to maximize $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ with respect to $\mathbf{P}$. For ease of explication, we proceed using a polytomous, or multi-category, logistic regression model, that is, for $d = 1, 2, 3$, $H(P_{\cdot d})$ is the logit which compares the $d$th category to the reference category. Thus, we may proceed by maximizing $Q(\cdot)$ with respect to the logistic parameter vector $\boldsymbol{\beta}$. As the double-summation in equation (4.2) is not a function of $\boldsymbol{\beta}$, we may rewrite the equation as

$$Q_{\beta}(\boldsymbol{\beta}|\boldsymbol{\theta}^{(t)}) = \sum_{i=1}^{N} C_{i1}^{*}\left[\mathbf{x}_i'\boldsymbol{\beta}_1 - \log\left(1 + \sum_{d=1}^{3} e^{\mathbf{x}_i'\boldsymbol{\beta}_d}\right)\right] + C_{i2}^{*}\left[\mathbf{x}_i'\boldsymbol{\beta}_2 - \log\left(1 + \sum_{d=1}^{3} e^{\mathbf{x}_i'\boldsymbol{\beta}_d}\right)\right] + C_{i3}^{*}\left[\mathbf{x}_i'\boldsymbol{\beta}_3 - \log\left(1 + \sum_{d=1}^{3} e^{\mathbf{x}_i'\boldsymbol{\beta}_d}\right)\right] - C_{i4}^{*}\log\left(1 + \sum_{d=1}^{3} e^{\mathbf{x}_i'\boldsymbol{\beta}_d}\right) + c,$$

where $c$ is constant with respect to $\boldsymbol{\beta}$ and $C_{id}^{*}$ is the stochastic approximation of $C_{id}$, for $d = 1, 2, 3, 4$. The estimate obtained from fitting a polytomous logistic regression model to the observed responses can be used as the initial estimate $\boldsymbol{\beta}^{(0)}$. The estimate is then updated via the recursion $\boldsymbol{\beta}^{(t+1)} = \arg\max_{\beta} Q_{\beta}(\boldsymbol{\beta}|\boldsymbol{\theta}^{(t)})$, where the maximization is performed using the Newton-Raphson algorithm.

The entire process is repeated until $|Q(\boldsymbol{\theta}^{(t+1)}|\boldsymbol{\theta}^{(t)}) - Q(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t-1)})| < \epsilon$, for some small $\epsilon > 0$. The vector of maximum likelihood estimates is given by $(\widehat{\mathbf{S}}_e', \widehat{\mathbf{S}}_p', \widehat{\boldsymbol{\beta}}')' \equiv (\mathbf{S}_e^{(T)'}, \mathbf{S}_p^{(T)'}, \boldsymbol{\beta}^{(T)'})'$, and for $d = 1, 2, 3$, the estimate of $P_{id}$ is given by $\widehat{P}_{id} = e^{\mathbf{x}_i'\widehat{\boldsymbol{\beta}}_d}/(1 + \sum_{b=1}^{3} e^{\mathbf{x}_i'\widehat{\boldsymbol{\beta}}_b})$, where $T$ is the iteration upon which the algorithm converges.

### 4.3.2 Boosted Generalized Linear Model

The concept of boosting can be applied to our generalized linear model through the use of a polytomous logistic regression base learner. This base learner implies that the parametric assumptions of our generalized linear model are retained. Our goal is again to maximize $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ with respect to $\boldsymbol{\theta}$, where $Q(\cdot)$ is given by equation (4.2). As before, we update $\mathbf{S}_e^{(t+1)}$ and $\mathbf{S}_p^{(t+1)}$ according to equation (4.4). To maximize $Q(\cdot)$ with respect to $\mathbf{P}$, we model the log-odds $\mathbf{F} = (\mathbf{F}_1, \mathbf{F}_2, \mathbf{F}_3)$, where

$\mathbf{F}_d = (F_{1d}, F_{2d}, \ldots, F_{Nd})'$, and $F_{id} = \log[P_{id}/P_{i4}]$ is the log-odds between the $d$th disease category and the reference category, for $d = 1, 2, 3$, and for $i = 1, 2, \ldots, N$.

We begin by reparamaterizing $Q(\cdot)$ as

$$\begin{aligned} Q_{\boldsymbol{\alpha},\boldsymbol{\lambda}}(\boldsymbol{\alpha}, \boldsymbol{\lambda}|\boldsymbol{\theta}^{(t)}) = \sum_{i=1}^{N} C_{i1}^* \left[ F_{i(m-1)1} + \alpha_1 + \lambda_1 x_{ip} - \log\left(1 + \sum_{b=1}^{3} e^{F_{i(m-1)b} + \alpha_b + \lambda_b x_{ip}}\right)\right] \\ + C_{i2}^* \left[ F_{i(m-1)2} + \alpha_2 + \lambda_2 x_{ip} - \log\left(1 + \sum_{b=1}^{3} e^{F_{i(m-1)b} + \alpha_b + \lambda_b x_{ip}}\right)\right] \\ + C_{i3}^* \left[ F_{i(m-1)3} + \alpha_3 + \lambda_3 x_{ip} - \log\left(1 + \sum_{b=1}^{3} e^{F_{i(m-1)b} + \alpha_b + \lambda_b x_{ip}}\right)\right] \\ - C_{i4}^* \log\left(1 + \sum_{b=1}^{3} e^{F_{i(m-1)b} + \alpha_b + \lambda_b x_{ip}}\right) + c, \quad (4.5) \end{aligned}$$

where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3)'$, $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \lambda_3)'$, $F_{i0d} = F_{id}^{(t)}$, $C_{id}^*$ is the stochastic approximation of $C_{id}$, and $c$ is constant with respect to $(\boldsymbol{\alpha}', \boldsymbol{\lambda}')'$. The estimate obtained from fitting a polytomous logistic regression model to the observed responses can be used as the initial estimate $\mathbf{F}^{(0)}$. At each iteration, we iteratively select the predictor variable $x_{\cdot p}$, $p \in \{1, 2, \ldots, r\}$, that results in the largest value of $Q_{\boldsymbol{\alpha},\boldsymbol{\lambda}}(\widehat{\boldsymbol{\alpha}}_p, \widehat{\boldsymbol{\lambda}}_p|\boldsymbol{\theta}^{(t)})$, where $(\widehat{\boldsymbol{\alpha}}_p', \widehat{\boldsymbol{\lambda}}_p')'$ are the values of $(\boldsymbol{\alpha}', \boldsymbol{\lambda}')'$ that maximize $Q_{\boldsymbol{\alpha},\boldsymbol{\lambda}}(\cdot)$. Let $x_{\cdot p^*}$ denote the selected predictor variable, and $(\widehat{\boldsymbol{\alpha}}_{p^*}', \widehat{\boldsymbol{\lambda}}_{p^*}')'$ denote the corresponding maximizers. The model for the $d$th log-odds is then updated as $F_{imd} = F_{i(m-1)d} + \tau(\widehat{\alpha}_{p^*d} + \widehat{\lambda}_{p^*d} x_{\cdot p^*})$, where $\tau$ is the learning rate. The updated log-odds of the $i$th individual is given by $F_{id}^{(t+1)} = F_{iMd}$, where $M$ is the iteration upon which $||\mathbf{F}_m - \mathbf{F}_{(m-1)}||$ is sufficiently small.

The entire process is repeated until $|Q(\boldsymbol{\theta}^{(t+1)}|\boldsymbol{\theta}^{(t)}) - Q(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t-1)})| < \epsilon$, for some small $\epsilon > 0$. The estimate of $(\mathbf{S}_e', \mathbf{S}_p')'$ is given by $(\widehat{\mathbf{S}}_e', \widehat{\mathbf{S}}_p')' \equiv (\mathbf{S}_e^{(T)'}, \mathbf{S}_p^{(T)'})'$, and the estimate of $\boldsymbol{\beta}$ is given by the multivariate linear regression of $\widehat{\mathbf{F}}_1 \equiv \mathbf{F}_1^{(T)}$, $\widehat{\mathbf{F}}_2 \equiv \mathbf{F}_2^{(T)}$, and $\widehat{\mathbf{F}}_3 \equiv \mathbf{F}_3^{(T)}$ on $\mathbf{X}$, where $T$ is the iteration upon which the algorithm converges. Lastly, the estimate of $P_{id}$ is given by $\widehat{P}_{id} = e^{\widehat{F}_{id}}/(1 + \sum_{b=1}^{3} e^{\widehat{F}_{ib}})$. A summary is provided in the description entitled "Algorithm 3" in the Appendix.

### 4.3.3 Boosted Regression Trees Model

In this section, we generalize our boosted regression trees algorithm to accommodate group testing protocols which simultaneously test for the presence of two diseases. Boosting is a nonparametric approach and thus places no assumptions on the parametric form of $P(\cdot)$. Rather, the log-odds $\mathbf{F} = (\mathbf{F}_1, \mathbf{F}_2, \mathbf{F}_3)$ is modeled directly, where $\mathbf{F}_d = (F_{1d}, F_{2d}, \ldots, F_{Nd})'$, and $F_{id} = \log[P_{id}/P_{i4}]$ is the log-odds between the $d$th disease category and the reference category, for $d = 1, 2, 3$, and for $i = 1, 2, \ldots, N$.

We again seek to maximize $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ with respect to $\boldsymbol{\theta}$, where $Q(\cdot)$ is given by equation (4.2). As before, we update $\mathbf{S}_e^{(t+1)}$ and $\mathbf{S}_p^{(t+1)}$ according to equation (4.4). To maximize $Q(\cdot)$ with respect to $\mathbf{P}$, we begin by initializing the model at $\mathbf{F}^{(0)} = \mathbf{0}$. Then, at each iteration, for $d = 1, 2, 3$, and for $i = 1, 2, \ldots, N$, we compute the negative gradients $G_{imd} \equiv [\partial Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})/\partial F_{id}]$, which are given by

$$G_{imd} = C_{id}^* - \frac{e^{F_{i(m-1)d}}}{1 + \sum_{b=1}^{3} e^{F_{i(m-1)b}}}, \tag{4.6}$$

where $F_{i0d} = 0$ and $C_{id}^*$ is the stochastic approximation of $C_{id}$. For each set of negative gradients we fit a $K$-node regression tree to a random subsample of $\{G_{imd}, \mathbf{x}_i\}_{i=1}^N$ of size $\eta N$, where $\eta$ is the desired proportion of individuals to be included in the subsample; the same set of individuals is used for each subsample. Trees use recursive binary partitioning to separate the predictor space into a set of disjoint regions $\{R_{kmd}\}_{k=1}^K$, where the method of least squares is used to determine the partition. Optimal step sizes are then found separately for each of these $K$ regions. A one-step Newton-Raphson algorithm yields

$$\gamma_{kmd} = \left( \sum_{i:\mathbf{x}_i \in R_{kmd}} G_{imd} \right) \Bigg/ \left( \sum_{i:\mathbf{x}_i \in R_{kmd}} \frac{e^{F_{i(m-1)d}}}{1 + \sum_{b=1}^{3} e^{F_{i(m-1)b}}} \times \left( 1 - \frac{e^{F_{i(m-1)d}}}{1 + \sum_{b=1}^{3} e^{F_{i(m-1)b}}} \right) \right)$$

as the step size for the $k$th region of the $d$th *log-odds*. The model is then updated as $F_{imd} = F_{i(m-1)d} + \tau \sum_{k=1}^{K} \gamma_{kmd} \times I(\mathbf{x}_i \in R_{kmd})$, where $\tau$ is the learning rate. Setting the complexity parameter ($cp$) to 0 allows the algorithm to continue splitting regardless

of the reduction in the least squares. The updated log-odds of the $i$th individual is then given by the average of $F_{iMd}, F_{i(M+1)d}, \ldots, F_{iM^*d}$, where $M^*$ is an arbitrarily chosen stopping iteration.

The entire process is repeated until the empirical sampling distributions become stationary. The vector of parameter estimates $(\widehat{\mathbf{S}}_e', \widehat{\mathbf{S}}_p', \widehat{\mathbf{F}}')'$ is given by the vector of means of these stationary distributions, and the estimate of $P_{id}$ is given by $\widehat{P}_{id} = e^{\widehat{F}_{id}}/(1 + \sum_{b=1}^{3} e^{\widehat{F}_{ib}})$. A summary is provided in the description entitled "Algorithm 4" in the Appendix.

### 4.4 Simulation

In this section, we present simulation evidence of the estimation performances of our generalized linear model (GLM), boosted generalized linear model (BGLM), and boosted regression trees model (BTM). We consider both Dorfman testing (DT) and array testing (AT) baseline protocols; see Section 1.1 for details. However, as we are testing for two diseases, further discussion of these protocols is required. For DT, if a master pool tests positively for either of the two diseases, individual retests are performed on its contributing specimens during the second stage of testing. For AT, if a column pool and a row pool both test positively for the same disease, a separate aliquot of the specimen which contributed to both pools is individually tested. In addition, if a column pool tests positively for a specific disease, but all row pools test negatively for that same disease, each specimen that contributed to the column pool is individually tested. Likewise, if a row pool tests positively for a specific disease, but all column pools test negatively for that same disease, each specimen that contributed to the row pool is individually tested.

Observations are simulated from the following population model:

$$\text{M1} : \text{logit}(P_{id}) = \mathbf{x}_i' \boldsymbol{\beta}_d,$$

where, for $d = 1, 2, 3$, logit$(P_{id})$ is the logit of the $d$th category to the fourth (reference) category, $\mathbf{x}_i = (1, x_{i1}, x_{i2})'$, $x_{i1} \sim \mathcal{N}(0, 1)$, $x_{i2} \sim$ Bernoulli(0.5), $\boldsymbol{\beta}_1 = (-3.5, 2, -1)'$, $\boldsymbol{\beta}_2 = (-4, 1, -0.5)'$, and $\boldsymbol{\beta}_3 = (-4.5, 1, -0.5)'$.

This population model yields prevalences of approximately 8.5% and 7.5% for the first and second diseases respectively. To reflect a high rate of co-infection, both diseases are present within approximately 6.5% of the population (i.e., $P_4 \approx 0.065$); that is, when one disease is present, it is more likely for the other disease to be present as well. We simulated 500 data sets for both DT and AT, with each data set containing $N = 5{,}000$ individuals. For $i = 1, 2, \ldots, N$, the true statuses of the $i$th individual are simulated according to a categorical$(P_{id})$ distribution, where $P_{id}$ is given by the population model. Each master pool consists of specimens from five randomly selected individuals. The observed status of the $j$th pool for the $q$th disease is simulated according to a Bernoulli$[S_{ejq}\widetilde{Z}_{jq} + (1 - S_{pjq})(1 - \widetilde{Z}_{jq})]$ distribution, where we consider the use of a single screening assay with accuracies $S_{e(1)1} = S_{e(1)2} = 0.95$ and $S_{p(1)1} = S_{p(1)2} = 0.99$ for pools, and $S_{e(2)1} = S_{e(2)2} = S_{p(2)1} = S_{p(2)2} = 0.98$ for individual specimens. These assay accuracy probabilities are not assumed to be known for the model building process. The tuning parameters of the BTM were set as follows: $K = 2$, $\eta = 0.6$, $\tau = 1$, $cp = 0$, and $minsplit = 0.05$.

### 4.4.1 Simulation Results

Table 4.1 displays the empirical bias and standard error estimates for the simulation study under population model M1. The BGLM displays slightly higher bias in its regression coefficient estimates as compared to the GLM. For DT, the assay accuracy estimates of the BTM are both less accurate and less precise than those of the parametric models. However, for AT, the discrepancy between the models is negligible, as all three models exhibit similar variabilities and little to no bias. Although the models perform equally well, the parametric models are perfectly specified (i.e., a

Table 4.1: Simulation study comparing the estimation performances of our generalized linear model (GLM), boosted generalized linear model (BGLM), and boosted regression trees model (BTM) under population model M1. The average bias (Bias) and estimated standard error (ESE) from 500 data sets is shown. The sample size for each data set is 5000. Dorfman testing and array testing use master pools of size five.

| | | Dorfman Testing | | | | | | Array Testing | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | GLM | | BGLM | | BTM | | GLM | | BGLM | | BTM | |
| Parameter | Description | Bias | ESE | Bias | ESE | Bias | ESE | Bias | ESE | Bias | ESE | Bias | ESE |
| $\beta_{01} = -3.5$ | - | $-0.01$ | 0.12 | 0.04 | 0.12 | - | - | $-0.02$ | 0.13 | 0.02 | 0.12 | - | - |
| $\beta_{11} = 2$ | - | 0.00 | 0.09 | $-0.05$ | 0.09 | - | - | 0.02 | 0.09 | $-0.03$ | 0.09 | - | - |
| $\beta_{21} = -1$ | - | 0.00 | 0.13 | 0.01 | 0.13 | - | - | 0.00 | 0.14 | 0.01 | 0.14 | - | - |
| $\beta_{02} = -4$ | - | $-0.04$ | 0.20 | 0.10 | 0.17 | - | - | $-0.02$ | 0.17 | 0.03 | 0.16 | - | - |
| $\beta_{12} = 1$ | - | 0.01 | 0.15 | $-0.24$ | 0.12 | - | - | 0.00 | 0.13 | $-0.08$ | 0.12 | - | - |
| $\beta_{22} = -0.5$ | - | 0.01 | 0.25 | 0.05 | 0.25 | - | - | $-0.01$ | 0.25 | 0.01 | 0.24 | - | - |
| $\beta_{03} = -4.5$ | - | $-0.05$ | 0.27 | 0.13 | 0.22 | - | - | $-0.04$ | 0.24 | 0.02 | 0.22 | - | - |
| $\beta_{13} = 1$ | - | 0.00 | 0.21 | $-0.32$ | 0.14 | - | - | 0.01 | 0.18 | $-0.10$ | 0.16 | - | - |
| $\beta_{23} = -0.5$ | - | 0.01 | 0.34 | 0.05 | 0.34 | - | - | 0.00 | 0.33 | 0.02 | 0.33 | - | - |
| $S_{e(1)1} = 0.95$ | $\text{Pool}_1$ | 0.00 | 0.01 | 0.00 | 0.01 | $-0.17$ | 0.10 | 0.00 | 0.01 | 0.00 | 0.01 | $-0.02$ | 0.01 |
| $S_{e(2)1} = 0.98$ | $\text{Individual}_1$ | 0.00 | 0.01 | 0.00 | 0.01 | $-0.02$ | 0.03 | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{p(1)1} = 0.99$ | $\text{Pool}_1$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.01 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 | 0.01 | 0.01 |
| $S_{p(2)1} = 0.98$ | $\text{Individual}_1$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.02 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 | 0.01 | 0.01 |
| $S_{e(1)2} = 0.95$ | $\text{Pool}_2$ | 0.00 | 0.01 | 0.00 | 0.01 | $-0.19$ | 0.12 | 0.00 | 0.01 | 0.00 | 0.01 | $-0.01$ | 0.01 |
| $S_{e(2)2} = 0.98$ | $\text{Individual}_2$ | 0.00 | 0.01 | 0.00 | 0.01 | $-0.03$ | 0.03 | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{p(1)2} = 0.99$ | $\text{Pool}_2$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.01 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 | 0.01 | 0.01 |
| $S_{p(2)2} = 0.98$ | $\text{Individual}_2$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.02 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 | 0.01 | 0.01 |

logistic model fit to a logistic population), whereas the BTM makes no assumptions about the functional form of $P(\cdot)$. The GLM and BGLM had an average runtime of between one and two minutes per data set, while the BTM averaged 13 to 14 minutes per data set.

## 4.5 Data Application

In this section, we analyze the clinical screening data provided by the State Hygienic Laboratory at the University of Iowa. These data contain the observed chlamydia status, the observed gonorrhea status, and the covariate information for 13,862 female subjects. The covariates, which are observed on each individual, include age ($x_1$), whether the individual is Caucasian ($x_2 = 1$), whether the individual reports having a new sexual partner within the last 90 days ($x_3 = 1$), whether the individual reports having multiple partners within the last 90 days ($x_4 = 1$), whether the individual reports having a partner with an STD within the last year ($x_5 = 1$), and whether the individual shows symptoms of infection ($x_6 = 1$); see Section 3.5 for more details.

We fit each of our models to the data, allowing the assay accuracies to depend upon both pool size (pooled vs unpooled) and specimen type (swab vs urine). This yields three sets of assay accuracy probabilities which our algorithms estimate along with the other model parameters. The tuning parameters of the BTM were set as follows: $K = 2$, $\eta = 0.6$, $\tau = 1$, $cp = 0$, and $minsplit = 0.05$. The model fitting process for the GLM, BGLM, and BTM took 49, 28, and 138 minutes respectively.

Table 4.2 displays the model parameter estimates for the Iowa screening data. Standard error estimates (ESE) were calculated using 500 bootstrap samples; a modified bootstrap procedure was used to preserve the proportions of the key features of the data (i.e., positive pools and resulting retests, negative pools, individual swab specimens, and urine specimens). The first block in the table contains the regression-coefficient estimates for comparing the log-odds of an individual having both diseases

Table 4.2: Iowa chlamydia and gonorrhea data parameter estimates for our generalized linear model (GLM), boosted generalized linear model (BGLM), and boosted regression trees model (BTM).

| | | GLM | | BGLM | | BTM | |
|---|---|---|---|---|---|---|---|
| Parameter | Description | Est. | ESE | Est. | ESE | Est. | ESE |
| $\beta_{01}$ | | $-4.227$ | 0.624 | $-4.253$ | 0.590 | - | - |
| $\beta_{11}$ | Age | $-0.033$ | 0.021 | $-0.033$ | 0.020 | - | - |
| $\beta_{21}$ | Race | $-1.056$ | 0.265 | $-1.044$ | 0.269 | - | - |
| $\beta_{31}$ | New partner | 0.006 | 0.267 | 0.010 | 0.277 | - | - |
| $\beta_{41}$ | Multiple partners | 0.805 | 0.308 | 0.802 | 0.328 | - | - |
| $\beta_{51}$ | Contact with STD | 1.823 | 0.349 | 1.909 | 0.345 | - | - |
| $\beta_{61}$ | Symptoms | 0.567 | 0.280 | 0.563 | 0.275 | - | - |
| $\beta_{02}$ | | $-0.928$ | 0.185 | $-1.112$ | 0.162 | - | - |
| $\beta_{12}$ | Age | $-0.073$ | 0.007 | $-0.066$ | 0.006 | - | - |
| $\beta_{22}$ | Race | $-0.296$ | 0.087 | $-0.275$ | 0.079 | - | - |
| $\beta_{32}$ | New partner | 0.281 | 0.066 | 0.311 | 0.063 | - | - |
| $\beta_{42}$ | Multiple partners | 0.297 | 0.097 | 0.286 | 0.091 | - | - |
| $\beta_{52}$ | Contact with STD | 1.385 | 0.112 | 1.495 | 0.125 | - | - |
| $\beta_{62}$ | Symptoms | 0.259 | 0.082 | 0.246 | 0.073 | - | - |
| $\beta_{03}$ | | $-4.849$ | 0.522 | $-4.888$ | 0.482 | - | - |
| $\beta_{13}$ | Age | $-0.012$ | 0.016 | $-0.012$ | 0.015 | - | - |
| $\beta_{23}$ | Race | $-0.611$ | 0.290 | $-0.623$ | 0.277 | - | - |
| $\beta_{33}$ | New partner | 0.038 | 0.308 | 0.117 | 0.293 | - | - |
| $\beta_{43}$ | Multiple partners | 0.304 | 0.399 | 0.290 | 0.367 | - | - |
| $\beta_{53}$ | Contact with STD | 2.414 | 0.330 | 2.453 | 0.291 | - | - |
| $\beta_{63}$ | Symptoms | 0.322 | 0.278 | 0.324 | 0.262 | - | - |
| $S_{e(1)1}$ | Swab pool | 0.999 | 0.003 | 0.998 | 0.003 | 0.999 | 0.001 |
| $S_{e(2)1}$ | Swab individual | 0.999 | 0.001 | 0.999 | 0.001 | 0.999 | 0.001 |
| $S_{e(3)1}$ | Urine individual | 0.959 | 0.065 | 0.943 | 0.053 | 0.999 | 0.059 |
| $S_{p(1)1}$ | Swab pool | 0.999 | 0.001 | 0.999 | 0.001 | 0.999 | 0.001 |
| $S_{p(2)1}$ | Swab individual | 0.978 | 0.005 | 0.980 | 0.005 | 0.980 | 0.005 |
| $S_{p(3)1}$ | Urine individual | 0.993 | 0.006 | 0.991 | 0.005 | 0.998 | 0.004 |
| $S_{e(1)2}$ | Swab pool | 0.999 | 0.001 | 0.999 | 0.001 | 0.999 | 0.001 |
| $S_{e(2)2}$ | Swab individual | 0.999 | 0.001 | 0.999 | 0.001 | 0.999 | 0.001 |
| $S_{e(3)2}$ | Urine individual | 0.999 | 0.111 | 0.997 | 0.092 | 0.999 | 0.139 |
| $S_{p(1)2}$ | Swab pool | 0.999 | 0.001 | 0.999 | 0.001 | 0.999 | 0.001 |
| $S_{p(2)2}$ | Swab individual | 0.999 | 0.001 | 0.999 | 0.001 | 0.999 | 0.001 |
| $S_{p(3)2}$ | Urine individual | 0.999 | 0.001 | 0.999 | 0.001 | 0.999 | 0.001 |

to having neither disease, the second block contains the regression-coefficient estimates for comparing the log-odds of an individual having chlamydia but not gonorrhea to having neither disease, and the third block contains the regression-coefficient estimates for comparing the log-odds of an individual having gonorrhea but not chlamydia to having neither disease. The fourth and fifth blocks contain the assay accuracy estimates for detecting chlamydia, and the final two blocks contain the assay accuracy estimates for detecting gonorrhea.

We can see from the table, that the estimates in the first and third blocks have relatively high standard errors. This is due to a sparsity of individuals falling into these categories. The estimates from the second block (i.e., chl. $= 1$, gon. $= 0$ vs chl. $= 0$, gon. $= 0$) are quite similar to the estimates in Chapter 3, where chlamydia was the sole response variable of interest. The inclusion of gonorrhea seems to have stabilized the chlamydia sensitivity estimates for pooled swab specimens, which are now higher for all three models than they were in Chapter 3. However, the chlamydia sensitivity estimates for urine specimens remain highly imprecise. The standard errors of the assay accuracy estimates for detecting gonorrhea are all quite low, with the exception of the sensitivity estimates for urine specimens, which are even less precise than those for chlamydia.

## 4.6 Discussion

In this chapter, we extended the regression methods presented in Chapters 2 and 3 to accommodate multiple-disease group testing data. These methods are the first within this setting to implement machine learning techniques. We demonstrated the estimation performance of our models using simulation, and used our algorithms to estimate population-level regression models for chlamydia and gonorrhea using data obtained from the State Hygienic Laboratory in Iowa. Although we focused on the

two-disease case, generalizing our methods to account for any number of diseases would not be theoretically difficult, but more an arduous exercise in combinatorics.

# Chapter 5

# Generalized Additive Modeling for Group Testing Data

## 5.1 Introduction

In this chapter, we advance the methods presented in Chapter 2 by implementing a generalized additive modeling framework.

## 5.2 Preliminaries

For our generalized additive model, we continue to assume the observed testing outcomes are conditionally independent given the individual true statuses, and the conditional distribution $\mathbf{Z}|\widetilde{\mathbf{Y}}$ does not depend on the covariates. However, we relax the linearity assumptions of Chapter 2, and instead assume the relationship between $\widetilde{Y}_i$ and $\mathbf{x}_i$, for $i = 1, 2, \ldots, N$, is of the form

$$H\{\text{pr}(\widetilde{Y}_i = 1|\mathbf{x}_i)\} = \beta_0 + \sum_{q=1}^{r_1} g_q(x_{iq}) + \sum_{q=1}^{r_2} \beta_q x_{i(r_1+q)}, \tag{5.1}$$

where $H(\cdot)$ is a known binary link function, $g_q(\cdot),\ q = 1, 2, \ldots, r_1$, are unspecified smooth functions, $\beta_q,\ q = 1, 2, \ldots, r_2$, are regression coefficients, and $r_1 + r_2 = r$. This generalized additive form allows for the presence of both linear and nonlinear effects. In addition, nothing prevents $r_1 = r$ or $r_2 = r$, the latter of which would result in a standard generalized linear model. See Section 2.2 for a review of preliminary notation.

### 5.2.1 Generalized Additive Models

Before proceeding with the methodology, we provide a brief exposition of the standard generalized additive model (GAM). Fathered by Hastie and Tibshirani (1986; 1987) in the 1980s, GAMs remain one of the most recognized regression methods of today. Their flexibility simultaneously allows for both parametric and nonparametric model fits to be utilized within the same modeling structure. Linear components are fit parametrically using weighted linear least squares, while nonlinear components are fit using unspecified smooth functions. This is done by iteratively smoothing partial residuals and re-weighting the additive components accordingly, a process which can be seen as a Gauss-Seidel algorithm for fitting additive models.

For a large class of GAMs, including those for binary regression, the objective function which we seek to maximize is a penalized log-likelihood. One may at first consider the approach of fitting a single scatterplot smoother to the data; however, it has been well established that such procedures suffer greatly from the so called *curse of dimensionality* (Friedman and Stuetzle, 1981). This is the primary motivation for additive modeling, which instead fits a smooth function using each covariate individually. In addition, this additive form allows one to glean information on the contribution of each individual covariate. Within the general regression setting, a GAM has the form

$$\eta(\mathbf{X}) = f(\mu) = \beta_0 + \sum_{q=1}^{r} g_q(X_q),$$

where $\mu = \mathbb{E}(Y|\mathbf{X})$, $Y$ is the response variable, $\mathbf{X}$ is the $N \times r$ covariate matrix, and $g_q(\cdot),\ q = 1, 2, \ldots, r$, are unknown smooth functions. Here, $\eta(\cdot)$ is analogous to the linear predictor within the context of a generalized linear model.

The process by which the objective function is maximized is referred to as local scoring, its name stemming from the use of local score estimates within the Fisher scoring updates. Broadly speaking, the method works by repeatedly smoothing a transformed dependent variable on $X_q, q = 1, 2, \ldots, r$, and adjusting each smooth in

turn, a process which is analogous to a weighted backfitting algorithm. For identifiability reasons, it is standard convention to assume the functions average to zero over the data, that is, $\sum_{i=1}^{N} g_q(x_{iq}) = 0 \ \forall q$. It is easily seen that $\hat{\beta}_0 = f\{\mathbb{E}(Y)\}$ under this assumption, and this value remains constant throughout the iterative process. We initialize the model at $g_{q(0)} = 0 \ \forall q$. We then define our adjusted response variable $Y^* = \eta_{m-1} + (Y - \mu_{m-1})(\partial\eta/\partial\mu_{m-1})$, where $\eta_{m-1} = f(\mu_{m-1}) = \hat{\beta}_0 + \sum_{q=1}^{r} g_{q(m-1)}(X_q)$, and where the subscript $m$ denotes an estimate obtained from the $m$th iteration of the algorithm. The data is then transformed using weights $W = (\partial\mu/\partial\eta_{m-1})^2 V^{-1}$, where $V$ is the variance of $Y$ at the fitted values $\hat{\mu}$. The update $g_{qm}, q = 1, 2, \ldots, r$, is obtained by fitting a smoother to the partial residuals $R_q$ which are given by

$$R_q = Y^* - \hat{\beta}_0 - \sum_{k=1}^{q-1} g_{km}(X_k^*) - \sum_{k=q+1}^{r} g_{k(m-1)}(X_k^*),$$

where $X_k^*, \ k = 1, 2, \ldots, q-1, q+1, \ldots, r$, are the weighted covariates. These updated functions then yield new working responses and weights. This process is repeated until the change in the objective function becomes sufficiently small. Although we have referred to the $g_q(\cdot)$'s as nonparametric smooth functions, nothing prevents one or more of these functions from being replaced by a simple regression on the corresponding covariate. In fact, if each smoother is replaced in this way, the local scoring algorithm converges to the standard (weighted) multiple regression (e.g., weighted generalized linear model).

So far, we have yet to address the choice of scatterplot smoother. Although a number of smoothers are available to choose from, most modern GAM algorithms rely on reduced rank smoothing approaches to reduce computational costs. These approaches represent the unknown smooth functions in terms of basis expansions

$$g_q(X_q) = \sum_{b=1}^{B} \alpha_{qb} \mathcal{B}_{qb}(X_q), \ q = 1, 2, \ldots, r,$$

where the $\alpha_{qb}$'s are model coefficients, the $\mathcal{B}_{qb}(\cdot)$'s are known basis functions, commonly radial basis functions (thin-plate splines) or B-spline basis functions (smooth-

ing splines), and $B$ is the basis dimension, which provides an upper limit on the degrees of freedom associated with $g_q(\cdot)$. In practice, the exact choice of $B$ is generally not critical. It should be large enough such that the degrees of freedom are sufficient to represent the underlying function reasonably well (to some degree of confidence), but small enough to maintain computational efficiency. However, the actual *effective* degrees of freedom associated with a smooth are affected by the degree of penalization selected during fitting (i.e., the estimation of the penalty (smoothness) parameter), which is commonly estimated using generalized cross validation, Akaike information criterion (AIC), or restricted maximum likelihood.

## 5.3 Methodology

Our goal in this section is to extend the standard generalized additive model, which is classically used for supervised individual-level data, to handle the complexities of partially-supervised pooled data. Much of the methodology follows directly from Chapter 2, but we present it again here for completeness. We begin by formulating the complete log-likelihood function for any group testing protocol, which can be written as

$$l(\boldsymbol{\theta}|\mathbf{Z},\widetilde{\mathbf{Y}},\mathbf{X}) = \sum_{l=1}^{L}\sum_{j\in\mathcal{M}(l)}\{\widetilde{Z}_j\log[S_{e(l)}^{Z_j}(1-S_{e(l)})^{1-Z_j}]+(1-\widetilde{Z}_j)\log[(1-S_{p(l)})^{Z_j}S_{p(l)}^{1-Z_j}]\} + \sum_{i=1}^{N}\{\widetilde{Y}_i\log P_i+(1-\widetilde{Y}_i)\log(1-P_i)\}, \quad (5.2)$$

where $\boldsymbol{\theta} = (\mathbf{S}_e', \mathbf{S}_p', \mathbf{P}')'$, $\mathbf{S}_e = (S_{e(1)}, S_{e(2)}, \ldots, S_{e(L)})'$, $\mathbf{S}_p = (S_{p(1)}, S_{p(2)}, \ldots, S_{p(L)})'$, $\mathbf{P} = (P_1, P_2, \ldots, P_N)'$, $\mathbf{X} = (\mathbf{x}_1\ \mathbf{x}_2\ \cdots\ \mathbf{x}_N)'$, $P_i = \text{pr}(\widetilde{Y}_i = 1|\mathbf{x}_i), i = i, \ldots, N$, and $L$ is the number of sets of assay accuracy probabilities. Due to the latency of the true disease statuses, direct maximization of the log-likelihood function is not possible. To implement the EM algorithm, we begin by calculating the conditional expectation of

the log-likelihood function in equation (5.2), which is given by

$$
\begin{aligned}
Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) &= \mathbb{E}[l(\boldsymbol{\theta}|\mathbf{Z}, \widetilde{\mathbf{Y}}, \mathbf{X})|\mathbf{Z}, \mathbf{X}, \boldsymbol{\theta}^{(t)}] = \\
&\sum_{l=1}^{L} \sum_{j \in \mathcal{M}(l)} \left\{ \underbrace{\text{pr}\left( \sum_{i \in \mathcal{P}_j} \widetilde{Y}_i > 0 \middle| \mathbf{Z}, \mathbf{X}, \boldsymbol{\theta}^{(t)} \right)}_{C_j} \times \log[S_{e(l)}^{Z_j}(1 - S_{e(l)})^{1-Z_j}] \right. \\
&\left. + \left[ 1 - \text{pr}\left( \sum_{i \in \mathcal{P}_j} \widetilde{Y}_i > 0 \middle| \mathbf{Z}, \mathbf{X}, \boldsymbol{\theta}^{(t)} \right) \right] \times \log[(1 - S_{p(l)})^{Z_j} S_{p(l)}^{1-Z_j}] \right\} \\
&+ \sum_{i=1}^{N} \{\text{pr}(\widetilde{Y}_i = 1|\mathbf{Z}, \mathbf{x}_i, \boldsymbol{\theta}^{(t)}) \log P_i + [1 - \text{pr}(\widetilde{Y}_i = 1|\mathbf{Z}, \mathbf{x}_i, \boldsymbol{\theta}^{(t)})] \log(1 - P_i)\}, \quad (5.3)
\end{aligned}
$$

where the superscript $t$ denotes an estimate from the $t$th iteration of the algorithm. To complete the E-step, we must first derive expressions for $\text{pr}(\widetilde{Y}_i = 1|\cdot)$ and $C_j$, where $C_j$ is the conditional probability of the $j$th pool being truly positive. Unfortunately, for most group testing protocols these expectations are intractable. Therefore, we stochastically approximate these probabilities by repeatedly sampling from the conditional distributions of the individual true statuses using Markov Chain Monte Carlo. We have $\widetilde{Y}_i|\mathbf{Z}, \widetilde{\mathbf{Y}}_{-i}, \mathbf{X}, \boldsymbol{\theta}$ follows a Bernoulli distribution with success probability $p_{i1}^*/(p_{i0}^* + p_{i1}^*)$, where

$$
p_{i1}^* = P_i \prod_{j \in \mathcal{A}_i} S_{ej}^{Z_j}(1 - S_{ej})^{1-Z_j},
$$

$$
p_{i0}^* = (1 - P_i) \prod_{j \in \mathcal{A}_i} [S_{ej}^{Z_j}(1 - S_{ej})^{1-Z_j}]^{I(\Sigma_{i' \in \mathcal{P}_{ij}} \widetilde{Y}_{i'} > 0)} \times [(1 - S_{pj})^{Z_j} S_{pj}^{1-Z_j}]^{I(\Sigma_{i' \in \mathcal{P}_{ij}} \widetilde{Y}_{i'} = 0)}, \quad (5.4)
$$

$\widetilde{\mathbf{Y}}_{-i} = \{\widetilde{Y}_1, \ldots, \widetilde{Y}_{i-1}, \widetilde{Y}_{i+1}, \ldots, \widetilde{Y}_N\}$, $\mathcal{A}_i = \{j : i \in \mathcal{P}_j\}$, $S_{ej}$ and $S_{pj}$ are the sensitivity and specificity respectively of the assay used to test the $j$th pool, and $\mathcal{P}_{ij} = \{i' \in \mathcal{P}_j : i' \neq i\}$. The estimate of $\text{pr}(\widetilde{Y}_i = 1|\cdot)$ is given by the proportion of the imputed statuses of the $i$th individual which are positive. Similarly, the estimate of $C_j$ is given by the proportion of the imputed statuses of the $j$th pool which are positive.

The M-step consists of maximizing the conditional expectation in equation (5.3) with respect to $\boldsymbol{\theta}$, which may at first seem daunting, but notice only the final line is

a function of $\mathbf{P}$ (and only $\mathbf{P}$). As such, we need only focus our attention on

$$Q_{\mathbf{P}}(\mathbf{P}|\mathbf{P}^{(t)}) = \sum_{i=1}^{N}\{\text{pr}(\widetilde{Y}_i = 1|\mathbf{Z}, \mathbf{x}_i, \boldsymbol{\theta}^{(t)}) \log P_i + [1 - \text{pr}(\widetilde{Y}_i = 1|\mathbf{Z}, \mathbf{x}_i, \boldsymbol{\theta}^{(t)})] \log(1 - P_i)\} \tag{5.5}$$

when maximizing with respect to $\mathbf{P}$. It is easily seen that equation (5.5) has the standard form of a log-likelihood for individual-level binary data. Thus, all of the GAM methodology outlined in the previous section can be integrated within our EM algorithm for group testing data by simply viewing the conditional success probabilities at each M-step as weighted binary responses. To adapt the general notation of the previous section to our algorithm, we can see that $Y \equiv \text{pr}(\widetilde{Y}_i = 1|\mathbf{Z}, \mathbf{x}_i, \boldsymbol{\theta}^{(t)})$ is the response variable, $\mu \equiv \text{pr}(\widetilde{Y}_i = 1|\mathbf{x}_i) \equiv P_i$ is the mean of the binary response, and $\eta(\cdot) \equiv H(\cdot)$ is the binary link function.

To complete the M-step, we maximize $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ directly with respect to $\mathbf{S}_e$ and $\mathbf{S}_p$, which yields the closed form solutions

$$S_{e(l)}^{(t+1)} = \frac{\sum_{j\in\mathcal{M}(l)} C_j^* Z_j}{\sum_{j\in\mathcal{M}(l)} C_j^*} \text{ and } S_{p(l)}^{(t+1)} = \frac{\sum_{j\in\mathcal{M}(l)}(1 - C_j^*)(1 - Z_j)}{\sum_{j\in\mathcal{M}(l)}(1 - C_j^*)},\ l = 1, 2, \ldots, L, \tag{5.6}$$

where $C_j^*$ is the stochastic approximation of $C_j$. The entire process is iterated until the change in $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ becomes sufficiently small. A summary is provided in the description entitled "Algorithm 5" in the Appendix.

## 5.4 Simulation

In this section, we use simulation to demonstrate the computational advantages of our procedure over the Bayesian approach of Liu et al. (2020). Their competing methodology uses both Gaussian process and predictive process priors, the estimation performances of which are nearly identical, with the Gaussian predictive process priors having greater computational efficiency. Thus, all comparisons made between our models are made with respect to their more efficient method. We consider two testing scenarios: Dorfman testing (DT) and array testing (AT); see Section 1.1 for

details. Observations are simulated from two population models, both of which are of the form

$$\text{probit}(P_i) = \beta_0 + g_1(x_{i1}) + g_2(x_{i2}) + \beta_1 x_{i3} + \beta_2 x_{i4},$$

where $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2)' = (-1.8, 0.5, 0.5)'$, $x_{i1}, x_{i2} \sim \mathcal{U}(-3, 3)$, $x_{i3} \sim \mathcal{N}(0, 1)$, and $x_{i4} \sim$ Bernoulli(0.5). For the first population model (M1), the nonlinear functions are given by

$$g_1(x_1) = 0.7 \exp\{-[1.2^2 I(x > 0) + 1.2^{-2} I(x < 0)] x_1^2 / 6.25\} - 0.468, \text{ and}$$
$$g_2(x_2) = 0.6 \exp\left\{-\frac{(x_2 + 1.5)^2}{0.72}\right\} + 0.4 \exp\left\{-\frac{(x_2 - 1.5)^2}{1.28}\right\} - 0.279,$$

while in the second model (M2),

$$g_1(x_1) = \frac{0.2 \sin\{\pi(x_1 + 0.2)/2.5\} + 0.4}{\exp\{[x_1 + (x_1 - 0.3)^2 I(x_1 > 0.3)]/6\}} - 0.351, \text{ and}$$
$$g_2(x_2) = \frac{4 \exp(1 + 1.5x_2)}{6 + 6 \exp(1 + 1.5x_2)} - 0.406.$$

The functions $g_1(\cdot)$ and $g_2(\cdot)$ were chosen to represent a wide array of nonlinear patterns. Both population models yield a prevalence of approximately 9%, which is consistent with the data application presented in Section 5.5. Under each population model, we simulated 500 data sets for both DT and AT, with each data set containing $N = 5{,}000$ individuals. For $i = 1, 2, \ldots, N$, the true status of the $i$th individual is simulated according to a Bernoulli($P_i$) distribution, where $P_i$ is given by the corresponding population model. Each master pool consists of specimens from five randomly selected individuals. The observed status of the $j$th pool is simulated according to a Bernoulli$[S_{ej}\widetilde{Z}_j + (1 - S_{pj})(1 - \widetilde{Z}_j)]$ distribution; to capture a mild dilution effect, we consider the use of a single screening assay with accuracies $S_{e(1)} = 0.95$ and $S_{p(1)} = 0.98$ for pools, and $S_{e(2)} = 0.98$ and $S_{p(2)} = 0.99$ for individual specimens. These assay accuracy probabilities are not assumed to be known for the model building process.

To fit our model, we integrated Trevor Hastie's *gam* function from the *mgcv* R package into our algorithm. For both population models, preliminary graphical

Table 5.1: Simulation study comparing the estimation performance of our generalized additive model (GAM) to that of the Bayesian approach of Liu et al. (2020) ($\text{GAM}_B$) under population models M1 and M2. The average bias (Bias) and estimated standard error/average posterior standard deviation (ESE) from 500 data sets is shown. The sample size for each data set is 5000. Dorfman testing (DT) and array testing (AT) use master pools of size five. For our GAM, the link was intentionally misspecified as logit for model M2.

| Dorman Testing | M1/GAM | | M1/$\text{GAM}_B$ | | M2/GAM | | M2/$\text{GAM}_B$ | |
|---|---|---|---|---|---|---|---|---|
| Parameter | Bias | ESE | Bias | ESE | Bias | ESE | Bias | ESE |
| $\beta_1 = 2$ | 0.00 | 0.03 | 0.02 | 0.04 | - | - | 0.02 | 0.04 |
| $\beta_2 = -1$ | 0.00 | 0.06 | 0.02 | 0.06 | - | - | 0.02 | 0.06 |
| $S_{e(1)} = 0.95$ | −0.01 | 0.04 | −0.05 | 0.06 | −0.03 | 0.05 | −0.05 | 0.05 |
| $S_{e(2)} = 0.98$ | 0.00 | 0.02 | 0.00 | 0.01 | −0.01 | 0.02 | 0.00 | 0.01 |
| $S_{p(1)} = 0.98$ | 0.00 | 0.01 | −0.01 | 0.02 | 0.00 | 0.01 | −0.01 | 0.02 |
| $S_{p(2)} = 0.99$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |
| Array Testing | M1/GAM | | M1/$\text{GAM}_B$ | | M2/GAM | | M2/$\text{GAM}_B$ | |
| Parameter | Bias | ESE | Bias | ESE | Bias | ESE | Bias | ESE |
| $\beta_1 = 2$ | 0.00 | 0.03 | 0.01 | 0.03 | - | - | 0.01 | 0.03 |
| $\beta_2 = -1$ | 0.00 | 0.06 | 0.01 | 0.06 | - | - | 0.01 | 0.06 |
| $S_{e(1)} = 0.95$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{e(2)} = 0.98$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{p(1)} = 0.98$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 |
| $S_{p(2)} = 0.99$ | 0.00 | 0.01 | 0.00 | 0.01 | 0.00 | 0.01 | −0.01 | 0.01 |

analysis easily revealed the presence of nonlinear effects for the first two covariates. As such, thin-plate splines were used to model these effects, while the remaining two effects were modeled linearly. The probit link was used for model M1, while for model M2, the link was intentionally misspecified as the logit link to investigate the method's robustness to link misspecification.

### 5.4.1 Simulation Results

Table 5.1 displays the model estimates for the simulations under population models M1 and M2. The empirical bias and estimated standard error/average posterior standard deviation are displayed for our model (GAM) and for the competing Bayesian methodology ($\text{GAM}_B$). In both simulation settings, the methods perform similarly,

although the GAM displays slightly lower bias for many of the parameter estimates. In addition, our method is much more efficient in terms of computational speed. For population model M1, our method had an average runtime of 4.3 minutes, while the Bayesian method averaged 47 minutes per data set; for population model M2, our method had an average runtime of 2.4 minutes for Dorfman testing and 3.3 minutes for array testing, while the Bayesian method had an average runtime of 48 to 49 minutes for both testing protocols.[1]

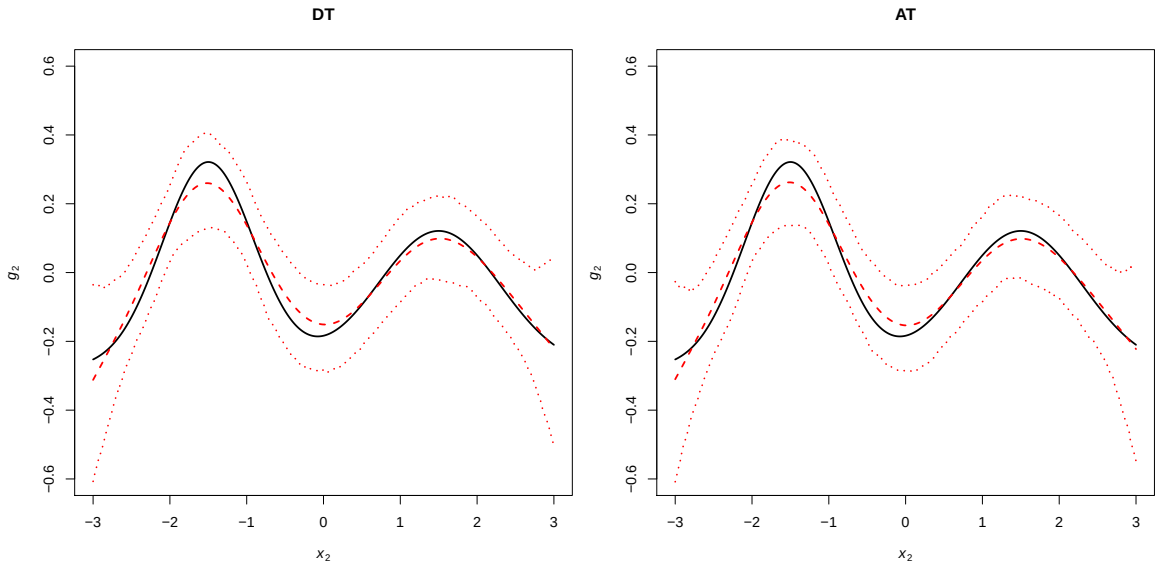


Figure 5.1: Estimation of nonlinear function $g_2$ under population model M1 using our generalized additive model. The solid curve represents the true function, the dashed curve represents the mean of 500 estimates of $g_2$, and the dotted curves represent the 0.025 and 0.975 quantiles of 500 estimates of $g_2$. Left: Dorfman testing (DT). Right: Array testing (AT).

Figure 5.1 displays our model's estimation of nonlinear function $g_2$ under population model M1. The average of the model estimates captures the nonlinear pattern quite well, and the true function lies entirely within each of the 95% empirical credible intervals. Figure 5.2 displays our model's predicted probability of disease against the

[1]The computation times for the approach of Liu et al. (2020) are for an equivalent simulation, but with known assay accuracies. Thus, the Bayesian method may actually take longer to converge than indicated here.

true probability of disease for a sample data set simulated under population model M2. Although there is a mild departure between the correspondence of the predicted and true disease probabilities for a handful of observations (those having the highest true probability of disease), the method overall appears quite resistant to the potential adverse effects of link misspecification.

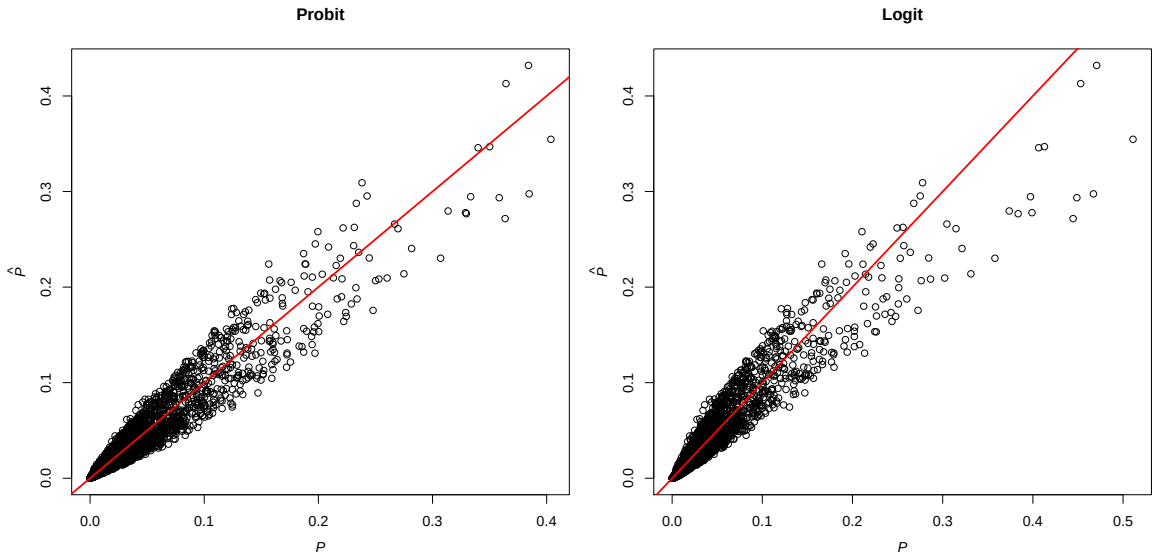


Figure 5.2: Effect of model-link misspecification. Our generalized additive model's predicted probability of disease ($\widehat{P}$) against the true probability of disease ($P$) for a sample data set simulated under population model M2. Left: Probit link. Right: Logit link.

## 5.5 Data Application

We apply our generalized additive model (GAM) to the chlamydia screening data provided by the State Hygienic Laboratory at the University of Iowa. These data contain the observed chlamydia status and the covariate information for 13,862 female subjects. The covariates, which are observed on each individual, include age ($x_1$), whether the individual is Caucasian ($x_2 = 1$), whether the individual reports having a new sexual partner within the last 90 days ($x_3 = 1$), whether the individual reports having multiple partners within the last 90 days ($x_4 = 1$), whether the individual

Table 5.2: Iowa chlamydia data parameter estimates for our generalized additive model.

| Parameter | Description | Est. | ESE |
|---|---|---|---|
| $\beta_0$ | | $-1.459$ | 0.327 |
| $\beta_1$ | Race | $-0.174$ | 0.043 |
| $\beta_2$ | New partner | 0.142 | 0.034 |
| $\beta_3$ | Multiple partners | 0.175 | 0.050 |
| $\beta_4$ | Contact with STD | 0.752 | 0.063 |
| $\beta_5$ | Symptoms | 0.146 | 0.040 |
| $S_{e(1)}$ | Swab pool | 0.949 | 0.049 |
| $S_{e(2)}$ | Swab individual | 0.999 | 0.001 |
| $S_{e(3)}$ | Urine individual | 0.923 | 0.081 |
| $S_{p(1)}$ | Swab pool | 0.999 | 0.001 |
| $S_{p(2)}$ | Swab individual | 0.975 | 0.007 |
| $S_{p(3)}$ | Urine individual | 0.991 | 0.007 |

reports having a partner with an STD within the last year ($x_5 = 1$), and whether the individual shows symptoms of infection ($x_6 = 1$); see Section 3.5 for more details.

Of primary interest is the relationship between age and disease status. Age is the only quantitative covariate within the data set, and thus provides the only relationship which can be modeled with a smooth function. The GAM was fit using the logit and probit link functions, with both links producing similar model fits. As such, the probit link function was chosen to provide a more direct comparison to the approach of Liu et al. (2020). We allow the assay accuracies to depend upon both pool size (pooled vs unpooled) and specimen type (swab vs urine). This yields three sets of assay accuracy probabilities which are estimated along with the other model parameters. The model fitting process took approximately 30 minutes to complete.

Table 5.2 displays the model parameter estimates for the Iowa chlamydia data. Standard error estimates (ESE) were calculated using 500 bootstrap samples; a modified bootstrap procedure was used to preserve the proportions of the key features of the data (i.e., positive pools and resulting retests, negative pools, individual swab specimens, and urine specimens). With the exception of the sensitivity estimates for

urine specimens and pooled swab specimens, the parameter estimates provided by our GAM closely match those from Liu et al. (2020). The lack of congruency between the sensitivity estimates can be attributed to their large standard errors. These findings are also consistent with our analyses in Chapter 3.

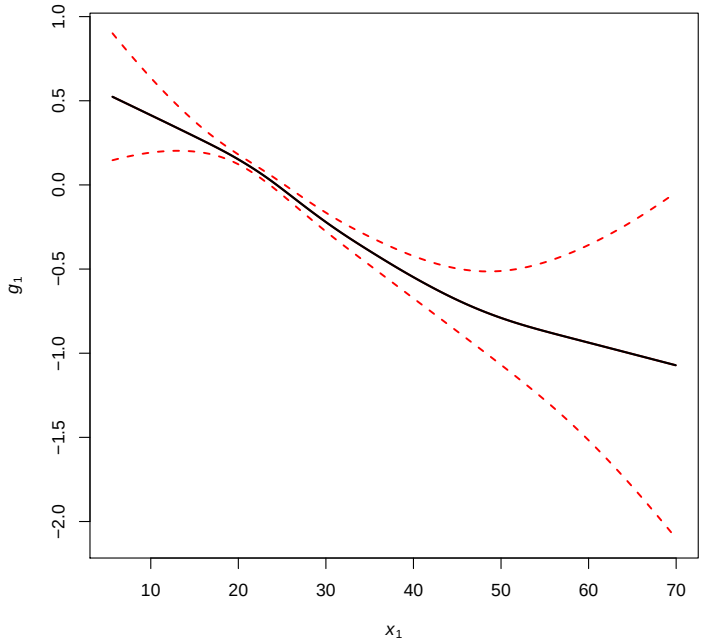


Figure 5.3: Estimation of age effect ($x_1$) for Iowa chlamydia data using our generalized additive model. The solid curve represents the estimated smooth function, while the dashed curves represent the set of approximate 95% pointwise confidence intervals.

Figure 5.3 displays the model-estimated effect of age on chlamydia status. The solid curve represents the estimated smooth function, while the dashed curves represent the set of approximate 95% pointwise confidence intervals. Our model suggests that there may be a slight nonlinear trend in the tails, but that the age effect is largely linear. This is in opposition to the results of Liu et al. (2020), who found a strong nonlinear trend for the age effect. However, this is not all that surprising if one considers the variability of these estimates. We can clearly see from the figure that the estimates in the tails of the smooth function are highly imprecise, especially

in the upper-tail. This is due to an inadequacy of data in these regions. Thus, it would be prudent to obtain more data, especially from individuals 50 years of age or older, before rendering conclusions on the linearity of the relationship between age and chlamydia status.

## 5.6 Discussion

In this chapter, we developed a generalized additive model for group testing data which allows linear effects to be identified and retain their interpretability, while more complex relationships are modeled with smooth functions. We addressed many of the shortcomings present in the competing methodology of Liu et al. (2020). Their method relies on complex Gaussian process and predictive process priors, which in addition can lead to model misspecification when not formulated properly. Our approach has no need for prior information and so avoids this problem altogether. Also, by taking a more simplistic approach, our method is substantially more efficient in terms of computational speed. Without further augmentation, their model is capable of running only the probit link function. Our model can accommodate both the probit and logit link functions, the latter of which is by and far the most common link function used in practice. We demonstrated the robustness of our method to link misspecification and provided simulation evidence that our method may further be preferred on the basis of estimation accuracy. Lastly, we applied our method to disease screening data obtained from the University of Iowa.

# Bibliography


Amemiya, C., Algeria-Hartman, M., Aslanidis, C., Chen, C., Nikolic, J., Gingrich, J., and De Jong, P. (1992). A two-dimensional YAC pooling strategy for library screening via STS and Alu-PCR methods. *Nucleic Acids Research* **25**, 2559–2563.

Barillot, E., Lacroix, B., and Cohen, D. (1991). Theoretical analysis of library screening using a N-dimensional pooling strategy. *Nucleic Acids Research* **19**, 6241–6247.

Berger, T., Mandell, J., and Subrahmanya, P. (2000). Maximally efficient two-stage screening. *Biometrics* **56**, 833–840.

Bilder, C. and Tebbs, J. (2005). Empirical Bayesian estimation of the disease transmission probability in multiple-vector-transfer designs. *Biometrical Journal* **47**, 502–516.

Bilder, C. and Tebbs, J. (2009). Bias, efficiency, and agreement for group-testing regression models. *Journal of Statistical Computation and Simulation* **79**, 67–80.

Bilder, C. and Tebbs, J. (2012). Pooled testing procedures for screening high volume clinical specimens in heterogeneous populations. *Statistics in Medicine* **31**, 3261–3268.

Bilder, C., Tebbs, J., and Chen, P. (2010). Informative retesting. *Journal of the American Statistical Association* **105**, 942–955.

Black, M., Bilder, C., and Tebbs, J. (2012). Group testing in heterogeneous populations by using halving algorithms. *Journal of the Royal Statistical Society. Series C* **61**, 277–290.

Brennan, T. (1991). *Just Doctoring: Medical Ethics in the Liberal State*. Berkeley: University of California Press.

Bruno, W., Knill, E., Balding, D., Bruce, D., Doggett, N., Sawhill, W., Stallings, R., Whittaker, C., and Torney, D. (1995).Efficient pooling designs for library screening. *Genomics* **26**, 21–30.

Burrows, P. (1987). Improved estimation of pathogen transmission rates by group testing. *Phytopathology* **77**, 363–365.

Chatterjee, A. and Bandyopadhyay, T. (2020). Regression models for group testing: Identifiability and asymptotics. *Journal of Statistical Planning and Inference* **204**, 141–152.

Chaubey, Y. and Li, W. (1995). Comparison between maximum likelihood and Bayes methods for estimation of binomial probability with samples composition. *Journal of Official Statistics* **11**, 379–390.

Chen, P., Tebbs, J., and Bilder, C. (2009). Group testing regression models with fixed and random effects. *Biometrics* **65**, 1270–1278.

Delaigle, A. and Hall, P. (2012). Nonparametric regression with homogeneous group testing data. *Annals of Statistics* **40**, 131–158.

Delaigle, A. and Hall, P. (2015). Nonparametric methods for group testing data, taking dilution into account. *Biometrika* **102**, 871–887.

Delaigle, A., Hall, P., and Wishart, J. (2014). New approaches to non- and semi-parametric regression for univariate and multivariate group testing data. *Biometrika* **101**, 567–585.

Delaigle, A., Huang, W., and Lei, S. (2019). Estimation of conditional prevalence from group testing data with missing covariates. *Journal of the American Statistical Association* **00**, 1–14.

Delaigle, A. and Meister, A. (2011). Nonparametric regression analysis for group testing data. *Journal of the American Statistical Association* **106**, 640–650.

Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood for incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B* **39**, 1–38.

Dorfman, R. (1943). The detection of defective members of large populations. *Annals of Mathematical Statistics* **14**, 436–440.

Einwalter, L., Ritchie, J., Ault, K., and Smith, E. (2005). Gonorrhea and chlamydia infection among women visiting family planning clinics: Racial variation in prevalence and predictors. *Perspectives on Sexual and Reproductive Health* **37**, 135–140.

Evans, G. and Lewis, K. (1989). Physical mapping of complex genomes by cosmid multiplex analysis. *Genetics* **86**, 5030–5034.

Farach, M., Kannan, S., Knill, E., and Muthukrishnan, S. (1997). Group testing problems with sequences in experimental molecular biology. *Proceedings. Compression and Complexity of SEQUENCES*, 357–367. IEEE Press.

Farrington, C. (1992). Estimating prevalence by group testing using generalized linear models. *Statistics in Medicine* **11**, 1591–1597.

Freund, Y. (1995). Boosting a weak learning algorithm by majority. *Information and Computation* **121**, 256–285.

Freund Y. and Schapire, R. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences* **55**, 119–139.

Friedman, J., Hastie, T., and Tibshirani, R. (2000). Additive logistic regression: A statistical view of boosting. *Annals of Statistics* **28**, 337–407.

Friedman, J., Hastie, T., and Tibshirani, R. (2001). *Elements of Statistical Learning*. Springer, New York.

Friedman, J. (2002). Stochastic gradient boosting. *Computational Statistics and Data Analysis* **38**, 367–378.

Friedman, J. and Stuetzle, W. (1981). Projection pursuit regression. *Journal of the American Statistical Association* **76**, 817–823.

Gastwirth, J. (2000). The efficiency of pooling in the detection of rare mutations. *American Journal of Human Genetics* **67**, 1036–1039.

Gastwirth, J. and Johnson, W. (1994). Screening with cost effective quality control: Potential applications to HIV and drug testing. *Journal of the American Statistical Association* **89**, 972–981.

Gregory, K., Wang, D., and McMahan, C. (2019). Adaptive elastic net for group testing. *Biometrics* **75**, 13–23.

Haber, G. and Malinovsky, Y. (2017). Random walk designs for selecting pool sizes in group testing estimation with small samples. *Biometrical Journal* **59**, 1382–1398.

Hastie, T. and Tibshirani, R. (1986). Generalized additive models (with discussion). *Statistical Science* **1**, 297–318.

Hou, P., Tebbs, J., Bilder, C., and McMahan, C. (2017). Hierarchical group testing for multiple infections. *Biometrics* **73**, 656–665.

Huang, X. (2009). An improved test of latent-variable model misspecification in structural measurement error models for group testing data. *Statistics in Medicine* **28**, 3316–3327.

Huang, X. and Warasi, M. (2017). Maximum likelihood estimators in regression models for error-prone group testing data. *Scandinavian Journal of Statistics* **44**, 918–931.

Hughes-Oliver, J. and Swallow, W. (1994). A two-stage adaptive group-testing procedure for estimating small proportions. *Journal of the American Statistical Association* **89**, 982–993.

Hung, M. and Swallow, W. (1999). Robustness of group testing in the estimation of proportions. *Biometrics* **55**, 231–237.

Hyun, N., Gastwirth, J., Graubard, B. (2018). Grouping methods for estimating prevalences of rare traits for complex survey data that preserve confidentiality of respondents. *Statistics in Medicine* **37**, 2174–2186.

Johnson, R., Newhall, W., Papp, J., et al. (2002). Screening tests to detect *Chlamydia trachomatis* and *Neisseria gonorrhoeae* infections. *MMWR Recommendations and Reports* **51**, 1–38.

Kim, H. and Hudgens, M. (2009). Three-dimensional array-based group testing algorithms. *Biometrics* **65**, 903–910.

Kim, H., Hudgens, M., Dreyfuss, J., Westreich, D., and Pilcher, C. (2007). Comparison of group testing algorithms for case identification in the presence of testing error. *Biometrics* **63**, 1152–1163.

Kleinman, S., Strong, D., Tegtmeier, G., Holland, P., Gorlin, J., Cousins, C., Chiacchierini, R., and Pietrelli, L. (2005). Hepatitis B virus (HBV) DNA screening of blood donations in minipools with the COBAS AmpliScreen HBV test. *Transfusion* **45**, 1247–1257.

Krajden, M., Cook, D., Mak, A., Chu, K., Chahil, N., Steinberg, M., Rekart, M., and Gilbert, M. (2014). Pooled nucleic acid testing increases the diagnostic yield of acute HIV infections in a high-risk population compared to 3rd and

4th generation HIV enzyme immunoassays. *Journal of Clinical Virology* **61**, 132–137.

Lewis, J., Lockary, V., and Kobic, S. (2012). Cost savings and increased efficiency using a stratified specimen pooling strategy for *Chlamydia trachomatis* and *Neisseria gonorrhoeae*. *Sexually Transmitted Diseases* **39**, 46–48.

Li, Q., Liu, A., and Xiong, W. (2017). D-optimality of group testing for joint estimation of correlated rare diseases with misclassification. *Statistica Sinica* **27**, 823–838.

Lin, J., Wang, D., and Zheng, Q. (2019). Regression analysis and variable selection for two-stage multiple-infection group testing data. *Statistics in Medicine* **38**, 4519–4533.

Litvak, E., Tu, X., and Pagano, M. (1994). Screening for the presence of a disease by pooling sera samples. *Journal of the American Statistical Association* **89**, 424–434.

Liu, Y., McMahan, C., and Gallagher, C. (2017). A general framework for the regression analysis of pooled biomarker assessments. *Statistics in Medicine* **36**, 2363–2377

Liu, Y., McMahan, C., Tebbs, J., Gallagher, C., and Bilder, C. (2020). Generalized additive regression for group testing data. *Biostatistics* **0**, 1–17

Louis, T. (1982). Finding the observed information matrix when using the EM algorithm. *Journal of the Royal Statistical Society. Series B* **44**, 226–233

McMahan, C., Tebbs, J., and Bilder, C. (2012a). Informative Dorfman screening. *Biometrics* **68**, 287–296.

McMahan, C., Tebbs, J., and Bilder, C. (2012b). Two-dimensional informative array testing. *Biometrics* **68**, 793–804.

McMahan, C., Tebbs, J., and Bilder, C. (2013). Regression models for group testing data with pool dilution effects. *Biostatistics* **14**, 284–298.

McMahan, C., Tebbs, J., Hanson, T., and Bilder, C. (2017). Bayesian regression for group testing data. *Biometrics* **73**, 1443–1452.

Navarro, C., Jolly, A., Nair, R., and Chen, Y. (2002). Risk factors for genital chlamydial infection. *Canadian Journal of Infectious Diseases* **13**, 195–207.

Nguyen, N., Bish, E., and Aprahamian, H. (2018). Sequential prevalence estimation with pooling and continuous test outcomes. *Statistics in Medicine* **37**, 2391–2426.

Nwokolo, N., Dragovic, B., Patel, S., Tong, C., Barker, G., and Radcliffe, K. (2016). 2015 UK national guideline for the management of infection with *Chlamydia trachomatis*. *International Journal of STD and AIDS* **27**, 251–267.

Papp, J., Schachter, J., Gaydos, C., and Van Der Pol, B. (2014). Recommendations for the laboratory-based detection of *Chlamydia trachomatis* and *Neisseria gonorrhoeae*. *MMWR Recommendations and Reports* **63**, 1–19.

Peeling, R., Toye, B., Jessamine, P., and Gemmill, I. (1998). Pooling of urine specimens for PCR testing: A cost saving strategy for *Chlamydia trachomatis* control programmes. *Sexually Transmitted Infections* **74**, 66–70.

Phatarfod, R. and Sudbury, A. (1994). The use of a square array scheme in blood testing. *Statistics in Medicine* **13**, 2337–2343.

Schapire, R. (1990). The strength of weak learnability. *Machine Learning* **5**, 197–227.

Sobel, M. and Elashoff, R. (1975). Group testing with a new goal, estimation. *Biometrika* **62**, 182–193.

Sterrett, A. (1957). On the detection of defective members of large populations. *Annals of Mathematical Statistics* **28**, 1033–1036.

Swallow, W. (1985). Group testing for estimating infection rates and probabilities of disease transmission. *Phytopathology* **75**, 882–889.

Tebbs, J., Bilder, C., and Moser, B. (2003). An empirical Bayes group-testing approach to estimating small proportions. *Communications in Statistics: Theory and Methods* **32**, 983–995.

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.

Thompson, K. (1962). Estimation of the proportion of vectors in a natural population of insects. *Biometrics* **18**, 568–578.

Tibshirani, R. and Hastie, T. (1987). Local likelihood estimation. *Journal of the American Statistical Association* **82**, 559–568.

Tu, X., Kowalski, J., and Jia, G. (1999). Bayesian analysis of prevalence with covariates using simulation-based techniques: Applications to HIV screening. *Statistics in Medicine* **18**, 3059–3073.

Vansteelandt, S., Goetghebeur, E., and Verstraeten, T. (2000). Regression models for disease prevalence with diagnostic tests on pools of serum samples. *Biometrics* **56**, 1126–1133.

Wang, D., McMahan, C., Gallagher, C., and Kulasekera, K. (2014). Semiparametric group testing regression models. *Biometrika* **101**, 587–598.

Wang, D., McMahan, C., Tebbs, J., and Bilder, C. (2018). Group testing case identification with biomarker information. *Computational Statistics and Data Analysis* **122**, 156–166.

Wang, D., Zhou, H., and Kulasekera, K. (2013). A semi-local likelihood regression estimator of the proportion based on group testing data. *Journal of Nonparametric Statistics* **25**, 209–221.

Warasi, M., Tebbs, J., McMahan, C., and Bilder, C. (2016). Estimating the prevalence of multiple diseases from two-stage hierarchical pooling. *Statistics in Medicine* **35**, 3851–3864.

Ward, G., Hastie, T., Barry, S., Elith, J., and Leathwick, J. (2009). Presence-only data and the EM algorithm. *Biometrics* **65**, 554–563.

Wein, L. and Zenios, S. (1996). Pooled testing for HIV screening: Capturing the dilution effect. *Operations Research* **44**, 543–569.

Xie, M. (2001). Regression analysis of group testing samples. *Statistics in Medicine* **20**, 1957–1969.

Yasui, Y., Pepe, M., Hsu, L., Adam, B., and Feng, Z. (2004). Partially supervised learning using an EM-boosting algorithm. *Biometrics* **60**, 199–206.

Zhang, B., Bilder, C., and Tebbs, J. (2013a). Group testing regression model estimation when case identification is a goal. *Biometrical Journal* **55**, 173–189.

Zhang, B., Bilder, C., and Tebbs, J. (2013b). Regression analysis for multiple-disease group testing data. *Statistics in Medicine* **32**, 4954–4966.

# Appendix A: Algorithms

**Algorithm 1:** Boosted Logistic Regression Model for Group Testing Data

**input :** $\mathbf{Z}$, $\mathbf{X}$, $\mathbf{S}_e^{(0)}$, $\mathbf{S}_p^{(0)}$, $\mathbf{F}^{(0)}$

**repeat**

  For $i = 1, 2, \ldots, N$, and for $j = 1, 2, \ldots, J$, estimate $\text{pr}(\widetilde{Y}_i = 1 | \mathbf{Z}, \mathbf{x}_i, \boldsymbol{\theta}^{(t)})$ and $C_j$ according to equation (2.3);

  Update $\mathbf{S}_e^{(t+1)}$ and $\mathbf{S}_p^{(t+1)}$ according to equation (2.4);

  **repeat**

    For $p = 1, 2, \ldots, r$, compute $(\widehat{\alpha}_p, \widehat{\beta}_p) \equiv \arg\max_{\alpha,\beta} Q_{\alpha,\beta}(\alpha, \beta | \boldsymbol{\theta}^{(t)})$, where $Q_{\alpha,\beta}(\cdot)$ is given by equation (3.1);

    Update $F_{im} = F_{i(m-1)} + \tau(\widehat{\alpha}_{p^*} + \widehat{\beta}_{p^*} x_{\cdot p^*})$, where $\tau$ is the learning rate and $p^*$ is the value of $p$ that results in the largest value of $Q_{\alpha,\beta}(\widehat{\alpha}_p, \widehat{\beta}_p | \boldsymbol{\theta}^{(t)})$;

  **until** $||\mathbf{F}_m - \mathbf{F}_{(m-1)}|| < \epsilon$;

  Update $F_i^{(t+1)} = F_{iM}$, where $M$ is the iteration upon which $||\mathbf{F}_m - \mathbf{F}_{(m-1)}|| < \epsilon$;

**until** $|Q(\boldsymbol{\theta}^{(t+1)} | \boldsymbol{\theta}^{(t)}) - Q(\boldsymbol{\theta}^{(t)} | \boldsymbol{\theta}^{(t-1)})| < \epsilon$;

**output:** $(\mathbf{S}_e^{(T)\prime}, \mathbf{S}_p^{(T)\prime}, \mathbf{F}^{(T)\prime})'$ where $T$ is the iteration upon which the algorithm converges.

**Algorithm 2:** Boosted Regression Trees Model for Group Testing Data

**input :** $\mathbf{Z}$, $\mathbf{X}$, $\mathbf{S}_e^{(0)}$, $\mathbf{S}_p^{(0)}$, $\mathbf{F}^{(0)}$

**for** $t = 0$ **to** $T$ **do**

- For $i = 1, 2, \ldots, N$, and for $j = 1, 2, \ldots, J$, estimate $\text{pr}(\widetilde{Y}_i = 1 | \mathbf{Z}, \mathbf{x}_i, \boldsymbol{\theta}^{(t)})$ and $C_j$ according to equation (2.3);
- Update $\mathbf{S}_e^{(t+1)}$ and $\mathbf{S}_p^{(t+1)}$ according to equation (2.4);
- **for** $m = 1$ **to** $M$ **do**
  - For $i = 1, 2, \ldots, N$, compute the negative gradient $G_{im}$ according to equation (3.2);
  - Fit a $K$-node regression tree to a random subsample of $\{G_{im}, \mathbf{x}_i\}_{i=1}^N$ of size $\eta N$, where $\eta$ is the desired proportion of individuals to be included in the subsample. Output $\{R_{km}\}_{k=1}^K$, which form a partition of the predictor space;
  - For $k = 1, 2, ..., K$, compute the step size $\gamma_{km}$ via

$$\gamma_{km} = \left( \sum_{i: \mathbf{x}_i \in R_{km}} G_{im} \right) \Bigg/ \left( \sum_{i: \mathbf{x}_i \in R_{km}} \frac{1}{1 + e^{-F_{i(m-1)}}} \times \frac{1}{1 + e^{F_{i(m-1)}}} \right)$$

    ;
  - Update $F_{im} = F_{i(m-1)} + \tau \sum_{k=1}^K \gamma_{km} \times I(\mathbf{x}_i \in R_{km})$, where $\tau$ is the learning rate;
- **end**
- Update $F_i^{(t+1)} = \sum_{m=M-B+1}^M F_{iM}/B$, where $B$ is the desired number of iterations over which to average;

**end**

**output:** $(\widehat{\mathbf{S}}_e', \widehat{\mathbf{S}}_p', \widehat{\mathbf{F}}')'$ where these quantities are defined to be the averages of the corresponding stationary distributions.

**Algorithm 3:** Boosted Logistic Regression Model for Multiplex Group Testing Data

**input :** $\mathbf{Z}$, $\mathbf{X}$, $\mathbf{S}_e^{(0)}$, $\mathbf{S}_p^{(0)}$, $\mathbf{F}^{(0)}$

**repeat**

  For $i = 1, 2, \ldots, N$, for $j = 1, 2, \ldots, J$, and for $d = 1, 2, 3$, estimate $C_{id}$ and $C_{jd}$ according to equation (4.3);

  Update $\mathbf{S}_e^{(t+1)}$ and $\mathbf{S}_p^{(t+1)}$ according to equation (4.4);

  **repeat**

    For $p = 1, 2, \ldots, r$, compute $(\widehat{\boldsymbol{\alpha}}_p', \widehat{\boldsymbol{\lambda}}_p')' \equiv \arg\max_{(\boldsymbol{\alpha}', \boldsymbol{\lambda}')'} Q_{\boldsymbol{\alpha},\boldsymbol{\lambda}}(\boldsymbol{\alpha}, \boldsymbol{\lambda} | \boldsymbol{\theta}^{(t)})$, where $Q_{\boldsymbol{\alpha},\boldsymbol{\lambda}}(\cdot)$ is given by equation (4.5);

    For $d = 1, 2, 3$, update $F_{imd} = F_{i(m-1)d} + \tau(\widehat{\alpha}_{p^*d} + \widehat{\lambda}_{p^*d} x_{\cdot p^*})$, where $\tau$ is the learning rate and $p^*$ is the value of $p$ that results in the largest value of $Q_{\boldsymbol{\alpha},\boldsymbol{\lambda}}(\widehat{\boldsymbol{\alpha}}_p, \widehat{\boldsymbol{\lambda}}_p | \boldsymbol{\theta}^{(t)})$;

  **until** $||\mathbf{F}_m - \mathbf{F}_{(m-1)}|| < \epsilon$;

  For $d = 1, 2, 3$, update $F_{id}^{(t+1)} = F_{iMd}$, where $M$ is the iteration upon which $||\mathbf{F}_m - \mathbf{F}_{(m-1)}|| < \epsilon$;

**until** $|Q(\boldsymbol{\theta}^{(t+1)} | \boldsymbol{\theta}^{(t)}) - Q(\boldsymbol{\theta}^{(t)} | \boldsymbol{\theta}^{(t-1)})| < \epsilon$;

**output:** $(\mathbf{S}_e^{(T)'}, \mathbf{S}_p^{(T)'}, \mathbf{F}^{(T)'})'$ where $T$ is the iteration upon which the algorithm converges.

**Algorithm 4:** Boosted Regression Trees Model for Multiplex Group Testing Data

**input :** $\mathbf{Z}$, $\mathbf{X}$, $\mathbf{S}_e^{(0)}$, $\mathbf{S}_p^{(0)}$, $\mathbf{F}^{(0)}$

**for** $t = 0$ **to** $T$ **do**

  For $i = 1, 2, \ldots, N$, for $j = 1, 2, \ldots, J$, and for $d = 1, 2, 3$, estimate $C_{id}$ and $C_{jd}$ according to equation (4.3);

  Update $\mathbf{S}_e^{(t+1)}$ and $\mathbf{S}_p^{(t+1)}$ according to equation (4.4);

  **for** $m = 1$ **to** $M$ **do**

    **for** $d = 1$ **to** $3$ **do**

      For $i = 1, 2, \ldots, N$, compute the negative gradient $G_{imd}$ according to equation (4.6);

      Fit a $K$-node regression tree to a random subsample of $\{G_{imd}, \mathbf{x}_i\}_{i=1}^{N}$ of size $\eta N$, where $\eta$ is the desired proportion of individuals to be included in the subsample. Output $\{R_{kmd}\}_{k=1}^{K}$, which form a partition of the predictor space;

      For $k = 1, 2, ..., K$, compute the step size $\gamma_{kmd}$ via

$$\gamma_{kmd} = \frac{\sum_{i:\mathbf{x}_i \in R_{kmd}} G_{imd}}{\sum_{i:\mathbf{x}_i \in R_{kmd}} \frac{e^{F_{i(m-1)d}}}{1+\sum_{b=1}^{3} e^{F_{i(m-1)b}}} \times \left(1 - \frac{e^{F_{i(m-1)d}}}{1+\sum_{b=1}^{3} e^{F_{i(m-1)b}}}\right)}$$

      ;

      Update $F_{imd} = F_{i(m-1)d} + \tau \sum_{k=1}^{K} \gamma_{kmd} \times I(\mathbf{x}_i \in R_{kmd})$, where $\tau$ is the learning rate;

    **end**

  **end**

  Update $F_{id}^{(t+1)} = \sum_{m=M-B+1}^{M} F_{iMd}/B$, where $B$ is the desired number of iterations over which to average;

**end**

**output:** $(\widehat{\mathbf{S}}_e', \widehat{\mathbf{S}}_p', \widehat{\mathbf{F}}')'$ where these quantities are defined to be the averages of the corresponding stationary distributions.

**Algorithm 5:** Generalized Additive Model for Group Testing Data

**input :** $\mathbf{Z}$, $\mathbf{X}$, $\mathbf{S}_e^{(0)}$, $\mathbf{S}_p^{(0)}$, $\mathbf{F}^{(0)}$

**repeat**

  For $i = 1, 2, \ldots, N$, and for $j = 1, 2, \ldots, J$, estimate $\text{pr}(\widetilde{Y}_i = 1|\mathbf{Z}, \mathbf{x}_i, \boldsymbol{\theta}^{(t)})$ and $C_j$ according to equation (2.3);

  Update $\mathbf{S}_e^{(t+1)}$ and $\mathbf{S}_p^{(t+1)}$ according to equation (2.4);

  **repeat**

    For $i = 1, 2, \ldots, N$, form the adjusted response variable

$$Y_i^* = F_{i(m-1)} + \left(\text{pr}(\widetilde{Y}_i = 1|\mathbf{Z}, \mathbf{x}_i, \boldsymbol{\theta}^{(t)}) - P_{i(m-1)}\right)\left(\frac{\partial F_i}{\partial P_{i(m-1)}}\right),$$

    where $F_i \equiv H\{P_i\}$ is given by equation 5.1;

    For $i = 1, 2, \ldots, N$, form the weights $W_i = \left(\frac{\partial P_i}{\partial F_{i(m-1)}}\right)^2 V^{-1}$, where $V$ is the variance of the response at the fitted values;

    For $q = 1, 2, \ldots, r$, update $g_{qm}$ by fitting a smooth function to the partial residuals $\mathbf{R}_q$, which are given by

$$\mathbf{R}_q = \mathbf{Y}^* - \hat{\beta}_0 - \sum_{k=1}^{q-1} g_{km}(\mathbf{X}_k^*) - \sum_{k=q+1}^{r} g_{k(m-1)}(\mathbf{X}_k^*),$$

    where $\mathbf{X}_k^*,\ k = 1, 2, \ldots, q-1, q+1, \ldots, r$, are the weighted covariates;

    Update $F_{im} = \hat{\beta}_0 + \sum_{q=1}^{r} g_{qm}(X_q)$

  **until** $||\mathbf{F}_m - \mathbf{F}_{(m-1)}|| < \epsilon$;

  Update $F_i^{(t+1)} = F_{iM}$, where $M$ is the iteration upon which $||\mathbf{F}_m - \mathbf{F}_{(m-1)}|| < \epsilon$;

**until** $|Q(\boldsymbol{\theta}^{(t+1)}|\boldsymbol{\theta}^{(t)}) - Q(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t-1)})| < \epsilon$;

**output:** $(\mathbf{S}_e^{(T)\prime}, \mathbf{S}_p^{(T)\prime}, \mathbf{F}^{(T)\prime})'$ where $T$ is the iteration upon which the algorithm converges.