Bias and Efficiency of Logistic Regression involving a Binary Covariate with Missing Observations
Bias and Efficiency of Logistic Regression involving a Binary Covariate with Missing Observations

By
Kai Zhao

A Thesis
Submitted to the School of Graduate Studies
in Partial Fulfilment of the Requirements
for the Degree
Master of Science

McMaster University
© Copyright by Kai Zhao August 2010
MASTER OF SCIENCE (2010) McMaster University
(Statistics) Hamilton, Ontario

TITLE: Bias and Efficiency of Logistic Regression
involving a Binary Covariate with Missing Observations

AUTHOR: Kai Zhao, B.Sc.
(McMaster University, Canada)

SUPERVISOR: Dr. Stephen Walter

NUMBER OF PAGES: xi, 63
Abstract

In the statistical analysis of a health research study, it is quite common to have some missing data after data collection. Typically in a clinical trial, the treatment variable is completely recorded most of the time, but the associated covariates may not be. The multi-variable analysis is often conducted by including all the important medically relevant covariates with the expectation that a valid estimate of the treatment effect could be obtained by properly adjusting for these covariates. In this scenario, if the data of the covariates are Missing Not at Random (MNAR), the situation becomes complicated. The estimate of the treatment effect obtained will be invalid. The situation when the data are Missing Completely at Random (MCAR) is interesting since a dilemma exists: if you include the covariates with a high missing proportion, the analysis loses power although the validity might be good. If the covariates with a high missing proportion are excluded, the validity might be of question but the precision is good. Although the literature suggests that the validity is more important, there might be cases where the precision would improve substantially with a little sacrifice on validity by omitting the covariate from the analysis. In this thesis, this dilemma will be evaluated in the context of multivariable logistic regression with the hope that some of the results from this work would shed light on the understanding of the situation. This work is significant in that it could potentially change the data collection process. For example, in the research design stage, if we expect that a covariate would have a high rate of missingness,
there might be little to gain by collecting this information. Furthermore, the results from this work may guide decisions about data collection. If we decide that a covariate does not need to be collected, then the relevant resources could be released to apply to other important aspects of a study.
Acknowledgements

I would like to thank my supervisor, Dr. Stephen Walter, whose supervision, guidance and encouragement made this project possible and enjoyable.

I would also like to thank Dr. P. Macdonald and Dr. A. Childs for serving on my supervisory committee. Dr. P. Macdonald advised on the R code and Dr. A. Childs provided valuable comments and advice.

Special thanks to Diane Heels-Ansdell who works in CE&B and helped me getting approval to use the data in SPRINT and RESTORE clinical trials and prepared the data set for me.

I would like to thank Dr. R. Viveros, Dr. R. Zhu, Dr. A. Canty, Dr. F. Hoppe and Dr. S. Feng for their teaching and help in my M.Sc. study. I also like to thank all of my friends for their friendship and help.

Finally, I would like to thank my family members, whose encouragement and support enable me to complete this project successfully.
2.4.3 Case 3: Covariate mean difference $\delta \neq 0$ and sample size $N = 50$ ... 25

2.4.4 Case 4: Covariate mean difference $\delta \neq 0$ and sample size $N = 200$ ... 31

2.5 Summary of Simulation Results ................................................. 39

2.5.1 Sample size effect ................................................................. 39

2.5.2 Baseline risk effect ............................................................... 42

2.5.3 Covariate missing effect ......................................................... 42

2.5.4 Covariate mean difference effect ........................................... 42

3 Real Applications ................................................................. 44

3.1 SPRINT study ........................................................................... 44

3.2 RESTORE study ...................................................................... 47

4 Discussion and Future Work ..................................................... 50

REFERENCES ................................................................................. 51

APPENDICES .................................................................................. 53

A Flowchart for Simulation ............................................................. 54

B Source code in R for Simulation ................................................ 54

C Source code in R for Graphs ....................................................... 58
List of Tables

2.1 Simulation Parameter Values ........................................... 12
2.2 Model Specifications in Figures ........................................ 14
2.3 Figures and the corresponding Simulation Parameters .............. 14
2.4 Bias and variance trend when one of the parameters increases .... 19
2.5 Bias and variance trend when one of the parameters increases .... 20
2.6 Bias and variance trend when one of the parameters increases .... 30
2.7 Best model for different parameters when δ = 0.4 and N = 50 ......... 31
2.8 Bias and variance trend when one of the parameters increases .... 34
2.9 Best model for different parameters when δ = 0.4 and N = 200 ......... 39

3.1 Standard error of $\hat{\beta}_1$ and its relative precision of different models adjusting or not adjusting for SMOKE on OPEN fracture SPRINT data comparing with the simulation using the same parameter .......................... 45

3.2 Standard error of $\hat{\beta}_1$ and its relative precision of different models adjusting or not adjusting for SMOKE on CLOSE fracture SPRINT data comparing with the simulation using the same parameter .......................... 46
3.3 Standard error of $\hat{\beta}_1$ and its relative precision of different models adjusting or not adjusting for AGE on RESTORE data comparing with the simulation using the same parameter ........................................ 48

3.4 Standard error of $\hat{\beta}_1$ and its relative precision of different models adjusting or not adjusting for ANTPOS on RESTORE data comparing with the simulation using the same parameter ........................................ 49

3.5 Standard error of $\hat{\beta}_1$ and its relative precision of different models adjusting or not adjusting for MEDLAT on RESTORE data comparing with the simulation using the same parameter ........................................ 49
List of Figures

2.1 Bias comparison of the 7 models when $N = 50$, $p = 0.1$, and $\delta = 0$ . . . . . . . 15

2.2 Variance comparison of the 7 models when $N = 50$, $p = 0.1$, and $\delta = 0$ . . . . 16

2.3 Bias comparison of the 7 models when $N = 50$, $p = 0.4$, and $\delta = 0$ . . . . . . . 17

2.4 Variance comparison of the 7 models when $N = 50$, $p = 0.4$, and $\delta = 0$ . . . . 18

2.5 Bias comparison of the 7 models when $N = 200$, $p = 0.1$, and $\delta = 0$ . . . . . . 21

2.6 Variance comparison of the 7 models when $N = 200$, $p = 0.1$, and $\delta = 0$ . . . 22

2.7 Bias comparison of the 7 models when $N = 200$, $p = 0.4$, and $\delta = 0$ . . . . . . 23

2.8 Variance comparison of the 7 models when $N = 200$, $p = 0.4$, and $\delta = 0$ . . . 24

2.9 Bias comparison of the 7 models when $N = 50$, $p = 0.1$, and $\delta = 0.4$ . . . . . . 26

2.10 Variance comparison of the 7 models when $N = 50$, $p = 0.1$, and $\delta = 0.4$ . . . 27

2.11 Bias comparison of the 7 models when $N = 50$, $p = 0.4$, and $\delta = 0.4$ . . . . . . 28

2.12 Variance comparison of the 7 models when $N = 50$, $p = 0.4$, and $\delta = 0.4$ . . . 29

2.13 Bias comparison of the 7 models when $N = 50$, $\beta_1 = 0.5 \times \ln 2$, and $\delta = 0.4$ . 32

2.14 Variance comparison of the 7 models when $N = 50$, $\beta_1 = 0.5 \times \ln 2$, and $\delta = 0.4$ 33

2.15 Bias comparison of the 7 models when $N = 200$, $p = 0.1$, and $\delta = 0.4$ . . . . . . 35
2.16 Variance comparison of the 7 models when $N = 200$, $p = 0.1$, and $\delta = 0.4$ .

2.17 Bias comparison of the 7 models when $N = 200$, $p = 0.4$, and $\delta = 0.4$ .

2.18 Variance comparison of the 7 models when $N = 200$, $p = 0.4$, and $\delta = 0.4$ .

2.19 Bias comparison of the 7 models when $N = 200$, $\beta_1 = 0.5 \times \ln 2$, and $\delta = 0.4$ .

2.20 Variance comparison of the 7 models when $N=200$, $\beta_1 = 0.5 \times \ln 2$, and $\delta = 0.4$ .

4.1 Flowchart for simulation .
Chapter 1

Introduction

1.1 Project Overview

In health research, researchers are interested to know the efficacy or effectiveness of a treatment such as quantifying the treatment effect of a new medicine on a specific disease. The interested clinical outcome is usually binary. For example, the patient is either cured or not cured. The logistic regression is suitable to be used when the outcome is binary. In logistic model, the treatment effect is estimated by adjusting other medically relevant factors, which are also typically collected at baseline.

The main interest of this study is to compare relative efficiency, MSE and bias of the treatment coefficient estimate between an unadjusted logistic regression model with complete data and an adjusted logistic regression model with a partially missing baseline measurement (covariate).

Robinson (1991) studied the effect of adjusting for covariate on precision in logistic regression. White (2005) studied adjusting for partially missing continuous baseline covariate. A lot of studies had performed simulations using parameters on a specific data set from
A clinical trial so that the results and conclusions are only exploratory (White, 2005). A pure simulation is used in this work through fitting logistic regression models based on the characteristics of clinical trial data in general. By choosing a series of values of different parameters in the logistic regression model, the trend of MSE, variance, and bias difference of the estimated treatment effect between models are studied. The logistic model in this work typically has two independent factor variables, one is the treatment variable and one is a baseline measurement. The simulation parameters are the sample size $N$, risk of disease $p$, treatment effect $\beta_1$, baseline effect $\beta_2$, baseline mean difference between treatment groups $\delta$ and baseline covariate missing proportion $\eta$. Values of the treatment variable $x_1$ are 1 and 0, where 1 denotes the treatment group and 0 denotes the control group. The baseline measurement variable $x_2$ is only assumed dichotomous in this simulation study. Samples are generated unconditionally by these parameters. However, it does not mean that sample data have these properties. The unconditional logistic regression method is used. Also, the treatment variable and the baseline variable of interest are assumed to be independent when $\delta = 0$ in the sense that the treatment assignment $x_1$ does not depend on the baseline variable $x_2$, which is typical in a randomized clinical trial. Missing mechanism of Missing Completely at Random (MCAR) is assumed in the simulation.

Details of simulation parameter value selection and simulation method are included in Chapter 2. Some heuristics based on the simulation results are summarized and conclusions drawn based on different parameter combinations. In Chapter 3, data from two real clinical trials, one with binary baseline measurement (SPRINT study) and one with continuous baseline measurement (RESTORE study), are used to verify the findings in the simulation. Discussion and future work are included in Chapter 4 of this thesis.
1.2 Motivation of the study

Many researchers (White 2005, Negassa and Hanley 2006) have done work in this topic. However, none of them have conducted a pure simulation study on binary covariate missing effect. Negassa (2006) did simulation on the effect of omitted covariates, but didn't investigate the effect of the missingness. White (2005) did simulation on binary outcome with continuous missing covariate but didn't perform simulation on binary missing covariate. Our simulation is based on binary outcome with binary covariate with different missing proportions. A simulation study has the advantage of exploring more situations by flexibly adjusting different parameters so that it has a broader view, although research based on one real study may be more convincing to clinicians. In health research, resources are limited and researchers are looking for different ways to improve resource allocation. One typical example is to use the smallest possible sample size to gain enough power. In the design stage, the sample size will be determined with the goal to detect a treatment effect which is clinically important.

Missing data are common in real situations and can have serious impact. Depending on different missing mechanism, the effect of missing is different. Missing Completely at Random (MCAR) is data are missing independent of observable variables and unobservable parameters of interest. Missing at Random (MAR) is that the data are missing dependent on covariates but not the missing data itself. When neither MCAR or MAR hold, the data are Missing Not at Random (MNAR). If data are MCAR, the study will lose power. If data are MAR, the data analysis should take into account the missing-relevant covariates and observations to obtain a valid estimate of treatment effect. If data are MNAR, then invalid estimates of the treatment effect are usually obtained. Although a lot of statistical models have been proposed, none of them is fully convincing as the missing depends on the unobservable (Molenberghs and Kenward, 2007).

Previous researches were in different settings. For example, many research papers (Gail

3
1984, Begg 1993, Hauck, Anderson, and Marcus 1998, Negassa and Hanley 2006) focused on when to adjust a covariate, which is trying to omit the covariate problem. Their solutions fall into model selection methods, without considering covariate missing proportions. According to Gail (1984), omission of an important predictive covariate generates biased estimate of treatment effects. Another big portion of research focused on the missing data issue. In these papers, the missing data is due to lost follow-up or patient withdrawals from the study at a later stage. Two strategies are commonly used to deal with the missing covariate problem: omitting all subjects with missing values or introducing additional categories for missing values if they occur. The first method result in inconsistent estimates and the second almost always causes serious bias (Vach 1993). Another portion of research focused on handling missing covariates and comparison of different approaches (Vach 1993, Peng 2008, and Greenland 1995). There are different types of imputation methods. However, how to handle missing values is out of the scope of this study. This work focuses on missing baseline covariate and evaluates the appropriateness of adjusting a baseline covariate with different missing proportions.

This work could help to provide a guideline on data collection based on medical experts’ view of baseline predictive power and the possibility of percentage missing before the data are actually collected, which could potentially improve cost efficiency. In addition, this work would help statistical analysts to choose the appropriate covariates to be adjusted in the multivariable model after data collection.

1.3 Logistic Regression

Logistic regression is commonly used in the data analysis in epidemiological studies. What distinguishes logistic regression from linear regression is that the outcome variable is dichotomous. In any regression problem, the key quantity is the mean value of the outcome variable
given values of independent variables. This quantity is called the conditional mean and is expressed as $E(Y|x)$, where $Y$ denotes the outcome variable and $x$ denotes the independent variables. The logistic regression can be expressed as $\log \frac{p(Y)}{1-p(Y)} = \beta_0 + \sum \beta_i x_i$, where $p(Y)$ denotes the probability of an outcome event. The left side of this expression is called the logit transformation. Logistic regression is either used for the prediction of the probability of occurrence of an event or for examining the relationship between the occurrence of an event and the treatment/exposure. Using logistic regression to study the relationship between the occurrence of an event and the treatment/exposure is more often in the medical sciences as researchers in this field are usually more interested in knowing the treatment effect.

In unadjusted or univariable logistic regression, the predictor could be dichotomous, categorical or continuous. In the case of a dichotomous predictor, logistic regression reduces to a $2 \times 2$ contingency table, for which various other statistical methods have been proposed to study the association between the outcome and treatment/exposure. The coefficient estimates in logistic regression in this case are equivalent to the odds ratio (OR) obtained from the $2 \times 2$ table. The test statistic for testing the null hypothesis of no treatment effect is equivalent to the Chi-square test. With a categorical predictor having $n$ levels, logistic regression reduces to a $2 \times n$ contingency table. In adjusted logistic regression, the relationship between the occurrence of an event and the treatment/exposure is studied by controlling the effect of other variables. The odds ratio obtained is believed to be more valid than the odds ratio obtained from univariable logistic regression.

1.4 Logistic Regression on RCT

A Randomized Controlled Trial (RCT) is a type of scientific experiment commonly used in testing the efficacy or effectiveness of health care services such as medicine or nursing, or health technologies such as pharmaceuticals, medical devices, or surgery. It has the most
internally valid assessment of efficacy and safety among all of the clinical trial methodologies (Hulley 2007).

For an RCT with a dichotomous outcome, logistic regression is often used in the data analysis. The outcome event is usually a clinical event such as getting a disease or not, so the outcome $Y$ can be represented by a dichotomous variable that has value 0 or 1. Logistic regression in which the response is a clinical outcome $Y$ and the predictive variables are a treatment variable $x_1$ and a baseline characteristic $x_2$ can be expressed as:

$$\log \mu = \beta_0 + \beta_1 x_1 + \beta_2 x_2, \text{ where } \mu = E(Y|x_1, x_2)$$

Here, $x_1$ is a dichotomous variable indicating either getting the treatment or not, while $x_2$ can be dichotomous or continuous. Here dichotomous $x_2$ is first studied, for example the covariate smoking. In a well-designed RCT, randomization is intended to ensure well-balanced/comparable intervention groups and hence minimizes the potential confounding. However, if the sample size is modest or small, that might not be the case. For studies with small-to-moderate sample size, chance of disparity imbalance in the distribution of important covariates is still a possibility after randomization unless stratified block randomization is employed (Negassa and Hanley, 2006). Therefore the covariate mean difference $\delta$ is first set to zero, which represents the non-confounding case, and then set to non-zero which represents the confounding case. The non-confounding case can be a RCT in a real study and the confounding case can be an observational study. The simulation was performed using unconditional logistic regression which means that even when we set $\delta = 0$, the data set generated may actually have a nonzero covariate mean difference between two treatment groups.

Baseline variable missing will reduce the study power. The missing proportion is usually different from that stated in the study protocol. It is usually predicted by referencing other similar studies. Currently, this proportion might be used to adjust the sample size, but the
implication and benefit on data collection has never been studied.

Outcome $Y$ missing is often caused by patient withdrawal or lost follow-up. Treatment allocation $x_1$ is rarely missing because the patient will not be included in the study unless he/she is randomized. In this work, the interest focuses on the impact from missing covariate $x_2$. Specifically, an important covariate will be missing different proportions and the impact on the treatment effect estimate will be evaluated by adjusting or not adjusting this covariate.

1.5 Adjusted versus Unadjusted Logistic Regression

Confounding is a common problem in observational studies. In statistical theory, a confounding variable is an extraneous variable in a statistical model that correlates (positively or negatively) with both the dependent variable and the independent variable. When a covariate is a confounding variable, the treatment effect in a multivariable model can only be validly estimated by including this variable.

When a covariate is a non-confounding variable, the inclusion of this covariate in a multivariable model depends on the model. In classic linear regression, the adjustment for a non-confounding predictive covariate results in improved precision, whereas such adjustment in logistic regression results in loss of precision (Robinson, 1991). However, when testing for a treatment effect in randomized trials studies, Robinson (1991) proved it is more efficient to adjust for predictive covariates when logistic models are used, so for testing the behavior of the logistic regression is the same as that of classic linear regression.

In clinical trial design, sample size is usually predetermined to ensure that the study has enough power to detect the treatment effect. Baseline information to be collected is confirmed by medical specialists. Normally only those baseline characteristics which have or may have some predictive power related to the outcome will be collected among the enrolled
patients. Most baseline data collected may have some predictive power. In this work, missing is only considered on those covariates which have certain predictive power. Those covariates might be confounding or non-confounding variables.

Because these baseline covariates are predictive, in common sense, they should be adjusted for in the logistic regression analysis. However, if it has very little predictive power ($\beta_2$ is very small), the covariate can be safely excluded without substantial impact on the estimated treatment effect, even when the data are complete. If the covariate has moderate effect or high effect ($\beta_2$ is not too small), and the data are complete, then it should be adjusted for.

If this covariate is a confounding covariate, then not adjusting will lead to a biased treatment effect estimate. On the other hand, if it is adjusted for but has a high proportion of missing, the adjustment will substantially reduce the precision of the treatment effect estimate.

In the case that the covariate is a non-confounding covariate, not adjusting can still produce valid results, while adjusting for this covariate may result in a loss of precision due to missing values. Robinson (1991) suggested that even with complete data, adjusting for a covariate in logistic regression will result in a loss of precision. Since both unadjusted and adjusted models will produce valid results, and considering that the adjusted model has less precision, we should simply choose omitting the non-confounding $x_2$ from the model. Therefore, for a non-confounding predictive covariate, precision is the key to model selection.

1.6 Validity and Precision

In the multivariable logistic regression analysis, if the relationship between the occurrence of an event and the treatment/exposure is of interest, then the estimate of the treatment
coefficient and its test statistic are the focus. If the test statistic is statistically significant, it indicates that the treatment effect is real. If the coefficient is large enough to be medically interesting, then the effect is clinically significant. The confidence interval of this coefficient shows how precise this estimate would be. In the multivariable logistic model, if all the important covariates are properly adjusted, then the resulting estimates are called valid results. The ideal situation is that the estimate is valid and its precision is excellent. In reality, this is usually not the case. In logistic regression, complete data is required for all the covariates. If the missing data is MNAR, then it is not possible to obtain a valid answer for the research question. If the missing data is MAR or MNAR, the missingness will decrease the probability of detecting the treatment effect. The missingness is assumed to be MCAR in the subsequent discussion. Frequently, researchers have the following dilemma: if a variable has a high missing rate, as an explanatory variable, the logistic regression including this variable will have a valid estimate with low power and low precision which is indicated by a large standard error; the logistic regression excluding this variable will have a biased estimate but with a good precision. In the literature, it suggests that validity is superior to precision. However, when the validity loss is small compared with the large gain in precision, the biased estimate is usually recommended. This work will shed some insights on this topic and provide a guideline to help researchers making the better choice. To study the validity and precision tradeoff, the bias, variance and MSE of $\hat{\beta}_1$ will be calculated. In addition, the relative efficiency (RE), which is defined as the ratio of standard error between two models, is calculated for the two real applications. The unadjusted model which is only contains the treatment covariate is a reference model. The other model is an adjusted model which has a baseline covariate that contains different amount of missing values. The relative efficiency is a measure for precision. The bias will also be calculated to be a measure of validity. The usefulness of this work relies on identifying the situations where the two variable model is preferred.
Chapter 2

Simulation Details

2.1 Simulation Parameters

The logistic model is specified as: \( \log(\text{odds}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \), in which \( \beta_0, \beta_1 \), and \( \beta_2 \) are model coefficients, \( x_1 \) denotes the treatment variable, and \( x_2 \) denotes a dichotomous baseline covariate. The probability of having the outcome event can be obtained with \( P(Y) = \frac{1}{1+e^{-z}} \) where \( z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \), and \( Y \) is the dichotomous study outcome.

In another form, the model can be written as \( \text{Logit} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \). To conduct a simulation, the first task is to generate data. The following describes how the data have been generated. The first step is to choose proper values of \( \beta_0, \beta_1 \), and \( \beta_2 \). The second step is to generate a treatment variable \( x_1 \) and a dichotomous baseline covariate \( x_2 \), which are independent of each other when \( \delta = 0 \). In other words, the treatment group allocation is unconditional on the baseline values.

For the treatment coefficient \( \beta_1 \), two-fold odds ratio is considered as a substantial treatment effect based on most clinical trial results, which corresponds to \( \ln 2 \) in the value of \( \beta_1 \). \( \text{OR} = 1 \) represents no difference between the treatment and placebo group. Therefore,
the range of $\beta_1$ to be considered is from ln 1 to ln 2. Similarly the baseline coefficient $\beta_2$ is considered to be in a range from ln 1 to ln 2 as well. $\beta_0$ is calculated from the value of the baseline risk $p$. In real cases, the odds ratio sometimes can be between 2 and 10, but they are not considered here because such large treatment effects are unusual in clinical trials. As a first step, a typical value of 2 in OR has been chosen. The baseline risk $p$ is considered from 10% to 50%, which is a reasonable range in epidemiology studies.

Here, $\delta$ is the mean difference of $x_2$ between the two treatment groups. Ideally, in RCT, the expectation of $\delta$ should be 0. However it is not true for non-RCT, and may not even be true in small sample RCTs. The simulation with $\delta = 0$ is first attempted and followed by the simulations with $\delta \neq 0$, which represents the confounding effect.

The treatment group allocation variable $x_1$ could be easily generated as a dichotomous variable where 1 represents the treatment group and 0 represents the placebo group, with equal probability. More complicated treatment group allocations are not considered in this work.

The outcome variable $Y$ can be generated from given values of $\beta_0$, $\beta_1$, $\beta_2$ and $x_1$, $x_2$, assuming that it follows the binomial distribution with parameters sample size $N$ and probability $\mu$, where $\mu = \frac{e^z}{1+e^z}$ and $z = \beta_0 + \beta_1 x_1 + \beta_2 x_2$.

### 2.2 Simulation Method

As mentioned before, simulated data from a logistic model is generated first. The specific model represents the actual underlying relationship. The mathematical model is $Logit = \beta_0 + \beta_1 x_1 + \beta_2 x_2$, where $x_1$ denotes the treatment variable, $x_2$ denotes the dichotomous covariate, which could be adjusted, and $\beta_0$, $\beta_1$, $\beta_2$ denote the coefficients in the logistic regression model. Table 2.1 contains a list of values or ranges which have been chosen for
the simulation parameters.

<table>
<thead>
<tr>
<th>Simulation Parameter</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>disease event rate $p$</td>
<td>10% 20% 30% 40% 50%</td>
</tr>
<tr>
<td>$\beta_0 = \ln(p/(1-p))$</td>
<td>0.1 $\ln 2$ 0.2 $\ln 2$ ... 0.9 $\ln 2$ $\ln 2$</td>
</tr>
<tr>
<td>treatment coefficient $\beta_1$</td>
<td>0.1 $\ln 2$ 0.2 $\ln 2$ ... 0.9 $\ln 2$ $\ln 2$</td>
</tr>
<tr>
<td>baseline coefficient $\beta_2$</td>
<td>0.1 $\ln 2$ 0.2 $\ln 2$ ... 0.9 $\ln 2$ $\ln 2$</td>
</tr>
<tr>
<td>covariate mean difference $\delta$</td>
<td>0(RCT) 0.1 0.2 0.3 0.4</td>
</tr>
<tr>
<td>$x_1$</td>
<td>1(treatment) 0(control)</td>
</tr>
<tr>
<td>$x_2$</td>
<td>1(smoker) 0(non-smoker)</td>
</tr>
</tbody>
</table>

The maximum effect of these coefficients represents a 2-fold increase in the odds ratio, which is widely used in such simulations.

After a logistic model is specified, the sample is randomly generated. Different sample sizes are chosen to generate different data sets for subsequent analysis. This is designed for the purpose of studying the influence of sample size. The sample size $N = 50$ and $N = 200$ are simulated.

The first generated variable is the treatment variable $x_1$. Based on the sample size, half of the subjects are allocated to the treatment group and half to the control group. The variable $x_2$ is generated in a fashion that there will be two distributions with the mean difference $\delta$ between treatment and control group. $\delta$ has the values of 0 to 0.4 for binary covariate $x_2$, which represents different levels and direction of confounding effect. If $\delta$ has a value other than zero, $x_2$ will be correlated with the treatment. As it is already known that $x_2$ has certain predictive power, if $\delta$ is non-zero, $x_2$ satisfies the definition of confounding, which is an effect that should be adjusted for. Once $x_1$ and $x_2$ have been generated, the probability of the outcome can be obtained through the logistic model. This probability will be used to generate the dichotomous outcome variable $Y$, which follows a binomial distribution. At this point, a complete data set with variables $Y$, $x_1$ and $x_2$ is ready for subsequent statistical
analysis. For this complete data set, two different models will be tried to find the original relationship. The two models are the logistic model with or without the adjusted covariate $x_2$. The next step is to randomly delete some of the $x_2$ values and try the adjusted logistic regression. The missing proportions $\eta$ are 10\%, 20\%, 30\%, 40\% and 50\%. Corresponding to each specific randomly generated sample, there will be 7 models in total. The estimated $\hat{\beta}_1$ in these models are then compared with the underlying true value. Each model is simulated 1000 times, and then the MSE, bias and variance of $\hat{\beta}_1$ will be obtained. The reference model is the model with complete data without adjusting for the covariate $x_2$. The bias could be obtained through subtracting the estimated $\hat{\beta}_1$ from $\beta_1$ in the initial specification. The variance and bias will be summarized and displayed in the next section.

2.3 Figure Layout for Simulation Results

Figure 2.1 and 2.2 show how the bias and variance change with $\beta_1$, $\beta_2$ and the missing proportion $\eta$. These are the simulation results with the simulation parameters $\delta = 0$, $N = 50$, and $p = 0.1$, where $\delta$ denotes the mean difference between treatment group and control group of the second covariate, $N$ is the number of observations generated in the simulation and $p$ is the probability of the event at baseline (baseline risk). In both figures, there are 10 panels which are associated with different $\beta_2$ values from $0.1 \times \ln 2$ to $\ln 2$ by the step $0.1 \times \ln 2$ from upper left to lower right. In each panel, the horizontal axis represents $\beta_1$ whose values range from $0.1 \times \ln 2$ to $\ln 2$. the vertical axis represents the bias of $\hat{\beta}_1$ in Figure 2.1 and the variance of $\hat{\beta}_1$ in Figure 2.2. There are seven lines in each panel which represents the models with different missing proportions $\eta$. The associated models are summarized in Table 2.2 as follows:

In Figure 2.1 and Figure 2.2, the patterns of how the bias and precision vary with $\beta_1$, $\beta_2$ and the missing proportion in the seven models are shown.
Table 2.2: Model Specifications in Figures

<table>
<thead>
<tr>
<th>Model</th>
<th>Covariates in the model</th>
<th>Model Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$x_1, x_2$</td>
<td>adjusted with $x_2$ complete</td>
</tr>
<tr>
<td>2</td>
<td>$x_1, x_2$</td>
<td>adjusted with $x_2$ 10% missing</td>
</tr>
<tr>
<td>3</td>
<td>$x_1, x_2$</td>
<td>adjusted with $x_2$ 20% missing</td>
</tr>
<tr>
<td>4</td>
<td>$x_1, x_2$</td>
<td>adjusted with $x_2$ 30% missing</td>
</tr>
<tr>
<td>5</td>
<td>$x_1, x_2$</td>
<td>adjusted with $x_2$ 40% missing</td>
</tr>
<tr>
<td>6</td>
<td>$x_1, x_2$</td>
<td>adjusted with $x_2$ 50% missing</td>
</tr>
<tr>
<td>7</td>
<td>$x_1$</td>
<td>unadjusted</td>
</tr>
</tbody>
</table>

For each combination of other simulation parameters, the bias and variance of the seven models will be shown in the figures in section 2.4, which have the same style as Figure 2.1 and 2.2. The figure number and the corresponding simulation parameters are summarized in Table 2.3.

Table 2.3: Figures and the corresponding Simulation Parameters

<table>
<thead>
<tr>
<th>Bias Trends</th>
<th>Variance Trends</th>
<th>Sample Size ($N$)</th>
<th>Baseline Risk ($p$)</th>
<th>$x_2$ Mean difference ($\delta$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 2.1</td>
<td>Figure 2.2</td>
<td>50</td>
<td>0.1</td>
<td>0</td>
</tr>
<tr>
<td>Figure 2.3</td>
<td>Figure 2.4</td>
<td>50</td>
<td>0.4</td>
<td>0</td>
</tr>
<tr>
<td>Figure 2.5</td>
<td>Figure 2.6</td>
<td>200</td>
<td>0.1</td>
<td>0</td>
</tr>
<tr>
<td>Figure 2.7</td>
<td>Figure 2.8</td>
<td>200</td>
<td>0.4</td>
<td>0</td>
</tr>
<tr>
<td>Figure 2.9</td>
<td>Figure 2.10</td>
<td>50</td>
<td>0.1</td>
<td>0.4</td>
</tr>
<tr>
<td>Figure 2.11</td>
<td>Figure 2.12</td>
<td>50</td>
<td>0.4</td>
<td>0.4</td>
</tr>
<tr>
<td>Figure 2.15</td>
<td>Figure 2.16</td>
<td>200</td>
<td>0.1</td>
<td>0.4</td>
</tr>
<tr>
<td>Figure 2.17</td>
<td>Figure 2.18</td>
<td>200</td>
<td>0.4</td>
<td>0.4</td>
</tr>
</tbody>
</table>

14
Figure 2.1: Bias comparison of the 7 models when $N = 50$, $p = 0.1$, and $\delta = 0$. 

$N = 50$, $\delta = 0$, $p = 0.1$
Figure 2.2: Variance comparison of the 7 models when $N = 50, \rho = 0.1$, and $\delta = 0$.
Figure 2.3: Bias comparison of the 7 models when $N = 50$, $p = 0.4$, and $\delta = 0$. 

$N = 50$, $\delta = 0$, $p = 0.4$
0 = \phi = \theta = 0.4, \text{ and } N = N^* = 10^2, \text{ and } p = 0.4.
2.4 Simulation Explanation

2.4.1 Case 1: Covariate mean difference $\delta = 0$ and sample size $N = 50$

Simulation with different values of $p(\beta_0)$, $\beta_1$, and $\beta_2$ have been run for the seven different models for sample size $N = 50$ with 1000 iterations at each run. The bias and variance of the seven models are shown in Figure 2.1, 2.3, 2.2, and 2.4. The trends and patterns of the bias and variance of the seven models are shown in Table 2.4. The bias and variance change monotonously with $\beta_1$, $\beta_2$, missing proportion $\eta$, and baseline risk $p$. The best model with the minimum bias or variance is suggested.

<table>
<thead>
<tr>
<th>As one of the following increases</th>
<th>Missing $\beta_1$</th>
<th>Baseline $\beta_2$</th>
<th>$\eta$</th>
<th>Risk ($p$)</th>
<th>Best Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bias trend</td>
<td>bigger</td>
<td>a bit smaller</td>
<td>bigger</td>
<td>smaller</td>
<td>unadjusted model</td>
</tr>
<tr>
<td>Variance trend</td>
<td>smaller</td>
<td>a bit smaller</td>
<td>bigger</td>
<td>smaller</td>
<td>unadjusted model</td>
</tr>
</tbody>
</table>

In Figure 2.1, for all seven models, the bias of $\hat{\beta}_1$ gets bigger when $\beta_1$ increases. This might be easy to understand as $\beta_1$ has bigger absolute value tends to have bigger absolute bias. Alternatively the relative bias of $\frac{\hat{\beta}_1}{\beta_1}$ may stay the same. Another trend is that the bias gets smaller when $\beta_2$ increases, probably because $\beta_2$ takes a bigger portion of the effect so that little has been left for $\beta_1$. A bigger missing proportion $\eta$ of the second covariate leads to bigger treatment effect bias. The unadjusted model always has the least bias among the seven models, and has very little difference with the adjusted model based on complete data. It indicates that without the confounding effect, the adjusted model with complete data is not much different from the unadjusted model with complete data. In Figure 2.2, the variance of $\hat{\beta}_1$ gets smaller as $\beta_1$ increases. It also gets smaller as $\beta_2$ increases. The unadjusted model
always has the least variance among the seven models and the adjusted model with complete data is very close to the unadjusted model. This is compliant with Robinson (1991) which suggested that in a logistic regression model, adjusting for covariate decreases the precision of estimation. This simulation is repeated with the baseline risk changed from 0.1 to 0.4. The associated results are presented in Figure 2.3 and 2.4. Comparing Figure 2.1 with 2.3, we see that a bigger baseline risk leads to smaller bias. Comparing Figure 2.2 with 2.4, shows that a bigger baseline risk leads to smaller variance. This is because when $p = 0.5$, in the $2 \times 2$ table, the variance is the smallest.

To examine the impact of the sample size, the simulation corresponding to Figure 2.1 and 2.2 is repeated with the sample size changed from 50 to 200 in the next section.

### 2.4.2 Case 2: Covariate mean difference $\delta = 0$ and sample size $N = 200$

The simulation results are presented in Figure 2.5, 2.6, 2.7 and 2.8.

<table>
<thead>
<tr>
<th>As one of the following increases</th>
<th>Best Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta_1$</td>
<td>Missing Baseline Risk ($p$)</td>
</tr>
<tr>
<td>$\beta_2$</td>
<td>$\eta$</td>
</tr>
<tr>
<td>Bias trend</td>
<td>no difference no difference bigger smaller unadjusted model</td>
</tr>
<tr>
<td>Variance trend</td>
<td>no difference no difference bigger smaller unadjusted model</td>
</tr>
</tbody>
</table>

It is clear that both the variance and bias get smaller when the sample size increases. The bias and variance trends as $\beta_1$, $\beta_2$ increase are different from smaller sample size. Both the bias and variance are stabilized given a large enough sample. The bias is consistent for the seven models and is almost always close to zero except for the case when $p = 0.1$ and $\beta_2$ is very small, which showed some bias as can be seen in Figure 2.5. The unadjusted
Figure 2.5: Bias comparison of the 7 models when \( N = 200, \delta = 0, p = 0.1 \) and \( \delta = 0 \).
Figure 2.6: Variance comparison of the 7 models when N = 200, p = 0.1, and δ = 0.
Figure 2.7: Bias comparison of the 7 models when $N = 200$, $p = 0.4$, and $\delta = 0$. 

$N = 200$, $\delta = 0$, $p = 0.4$
Figure 2.8: Variance comparison of the 7 models when $N = 200, \delta = 0$, and $\rho = 0.4$. 

$N = 200, \delta = 0, \rho = 0.4$
model always has the least bias when the other parameters are the same among the seven models. The variance, like the bias, does not change much with $\beta_1$ and $\beta_2$ except for the case when $p = 0.1$. Figure 2.6 shows that for all seven models the variance gets smaller as $\beta_2$ increases and it does not change with $\beta_1$ increases. The unadjusted model always has the least variance when the other parameters are the same among the seven models.

A bigger missing proportion leads to larger bias and variance. This is the same as the previous simulation with sample size 50.

To compare the baseline risk effect, a bigger baseline risk ($p = 0.4$) as an input parameter is used to run the simulation. The same result has been obtained as in the simulation of sample size 50 in the sense that bigger baseline risk lead to smaller bias as shown in Figure 2.5 and Figure 2.7. It also leads to smaller variance as shown in Figure 2.6 and Figure 2.8. This is because when data is more balanced in the $2 \times 2$ table, the variances get smaller. According to the above result, for the case when $\delta = 0$, and $N = 200$, we choose omitting the covariate when it has OR of 1 to 2.

2.4.3 Case 3: Covariate mean difference $\delta \neq 0$ and sample size $N = 50$

When the mean difference of the covariate between the treatment group and control group is not zero, the above simulation process is repeated. With $\delta = 0.4$, simulation with different values of $p(\beta_0)$, $\beta_1$, $\beta_2$ have been run for seven different models for sample size $N = 50$ with 1000 iterations at each run. The bias and variance of the seven models are shown in Figure 2.9, 2.11, 2.10 and 2.12. In Table 2.6, the trends of the bias and variance of the seven models are shown with increasing values of $\beta_1$, $\beta_2$, missing proportion $\eta$, and baseline risk $p$. The best model is suggested according to the minimum bias or variance shown in the figures.

Similar to $\delta$ equal to 0 case, the bias for the smaller sample size $N = 50$ is significantly bigger than the same parameter setting for sample size $N = 200$, as shown in Figure 2.9 and
Figure 2.9: Bias comparison of the 7 models when \( N = 50 \), \( d = 0.1, \) and \( \theta = 0.4 \).
Figure 2.10: Variance comparison of the 7 models when $N = 50$, $p = 0.1$, and $\delta = 0.4$
Figure 2.11: Bias comparison of the 7 models when $N = 50$, $p = 0.4$, and $\delta = 0.4$
Figure 2.12: Variance comparison of the 7 models when \( N = 50 \), \( p = 0.4 \), and \( \delta = 0.4 \)
Table 2.6: Bias and variance trend when one of the parameters increases
\((\delta = 0.4 \text{ and } N = 50)\)

<table>
<thead>
<tr>
<th>As one of the following increases</th>
<th>Missing</th>
<th>Baseline</th>
<th>Best Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\beta_1)</td>
<td>bigger</td>
<td>a bit smaller</td>
<td>bigger</td>
</tr>
<tr>
<td>(\beta_2)</td>
<td>smaller</td>
<td>smaller</td>
<td>smaller</td>
</tr>
<tr>
<td>(\eta)</td>
<td>bigger</td>
<td>smaller</td>
<td>adjusted model</td>
</tr>
<tr>
<td>Risk ((p))</td>
<td>smaller</td>
<td>bigger</td>
<td>unadjusted model</td>
</tr>
</tbody>
</table>

Figure 2.15. When the sample size is \(N = 50\), the trends of bias and variance when \(\beta_1, \beta_2, \) missing proportion \(\eta\), and baseline risk \(p\) increase are all similar to the \(\delta\) equal to 0 case. The only difference is that the adjusted model with complete data (the adjusted model in Figure 2.9) has the least bias when \(p = 0.1\) and \(\beta_2\) is greater than \(0.7 \times \ln 2\) and the unadjusted model (Model 7 in Figure 2.9) has the least bias when \(\beta_2\) is less than \(0.7 \times \ln 2\). However, when \(p = 0.4\), the adjusted model always has the least bias, as shown in Figure 2.11. With the increase of \(\beta_2\), the unadjusted model shows a bigger bias as can be seen in Figure 2.11.

This is because when \(\delta\) is not zero, which means it is not a RCT case, with the increasing covariate effect, omitting the covariate will lead to a biased result. While in the \(\delta\) equal to 0 case, the unadjusted model always has the least bias, as shown in Figure 2.11 and Figure 2.3.

The change from being in favor of the unadjusted model to the adjusted model as \(\delta\) increases shows the confounding effect of the covariate. As in the \(\delta\) equal to 0 case, the variance for the smaller sample size is a lot bigger than in the case of \(N = 200\) with the same parameter setting, as shown in Figure 2.10 and Figure 2.16. For sample size \(N = 50\), the variance gets smaller as \(\beta_1\) or \(\beta_2\) increases, as shown in Figure 2.10. The unadjusted model always has the least variance when the other parameters are the same among the seven models, as shown in Figure 2.10. This again agrees with Robinson’s (1991) result. The variance trends are similar to the \(\delta\) equal to 0 case, as shown in Figure 2.10 and Figure 2.2. According to the above result, for the \(\delta\) not equal to 0 case (non-RCT), when the sample size is 50, we
choose to adjust or not based on both effects. To study this in more detail, we drew another
two graphs (Figure 2.13 and Figure 2.14) that show the bias and variance trend with the
changing effect of baseline risk $p$ when $\delta$ is 0.4, which is relatively big, and $\beta_1 = 0.5 \times \ln 2$,
which is around the middle of the range of interest.

From these graphs, we can see that when the sample size is 50, the baseline risk is low
(less than 0.3), and the missing proportion is high, both the bias and variance are very large
for the adjusted model. We summarized the results from these graphs in Table 2.7. We can
see that when the sample size is 50 and $\delta$ is large, only when the baseline risk $p$ is high and
$\beta_2$ is high the adjusted model is better than the unadjusted model.

Table 2.7: Best model for different parameters when $\delta = 0.4$ and $N = 50$

<table>
<thead>
<tr>
<th>Baseline $p$</th>
<th>$\beta_2$</th>
<th>Missing%</th>
<th>Best Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline Risk</td>
<td>$\beta_2$ is low</td>
<td>Missing on $x_2$ is low</td>
<td>unadjusted</td>
</tr>
<tr>
<td>$p$ is low</td>
<td>$\beta_2$ is high</td>
<td>Missing on $x_2$ is high</td>
<td>unadjusted</td>
</tr>
<tr>
<td>($p \leq 0.3$)</td>
<td></td>
<td>Missing on $x_2$ is low</td>
<td>unadjusted/adjusted</td>
</tr>
<tr>
<td></td>
<td>$\beta_2$ is high</td>
<td>Missing on $x_2$ is high</td>
<td>unadjusted</td>
</tr>
</tbody>
</table>

| Baseline Risk| $\beta_2$ is low | Missing on $x_2$ is low | unadjusted/adjusted |
| $p$ is high  | $\beta_2$ is high | Missing on $x_2$ is high | adjusted          |
| ($p > 0.3$)  |            | Missing on $x_2$ is high | adjusted          |

2.4.4 Case 4: Covariate mean difference $\delta \neq 0$ and sample size $N = 200$

The trends of bias and variance of the seven models are shown in Figure 2.15, 2.17, 2.16 and
2.18. The bias is changing with $\beta_2$ for the seven models. As $\beta_2$ grows, the unadjusted model
has bigger bias. The bias becomes the biggest among the seven models when $\beta_2$ approaches
$\ln 2$ as shown in Figure 2.15. A bigger missing proportion did not show much effect on the
bias in Figure 2.17. This is because even with the missing proportion as much as 50%, the
Figure 2.13: Bias comparison of the 7 models when $N = 50$, $\beta_1 = 0.5 \times \ln 2$, and $\delta = 0.4$
Figure 2.14. Variance comparison of the 7 models when $N = 50, \beta_1 = 0.5 \times \ln 2$, and $\delta = 0.4$.
sample size is large enough to produce a valid result. The adjusted model without missing values always has the least bias when the other parameters are the same among the seven models.

Table 2.8: Bias and variance trend when one of the parameters increases

<table>
<thead>
<tr>
<th>As one of the following increases</th>
<th>Best Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta_1$</td>
<td>$\beta_2$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Bias trend (range [0-0.4])</th>
<th>no difference</th>
<th>smaller$^1$</th>
<th>no difference</th>
<th>smaller</th>
<th>adjusted model without missing</th>
</tr>
</thead>
</table>

| Variance trend (range [0-4]) | a bit smaller | a bit smaller | bigger | smaller | unadjusted model |

1 Especially the bias of $\beta_1$ of the unadjusted model becomes bigger comparing with other models as $\beta_2$ increases

The variance gets a little smaller as either $\beta_1$ or $\beta_2$ increases, as shown in Figure 2.16. A bigger missing proportion leads to a bigger variance. A bigger baseline risk leads to a smaller variance, as shown in Figure 2.16 and Figure 2.18. These are still the same with previous situations. The unadjusted model always has the least variance when the other parameters are the same among the seven models.

According to the above results, for the $\delta$ not equal to 0 case, when the sample size $N = 200$, we choose to adjust for the covariate or not based on the overall effect of bias and variance. When $\beta_2$ is closer to ln2, the bias has a non-ignorable effect, and we choose the adjusted model. When $\beta_2$ is closer to 0, the unadjusted model can be considered.

Similar to the $\delta$ equal to 0 case, two graphs (Figure 2.19 and Figure 2.20) were drawn to show the bias and variance changes with the baseline risk $p$ when $\delta$ is 0.4, which is relatively
Figure 2.15: Bias comparison of the 7 models when $N = 200$, $p = 0.1$, and $\delta = 0.4$.
Figure 2.16: Variance comparison of the 7 models when $N = 200$, $\delta = 0.4$, and $\rho = 0.1$. 

$N = 200$, $\delta = 0.4$, $\rho = 0.1$
Figure 2.17: Bias comparison of the 7 models when $N = 200$, $\delta = 0.4$, and $p = 0.4$. 

- $\beta_2 = 0.1 \log 2$
- $\beta_2 = 0.2 \log 2$
- $\beta_2 = 0.3 \log 2$
- $\beta_2 = 0.4 \log 2$
- $\beta_2 = 0.5 \log 2$
- $\beta_2 = 0.6 \log 2$
- $\beta_2 = 0.7 \log 2$
- $\beta_2 = 0.8 \log 2$
- $\beta_2 = 0.9 \log 2$
- $\beta_2 = \log 2$
Figure 2.18: Variance comparison of the 7 models when \( N = 200, \; \delta = 0.4, \; p = 0.4 \)
big, and $\beta_1 = 0.5 \times \ln 2$, which is in the middle of the range of interest.

Table 2.9: Best model for different parameters when $\delta = 0.4$ and $N = 200$

<table>
<thead>
<tr>
<th>Baseline $p$</th>
<th>$\beta_2$</th>
<th>Missing%</th>
<th>Best Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline Risk</td>
<td>$\beta_2$ is low</td>
<td>Missing on $x_2$ is low</td>
<td>adjusted</td>
</tr>
<tr>
<td>$p$ is low $(p \leq 0.2)$</td>
<td>$\beta_2$ is high</td>
<td>Missing on $x_2$ is high</td>
<td>unadjusted</td>
</tr>
<tr>
<td>Baseline Risk</td>
<td>$\beta_2$ is low</td>
<td>Missing on $x_2$ is low</td>
<td>adjusted</td>
</tr>
<tr>
<td>$p$ is low $(p &gt; 0.2)$</td>
<td>$\beta_2$ is high</td>
<td>Missing on $x_2$ is high</td>
<td>adjusted</td>
</tr>
<tr>
<td>Baseline Risk</td>
<td>$\beta_2$ is low</td>
<td>Missing on $x_2$ is low</td>
<td>adjusted</td>
</tr>
<tr>
<td>$p$ is low $(p \leq 0.2)$</td>
<td>$\beta_2$ is high</td>
<td>Missing on $x_2$ is high</td>
<td>adjusted</td>
</tr>
</tbody>
</table>

Table 2.9 was summarized from Figure 2.19 and 2.20. We can see that when the sample size is 200 and $\delta$ is large, only when the baseline risk is low, $\beta_2$ is low, and missing is high, the unadjusted model is preferred rather than the adjusted model. In other words, if either the baseline risk $p$ is high or $\beta_2$ is high, we can use the adjusted model even if missing is high because when the sample size is large and either $p$ or $\beta_2$ is large, even if the covariate was 50% missing, we still have a big enough sample to produce a valid result.

2.5 Summary of Simulation Results

2.5.1 Sample size effect

A bigger sample size will lead to more precise estimates. The variance of $\hat{\beta}_1$ decreases significantly as the sample size increases.
Figure 2.10: Bias comparison of the 7 models when $N = 200$, $\beta_1 = 0.5 \times \ln 2$, and $\delta = 0.4$.
Figure 2.2: Variance comparison of the $\tau$ modes when $N=200$, $\beta_1 = 0.5 \times \ln 2$, and $\delta = 0.4$.
2.5.2 Baseline risk effect

A bigger baseline risk, such as a value close to 0.5, will make the $2 \times 2$ table more balanced therefore a smaller sample will be adequate. More balanced data will lead to less variation.

2.5.3 Covariate missing effect

Both the bias and the variance get bigger as the covariate missing proportion increases because the effect is similar to that of a smaller sample. However, when the sample size is sufficiently large, such as $N = 200$, and if the baseline risk $p$ is big too, then the effect of covariate missing will diminish.

2.5.4 Covariate mean difference effect

When $\delta$ is zero, which means the predictive covariate $x_2$ is a non-confounding variable so the unadjusted model will not have bigger bias than the adjusted model and the variance will be smaller than the adjusted model with complete data according to both Robinson (1991) and the above simulation result. With missing data on the covariate, the precision will be worse if we use the adjusted model. Therefore, for the $\delta$ equal to 0 case, the unadjusted model is suggested. There is no need to collect this baseline information if the mean difference of the covariate between different treatment groups is close to 0. When $\delta$ is not equal to zero, the unadjusted model has a bigger bias as the covariate effect $\beta_2$ increases. This is because when the covariate has correlation with the outcome, adjusting for the covariate will give more valid results. However, the variance gets smaller as $\beta_2$ increases and the trends of variance are relying on the sample size more than relying on $\beta_2$. This is when we need to choose to adjust for covariate or not based on the overall effect of $\delta$, the sample size, the baseline risk and $\beta_2$. Similarly, when $\delta$ is not zero, the unadjusted model will have a bigger bias as $\delta$
increases. In this case, the adjusted model will be in more favor of. However, even with a very big $\delta$ (such as 0.4), simulation result showed for a small sample, only when the baseline risk and $\beta_2$ are both large, the adjusted model is a better choice. So for medium or small sample study, a bigger baseline risk and $\beta_2$ are required to use the adjusted model. As the sample size gets larger, it will gain back the precision loss from adjusting for the covariate. The variance of $\hat{\beta}_1$ decrease significantly as the sample size gets larger. Therefore when $\delta$ is big and the sample size is large, the adjusted model is preferred except for the case when the baseline risk is low and $\beta_2$ is low and baseline covariate missing is high.
Chapter 3

Real Applications

To check if the simulation reflects real case, data from two clinical trials have been used.

3.1 SPRINT study

SPRINT study is intended to compare reamed and unreamed intramedullary nailing for treatment of tibial shaft fractures. It is a multicenter blinded randomized trial. The primary outcome event is an expanded primary composite outcome decided after first interim analysis. It includes dynamization of fracture in the operating room or in the outpatient clinic, removal of locking screws because of hardware breakage or loosening; autodynamization; fasciotomy and drainage of hematomas.

SPRINT data was analyzed based on different treatment groups, either reamed or unreamed nailing, and open or closed fracture, because of the interaction observed. As the result of the study suggested (The SPRINT investigators, 2008), among 1226 participants completed one year of follow-up, 622 patients were randomized to reamed nailing and 604 were randomized to unreamed nailing. Among all patients, 105 in the reamed nailing group
and 114 in the undreamed nailing group experienced a primary outcome event (relative risk, 0.90; 95% CI, 0.71 to 1.15). In patients with closed fractures, 45(11%) of 416 in the reamed nailing group and 68(17%) of 410 in the unreamed nailing group experienced a primary event (relative risk, 0.67; 95% CI, 0.47 to 0.96). In patients with open fractures, 60(19%) of 206 in the reamed nailing group and 46(24%) of 194 in the undreamed nailing group experienced a primary event (relative risk, 1.27; 95% CI, 0.91 to 1.78). The study shows a possible benefit for reamed intramedullary nailing in the patients with close fractures, however, no difference in patients with open fractures.

To show the consistency between the simulation and this real case, baseline binary variable "smoke" is chosen to test the effect of missing. Because of the interaction, analysis was based on two groups. One is open fracture group and the other is close fracture group.

Table 3.1: Standard error of \( \hat{\beta}_1 \) and its relative precision of different models adjusting or not adjusting for SMOKE on OPEN fracture SPRINT data comparing with the simulation using the same parameter

<table>
<thead>
<tr>
<th></th>
<th>Unadjusted</th>
<th>Adjusted</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Complete Data</td>
<td>Missing rate of covariate SMOKE</td>
</tr>
<tr>
<td></td>
<td>Complete data</td>
<td>Complete data</td>
</tr>
<tr>
<td>( \hat{\beta}_1 )</td>
<td>0.27</td>
<td>0.275</td>
</tr>
<tr>
<td>Result from</td>
<td>se(( \hat{\beta}_1 ))</td>
<td>0.2288</td>
</tr>
<tr>
<td>SPRINT</td>
<td>RP&lt;sup&gt;1&lt;/sup&gt;</td>
<td>0.99</td>
</tr>
<tr>
<td></td>
<td>se(( \hat{\beta}_1 ))</td>
<td>0.21</td>
</tr>
<tr>
<td>Simulation</td>
<td>RP&lt;sup&gt;1&lt;/sup&gt;</td>
<td>0.99</td>
</tr>
</tbody>
</table>

<sup>1</sup> Relative precision, RP of model a to model b is variance of model b divided by variance of model a.

The result from open fracture group adjusted by smoke is in Table 3.1. In this analysis, baseline risk \( p = 0.264(105/398) \) for open fracture and sample size \( N = 398 \) are unchanged. Mean difference between two treatment groups \( \delta \) is \(-0.028(64/194 - 73/204)\). From the logistic regression with complete data adjusted for covariate smoke, \( \hat{\beta}_1 \) is 0.275 and \( \hat{\beta}_2 \) is
Table 3.2: Standard error of \( \hat{\beta}_1 \) and its relative precision of different models adjusting or not adjusting for SMOKE on CLOSE fracture SPRINT data comparing with the simulation using the same parameter

<table>
<thead>
<tr>
<th></th>
<th>Complete Data</th>
<th>Missing rate of covariate SMOKE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Complete</td>
<td>Complete</td>
</tr>
<tr>
<td></td>
<td>data</td>
<td>data</td>
</tr>
<tr>
<td>( \hat{\beta}_1 )</td>
<td>0.49</td>
<td>0.479</td>
</tr>
<tr>
<td>Result from</td>
<td></td>
<td></td>
</tr>
<tr>
<td>se(( \hat{\beta}_1 ))</td>
<td>0.206</td>
<td>0.207</td>
</tr>
<tr>
<td>SPRINT</td>
<td>RP(^1)</td>
<td>1</td>
</tr>
<tr>
<td>se(( \hat{\beta}_1 ))</td>
<td>0.201</td>
<td>0.202</td>
</tr>
<tr>
<td>Simulation</td>
<td>RP(^1)</td>
<td>0.988</td>
</tr>
</tbody>
</table>

1 Relative precision, RP of model a to model b is variance of model b divided by variance of model a

0.192. The standard error of \( \hat{\beta}_1 \) as well as the relative precision are presented in the Table 3.1. These results should roughly match with the simulation result of sample size \( N = 400 \), \( p = 0.3 \), \( \delta = 0 \), \( \beta_1 = 0.275 \), and \( \beta_2 = 0.192 \) case. Using them as input parameters to our simulation program and result are put into the Table 3.1 as well.

For close fracture group, we did the same analysis adjusted for the smoke variable. In Table 3.2, baseline risk \( p \) is 0.12(45/369) and sample size is \( N = 823 \). Mean difference from different treatment groups \( \delta \) is \(-0.04(127/404 - 142/409)\). Then \( \hat{\beta}_1 = 0.5 \) and \( \hat{\beta}_2 = 0.34 \) are obtained from logistic regression adjusted for complete smoke variable. This case should roughly match our simulation result for \( N = 800 \), \( \delta = 0 \) and \( p = 0.1 \), with \( \beta_1 = 0.5 \) and \( \beta_2 = 0.34 \) case. Then the simulation program was run using them as input parameters and the result is put in the Table 3.2. Compared the standard error of \( \hat{\beta}_1 \) and the relative precision of \( \hat{\beta}_1 \) of the SPRINT data set with the simulation result, it is verified that our simulation result matches with the real application.
3.2 RESTORE study

RESTORE is a multicenter randomized trial with 62 patients to compare the effectiveness of a standardized method of rotator cuff repair with or without augmentation using porcine small intestine submucosa (SIS) in patients with large rotator cuff tears. The study outcome is failure defined as a remaining full thickness defect greater than 5 mm in any dimension (Bryant).

This study has small sample size. There are 62 patients enrolled and three of them have missing outcome results. Out of those 3 patients, one is from allocation 1 and two are from allocation 2. Therefore, 59 patients can be used for analysis.

Three covariates “age”, “antpos” and “medlat” will be used for demonstration. Covariate “antpos” is the size of tear from anterior to posterior prior to repair. Covariate “medlat” is the size of tear from medial to lateral prior to repair.

In order to have direct comparison with our simulation result, in which case only the binary covariate is used, we converted these three continuous covariates to binary variables. The method we used is to set the variable to 1 if it is greater than mean and set it to 0 elsewhere. After this continuous to binary conversion, the following results are produced.

Binary Age-adjusted results are in Table 3.3:

Simulation parameters in Table 3.3 for AGE are $N = 50$, $p = 0.55$, $\delta = 0.1$, $\hat{\beta}_1 = 0.32$, $\hat{\beta}_2 = 1.51$.

Binary ANTPOS-adjusted results are in Table 3.4 and simulation parameters for ANTPOS are $N = 50$, $p = 0.55$, $\delta = -0.15$, $\hat{\beta}_1 = 0.74$, $\hat{\beta}_2 = 1.54$.

Binary MEDLAT-adjusted results are in Table 3.5 and simulation parameters for MEDLAT are $N = 50$, $p = 0.55$, $\delta = 0.084$, $\hat{\beta}_1 = 0.38$, $\hat{\beta}_2 = 1.39$. 

47
Table 3.3: Standard error of $\hat{\beta}_1$ and its relative precision of different models adjusting or not adjusting for AGE on RESTORE data comparing with the simulation using the same parameter

<table>
<thead>
<tr>
<th></th>
<th>Unadjusted</th>
<th>Adjusted</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Complete Data</td>
<td>Missing rate of covariate AGE</td>
</tr>
<tr>
<td></td>
<td>Complete data</td>
<td>Complete data</td>
</tr>
<tr>
<td>$\hat{\beta}_1$</td>
<td>0.45</td>
<td>0.32</td>
</tr>
<tr>
<td>Result from $se(\hat{\beta}_1)$</td>
<td>0.54</td>
<td>0.58</td>
</tr>
<tr>
<td>RESTORE $RP^1$</td>
<td>0.87</td>
<td>0.77</td>
</tr>
<tr>
<td>$se(\hat{\beta}_1)$</td>
<td>0.32</td>
<td>0.33</td>
</tr>
<tr>
<td>Simulation $RP^1$</td>
<td>0.94</td>
<td>0.57</td>
</tr>
</tbody>
</table>

1 Relative precision, $RP$ of model a to model b is variance of model b divided by variance of model a

It’s easy to see that our simulation result is quite close to the analysis result from real clinical trial data even smaller sample size made the variation a little bigger. This again proved the validity of our simulation.
Table 3.4: Standard error of $\hat{\beta}_1$ and its relative precision of different models adjusting or not adjusting for ANTPOS on RESTORE data comparing with the simulation using the same parameter

<table>
<thead>
<tr>
<th></th>
<th>Unadjusted Complete Data</th>
<th>Complete data</th>
<th>Adjusted Missing rate of covariate ANTPOS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>10%</td>
</tr>
<tr>
<td>$\hat{\beta}_1$</td>
<td>0.45</td>
<td>0.74</td>
<td>1.12</td>
</tr>
<tr>
<td>Result from</td>
<td>$se(\hat{\beta}_1)$</td>
<td>0.54</td>
<td>0.65</td>
</tr>
<tr>
<td>RESTORE</td>
<td>$RP^1$</td>
<td>0.85</td>
<td>0.70</td>
</tr>
<tr>
<td>Simulation</td>
<td>$se(\hat{\beta}_1)$</td>
<td>0.34</td>
<td>0.44</td>
</tr>
<tr>
<td></td>
<td>$RP^1$</td>
<td>0.91</td>
<td>0.61</td>
</tr>
</tbody>
</table>

1 Relative precision, $RP$ of model a to model b is variance of model b divided by variance of model a

Table 3.5: Standard error of $\hat{\beta}_1$ and its relative precision of different models adjusting or not adjusting for MEDLAT on RESTORE data comparing with the simulation using the same parameter

<table>
<thead>
<tr>
<th></th>
<th>Unadjusted Complete Data</th>
<th>Complete data</th>
<th>Adjusted Missing rate of covariate MEDLAT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>10%</td>
</tr>
<tr>
<td>$\hat{\beta}_1$</td>
<td>0.45</td>
<td>0.38</td>
<td>0.87</td>
</tr>
<tr>
<td>Result from</td>
<td>$se(\hat{\beta}_1)$</td>
<td>0.54</td>
<td>0.64</td>
</tr>
<tr>
<td>RESTORE</td>
<td>$RP^1$</td>
<td>0.89</td>
<td>0.72</td>
</tr>
<tr>
<td>Simulation</td>
<td>$se(\hat{\beta}_1)$</td>
<td>0.43</td>
<td>0.52</td>
</tr>
<tr>
<td></td>
<td>$RP^1$</td>
<td>0.96</td>
<td>0.70</td>
</tr>
</tbody>
</table>

1 Relative precision, $RP$ of model a to model b is variance of model b divided by variance of model a
Chapter 4

Discussion and Future Work

The purpose of this thesis is to explore the characteristics of the logistic regression with binary covariate and to make some suggestion on adjusting or not when missingness is present. It is not intended to make a strict guideline on how to analyze the data. In real application, statisticians still have to analysis the data according to different situations. White (2005) suggested that imputation method has better precision than both the unadjusted and the adjusted methods. That could still be the way to go in real studies.

Sample size of 100 has been explored after the simulations on sample size of 50 and 200. It is shown that 100 samples has the characteristics that is in between of 50 and 200 samples. It will be harder to decide which model to use than in the 50 or 200 samples because the tradeoff is not as obvious.

Continuous covariate has also been explored and the results are very similiar to those of binary covariate.

The largest $\beta_2$ is $\ln 2$, which means odds ratio of 2 in the simulation. In the real world, although not very common, there are cases that exceed this value. In the case that covariate effect is bigger than treatment effect, covariate can not be omitted.
Categorical covariates that have more than two levels could be explored in the simulation in the future. More baseline covariates can be added to study the effect. In addition, mixture of different type of covariates can be studied. More clinical trials in practice may be needed to verify the findings.
References


Bryant, D., RESTORE Pilot study. (not published).


Trials 28, 242-8.


Appendix

A. Flowchart for Simulation

Figure 4.1 is the flowchart for simulation.

B. Source code in R for Simulation

large_genres_delta is the code to prepare data set to call by new_aabbdraw and new_aabbdraw_var functions to draw figures 2.1-2.12, 2.15-2.18.

```r
> large_genres_delta<-function (delta=0.4,pp=1,NN=200)
> { options(warn=-1)
> p=pp/10
> betaO=log(p/(1-p))
> beta1=(1:10)/10*log(2)
> beta2=(1:10)/10*log(2)
> del=delta
> n=NN
> input = matrix(0,nr=length(n)*length(del)*length(betaO)*
> length(beta1)*length(beta2),nc=26)
> colnames(input)=c("N","delta","b0","b1","b2","b1m0var","b1m0mse",
> "b1mObias","b1m1var","b1m1mse","b1m1bias","b1m2var","b1m2mse",
> "b1m2bias","b1m2mse")
> "b1m2bias","b1m2mse")
> }
Sample Size $N = 50,200$
Baseline Risk $p = 0.1, 0.2, 0.3, 0.4, 0.5$
mean difference($\delta$) = 0, 0.1, 0.2, 0.3, 0.4

In each ($N, \delta, p$) combination
Generate different combination of $p_1$ and $p_2$ (10x10 combinations in total)

Iterate 1000 times

For each combination of $N$, $\delta$, ($p$, $p_1$ and $p_2$)
Generate $N$ samples, set
missing for 10-50%.
Calculate $p_1$ estimate for 7 different
models, get 7 estimated $p_1$ as
output each time.

Get a matrix of
1000x7 values

Calculate variance, MSE, bias for 7 models
based on 1000 estimates from one specific
$N$, $\delta$, ($p$, $p_1$ and $p_2$).

Generated a dataset of 100 rows for
each $p_1$, $p_2$ combination for one
specific ($N, \delta, p$).

In the end, 50 datasets
could be generated. Each
has 100 rows.

Figure 4.1: Flowchart for simulation
"b1m2bias","b1m3var","b1m3mse","b1m4bias","b1m4mse","b1m4var","b1m5bias","b1m5mse","b1m5var","b1m0bias","b1m0var","b1m0mse","b1m0xvar","b1m0xmse"

input[,1] = rep(n, each=length(beta0)*length(beta1)*length(beta2))
input[,2] = rep(del, each=length(beta0)*length(beta1)*length(beta2))
input[,3] = rep(rep(beta0, each=length(beta1)*length(beta2)), length(beta0))
for(i in 1:nrow(input)){
  para = c(input[i,1],input[i,2],input[i,3],input[i,4],input[i,5])
  input[i,6:26] = large_tloop_delta(para)
  print(i)
}
options(warn=0) return(input)

> large_tloop_delta<-function (para)
  {
    ntr = para[1]*0.5
    paran = c(para,ntr)
    # nr is number of iterations
    nr = 1000
    inpara = matrix(rep(paran,nr),nrow=nr, byrow=T)
    tp = t(apply(inpara,i,large_testkai))
    b1 = para[4]
    # (mean(bihat)-b1)^2+var(bihat) == mse
    # and bias
    return(c( var(tp[,1]),(mean(tp[,1])-b1)^2+var(tp[,1]),
      (mean(tp[,1])-b1), var(tp[,3]),(mean(tp[,3])-b1)^2+var(tp[,3]),
      (mean(tp[,3])-b1), var(tp[,5]),(mean(tp[,5])-b1)^2+var(tp[,5]),
      (mean(tp[,5])-b1), var(tp[,7]),(mean(tp[,7])-b1)^2+var(tp[,7]),
      (mean(tp[,7])-b1), var(tp[,9]),(mean(tp[,9])-b1)^2+var(tp[,9]),
      (mean(tp[,9])-b1), var(tp[,11]),(mean(tp[,11])-b1)^2+var(tp[,11]),
      (mean(tp[,11])-b1), var(tp[,13]),(mean(tp[,13])-b1)^2+var(tp[,13]),
      (mean(tp[,13])-b1))
  }

> large_testkai<-function (para)
  {
    ntot=para[1]
    delta=para[2]
    b0=para[3]
    b1=para[4]
    b2=para[5]
    ntr=para[6]
ind = 1
# loop until one converged is found
while (ind==1){

# generate X1 as
# number of placebo group as 0 (ntot-ntr)
# number of treatment group as 1 (ntr)
x1 = rep(c(0, 1), c(ntot - ntr, ntr))

# Change to the two lines under if x2 is continuous
# generate X2~N(ntr, delta, 10)
# x2 = c(rnorm((ntot - ntr), 0, 10), rnorm(ntr, delta, 10))

# generate x2
x2<- c(rbinom((ntot-ntr), 1, 0.5),rbinom(ntr, 1, (delta+0.5)))

# X2ml to X2m6 are 0%-50% missing randomly assigned
x2m1 = x2
x2m1[sample(ntot, 0 * ntot)] <- NA
x2m2 = x2
x2m2[sample(ntot, 0.1 * ntot)] <- NA
x2m3 = x2
x2m3[sample(ntot, 0.2 * ntot)] <- NA
x2m4 = x2
x2m4[sample(ntot, 0.3 * ntot)] <- NA
x2m5 = x2
x2m5[sample(ntot, 0.4 * ntot)] <- NA
x2m6 = x2
x2m6[sample(ntot, 0.5 * ntot)] <- NA

# mu is the mean value of outcome variable
mu <- exp(b0 + b1 * x1 + b2 * x2)/(1 + exp(b0 + b1 * x1 + b2 * x2))

# randomly generate outcome variable y~binom(ntot,1,mu)
y <- rbinom(ntot, 1, mu)

# create data frame(dat) with y, x1,x2m1,x2m2,....,x2m6 columns, and ntot rows
dat=data.frame(y = y, x1 = x1, x2m1 = x2m1, x2m2 = x2m2, x2m3=x2m3,
x2m4 = x2m4,x2m5=x2m5,x2m6=x2m6)

# run logistic regression with x1 and x2 complete
fit1 = summary(glm(y~factor(x1)+x2m1,family=binomial("logit"), data=dat))

# run logistic regression with x1 and x2 10% missing
fit2 = summary(glm(y~factor(x1)+x2m2,family=binomial("logit"), data=dat))
# run logistic regression with x1 and x2 20% missing
fit3 = summary(glm(y~factor(x1)+x2m3,family=binomial("logit"), data=dat))

# run logistic regression with x1 and x2 30% missing
fit4 = summary(glm(y~factor(x1)+x2m4,family=binomial("logit"), data=dat))

# run logistic regression with x1 and x2 40% missing
fit5 = summary(glm(y~factor(x1)+x2m5,family=binomial("logit"), data=dat))

# run logistic regression with x1 and x2 50% missing
fit6 = summary(glm(y~factor(x1)+x2m6,family=binomial("logit"), data=dat))

# run logistic regression with x1 only
fit7 = summary(glm(y~factor(x1),family=binomial("logit"), data=dat))

# return regression result (b1 estimate, (b1 estimate - b1)^2)
# based on sample size ntot=50, with different missing rates
temp = c(fit$coef[2], (fit$coef[2]-b1)^2,fit2$coef[2],
(fit2$coef[2]-b1)^2,fit3$coef[2], (fit3$coef[2]-b1)^2,fit4$coef[2],
(fit4$coef[2]-b1)^2,fit5$coef[2], (fit5$coef[2]-b1)^2,fit6$coef[2],
(fit6$coef[2]-b1)^2,fit7$coef[2], (fit7$coef[2]-b1)^2)

bv = sum(temp>1000)
if (bv==0) ind = 0
}

# end of while loop
return(temp)
}

C. Source code in R for Graphs

Function new_aabbdraw is used to draw Figures 2.1, 2.3, 2.5, 2.7, 2.9, 2.11, 2.13, 2.15, 2.17, 2.19.

new_aabbdraw<- function (indat=N200_I1000_p1_d04) {
  library(lattice)
  dat = as.data.frame(indat)
  dat1 = dat[,c(4,5,8)]
  dat2 = dat[,c(4,5,11)]
  dat3 = dat[,c(4,5,14)]
  dat4 = dat[,c(4,5,17)]
  dat5 = dat[,c(4,5,20)]
  dat6 = dat[,c(4,5,23)]
dat7 = dat[,c(4,5,26)]
names(dat1) = c("Beta1", "Beta2", "Bias")
names(dat2) = c("Beta1", "Beta2", "Bias")
names(dat3) = c("Beta1", "Beta2", "Bias")
names(dat4) = c("Beta1", "Beta2", "Bias")
names(dat5) = c("Beta1", "Beta2", "Bias")
names(dat6) = c("Beta1", "Beta2", "Bias")
names(dat7) = c("Beta1", "Beta2", "Bias")

datcom = rbind(dat1, dat2, dat3, dat4, dat5, dat6, dat7)

datcom$rnisslbl = rep(c("Cornplete 2", "Miss 10\%", "Miss 20\%", "Miss 30\%", "Miss 40\%", "Miss 50\%", "Complete 1"), each=100)
datcom$b2lbl = rep(c("0.1*log2", "0.2*log2", "0.3*log2", "0.4*log2", "0.5*log2", "0.6*log2", "0.7*log2", "0.8*log2", "0.9*log2", "log2"), 70)
datcom = datcom[order(datcom$b2lbl),]

pdf(file="C:\Kai\thesis\new_version\largersample\csAug19\Bias_N200_pl_d04.pdf", 10, 7, "a4r")

drawkey = list(space="right", text=list(modelbl), lines=list(col=c("black", "red", "green", "orange", "purple", "red", "blue")), lwd=c(2,1,1,1,1,1,3), lty=c(1,2,3,4,5,6,3))

ic = xyplot(Bias ~ Beta1 Ifactor(b2lbl), data=datcom, layout=c(5,2),
xlab=expression(beta[1]), ylab="Bias", main=expression(paste("N = 200, "delta," = 0.4, P = 0.1")), key=drawkey, strip = strip.custom(factor.levels = expression(paste(b2lbl, " = 0.1*log2")),
paste(b2lbl, " = 0.2*log2"), paste(b2lbl, " = 0.3*log2"),
paste(b2lbl, " = 0.4*log2"), paste(b2lbl, " = 0.5*log2"),
paste(b2lbl, " = 0.6*log2"), paste(b2lbl, " = 0.7*log2"),
paste(b2lbl, " = 0.8*log2"), paste(b2lbl, " = 0.9*log2"),
paste(b2lbl, " = log2"))

panel.xyplot(x[1:10], y[1:10], type="1", col="black", lty=1, lwd=2)
panel.xyplot(x[11:20], y[11:20], type="1", col="red", lty=2, lwd=1)
panel.xyplot(x[21:30], y[21:30], type="1", col="green", lty=3, lwd=1)
panel.xyplot(x[31:40], y[31:40], type="1", col="orange", lty=4, lwd=1)
panel.xyplot(x[41:50], y[41:50], type="1", col="purple", lty=5, lwd=1)
panel.xyplot(x[51:60], y[51:60], type="1", col="red", lty=6, lwd=1)
panel.xyplot(x[61:70], y[61:70], type="1", col="blue", lty=3, lwd=3)

pic = pic[c(6:10,1:5)]
print(pic)
dev.off()
return(pic)
Function `new_aabbdraw_var` is used to draw Figures 2.2, 2.4, 2.6, 2.8, 2.10, 2.12, 2.16, 2.18.

```r
new_aabbdraw_var <- function (indat=N200_Il000_pl_d04) {
  library(lattice)
  dat = as.data.frame(indat)
  dat1 = dat[,c(4,5,6)]
  dat2 = dat[,c(4,5,9)]
  dat3 = dat[,c(4,5,12)]
  dat4 = dat[,c(4,5,15)]
  dat5 = dat[,c(4,5,18)]
  dat6 = dat[,c(4,5,21)]
  dat7 = dat[,c(4,5,24)]
  names(dat1) = c("Beta1", "Beta2", "Variance")
  names(dat2) = c("Beta1", "Beta2", "Variance")
  names(dat3) = c("Beta1", "Beta2", "Variance")
  names(dat4) = c("Beta1", "Beta2", "Variance")
  names(dat5) = c("Beta1", "Beta2", "Variance")
  names(dat6) = c("Beta1", "Beta2", "Variance")
  names(dat7) = c("Beta1", "Beta2", "Variance")
  datcom = rbind(dat1, dat2, dat3, dat4, dat5, dat6, dat7)
  datcom$misslbl = rep(c("Complete 2", "Miss 10\%", "Miss 20\%", "Miss 30\%", "Miss 40\%", "Miss 50\%", "Complete 1"), each=100)
  datcom$b2lbl = rep(c("0.1*log2", "0.2*log2", "0.3*log2", "0.4*log2", "0.5*log2", "0.6*log2", "0.7*log2", "0.8*log2", "0.9*log2", "log2"), 70)
  datcom = datcom[order(datcom$b2lbl),]
  pdf(file="C:\Kai\thesis\new_version\largersample\csAug19\Variance_N200_pLd04.pdf", 10, 7, "a4r")
  drawkey=list(space="right", text=list(modellbl), lines=list(col=c("black", "red", "green", "orange", "purple", "red", "blue")),
  lwd=c(2,1,1,1,1,1,3), lty=c(1,2,3,4,5,6,3))
  pic=xyplot(Variance ~ Beta1 | factor(b2lbl), data=datcom, layout=c(5,2),
  xlab=expression(beta[l]), ylab="Variance", main= expression(paste("N = 200, \"delta\", 0.4, \"p = 0.1\"")),
  key=drawkey, strip = strip.custom(factor.levels = expression(paste("N = 200, \"delta\", 0.4, \"p = 0.1\"")),
  panel=function(x,y){
    panel.xyplot(x[1:10], y[1:10], type="l", col="black", lty=1, lwd=2)
    panel.xyplot(x[11:20], y[11:20], type="l", col="red", lty=2, lwd=1)
    panel.xyplot(x[21:30], y[21:30], type="l", col="green", lty=3, lwd=1)
    panel.xyplot(x[31:40], y[31:40], type="l", col="orange", lty=4, lwd=1)
  })
```
Function new_ppppdraw is used to draw Figures 2.14, 2.19. temp1 is made by combining 5 different p values result data sets into one big data set and adding column p accordingly. temp_N50_b1_5 is part of temp1 where $\beta_1 = 0.5 \cdot ln(2)$.

```r
new_ppppdraw <- function (indat= temp_N50_b1_5) {
  library(lattice)
  dat = indat
  dat1 = dat[,c(27,5,8)]
  dat2 = dat[,c(27,5,11)]
  dat3 = dat[,c(27,5,14)]
  dat4 = dat[,c(27,5,17)]
  dat5 = dat[,c(27,5,20)]
  dat6 = dat[,c(27,5,23)]
  dat7 = dat[,c(27,5,26)]
  names(dat1) = c("p", "Beta2", "Bias")
  names(dat2) = c("p", "Beta2", "Bias")
  names(dat3) = c("p", "Beta2", "Bias")
  names(dat4) = c("p", "Beta2", "Bias")
  names(dat5) = c("p", "Beta2", "Bias")
  names(dat6) = c("p", "Beta2", "Bias")
  names(dat7) = c("p", "Beta2", "Bias")
  datcom = rbind(dat1,dat2,dat3,dat4,dat5,dat6,dat7)
  datcom$misslbl = rep(c("Complete 2","Miss 10%","Miss 20%","Miss 30%","Miss 40%","Miss 50%","Complete 1"),each=50)
  datcom$b2lbl = rep(c("Beta2 = 0.1*log2","Beta2 = 0.2*log2","Beta2 = 0.3*log2","Beta2 = 0.4*log2","Beta2 = 0.5*log2","Beta2 = 0.6*log2","Beta2 = 0.7*log2","Beta2 = 0.8*log2","Beta2 = 0.9*log2","Beta2 = log2"),35)
  datcom=modelcom[order(datcom$misslbl),]
  pdf(file="C:\Kai\thesis\new_version\largersample\csAug19\Bias_N50_byp_b1_5.pdf", 10, 7, "a4r")
  modellbl = c("Adjusted","Model 2","Model 3","Model 4","Model 5")
  dev.off()
  return(datcom)
}
```
Function new_pppdraw_var is used to draw Figures 2.15, 2.20.

new_pppdraw_var <- function (indat=temp_N50_b1_5) {
  library(lattice)
  dat = indat
  dat1 = dat[,c(27,5,6)]
  dat2 = dat[,c(27,5,9)]
  dat3 = dat[,c(27,5,12)]
  dat4 = dat[,c(27,5,15)]
  dat5 = dat[,c(27,5,18)]
  dat6 = dat[,c(27,5,21)]
  dat7 = dat[,c(27,5,24)]
  names(dat1) = c("p","Beta2","Variance")
  names(dat2) = c("p","Beta2","Variance")
  names(dat3) = c("p","Beta2","Variance")
  names(dat4) = c("p","Beta2","Variance")
  names(dat5) = c("p","Beta2","Variance")
  names(dat6) = c("p","Beta2","Variance")
  names(dat7) = c("p","Beta2","Variance")
  datcom = rbind(dat1,dat2,dat3,dat4,dat5,dat6,dat7)
datcom$misslbl = rep(c("Complete 2", "Miss 10%", "Miss 20%", "Miss 30%", "Miss 40%", "Miss 50%", "Complete 1"), each=50)
datcom$b2lbl = rep(c("Beta2 = 0.1*log2", "Beta2 = 0.2*log2", "Beta2 = 0.3*log2", "Beta2 = 0.4*log2", "Beta2 = 0.5*log2", "Beta2 = 0.6*log2", "Beta2 = 0.7*log2", "Beta2 = 0.8*log2", "Beta2 = 0.9*log2", "Beta2 = log2"), 35)
datcom = datcom[order(datcom$b2lbl),]

pdf(file="C:\Kai\thesis\new_version\largersample\csSept5th\Variance_NS50_byp_b2l_bl_S.pdf", 10, 7, "a4r")
modellbl = c("Adjusted", "Model 2", "Model 3", "Model 4", "Model 5", "Model 6", "Unadjusted")
drawkey = list(space="right", text=list(modellbl), lines=list(col=c("black", "red", "green", "orange", "purple", "red", "blue")), lwd=c(2,1,1,1,1,3,1), lty=c(1,2,3,4,5,6,3))
datcom$model = rep(rep(modellbl, each=5), 10)
drawkey = list(space="right", text=list(modellbl), lines=list(lty=c(1,2,3,4,5,6,3), col=c("black", "red", "green", "orange", "purple", "red", "blue")), lwd=c(2,1,1,1,1,3,1))

pic = xyplot(Variance ~ plfactor(b2lbl), data=datcom, layout=c(5,2), xlab="p", ylab="Variance", main=expression(paste("N = 50, \"", delta, \" = 0.4, \"", beta[1], \" = 0.5*ln2\"))), group=model, type="b", col=c("black", "red", "green", "orange", "purple", "red", "blue"), lty=c(1,2,3,4,5,6,3), key=drawkey, strip = strip.custom(factor.levels = expression(paste(b2lbl[1], " = 0.1*log2"), paste(b2lbl[2], " = 0.2*log2"), paste(b2lbl[3], " = 0.3*log2"), paste(b2lbl[4], " = 0.4*log2"), paste(b2lbl[5], " = 0.5*log2"), paste(b2lbl[6], " = 0.6*log2"), paste(b2lbl[7], " = 0.7*log2"), paste(b2lbl[8], " = 0.8*log2"), paste(b2lbl[9], " = 0.9*log2"), paste(b2lbl[10], " = log2")) ), lwd=c(2,1,1,1,1,3))

pic = pic[c(6:10,1:5)]
print(pic)
dev.off()
return(pic)