INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600
Evaluation of Candidate Genes in Family Studies: Generalized Estimating Equations and Bootstrap Approaches

by

Janey Shin

A Thesis submitted in conformity with the requirements for the Degree of Master of Science
Graduate Department of Community Health
University of Toronto

© Copyright by Janey Shin, 1998
The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author’s permission.

L’auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L’auteur conserve la propriété du droit d’auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-40723-3
Evaluation of Candidate Genes in Family Studies: Generalized Estimating Equations and Bootstrap Approaches

Master of Science - 1998
Janey Shin
Graduate Department of Community Health
University of Toronto, Canada

Abstract

In family data, observations within the same family cannot be assumed to be independent. In genetic epidemiological studies, when the goal is to make inferences on the population average of certain covariate effects, one must not ignore the correlation in the data analysis. Two methods for analysing binary family data are evaluated. One method is the generalized estimating equations approach (GEEs). This is a semiparametric method that requires that the mean structure and a working correlation matrix are specified. The GEE is mainly used to model the relationship between covariates and the response, while treating the correlation as a nuisance parameter. The other method is a nonparametric bootstrap approach. A hierarchical bootstrap approach is proposed that involves two strategies to handle dependent data from nested structures, since the family structure can be viewed as such. A one-step logistic regression model is applied on bootstrap data ignoring the family structure, letting the bootstrap determine the variation. A simulation study is performed to compare the two methods using a multivariate logit-normal model to generate family data with a general correlation structure. An underlying multifactorial genetic model is assumed where affected status is determined by multiple candidate genes with both additive and dominant effects, an environmental factor, and a residual familial component that would exhibit the effect of unmeasured genes and/or environmental effects. The GEE and bootstrap methods are compared under different scenarios given in the simulation study. An application is made to a Genetic Analysis Workshop 9 data set.
ACKNOWLEDGEMENTS

I would like to give a million thanks to my supervisor, Dr. Shelley Bull, who was a wonderful, patient teacher and who always seemed to have unlimited time to help me.

I would also like to thank my committee members, Dr. Gerarda Darlington and Dr. Mary Corey, for helpful discussions and comments. Their encouragement and kind words were extremely helpful during the tumultuous times while I was putting this thesis together.

I would also like to thank Dr. Mary Corey for her patience and understanding during the last few months leading up to my defense.

I would also like to thank Professor Paul Corey for giving me a chance to pursue a career in biostatistics.

I would also like to thank my fellow biostatistics students: Carmen, Celia, Fang, Mike and Rafal for helpful discussions and company on many late nights in the computer room.

I would also like to thank the gang at the Hospital for Sick Children: Lenny and Namita for their friendship.

I would also like to thank my best friends Lisa, Christina, Gillian and Mike for providing a network of support and a lot of laughs.

I would also like to thank the bobos for just being bobos.
I would also like to thank my mother, Rita, and my brother, Richard, for their unconditional love and support. I appreciate their patience during those times I neglected to see them.

Finally, I would like to thank my special friend Peter who always encouraged me, supported me and who was always by my side. I'll always appreciate his patience and his unconditional love.
Contents

1 Introduction .................................................. 1
  1.1 Some Background in Genetics ........................... 3

2 An Overview of the GEE and Bootstrap Approaches for the Analysis
   of Association in Correlated Binary Data .................. 10
  2.1 Generalized Linear Models .............................. 12
  2.2 Quasi-Likelihood Functions ............................. 16
  2.3 Generalized Estimating Equations ....................... 17
     2.3.1 Estimation and Testing ........................... 17
     2.3.2 Performance of the GEE Method .................. 21
     2.3.3 Summary and Discussion .......................... 25
  2.4 Misspecification Effects for Models Fitted to Dependent Data .... 27
  2.5 The Bootstrap ............................................ 30
     2.5.1 Background .................................... 30
     2.5.2 Bootstrap Estimates of Standard Error and Bias ... 32
     2.5.3 Bootstrap Confidence Intervals ................... 33
     2.5.4 Some Bootstrap Methods Applied to Dependent Data .. 36
     2.5.5 Hierarchical Bootstrap ........................... 38
     2.5.6 Bootstrapping Logistic Regression Models .......... 42
     2.5.7 Summary of the Bootstrap Method .................. 43
  2.6 Performance of the Bootstrap versus the GEE ............ 44
2.6.1 Conclusion ................................................. 48

3 A Multifactorial Model for Family Data

with a Note on Cluster-Specific and Population Averaged Approaches 50

3.1 Generation of Nuclear Pedigrees ................................................. 51
  3.1.1 Genetic and Environmental Covariates ................................................. 51
  3.1.2 Familial Correlations .......................................................... 52
  3.1.3 Model Based Simulation ......................................................... 54
  3.1.4 Choice of Values for Logit-Normal Variance ................................................. 58

3.2 Population-Averaged Approaches vs Cluster-Specific Approaches ... 60

4 Design and Analysis of the Simulation Study 68

4.1 Introduction ................................................................. 68

4.2 Statistical Methods .......................................................... 69
  4.2.1 Some Definitions .......................................................... 69
  4.2.2 Measures of Performance ......................................................... 69
  4.2.3 Numerical Issues .......................................................... 71

4.3 Stage One: The Preliminary Simulation Study for the Selection of Parameter Values ......................................................... 72
  4.3.1 Methods ................................................................. 72
  4.3.2 Results ................................................................. 75

4.4 Stage Two: The Simulation Results for the Evaluation of GEE and Bootstrap Approaches ......................................................... 82
  4.4.1 Percent Relative Bias of Regression Coefficients ................................................. 82
  4.4.2 Efficiency of Regression Coefficients ......................................................... 82
  4.4.3 Coverage Probabilities of Regression Coefficients ................................................. 86
  4.4.4 Analysis of Variance of Coverage Probabilities ................................................. 91

4.5 Recommendations for Practice ................................................. 100
List of Tables

1.1 Transmission Probabilities for a single autosomal locus with two alleles 5
3.1 Point Estimates (standard errors) of regression coefficients for 4 approaches applied to simulated family data .......................... 65
3.2 Wald statistics to test for covariate effects for 4 approaches applied to simulated family data .......................... 65
4.1 Expected Prevalence for various intercept and variance combinations 74
4.2 Expected Prevalence for various intercept and variance combinations 78
4.3 Empirical intra- and inter-familial correlations for affection status estimated by averaging over 500 replicates .......................... 80
4.4 Empirical intra- and inter-familial correlations for covariates CG1, CG2 and EF estimated by averaging over 500 replicates. ................. 80
4.5 Misspecification Effect (ME) for $\beta_1=CG1$ slope, $\beta_2=CG2$ slope, $\beta_3=CG3$ slope over 500 simulations. ..................... 81
4.6 Percent Relative Bias for $\hat{\beta}_1=CG1$ slope estimate over 500 simulations. 83
4.7 Percent Relative Bias of $\hat{\beta}_2=CG2$ slope estimate over 500 simulations. 84
4.8 Percent Relative Bias of $\hat{\beta}_3=EF$ slope estimate over 500 simulations. 85
4.9 Variance of $\hat{\beta}_1=CG1$ slope estimate over 500 simulations. .......... 87
4.10 Variance of $\hat{\beta}_2=CG2$ slope estimate over 500 simulations. .......... 88
4.11 Variance ($\times 10^{-2}$) of $\hat{\beta}_3=EF$ slope estimate over 500 simulations. .. 89
4.12 Observed Coverage of the nominal 95% Confidence Interval (with Average Interval Length) for $\hat{\beta}_1 = CG_1$ slope estimate over 500 simulations.

4.13 Observed Coverage of the nominal 95% Confidence Interval (with Average Interval Length) for $\hat{\beta}_2 = CG_2$ slope estimate over 500 simulations.

4.14 Observed Coverage of the nominal 95% Confidence Interval (with Average Interval Length) for $\hat{\beta}_3 = EF$ slope estimate over 500 simulations.

4.15 The number of misses above (upper) and below (lower) the 95% confidence intervals for $\hat{\beta}_1 = CG_1$ slope estimate under the null model.

4.16 The number of misses above (upper) and below (lower) the 95% confidence intervals for $\hat{\beta}_2 = CG_2$ slope estimate under the null model.

4.17 The number of misses above (upper) and below (lower) the 95% confidence intervals for $\hat{\beta}_3 = EF$ slope estimate under the null model.

4.18 Comparison of methods using contrasts for $\beta_1 = CG_1$ slope estimate.

4.19 Comparison of methods using contrasts for $\beta_2 = CG_2$ slope estimate.

4.20 Comparison of methods using contrasts for $\beta_3 = EF$ slope estimate.

4.21 The average of the logit of the coverage probabilities and average of the coverage probabilities over all sample sizes and models.

4.22 Coverage probabilities for model 5 in table 4.2 using different starting seeds.

5.1 Coefficient estimates (standard errors) model fitted to GAW9 data using C5, EF (centered), age and sex.

5.2 Misspecification Effects for GAW9 data set.

5.3 95% confidence intervals for the regression coefficients using the GEE with length in brackets for GAW9 data set.

5.4 95% confidence intervals for the regression coefficients using the bootstrap.

B.1 Prevalence for various parameter combinations with $\sigma^2 = 9$. 

ix
List of Figures

2.1 Structure of a Nuclear Pedigree Seen As A Nested Design .................. 40

3.1 The univariate logit-normal density with different variances .............. 57

4.1 Power Curves for $\hat{\beta}_1$ with $\sigma^2 = 9, 16, 25$. .................. 76
4.2 Power Curves for $\hat{\beta}_2$ with $\sigma^2 = 9, 16, 25$. .................. 77

5.1 Generating model for GAW9 data set ........................................ 102
5.2 Chromosomal locations of major loci and candidate loci for GAW9 data set. .......................................................... 103
5.3 Bootstrap distribution for C5 in GAW9 data set. .......................... 107
Chapter 1

Introduction

In genetic epidemiological studies, association of disease of unknown etiology with genetic markers or candidate genes as risk factors may suggest the presence of susceptibility loci if an association is present. Candidate genes are genes that have been localized and mapped and have a known or suspected function in the etiology of a disease. When candidate genes are identified, association studies (as in cohort studies where family data are collected) can be used to examine the impact of such candidate genes with other shared risk factors on complex diseases. Standard statistical methods that test for association between these risk factors and disease where individual family members are treated as independent are not valid since they ignore the correlation among family members.

For quantitative traits, statistical methods that incorporate the familial correlation are well-established because most of the models belong to a rich class of distributions such as the multivariate Normal. However for qualitative traits, parametric models for correlated categorical outcomes are not as well developed since the joint distributions for nonnormal data are difficult to construct. Methods that can incorporate the correlation among binary responses include a generalized estimating equation (GEE) approach (Liang and Zeger, 1986) which is an extension of a generalized linear model, a generalized linear mixed model (GLMM) approach which allows for both
random and fixed effects (Laird and Ware, 1982), and various conditional logistic regression (CLR) models (Rosner, 1984, Connelly and Liang, 1988 and Breslow and Day, 1980) (see chapter 3).

The goal of this thesis is to evaluate two statistical methods that both avoid strict distributional assumptions for correlated binary data. A semi-parametric approach is taken since it is often difficult for an investigator to determine or even postulate the underlying correlation structure of the data which is required for parametric methods. A semi-parametric method avoids having to specify this correlation structure. One popular semi-parametric method used to analyze correlated binary data is the GEE approach. The GEE is a marginal model that is an extension of the generalized linear model, but only the mean and the variance relationship need to be specified. This method is appealing because it does not require a joint distribution for the correlated measurements. An alternative to the GEE is the nonparametric bootstrap which has also become popular as a nonparametric method for providing inferential statistics but has not been widely used as a method to analyze correlated data in particular, correlated binary data in the regression setting. The bootstrap is a resampling method that exploits computational effort in place of theoretical asymptotic results. Both methods are simple to apply. Also, when the bootstrap is used to estimate a parameter from a marginal model, both methods have intuitive parameter interpretations. That is, both have a population-averaged interpretation which is often the scientific question of interest in association studies.

This thesis presents an evaluation and comparison of these approaches to analyze family data in the presence of large unexplained variability such as unmeasured genetic and environmental factors. In addition, different methods used to analyze correlated binary data will be examined which differ according to research interests and lead to different interpretations of the regression coefficients.

The structure of the thesis is as follows: in chapter 1, some basic concepts in
genetics will be presented; in chapter 2, some background on the GEE and the non-parametric bootstrap approach will be presented along with a review of relevant literature for each of the methods; in chapter 3, a multifactorial model for family data is introduced as well as a discussion on cluster-specific and population-averaged models; in chapter 4 the design and analysis of the simulation study will be presented; in chapter 5, an application using the GEE and bootstrap will be presented on a simulated data set given at the Genetics Analysis Workshop 9 (GAW9); and in chapter 6, a discussion and some remarks about areas for further research will be presented.

1.1 Some Background in Genetics

Most of the material presented in this section was taken from Vogel and Motulsky (1997), Thompson et al. (1991) and Khoury et al. (1994).

The simplest genetic model is derived from Mendel who showed that it was possible to make observations about the nature of the mode of transmission of information from parents to offspring. The basic underlying concept of this model is that each individual carries two discrete genes that are of allelic forms which then define the characteristic for the trait of interest. The two genes are segregated from parent to offspring where one is a randomly chosen copy of two genes from the father and the other, a copy of those carried by the mother. A genotype is an unordered pair of genes carried by the individual.

These genes are located on chromosomes which are made up of deoxyribonucleic acid (DNA), the basic genetic material. Every cell in normal human individuals contains forty-six chromosomes where forty-four of them are autosomes. Thus for all twenty-three pairs, one set will be inherited from the father and the other set from the mother. An autosomal trait is affected by genes lying on an autosomal locus (which is a position on the chromosome). The remaining two chromosomes are the sex chromosomes or the X and Y chromosomes. Genes on these two chromosomes
segretate differently for males. A male has one X chromosome which is randomly chosen from his mother's two X chromosomes and one Y chromosome inherited from his father which is then passed onto each subsequent son. Conversely for a female, one X chromosome is inherited from her mother and one from her father.

A phenotype is an observed trait. Examples are the colour of one's hair or eyes, one's height or being predisposed to a disease. A simple genetic model consists of a dichotomous phenotype for a trait resulting from a genotype at a single autosomal locus. This dichotomous phenotype will often be classified as affected and unaffected or normal. Several terms are used to describe how the susceptibility to a phenotype is related to an underlying genotype. Assume there are only two alleles present at a locus, $A_1$ and $A_2$, which result in three possible genotypes: $A_1A_1$, $A_1A_2$, and $A_2A_2$. Individuals who carry two different alleles are heterozygous and those who carry the same alleles are homozygous. If the result of having a particular disease depends on having only one copy of an abnormal allele, the disease is dominant. That is, if allele $A_1$ is dominant over allele $A_2$ then the two genotypes, $A_1A_1$ and $A_1A_2$ are indistinguishable with respect to phenotype. However, if two alleles are required, that is, an affected person is homozygous for the disease allele ($A_1A_1$), then the disease is recessive. Heterozygotes with a recessive allele who are unaffected but carry one copy of the $A_1$ allele, are known as carriers. Alleles can also influence a trait additively, that is, the severity of disease influenced by an additive gene would vary depending on the allele combination. So if $A_2$ was the more 'dominant' allele, then in terms of trait severity, the genotype $A_2A_2 > A_2A_1 > A_1A_1$.

A mating type is the representation of a parental mating at a locus where either one parent is homozygous and the other is heterozygous or both homozygous or both heterozygous. In simple Mendelian models, each mating type transmits a particular genotype to an offspring with some probability which can be formulated as a transmission parameter. Under a single-locus two-allele model, it can shown that there are three basic transmission parameters to specify all the genotypic probabilities for
Table 1.1: Transmission Probabilities for a single autosomal locus with two alleles

offspring. Taken from Khoury (1993), table 1.1 shows the transmission probabilities for all possible mating types for a single-locus two-allele ($A_1$ and $A_2$) model.

Penetrance is defined as the probability of observing a particular phenotype conditional on having a particular genotype. A disease is fully penetrant when an abnormal allele(s) always leads to the development of disease. Conversely, a disease can also have incomplete penetrance when the same set of abnormal allele(s) do not necessarily lead to the development of disease. One example is retinoblastoma, a rare malignant tumor of the retina in infants that is inherited as dominant trait. Some individuals that carry the mutant allele are not affected but will have affected parents and affected children (Khoury, 1993).

In population genetics, one important principle for understanding the relationship between the frequency of genotypes in a population and the prevalence of a Mendelian disease is known as the Hardy-Weinberg (H-W) principle. This principle states that allele frequencies in a population will remain constant given that there is random mating, no selection, no mutation and no migration. When the population is in H-W equilibrium, that is if all of the previous conditions are met, the genotype and phenotype frequencies can be estimated based on allele frequencies, and, the frequencies of genotypes will be the same in all subsequent generations.
The genetic transmission of a disease phenotype can be modelled by using a two-allele ($A_1$ and $A_2$) single locus model,

$$V = p^2 f_{A_1A_1} + 2pq f_{A_1A_2} + q^2 f_{A_2A_2}$$ \hspace{1cm} (1.1)

where $V$ is the population prevalence, $p$ the allele frequency for $A_1$ and $q = 1 - p$ the allele frequency for $A_2$, and $f_{A_1A_1}$, $f_{A_1A_2}$ and $f_{A_2A_2}$ are the penetrances for the three possible genotypes produced by this model, $A_1A_1$, $A_1A_2$, and $A_2A_2$ respectively. The penetrances are allowed to vary between 0 and 1. Therefore, one can estimate the frequency of the disease phenotype if given the estimates of the allele frequency and penetrances.

Mendelian traits have a simple pattern of inheritance that is determined by a single genetic locus with complete penetrance and no phenocopies (phenotypes not governed by the underlying disease allele(s) but expressing the same attributes as the true phenotype). On the other hand, if a trait is assumed to be governed by more than one locus, has incomplete penetrance or has phenocopies present with the same trait, then the disease is described as having a complex inheritance pattern. A complex genetic model can be polygenic (many genes with small additive effects) or oligogenic (a small collection of genes each having a major effect on disease). When an environmental influence is involved, then the pattern of inheritance is assumed to be multifactorial. In a multifactorial model, the phenotype can be modelled as a function of independent, unobservable genetic and environmental factors. Since family members share genes and common environments, the covariance or correlation among relatives can be separated into components attributable to genetic factors and components attributable to environmental factors. Usually multifactorial inheritance disorders are common disorders that appear to run in families. Examples are coronary artery disease, diabetes mellitus, hypertension, obesity, many forms of cancer, as well as common psychiatric illnesses, such as bipolar disorder and schizophrenia. These multifactorial models may involve one, few or many genes located at separate loci which may be on one or several chromosomes.
The variability of a trait can be decomposed into separate components of variance attributed to genetic and environmental components. Amos (1994) and Lange et al (1976) use a variance-components method to assess evidence for genetic linkage (linkage of a disease trait to a specific genetic location). The variability among the responses for a quantitative trait from pedigree members can be expressed in terms of fixed effects from covariates, effects due to an unobserved genetic component(s) and residual nongenetic variance such as an environmental measure. For example, suppose the trait value of the \( i \)th relative is,

\[
Y_i = \mu + g_i + G_i + \sum_{k=1}^{p} \beta_k x_{ik} + e_i
\]

where \( \mu \) is the overall mean, \( g_i \) is a fixed unobserved major gene component, \( G_i \) is a random polygenic effect, \( \beta_k \) is a covariate effect of the measured \( x_{ik} \) that is independent from the genetic factors and \( e_i \) is the residual variability uncorrelated with the genetic factors and covariates. When the phenotype is discrete, the statistical model assumes that there is an underlying continuous trait that determines disease. This idea is used later in the thesis, in the design of the simulation study to simulate unobserved genetic and environmental variance.

Recently, attention has focussed on using candidate genes or genetic markers in various forms of association studies. One study design for examining associations is the case-control approach where unrelated affected individuals (cases) are compared to unrelated unaffected individuals (controls) (Lilienfeld and Lilienfeld, 1980). If the frequency of an allele for a gene of interest is observed to be significantly higher among the cases than the controls, then the allele is suspected to be associated with disease. Another study design used in association studies is the cohort approach. This approach compares the frequency of disease in those with a specific genotype to those without. A special case of the cohort design would involve a random selection of families. Another method used in association studies is the transmission disequilibrium test (TDT) (Spielman et al., 1993). The TDT uses parents who are heterozygous (both having different alleles) with an allele suspected to be associated with disease.
The TDT examines the frequency of the suspected allele that is being transmitted to the affected offspring and compares the number of alleles that are transmitted from the parents to an affected child to the number of alleles that are not transmitted. This approach has focussed on locating genes but can also be used to evaluate candidate genes. In this thesis, focus is on statistical association of candidate genes adjusted for other risk factors using family data. In this framework, either between or within association in families may be examined depending on the research question.

Candidate gene approaches have been successful for understanding a number of diseases such as, the role of the amyloid precursor protein gene in Alzheimer’s disease (Murphy, 1992) and the human leukocyte antigens (HLAs) on chromosome 6 which were first to be associated with insulin-dependent diabetes mellitus (IDDM). The association of particular HLA markers with IDDM is an example where associations (using a case-control approach) in genetic analysis can be used to identify susceptibility genes in complex disorders (Cox and Bell, 1989). Interest in using candidate genes has been mainly in cases where multifactorial or complex diseases are studied. Often in such diseases, the etiology is unclear, there is possible genetic heterogeneity (multiple genes acting independently) with possible genes acting additively (polygenic inheritance) and low penetrance from susceptible genotypes, all which make it increasingly difficult to identify the susceptibility genes by linkage analysis.

As the use of association studies in complex genetic disorders increases, so does the need for robust statistical methods that use family data to examine statistical or epidemiological association. It is likely that as more genes are identified, association studies that examine the influence of genes and other familial and environmental risk factors to disease will become more widely used. This was the motivation for evaluating statistical methods for use in family data. For association studies that use a cohort design, a population-averaged approach such as the generalized estimating equations (GEE) approach is attractive because it is conceptually simple for investigators to use and interpret. However, limitations to this approach include specification
of the underlying correlation structure when a model-based covariance structure is used (Sullivan Pepe and Anderson, 1994). In this thesis, the nonparametric bootstrap is proposed as a method that avoids having to make correlation assumptions. These two methods, the GEE and the nonparametric bootstrap, are evaluated as tools for association studies.
Chapter 2

An Overview of the GEE and Bootstrap Approaches for the Analysis of Association in Correlated Binary Data

In clustered binary data, observations within the same cluster cannot be assumed to be independent. In many cases, the variation within a cluster will likely be smaller among similar subcluster units than the variation between independent clusters. Some examples are longitudinal studies where observations are collected for each subject over a period of time, multistage survey sampling data where subunits are sampled to reduce cost, and, association studies in families where family members share genetic characteristics and environmental influence(s). When the goal is to make inferences on the population average of certain covariate effects, ignoring the clustering of data in the analysis would be inappropriate. Donner (1984) showed that when analyzing correlated data using linear regression, the standard error of a regression coefficient computed from ordinary least squares underestimated/overestimated the true
standard error based on whether the within-cluster correlation of the corresponding covariate was positive/negative. Others (Scott & Holt, 1982, Liang & Zeger, 1986, Neuhaus & Segal, 1993) have also shown that this problem exists when the correlation is not taken into account properly. In the section on misspecification effects (2.3) it will be shown that it is inappropriate to use standard statistical methods for independent observations in the analysis of dependent data since this will often give misleading statistical results.

The following notation will be used throughout the thesis. Let $y_{ij}$ represent the outcome or phenotype for individual $j$ in family $i$ for families $i = 1, \ldots, m$ comprised of $j = 1, \ldots, n_i$ family members. Each outcome $y_{ij}$ is associated with a vector of covariates $x_{ij} = [x_{1ij}, x_{2ij}, \ldots, x_{ pij}]$ that may consist of genetic and environmental factors. Family data will consist of observations that are correlated within each family unit since family members are likely to share similar genetic and environmental conditions. In this thesis, the analyses of nuclear families will include information from two parents and a variable number of offspring. In nuclear families, it can be assumed that there are three different correlation coefficients: between parents ($\rho_{PP}$), between siblings ($\rho_{SS}$) and between a parent and offspring ($\rho_{PS}$), each of which may represent genetic heritability and/or external influences such as a common environmental effect. Correlation between family members can in general, be classified into two broad categories (Eliasziw and Donner, 1991): 1) *intra*class correlations that measure the association for a particular trait between two members from the same class of individuals in a family such as sib-sib, cousin-cousin correlations and; 2) *inter*class correlations that measure the association between two members from different classes of individuals in a family such as father-mother, mother-daughter, sister-brother.

This chapter will begin with some background on generalized linear models and quasi-likelihood models. Generalized linear models go beyond the classical linear model by describing data through a class from the exponential family of distributions.
An extension of the generalized linear model known as the generalized estimating equations (GEEs) is based on the quasi-likelihood which provides more flexibility by relaxing the distributional assumptions. This is followed by a discussion of a measure known as the *misspecification effect*. This is used to assess the extent of the clustering effect on the variances of the parameters in a regression model when correlation is not accounted for in the modelling process. Finally, an overview of the two methods that will be used for analyzing correlated data will be given; the generalized estimating equations approach and the bootstrap approach.

### 2.1 Generalized Linear Models

The generalized linear model (GLM) introduced by Nelder and Wedderburn (1972) allows the modelling of both continuous and discrete data in a unified way by using a general family of distributions for fitting linear models. It assumes that the underlying distribution of the responses is a member of the exponential family which includes the binomial, Poisson and normal distributions. This section provides a short overview of GLMs.

Assume that the observed data consist of a set of independent observations, \( y_i, i = 1, 2, \ldots, n \) where \( y_i \) has mean, \( \mu_i = E(y_i) \) and variance \( \sigma_i^2 = \text{var}(y_i) \) and where each is independently observed at fixed covariate values, \( x_1, \ldots, x_p \). A GLM is comprised of three components (McCullagh and Nelder, 1992):

1. Response variables \( y_1, \ldots, y_n \) which share the same distribution function from the exponential family and are independent,

\[
f(y_i; \theta_i, \phi) = \exp\{(y_i\theta_i - b(\theta_i))/a(\phi) + c(y_i, \phi)\}
\]

(2.1)

where \( \phi \) is the *overdispersion parameter* and parameter \( \theta_i \), specifies the distribution of \( y_i \). This is known as the random component.
2. A $p$-vector of parameters $\beta = (\beta_1, \cdots, \beta_p)^T$ and a corresponding $p$-vector of predictor variables $x = (x_1, \cdots, x_p)$ which define a linear predictor $\eta_i = x_i \beta, i = 1, \cdots, n$. This is known as the systematic component.

3. A monotone differentiable link function $g$ such that $g(\mu_i) = \eta_i$ relates the mean response, $\mu_i = E(y_i)$ to the linear predictor $\eta_i = x_i \beta$.

In GLMs, the variance function $\nu(\cdot)$ relates the variance to the mean by $\text{var}(y_i) = a(\phi)\nu(\mu_i)$.

Parameter estimates in a generalized linear model are obtained by the method of maximum likelihood. This is usually computed numerically by an iterative re-weighted least squares (IRLS) procedure. If $f(y_i|\theta_i, \phi)$ is the density or probability mass function for the observation $y_i$ given $\theta$ and $\phi$, then the individual contribution to the $i$th log likelihood would be,

$$l_i(\theta_i) = \log f(y_i|\theta_i, \phi)$$

Since $y_1, \cdots, y_n$ are independent random variables, the log-likelihood of the sample is the sum of the individual contributions,

$$l(\beta) = \sum_i l_i(\beta).$$

(2.2)

The maximum-likelihood estimates are computed by solving the $p$-dimensional score equations which are the first derivatives with respect to $\beta$ of (2.2) set equal to 0,

$$U^{GLM}(\beta) = \frac{\partial l}{\partial \beta} = \sum_i U^{GLM}_i(\beta) = \sum_{i=1}^n \left( \frac{\partial \mu_i(\beta)}{\partial \beta} \right)^T \text{var}(y_i)^{-1}(y_i - \mu_i(\beta)) = 0_{p \times 1}$$

(2.3)

where $\left( \frac{\partial \mu_i(\beta)}{\partial \beta} \right)^T$ is a $p \times 1$ matrix. These likelihood or score equations are generally nonlinear and must be solved iteratively using a Fisher scoring method. Starting with
an initial estimate $\hat{\beta}^0$, each Fisher scoring iteration is computed as,

$$\hat{\beta}^{(t+1)} = \hat{\beta}^{(t)} + F^{-1}(\hat{\beta}^{(t)})U^{GLM}(\hat{\beta}^{(t)}), \quad t = 0, 1, 2, \cdots \quad (2.4)$$

where $F(\beta)$ is the expected Fisher information matrix,

$$F(\beta) = \text{cov}[U^{GLM}(\beta)] = \left[ \frac{\partial \mu_i(\beta)}{\partial \beta} \right]^T \text{cov}(y) \left[ \frac{\partial \mu_i(\beta)}{\partial \beta} \right]$$

and $U^{GLM}(\hat{\beta}^{(t)})$ is the vector of first derivatives ($\partial l/\partial \beta$) evaluated at $\beta = \hat{\beta}^{(t)}$. It can be shown that the iterative equation used in the method of scoring (2.4) can be written as a form similar to the normal equations used in the weighted least squares estimation for the linear model. If both sides of (2.4) are multiplied by $F(\hat{\beta}^{(t)})$ then,

$$F(\hat{\beta}^{(t)})\hat{\beta}^{(t+1)} = F(\hat{\beta}^{(t)})\hat{\beta}^{(t)} + U^{GLM}(\hat{\beta}^{(t)}) \quad (2.6)$$

$F(\beta)$ can be rewritten as,

$$F(\beta) = XTWX$$

where $X$ is the design matrix and $W$ is the $n \times n$ diagonal weight matrix with elements,

$$w_{ii} = \frac{1}{\text{var}(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2.$$

Elements from the vector on the right-hand of (2.6) can be written as

$$\sum_{i} \frac{x_{ij} x_{ik}}{\text{var}(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)^2 \beta_k^{(t)} + \sum_{i} \frac{(y_i - \mu_i) x_{ij}}{\text{var}(y_i)} \left( \frac{\partial \mu_i}{\partial \eta_i} \right)$$

where $A$ is the $j,k^{th}$ element of $F(\beta)$ evaluated at $\hat{\beta}^{(t)}$ and $B$ is the $j^{th}$, $j = 1, \cdots, p$ element of $U^{GLM}(\beta)$ evaluated at $\hat{\beta}^{(t)}$. The expression, $F(\beta)\hat{\beta}_{(t+1)} = XTWX\hat{\beta}$ is equivalent to the matrix expression, $XTWXz$. The elements in the vector $z$ are,

$$z_i = \sum_k x_{ik} \beta_k^{(t)} + (y_i - \mu_i) \left( \frac{\partial \mu_i}{\partial \eta_i} \right)$$
where \( \mu_i \) and \( \partial \eta_i / \partial \mu_i \) are evaluated at \( \hat{\beta}^{(t)} \). Hence, the maximum likelihood estimators in a generalized linear model are obtained by an IRLS procedure,

\[
X^T W X \hat{\beta}^{(t)} = X^T W z
\]

which is similar to the normal equations for a linear model. Since \( z \) and \( W \) depend on \( \beta \), these equations (2.7) must be solved iteratively. Iterations are stopped when some criterion that compares the new estimate of \( \hat{\beta}^{(t+1)} \) to the old estimate of \( \hat{\beta}^{(t)} \) is attained.

In this thesis, focus is limited to models with binary outcomes. Binary data, \( y_i, i = 1, 2, \cdots, n \), which can only take on two possible values, 0 or 1, are usually modelled by the ordinary logistic regression model,

\[
\text{logit}(Pr(y_i = 1)) = \logit(p_i) = \log \left( \frac{p_i}{1 - p_i} \right) = x_i^T \beta
\]

where \( x_i \) is the \( i \)th \( p \times 1 \) covariate vector, \( \beta \) a \( p \times 1 \) parameter vector and \( p_i \) is the response probability. The link function is the logit function which is \( \log(p_i/(1 - p_i)) \). The mean and the variance of the binomial distribution are given by \( E(y_i) = np_i \) and \( \text{var}(y_i) = np_i(1 - p_i) \) respectively. It can easily be seen that any value of \( p_i \) which lies in the range \([0,1]\), corresponds to a value of \( \logit(p_i) \) which lies in the range \([-\infty, \infty]\]. That is, when \( p \to 0 \) then \( \logit(p) \to -\infty \); as \( p \to 1 \) then \( \logit(p) \to \infty \). Another feature of the logistic function is that it is approximately linear between \( p=0.2 \) to \( p=0.8 \) and non-linear outside this range so it appears in the form of a sigmoid curve (Collet, 1993). The \( \logit(p_i) \) is the \textit{log of the odds of a success}. The exponential of each \( \beta \) parameter is interpreted as an \textit{odds ratio}. For example, in a model with one dichotomous covariate, \( \exp(\beta) \) would be interpreted as the odds ratio of a positive response when \( x=1 \) relative to \( x=0 \). Similarly, if \( x \) was continuous, \( \exp(\beta) \) would be the multiplicative increase in odds associated with one unit increase in \( x \).
2.2 Quasi-Likelihood Functions

An extension of the generalized linear model is the quasi-likelihood (McCullagh and Nelder, 1989). Quasi-likelihoods (QL) provide more flexibility in applications than GLMs because unlike GLMs, QLs do not require that the density of the responses follow a distribution from the exponential family class. In addition to relaxing the exponential family assumption, the mean and the variance structures are independent whereas in GLMs, apart from the normal family, choice of the mean structure, \( \mu = g(x'\beta) \) implies a certain variance structure, \( \nu(\mu) = \nu(g^{-1}(x'\beta)) \). No full distributional assumptions for \( y_i \) are necessary and only the mean and variance need be specified. This method has been useful for data that exhibit overdispersion or have variance of \( y_i \) greater than \( E(y_i) \). These moments are assumed to be correctly specified by

\[
E(y_i|x_i) = \mu_i = g^{-1}(x'_i\beta), \quad \text{var}(y_i) = \phi \nu(\mu_i)
\]  

(2.9)

where \( \nu(\cdot) \) is a known function and \( \phi \) is an overdispersion parameter whose value may be known or unknown and accounts for the variation of \( y_i \) not explained by \( \nu(\mu_i) \). The quasi-likelihood score equations are similar to the GLM score equations, however the variance component in (2.3) of the GLM score function is replaced by the variance expression in (2.9). The parameter estimates can be computed using the same methods described in section 2.1; however, these estimates are called maximum quasi-likelihood estimates. The quasi-likelihood model plays an important role in problems where the responses are correlated since, with the exception of the normal distribution, it is often difficult to construct joint likelihoods for repeated data. This model is the basis for generalized estimating equations methods.
2.3 Generalized Estimating Equations

Liang and Zeger (1986) extended the quasi-likelihood method and proposed the generalized estimating equations (GEE) approach to analyze longitudinal data. For correlated binary data, it is often difficult to construct a joint likelihood for the binary responses especially if the cluster size is large. The GEE is used frequently because it does not require the complete specification of the joint distribution or the likelihood of the repeated responses. This method requires that the mean structure and a working correlation matrix for the vector of responses are specified. Specifying a working correlation structure correctly can improve estimation efficiency but even if it is misspecified, the GEE is robust, that is, asymptotically unbiased (as was shown by Liang & Zeger, 1986). Hence it is semi-parametric. This approach is mainly used to model the relationship between covariates and the response, while treating the correlation as a nuisance parameter.

2.3.1 Estimation and Testing

In the GEE approach, one specifies the marginal distribution of the phenotype $y_{ij}$, for the $j$th individual in the $i$th family, $i = 1, \ldots, m$, $j = 1, \ldots, n_i$ from the exponential family class as in (2.7), $f(y_{ij}) = \exp[y_{ij}\theta_{ij} - b(\theta_{ij})/a(\phi) + c(y_{ij}, \phi)]$. Then the marginal expectation of $y_{ij}$ is $E(y_{ij}) = \mu_{ij}$ which is explained by the covariate vector $x_{ij}$ (of dimension $n_i \times p$ through the link function $g(\mu_{ij}) = x_{ij}^T\beta$ where the regression parameter $\beta$ is of dimension $p$. The marginal variance of $y_{ij}$ depends on the marginal mean, $\text{var}(y_{ij}) = \nu(\mu_{ij})\phi$ where $\nu$ is a known variance function. For example, $\nu(\mu_{ij}) = \pi_{ij}(1 - \pi_{ij})$ for binomial data where $\pi_{ij} = P(y_{ij} = 1)$. The parameter $\phi$ is the overdispersion parameter which is, for example, equal to 1 that follows the Binomial probability model. Then, a consistent estimate of the parameter $\beta$ is found by solving the estimating equations,

$$U^{GEE}(\beta, \alpha) = \sum_{i=1}^{m} D_iV_i^{-1}(y_i - \mu_i(\beta)) = 0.$$ (2.10)
The first term $D_i = (\partial \mu_i(\beta) / \partial \beta)^T = (\partial \mu_i / \partial \beta \cdots \partial \mu_{in_i} / \partial \beta)^T = A_i \Delta_i X_i$ is an $n_i \times p$ matrix where the $(j, k)$th element is $\partial \mu_{ij} / \partial \beta_k$ where $\mu_i(\beta) = E(y_i)$ the marginal expectation for $y_i$, $i = 1, 2, \cdots, m$. The second term is the inverse of $V_i = A_i^{1/2} R_i(\alpha) A_i^{1/2} / \phi$ of dimension $n_i \times n_i$. The diagonal elements of $A_i$ contain $\text{var}(y_{ij}) = \nu(\mu_{ij}) \phi = \pi_{ij}(1 - \pi_{ij})$ in the Binomial model and $R_i(\alpha)$ is the "working correlation matrix" or the "model-based" assumption of the correlation for the $n_i$ subunits or family members in cluster $i$ which is fully specified by the $s \times 1$ vector of parameters $\alpha$. The last term is an $n_i \times 1$ residual vector. The GEEs have a similar form as the score equations used to solve for GLM parameters. The difference is that now $y_i$ is an $n_i \times 1$ vector which is comprised of $n_i$ observations from the $i$th family, and the covariance matrix, $V_i$, for $y_i$ depends not only on $\beta$ but on some parameter vector $\alpha$, which is used to describe the within-cluster dependence. Some common choices for $R(\alpha)$ are the independence structure where the correlation matrix is the identity matrix, the exchangeable structure where correlations are assumed to be equal between observations within a cluster and some autoregressive structures where correlations have a serial structure in which observations further apart are less correlated. Note that when the working correlation matrix is the identity matrix and $\phi = 1$ then, $V_i = A_i$ which would make the GEE approach equivalent to the likelihood equations for the independence model for binomial data.

To solve for $\hat{\beta}^{GEE}$, Liang and Zeger (1986) suggest that one iterates between a modified Fisher scoring for $\beta$ and moment estimation for $\alpha$ and $\phi$. The estimation of $\alpha$ depends on the choice of $R_i(\alpha)$. Generally, it is estimated as a function of

$$R_{uv} = \frac{\sum_{i=1}^m \hat{\tau}_{iu} \hat{\tau}_{iv}}{\sum_{i=1}^m n_i - p}. \quad (2.11)$$

Liang and Zeger (1986) suggest using Pearson residuals,

$$\hat{\tau}_{ij} = \frac{y_{ij} - \hat{\mu}_{ij}}{[\nu(\mu)]_{ij}^{1/2}} \quad (2.12)$$

to estimate the correlation parameters in $\alpha$ and,

$$\hat{\phi} = \frac{\sum_{i=1}^m \sum_{j=1}^{n_i} \hat{\tau}_{ij}^2}{\sum_{i=1}^m n_i - p} \quad (2.13)$$
to estimate $\phi$. If $R_i(\alpha)$ was specified as an exchangeable correlation matrix, that is, $\text{corr}(y_{ij}, y_{ik}) = \alpha$ for all $j \neq j'$, then $\alpha$ can be estimated as

$$\hat{\alpha} = \frac{\sum_{i=1}^m \sum_{j>j'} \hat{r}_{ij}\hat{r}_{ij'}}{\hat{\phi}[\sum_{i=1}^m \frac{1}{2} n_i(n_i - 1) - p]}.$$  

(2.14)

If $R_i(\alpha)$ was unspecified so that the number of distinct $\alpha$ parameters were $s = \frac{1}{2} n_i(n_i - 1)$, then $R_i$ can be estimated as,

$$\frac{\sum_{i=1}^m A_i^{-1/2}(y_i - \mu_i(\beta))(y_i - \mu_i(\beta))' A_i^{-1/2}}{\hat{\phi} m}$$

(2.15)

which is useful only if $n_i$ is relatively small.

Estimates for the GEE are found iteratively. A good starting value for the parameter estimate, $\hat{\beta}^0$ is the maximum likelihood estimate from a generalized linear model. The GEE algorithm can be stated as follows for iteration $k$:

1. Given a current estimate of $\hat{\beta}$, the parameters $\hat{\alpha}$ and $\hat{\phi}$ can be estimated from current Pearson residuals as in (2.12) and the estimator for the dispersion parameter $\phi$ as in (2.13).

2. Then given current estimates $\hat{\alpha}^k$ and $\hat{\phi}^k$, the GEE (2.10) for $\hat{\beta}^{k+1}$ is solved by

$$\hat{\beta}^{(k+1)} = \hat{\beta}^{(k)} + F^{-1}(\hat{\beta}^{(k)})U_i^{GEE}(\hat{\beta}^{(k)})$$

where $F(\hat{\beta}^{(k)})$ is the \textit{working Fisher matrix},

$$F(\hat{\beta}^{(k)}) = \sum_{i=1}^m D_i^T(\hat{\beta}^{(k)})V_i^{-1}(\hat{\beta}^{(k)}, \hat{\alpha}^{(k)}, \hat{\phi}^{(k)})D_i(\hat{\beta}^{(k)})$$

Liang and Zeger (1986) showed that the GEE approach led to consistent estimates of $\beta$ even when the covariance structure was misspecified. They showed that when independence was assumed between observations in the same cluster the covariance matrix for $\hat{\beta}$ would approach,

$$V_i(\hat{\beta}) = \left(\sum_{i=1}^m X_i^T \Delta_i A_i \Delta_i X_i\right)^{-1} \left(\sum_{i=1}^m X_i^T \Delta_i \text{cov}(y_i) \Delta_i X_i\right) \left(\sum_{i=1}^m X_i^T \Delta_i A_i \Delta_i X_i\right)$$

(2.16)
as the number of \( m \) clusters increased. However, the better \( R_i(\alpha) \) was specified, the more efficient were the estimates \( \hat{\beta} \) and \( V(\hat{\beta}) \) (note if \( R_i(\alpha) \) was correctly specified then \( V_i = \text{cov}(y_i) \)). If we assume that \( V_i \) has been correctly specified, then the model-based estimate of \( \text{var}(\hat{\beta}^{GEE}) \) is

\[
V_N(\hat{\beta}^{GEE}) = \sum_{i=1}^{m} D_i^T \hat{V}_i^{-1} D_i
\]

(2.17)

where \( D_i = (\partial \mu_i / \partial \beta) \) evaluated at \( \hat{\beta}^{GEE} \) and \( \hat{V}_i \) is evaluated at \( \hat{\beta}^{GEE} \), \( \hat{\alpha}^{GEE} \) and \( \hat{\phi}^{GEE} \). Note that this is similar to the variance of the maximum likelihood estimator \( \hat{\beta}_{GLM} \) (which is the inverse of the information matrix, \( \left[ \sum_{i=1}^{m} D_i^T \hat{A}_i^{-1} D_i \right]^{-1} \)) except that \( \hat{A}_i \) has been replaced by \( \hat{V}_i \) in the GEE approach. If \( V_i \) is misspecified (or if \( R(\alpha) \) has been incorrectly assumed), then the variance of \( \hat{\beta} \) will also be incorrect. Liang and Zeger (1986) proposed a robust estimate of variance for \( \hat{\beta}^{GEE} \) or the sandwich estimator (Royall, 1986) \( V_R(\hat{\beta}^{GEE}) \) of \( \beta^{GEE} \) that would provide valid inferences for \( \beta^{GEE} \) even when the covariance structure in (2.17) is misspecified. An empirical estimate for \( V_i \),

\[
V_{0i} = (y_i - \mu_i(\beta))(y_i - \mu_i(\beta))^T
\]

is used in the sandwich estimator,

\[
V_R(\hat{\beta}^{GEE}) = V_N(\hat{\beta}^{GEE}) \left( \sum_{i=1}^{m} D_i^T \hat{V}_i^{-1} \tilde{V}_{0i} \hat{V}_i^{-1} D_i \right) V_N^{-1}(\hat{\beta}^{GEE})
\]

(2.18)

which is robust to an incorrectly specified working correlation. The unknown \( \mu_i(\beta) \)'s in \( V_{0i} \) are replaced by their estimates \( \mu_i(\hat{\beta}) \) for \( \hat{V}_{0i} \).

Confidence intervals for \( \hat{\beta} \) are constructed assuming an asymptotic standard normal distribution. These confidence intervals can be constructed using either the model-based variance estimate or the robust variance estimate of the regression coefficient. Similarly, the Wald test can be used to test a null hypothesis of no slope effect. These test statistics are computed as \( \hat{\beta} / \text{se} \hat{\beta} \) where either the model-based variance estimate or the robust variance estimate can be used.
2.3.2 Performance of the GEE Method

Several authors have assessed the performance of the GEE in simulation studies and compared it to other methods such as a maximum likelihood method under various conditions. These conditions have been restricted to sample size, cluster size and intracluster correlation.

Liang and Zeger (1986) examined the asymptotic efficiency of a GEE estimate for a regression parameter (\( \beta^{GEE} \)) assuming various working correlation structures and compared it to a maximum likelihood estimator when the correlation matrix was correctly specified. The data were generated from a multivariate Gaussian distribution with modest (\( \rho_Y = 0.3 \)) or high (\( \rho_Y = 0.7 \)) levels of correlation for one-dependence (tridiagonal correlation matrix), exchangeable and AR-1 correlation structures. They showed for a modest correlation that all of the GEE estimates were similar and highly asymptotically efficient regardless of the true underlying correlation structure. However, when the true correlation was large, the \( \beta^{GEE} \)'s were inefficient when the working correlation matrix was misspecified compared to when a correct working correlation matrix was specified.

Sharples and Breslow (1992) evaluated small sample properties of the GEE for binary outcomes and compared it to a maximum likelihood approach that was based on Bahadur's representation (Bahadur, 1961) of a multivariate binary distribution. In their simulation study, they considered moderate to low levels of correlation (they assumed an exchangeable structure of \( \rho_Y = 0.1 \) and \( \rho_Y = 0.3 \)) for a small sample size of 80 observations and large sample sizes of 100 and 200 observations under a specified logistic regression model. These were comprised of 20, 50 and 100 clusters which varied in size from 2 to 4 observations. In their model they used two covariates where one was cluster-specific and the other was individual-specific. They looked at bias, efficiency and estimates of variance of the regression parameter estimates to compare the ML approach and the GEE approach.

Under a correctly specified working correlation matrix, Sharples and Breslow
(1992) observed some bias in the GEE parameter estimates in small samples but the bias decreased when the sample size increased. They showed that the bias was higher in the cluster level parameter estimate than in the individual level parameter estimate. This was independent of the level of correlation whether \( \rho_Y = 0 \) or \( \rho_Y > 0 \) in the data. As expected, there was an increase in efficiency of the estimates when the sample size increased. They also showed that there was an improvement in efficiency and bias by increasing the number of clusters rather than increasing the cluster size for the same total number of observations. When they compared the small sample bias and efficiency to the maximum likelihood approach using Bahadur's representation, they observed both approaches produced similar regression parameter estimates and their corresponding variances. They looked at relative efficiency which is defined as the ratio of the mean square error of the GEE estimate using a robust variance to the mean square error of the maximum likelihood estimates. They observed that the GEE approach was close to being fully efficient for estimating regression parameters in small samples and sometimes more efficient.

When the working correlation matrix was incorrectly specified, they observed that the bias was larger in small samples (100 observations) compared to when the correlation matrix was correctly specified. However, as the sample size increased to 200 observations, there was less bias than in the smaller sample size of 100 observations.

They looked at the efficiency of the GEE approach under different assumed working correlations when the true underlying correlation was the exchangeable correlation. In smaller sample sizes, the parameter estimates were less efficient than in larger sample sizes. This difference was more noticeable when the underlying correlation was higher. In their examination of bias and efficiency of the GEE, they confirmed Liang and Zeger's (1986) result which showed that the independence assumption for the working correlation matrix was adequate in large samples.

Sharples and Breslow (1992) also looked at the performance of the model-based
and robust estimate of variance of the parameter estimates. When the working correlation matrix was correctly specified, neither estimate appeared to be close to the true variance. Contrary to what they expected, they observed in some situations that the robust estimate of variance was smaller than the model-based estimate of variance such as, in the estimate of the cluster-level covariate. In other situations, the robust estimate performed better than the model-based.

When the working correlation matrix was misspecified, they naturally expected that the robust estimate would perform better than the model-based estimate. This was not necessarily the case for the binary cluster level covariate. Rather, its robust estimate of variance was too low and conversely, too high for the subject-specific covariate. They note that Sharples (1989) has shown in theoretical results that for cluster level covariates, standard logistic regression consistently underestimates the variance. However, for subject level covariates, the variance may be over- or underestimated depending on the correlation structure and the distribution of the covariate which was also shown by Neuhaus and Segal (1993).

Lee et al (1992) also examined the behaviour of GEE estimates for bivariate binary data. They provided an asymptotic proof that showed that when the independence assumption was used for the working correlation matrix, the variance of a subject-specific coefficient always underestimated the usual asymptotic approximation to the true variance matrix of the regression coefficient. In their simulation study, they were able to validate their asymptotic result as well as observe that there was negligible bias in the estimation of their regression coefficients and that the efficiency of coefficient estimates decreased as the correlation within clusters increased.

Fitzmaurice (1995) demonstrated that in certain situations assuming an independence working correlation (with a robust variance correction) can lead to a substantial loss of efficiency in the parameter estimates. This was most evident when the intra-cluster correlation was high and when the data included individual-specific covariates. Fitzmaurice used the asymptotic relative efficiency (ARE) which is the ratio of a GEE
variance estimate to the asymptotic variance for the regression coefficient. He showed that the ARE depended on both the intracluster correlation for the covariate ($\rho_X$) and the correlation between the responses ($\rho_Y$). The maximum likelihood estimator was computed using Bahadur's likelihood representation as the true underlying model for an AR(1) process. When Fitzmaurice examined cluster-level covariates, he observed that the ARE was similar whether using an independence working assumption or one-dependence working assumption for the GEE. Under both working assumptions, there was some loss of efficiency in the estimates especially when $\rho_Y > 0.4$. He noted that the decline in efficiency is due to the failure of the GEE estimators “to exploit all the information about the mean or regression parameters in the second and third moment parameters”. However, when individual-specific covariates were examined, there were noticeable differences between the independence and one-dependence working assumptions. He showed that the loss of efficiency for the independence estimator declined more rapidly with increasing correlation compared to the one-dependence estimator. Fitzmaurice suggests that some attempt should be made to model the correlation between responses.

Trégouët et al (1997) used simulations to study the behaviour of the GEE in several practical situations of testing the association between candidate-gene markers and phenotype in related individuals. To date, this is the only known study of performance specifically designed for family data where genetic data are involved. They generated mixed samples of either fixed or variable sized nuclear families and included individuals that were not related. Rather than using an exchangeable correlation matrix for all pairs of responses, they estimated the correlations between binary responses using closed-form expressions. They note that there is better efficiency in estimating the regression coefficients and their corresponding variances when the working correlation matrix is better specified. As noted by Rotnitzky and Jewell (1990), misspecifying the correlation matrix has a large impact on the efficiency of the GEE estimate when the cluster size is not constant, as is often the case with
family data.

For fixed clusters, Trégouët et al (1997) also evaluated alternative models of various levels of association parameters. They observed for a binary phenotype, that the power of the GEE test statistic decreased as the cluster sizes and intracluster correlation increased. In large samples such as \( m \geq 50 \), the bias was negligible and the coverage probability was close to the desired nominal level. However in small samples such as \( m < 50 \), the bias was larger and the coverage probability was lower than the desired probability level of 0.95. Power was low when samples of variable cluster sizes were used and when intracluster correlation increased.

When they examined a null model of no association, they observed that the type I errors for the GEE estimates were inflated in small samples and in large cluster sizes. There was some efficiency loss in the GEE estimates when samples consisting of different cluster sizes were used compared to GEE estimates computed from samples of fixed cluster sizes. The type I error was inflated when the clusters were large; if the sample consisted of various sized clusters (heterogeneous) and; if the sample size was small especially if it was heterogeneous.

**2.3.3 Summary and Discussion**

In general, most observed that the performance of the GEE was weaker when the sample consisted of a small number of clusters, when the cluster size was large or unbalanced and when the intracluster correlation was high.

In all of the simulation studies presented, the various authors observed that under a *modest* degree of residual correlation (such as \( \rho = 0.3 \) in the exchangeable structure), the parameter estimates were efficient. Liang and Zeger (1986) showed that using any working correlation structure provided asymptotically efficient parameter estimates (when the intracluster correlation was modest). However, in small sample sizes, Sharples and Breslow (1992) observed that misspecification of the working
correlation matrix did not produce efficient GEE estimates whereas correctly specifying the working correlation matrix did. Fitzmaurice (1995) clarified some of the paradoxical findings in the GEE literature where some authors were prescribing the independence working assumption with robust variance correction in practical situations whereas others were warning against the use of the independence assumption because it would lead to a substantial loss of efficiency in the parameter estimates. Fitzmaurice showed that when the covariate design included individual-specific covariates, assuming independence led to a substantial loss of efficiency for the parameter estimate associated with that covariate. As the correlation increased, all authors observed that the efficiency of the parameter estimates decreased when the working correlation structure was misspecified. Liang and Zeger (1986), Sharples and Breslow (1992) and Trégouët et al. (1997) examined clusters of different sizes. Both Liang and Zeger (1986) and Trégouët et al (1997) observed some efficiency loss when the cluster sizes varied whereas Sharples and Breslow (1992) saw no evidence of this in their study.

Some finite sample bias was observed in all simulation studies with the exception of Lee et al. (1992). In the simulation study conducted by Lee et al. (1992), only bivariate binary data were considered. They observed negligible bias in the parameter estimates even in a small sample of 20 clusters. This may have been due to using clusters of size 2 instead of the larger cluster sizes which were used by other investigators in their simulation studies. Under a correctly specified working correlation structure, Sharples and Breslow (1992) observed some bias in small sample sizes. The bias of the GEE parameter estimates increased when the working correlation matrix was misspecified. The bias was larger for the cluster-level covariate than the subcluster-level covariate. They observed that the bias decreased as the number of clusters increased and to a lesser extent, when the cluster sizes increased. Trégouët et al. (1997) observed some bias in small sample sizes.

It is apparent that assuming an independence working correlation structure is not
always the best approach. In general, misspecifying the correlation structure can lead to some efficiency loss especially when there is strong correlation and individual-specific covariates. Also, Sullivan Pepe and Anderson (1995) noted that misspecification of the correlation could also lead to seriously biased estimates if a condition that they specified was violated. This condition involved the equivalence of the marginal mean to the partly conditional mean of the model for the data at hand. This is one of the drawbacks when using the GEE. It is difficult to specify a correlation structure especially when the underlying correlation is unknown. In the section 2.5, the non-parametric bootstrap is proposed as a more robust method to handle the underlying correlation in a regression setting where the correlation between responses need not be known.

2.4 Misspecification Effects for Models Fitted to Dependent Data

Conventional analyses of family data or other forms of dependent data will yield variance estimates for regression coefficients that can be severely overestimated or underestimated if not properly corrected (Donner, 1984). Scott and Holt (1984) describe a misspecification effect as the inflation factor needed to correct standard results obtained from ordinary linear regression for the effect of intracluster correlation. This measure of the effect of clustering on an estimator is calculated as the ratio of the variance of the estimator that adjusts for dependent data to the variance of an estimator that does not adjust for dependent data, that is, assuming the data were independent.

To illustrate the misspecification effect, suppose there are $m$ families indexed by $i = 1, \ldots, m$ with $j = 1, \ldots, n_i$ members within each family. Assume that for each family member a continuous outcome variable $y_{ij}$ is observed that is explained by $p$
predictor variables $x_{ij} = (x_{1ij}, \cdots, x_{p_{ij}})$ and represented by a linear regression model,

$$E(y) = x_{ij}'\beta$$

where $y$ is a data vector of all realizations. Since observations within each family are not independent, Scott and Holt (1984) assumed that the covariance matrix of $y$ took on the form

$$V_C(y) = \sigma^2 C$$

where $C$ is a block diagonal matrix such that each $n_i \times n_i$ block corresponds to the $i$th family (cluster) and where the responses are correlated within families but uncorrelated between families. For regression coefficients estimated under ordinary least squares, these authors showed that the true covariance matrix of $\hat{\beta}$ (accounting for correlation) was

$$V_C(\hat{\beta}) = \sigma^2 (X^T X)^{-1} (X^T V X) (X^T X)^{-1}$$

$$= \sigma^2 (X^T X)^{-1} D \quad \text{(2.19)}$$

where $X = [x_1^T, \cdots, x_p^T]$ is the design matrix and $V = \text{cov}(y)$. The misspecification effect of the regression coefficients are the diagonal elements of $D = (X^T V X) (X^T X)^{-1}$. Scott & Holt (1984) simplified the matrix of $D$ by assuming a common correlation (exchangeable correlation) between pairs of responses,

$$D = I + (M - I) \rho \quad \text{(2.20)}$$

where $I$ is the identity matrix and the matrix $M$ depends only on the design matrix $X$. Scott & Holt (1984) considered the case of a simple linear regression to show the effects of the response correlation and the covariate correlation on the variance estimate of a slope parameter, $\beta$. The inflation factor associated with the slope estimate is the lower right-hand entry in $D$,

$$D = I + (M - I) \rho$$

$$= \begin{pmatrix}
1 + (n - 1) \rho_Y & 0 \\
0 & 1 + (n - 1) \hat{\rho}_X \rho_Y
\end{pmatrix} \quad \text{(2.21)}$$
if equal cluster sizes are assumed. The quantity $\hat{\rho}_X$ which is the common intracluster correlation between the covariate values within each cluster is estimated as,

$$\hat{\rho}_X = \left[ \frac{m(xx)_B}{(xx)_W} - 1 \right] / (m - 1)$$

(2.23)

where $(xx)_B = \sum (\bar{x}_i - \bar{x})^2$ and $(xx)_W = \sum \sum (x_{ij} - \bar{x})^2 - (xx)_B$ are the between and within sum of squares for the covariate $x$ and $\rho_Y$ is the residual correlation (Donner, 1984). The residual correlation is assumed to be non-negative in the inflation factor.

If the covariate value $X$ is chosen randomly, the variation between subjects will tend to be greater than the variation within. This would tend to make $\hat{\rho}_X$ positive making the inflation factor $> 1$. This would imply that the standard error would underestimate the true standard error leading to a smaller p-value. Conversely, a negative value of $\hat{\rho}_X$ would suggest that the variation between subjects was smaller than the variation within subjects. This would lead to an inflation factor $< 1$. The standard errors will be overestimated and the p-value large.

Neuhaus and Segal (1992) extended this method to generalized linear models for binary responses. They found that the analytical results for the misspecification effects described by Scott & Holt for the linear regression model were also good approximations for binary regression models. When independence among responses is assumed within clusters, the asymptotic covariance matrix of the estimated generalized linear model coefficients $\hat{\beta}$ is given by,

$$V_I(\hat{\beta}) = (X^T \Delta A \Delta X)^{-1}(X^T \Delta \text{var}(Y_{ij}) \Delta X)(X^T \Delta A \Delta X)^{-1}$$

(2.24)

where $\Delta = \text{diag}(\partial \theta_{ij}/\partial \eta_{ij})$ and $A = \text{diag}[\phi \text{var}(Y_{ij})]$. Neuhaus & Segal reduced (2.24) into a form that was similar to (2.20) by letting $\tilde{X} = A^{1/2} \Delta X$ and $\tilde{V} = A^{-1/2} \text{var}(y) A^{-1/2}$ so that

$$V(\hat{\beta}) = (\tilde{X}^T \tilde{X})^{-1}(\tilde{X}^T \tilde{V} \tilde{X})(\tilde{X}^T \tilde{X})^{-1} = V_I(\hat{\beta}) \tilde{D}.$$
2.5 The Bootstrap

The bootstrap is a simple method to apply and can be used as an alternative method to the GEE method in the analysis of correlated data. An advantage of the bootstrap is that the variance of an estimate can be computed without having to make any assumptions about the underlying correlation structure. Originally, the bootstrap was developed for problems that assumed independent, identically distributed data (i.i.d.). However, for dependent data, the ordinary bootstrap falls short in capturing the dependence structure of the data because the correlation structure is not preserved when individual data points are bootstrapped. As a result, some bootstrap methods that deal with dependent data have been proposed ranging from simple to complex modifications. Some background on the bootstrap will be presented followed by a discussion of some bootstrap methods that have been suggested for dependent data. Finally, a hierarchical nonparametric bootstrap method is proposed that involves two strategies to handle dependent data from nested structures, since the family structure can be viewed as such. The section closes with discussion of a one-step GLM approach for bootstrapping binary data.

2.5.1 Background

Efron first proposed the idea of the bootstrap in 1979. This method was used for assessing the variability of estimated parameters from a set of i.i.d. observations, by examining the variability of the estimate across a large number \( B \) of bootstrap samples. A bootstrap sample is obtained by taking a random sample of size \( n \) from the original data \( x_1, x_2, \ldots, x_n \) using random sampling with replacement. The variability between results from bootstrap samples can be investigated. There are two ways to compute a bootstrap sample: 1) analytically (parametric) and 2) using Monte Carlo methods (nonparametric). The latter can be used to compute bootstrap samples in all cases since minimal model assumptions are required whereas the former is more
limited by distributional assumptions. Focus will be on the nonparametric bootstrap approach due to its flexibility and simplicity and because it assumes very little about the form of a probability distribution $F$, apart from its existence. One of the biggest attractions for using the nonparametric bootstrap is that it can substitute computer power for complicated theoretical analysis.

To understand how the bootstrap works, suppose a random sample of size $n$ is observed from a probability distribution $F$,

$$F \rightarrow (x_1, x_2, \ldots, x_n).$$

The empirical distribution function $\hat{F}$ is defined as a discrete distribution that places probability $1/n$ on each value $x_i$, $i = 1, 2, \ldots, n$,

$$\hat{F}(x) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}\{x_i \leq x\},$$

where $\mathbb{I}\{A\}$ is an indicator function of the set $A$. The bootstrap sample $x^* = (x_1^*, x_2^*, \ldots, x_n^*)$, is defined to be a random sample of size $n$ drawn with replacement from $\hat{F}$,

$$\hat{F} \rightarrow (x_1^*, x_2^*, \ldots, x_n^*).$$

The bootstrap constructs an empirical distribution function, $\hat{F}$, which can be seen as a simple estimate of the entire distribution $F$. Therefore, to estimate some parameter of $F$ it would be natural to examine its corresponding statistic in $\hat{F}$, such as $\hat{\theta} = s(x)$.

For every bootstrap sample, $x^*$, the function $s(\cdot)$ is applied to the bootstrap sample giving the bootstrap replicate, $\hat{\theta}^* = s_{\hat{F}}(x^*)$. A bootstrap sample of size $n$ is replicated a large number of $B$ times using a random number generator, and analyzed to produce $B$ bootstrap estimates of $\hat{\theta}(b)$, $b = 1, \ldots, B$ of the statistic of interest, $\hat{\theta} = s_F(x)$. By using this collection of $B$ bootstrap estimates $\hat{\theta}^*(b)$, $b = 1, \ldots, B$, inferences can be made on the original parameter, $\hat{\theta}$, concerning its standard error and bias and can construct confidence intervals for the population statistic of interest $\theta$.  

It is clear that for the bootstrap to work, it must be assumed that the sample which is represented by the empirical distribution function is a good estimator of the population distribution function that generated the sample in the first place. That is, there must be faith that all of the relevant characteristics of the population are represented in the sample. The consistency of the bootstrap rests on the fact that if the original sample size is sufficiently large then as the number of bootstrap samples approaches infinity, the bootstrap estimate \( \hat{\theta}^B = f(\hat{\theta}^*(b)) = \hat{s}_B(x^*) \) for \( b = 1, \cdots, B \) bootstrap replications, approaches the *ideal* bootstrap estimate, \( \hat{\theta}^* = \hat{s}_F(x^*) \) which is the estimate of the statistic, \( \hat{\theta} = s_F(x) \).

Efron (1982) has stated that the number of \( B \) bootstrap samples required for satisfactory results depends on the statistics to be estimated and the accuracy desired. For statistics such as the bias and standard error, anything over \( B > 200 \) provides only a slight improvement in the estimator. For bootstrap confidence intervals, a minimum of \( B = 1000 \) is required for accurate estimation of the tail probabilities.

### 2.5.2 Bootstrap Estimates of Standard Error and Bias

Sometimes for complicated estimators, it is difficult to assess accuracy (such as the variance and bias) because either closed form analytical expressions are not available or it is simply too difficult to compute. When this situation arises, the bootstrap is used frequently since it provides a way to compute numerically an approximation to a theoretical statistic.

The bootstrap estimate of \( \theta \) is given by,
\[
\hat{\theta}^B = \frac{\sum_{i=1}^{B} \hat{\theta}_i^*}{B} = \hat{\theta}^*(\cdot)
\] (2.25)

The bootstrap estimate of standard error for \( \hat{\theta} \) is the sample standard deviation of the \( \hat{\theta}^*(b) \) values,
\[
\hat{se}_B = \sqrt{\frac{\sum_{b=1}^{B} \left[ \left( \hat{\theta}^*(b) - \hat{\theta}^*(\cdot) \right)^2 \right]}{B - 1}}^{1/2}
\] (2.26)
Another statistical measure of the accuracy of an estimator $\hat{\theta}$ is the bias,

$$bias_F = bias_F(\hat{\theta}, \theta) = E_F[\hat{\theta}] - \theta$$

(2.27)

which is defined as the difference between the expectation of $\hat{\theta}$ and the true parameter $\theta$. To obtain the bootstrap estimate of bias, $\hat{F}$ would replace $F$ in (2.27) that is, $\hat{\theta}^*$ would substitute for $\hat{\theta}$ and $\theta$ would substitute for $\theta$,

$$bias_{\hat{F}} = E_{\hat{F}}(\hat{\theta}^*) - \hat{\theta},$$

(2.28)

which is the theoretical bootstrap result based on an infinite number of bootstrap replications. The expression in (2.28) can be approximated by using $\hat{\theta}^*(\cdot)$ (defined in (2.25)), based on $B$ bootstrap replications, in place of $E_{\hat{F}}(\hat{\theta}^*)$. Therefore, the bootstrap estimate of bias based on $B$ bootstrap replications is

$$\text{bias}_B = \hat{\theta}^*(\cdot) - \hat{\theta}.$$ 

(2.29)

### 2.5.3 Bootstrap Confidence Intervals

Since Efron (1979) first introduced the bootstrap, a vast amount of research has focussed on bootstrap confidence intervals. Two widely used reliable methods are the percentile and bias-corrected and accelerated confidence intervals.

**Percentile Method**

A simple method to construct bootstrap confidence intervals is the percentile method (Efron and Tibshirani, 1986). To construct a percentile interval, bootstrap replications $\hat{\theta}^*(b)$ are computed to generate the bootstrap distribution for the statistic of interest, $\hat{\theta}$. The $1 - 2\alpha$ percentile interval is given by the two values that encompass the central $(1 - 2\alpha)\%$ of the distribution,

$$[\hat{\theta}_{\%lo}, \hat{\theta}_{\%up}] = [\hat{\theta}^*(\alpha), \hat{\theta}^*(1-\alpha)]$$

(2.30)
where the lower endpoint of this interval \((lo)\) is the \(100\alpha^{th}\) percentile of the bootstrap distribution and the upper endpoint \((up)\) is the \(100(1-\alpha)^{th}\) percentile of the bootstrap distribution. However, this interval is for the ideal bootstrap situation, where an infinite number of bootstrap samples are taken. To construct an approximate \(1 - 2\alpha\) percentile interval from a fixed set of \(B\) bootstrap samples, \(B\) bootstrap estimates of the parameter of interest are computed and then ordered. The \(B \cdot \alpha\) value in the ordered list of the \(B\) replications of \(\tilde{\theta}^*\) is the empirical \(100 \cdot \alpha\) percentile of the \(\tilde{\theta}^*(b)\) values denoted by \(\tilde{\theta}^*_{B(\alpha)}\) and similarly, the \(B \cdot (1 - \alpha)\) value is the \(100 \cdot (1 - \alpha)\) empirical percentile of the \(\tilde{\theta}^*(b)\) values denoted by \(\tilde{\theta}^*_{B(1-\alpha)}\). For example, a 95% percentile interval is given by the value that exceeds 2.5% of the generated distribution and the value that exceeds 97.5% of the generated distribution. Hence, the approximate \(1 - 2\alpha\) percentile interval is

\[
[\tilde{\theta}_{B,lo}; \tilde{\theta}_{B,up}] \approx [\tilde{\theta}^*_{B(\alpha)}; \tilde{\theta}^*_{B(1-\alpha)}].
\]  

(2.31)

One of the attractive features of the percentile method is that it has a \textit{transformation-respecting} property. In other words, for parameters with an unknown underlying distribution, the percentile method will provide a good approximation of the \(1 - 2\alpha\) confidence interval, especially for nonlinear parameters estimated by small sample sizes (Efron and Tibshirani, 1993). It is called \textit{transformation respecting} because if a transformation was applied to a nonlinear parameter so that its bootstrap distribution is made more Gaussian, then a confidence interval based on the normal approximation would give a correct interval for this transformed variable. When the endpoints of the standard confidence interval are mapped back onto the scale of the original parameter, the endpoints will agree with the percentile interval endpoints. Also, the percentile intervals will provide more reliable intervals than the standard normal approximation method since the percentile intervals are based on bootstrap statistics that follow any restrictions that the statistic of interest, \(\tilde{\theta}\) may have, such as the correlation coefficient which is restricted to lie between -1 and 1. In other words, the percentile interval is \textit{range preserving} of parameters with restriction to possible values that lie in the
Bias-Corrected and Accelerated Percentile Method

The *bias-corrected and accelerated* (BC$_a$) percentile method (Efron and Tibshirani, 1986) is an improved version of the percentile method which works better for distributions of $\hat{\theta}$ that may be biased. The BC$_a$ interval endpoints are also computed by using percentiles of the bootstrap distribution. Two quantities, the *acceleration* ($\hat{a}$) which corrects for skewness and the *bias-correction* ($\hat{z}_0$) are used in the construction of the BC$_a$ interval. The bias-correction is defined as

$$\hat{z}_0 = \Phi^{-1}\left(\frac{\#\{\hat{\theta}^*(b) < \hat{\theta}\}}{B}\right)$$

which is the inverse function of a standard normal CDF of the proportion of bootstrap replications less than $\hat{\theta}$. For example, $\Phi^{-1}(0.95) = 1.645$. It corrects for any discrepancy that might exist between the median of $\hat{\theta}^*$ and $\hat{\theta}$ in normal units. The acceleration constant ($\hat{a}$) is computed by using the *jackknife* statistic, $\hat{a}(i) = s(x_{(i)})$ where $(i)$ denotes computation with the $i$th point $x_i$ removed (Efron and Tibshirani, 1993). Thus,

$$\hat{a} = \frac{\sum_{i=1}^{n} (\hat{\theta} - \hat{\theta}(i))^3}{6\{\sum_{i=1}^{n} (\hat{\theta} - \hat{\theta}(i))^2\}^{3/2}}$$

where $\hat{\theta}(i) = \sum_{i=1}^{n} \hat{\theta}(i)/n$. The BC$_a$ interval endpoints are those values that cover $(1 - 2\alpha)$% of the distribution,

$$[\hat{\theta}_{\%lo}, \hat{\theta}_{\%up}] = [\hat{\theta}^*_B(\alpha_1), \hat{\theta}^*_B(\alpha_2)],$$

where,

$$\alpha_1 = \Phi\left(\frac{\hat{z}_0 + z(\alpha)}{1 - \hat{a}(\hat{z}_0 + z(\alpha))}\right)$$

and

$$\alpha_2 = \Phi\left(\frac{\hat{z}_0 + z(1-\alpha)}{1 - \hat{a}(\hat{z}_0 + z(1-\alpha))}\right),$$
where \( z^{(\alpha)} \) is the 100-\( \alpha \) percentile point of a standard normal distribution, for example, \( z^{(0.05)} = -1.645 \). Just like the percentile interval, the BC\( _a \) interval is transformation-respecting, however, it is second-order accurate compared to the percentile interval which is only first-order accurate. In other words, the BC\( _a \) interval should have better coverage accuracy than the percentile interval. The BC\( _a \) intervals are an improvement over the percentile intervals both theoretically and practically (Efron and Tibshirani, 1993).

A problem with the BC\( _a \) method arises when \( \alpha_1 \) is close to 0 and \( \alpha_2 \) is close to 1. This can occur when the bootstrap distribution of a parameter is biased. When the bootstrap distribution is biased to the right of the empirical parameter estimate \( \hat{\theta} \), the bias-correction in (2.32) will push \( \alpha_1 \) towards 0. Similarly, if the bootstrap distribution is biased to the left of the empirical parameter estimate \( \hat{\theta} \), the bias-correction in (2.32) will push \( \alpha_2 \) towards 1. When this occurs, the quantile of the corresponding bootstrap estimate cannot be calculated since \( B \cdot \alpha_1 \) or \( B \cdot \alpha_2 \) could be less than 1 or greater than \( B \). In this situation, it is appropriate to use the extreme value of the ordered \( \hat{\theta}^*(b) \) values, that is one could use the first ordered value or the last ordered value of the \( \hat{\theta}^*(b) \)'s (Davison and Hinkley, 1997).

### 2.5.4 Some Bootstrap Methods Applied to Dependent Data

If an ordinary bootstrap is applied to dependent data by taking individual observations as the bootstrap unit, naturally, the variance estimators for a particular parameter would be biased. Singh (1981) showed the inconsistency of the bootstrap with dependent data where he looked at the sample mean from a simple \( m \)-dependent structure composed of contiguous groups of \( m \) random variables, \( y_i, i = 1, \ldots, n \) that were correlated within but not between. He noted that according to the Central Limit Theorem, this \( m \)-dependent process followed,

\[
n^{1/2}(\hat{\theta} - \theta) \rightarrow N(0, \sigma^2 \sum_{i=1}^{m-1} \text{cov}(y_1, y_{1+i})).
\]  

(2.34)
However, according to bootstrap theory,

\[ n^{1/2}(\hat{\theta}^* - \bar{\theta}) \rightarrow N(0, \sigma^2) \]  

(2.35)

where \( \hat{\theta}^* \) is the bootstrap estimate of the sample mean. Therefore for an \( m \)-dependent process the ordinary bootstrap will not converge to (2.35). Hence, even under the simplest dependent data structure, the bootstrap would not be expected to provide a consistent approximation. The general idea behind bootstrapping dependent data is to choose a bootstrap unit so that each bootstrap unit is independent from each other.

For general dependent data, several authors have proposed various different bootstrapping schemes. One of the first suggestions was given by Carlstein (1986) and Shi (1986), based on the notion of grouping. Briefly, the grouped bootstrap groups dependent data into \( k \) groups with \( h \) observations in each, such that the groups are (nearly) independent. The bootstrap is then applied to each group by treating the groups as primary sampling units. The moving block bootstrap proposed by Künsch (1989), Liu and Singh (1992) and Politis and Romano (1992) is similar to the grouped bootstrap except that the blocks are overlapped. These methods are useful for time-series data where blocks of data are not as explicitly defined. When blocks of data are explicitly defined, that is when the data present themselves in a clustered form, a simple and intuitive approach to bootstrapping correlated data is to resample the distinct clusters of dependent data in order to maintain the correlation structure. This is known as block resampling (Davison and Hinkley, 1997) or the all block bootstrap (Sherman and Le Cessie, 1997).

In the next section, an approach to analysing dependent data is presented which does not explicitly model the within-cluster correlation. This nonparametric bootstrap method exploits the nuclear family structure to obtain the correct standard errors.
2.5.5 Hierarchical Bootstrap

If it is assumed that a collection of families are completely independent from each other, then an intuitive way to apply a nonparametric bootstrap on this data would be to bootstrap distinct families. Hence, the structure and correlation within each family would be preserved. This all block bootstrap (Sherman and Le Cessie (1997), Davison and Hinkley (1997)) is the easiest way to apply the standard bootstrap to dependent data. If $y_1, \ldots, y_m$ denote family vectors for families $1$ to $m$ consisting of $n_i$ family members for each, then the all block bootstrap takes a bootstrap sample with equal probabilities $m^{-1}$ from $y_i$'s, $i = 1, \ldots, m$.

The all block bootstrap can be extended to a resampling scheme that takes into account two sources of variation observed in family data. These two sources of variation are the variation between families and the variation within families. A hierarchical bootstrap was described by Davison and Hinkley (1997) as a method for computing bootstrap estimates in which the resampling scheme follows a two-stage nested design. The composition of the nuclear family can be viewed as one with a nested or hierarchical structure (see figure 2.1).

Davison and Hinkley (1997) presented theoretical bootstrap results assuming equal-sized clusters. Their results are generalized to the unbalanced case since it is of interest to bootstrap families of unequal sizes. Assume that there are $m$ families, $i = 1, 2, \ldots, m$ each consisting of $n_i$ family members (where each is indexed by $j = 1, 2, \ldots, n_i$). Also assume that each response can be written as a general linear random effects model of a two-stage nested design,

$$y_{ij} = x_i + z_{ij}, \quad i = 1, \ldots, m \text{ families,}$$

$$j = 1, \ldots, n_i \text{ individuals in family } i.$$  \hspace{1cm} (2.36)

The $x_i$'s describe a feature that is unique to the $i$th cluster. They come from some distribution $F_x$ with mean $E(x_i) = 0$ and $Var(x_i) = \sigma_x^2$. The differences between the $x_i$'s are due to the between-family variability, $Var(x_i) = \sigma_x^2$. The $z_{ij}$'s are unique to
the \(j\)th family member in the \(i\)th family and belong to some other distribution \(F_z\) with mean \(E(z_{ij}) = 0\) and \(Var(z_{ij}) = \sigma^2_z\). If the \(x_i\)'s and the \(z_{ij}\)'s are independent, then for (2.36),

\[
\text{var}(y_{ij}) = \sigma^2_x + \sigma^2_z \tag{2.37}
\]

and

\[
\text{cov}(y_{ij}, y_{ik}) = E(y_{ij}y_{ik}) - E(y_{ij})E(y_{ik})
\]
\[
= E[(x_i + z_{ij})(x_i + z_{ik})] - E(x_i + z_{ij})E(x_i + z_{ik})
\]
\[
= E(x_i^2 + x_i z_{ik} + x_i z_{ij} + z_{ij} z_{ik}) - E^2(x_i) - E(x_i)E(z_{ik}) - E(z_{ij})E(x_i) - E(z_{ij})E(z_{ik})
\]
\[
\text{(since } x \text{ and } z \text{ are independent)}
\]
\[
= E(x_i^2) - E^2(x_i)
\]
\[
= \sigma^2_x, \ j \neq k. \quad (2.38)
\]

Then \(\text{corr}(y_{ij}, y_{ik}) = \sigma^2_x/(\sigma^2_x + \sigma^2_z)\). This correlation within a family complicates resampling. Davison & Hinkley (1997) presented two strategies for bootstrapping data with a nested structure where sampling is performed at two levels. These strategies are adapted for family data:

**Strategy 1:**
*Stage 1:* randomly sample families with replacement, keep parents.
*Stage 2:* for those families selected in stage 1, randomly sample offspring **within** each family **without** replacement.

**Strategy 2:**
*Stage 1:* randomly sample families with replacement, keep parents.
*Stage 2:* for those families selected in stage 1, randomly sample offspring **within** each family **with** replacement.

Note that the first strategy is the same as the **all block** bootstrap described at the beginning of this section since the sampling at the second stage is performed without
replacement. Also note in stage 2 that only the offspring are randomly sampled since it would not make sense genetically to include the parents in the resampling scheme.

For these strategies, resampling at the second stage can be viewed as resampling from a fixed set of zygotes produced by the parental union (strategy 1) or as resampling from an infinite pool of zygotes available from the parental union (strategy 2). The bootstrap can be applied assuming that the nuclear family data were sampled as a two-stage design.

For balanced data structures, Davison & Hinkley argued that the theoretical bootstrap under strategy 1 was better than strategy 2 because it closely resembled the variability of the nested design of the original data. This is also true in the unbalanced case when the second moments of the resampled (bootstrapped) data were examined. Some results are presented to show the differences between the two strategies. Details of the intermediate steps can be found in appendix A. Assume that the empirical distribution function $\hat{F}$ contains $m$ families each with $n_i$ family members. Then, the bootstrap sample drawn from $\hat{F},$

$$\hat{F} \rightarrow (y_1^*, y_2^*, \ldots, y_m^*)$$

consists of those families resampled. In stage 1, a bootstrap sample of $m^*$ families

Figure 2.1: *Structure of a Nuclear Pedigree Seen As A Nested Design*
is obtained which is essentially a bootstrap sample of the parents along with their children. Given a particular bootstrap sample, \( b \), and a particular family, \( i^* \), it is seen that under both strategies

\[
E_{\hat{F}}(y_{ij}^*|y_i^* b) = \bar{y}_i^* \quad \text{and} \quad E_{\hat{F}}(y_{ij}^{*2}|y_i^* b) = \sum_{j=1}^{n_i} \frac{1}{n_i} y_{ij}^{*2}
\]

regardless of whether sampling was done with or without replacement in stage 2.

However, the differences between the sampling types lie in the expression,

\[
E_{\hat{F}}(y_{ij}^* y_{ik}^*|y_i^* b) = \begin{cases} \sum_{1 \leq k \neq l \leq n_i} \frac{1}{n_i(n_i-1)} y_{ik}^* y_{il}^* & \text{Strategy 1} \\ \sum_{k,l=1}^{n_i} \frac{1}{n_i^2} y_{ik}^* y_{il}^* & \text{Strategy 2} \end{cases}
\]

since \( \text{Prob}(y_{ij}, y_{ik}) = \frac{1}{n_i(n_i-1)} \) if random sampling without replacement and \( \text{Prob}(y_{ij}, y_{ik}) = \frac{1}{n_i^2} \) if random sampling with replacement. Hence, over all infinite bootstrap samples,

\[
E_{\hat{F}}(y_{ij}^*) = \bar{y}_i, \quad \text{Var}_{\hat{F}} = \frac{SS_B}{m} + \frac{1}{m} \sum_{i=1}^{m} \frac{SS_W[i]}{n_i}
\]

and

\[
\text{Cov}_{\hat{F}}(y_{ij}^* y_{ik}^*) = \begin{cases} \frac{SS_B}{m} - \frac{1}{m} \sum_{i=1}^{m} \frac{SS_W[i]}{n_i(n_i-1)} & \text{Strategy 1} \\ \frac{SS_B}{m} & \text{Strategy 2} \end{cases}
\]

where \( \bar{y}_i^* = \sum_{i=1}^{m} \bar{y}_i^* \), \( SS_B = \sum_{i=1}^{m} (\bar{y}_i^* - \bar{y}_i)^2 \) and \( SS_W[i] = \sum_{j=1}^{n_i} (y_{ij}^* - \bar{y}_i)^2 \).

Following Davison & Hinkley, the next step is to compare the expectations of (2.40) to theoretical counterparts by using the results,

\[
E(SS_B) = m \sigma_x^2 \left[ 1 - \frac{\sum_i n_i^2}{(\sum_i n_i)^2} \right] + \sigma_x^2 \left[ \sum_i \frac{1}{n_i} - \frac{a}{\sum_i n_i} \right]
\]

and

\[
E(SS_W) = \sum_{i=1}^{m} (n_i - 1) \sigma_x^2
\]

which produced,

\[
\text{Ecov}_{\hat{F}}(y_{ij}^*, y_{ik}^*) = \sigma_x^2 \left[ 1 - \frac{\sum_i n_i^2}{(\sum_i n_i)^2} \right] - \frac{1}{\sum_i n_i} \sigma_x^2
\]

Strategy 1

\[
= \sigma_x^2 \left[ 1 - \frac{\sum_i n_i^2}{(\sum_i n_i)^2} \right] + \frac{\sigma_x^2}{m} \left[ \sum_i \frac{1}{n_i} - \frac{m}{\sum_i n_i} \right]
\]

Strategy 2.
Strategy 1 appears closer to the true variability than strategy 2, which is larger. However, strategy 2 may be useful in situations where there are small number of clusters and large number of subunits in each cluster, since it accounts for more within-cluster variation than strategy 1. Generally, strategy 1 would be expected to pick up the true variability whereas strategy 2 would be expected to overestimate the true variability of the outcomes. However, in a sample with a small number of clusters, strategy 1 might underestimate the variability whereas strategy 2 might provide better estimates closer to the true variability.

Using the bootstrapped data from strategy 1 and strategy 2, it is easily seen that for complex statistics such as a regression coefficient ($\beta$), the variance of $\hat{\beta}$ will also differ according to the bootstrap strategy used. This difference can be approximated by substituting the covariances in (2.43) and (2.44) into the asymptotic covariance-variance matrix for $\hat{\beta}$ from a generalized linear model,

$$V(\hat{\beta}) = (X^T \Delta A \Delta X)^{-1} (X^T \Delta \text{cov}(y_i, y_j) \Delta X)(X^T \Delta A \Delta X)^{-1}$$

where $\Delta = \text{diag}(\partial \theta_{ij}/\partial \eta_{ij})$ and $A = \text{diag}[\text{var}(y_{ij})]$. Clearly the variance estimates for $\hat{\beta}$ using the two strategies would show that the $\text{var}(\hat{\beta}_{S1})$ is less than the $\text{var}(\hat{\beta}_{S2})$.

2.5.6 Bootstrapping Logistic Regression Models

When a logistic regression model is applied to data in a bootstrap sample, obtaining degenerate data replications can occur occasionally. A degenerate data set may contain data where very few events are present or where there are no events or all events. Hence, the fitting of a logistic model can break down if the data are truly degenerate. This phenomenon of degenerate data sets that produces nonexistent MLEs in logistic regression models was discussed by Albert and Anderson (1984). They described how data could be completely separated or quasicompletely separated. The estimates were often nonunique and infinite, and occur mainly in small samples. This problem is further discussed in section 4.2.3.
In addition, if there were a large number of covariates, several iterations would be needed for convergence. To circumvent these problems, Moulton and Zeger (1991) proposed a method for bootstrapping generalized linear models. Essentially, their method uses a one-step approximation to the bootstrap distribution of the coefficient, $\tilde{\beta}_i$, which could be used to assess its variability. In other words, only one iteration is performed for each bootstrap replication rather than allowing the iteratively reweighted least squares (IRLS) algorithm to converge. This method is also useful in small samples where degenerate bootstrap data sets are more likely to occur since on occasion no observations are represented in some covariate groups. In addition, the method is also useful for complicated models and large data sets since computing time is reduced as a result of only one iteration being performed.

In this thesis, a logistic regression model is applied to the bootstrap data sets ignoring the cluster structure. The bootstrap is used to determine the variation of the estimates.

### 2.5.7 Summary of the Bootstrap Method

In summary, the hierarchical bootstrap with two strategies, strategy 1 and strategy 2 (S1 and S2) is proposed as a bootstrap method to handle dependent block data. Unlike other methods which require the specification of the correlation structure, the bootstrap can automatically correct for the correlation structure without specifying it in the modelling process. When the hierarchical bootstrap is used on family data where it is likely that a collection of families will consist of different family sizes, an adjustment should be made to each bootstrap sample in order to have comparable bootstrap replications. Efron and Tibshirani (1993) suggested applying a correction factor to each bootstrap estimate such as $(n^*/n)^{1/2} \tilde{\theta}^*(b)$ where $\tilde{\theta}^*(b)$ is the bootstrap estimate from a bootstrap sample of size $n^*$ and where the empirical distribution is of size $n$. This adjustment is also made to strategy 1 and strategy 2, S1$_{ADJ}$ and S2$_{ADJ}$. The hierarchical bootstrap will be used to compute the coefficient estimates and their
standard errors from a logistic regression model ignoring the cluster structure, using the one-step approach as discussed in section 2.5.6.

2.6 Performance of the Bootstrap versus the GEE

Very few authors have compared the performance of both the bootstrap and the GEE as two methods which do not require the specification of a parametric likelihood to analyze correlated data.

Sherman and Le Cessie (1997) compared the bootstrap with the GEE for correlated outcomes in generalized linear models. They used a bootstrap scheme which they referred to as the “all blocks” bootstrap which is equivalent to strategy 1 of the hierarchical bootstrap. They also used an adjustment on the bootstrap estimates to account for the variability in the sample sizes. This is equivalent to the adjusted version of strategy 1 of the hierarchical bootstrap, S1_{ADJ}. They compared the bootstrap with the GEE using independence and exchangeable working correlations in two examples and several simulation settings with different exchangeable dependence structures. For each bootstrap sample, the regression parameters were estimated by maximizing the likelihood under independence. Confidence intervals for the parameter estimate were used as a measure of comparison. For the bootstrap, the bootstrap-t, percentile, BC and BCα intervals were used. The bootstrap-t confidence interval is the bootstrap version of the t-distribution. It uses every bootstrap replication to construct a pivot and the collection of pivots are used to produce a bootstrap-t distribution. These pivots are similar to the t-statistic such that each bootstrap pivot is computed as \( t^*(b) = \frac{\hat{\beta}^*}{se^*} \). The standard error of each bootstrap-t pivot can be computed using a second level of bootstrap sampling or another reliable estimate for the standard error. Sherman and Le Cessie (1997) used the robust GEE estimates of standard error to avoid the second level of bootstrapping. It is not certain what working correlation they used to obtain the robust estimate. In this thesis, the bootstrap-t
and the BC interval were not examined for several reasons. The bootstrap-t has been known to produce erratic results (Efron and Tibshirani, 1993) and can be unreliable if the covariate distribution is skewed. This type of erratic behaviour was encountered by Sherman and Le Cessie (1997) in some of their examples. In addition, the process of estimating the standard error for each pivot using the GEE would have resulted in a longer bootstrap processing time. The BC interval has been shown to be less powerful than the BCα interval in some situations (Efron, 1987) since the BCα has the added feature of correcting for skewness. Although the BC interval produces better coverage than the percentile interval (Efron, 1987), the percentile interval is examined because of its simplicity in constructing the interval whereas the BCα or BC interval require extra steps.

In the first example, Sherman and Le Cessie (1997) analyzed data with binary outcomes that comprised single independent individuals and various blocks of size 2 and 3 monozygotic and dizygotic siblings. They fitted a logistic regression model with two continuous covariates, one that was cluster specific and the other, individual specific. Overall, the intervals based on the bootstrap were comparable to the intervals for the GEE where the robust estimate of standard error was used. They did not report the observed intra-cluster correlations thus it is possible that the similarities between the methods in this example could be due to an underlying modest correlation and small block sizes. In their second example, they looked at extremely variable Poisson data where the dependence structures within clusters were difficult to model. Their results showed that the confidence intervals varied widely across the bootstrap and GEE methods. They used the empirical distribution of the bootstrap replicates, \(\hat{\beta}^*(b)\), as a diagnostic tool. The distribution was bimodal which explained the variability in the intervals. Further examination indicated that one cluster was influential. In this example, a small number of clusters were used where one was an influential point. Thus in this situation the bootstrap is not an ideal method to use since it is sensitive to outlying values, however it still provides an excellent diagnostic
Sherman and Le Cessie (1997) also performed two simulation experiments, one under a linear regression setting and the other under a logistic regression setting. In the linear regression setting they considered two models where each had one covariate and a group random effect which was used to induce correlation between the outcomes. In the first model (model 1), the covariate was constant within a group but varied between groups (cluster-specific) and in the second model (model 2), the covariate varied within groups (individual-specific). They used 500 replications in the simulation study with B=2000 bootstrap replications for the bootstrap analysis. They looked at balanced samples comprised of 10, 15, 20 and 25 clusters where each cluster was of size 10. In this simulation study the exact or true variance of the ordinary least squares parameter estimate was known and was compared with the bootstrap estimator and the GEE robust estimator using an exchangeable working correlation. In the first setting (model 1), they observed that the bootstrap variance estimate was closer to the true variance than the GEE robust estimate across all sample sizes. However, in small sample sizes the bootstrap standard error was on average too large whereas the GEE robust standard error was on average too small. The coverage rates for the percentile and bootstrap-t confidence intervals were closer to the nominal level of 90% than the robust intervals in most cases but they still undercovered in most cases. The bootstrap-t interval performed the best out of all 3 intervals. In the second setting, the bootstrap estimate of variance and the robust estimate of variance were both comparable to the true variance. The coverage rates of the percentile and robust intervals were similar, with the bootstrap-t interval performing the best. All undercovered. The coverage and variance estimates provided better results in model 2 when the covariate was individual-specific than in model 1 when the covariate was cluster-specific. This is not surprising since more information within the clusters would contribute to the total variability for individual-specific covariates than cluster-specific covariates providing better variance estimates of the
coefficients. Although in both settings (model 1 and model 2), the bootstrap-t intervals provided the best coverage, Sherman and Le Cessie (1997) note that it also provided the widest intervals and occasionally had "wild" endpoints.

In the second simulation study, a logistic regression model with one covariate was used in a variety of settings. In two settings they considered independent binary outcomes which will not be discussed further since correlated outcomes are the focus of this thesis. In other settings they generated correlated binary data with a common tetrachoric correlation in blocks of size 2 (model 3) and blocks of size 10 (model 4). A tetrachoric correlation measures the dependence of the binary outcomes which are assumed to be realizations of continuous latent variables. These latent variables are distributed as multivariate normal with mean vector 0 and correlation matrix \( \Sigma \). In model 3 for 25, 50 and 100 clusters, Sherman and Le Cessie (1997) observed that the robust-GEE interval, BC\(_a\) interval and bootstrap-t interval coverage rates were comparable. In model 4 for 10 and 20 clusters where the block sizes were larger and thus the dependence was heavier, the bootstrap-t interval outperformed the other intervals. When comparing the interval widths, Sherman and Le Cessie observed that not only did the percentile intervals undercover but they were wider than the other intervals. On the other hand, the BC\(_a\) had shorter lengths than its bootstrap competitors. However, the BC\(_a\) intervals were wider than the robust-GEE intervals, but with similar coverage and thus gave less precise information. One problem that Sherman and Le Cessie encountered was the problem of infinite parameter estimates in the logistic model in the independent setting. In this situation, the bootstrap-t and percentile intervals did not work well.

Feng, McLerran and Grizzle (1996) compared the GEE and the bootstrap under a linear model for correlated data with Gaussian error. They focussed on situations in which the number of clusters was small, when cluster sizes were big, and the correlation within the cluster was weak. They examined bias in the estimation of the regression parameters, their standard errors and the type I error of the test statistics.
For the GEE, they assumed an independence working correlation and used the robust variance estimate in their analysis. For the bootstrap, they bootstrapped independent clusters which is strategy 1 of the hierarchical bootstrap. In their simulation study, Feng et al. (1996) used a mixed linear model that is, a random coefficients model to induce correlation between the continuous outcomes. This was generated using the model

$$Y_i = X_i \beta + Z_i b_i + \epsilon_i,$$

where $Y_i = (y_{i1}, \ldots, y_{in_i})$, $i = 1, \ldots, K$ clusters, $\beta$ is a $p \times 1$ vector of fixed regression parameters, $X_i$ is a $n_i \times p$ matrix of covariates that included both subject-specific covariate and cluster-specific covariates, $Z_i$ is a $n_i \times q$ known matrix, $b_i$ is a $q \times 1$ vector of random regression coefficients independently distributed as $N(0, \Psi)$ and $\epsilon_i$ is distributed $N(0, \sigma^2 I_{n_i})$ that is independent of the $b_i$s. They looked at the performance of both methods under different combinations of cluster sizes of 10, 30 and 100 in sample sizes of 10, 20 and 50 clusters. Feng et al. (1996) observed that the MSEs under all configuration combinations from both the bootstrap and the GEE were similar although the bootstrap had a smaller MSE when the sample size was small. The biases were all very small since both procedures would produce unbiased parameter estimates under a Gaussian linear model.

### 2.6.1 Conclusion

It is apparent that the bootstrap is not a frequently used tool to analyze correlated data in regression analysis. Both Feng et al. (1996) and Sherman and Le Cessie (1997) used the bootstrap approach on independent clusters. There does not appear to be any work published that uses a bootstrapping scheme on subcluster units. Although these authors found the bootstrap approach superior in small and in large samples compared to other existing methods used for correlated data such as the GEE, it is of interest to examine a bootstrapping scheme that uses the within cluster information because it may perform better in small samples. This was the motivation.
for proposing the hierarchical bootstrap which takes into account the within cluster information by random sampling with replacement within each cluster (strategy 2) as well as using the clusters as bootstrapping units (strategy 1).

Sherman and Le Cessie did not consider more complicated correlation structures other than the exchangeable correlation in their simulations. Although Feng et al. used a random coefficient model to induce a general correlation structure in their simulated data, they did not consider binary data. Also, these authors focussed on clusters of fixed sizes and not on clusters with sizes that varied. In family data, one cannot assume that there is a common intra-familial correlation. That is, is it safe to assume that the correlation between parents is the same as the correlation between parents and offspring? Also, it is likely that in a population-based association study, a collection of families will consist of families of different sizes. These issues were considered when designing the simulation study.
Chapter 3

A Multifactorial Model for Family Data

with a Note on Cluster-Specific and Population Averaged Approaches

The simulation study was conducted to compare the methods discussed in the previous chapter for the analysis of association studies using family data. Some assumptions were made about the true underlying nature of the disease status for all subjects and used to simulate the data. Once the data were simulated, parameter estimates describing genetic and environmental effects using the different methods were compared to the "true" underlying values. Randomly sampled families from the simulation were used to assess the coverage probability, percent relative bias, relative efficiency and misspecification effect of the parameter estimates of the regression estimates for two GEE and the four hierarchical bootstrap methods. The impact of
sample size, degree of correlation, and genetic model were considered.

To generate relevant data for the simulation study it was necessary to specify an underlying multifactorial genetic model which included major candidate genes, an environmental effect with familial sharing, and a residual component that would exhibit the effect of polygenes with other possibly unmeasured environmental factors.

3.1 Generation of Nuclear Pedigrees

The structure of the nuclear families comprised two parents and a random number of offspring assumed to follow a sibship size distribution. The sibship distribution was assumed to be a geometric distribution with a mean of 0.4551 (Suarez and Van Eerdwegh, 1984) which generated a mean sibship size of 2.2. For each family member, individual-specific covariates were also simulated, which included: sex, genotypes for two candidate genes, and an environmental factor. All were simulated following specific distributions.

3.1.1 Genetic and Environmental Covariates

A multifactorial genetic model, which can include multiple genes and environmental factors was developed for use in the simulation study. The genetic component consisted of candidate genes which are genes that have been localized and mapped and have a known or suspected function in the etiology of a disease. It was assumed that each candidate gene was diallelic (two alleles) and on different chromosomes and therefore segregated independently. Candidate gene 1 (CG1) was assumed to have a dominant pattern of penetrance and candidate gene 2 (CG2) was assumed to have an additive pattern of penetrance.

The parental genotype was simulated based on population allele frequencies that were predefined. Allele frequencies for the candidate genes were fixed at 0.40 for one allele of CG1 and 0.40 for one allele of CG2. The parental genotypes were
assigned assuming Hardy-Weinberg equilibrium (see section 1.1). The genotypes for the offspring were assigned following Mendelian transmission probabilities (see table 1.1) based on the parental mating type. Random number generators in the S-plus software package were used for assigning genotypes for parents and offspring.

The genotypes were then coded based on dominant and additive inheritance. For CG1, two alleles were assigned to the locus to form the genotype. A value was assigned to reflect dominant inheritance for the following genotypes: \( AA=1, Aa=1 \) and \( aa=0 \). Similarly, for CG2 two alleles were also assigned on a separate chromosome: \( B \) and \( b \). A value was given to reflect additive inheritance for the following genotypes: \( BB=2, Bb=1, bb=0 \).

The environmental factor (EF) was generated assuming that each family shared the same level of environmental influence. For each family, a random normal variate with a mean of 31.5 and standard deviation of 5.3 (Genetic Analysis Workshop 9, 1995), was assigned to represent the mean family environmental influence. Individual EF levels were then generated from a second normal distribution using the family EF level for its mean and a standard deviation of 2 so that there was less within-family variability relative to between-family variability.

### 3.1.2 Familial Correlations

Two sources of familial correlation were incorporated into the simulated data. One source was at the covariate level, as described in the previous section, and the other was at the residual level. At the covariate level, it is inherent that family members will share common genes and/or household environment.

A candidate gene covariate will have a covariate intracluster correlation, \( \rho_X \) that will always be positive within families. Clearly this is to be expected as one would expect sharing of alleles between siblings and parent-offspring since the genes of an
offspring are determined solely by parental genes and governed by transmission probabilities. The value of $\rho_X$ when the covariate $X$ is a candidate gene has been analytically shown by Bull et al. (1998) to have an expected value that depends on population allele frequencies and on Mendelian transmission probabilities. These expressions are gene inheritance-specific, that is, for dominant inheritance,

$$E_D(\hat{\rho}_X) \approx \frac{(1 - \frac{3}{4}p)}{2 - p}$$

and for recessive inheritance,

$$E_R(\hat{\rho}_X) \approx \frac{(\frac{1}{4} + \frac{3}{4}p)}{1 + p}$$

where $p$ is the population allele frequency for a diallelic gene with two distinct alleles, $A$ and $a$. One can infer from these expressions that the expected intraclass correlation for a genotype will range between $\frac{1}{4}$ and $\frac{1}{2}$ for siblings and for parents and offspring depending on the allele frequency $p$. Since spouses are not related, the expected intraclass correlation for parents will be close to 0.

The environmental factor generated in the simulation study was correlated among members of the same family since within environmental variation was smaller than between environmental variation. Details were described in section 3.1.1. The environmental factor can be expressed as a value arising from a one-way ANOVA model with random effects,

$$e_{f_{ij}} = 31.5 + e_f + e_{ij}$$

where $e_f \sim N(0, 5.3)$ is the family-specific environmental factor and $e_{ij} \sim N(0, 2)$. If equal-sized families are assumed, then the intrafamilial correlation can be computed as

$$\rho_{\text{intra}} = \frac{\sigma_B^2}{\sigma_B^2 + \sigma_W^2}$$

where $\sigma_B^2$ is the variance between families and $\sigma_W^2$ is the variance within families. For the environmental factor generated in the simulations, $\sigma_B^2 = 5.3$ and $\sigma_W^2 = 2$ yielding a intrafamilial correlation, $\rho \approx 0.73$. The intra-familial correlation will vary in the
simulation since the families generated are of different sizes. However, the value of the intra-familial correlation should be close to the above expression.

In addition, a residual familial component was added. This family residual was generated as a variate from a multivariate normal distribution with mean vector zero and a variance-covariance matrix reflecting the familial correlation structure among spouses, parent-offspring and sibs. The goal was to simulate a residual component that would reflect the presence of other gene(s) or shared environmental factors that were unmeasured or unknown. The residual component for each family contained correlations that were small for spouses but larger for related members in the family, that is, parent-offspring and siblings.

3.1.3 Model Based Simulation

The model developed for the simulation studies was a multifactorial model where the affection status for each individual was determined by the genotype effects and the environmental effect. In addition to these factors contributing to disease, a residual familial component was generated in order to induce possible genetic variability that would be unexplained by the existing measured covariates. This type of extra variation is known as overdispersion or extra-binomial variation. When overdispersion is observed, some of the reasons may be that there are unobserved sources of heterogeneity not accounted for by the covariates in a linear predictor and/or positive correlation between individual binary responses of family members.

Several methods to generate correlated binary data have been proposed (Emrich and Piedmonte, 1991 and Park et al., 1996). Most however, assume exchangeable correlation between observations. Other methods that do generate a correlation structure other than the exchangeable have limitations. For example the Bahadur model, which is feasible for modelling 2 or 3 different correlations and time series models, for correlations that are serial in nature, is not suitable for familial correlations since families can exhibit more than 2 different correlations and unlike time series data, do
not have a natural ordering. The goal was to generate correlated binary data that exhibited a general dependence structure such as one that is familial. Such a model is the multivariate logit-normal (Joe, 1997) with a probability density function,

$$f(y) = \int_{[0,1]^{n_i}} \prod_{j=1}^{n_i} p_j^{y_j} (1 - p_j)^{1-y_j} G(dp)$$  

(3.3)

where $G$ is a cumulative distribution function with support in $[0, 1]$, $p_j = \Pr(y_j = 1)$ and $y_j$ is the $j$th individual response, $j = 1, \ldots, n_i$ in $i = 1, \ldots, m$ families.

In keeping with the notation introduced in Chapter 2, recall that each of $m$ families contain $n_i$ family members. If the responses are binary, a $n_i \times 1$ vector, $y_m = (y_{mi}, \ldots, y_{mni})^T$ can be constructed where $y_{mi} = 1$ if individual $n_i$ from family $i$ is affected with disease and 0 otherwise. Then for each family, a vector of probabilities $p_i = (p_1, \ldots, p_{n_i})$, is assumed to arise from a multivariate logit normal with parameters $\mu$ and $\Sigma = (\sigma_{jk})$:

$$(\log[p_1/(1 - p_1)], \ldots, \log[p_{n_i}/(1 - p_{n_i})]) \sim N_{n_i}(\mu, \Sigma)$$  

(3.4)

where $N_{n_i}(\mu, \Sigma)$ is a $n_i$-variate normal distribution.

The univariate logit-normal density with parameters $\mu$ and $\sigma$ is

$$\phi([\log(p/(1 - p)) - \mu]/\sigma)(\sigma p(1 - p))^{-1}, \quad 0 < p < 1$$  

(3.5)

where $\phi$ is the standard normal pdf. This univariate family has the approximate shapes of the family of beta densities. The density is unimodal if $\sigma$ is small and U-shaped if $\sigma$ is sufficiently large (see figure 3.1).

Let $\sigma_{jj} = \sigma^2$ for all $j$ and $\Sigma = \sigma^2 R$ where $R$ is the correlation matrix with elements $\rho_{ij}$ and ones along the diagonal. If $z \sim N_{n_i}(0, R)$ and assuming (3.4) then $p_i$ can be represented by

$$p_j = \frac{1}{1 + \exp -\{\sigma z_j + \mu_j\}} \quad j = 1, \ldots, n_i.$$  

(3.6)

The marginal parameters from the multivariate binary distribution (3.3) are

$$\pi_j = E(p_j) = \Pr(y_j = 1)$$  

(3.7)
and

\[ Pr(y_{ij}, j = 1, \cdots, n_i) = E\left[\prod_{j=1}^{n_i} p_j^{y_{ij}}(1 - p_j)^{1-y_{ij}}\right] \]

\[ = E\left\{\prod_{j=1}^{n_i} [1 + \exp\{(1 - 2y_{ij})(\sigma z_{ij} + \mu_j)\}]^{-1}\right\}. \tag{3.8} \]

From (3.6) it is seen that when \( \sigma^2 \) is fixed, \( \pi_j \) increases with \( \mu_j \) for \( 1 \leq j \leq m \). When \( \sigma^2 \) is allowed to vary with \( \mu_j \) fixed, then as \( \sigma^2 \to 0 \), \( \pi_j \to (1 + \exp(-\mu_j))^{-1} \). As \( \sigma^2 \to \infty \), \( \pi_j \to 0.5 \). Since when \( z_j > 0 \) then \( p_j \to 1 \) and when \( z_j < 0 \) then \( p_j \to 0 \) and since \( z_j \) in (3.6) is standard normal with mean 0, when \( p_j \to 1 \) then \( z_j > 0 \) with probability 0.5. Hence, \( \pi_j \to 0.5 \).

When \( \mu_j \) and \( \pi_j \) are fixed for \( j = 1, \cdots, n_i \) it is clear that the \( \text{corr}(y_j, y_k) \) increases as \( \rho_{jk} \) increases. When \( \sigma^2 \) increases there is a wider range for \( \text{corr}(y_j, y_k) \) since there is a wider range of dependence for \( p_j \) and \( p_k \). As \( \sigma^2 \to \infty \) then \( p_j \to y_j \) which implies that \( \text{corr}(y_j, y_k) \to \text{corr}(p_j, p_k) \). Similarly, as \( \sigma^2 \to 0 \), \( \text{corr}(y_j, y_k) \to 0 \) for all \( j \neq k \) (see figure 3.1). Hence to use model (3.3) \( \sigma^2 \) should be set large enough in order to obtain a wider range of dependence (Joe, 1997).

The logistic model used to simulate outcomes for each family \( i \) of size \( n_i \) was

\[ \log\left(\frac{p_{ij}}{1 - p_{ij}}\right) = \beta_0 + \beta_1 x_{1ij} + \beta_2 x_{2ij} + \beta_3 x_{3ij} + q_{ij} \tag{3.9} \]

where \( q_{ij} \) is an element from the correlated vector \( Q_i \), \( \beta_1 = \log \text{odds ratio for the effect of CG1}, \beta_2 = \log \text{odds ratio for the effect of CG2} \) and \( \beta_3 = \log \text{odds ratio for the effect of EF} \). The covariates, \( x_1 = \text{CG1}, x_2 = \text{CG2} \) and \( x_3 = \text{EF} \) are generated as described in section 3.1.1. The residual component in (3.9) is the family residual, generated from \( Q_i \sim N_{n_i}(0, \Sigma_i) \), where \( \Sigma_i = \sigma^2 K_i \) is the family variance-covariance matrix that is equal to some constant \( \sigma^2 \) multiplied with the \( i \)th family correlation matrix. Here \( q_{ij} \) is related to \( z_{ij} \) in (3.6) by \( q_{ij} = \sigma z_{ij} \).

Affection status was determined as a stochastic representation of the affection probability. A uniform random number \( (D \sim \text{Unif}(0,1)) \) was generated and an individual, \( ij \), was affected if \( p_{ij} \) (calculated from (3.9)), the probability of affection, was
Figure 3.1: The univariate logit-normal density and a fixed mean of -2 with different variances.
above $D$ or not affected if below $D$. In other words, an individual was more likely to be assigned as affected if their probability of being affected ($p_{ij}$) was high since $D$ would be more likely to be less than $p_{ij}$, the probability of affection for individual $j$ in family $i$.

### 3.1.4 Choice of Values for Logit-Normal Variance

Variance components were introduced in chapter 1 as a method to distinguish genetic and environmental variability of the trait of interest. In (3.9), the model that is used to simulate the data on the logit scale includes a residual component from a correlated vector distributed as a multivariate Normal with mean vector $0$ and a family covariance matrix. This covariance matrix is the product of a common variance $\sigma^2$ and a familial correlation matrix, $K_i$. In variance components analysis, it is assumed that the quantitative trait is normally distributed and in its simplest case, its variance is linearly determined by the genetic variance ($\sigma^2_G$) and the nongenetic variance ($\sigma^2_E$),

$$\sigma^2 = \sigma^2_G + \sigma^2_E.$$  

Therefore, in the logit model (3.9), the variance $\sigma^2$ in the covariance matrix $K_i$ can be viewed as the residual variance component that contributes to the total phenotypic variability. This residual variance component may be additively comprised of smaller polygenic variance components and environmental variance components. In chapter 4, when designing the simulation study, one of the issues that arose was how large to set the residual variance component, $\sigma^2$ in $K_i$. Some examples were sought where authors had used the variance components approach in genetic studies.

Hopper et al. (1982) analyzed systolic blood pressure in 78 Melbourne families using variance components methods. They were interested in examining if the genetic loci which influenced human immune response would also influence the risk of hypertension. Using the residuals obtained from a regression model that was adjusted for age and sex, they estimated the components of residual variation arising from genetic
or environmental sources. For example, using the measured genetic markers or candidate genes (HLA and Gm) known to be associated with human immune response, the variance components method estimates the proportion of variance in blood pressure attributable to these loci. They considered a model where it was assumed that the average contribution to systolic blood pressure variation could be explained as additive effects ($\sigma_A^2$) of a number of unmeasured genes and from HLA ($\sigma_H^2$) and Gm ($\sigma_G^2$). They found that the total phenotypic variance about the age and sex adjusted systolic blood pressure was estimated as $\sigma_T^2 = 182.4$. This was comprised of components due to unmeasured additive genes ($\sigma_A^2 = 57.4$), components due to HLA ($\sigma_H^2 = 20.4$), Gm ($\sigma_G^2 = 21.6$) and a residual environmental component ($\sigma_E^2 = 83.0$). These estimates of the importance of the HLA and Gm loci supported the hypothesis that immunological and immunogenetic factors may play some part in the pathogenesis of hypertension.

Daiger et al. (1984) collected data on human group-specific component (Gc), a plasma transport protein for vitamin D, from 31 monzygotic twin pairs, 13 dizygotic twin pairs and 45 unrelated controls. They observed that there were differences in Gc concentrations between the three distinct Gc genotypes. Using variance components methods they wanted to determine whether these differences were attributable to high heritability, that is, if the square root of the proportion of genetic variance to the total phenotypic variance was high. They estimated that the total genetic variance component ranged from $\sigma_g^2 = 4.22$ for females to $\sigma_g^2 = 20.55$ for males. The environmental variance component was estimated at $\sigma_e^2 = 3.12$ for females and $\sigma_e^2 = 0.98$ for males. Thus the estimated total phenotypic variance was $\sigma_T^2 = 7.34$ for females and $\sigma_T^2 = 21.53$ for males.

These studies represent some examples for values of variance components. In Hopper et al. (1982), the unmeasured additive genes and unmeasured residual environmental factor represented 31% and 46% respectively, of the total variation. In Daiger et al. (1984), the variances of the unmeasured environmental factor represented 43% for females and 5% for males of the total variability. These two examples
show that a total phenotypic variance can be as low as $\sigma^2 = 7.34$ or as high as $\sigma^2 = 182.4$ where the proportion of the unexplained residual component to the total variation can be as low as 5% or as high as 46%. Therefore, it is not unrealistic to assume that the proportion of the variance of an unmeasured residual component to the total variance is high when designing a simulation study.

### 3.2 Population-Averaged Approaches vs Cluster-Specific Approaches

One of the issues that arose in this thesis research, was the distinction between cluster-specific and population-averaged models for binary data. The model that was used to generate the simulated family data (3.9), is a cluster-specific model in which there are individual-specific covariates for each family member and the residual vector, $Q_i$, was used to induce additional unexplained familial correlation. In the binary response setting, the cluster-specific model and the population-averaged model have two very different parameter interpretations.

A simple cluster-specific model can be written as,

$$g(\mu_{ij}) = g(\mathbb{E}[Y_{ij}|r_i]) = (\beta_{0CS} + r_i) + \beta^{CS}X_{ij}$$

(3.10)

where $r_i$ is assumed to be i.i.d. and assumed to follow some distribution, typically a Normal distribution (Stiratelli et al., 1984) where $E(r) = 0$ and $Var(r) = \sigma_r^2$. Given $r_i$, it is assumed that the within-cluster observations for the $i$th cluster are independent. The correlation between the responses in the $i$th cluster arises from the sharing of the unobserved $r_i$. For a binary covariate $X_{ij}$, the interpretation of $\beta^{CS}$ is the log odds for disease in family member $j$ with $X_{ij} = 1$ relative to another family member $j'$ within the same family $i$ that has $X_{ij} = 0$.

A population-averaged model can be written similarly, however, without explicitly
accounting for cluster to cluster heterogeneity,

\[ g(\mu_{ij}) = g(E[Y_{ij}]) = \beta_0 + \beta^{PA}X_{ij}. \]  

(3.11)

For a binary covariate \( X_{ij} \), \( \beta^{PA} \) would be interpreted as the log odds for disease for those in the population with \( X_{ij} = 1 \) relative to those in the population with \( X_{ij} = 0 \).

Zeger, Liang and Albert (1988) suggested analyzing generalized linear random effects models within a marginal framework for discrete and continuous outcomes. Using this approach, if it is assumed that the cluster-specific model, (3.10) holds, then,

\[ \mu_{ij} = E[Y_{ij}|r_i] \]

which denotes the conditional mean of \( Y_{ij} \) given \( r_i \). The marginal mean of \( Y_{ij} \) is obtained by integrating out the random effects, \( r_i \), from the conditional mean,

\[ \nu_{ij} = E[Y_{ij}] = E[E[Y_{ij}|r_i]] = \int g^{-1}(r_i + \beta^{CS}X_{ij})f(r_i)\partial r. \]  

(3.12)

It is easy to see for linear models that have a natural link function,

\[ \nu_{ij} = E[Y_{ij}] = \beta^{CS}X_{ij} + \int r_if(r)\partial r = \beta^{CS}X_{ij} \]  

(3.13)

since \( E(r_i) = 0 \). So, the distinction between the two approaches is irrelevant in the linear case since the coefficient, \( \beta^{CS} \) measures the effect of the covariate \( X \) averaged over the population for both cluster-specific and population-averaged approaches; the regression parameters of these models have both interpretations. However for a logistic regression model, \( g(\nu_{ij}) \) in 3.12 is the logit link function,

\[ \nu_{ij} = E[Y_{ij}] = \int (1 + e^{-r_i - \beta^{CS}X_{ij}})^{-1}f(r)\partial r \]  

(3.14)

which is not equivalent to

\[ E[Y_{ij}] = P(Y_{ij} = 1) = (1 + e^{-\beta_0 - \beta^{PA}X_{ij}})^{-1}, \]

the usual binary logistic form for a population-averaged model. Thus the marginal mean (or the population-averaged mean) of the conditional mean from model 3.10 is
not the same as the marginal mean from model 3.11. Hence for nonlinear models, the distinction between cluster-specific and population-averaged models is important. The regression parameters have different interpretations concerning the effects of covariates on the response probabilities.

Zeger et al. (1988) and Neuhaus, Kalbfleisch and Hauck (1991) compared the cluster-specific and population-averaged models for the binary case and presented similar approximations that showed their relationship to each other. This relationship was given by Neuhaus et al. under the assumption of a cluster-specific model which holds when \( \beta^{CS} \) is not too far from zero,

\[
\beta^{PA} \approx \beta^{CS} \left( 1 - \frac{\text{Var}(p)}{E(p)E(q)} \right) = \beta^{CS}[1 - \rho_Y]
\]

(3.15)

where \( q = 1 - p \) and \( \rho_Y \) is the residual intracluster correlation between responses \( Y_{ij} \) and \( Y_{ik} \), \( 0 \leq \rho_Y \leq 1 \). Using the definition of \( \beta^{PA} \),

\[
\beta^{PA} = \log \left\{ \frac{P(Y = 1|X + 1)/P(Y = 0|X + 1)}{P(Y = 1|X)/P(Y = 0|X)} \right\}
\]

(3.16)

the probability components in (3.16) can be substituted with the population-averaged probabilities from the cluster-specific model (3.14) to express \( \beta^{PA} \) in terms of \( \beta^{CS} \),

\[
\beta^{PA} = \log \left\{ \frac{E[(1 + e^{-r-\beta^{CS}(X+1)})^{-1}]E[(1 + e^{r+\beta^{CS}(X)})^{-1}]}{E[(1 + e^{r+\beta^{CS}(X+1)})^{-1}]E[(1 + e^{-r-\beta^{CS}(X)})^{-1}]} \right\}
\]

(3.17)

where the expectations are with respect to the distribution \( f(\alpha) \). The right-hand side of this expression is approximated by using a Taylor’s expansion around \( \beta^{CS} = 0 \) as suggested by Neuhaus (1992). With some simplification, expression (3.15) is obtained. Intermediate steps leading to this result can be found in appendix A. Expression (3.15) shows that when data are assumed to come from a cluster-specific model, the population-averaged regression coefficients estimated from a marginal model are attenuated. It is clear that as \( \rho_Y \) (or \( \text{var}(p) \)) increases, there is more attenuation. Conversely, when \( \rho_Y = 0 \) (or \( \text{var}(p) = 0 \)) that is, when there are no random (cluster-specific) effects, \( \beta^{CS} = \beta^{PA} \). Also note that for binary data, the population-averaged
parameters will be closer to zero than the cluster-specific parameters (Neuhaus and Jewell, 1993). The attenuation of the regression coefficients also extends to their corresponding estimated standard errors with the same constant of proportionality,

$$se(\hat{\beta}^P) \approx (1 - \rho_Y)se(\hat{\beta}^{CS}).$$ (3.18)

Hence for large samples, dividing the population-averaged parameter (for example, a GEE estimate) by an estimate of its standard error yields a statistic that is approximately the same as the analogous statistic based on the cluster-specific model (Neuhaus, 1993) if the true coefficient, $\beta$ is close to the null, that is if $\beta$ is close to zero. Neuhaus used the relationships of the regression coefficients and the asymptotic standard errors from the population-averaged and cluster-specific models to compare the relative power of tests of the null hypothesis of $\beta^{CS} = 0$ and the estimation efficiency via the Pitman efficiency of the population-averaged approach to the cluster-specific approach. Neuhaus found that for the GEE exchangeable approach the Pitman efficiency was equal to 1. This implied that the Wald tests for the GEE exchangeable approach and cluster-specific approaches would be quite similar. Conversely, for the GEE independence approach, the Pitman efficiency was less than or equal to 1. This suggested that the Wald tests were less powerful under the GEE independence approach than tests based on the cluster-specific or GEE exchangeable approaches since the Pitman efficiency decreased as the intracluster correlation among the responses increased.

Depending on the research interest, the parameters in the marginal (population-averaged) model can be estimated using the GEE as outlined in section 2.4 or, for parameters in a random intercept model (a simple cluster-specific model), maximum likelihood methods for mixed models as well as conditional likelihood methods (which condition on the intercept parameters)(Breslow and Day, 1980) can be used to estimate the slope parameters.

In a conditional logistic regression model, it is assumed that outcomes (such as disease) are independent between family members when conditioned on the family
level covariate such as the intercept term in a family-specific intercept model,

$$\logit(\Pr(y_{ij} = 1|\alpha_i)) = \alpha_i + x_{ij}^T\beta.$$ 

A sufficient statistic for $\alpha_i$ is the sum of the responses in each family. When conditioned on this sufficient statistic, the nuisance parameter $\alpha_i$ is removed from the likelihood which allows the $\beta$s to estimated. The conditional likelihood for $\beta$ has the form

$$\prod_{i=1}^m \frac{\exp(\sum_{j=1}^{n_i} y_{ij}x_{ij}^T\beta)}{\sum_{R_i} \exp(\sum_{j=1}^{n_i} x_{ij}^T\beta)} \quad (3.19)$$

where $y_{i.} = \sum_{j=1}^{n_i} y_{ij}$ and the index set $R_i$ ranges over the $n_i$ choose $y_{i.}$ ways of choosing $y_{i.}$ positive responses out of $n_i$ family members. The structure of the conditional likelihood is similar to the partial likelihood used to fit the Cox survival model which makes it simple to fit this conditional logistic model to binary correlated data. One of the drawbacks with the conditional likelihood approach is that families that have concordant observations do not contribute to the likelihood. That is, families with $y_{i.} = n_i$ or $y_{i.} = 0$ provide no information about the regression coefficients. Therefore, standard errors of the regression coefficients tend to be larger than those from the population-averaged or cluster-specific effects approaches.

In the examples in table 3.1, parameter estimates from the cluster-specific model and conditional model are compared with the estimates from the GEE model to illustrate the attenuation of the estimated parameters. Two simulated data sets of 500 replicates of 100 families, one with a small cluster variance ($\sigma^2 = 9$) and one with a large cluster variance ($\sigma^2 = 25$) are used to show the degree of attenuation. A cluster-specific model was used to generate the data where the true cluster-specific parameters were fixed at $\beta_{CSC1}^{CS} = 2$, $\beta_{CSC2}^{CS} = 3$, $\beta_{ESP}^{CS} = 0.01$. Clearly, the estimates from the marginal approaches are greatly attenuated compared to the cluster-specific approaches. As expected, the standard errors from the conditional likelihood approach were larger than the population-averaged and the cluster-specific approaches. It should be noted in this example, that expression (3.15) may not hold precisely
Table 3.1: Point Estimates (standard errors) of regression coefficients for 4 approaches applied to simulated family data. Outcome is binary. The true cluster-specific parameters were fixed at $\beta_{CG1}^{CS} = 2$, $\beta_{CG2}^{CS} = 3$, $\beta_{EF}^{CS} = 0.01$.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Cluster-Specific (Random Int) $\beta^{CS}$</th>
<th>Conditional Likelihood $\beta^{CS}$</th>
<th>GEE (Indep - R) $\beta^{PA}$</th>
<th>GEE (Exch - N) $\beta^{PA}$</th>
<th>GEE (Exch - R) $\beta^{PA}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma^2 = 9$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CG1</td>
<td>1.45 (0.33)</td>
<td>1.55 (0.66)</td>
<td>1.02 (0.33)</td>
<td>1.01 (0.28)</td>
<td>1.01 (0.29)</td>
</tr>
<tr>
<td>CG2</td>
<td>2.12 (0.24)</td>
<td>2.24 (0.44)</td>
<td>1.49 (0.23)</td>
<td>1.47 (0.20)</td>
<td>1.47 (0.21)</td>
</tr>
<tr>
<td>EF</td>
<td>0.005 (0.03)</td>
<td>0.006 (0.06)</td>
<td>0.003 (0.03)</td>
<td>0.004 (0.02)</td>
<td>0.004 (0.02)</td>
</tr>
<tr>
<td>$\sigma^2 = 25$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CG1</td>
<td>1.04 (0.31)</td>
<td>1.15 (0.55)</td>
<td>0.66 (0.30)</td>
<td>0.65 (0.25)</td>
<td>0.65 (0.30)</td>
</tr>
<tr>
<td>CG2</td>
<td>1.51 (0.22)</td>
<td>1.65 (0.38)</td>
<td>0.95 (0.21)</td>
<td>0.94 (0.18)</td>
<td>0.94 (0.21)</td>
</tr>
<tr>
<td>EF</td>
<td>0.006 (0.03)</td>
<td>0.008 (0.06)</td>
<td>0.003 (0.03)</td>
<td>0.003 (0.02)</td>
<td>0.003 (0.03)</td>
</tr>
</tbody>
</table>

Table 3.2: Wald statistics to test for covariate effects for 4 approaches applied to simulated family data. Outcome is binary.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Cluster-Specific (Random Int) $\beta^{CS}$</th>
<th>Conditional Likelihood $\beta^{CS}$</th>
<th>GEE (Indep - R) $\beta^{PA}$</th>
<th>GEE (Exch - N) $\beta^{PA}$</th>
<th>GEE (Exch - R) $\beta^{PA}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma^2 = 9$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CG1</td>
<td>4.29</td>
<td>2.60</td>
<td>3.12</td>
<td>3.55</td>
<td>3.40</td>
</tr>
<tr>
<td>CG2</td>
<td>8.82</td>
<td>5.15</td>
<td>6.42</td>
<td>7.23</td>
<td>6.91</td>
</tr>
<tr>
<td>EF</td>
<td>0.18</td>
<td>0.10</td>
<td>0.14</td>
<td>0.17</td>
<td>0.16</td>
</tr>
<tr>
<td>$\sigma^2 = 25$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CG1</td>
<td>3.28</td>
<td>2.17</td>
<td>2.18</td>
<td>2.70</td>
<td>2.55</td>
</tr>
<tr>
<td>CG2</td>
<td>6.71</td>
<td>4.33</td>
<td>4.49</td>
<td>5.56</td>
<td>5.19</td>
</tr>
<tr>
<td>EF</td>
<td>0.19</td>
<td>0.14</td>
<td>0.11</td>
<td>0.15</td>
<td>0.15</td>
</tr>
</tbody>
</table>
for the covariates CG1 and CG2 since these provided strong associations. For EF where the association was small, the magnitude of the attenuation increases as the correlation increases, that is when $\sigma^2 = 9$ compared to when $\sigma^2 = 25$.

Since the parameters for CG1 and CG2 were not close to zero, the Wald test statistics in table 3.2 are not comparable across methods. However, for EF where the association was small, the Wald test statistic for the GEE exchangeable approach was similar to the cluster-specific approach when $\sigma^2 = 9$ but less similar when $\sigma^2 = 25$, although it was closer to the cluster-specific value than either of the conditional likelihood and GEE independence approach. This may have been due to using a finite number of samples.

A natural way to collect genetic data is by families which is assumed in model 3.9. The familial residual vector, $Q_i$, induces heterogeneity between families producing a cluster (family)-specific effect in the data. Thus, model 3.9 is essentially a random effects or a cluster-specific model. A population-averaged approach includes models such as the GEE which unlike the cluster-specific models, do not explicitly model the heterogeneity. For linear models, the distinction between the population-averaged (PA) approach and the cluster-specific (CS) approach is not important since both will yield the same estimates and have the same interpretations of the regression coefficients. However, with non-linear models for discrete data, such as logistic regression, population-averaged and cluster-specific models will provide different interpretations for the regression coefficients where each describes different types of effects on the response probabilities. Therefore, the objectives of an analysis must be carefully laid out before choosing an approach.

Different interpretations of research results depend on the question of interest. In family association studies, one may be interested in the population-averaged coefficient value of covariates (such as genetic and environmental factors) on disease in which a GEE model would be appropriate. In other cases, research interest maybe in the relative risk of disease of one family member to another, in which a random-effects
or conditional model would be suitable.

The next chapter discusses the results of the simulation study that utilizes the assumptions made in a multifactorial model for family data presented in this chapter. These assumptions include the construction of family size, genetic and environmental components and familial correlations all which are used in a cluster-specific model to generate correlated binary data.
Chapter 4

Design and Analysis of the Simulation Study

4.1 Introduction

The chapter describes the simulation study which was performed in two stages. In stage one, simulations were conducted on 200 replicates and in stage two on 500 replicates for 25, 50 and 100 families. In order to make the simulations comparable with respect to family size, the same starting seed was used in different models throughout the simulations. This ensured that the same family structures were generated for each family size for making valid comparisons. The two stages of the simulation study were as follows:

1. Select parameter combinations for models which would provide an approximate disease prevalence of 25%. A subset of models was selected where the power (computed using robust-GEE test statistics assuming independence) to reject the null hypothesis of no parameter effect was moderate.
2. Compare the bootstrap and the GEE methods (discussed in chapter 2) for the chosen parameters from stage one. These comparisons were based on various measures of performance: percent relative bias, mean square error, relative efficiency and coverage probabilities.

4.2 Statistical Methods

4.2.1 Some Definitions

In stage one simulations, the GEE test statistic was calculated as the slope estimate over its corresponding standard error estimate. The power of the GEE test statistic was defined as the probability of rejecting a null hypothesis when an alternative hypothesis was true at a given nominal level $\alpha$, for example $\alpha = 0.05$. The power was computed as the proportion of times (out of a total number of simulation runs) the test statistic exceeded a critical value which was determined by the predefined nominal level $\alpha = 0.05$.

In stage two simulations, the GEE confidence intervals were based on the normality assumption of the distribution of the coefficients. Therefore, all 95% confidence intervals for the GEE estimates were constructed using $z$-statistics taken from the standard normal distribution along with the standard error estimate computed from each GEE method. The bootstrap percentile and $BC_a$ intervals discussed in chapter 2 section 2.5.3 were constructed based on 1000 bootstrap replications. Confidence interval coverage was chosen as the primary criterion to compare performance of the methods in stage two.

4.2.2 Measures of Performance

Percent relative bias, mean square error, relative efficiency and coverage of the estimated regression coefficients from the simulations in stage two were examined for
each of the GEE and bootstrap methods. Problems sometimes arose when the GEE method was applied to the 500 simulated replications. Some parameter estimates could not be observed due to lack of convergence (see section 4.2.3) as a result of degenerate data sets. Therefore, the number of nondegenerate data sets, denoted by \( ND \), was used to assess the GEE and bootstrap methods using these measures of performance.

The percent relative bias for the GEE was computed as,

\[
\frac{1}{ND} \sum_{s=1}^{ND} \left( \frac{\hat{\beta}_s - \beta}{\beta} \right) \times 100\% \tag{4.1}
\]

and for the bootstrap approach,

\[
\frac{1}{ND} \sum_{s=1}^{ND} \left( \frac{\hat{\beta}_s^*(\cdot) - \beta}{\beta} \right) \times 100\% \tag{4.2}
\]

where \( \hat{\beta}_s^*(\cdot) = \sum_{b=1}^{B} \hat{\beta}_s^*(b)/B \) (see (2.25)) is the average of a parameter over all \( b \) bootstrap replicates from the \( s \)th simulation replicate.

The mean square error (MSE) which was used to assess efficiency was computed as,

\[
\frac{1}{ND} \sum_{s=1}^{ND} \left( \frac{\hat{\beta}_s - \beta}{\beta} \right)^2 \tag{4.3}
\]

where \( \hat{\beta}_s \) denotes the regression coefficient estimate of the true population-averaged parameter from the \( s \)th simulation. The MSE can also be rewritten as,

\[
\text{var}(\hat{\beta}) + \text{bias}^2,
\]

where the true variance, computed as the variance of \( \hat{\beta} \) over all replications is,

\[
\text{var}(\hat{\beta}) = \frac{1}{ND} \sum_{s=1}^{ND} \left( \frac{\hat{\beta}_s - \bar{\beta}}{\bar{\beta}} \right)^2
\]

where \( \bar{\beta} = \sum_{s=1}^{ND} \hat{\beta}_s / ND = E(\hat{\beta}_s) \). Note that \( \bar{\text{var}} = \sum_{s=1}^{ND} \text{var}(\hat{\beta}_s)/ND \), the average of the variance estimates for the coefficient estimate computed for each replication.
For the GEE the bias is, $\text{bias} = E(\hat{\beta}_n) - \beta$. For the bootstrap, the estimate of bias is, $\hat{\text{bias}}_B = \hat{\theta}^*(.) - \hat{\theta}$.

The coverage probability was computed as the proportion of 95% confidence intervals out of $ND$, that contained the 'true' value of the logistic slope parameter. Each coverage probability was tested (using a one tail test) against the null hypothesis of $H_0: p = 0.95$ (where $p$ is the coverage probability) to determine if the observed coverage probability was significantly different from the nominal level of 95%. The confidence interval length averaged over all replicates was computed to show the overall accuracy of the interval under different conditions such as method or sample size. In addition, the number of misses below and above the confidence intervals under the null model were counted to observe the probability of the confidence interval not covering the true parameter value.

### 4.2.3 Numerical Issues

As previously mentioned, during the fitting process, there were situations when the parameter estimates did not converge. In logistic regression, this phenomenon is known as \textit{separation}. As demonstrated by Albert and Anderson (1984), separation occurs when some level or value of a covariate $x$, can perfectly predict the response $Y$ (known as \textit{complete separation}) or when prediction is perfect except for one or more $x$-values that correspond to both $Y=0$ or $Y=1$ (known as \textit{quasicomplete separation}). This results in nonconvergence of maximum likelihood estimates and standard errors of the estimated coefficients that cannot be estimated. Only when there is overlap in the covariate distribution, such as $x$-values corresponding to $Y=1$ exceeding some but not all of the $x$-values corresponding to $Y=0$ will the maximum likelihood estimates exist. Separation is commonly encountered in small samples and was seen in some of the simulated data sets of 25 families. There is no simple solution to this problem (Albert and Anderson, 1984) and Hosmer and Lemeshow (1989) state that it is a problem one "will have to work around." With this in mind, simulation data sets
were excluded that were degenerate when a GEE model was used. Degenerate data sets occurred when disease susceptibility allele frequencies were set too low and/or when \( \beta_0 \) or \( \sigma^2 \) when using the simulation model of 3.9 were also set too low, producing relatively few outcomes, that is all \( Y=0 \). Therefore, in the simulation study, the disease susceptibility allele frequency was set at 0.40 (GAW9, 1995) for each locus corresponding to CG1 and CG2. For this allele frequency, the combined genotype frequencies for each level of penetrance were kept relatively low, yielding a low disease prevalence. For CG1 and CG2, the genotype frequencies for an individual predisposed with various alleles were as given in the following table:

<table>
<thead>
<tr>
<th>Genotype for CG1 &amp; CG2</th>
<th>AA &amp; BB</th>
<th>AA &amp; Bb</th>
<th>AA &amp; bb</th>
<th>Aa &amp; BB</th>
<th>Aa &amp; Bb</th>
<th>aa &amp; BB</th>
<th>aa &amp; bb</th>
<th>aa &amp; Bb</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>0.03</td>
<td>0.08</td>
<td>0.06</td>
<td>0.08</td>
<td>0.23</td>
<td>0.06</td>
<td>0.13</td>
<td>0.08</td>
</tr>
</tbody>
</table>

4.3 Stage One: The Preliminary Simulation Study for the Selection of Parameter Values

4.3.1 Methods

Different parameter combinations of the simulation model (3.9) were examined which would obtain a desired disease prevalence of 25%. The prevalence was computed by extending the function (1.1) discussed in Chapter 1. Recall that for a disease that is influenced by a single diallelic gene, the prevalence of disease is a function of the genotype frequencies and its corresponding penetrances:

\[ V = p^2 f_{AA} + 2pq f_{Aa} + q^2 f_{aa}. \]

For the genetic model considered here, two candidate genes (CG1 and CG2) have been measured and are assumed to segregate independently. If alleles \( A \) and \( a \) make up CG1 with allele frequency, \( P(A) = p_1 \) and alleles \( B \) and \( b \) make up CG2 with allele
frequency, $P(B) = p_2$ then model (1.1) can be extended to include the probabilities
and penetrance functions of these genes:

$$V_2 = q_1^2 p_2^2 f_{aaBB} + q_1^2 q_2 p_2 q_2 f_{aaBb} + q_1^2 q_2^2 f_{aabb}$$

$$+ 2 p_1 q_1 p_2^2 f_{AaBB} + p_1^2 p_2^2 f_{AAAB} + 2 p_1 q_1 2 p_2 q_2 f_{AaBb}$$

$$+ p_1^2 2 p_2 q_2 f_{AABB} + 2 p_1 q_1 q_2^2 f_{Aabb} + p_1^2 q_2^2 f_{AAAB}$$

(4.4)

where $f_{xyYy}$ is the penetrance associated with genotype $xx$ and genotype $YY$. The
penetrance was computed using the logit-normal model (3.6) where $\mu_j$ for each in-
dividual was equal to the linear predictor of (3.9) at different slope values and at
different $\sigma^2$s. For example, to calculate the penetrance estimate for genotype $aa$ and
genotype $BB$ at $\sigma^2 = 9$ and $\beta = (\beta_0, \beta_1, \beta_2, \beta_3) = (-6, 2, 2, 0.01)$ equation (3.6) was
calculated using $n=50,000$ standard normal variates $z_j$,

$$p_j = \frac{1}{1 + exp \{ \sigma z_j + \mu_j \}} \quad j = 1, \ldots, n$$

with

$$\mu_j = \beta_0 + \beta_1 \times CG1(aa) + \beta_2 \times CG2(BB) + \beta_3 \times \text{avg}(EF)$$

$$= -6 + 2(0) + 2(2) + 0.01(33)$$

since the genotype value for $CG1(aa) = 0$ and $CG2(BB) = 2$. The penetrance
estimate was calculated as the mean of all 50,000 penetrance values. The prevalence
for each parameter combination was calculated using formula (4.4) using various
penetrance value configurations. In appendix B, tables B.1, B.2 and B.3 show the
expected prevalence for different parameter combinations assuming $p_1 = 0.40$ (for
one allele of the diallelic gene, $CG1$) and $p_2 = 0.40$ (for one allele of the diallelic
gene, $CG2$). The gene $CG1$ is inherited in a dominant fashion and $CG2$ is inherited
in an additive fashion. The different values of $\sigma^2$ were chosen based on some of the
plausible values discussed in section 3.1.4.
Table 4.1: Expected Prevalence for various intercept and variance combinations.

<table>
<thead>
<tr>
<th>$\beta_0$</th>
<th>$\sigma^2 = 9$</th>
<th>$\sigma^2 = 16$</th>
<th>$\sigma^2 = 25$</th>
<th>$\sigma^2 = 36$</th>
</tr>
</thead>
<tbody>
<tr>
<td>-4</td>
<td>0.13</td>
<td>0.18</td>
<td>0.23</td>
<td>0.26</td>
</tr>
<tr>
<td>-5</td>
<td>0.08</td>
<td>0.13</td>
<td>0.18</td>
<td>0.22</td>
</tr>
<tr>
<td>-6</td>
<td><strong>0.04</strong></td>
<td><strong>0.09</strong></td>
<td><strong>0.13</strong></td>
<td><strong>0.17</strong></td>
</tr>
<tr>
<td>-7</td>
<td>0.02</td>
<td>0.06</td>
<td>0.10</td>
<td>0.13</td>
</tr>
</tbody>
</table>

Before the slope parameters were selected, the parameters $\beta_0$ and the variance component of the familial covariance matrix $\Sigma_i = \sigma^2 K_i$ (see section 3.1.3) were chosen to obtain a desired baseline population prevalence without any measured genetic and environmental effects. An arbitrary baseline population prevalence $< 10\%$ without covariate effects was assumed. Table 4.1 shows the average prevalence for various intercepts and variances using (3.6) based on 50,000 standard normal variates, $z_i$'s. Models with intercept -6 along with variances 9,16, and 25, respectively, were chosen since these combinations provided the desired range of baseline population prevalence. Notice that as the variance increases, the prevalence also increases as a result of the probabilities becoming more polarized, that is, the probabilities approach either 0 or 1.

The family correlation matrix, $K_i$ of (3.9), was fixed so that it would simulate an unmeasured genetic effect with a possible common environmental effect. Since $K_i$ is fixed on the logit scale, the residual correlations must be set higher to achieve a desired correlation level on the binary level. This is because the residual correlations of the binary outcome are attenuated from the residual correlation on the logit scale. The residual correlation on the logit scale for both siblings and parent-offspring were set to 0.80. The residual correlation between spouses was set to 0.30 which represented shared environment only. This allowed the correlations to include both genetic and environmental components. For example, for a given family with 5 members, $K_i$
would look like,

\[
\begin{pmatrix}
1 & 0.30 & 0.80 & 0.80 & 0.80 \\
0.30 & 1 & 0.80 & 0.80 & 0.80 \\
0.80 & 0.80 & 1 & 0.80 & 0.80 \\
0.80 & 0.80 & 0.80 & 1 & 0.80 \\
0.80 & 0.80 & 0.80 & 0.80 & 1
\end{pmatrix}
\]

Table 4.3 shows the empirical correlations from the simulations between spouses, parent-offspring and siblings for the affection status.

Data sets were generated under both the null hypothesis of \( H_0 : \beta_i = 0, \ i = 1, 2, 3 \) and under the alternative hypothesis of \( H_0 : \beta_i \neq 0, \ i = 1, 2, 3 \). For the alternative models, several parameter combinations were considered so that the aggregate covariate effects would provide an approximate population prevalence of 25%. Parameter values that were used ranged from \( \beta_1 = 1 \) to \( \beta_1 = 3 \) and \( \beta_2 = 1 \) to \( \beta_2 = 3 \). The environmental regression coefficient was fixed at 0.01.

### 4.3.2 Results

'Optimal' parameter combinations to be used in simulations evaluating the GEE and bootstrap methods were selected by initially assessing the power of the robust-GEE test statistics using the model-based assumption of independence for each of the 25, 50 and 100 families. Parameter combinations were included in further simulations if there was moderate power of 50% to 80% of the test statistic for the two candidate gene covariates, CG1 and CG2. This was to ensure some delineation of the performance of power between the two GEE and the four bootstrap methods.

In figures 4.1 and 4.2, the power curves for each parameter at different parameter combinations are shown for all 3 sample sizes. Based on these results, the models in table 4.2 were chosen to depict interesting scenarios of small and large genetic effects in the presence of small and large residual effects.
Figure 4.1: Power Curves for $\beta_1$ with varying $\beta_2$ levels and $\beta_3=0.01$ for $\sigma^2 = 9, 16, 25$
Figure 4.2: Power Curves for $\beta_2$ with varying $\beta_1$ levels and $\beta_3=0.01$ for $\sigma^2 = 9, 16, 25$
Recall that the simulated data were generated under a cluster-specific model as discussed in section 3.2. Therefore, the true cluster-specific slope values used to generate the simulated data would not equal the population-averaged values because of the attenuation of parameter estimates. To obtain “true” population-averaged parameters, a large population of 100,000 individuals was generated using equation (3.6) where \( \mu_j \) represented the explained portion (the linear predictor) of model (3.9) for the selected models in table 4.2. The “true” parameters were estimated using an ordinary logistic model since asymptotically, the coefficients are consistent in both the independent and correlated data settings (Liang and Zeger, 1986). The convergence of the “true” parameters can also be justified using another argument. That is, the maximum likelihood estimate for the population-averaged parameter, \( \tilde{\beta}^{PA} \) estimated under a misspecified model (that is, a population-averaged model when the cluster-specific model is the true one) converges to the value \( (\beta^{PA})^* \) which minimizes the Kullback-Leibler divergence (Neuhaus and Hauck, 1992). These “true” population-averaged parameters given in table 4.2 were used to obtain coverage probabilities and to compute bias for comparing the GEE and bootstrap methods.

For all of the 6 selected models, the susceptibility allele frequencies for CG1 and CG2 were kept constant at 0.40. It would have been interesting to examine different combinations of allele frequencies as well as different inheritance models for CG1 and
CG2. However this would have required performing analyses on a large number of models. This was judged to be infeasible for the bootstrap approach which on average took 4 days to process each experiment. The focus of the simulation study was to compare the methods of variance estimation, hence it was decided to keep the allele frequencies fixed.

The variance components of $\sigma^2 = 9$ and $\sigma^2 = 25$ were chosen to give two different levels of correlation and heterogeneity. The intraclass and interclass correlations of the affection status were estimated from the simulated data sets since these would differ from the fixed correlations at the linear logit level of model (3.9). As mentioned in section 4.3.1, the correlations are attenuated when the logit transformed outcomes of (3.9) are probabilistically dichotomized. Simple Pearson correlation coefficient estimates were computed to provide a rough idea of the correlation induced from model (3.9) on the affection status (table 4.3). The familial correlations for the measured genetic covariates (CG1 and CG2), remained constant for each model since the allele frequencies remained fixed as well as the measured environment (EF). Table 4.4 shows the correlation between family members for each covariate used in the simulation study. The empirical correlations show that for the dominant gene, CG1, there is almost no correlation between spouses and more correlation between siblings than parent-offspring, both of which are less than $\frac{1}{2}$. Similarly, for the additive gene CG2, there is no correlation between spouses but the sibling and parent-offspring correlation are similar at 0.49. For EF, the correlations are much larger than the genetic components.

The misspecification effect (section 2.1) was calculated as the ratio of $\text{var}(\hat{\beta})$ over the simulations using the MLE $\hat{\beta}$ to $\bar{\text{var}}$, the average of the $\text{var}_N(\hat{\beta})$s, the naive estimate of the regression coefficient under the independence assumption. This measure was calculated to estimate the effect of clustering on the variance of the coefficients.
Table 4.3: Empirical intra- and inter-familial correlations for affection status estimated by averaging over 500 replicates. The parent-child interclass correlation is the average of the interclass correlation of father-child and mother-child relationships which were very similar.

<table>
<thead>
<tr>
<th>Model</th>
<th>Husband/Wife</th>
<th>Parent/Child</th>
<th>Siblings</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.110</td>
<td>0.289</td>
<td>0.327</td>
</tr>
<tr>
<td>2</td>
<td>0.157</td>
<td>0.400</td>
<td>0.455</td>
</tr>
<tr>
<td>3</td>
<td>0.109</td>
<td>0.315</td>
<td>0.339</td>
</tr>
<tr>
<td>4</td>
<td>0.161</td>
<td>0.403</td>
<td>0.451</td>
</tr>
<tr>
<td>5</td>
<td>0.100</td>
<td>0.324</td>
<td>0.344</td>
</tr>
<tr>
<td>6</td>
<td>0.154</td>
<td>0.403</td>
<td>0.439</td>
</tr>
</tbody>
</table>

Table 4.4: Empirical intra- and inter-familial correlations for covariates CG1, CG2 and EF estimated by averaging over 500 replicates. The parent-child interclass correlation is the average of the interclass correlation of father-child and mother-child relationships which were very similar.

<table>
<thead>
<tr>
<th>CG1</th>
<th>Husband/Wife</th>
<th>Parent/Child</th>
<th>Siblings</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.009</td>
<td>0.401</td>
<td>0.480</td>
</tr>
<tr>
<td>CG2</td>
<td>Husband/</td>
<td>Parent/</td>
<td>Siblings</td>
</tr>
<tr>
<td></td>
<td>Wife</td>
<td>Child</td>
<td></td>
</tr>
<tr>
<td></td>
<td>-0.003</td>
<td>0.490</td>
<td>0.490</td>
</tr>
<tr>
<td>EF</td>
<td>Husband/</td>
<td>Parent/</td>
<td>Siblings</td>
</tr>
<tr>
<td></td>
<td>Wife</td>
<td>Child</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.634</td>
<td>0.719</td>
<td>0.857</td>
</tr>
</tbody>
</table>
Table 4.5: Misspecification Effect (ME) for $\beta_1=$CG1 slope, $\beta_2=$CG2 slope, $\beta_3=$CG3 slope over 500 simulations. Note $\text{var}_N =$ naive variance estimate and $\text{var}_R =$ robust variance estimate. Variance estimate for EF is variance $\times 10^{-2}$.

For all models in table 4.2, the misspecification effect was computed using 500 replications which is given in table 4.5.

In summary, the parameter combinations given in table 4.2 were chosen to represent different scenarios. For example, data sets generated from models 1 and 2 depict a scenario where the relative risk for CG1 ($e^{\beta_1^{PA}} = e^{1.0889} = 2.97$) is greater than the relative risk for CG2 ($e^{\beta_2^{PA}} = e^{0.5211} = 1.68$) in the presence of a moderate level of unmeasured variation (Model 1) and a high level of unmeasured variation (Model 2). Models 3 to 6 have similar interpretations.
4.4 Stage Two: The Simulation Results for the Evaluation of GEE and Bootstrap Approaches

4.4.1 Percent Relative Bias of Regression Coefficients

In general, the percent relative bias of the bootstrap and the GEE method using the independence working correlation assumption were comparable (see tables 4.6 to 4.8). However, the percent relative bias for the GEE when assuming an exchangeable working correlation for the GEE was always smaller in all sample sizes and models. This may have been due to the GEE with exchangeable working correlation producing more efficient estimates (see next section). For EF, the percent relative bias was much higher since the true EF effect was small.

4.4.2 Efficiency of Regression Coefficients

Tables 4.9, 4.10 and 4.11 show the average of the variance estimates, \( \bar{\text{var}} \), (which were computed as the average of the estimated variances from each simulation over all replicates, see page 73) and mean squared errors (MSE) for covariates CG1, CG2 and EF respectively. It was interesting to observe that the MSEs for all bootstrap strategies were almost equivalent. The MSEs were the smallest for the GEE-exchangeable, the bootstrap strategies had the next smallest MSEs and the GEE-independence MSEs were the largest. This pattern was observed in all models and for all sample sizes. This leads to the interpretation that the GEE-exchangeable gives the most efficient estimates, followed by the bootstrap strategies and the GEE-independence method. Although the underlying correlation structure in this simulation study was not exchangeable, the superior performance of the GEE-exchangeable in terms of efficiency could be attributed to the family structure that was used. In other words, in a nuclear family setting, the familial correlation that was assumed in this simulation was not
<table>
<thead>
<tr>
<th>Model (Size)</th>
<th>GEE</th>
<th>BOOTSTRAP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>I</td>
<td>E</td>
</tr>
<tr>
<td>1 (25)</td>
<td>-2.82</td>
<td>-2.63</td>
</tr>
<tr>
<td>(50)</td>
<td>9.03</td>
<td>7.37</td>
</tr>
<tr>
<td>(100)</td>
<td>3.27</td>
<td>1.96</td>
</tr>
<tr>
<td>2 (25)</td>
<td>4.3</td>
<td>1.64</td>
</tr>
<tr>
<td>(50)</td>
<td>7.51</td>
<td>5.14</td>
</tr>
<tr>
<td>(100)</td>
<td>2.78</td>
<td>1.36</td>
</tr>
<tr>
<td>3 (25)</td>
<td>4.48</td>
<td>1.48</td>
</tr>
<tr>
<td>(50)</td>
<td>8.66</td>
<td>6.54</td>
</tr>
<tr>
<td>(100)</td>
<td>4.72</td>
<td>3.03</td>
</tr>
<tr>
<td>4 (25)</td>
<td>3.74</td>
<td>1.87</td>
</tr>
<tr>
<td>(50)</td>
<td>8.32</td>
<td>6.38</td>
</tr>
<tr>
<td>(100)</td>
<td>5.42</td>
<td>3.91</td>
</tr>
<tr>
<td>5 (25)</td>
<td>7.37</td>
<td>4.98</td>
</tr>
<tr>
<td>(50)</td>
<td>5.78</td>
<td>4.31</td>
</tr>
<tr>
<td>(100)</td>
<td>2.62</td>
<td>1.21</td>
</tr>
<tr>
<td>6 (25)</td>
<td>5.77</td>
<td>2.99</td>
</tr>
<tr>
<td>(100)</td>
<td>5.53</td>
<td>4.26</td>
</tr>
</tbody>
</table>

Table 4.6: Percent Relative Bias for $\hat{\beta}_1 =$ CG1 slope estimate over 500 simulations. Results for GEE: 'I' = independent working correlation and 'E' = exchangeable working correlation. Results for bootstrap: 'S1' = strategy 1 of the bootstrap, 'S2' = strategy 2 of the bootstrap, 'S1<sub>ADJ</sub>' = strategy 1 of the bootstrap with adjustment factor, and 'S2<sub>ADJ</sub>' = strategy 2 of the bootstrap with adjustment factor.
Table 4.7: Percent Relative Bias of $\hat{\beta}_2=CG2$ slope estimate over 500 simulations. Results for GEE: 'I' = independent working correlation and 'E' = exchangeable working correlation. Results for bootstrap: 'S1' = strategy 1 of the bootstrap, 'S2' = strategy 2 of the bootstrap, 'S1_{ADJ}' = strategy 1 of the bootstrap with adjustment factor, and 'S2_{ADJ}' = strategy 2 of the bootstrap with adjustment factor.
Table 4.8: Percent Relative Bias of $\hat{\beta}_3$ = EF slope estimate over 500 simulations. Results for GEE: 'I' = independent working correlation and 'E' = exchangeable working correlation. Results for bootstrap: 'S1' = strategy 1 of the bootstrap, 'S2' = strategy 2 of the bootstrap, 'S1_{adj}' = strategy 1 of the bootstrap with adjustment factor, and 'S2_{adj}' = strategy 2 of the bootstrap with adjustment factor.

<table>
<thead>
<tr>
<th>Model (Size)</th>
<th>GEE</th>
<th>BOOtstrap</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>I</td>
<td>E</td>
</tr>
<tr>
<td>1 (25)</td>
<td>42.1</td>
<td>39.6</td>
</tr>
<tr>
<td>(50)</td>
<td>-28.1</td>
<td>-20.4</td>
</tr>
<tr>
<td>(100)</td>
<td>-62.6</td>
<td>-57.5</td>
</tr>
<tr>
<td>2 (25)</td>
<td>37.6</td>
<td>-21.68</td>
</tr>
<tr>
<td>(50)</td>
<td>-44.2</td>
<td>-64.1</td>
</tr>
<tr>
<td>(100)</td>
<td>-61.9</td>
<td>-46.4</td>
</tr>
<tr>
<td>3 (25)</td>
<td>34.8</td>
<td>33.1</td>
</tr>
<tr>
<td>(50)</td>
<td>-24.2</td>
<td>-28.5</td>
</tr>
<tr>
<td>(100)</td>
<td>-49.7</td>
<td>-44.7</td>
</tr>
<tr>
<td>4 (25)</td>
<td>53.8</td>
<td>37.6</td>
</tr>
<tr>
<td>(50)</td>
<td>-40.3</td>
<td>-67.2</td>
</tr>
<tr>
<td>(100)</td>
<td>-53.8</td>
<td>-29.6</td>
</tr>
<tr>
<td>5 (25)</td>
<td>88.1</td>
<td>69.7</td>
</tr>
<tr>
<td>(50)</td>
<td>11.8</td>
<td>9.04</td>
</tr>
<tr>
<td>(100)</td>
<td>-29.1</td>
<td>-20.7</td>
</tr>
<tr>
<td>6 (25)</td>
<td>49.5</td>
<td>5.63</td>
</tr>
<tr>
<td>(50)</td>
<td>-34.2</td>
<td>-39</td>
</tr>
<tr>
<td>(100)</td>
<td>-10.5</td>
<td>13.5</td>
</tr>
</tbody>
</table>
far from the exchangeable correlation structure since the correlations between parents and offspring and between siblings were assumed equal while the correlation between spouses was negligible.

For the GEE, under the independence working correlation assumption, all of the average variances were less than the MSE indicating that the robust standard errors may be somewhat too small, especially in very small samples. This was true for all sample sizes and models. Under the exchangeable working correlation assumption, both the naive and robust estimates of mean variances were less than the MSE in small samples. However, the robust variance estimates were closer on average to the MSE than the naive variance estimates. This pattern was observed for all sample sizes and models.

For the bootstrap, the variance estimates from all of the strategies were less than the MSE. For strategy 1 (S1), the average of the variance estimates ($\bar{\text{var}}(\hat{\beta})$) was less than the MSE among all the bootstrap strategies. The adjusted strategies produced average variance estimates that were closer to the MSE compared to their unadjusted counterparts. In most cases, strategy 2 for both the unadjusted and adjusted overestimated the true variance. This was to be expected since it was shown analytically that strategy 2 of the hierarchical bootstrap is expected to produce a larger variance than strategy 1. These patterns were observed for all of the sample sizes and all 6 models.

As expected, the efficiency of the parameter estimates increased as the number of families in the sample increased for all methods.

### 4.4.3 Coverage Probabilities of Regression Coefficients

All coverage probabilities are given in tables 4.12 to 4.14. The most noticeable difference between the GEE and bootstrap methods was in the smaller sample size of 25
Table 4.9: Variance of $\hat{\beta}_1 = CGI$ slope estimate over 500 simulations. Results for GEE: ‘I’ = independent working correlation ($\text{var}_R$ = robust variance estimate) and ‘E’ = exchangeable working correlation ($\text{var}_N$ = naive variance estimate and $\text{var}_R$ = robust variance estimate). Results for bootstrap: ‘S1’ = strategy 1 of the bootstrap, ‘S2’ = strategy 2 of the bootstrap, ‘S1\text{ADJ}’ = strategy 1 of the bootstrap with adjustment factor, and ‘S2\text{ADJ}’ = strategy 2 of the bootstrap with adjustment factor.
<table>
<thead>
<tr>
<th>Model (Size)</th>
<th>GEE</th>
<th>Bootstrap</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\hat{\beta}_2$</td>
<td>$\hat{\beta}_2$</td>
</tr>
<tr>
<td></td>
<td>$\text{var}_R$ MSE</td>
<td>$\text{var}_N$ MSE</td>
</tr>
<tr>
<td></td>
<td>$\text{var}_R$ MSE</td>
<td>$\text{var}_{S1}$ MSE</td>
</tr>
<tr>
<td>1 (25)</td>
<td>0.22 0.30</td>
<td>0.20 0.18 0.26</td>
</tr>
<tr>
<td>(50)</td>
<td>0.11 0.14</td>
<td>0.09 0.09 0.11</td>
</tr>
<tr>
<td>(100)</td>
<td>0.06 0.06</td>
<td>0.04 0.05 0.05</td>
</tr>
<tr>
<td>2 (25)</td>
<td>0.18 0.24</td>
<td>0.13 0.13 0.17</td>
</tr>
<tr>
<td>(50)</td>
<td>0.09 0.10</td>
<td>0.06 0.07 0.07</td>
</tr>
<tr>
<td>(100)</td>
<td>0.05 0.06</td>
<td>0.03 0.03 0.04</td>
</tr>
<tr>
<td>3 (25)</td>
<td>0.20 0.28</td>
<td>0.17 0.17 0.22</td>
</tr>
<tr>
<td>(50)</td>
<td>0.10 0.11</td>
<td>0.08 0.08 0.09</td>
</tr>
<tr>
<td>(100)</td>
<td>0.05 0.06</td>
<td>0.04 0.04 0.05</td>
</tr>
<tr>
<td>4 (25)</td>
<td>0.18 0.22</td>
<td>0.12 0.13 0.17</td>
</tr>
<tr>
<td>(50)</td>
<td>0.09 0.10</td>
<td>0.06 0.06 0.07</td>
</tr>
<tr>
<td>(100)</td>
<td>0.04 0.05</td>
<td>0.03 0.03 0.03</td>
</tr>
<tr>
<td>5 (25)</td>
<td>0.23 0.39</td>
<td>0.20 0.21 0.33</td>
</tr>
<tr>
<td>(50)</td>
<td>0.11 0.13</td>
<td>0.09 0.09 0.10</td>
</tr>
<tr>
<td>(100)</td>
<td>0.05 0.07</td>
<td>0.04 0.05 0.06</td>
</tr>
<tr>
<td>6 (25)</td>
<td>0.19 0.27</td>
<td>0.13 0.14 0.21</td>
</tr>
<tr>
<td>(50)</td>
<td>0.09 0.11</td>
<td>0.06 0.07 0.08</td>
</tr>
<tr>
<td>(100)</td>
<td>0.05 0.05</td>
<td>0.03 0.03 0.04</td>
</tr>
</tbody>
</table>

Table 4.10: Variance of $\hat{\beta}_2 = \text{CG2}$ slope estimate over 500 simulations. Results for GEE: 'I' = independent working correlation ($\text{var}_R$ = robust variance estimate) and 'E' = exchangeable working correlation ($\text{var}_N$ = naive variance estimate and $\text{var}_R$ = robust variance estimate). Results for bootstrap: 'S1' = strategy 1 of the bootstrap, 'S2' = strategy 2 of the bootstrap, 'S1, ADJ' = strategy 1 of the bootstrap with adjustment factor, and 'S2, ADJ' = strategy 2 of the bootstrap with adjustment factor.
Table 4.11: Variance of $\hat{\beta}_3=EF$ slope estimate over 500 simulations. Results for GEE: 'I' = independent working correlation (\(\text{var}_R\) = robust variance estimate) and 'E' = exchangeable working correlation (\(\text{var}_N\) = naive variance estimate and \(\text{var}_R\) = robust variance estimate). Results for bootstrap: 'S1' = strategy 1 of the bootstrap, 'S2' = strategy 2 of the bootstrap, 'S1_{ADJ}' = strategy 1 of the bootstrap with adjustment factor, and 'S2_{ADJ}' = strategy 2 of the bootstrap with adjustment factor.
families for which the bootstrap consistently provided the best coverage for the slope values. This pattern was seen across all models and for all of the covariates, CG1, CG2 and EF. Unlike the bootstrap methods, the GEE confidence intervals in 25 families in most cases tended to undercover and were significantly different from the nominal level of 95%. In most cases, the percentile interval tended to overcover for both bootstrap strategies (S1 and S2) as well as their adjusted versions (S1_{ADJ} and S2_{ADJ}). As expected, strategy 2 tended to cover more than strategy 1 since the variance estimates of the coefficients were larger. This is also seen in the longer average interval length for S2. At all sample sizes, the interval lengths for all bootstrap strategies were comparable to the robust-GEE confidence interval assuming an independence working correlation however; the bootstrap had coverage close to the nominal level of 95% more often than the GEE. In general, the bootstrap confidence intervals had longer lengths corresponding to the higher coverage probabilities. Although the GEE produced shorter intervals than the bootstrap, one would not conclude that the GEE intervals are more precise. The GEE uses the standard normal assumption to construct its intervals which may not always be appropriate. The bootstrap on the other hand uses the data at hand to construct its intervals and thus is probably more robust to nonnormal slope estimate distributions unlike the GEE which assumes a normal distribution.

Among the GEE methods, the coverage probabilities were comparable between the exchangeable working correlation and the independence working correlation. However, the average interval lengths from the exchangeable working correlation assumption were consistently shorter, thus providing better accuracy than using the independence assumption.

Tables 4.15 to 4.17 show the miscoverage rate for below and above the confidence intervals for data sets generated under the null hypothesis of no covariate effects. This
is the proportion of times the confidence intervals missed the true value on the left and right sides in 500 simulations. Since this value is analogous to a type I error, the desired miscoverage is 2.5% on each side. When the miscoverage rate is greater than 2.5%, then the interval undercovers on the side that is observed. Similarly, when the miscoverage rate is less than 2.5%, then the interval overcovers on the side that is observed. Similar patterns seen in the non-null models were observed for the sample of 25 families across all methods. The GEE consistently undercovered on both sides of the intervals whereas the bootstrap undercovered and overcovered depending on the bootstrap interval used, however, it tended to produce miscoverage rates closer to the target rate of 2.5% than the GEE. For strategy 2 of the bootstrap in 25 families, both the percentile and $BC_a$ intervals tended to overcover with the percentile overcovering more times than the $BC_a$. In strategy 1 of the bootstrap, the percentile intervals were comparable to the $BC_a$ intervals for all sample sizes.

4.4.4 Analysis of Variance of Coverage Probabilities

The primary measure used to compare the two methods was the coverage probability. The previous section highlighted several patterns but did not suggest the preferred method overall. The GEE and bootstrap methods can be assessed by performing an analysis of variance of the coverage to suggest the better method.

Analysis of variance was performed on the logit transformations of coverage probabilities where the factors of interest were: method (6), model (6) and sample size (3). In this $6 \times 6 \times 3$ factorial model, there were 3 main effects: method, model and size; three two-way interactions: method $\times$ size, model $\times$ size and method $\times$ model; and one three-way interaction: method $\times$ model $\times$ size. In this unreplicated factorial design, it was assumed that the higher order interaction, method $\times$ model $\times$ size was negligible. Its mean squares was used to estimate error with 100 degrees of freedom.
Table 4.12: Observed Coverage at the nominal 95 % Confidence Interval (with Average Interval Length) for $\hat{\beta}_1=CG1$ slope estimate over 500 simulations. Asterisks denote coverage probabilities that are significantly different from the nominal level of 95% using a one tailed test. Results for GEE: 'I' = independent working correlation and 'E' = exchangeable working correlation. Results for bootstrap: 'S1' = strategy 1 of the bootstrap, 'S2' = strategy 2 of the bootstrap, 'S1_{ADJ}' = strategy 1 of the bootstrap with adjustment factor, and 'S2_{ADJ}' = strategy 2 of the bootstrap with adjustment factor.
Table 4.13: Observed Coverage at the nominal 95% Confidence Interval (with Average Interval Length) for $\beta_2=CG2$ slope estimate over 500 simulations. Asterisks denote coverage probabilities that are significantly different from the nominal level of 95% using a one tail test. Results for GEE: ’I’ = independent working correlation and ’E’ = exchangeable working correlation. Results for bootstrap: ’S1’ = strategy 1 of the bootstrap, ’S2’ = strategy 2 of the bootstrap, ’S1_{ADJ}’ = strategy 1 of the bootstrap with adjustment factor, and ’S2_{ADJ}’ = strategy 2 of the bootstrap with adjustment factor.
Table 4.14: Observed Coverage at the nominal 95% Confidence Interval (with Average Interval Length) for $\beta_3=EF$ slope estimate over 500 simulations. Asterisks denote coverage probabilities that are significantly different from the nominal level of 95% using a one tail test. Results for GEE: 'I' = independent working correlation and 'E' = exchangeable working correlation. Results for bootstrap: 'S1' = strategy 1 of the bootstrap, 'S2' = strategy 2 of the bootstrap, 'S1_{ADJ}' = strategy 1 of the bootstrap with adjustment factor, and 'S2_{ADJ}' = strategy 2 of the bootstrap with adjustment factor.
Table 4.15: The number of misses above (upper) and below (lower) the 95% confidence intervals for \( \hat{\beta}_1 = \text{CG1} \) slope estimate under the null model. Results for GEE: 'I' = independent working correlation and 'E' = exchangeable working correlation. Results for bootstrap: 'S1' = strategy 1 of the bootstrap, 'S2' = strategy 2 of the bootstrap, 'S1_{ADJ}' = strategy 1 of the bootstrap with adjustment factor, and 'S2_{ADJ}' = strategy 2 of the bootstrap with adjustment factor.

<table>
<thead>
<tr>
<th>Model</th>
<th>GEE</th>
<th>BOOTSTRAP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>I</td>
<td>E</td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>N</td>
</tr>
<tr>
<td>( \sigma^2 = 9 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(25)</td>
<td>0.04</td>
<td>0.028</td>
</tr>
<tr>
<td>(50)</td>
<td>0.032</td>
<td>0.04</td>
</tr>
<tr>
<td>(100)</td>
<td>0.026</td>
<td>0.028</td>
</tr>
<tr>
<td>( \sigma^2 = 25 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(25)</td>
<td>0.056</td>
<td>0.058</td>
</tr>
<tr>
<td>(50)</td>
<td>0.046</td>
<td>0.046</td>
</tr>
<tr>
<td>(100)</td>
<td>0.034</td>
<td>0.020</td>
</tr>
</tbody>
</table>

Table 4.16: The number of misses above (upper) and below (lower) the 95% confidence intervals for \( \hat{\beta}_2 = \text{CG2} \) slope estimate under the null model. Results for GEE: 'I' = independent working correlation and 'E' = exchangeable working correlation. Results for bootstrap: 'S1' = strategy 1 of the bootstrap, 'S2' = strategy 2 of the bootstrap, 'S1_{ADJ}' = strategy 1 of the bootstrap with adjustment factor, and 'S2_{ADJ}' = strategy 2 of the bootstrap with adjustment factor.

<table>
<thead>
<tr>
<th>Model</th>
<th>GEE</th>
<th>BOOTSTRAP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>I</td>
<td>E</td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>N</td>
</tr>
<tr>
<td>( \sigma^2 = 9 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(25)</td>
<td>0.044</td>
<td>0.038</td>
</tr>
<tr>
<td>(50)</td>
<td>0.032</td>
<td>0.030</td>
</tr>
<tr>
<td>(100)</td>
<td>0.022</td>
<td>0.034</td>
</tr>
<tr>
<td>( \sigma^2 = 25 )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(25)</td>
<td>0.036</td>
<td>0.038</td>
</tr>
<tr>
<td>(50)</td>
<td>0.036</td>
<td>0.028</td>
</tr>
<tr>
<td>(100)</td>
<td>0.038</td>
<td>0.030</td>
</tr>
<tr>
<td></td>
<td>0.030</td>
<td>0.030</td>
</tr>
</tbody>
</table>
Table 4.17: The number of misses above (upper) and below (lower) the 95% confidence intervals for $\hat{\beta}_3 = \text{EF}$ slope estimate under the null model. Results for GEE: ‘I’ = independent working correlation and ‘E’ = exchangeable working correlation. Results for bootstrap: ‘S1’ = strategy 1 of the bootstrap, ‘S2’ = strategy 2 of the bootstrap, ‘S1\text{ADJ}’ = strategy 1 of the bootstrap with adjustment factor, and ‘S2\text{ADJ}’ = strategy 2 of the bootstrap with adjustment factor.

For all of the three covariates, CG1, CG2 and EF, there were significant interactions between model $\times$ size and method $\times$ size but these appeared to be due to variability from using a finite number of replications and possibly, the choice of a starting seed. That is, there was some lack of parallalism between the mean plots of the size and model and between the mean plots of size and methods in some families. There was no significant overall difference among all 6 models. However, there were significant differences between methods.

To see these differences among methods more clearly, tests were performed by using contrasts between a subset of methods. In tables 4.18, 4.19 and 4.20, there are a total of 9 comparison tests. Since there were multiple comparisons, a Bonferroni correction was made by adjusting the significance level of $\alpha = 0.05$ to $\alpha' \approx 0.05/9 \approx 0.005$. For CG1 and CG2, strategy 1 of the bootstrap gave an average coverage probability that was closer to the nominal level (when averaged over all models and sample


<table>
<thead>
<tr>
<th>Comparison</th>
<th>F Value</th>
<th>Pr &gt; F</th>
</tr>
</thead>
<tbody>
<tr>
<td>GEE vs Bootstrap</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GEE (I-R) vs Bootstrap (S1)</td>
<td>2.87</td>
<td>0.0931</td>
</tr>
<tr>
<td>GEE (E) vs Bootstrap (S1)</td>
<td>0.94</td>
<td>0.3335</td>
</tr>
<tr>
<td>GEE (I-R) vs Bootstrap (S2)</td>
<td>8.11</td>
<td>0.0053</td>
</tr>
<tr>
<td>GEE (E) vs Bootstrap (S2)</td>
<td>59.56</td>
<td>0.0001</td>
</tr>
<tr>
<td>GEE (I-R) vs Bootstrap - (S1 - ADJ)</td>
<td>1.48</td>
<td>0.2262</td>
</tr>
<tr>
<td>GEE (E) vs Bootstrap - (S1 - ADJ)</td>
<td>6.48</td>
<td>0.0124</td>
</tr>
<tr>
<td>GEE (I-R) vs Bootstrap - (S2 - ADJ)</td>
<td>111.74</td>
<td>0.0001</td>
</tr>
<tr>
<td>GEE (E) vs Bootstrap - (S2 - ADJ)</td>
<td>79.37</td>
<td>0.0001</td>
</tr>
</tbody>
</table>


sizes) and was significantly different from the GEE using an exchangeable working correlation. For CG2, the adjusted version of strategy 1 of the bootstrap which also provided better coverage on average, was significantly different from the GEE with an exchangeable working correlation. For EF which had a stronger intracluster correlation, strategy 1 and its adjusted method of the bootstrap were significantly different from the GEE with an exchangeable working correlation which on average produced lower coverage. For all three covariates, strategy 2 of the bootstrap along with its correction (ADJ), were significantly different from the GEE with an independence working correlation and the GEE with an exchangeable working correlation. However, this difference was due to the tendency of strategy 2 of the bootstrap to overcover.

It should be noted that some inconsistencies were observed in the simulation results as evident in the coverage probabilities where it appeared that a sample size of 50 families provided better coverage than a sample size of 100 families. One would expect the reverse to hold since increasing sample size would increase the
Comparison | F Value | Pr > F |
---|---|---|
GEE vs Bootstrap | 3.29 | 0.0725 |
GEE (I-R) vs Bootstrap (S1) | 3.45 | 0.0661 |
GEE (E) vs Bootstrap (S1) | 11.92 | 0.0008 |
GEE (I-R) vs Bootstrap (S2) | 39.67 | 0.0001 |
GEE (E) vs Bootstrap (S2) | 180.67 | 0.0001 |
GEE (I-R) vs Bootstrap - (S1 - ADJ) | 2.85 | 0.0947 |
GEE (E) vs Bootstrap - (S1 - ADJ) | 13.40 | 0.0004 |
GEE (I-R) vs Bootstrap - (S2 - ADJ) | 125.69 | 0.0001 |
GEE (E) vs Bootstrap - (S2 - ADJ) | 64.06 | 0.0001 |


Comparison | F Value | Pr > F |
---|---|---|
GEE vs Bootstrap | 29.07 | 0.0001 |
GEE (I-R) vs Bootstrap (S1) | 13.60 | 0.0004 |
GEE (E) vs Bootstrap (S1) | 6.38 | 0.0131 |
GEE (I-R) vs Bootstrap (S2) | 71.41 | 0.0001 |
GEE (E) vs Bootstrap (S2) | 302.45 | 0.0001 |
GEE (I-R) vs Bootstrap - (S1 - ADJ) | 0.08 | 0.7835 |
GEE (E) vs Bootstrap - (S1 - ADJ) | 54.44 | 0.0001 |
GEE (I-R) vs Bootstrap - (S2 - ADJ) | 71.39 | 0.0001 |
GEE (E) vs Bootstrap - (S2 - ADJ) | 10.93 | 0.0013 |

Table 4.21: The average of the logit of the coverage probabilities and average of the coverage probabilities over all sample sizes and models. Results for GEE: 'I' = independent working correlation and 'E' = exchangeable working correlation where 'N' = naive variance and 'R' = robust variance. Results for bootstrap: 'S1' = strategy 1 of the bootstrap, 'S2' = strategy 2 of the bootstrap, 'S1_ADJ' = strategy 1 of the bootstrap with adjustment factor, and 'S2_ADJ' = strategy 2 of the bootstrap with adjustment factor.

<table>
<thead>
<tr>
<th>β</th>
<th>GEE</th>
<th>Bootstrap</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>I</td>
<td>E</td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>N</td>
</tr>
<tr>
<td>CG1</td>
<td>2.59</td>
<td>2.67</td>
</tr>
<tr>
<td></td>
<td>93.0</td>
<td>93.5</td>
</tr>
<tr>
<td>CG2</td>
<td>2.60</td>
<td>2.63</td>
</tr>
<tr>
<td></td>
<td>93.1</td>
<td>93.3</td>
</tr>
<tr>
<td>EF</td>
<td>2.48</td>
<td>2.77</td>
</tr>
<tr>
<td></td>
<td>92.3</td>
<td>94.1</td>
</tr>
</tbody>
</table>

precision of the confidence intervals. One possible reason for this was the use of a finite number of simulation replicates. It would have been ideal to use more replicates, however, this was not feasible due to the demands that the bootstrap imposed on computational time. As a check, different replicates of models were generated using several starting seeds (see table 4.22). The robust-GEE was applied to different replicates (since its processing time was relatively fast compared to the bootstrap) to observe whether the same inconsistencies (which were constant across the GEE and bootstrap methods) as in the original simulation study would arise. Table 4.22 shows the coverage probabilities using different starting seeds from the same generating model. Using seed 77, it appears that a sample of 50 families has better coverage than 100 families. Using seed 199, a sample of 100 families has better coverage than 50 families. It is probably safe to assume that the anomaly observed in the simulation study was due to the choice of the starting seed and using a finite number of replicates.
Table 4.22: Coverage probabilities for model 5 in table 4.2 using different starting seeds.

<table>
<thead>
<tr>
<th>Family Size</th>
<th>CG1</th>
<th>CG2</th>
<th>EF</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEED=77</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>0.932</td>
<td>0.911</td>
<td>0.907</td>
</tr>
<tr>
<td>50</td>
<td>0.934</td>
<td>0.938</td>
<td>0.948</td>
</tr>
<tr>
<td>100</td>
<td>0.93</td>
<td>0.922</td>
<td>0.938</td>
</tr>
<tr>
<td>SEED=199</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>0.925</td>
<td>0.903</td>
<td>0.892</td>
</tr>
<tr>
<td>50</td>
<td>0.932</td>
<td>0.934</td>
<td>0.91</td>
</tr>
<tr>
<td>100</td>
<td>0.95</td>
<td>0.942</td>
<td>0.934</td>
</tr>
</tbody>
</table>

4.5 Recommendations for Practice

It is evident from the simulations that in small sample sizes, such as 25 families, that in terms of coverage probabilities, the bootstrap outperforms the GEE with either independence or exchangeable working correlations. However in some cases, strategy 2 of the bootstrap tended to overcover. In general, strategy 1 of the bootstrap and its adjusted form, appeared to provide better coverage than the exchangeable GEE. However, from the simulations and the ANOVA analysis, strategy 1 of the bootstrap is not significantly different from the robust-GEE with independence assumption. Nevertheless, strategy 1 of the bootstrap and its adjusted form is the method of choice over the GEE methods since it performs well in all sample sizes from small to moderate. The biggest drawback in using the bootstrap is its demands on computational time relative to the GEE that can compute estimates in very little time. As noted previously, the bootstrap can take up to 4 days to process compared to the S-plus GEE function written by Vince Carey (Statlib, 1997) which only uses up to 10 minutes to process. Therefore, it seems that accuracy comes with some cost in computational time.
Chapter 5

Application to Genetic Analysis

Workshop 9 Data Set

The Genetic Analysis Workshops (GAWs) are held every two years for statistical geneticists and genetic epidemiologists to compare and assess statistical methods in genetic analysis. For each GAW, real and computer-simulated data sets are distributed worldwide. The focus has been to examine the analytical methods used in statistical genetic analysis as well as to compare the range of conclusions that could be drawn from the same set of data using different methods of analysis.

5.1 The Data

For the Genetic Analysis Workshop 9 (GAW9), one of the simulated data sets provided was of a common oligogenic disease, defined by an underlying quantitative trait in 23 randomly ascertained, extended pedigrees. Altogether there were 1497 individuals. The data comprised a few related quantitative traits, an environmental factor, candidate locus genotypes, and genotypes for highly polymorphic marker loci. In this data set, the prevalence of disease was approximately 10%.
Each individual was genotyped at 180 highly polymorphic loci on 6 chromosomes of which 168 were anonymous DNA markers and the remaining 12 were candidate loci. The anonymous DNA markers had 4 to 9 alleles, and the candidate loci had 2 to 3 alleles.

The model that was used to generate the data had 4 major genes (MG1 ... MG4) influencing one or more of the four quantitative traits, Q1 to Q4. Other factors in the model included, age, an environmental factor (EF) which was individual-specific and normally distributed with mean 31.5 and standard deviation 5.3 and other random environmental factors. Figure 5.1 which is taken from the GAW9 issue of Genetic Epidemiology (1995) describes the generating model graphically and fig 5.2 shows the locations of the major genes and candidate loci for all 6 chromosomes. Affection status for every individual was defined if their Q1 trait value was greater than the
Figure 5.2: Chromosomal locations of major loci and candidate loci for GAW9 data set. The genetic distance between loci are given in centiMorgans which is shown above the line segments connecting each locus. (Editors-in-Chief: Chakravarti and Mulvihill, Genetic Epidemiology vol. 12, no. 6, 1995. Reprinted by permission of Wiley-Liss Inc., a subsidiary of John Wiley & Sons Inc.)

predefined threshold of 87.5. The Q1 trait was directly influenced by major genes MG1 and MG2 which were also candidate genes C5 and C2 (8% and 16% of the phenotypic variance) respectively. There were no other direct associations between the remaining candidate genes and Q1. Q1 was indirectly influenced through the intervening quantitative trait Q3 (11% of phenotypic variance), by polygenes (3%), by age (20%), by EF (10%) and by random environmental factors (32%).

5.2 Methods

Nuclear families were extracted out of the 23 extended pedigrees such that each included the proband (a family member that has met some criteria and brings the family into a sample). Altogether there were 122 individuals. Only the candidate genes were used in identifying the major genes since these were known to have influence on the
traits associated with the disease.

Candidate gene (C5) was examined, the major gene MG1, which directly influenced Q1. Out of the 3 alleles in C5, allele 3 was known to be the disease susceptibility allele. Therefore, C5 was coded as a binary variable where C5=1 if the genotype included allele 3 and C5=0 if it did not. The original cutoff of Q1>87.5 produced only 6% of the individuals as affected in this data set of nuclear families. Therefore, in order to increase the number of affected individuals in this trimmed data set, an arbitrary cutoff value for Q1 of ≥ 72 was used which corresponded to a population prevalence of 31.1% (out of the original 1000 individuals). In an initial analysis performed using the original cutoff, very little power was available to detect the major gene effect (C5) using both methods.

5.3 Results

The GEE confidence intervals were able to detect MG1 (C5) as being significantly associated with disease. Both bootstrap confidence intervals (percentile and $BC_a$) for strategies one and two showed that C5 was not significantly associated with disease. However for strategy one, the confidence intervals slightly covered zero indicating that C5 was borderline significant. In this data set, 13 individuals carried allele 3 in C5 (that is, C5=1) where only 4 individuals were affected and 9 individuals were not affected. Out of 8 individuals who were affected, 4 carried allele 3 and the other 4 did not. Due to the sparseness of the data, the bootstrap approach occasionally produced degenerate bootstrap samples which were about 2% of 5000 bootstrap samples used. Using these degenerate bootstrap samples resulted in very large regression coefficients, large estimated standard errors and negative standard errors caused by numerical problems in the decomposition of the covariance matrix (for the slope estimate) in the fitting process. The degenerate data sets from strategy
<table>
<thead>
<tr>
<th>Model</th>
<th>I-R</th>
<th>E-N</th>
<th>E-R</th>
<th>S1</th>
<th>S2</th>
<th>S1_ADJ</th>
<th>S2_ADJ</th>
</tr>
</thead>
<tbody>
<tr>
<td>C5</td>
<td>2.42 (0.74)</td>
<td>2.49 (0.86)</td>
<td>2.49 (0.80)</td>
<td>2.36 (0.94)</td>
<td>2.85 (1.73)</td>
<td>2.37 (0.95)</td>
<td>2.86 (1.74)</td>
</tr>
<tr>
<td>EF(c)</td>
<td>0.24 (0.04)</td>
<td>0.24 (0.06)</td>
<td>0.24 (0.04)</td>
<td>0.23 (0.04)</td>
<td>0.23 (0.06)</td>
<td>0.23 (0.05)</td>
<td>0.23 (0.06)</td>
</tr>
<tr>
<td>AGE</td>
<td>0.06 (0.02)</td>
<td>0.06 (0.02)</td>
<td>0.06 (0.02)</td>
<td>0.06 (0.02)</td>
<td>0.06 (0.03)</td>
<td>0.06 (0.02)</td>
<td>0.06 (0.03)</td>
</tr>
<tr>
<td>SEX</td>
<td>-0.77 (0.38)</td>
<td>-0.76 (0.48)</td>
<td>-0.76 (0.39)</td>
<td>-0.77 (0.40)</td>
<td>-0.68 (0.61)</td>
<td>-0.77 (0.40)</td>
<td>-0.68 (0.61)</td>
</tr>
</tbody>
</table>

Table 5.1: Coefficient estimates (standard errors) from a model fitted to GAW9 data using C5, EF (centered), age and sex.

one did not appear to produce large slope values but only negative standard errors of the coefficients. The same degenerate data sets were also in strategy two in addition to other degenerate data sets that produced large coefficient estimates. Therefore, only those degenerate bootstrap samples that produced negative standard errors in both bootstrap strategies were excluded in the analysis. It is very unlikely that removing bootstrap samples that gave negative standard errors (about 1% of the total number of bootstrap replications) would bias the results since 5000 bootstrap samples were used. The degenerate data sets that produced large slope values (which were in strategy two only), were not excluded. It was difficult to assess what was considered a large slope value since the hierarchical bootstrap that uses a one-step iteration in the fitting process stops the estimate from approaching infinity after the first step.

Figure 5.3 shows the bootstrap distribution (strategy one) for C5. The bootstrap distribution for C5 is skewed to the left and extends below zero. Since 4950 bootstrap samples were used, figure 5.3 is a good approximation of the empirical distribution for the regression coefficient which does not appear to be normally distributed. Therefore, the GEE confidence intervals may be incorrect since the confidence intervals are constructed assuming the coefficient estimate for C5 is normally distributed.

Table 5.1 summarizes the results for C5 and other covariates. For the environmental factor (EF(c)), and age both the GEE and bootstrap approaches produced
Table 5.2: Misspecification Effects.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Robust Variance</th>
<th>Naive Variance</th>
<th>Misspecification Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>C5</td>
<td>0.54</td>
<td>0.63</td>
<td>0.86</td>
</tr>
<tr>
<td>EF(c)</td>
<td>0.002</td>
<td>0.003</td>
<td>0.60</td>
</tr>
<tr>
<td>AGE</td>
<td>0.0005</td>
<td>0.0004</td>
<td>1.44</td>
</tr>
<tr>
<td>SEX</td>
<td>0.14</td>
<td>0.24</td>
<td>0.60</td>
</tr>
</tbody>
</table>

Table 5.3: 95% confidence intervals for the regression coefficients using the GEE with length in brackets.

<table>
<thead>
<tr>
<th>Variable</th>
<th>I R</th>
<th>E R</th>
</tr>
</thead>
<tbody>
<tr>
<td>C5</td>
<td>[0.98, 3.68]</td>
<td>[0.80, 4.18]</td>
</tr>
<tr>
<td></td>
<td>(2.7)</td>
<td>(3.38)</td>
</tr>
<tr>
<td>EF(c)</td>
<td>[0.17, 0.32]</td>
<td>[0.12, 0.36]</td>
</tr>
<tr>
<td></td>
<td>(0.15)</td>
<td>(0.24)</td>
</tr>
<tr>
<td>AGE</td>
<td>[0.02, 0.11]</td>
<td>[0.03, 0.10]</td>
</tr>
<tr>
<td></td>
<td>(0.09)</td>
<td>(0.07)</td>
</tr>
<tr>
<td>SEX</td>
<td>[-1.52, -0.03]</td>
<td>[-1.70, 0.18]</td>
</tr>
<tr>
<td></td>
<td>(1.49)</td>
<td>(1.88)</td>
</tr>
</tbody>
</table>

Comparable results. Table 5.2 shows the misspecification effect for the main effects that include the major gene effect. This is calculated as the ratio of the robust variance to the naive variance assuming independence. For the candidate gene main effect, C5, the robust-GEE variance (assuming independence) was smaller than the naive variance. Bull et al. (1995) also observed a similar misspecification effect. They attributed this unexpected result to a lack of within-family gene variation.

Tables 5.3 and 5.4 show the 95% confidence intervals for the covariates using the GEE and the bootstrap. In table 5.3, the robust-GEE confidence intervals with independence correlation structure were similar to the robust-GEE with an exchangeable correlation assumption. This may have been due to minimal correlation in the data as was seen in the misspecification effects. The naive GEE under the exchangeable assumption produced wider intervals due to the larger standard error estimate of the coefficient. In table 5.4, for strategy one, the $BC_a$ confidence intervals which are
Table 5.4: 95% confidence intervals for the regression coefficients using the bootstrap with length in brackets.

<table>
<thead>
<tr>
<th>Variable</th>
<th>S1</th>
<th>S2</th>
<th>S1\text{adj}</th>
<th>S2\text{adj}</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PI</td>
<td>BCₐ</td>
<td>PI</td>
<td>BCₐ</td>
</tr>
<tr>
<td>C5</td>
<td>-0.11, 0.37</td>
<td>-0.76, 0.66</td>
<td>-0.79, 0.82</td>
<td>-0.75, 0.37</td>
</tr>
<tr>
<td>EF(c)</td>
<td>0.14, 0.32</td>
<td>0.16, 0.33</td>
<td>0.13, 0.35</td>
<td>0.14, 0.35</td>
</tr>
<tr>
<td>AGE</td>
<td>0.01, 0.11</td>
<td>0.01, 0.10</td>
<td>0.01, 0.12</td>
<td>0.01, 0.12</td>
</tr>
<tr>
<td>SEX</td>
<td>-1.52, 0.001</td>
<td>-1.53, -0.001</td>
<td>-1.88, 0.005</td>
<td>-2.04, 0.39</td>
</tr>
</tbody>
</table>

Figure 5.3: Bootstrap distribution for C5. Left histogram is strategy 1 and right histogram is strategy 1 (adjusted).
more robust to skewness are likely more accurate. These $BC_a$ confidence intervals also provided better coverage more times than the GEE for small sample sizes in the simulation study. For strategy two, it is difficult to make any inferences for C5 since the confidence intervals may be biased from large coefficients. The bootstrap percentile intervals for strategy 1 (and its adjusted form) provided shorter intervals compared to the $BC_a$. However, it is probably not wise to infer that the percentile intervals are more accurate in this case. The percentile interval may not provide the best coverage since the underlying distribution is skewed or biased. There were no differences between the unadjusted and adjusted $BC_a$ bootstrap intervals for strategy one.

This data set showed some of the potential problems with the hierarchical bootstrap approach when the data is sparse and the sample size is small. Although this approach uses a one-step iteration process, it is clear from this example that it is still possible to sample degenerate bootstrap data sets. However, there were relatively few problems with strategy one compared to strategy two. It is probable that the $BC_a$ interval from strategy one is more correct than the GEE confidence intervals since evidence from the simulation study suggests that the $BC_a$ interval provides better coverage under strategy 1 more times than the GEE in smaller sample sizes. Furthermore, the histogram showing the bootstrap distribution for C5 indicates that the distribution is skewed making it more likely that the $BC_a$ interval is more accurate than the GEE confidence interval.
Chapter 6

Discussion and Areas for Further Research

Two approaches, one semi-parametric the other nonparametric, to analyze correlated binary data were evaluated and compared. These were the generalized estimating equations approach and the bootstrap approach. They each have advantages and disadvantages. One of the advantages of the GEE approach that the bootstrap lacks, is that much existing software exists that is relatively simple to use and computation- ally efficient. The GEE function that was used for this thesis was the S-plus version written by Vince Carey (Statlib, 1997). Others include the SAS GEE macro written by Karim and Zeger (Karim MR, 1989) and the QGE software (Fred Hutchinson Cancer Research Center, 1994). On the other hand, although bootstrap functions exist, the existing ones cater to individual data points that are assumed independent. A bootstrap function was written specifically for the bootstrap approaches used in this thesis. In addition, although the function was written as efficiently as possible, it placed enormous demands on computational time. For example, for one simulation, the GEE would take approximately 10 minutes to process whereas the bootstrap would take approximately 3-4 days to process.
However, there were advantages to the bootstrap. Most of the time it outperformed the GEE in terms of coverage probabilities in the smallest sample size of 25 families used in the simulations. In terms of efficiency and bias, it was a slight improvement over the robust-GEE. Furthermore, no assumptions about the underlying correlation structure needed to be specified.

In most cases, the GEE confidence intervals, using either the exchangeable (model-based and robust variance estimates) or the independence assumption (robust variance estimate only) produced poor coverage probabilities for the smallest sample size of 25 families. In terms of efficiency, the GEE with exchangeable working correlation produced the most efficient estimates among all the methods. The independence GEE was the least efficient. Using the exchangeable working correlation was appropriate for nuclear families in this simulation design because the correlation among the relationship types were similar.

Throughout the development of this thesis, many issues arose. In the existing literature, there are few algorithms or methods dealing with the generation of correlated binary data with a general dependence structure in the marginal framework. For this reason, it was necessary in this thesis to develop a model that would induce a general correlation structure. This model was based on a cluster-specific model which introduced issues surrounding the interpretation of research results for binary data. That is, the different interpretation of regression coefficients from population-averaged and cluster-specific models. This stressed the importance of establishing the research question before an analysis is conducted. More work is needed to develop new algorithms which would better accommodate the testing of new methods for marginal models developed for correlated binary data.

This research considered only nuclear family structures. It would be interesting to examine similar methods for extended families. The hierarchical bootstrap approach
could be modified for multi-stage nested designs relevant to the extended family structure. This modified hierarchical bootstrap may be more suitable than the GEE to handle more complex correlation structures as well as smaller sample sizes of extended families. Strategy 2 of the hierarchical bootstrap may be a strong competitor to the other bootstrap strategies and the GEE since it uses information on variability within families as well. Although the GEE with exchangeable working correlation may be a reasonable assumption to make in analyzing nuclear families, it may be inappropriate for extended families since it is likely the correlation structure will be more complex.

Finally, it would be interesting to examine the behaviour of the GEE and bootstrap estimates when the population consists of several mixed sub-populations. One example is ethnicity. In some ethnic groups, allele frequencies for certain disease genes are higher than other ethnic groups. Associations may result from population stratification rather than a direct association of genes. Furthermore, environmental exposures may also be ethnic-specific. The implications on population-averaged estimates when the data is a mixed population would be an interesting area for further research.
Appendix A

Intermediate Steps for Some Results

A.1 Hierarchical Bootstrap

In chapter 2 the hierarchical bootstrap is proposed as a bootstrap method to resample pedigree data. The results of Davison and Hinkley’s (1997) is generalized from the balanced case to the unbalanced case since bootstrapping families of unequal size is examined.

Let $y_{ij}^*$ denote the phenotype for individual $j$ in family $i$, $i = 1, 2, \cdots, a$, $j = 1, 2, \cdots, n_i$.

In (2.35), the theoretical bootstrap result for $\text{var}_\bar{P}$ is derived as follows:

\[
\text{var}_\bar{P}(y_{ij}^*) = \frac{1}{m} \sum_{i=1}^{m} \sum_{j=1}^{n_i} \frac{1}{n_i} (y_{ij}^* - \bar{y}_{..})^2
\]
\[
= \frac{1}{m} \sum_{i=1}^{m} \sum_{j=1}^{n_i} \frac{1}{n_i} [(y_{ij}^* - \bar{y}_i^*) + (\bar{y}_i^* - \bar{y}_{..})]^2
\]
\[
= \frac{1}{m} \sum_{i=1}^{m} \sum_{j=1}^{n_i} \frac{1}{n_i} [(y_{ij}^* - \bar{y}_i^*)^2 + 2(y_{ij}^* - \bar{y}_i^*)(\bar{y}_i^* - \bar{y}_{..}) + (\bar{y}_i^* - \bar{y}_{..})^2]
\]
\[ I = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{n_i} \left[ \sum_{j=1}^{n_i} (y_{ij}^* - \bar{y}_i^*)^2 + n_i (\bar{y}_i^* - \bar{y}^*)^2 \right] \]
\[ = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{n_i} SS_{W|i} + \frac{1}{m} \sum_{i=1}^{m} (\bar{y}_i^* - \bar{y}^*)^2 \]
\[ = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{n_i} SS_{W|i} + \frac{SS_B}{m} \]

\[ \bar{y}_{i.}^2 = \left( \frac{1}{n_i} \sum_{j=1}^{n_i} y_{ij}^* \right)^2 \]
\[ = \frac{1}{n_i^2} (y_{i1}^* + \cdots + y_{in_i}^*)^2 \]
\[ = \frac{1}{n_i^2} \left( \sum_{j=1}^{n_i} y_{ij}^2 + \sum_{u,v=1}^{n_i} y_{iu}^* y_{iv}^* \right) \]

Therefore,
\[ \sum_{i=1}^{m} \bar{y}_{i.}^2 = \sum_{i=1}^{m} \sum_{j=1}^{n_i} \frac{1}{n_i} y_{ij}^2 + \sum_{u,v=1}^{n_i} \sum_{i=1}^{m} \frac{1}{n_i^2} y_{iu}^* y_{iv}^* \] \hspace{1cm} (A.1)

\[ SS_B = \sum_{i=1}^{m} (\bar{y}_i^* - \bar{y}^*)^2 \]
\[ = \sum_{i=1}^{m} \bar{y}_{i.}^2 - 2\bar{y}_i^* \bar{y}^* + \bar{y}^* \]
\[ = \sum_{i=1}^{m} \bar{y}_{i.}^2 - a\bar{y}^* \] \hspace{1cm} (A.2)

\[ SS_W = \sum_{i=1}^{m} \sum_{j=1}^{n_i} (y_{ij}^* - \bar{y}_i^*)^2 \]
\[ = \sum_{i=1}^{m} \sum_{j=1}^{n_i} y_{ij}^2 + \sum_{i=1}^{m} \sum_{j=1}^{n_i} \bar{y}_i^* \]
\[ - 2 \sum_{i=1}^{m} \bar{y}_i^* \sum_{j=1}^{n_i} y_{ij}^* + \sum_{i=1}^{m} \sum_{j=1}^{n_i} \frac{y_{ij}^*}{n_i} \bar{y}_i^* \]
\[ = \sum_{i=1}^{m} \sum_{j=1}^{n_i} y_{ij}^2 - \sum_{i=1}^{m} n_i \bar{y}_i^2 \]
\[ = \sum_{i=1}^{m} \sum_{j=1}^{n_i} y_{ij}^2 - \sum_{i=1}^{m} n_i \bar{y}_i^2 \] \hspace{1cm} (A.3)
In the unbalanced case, the $SS_W$ will be family-specific since it will depend on the family size,

$$SS_{W[i]} = \sum_{j=1}^{n_i} y_{ij}^* - n_i\bar{y}_i^*$$

$$= \sum_{j=1}^{n_i} (y_{ij}^* - \bar{y}_i^*)$$  \hspace{1cm} (A.4)

where $SS_{W[i]}$ is the within sum of squares for the $i$th family. Therefore,

$$SS_W = \sum_i^m SS_{W[i]}$$  \hspace{1cm} (A.5)

For strategy 1,

$$\text{cov}_F(y_{ij}^*, y_{ik}^*) = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{n_i(n_i - 1)} \left[ \sum_{u,v=1}^{n_i} y_{iu}^* y_{iv}^* \right] - \bar{y}_i^*$$

$$= \frac{1}{m} \sum_{i=1}^{m} \frac{1}{n_i(n_i - 1)} \left[ n_i^2 \bar{y}_{i*}^2 \sum_{j=1}^{n_i} y_{ij}^* \right] - \bar{y}_i^*$$

$$= \frac{1}{m} \sum_{i=1}^{m} \frac{1}{n_i(n_i - 1)} \left[ n_i \bar{y}_{i*}^2 - \bar{y}_i^* \right] \left[ \sum_{j=1}^{n_i} y_{ij}^* \right] - \bar{y}_i^*$$

$$= \frac{1}{m} SS_B + \frac{1}{m} \sum_{i=1}^{m} \frac{1}{n_i(n_i - 1)} \left[ \frac{n_i}{n_i - 1} \bar{y}_{i*}^2 - \frac{n_i}{n_i - 1} \bar{y}_{i*}^2 \right] - \frac{1}{m} \sum_{i=1}^{m} \sum_{j=1}^{n_i} \frac{y_{ij}^*}{n_i(n_i - 1)}$$

$$= \frac{1}{m} SS_B - \frac{1}{m} \sum_{i=1}^{m} \frac{1}{n_i(n_i - 1)} \left( \sum_{j=1}^{n_i} y_{ij}^* \right)^2$$

substituting A.4 for the expression in the parenthesis,

$$\text{cov}_F(y_{ij}^*, y_{ik}^*) = \frac{1}{m} SS_B - \frac{1}{m} \sum_{i=1}^{m} \frac{SS_{W[i]}}{n_i(n_i - 1)}$$  \hspace{1cm} (A.6)

For strategy 2,

$$\text{cov}_F(y_{ij}^*, y_{ik}^*) = \frac{1}{m} \sum_{i=1}^{m} \sum_{u,v=1}^{n_i} \frac{1}{n_i^2} y_{iu}^* y_{iv}^* - \bar{y}_i^*$$

$$= \frac{1}{m} \left( \sum_{i=1}^{m} \sum_{u,v=1}^{n_i} y_{iu}^* y_{im}^* \frac{1}{n_i^2} \right) - \bar{y}_i^*$$
Davison and Hinkley looked at the expectations of $SS_B$ and $SS_W$ to see how well it mimicked the theoretical variability. Recall that for

$$SS_B = \sum_{i=1}^{m} \bar{y}_{i}^2 - m\bar{y}^2.$$  

From (2.33),

$$\bar{y}_{..} = \bar{x} + \bar{z}.$$  

then  

$$E(\bar{y}_{..}) = \mu_x$$  

and

$$\bar{y}_{i} = \bar{x}_i + \bar{z}_i.$$  

then  

$$E(\bar{y}_{i}) = \mu_x.$$  

Therefore,

$$SS_B = \sum_{i=1}^{m} (x_i + \bar{z}_i)^2 - m(\bar{x} + \bar{z}.)^2$$

$$= \sum_{i=1}^{m} (x_i^2 + 2x_i\bar{z}_i + \bar{z}_i^2) - m(\bar{x} + 2\bar{z}_. + \bar{z}.)$$

(A.8)

So,

$$E(SS_B) = E[\sum_{i=1}^{m} (x_i^2 + \bar{z}_i^2)] - mE[\bar{x}^2] - mE[\bar{z}^2]$$

(A.10)

Note that,

$$E(\bar{z}_i^2) = \text{Var}(\bar{z}_i) = \frac{\sigma^2_z}{n_i}$$

$$E(\bar{z}^2) = \text{Var}(\bar{z}_.) = \text{Var} \left( \frac{\sum_i \sum_j z_{ij}}{\sum_i n_i} \right) = \text{Var} \left( \frac{\sum_i n_i \bar{z}_i}{\sum_i n_i} \right) = \frac{\sigma^2_z}{\sum_i n_i}$$

$$E(x^2_i) = \text{Var}(x_i) + E^2(x_i) = \sigma^2_x + \mu^2_x$$

$$E(\bar{x}^2) = \text{Var}(\bar{x}) + E^2(\bar{x}) = \text{Var} \left( \frac{\sum_i n_i x_i}{\sum_i n_i} \right) + E^2(\bar{x}) = \frac{\sum_i n_i^2 \sigma^2_x}{(\sum_i n_i)^2} + \mu^2_x$$
Therefore,

\[ E[SS_B] = m\sigma_x^2 \left[ 1 - \frac{\sum_i n_i^2}{(\sum_i n_i)^2} \right] + \sigma_z^2 \left[ \sum_i \frac{1}{n_i} - \frac{a}{\sum_i n_i} \right] \]  \hspace{1cm} (A.11)

and,

\begin{align*}
E[SS_W] &= E\left[\sum_{i=1}^{m} SS_W[i]\right] \\
&= \sum_{i=1}^{m} E[SS_W[i]] \\
&= \sum_{i=1}^{m} (n_i - 1)E \left[ \frac{\sum_{j=1}^{n_i} (y_{ij} - \bar{y}_i)^2}{n_i - 1} \right] \\
&= \sum_{i=1}^{m} (n_i - 1)\sigma_z^2 \\
&= \sum_{i=1}^{m} (n_i - 1)\sigma_z^2 \hspace{1cm} (A.12)
\end{align*}

since,

\[ SS_W[i] = \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_i)^2 \]

\[ = \sum_{j=1}^{n_i} (z_{ij} - \bar{z}_i)^2 \]

then,

\[ E[SS_W[i]] = E\left[\sum_{j=1}^{n_i} (z_{ij} - \bar{z}_i)^2\right] \\
= (n_i - 1)E \left[ \frac{\sum_{j=1}^{n_i} (z_{ij} - \bar{z}_i)^2}{n_i - 1} \right] \\
= (n_i - 1)\sigma_z^2 \\
&= \sum_{j=1}^{n_i} (n_i - 1)\sigma_z^2 \hspace{1cm} (A.13)
\]

Therefore for strategy 1,

\[ E[cov_F(y_{ij}, y_{ik})] = \frac{1}{m} E(SS_B) - \frac{1}{m} \sum_i \frac{1}{n_i(n_i - 1)} E(SS_W[i]) \]

\[ = \sigma_x^2 \left[ 1 - \frac{\sum_i n_i^2}{(\sum_i n_i)^2} \right] + \frac{1}{m}\sigma_z^2 \left[ \sum_i \frac{1}{n_i} - \frac{m}{\sum_i b_i} \right] - \frac{1}{m} \sum_i \frac{1}{n_i(n_i - 1)} (n_i - 1)\sigma_z^2 \\
= \sigma_x^2 \left[ 1 - \frac{\sum_i n_i^2}{(\sum_i n_i)^2} \right] - \frac{1}{\sum_i n_i} \sigma_z^2 \hspace{1cm} (A.14)
\]
and for strategy 2,

$$E[\text{cov}_F(y^*_i, y^*_k)] = \sigma^2_z \left[ 1 - \frac{\sum_i n_i^2 z_i}{(\sum_i n_i)^2} \right] + \frac{\sigma^2_z}{m} \left[ \sum_i \frac{1}{n_i} - \frac{m}{\sum_i n_i} \right] \quad (A.16)$$

### A.2 Relationship between Cluster-Specific and Population-Averaged Regression Parameters

Given:

$$\beta_{PA} = \log \left( \frac{E \left( \frac{1}{1 + e^{-r - \beta(X+1)}} \right) E \left( \frac{1}{1 + e^{r+\beta X}} \right)}{E \left( \frac{1}{1 + e^{r+\beta(X+1)}} \right) E \left( \frac{1}{1 + e^{-r - \beta X}} \right)} \right)$$

$$= F(X; \beta) \quad (A.17)$$

Since the expectation expressions in A.10 is difficult to compute, the right-hand side of (A.17) is evaluated using a first-order Taylor’s expansion around $\beta^* = 0$,

$$F(X; \beta) \approx F(X; \beta = \beta^*) + \frac{\partial F(X; \beta = \beta^*)}{\partial \beta} (\beta - \beta^*).$$

Note that,

$$F(X; \beta = \beta^*) = \log \left( \frac{E \left( \frac{1}{1 + e^{-r}} \right) E \left( \frac{1}{1 + e^r} \right)}{E \left( \frac{1}{1 + e^r} \right) E \left( \frac{1}{1 + e^{-r}} \right)} \right)$$

$$= \log 1 = 0 \quad (A.18)$$

Let

$$r(X; \beta) = E \left( \frac{1}{1 + e^{-r - \beta(X+1)}} \right)$$

$$s(X; \beta) = E \left( \frac{1}{1 + e^{r+\beta X}} \right)$$

$$t(X; \beta) = E \left( \frac{1}{1 + e^{r+\beta(X+1)}} \right)$$

and,

$$u(X; \beta) = E \left( \frac{1}{1 + e^{-r - \beta X}} \right)$$
Then,

\[
\frac{\partial F(X; \beta = \beta^*)}{\partial \beta} = \frac{t(X)u(X)[r'(X)s(X) + s'(X)r(X)] - r(X)s(X)[t'(X)u(X) + u'(X)t(X)]}{[t(X)u(X)]^2}
\]

where each component on the right-hand side is evaluated at \( \beta = \beta^* \),

\[
\begin{align*}
\frac{\partial}{\partial \beta} \left( \frac{1}{1 + e^{-r-\beta x}} \right) &= E \left( \frac{(X + 1)e^{-r-\beta(X+1)}}{1 + e^{-r-\beta(X+1)}} \right) = E \left( \frac{X e^{-r}}{1 + e^{-r}} \right) \\
\frac{\partial}{\partial \beta} \left( \frac{1}{1 + e^{r+\beta x}} \right) &= E \left( \frac{-X e^{r+\beta X}}{1 + e^{r+\beta X}} \right) = E \left( \frac{-X e^{r}}{1 + e^{r}} \right) \\
\frac{\partial}{\partial \beta} \left( \frac{1}{1 + e^{-r-\beta X}} \right) &= E \left( \frac{X e^{-r-\beta X}}{1 + e^{-r-\beta X}} \right) = E \left( \frac{X e^{-r}}{1 + e^{-r}} \right)
\end{align*}
\]

Since \( \text{logit}(p) = r \) this implies that

\[
\begin{align*}
p &= \frac{e^r}{1 + e^r} = \frac{1}{1 + e^{-r}}, \\
q &= 1 - p = \frac{1}{1 + e^{-r}} = \frac{e^{-r}}{1 + e^{-r}} \quad \text{and} \\
pq &= \frac{e^{-r}}{1 + e^{-r}} = \frac{e^r}{1 + e^r}
\end{align*}
\]

The components of A.12 are then simplified:

\[
\begin{align*}
r(X; \beta = \beta^*) &= E(q) \quad \text{and} \quad r'(X; \beta = \beta^*) = (X + 1)E(pq) \\
s(X; \beta = \beta^*) &= E(p) \quad \text{and} \quad s'(X; \beta = \beta^*) = -XE(pq) \\
t(X; \beta = \beta^*) &= E(q) \quad \text{and} \quad t'(X; \beta = \beta^*) = -(X + 1)E(pq) \\
u(X; \beta = \beta^*) &= E(p) \quad \text{and} \quad u'(X; \beta = \beta^*) = XE(pq).
\end{align*}
\]
After some simplification,

\[
\frac{\partial F(X; \beta = \beta^*)}{\partial \beta} = \frac{E(pq)}{E(p)E(q)}
\]

\[
= \frac{E(p)E(q) - E(p)E(q) + E(pq)}{E(p)E(q)}
\]

\[
= 1 + \frac{\text{cov}(p, q)}{E(p)E(q)} \quad \text{(since cov}(p, q) = E(pq) - E(p)E(q))
\]

\[
= 1 + \frac{\text{cov}(p, 1-p)}{E(p)E(q)}
\]

\[
= 1 + \frac{\text{cov}(p, -p)}{E(p)E(q)}
\]

\[
= 1 - \frac{\text{var}(p)}{E(p)E(q)} \quad \text{(A.20)}
\]

Thus,

\[
\beta^p \approx \beta^{CS} \left[ 1 - \frac{\text{var}(p)}{E(p)E(q)} \right].
\]
Appendix B

Tables of Estimated Prevalence for Various Parameter Combinations
Table B.1: Prevalence for various parameter combinations with $\sigma^2 = 9$

<table>
<thead>
<tr>
<th>$\beta^T = (\beta_0, \beta_1, \beta_2, \beta_3)$</th>
<th>Penetrence Estimate</th>
<th>Expected Prevalence ($\hat{K}_2$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(-6,0,0,0,01)</td>
<td>$f_{AAABB}$</td>
<td>$f_{AABB}$</td>
</tr>
<tr>
<td>(-6,1,0,0,01)</td>
<td>0.054</td>
<td>0.054</td>
</tr>
<tr>
<td>(-6,0,1,0,01)</td>
<td>0.054</td>
<td>0.054</td>
</tr>
<tr>
<td>(-6,1,1,0,01)</td>
<td>0.054</td>
<td>0.054</td>
</tr>
<tr>
<td>(-6,1,2,0,01)</td>
<td>0.054</td>
<td>0.054</td>
</tr>
<tr>
<td>(-6,2,0,0,01)</td>
<td>0.054</td>
<td>0.054</td>
</tr>
<tr>
<td>(-6,1,3,0,01)</td>
<td>0.054</td>
<td>0.054</td>
</tr>
<tr>
<td>(-6,2,1,0,01)</td>
<td>0.054</td>
<td>0.054</td>
</tr>
<tr>
<td>(-6,2,2,0,01)</td>
<td>0.054</td>
<td>0.054</td>
</tr>
<tr>
<td>(-6,3,0,0,01)</td>
<td>0.054</td>
<td>0.054</td>
</tr>
</tbody>
</table>

Table B.2: Prevalence for various parameter combinations with $\sigma^2 = 16$

<table>
<thead>
<tr>
<th>$\beta^T = (\beta_0, \beta_1, \beta_2, \beta_3)$</th>
<th>Penetrence Estimate</th>
<th>Expected Prevalence ($\hat{K}_2$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(-6,0,0,0,01)</td>
<td>$f_{AAABB}$</td>
<td>$f_{AABB}$</td>
</tr>
<tr>
<td>(-6,1,0,0,01)</td>
<td>0.100</td>
<td>0.100</td>
</tr>
<tr>
<td>(-6,0,1,0,01)</td>
<td>0.100</td>
<td>0.100</td>
</tr>
<tr>
<td>(-6,1,1,0,01)</td>
<td>0.100</td>
<td>0.100</td>
</tr>
<tr>
<td>(-6,1,2,0,01)</td>
<td>0.100</td>
<td>0.100</td>
</tr>
<tr>
<td>(-6,2,0,0,01)</td>
<td>0.100</td>
<td>0.100</td>
</tr>
<tr>
<td>(-6,1,3,0,01)</td>
<td>0.100</td>
<td>0.100</td>
</tr>
<tr>
<td>(-6,2,1,0,01)</td>
<td>0.100</td>
<td>0.100</td>
</tr>
<tr>
<td>(-6,2,2,0,01)</td>
<td>0.100</td>
<td>0.100</td>
</tr>
<tr>
<td>(-6,3,0,0,01)</td>
<td>0.100</td>
<td>0.100</td>
</tr>
<tr>
<td>(-6,1,3,0,01)</td>
<td>0.100</td>
<td>0.100</td>
</tr>
<tr>
<td>(-6,3,2,0,01)</td>
<td>0.100</td>
<td>0.100</td>
</tr>
<tr>
<td>(-6,3,3,0,01)</td>
<td>0.100</td>
<td>0.100</td>
</tr>
</tbody>
</table>
Table B.3: Prevalence for various parameter combinations with $\sigma^2 = 25$

<table>
<thead>
<tr>
<th>$\beta^T$</th>
<th>$\beta_0$, $\beta_1$, $\beta_2$, $\beta_3$</th>
<th>$f_{aaBB}$</th>
<th>$f_{aaBb}$</th>
<th>$f_{aabb}$</th>
<th>$f_{AABB}$</th>
<th>$f_{AABb}$</th>
<th>$f_{AAbb}$</th>
<th>$f_{AABb}$</th>
<th>$f_{AAbb}$</th>
<th>Expected Prevalence ($K_2$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(-6,0,0,0.01)</td>
<td>0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145</td>
<td>0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145</td>
<td>0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(-6,1,0,0.01)</td>
<td>0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145</td>
<td>0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145</td>
<td>0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145 0.145</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(-6,0,1,0.01)</td>
<td>0.248 0.193 0.145 0.248 0.248 0.248 0.193 0.193 0.193 0.193</td>
<td>0.248 0.193 0.145 0.248 0.248 0.248 0.193 0.193 0.193 0.193</td>
<td>0.248 0.193 0.145 0.248 0.248 0.248 0.193 0.193 0.193 0.193</td>
<td>0.248 0.193 0.145 0.248 0.248 0.248 0.193 0.193 0.193 0.193</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(-6,1,2,0.01)</td>
<td>0.380 0.248 0.145 0.453 0.453 0.453 0.311 0.311 0.311 0.311</td>
<td>0.380 0.248 0.145 0.453 0.453 0.453 0.311 0.311 0.311 0.311</td>
<td>0.380 0.248 0.145 0.453 0.453 0.453 0.311 0.311 0.311 0.311</td>
<td>0.380 0.248 0.145 0.453 0.453 0.453 0.311 0.311 0.311 0.311</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(-6,1,3,0.01)</td>
<td>0.527 0.311 0.145 0.601 0.601 0.601 0.380 0.380 0.380 0.380</td>
<td>0.527 0.311 0.145 0.601 0.601 0.601 0.380 0.380 0.380 0.380</td>
<td>0.527 0.311 0.145 0.601 0.601 0.601 0.380 0.380 0.380 0.380</td>
<td>0.527 0.311 0.145 0.601 0.601 0.601 0.380 0.380 0.380 0.380</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(-6,2,1,0.01)</td>
<td>0.248 0.193 0.145 0.380 0.380 0.380 0.311 0.311 0.311 0.311</td>
<td>0.248 0.193 0.145 0.380 0.380 0.380 0.311 0.311 0.311 0.311</td>
<td>0.248 0.193 0.145 0.380 0.380 0.380 0.311 0.311 0.311 0.311</td>
<td>0.248 0.193 0.145 0.380 0.380 0.380 0.311 0.311 0.311 0.311</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(-6,2,2,0.01)</td>
<td>0.380 0.248 0.145 0.527 0.527 0.527 0.380 0.380 0.380 0.380</td>
<td>0.380 0.248 0.145 0.527 0.527 0.527 0.380 0.380 0.380 0.380</td>
<td>0.380 0.248 0.145 0.527 0.527 0.527 0.380 0.380 0.380 0.380</td>
<td>0.380 0.248 0.145 0.527 0.527 0.527 0.380 0.380 0.380 0.380</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(-6,2,3,0.01)</td>
<td>0.527 0.311 0.145 0.671 0.671 0.671 0.453 0.453 0.453 0.453</td>
<td>0.527 0.311 0.145 0.671 0.671 0.671 0.453 0.453 0.453 0.453</td>
<td>0.527 0.311 0.145 0.671 0.671 0.671 0.453 0.453 0.453 0.453</td>
<td>0.527 0.311 0.145 0.671 0.671 0.671 0.453 0.453 0.453 0.453</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(-6,3,1,0.01)</td>
<td>0.248 0.193 0.145 0.453 0.453 0.453 0.380 0.380 0.380 0.380</td>
<td>0.248 0.193 0.145 0.453 0.453 0.453 0.380 0.380 0.380 0.380</td>
<td>0.248 0.193 0.145 0.453 0.453 0.453 0.380 0.380 0.380 0.380</td>
<td>0.248 0.193 0.145 0.453 0.453 0.453 0.380 0.380 0.380 0.380</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(-6,3,2,0.01)</td>
<td>0.380 0.248 0.145 0.601 0.601 0.601 0.453 0.453 0.453 0.453</td>
<td>0.380 0.248 0.145 0.601 0.601 0.601 0.453 0.453 0.453 0.453</td>
<td>0.380 0.248 0.145 0.601 0.601 0.601 0.453 0.453 0.453 0.453</td>
<td>0.380 0.248 0.145 0.601 0.601 0.601 0.453 0.453 0.453 0.453</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(-6,3,3,0.01)</td>
<td>0.527 0.311 0.145 0.736 0.736 0.736 0.527 0.527 0.527 0.527</td>
<td>0.527 0.311 0.145 0.736 0.736 0.736 0.527 0.527 0.527 0.527</td>
<td>0.527 0.311 0.145 0.736 0.736 0.736 0.527 0.527 0.527 0.527</td>
<td>0.527 0.311 0.145 0.736 0.736 0.736 0.527 0.527 0.527 0.527</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Appendix C

Simulation Programs

C.1 S Code to Generate Nuclear Pedigrees

```r
pedsim<-function(n,f1,f2,p,o,effsd,sig,b0,b1,b2,b3,b12,b2,b3,b4,h4)
#
# This program simulates nuclear pedigrees along with a set of covariates
# for each individual. The data will be generated using a logit-normal
# model with error that is distributed MVN. The outcome of interest is
# affection status.
# The covariates of interest include two candidate genes and an
# environmental factor.
# EF will be a continuous variable. Each family will be randomly assigned an
# EF family mean (randomly generated from a standard normal distn).
# Each family member will then be randomly assigned a EF value
# according to a normal distribution with the family EF mean and with
# a smaller variance determined by the user.
# The genetic model assumed here will be a multifactorial inheritance
# model (assuming Hardy-Weinberg equilibrium) with dominant and
# additive effects.
# Both candidate genes are diallelic (two alleles). It will be assumed
# that each gene are on separate chromosomes and thus are segregated
# independently.
# The genotypes for the offspring will be generated
# using transmission parameters given the parental mating type.
# For the parental genotypes, a simple random number generator will be used
# where a particular allele will be assigned if it falls below or above
```
the population frequency for allele 1 (disease allele).

The simulation will involve a two-step process: 1) First the parent's data will be generated and 2) the offspring data will be generated given the parental data where the number of sibs follows a sibship distribution.

n represents the number of pedigrees i.e. no. of sets of founders

b's = fixed slopes of genotype covariates and ef

f1 and f2 = (population) frequencies of alleles 1 and 2 respectively

```r
# generating parental data
for(i in 1:n) {
# candidate gene 1
fm1.all[i, ] <- c(runif(1, 0, 1), runif(1, 0, 1))
ff1.all[i, ] <- c(runif(1, 0, 1), runif(1, 0, 1))
if(fm1.all[i, 1] < f1)
  fm1.all[i, 1] <- 1
else fm1.all[i, 1] <- 2
if(fm1.all[i, 2] < f1)
  fm1.all[i, 2] <- 1
else fm1.all[i, 2] <- 2
if(ff1.all[i, 1] < f1)
  ff1.all[i, 1] <- 1
else ff1.all[i, 1] <- 2
if(ff1.all[i, 2] < f1)
  ff1.all[i, 2] <- 1
else ff1.all[i, 2] <- 2
# candidate gene 2
fm2.all[i, ] <- c(runif(1, 0, 1), runif(1, 0, 1))
ff2.all[i, ] <- c(runif(1, 0, 1), runif(1, 0, 1))
if(fm2.all[i, 1] < f2)
  fm2.all[i, 1] <- 3
else fm2.all[i, 1] <- 4
if(fm2.all[i, 2] < f2)
  fm2.all[i, 2] <- 3
```
else fm2.all[i, 2] <- 4
if(ff2.all[i, 1] < f2)
    ff2.all[i, 1] <- 3
else ff2.all[i, 1] <- 4
if(ff2.all[i, 2] < f2)
    ff2.all[i, 2] <- 3
else ff2.all[i, 2] <- 4

no.sibs[i, ] <- rgeom(i, 0.4651) + 1
fam.id[i, ] <- 1
fam <- cbind(fam.id, fm1.all, fm2.all, ff1.all, ff2.all, no.sibs)
fam <- round(fam, digits = 1)
dimnames(fam) <- list(NULL, c("Fam ID", "FM1(allel1)",
                             "FM1(allel12)", "FM2(allel1)", "FM2(allel12)",
                             "FP1(allel1)", "FP1(allel12)", "FP2(allel1)",
                             "FP2(allel12)", "No. sibs")
}

# generating covariate data for each family

fam.size <- no.sibs + 2
family <- NULL
for(i in 1:n) {
    size <- fam.size[i, ]
    for(j in 1:size) {
        fam.par <- matrix(0, 2, 12)
ef.mean <- rnorm(1, 133, 5)
        if(ff1.all[i, 1] == 1 && ff1.all[i, 2] == 1)
            geno1.f <- h1
        if(ff1.all[i, 1] == 1 && ff1.all[i, 2] == 2)
            geno1.f <- h12
        if(ff1.all[i, 1] == 2 && ff1.all[i, 2] == 1)
            geno1.f <- h12
        if(ff1.all[i, 1] == 2 && ff1.all[i, 2] == 2)
            geno1.f <- h2
        if(fm1.all[i, 1] == 1 && fm1.all[i, 2] == 1)
            geno1.m <- h1
        if(fm1.all[i, 1] == 1 && fm1.all[i, 2] == 2)
            geno1.m <- h12
        if(fm1.all[i, 1] == 2 && fm1.all[i, 2] == 1)
            geno1.m <- h1
        if(fm1.all[i, 1] == 2 && fm1.all[i, 2] == 2)
            geno1.m <- h12
    }
}
geno1.m <- h12
if(fml.all[i, 1] == 2 & fml.all[i, 2] == 2)
geno1.m <- h2

if(fml.all[i, 1] == 3 & fml.all[i, 2] == 3)
geno2.f <- h3
if(fml.all[i, 1] == 3 & fml.all[i, 2] == 4)
geno2.f <- h4
if(fml.all[i, 1] == 4 & fml.all[i, 2] == 3)
geno2.f <- h3
if(fml.all[i, 1] == 4 & fml.all[i, 2] == 4)
geno2.f <- h4

# father data
fam.par[1, ] <- c(fam.id[i, ], fam.id[i, ] + 1/100, 1,
                   fml.all[i, ], fml.all[i, ],
                   rnorm(1, ef.mean, effsd), ef.mean,
                   geno1.f, geno2.f)

# mother data
fam.par[2, ] <- c(fam.id[i, ], fam.id[i, ] + 2/100, 1,
                   fml.all[i, ], fml.all[i, ],
                   rnorm(1, ef.mean, 5.3), ef.mean,
                   geno1.m, geno2.m)

# sib data
sib.data <- NULL
for(k in 1:no.sibs[i, ]){
    family.id <- fam.id[i, ]
    sib.id <- fam.id[i, ] + (k + 2)/100
    sib.sex <- rbinom(1, 1, 0.5)
u <- runif(1, 0, 1)
# the offspring genotype is dependent on the parental genotypes
# and the transmission probabilities that depend on the genetic model

# parental genotype: 1 1 X 1 1
if(fam.par[1, 5] == 1 && fam.par[1, 6] == 1 &&
  fam.par[2, 5] == 1 && fam.par[2, 6] == 1) {
  sib.gen01 <- c(1, 1)
}

# parental genotype: 2 2 X 2 2
if(fam.par[1, 5] == 2 && fam.par[1, 6] == 2 &&
  fam.par[2, 5] == 2 && fam.par[2, 6] == 2) {
  sib.gen01 <- c(2, 2)
}
if(u < 0.5) {
 # parental genotype: (1 1 X 1 2) (1 1 X 2 1) (1 2 X 1 1) (2 1 X 1 1)
if(fam.par[1, 5] == 1 && fam.par[1, 6] == 1 &&
  fam.par[2, 5] == 1 && fam.par[2, 6] == 2) {
  sib.gen01 <- c(1, 1)
}
if(fam.par[1, 5] == 1 && fam.par[1, 6] == 1 &&
  fam.par[2, 5] == 2 && fam.par[2, 6] == 1) {
  sib.gen01 <- c(1, 1)
}
if(fam.par[1, 5] == 1 && fam.par[1, 6] == 2 &&
  fam.par[2, 5] == 1 && fam.par[2, 6] == 1) {
  sib.gen01 <- c(1, 1)
}
if(fam.par[1, 5] == 2 && fam.par[1, 6] == 1 &&
  fam.par[2, 5] == 1 && fam.par[2, 6] == 1) {
  sib.gen01 <- c(1, 1)
}
 # parental genotype: (1 2 X 2 2) (2 1 X 2 2) (2 2 X 2 1) (2 2 X 1 2)
if(fam.par[1, 5] == 1 && fam.par[1, 6] == 2 &&
  fam.par[2, 5] == 2 && fam.par[2, 6] == 2) {

sib.gen01 <- c(1, 2)
}
  {sib.gen01 <- c(1, 2)}
  {sib.gen01 <- c(1, 2)}
  {sib.gen01 <- c(1, 2)}
if(u > 0.5) {
  # parental genotype: (1 1 X 1 2) (1 1 X 2 1) (1 2 X 1 1) (2 1 X 1 1)
    {sib.gen01 <- c(1, 2)}
    {sib.gen01 <- c(1, 2)}
    {sib.gen01 <- c(1, 2)}
    {sib.gen01 <- c(1, 2)}
}
# parental genotype: (1 2 X 2 2) (2 1 X 2 2) (2 2 X 2 1) (2 2 X 1 2)

```r
if(fam.par[1, 6] == 1 && fam.par[1, 6] == 2 &&
    fam.par[2, 6] == 2 && fam.par[2, 6] == 2)
{
    sib.gen01 <- c(2, 2)
}
```

if(fam.par[1, 6] == 2 && fam.par[1, 6] == 1 &&
    fam.par[2, 6] == 2 && fam.par[2, 6] == 2)
{
    sib.gen01 <- c(2, 2)
}

if(fam.par[1, 6] == 2 && fam.par[1, 6] == 2 &&
{
    sib.gen01 <- c(2, 2)
}

if(fam.par[1, 6] == 2 && fam.par[1, 6] == 2 &&
    fam.par[2, 6] == 1 && fam.par[2, 6] == 2)
{
    sib.gen01 <- c(2, 2)
}

if(fam.par[1, 6] == 2 && fam.par[1, 6] == 1 &&
    fam.par[2, 6] == 1 && fam.par[2, 6] == 2)
{
    sib.gen01 <- c(1, 2)
}

if(fam.par[1, 6] == 2 && fam.par[1, 6] == 2 &&
{
    sib.gen01 <- c(1, 2)
}

if(u < 0.25) {

# parental genotype: (1 1 X 2 2)

```r
if(fam.par[1, 6] == 1 && fam.par[1, 6] == 1 &&
    fam.par[2, 6] == 2 && fam.par[2, 6] == 2) {
    sib.gen01 <- c(1, 2)
}
```
}

if(fam.par[1, 6] == 2 && fam.par[1, 6] == 1 &&
{
    sib.gen01 <- c(1, 2)
}

if(fam.par[1, 6] == 1 && fam.par[1, 6] == 1 &&
{
    sib.gen01 <- c(1, 1)
}

```
if(fam.par[1, 6] == 1 && fam.par[1, 6] == 2 &&
```
{ 
  sib.genol <- c(1, 1)
}

if(fam.par[1, 5] == 2 && fam.par[1, 6] == 1 &&
   fam.par[2, 5] == 1 && fam.par[2, 6] == 2)
{
  sib.genol <- c(1, 1)
}

if(fam.par[1, 5] == 2 && fam.par[1, 6] == 1 &&
   fam.par[2, 5] == 2 && fam.par[2, 6] == 1)
{
  sib.genol <- c(1, 1)
}

if(u < 0.75 && u >= 0.25) {
  if(fam.par[1, 5] == 1 && fam.par[1, 6] == 2 &&
    fam.par[2, 5] == 1 && fam.par[2, 6] == 2)
    { sib.genol <- c(1, 2) }
  if(fam.par[1, 5] == 1 && fam.par[1, 6] == 2 &&
      fam.par[2, 5] == 2 && fam.par[2, 6] == 1)
    { sib.genol <- c(1, 2) }
  if(fam.par[1, 5] == 2 && fam.par[1, 6] == 1 &&
      fam.par[2, 5] == 1 && fam.par[2, 6] == 2)
    { sib.genol <- c(1, 2) }
  if(fam.par[1, 5] == 2 && fam.par[1, 6] == 1 &&
      fam.par[2, 5] == 2 && fam.par[2, 6] == 1)
    { sib.genol <- c(1, 2) }
}

if(u >= 0.75) {
  if(fam.par[1, 5] == 1 && fam.par[1, 6] == 2 &&
    fam.par[2, 5] == 1 && fam.par[2, 6] == 2)
    { }
sib.gen1 <- c(2, 1)
}

if(fam.par[1, 5] == 1 && fam.par[1, 6] == 2 &&
    fam.par[2, 5] == 2 && fam.par[2, 6] == 1)
{
    sib.gen1 <- c(2, 2)
}

if(fam.par[1, 5] == 2 && fam.par[1, 6] == 1 &&
    fam.par[2, 5] == 1 && fam.par[2, 6] == 2)
{
    sib.gen1 <- c(2, 2)
}

if(fam.par[1, 6] == 2 && fam.par[1, 6] == 1 &&
{
    sib.gen1 <- c(2, 2)
}

# parental genotype : 3 3 X 3 3
if(fam.par[1, 7] == 3 && fam.par[1, 8] == 3 &&
    fam.par[2, 7] == 3 && fam.par[2, 8] == 3)
{
    sib.gen2 <- c(3, 3)
}

# parental genotype: 4 4 X 4 4
if(fam.par[1, 7] == 4 && fam.par[1, 8] == 4 &&
    fam.par[2, 7] == 4 && fam.par[2, 8] == 4)
{
    sib.gen2 <- c(4, 4)
}

if(u < 0.5)
{
    # parental genotype: (3 3 X 3 4) (3 3 X 4 3) (3 4 X 3 3) (4 3 X 3 3)
    if(fam.par[1, 7] == 3 && fam.par[1, 8] == 3 &&
        fam.par[2, 7] == 3 && fam.par[2, 8] == 4)
    {
        sib.gen2 <- c(3, 3)
    }

    # parental genotype: (3 3 X 3 4) (3 3 X 4 3) (3 4 X 3 3) (4 3 X 3 3)
    if(fam.par[1, 7] == 3 && fam.par[1, 8] == 3 &&
        fam.par[2, 7] == 4 && fam.par[2, 8] == 3)
    {
        sib.gen2 <- c(3, 3)
    }
if(fam.parC1, 71 = 3 && fam.par[1, 8] = 4 &&
fam.par[2, 7] = 3 && fam.par[2, 8] = 3)
{
    sib.geno2 <- c(3, 3)
}
if(fam.par[1, 7] = 4 && fam.par[1, 8] = 3 &&
fam.par[2, 7] = 3 && fam.par[2, 8] = 3)
{
    sib.geno2 <- c(3, 3)
}
# parental genotype: (3 4 X 4 4) (4 3 X 4 4) (4 4 X 4 3) (4 4 X 3 4)
if(fam.par[1, 7] = 3 && fam.par[1, 8] = 4 &&
fam.par[2, 7] = 4 && fam.par[2, 8] = 3)
{
    sib.geno2 <- c(3, 4)
}
if(fam.par[1, 7] = 4 && fam.par[1, 8] = 3 &&
fam.par[2, 7] = 4 && fam.par[2, 8] = 4)
{
    sib.geno2 <- c(3, 4)
}
if(fam.par[1, 7] = 4 && fam.par[1, 8] = 4 &&
fam.par[2, 7] = 4 && fam.par[2, 8] = 3)
{
    sib.geno2 <- c(3, 4)
}
if(fam.par[1, 7] = 4 && fam.par[1, 8] = 4 &&
fam.par[2, 7] = 3 && fam.par[2, 8] = 4)
{
    sib.geno2 <- c(3, 4)
}

if(u > 0.5) {
# parental genotype: (3 3 X 3 4) (3 3 X 4 3) (3 4 X 3 3) (4 3 X 3 3)
if(fam.par[1, 7] = 3 && fam.par[1, 8] = 3 &&
fam.par[2, 7] = 3 && fam.par[2, 8] = 4)
{
    sib.geno2 <- c(3, 4)
}
if(fam.par[1, 7] == 3 && fam.par[1, 8] == 3 &&
    fam.par[2, 7] == 4 && fam.par[2, 8] == 3)
{
    sib.geno2 <- c(3, 4)
}
if(fam.par[1, 7] == 3 && fam.par[1, 8] == 4 &&
    fam.par[2, 7] == 3 && fam.par[2, 8] == 3)
{
    sib.geno2 <- c(3, 4)
}
if(fam.par[1, 7] == 4 && fam.par[1, 8] == 3 &&
    fam.par[2, 7] == 3 && fam.par[2, 8] == 3)
{
    sib.geno2 <- c(3, 4)
}
# parental genotype: (3 4 4 4) (4 3 4 4) (4 4 3 4) (4 4 X 4 3)
if(fam.par[1, 7] == 3 && fam.par[1, 8] == 4 &&
    fam.par[2, 7] == 4 && fam.par[2, 8] == 4)
{
    sib.geno2 <- c(4, 4)
}
if(fam.par[1, 7] == 4 && fam.par[1, 8] == 3 &&
    fam.par[2, 7] == 4 && fam.par[2, 8] == 4)
{
    sib.geno2 <- c(4, 4)
}
if(fam.par[1, 7] == 4 && fam.par[1, 8] == 4 &&
    fam.par[2, 7] == 4 && fam.par[2, 8] == 3)
{
    sib.geno2 <- c(4, 4)
}
if(fam.par[1, 7] == 4 && fam.par[1, 8] == 4 &&
    fam.par[2, 7] == 3 && fam.par[2, 8] == 4)
{
    sib.geno2 <- c(4, 4)
}
if(fam.par[1, 7] == 4 && fam.par[1, 8] == 4 &&
    fam.par[2, 7] == 3 && fam.par[2, 8] == 4)
{
    sib.geno2 <- c(4, 4)
}
# parental genotype: 3 3 X 4 4
if(fam.par[1, 7] == 3 && fam.par[1, 8] == 3 &&
    fam.par[2, 7] == 4 && fam.par[2, 8] == 4) {
sib.geno2 <- c(3, 4)

if(fam.par[1, 7] == 4 && fam.par[1, 8] == 4 &&
   fam.par[2, 7] == 3 && fam.par[2, 8] == 3)
  sib.geno2 <- c(3, 4)

if(u < 0.25) {
  # parental genotype: (3 4 X 3 4) (3 4 X 4 3) (4 3 X 3 4) (4 3 X 4 3)
  if(fam.par[1, 7] == 3 && fam.par[1, 8] == 4 &&
      fam.par[2, 7] == 3 && fam.par[2, 8] == 4)
    sib.geno2 <- c(3, 3)
  }

  if(fam.par[1, 7] == 3 && fam.par[1, 8] == 4 &&
      fam.par[2, 7] == 4 && fam.par[2, 8] == 3)
    sib.geno2 <- c(3, 3)
  }

  if(fam.par[1, 7] == 4 && fam.par[1, 8] == 3 &&
      fam.par[2, 7] == 3 && fam.par[2, 8] == 4)
    sib.geno2 <- c(3, 3)
  }

  if(fam.par[1, 7] == 4 && fam.par[1, 8] == 3 &&
      fam.par[2, 7] == 4 && fam.par[2, 8] == 3)
    sib.geno2 <- c(3, 3)
  }

  if(u < 0.75 && u >= 0.25) {
    if(fam.par[1, 7] == 3 && fam.par[1, 8] == 4 &&
        fam.par[2, 7] == 3 && fam.par[2, 8] == 4)
      sib.geno2 <- c(3, 4)
    }

if(fam.par[1, 7] == 3 && fam.par[1, 8] == 4 &&
   fam.par[2, 7] == 4 && fam.par[2, 8] == 3)
  sib.geno2 <- c(3, 4)
if(fam.par[1, 7] == 4 && fam.par[1, 8] == 3 &&
fam.par[2, 7] == 3 && fam.par[2, 8] == 4)
{
  sib.geno2 <- c(3, 4)
}
if(fam.par[1, 7] == 4 && fam.par[1, 8] == 3 &&
fam.par[2, 7] == 4 && fam.par[2, 8] == 3)
{
  sib.geno2 <- c(3, 4)
}
if(u >= 0.75) {
  if(fam.par[1, 7] == 3 && fam.par[1, 8] == 4 &&
fam.par[2, 7] == 3 && fam.par[2, 8] == 4)
  {
    sib.geno2 <- c(4, 4)
  }
  if(fam.par[1, 7] == 4 && fam.par[1, 8] == 4 &&
fam.par[2, 7] == 4 && fam.par[2, 8] == 3)
  {
    sib.geno2 <- c(4, 4)
  }
  if(fam.par[1, 7] == 4 && fam.par[1, 8] == 3 &&
fam.par[2, 7] == 3 && fam.par[2, 8] == 4)
  {
    sib.geno2 <- c(4, 4)
  }
  if(fam.par[1, 7] == 4 && fam.par[1, 8] == 3 &&
fam.par[2, 7] == 4 && fam.par[2, 8] == 3)
  {
    sib.geno2 <- c(4, 4)
  }
}
sib.ef <- rmnorm(1, ef.mean, effsd)
genoi <- h1
genoi <- h12
genoi <- h12
geno1 <- h2

geno2 <- h3
geno2 <- h34
geno2 <- h34
geno2 <- h4

sib <- c(family.id, sib.id, 2, sib.sex,
         sib.geno1, sib.geno2, sib.ef, ef.mean,
         geno1, geno2)
sib.data <- rbind(sib.data, sib)
}

# closing k iteration
fam.one <- rbind(fam.par, sib.data)

# We now begin generating probabilites of being affected for each
# member in the dataset. The model assumed in the simulation will
# be the logit-normal model.
# Correlation within each family will be handled in two ways:
# 1) within the generation of covariate values
# 2) The second type of correlation can be considered a
# residual type of correlation or a familial type of correlation
# where a random error component will
# be attached to the logit-normal model. This residual follows a multivariate
# normal distribution with mean 0 and a variance-covariance matrix
# that is family-specific i.e. each element of the matrix would
# correspond to the degree of correlation between family members.
# Therefore the size of the matrix would depend on the number of
# family members in each pedigree. There are only three types of
# relationships: spousal, parent-offspring, sibs.
# Different combinations of parameters will be examined.
# The coefficients will be set at:
# B0=6 (intercept)  B1=C1  B2=C2  B3=EF
# Generating the variance-covariance matrix: the correlation between
# relative-types will be held constant across families, i.e.
# 1) Spouse-spouse = p
# 2) Parent-offspring = po
# 3) Siblings = o

# g.x represents the explained linear portion of the logistic model

r <- matrix(0, size, size)
r[1:size, 1] <- rep(0, size)
r[, 1:2] <- po
r[1:2, 1] <- po

# creating a covariance matrix with initially starting values of all o.

diag(r) <- rep(1, size)

# placing 1's along diagonal of covariance
# matrix indicating perfect correlation between
# father-father, mother-mother etc.

r[1, 2] <- p
r[2, 1] <- p

v <- sigx
mu <- rep(0, size) # both are spousal correlation
e.c <- rmultnorm(1, mu, v) # 1 correlated 'size' vector
beta <- rbind(b0, b1, b2, b3) # beta=(B0, B1, B2, B3)'
x0 <- rep(1, size) # design column
x1 <- fam.one[, 11] # genotype1 categorization
x2 <- fam.one[, 12] # genotype2 categorization
x3 <- fam.one[, 9] # environmental factor
x.data <- cbind(x0, x1, x2, x3)
g.resid <- x.data %*% beta
p.resid <- exp(g.resid)/(1+exp(g.resid))
g.x <- x.data %*% x + t(e.c) # x.data is a size X size matrix
p.x <- exp(g.x)/(1 + exp(g.x)) # beta is a size X 1 matrix
aff <- matrix(0, size, 1)
for (l in 1:size){ # assigning affection status
  raff <- runif(1, 0, 1)
  if (raff<p.x[l]) aff[l,] <- 1 else aff[l,] <- 0
}
resid <- aff-p.resid
fam.one <- cbind(fam.one, g.resid, p.resid, resid, g.x, p.x, aff)

# closing j (for each family member) iteration
family <- rbind(family, fam.one)

# closing the i (for each family) iteration

C.2 S Code for the Hierarchical Bootstrap

bootped<-function(nboot, k0, k1, k2, k3, simglmest, r)
{
    no.fam <- length(unique(sim[, 1]))
    nuc.boot.glm.s1 <- matrix(0, nboot, 12)
    nuc.boot.glm.s2 <- matrix(0, nboot, 12)
    nuc.boot.n.s1 <- matrix(0, nboot, 1)
    nuc.boot.n.s2 <- matrix(0, nboot, 1)
    sim.n <- length(sim[, 1])
    kull.lieb <- cbind(rep(k0, nboot), rep(k1, nboot), rep(k2, nboot),
                      rep(k3, nboot))
    glm.est<-cbind(rep(simglmest[1], nboot), rep(simglmest[2], nboot),
                    rep(simglmest[3], nboot), rep(simglmest[4], nboot))
    sum.nuc.boot.glm.s1.b0<-matrix(0,nboot,12)
    sum.nuc.boot.glm.s1.b1<-matrix(0,nboot,12)
    sum.nuc.boot.glm.s1.b2<-matrix(0,nboot,12)
    sum.nuc.boot.glm.s1.b3<-matrix(0,nboot,12)
    sum.nuc.boot.glm.s2.b0<-matrix(0,nboot,12)
    sum.nuc.boot.glm.s2.b1<-matrix(0,nboot,12)
    sum.nuc.boot.glm.s2.b2<-matrix(0,nboot,12)
    sum.nuc.boot.glm.s2.b3<-matrix(0,nboot,12)

    # kullback-liebler result to be used for coverage probability
    # and boot-t
    # r.se to be used for boot-t
    boot.s1.m <- matrix(rep(sim[, 1], no.fam), ncol = no.fam)
for(i in 1:nboot) {
    # bootstrapping families
    nuc.boot.si.m <- sample(1:no.fam, replace = TRUE)
    boot.si.m2 <- t(matrix(rep(nuc.boot.si.m, nrow(boot.si.m)),
        ncol = nrow(boot.si.m)))
    match.fam <- (boot.si.m == boot.si.m2)
    locations <- match.fam * (1:nrow(boot.si.m))
    locations.v <- matrix(locations, ncol = 1)
    nuc.boot.si.sam <- sim[locations.v, ]
    nuc.boot.n.si[i, ] <- length(nuc.boot.si.sam[, 1])
    x0 <- rep(1, length(nuc.boot.si.sam[, 1]))
    x <- cbind(x0, nuc.boot.si.sam[, 11], nuc.boot.si.sam[, 12],
        nuc.boot.si.sam[, 9])
    mle <- x %*% simglmest
    nuc.boot.glm <- summary(glm(nuc.boot.si.sam[, 18] ~
        nuc.boot.si.sam[, 11] + nuc.boot.si.sam[, 12] +
        nuc.boot.si.sam[, 9], family = binomial, start = mle,
        maxit = 1))
    nuc.boot.glm.si[i, ] <- cbind(t(nuc.boot.glm$coef[, 1]), t(1
        nuc.boot.glm$coef[, 2]), t(nuc.boot.glm$coef[, 3]))

    # columns 1-4 are estimates b0-b3
    # columns 5-8 are se of est b0-b3
    # columns 9-12 are z-scores for b0-b3
    # sampling with replacement for offsprings
    no.fam.boot <- tapply(sort(nuc.boot.si.m), sort(nuc.boot.si.m),
        length) # frequency of families bootstrapped
    no.sibs.boot <- ((tapply(nuc.boot.si.sam[, 2], nuc.boot.si.sam[, 1], max) -
        unique(sort(nuc.boot.si.sam[, 1]))) * 100) - 2
    all.fam.boot <- rep(no.sibs.boot, no.fam.boot)
    all.fam.boot2 <- as.single(all.fam.boot)
    all.fam.boot3 <- as.matrix(all.fam.boot2)
    sibs.rand <- unlist(apply(all.fam.boot3, 1, sample, replace = T
        )) + 2
    sibs.rand.fam.id <- sibs.rand/100 + sort(nuc.boot.si.sam[, 1][
        nuc.boot.si.sam[, 3] == 2])
    sibs.boot.match <- match(sibs.rand.fam.id, nuc.boot.si.sam[, 2]
        )
    sibs.boot.sam <- nuc.boot.si.sam[sibs.boot.match, ]
par.boot <- nuc.boot.sl.sam[nuc.boot.sl.sam[, 3] == 1, ]
nuc.boot.s2r.sam <- rbind(par.boot, sibs.boot.sam)
locat.r <- sort(match(nuc.boot.s2r.sam[, 2], nuc.boot.sl.sam[, 2]))
nuc.boot.s2.sam <- nuc.boot.sl.sam[locat.r, ]
nuc.boot.n.s2[i, ] <- length(nuc.boot.s2.sam[, 1])
x0.r <- rep(1, length(nuc.boot.s2.sam[, 1]))
x.r <- cbind(x0.r, nuc.boot.s2.sam[, 11], nuc.boot.s2.sam[, 12], nuc.boot.s2.sam[, 9])
ml.r <- x.r %*% simglm
nuc.boot.glm.s2[i, ] <- cbind(t(nuc.boot.glm.r$coef[, 1]), t(nuc.boot.glm.r$coef[, 2]), t(nuc.boot.glm$coef[, 3]))

# closing i
# There are two types of bootstrap data sets.
# The first is nuc.boot.sl.sam which is strategy 1
# and where all the bootstrap data results is stored in nuc.boot.glm.sl
# for each replicate
# The second is nuc.boot.s2.sam which is strategy 2
# and where all the bootstrap data results is stored in
# nuc.boot.glm.s2 for each replicate

# Summary Statistics #
int.boot <- mean(nuc.boot.glm.sl[, 1])
# bootstrap coefficient estimates
b1.boot <- mean(nuc.boot.glm.sl[, 2])  # from B bootstrap samples
b2.boot <- mean(nuc.boot.glm.sl[, 3])
b3.boot <- mean(nuc.boot.glm.sl[, 4])
int.boot.adj <- mean((nuc.boot.n.s1/sim.n) * nuc.boot.glm.sl[, 1])
b1.boot.adj <- mean((nuc.boot.n.s1/sim.n) * nuc.boot.glm.sl[, 2])
b2.boot.adj <- mean((nuc.boot.n.s1/sim.n) * nuc.boot.glm.sl[, 3])
b3.boot.adj <- mean((nuc.boot.n.s1/sim.n) * nuc.boot.glm.sl[, 4])
se.int.boot <- sqrt(var(nuc.boot.glm.sl[, 1]))
# bootstrap estimate of SE
se.b1.boot <- sqrt(var(nuc.boot.glm.s1[, 2]))
se.b2.boot <- sqrt(var(nuc.boot.glm.s1[, 3]))
se.b3.boot <- sqrt(var(nuc.boot.glm.s1[, 4]))
se.int.boot.adj <- sqrt(var((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s1[, 1]))
se.b1.boot.adj <- sqrt(var((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s1[, 2]))
se.b2.boot.adj <- sqrt(var((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s1[, 3]))
se.b3.boot.adj <- sqrt(var((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s1[, 4]))

boot.mean.s1 <- c(int.boot, b1.boot, b2.boot, b3.boot, int.boot.adj, b1.boot.adj, b2.boot.adj, b3.boot.adj, se.int.boot, se.b1.boot, se.b2.boot, se.b3.boot, se.int.boot.adj, se.b1.boot.adj, se.b2.boot.adj, se.b3.boot.adj)

# for second method ie random sample for each bootstrap family
int.boot.r <- mean(nuc.boot.glm.s2[, 1])

# bootstrap coefficient estimates
b1.boot.r <- mean(nuc.boot.glm.s2[, 2]) # from B bootstrap samples
b2.boot.r <- mean(nuc.boot.glm.s2[, 3])
b3.boot.r <- mean(nuc.boot.glm.s2[, 4])
int.boot.r.adj <- mean((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s2[, 1])

# adjusted bootstrap coefficient estimates
b1.boot.r.adj <- mean((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s2[, 2])
# from B bootstrap samples
b2.boot.r.adj <- mean((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s2[, 3])
b3.boot.r.adj <- mean((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s2[, 4])

se.int.boot.r <- sqrt(var(nuc.boot.glm.s2[, 1]))

# bootstrap estimate of SE
se.b1.boot.r <- sqrt(var(nuc.boot.glm.s2[, 2]))
se.b2.boot.r <- sqrt(var(nuc.boot.glm.s2[, 3]))
se.b3.boot.r <- sqrt(var(nuc.boot.glm.s2[, 4]))
se.int.boot.r.adj <- sqrt(var((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s2[, 1]))

# adjusted bootstrap estimate of SE
se.b1.boot.r.adj <- sqrt(var((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s2[, 2]))
se.b2.boot.r.adj <- sqrt(var((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s2[, 3]))
se.b3.boot.r.adj <- sqrt(var((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s2[, 4]))
43)

```r
boot.mean.s2 <- c(int.boot.r, b1.boot.r, b2.boot.r, b3.boot.r,
       int.boot.r.adj, b1.boot.r.adj, b2.boot.r.adj, b3.boot.r(adj,
       se.int.boot.r, se.b1.boot.r, se.b2.boot.r, se.b3.boot.r,
       se.int.boot.r.adj, se.b1.boot.r.adj, se.b2.boot.r.adj, se.b3.boot.r.adj)
sum.nuc.boot.glm.sl.b0<-sum(apply(slot('nuc.boot', 'glm.sl', 1), summary(slot('nuc.boot', 'glm.sl', 2)), summary(slot('nuc.boot', 'glm.sl', 3)), summary(slot('nuc.boot', 'glm.sl', 4))))

# Constructing the confidence intervals #

# 1) Percentile

# 2) BCa

# 1) Percentile Confidence Intervals
# Strategy 1

percent.s1.ci <- matrix(0, 4, 2)
p.s1.b0 <- sort(slot('nuc.boot', 'glm.sl', 1))
p.s1.b1 <- sort(slot('nuc.boot', 'glm.sl', 2))
p.s1.b2 <- sort(slot('nuc.boot', 'glm.sl', 3))
p.s1.b3 <- sort(slot('nuc.boot', 'glm.sl', 4))
percent.s1.ci[, 1] <- c(p.s1.b0[25], p.s1.b0[75])
# 95% percentile ci for b0
percent.s1.ci[, 2] <- c(p.s1.b1[25], p.s1.b1[75])
# 95% percentile ci for b1
percent.s1.ci[, 3] <- c(p.s1.b2[25], p.s1.b2[75])
# 95% percentile ci for b2
percent.s1.ci[, 4] <- c(p.s1.b3[25], p.s1.b3[75])
# 95% percentile ci for b3
length.percent.s1.ci <- abs(percent.s1.ci[, 2] - percent.s1.ci[, 1])
percent.s1.ci.all <- c(percent.s1.ci[, 1], percent.s1.ci[, 2],
percent.s1.ci[, 3], percent.s1.ci[, 4], length.percent.s1.ci)
```

# Strategy 2
percent.s2.ci <- matrix(0, 4, 2)
p.s2.b0 <- sort(nuc.boot.glm.s2[, 1])
p.s2.b1 <- sort(nuc.boot.glm.s2[, 2])
p.s2.b2 <- sort(nuc.boot.glm.s2[, 3])
p.s2.b3 <- sort(nuc.boot.glm.s2[, 4])
percent.s2.ci[1, ] <- c(p.s2.b0[25], p.s2.b0[975])
# 95% percentile ci for b0
percent.s2.ci[2, ] <- c(p.s2.b1[25], p.s2.b1[975])
# 95% percentile ci for b1
percent.s2.ci[3, ] <- c(p.s2.b2[25], p.s2.b2[975])
# 95% percentile ci for b2
percent.s2.ci[4, ] <- c(p.s2.b3[25], p.s2.b3[975])
# 95% percentile ci for b3
length.percent.s2.ci <- abs(percent.s2.ci[, 2] - percent.s2.ci[, 1])
percent.s2.ci.all <- c(percent.s2.ci[1, ], percent.s2.ci[2, ],
                        percent.s2.ci[3, ], percent.s2.ci[4, ], length.percent.s2.ci)
if (percent.s1.ci[1, 1] < kull.lieb[, 1][1] && percent.s1.ci[1, 2] >
kull.lieb[, 1][1])
  percent.0.s1.cover <- 1
else percent.0.s1.cover <- 0
if (percent.s1.ci[2, 1] < kull.lieb[, 2][1] && percent.s1.ci[2, 2] >
kull.lieb[, 2][1])
  percent.1.s1.cover <- 1
else percent.1.s1.cover <- 0
if (percent.s1.ci[3, 1] < kull.lieb[, 3][1] && percent.s1.ci[3, 2] >
kull.lieb[, 3][1])
  percent.2.s1.cover <- 1
else percent.2.s1.cover <- 0
if (percent.s1.ci[4, 1] < kull.lieb[, 4][1] && percent.s1.ci[4, 2] >
kull.lieb[, 4][1])
  percent.3.s1.cover <- 1
else percent.3.s1.cover <- 0
if (percent.s2.ci[1, 1] < kull.lieb[, 1][1] && percent.s2.ci[1, 2] >
kull.lieb[, 1][1])
  percent.0.s2.cover <- 1
else percent.0.s2.cover <- 0
if (percent.s2.ci[2, 1] < kull.lieb[, 2][1] && percent.s2.ci[2, 2] >
kull.lieb[, 2][1])
  percent.1.s2.cover <- 1
else percent.1.s2.cover <- 0

if(percent.s2.ci[3, 1] < kull.lieb[, 3][1] & percent.s2.ci[3, 2] >
   kull.lieb[, 3][1])
   percent.s2.cover <- 1
else percent.s2.cover <- 0
if(percent.s2.ci[4, 1] < kull.lieb[, 4][1] & percent.s2.ci[4, 2] >
   kull.lieb[, 4][1])
   percent.s2.cover <- 1
else percent.s2.cover <- 0
percent.s1.coverage <- c(percent.0.s1.cover, percent.1.s1.cover,
   percent.2.s1.cover, percent.3.s1.cover)
percent.s2.coverage <- c(percent.0.s2.cover, percent.1.s2.cover,
   percent.2.s2.cover, percent.3.s2.cover) # ADJUSTED

# Strategy 1
percent.adj.s1.ci <- matrix(0, 4, 2)
p.adj.s1.b0 <- sort((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s1[, 1])
p.adj.s1.b1 <- sort((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s1[, 2])
p.adj.s1.b2 <- sort((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s1[, 3])
p.adj.s1.b3 <- sort((nuc.boot.n.s1/sim.n) * nuc.boot.glm.s1[, 4])
percent.adj.s1.ci[1, ] <- c(p.adj.s1.b0[26], p.adj.s1.b0[975])
   # 95% percentile ci for b0
percent.adj.s1.ci[2, ] <- c(p.adj.s1.b1[26], p.adj.s1.b1[975])
   # 95% percentile ci for b1
percent.adj.s1.ci[3, ] <- c(p.adj.s1.b2[26], p.adj.s1.b2[975])
   # 95% percentile ci for b2
percent.adj.s1.ci[4, ] <- c(p.adj.s1.b3[26], p.adj.s1.b3[975])
   # 95% percentile ci for b3
length.percent.adj.s1.ci <- abs(percent.adj.s1.ci[2] -
   percent.adj.s1.ci[1])
percent.adj.s1.ci.all <- c(percent.adj.s1.ci[1, ], percent.adj.s1.ci[2, ], percent.adj.s1.ci[3, ], percent.adj.s1.ci[4, ],
   length.percent.adj.s1.ci) # Strategy 2
percent.adj.s2.ci <- matrix(0, 4, 2)
p.adj.s2.b0 <- sort((nuc.boot.n.s2/sim.n) * nuc.boot.glm.s2[, 1])
p.adj.s2.b1 <- sort((nuc.boot.n.s2/sim.n) * nuc.boot.glm.s2[, 2])
p.adj.s2.b2 <- sort((nuc.boot.n.s2/sim.n) * nuc.boot.glm.s2[, 3])
p.adj.s2.b3 <- sort((nuc.boot.n.s2/sim.n) * nuc.boot.glm.s2[, 4])
percent.adj.s2.ci[1, ] <- c(p.adj.s2.b0[26], p.adj.s2.b0[975])
   # 95% percentile ci for b0
percent.adj.s2.ci[2, ] <- c(p.adj.s2.b1[26], p.adj.s2.b1[975])
   # 95% percentile ci for b1
percent adj.s2.ci[3, ] <- c(adj.s2.b2[25], adj.s2.b2[975])
# 95% percentile ci for b2
percent adj.s2.ci[4, ] <- c(adj.s2.b3[25], adj.s2.b3[975])
# 95% percentile ci for b3
length percent adj.s2.ci <- abs(percent adj.s2.ci[, 2] -
  percent adj.s2.ci[, 1])
percent adj.s2.ci.all <- c(percent adj.s2.ci[1, 1], percent adj.s2.ci[2, 1], percent adj.s2.ci[3, 1], percent adj.s2.ci[4, 1], length percent adj.s2.ci)
if(percent adj.s1.ci[1, 1] < kull lieb[, 1][1] && percent adj.s1.ci[1, 2] > kull lieb[, 1][1])
  percent 0 adj.s1.cover <- 1
else percent 0 adj.s1.cover <- 0
if(percent adj.s1.ci[2, 1] < kull lieb[, 2][1] && percent adj.s1.ci[2, 2] > kull lieb[, 2][1])
  percent 1 adj.s1.cover <- 1
else percent 1 adj.s1.cover <- 0
if(percent adj.s1.ci[3, 1] < kull lieb[, 3][1] && percent adj.s1.ci[3, 2] > kull lieb[, 3][1])
  percent 2 adj.s1.cover <- 1
else percent 2 adj.s1.cover <- 0
if(percent adj.s1.ci[4, 1] < kull lieb[, 4][1] && percent adj.s1.ci[4, 2] > kull lieb[, 4][1])
  percent 3 adj.s1.cover <- 1
else percent 3 adj.s1.cover <- 0
if(percent adj.s2.ci[1, 1] < kull lieb[, 1][1] && percent adj.s2.ci[1, 2] > kull lieb[, 1][1])
  percent 0 adj.s2.cover <- 1
else percent 0 adj.s2.cover <- 0
if(percent adj.s2.ci[2, 1] < kull lieb[, 2][1] && percent adj.s2.ci[2, 2] > kull lieb[, 2][1])
  percent 1 adj.s2.cover <- 1
else percent 1 adj.s2.cover <- 0
if(percent adj.s2.ci[3, 1] < kull lieb[, 3][1] && percent adj.s2.ci[3, 2] > kull lieb[, 3][1])
  percent 2 adj.s2.cover <- 1
else percent 2 adj.s2.cover <- 0
if(percent adj.s2.ci[4, 1] < kull lieb[, 4][1] && percent adj.s2.ci[4, 2] > kull lieb[, 4][1])
  percent 3 adj.s2.cover <- 1
else percent 3 adj.s2.cover <- 0
else percent.3.adj.s2.cover <- 0
percent.adj.s1.coverage <- c(0, percent.0.adj.s1.coverage, percent.1.adj.s1.coverage, percent.2.adj.s1.coverage, percent.3.adj.s1.coverage)
percent.adj.s2.coverage <- c(0, percent.0.adj.s2.coverage, percent.1.adj.s2.coverage, percent.2.adj.s2.coverage, percent.3.adj.s2.coverage)

# 3) Bias and accelerated confidence intervals.
# the acceleration constant (a) is performed only once for both bootstrap strategies.
jack.glm.est <- matrix(0, no.fam, 4)
# holds jackknife results with the ith row corresponding to the ith family that is deleted.
for(k in 1:no.fam) {
  sim.jack <- sim[, 1] != k, ]
  jack.glm <- summary(glm(sim[, 18] ~ sim[, 11] + sim[, 12] + sim[, 9], family = binomial, maxit = 50))
  jack.glm.est[k, ] <- jack.glm$coeff[, 1]
}
jack.glm.est.mean <- c(mean(jack.glm.est[, 1]), mean(jack.glm.est[, 2]), mean(jack.glm.est[, 3]), mean(jack.glm.est[, 4]))

# a0=acceleration constant for intercept
# a1=acceleration constant for b1
# a2=acceleration constant for b2
# a3=acceleration constant for b3
a0 <- sum((jack.glm.est.mean[1] - jack.glm.est[, 1])^3)/(6 * (sum((jack.glm.est.mean[1] - jack.glm.est[, 1])^2)))^1.5
a1 <- sum((jack.glm.est.mean[2] - jack.glm.est[, 2])^3)/(6 * (sum((jack.glm.est.mean[2] - jack.glm.est[, 2])^2)))^1.5
a <- c(a0, a1, a2, a3)

# bca.sl.ci <- matrix(0, 4, 2)
z0.s1.0 <- qnorm(sum((nuc.boot.glm.s1[, 1] < glm.est[, 1]) * 1)/nboot)
z0.s1.1 <- qnorm(sum((nuc.boot glm.s1[, 2] < glm.est[, 2]) * 1)/nboot
   )
z0.s1.2 <- qnorm(sum((nuc.boot glm.s1[, 3] < glm.est[, 3]) * 1)/nboot
   )
z0.s1.3 <- qnorm(sum((nuc.boot glm.s1[, 4] < glm.est[, 4]) * 1)/nboot
   )
z0.s1 <- c(z0.s1.0, z0.s1.1, z0.s1.2, z0.s1.3)
bcas2.ci <- matrix(0, 4, 2)
z0.s2.0 <- qnorm(sum((nuc.boot glm.s2[, 1] < glm.est[, 1]) * 1)/nboot
   )
z0.s2.1 <- qnorm(sum((nuc.boot glm.s2[, 2] < glm.est[, 2]) * 1)/nboot
   )
z0.s2.2 <- qnorm(sum((nuc.boot glm.s2[, 3] < glm.est[, 3]) * 1)/nboot
   )
z0.s2.3 <- qnorm(sum((nuc.boot glm.s2[, 4] < glm.est[, 4]) * 1)/nboot
   )
z0.s2 <- c(z0.s2.0, z0.s2.1, z0.s2.2, z0.s2.3)
z.025 <- qnorm(0.025)
z.975 <- qnorm(0.975)
alpha1.s1.0 <- pnorm(z0.s1.0 + ((z0.s1.0 + z.025)/(1 - a0 * (z0.s1.0 + z.025))))
alpha2.s1.0 <- pnorm(z0.s1.0 + ((z0.s1.0 + z.975)/(1 - a0 * (z0.s1.0 + z.975))))
bcas1.ci[1, ] <- c(sort(nuc.boot glm.s1[, 1])[ceiling(nboot *
   alpha1.s1.0)], sort(nuc.boot glm.s1[, 1])[ceiling(nboot *
   alpha2.s1.0)]
alpha1.s1.1 <- pnorm(z0.s1.1 + ((z0.s1.1 + z.025)/(1 - a1 * (z0.s1.1 +
   z.025))))
alpha2.s1.1 <- pnorm(z0.s1.1 + ((z0.s1.1 + z.975)/(1 - a1 * (z0.s1.1 +
   z.975))))
bcas1.ci[2, ] <- c(sort(nuc.boot glm.s1[, 2])[ceiling(nboot *
   alpha1.s1.1)], sort(nuc.boot glm.s1[, 2])[ceiling(nboot *
   alpha2.s1.1)]
alpha1.s1.2 <- pnorm(z0.s1.2 + ((z0.s1.2 + z.025)/(1 - a2 * (z0.s1.2 +
   z.025))))
alpha2.s1.2 <- pnorm(z0.s1.2 + ((z0.s1.2 + z.975)/(1 - a2 * (z0.s1.2 +
   z.975))))
bcas1.ci[3, ] <- c(sort(nuc.boot glm.s1[, 3])[ceiling(nboot *
   alpha1.s1.2)], sort(nuc.boot glm.s1[, 3])[ceiling(nboot *
   alpha2.s1.2)]
alpha1.s1.3 <- pnorm(z0.s1.3 + ((z0.s1.3 + z.025)/(1 - a3 * (z0.s1.3 + z.025))))
alpha2.s1.3 <- pnorm(z0.s1.3 + ((z0.s1.3 + z.975)/(1 - a3 * (z0.s1.3 + z.975))))
bcas1.ci[4, ] <- c(sort(nuc.boot.glm.s1[, 4])[ceiling(nboot * 
alpha1.s1.3)], sort(nuc.boot.glm.s1[, 4])[ceiling(nboot * 
alpha2.s1.3)])
alpha.s1 <- c(alpha1.s1.0, alpha2.s1.0, alpha1.s1.1, alpha2.s1.1, 
alpha1.s1.2, alpha2.s1.2, alpha1.s1.3, alpha2.s1.3)
length.bca.s1.ci <- abs(bca.s1.ci[2] - bca.s1.ci[1])
bcas1.ci.all <- c(bcas1.ci[1, ], bcas1.ci[2, ], bcas1.ci[3, ], 
bcas1.ci[4, ], length.bca.s1.ci)
alpha1.s2.0 <- pnorm(z0.s2.0 + ((z0.s2.0 + z.025)/(1 - a0 * (z0.s2.0 + 
z.025))))
alpha2.s2.0 <- pnorm(z0.s2.0 + ((z0.s2.0 + z.975)/(1 - a0 * (z0.s2.0 + 
z.975))))
bcas2.ci[1, ] <- c(sort(nuc.boot.glm.s2[, 1])[ceiling(nboot * 
alpha1.s2.0)], sort(nuc.boot.glm.s2[, 1])[ceiling(nboot * 
alpha2.s2.0)])
alpha1.s2.1 <- pnorm(z0.s2.1 + ((z0.s2.1 + z.025)/(1 - a1 * (z0.s2.1 + 
z.025))))
alpha2.s2.1 <- pnorm(z0.s2.1 + ((z0.s2.1 + z.975)/(1 - a1 * (z0.s2.1 + 
z.975))))
bcas2.ci[2, ] <- c(sort(nuc.boot.glm.s2[, 2])[ceiling(nboot * 
alpha1.s2.1)], sort(nuc.boot.glm.s2[, 2])[ceiling(nboot * 
alpha2.s2.1)])
alpha1.s2.2 <- pnorm(z0.s2.2 + ((z0.s2.2 + z.025)/(1 - a2 * (z0.s2.2 + 
z.025))))
alpha2.s2.2 <- pnorm(z0.s2.2 + ((z0.s2.2 + z.975)/(1 - a2 * (z0.s2.2 + 
z.975))))
bcas2.ci[3, ] <- c(sort(nuc.boot.glm.s2[, 3])[ceiling(nboot * 
alpha1.s2.2)], sort(nuc.boot.glm.s2[, 3])[ceiling(nboot * 
alpha2.s2.2)])
alpha1.s2.3 <- pnorm(z0.s2.3 + ((z0.s2.3 + z.025)/(1 - a3 * (z0.s2.3 + 
z.025))))
alpha2.s2.3 <- pnorm(z0.s2.3 + ((z0.s2.3 + z.975)/(1 - a3 * (z0.s2.3 + 
z.975))))
bcas2.ci[4, ] <- c(sort(nuc.boot.glm.s2[, 4])[ceiling(nboot * 
alpha1.s2.3)], sort(nuc.boot.glm.s2[, 4])[ceiling(nboot * 
alpha2.s2.3)])
alpha.s2 <- c(alpha1.s2.0, alpha2.s2.0, alpha1.s2.1, alpha2.s2.1, 
    alpha1.s2.2, alpha2.s2.2, alpha1.s2.3, alpha2.s2.3)
length.bca.s2.ci <- abs(bca.s2.ci[, 2] - bca.s2.ci[, 1])

bca.s2.ci.all <- c(bca.s2.ci[1, ], bca.s2.ci[2, ], bca.s2.ci[3, ],
    bca.s2.ci[4, 1, length.bca.s2.ci])

if(bca.s1.ci[1, 1] < kull.lieb[, 1][1] & & bca.s1.ci[1, 2] > kull.lieb[, 1][1])
    bca.0.s1.cover <- 1
else bca.0.s1.cover <- 0

if(bca.s1.ci[2, 1] < kull.lieb[, 2][1] & & bca.s1.ci[2, 2] > kull.lieb[, 2][1])
    bca.1.s1.cover <- 1
else bca.1.s1.cover <- 0

if(bca.s1.ci[3, 1] < kull.lieb[, 3][1] & & bca.s1.ci[3, 2] > kull.lieb[, 3][1])
    bca.2.s1.cover <- 1
else bca.2.s1.cover <- 0

if(bca.s1.ci[4, 1] < kull.lieb[, 4][1] & & bca.s1.ci[4, 2] > kull.lieb[, 4][1])
    bca.3.s1.cover <- 1
else bca.3.s1.cover <- 0

if(bca.s2.ci[1, 1] < kull.lieb[, 1][1] & & bca.s2.ci[1, 2] > kull.lieb[, 1][1])
    bca.0.s2.cover <- 1
else bca.0.s2.cover <- 0

if(bca.s2.ci[2, 1] < kull.lieb[, 2][1] & & bca.s2.ci[2, 2] > kull.lieb[, 2][1])
    bca.1.s2.cover <- 1
else bca.1.s2.cover <- 0

if(bca.s2.ci[3, 1] < kull.lieb[, 3][1] & & bca.s2.ci[3, 2] > kull.lieb[, 3][1])
    bca.2.s2.cover <- 1
else bca.2.s2.cover <- 0

if(bca.s2.ci[4, 1] < kull.lieb[, 4][1] & & bca.s2.ci[4, 2] > kull.lieb[, 4][1])
    bca.3.s2.cover <- 1
else bca.3.s2.cover <- 0

bca.s1.coverage <- c(bca.0.s1.cover, bca.1.s1.cover, bca.2.s1.cover, 
    bca.3.s1.cover)
bca.s2.coverage <- c(bca.0.s2.cover, bca.1.s2.cover, bca.2.s2.cover, 
    bca.3.s2.cover)
# ADJUSTED

# Strategy 1 and Strategy 2

# proportion of bootstrap estimates for beta (out of nboot) that

# is less than the the Kullback-Leibler result

```r
bca.adj.sl.ci <- matrix(0, 4, 2)
z0.adj.sl.0 <- qnorm(sum(((nuc.boot.n.sl/sim.n) * nuc.boot glm.sl[, 1] <
glm.est[, 1]) * 1)/nboot)
z0.adj.sl.1 <- qnorm(sum(((nuc.boot.n.sl/sim.n) * nuc.boot glm.sl[, 2] <
glm.est[, 2]) * 1)/nboot)
z0.adj.sl.2 <- qnorm(sum(((nuc.boot.n.sl/sim.n) * nuc.boot glm.sl[, 3] <
glm.est[, 3]) * 1)/nboot)
z0.adj.sl.3 <- qnorm(sum(((nuc.boot.n.sl/sim.n) * nuc.bootglm.sl[, 4] <
glm.est[, 4]) * 1)/nboot)
z0.adj.sl <- c(z0.adj.sl.0, z0.adj.sl.1, z0.adj.sl.2, z0.adj.sl.3)
bca.adj.s2.ci <- matrix(0, 4, 2)
z0.adj.s2.0 <- qnorm(sum(((nuc.boot.n.s2/sim.n) * nuc.boot glm.s2[, 1] <
glm.est[, 1]) * 1)/nboot)
z0.adj.s2.1 <- qnorm(sum(((nuc.boot.n.s2/sim.n) * nuc.boot glm.s2[, 2] <
glm.est[, 2]) * 1)/nboot)
z0.adj.s2.2 <- qnorm(sum(((nuc.boot.n.s2/sim.n) * nuc.boot glm.s2[, 3] <
glm.est[, 3]) * 1)/nboot)
z0.adj.s2.3 <- qnorm(sum(((nuc.boot.n.s2/sim.n) * nuc.boot glm.s2[, 4] <
glm.est[, 4]) * 1)/nboot)
z0.adj.s2 <- c(z0.adj.s2.0, z0.adj.s2.1, z0.adj.s2.2, z0.adj.s2.3)
z0.025 <- qnorm(0.025)
z0.975 <- qnorm(0.975)
alphal.adj.sl.0 <- pnorm(z0.adj.sl.0 + ((z0.adj.sl.0 + z.025)/(1 - a0 * (z0.adj.sl.0 + z.025))))
alphal.adj.sl.1 <- pnorm(z0.adj.sl.1 + ((z0.adj.sl.1 + z.025)/(1 - a1 * (z0.adj.sl.1 + z.025))))
bcadj.sl.ci[1, ] <- c(sort((nuc.boot.n.sl/sim.n) * nuc.boot glm.sl[, 1])[
ceilingsim.n * alphal.adj.sl.0], sort((nuc.boot.n.sl/sim.n) * nuc.boot glm.sl[, 1])[
ceilingsim.n * alphal.adj.sl.0]
) 
alphal.adj.sl.1 <- pnorm(z0.adj.sl.1 + ((z0.adj.sl.1 + z.025)/(1 - a1 * (z0.adj.sl.1 + z.025))))
alphal.adj.sl.1 <- pnorm(z0.adj.sl.1 + ((z0.adj.sl.1 + z.975)/(1 - a1 * (z0.adj.sl.1 + z.975))))
bcadj.sl.ci[2, ] <- c(sort((nuc.boot.n.sl/sim.n) * nuc.boot glm.sl[, 2])[
ceilingsim.n * alphal.adj.sl.1], sort((nuc.boot.n.sl/
sim.n) * nuc.boot glm.sl[, 2])[(ceiling(nboot * alpha2.adj.sl.1.1)
})
alpha1.adj.sl.1.2 <- pnorm(z0.adj.sl.1.2 + ((z0.adj.sl.1.2 + z.025)/(1 - a2 * (z0.adj.sl.1.2 + z.025))))
alpha2.adj.sl.1.2 <- pnorm(z0.adj.sl.1.2 + ((z0.adj.sl.1.2 + z.975)/(1 - a2 * (z0.adj.sl.1.2 + z.975))))
bc.a.adj.sl.ci[3, ] <- c(sort((nuc.boot.n.sl/sim.n) * nuc.boot glm.sl[, 3])[(ceiling(nboot * alpha1.adj.sl.2]), sort((nuc.boot.n.sl/sim.n) * nuc.boot glm.sl[, 3])[(ceiling(nboot * alpha2.adj.sl.1.2)
})
alpha1.adj.sl.1.3 <- pnorm(z0.adj.sl.1.3 + ((z0.adj.sl.1.3 + z.025)/(1 - a3 * (z0.adj.sl.1.3 + z.025))))
alpha2.adj.sl.1.3 <- pnorm(z0.adj.sl.1.3 + ((z0.adj.sl.1.3 + z.975)/(1 - a3 * (z0.adj.sl.1.3 + z.975))))
bc.a.adj.sl.ci[4, ] <- c(sort((nuc.boot.n.sl/sim.n) * nuc.boot glm.sl[, 4])[(ceiling(nboot * alpha1.adj.sl.3]), sort((nuc.boot.n.sl/sim.n) * nuc.boot glm.sl[, 4])[(ceiling(nboot * alpha2.adj.sl.1.3)
})
alpha.adj.sl.1 <- c(alpha1.adj.sl.1.0, alpha2.adj.sl.1.0, alpha1.adj.sl.1.1, alpha2.adj.sl.1.1, alpha1.adj.sl.1.2, alpha2.adj.sl.1.2, alpha1.adj.sl.1.3, alpha2.adj.sl.1.3)
length.bca.adj.sl.ci <- abs(bca.adj.sl.ci[, 2] - bca.adj.sl.ci[, 1])
bc.a.adj.sl.ci.all <- c(bca.adj.sl.ci[1, ], bca.adj.sl.ci[2, ], bca.adj.sl.ci[3, ], bca.adj.sl.ci[4, ], length.bca.adj.sl.ci)
alpha1.adj.s2.0 <- pnorm(z0.adj.s2.0 + ((z0.adj.s2.0 + z.025)/(1 - a0 * (z0.adj.s2.0 + z.025))))
alpha2.adj.s2.0 <- pnorm(z0.adj.s2.0 + ((z0.adj.s2.0 + z.975)/(1 - a0 * (z0.adj.s2.0 + z.975))))
bc.a.adj.s2.ci[1, ] <- c(sort((nuc.boot.n.s2/sim.n) * nuc.boot glm.s2[, 1])[(ceiling(nboot * alpha1.adj.s2.0]), sort((nuc.boot.n.s2/sim.n) * nuc.boot glm.s2[, 1])[(ceiling(nboot * alpha2.adj.s2.0)
})
alpha1.adj.s2.1 <- pnorm(z0.adj.s2.1 + ((z0.adj.s2.1 + z.025)/(1 - a1 * (z0.adj.s2.1 + z.025))))
alpha2.adj.s2.1 <- pnorm(z0.adj.s2.1 + ((z0.adj.s2.1 + z.975)/(1 - a1 * (z0.adj.s2.1 + z.975))))
bc.a.adj.s2.ci[2, ] <- c(sort((nuc.boot.n.s2/sim.n) * nuc.boot glm.s2[, 2])[(ceiling(nboot * alpha1.adj.s2.1]), sort((nuc.boot.n.s2/sim.n) * nuc.boot glm.s2[, 2])[(ceiling(nboot * alpha2.adj.s2.1)
})
alpha1.adj.s2.2 <- pnorm(z0.adj.s2.2 + ((z0.adj.s2.2 + z.025)/(1 - a2 * (z0.adj.s2.2 + z.025))))
alpha2.adj.s2.2 <- pnorm(z0.adj.s2.2 + ((z0.adj.s2.2 + z.975)/(1 - a2 * (z0.adj.s2.2 + z.975))))

bcadj.s2.ci[3, ] <- c(sort((nuc.boot.n.s2/sim.n) * nuc.boot.glm.s2[, 3]) [ceiling(nboot * alpha1.adj.s2.2)], sort((nuc.boot.n.s2/sim.n) * nuc.boot.glm.s2[, 3]) [ceiling(nboot * alpha2.adj.s2.2)]

alpha1.adj.s2.3 <- pnorm(z0.adj.s2.3 + ((z0.adj.s2.3 + z.025)/(1 - a3 * (z0.adj.s2.3 + z.025))))
alpha2.adj.s2.3 <- pnorm(z0.adj.s2.3 + ((z0.adj.s2.3 + z.975)/(1 - a3 * (z0.adj.s2.3 + z.975))))

bcadj.s2.ci[4, ] <- c(sort((nuc.boot.n.s2/sim.n) * nuc.boot.glm.s2[, 4]) [ceiling(nboot * alpha1.adj.s2.3)], sort((nuc.boot.n.s2/sim.n) * nuc.boot.glm.s2[, 4]) [ceiling(nboot * alpha2.adj.s2.3)]

alpha.adj.s2 <- c(alpha1.adj.s2.0, alpha2.adj.s2.0, alpha1.adj.s2.1, alpha2.adj.s2.1, alpha1.adj.s2.2, alpha2.adj.s2.2, alpha1.adj.s2.3, alpha2.adj.s2.3)

length.bca.adj.s2.ci <- abs(bca.adj.s2.ci[, 2] - bca.adj.s2.ci[, 1])
bca.adj.s2.ci.all <- c(bca.adj.s2.ci[1, ], bca.adj.s2.ci[2, ], bca.adj.s2.ci[3, ], bca.adj.s2.ci[4, ], length.bca.adj.s2.ci)

if(bca.adj.s1.ci[1, 1] < kull.lieb[, 1][1] & bca.adj.s1.ci[1, 2] > kull.lieb[, 1][1])
  bca.0.adj.s1.cover <- 1
else bca.0.adj.s1.cover <- 0

if(bca.adj.s1.ci[2, 1] < kull.lieb[, 2][1] & bca.adj.s1.ci[2, 2] > kull.lieb[, 2][1])
  bca.1.adj.s1.cover <- 1
else bca.1.adj.s1.cover <- 0

if(bca.adj.s1.ci[3, 1] < kull.lieb[, 3][1] & bca.adj.s1.ci[3, 2] > kull.lieb[, 3][1])
  bca.2.adj.s1.cover <- 1
else bca.2.adj.s1.cover <- 0

if(bca.adj.s1.ci[4, 1] < kull.lieb[, 4][1] & bca.adj.s1.ci[4, 2] > kull.lieb[, 4][1])
  bca.3.adj.s1.cover <- 1
else bca.3.adj.s1.cover <- 0
bca.0.adj.s2.cover <- 1
else bca.0.adj.s2.cover <- 0
if(bca.adj.s2.ci[2, 1] < kull.lieb[, 2][1] && bca.adj.s2.ci[2, 2] >
kull.lieb[, 2][1])
  bca.1.adj.s2.cover <- 1
else bca.1.adj.s2.cover <- 0
if(bca.adj.s2.ci[3, 1] < kull.lieb[, 3][1] && bca.adj.s2.ci[3, 2] >
kull.lieb[, 3][1])
  bca.2.adj.s2.cover <- 1
else bca.2.adj.s2.cover <- 0
if(bca.adj.s2.ci[4, 1] < kull.lieb[, 4][1] && bca.adj.s2.ci[4, 2] >
kull.lieb[, 4][1])
  bca.3.adj.s2.cover <- 1
else bca.3.adj.s2.cover <- 0
bca.adj.s1.coverage <- c(bca.0.adj.s1.cover, bca.1.adj.s1.cover,
bca.2.adj.s1.cover, bca.3.adj.s1.cover)
bca.adj.s2.coverage <- c(bca.0.adj.s2.cover, bca.1.adj.s2.cover,
bca.2.adj.s2.cover, bca.3.adj.s2.cover)
write(t(z0.s1), "z0.s1", ncol = 4, append = T)
write(t(z0.s2), "z0.s2", ncol = 4, append = T)
write(t(a), "a", ncol = 4, append = T)
write(t(alpha.s1), "alpha.s1", ncol = 8, append = T)
write(t(alpha.s2), "alpha.s2", ncol = 8, append = T)
write(t(percent.s1.ci.all), "percent.s1.ci.all", ncol = 12, append = T)
write(t(percent.s2.ci.all), "percent.s2.ci.all", ncol = 12, append = T)
write(t(percent.s1.coverage), "percent.s1.coverage", ncol = 4, append = T)
write(t(percent.s2.coverage), "percent.s2.coverage", ncol = 4, append = T)
write(t(bca.s1.ci.all), "bca.s1.ci.all", ncol = 12, append = T)
write(t(bca.s2.ci.all), "bca.s2.ci.all", ncol = 12, append = T)
write(t(bca.s1.coverage), "bca.s1.coverage", ncol = 4, append = T)
write(t(bca.s2.coverage), "bca.s2.coverage", ncol = 4, append = T)
write(t(z0.adj.s1), "z0.adj.s1", ncol = 4, append = T)
write(t(z0.adj.s2), "z0.adj.s2", ncol = 4, append = T)
write(t(alpha.adj.s1), "alpha.adj.s1", ncol = 8, append = T)
write(t(alpha.adj.s2), "alpha.adj.s2", ncol = 8, append = T)
write(t(percent.adj.s1.ci.all), "percent.adj.s1.ci.all", ncol = 12, append = T)
write(t(percent.adj.s2.ci.all), "percent.adj.s2.ci.all", ncol = 12,
append = T)
write(t(percent.adj.s1.coverage), "percent.adj.s1.coverage", ncol = 4,
append = T)
write(t(percent.adj.s2.coverage), "percent.adj.s2.coverage", ncol = 4,
append = T)
write(t(bca.adj.s1.ci.all), "bca.adj.s1.ci.all", ncol = 12, append = T)
write(t(bca.adj.s2.ci.all), "bca.adj.s2.ci.all", ncol = 12, append = T)
write(t(bca.adj.s1.coverage), "bca.adj.s1.coverage", ncol = 4, append
= T)
write(t(bca.adj.s2.coverage), "bca.adj.s2.coverage", ncol = 4, append
= T)
write(t(boot.mean.s1), "boot.mean.s1", ncol = 16, append = T)
write(t(boot.mean.s2), "boot.mean.s2", ncol = 16, append = T)
write(t(sum.nuc.boot.glm.s1.b0), "sum.nuc.boot.glm.s1.b0", ncol=12,append=T)
write(t(sum.nuc.boot.glm.s1.b1), "sum.nuc.boot.glm.s1.b1", ncol=12,append=T)
write(t(sum.nuc.boot.glm.s1.b2), "sum.nuc.boot.glm.s1.b2", ncol=12,append=T)
write(t(sum.nuc.boot.glm.s1.b3), "sum.nuc.boot.glm.s1.b3", ncol=12,append=T)
write(t(sum.nuc.boot.glm.s2.b0), "sum.nuc.boot.glm.s2.b0", ncol=12,append=T)
write(t(sum.nuc.boot.glm.s2.b1), "sum.nuc.boot.glm.s2.b1", ncol=12,append=T)
write(t(sum.nuc.boot.glm.s2.b2), "sum.nuc.boot.glm.s2.b2", ncol=12,append=T)
write(t(sum.nuc.boot.glm.s2.b3), "sum.nuc.boot.glm.s2.b3", ncol=12,append=T)
}
Bibliography


