2016

Model choices for complex survey analysis

Preeya Riyapan
University of Wollongong

Model Choices for Complex Survey Analysis

A thesis submitted in fulfilment of

the requirements for the award of the degree

Doctor of Philosophy

from

University of Wollongong

by

Preeya Riyapan BSc, MSc

School of Mathematics and Applied Statistics

2016

CERTIFICATION

I, Preeya Riyapan, declare that this thesis, submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the Department of Mathematics and Applied Statistics, University of Wollongong, is wholly my own work unless otherwise referenced or acknowledged. The document has not been submitted for qualifications at any other academic institution.

Preeya Riyapan

May 30, 2016

Acknowledgements

This thesis would never have been possible without the help and support of many kind people. I would firstly like to express my deepest gratitude to my supervisor, Professor Ray Chambers, for his continuous support from the initial to the final level, for his great patience, motivation, excellent guidance, and immense knowledge. I could not have imagined having a better or friendlier supervisor for this thesis.

I would like to acknowledge the financial support of a Royal Thai Government Scholarship that provided the necessary financial support for this research. I am very grateful for that. I also thank the Prince of Songkla University for their financial support for first arrival and for giving me this opportunity.

Amongst my fellow PhD students; Klairung, Bothaina, Rebecca, Amirah, and Diane, I would like to thanks them all for sharing the experiences, and encouraging each other.

Finally, I would like to thank my parents, elder brother, sister-in-law, dear nephew and niece for always supporting me and encouraging me with their love and best wishes. Most importantly, I would like to thank my beloved boyfriend, Santi Chotkaew, for his patiently waiting for me, encouragement, and advice in programming.

Abstract

Survey data are an important source of information for modern society. However, the complex structures of modern populations require sampling designs for surveys that are more complex than simple random sampling in order to be effective. With large national population surveys, the sample data collected via these designs typically include sample weights that allow analysis to take account of these complex population structures. As a consequence, these sample weights need to be taken into consideration when modelling the sample data, e.g. when the target of estimation is the coefficients of a regression model for the target population. In this situation, it is important to know whether these weights should be used when identifying an appropriate model specification and also whether they should be used when fitting this model to the survey data. Given the complexity of both model choice and model fitting and the limited literature on this issue, there is clearly scope for theoretical and methodological development in order to help with these decisions.

The principal aim of this thesis is to develop and evaluate strategies for population modelling using complex sample survey data. More specifically, since both linear and logistic regression analysis are very widely used statistical modelling methods, our goal is to develop procedures for analysing complex sample survey data in order to choose appropriate linear and logistic regression models based on either unweighted or weighted modelling of the survey data. In particular we develop two approaches to regression model choice and consequent regression model fit given complex survey data. These are a likelihood-based approach and a prediction-based approach. Both approaches allow us to identify a final model given two competing

models suggested by model search methods based on application of different inferential paradigms. The likelihood approach is based on the non-nested test suggested by Vuong (1989), while the predictive approach uses cross-validation. The two model choice methods differ in terms of whether or not they use the sample weights. That is, we investigate four modelling strategies defined by the combination of two different approaches to model identification (likelihood-based versus cross-validation) and two paradigms for model search (unweighted versus weighted).

In order to evaluate these strategies in realistic circumstances, we simulate their performance under three scenarios, which we label as Non-Informative Sampling (Non-Informative Sampling (NIS)), Missing Stratification Information (Missing Stratification Information (MSI)) and Response-Based Sampling (Response-Based Sampling (RBS)), and for two types of regression models: linear and logistic. All three simulation scenarios are based on a real survey data set. Our results indicate that it is possible to recover the underlying population model in the first and the third of the scenarios that we investigate. However, in the case of the second scenario, our results indicate that the model identified by the suggested procedure does not recover the actual population model. Finally, we apply our modelling approach to the original real survey data set in order to assess its practical usefulness.

Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

CV_W	The weighted cross-validation statistic
CV_{NW}	The unweighted cross-validation statistic
VS_{large}	The voting-system strategy with a larger model chosen in case of a tie
VS_{small}	The voting-system strategy with a smaller model chosen in case of a tie
V_W	The weighted Vuong test statistic
$V_{NW_{small}}$	The unweighted Vuong strategy with a smaller model chosen when the unweighted and weighted model specifications are not significantly different
V_{NW}	The unweighted Vuong test statistic
$V_{W_{small}}$	The weighted Vuong strategy with a smaller model chosen when the unweighted and weighted model specifications are not significantly different
$Vuong_W$	The strategy that used the weighted Vuong test statistic
$Vuong_{NW}$	The strategy based on the unweighted Vuong test statistic
CV	Cross-Validation

DD	The test statistic proposed by DuMouchel and Duncan (1983)
KLIC	Kullback-Leibler Information Criteria
MLE	Maximum Likelihood Estimator
MSI	Missing Stratification Information
MSPE	Mean Squared Prediction Error
NIS	Non-Informative Sampling
OLS	Ordinary Least Squares
PE	Prediction Error
PS	The test statistic proposed by Pesaran (1974)
RB	Relative Bias
RBS	Response-Based Sampling
RRMSE	Relative Root Mean Squared Error
RV	Relative Variance
UMS	Unweighted Model Specification
WLS	Weighted Least Squares
WMS	Weighted Model Specification

Chapter 1

Introduction

Data from surveys are an important source of information for modern society, in the sense that informed decision making benefits from the information collected in surveys, provided this information is accurate. These include: the implementation and changes to programs to better target the needs of population of a country, the development of community policies and projects, determining priorities in allocating funds for government agencies, public inquiries, allocation of funds and federal seats, planning and administration of local and federal governments, improving delivery of nursing care and building better shopping centres and areas where they are most needed. Bringing together data and information from different surveys can also help improve health delivery issues around the globe as well as help policy makers address emerging social and economic issues associated with climate change. In this context, social survey data are one of the most important data sources for understanding society and changes in social trends, allowing one to monitor changes in the well being of citizens as well as providing information on specific social policy issues. Similarly, health surveys are essential for providing the public health data that inform policy makers as well as members of society about important health issues for which health policy and procedures need to be implemented. Information from surveys therefore represents one of the most important contributions to decision making processes aimed at effectively implementing international and government policies. It is important to have reliable and unbiased methods for extracting information from a

survey, particularly since this information will be used as the basis for making decisions about the large target population of the survey. In other words, we need reliable ways of inferring the relationships that characterise a large population of interest from those observed in a small sample of it.

In making a decision based on survey data, it is important to ensure that good sample design has been used to select the sample from the target population. However, it is just as important to use good methods of analysis after the survey data are collected. In particular, when poor design is unavoidable, one should focus on the analysis because the negative impact of poor design can be made ignorable when the design is allowed for in the analysis, e.g. by appropriate weighting (Smith, 1976). However, the necessity to account for how the survey data were obtained makes the analysis of survey data more cumbersome since these data then represent the outcome of two random processes. As Pfeffermann (1998) points out

> *"Survey data may be viewed as the outcome of two random processes: The process generating the values in the finite population, often referred to as the 'superpopulation model', and the process selecting the sample data from the finite population values, known as the 'sample selection mechanism'."* (Pfeffermann, 1998, p.1087)

That is, the process generating the population data (the superpopulation model) and the process used to select the sample from the corresponding population are both required to be part of any realistic statistical model for the sample survey data. Consequently, analysis of these data should take both processes into account. However, the use of complex sampling mechanisms make this a difficult exercise.

Due to the complex structures of large finite populations, quite sophisticated mechanisms are typically employed to select the sample data. For instance, the use of highly detailed stratified sampling indicates that population variability changes considerably between the different layers of the finite population defined by the strata. Similarly, multi-level sampling, including cluster sampling, is typically used in finite populations that can be represented as a hierarchical structure. These are

examples of sampling designs that we will refer to from now on as *complex sampling designs*. Data obtained using a complex sampling design are usually called *complex survey data*. Note that implicit in this notation is the assumption that one has access to *identifier variables* (e.g. stratum identifiers, cluster identifiers) that allow one to 'see' these complex population structures in the survey data.

Analyses of complex survey data are usually carried out with two aims in mind: descriptive and analytic. In the past, most uses of surveys were descriptive. That is, the objective in general was to estimate finite population quantities such as population totals and means. More recently, analytic uses have increased in importance. In particular these uses have focussed on the description of relationships in complex survey data. For example, a common analytic use is the estimation of regression coefficients, with the aim of making an inference about the parameters of a target model for the population. An advantage of analytic use over descriptive use of a survey is that it enables prediction of interesting issues under similar conditions without requiring that the survey be repeated. In particular, analytic uses of surveys are now more common than descriptive uses (Thompson, 1997). As a consequence, the main focus of this thesis will be the analytic use of complex survey data for the estimation of population regression coefficients.

Why regression? This can be answered by the following quote.

> *"Study regression. All of statistics is regression."*
> (Heeringa, West & Berglund, 2010, p.179)

This quote is essentially equivalent to saying that regression is how statistics represents natural phenomena in a form suitable for analysis and interpretation. More importantly, regression coefficients provide measures of association between an outcome variable and other variables of interest that help to explain the mechanisms in social processes that are the focus of surveys. In this context, the simplest form of regression is linear regression, where one explains the population distribution of a scalar continuous variable in terms of a linear combination of the values of a set of independent variables, and logistic regression where one explains the population

distribution of a binary variable as the set of outcomes of independent Bernoulli variables with 'success' probabilities that, after a logit transformation, are also a linear combination of the values of a set of independent variables. Both these methods of regression modelling are widely used. Consequently, this thesis will be devoted to the estimation of regression coefficients of the target population under both linear and logistic regression.

The target populations underpinning the analytic uses of a survey are often indefinite. In order to fix this concept, it is common to base inference on a hypothetical population that underpins the survey target population (and hence sample) values. This population corresponds to all the possible values that could occur along with the probability of occurrence and is usually characterised in terms of a model usually referred to as the superpopulation model. This allows one to define the hypothetical population from which the actual target population is drawn (Thompson, 1997). Consequently, both the target population and the hypothetical population are assumed to follow the same superpopulation model. Furthermore, even if we do not know the true model used to create the superpopulation, we assume that as the size of the target population increases, the coefficient parameters of this model fitted to the population values become arbitrarily close to those of the superpopulation model. Based on this assumption, it is then natural to view the regression coefficient parameters of the finite population model (i.e. the model fitted to the finite population values) as the estimation target for inference.

So far, the issue of exactly how one chooses the model that best explains the sample values, and the way these sample values were selected, has not concerned us. However, when the target of estimation is the regression coefficients of the true finite population model, the issue of whether the sampling method used is non-informative or informative needs to be considered. To illustrate this issue, consider first the case of non-informative sampling from a finite population. Let Y, $\mathbf{X}$ and I represent in turn a dependent variable, a matrix of independent variables and an indicator variable for whether a particular population element is selected. Given independent

population elements, a *non-informative* sampling scheme is then defined as one where the sample inclusion variable I satisfies

$$Y \perp I|\mathbf{X} \tag{1.1}$$

where the sign $\perp$ indicates that the two variables on either side of it are independent, and was originally suggested by Dawid (1979). It is commonly stated that the sampling design corresponding to I is *ignorable* when condition (1.1) is true (Sugden & Smith, 1984), i.e. the design does not matter when analysing the relationship between Y and $\mathbf{X}$ using sample data obtained under such a design. Another interpretation of (1.1) is that it indicates that the sample distribution of Y does not differ from the corresponding population distribution of Y, or in other words the sample model for the relationship between Y and $\mathbf{X}$ is no different from the corresponding population model (Pfeffermann, 2009). An immediate consequence is that the estimated coefficients of the sample model are also estimates of the coefficients of the target population model under this condition. This is very convenient for modellers who have already specified a model for the target population, since all the information they need to fit this model is already available in the sample data. That is, classical regression analyses can be directly applied and used in this case.

On the other hand, *informative sampling* occurs when condition (1.1) is not true. An immediate consequence is that informative sampling generally induces a difference between the sample model and the corresponding population model. Indeed, disregarding this difference will lead to misleading results as Pfeffermann (2009) points out

> "*Failure to account for the difference between the sample model and the census model can result in biased and inconsistent parameter estimators, poor coverage of confidence intervals, wrong predictions, and ultimately erroneous conclusions.*" (Pfeffermann, 2009, p. 425)

It is quite clear from this quote that the informativeness of the sampling method should be taken into account when modelling complex sample survey data.

Since informativeness is generally the absence of non-informativeness, it is impossible to specify precisely. For this reason we focus in this thesis on two specific types of informative sampling defined by two general situations where the sample and population models diverge. Both correspond to a form of stratified sampling. In the first, which we refer to as *Missing Stratification Information* in what follows, informativeness arises through misspecification of the sample model through omission of a key regressor. In particular we assume that the population model is stratified on a categorical variable Z that is distinct from $\mathbf{X}$ (i.e. the relationship between Y and $\mathbf{X}$ varies from one level of Z to another, and $\mathbf{X}$ does not include Z), but the sample data (and hence the sample model) ignore Z. This is equivalent to saying that (1.1) does not hold because

$$Y \perp I|\mathbf{X}, Z$$

holds instead. In the second, which we refer to as *Response-Based Stratification* in what follows, we also have a stratified sampling design, but this time the stratification is on Y, in the sense that different ranges (or categories) of Y correspond to different strata. In this case it is quite clear that (1.1) cannot hold.

As pointed out earlier, not checking whether (1.1) actually holds, and proceeding on the basis that it does, can lead to erroneous inferences, as clearly stated by Pfeffermann & Sverchkov (2009, p455). In this thesis we therefore focus our study on the consequences for regression modelling of sample data when the population regression model is unknown and when one of three scenarios: Non-Informative Sampling (NIS), Missing Stratification Information (MSI) and Response-Based Sampling (RBS), holds. In doing so, we adopt a regression modelling strategy with two main steps. In the first step we identify the variables in $\mathbf{X}$ that we believe underpin the population regression model for Y. In the second step we then use these variables, together with Y, to obtain the final model. We refer to the first step as the *model choice* step and to the second step as the *model fit* step.

The standard advice (Pfeffermann & Holmes, 1985; DuMouchel & Duncan, 1983;

Kott, 1991) when carrying out regression modelling of complex survey data with the aim of making an inference about a population regression model is to incorporate the *sample weights* into the modelling process. However, it is unclear how this advice should be implemented in the context of the two step modelling process above. Furthermore, it is unclear how incorporation of sample weights themselves is an effective strategy for protecting inference in the three situations (NIS, MSI and RBS) described above. In fact, from a theoretical perspective, sample weights should be unnecessary under NIS. A goal of this thesis will therefore be to explore more fully how sample weights can be incorporated into the the two-step modelling strategy above and to assess whether they are effective tools in regression modelling of complex survey data. In order to do this, we compare use of sample weights vs. ignoring these weights at both steps of the modelling process, leading to the four options set out in Figure 1.1 below. In doing so we aim to develop a standard survey data modelling decision process for non-statistical analysts who are not familiar with analysis of complex sample survey data.

Sample data → Model specification ignoring weights → Model fit ignoring weights
Sample data → Model specification ignoring weights → Model fit using weights
Sample data → Model specification using weights → Model fit ignoring weights
Sample data → Model specification using weights → Model fit using weights

Figure 1.1: The four options of the modelling process

As we can see from Figure 1.1 that there are four possible pathways that we can adopt when thinking about whether or not to use sample weights in modelling. These are defined by whether or not we use sample weights in specifying the model (i.e. choosing the regressors) and whether or not we use the sample weights in fitting the chosen model. Equivalently, there are two basic questions that need to be answered when choosing which one of these pathways one should take:

(i) Given two model specifications, one developed using weights, and the other

developed ignoring the sample weights, which one should be adopted?

(ii) Given the model specification adopted in (i) above, should one then fit the model using the sample weights or should one fit it ignoring the sample weights?

Clearly, the answer to (i) will depend on how much the two model specifications differ. If they are the same (i.e. the same set of regressors are in both models) then only question (ii) is of concern. However, if the two model specifications in (i) are different, then we need methods for deciding between them. Similarly, we need methods for deciding when to use the sample weights and when to ignore them when fitting the model chosen after (i).

Unfortunately, although methods for resolving (ii) exist in the literature, there do not appear to be any results to guide our answer to (i). As a consequence, the research described in this thesis will develop and evaluate procedures for answering (i). These new procedures will then be evaluated under the NIS, MSI and RBS scenarios, as will their extensions to resolving (ii). In all cases, our aim will be to assess the fitted models suggested by these new methods in terms of their ability to recover the true population regression model.

This thesis is organised as follows. In Chapter 2 we review the relevant literature on model choice and model fit for regression given complex survey data, and in particular, how researchers have allowed for the availability of survey weights and stratification information. The first three sections of this Chapter will review general concepts relevant to the analysis of survey data. The following sections will then review the literature on allowing for the effects of sampling design in regression modelling using complex sample survey data. Given the limited literature on the impact of sample design on model specification (question (i) above), we then review some general methods that may be applied to resolving this issue. Chapter 2 concludes with a review of the three types of models (nested, overlapping and non-nested) that define the decision problem (i) and concern in the problem (ii).

Chapter 3 then develops the inferential and decision framework that is the basis

of the research reported in this thesis. In this Chapter we discuss the creation of sampling weights, and how a structured analysis of the complex survey data leads to our proposed regression modelling procedure. The main focus of this Chapter is therefore the development of this procedure, together with a discussion of how it can be implemented. In Chapter 4, we provide a theoretical development of the impact of model misspecification when important population model covariates are omitted from the sample model, as is the case in the MSI scenario. Here we focus on the bias that ensues when simple linear and logistic regression models are appropriate.

The decision process set out in Chapter 3 cannot be implemented without appropriate tools that allow us to answer questions (i) above. This leads to theoretical and methodological developments set out in Chapter 5 that enable us to choose the final model. In particular, in this chapter we provide a detailed derivation of the two methodologies investigated in this thesis: the likelihood-based approach and the prediction-based approach.

Chapter 6 describes results from a sequence of simulation experiments, based on a real survey data set, that were used to test the efficiency of the modelling procedure proposed in Chapter 3 and the methodologies developed for model choice in Chapter 5. This Chapter includes a description of this data set and the criteria that were used in evaluating the final model fits. The simulation experiments recreated versions of the NIS, MSI and RBS scenarios using this data set and then implemented the modelling procedure of Chapter 3. In addition, simulation results as a consequence of theoretical modelling bias developed in Chapter 4 are presented. Chapter 6 concludes with an application of the proposed modelling procedure to the original survey data.

An approach to resolving the decision problem (ii) is set out in Chapter 7, which includes an implementation of the modelling procedure, two statistical tests and associated simulation results. The tests are the one proposed by DuMouchel and Duncan (1983) and the one suggested by Pesaran (1974), while the simulations evaluate the efficiency of these procedures using the same simulation framework

as that used in Chapter 6. Finally, we conclude the thesis with a discussion and suggestions for future research in Chapter 8.

Chapter 2

Literature Review

2.1 Introduction

In the previous chapter we raised two questions that need to be answered if one wishes to use a consistent framework for parametric inference from complex survey data. In particular, they need to be answered before one can claim to have a cohesive approach to population modelling via regression analysis of complex survey data based on a valid inferential approach, and using effective model search strategies. In this chapter we review the recent literature that focuses on this topic. The first section below is therefore a review of general procedures used in regression analysis of complex sample survey data. In the next section, inferential approaches are reviewed, with reference to the general application of the three main frequentist approaches in current use: model-based, design-based and model-design randomization-based. An important consideration here is failure to take account of the non-informative sampling assumption (1.1) when fitting regression models based on complex survey data. This leads to consideration of the appropriate way to take account of the effects of both the sampling design and the failure of (1.1) in regression modelling. A review of the literature discusses both issues. Given the limited literature addressing question (i), we will tend to focus mostly on literature relevant to (ii) and the general issue of model choice methodologies in statistical analysis. This review will then guide our discussion of model search strategies aimed

at answering question (i) in the last section of this chapter.

2.2 Regression Analysis for Complex Sample Survey Data

Heeringa *et al.* (2010) sets out the following four-step strategy for regression analysis of complex survey data.

1. **Model Formulation:** This step aims to identify the regression models that reflect the relationships among the variables of interest. Survey data sets typically include many variables, and the choice of the independent variables that significantly relate to the target dependent variable can be made through *stepwise variable selection* procedures such as *forward selection* and *backward elimination*.

2. **Model Estimation:** This step is concerned with the statistical methods employed to compute estimates of the parameters of the regression model identified in Step 1. The most popular estimation methods applied to complex sample survey data are the *ordinary least squares (Ordinary Least Squares (OLS)) method* and the *weighted least squares (Weighted Least Squares (WLS)) method*. The OLS method minimises the sum of squared residuals defined by the model fit and so ignores sample weights in the estimation process. We refer to this estimation method as *ordinary least squares estimation* from now on. The WLS method minimises the sample weighted sum of squared residuals and so explicitly builds the sample weights into the fitted model. We refer to this estimation method as *weighted least squares estimation* from now on.

3. **Model Evaluation:** This step assesses the adequacy of the model fitted in the previous step. It's aim is to examine whether the regression model estimated in Step 2 is adequate for inference in the next step. In linear regression modelling, standard measures for assessing adequacy of the fitted model in-

clude an examination of the model assumptions (i.e. conditional independence given model covariates, constant variance and normality of the model errors), an evaluation of the model goodness of fit, and a consideration of unusual observations that might affect the fit of the model. Heeringa *et al.* (2010) state that these standard diagnostic methods can be directly applied to complex sample survey data when fitting a regression model for the finite target population of the survey.

4. **Inference:** This final step completes the analysis process by inferring the 'true' values of the regression coefficients of the target population model from the estimated parameters obtained in the previous steps. Classical methods (e.g. t-statistics and confidence intervals) that are typically employed for making inferences can be applied to complex sample survey data.

Note that the four-step regression analysis process detailed above is an iterative process, and can have multiple iterations as described in Heeringa *et al.* (2010). Furthermore, these authors suggest that the analyst needs to be careful at each step if the aim is to obtain a plausible model fit to the complex sample survey data. In this thesis we focus on the first three steps of this four-step regression analysis process: model choice, model fit and model evaluation. Details of our proposed strategy for dealing with the different decisions that need to be made when implementing these steps will be detailed in Chapter 3.

2.3 Inferential Approaches

Central to estimating population features of interest is the mechanism used for inference from samples to populations (Sterba, 2009). From a frequentist perspective there are two main mechanisms, model-based and design-based, used for making this inference. A detailed review of those two mechanisms was presented by Sterba (2009), and briefly summarised here as follows:

1. **Model-Based Approach:** This inferential mechanism hinges on a statistical model that should play the central role in the data analysis. The targets of inference under this mechanism are the model parameters (for example, regression coefficients). To implement this approach, one needs to:

   (i) Formulate a statistical model that describes how the target response variable is generated. A hypothetical (infinite) population is then defined by all possible values of the target response variable generated by the model.

   (ii) Make an assumption about the source of the underlying variability in the population data. Typically, a parametric distributional assumption is required for the population values of the response variable y to be validly treated as the outcomes of random variable. For example, such an assumption allows us to treat the errors in a regression model as independently and identically distributed (iid) realisations of a random variable with mean zero and a constant variance.

2. **Design-Based Approach:** In some ways, the idea behind this approach is the opposite of that underpinning the model-based approach. In particular, population values are treated as fixed. The only source of variability is the randomness in the observed sample values caused by the sampling process (Kish, 1965). Population characteristics that are the targets of this approach are assumed to be fixed quantities (Binder & Roberts, 2009). To implement this approach, one needs to:

   (i) Specify the random sampling design that was used to obtain the sample data. Typically this includes specification of a sampling frame, a sampling design, and scheme to draw the sample from the finite population values of y. Sterba (2010) defines these three terms as follows: *"The sampling frame is the list of primary sampling units in the finite population; the sampling design assigns nonzero probabilities of selection to each sample that could be drawn from the frame; the sampling scheme is a mechanism*

for implementing the sampling design." (Sterba, 2010, p. 724-725)

(ii) Use the specified sampling design to draw a sample from the finite population, and then calculate sample weights. Note that no model or distributional assumption is required to allow the sample values of y to be treated as the realisations a random variable as in the model-based approach.

Both approaches have limitations. For example, in the model-based approach, model specification can be complicated by the need to condition on all stratifying variables (Pfeffermann, 1996). Such conditioning can then lead to a more complicated interpretation of the model parameters (Pfeffermann *et al.*, 1998). Consequently incomplete conditioning on the sampling design can result in potential bias (Sterba, 2009). In contrast, the most serious disadvantage of the design-based approach is that it essentially ignores analytic and causal inference (Sterba, 2009), focussing instead on inference about characteristics of the target population. This naturally leads to the question *"Which approach should be used in inference when both have limits?"* Over the past four decades, numerous studies have attempted to reconcile the two approaches (Smith, 1976, 1984; Hansen *et al.*, 1983; Iachan, 1984; Gregorie, 1998; Brewer, 1999; Geuna, 2000; Wheeler *et al.*, 2008; Binder & Roberts, 2009; Sterba, 2009), but with limited success.

A compromise inferential approach is reviewed by Binder and Roberts (2009), who refer to it as *model-design-based randomization.* This approach combines those two approaches reviewed above by allowing for both model-based and design-based variability. In particular, it allows for the population values of y to be the realisations of a random variable. Binder and Roberts (2009) describe this process as made up of three phases as follows:

(i) In the first phase, values of the characteristics of a finite target population are generated, based on realisations of random variables described by a statistical model.

(ii) The second phase augments the finite population variables with design variables, such as stratification and cluster identifiers. The

values of these design variables can depend on the outcomes of the random variables in the first phase and may also be random.

(iii) In the third phase, a probability sample is selected from the finite population using the population values of the design variables to specify the sampling mechanism.

(Binder & Roberts, 2009, p.39)

We see that this approach gives the design variables the opportunity to be included in the statistical model (although this is not mandatory). In addition, it provides a challenge to the model-based approach since it implies that one has to allow for an informative sampling mechanism as a potential factor that could lead to erroneous inference. As a result, we will use the model-based approach in inferential process, together with the three-phase process proposed by Binder and Roberts (2009) above, throughout this thesis. An obvious reason for doing this is that we are then able to consider the analytic uses of the fitted model as well as to take account of the impact of informative sampling on the modelling process.

2.4 Relevant Literature

As far as we are aware, there appears to be no previous research aimed at answering question (i) posed in the previous chapter, i.e. how to choose between two model specifications, one derived using the sample weights and the other derived ignoring these weights. Consequently, this section briefly reviews previous research aimed at tackling question (ii) - i.e. given weighted sample survey data, should one use these weights in model fitting?

2.4.1 Using Sample Design Information in Regression

There are a large number of published studies describing the use of sampling design information when model fitting is carried out using complex sample survey data. Many of these focus on the issue of whether sample weights should be used when

fitting a model to such data. It should also be pointed out that this is not a new problem. In fact, over the past 30 years, there have been many research papers that aim to answer this question (Pfeffermann & Nathan, 1981; DuMouchel & Duncan, 1983; Lee *et al.*, 1986; Nordberg, 1989; Preffermann, 1993; Lohr & Liu, 1994; Winship & Radbill, 1994; Korn & Graubard, 1995; ; Pfeffermann & Sverchkov, 1999, 2003; Chambers et al., 2003; Wu & Fuller, 2005; Eideh & Nathan, 2006). Most of these papers not only point out the solution to the question but also provide useful guidelines regarding how sample weights should be used in model fitting.

DuMouchel and Duncan (1983) assume that a stratified sampling design had been used to obtain the realised sample, and investigate the impact of sample weights on model fit with respect to the population model of interest. In their review of the use of sample weights in regression, they suggest that:

- Sample weights are unnecessary if the target population model follows the *iid* assumption.

- Sample weights are essential tools in fitting a model if the target of estimation is the regression parameter of a census model, i.e. the fitted regression model parameter when the entire population data are used.

- If the target population follows a model in which parameters vary by stratum, and if the assumed model does not allow for this, then it is questionable whether incorporating sample weights into the fit is of any use since both fits (that is, unweighted and weighted) lead to biased estimators. They suggest that further investigation of this problem is required.

In their major study, they propose a strategy for choosing an appropriate model given the results from unweighted and weighted fits to the sample data. This strategy is based on the calculation of an appropriate F-statistic. In Chapter 7 we provide details of how this strategy is applied in this thesis in order for (ii) to be resolved. It should be noted however, that this strategy assumes that there is only one model specification. That is, they make no attempt to develop a strategy for deciding

between two competing model specifications.

Lee *et al.* (1986) present a similar study of the use of sample weights in estimation, based on three examples of complex survey data analysis. In this study, the sample used for analytic regression modelling was selected via a multi-stage sampling design. From their analysis of all three examples these authors recommend that:

- Sample weights are an important component of analytic regression, playing a key role in estimation.
- Any analysis of complex survey data that ignores both sample weights and the sampling design may lead to biased estimation and inaccurate inference.

Heeringa *et al.* (2010) illustrate the role of sample weights in model fitting through a comparison of three methods: ordinary least squares, classical weighted least squares and sample weighted least squares methods. Sample weights are ignored in estimation for the first two methods whereas, in the third method, they are used for estimation. In their study these authors show that it is essential to incorporate sample weights into the fit of the model when the target is to estimate the regression coefficient parameters of the population model. They therefore suggest that sample weights should be employed in model fitting to complex sample survey data.

Korn and Graubard (1995) study the difference between two estimates, one fitted by ignoring sample weights, and the other fitted by using the weights, using survey data collected in the 1988 National Maternal and Infant Health Survey. In this case simple random stratified sampling was used to select the data from six strata with unequal probabilities of selection within each stratum. That is, the sampling weights varied from stratum to stratum. The main finding of their study is that when the target of estimation is the parameter of the population model, then unweighted estimators are biased while weighted estimators are asymptotically unbiased. In particular, the weighted estimators are consistent (Pfeffermann, 1993; Lohr & Liu,

1994). As consequence, these authors recommend the use of sample weights in the model fitting process.

Winship and Radbill (1994) investigate the use sample weights in regression analysis through consideration of two situations: the first one where the variables used in the construction of the sample weights are included in the regression model as independent variables, and the second one where the target dependent variable is used in the construction of the sample weights. Two types of regression model fit were then carried in these two situations. The first type corresponded to the the fit that ignored the sample weights, while the second one corresponded to the fit using these weights. Their Monte Carlo experiments relied on a model-based approach where data are generated based on a specified model, and consisted of repeated regression analyses based on five iterated steps that are summarised as follows:

1. Create a data set by first independently generating values of independent variables and then using these values to calculate the expected values of the target dependent variable.

2. Separately generate an error, and add it to each calculated expected value of the dependent variable.

3. Estimate the model parameters using these data, either ignoring the sample weights or using them.

4. Record these estimates.

5. Repeat Step 2-4 until 750 sets of parameter estimates are obtained.

They use average values of estimates based on 750 iterations of the above procedure to investigate whether the estimators are biased and came to the following conclusions.

- In the first situation (i.e. where the variables used in the construction of the sample weights are included in the regression model as independent variables),

the unweighted estimators are preferred to the weighted estimators. In other words, sample weights are unnecessary for this case.

- In the second situation (i.e. where the target dependent variable is used in the construction of the sample weights), sample weights are essential for the model fit process. That is, the weighted estimators are preferred to the unweighted estimators for this case.

Taken together, these findings suggest a procedure for choosing a model when analysing a set of complex sample survey data that include sample weights. This can be summarised as follows:

1. Fit two models: one without using the sample weights, and one using the sample weights.

2. Compare the estimated model coefficients for those two models. If they are substantively similar, then choose the unweighted estimates. If not, use the F test proposed by DuMouchel and Duncan (1983) in order to check that whether the sample weights have an impact on the model fit. If the test suggests that sample weights have an impact on the fit, then revise the model specification and repeat the process above until estimates from the two models fits are substantively similar.

3. If it is impossible to bridge the gap between these two sets of estimates through model respecification, then Winship and Radbill (1994) suggest that the weighted estimates should be used because of their consistency properties.

This procedure seems practically reasonable. However, it has the obvious weakness that a lot of time may be required for model fitting if the model has to be respecified several times. Another problem with this procedure is that it fails to take the two-step modelling strategy into account when assessing the quality of the final fit, e.g. when computing standard errors.

Reiter *et al.* (2005) suggest a similar strategy to that of Winship and Radbill (1994). Their study used three real data sets to examine the model fit problem

for complex survey data under two types of regression models: linear and logistic. Guidelines from previous studies based on these data were used for model specification, and their suggested modelling procedure was as follows:

1. Fit two models based on two different inferential approaches: model-based and design-based (that is, unweighted vs. weighted).

2. Consider whether the two model fits in Step 1 differ. If not, the unweighted fit should be better due to smaller standard errors.

3. If these model fits differ, the decision on which one to adopt is left up to the analyst.

Comments by these authors regarding how one should approach statistical modelling using complex sample survey data can be summarised as:

- In order to obtain reliable conclusions, consideration should be given to the impact of the sampling design on regression modelling.

- Under non-informative sampling, unweighted estimators are preferable to weighted estimators when stratification variables are incorporated into the model.

Lohr and Liu (1994) provide an excellent overview of the use of sample weights in regression analysis. Gathering together results from previous studies, they show how sample weights can have an impact on model fit as well as how to take account of these weights to regression modelling. We summarise the main points in their their paper as follows.

- Sample weights are useless when the target population model is correctly specified.

- When the variables underpinning the sampling design are correlated with the target dependent variable, sample weights should be employed in model fitting (DuMouchel & Duncan, 1983).

- When there is model misspecification, and this cannot be fixed by respecification of the model, then incorporating sample weights into the model fit should be taken into consideration since sampling weights can protect in the case of model misspecification (Pfeffermann & Holmes, 1985; DuMouchel & Duncan, 1983; Kott, 1991).

- Both unweighted and weighted estimators are biased in the case of Missing Stratification Information (MSI) (Kott, 1991).

- One way of attempting to correct the difference between the unweighted and weighted estimators under MSI is to search for independent variables that are correlated with the variables used in constructing the sample weights (DuMouchel & Duncan, 1983).

- Sample weighted estimators might be reasonable in inference for the target regression coefficient parameter when stratification variables are unavailable or cannot be accessed (Little, 1991; Smith, 1988).

Overall, Lohr (1999) summarises the guidance in the literature on how regression estimators based on complex sample survey data should be used to infer model parameters as follows:

- The use of unweighted estimators obtained under the model-based approach is appropriate when unequal probabilities of selection are employed and the sample size is small. This is because the potential increase in bias from adopting this approach is more than offset by the smaller variability of the unweighted estimators.

- If one would like to estimate model parameters so as to inform decision making related to national or public policies, and the target population and sample are large, then weighted estimators may be more appropriate than unweighted estimators. However, the model then has to be precisely specified.

- An alternative model should be searched for if a situation arises where the unweighted and the weighted estimators differ substantially.

In summary, we see that there does not appear to be a general consensus about how one should answer the question (ii) posed in the previous chapter, i.e. whether or not sample weights should be used in model fitting. The different recommendations can be summarized under two headings: the first concerns situations where sample weights should be used in model fitting, and the second concerns situations where sample weights should be unnecessary.

1. Sample weights are essential tools in fitting a model in the following situations:

    - When the target of estimation is the regression parameter of a census model (DuMouchel & Duncan, 1983; Heeringa *et al.*, 2010; Korn & Graubard, 1995), and stratification variables are unavailable or cannot be accessed (Little, 1991; Smith, 1988).
    - The target dependent variable is used in the construction of the sample weights (Winship & Radbill, 1994).
    - The variables used in the sample design are correlated with the target dependent variable (DuMouchel & Duncan, 1983).
    - When there is model misspecification (Pfeffermann & Holmes, 1985; DuMouchel & Duncan, 1983; Kott, 1991).

2. Sample weights are unnecessary when fitting a model under the following circumstances:

    - When the target population model follows the *iid* assumption (DuMouchel & Duncan, 1983) and is correctly specified (Lohr & Liu, 1994).
    - The variables used in the construction of the sample weights are included in the regression model as independent variables (Winship & Radbill, 1994).

- Under non-informative sampling, e.g. when stratification variables are incorporated into the model (Reiter *et al.*, 2005).

2.4.2 The Interaction Between Sampling Mechanisms and Population Modelling

As stated in the previous chapter, the sampling mechanisms that are relevant to this thesis correspond to three types of sampling: Non-Informative Sampling (NIS), Missing Stratification Information (MSI) and Response-Based Sampling (RBS). Under each, an essential condition that needs to be taken into account in inference is whether (1.1) holds. In this context, we note that most studies of regression modelling with complex survey data have assumed non-informative sampling (that is, (1.1) holds) even if, in reality, this is not the case.

Two notable exceptions are Winship and Radbill (1994) and Pfeffermann (1996). The former study considers all those three sampling mechanisms: Non-Informative Sampling (NIS), Missing Stratification Information (MSI) and Response-Based Sampling (RBS), in an investigation of whether sample weights should be used in regression analysis of complex sample survey data. Their findings indicate that sample weights could be a major factor is determining whether inference is unbiased. In particular, they claim that sample weights do not lead to significant numerical differences between unweighted and weighted fits under NIS. Whereas, under MSI, use of these weights can be an indication that a key regressor has been omitted from the model (that is, both unweighted and weighted fits provide different estimates). Nevertheless, they show that the unweighted fit remains preferable to weighted fit in this case, i.e. the weights are useless for the MSI situation. Under RBS, the use of these weights has important implications for estimating the parameters of the target population model. Here Pfeffermann (1996) explicitly considers how one should use the sample weights in modelling when Response-Based Sampling (RBS) occurs. His study focuses on multistage cluster sampling, and its main results can be summarised as follows:

- Biased estimators can arise under RBS even when there are no unequal selection probabilities.

- The use of sample weights can fix the two main problems caused by RBS. These are when the model for the sample data (i.e. the sample model) and the corresponding population model differ, and when the target response variable is correlated with the probabilities of selection.

Finally, we note that several of the papers referenced above also state that the use of sample weights can protect inference in the MSI situation by virtue of the fact that one can ignore whether or not the model is correctly specified under a design-based analysis (DuMouchel & Duncan, 1983; Pfeffermann & Homes, 1985). Kott (1991) adopts this approach when arguing that the use of these weights is vital for regression analysis in the MSI case.

2.5 Model Choice Methodologies

In this section, we return to the basic issue raised by question (i) posed in the previous chapter, i.e. how to choose between two model specifications. We review general methodologies that are commonly used for model choice, and how these can be applied in this thesis in order to provide a solution to this question.

There are many research papers on the issue of model choice in regression analysis (Linhart & Zucchini, 1986; Zucchini, 2000; Lahiri, 2001; Hossain & Bhatti, 2003). Different methods for model choice are reviewed by Lahiri (2001), who classifies them into four categories: information-theoretic (i.e. likelihood-based), cross-validation (Cross-Validation (CV)), classical hypothesis testing, and predictive error-based.

Deciding which type of methods should be adopted is an important issue for a statistical analyst. Since they appear to be widely used, we will focus on the the first two categories, i.e. methods that take a likelihood-based approach, and methods that base model choice on cross-validation. Granger *et al.* (1995), for example, claims that likelihood-based methods have become popular for model selection in

economics because these procedures have less limitations than procedures based on classic hypothesis testing. However, even though likelihood-based methods are popular for model selection in applications, there appears to be no evidence to support the use of sampling weights when applying this approach to model identification in regression analysis.

In contrast, there has been a growing statistical literature on applications of CV methods over the last 30 years, with many of these applications focussed on model and variable selection (Stone, 1974; Picard & Cook, 1984; Shao, 1993; Zhang, 1993; Droge, 1999). For example, Arlot and Celisse (2010) note that CV is widely used for model choice because of its simplicity and flexibility. However, although the CV method is now widely used in statistics, there appears to be no study of the application of this method in the context of regression modelling of complex survey data, and especially when the issue of how to deal with the sample weights in modelling needs to be addressed.

We describe in detail in Chapter 3 how we adapt both the likelihood-based and the CV-based approaches to regression model choice when weighted sample data are available. In the next two subsections, however, we first provide a brief general review of these two methods.

2.5.1 The Likelihood-Based Approach to Model Choice

A traditional method used for selecting models is the *likelihood-based approach*. This approach uses the likelihood function, which is fundamental for all statistical inference (Reid, 2010), as well as derived quantities that rely on the likelihood function. Following Reid (2012), we note the following two reasons which indicate why this approach plays such an important role in inference.

- In theory, all information in the data about the corresponding population of interest is captured by the likelihood function, since it is a function of the complete set of sufficient statistics.

- In practice, "*the likelihood function provides a set of summary statistics with*

> *known limiting distributions, and this leads to the construction of approximately pivotal functions that are easily used for inference based on the limiting normal distribution of these statistics.*" (Reid, 2012, p.731)

As a result, the likelihood-based approach has been one of the most widely used methods of carrying out model choice in traditional parametric inference. See Reid (2010).

A target of this thesis is to choose one model from two rival models, where one is specified ignoring sample weights, and the other is specified using the weights. That is, the two rival models can be classified as being of three general types. These types are non-nested, nested, and overlapping as defined by Vuong (1989) in the following. To start, we define the two competing conditional models as sets containing conditional density functions as follows:

$$F_\Theta = \{f(y|x;\theta); \theta \in \Theta\},$$

and

$$G_\Gamma = \{g(y|x;\gamma); \gamma \in \Gamma\},$$

where f and g are specified functions, and θ and γ are the parameters characterising the models F_θ and G_γ respectively. We use ϕ to denote the empty set.

Definition 2.1 (Non-nested models) *Two conditional models F_θ and G_γ are strictly non-nested if and only if: $F_\theta \cap G_\gamma = \phi$ (Vuong, 1989, p.317)*

Definition 2.2 (Nested models) *Conditional model G_γ is nested in F_θ if and only if: $G_\gamma \subset F_\theta$ (Vuong, 1989, p.323)*

Definition 2.3 (Overlapping models) *Two conditional models F_θ and G_γ are overlapping if and only if:*
(i) $F_\theta \cap G_\gamma \neq \phi$
(ii) $F_\theta \nsubseteq G_\gamma$ and $G_\gamma \nsubseteq F_\theta$ (Vuong, 1989, p.320)

Several published papers have proposed statistical tests for choosing a model in these three general situations (Cox, 1961, 1962; Pesaran, 1974; Davidson & MacKinnon, 1980; Vuong, 1989). Of these, the test proposed by Vuong (1989) is one of the most widely used and popular because of its simplicity in practice, and because it can be easily applied in all three general situations above. In this thesis, the Vuong test is therefore employed as a strategy for model specification search, and its details are described in Section 5.1.

2.5.2 Cross-Validation

Cross-Validation (CV) is based on a predictive approach to model specification (Han & Kamber, 2006). It can be defined here as follows:

> " **Cross-validation:** *The division of data into two approximately equal sized subsets, one of which is used to estimate the parameters in some model of interest, and the second is used to assess whether the model with these parameters values fits adequately.*" (Everitt, 2002, p.102)

The first subset of the data is generally called the *training set*, and the second subset is called the *testing set*. The basic approach in model specification consist of two steps: the first is where the model of interest is fitted using the training set, and second is where this fitted model is used to predict the values in the testing set (Kantardzic, 2011). Different model specifications can be tested in this way and the best one (in terms of predicting best) can then be adopted. The main assumption that needed to justify this *naive strategy* is that the training set and the testing set are chosen to be representatives of the same, unknown data distribution characterising the population of interest (Kantardzic, 2011). An obvious weakness of this strategy is that this assumption may be violated for a small data set. However, it should be reasonable given a large data set (Kantardzic, 2011). This is often the case for survey data, especially data derived from national population surveys. The cross-validation technique is therefore useful for model specification when the data are obtained from surveys with large sample sizes.

There are several variations on the CV procedure, with Leave-One-Out CV and K-fold CV representing the most commonly used variations. These are reviewed in Arlot and Celisse (2010), and we briefly summarise them as follows:

- *Leave-One-Out CV:* A single element of the original sample is used as the testing set, with the remaining sample elements defining the training set. The training/prediction process is repeated until each element in the sample has been used as the testing set.

- *K-fold CV:* This type of CV process is defined by first dividing the original sample into K random subsamples, and then using each of these subsamples as the testing set, with the remaining subsamples constituting the training set. The procedure is complete when each of these subsamples has been used as the testing set once and only once.

An advantage of K-fold CV is that it can deal with small sample sizes (Kantardzic, 2011). It is also one of the most useful CV methods because of its simplicity, efficiency and reliability (Anthony and Holden, 1998). Furthermore, it often provides the best result among different CV methods (Zhang, 1993). As a result, we will use K-fold CV as the CV strategy for model specification search in this thesis.

An important factor that needs to be taken into account when using K-fold CV is the number of folds K. Ten-fold CV is widely used (Witten & Frand, 2000). However, with small data sizes it is preferable to reduce K. Since there is no strong evidence to indicate that the ten-fold CV provides better results than five-fold CV (Feng *et al.*, 2005), we will use five-fold CV throughout this thesis. An added advantage is that five-fold CV also allows us to implement an effective cross-validation technique for model selection in the context of the small sample sizes used in the simulation study reported later in the thesis.

2.6 Conclusion

Although extensive research has been carried out on both the effect of sampling design and the use of sampling design information on regression model fitting (e.g. Heeringa et al., 2010), there has been almost no investigation of the impact of sample design, and in particular sample weights, on regression model specification.

In addition, most of the studies referred to above have focussed on the situation where the target population model is formulated under non-informative sampling (NIS). As a consequence, these studies have addressed the issue of model fit using complex sample survey data in cases where the population model and the sample model are the same. Although there have been some studies that have considered the MSI and the RBS scenarios, they have still focussed on the model fit problem. Thus, even though much is now known about the impact of sample weights when fitting a specified model to complex survey data, it is much less obvious how one should incorporate sample weights into the model specification process, and in particular into the two-step modelling process (i.e. model choice then model fit) in order to recover the target population model. Finally, it is unclear how one should use sample weights as an effective strategy to protect inference when the sample data can potentially be drawn from any one of the three scenarios: NIS, MSI and RBS.

Most of the research described in this thesis (Chapters 3-6) is devoted to the issue of how one should use sample weights in order answer the question (i) posed earlier. However, in Chapter 7 of this thesis we also address question (ii).

Chapter 3

A Theoretical Framework for Survey Data Analysis

3.1 Introduction

From a model-based perspective, a key consideration when analysing complex sample survey data is obtaining a model that fits the unknown target population as closely as possible given the data at hand. However, implementing a process that leads to such fit using data from complex surveys is complicated for anyone who is unfamiliar with the analysis of such data.

The four steps that make up the regression analysis process are set out in Heeringa *et al.* (2010). Applied to this situation, they reduce to three main components: model choice, model fit and model evaluation. Incorporating sample weights into these three components increases their complexity but also increases the chance of a reliable outcome. As a result, we use this three stage modelling framework throughout this thesis since our aim is to recover the underlying target population model given data obtained via a complex sample survey.

This chapter sets out the basic assumptions underpinning this modelling framework (Section 3.2), the construction of sample weights (Section 3.3), and the backward elimination procedure that we use in model selection (Section 3.4). The stages

of the suggested modelling process, which are based on the three components of regression analysis described in Chapter 2, are then explained and incorporated into the proposed procedure (Section 3.5). Finally, an implementation of the proposed procedure is provided (Section 3.6).

3.2 Basic Assumptions

Throughout we assume a modelling procedure that is based on the following assumptions.

- The sample data are drawn from a finite population of values.
- The sampling design employed is stratified sampling.
- There is complete response, or equivalently, any non-response is ignorable. This is necessary since non-response can be modelled as an extra, uncontrolled, stage of sample selection, and so its inclusion in our analysis adds extra, unnecessary, complication to what is already a complicated situation. Furthermore, even though non-response does not necessarily lead to bias in the associations among survey variables, survey practice shows that non-response can cause estimates to be biased (see Van Loon *et.al.*, 2003). By assuming complete response we avoid these issues.
- There are no outliers or unusual observations. Here again, we sidestep another important issue that is outside the scope of the research reported in this thesis. Outliers can have a significant impact on regression parameter estimates, and methods for detecting and dealing with them are necessary (see Li & Valliant, 2011). However, these methods are not the focus of this thesis.

3.3 Sample Weights

In this thesis, we assume that sample weights correspond to base weights as detailed in Kalton and Piesse (2011). Base weights are calculated as reciprocals of proba-

bilities of selection. The sum of these weights is the population size (Lohr, 1999; Kalton & Piesse, 2011), denoted here by N.

To illustrate, suppose that π_{hi} and w_{hi} denotes in turn the probability of selecting the ith unit in the hth stratum, and the sampling weight of the ith unit selected in the hth stratum. Then

$$w_{hi} = \pi_{hi}^{-1} \tag{3.1}$$

where $\pi_{hi} = n_h N_h^{-1}$; here N_h and n_h denote the population size in the hth stratum and the sample size in the hth stratum respectively.

3.4 Backward Elimination for Model Choice

The *Backward elimination process* is widely used when the aim is to choose a parsimonious model for a relation among variables of interest. The main reasons for employing this process are its attractive features of simplicity and greater practicality. Faraway (2005) identifies Backward elimination as the simplest procedure among the three main variable selection procedures (stepwise selection, forward selection and backward elimination) and notes the ease of its implementation with the `R` programme.

Two types of Backward elimination are employed in this thesis. The first is standard Backward elimination (that is, no weights are used in the model specification search), referred to below as *unweighted backward elimination.* The second, referred to below as *weighted backward elimination* is an extension of standard Backward elimination to the situation where sample weights are used in the model specification search. Note that irrespective of whether traditional (weighted) or model-based (unweighted) inference is eventually carried out, both of these approaches to model choice can be applied. That is, we consider both to be tools for model choice only.

The backward elimination procedure is fundamentally defined as a sequence of tests for the significance of candidate independent variables based on the following algorithm.

1. Fit a regression of the full model that consists of all independent variables. Note that the full model fitted by the ordinary least squares for the unweighted backward elimination method (or the weighted least squares for the weighted backward elimination method). Then, let the full model be the current model.

2. Based on the current model and a significance level of the test specified, choose the independent variable with the highest p-value. If that p-value is less than the significance level of the test specified, then stop the process; otherwise, go to the next step.

3. Modify the current model by removing the independent variable with the highest p-value from the model. Refit the model. If all nonsignificant independent variables are removed, then stop the process; otherwise, go back to step 2.

Both methods of Backward elimination were carried out using R software packages at a significance level of 0.05. Note that the 'survey' package was used for weighted backward elimination, with p-values and standard errors of regression estimators produced using the command 'svyglm( )' that is part of this package.

3.5 The Regression Modelling Process

The regression modelling procedure for sample survey data that underpins the research reported in this thesis is defined as consisting of the following steps:

1. **Model Choice:** A regression model for the sample data is identified. This model is supposed to summarise the relationship between the target response variable and independent variables. We assume that both unweighted and weighted backward elimination methods are used to identify the most parsimonious set of the regressors in each case. This implies that in general two distinct models will be identified, one via unweighted backward elimination and the other via weighted backward elimination.

2. **Model Fit:** Both models (or equivalently, sets of regressors) identified in the previous step are fitted to the target response variable using the available sample survey data. Here, again, we theoretically have the choice of either carrying out an unweighted model fit, or using the weights in the model fit. For the former we only consider the case where the unweighted fit corresponds to ordinary least squares (OLS), while in the latter, the fit is via weighted least squares (WLS), using the sample weights. Note that the end result is then (at least theoretically) four distinct fitted models for the target population - an OLS-based model derived from an unweighted model specification search; a WLS-based model derived from an unweighted model specification search; an OLS-based model derived from a weighted model specification search; and a WLS-based model derived from an weighted model specification search.

3. **Model Evaluation:** The previous steps will generally identify multiple models for the target population.The purpose of this final step is therefore to decide which of these competing models is the one most appropriate for 'explaining' the behaviour of the target variable in the target population given the available survey data. This in turn requires that we have procedures for choosing between different models (as would be the case if the weighted and unweighted backward elimination approaches selected different regressors), and procedures for choosing between different fits of the same model (as would be the case when we have to decide between an OLS fit and a WLS fit). We consider both issues below:

   (a) **Non-equivalent Regressor Sets:** This case arises when the unweighted and weighted backward elimination strategies used in the model choice step above leads to the identification of two distinct models - i.e. two different sets of regressors. In general these models will not be nested, and so standard methods for comparing them (e.g. likelihood ratios) cannot be used. Consequently we propose two approaches to model choice in this case. The first uses the test described by Vuong (1989) and the

second uses cross-validation (CV). Details of both approaches are set out in Chapter 5. Both approaches can be undertaken either ignoring the sample weights or incorporating them, so for any particular model comparison the eventual choice can depend on the outcomes of four test statistics:

- Strategy I: The unweighted Vuong test statistic (V_{NW}). See Vuong (1989).
- Strategy II: The weighted Vuong test statistic (V_W), obtained by extending the method described in Vuong (1989) to the situation where sample weights are available.
- Strategy III: The unweighted cross-validation statistic (CV_{NW}), obtained using the OLS fitting procedure in the test and validation sub-samples.
- Strategy IV: The weighted cross-validation statistic (CV_W), obtained using the WLS fitting procedure in the test and validation sub-samples.

(b) **Equivalent Regressor Sets:** Here the same model is specified by both model specification searches in the model choice step, and one needs to decide between an unweighted (OLS) fit and a weighted (WLS) fit of this model. We propose two approaches to testing whether sample weights should be used in the fit of the model. The first is the test proposed by DuMouchel and Duncan (1983). The second is the test proposed by Pesaran (1974), which treats the weighted and weighted fits as defining two non-nested fits, and then compares them. That is, we have two test statistics to consider:

- Test I: The test statistic proposed by DuMouchel and Duncan (1983) (DD).
- Test II: The test statistic proposed by Pesaran (1974) (PS).

Note that both tests depend on distributional assumptions. The Pesaran

test assumes that the underlying model errors are normally distributed while the DuMouchel and Duncan test assumes that the test statistic has an F-distribution. See Chapter 7 for more details.

A flowchart of this modelling process is set out in Figure 3.1.

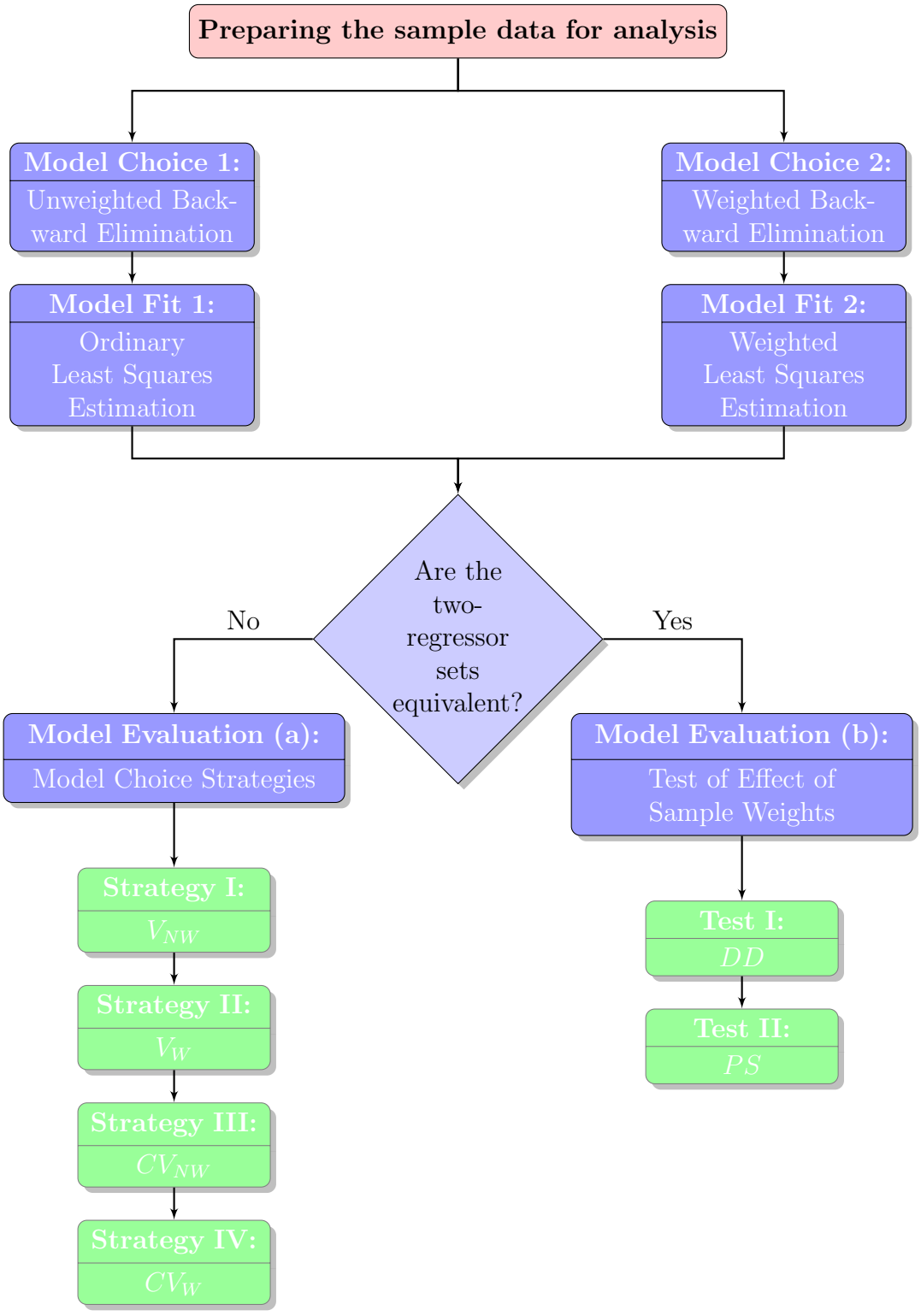

Figure 3.1: The proposed three-step decision tree procedure for regression modelling of complex sample survey data

3.6 Implementing the Proposed Modelling Procedure

The modelling procedure specified in the previous Sub-section and the decision process outlined in Figure 3.1 can be implemented as follows:

1. Prepare the sample data for analysis.

2. Use standard (unweighted) backward elimination to specify one candidate model for the target population. Call this Model U.

3. Use weighted backward elimination to specify another candidate model for the target population. Call this Model W.

4. Use ordinary least squares to fit Model U to the sample data.

5. Use weighted least squares to fit Model W to the sample data.

6. If Model U and Model W are not the same then use model search strategies I to IV (i.e. defined by the test statistics V_{NW}, V_W, CV_{NW} and CV_W) to make a decision on which model fit should be adopted. A voting strategy then decides the final model, i.e. the model that is most often favoured by these four strategies is adopted.

7. If Model U and Model W are the same then use tests I and II (i.e. defined by the test statistics DD and PS) to decide whether to use sample weights in the model fit. If both tests indicate the use of sample weights, then these are used in the final model fit. Otherwise, the decision on whether or not to use weights is determined by the outcome of the test statistics PS.

Note that in simulation results reported later in this thesis (see Chapter 6) we evaluate the efficiency of this modelling procedure by examining the relative bias, relative root mean square error and relative variance of the resulting parameter estimates.

Chapter 4

Misspecified Models, Targets of Inference and Bias

4.1 Introduction

A key objective of this thesis is to investigate the statistical properties of parameter estimators for population regression models under three scenarios: Non-Informative Sampling (NIS), Missing Stratification Information (MSI) and Response-Based Sampling (RBS). In this context, we focus on what is usually considered to be the most important statistical property of an estimator - its bias. However, bias can only be evaluated with respect to a target of inference, which in this thesis we consider to be a parameter of a population level regression model. This chapter investigates the bias properties of unweighted and sample-weighted regression parameter estimators under these three scenarios. Note that since the target of inference is a model characteristic, bias is evaluated from a model-based perspective. We also assume that the regression model underpinning the estimator makes use of all available regressors (i.e. model choice is not an issue).

To start, we consider the NIS situation. Here, since the sample and population regression models are the same, it is clear that the estimator of any parameter in this model will be unbiased.

Next, we consider the RBS situation. Here it is well-known known that the weighted estimators are design consistent (Pfeffermann, 1993; Lohr & Liu, 1994), provided the weights used correctly represent the inverses of the selection probabilities of the sample units. The same argument can be used to show that the weighted estimators are approximately unbiased from a model-based perspective, while the unweighted estimators are not. This is because both the weighted and unweighted estimators can be represented as ratios of population averages, and the model expectation of a weighted average, which also takes into account the underlying selection process, is unbiased for its corresponding population value, while this is not true of an unweighted average.

The case of most interest, therefore, is the MSI situation. Recollect that this corresponds to where the values of the response variable Y and a regressor X are available on the sample, but the population regression also depends on a stratifying variable Z, whose values are not available on the sample. Instead, the impact of the stratified sampling design is reflected in the values of the sample weights, which are the ratios of the population number in a stratum to the corresponding sample number in the stratum. Here, although many studies (reviewed in Chapter 2) have recommended that the weighted estimators indicated by the design-based approach should be adopted in spite of model misspecification (DuMouchel & Duncan, 1983; Pfeffermann & Holmes, 1985; Kott, 1991), it is not at all clear whether this advice remains appropriate when a model-based perspective is adopted. Consequently, we now investigate the model-bias properties of weighted and unweighted regression parameter estimators under MSI. Our aim is expository, so we only consider population models where the regressor X is a binary variable and which correspond to either simple linear regression (Y is scalar) or to logistic regression (Y is binary). Our reason for the restriction of X to being binary is simple: In most social research regressors are typically categorical., and the simplest version of a categorical variable is a binary variable. Consequently we assume that the regressor X is binary. We also restrict our analysis to the case where the missing stratification information

Z corresponds to a binary variable, i.e. the underlying sample design has just two strata.

4.2 MSI-Generated Bias for the Linear Regression Model

4.2.1 Targets of Inference

We start by noting that there are in fact two population level linear regression models that can define the targets of inference in this case. These are the linear regression model defined by the 'complete' population data (Y, X and Z):

$$E(Y|X,Z) = a + bX + cZ \tag{4.1}$$

and the linear regression model defined by the 'incomplete' sample data (Y and X):

$$E(Y|X) = a^* + b^*X. \tag{4.2}$$

Here Y is a continuous response variable, X is a binary independent variable and Z is a binary stratifying variable that can be used to classify the finite population data of size N into two strata.

We now show how Equation (4.2) can be obtained from Equation (4.1). First, observe that we can write

$$\begin{aligned} E(y_i|x_i) &= E[E(y_i|x_i,z_i)|x_i] \\ &= E(a + bx_i + cz_i|x_i) \\ &= a + bx_i + cE(z_i|x_i; i \in s) \ \ ; i = 1, ..., n. \end{aligned} \tag{4.3}$$

Put $P_0 = Pr(z_i = 1|x_i = 0; i \in s)$, and $P_1 = Pr(z_i = 1|x_i = 1; i \in s)$. Then

$E(z_i|x_i; i \in s)$ in Equation (4.3) can be written as

$$\begin{aligned} E(z_i|x_i; i \in s) &= Pr(z_i = 1|x_i; i \in s) \\ &= x_i Pr(z_i = 1|x_i = 1; i \in s) + (1 - x_i) Pr(z_i = 1|x_i = 0; i \in s) \\ &= P_0 + (P_1 - P_0)x_i \end{aligned}$$

Substituting $E(z_i|x_i; i \in s)$ above back into Equation (4.3), we then obtain

$$\begin{aligned} E(y_i|x_i) &= a + bx_i + cE(z_i|x_i; i \in s) \\ &= a + bx_i + c[P_0 + (P_1 - P_0)x_i] \\ &= (a + cP_0) + [b + c(P_1 - P_0)]x_i \\ &= a^* + b^* x_i \end{aligned} \tag{4.4}$$

where $a^* = a + cP_0$ and $b^* = b + c(P_1 - P_0)$.

In what follows, the regression coefficient parameters (a, b, c) in Equation (4.1) will be referred to as the parameters of the 'conditional model' (i.e. the model that conditions on Z), and regression coefficient parameters (a^*, b^*) in Equation (4.2) will be referred to as the parameters of the 'unconditional model' (i.e. the model that averages over Z). Throughout we assume that the population values of Y are actually generated via the conditional model, i.e. via Equation (4.1). We show that both weighted and unweighted estimators based on the unconditional model (the natural one to fit given the sample data) are then biased if the targets of inference are the conditional model parameters. However, these estimators are unbiased if the targets of inference are the unconditional model parameters.

4.2.2 Assumptions Used in the Derivation

Our derivation of modelling bias depends on two main assumptions. These are:

$$E(\sum_s x_i) = \sum_{i=1}^{n} x_i$$

and

$$E(\sum_s w_i x_i) = \sum_{i=1}^{N} x_i.$$

Here w_i is the sample weight of the ith element in the sample s, and we have $\sum_s w_i = N$.

Note that the expectation of the summation term in the first assumption corresponds to the sample size when $X = 1$, denoted by n_1. In addition, we define n_0 as the sample size when $X = 0$, n_{00} as the sample size when $X = 0$ and $Z = 0$, n_{01} as the sample size when $X = 0$ and $Z = 1$, n_{10} as the sample size when $X = 1$ and $Z = 0$ and n_{11} as the sample size when $X = 1$ and $Z = 1$. Similarly, the expectation of the summation term in the second assumption is equivalent to the population size when $X = 1$, denoted by N_1. In addition, we suppose that N_0 is the population size when $X = 0$, N_{00} is the population size when $X = 0$ and $Z = 0$ and N_{01} is the population size when $X = 0$ and $Z = 1$, N_{10} is the population size when $X = 1$ and $Z = 0$ and N_{11} is the population size when $X = 1$ and $Z = 1$.

4.2.3 MSI-Generated Bias of Unweighted Estimators

In the following, we show that the unweighted intercept and slope estimators are biased when the targets of inference are the conditional model parameters, see Equation (4.1). In contrast, these estimators are unbiased when the targets of inference are the unconditional model parameters, see Equation (4.2).

As in Lohr (1999), the intercept and slope estimators defined by an unweighted simple linear regression fit can be expressed as

$$\hat{a} = \sum_s \frac{1}{n} \left[1 - \frac{x_i \sum_s x_j - \frac{(\sum_s x_j)^2}{n}}{\sum_s x_j^2 - \frac{(\sum_s x_j)^2}{n}} \right] y_i \tag{4.5}$$

and

$$\hat{b} = \sum_s \left[\frac{x_i - \frac{\sum_s x_j}{n}}{\sum_s x_j^2 - \frac{(\sum_s x_j)^2}{n}} \right] y_i. \tag{4.6}$$

By taking expectations on both sides of Equation (4.5)-(4.6), we obtain

$$\begin{aligned}
E(\hat{a}|X,Z) &= \sum_s \frac{1}{n}\left[1 - \frac{x_i \sum_s x_j - \frac{(\sum_s x_j)^2}{n}}{\sum_s x_j^2 - \frac{(\sum_s x_j)^2}{n}}\right] E(y_i|x_i,z_i) \\
&= \sum_s \frac{1}{n}\left[1 - \frac{x_i \sum_s x_j - \frac{(\sum_s x_j)^2}{n}}{\sum_s x_j^2 - \frac{(\sum_s x_j)^2}{n}}\right] (a + bx_i + cz_i) \\
&= \frac{n_{00}}{n}\left[1 - \frac{-\frac{n_1^2}{n}}{n_1 - \frac{n_1^2}{n}}\right] a + \frac{n_{01}}{n}\left[1 - \frac{n_1 - \frac{n_1^2}{n}}{n_1 - \frac{n_1^2}{n}}\right] (a+c) \\
&= \frac{n_{00}}{n}\left[1 + \frac{n_1}{n_0}\right] a + \frac{n_{01}}{n}\left[1 + \frac{n_1}{n_0}\right] (a+c) \\
&= a + \left(\frac{n_{01}}{n_0}\right) c \qquad (4.7)
\end{aligned}$$

and

$$\begin{aligned}
E(\hat{b}|X,Z) &= \sum_s \left[\frac{x_i - \frac{\sum_s x_j}{n}}{\sum_s x_j^2 - \frac{(\sum_s x_j)^2}{n}}\right] E(y_i|x_i,z_i) \\
&= \sum_s \left[\frac{x_i - \frac{\sum_s x_j}{n}}{\sum_s x_j^2 - \frac{(\sum_s x_j)^2}{n}}\right] (a + bx_i + cz_i) \\
&= n_{00}\left[\frac{-\frac{n_1}{n}}{n_1 - \frac{n_1^2}{n}}\right] a + n_{01}\left[\frac{-\frac{n_1}{n}}{n_1 - \frac{n_1^2}{n}}\right] (a+c) \\
&\quad + n_{10}\left[\frac{1 - \frac{n_1}{n}}{n_1 - \frac{n_1^2}{n}}\right] (a+b) + n_{11}\left[\frac{1 - \frac{n_1}{n}}{n_1 - \frac{n_1^2}{n}}\right] (a+b+c) \\
&= b + \left(\frac{n_{11}}{n_1} - \frac{n_{01}}{n_0}\right) c \qquad (4.8)
\end{aligned}$$

Clearly, the two estimators in Equation (4.5)-(4.6) will be biased if the targets of inference are the conditional model parameters (a, b) in Equation (4.1). In addition, it can be seen that the bias terms (i.e. $\left(\frac{n_{01}}{n_0}\right) c$, $\left(\frac{n_{11}}{n_1} - \frac{n_{01}}{n_0}\right) c$) depend on the relationship between the target response variable Y and the stratifying variable Z (i.e. via the conditional model parameter c) - the stronger the relationship, the more the bias. On the other hand, the bias decreases as the relationship between Y and Z becomes weaker. Furthermore, the unweighted slope estimator $\hat{b}$ will be unbiased for the slope parameter b of the conditional model if the sample distribution of Z is effectively independent of the sample distribution of X, i.e. $\frac{n_{11}}{n_1} = \frac{n_{01}}{n_0}$.

Similarly, the bias of the unweighted parameter estimators under the unconditional model, see Equation (4.2), can be obtained by taking expectations on both sides of Equation (4.5)-(4.6).This leads to

$$\begin{aligned} E(\hat{a}|X) &= \sum_s \frac{1}{n}\left[1 - \frac{x_i \sum_s x_j - \frac{(\sum_s x_j)^2}{n}}{\sum_s x_j^2 - \frac{(\sum_s x_j)^2}{n}}\right] E(y_i|x_i) \\ &= \sum_s \frac{1}{n}\left[1 - \frac{x_i \sum_s x_j - \frac{(\sum_s x_j)^2}{n}}{\sum_s x_j^2 - \frac{(\sum_s x_j)^2}{n}}\right] (a^* + b^* x_i) \\ &= \frac{n_0}{n}\left[1 - \frac{-\frac{n_1^2}{n}}{n_1 - \frac{n_1^2}{n}}\right] a^* + \frac{n_1}{n}\left[1 - \frac{n_1 - \frac{n_1^2}{n}}{n_1 - \frac{n_1^2}{n}}\right] (a^* + b^*) \\ &= a^* \end{aligned} \tag{4.9}$$

and

$$\begin{aligned} E(\hat{b}|X) &= \sum_s \left[\frac{x_i - \frac{\sum_s x_j}{n}}{\sum_s x_j^2 - \frac{(\sum_s x_j)^2}{n}}\right] E(y_i|x_i) \\ &= \sum_s \left[\frac{x_i - \frac{\sum_s x_j}{n}}{\sum_s x_j^2 - \frac{(\sum_s x_j)^2}{n}}\right] (a^* + b^* x_i) \\ &= n_0 \left[\frac{-\frac{n_1}{n}}{n_1 - \frac{n_1^2}{n}}\right] a^* + n_1 \left[\frac{1 - \frac{n_1}{n}}{n_1 - \frac{n_1^2}{n}}\right] (a^* + b^*) \\ &= b^* \end{aligned} \tag{4.10}$$

That is, the unweighted regression parameters estimators in Equations (4.5)-(4.6) are unbiased if the targets of inference are the unconditional model parameters (a^*, b^*).

4.2.4 MSI-Generated Bias of Weighted Estimators

As in the previous subsection, we now show that the weighted estimators of the intercept and slope parameters are biased when the targets of inference are the conditional model parameters in Equation (4.1), but are unbiased when the targets of inference are the unconditional model parameters in Equation (4.2).

The weighted intercept and slope estimators are:

$$\hat{a}_w = \sum_s \frac{1}{N}\left[w_i - \frac{x_i w_i \sum_s w_j x_j - \frac{w_i(\sum_s w_j x_j)^2}{N}}{\sum_s w_j x_j^2 - \frac{(\sum_s w_j x_j)^2}{N}}\right] y_i \tag{4.11}$$

and

$$\hat{b}_w = \sum_s \left[\frac{w_i x_i - w_i \frac{\sum_s w_j x_j}{N}}{\sum_s w_j x_j^2 - \frac{(\sum_s w_j x_j)^2}{N}}\right] y_i \tag{4.12}$$

where w_i is the sample weight of the ith element in the sample s.

The bias of these estimators for the conditional model parameters in Equation (4.1) can be obtained by taking expectations on both sides of Equations (4.11)-(4.12). This leads to

$$\begin{aligned}
E(\hat{a}_w|X,Z) &= \sum_s \frac{1}{N}\left[w_i - \frac{x_i w_i \sum_s w_j x_j - w_i \frac{(\sum_s w_j x_j)^2}{N}}{\sum_s w_j x_j^2 - \frac{(\sum_s w_j x_j)^2}{N}}\right] E(y_i|x_i,z_i) \\
&= \sum_s \frac{1}{N}\left[w_i - \frac{x_i w_i \sum_s w_j x_j - w_i \frac{(\sum_s w_j x_j)^2}{N}}{\sum_s w_j x_j^2 - \frac{(\sum_s w_j x_j)^2}{N}}\right] (a + bx_i + cz_i) \\
&= \frac{1}{N}\left[N_{00} + \frac{N_{00}\frac{N_1^2}{N}}{N_1 - \frac{N_1^2}{N}}\right] a + \frac{1}{N}\left[N_{01} + \frac{N_{01}\left(\frac{N_1^2}{N}\right)}{N_1 - \frac{N_1^2}{N}}\right] (a+c) \\
&= a + \left(\frac{N_{01}}{N_0}\right) c
\end{aligned} \tag{4.13}$$

and

$$\begin{aligned}
E(\hat{b}_w|X,Z) &= \sum_s \left[\frac{w_i x_i - w_i \frac{\sum_s w_j x_j}{N}}{\sum_s w_j x_j^2 - \frac{(\sum_s w_j x_j)^2}{N}}\right] E(y_i|x_i,z_i) \\
&= \sum_s \left[\frac{w_i x_i - w_i \frac{\sum_s w_j x_j}{N}}{\sum_s w_j x_j^2 - \frac{(\sum_s w_j x_j)^2}{N}}\right] (a + bx_i + cz_i) \\
&= \left[\frac{-N_{00}\frac{N_1}{N}}{N_1 - \frac{N_1^2}{N}}\right] a + N_{01}\left[\frac{-\frac{N_1}{N}}{N_1 - \frac{N_1^2}{N}}\right] (a+c) \\
&\quad + N_{10}\left[\frac{1 - \frac{N_1}{N}}{N_1 - \frac{N_1^2}{N}}\right] (a+b) + N_{11}\left[\frac{1 - \frac{N_1}{N}}{N_1 - \frac{N_1^2}{N}}\right] \\
&= b + \left(\frac{N_{11}}{N_1} - \frac{N_{01}}{N_0}\right) c
\end{aligned} \tag{4.14}$$

As we can see, the results set out above in Equations (4.13)-(4.14) are similar to the results shown in Equations (4.7)-(4.8). We interpret the bias terms (i.e.

$\left(\frac{N_{01}}{N_0}\right)c$ and $\left(\frac{N_{11}}{N_1} - \frac{N_{01}}{N_0}\right)c$) here similarly to those defined by Equations (4.7)-(4.8); Here N_0 and N_1 are the corresponding population size when $X = 0$ and $X = 1$ respectively, N_{01} denotes the population size when $X = 0$ and $Z = 1$, and N_{11} denotes the population size when $X = 1$ and $Z = 1$. It is clear that (4.5)-(4.6) will therefore be biased if the targets of inference are the conditional model parameters, with the exception that the weighted slope estimator $\hat{b}_w$ will be unbiased for the slope parameter b if the population distribution of Z is effectively independent of the population distribution of X, i.e. when $\frac{N_{01}}{N_0} = \frac{N_{11}}{N_1}$.

When the targets of inference are the unconditional model parameters in Equation (4.2), the bias of the weighted estimators can be obtained by taking expected values on both sides of Equations (4.11)-(4.12) conditioning only on X. This leads to

$$\begin{aligned}
E(\hat{a}_w|X) &= \sum_s \frac{1}{N}\left[w_i - \frac{x_i w_i \sum_s w_j x_j - w_i \frac{(\sum_s w_j x_j)^2}{N}}{\sum_s w_j x_j^2 - \frac{(\sum_s w_j x_j)^2}{N}}\right] E(y_i|x_i) \\
&= \sum_s \frac{1}{N}\left[w_i - \frac{x_i w_i \sum_s w_j x_j - w_i \frac{(\sum_s w_j x_j)^2}{N}}{\sum_s w_j x_j^2 - \frac{(\sum_s w_j x_j)^2}{N}}\right] (a^* + b^* x_i) \\
&= \frac{1}{N}\left[N_0 + N_0\left(\frac{\frac{N_1^2}{N}}{N_1 - \frac{N_1^2}{N}}\right)\right] a^* + \frac{1}{N}\left[N_1 - \frac{N_1^2 - \frac{N_1^3}{N}}{N_1 - \frac{N_1^2}{N}}\right](a^* + b^*) \\
&= a^* \qquad (4.15)
\end{aligned}$$

and

$$\begin{aligned}
E(\hat{b}_w|X) &= \sum_s \left[\frac{w_i x_i - w_i \frac{\sum_s w_j x_j}{N}}{\sum_s w_j x_j^2 - \frac{(\sum_s w_j x_j)^2}{N}}\right] E(y_i|x_i) \\
&= \sum_s \left[\frac{w_i x_i - w_i \frac{\sum_s w_j x_j}{N}}{\sum_s w_j x_j^2 - \frac{(\sum_s w_j x_j)^2}{N}}\right] (a^* + b^* x_i) \\
&= \left[\frac{-N_0 \frac{N_1}{N}}{N_1 - \frac{N_1^2}{N}}\right] a^* + \left[\frac{N_1 - \frac{N_1^2}{N}}{N_1 - \frac{N_1^2}{N}}\right](a^* + b^*) \\
&= b^* \qquad (4.16)
\end{aligned}$$

That is, the two weighted estimators defined by Equations (4.11)-(4.12) are un-

biased under MSI if the targets of inference are the unconditional model parameters in Equation (4.2).

4.3 MSI-Generated Bias for the Logistic Regression Model

4.3.1 Targets of Inference

As in the linear case, there are two population level logistic regression models that can define the targets of inference. These are the conditional logistic regression model defined by the 'complete' population data (Y, X and Z):

$$logitPr(Y = 1|X, Z) = a + bX + cZ \tag{4.17}$$

and the unconditional logistic regression model defined by the 'incomplete' sample data (Y and X):

$$logitPr(Y = 1|X) = a + bX + g(f(X)) \tag{4.18}$$

where $g(f(X)) = log\left(\frac{1+f(X)}{1-e^{a+bX}f(X)}\right)$; $f(X) = \frac{e^c-1}{1+e^{a+bX+c}}[P_0 + (P_1 - P_0)X]$. Here Y is a binary response variable; the other variables (X and Z), P_0 and P_1 are the same as in the previous section.

In order to show how Equation (4.18) can be obtained from Equation (4.17), we note that since

$$Pr(Y = 1|X, Z) = \frac{e^{a+bX+cZ}}{1 + e^{a+bX+cZ}} = E(Y|X, Z),$$

we then have

$$\begin{aligned}
E(Y|X) &= E[E(Y|X,Z)|X] \\
&= E\left[\frac{e^{a+bX+cZ}}{1+e^{a+bX+cZ}}|X\right] \\
&= \frac{e^{a+bX+c}}{1+e^{a+bX+c}}[P_1X + P_0(1-X)] + \frac{e^{a+bX}}{1+e^{a+bX}}[(1-P_1)X + (1-P_0)(1-X)] \\
&= \frac{e^{a+bX}}{1+e^{a+bX}}\left[1 + \left(\frac{e^c-1}{1+e^{a+bX+c}}\right)\{P_0 + (P_1-P_0)X\}\right] \\
&= \frac{e^{a+bX}}{1+e^{a+bX}}[1+f(X)] \\
&= Pr(Y=1|X).
\end{aligned}$$

Taking natural logarithms above, we obtain

$$logPr(Y=1|X) = a + bX - log(1+e^{a+bX}) + log[1+f(X)].$$

Furthermore, since

$$\begin{aligned}
1 - Pr(Y=1|X) &= 1 - \frac{e^{a+bX}}{1+e^{a+bX}}[1+f(X)] \\
&= \frac{1-e^{a+bX}f(X)}{1+e^{a+bX}},
\end{aligned}$$

it follows that

$$\begin{aligned}
logPr(Y=1|X) &= a + bX - log(1+e^{a+bX}) + log[1+f(X)] \\
&= a + bX - log\left(\frac{1-e^{a+bX}f(X)}{1-Pr(Y=1|X)}\right) + log[1+f(X)].
\end{aligned}$$

Therefore

$$logitPr(Y=1|X) = a + bX + log\left[\frac{1+f(X)}{1-e^{a+bX}f(X)}\right].$$

In order to obtain Equation (4.18) from Equation (4.17), it only remains to put $f(X) = \frac{e^c-1}{1+e^{a+bX+c}}[P_0 + (P_1-P_0)X]$.

In what follows we refer to the regression coefficients (a, b, c) in Equation (4.17) below as the parameters of the conditional model, and the regression coefficients (a, b) in Equation (4.18) as the parameters of the unconditional model. Again, as in the linear case, we assume that the population values of Y are actually generated through the conditional model, i.e. through Equation (4.17). We now show that both the unweighted and weighted estimators of the intercept and slope of a logistic fit to the sample data are biased if the targets of inference are the corresponding conditional model parameters. In contrast, these estimators (and in particular the slope estimator) are approximately unbiased if the targets of inference are the parameters of the unconditional model.

4.3.2 Assumptions Used in the Derivation

The assumptions set out in Subsection 4.2.2 are used again in the development below. Also, since parameter estimators in the case of logistic regression are only implicitly defined as solutions of estimating equations, a further assumption is required that allows these parameter estimators to be approximated by first order Taylor series expansions around the true values (a, b) of the conditional model in Equation (4.17).

4.3.3 MSI-Generated Bias of Unweighted Estimators

First order approximations to the biases of the parameters of the unweighted logistic fit under the population models defined by Equations (4.17)-(4.18) are set out below. As usual, we assume that the sample s corresponds to a set of data that does not contain the values of Z. Our 'sample model' for Y is then

$$\widetilde{Pr}(y_i = 1|x_i) = \frac{e^{a^*+b^*x_i}}{1 + e^{a^*+b^*x_i}} \;\; ; i = 1, ..., n.$$

It follows that we can write

$$logit\widetilde{Pr}(y_i = 1|x_i) = a^* + b^*x_i \;\; ; i = 1, ..., n. \tag{4.19}$$

where a^* and b^* are the regression coefficients of the sample model.

The logistic function is nonlinear. Consequently exact expressions for the unweighted estimators $\hat{a^*}$ and $\hat{b^*}$ are not available, and the use of approximations becomes an important tool when assessing the statistical properties of these estimators. One popular approach to approximation uses Taylor series linearisation (Casella & Berger, 2002). In the following, we illustrate how this approach can be used to obtain approximations to estimators of the regression coefficients a^* and b^* in Equation (4.19).

For the pair (x_i, y_i) corresponding to the ith element of the sample s, put

$$h(x_i) = [\widetilde{Pr}(y_i = 1|x_i)]^{y_i}[1 - \widetilde{Pr}(y_i = 1|x_i)]^{1-y_i}.$$

Let $\beta^* = (a^*, b^*)^T$. Since the n observations are assumed to be independent, the log-likelihood can be expressed by

$$\begin{aligned} l(\beta^*) &= log\prod_{i=1}^{n} h(x_i) \\ &= \sum_{i=1}^{n}\left[y_i log\widetilde{Pr}(y_i = 1|x_i) + (1 - y_i)log(1 - \widetilde{Pr}(y_i = 1|x_i))\right] \\ &= \sum_{i=1}^{n}\left[y_i(a^* + b^*x_i) - log(1 + e^{a^*+b^*x_i})\right]. \end{aligned}$$

By definition, $l'(\hat{\beta}^*) = 0$. We therefore use a first order expansion of $l'(\beta^*)$ around the true values $\beta = (a, b)^T$ in Equation (4.17) to re-express this identity in the form

$$l'(\hat{\beta}^*) \approx l'(\beta) + l''(\beta)(\hat{\beta}^* - \beta) = 0$$

implying

$$\hat{\beta}^* \approx \beta - \frac{l'(\beta)}{l''(\beta)}. \tag{4.20}$$

In the context of the unweighted estimators $\hat{a}^*$ and $\hat{b}^*$, this leads to the approx-

imations:

$$\hat{a}^* \approx a + \frac{\sum_s \{y_i - logit^{-1}(a + bx_i)\}}{\sum_s logit^{-1}(a + bx_i)} \tag{4.21}$$

and

$$\hat{b}^* \approx b + \frac{\sum_s x_i\{y_i - logit^{-1}(a + bx_i)\}}{\sum_s x_i^2 logit^{-1}(a + bx_i)}. \tag{4.22}$$

First order approximations to the biases of $\hat{a}^*$ and $\hat{b}^*$ under the conditional model defined by Equation (4.17) can be obtained by taking appropriate expectations on both sides of Equations (4.21)-(4.22). This leads to

$$\begin{aligned} E(\hat{a}^*|X, Z) &\approx a + \frac{\sum_s \{E(y_i|x_i, z_i) - logit^{-1}(a + bx_i)\}}{\sum_s logit^{-1}(a + bx_i)} \\ &= a + \frac{\sum_s \{logit^{-1}(a + bx_i + cz_i) - logit^{-1}(a + bx_i)\}}{\sum_s logit^{-1}(a + bx_i)} \\ &= a + \frac{n_{01} logit^{-1}(a)\frac{e^c-1}{1+e^{a+c}} + n_{11} logit^{-1}(a + b)\frac{e^c-1}{1+e^{a+b+c}}}{n_0 logit^{-1}(a) + n_1 logit^{-1}(a + b)} \end{aligned} \tag{4.23}$$

and

$$\begin{aligned} E(\hat{b}^*|X, Z) &\approx b + \frac{\sum_s x_i\{E(y_i|x_i, z_i) - logit^{-1}(a + bx_i)\}}{\sum_s x_i^2 logit^{-1}(a + bx_i)} \\ &= b + \frac{\sum_s x_i\{logit^{-1}(a + bx_i + cz_i) - logit^{-1}(a + bx_i)\}}{\sum_s x_i^2 logit^{-1}(a + bx_i)} \\ &= b + \frac{n_{11}}{n_1}\left(\frac{e^c - 1}{1 + e^{a+b+c}}\right). \end{aligned} \tag{4.24}$$

It is obvious from Equations (4.23)-(4.24) above that the estimators $\hat{a}^*$ and $\hat{b}^*$ are biased for the conditional model parameters a and b unless the stratifying variable Z is not related to the target response variable Y (i.e. $c = 0$). Although these biases (i.e. the second terms in the right-hand sides of Equations (4.23)-(4.24)) cannot be expressed explicitly, their magnitudes mainly depend on the value of the conditional model parameter c, as we now demonstrate.

Suppose that the two terms related to the conditional model parameter c are $U = \frac{e^c-1}{1+e^{a+c}}$ and $V = \frac{e^c-1}{1+e^{a+b+c}}$. For negative values of a (as is common), both U and V approach positive infinity as c approaches positive infinity; whereas both U and

V converge to zero as c approaches negative infinity. The former behaviour of U and V indicates that the biases of $\hat{a}^*$ and $\hat{b}^*$ (as estimators of a and b) will increase in a situation where the missing stratifying variable Z is closely related to the target response variable Y. In contrast, the latter behaviour of U and V indicates that these biases will decrease in a situation where the missing stratifying variable Z is not related to the target response variable Y.

Turning now to the unconditional model, see Equation (4.18), we again take appropriate expectations on both sides of Equations (4.21)-(4.22) to obtain:

$$\begin{aligned} E(\hat{a}^*|X) &\approx a + \frac{\sum_s \{E(y_i|x_i) - logit^{-1}(a+bx_i)\}}{\sum_s logit^{-1}(a+bx_i)} \\ &= a + \frac{\sum_s \{logit^{-1}(a+bx_i)[1+f(x_i)] - logit^{-1}(a+bx_i)\}}{\sum_s logit^{-1}(a+bx_i)} \\ &= a + \frac{n_0 logit^{-1}(a)\frac{e^c-1}{1+e^{a+c}}P_0 + n_1 logit^{-1}(a+b)\frac{e^c-1}{1+e^{a+b+c}}P_1}{n_0 logit^{-1}(a) + n_1 logit^{-1}(a+b)} \end{aligned} \quad (4.25)$$

and

$$\begin{aligned} E(\hat{b}^*|X) &\approx b + \frac{\sum_s x_i\{E(y_i|x_i) - logit^{-1}(a+bx_i)\}}{\sum_s x_i^2 logit^{-1}(a+bx_i)} \\ &= b + \frac{\sum_s x_i\{logit^{-1}(a+bx_i)[1+f(x_i)] - logit^{-1}(a+bx_i)\}}{\sum_s x_i^2 logit^{-1}(a+bx_i)} \\ &= b + \left(\frac{e^c-1}{1+e^{a+b+c}}\right)P_1. \end{aligned} \quad (4.26)$$

The unconditional model biases shown in Equations (4.25)-(4.26) are very similar to those shown in Equations (4.23)-(4.24). In particular, we see that the bias terms here (i.e. the second terms on the right-hand sides of Equations (4.25)-(4.26)) are equal to the corresponding bias terms in Equations (4.23)-(4.24) provided we have unweighted sample balance on Z, i.e. $P_0 = Pr(Z=1|X=0) = \frac{n_{01}}{n_0}$ and $P_1 = Pr(Z=1|X=1) = \frac{n_{11}}{n_1}$.

4.3.4 MSI-Generated Bias of Weighted Estimators

Here sample weights are incorporated into the iterative process used to estimate the logistic model parameters, leading to the weighted estimators $\hat{a}_w^*$ and $\hat{b}_w^*$. We again assume that we can approximate these weighted estimators via a first order Taylor Series expansion around the true values (a, b) based on Equation (4.17). This leads to the approximations:

$$\hat{a}_w^* \approx a + \frac{\sum_s w_i\{y_i - logit^{-1}(a + bx_i)\}}{\sum_s w_i logit^{-1}(a + bx_i)} \tag{4.27}$$

and

$$\hat{b}_w^* \approx b + \frac{\sum_s w_i x_i\{y_i - logit^{-1}(a + bx_i)\}}{\sum_s w_i x_i^2 logit^{-1}(a + bx_i)}. \tag{4.28}$$

By taking expectations conditional on X and Z on both sides of Equations (4.27)-(4.28), we obtain first order approximations to the biases of the weighted estimators when the targets of inference are the unconditional model parameters defined in Equation (4.17). These are:

$$\begin{aligned} E(\hat{a}_w^*|X, Z) &\approx a + \frac{\sum_s w_i\{E(y_i|x_i, z_i) - logit^{-1}(a + bx_i)\}}{\sum_s w_i logit^{-1}(a + bx_i)} \\ &= a + \frac{\sum_s w_i\{logit^{-1}(a + bx_i + cz_i) - logit^{-1}(a + bx_i)\}}{\sum_s w_i logit^{-1}(a + bx_i)} \\ &= a + \frac{N_{01} logit^{-1}(a)\frac{e^c-1}{1+e^{a+c}} + N_{11} logit^{-1}(a + b)\frac{e^c-1}{1+e^{a+b+c}}}{N_0 logit^{-1}(a) + N_1 logit^{-1}(a + b)} \end{aligned} \tag{4.29}$$

and

$$\begin{aligned} E(\hat{b}_w^*|X, Z) &\approx b + \frac{\sum_s w_i x_i\{E(y_i|x_i, z_i) - logit^{-1}(a + bx_i)\}}{\sum_s w_i x_i^2 logit^{-1}(a + bx_i)} \\ &= b + \frac{\sum_s w_i x_i\{logit^{-1}(a + bx_i + cz_i) - logit^{-1}(a + bx_i)\}}{\sum_s w_i x_i^2 logit^{-1}(a + bx_i)} \\ &= b + \frac{N_{11}}{N_1}\left(\frac{e^c - 1}{1 + e^{a+b+c}}\right). \end{aligned} \tag{4.30}$$

It is clear again that the two estimators in Equations (4.27)-(4.28) are biased for

the complete model parameters (a, b) in Equation (4.17), except in the situation where there is no relationship between the target response variable Y and the stratifying variable Z (i.e. $c = 0$). The magnitudes of the bias terms (i.e. the second terms in the right-hand side of Equations (4.29)-(4.30)) are driven by the value of the complete model parameter c. Recall that $U = \frac{e^c-1}{1+e^{a+c}}$ and $V = \frac{e^c-1}{1+e^{a+b+c}}$. For negative values of a, both U and V approach positive infinity when c approaches positive infinity. This means that the biases of the two estimators (i.e. $\hat{a}_w^*$, $\hat{b}_w^*$) will increase if the variables Y and Z are positively related to each other. On the other hand, values of U and V are both close to zero when c approaches negative infinity, which means that the biases of these estimators will decrease if the variables Y and Z are not related to each other.

First order approximations to the biases of $\hat{a}_w^*$ and $\hat{b}_w^*$ under the unconditional model defined by Equation (4.18) are obtained by taking expected values under this model on both sides of Equations (4.27)-(4.28). This leads to

$$\begin{aligned} E(\hat{a}_w^*|X) &\approx a + \frac{\sum_s w_i\{E(y_i|x_i) - logit^{-1}(a+bx_i)\}}{\sum_s w_i logit^{-1}(a+bx_i)} \\ &= a + \frac{\sum_s w_i\{logit^{-1}(a+bx_i)f(x_i) - logit^{-1}(a+bx_i)\}}{\sum_s w_i logit^{-1}(a+bx_i)} \\ &= a + \frac{N_0 logit^{-1}(a)\frac{e^c-1}{1+e^{a+c}}P_0 + N_1 logit^{-1}(a+b)\frac{e^c-1}{1+e^{a+b+c}}P_1}{N_0 logit^{-1}(a) + N_1 logit^{-1}(a+b)} \end{aligned} \quad (4.31)$$

and

$$\begin{aligned} E(\hat{b}_w^*|X) &\approx b + \frac{\sum_s w_i x_i\{E(y_i|x_i) - logit^{-1}(a+bx_i)\}}{\sum_s w_i x_i^2 logit^{-1}(a+bx_i)} \\ &= b + \frac{\sum_s w_i x_i\{logit^{-1}(a+bx_i)f(x_i) - logit^{-1}(a+bx_i)\}}{\sum_s w_i x_i^2 logit^{-1}(a+bx_i)} \\ &= b + \left(\frac{e^c-1}{1+e^{a+b+c}}\right)P_1. \end{aligned} \quad (4.32)$$

The bias terms on the right-hand side of Equations (4.31)-(4.32) can be interpreted in the same way as the bias terms in Equations (4.29)-(4.30). In particular, we see that Equation (4.31) will be the same as Equation (4.29) and Equation (4.32)

will be the same as Equation (4.30) when we have weighted sample balance on Z, i.e. when $P_0 = Pr(Z = 1|X = 0) = \frac{N_{01}}{N_0}$ and $P_1 = Pr(Z = 1|X = 1) = \frac{N_{11}}{N_1}$.

4.4 Conclusion

This Chapter has focussed on bias due to missing stratification information (MSI) when modelling sample survey data. In particular, we consider the special case of the MSI situation where the values of the variable Z used for classifying the finite population data into two strata are omitted from the sample data set and we develop expressions for two different types of bias (with respect to the actual model that generated the data, the conditional model, and with respect to a model that averages over the population values of Z, the unconditional model) under two types of simple regression models (linear and logistic), and two approaches to model fitting (unweighted and weighted). Our results indicate the following conclusions can be drawn:

- **Linear:** If the targets of inference are the conditional model parameters, then both unweighted and weighted estimators of these parameters are biased. On the other hand, if the targets of inference are the unconditional model parameters, then both unweighted and weighted estimates of these parameters are unbiased.

- **Logistic:** The estimators of both the intercept and slope coefficients derived from a logistic model fit under both the unweighted and the weighted approaches are biased no matter how these parameters are defined (i.e. with respect to the conditional model or with respect to the unconditional model). This is not unexpected since MSI corresponds to a type of non-ignorable sample design, and it is well known that such designs leads to biased estimation in the logistic case. For example, the intercept term in a logistic model fit in a RBS situation is biased (Prentice & Pyke, 1979).

Chapter 5

Choosing Between Competing Models

In this chapter we describe the two main methods used as model search strategies in the proposed three-step modelling procedure set out in Chapter 3. These strategies enable us to choose between two competing models when their regressor sets are non-equivalent, and correspond to the likelihood-based and the prediction-based approaches reviewed in Chapter 2. In particular, we focus on the test proposed by Vuong (1989), which relies on the likelihood-based approach (Section 5.1), and the cross-validation technique, which relies on the prediction-based approach (Section 5.2).

5.1 The Likelihood-Based Approach

A general statistical test for choosing between two competing models based on the likelihood-based approach was proposed by Vuong (1989). The most important advantage of this test is the fact that it can be applied to model choice for all types of models (i.e. non-nested, nested and overlapping). Statistical hypothesis testing and the likelihood ratio statistic are a fundamental part of this test.

In this section, we describe the test statistic proposed by Vuong (1989). Next, specific statistical tests for both the unweighted and weighted inference paradigms

are provided for linear and logistic regression. Throughout, we employ the same notation as in Vuong (1989).

5.1.1 The Vuong Test Statistic

We begin with a general form of two competing models:

$$F_\theta = \{f(Y|X;\theta); \theta \in \Theta\}$$

and

$$G_\gamma = \{g(Y|X;\gamma); \gamma \in \Gamma\},$$

where Y is a target response variable, X is a matrix of independent variables, θ and γ are the true parameters of the corresponding models F_θ and G_γ.

As mentioned before, there are three general ways in which we can characterise these two competing models. They can be non-nested, nested, and overlapping as defined in Subsection 2.5.1. The first situation (i.e. non-nested models) is not of interest as far as this thesis is concerned because the two competing models have the same distributional assumptions (i.e. either normal or binomial distribution) and functional forms (i.e. either linear or logistic). We therefore confine our attention to the two latter cases (i.e. nested and overlapping models).

Since consideration of model choice here relates to only the case of non-equivalent regressor sets, the two competing models are treated as two different models in terms of independent variables. That is, the unweighted model specification (i.e. the model specified by ignoring sample weights) is defined as the model F_θ, and the weighted model specification (i.e. the model specified by using sample weights) is defined as the model G_γ.

In order to decide which of two competing models is better, Vuong's test relies on the Kullback-Leibler information criteria (Kullback-Leibler Information Criteria (KLIC)):

$$KLIC \equiv E^0[log\ p^0(Y|X)] - E^0[logf(Y|X;\theta_*)], \tag{5.1}$$

where $p^0(\cdot|\cdot)$ is the true conditional density of Y given X (that is, the unknown true model), E^0 denotes the expectation with respect to the true distribution, and $\theta_* = \arg\min_{\theta\in\Theta} E^0 log f(Y|X;\theta)$ is called the *pseudo-true value* of θ. Similarly KLIC and γ_* are defined for the G_γ. The better model is the model that minimizes Equation (5.1). As a result, the model that maximizes $E^0[log\ f(Y|X;\theta_*)]$ or $E^0[log\ g(Y|X;\gamma_*)]$ is chosen as the better model.

To create hypotheses appropriate for testing model choice, Vuong (1989) proposed to choose between the two competing models F_θ and G_γ as follows: The null hypothesis of the test expressed by

$$H_0 : E^0\left[log\frac{f(Y|X;\theta_*)}{g(Y|X;\gamma_*)}\right] = 0$$

meaning that the two rival models F_θ and G_γ are equivalent. The alternative hypotheses of the test given by

$$H_f : E^0\left[log\frac{f(Y|X;\theta_*)}{g(Y|X;\gamma_*)}\right] > 0$$

meaning that F_θ is better than G_γ

and

$$H_g : E^0\left[log\frac{f(Y|X;\theta_*)}{g(Y|X;\gamma_*)}\right] < 0$$

meaning that G_γ is better than F_θ.

In order to create a general statistic for choosing between two competing models, Vuong (1989) uses the likelihood ratio statistic as follows. Suppose that log-likelihood functions defined by the two competing models are:

$$l(\theta) \equiv \sum_{i=1}^{n} log\ f(y_i|x_i;\theta)$$

for the model F_θ, and

$$l(\gamma) \equiv \sum_{i=1}^{n} log\ g(y_i|x_i;\gamma)$$

for the model G_γ.

Given a sample of size n, let $\hat{\theta}$ and $\hat{\gamma}$ denote the corresponding maximum likelihood estimators (Maximum Likelihood Estimator (MLE)) of θ_* and γ_* respectively. Then $\hat{\theta}$ can be estimated by solving the score equation

$$l'(\theta) = 0,$$

Similarly, $\hat{\gamma}$ can be estimated by solving

$$l'(\gamma) = 0.$$

Therefore, the likelihood ratio statistic is

$$\begin{aligned} l(\hat{\theta}, \hat{\gamma}) &\equiv l(\hat{\theta}) - l(\hat{\gamma}) \\ &= \sum_{i=1}^{n} log\left[\frac{f(y_i|x_i;\hat{\theta})}{g(y_i|x_i;\hat{\gamma})}\right] \end{aligned} \tag{5.2}$$

where $l(\hat{\theta}) = sup_{\theta\in\Theta} l(\theta)$, and $l(\hat{\gamma}) = sup_{\theta\in\Theta} l(\gamma)$, and the variance of the likelihood ratio statistic can be directly calculated as

$$\hat{v}_*^2 = E^0[l(\hat{\theta}, \hat{\gamma})]^2 - [E^0[l(\hat{\theta}, \hat{\gamma})]]^2 \tag{5.3}$$

Vuong (1989) then notes that the following conditions are required for the likelihood ratio statistic to follow a normal distribution:

Assumption 5.1 *Given two competing models F_θ and G_γ, we assume that the following conditions hold:*

1. $f(\cdot|\cdot;\theta_*) \neq g(\cdot|\cdot;\gamma_*)$
2. $\frac{1}{n} l(\hat{\theta}, \hat{\gamma}) \xrightarrow{a.s.} E^0\left[log\frac{f(y_i|x_i;\theta_*)}{g(y_i|x_i;\gamma_*)}\right]$

As a consequence, Vuong (1989) obtained the following theorem.

Theorem 5.1 *Given Assumption 5.1, then*

$$\sqrt{n}\left[\frac{1}{n}l(\hat{\theta},\hat{\gamma}) - E^0\left[log\frac{f(y_i|x_i;\theta_*)}{g(y_i|x_i;\gamma_*)}\right]\right] \xrightarrow{d} N(0, v_*^2),$$

where the variance of $log\frac{f(y_i|x_i;\theta_*)}{g(y_i|x_i;\gamma_*)}$ *denoted by* v_*^2 *can be computed via*

$$v_*^2 = E^0\left[log\frac{f(Y|X;\theta_*)}{g(Y|X;\gamma_*)}\right]^2 - \left[E^0\left[log\frac{f(Y|X;\theta_*)}{g(Y|X;\gamma_*)}\right]\right]^2. \tag{5.4}$$

Let $V_{NW} = \frac{1}{n}l(\hat{\theta},\hat{\gamma})$ be the (unweighted) Vuong statistic. From Theorem 5.1, we note that V_{NW} is then asymptotically normally distributed with mean: $E^0\left[log\frac{f(y_i|x_i;\theta_*)}{g(y_i|x_i;\gamma_*)}\right]$ and variance $\frac{v_*^2}{n}$ given by

$$\frac{1}{n}\left(E^0\left[log\frac{f(Y|X;\theta_*)}{g(Y|X;\gamma_*)}\right]^2 - \left[E^0\left[log\frac{f(Y|X;\theta_*)}{g(Y|X;\gamma_*)}\right]\right]^2\right). \tag{5.5}$$

This leads to the following theorem.

Theorem 5.2 *Given Assumption 5.1 and Theorem 5.1, then*

$$\begin{aligned}
under H_0 : \frac{\sqrt{n}V_{NW}}{\hat{v}_*} &\xrightarrow{d} N(0,1) \\
under H_f : \frac{\sqrt{n}V_{NW}}{\hat{v}_*} &\xrightarrow{a.s.} +\infty \\
under H_g : \frac{\sqrt{n}V_{NW}}{\hat{v}_*} &\xrightarrow{a.s.} -\infty.
\end{aligned}$$

See Vuong (1989) for the proofs of Theorems 5.1-5.2.

Note that the Vuong statistic, which is asymptotically normally distributed as described in Theorem 5.1, is used throughout this thesis for both nested and overlapping models for practical simplicity. It is true that for the nested case using an asymptotic normal distribution for this statistic is not always as effective as using a chi-squared distribution. However, 'not as effective' does not mean 'not effective'. Both distributions are related, with the square of a standard normal variable having chi-squared distribution with one degree of freedom (Brown & Hollander, 1977). Given the conditions specified in Assumption 5.1 above, the statistic employed for

both nested and overlaping models will therefore follow Theorem 5.1 and Theorem 5.2.

Given Assumption 5.1 and Theorems 5.1-5.2 hold, a critical value is required in order to decide which model should be adopted. Throughout this thesis this critical value will be $z_{0.05}(1.96)$. That is, suppose that c_{cal} denotes the value of the statistic $\frac{\sqrt{n}V_{NW}}{\hat{v}_*}$ given a sample of size n. A final model can then be obtained via the following decision process:

(a) If $c_{cal} > z_{0.05}$, the unweighted model specification is adopted;

(b) If $c_{cal} < -z_{0.05}$, the weighted model specification is adopted;

(c) If $|c_{cal}| \leq z_{0.05}$, then there are three options;

  (c.1) the smaller of the two model specifications is adopted (i.e. the model with fewer covariates),

  (c.2) the unweighted model specification is adopted,

  (c.3) the weighted specification is adopted.

Note that c.1-c.3 are options for model choice when the test is not significant. Deciding which of these to adopt is essentially the responsibility of the researcher.

5.1.2 The Unweighted Vuong Statistic (V_{NW})

In the previous subsection the general form of the Vuong statistic was developed, as set out in Theorem 5.1. In this subsection, we provide expressions for the Vuong statistic that are applicable under linear and logistic regression.

5.1.2.1 The Unweighted Vuong Test Statistic for Linear Regression

Suppose that the target response variable is normally distributed with mean zero and a constant variance σ^2. Let $\theta = (\beta_f^T, \sigma_f^2)$ denotes the parameters of the model F_θ, and $\gamma = (\beta_g^T, \sigma_g^2)$ denote the parameters of the model G_γ; where $\beta_{(\cdot)}$ and $\sigma^2_{(\cdot)}$

represent in turn the regression coefficients and the constant variance under the corresponding model. The log-likelihood functions of the two rival models are then

$$l(\theta) = -\frac{n}{2}log(2\pi\sigma_f^2) - \frac{1}{2\sigma_f^2}(Y - X\beta_f)^T(Y - X\beta_f)$$

and

$$l(\gamma) = -\frac{n}{2}log(2\pi\sigma_g^2) - \frac{1}{2\sigma_g^2}(Y - X\beta_g)^T(Y - X\beta_g).$$

The parameters θ and γ can be estimated by solving the corresponding equations $l'(\theta) = 0$ and $l'(\gamma) = 0$. The unweighted Vuong test statistic for the linear regression case is then obtained by replacing $l(\hat{\theta})$ and $l(\hat{\gamma})$ into Equation (5.2) and divided by n. This leads to

$$V_{NW_{linear}} = \frac{1}{2}log(\frac{\hat{\sigma}_g^2}{\hat{\sigma}_f^2}) \tag{5.6}$$

where $V_{NW_{linear}}$ denotes the unweighted Vuong statistic for linear regression, and $\hat{\sigma}_f^2$ and $\hat{\sigma}_g^2$ denote in turn the estimated variance under the models F_θ and G_γ. The variance of the statistic $V_{NW_{linear}}$ can be computed as $\frac{\hat{v}_*^2}{n}$ where $\hat{v}_*^2$ is defined by Equation (5.3).

5.1.2.2 The Unweighted Vuong Test Statistic for Logistic Regression

In case of a logistic regression model, the target response variable is binary valued, with a 'success' probability that is linear on the logistic scale, i.e. the logit of this probability is a linear combination of the model covariates. Let β_f be the vector of regression coefficients under the model F_θ, and β_g be the vector of regression coefficients under the model G_γ. The log-likelihood function for the model F_θ is then

$$l(\beta_f) = \sum_{i=1}^{n}\left[y_i\sum_{j=0}^{k}x_{ij}\beta_{f_j} - log(1 + exp(\sum_{j=0}^{k}x_{ij}\beta_{f_j}))\right].$$

The regression coefficients β_f can be estimated by solving

$$l'(\beta_f) = 0.$$

Similarly, the log-likelihood function for the model G_γ is

$$l(\beta_g) = \sum_{i=1}^{n} \left[y_i \sum_{m=0}^{q} x_{im}\beta_{g_m} - log(1 + exp(\sum_{m=0}^{q} x_{im}\beta_{g_m}) \right].$$

The regression coefficients β_g can be estimated by solving

$$l'(\beta_g) = 0.$$

In doing so, suppose that we then obtain estimates denoted by $\hat{\beta}_f$ and $\hat{\beta}_g$ for the corresponding models F_θ and G_γ, and also the values $l(\hat{\beta}_f)$ and $l(\hat{\beta}_g)$. We replace $l(\hat{\beta}_f)$ and $l(\hat{\beta}_g)$ in Equation (5.2) by these values, leading to the unweighted Vuong statistic for model choice in the case of logistic regression. This is,

$$V_{NW_{logistic}} = \frac{1}{n} \sum_{i=1}^{n} \left[y_i \left(\sum_{j=0}^{k} x_{ij}\hat{\beta}_{f_j} - \sum_{m=0}^{q} x_{im}\hat{\beta}_{g_m} \right) - log(\frac{1 + exp(\sum_{j=0}^{k} x_{ij}\hat{\beta}_{f_j})}{1 + exp(\sum_{m=0}^{q} x_{im}\hat{\beta}_{g_m})} \right] \tag{5.7}$$

where $\hat{\beta}_f$ and $\hat{\beta}_g$ denote the estimated regression coefficients of the corresponding model F_θ and G_γ. Again, the variance of the statistic $V_{NW_{logistic}}$ can be computed by $\frac{\hat{v}_*^2}{n}$ where $\hat{v}_*^2$ is defined by Equation (5.3).

5.1.3 The Weighted Vuong Statistic (V_W)

Developing the weighted version of the Vuong test statistic is a straightforward extension of the development of the unweighted version of this statistic as set out in Section 5.1.1. Here, sample weights are employed throughout the process.

Given the same two competing models F_θ and G_γ, and the same hypotheses detailed in Section 5.1.1, we use a pseudo-likelihood (weighted likelihood) function

to define the weighted Vuong statistic in what follows. In particular, suppose that

$$\tilde{l}(\theta) \equiv \sum_{i=1}^{n} w_i log\ f(y_i|x_i;\theta),$$

is the pseudo-log-likelihood function defined by the model F_θ, and

$$\tilde{l}(\theta) \equiv \sum_{i=1}^{n} w_i log\ g(y_i|x_i;\gamma),$$

is the pseudo-log-likelihood function defined by the model G_γ. Once again, the parameters θ and γ can be estimated through solution of the equations obtained by setting the first derivatives of these pseudo-log-likelihood functions to zero. Assume that $\hat{\theta}$ and $\hat{\gamma}$ denote the corresponding pseudo-maximum likelihood estimators of θ_* and γ_*, obtained by solving

$$\tilde{l}'(\theta) = 0,$$

and

$$\tilde{l}'(\gamma) = 0.$$

The maximum pseudo-likelihood estimates for both models F_θ and G_γ are then defined as follows: $\tilde{l}(\hat{\theta}) = sup_{\theta\in\Theta}\tilde{l}(\theta)$, and $\tilde{l}(\hat{\gamma}) = sup_{\theta\in\Theta}\tilde{l}(\gamma)$, and we again use Equation (5.2) to obtain the weighted Vuong test statistic. This is

$$\begin{aligned} \tilde{l}(\hat{\theta},\hat{\gamma}) &\equiv \tilde{l}(\hat{\theta}) - \tilde{l}(\hat{\gamma}) \\ &= \sum_{i=1}^{n} w_i\ log\left[\frac{f(y_i|x_i;\hat{\theta})}{g(y_i|x_i;\hat{\gamma})}\right], \end{aligned} \tag{5.8}$$

with asymptotic variance:

$$\hat{v}^2 = E^0\left[\tilde{l}(\hat{\theta},\hat{\gamma})\right]^2 - \left[E^0\left[\tilde{l}(\hat{\theta},\hat{\gamma})\right]\right]^2. \tag{5.9}$$

In order for asymptotic properties to hold, we need to convert the expression (5.8) above to a weighted mean. That is, in what follows we write $V_W = \frac{1}{\sum_{i=1}^{n} w_i}\tilde{l}(\hat{\theta},\hat{\gamma})$ as the weighted Vuong statistic. We also note that the following additional assumption

is required to hold for this weighted Vuong statistic to have an asymptotic normal distribution.

Assumption 5.2 *Assuming Assumption 5.1 holds, and given two rival models F_θ and G_γ, we assume that the weighted Vuong test statistic V_W satisfies:*

$$V_W \xrightarrow{a.s.} E^0\left[log\frac{f(y_i|x_i;\theta_*)}{g(y_i|x_i;\gamma_*)}\right],$$

where $V_W = \frac{1}{\sum_{i=1}^n w_i}\tilde{l}(\hat{\theta},\hat{\gamma})$.

Given Assumption 5.2, we then have the following theorem.

Theorem 5.3 *Given Assumption 5.2, then*

$$\sqrt{n}\left[V_W - E^0\left[log\frac{f(y_i|x_i;\theta_*)}{g(y_i|x_i;\gamma_*)}\right]\right] \xrightarrow{d} N(0,\omega^2).$$

That is, the weighted Vuong test statistic $V_W = \frac{1}{\sum_{i=1}^n w_i}\tilde{l}(\hat{\theta},\hat{\gamma})$ *is asymptotically normally distributed with mean:* $E^0\left[log\frac{f(y_i|x_i;\theta_*)}{g(y_i|x_i;\gamma_*)}\right]$ *and variance*

$$\frac{\omega^2}{n} = \frac{\sum_s w_i^2}{(\sum_s w_i)^2}v_*^2, \tag{5.10}$$

where v_*^2 *is the variance of* $log\frac{f(y_i|x_i;\theta_*)}{g(y_i|x_i;\gamma_*)}$ *as defined in Equation (5.4).*

Proof of Theorem 5.3 is in Appendix A.

Given Theorem 5.3, it is then straightforward to prove the following theorem.

Theorem 5.4 *Given Assumption 5.2 and Theorem 5.3, then*

$$\begin{aligned} under H_0 : \frac{\sqrt{n}V_W}{\hat{\omega}} &\xrightarrow{d} N(0,1) \\ under H_f : \frac{\sqrt{n}V_W}{\hat{\omega}} &\xrightarrow{a.s.} +\infty \\ under H_g : \frac{\sqrt{n}V_W}{\hat{\omega}} &\xrightarrow{a.s.} -\infty. \end{aligned}$$

As a consequence, the same criteria as described in Section 5.1.1 can be used to

make a decision regarding which model should be adopted at the specified significance level of the test.

5.1.3.1 The Weighted Vuong Test Statistic for Linear Regression

The following development of the weighted Vuong test statistic for linear regression is a straightforward generalisation of earlier results.

Once again, suppose that the target response variable is normally distributed with mean zero and a constant variance σ^2, $\theta = (\beta_f', \sigma_f^2)$ denotes the parameters of the model F_θ, and $\gamma = (\beta_g', \sigma_g^2)$ denotes the parameters of the model G_γ; where $\beta_{(\cdot)}$ and $\sigma^2_{(\cdot)}$ represent regression coefficients and a variance parameter respectively under the corresponding model. The pseudo-log-likelihood functions of the two rival models are

$$\tilde{l}(\theta) = -\frac{\sum_{i=1}^{n} w_i}{2} log(2\pi\sigma_f^2) - \frac{1}{2\sigma_f^2}(Y - X\beta_f)^T W (Y - X\beta_f)$$

and

$$\tilde{l}(\gamma) = -\frac{\sum_{i=1}^{n} w_i}{2} log(2\pi\sigma_g^2) - \frac{1}{2\sigma_g^2}(Y - X\beta_g)^T W (Y - X\beta_g),$$

where W is a diagonal matrix of sample weights. The parameters θ and γ can be estimated by solving the equations $\tilde{l}'(\theta) = 0$ and $\tilde{l}'(\gamma) = 0$, respectively. The weighted Vuong test statistic for linear regression is then obtained by substituting $\tilde{l}(\hat{\theta})$ and $\tilde{l}(\hat{\gamma})$ into Equation (5.8). This leads to

$$V_{W_{linear}} = \frac{1}{2} log(\frac{\hat{\tilde{\sigma}}_g^2}{\hat{\tilde{\sigma}}_f^2}) \tag{5.11}$$

where $V_{W_{linear}}$ denotes the weighted Vuong test statistic for linear regression, and $\hat{\tilde{\sigma}}_f^2$ and $\hat{\tilde{\sigma}}_g^2$ denote the weighted estimates of the variance parameters of the corresponding models F_θ and G_γ. The variance of $V_{W_{linear}}$ can be computed via Equation (5.10), with the estimated variance $\hat{v}_*^2$ following from Equation (5.9).

5.1.3.2 The Weighted Vuong Test Statistic for Logistic Regression

For logistic regression, we again suppose that the target response variable is binary valued with a 'success' probability that is linear on a logistic scale and characterised by a vector of regression coefficients. Let β_f be a vector of regression coefficients under the model F_θ, and let β_g be the vector of regression coefficients under the model G_γ. The pseudo log-likelihood function of the model F_θ is given by

$$\tilde{l}(\beta_f) = \sum_{i=1}^{n} w_i \left[y_i \sum_{j=0}^{k} x_{ij}\beta_{f_j} - log(1 + exp(\sum_{j=0}^{k} x_{ij}\beta_{f_j})) \right]$$

and the regression coefficients β_f can therefore be estimated by solving

$$\tilde{l}'(\beta_f) = 0.$$

Similarly, the pseudo log-likelihood function of the model G_γ is given by

$$\tilde{l}(\beta_g) = \sum_{i=1}^{n} w_i \left[y_i \sum_{m=0}^{q} x_{im}\beta_{g_m} - log(1 + exp(\sum_{m=0}^{q} x_{im}\beta_{g_m}) \right]$$

and the regression coefficients β_g can be estimated by solving

$$\tilde{l}'(\beta_g) = 0.$$

Suppose that these estimates are $\tilde{l}(\hat{\beta}_f)$ and $\tilde{l}(\hat{\beta}_g)$. By replacing $\tilde{l}(\hat{\beta}_f)$ and $\tilde{l}(\hat{\beta}_g)$ into Equation (5.8), the weighted Vuong test statistic for logistic regression is then:

$$V_{W_{logistic}} = \frac{1}{\sum_{i=1}^{n} w_i} \sum_{i=1}^{n} w_i \left[y_i \left(\sum_{j=0}^{k} x_{ij}\hat{\beta}_{f_j} - \sum_{m=0}^{q} x_{im}\hat{\beta}_{g_m} \right) - log(\frac{1 + exp(\sum_{j=0}^{k} x_{ij}\hat{\beta}_{f_j})}{1 + exp(\sum_{m=0}^{q} x_{im}\hat{\beta}_{g_m})} \right] \tag{5.12}$$

where $\hat{\beta}_f$ and $\hat{\beta}_g$ denote the estimated regression coefficients under the corresponding models. Again, the variance of $V_{W_{logistic}}$ can be computed using Equation (5.10), with the estimated variance $\hat{v}_*^2$ derived via Equation (5.9).

5.2 Prediction-Based Approach

This section describes prediction-based cross-validation (CV) methodology, with reference to the CV procedure as well as the CV criteria employed in this thesis.

5.2.1 The Cross-Validation Procedure

The CV procedure was introduced in Chapter 2, Here we note that five-fold CV will be used throughout this thesis in order to choose a final model from two competing models (i.e. the unweighted and weighted model specifications). Five-fold CV proceeds via the following three main steps: data splitting, model fitting and model evaluation.

1. **Data Splitting:** The sample is randomly separated into five approximately equal-sized subsamples. One of these subsamples is called the testing set, and the other subsamples are referred to as the training set.

2. **Model Fitting:** For each model specification, the training set is used to fit two models. One is fitted by ignoring sample weights, and the other is fitted using the sample weights.

3. **Model Evaluation:** Given the two fitted models obtained from Step 2, their predictive efficiency is assessed via a criterion based on how well they predict the values in the testing set. We describe the predictive efficiency criteria used for this evaluation in the next subsection.

The procedure is complete when all five subsamples have been used as the testing set.

5.2.1.1 The CV Procedure for Linear Regression

In the following, the CV procedure is applied to the case of linear regression. We focus on steps 2-3 of the CV procedure as described in the previous subsection, and provide details for one iteration of the five-fold CV. We use the subscript j to

denote that the jth subsample is treated as a testing set, and the subscript $(-j)$ to denote the training set that omits the jth subsample. To illustrate, assume a linear regression model of the form

$$Y = X\beta + \varepsilon$$

where Y is a vector of values for a continuous dependent variable, X is a matrix of values for a set of independent variables, β is a vector of unknown parameters, and ε is a vector of errors.

In Step 2 of the three-step CV procedures, we have a training set including $Y_{(-j)}, X_{(-j)}$ and $W_{(-j)}$ corresponding to a vector of dependent variable values, a matrix of values for the independent variables and a diagonal matrix of the sample weights. The unweighted estimator $\hat{\beta}_{(-j)}$ can be obtained by

$$\hat{\beta}_{(-j)} = (X_{(-j)}^T X_{(-j)})^{-1} X_{(-j)}^T Y_{(-j)}.$$

Similarly, the weighted estimator $\hat{\beta}_{*(-j)}$ can be obtained by

$$\hat{\beta}_{*(-j)} = (X_{(-j)}^T W_{(-j)} X_{(-j)})^{-1} X_{(-j)}^T W_{(-j)} Y_{(-j)}.$$

In Step 3 of the three-step CV procedures, both the unweighted and weighted estimators obtained from Step 2 are then used to predict the target dependent variable Y_j in the jth subsample, i.e. the jth testing set. Next, two versions of the resulting mean squared prediction error (Mean Squared Prediction Error (MSPE)) are used as the criteria for evaluating the predictive efficiency of the two fitted models. These are the (unweighted) MSPE ignoring sample weights, and the weighted MSPE using the sample weights. Details of both the unweighted and weighted MSPE are in Subsection 5.2.2.1.

5.2.1.2 The CV Procedure for Logistic Regression

Here we provide details for the application of the three-step CV procedures to the case of logistic regression. Again, we focus on Steps 2-3 of the three-step CV proce-

dure, and for one iteration of the five-fold CV procedure. We use the same subscript notation as previously defined for the linear case to identify both the training and testing sets. To start, suppose that a logistic regression model given by

$$logit\ Pr(Y = 1|X) = X\beta, \tag{5.13}$$

where Y is a vector of values of a binary response variable, X is a matrix of values for a specified set of independent variables, β is a vector of unknown parameters, and $Pr(Y = 1|X)$ is the conditional probability of the 'success' outcome for Y.

Step 2 of the three-step CV procedure assumes a training set that includes $Y_{(-j)}, X_{(-j)}$ and $W_{(-j)}$ as defined before for the linear case. Since the unknown parameters β set out in Equation (5.13) can be estimated via maximum likelihood (Hosmer & Lemeshow, 1989) using these data, we can calculate estimates of the parameters $\beta_{(-j)}$ for the training set. Suppose that the subscript i represents the ith element in the training set which is of size $n_{(-j)}$. The unweighted estimators $\beta_{(-j)}$ are then obtained by solving

$$\sum_{i=1}^{n_{(-j)}} \left\{ y_{i(-j)} log \left[\frac{exp(x_{i(-j)}\beta_{(-j)})}{1 + exp(x_{i(-j)}\beta_{(-j)})} \right] + (1 - y_{i(-j)}) log \left[\frac{1}{1 + exp(x_{i(-j)}\beta_{(-j)})} \right] \right\} = 0$$

(Lohr, 1999).

Similarly, the weighted estimators $\tilde{\beta}_{(-j)}$ are obtained by solving

$$\sum_{i=1}^{n_{(-j)}} w_{(-j)} \left\{ y_{i(-j)} log \left[\frac{exp(x_{i(-j)}\tilde{\beta}_{(-j)})}{1 + exp(x_{i(-j)}\tilde{\beta}_{(-j)})} \right] + (1 - y_{i(-j)}) log \left[\frac{1}{1 + exp(x_{i(-j)}\tilde{\beta}_{(-j)})} \right] \right\} = 0$$

(Binder, 1983; Chambless & Boyle, 1985).

In Step 3 of the three-step procedures, both the unweighted and weighted estimators obtained from Step 2 are used to predict the values of the target response variable Y_j in the jth subsample or the testing set. The predictive efficiency of the two fitted models can be assessed through the values of the prediction error (Prediction Error (PE)) then generated. Two versions of the PE are computed. These

correspond to the (unweighted) PE ignoring sample weights, and the weighted PE that uses sample weights. Details of both the unweighted and weighted PE are in Subsection 5.2.2.2.

5.2.2 Cross-Validation Criteria

In this subsection, we provide the formula for the two criteria (MSPE and PE) applicable under the two paradigms (unweighted and weighted) used to evaluate the predictive efficiency of each subsample in the three-step CV procedure.

5.2.2.1 Cross-Validation Criteria for Linear Regression

In the linear case, the unweighted mean square prediction error for the jth testing set denoted by $MSPE_j$ is defined as

$$MSPE_j = \frac{1}{n_j} \sum_{i=1}^{n_j} \left(y_{ij} - \hat{y}_{i(-j)} \right)^2$$

where n_j denotes the sample size of the jth testing set, y_{ij} denotes the ith observation of the dependent variable in the jth testing set; here y_{ij} is treated as the true value to be predicted, and $\hat{y}_{i(-j)}$ is the predicted value of the ith observation of the dependent variable in the testing set that omits the jth subsample.

Similarly, the weighted mean square prediction error for the jth testing set denoted by $MSPE_{*_j}$ is defined as

$$MSPE_{*j} = \frac{1}{\sum_{i=1}^{n_j} w_{ij}} \sum_{i=1}^{n_j} w_{ij} \left(y_{ij} - \hat{y}_{i(-j)} \right)^2$$

where w_{ij} is the sample weight of the ith element in the jth testing set, and the other quantities are defined as in $MSPE_j$.

5.2.2.2 Cross-Validation Criteria for Logistic Regression

In the logistic case, the unweighted prediction error for the jth testing set denoted by PE_j is defined as

$$PE_j = \frac{n_{\Delta j}}{n_j}$$

where $n_{\Delta j}$ denotes the number of incorrectly predicted cases in the jth testing set, (i.e. the observed value and the predicted value differ), and n_j denotes the sample size of the jth testing set. Note that the predicted value for a case is one when its estimated probability is at least one-half. Otherwise, the predicted value is zero. Similarly, the weighted prediction error for the jth testing set denoted by PE_{*j} is defined as

$$PE_{*j} = \frac{\sum_{i=1}^{n_j} w_{\Delta ij}}{\sum_{i=1}^{n_j} w_{ij}}$$

where $w_{\Delta ij}$ denotes the sample weight of the ith incorrectly predicted case in the jth testing set, and w_{ij} denotes the sample weight of the ith element in the jth testing set.

5.2.2.3 Cross-Validation Criteria for A Final Model

The final step in the five-fold CV process is to combine the five separate values of each criterion defined above using their average value. That is, the decision on which model to adopt depends on which model generates the smallest such average value.

5.3 Conclusion

This chapter describes the two main methods that can be used as model search strategies for the case where the two competing models are non-equivalent in term of their regressor sets. The first method uses the Vuong test statistic and is based on a likelihood approach, while the second uses the cross-validation technique which is essentially a prediction-based approach. Criteria for choosing a final model are also

provided for both approaches. Note that the methods for model search described here are theoretical and have been formulated so that they can be used as model search strategies within the proposed modelling procedure described in Chapter 3. Their practicality, in terms of being useful for model choice is assessed via the simulation study that we describe in the next chapter.

Chapter 6

Simulation Study and Application

In previous chapters we developed a three-step modelling procedure for weighted survey data that included a model search strategy based on backward elimination with and without use of survey weights and alternative decision processes for choosing between competing models when this search identified distinct models for the survey population. In this chapter we provide simulation results that illustrate the comparative performances of these decision processes based on an application to a realistic survey data set where non-identical models (or equivalently, regressor sets) are identified. We also include its application to this survey data set in the last section. As a consequence in this chapter we illustrate application of our general approach via both simulation and application based on the same realistic data set.

6.1 Data

Our motivating application is based on one designed (among other things) to help a government make decisions about allocation of resources for national health care services. The data used for these simulations are based on a subset of the variables contained in the 4617 household records from a single State in a round of the Indian National Family Health Survey (INFHS), which is a large annual survey that collects data relevant to the health status of individuals in Indian households. Six point summaries for the variables contained in these INFHS data are set out in Table 6.1,

with Table 6.2 showing the definitions of the variables in Table 6.1. An important objective of the survey is to provide information to assist the government of India allocate resources for provision of health care services in the country. In order to achieve the purpose, population modelling of the survey data is necessary in order to identify important factors underpinning the health status of Indian households. In particular, both linear and logistic regression models are used for this purpose. In this chapter we use the variables *region*, *hhtype*, *headsex*, *headage*, *headocc* and *headed* in a simulation study aimed at evaluating the three-step modelling procedures developed in the previous chapters. We then investigate how these procedures can be used to actually model the regression relationships of interest in these data. The variables that we focus on are for illustrative purposes only, but have been chosen to relate to national health care services provision. The dependent variable for our linear regression analysis is *density* defined as *hhsize* divided by *hhrooms*, reflecting quality of housing, while for our logistic regression analysis we use the binary response variable *pipe* as the dependent variable, reflecting access to sanitation services. In particular, we simulated values of these two variables based on a known relationship to the other variables used in the simulation (i.e. *region*, *hhtype*, *headsex*, *headage*, *headocc* and *headed*), creating an artificial INFHS sample with fixed size $N = 4617$. These data were then modelled using the three-step procedure, and the performance of the parameter estimates thus obtained were analysed. In the last section of the chapter we then use the actual INFHS data to model the regression of *density* and *pipe* on *region*, *hhtype*, *headsex*, *headage*, *headocc* and *headed*.

In the simulation process, the variable *headage*, labeled by Z, is used as a stratifying variable. Similarly, the variables *region*, *hhtype*, *headsex*, *headocc* and *headed* are used as independent variables, labeled by $\mathbf{X}$. Note that although *age* and *sex* are common stratifying variables (Lumley, 2010), only *headage* was used for stratification here because joint stratification on *headage* and *headsex* led to strata with very small numbers of cases under non-proportional allocation.

Variable	Summary Statistics					
	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
hhstate	3.000	7.000	9.000	10.650	12.000	24.000
weight	1389	5023	13290	22190	26560	437800
region	1.000	1.000	2.000	2.004	3.000	3.000
hhtype	1.000	1.000	2.000	1.525	2.000	2.000
headsex	1.000	1.000	1.000	1.100	1.000	2.000
headage	1.000	4.000	6.000	4.800	6.000	6.000
headmar	1.000	1.000	1.000	1.316	1.000	5.000
headocc	1.000	2.000	3.000	2.817	3.000	4.000
headed	1.000	1.000	2.000	2.111	3.000	4.000
religion	1.000	1.000	1.000	1.232	1.000	3.000
hhsize	1.000	3.000	5.000	5.665	7.000	20.000
hhrooms	1.000	2.000	3.000	3.477	4.000	33.000
toilet	1.000	2.000	3.000	2.594	3.000	4.000
density	0.121	1.000	1.667	1.970	2.500	12.000
pipe	0.000	0.000	1.000	0.730	1.000	1.000

Table 6.1: Summary statistics in the INFHS data

6.2 Evaluation Criteria for the Final Model

Three criteria: relative bias (Relative Bias (RB)), relative root mean squared error (Relative Root Mean Squared Error (RRMSE)), and relative variance (Relative Variance (RV)), were used to evaluate the final models achieved as the outcome of the three-step modelling procedure.

Let φ be a $p \times 1$ vector of regression coefficients associated with the target population model, and let $\hat{\varphi}$ be an estimator of φ. Assuming t iterations, the relative bias (RB) can be expressed in form

$$\frac{1}{|\varphi|}\left[\frac{1}{t}\sum_{j=1}^{t}(\hat{\varphi}_j - \varphi)\right],$$

the relative root mean squared error (RRMSE) can be expressed as

$$\frac{1}{|\varphi|}\left[\frac{1}{t}\sum_{j=1}^{t}(\hat{\varphi}_j - \varphi)^2\right]^{0.5},$$

and the relative variance (RV) can be expressed

$$\frac{1}{|\varphi|}\left[\frac{1}{t}\sum_{j=1}^{t}(\hat{\varphi}_j - \bar{\varphi})^2\right]^{0.5}.$$

hhstate (8)	State where the household is located
weight	Sample weight of household
region(3)	Region of household
hhtype (2)	Type of household (urban or rural)
headsex (2)	Sex of the head of household
headage(7)	Age of the head of household
headmar (5)	Marital status of the head of household
headocc(5)	Occupation of the head of household
headed (4)	Education level of the head of household
religion(4)	Religion of the head of household
hhsize	The number of people in the house
hhrooms	The number of rooms in house
toilet (4)	The number of toilets in the house
density	The density of the household, defined as *hhsize* divided by *hhrooms*
pipe(2)	Piped water supply status of the household
	(0 = no supply, 1 = supply)

Table 6.2: Definitions of variables set out in Table 6.1

Note that the figures in parentheses correspond to number of categories for the associated variable

Note that since relative variance does not include a bias term, it is a useful discriminating criterion in a situation where all estimators are biased.

6.3 Simulation of Linear Regression

For each simulation of a finite population of size $N = 4617$, let $\underset{\sim}{a}$ and $\underset{\sim}{b}$ denote vectors of regression coefficient parameters that correspond to the regression coefficients of the stratification variable $\mathbf{Z}$ and the independent variables $\mathbf{X}$, respectively. Values of a $N \times 5$ matrix of $\mathbf{Z}$ and a $N \times 11$ matrix of $\mathbf{X}$ were obtained from the INFHS data set in order to generate the corresponding values of Y via the model

$$Y = 2 + \mathbf{Z}\underset{\sim}{a} + \mathbf{X}\underset{\sim}{b} + \varepsilon \tag{6.1}$$

where $\underset{\sim}{a} = (4, 4, 4, 4, 3)^T$, $\underset{\sim}{b} = (1, -1, -2, 2, 1, 2, 3, 1.5, 1.5, 1)^T$, and ε denotes a vector of *iid* realisations from a N(0,3) distribution.

Finite population data were obtained by merging values of Y generated from equation (6.1) with their corresponding values of $\mathbf{Z}$ and $\mathbf{X}$. The values of the stratifying variable $\mathbf{Z}$ were then used to classify these finite population data into strata. These strata formed the basis of sample design in the NIS and MSI scenarios; whereas the sample design under the RBS scenario strata were defined using the values of the target response variable Y. Next, a sample was drawn from this finite population using simple stratified random sampling and standard sample expansion-type weights constructed as set out in Table 6.3. Finally, the proposed modelling procedure, including model search, and as set out in Figure 3.1, was used to identify a final model for the population.

The above sampling procedure was independently repeated 10000 times using the R code set out in Appendix B. Table 6.4 shows the classification of final models obtained under the NIS, MSI and RBS scenarios when these models had non-equivalent regressor sets. The first column in this table shows the strategies used when searching for a final model. Here $Vuong_{NW}$ is the strategy based on the unweighted Vuong

stratum (h)	N_h	n_h	w_h
Scenario 1: Non-Informative Sampling Scenario 2: Missing Stratification Information			
1	120	100	12.00
2	371	150	2.47
3	522	125	4.18
4	641	75	8.55
5	608	100	6.08
6	2355	50	47.10
Scenario 3: Response-Based Sampling			
1	1500	50	30.00
2	1300	100	13.00
3	1200	200	6.00
4	617	250	2.47

Table 6.3: Stratum sample allocations and corresponding expansion-type sample weights used in simulation of linear regression modelling under the three scenarios.

Strategy	Final Model Choices		
	Unweighted Model Specification	Weighted Model Specification	Equivalence
Scenario 1: Non-Informative Sampling (NIS)			
$Vuong_{NW}$	67	608	8844
$Vuong_{W}$	9519	-	-
CV_{NW}	9499	20	-
CV_{W}	7141	2378	-
Voting System	7175	3	2341
Scenario 2: Missing Stratification Information (MSI)			
$Vuong_{NW}$	1	4057	5371
$Vuong_{W}$	9429	-	-
CV_{NW}	9389	40	-
CV_{W}	5663	3766	-
Voting System	5703	17	3709
Scenario 3: Response-Based Sampling (RBS)			
$Vuong_{NW}$	1	4434	364
$Vuong_{W}$	81	1785	2933
CV_{NW}	4337	462	-
CV_{W}	217	4582	-
Voting System	331	2687	1781

Table 6.4: Simulation results for final model choices using four model search strategies for the case of linear regression.

test statistic, $Vuong_W$ stands for the strategy that used the weighted Vuong test statistic, CV_{NW} is the strategy corresponding to unweighted cross-validation, CV_W is the weighted cross-validation strategy, and *Voting System* denotes the strategy where the choice of a final model is based on the one most often selected among these four strategies (i.e. $Vuong_{NW}$, $Vuong_W$, CV_{NW} and CV_W). Recollect that we are only dealing with situations where two distinct models are identified by the forward selection procedure. Given this situation (non-equivalent regressor sets), the second column indicates the number of a situations where the model specified by ignoring sample weights (unweighted model specification) was chosen, whereas the third column indicates the number of a situations where the model specified by using sample weights (weighted model specification) was chosen. The fourth column indicates the number of situations where the two competing models (unweighted and weighted model specifications) were not significantly different following application of either Vuong-based strategy or were there was a tie between competing models following application of the voting system strategy.

Table 6.5-6.13 show the empirical values of relative biases, relative root mean squared errors and relative variances of regression coefficient estimators of the final models achieved by five model search strategies for all three scenarios as detailed in Table 6.4.

The numbers set out in Table 6.4 indicate that over 10000 simulations, there were 9519, 9429 and 4799 simulations that resulted in non-equivalent regressor sets under the NIS, MSI and RBS scenarios, respectively. For the NIS and MSI scenarios we see that, except for the $Vuong_{NW}$ strategy, all four of the other strategies strongly prefer the unweighted model specification to the weighted model specification. That is, under these scenarios sample weights are indicated as unnecessary, as we expect from the theoretical perspective. In contrast, under the RBS scenario we see all strategies except for CV_{NW} strongly prefer the weighted model specification to the unweighted model specification. That is, in this case (RBS) use of sample weights are indicated as essential for modelling purposes.

The last column of Table 6.4 shows the number of simulations where the two competing model specifications (weighted and unweighted) were seen to be equivalent after using either the Vuong-based or voting strategies. In these situations, we chose the smaller of the two models (i.e. the one with fewer regressors) in the case of the The strategy based on the unweighted Vuong test statistic ($Vuong_{NW}$) and The strategy that used the weighted Vuong test statistic ($Vuong_W$) strategies, while in the case of the *voting system* strategy we kept both competing models (the smaller and the larger). This eventually led to six strategies that we could then examine for efficiency as a model-choice procedure in terms of RB, RV and RRMSE in what follows. In particular, in what follows The unweighted Vuong strategy with a smaller model chosen when the unweighted and weighted model specifications are not significantly different ($V_{NW_{small}}$) denotes the unweighted Vuong stategy with a smaller model chosen when the unweighted and weighted model specifications are not significantly different, CV_{NW} denotes the unweighted CV strategy, The weighted Vuong strategy with a smaller model chosen when the unweighted and weighted model specifications are not significantly different ($V_{W_{small}}$) denotes the weighted Vuong strategy with a smaller model chosen when the unweighted and weighted model specifications are not significantly different, CV_W denotes the weighted CV strategy, The voting-system strategy with a smaller model chosen in case of a tie (VS_{small}) denotes the voting-system strategy with a smaller model chosen in the case of a tie, and The voting-system strategy with a larger model chosen in case of a tie (VS_{large}) denotes the voting-system strategy with a larger model chosen in the case of a tie. CV_{NW} and CV_W are the same as defined earlier.

In the following subsections, we show the empirical values of RB, RV and RRMSE of the regression coefficient estimates obtained using these six modelling strategies. These results are set out in a common table form - the first column gives the name of the regressor variable corresponding to the regression coefficient in the target population model. The second column shows the true value of this regression coefficient as specified by Equation (6.1). The third column shows the values of the relevant cri-

terion in the case where the model indicated by either an unweighted or a weighted selection process is fitted. Here Unweighted Model Specification (UMS) stands for 'unweighted model specification', and Weighted Model Specification (WMS) stands for 'weighted model specification'. The last set of columns show the values of the relevant criterion after using the six modelling strategies.

6.3.1 Simulation Results for Scenario 1: NIS

We recall that Non-Informative Sampling (NIS) is where the sample and the corresponding population have the same distribution for the target response variable. In this case we expect that the model specified by ignoring sample weights should be the one chosen, and that, on average, it should be a 'closer' model to the true target population model than the model chosen using the sample weights. This is because sample weights do not matter for the NIS scenario, as mentioned in Chapter 2.

From the results set out in Table 6.4, we see that there are 9519 simulations (from 10000) where non-equivalent regressor sets were identified under NIS. The five main strategies (i.e. $Vuong_{NW}$, $Vuong_{W}$, CV_{NW}, CV_{W} and *Voting System*) were then used to choose a final model using the sample data obtained in each simulation. The values set out in Table 6.5-6.7 show the relative biases, relative variances and relative root mean squared errors of regression estimators respectively generated by these final models, averaged over the 9519 cases, under the six strategies (i.e. $V_{NW_{small}}$, CV_{NW}, $V_{W_{small}}$, CV_{W}, VS_{small} and VS_{large}).

The Monte Carlo biases set out in the third column of Table 6.5 indicate that estimators obtained from the unweighted model specification are less biased than estimators obtained from the weighted model specification (UMS) in the case of direct choice, as expected. In the fourth column of this table, we can see that the final models obtained using the CV_{NW}, $V_{W_{small}}$ and VS_{large} strategies provide the lowest values of RB for almost all of the estimators. The biases generated by these three strategies are also almost identical. In contrast, the biases generated by $V_{NW_{small}}$ are generally larger, in absolute terms, than those generated by the other

strategies.

The average RV values set out in Table 6.6 indicate that UMS-based estimators generate values for average RV that are less than those generated by WMS-based estimators in all cases. Furthermore, the three strategies CV_{NW}, $V_{W_{small}}$ and VS_{large} are again the best performers and provide almost identical outcomes for this measure. This is also the case for the average RRMSE values displayed in Table 6.7.

We conclude that the three strategies i.e. CV_{NW}, $V_{W_{small}}$ and VS_{large} are better than the alternative strategies in choosing a final model under the NIS scenario, recording virtually identical performances in our simulations. In comparison, the $V_{NW_{small}}$ strategy is the worst among the different strategies that we compared. We conclude that the three strategies i.e. CV_{NW}, $V_{W_{small}}$ and VS_{large} represent reasonable tools for model choice tool under the NIS scenario.

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	2.0	-0.063	0.312	0.303	-0.062	-0.063	0.036	0.036	-0.063
headage2	4.0	0.000	-0.007	-0.007	0.000	0.000	-0.002	-0.002	0.000
headage3	4.0	0.000	-0.007	-0.006	0.000	0.000	-0.002	-0.002	0.000
headage4	4.0	-0.001	-0.004	-0.004	-0.001	-0.001	-0.001	-0.001	-0.001
headage5	4.0	0.002	0.000	0.000	0.002	0.002	0.002	0.002	0.002
headage6	3.0	-0.001	-0.014	-0.014	-0.001	-0.001	-0.004	-0.004	-0.001
headsex2	1.0	-0.240	-0.405	-0.414	-0.240	-0.240	-0.264	-0.269	-0.240
headocc2	-1.0	0.249	0.043	0.059	0.248	0.249	0.191	0.195	0.249
headocc3	-2.0	0.097	0.056	0.064	0.097	0.097	0.082	0.084	0.097
headocc4	2.0	0.132	0.049	0.060	0.131	0.132	0.104	0.107	0.132
headedu2	1.0	-0.102	-0.476	-0.479	-0.102	-0.102	-0.189	-0.191	-0.102
headedu3	2.0	-0.036	-0.211	-0.211	-0.036	-0.036	-0.077	-0.078	-0.036
headedu4	3.0	-0.009	-0.164	-0.162	-0.009	-0.009	-0.048	-0.048	-0.009
region2	1.5	-0.006	-0.206	-0.204	-0.006	-0.006	-0.065	-0.066	-0.006
region3	1.5	0.013	-0.150	-0.147	0.013	0.013	-0.039	-0.039	0.013
hhtype2	1.0	-0.017	-0.370	-0.368	-0.018	-0.017	-0.100	-0.101	-0.018

Table 6.5: Simulation results for relative biases of estimators of linear regression coefficients under NIS

[1]Unweighted Model Specification
[2]Weighted Model Specification

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	2.0	0.533	0.939	0.936	0.534	0.533	0.683	0.682	0.533
headage2	4.0	0.098	0.104	0.104	0.098	0.098	0.100	0.100	0.098
headage3	4.0	0.103	0.111	0.111	0.103	0.103	0.105	0.105	0.103
headage4	4.0	0.117	0.125	0.125	0.117	0.117	0.120	0.120	0.117
headage5	4.0	0.109	0.120	0.120	0.109	0.109	0.113	0.113	0.109
headage6	3.0	0.178	0.188	0.188	0.178	0.178	0.180	0.180	0.178
headsex2	1.0	0.807	1.049	1.037	0.807	0.807	0.888	0.884	0.807
headocc2	-1.0	0.944	1.388	1.381	0.945	0.944	1.077	1.076	0.944
headocc3	-2.0	0.478	0.821	0.818	0.478	0.478	0.585	0.585	0.478
headocc4	2.0	0.586	0.919	0.917	0.586	0.586	0.689	0.689	0.586
headedu2	1.0	0.464	0.762	0.755	0.464	0.464	0.580	0.579	0.464
headedu3	2.0	0.196	0.460	0.457	0.196	0.196	0.306	0.305	0.196
headedu4	3.0	0.206	0.461	0.458	0.206	0.206	0.309	0.308	0.206
region2	1.5	0.212	0.572	0.567	0.212	0.212	0.375	0.374	0.212
region3	1.5	0.204	0.540	0.537	0.205	0.204	0.356	0.355	0.205
hhtype2	1.0	0.349	0.709	0.706	0.350	0.349	0.494	0.494	0.349

Table 6.6: Simulation results for relative variances of estimators of linear regression coefficients under NIS

[1]Unweighted Model Specification
[2]Weighted Model Specification

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	2.0	0.537	0.990	0.984	0.538	0.537	0.684	0.683	0.537
headage2	4.0	0.098	0.105	0.104	0.098	0.098	0.100	0.100	0.098
headage3	4.0	0.103	0.111	0.111	0.103	0.103	0.105	0.105	0.103
headage4	4.0	0.117	0.125	0.125	0.117	0.117	0.120	0.120	0.117
headage5	4.0	0.109	0.120	0.120	0.109	0.109	0.113	0.113	0.109
headage6	3.0	0.178	0.189	0.188	0.178	0.178	0.180	0.180	0.178
headsex2	1.0	0.842	1.124	1.117	0.842	0.842	0.926	0.924	0.842
headocc2	-1.0	0.976	1.389	1.383	0.977	0.976	1.094	1.093	0.976
headocc3	-2.0	0.487	0.823	0.821	0.488	0.487	0.590	0.591	0.487
headocc4	2.0	0.600	0.921	0.919	0.601	0.600	0.697	0.697	0.600
headedu2	1.0	0.475	0.898	0.895	0.475	0.475	0.610	0.610	0.475
headedu3	2.0	0.199	0.506	0.503	0.199	0.199	0.315	0.315	0.199
headedu4	3.0	0.206	0.490	0.486	0.206	0.206	0.312	0.312	0.206
region2	1.5	0.212	0.608	0.603	0.212	0.212	0.381	0.379	0.212
region3	1.5	0.205	0.561	0.557	0.205	0.205	0.358	0.357	0.205
hhtype2	1.0	0.349	0.800	0.796	0.350	0.349	0.504	0.504	0.350

Table 6.7: Simulation results for relative root mean squared errors of estimators of linear regression coefficients under NIS

[1]Unweighted Model Specification
[2]Weighted Model Specification

6.3.2 Simulation Results for Scenario 2: MSI

We recall that the Missing Stratification Information (MSI) scenario corresponds to a situation where the sample and the corresponding population have a different regression function for the target response variable because of missing stratification variables. In this case, the stratifying variable used for classifying the target population data is *headage* , so all results related to this variable are absent in the following tables of results. As we have already shown in Chapter 4, both model specifications (UMS and WMS) are then biased because they in fact produce estimates of the regression coefficient parameters of another population model rather than that of the target population model. There we also found that sample weights cannot fix this situation, in the sense of recovering the target population model coefficients for the included variables. Here therefore we consider the situation where, given that any model is erroneous for estimation, should the model specified by ignoring sample weights be chosen rather than the model specified by using sample weights.

As results displayed in Table 6.4 indicate, there are 9429 cases where non-equivalent regressor sets were chosen out of the 10000 replicates under MSI. Again, we apply the five strategies (i.e. $Vuong_{NW}$, $Vuong_{W}$, CV_{NW}, CV_{W} and *Voting System*) to choose a final model for each simulation, and the resulting values of relative bias, relative variance and relative root mean squared error generated by the strategies $V_{NW_{small}}$, CV_{NW}, $V_{W_{small}}$, CV_{W}, VS_{small} and VS_{large} are set out in Tables 6.8-6.10.

The results displayed in Table 6.8 for the case of direct choice (all available variables in the model specification) show that the estimators of the regression coefficients associated with *headdocc*2, *headocc*3, *headedu*2, *headedu*3, *headedu*4 and *region*2 obtained under the UMS strategy typically show smaller RB than the corresponding estimators defined under the WMS strategy. Turning to the model choice strategies in the fourth column, it can be seen that the $V_{W_{small}}$ strategy provides the best result. In addition, the the CV_{NW} and VS_{large} strategies also perform well with respect to RB, with CV_{W} also performing reasonably. In contrast, the $V_{NW_{small}}$ and

the VS_{small} strategies do not perform well in terms of RB.

Some perspective on these results can be obtained if we now turn to Subsection 6.5.1 where we see an empirical verification of the results on modelling bias under MSI that were developed in Chapter 4 for the case of a very simple linear regression model. In particular, the results shown in Table 6.25 indicate that both model specifications (UMS and WMS) are biased when the target of inference is the model that includes the missing stratified variable. On the other hand, both model specifications are unbiased when the target of inference is the model that averages over the missing stratification variable. Since our focus here is bias relative to the true model (which includes the stratifying variable) we can see why the RB values displayed in Table 6.8 are comparatively high.

Returning to Tables 6.9-6.10 we next consider the simulation results for RV and RRMSE, respectively. The values of RV shown in Table 6.9 indicate that, when bias term is ignored, the $V_{W_{small}}$ strategy has the smallest values of RV among all strategies considered here. Again, we see that CV_{NW} and VS_{large} also perform well. Note however that $V_{NW_{small}}$, CV_W and VS_{small}) do not perform well, with high values of RV; in particular, the $V_{NW_{small}}$ strategy is the worst of these three strategies in terms of RV. Consequently, if we base our decision on RV (i.e. we downplay the importance of bias), then the three strategies (i.e. $V_{W_{small}}$, CV_{NW} and VS_{large}) emerge as reasonable model choice tools for the MSI scenario. We note furthermore that the results set out in Table 6.10 are generally consistent with those displayed in Table 6.9. We therefore conclude that the three strategies CV_{NW}, $V_{W_{small}}$ and VS_{large} should be used for model choice under MSI, provided one accepts the inevitable bias due to the missing stratification variable.

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	2.0	1.778	1.942	1.947	1.779	1.778	1.814	1.817	1.779
headage2	4.0	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000
headage3	4.0	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000
headage4	4.0	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000
headage5	4.0	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000
headage6	3.0	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000
headsex2	1.0	0.434	-0.202	-0.203	0.433	0.434	0.147	0.147	0.433
headocc2	-1.0	0.122	0.138	0.140	0.121	0.122	0.153	0.153	0.122
headocc3	-2.0	-0.027	0.106	0.108	-0.027	-0.027	0.048	0.048	-0.027
headocc4	2.0	-0.554	-0.104	-0.108	-0.554	-0.554	-0.304	-0.307	-0.554
headedu2	1.0	-0.170	-0.408	-0.417	-0.171	-0.170	-0.232	-0.237	-0.171
headedu3	2.0	-0.057	-0.123	-0.127	-0.057	-0.057	-0.062	-0.064	-0.057
headedu4	3.0	0.009	-0.082	-0.084	0.009	0.009	-0.009	-0.010	0.009
region2	1.5	-0.028	-0.216	-0.219	-0.028	-0.028	-0.106	-0.107	-0.028
region3	1.5	0.166	-0.138	-0.137	0.165	0.166	0.020	0.021	0.166
hhtype2	1.0	-0.440	-0.427	-0.438	-0.440	-0.440	-0.392	-0.398	-0.440

Table 6.8: Simulation results for relative biases of estimators of linear regression coefficients under MSI

[1]Unweighted Model Specification
[2]Weighted Model Specification

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	2.0	0.517	0.870	0.869	0.518	0.517	0.689	0.689	0.518
headage2	4.0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
headage3	4.0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
headage4	4.0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
headage5	4.0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
headage6	3.0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
headsex2	1.0	0.872	1.157	1.155	0.873	0.872	1.036	1.036	0.872
headocc2	-1.0	0.936	1.319	1.316	0.938	0.936	1.104	1.103	0.937
headocc3	-2.0	0.471	0.776	0.775	0.472	0.471	0.621	0.622	0.472
headocc4	2.0	0.557	0.871	0.871	0.557	0.557	0.763	0.763	0.557
headedu2	1.0	0.524	0.795	0.790	0.526	0.524	0.664	0.662	0.525
headedu3	2.0	0.213	0.454	0.452	0.214	0.213	0.335	0.335	0.213
headedu4	3.0	0.219	0.452	0.451	0.220	0.219	0.336	0.336	0.220
region2	1.5	0.227	0.577	0.575	0.228	0.227	0.427	0.426	0.227
region3	1.5	0.209	0.545	0.545	0.210	0.209	0.430	0.430	0.210
hhtype2	1.0	0.459	0.692	0.690	0.460	0.459	0.571	0.570	0.459

Table 6.9: Simulation results for relative variances of estimators of linear regression coefficients under MSI

[1]Unweighted Model Specification
[2]Weighted Model Specification

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	2.0	1.852	2.128	2.132	1.853	1.852	1.940	1.943	1.853
headage2	4.0	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
headage3	4.0	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
headage4	4.0	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
headage5	4.0	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
headage6	3.0	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
headsex2	1.0	0.974	1.174	1.173	0.975	0.974	1.046	1.046	0.974
headocc2	-1.0	0.944	1.326	1.323	0.946	0.944	1.115	1.114	0.945
headocc3	-2.0	0.472	0.783	0.783	0.473	0.472	0.623	0.624	0.472
headocc4	2.0	0.786	0.877	0.878	0.786	0.786	0.821	0.823	0.786
headedu2	1.0	0.551	0.893	0.893	0.553	0.551	0.703	0.703	0.552
headedu3	2.0	0.220	0.470	0.470	0.221	0.220	0.341	0.341	0.221
headedu4	3.0	0.220	0.459	0.459	0.220	0.220	0.336	0.336	0.220
region2	1.5	0.228	0.616	0.615	0.229	0.228	0.440	0.439	0.229
region3	1.5	0.267	0.562	0.562	0.268	0.267	0.431	0.431	0.267
hhtype2	1.0	0.636	0.813	0.817	0.637	0.636	0.693	0.695	0.636

Table 6.10: Simulation results for relative root mean squared errors of estimators of linear regression coefficients under MSI

[1]Unweighted Model Specification
[2]Weighted Model Specification

6.3.3 Simulation Results for Scenario 3: RBS

The Response-Based Sampling (RBS) scenario considered in this thesis is one where the sample and the corresponding population have different distributions for the target response variable Y because the stratification variable used in the sample design is also Y. From the review of this situation in Chapter 2, we know that sample weighting can fix this issue, provided these weights are first re-scaled by multiplying by nN^{-1} in order to ensure that summation of the sample weights for all sample units equals the actual sample size n. The necessity for this re-scaling was confirmed in our simulations, where results obtained using re-scaled sample weights lead to more reasonable results than those obtained using the usual inverse selection probability sample weights. As a result, for this case it is clear that when comparing UMS and WMS, the model fit defined using sample weights (WMS) should be used instead of the model fit defined by ignoring sample weights (UMS).

From Table 6.4, we see that there are 4799 cases out of 10000 Monte Carlo replicates where non-equivalent regressor sets were identified when model selection was carried out with and without sample weighting under RBS. Again, we used the five model selection strategies outlined earlier (i.e. $Vuong_{NW}$, $Vuong_{W}$, CV_{NW}, CV_{W} and *Voting System*) to choose a final model in each simulation, and show the resulting relative values of bias, variance and root mean squared error for the actual regression model coefficients in Tables 6.11-6.13.

As already noted above, the bias results displayed in Table 6.11 in the case of correct model specification (direct choice, third column) show that the regression estimators obtained under WMS typically provide lower values of RB compared with the estimators obtained under UMS. Turning to the fourth column of this table, it can be seen that $V_{NW_{small}}$ and CV_{W} are the two strategies that lead to estimators with the least bias in almost all cases. All the remaining strategies have comparatively larger biases, with the CV_{NW} strategy the worst in this regard.

Turning next to the RV results displayed in Table 6.12, these show that, for the direct-choice case, the WMS fit should be adopted because its outcomes exhibit

smaller relative variance than those generated by the UMS fit. In the fourth column, we see that $V_{NW_{small}}$ and CV_W are the two strategies that lead to the lowest values of RV for almost all estimators of model coefficients. In contrast, the CV_{NW} strategy leads to the worst RV results. Furthermore the relative RMSE results set out in Table 6.13 are in the same direction as those displayed in Table 6.12. What is more interesting, however, is that, with the exception of CV_{NW}, all the model choice strategies lead to lower values of RRMSE when compared with the RRMSE values recorded by UMS under the direct-choice approach.

We conclude that the $V_{NW_{small}}$ and CV_W strategies represent the best strategies for choosing a final model under the RBS scenario in our simulations, while the CV_{NW} strategy appears to be the worst. In addition, all five 'acceptable' strategies (i.e. $V_{NW_{small}}$, $V_{W_{small}}$, CV_W, VS_{small} and VS_{large}) appear reasonable options for use for model choice because they all lead to lower values of RRMSE when compared with the RRMSE values generated by UMS under the RBS scenario.

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	2.0	0.423	0.098	0.110	0.400	0.213	0.113	0.120	0.214
headage2	4.0	-0.010	-0.008	-0.008	-0.010	-0.008	-0.007	-0.008	-0.008
headage3	4.0	-0.002	-0.005	-0.005	-0.002	-0.003	-0.005	-0.005	-0.003
headage4	4.0	-0.007	-0.007	-0.007	-0.007	-0.006	-0.007	-0.007	-0.006
headage5	4.0	0.005	0.000	0.000	0.005	0.003	0.000	0.000	0.003
headage6	3.0	-0.002	-0.008	-0.008	-0.003	-0.004	-0.007	-0.007	-0.004
headsex2	1.0	-0.256	-0.152	-0.156	-0.245	-0.213	-0.152	-0.156	-0.206
headocc2	-1.0	0.149	0.135	0.137	0.141	0.188	0.134	0.137	0.178
headocc3	-2.0	0.057	0.053	0.053	0.054	0.073	0.052	0.053	0.069
headocc4	2.0	0.077	0.066	0.067	0.074	0.094	0.066	0.067	0.090
headedu2	1.0	-0.152	-0.093	-0.094	-0.149	-0.121	-0.095	-0.096	-0.120
headedu3	2.0	-0.014	-0.002	-0.002	-0.012	-0.006	-0.002	-0.002	-0.006
headedu4	3.0	-0.017	0.004	0.003	-0.015	-0.001	0.003	0.003	-0.002
region2	1.5	-0.820	-0.475	-0.487	-0.799	-0.626	-0.491	-0.499	-0.624
region3	1.5	-0.741	-0.721	-0.720	-0.756	-0.740	-0.736	-0.731	-0.744
hhtype2	1.0	-0.028	-0.003	-0.004	-0.026	-0.009	-0.004	-0.004	-0.010

Table 6.11: Simulation results for relative biases of estimators of linear regression coefficients under RBS

[1]Unweighted Model Specification
[2]Weighted Model Specification

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	2.0	0.600	0.559	0.569	0.596	0.619	0.549	0.563	0.612
headage2	4.0	0.171	0.159	0.159	0.169	0.164	0.160	0.160	0.163
headage3	4.0	0.165	0.153	0.154	0.164	0.159	0.154	0.154	0.158
headage4	4.0	0.162	0.151	0.151	0.161	0.156	0.152	0.152	0.156
headage5	4.0	0.161	0.151	0.151	0.161	0.157	0.152	0.152	0.156
headage6	3.0	0.202	0.188	0.188	0.201	0.195	0.190	0.189	0.195
headsex2	1.0	0.558	0.488	0.490	0.554	0.541	0.489	0.492	0.536
headocc2	-1.0	0.700	0.625	0.626	0.692	0.685	0.626	0.629	0.679
headocc3	-2.0	0.336	0.300	0.301	0.332	0.325	0.301	0.303	0.323
headocc4	2.0	0.354	0.313	0.314	0.351	0.337	0.314	0.315	0.335
headedu2	1.0	0.317	0.273	0.274	0.315	0.290	0.274	0.275	0.289
headedu3	2.0	0.148	0.133	0.133	0.147	0.139	0.134	0.134	0.139
headedu4	3.0	0.154	0.139	0.139	0.153	0.146	0.139	0.140	0.146
region2	1.5	0.385	0.371	0.381	0.388	0.385	0.353	0.367	0.381
region3	1.5	0.280	0.256	0.257	0.279	0.269	0.262	0.261	0.270
hhtype2	1.0	0.232	0.212	0.212	0.231	0.216	0.213	0.212	0.217

Table 6.12: Simulation results for relative variances of estimators of linear regression coefficients under RBS

[1]Unweighted Model Specification
[2]Weighted Model Specification

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	2.0	0.734	0.567	0.580	0.718	0.655	0.560	0.576	0.649
headage2	4.0	0.171	0.159	0.159	0.170	0.164	0.160	0.160	0.164
headage3	4.0	0.165	0.154	0.154	0.164	0.159	0.154	0.154	0.158
headage4	4.0	0.162	0.151	0.151	0.161	0.157	0.152	0.152	0.156
headage5	4.0	0.161	0.151	0.151	0.161	0.157	0.152	0.152	0.156
headage6	3.0	0.202	0.188	0.188	0.201	0.195	0.190	0.190	0.195
headsex2	1.0	0.614	0.511	0.514	0.606	0.581	0.512	0.516	0.574
headocc2	-1.0	0.715	0.639	0.641	0.706	0.710	0.641	0.644	0.702
headocc3	-2.0	0.341	0.305	0.306	0.337	0.333	0.306	0.307	0.330
headocc4	2.0	0.363	0.320	0.321	0.359	0.350	0.321	0.322	0.347
headedu2	1.0	0.352	0.288	0.290	0.348	0.314	0.290	0.291	0.313
headedu3	2.0	0.148	0.133	0.133	0.147	0.139	0.134	0.134	0.139
headedu4	3.0	0.155	0.139	0.139	0.154	0.146	0.139	0.140	0.146
region2	1.5	0.906	0.603	0.618	0.888	0.735	0.605	0.620	0.731
region3	1.5	0.792	0.766	0.765	0.805	0.787	0.782	0.777	0.792
hhtype2	1.0	0.234	0.212	0.212	0.233	0.216	0.213	0.212	0.217

Table 6.13: Simulation results for relative root mean squared errors of linear regression coefficients under RBS

[1]Unweighted Model Specification
[2]Weighted Model Specification

6.4 Simulation of Logistic Regression

We use the same values of $\mathbf{Z}$ and $\mathbf{X}$ as employed in the simulations reported in the previous section. This leads to a simulation of a population of size $N = 4617$ with corresponding values of Y defined by

$$logit\{\Pr(Y = 1|Z, \mathbf{X})\} = -1 + Z\underset{\sim}{a} + \mathbf{X}\underset{\sim}{b} \tag{6.2}$$

where $logit\{\Pr(Y = 1|Z, \mathbf{X})\} = \log\left(\frac{\Pr(Y=1|Z,\mathbf{X})}{1-\Pr(Y=1|Z,\mathbf{X})}\right)$; $\Pr(Y = 1|Z, \mathbf{X})$ denotes the probability that the binary response Y takes the value 1 given the values of the stratification variables $\mathbf{Z}$ and a set of independent variables $\mathbf{X}$, and the model parameters in equation (6.2) are the vectors

$$\underset{\sim}{a} = (3, 3, 3, 3, 3)^T \text{ and } \underset{\sim}{b} = (0.5, -0.5, -1, 0.5, 0.5, 1, 2, 0.5, 0.5, 0.5)^T.$$

A finite population was then created by combining the values of the binary response variable Y generated from equation (6.2) and the existing data sets of $\mathbf{Z}$ and $\mathbf{X}$. These finite population data were then classified into strata using the values of the stratifying variable appropriate to the scenario employed. That is, the finite population data were classified using the $\mathbf{Z}$ variable for the NIS and MSI scenarios, and using the Y variable for the RBS scenario. A sample was then randomly drawn from this finite population using simple stratified random sampling, and sample weights were constructed as detailed in Table 6.14. Finally, the suggested model specification and model selection tools were employed in order to achieve a final model for inference.

An total of 10000 independent simulations were carried out. These simulations were executed using the `R` code set out in Appendix B. The `R` software package was used for selecting samples and fitting the logistic regression model. Table 6.15 shows the number of simulations where final models corresponded to non-equivalent regressor sets. That is, the numbers set out in Table 6.15 indicate that there were 9742, 9794 and 10000 simulations out of 10000 where non-equivalent regressor sets

were achieved under NIS, MSI and RBS scenarios respectively. Tables 6.16-6.24 show the Monte Carlo values of relative biases, relative root mean squared errors and relative variances of the regression coefficient estimators defined by the final models identified using the model selection strategies shown in Table 6.15.

stratum (h)	N_h	n_h	w_h
Scenario 1: Non-Informative Sampling Scenario 2: Missing Stratification Information			
1	120	100	12.00
2	371	150	2.47
3	522	125	4.18
4	641	75	8.55
5	608	100	6.08
6	2355	50	47.10
Scenario 3: Response-Based Sampling			
1	2000	400	5.00
2	2617	200	13.09

Table 6.14: Stratum sample allocations and corresponding expansion-type sample weights used in simulation of logistic regression modelling under the three scenarios.

Strategy	Final Model Choices		
	Unweighted Model Specification	Weighted Model Specification	Equivalence
Scenario 1: Non-Informative Sampling (NIS)			
$Vuong_{NW}$	2073	1080	6589
$Vuong_{W}$	10	823	8909
CV_{NW}	6373	3369	-
CV_{W}	6312	3430	-
Voting System	3829	2737	3176
Scenario 2: Missing Stratification Information (MSI)			
$Vuong_{NW}$	31	7513	2250
$Vuong_{W}$	-	9409	385
CV_{NW}	576	9218	-
CV_{W}	553	9241	-
Voting System	53	9240	501
Scenario 3: Response-Based Sampling (RBS)			
$Vuong_{NW}$	-	10000	-
$Vuong_{W}$	-	10000	-
CV_{NW}	165	9835	-
CV_{W}	13	9987	-
Voting System	-	9989	11

Table 6.15: Simulation results for final model choices using four model search strategies for the case of logistic regression.

6.4.1 Simulation Results for Scenario 1: NIS

As we have already noted, we expect that the optimal model choice under NIS should be that dictated by UMS. From Table 6.15 we see that there are 9742 cases over 10000 replications where non-equivalent regressor sets were generated under NIS. For these cases, Tables 6.16-6.18 show the relative biases, relative variances and relative root mean square errors of the estimators of the regression coefficients of the final models obtained under the five model search strategies (i.e. $Vuong_{NW}$, $Vuong_{W}$, CV_{NW}, CV_{W} and *Voting System*).

The results set out in Table 6.15 indicate that all strategies, except for $Vuong_{W}$, prefer the UMS model to the WMS model, as expected. Furthermore, for these 9742 cases of non-equivalent regressor sets, the results displayed in the third column of Table 6.16 show that UMS leads to estimators with smallest bias. In particular, the estimators of the intercept regression coefficient generated by the chosen models under the WMS approach had higher biases than those generated by the UMS-based models. The reason for this is explained in Prentice and Pyke (1979), who point out that this apparent bias is caused by the impact of the sampling design, which generates an effect even though the model itself is valid. This bias is increased because the iterative estimation method used to fit the logistic model interacts with the sample weights used for estimating the logistic regression model coefficients. From the results set out in Table 6.16, we can see that the VS_{large} strategy provides the best outcomes in terms of minimal RB. However, if we examine the RB values generated by the other strategies, we see that they lead to estimators with less bias than the estimators generated by WMS under direct choice. Furthermore, the two CV-based strategies appear to be more effective than the two strategies based on Vuong's method as far as RB associated with choosing the final model is concerned.

These conclusions based on the RB results set out in Table 6.16 are repeated when we consider the results displayed in Tables 6.17-6.18. Consequently we conclude that the VS_{large} strategy works reasonably well for choosing a final model for logistic regression under the NIS scenario. However, we also note that the other strategies

remain effective for model choice in this situation.

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	-1.0	0.558	3.934	1.936	1.755	2.241	1.737	2.346	1.368
headage2	3.0	0.007	0.069	0.048	0.032	0.040	0.032	0.050	0.027
headage3	3.0	0.021	0.082	0.061	0.045	0.056	0.045	0.064	0.041
headage4	3.0	0.031	0.098	0.077	0.061	0.072	0.061	0.080	0.056
headage5	3.0	0.035	0.096	0.076	0.060	0.071	0.060	0.079	0.056
headage6	3.0	0.027	0.164	0.135	0.120	0.131	0.121	0.138	0.115
headsex2	0.5	-0.450	0.309	-0.120	-0.178	-0.132	-0.174	-0.051	-0.259
headocc2	-0.5	-0.122	-6.958	-2.910	-2.569	-3.244	-2.530	-3.696	-1.738
headocc3	-1.0	-0.073	-3.544	-1.523	-1.293	-1.690	-1.272	-1.904	-0.883
headocc4	0.5	0.277	-5.572	-1.628	-1.535	-2.202	-1.486	-2.297	-0.793
headedu2	0.5	-0.515	-0.432	-0.511	-0.520	-0.607	-0.520	-0.529	-0.548
headedu3	1.0	-0.263	-0.260	-0.248	-0.276	-0.378	-0.277	-0.283	-0.288
headedu4	2.0	-0.223	0.427	0.170	0.086	-0.017	0.076	0.168	0.014
region2	0.5	-0.515	-0.389	-0.474	-0.525	-0.539	-0.519	-0.493	-0.539
region3	0.5	-0.410	-0.419	-0.426	-0.473	-0.514	-0.473	-0.462	-0.479
hhtype2	0.5	-0.508	-0.661	-0.606	-0.575	-0.709	-0.585	-0.643	-0.582

Table 6.16: Simulation results for relative biases of estimators of logistic regression coefficients under NIS

[1]Unweighted Model Specification
[2]Weighted Model Specification

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	-1.0	1.192	6.337	4.428	4.170	4.661	4.146	4.884	3.502
headage2	3.0	0.136	0.165	0.154	0.142	0.158	0.144	0.154	0.142
headage3	3.0	0.150	0.176	0.166	0.156	0.170	0.157	0.167	0.155
headage4	3.0	0.183	0.266	0.259	0.252	0.262	0.254	0.260	0.252
headage5	3.0	0.167	0.205	0.196	0.188	0.198	0.189	0.196	0.187
headage6	3.0	0.235	0.655	0.655	0.655	0.654	0.655	0.655	0.655
headsex2	0.5	1.475	3.500	2.664	2.393	2.724	2.433	2.829	2.219
headocc2	-0.5	1.981	12.569	8.778	8.235	9.263	8.169	9.691	6.867
headocc3	-1.0	1.057	6.342	4.458	4.161	4.696	4.132	4.901	3.494
headocc4	0.5	2.036	12.976	8.673	8.116	9.361	8.112	9.711	6.722
headedu2	0.5	0.813	1.213	1.019	0.915	1.024	0.930	1.037	0.887
headedu3	1.0	0.572	0.838	0.734	0.661	0.768	0.670	0.749	0.657
headedu4	2.0	0.595	1.991	1.680	1.481	1.405	1.456	1.665	1.368
region2	0.5	0.795	1.226	1.033	0.894	1.065	0.912	1.051	0.877
region3	0.5	0.864	1.245	1.082	0.947	1.089	0.951	1.088	0.927
hhtype2	0.5	0.764	1.011	0.886	0.821	0.875	0.823	0.889	0.802

Table 6.17: Simulation results for relative variances of estimators of logistic regression coefficients under NIS

[1]Unweighted Model Specification
[2]Weighted Model Specification

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	-1.0	1.316	7.459	4.833	4.524	5.172	4.495	5.418	3.760
headage2	3.0	0.136	0.179	0.161	0.146	0.163	0.147	0.162	0.145
headage3	3.0	0.151	0.194	0.177	0.162	0.179	0.164	0.178	0.161
headage4	3.0	0.186	0.284	0.271	0.260	0.271	0.261	0.272	0.259
headage5	3.0	0.170	0.227	0.210	0.197	0.211	0.198	0.211	0.195
headage6	3.0	0.237	0.676	0.669	0.666	0.667	0.666	0.669	0.665
headsex2	0.5	1.542	3.514	2.667	2.400	2.728	2.439	2.830	2.234
headocc2	-0.5	1.985	14.366	9.248	8.626	9.814	8.552	10.372	7.084
headocc3	-1.0	1.060	7.265	4.711	4.358	4.991	4.323	5.258	3.604
headocc4	0.5	2.055	14.122	8.825	8.259	9.616	8.247	9.979	6.769
headedu2	0.5	0.962	1.288	1.140	1.052	1.191	1.066	1.164	1.042
headedu3	1.0	0.630	0.878	0.775	0.717	0.856	0.725	0.801	0.717
headedu4	2.0	0.636	2.037	1.688	1.484	1.406	1.458	1.674	1.368
region2	0.5	0.948	1.287	1.137	1.037	1.194	1.050	1.161	1.030
region3	0.5	0.957	1.314	1.163	1.058	1.204	1.062	1.182	1.044
hhtype2	0.5	0.918	1.208	1.074	1.002	1.126	1.010	1.097	0.991

Table 6.18: Simulation results for relative root mean squared errors of estimators of logistic regression coefficients under NIS

[1]Unweighted Model Specification
[2]Weighted Model Specification

6.4.2 Simulation Results for Scenario 2: MSI

Here the distributions of the target response variable in the sample and in the corresponding population are different because the variable used for stratification is missing from the set of variables used to specify the sample model. Here this stratification variable is *headage*. We know from theory developed in Chapter 4 that omitting this variable from the model then means that both UMS and WMS based on the incomplete covariate set lead to biased inference for the actual population model coefficients.

We see from Table 6.15 that there are 9794 cases of non-equivalent regressor sets generated by UMS and WMS over the 10000 simulations used in the study. Furthermore, all strategies (i.e. $Vuong_{NW}$, $Vuong_{W}$, CV_{NW}, CV_{W} and *Voting System*) tend to choose a WMS-based model rather than a UMS-based model. Tables 6.19-6.21 show the relative bias, relative variance and relative root mean square errors for the estimators of the population model coefficients as displayed in Table 6.15.

The third column (direct choice) in Table 6.19 shows that UMS leads to less biased estimators than WMS. However, it is also clear that both UMS and WMS lead to biased results. This is in line with the theory developed in Chapter 4. That is, both model specifications are biased due to model misspecification. As a consequence, the results set out in the fourth column of Table 6.19 indicate that all six model-choice strategies lead to highly biased estimators because they are largely based on the WMS specification. See Subsection 6.5.2 for supporting results for this situation. These results are consistent with those set out in Table 6.26, which indicate that the bias of the estimators in a MSI scenario depends on the target of inference. That is, both the unweighted and weighted model specifications are biased when the targets of inference are misspecified.

The results shown in in the third column of Table 6.20 indicate that UMS leads to smaller values of RV for estimators of the actual population regression model coefficients compared with WMS when the final models are directly chosen. In the fourth column of this table we see that CV_{NW} and VS_{large} are the two model-choice

strategies that provide the best results as far as RV is concerned. Furthermore, we see that the two CV-based model choice strategies lead to smaller values of RV than the two strategies based on Vuong's method. Finally, it can be seen that the results set out in Table 6.21 are generally in the same direction of those displayed in Table 6.20.

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	-1.0	2.572	7.008	6.732	6.704	6.989	6.707	6.966	6.698
headage2	3.0	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000
headage3	3.0	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000
headage4	3.0	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000
headage5	3.0	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000
headage6	3.0	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000	-1.000
headsex2	0.5	-0.105	0.198	0.179	0.236	0.196	0.233	0.205	0.234
headocc2	-0.5	0.772	-6.770	-6.249	-6.193	-6.732	-6.200	-6.689	-6.182
headocc3	-1.0	0.369	-3.478	-3.219	-3.180	-3.460	-3.185	-3.437	-3.176
headocc4	0.5	-0.784	-5.715	-5.283	-5.310	-5.678	-5.315	-5.650	-5.290
headedu2	0.5	-0.749	-0.618	-0.620	-0.658	-0.618	-0.654	-0.620	-0.654
headedu3	1.0	-0.457	-0.447	-0.437	-0.471	-0.446	-0.469	-0.446	-0.468
headedu4	2.0	-0.199	0.332	0.322	0.281	0.332	0.285	0.330	0.284
region2	0.5	-0.741	-0.540	-0.550	-0.570	-0.540	-0.569	-0.542	-0.567
region3	0.5	-0.157	-0.397	-0.375	-0.408	-0.396	-0.407	-0.396	-0.404
hhtype2	0.5	-1.017	-0.794	-0.812	-0.852	-0.794	-0.848	-0.798	-0.847

Table 6.19: Simulation results for relative biases of estimators of logistic regression coefficients under MSI

[1]Unweighted Model Specification
[2]Weighted Model Specification

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	-1.0	0.718	6.211	6.083	6.076	6.202	6.078	6.192	6.072
headage2	3.0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
headage3	3.0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
headage4	3.0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
headage5	3.0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
headage6	3.0	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
headsex2	0.5	1.526	3.334	3.265	3.202	3.332	3.211	3.324	3.228
headocc2	-0.5	1.062	12.320	12.016	11.975	12.299	11.979	12.268	11.968
headocc3	-1.0	0.678	6.223	6.077	6.057	6.213	6.060	6.199	6.055
headocc4	0.5	0.943	12.737	12.373	12.295	12.710	12.304	12.684	12.293
headedu2	0.5	0.543	0.998	0.986	0.929	0.998	0.934	0.993	0.935
headedu3	1.0	0.450	0.736	0.730	0.702	0.736	0.704	0.735	0.706
headedu4	2.0	0.509	1.954	1.937	1.895	1.954	1.898	1.950	1.898
region2	0.5	0.542	1.029	1.013	0.975	1.029	0.978	1.024	0.980
region3	0.5	0.802	1.184	1.180	1.138	1.184	1.140	1.181	1.142
hhtype2	0.5	0.253	0.798	0.770	0.706	0.797	0.714	0.791	0.715

Table 6.20: Simulation results for relative variances of estimators of logistic regression coefficients under MSI

[1]Unweighted Model Specification
[2]Weighted Model Specification

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	-1.0	2.670	9.364	9.073	9.048	9.344	9.051	9.320	9.040
headage2	3.0	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
headage3	3.0	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
headage4	3.0	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
headage5	3.0	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
headage6	3.0	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
headsex2	0.5	1.529	3.340	3.270	3.211	3.338	3.220	3.330	3.236
headocc2	-0.5	1.313	14.058	13.544	13.482	14.021	13.488	13.973	13.470
headocc3	-1.0	0.772	7.129	6.876	6.841	7.111	6.846	7.088	6.837
headocc4	0.5	1.227	13.960	13.454	13.392	13.921	13.403	13.885	13.382
headedu2	0.5	0.925	1.174	1.165	1.138	1.174	1.140	1.171	1.141
headedu3	1.0	0.641	0.861	0.851	0.845	0.861	0.846	0.860	0.847
headedu4	2.0	0.547	1.982	1.964	1.916	1.982	1.920	1.978	1.920
region2	0.5	0.918	1.162	1.153	1.130	1.162	1.131	1.159	1.132
region3	0.5	0.817	1.248	1.238	1.209	1.248	1.211	1.246	1.211
hhtype2	0.5	1.048	1.126	1.120	1.107	1.125	1.109	1.124	1.109

Table 6.21: Simulation results for relative root mean squared errors of estimators of logistic regression coefficients under MSI

[1]Unweighted Model Specification
[2]Weighted Model Specification

6.4.3 Simulation Results for Scenario 3: RBS

The RBS scenario corresponds to a situation where the distributions of the target response variable in the sample and in the corresponding population are different because the sample inclusion depends on the target response variable Y. In particular, the sample design uses strata based on Y. As discussed in Chapter 2, in this situation the WMS should be adopted because it leads to less biased estimators.

Tables 6.22-6.24 show the values of RB, RV and RRMSE respectively for estimators of population regression model coefficients obtained using the five model-choice strategies, $Vuong_{NW}$, $Vuong_{W}$, CV_{NW}, CV_{W} and *Voting System* under this RBS scenario.

To start, we note that the results set out in column three of Table 6.22 show that estimators based on WMS are less biased than those based on UMS in almost all cases when direct choice is used. That is, WMS should be adopted under RBS if bias of estimators is a key concern. However, we also note that in the fourth column this advantage is no longer as obvious since RB values generated by all six strategies are almost the same. This indicates that WMS is not as important when a model-choice strategy is adopted.

Turning to the RV results displayed in Table 6.23 we see that for the direct-choice case these results are the opposite of the results set out in Table 6.22. All estimators, except for *region3* and *hhtype2*, obtained from UMS have less RV than corresponding estimators obtained from WMS. In the fourth column, we see that the CV_{NW} strategy provides estimators with the lowest RV for almost all regression model coefficients. Furthermore, it can be seen that the two CV-based strategies record smaller values of RV than the two strategies defined by Vuong's method.

Finally, we note that the RRMSE results set in Table 6.24 are in the same direction of those displayed in Table 6.23.

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	-1.0	0.760	0.606	0.606	0.603	0.606	0.608	0.606	0.606
headage2	3.0	-0.887	-0.771	-0.771	-0.772	-0.771	-0.772	-0.771	-0.771
headage3	3.0	-0.878	-0.766	-0.766	-0.766	-0.766	-0.767	-0.766	-0.766
headage4	3.0	-0.873	-0.762	-0.762	-0.763	-0.762	-0.763	-0.762	-0.762
headage5	3.0	-0.875	-0.750	-0.750	-0.751	-0.750	-0.751	-0.750	-0.750
headage6	3.0	-0.874	-0.759	-0.759	-0.760	-0.759	-0.760	-0.759	-0.759
headsex2	0.5	-0.854	-0.913	-0.913	-0.913	-0.913	-0.913	-0.913	-0.913
headocc2	-0.5	0.993	0.643	0.643	0.655	0.643	0.643	0.643	0.643
headocc3	-1.0	0.756	0.697	0.697	0.701	0.697	0.697	0.697	0.697
headocc4	0.5	-1.095	-1.258	-1.258	-1.252	-1.258	-1.259	-1.258	-1.259
headedu2	0.5	-0.955	-1.029	-1.029	-1.028	-1.029	-1.029	-1.029	-1.029
headedu3	1.0	-0.960	-0.915	-0.915	-0.917	-0.915	-0.915	-0.915	-0.915
headedu4	2.0	-0.985	-0.970	-0.970	-0.971	-0.970	-0.970	-0.970	-0.970
region2	0.5	-1.000	-39.571	-39.571	-38.925	-39.571	-39.522	-39.571	-39.530
region3	0.5	1.081	-0.986	-0.986	-0.956	-0.986	-0.984	-0.986	-0.984
hhtype2	0.5	-0.840	-0.954	-0.954	-0.951	-0.954	-0.954	-0.954	-0.954

Table 6.22: Simulation results for relative biases of estimators of logistic regression coefficients under RBS

[1]Unweighted Model Specification
[2]Weighted Model Specification

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	-1.0	0.897	3.392	3.392	3.356	3.392	3.388	3.392	3.392
headage2	3.0	0.259	1.007	1.007	0.998	1.007	1.005	1.007	1.007
headage3	3.0	0.263	1.005	1.005	0.996	1.005	1.003	1.005	1.005
headage4	3.0	0.264	1.006	1.006	0.997	1.006	1.004	1.006	1.006
headage5	3.0	0.263	1.010	1.010	1.000	1.010	1.008	1.010	1.010
headage6	3.0	0.257	1.004	1.004	0.995	1.004	1.003	1.004	1.004
headsex2	0.5	0.505	0.515	0.515	0.514	0.515	0.514	0.515	0.514
headocc2	-0.5	0.495	2.588	2.588	2.543	2.588	2.588	2.588	2.589
headocc3	-1.0	0.340	1.315	1.315	1.292	1.315	1.315	1.315	1.315
headocc4	0.5	0.554	2.566	2.566	2.522	2.566	2.566	2.566	2.566
headedu2	0.5	0.253	0.294	0.294	0.293	0.294	0.294	0.294	0.294
headedu3	1.0	0.155	0.233	0.233	0.231	0.233	0.233	0.233	0.233
headedu4	2.0	0.103	0.140	0.140	0.139	0.140	0.140	0.140	0.140
region2	0.5	0.000	3.166	3.166	5.813	3.166	3.458	3.166	3.415
region3	0.5	0.501	0.230	0.230	0.321	0.230	0.241	0.230	0.241
hhtype2	0.5	0.380	0.286	0.286	0.289	0.286	0.287	0.286	0.287

Table 6.23: Simulation results for relative variances of estimators of logistic regression coefficients under RBS

[1]Unweighted Model Specification
[2]Weighted Model Specification

Coefficient	True Parameter	Direct Choice		Strategy					
				Unweighted		Weighted		Voting System	
		UMS[1]	WMS[2]	$V_{NW_{small}}$	CV_{NW}	$V_{W_{small}}$	CV_W	VS_{small}	VS_{large}
Intercept	-1.0	1.175	3.446	3.446	3.410	3.446	3.442	3.446	3.446
headage2	3.0	0.924	1.268	1.268	1.261	1.268	1.268	1.268	1.268
headage3	3.0	0.916	1.264	1.264	1.257	1.264	1.263	1.264	1.264
headage4	3.0	0.912	1.262	1.262	1.255	1.262	1.261	1.262	1.262
headage5	3.0	0.913	1.258	1.258	1.251	1.258	1.257	1.258	1.258
headage6	3.0	0.911	1.259	1.259	1.252	1.259	1.258	1.259	1.259
headsex2	0.5	0.992	1.048	1.048	1.047	1.048	1.048	1.048	1.048
headocc2	-0.5	1.109	2.667	2.667	2.626	2.667	2.667	2.667	2.667
headocc3	-1.0	0.829	1.488	1.488	1.470	1.488	1.488	1.488	1.488
headocc4	0.5	1.227	2.858	2.858	2.816	2.858	2.858	2.858	2.858
headedu2	0.5	0.988	1.071	1.071	1.069	1.071	1.071	1.071	1.071
headedu3	1.0	0.973	0.944	0.944	0.945	0.944	0.944	0.944	0.945
headedu4	2.0	0.990	0.980	0.980	0.981	0.980	0.981	0.980	0.981
region2	0.5	1.000	39.698	39.698	39.357	39.698	39.673	39.698	39.677
region3	0.5	1.192	1.012	1.012	1.009	1.012	1.013	1.012	1.013
hhtype2	0.5	0.922	0.996	0.996	0.995	0.996	0.996	0.996	0.996

Table 6.24: Simulation results for relative root mean squared errors of estimators of logistic regression coefficients under RBS

[1]Unweighted Model Specification
[2]Weighted Model Specification

6.5 Simulation of Modelling Bias Under MSI

In Chapter 4, we provided analytic results on modelling bias for both simple linear and logistic regression under a MSI scenario. In this section we present simulation results that serve to illustrate this theory.

6.5.1 Modelling Bias for Linear Regression

As in Chapter 4, we assume that x and z are a binary independent variable and a binary stratifying variable respectively. We generate values for these two variables by first generating values for X and Z, where

$$\begin{pmatrix} X \\ Z \end{pmatrix} \sim N(0, \Sigma)$$

with

$$\Sigma = \begin{bmatrix} 1 & 0.1 \\ 0.1 & 1 \end{bmatrix}.$$

Note that setting the covariance to 0.1 above corresponds to simulating a small positive correlation of 0.1 between X and Z. We then put $x = 1$ when $X > 0$, otherwise $x = 0$. Similarly we set $z = 1$ when $Z > 0$, otherwise $z = 0$. With this definition, $P_0 = Pr(z_i = 1|x_i = 0) \cong 0.48$ and $P_1 = Pr(z_i = 1|x_i = 1) \cong 0.54$.

At each simulation, a finite population dataset of values of x and z of size $N = 5000$ was generated. Given these values of x and z, values of y were generated as

$$\begin{aligned} y_i &= a + bx_i + cz_i + \varepsilon_i \\ &= 1 + x_i + z_i + \varepsilon_i \qquad ; i = 1, 2, ..., N. \end{aligned}$$

Here $\varepsilon_i \sim N(0, 1)$. That is, $E(y_i|x_i, z_i) = 1+x_i+z_i$ and so the parameters of the conditional model interest (the target model) are the values $(1, 1, 1)$. The corresponding

unconditional model is obtained by averaging over z, and is given by

$$\begin{aligned} E(y_i|x_i) &= a^* + b^* x_i \\ &\cong 1.48 + 1.06 x_i \qquad ; i = 1, 2, ..., N. \end{aligned}$$

where

$$a^* = a + cP_0 = 1.48$$

and

$$b^* = b + P_1 - P_0 = 1.06.$$

These finite population data were then classified into two strata based on their values of z, with stratum sizes specified by $n_1 = 600$ ($z = 0$) and $n_2 = 400$ ($z = 1$), i.e. a total sample of size n = 1000. A sample was then randomly drawn from these finite population data using simple stratified sampling. Given the sample data, both unweighted and weighted linear model fits (UMS and WMS) to the sample values of y and x were implemented. Parameter estimates generated by these two fitted models were then stored.

A total of 1000 independent simulations of the above process was carried out. The Monte Carlo expected values of the parameter estimates generated by these simulations were calculated for both the UMS and WMS fits, and their Monte Carlo biases relative to conditional and unconditional target parameters were computed. These biases are set out in Table 6.25 in relative terms, and are consistent with the theoretical results obtained in Chapter 4.

Target of Inference	Relative Monte Carlo Bias			
	Intercept		Slope	
	OLS	WLS	OLS	WLS
$E(y_i\|x_i, z_i)$, i.e. a and b	0.378	0.475	0.047	0.049
$E(y_i\|x_i)$, i.e. a^* and b^*	-0.096	0.001	-0.002	0.000

Table 6.25: Simulation results for relative biases of estimates of linear regression model parameters under MSI

6.5.2 Modelling Bias for Logistic Regression

Simulations were carried out in the same manner as in the previous subsection. Values of X and Z were again generated from

$$\begin{pmatrix} X \\ Z \end{pmatrix} \sim N(0, \Sigma)$$

where

$$\Sigma = \begin{bmatrix} 1 & 0.1 \\ 0.1 & 1 \end{bmatrix}.$$

For each simulation, a finite population of binary values of x and z of size $N = 5000$ was then generated as in the linear regression case. Given these values of x and z, independent Bernoulli values of y were then generated under a conditional logistic regression model of the form

$$\begin{aligned} logit\{\Pr(y_i = 1|x_i, z_i)\} &= a + bx_i + cz_i \\ &= 1 + x_i + z_i \qquad ; i = 1, 2, ..., N \end{aligned}$$

with the corresponding unconditional logistic regression model given by

$$logit\{\Pr(y_i = 1|x_i)\} = a + bx_i g(f(x_i)) \qquad ; i = 1, 2, ..., N$$

where $g(f(x)) = \log\left(\frac{f(x)}{1+e^{a+bX}(1-f(x))}\right)$; $f(x) = 1 + \frac{e^c-1}{1+e^{a+bX+c}}[P_0 + (P_1 - P_0)X]$.

Again, the finite population data were classified into two strata according to $z = 0$ and $z = 1$ with strata sizes specified by $n_1 = 600$ and $n_2 = 400$ respectively. A sample of size $n = 1000$ was then randomly drawn from this finite population using simple stratified sampling. Using the sample values of y and x thus obtained, unweighted and weighted logistic fits (UMS and WMS) were obtained using these sample data and the resulting estimates for the intercept and slope coefficients then stored.

A total of 1000 independent simulations of the above procedure was carried out. The average Monte Carlo biases generated by these simulations are set out in Table 6.26. As predicted by the theoretical results developed in Chapter 4, we see that use of a WMS-based approach is of little use here, with the UMS biases usually a little smaller than the WMS biases irrespective of whether the target of inference is the original (conditional) model or the unconditional model that averages over z.

Target of Inference	Average Monte Carlo Bias			
	Intercept		Slope	
	OLS	WLS	OLS	WLS
$logit(Y = 1\|X, Z)$, i.e. a and b	0.313	0.402	0.037	0.040
$logit(Y = 1\|X)$ i.e. a and $bg(f(x_i))$	0.276	0.347	0.018	-0.015

Table 6.26: Simulation results for average biases of estimates of logistic regression model parameters under MSI

6.6 Application of Modelling Procedure To Indian National Family Health Survey Data

In previous sections we used Monte Carlo simulation to evaluate the efficiency of the different modelling procedures described in Chapter 5. These simulated population data were based on variables collected in the INFHS. In this section we apply our modelling procedure to the original INFHS data set in order to fit linear and logistic regression models to these data.

As mentioned at the start of this chapter, one of the main the purposes for which the original INFHS data was collected was to carry out population modelling of the survey data in order to show how the health status of Indian households varies across the country. One way of doing this is to analyse the INFHS data using regression analysis. Here we use two types of regression: linear and logistic. The dependent variable employed in our linear regression analysis is *density* defined as *hhsize* divided by *hhrooms*. Household crowding, as measured by *density*, is important for the health of the household. The more crowded a household is, the more susceptible

its members are to the spread of infection. In addition, household crowding may represent the presence of large extended families, which are commonplace in India. The power structures and decision-making behaviour in such families have been shown to be influential in health care use. For our logistic regression analysis, we use the binary response variable *pipe* as a way of characterising the sanitation status of a household. This variable is coded 1 for piped water and 0 for other water sources. Sources of water other than a piped supply include public pipes, wells, tankers and lakes and rivers. These are considered inferior to having a water supply piped into the house, and are associated with an increased risk of infectious disease transmission. That is, if a household does not have piped water, it indicates that the household may have poor hygiene.

The independent variables available for these regression analyses are of three types: the geographic location of a household, the characteristics of the household head and the number of household toilets. The geographic location variables aref *region*, *hhtype* and *hhstate*. The variables that characterise the household head are *headsex*, *headage*, *headmar*, *headed*, *headocc* and *religion*. Finally, there is the variable *toilet*, which is the number of toilets in the house. All three types of variables can be used as potential dependent variables for regression modelling of *density*, while only the first two types are relevant for modelling whether there is piped water to the household (since it seems reasonable to assume that there must be piped water to the house if $toilet > 0$).

6.6.1 Linear Regression Modelling of Household Density

The distribution of *density* in the INFHS data is right-skewed and heteroskedastic, with variability increasing with *density*. Consequently we model the natural logarithmic transformation of *density* so that that the resulting transformed variable then meets the usual linear model assumptions. As can be seen in Table 6.2, the key sample design variable *hhstate* and the sampling weight are included in the set of potential model variables. This corresponds to the NIS scenario in our study. We

therefore model the regression of the natural log of household density as a function of the other variables in Table 6.2.

Following the modelling process set out in Figure 3.1 is then straightforward. We implement this procedure as follows:

1. Import the INFHS data.

2. Fit Model U by regressing log *density* on the complete set of independent variables defined in Table 6.2 using standard (unweighted) backward elimination.

3. Fit Model W by regressing log *density* on the complete set of independent variables defined in Table 6.2 using weighted backward elimination.

4. Use ordinary least squares to fit Model U to the INFHS data.

5. Use weighted least squares to fit Model W to the INFHS data.

6. For the INFHS data, Model U and Model W are not the same. We therefore use model search strategies I to IV defined by V_{NW}, V_W, CV_{NW} and CV_W to identify the final model.

With the exception of model search strategy V_W, the remaining three strategies all select Model U. The parameter estimates defined by this fit are shown in Table 6.27. They indicate that all three types of independent variables significantly affect the log of household density. That is, the significant factors associated with the density of a household are the state where the household is located, the age of the head of household, the sex of the head of household, the occupation of the head of household, the education level of the head of household, the religion of the head of household, whether the household is urban or rural, and number of toilets in household.

Coefficients	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	1.031	0.041	25.104	< 2e-16	***
hhstate4	0.146	0.019	7.517	6.71e-14	***
hhstate7	0.132	0.019	6.884	6.62e-12	***
hhstate8	-0.045	0.020	-2.220	0.026478	*
hhstate9	0.134	0.019	7.108	1.36e-12	***
hhstate12	0.110	0.018	6.015	1.94e-09	***
hhstate24	0.155	0.019	8.038	1.15e-15	***
headage2	0.136	0.039	3.476	0.000514	***
headage3	0.215	0.038	5.714	1.17e-08	***
headage4	0.213	0.037	5.761	8.91e-09	***
headage5	0.218	0.037	5.867	4.76e-09	***
headage6	0.148	0.035	4.241	2.27e-05	***
headsex2	-0.130	0.021	-6.253	4.39e-10	***
headocc2	0.044	0.018	2.380	0.017352	*
headocc3	0.063	0.017	3.740	0.000186	***
headed3	-0.090	0.015	-6.136	9.16e-10	***
headed4	-0.206	0.022	-9.417	< 2e-16	***
religion3	0.072	0.018	3.913	9.25e-05	***
hhtype2	-0.056	0.015	-3.780	0.000159	***
toilet2	0.091	0.033	2.781	0.005438	**
toilet3	0.141	0.019	7.467	9.78e-14	***
toilet4	0.107	0.021	5.182	2.28e-07	***

Table 6.27: The final linear regression model (Model U) for log *density*

6.6.2 Logistic Regression Modelling of Household Piped Water

The proposed modelling procedure set out in Figure 3.1 was again employed to define a logistic regression model for the variable *pipe* in the INFHS data. Note that in this case we did not use the *toilet* variable as a candidate independent variable in the logistic model. However all the other variables listed in Table 6.2 were considered. Once again, the procedure was implemented as follows:

1. Import the INFHS data.
2. Fit Model U by as the unweighted logistic regression of *pipe* on the independent variables defined in Table 6.2 (excluding *toilet*) using backward elimination based on standard (unweighted) ML estimation.
3. Fit Model W as the weighted logistic regression of *pipe* on the independent variables defined in Table 6.2 (excluding *toilet*) using backward elimination based on weighted ML (pseudo-ML) estimation.
4. Use standard unweighted ML estimation to fit Model U to the INFHS data.
5. Use weighted ML (pseudo-ML) estimation Model W to the INFHS data.
6. For the INFHS data, Model U and Model W are once again not the same. We therefore use model search strategies I to IV defined by V_{NW}, V_W, CV_{NW} and CV_W to identify the final model.

In this case, all strategies except for V_{NW} choose Model U or the unweighted model specification (UMS). The parameter estimates generated under Model U are set out in Table 6.28. These indicate that almost all the independent variables that we considered significantly affected the probability of a household having a water pipe. These significant variables are the state where the household is located, the age of the head of household, the occupation of the head of household, the education level of the head of household, and whether the household is rural or urban.

Coefficients	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	2.673	0.216	12.363	< 2e-16	***
hhstate7	-1.446	0.157	-9.245	< 2e-16	***
hhstate8	-2.223	0.168	-13.246	< 2e-16	***
hhstate9	-2.627	0.160	-16.405	< 2e-16	***
hhstate12	-1.355	0.158	-8.552	< 2e-16	***
hhstate18	-0.383	0.175	-2.190	0.02857	*
hhstate24	-1.852	0.161	-11.507	< 2e-16	***
headage3	-0.428	0.195	-2.190	0.02857	*
headage4	-0.548	0.189	-2.902	0.00372	**
headage5	-0.413	0.189	-2.178	0.02943	*
headage6	-0.770	0.163	-4.729	2.32e-06	***
headocc3	0.665	0.096	6.964	3.77e-12	***
headed2	-0.810	0.132	-6.131	9.47e-10	***
headed3	-1.360	0.121	-11.214	< 2e-16	***
headed4	-2.102	0.155	-13.573	< 2e-16	***
hhtype2	2.619	0.113	23.150	< 2e-16	***

Table 6.28: The final logistic regression model (Model U) for *pipe*

6.7 Conclusion

The results set out in this chapter have empirically demonstrated the effectiveness of the three-step modelling procedure developed in previous chapters through simulation results based on a subset of data obtained in the Indian National Family Health Survey (INFHS). These results were obtained using the three-step procedure described in Chapter 3 and are based on simulation results under the NIS, MSI and RBS scenarios in those cases where the two competing models (unweighted and weighted model specifications) have different regressor sets. Simulation results to illustrate the theory of modelling bias under MSI set out in Chapter 4 are also provided. Finally, the application of the proposed modelling procedure for both linear and logistic regression modelling of the original INFHS data are presented as well. Five model search strategies were used to choose a final model from the two competing models (UMS and WMS). They are: $Vuong_{NW}$, $Vuong_{W}$, CV_{NW}, CV_{W} and *Voting System*. The findings referred to above can be summarised as follows:

1. Simulation results of linear regression show that the three strategies i.e. CV_{NW}, $V_{W_{small}}$ and VS_{large} are reasonable tools for model choice tool under the NIS

and MSI scenarios. In addition, all five strategies (i.e. $Vuong_{NW}$, $Vuong_{W}$, CV_{NW}, CV_{W} and *Voting System*) are acceptable for model choice under the RBS scenario.

2. Simulation results of logistic regression indicate that all five strategies (i.e. $Vuong_{NW}$, $Vuong_{W}$, CV_{NW}, CV_{W} and *Voting System*) are effective for model choice under NIS. Under the missing stratification variable (MSI) scenario, all strategies (i.e. $Vuong_{NW}$, $Vuong_{W}$, CV_{NW}, CV_{W} and *Voting System*) appear ineffective for model choice because both the unweighted and weighted model specifications are biased. Under the RBS scenario, the two CV-based model choice strategies tend to be more effective than those defined by Vuong's method.

3. The simulation results for modelling bias under MSI for both linear and logistic regression models are consist with the theoretical results obtained in Chapter 4.

4. The application to linear regression of *density* based on the INFHS data shows that all three types of independent variables are important in the regression equation. The particular variables included in the linear regression model are the state where the household is located, the age of the head of household, the sex of the head of household, the occupation of the head of household, the education level of the head of household, the religion of the head of household, whether the household is urban or rural, and number of toilets in household.

5. An application to logistic regression of *pipe* based on the INFHS data shows that almost all the independent variables considered are highly related to the probability of a household having piped water. These variables are the state where the household is located, the age of the head of household, the occupation of the head of household, the education level of the head of household, and whether the household is rural or urban.

Chapter 7

Statistical Tests for Weighting when Fitting Models

In previous chapters we discussed implementation of the Model Evaluation (a) pathway in Figure 3.1. This was for the situation where unweighted and weighted model search strategies (using backward elimination) led to different model specifications. It included a definition of the model identification procedures to be used, the model search strategies and relevant simulations. In this chapter we focus on implementation of the Model Evaluation (b) pathway in Figure 3.1. That is, we consider model fitting when the two backward elimination model search procedures (unweighted and weighted) lead to the same model specification, i.e. the two regressor sets are the same. In Section 7.1 we outline the model choice process when regressor sets of the two competing models are the same. Then in Section 7.2 we discuss two tests that can be used to choose between the unweighted and weighted model fits based on this common model specification, and in Section 7.3 we provide simulation results that illustrate the performances of these two tests. Throughout, we restrict ourselves to the linear regression case.

7.1 The Model Choice Procedure

The following steps outline the proposed procedure as shown in Figure 3.1 for testing whether sample weights significantly affect model fit. We proceed as follows:

1. Prepare the sample data.
2. Use standard (unweighted) backward elimination to specify Model U. In the INFHS example, this is the regression of log *density* on all available independent variables.
3. Use weighted backward elimination to specify Model W which is again the regression of log *density* on all available independent variables.
4. Use ordinary least squares to fit Model U to the sample data.
5. Use weighted least squares to fit Model W to the sample data.
6. When Model U and Model W have the same regressor set, use the two tests defined by the DuMouchel and Duncan test (DD) and the Pesaran test (PS) as detailed in the next section to decide between these two fits.

7.2 Statistical Tests for Use of Weights in Model Fitting

We consider two statistical tests that can be employed in the case of equivalent regressor sets in order to choose between an unweighted and a weighted fit to the sample data. The first test was proposed by DuMouchel and Duncan (1983) and is labelled 'DD' in what follows, while the second was proposed by Pesaran (1974) and is labelled 'PS' in what follows. Both the DD and the PS tests can be summarised as follows.

- **The DD test:** An augmented matrix of regressors is created by merging the regressor set with an additional set of regressors defined by multiplying this

regressor set by the sample weights. The unweighted regression of the response variable on this augmented regressor set is then obtained, and an Analysis of Variance F-test is used to test whether the interaction terms defined by these additional regressor terms represent a significant contribution to this regression fit.

- **The PS test:** Pesaran(1974) proposed a test for comparing two non-nested model fits. This test can be directly applied in this situation as follows. Two model fits defined by 'unweighted fit' and 'weighted fit' are constructed. The unweighted fit is the unweighted regression of the response variable on the regressor set defined by the backward elimination model identification process, and the weighted fit is the unweighted regression of the response variable on a new regressor set created by multiplying this regressor set by the sample weights. The test statistic proposed by Pesaran (1974) is then used to decide which of these two model fits is significantly better.

Details of these two tests are set out in the next two subsections.

7.2.1 The DuMouchel and Duncan Test (DD)

To illustrate the DuMouchel and Duncan test, suppose that $\mathbf{y}$ is the vector of values of a response variable, $\mathbf{X}$ is a matrix of values of a set of independent variables, and $\mathbf{W}$ is a diagonal matrix of sample weights. The DD test then uses analysis of variance test for the effect of sample weights by fitting a regression model of the form

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{WX}\gamma + \varepsilon$$

where β and γ are unknown regression parameters, and ε is an error vector.
That is, testing whether sample weights are influential in the model fit is equivalent to testing whether $\gamma = 0$ using the F test. If the F test indicates that the hypothesis $\gamma = 0$ cannot rejected, then the sample weights are considered as not influential in model fitting, i.e. the unweighted fit is preferable. On the other hand, if the F

test rejects the hypothesis that $\gamma = 0$, then the sample weights are considered as influential in model fitting. In this case the weighted model fit should be used.

7.2.2 The Pesaran Test (PS)

Pesaran (1974) used the approach of Cox (1961, 1962) to define a statistical test for distinguishing between two non-nested regression models. Using the same notation as above, we can illustrate Cox's approach as follows. Suppose that

$$\begin{aligned} \text{Model A} : \mathbf{y} &= \mathbf{X}\beta + \varepsilon \qquad ; \varepsilon \sim \mathbf{N}(\mathbf{0}, \sigma^{\mathbf{2}}\mathbf{I}), \\ \text{Model B} : \mathbf{y} &= \mathbf{WX}\delta + \xi \qquad ; \xi \sim \mathbf{N}(\mathbf{0}, \nu^{\mathbf{2}}\mathbf{I}) \end{aligned}$$

are two non-nested models; where β and δ are unknown parameters. Let $\theta = (\beta, \sigma^2)$ and $\gamma = (\delta, \nu^2)$ define the unknown parameters in model A and model B respectively, with corresponding ML estimators $\hat{\theta}$ and $\hat{\gamma}$. The values of the log-likelihood functions generated under the two models at these maximum likelihood estimates are then

$$\begin{aligned} l_A(\hat{\theta}|\mathbf{y}) &= -\frac{n}{2}log(2\pi\hat{\sigma}^2) - \frac{1}{2\hat{\sigma}^2}\hat{\varepsilon}'\hat{\varepsilon} \\ l_B(\hat{\gamma}|\mathbf{y}) &= -\frac{n}{2}log(2\pi\hat{\nu}^2) - \frac{1}{2\hat{\nu}^2}\hat{\xi}'\hat{\xi}. \end{aligned}$$

Given Model A is true, Cox proposed the test statistic

$$C_A = l_A(\hat{\theta}) - l_B(\hat{\gamma}) - E_A\{l_A(\hat{\theta}) - l_B(\hat{\gamma})\}. \tag{7.1}$$

The test statistic proposed by Pesaran (1974) is derived from equation (7.1).and is given by

$$PS = \frac{n}{2}log\left(\frac{\hat{\nu^2}}{\hat{\nu}_A^2}\right)$$

where ν_A^2 denotes the expected value of $\hat{\nu}^2$ under Model A (i.e. ignoring the weights). Pesaran (1974) shows that PS is then normally distributed with mean zero and variance given by $V(PS) = \frac{\hat{\nu}^2}{\hat{\nu}_A^2}\hat{\xi}_*'\hat{\xi}_*$ where $\hat{\xi}_* = \mathbf{M}\hat{\nu}_A^2$; $\mathbf{M} = \mathbf{I}_n - \mathbf{X}(\mathbf{X}'\mathbf{X})^{-\mathbf{1}}\mathbf{X}'$; and

$\mathbf{I}_n$ is the identity matrix of order n.

Given that Model A is true, the hypotheses for the test are

H_0: There is no effect of sample weights on the model fit,

H_a: There is an effect of sample weights on the model fit.

If the observed value of PS is significant at a test level of 0.05, i.e. when the absolute value of $PS/\sqrt{V(PS)}$ is greater than $z_{0.05}$, we conclude that there is an effect associated with the use of sample weights on the model fit and the null hypothesis H_0 is rejected. On the other hand, if this is not the case then we cannot reject H_0 and we conclude that there is insufficient evidence for using the sample weights in model fitting.

7.3 Simulation Results

To examine the effect of sample weights on model fit, data were generated in the same way as in Section 6.3 to obtain a finite population data set and a sample for each simulation.

A total of 10000 independent simulations were carried out using the same random number seed as used in Section 6.3. The `R` code for these simulations is set out in Appendix B. Here we focus on those simulations where the weighted and unweighted model backward elimination procedures resulted in identification of the same set of regressors. Table 7.1 shows the number of 'same regressor set' simulations classified by whether sample weights are identified as influential or not after testing for the effect of sample weighting using the PS and DD tests. Corresponding empirical values of relative biases, relative root mean squared errors and relative variances of estimators of the population regression model coefficients are set out in Tables 7.2-7.10 for the same three scenarios (NIS, MSI, RBS) considered earlier.

Table 7.1 shows that for the NIS and MSI situations, both the DD and PS tests lead to essentially the same results, with the unweighted model fits selected in almost all cases. However, in the RBS scenario these tests give completely different results.

Test Strategy	Are sample weights influential in model fit?	
	No	Yes
Scenario 1: Non-Informative Sampling (NIS)		
DD	481	-
PS	481	-
Scenario 2: Missing Stratification Information (MSI)		
DD	562	9
PS	571	-
Scenario 3: Response-Based Sampling (RBS)		
DD	-	2559
PS	2559	-

Table 7.1: Simulation results for whether sample weights are influential in model fit

Using the DD test invariably leads to use of the weighted fit, while use of the PS test invariably leads to use of the unweighted fit.

7.3.1 Simulation Results for Scenario 1: NIS

Since the DD and PS tests lead to exactly the same conclusions in this scenario, the results displayed in Tables 7.2-7.4 show that the relative bias, relative root mean squared error and relative variance values generated by both the DD and PS tests are exactly the same, and are identical to those obtained by the unweighted fit of the model identified by unweighted backward elimination (UMS). Furthermore, we see that the results for weighted fits generated by the model based on weighted backward elimination (WMS) are generally inferior to these three options.

Coefficient	True Parameter	Direct Choice		Statistical Test	
		UMS	WMS	DD	PS
Intercept	2	-0.091	-0.242	-0.091	-0.091
headage2	4	-0.001	0.007	-0.001	-0.001
headage3	4	0.000	0.009	0.000	0.000
headage4	4	-0.002	0.006	-0.002	-0.002
headage5	4	0.002	0.008	0.002	0.002
headage6	3	0.005	0.016	0.005	0.005
headsex2	1	-0.420	-0.289	-0.420	-0.420
headocc2	-1	0.278	0.217	0.278	0.278
headocc3	-2	0.075	-0.002	0.075	0.075
headocc4	2	0.151	0.143	0.151	0.151
headedu2	1	-0.090	0.232	-0.090	-0.090
headedu3	2	-0.027	0.090	-0.027	-0.027
headedu4	3	-0.002	0.064	-0.002	-0.002
region2	1	0.040	0.157	0.040	0.040
region3	1	0.061	0.117	0.061	0.061
hhtype2	1	0.050	0.287	0.050	0.050

Table 7.2: Simulation results for relative biases of estimators of linear regression coefficients under NIS and equivalent regressor sets

Coefficient	True Parameter	Direct Choice		Statistical Test	
		UMS	WMS	DD	PS
Intercept	2	0.573	0.608	0.573	0.573
headage2	4	0.104	0.106	0.104	0.104
headage3	4	0.104	0.105	0.104	0.104
headage4	4	0.114	0.118	0.114	0.114
headage5	4	0.115	0.118	0.115	0.115
headage6	3	0.175	0.179	0.175	0.175
headsex2	1	0.832	1.026	0.832	0.832
headocc2	-1	1.043	1.144	1.043	1.043
headocc3	-2	0.523	0.531	0.523	0.523
headocc4	2	0.695	0.718	0.695	0.695
headedu2	1	0.542	0.724	0.542	0.542
headedu3	2	0.204	0.291	0.204	0.204
headedu4	3	0.204	0.290	0.204	0.204
region2	1	0.196	0.308	0.196	0.196
region3	1	0.179	0.288	0.179	0.179
hhtype2	1	0.372	0.482	0.372	0.372

Table 7.3: Simulation results for relative variances of estimators of linear regression coefficients under NIS and equivalent regressor sets

Coefficient	True Parameter	Direct Choice		Statistical Test	
		UMS	WMS	DD	PS
Intercept	2	0.580	0.654	0.580	0.580
headage2	4	0.104	0.106	0.104	0.104
headage3	4	0.104	0.105	0.104	0.104
headage4	4	0.114	0.119	0.114	0.114
headage5	4	0.115	0.119	0.115	0.115
headage6	3	0.175	0.180	0.175	0.175
headsex2	1	0.932	1.066	0.932	0.932
headocc2	-1	1.079	1.164	1.079	1.079
headocc3	-2	0.528	0.531	0.528	0.528
headocc4	2	0.711	0.732	0.711	0.711
headedu2	1	0.550	0.761	0.550	0.550
headedu3	2	0.206	0.304	0.206	0.206
headedu4	3	0.204	0.297	0.204	0.204
region2	1	0.200	0.346	0.200	0.200
region3	1	0.189	0.311	0.189	0.189
hhtype2	1	0.375	0.561	0.375	0.375

Table 7.4: Simulation results for relative root mean squared errors of estimators of linear regression coefficients under NIS and equivalent regressor sets

7.3.2 Simulation Results for Scenario 2: MSI

Under MSI, the results set out in Table 7.5 indicate that all estimators are essentially the same as far as bias is concerned, with the intercept estimator most affected. The relative variance results shown in Table 7.6 indicate that the unweighted fits are less variable than the weighted fits and, since both DD and PS generally lead to unweighted fits, these perform relatively well. Finally, in Table 7.7 we see that the model fits based on the PS test are slightly better than those based on the DD test, again because unweighted fits (UMS) are generally better than weighted fits (WMS) in this scenario.

Coefficient	True Parameter	Direct Choice		Statistical Test	
		UMS	WMS	DD	PS
Intercept	2	1.771	1.778	1.774	1.771
headage2	4	-1.000	-1.000	-1.000	-1.000
headage3	4	-1.000	-1.000	-1.000	-1.000
headage4	4	-1.000	-1.000	-1.000	-1.000
headage5	4	-1.000	-1.000	-1.000	-1.000
headage6	3	-1.000	-1.000	-1.000	-1.000
headsex2	1	0.355	0.503	0.358	0.355
headocc2	-1	0.072	-0.204	0.064	0.072
headocc3	-2	-0.072	-0.179	-0.076	-0.072
headocc4	2	-0.592	-0.511	-0.591	-0.592
headedu2	1	-0.157	0.156	-0.152	-0.157
headedu3	2	-0.048	0.069	-0.047	-0.048
headedu4	3	0.015	0.025	0.016	0.015
region2	1	0.022	0.153	0.023	0.022
region3	1	0.214	0.138	0.212	0.214
hhtype2	1	-0.346	-0.010	-0.336	-0.346

Table 7.5: Simulation results for relative biases of estimators of linear regression coefficients under MSI and equivalent regressor sets

Coefficient	True Parameter	Direct Choice		Statistical Test	
		UMS	WMS	DD	PS
Intercept	2	0.525	0.651	0.526	0.525
headage2	4	0.000	0.000	0.000	0.000
headage3	4	0.000	0.000	0.000	0.000
headage4	4	0.000	0.000	0.000	0.000
headage5	4	0.000	0.000	0.000	0.000
headage6	3	0.000	0.000	0.000	0.000
headsex2	1	1.028	1.162	1.031	1.028
headocc2	-1	0.975	1.275	0.990	0.975
headocc3	-2	0.491	0.582	0.495	0.491
headocc4	2	0.584	0.701	0.585	0.584
headedu2	1	0.573	0.793	0.579	0.573
headedu3	2	0.220	0.307	0.220	0.220
headedu4	3	0.217	0.288	0.218	0.217
region2	1	0.208	0.314	0.217	0.208
region3	1	0.197	0.291	0.203	0.197
hhtype2	1	0.459	0.706	0.477	0.459

Table 7.6: Simulation results for relative variances of estimators of linear regression coefficients under MSI and equivalent regressor sets

Coefficient	True Parameter	Direct Choice		Statistical Test	
		UMS	WMS	DD	PS
Intercept	2	1.847	1.893	1.850	1.847
headage2	4	1.000	1.000	1.000	1.000
headage3	4	1.000	1.000	1.000	1.000
headage4	4	1.000	1.000	1.000	1.000
headage5	4	1.000	1.000	1.000	1.000
headage6	3	1.000	1.000	1.000	1.000
headsex2	1	1.087	1.266	1.091	1.087
headocc2	-1	0.978	1.291	0.992	0.978
headocc3	-2	0.497	0.609	0.500	0.497
headocc4	2	0.831	0.867	0.832	0.831
headedu2	1	0.594	0.808	0.598	0.594
headedu3	2	0.225	0.315	0.225	0.225
headedu4	3	0.217	0.289	0.218	0.217
region2	1	0.209	0.350	0.218	0.209
region3	1	0.291	0.322	0.293	0.291
hhtype2	1	0.575	0.706	0.584	0.575

Table 7.7: Simulation results for relative root mean squared errors of estimators of linear regression coefficients under MSI and equivalent-regressor sets

7.3.3 Simulation Results for Scenario 3: RBS

For the RBS situation, the results set out in Table 7.8 show again that all estimators have similar bias behaviour. More interestingly, since PS always rejects weighted fits, while DD always rejects unweighted fits in this scenario, the real comparison is between UMS and WMS. Here the results displayed in Table 7.9-7.10 indicate that the weighted fits generally lead to estimators that are less efficient that estimators based on unweighted fits in this scenario.

Coefficient	True Parameter	Direct Choice		Statistical Test	
		UMS	WMS	DD	PS
Intercept	2	0.116	-0.492	-0.492	0.116
headage2	4	-0.007	-0.014	-0.014	-0.007
headage3	4	-0.007	-0.027	-0.027	-0.007
headage4	4	-0.009	-0.020	-0.020	-0.009
headage5	4	-0.003	-0.023	-0.023	-0.003
headage6	3	-0.007	-0.038	-0.038	-0.007
headsex2	1	-0.051	0.046	0.046	-0.051
headocc2	-1	0.074	0.020	0.020	0.074
headocc3	-2	0.034	0.005	0.005	0.034
headocc4	2	0.029	0.008	0.008	0.029
headedu2	1	-0.051	0.030	0.030	-0.051
headedu3	2	0.005	0.015	0.015	0.005
headedu4	3	0.009	0.037	0.037	0.009
region2	1	-0.371	0.159	0.159	-0.371
region3	1	-0.526	-0.503	-0.503	-0.526
hhtype2	1	0.008	0.050	0.050	0.008

Table 7.8: Simulation results for relative biases of estimators of linear regression coefficients under RBS and equivalent regressor sets

Coefficient	True Parameter	Direct Choice		Statistical Test	
		UMS	WMS	DD	PS
Intercept	2	0.454	0.520	0.520	0.454
headage2	4	0.138	0.177	0.177	0.138
headage3	4	0.130	0.167	0.167	0.130
headage4	4	0.131	0.168	0.168	0.131
headage5	4	0.131	0.167	0.167	0.131
headage6	3	0.163	0.208	0.208	0.163
headsex2	1	0.407	0.455	0.455	0.407
headocc2	-1	0.577	0.614	0.614	0.577
headocc3	-2	0.264	0.285	0.285	0.264
headocc4	2	0.266	0.292	0.292	0.266
headedu2	1	0.213	0.258	0.258	0.213
headedu3	2	0.115	0.142	0.142	0.115
headedu4	3	0.116	0.140	0.140	0.116
region2	1	0.314	0.246	0.246	0.314
region3	1	0.207	0.210	0.210	0.207
hhtype2	1	0.181	0.230	0.230	0.181

Table 7.9: Simulation results for relative variances of estimators of linear regression coefficients under RBS and equivalent regressor sets

Coefficient	True Parameter	Direct Choice		Statistical Test	
		UMS	WMS	DD	PS
Intercept	2	0.468	0.716	0.716	0.468
headage2	4	0.138	0.177	0.177	0.138
headage3	4	0.130	0.169	0.169	0.130
headage4	4	0.131	0.169	0.169	0.131
headage5	4	0.131	0.169	0.169	0.131
headage6	3	0.163	0.211	0.211	0.163
headsex2	1	0.410	0.458	0.458	0.410
headocc2	-1	0.582	0.614	0.614	0.582
headocc3	-2	0.267	0.285	0.285	0.267
headocc4	2	0.268	0.292	0.292	0.268
headedu2	1	0.219	0.260	0.260	0.219
headedu3	2	0.115	0.143	0.143	0.115
headedu4	3	0.117	0.145	0.145	0.117
region2	1	0.486	0.293	0.293	0.486
region3	1	0.565	0.545	0.545	0.565
hhtype2	1	0.181	0.235	0.235	0.181

Table 7.10: Simulation results for relative root mean squared errors of linear regression coefficients under RBS and equivalent regressor sets

Chapter 8

Conclusions and Future Research

8.1 Summary

Complex survey data are data obtained using a complex sampling design. These data frequently include sample weights and identifier variables (e.g. stratum identifiers, cluster identifiers) that characterise the complex structures of the target finite population from which the sample was drawn. More importantly, these data are widely used by government agencies which are often concerned with making decisions about the target population. In making a decision based on data obtained from such a survey, reliable ways of inferring the relationships that characterise the target population are required for reliable and accurate decisions.

In many cases the target of inference is the vector of coefficients corresponding to the unknown parameter of the regression model that characterises the relationship between a variable of interest in the target population and a set of explanatory independent variables. Fitting such a regression model using complex sample survey data needs consideration of whether the sampling method used is non-informative or informative. In other words, analysis based on an assumption that the sampling design used is non-informative (i.e. the sample distribution of the target response variable does not differ from the corresponding population distribution) can lead to misleading results and erroneous inferences if the sampling design is actually informative. In this thesis, we develop a framework for analysis in this

situation given a stratified sampling method and investigate three scenarios based on it: Non-Informative Sampling (NIS), Missing Stratification Information (MSI) and Response-Based Sampling (RBS) as defined in Chapter 1.

The principal aim of this thesis is to develop and evaluate strategies for population modelling using complex sample survey data. In particular, we focus on the consequences for regression modelling of sample data when the population regression model is unknown and when one of the three scenarios above hold. To achieve this aim, a procedure based on a regression modelling strategy given complex survey data has been developed for simple stratified sampling. The procedure provides an answer the two questions posed in Chapter 1. In Chapter 3, we focus on development of this procedure, as set out in Figure 3.1. We also investigate modelling bias under one of the three scenarios (MSI) in Chapter 4.

In order to apply the procedure proposed in Chapter 3 we require appropriate model search tools. In Chapter 5 we describe two approaches (the likelihood-based approach and the prediction-based approach) that can be used for this purpose. In particular, we focus on the likelihood approach based on the test suggested by Vuong (1989), and on the version of the predictive approach that uses cross-validation. This naturally leads to four modelling strategies defined by the combination of the two different approaches to model identification (Vuong test vs. cross-validation) and two model search strategies (unweighted vs. weighted backward elimination). This allows us to identify a final model when given two competing models suggested by the model search methods. Simulations results based on a real survey data set are then presented in Chapter 6, where we investigate the empirical performances of these four modelling strategies under the NIS, MSI and RBS scenarios, and for linear and logistic regression. The computer software `R` was used throughout the simulation study. Our results can be summarised as follows:

1. The suggested procedure diagrammatically set out in Figure 3.1 is likely to recover the true population regression model for all scenarios.

2. Under NIS (that is, where the sample distribution of the target response vari-

able does not differ from the corresponding population distribution of the target response variable), it can be conclude that

- **Linear Regression:** The three-step modelling procedure and five model search strategies ($Vuong_{NW}$, $Vuong_{W}$, CV_{NW}, $CV_{W}$and *Voting System*) work reasonably well together to recover the true population model in general. Overall, the unweighted model specification needs to be adopted.
- **Logistic Regression:** The proposed procedure and the five model search strategies work very well together to recover the true population model. The voting-system strategy provides the best results. However, the results obtained from the other model-choice strategies are all an improvement on those obtained from direct choice of a weighted model specification. Furthermore, the two CV-based strategies provide better results than the two strategies obtained from Vuong's approach. Here as well the unweighted model specification needs to be adopted.

3. Under MSI (that is, where the sample regression model and the population regression model differ because stratifying regressors are omitted from the sample model), we conclude that

   - **Linear Regression:** Although the proposed procedure and the five model search strategies fail to recover the true population model (i.e. the model includes regression parameters of the stratifying variables), in general they work reasonably well together when the targets of inference are the parameters of the model that averages over the stratifying variables. Again, the unweighted model specification is superior.
   - **Logistic Regression:** These results are similar to those for linear regression. We also note that the proposed procedure and both the Vuong-based model search strategies are capable of recovering the incomplete sample data model (i.e. the one that averages over the missing stratifying variables).

Note that when the weighted and unweighted model specifications are considered as *equivalent* in the case of the Vuong strategies, then the unweighed model specification should be adopted for both the NIS and MSI scenarios.

4. Under RBS (that is, where the sample and population distributions of the target response variable differ because the sampling design stratifies on the target response variable), sample weights are theoretically necessary if one intends to recover the target population model. It is therefore natural that the weighted model specification should be adopted for this situation. From our simulation results, we conclude that

    - **Linear Regression:** The proposed procedure and five model search strategies perform reasonably well in terms of recovering the true population model. Note however that this is only the case when re-scaled sample weights are used. As a result, the weighted model specification needs to be adopted in this case.
    - **Logistic Regression:** The proposed procedure and the five model search strategies work well in terms of recovering the true population model from the point of view of relative bias (RB). As far as RV and RRMSE are concerned, however, the proposed procedure based on the two CV model search strategies represents the best option. Here again the weighted model specification is identified as preferable.

In order to assess the practical usefulness of the two-step modelling procedure developed in this thesis, we then applied it to the original INFHS survey data in Chapter 6. Because all design information (i.e. sample weights and stratified variables) are available to be employed in the analysis, we consider this case as one that corresponds to the NIS scenario. Our results indicate that the four modelling strategies could detect an appropriate specification for the underlying population model for both linear and logistic regression.

Finally, in Chapter 7 we used the DuMouchel and Duncan test and the Pesaran

test to investigate in cases where the model specification is fixed, whether sample weights affect model fit under NIS, MSI and RBS for the linear regression model. Again, we use the `R` software package in our investigation. Here our simulation results show that sample weights should be ignored in fitting a model when the two competing models have the same set of regressors. That is, sample weights do not a matter for all three scenarios.

8.2 Future Research

This thesis represents a starting point in terms of building a framework for the development of strategies for population modelling using complex sample survey data. This implies that there are a lot of issues that need to be further developed and investigated in future research.

1. The modelling procedure set out in this thesis requires an extension to other sampling designs such as multi-stage sampling, including cluster sampling, which are more complex than the simple stratified sampling considered here. This means that the corresponding modelling strategies described in this thesis also need to be modified and developed further to allow for clustered population data.

2. In the case of MSI, the simulation results displayed in Chapter 6 show that the models suggested by the proposed modelling procedure are biased. This is because the target of inference is misspecified due to the omitted stratifying variables. However, this bias is decreased when the target model is the population model that averages over the missing stratifying variables. Some theory for this result is set out in Chapter 4, but this is only for a simple case of both linear and logistic regression. In reality, multivariate regression models are widely used. Therefore, an extension of Chapter 4 to the case of multivariate regression is necessary.

3. This thesis considers two model search strategies (Vuong and cross-validation). Although both seemed reasonable in their ability to recover the true population model, other, more efficient, methods should be developed. This is particularly necessary for the MSI scenario.

4. In this thesis we use backward elimination to specify candidate models for the target population model. Other model selection methods may be useful and should be investigated as the first step in the suggested modelling procedure.

5. We have restricted our investigation to models without interaction terms. In reality most model searches include interaction terms. As such, there is a need to extend our theoretical and methodological approach for this situation.

6. Finally, we note that since most analyses of complex survey data are carried out by non-statisticians, there is a need to integrate the modelling ideas set out in this thesis into standard statistical analysis software, including perhaps the creation of a user friendly statistical package that incorporates these ideas.

Appendix A

Proof of Theorem 5.3

Suppose that the existence of the matrices, defined in Vuong (1989)

$$A_f(\theta) \equiv E^0 \left[\frac{\partial^2 log\ f(Y_i|X_i;\theta)}{\partial\theta\partial\theta^T} \right]$$

and

$$B_f(\theta) \equiv E^0 \left[\frac{\partial log\ f(Y_i|X_i;\theta)}{\partial\theta} . \frac{\partial log\ f(Y_i|X_i;\theta)}{\partial\theta^T} \right].$$

Similar matrices $A_g(\gamma)$ and $B_g(\gamma)$ are defined for the model G_γ.

To prove Theorem 5.3, it requires the following lemmas as defined in Vuong (1989).

Lemma A.1 *Given the existence of the matrices defined above,*

$$\sqrt{n} \begin{bmatrix} \hat{\theta} - \theta \\ \hat{\gamma} - \gamma \end{bmatrix} \xrightarrow{d} N(0, \Sigma), \quad where\ \ \Sigma = \begin{bmatrix} A_f^{-1} B_f A_f^{-1} & A_f^{-1} B_{fg} A_g^{-1} \\ A_g^{-1} B_{gf} A_f^{-1} & A_g^{-1} B_g A_g^{-1} \end{bmatrix}.$$

Lemma A.2 *Given the existence of the matrices B_f defined above,*

$$\sqrt{n} \begin{bmatrix} \frac{1}{n} l'(\theta) \\ \frac{1}{n} l'(\gamma) \end{bmatrix} \xrightarrow{d} N \left(0, \begin{bmatrix} B_f & B_{fg} \\ B_{gf} & B_g \end{bmatrix} \right),$$

where $l'(\theta) = \sum_s^n \frac{\partial log f(Y_i|X_i;\theta)}{\partial\theta}$ *and* $l'(\gamma) = \sum_s^n \frac{\partial log g(Y_i|X_i;\gamma)}{\partial\gamma}$.

Proof of Lemma A.1-A.2, see Vuong (1989). In addition, it also requires the following theorem as expressed in Serfling (1980).

Theorem A.1 *Given a sequence X_n and a condition that $X_n \xrightarrow{rth} 0$, suppose that the smoothed sequence*

$$X_n^* = \frac{\sum_{i=1}^n w_i X_i}{\sum_{i=1}^n w_i}, \quad w_i \geqslant 0, \sum_{i=1}^{\infty} w_i = \infty.$$

A sufficient condition for X_n^* to converge to zero with proability 1 (wp1) is that

$$\sum_{n=1}^{\infty} \frac{E|X_n|^r}{n} < \infty,$$

where $E|X_n|^r$ represents the rth mean.

Proof of Theorem A.1, see Serfling (1980).

We recall that

$$\tilde{l}(\hat{\theta}) = \sum_{i=1}^{n} w_i log\ f(Y_i|X_i;\hat{\theta})$$

and

$$\tilde{l}(\hat{\gamma}) = \sum_{i=1}^{n} w_i log\ g(Y_i|X_i;\hat{\gamma}).$$

The Taylor expansion of $\underset{\sim}{l}(\hat{\theta})$ around $\hat{\theta} = \theta$ is then

$$\tilde{l}(\hat{\theta}) \approx \tilde{l}(\theta) + (\hat{\theta} - \theta)\tilde{l}'(\theta) + \frac{1}{2}(\hat{\theta} - \theta)^T \tilde{l}''(\theta)(\hat{\theta} - \theta).$$

Similar expansion of $\underset{\sim}{l}(\hat{\gamma})$ around $\hat{\gamma} = \gamma$ is

$$\tilde{l}(\hat{\gamma}) \approx \tilde{l}(\gamma) + (\hat{\gamma} - \gamma)\tilde{l}'(\theta) + \frac{1}{2}(\hat{\gamma} - \gamma)^T \tilde{l}''(\gamma)(\hat{\gamma} - \gamma).$$

Since $\tilde{l}(\hat{\theta}, \hat{\gamma}) = \tilde{l}(\hat{\theta}) - \tilde{l}(\hat{\gamma})$ and $\tilde{l}(\theta, \gamma) = \tilde{l}(\theta) - \tilde{l}(\gamma)$, we then obtain

$$\tilde{l}(\hat{\theta}, \hat{\gamma}) \approx \tilde{l}(\theta, \gamma) + (\hat{\theta} - \theta)\tilde{l}'(\theta) - (\hat{\gamma} - \gamma)\tilde{l}'(\gamma) + \frac{1}{2}(\hat{\theta} - \theta)^T \tilde{l}''(\theta)(\hat{\theta} - \theta) - \frac{1}{2}(\hat{\gamma} - \gamma)^T \tilde{l}''(\gamma)(\hat{\gamma} - \gamma). \tag{A.1}$$

From Equation (A.1), in what follows, we first begin with showing that $\tilde{l}'(\theta)$ and $\tilde{l}'(\gamma) \xrightarrow{\text{wp1}} 0$. Given Lemma A.2, we have

$$E\left[\frac{\partial}{\partial\theta} log f(Y|X;\theta)\right] = 0 \tag{A.2}$$

and

$$E\left[\frac{\partial}{\partial\theta} log g(Y|X;\gamma)\right] = 0. \tag{A.3}$$

As a consequence, given Theorem A.1, we obtain

$$(\sum_s w_i)^{-1}\tilde{l}'(\theta) \xrightarrow{\text{wp1}} 0 \text{ and } (\sum_s w_i)^{-1}\tilde{l}'(\gamma) \xrightarrow{\text{wp1}} 0.$$

Therefore,

$$\tilde{l}'(\theta) \xrightarrow{\text{wp1}} 0 \text{ and } \tilde{l}'(\gamma) \xrightarrow{\text{wp1}} 0. \tag{A.4}$$

It is similar for an expression of $\tilde{l}''(\theta)$ and $\tilde{l}''(\gamma)$ in what follows. As Equation (A.2)-(A.3), suppose that the response variable Y is a continuous variable, we have

$$\int \frac{\partial}{\partial\theta} f(Y|X;\theta)dy = 0$$

and

$$\int \frac{\partial}{\partial\gamma} g(Y|X;\gamma)dy = 0,$$

given a condition where all the derivatives exist for each $\theta \in \Theta$.

As a consequence, we also have

$$\int \frac{\partial^2}{\partial\theta^2} f(Y|X;\theta)dy = 0$$

and

$$\int \frac{\partial^2}{\partial\gamma^2} g(Y|X;\gamma)dy = 0,$$

Therefore,

$$\begin{aligned} E\left[\frac{\partial^2}{\partial\theta^2} log\ f(Y|X;\theta)\right] &= \int\left[\frac{1}{f(Y|X;\theta)}\frac{\partial^2}{\partial\theta^2}f(Y|X;\theta) - \left(\frac{1}{f(Y|X;\theta)}\frac{\partial}{\partial\theta}f(Y|X;\theta)\right)^2\right] \\ & f(Y|X;\theta)dy \\ &= -\int\left(\frac{\partial}{\partial\theta} log\ f(Y|X;\theta)\right)^2 f(Y|X;\theta)dy \\ &= -E\left[\left(\frac{\partial}{\partial\theta} log\ f(Y|X;\theta)\right)^2\right] \\ &= -nA_f(\theta). \end{aligned} \tag{A.5}$$

Similar expression for $E\left[\frac{\partial^2}{\partial\gamma^2} log\ g(Y|X;\gamma)\right]$ is that

$$\begin{aligned} E\left[\frac{\partial^2}{\partial\gamma^2} log\ g(Y|X;\gamma)\right] &= \int\left[\frac{1}{g(Y|X;\gamma)}\frac{\partial^2}{\partial\gamma^2}f(Y|X;\gamma) - \left(\frac{1}{g(Y|X;\gamma)}\frac{\partial}{\partial\gamma}g(Y|X;\gamma)\right)^2\right] \\ & g(Y|X;\gamma)dy \\ &= -\int\left(\frac{\partial}{\partial\gamma} log\ g(Y|X;\gamma)\right)^2 g(Y|X;\gamma)dy \\ &= -E\left[\left(\frac{\partial}{\partial\gamma} log\ g(Y|X;\gamma)\right)^2\right] \\ &= -nA_g(\gamma). \end{aligned} \tag{A.6}$$

Given Theorem A.1, from Equation (A.5)-(A.6), we then obtain

$$(\sum_s w_i)^{-1}\tilde{l}''(\theta) \xrightarrow{\text{wp1}} -nA_f(\theta) \text{ and } (\sum_s w_i)^{-1}\tilde{l}''(\gamma) \xrightarrow{\text{wp1}} -nA_g(\gamma).$$

Therefore,

$$\tilde{l}''(\theta) \xrightarrow{\text{wp1}} -n(\sum_s w_i)A_f(\theta) \text{ and } \tilde{l}''(\gamma) \xrightarrow{\text{wp1}} -n(\sum_s w_i)A_g(\gamma). \tag{A.7}$$

From Equation (A.1) given the results set out in Equation (A.4) and Equation (A.7), we obtain

$$\tilde{l}(\hat{\theta},\hat{\gamma}) \approx \tilde{l}(\theta,\gamma) - n\frac{\sum_s w_i}{2}(\hat{\theta}-\theta)^T A_f(\theta)(\hat{\theta}-\theta) + n\frac{\sum_s w_i}{2}(\hat{\gamma}-\gamma)^T A_g(\gamma)(\hat{\gamma}-\gamma). \tag{A.8}$$

By multiplying $\frac{1}{\sum_s w_i}$ on both sides Equation (A.8), we then obtain

$$\frac{1}{\sum_s w_i}\tilde{l}(\hat{\theta},\hat{\gamma}) \approx \frac{1}{\sum_s w_i}\tilde{l}(\theta,\gamma) - \frac{n}{2}(\hat{\theta}-\theta)^T A_f(\theta)(\hat{\theta}-\theta) + \frac{n}{2}(\hat{\gamma}-\gamma)^T A_g(\gamma)(\hat{\gamma}-\gamma). \quad \text{(A.9)}$$

As the weighted Vuong statistic $V_W = \frac{1}{\sum_s w_i}\tilde{l}(\hat{\theta},\hat{\gamma})$, given Lemma A.1 we see that $\sqrt{n}(\hat{\theta}-\theta)$ and $\sqrt{n}(\hat{\gamma}-\gamma)$ are $O_p(1)$. From (A.9), we thus obtain

$$\sqrt{n}\left[V_W - E^0\left[log\frac{f(Y_i|X_i;\theta_*)}{g(Y_i|X_i;\gamma_*)}\right]\right] \approx \sqrt{n}\left[\frac{1}{\sum_s w_i}\tilde{l}(\theta,\gamma) - E^0\left[log\frac{f(Y_i|X_i;\theta_*)}{g(Y_i|X_i;\gamma_*)}\right]\right] \quad \text{(A.10)}$$

Since $\frac{1}{n}l(\theta,\gamma)$ in Vuong (1989) has normal distributed with mean $\mu_* = E^0\left[log\frac{f(Y_i|X_i;\theta_*)}{g(Y_i|X_i;\gamma_*)}\right]$ and variance v_*^2 defined by Equation (5.4), as a consequence, $\frac{1}{\sum_s w_i}\tilde{l}(\theta,\gamma)$ also has normal distributed with the following mean:

$$\begin{aligned}
E^0\left[\frac{1}{\sum_s w_i}\tilde{l}(\theta,\gamma)\right] &= E^0\left[\frac{1}{\sum_s w_i}\sum_s w_i log\frac{f(Y_i|X_i;\theta_*)}{g(Y_i|X_i;\gamma_*)}\right] \\
&= \frac{1}{\sum_s w_i}\sum_s w_i E^0\left[log\frac{f(Y_i|X_i;\theta_*)}{g(Y_i|X_i;\gamma_*)}\right] \\
&= \frac{1}{\sum_s w_i}\sum_s w_i\mu_* \\
&= \mu_*
\end{aligned}$$

and the following variance:

$$\begin{aligned}
Var^0\left[\frac{1}{\sum_s w_i}\tilde{l}(\theta,\gamma)\right] &= Var^0\left[\frac{1}{\sum_s w_i}\sum_s w_i log\frac{f(Y_i|X_i;\theta_*)}{g(Y_i|X_i;\gamma_*)}\right] \\
&= \frac{1}{(\sum_s w_i)^2}\sum_s w_i^2 Var^0\left[log\frac{f(Y_i|X_i;\theta_*)}{g(Y_i|X_i;\gamma_*)}\right] \\
&= \frac{\sum_s w_i^2}{(\sum_s w_i)^2}v_*^2.
\end{aligned}$$

As a result, from Equation (A.10) given an assumptiong that the variance is finite and the Cauchy-Schwarz inequality, the term in the right-hand side converges in distribution to $N(0, \frac{\sum_s w_i^2}{(\sum_s w_i)^2}v_*^2)$ by the multivariate Central Limit Theorem (Serfling, 1980). The proof of Theorem 5.3 is then complete.

Appendix B

Simulation Study Code

The following functions are used for simulations of both linear and logistic regression under three scenarios: NIS, MSI and RBS.

```
# Storing regression estimators
coefStore   <- function(numCf,coef,colNames){
    tmp     <- c()
    tmp     <- rep(0,numCf)
    names(tmp) <- colNames
    for (i in 1:length(coef)){
        tmp[names(coef[i])] <- coef[i]
    }
    return(tmp)
}

# Relative bias
RB <- function(coefPop,numCf,coefMatrix){
    nr <- nrow(coefMatrix)
    RB <- c()
    for(i in 1:numCf){
        sumEstSq <- c(rep(0,numCf))
        for(j in 1:nr){
            sumEstSq[i] <- sumEstSq[i]+(coefMatrix[j,i]-
                           coefPop[i])
        }
        RB[i] <- (1/(nr*abs(coefPop[i,1])))*sumEstSq[i]
    }
    return(RB)
}

# Relative variance
RV <- function(coefPop,numCf,coefMatrix){
```

```
    nr <- nrow(coefMatrix)
    thetabar <- apply(coefMatrix,2,mean)
    rv <- c()
    for(i in 1:numCf){
        sumEstSq <- c(rep(0,numCf))
        for(j in 1:nr){
            sumEstSq[i] <- sumEstSq[i] +
                           (coefMatrix[j,i]-thetabar[i])^2
        }
        rv[i] <- (1/abs(coefPop[i,1]))*((1/nr)*sumEstSq[i])^0.5
    }
    return(rv)
}

# Relative root mean squared error
RRMSE <- function(coefPop,numCf,coefMatrix){
    nr <- nrow(coefMatrix)
    rrmse <- c()
    for(i in 1:numCf){
        sumEstSq <- c(rep(0,numCf))
        for(j in 1:nr){
            sumEstSq[i] <- sumEstSq[i]+(coefMatrix[j,i]-
                           coefPop[i])^2
        }
        rrmse[i] <- (1/abs(coefPop[i,1]))*((1/nr)*
                    sumEstSq[i])^0.5
    }
    return(rrmse)
}

# Vuong's choice
Vchoice <- function(vt){
    z05 <- 1.96
    if(vt>z05) {ch<-1}
    if(vt<(-z05)){ch<-2}
    if((vt>=(-z05))&&(vt<=z05)){ch<-3}
    return(ch)
}

# Final model choice: choosing a smaller model
# if the Vuong's test indicated that two model
# specifications have equivalent-regressor sets.
smVchoice <- function(v,nCF1,nCF2){
    if(v==1){M<-1}
    if(v==2){M<-2}
    if(v==3){M<-ifelse(nCF1<nCF2,1,2)}
    return(M)
}
```

```
# Voting system
voting <- function(s,loops){
    if((s==4)|(s==5)){r<-1}
    if((s==7)|(s==8)){r<-2}
    if(s==6){r<-3}
    return(r)
}

# For voting system, storing coefficients of
# the final model under a condition where
# a smaller model is chosen when equal voting.
finalCoefSM <- function(m,uCf,wCf,numCf){
    m <- as.vector(m)
    C <- matrix(0,length(m),numCf)
    l <- length(m)
    nu <- nrow(uCf)
    nw <- nrow(wCf)
    for(i in 1:l){
        if(m[i]==1){C[i,]<-uCf[i,]}else{
        if(m[i]==2){C[i,]<-wCf[i,]}else{
        if((m[i]==3)&&(nu<nw)){C[i,]<-uCf[i,]}
        else{
            C[i,]<-wCf[i,]
        }}}
    }
    return(C)
}

# For voting system, storing coefficients
# of the final model under a condition where
# a larger model is chosen when equal voting.
finalCoefLG <- function(m,uCf,wCf,numCf){
    m <- as.vector(m)
    C <- matrix(0,length(m),numCf)
    l <- length(m)
    nu <- nrow(uCf)
    nw <- nrow(wCf)
    for(i in 1:l){
        if(m[i]==1){C[i,]<-uCf[i,]}else{
        if(m[i]==2){C[i,]<-wCf[i,]}else{
        if((m[i]==3)&&(nu<nw)){C[i,]<-wCf[i,]}
        else{
            C[i,]<-uCf[i,]
        }}}
    }
    return(C)
}
```

```
# Storing coefficients for the choice of CV
cvCoef <- function(finalM,numCf,cvtest,uCf,wCf){
    nr <- nrow(finalM)
    cvcoef   <- matrix(0,nr,numCf)
    for(i in 1:nr){
        if(cvtest[i]==1) {cvcoef[i,]<-uCf[i,]}
        if(cvtest[i]==2) {cvcoef[i,]<-wCf[i,]}
    }
    return(cvcoef)
}

# Dividing a sample into five folds
getFolds <- function(nFolds,sample,sSize){
    nStrata <- ncol(sSize)
    count <- rep(0,nStrata)
    currentFold <- rep(1,nStrata)
    cPos <- rep(1,nFolds)
    nSample <- nrow(sample)
    f <- matrix(data=0,nrow=nFolds,
        ncol=(nSample/nFolds))
    s <- matrix(data=0,nrow=nFolds,
        ncol=(nSample/nFolds))
    idStrata <- matrix(data=0,nrow=nSample,
                ncol=2)
    idStrata[,1] <- rownames(sample)
    idStrata[,2] <- sample$strataY
    for (i in 1:nSample){
        str <- as.numeric(idStrata[i,2])
        id <- idStrata[i,1]
        cF <- currentFold[str]
        if (count[str] < sSize[cF,str]){
            #Add id to fold
            f[cF,cPos[cF]] <- id
            s[cF,cPos[cF]] <- str
            cPos[cF] <- cPos[cF] + 1
            #Update count
            count[str] <- count[str] + 1
        }else{
            #Change fold and add id to new fold
            currentFold[str]<-currentFold[str]+1
            cF <- currentFold[str]
            f[cF,cPos[cF]] <- id
            s[cF,cPos[cF]] <- str
            cPos[cF] <- cPos[cF] + 1
            #Update count
            count[str] <- 1
        }
```

```
    }
    return(f)
}

# The testing set for CV
fRest <- function(nFolds,sf,f){
    tmp <- NULL
    for (i in 1:nFolds){
        if(i != sf){
            tmp <- c(tmp,f[i,])
        }
    }
    return(tmp)
}

# Evaluation of the results
printResults <- function(cfPop,noCfPop,finalM,
                uCf,wCf,uSe,wSe,noCfs)
{
    # 1 Vuong
    # NWtest
    endVuCoefSmall <- finalCoefSM(finalM[,6],
                        uCf,wCf,noCfPop)
    # Wtest
    endVwCoefSmall <- finalCoefSM(finalM[,7],
                        uCf,wCf,noCfPop)

    # 2 CV
    endCVcoefU <- finalCoefSM(finalM[,8],
                    uCf,wCf,noCfPop)
    endCVcoefW <- finalCoefSM(finalM[,9],
                    uCf,wCf,noCfPop)

    # 3 Voting system
    endVoteCoefSmall <- finalCoefSM(finalM[,10],
                        uCf,wCf,noCfPop)
    endVoteCoefLong  <- finalCoefLG(finalM[,10],
                        uCf,wCf,noCfPop)

    # Evaluating final coefficients of each method
    # Evaluating of direct choice
    uRB <- RB(cfPop,noCfPop,uCf)
    wRB <- RB(cfPop,noCfPop,wCf)
    uRV <- RV(cfPop,noCfPop,uCf)
    wRV <- RV(cfPop,noCfPop,wCf)
    uRRMSE <- RRMSE(cfPop,noCfPop,uCf)
    wRRMSE <- RRMSE(cfPop,noCfPop,wCf)
```

```
# 1 Vuong
# NWtest
RBsmallVu  <- RB(cfPop,noCfPop,
                 endVuCoefSmall)
RVsmallVu  <- RV(cfPop,noCfPop,
                 endVuCoefSmall)
RRMSEsmallVu<- RRMSE(cfPop,noCfPop,
                 endVuCoefSmall)

# Wtest
RBsmallVw  <- RB(cfPop,noCfPop,
                 endVwCoefSmall)
RVsmallVw  <- RV(cfPop,noCfPop,
                 endVwCoefSmall)
RRMSEsmallVw<- RRMSE(cfPop,noCfPop,
                 endVwCoefSmall)

# 2 CV
RBcvU <- RB(cfPop,noCfPop,
                 endCVcoefU)
RBcvW <- RB(cfPop,noCfPop,
                 endCVcoefW)
RVcvU <- RV(cfPop,noCfPop,
                 endCVcoefU)
RVcvW <- RV(cfPop,noCfPop,
                 endCVcoefW)
RRMSEcvU <- RRMSE(cfPop,noCfPop,
                 endCVcoefU)
RRMSEcvW <- RRMSE(cfPop,noCfPop,
                 endCVcoefW)

# 3 Voting system
RBsmallVote  <- RB(cfPop,noCfPop,
                     endVoteCoefSmall)
RVsmallVote  <- RV(cfPop,noCfPop,
                     endVoteCoefSmall)
RRMSEsmallVote <- RRMSE(cfPop,noCfPop,
                     endVoteCoefSmall)
RBlongVote  <- RB(cfPop,noCfPop,
                     endVoteCoefLong)
RVlongVote  <- RV(cfPop,noCfPop,
                     endVoteCoefLong)
RRMSElongVote <- RRMSE(cfPop,noCfPop,
                     endVoteCoefLong)

# Results
allRBresults    <- round(cbind(uRB,wRB,
                     RBsmallVu,RBcvU,
```

```
                        RBsmallVw,RBcvW,
                        RBsmallVote,
                        RBlongVote),3)
    allRVresults <- round(cbind(uRV,wRV,
                        RVsmallVu,RVcvU,
                        RVsmallVw,
                        RVcvW,RVsmallVote,
                        RVlongVote),3)
    allRRMSEresults <- round(cbind(uRRMSE,
                        wRRMSE,RRMSEsmallVu,
                        RRMSEcvU,RRMSEsmallVw,
                        RRMSEcvW,RRMSEsmallVote,
                        RRMSElongVote),3)

    minRB  <- apply(abs(allRBresults[,-c(1:2)]),
                1,"min")
    minRV  <- apply(abs(allRVresults[,-c(1:2)]),
                1,"min")
    minRRMSE<- apply(allRRMSEresults[,-c(1:2)],
                1,"min")

    c1 <- cbind(coef.pop,allRBresults,minRB)
    c2 <- cbind(coef.pop,allRVresults,minRV)
    c3 <- cbind(coef.pop,allRRMSEresults,
            minRRMSE)
    print(c1);print(c2);print(c3)
}
```

B.1 Simulation of Linear Regression Model

In case of linear regression, the following functions are used for all of the three scenarios.

```
# Unweighted backward elimination
flag <- 0
uSelect <- function(model,sample) {
    # Covariate matrix
    V  <- as.data.frame(model.matrix(model))
    v  <- names(V)

    # Deleting "Intercept" before checking p-value
    v  <- v[-1]

    p.value <-  summary(model)$coefficients[,"Pr(>|t|)"]
    p.value <-  p.value[-1]

    while (max(p.value) >= 0.05) {
        # Find a position of a maximum p-value
```

```
        for (i in 1:length(p.value)){
            if (max(p.value) == p.value[i])
                {id <- i}
        }

        # Delete the variable with the highest p-value
        v  <- v[-id]

        if (length(v) == 0){
            flag <<- 1
            print ("flag is on")
            break
        }

        txt <- "Y~"
        # Refit the model
        for (i in 1:length(v)){
            txt <- paste(txt,"+",v[i])
        }
        model  <-  lm(txt,sample)
        V  <- as.data.frame(model.matrix(model))
        v  <- names(V)
        v  <- v[-1]
        p.value <-summary(model)$coefficients[,"Pr(>|t|)"]
        p.value <-  p.value[-1]
    }
    return(model)
}

# Weighted backward elimination
flag <- 0
wSelect <- function(model,design) {
    # Covariate matrix
    V  <- as.data.frame(model.matrix(model))
    v  <- names(V)

    # Deleting "Intercept" before checking p-value
    v  <- v[-1]

    p.value <-  summary(model)$coefficients[,"Pr(>|t|)"]
    p.value <-  p.value[-1]

    while (max(p.value) >= 0.05) {
        # Find a position of a maximum p-value
        for (i in 1:length(p.value)){
            if (max(p.value) == p.value[i])
                {id <- i}
        }
```

```
        # Delete the variable with the highest p-value
        v  <- v[-id]

        if (length(v) == 0){
            flag <<- 1
            print ("flag is on")
            break
        }

        txt <- "Y~"
        # Refit the model
        for (i in 1:length(v)){
            txt <- paste(txt,"+",v[i])
        }
        model  <-  svyglm(txt,design)
        V  <- as.data.frame(model.matrix(model))
        v  <- names(V)
        v  <- v[-1]
        p.value <-  summary(model)$coefficients[,"Pr(>|t|)"]
        p.value <-  p.value[-1]
    }
    return(model)
}

# Unweighted Vuong statistic
Vtest <- function(mod1,mod2,sample){
    sigsqmod1 <- (1/n)*t(residuals(mod1))%*%(residuals(mod1))
    sigsqmod2 <- (1/n)*t(residuals(mod2))%*%(residuals(mod2))
    LR        <- 0.5*log(sigsqmod2/sigsqmod1)
    # Variance of LRtest
    e1 <- residuals(mod1); e2 <- residuals(mod2)
    first <- 0
    for(i in 1:n){
        first <- first + (LR + (0.5*(1/sigsqmod2)*e2[i]^2) -
                (0.5*(1/sigsqmod1)*e1[i]^2))^2
    }
    varhat <- ((1/n)*(first)) - LR^2
    zLR <- LR/sqrt(varhat/n)
    return(zLR)
}

# Weighted Vuong statistic
wVtest  <- function(mod1,mod2,sample){
    w   <- as.matrix(diag(sample$weight))
    sum.w <- sum(diag(w))
    sigsqmod1   <- (1/sum.w)*t(residuals(mod1))%*%w%*%
                    (residuals(mod1))
```

```
    sigsqmod2   <- (1/sum.w)*t(residuals(mod2))%*%w%*%
                    (residuals(mod2))
    LR          <- 0.5*log(sigsqmod2/sigsqmod1)
    # Variance of LRtest
    e1      <- residuals(mod1); e2 <- residuals(mod2)
    s1sq    <-(1/n)*t(residuals(mod1))%*%(residuals(mod1))
    s2sq    <-(1/n)*t(residuals(mod2))%*%(residuals(mod2))
    first   <- 0
    sum.wi2 <- 0
    for(i in 1:n){
        first   <- first+((0.5*log(s2sq/s1sq))+
                    (0.5*(1/s2sq)*e2[i]^2)
                    - (0.5*(1/s1sq)*e1[i]^2))^2
        sum.wi2 <- sum.wi2 + w[i,i]^2
    }
    varhat  <- (sum.wi2/sum.w^2)*(((1/n)*(first))-((1/n)*
               ((0.5*log(s2sq/s1sq)) + (0.5*(1/s2sq)*e2[i]^2) -
               (0.5*(1/s1sq)*e1[i]^2)))^2)
    zLR     <- LR/sqrt(varhat)
    return(zLR)
}

# Unweighted model fit
Ufit <- function(Model,sample){
    XZs         <- model.matrix(Model)
    namesXZs    <- colnames(XZs) # Names of covariate variables
    txt <- "Y~"
    for(i in 2:length(namesXZs)){
        txt <- paste(txt,"+",namesXZs[i])
    }
    fit   <-  lm(txt,sample)
    return(fit)
}

# Weighted model fit
Wfit <- function(Model,Design){
    XZs  <- model.matrix(Model)
    namesXZs  <- colnames(XZs) # Names of covariate variables
    txt <- "Y~"
    for(i in 2:length(namesXZs)){
        txt <- paste(txt,"+",namesXZs[i])
    }
    Wfit  <-  svyglm(txt,Design)
    return(Wfit)
}


# Unweighted mean square prediction error
```

```
MSPE   <- function(y,yhat) (1/n)*sum((y-yhat)^2)

# Weighted mean square prediction error
MSPEw  <- function(y,yhat,w) (1/sum(w))*sum(w*(y-yhat)^2)

# CV choice for a final model
endMod  <- function(u,w) {
    e   <- c()
    if(u < w) {e <- 1}  # 1 == "unweighted"
    else{e <- 2}        # 2 == "weighted"
    return(e)
}
```

B.1.1 Simulation of Non-Informative Sampling

```
# Finite population data stratified by design variable (headage)
pop.fn <- function(INFdat,loops){
    Nf <- nrow(INFdat)
    strataY  <- rep(0,Nf)
    for(i in 1:Nf){
        if(INFdat$headage[i]==1){strataY[i]<-1}else
        if(INFdat$headage[i]==2){strataY[i]<-2}else
        if(INFdat$headage[i]==3){strataY[i]<-3}else
        if(INFdat$headage[i]==4){strataY[i]<-4}else
        if(INFdat$headage[i]==5){strataY[i]<-5}else
        {strataY[i]<-6}
    }
    INFdat2<- as.data.frame(cbind(strataY,INFdat))
    ep <- rnorm(Nf,0,3)
    INFdat2$Y <- INFdat2$Y + ep
    return(INFdat2)
}

# Construction of sample weights
sam.fn<- function(pop,str,N.h,n.h,loops){
    strY  <- strat(pop,stratanames=c("strataY"),
             size=n.h,method="srswor")
    sam   <- pop[strY$ID_unit,]
    s     <- as.vector(sam$str)
    weight <- c()
    for(i in 1:sum(n.h)){
        if(s[i]==1) {(weight[i]<-N.h[6]/n.h[6])} else
        if(s[i]==2) {(weight[i]<-N.h[2]/n.h[2])} else
        if(s[i]==3) {(weight[i]<-N.h[5]/n.h[5])} else
        if(s[i]==4) {(weight[i]<-N.h[4]/n.h[4])} else
        if(s[i]==5) {(weight[i]<-N.h[3]/n.h[3])} else
```

```
                    {(weight[i]<-N.h[1]/n.h[1])}
    }
    sam <- as.data.frame(cbind(weight,sam))
    return(sam)
}

# CV operation
cvOperation <- function(mod1,mod2,folds,sample,loops){
    Ss      <- as.vector(round(table(sample$strataY)/folds,0))
    sSize   <- matrix(data=0,nrow=5,ncol=12)
    sSize[1,] <- Ss
    sSize[2,] <- Ss
    sSize[3,] <- Ss
    sSize[4,] <- Ss
    sSize[5,] <- Ss

    f <- getFolds(folds,sample,sSize)

    # Create variables to storage MSPE
    mspe.u  <<- c()
    mspe.w  <<- c()
    mspew.u <<- c()
    mspew.w <<- c()

    for (i in 1:folds){ # Separate sample data to 5 folds
        # A testing set
        sampF  <- sample[f[i,],]# Sample of the i.th fold
        # A training set
        samp_F <- sample[fRest(folds,i,f),]
        # Design identification
        des_F  <- svydesign(id=~1,strata=~strataY,
                  weights=~weight,data=samp_F)

        # Fit models for the training set
        uUfitCV <<- Ufit(mod1,samp_F)
        wWfitCV <<- Wfit(mod2,des_F)

        # Predict the testing set
        uPred <<- predict(uUfitCV,newdata=sampF)
        wPred <<- predict(wWfitCV,newdata=sampF)

        # Calculate MSPE
        mspe.u[i]  <<- MSPE(sampF$Y,uPred)
        mspe.w[i]  <<- MSPE(sampF$Y,wPred)
        mspew.u[i] <<- MSPEw(sampF$Y,uPred,sampF$weight)
        mspew.w[i] <<- MSPEw(sampF$Y,wPred,sampF$weight)
    }
    # Average MSPE
```

```
    avgmspe.u[loops] <<- mean(mspe.u)
    avgmspe.w[loops] <<- mean(mspe.w)
    avgmspew.u[loops]<<- mean(mspew.u)
    avgmspew.w[loops]<<- mean(mspew.w)

    # Choose a final model
    uEndM <- endMod(avgmspe.u[loops],avgmspe.w[loops])
    wEndM <- endMod(avgmspew.u[loops],avgmspew.w[loops])
    return(c(uEndM,wEndM))
}

# Load Packages
library(MASS)
library(lpSolve)
library(sampling)
library(survey)
library(lattice)

# Regression coefficient parameters of the target population model
intercept<- 2
coefZ    <- as.matrix(c(4, 4, 4, 4, 3))
coefX    <- as.matrix(c(1, -1,  -2, 2, 1, 2 , 3, 1.5, 1.5, 1))
coefXZ   <- as.matrix(c(intercept,coefZ,coefX))
coef.pop <- as.matrix(c(intercept,coefZ,coefX))

# Import the real data set (INFHS)
hh.data  <-  as.data.frame(read.table("E:/ForPreeya/nfhs2.txt",
             header=T, sep = "\t", fill = T))

# Delete unused variables
Rdata <- hh.data[,-c(1:3,8,12:16)]

# Deal missing data (9) with the mode
nr <- nrow(Rdata)
for(i in 1:nr){
    if(Rdata$headage[i]==9) {Rdata$headage[i]<-6}
    if(Rdata$headocc[i]==9) {Rdata$headocc[i]<-3}
    if(Rdata$religion[i]==9){Rdata$religion[i]<-1}
}

# Dummy variables
attach(Rdata)
reg2   <- (region==2)*1
reg3   <- (region==3)*1
hht2   <- (hhtype==2)*1
hdsex2 <- (headsex==2)*1
hdage2 <- (headage==2)*1
hdage3 <- (headage==3)*1
```

```
hdage4 <- (headage==4)*1
hdage5 <- (headage==5)*1
hdage6 <- (headage==6)*1
hdocc2 <- (headocc==2)*1
hdocc3 <- (headocc==3)*1
hdocc4 <- (headocc==4)*1
hdedu2 <- (headed==2)*1
hdedu3 <- (headed==3)*1
hdedu4 <- (headed==4)*1
rel2   <- (religion==2)*1
rel3   <- (religion==3)*1
detach(Rdata)

# Create the target response variable Y
Nsp <- nrow(Rdata)
Y   <- coefXZ[1]+(coefXZ[2]*hdage2)+(coefXZ[3]*hdage3)+
       (coefXZ[4]*hdage4)+(coefXZ[5]*hdage5)+(coefXZ[6]*hdage6)+
       (coefXZ[7]*hdsex2)+(coefXZ[8]*hdocc2)+(coefXZ[9]*hdocc3)+
       (coefXZ[10]*hdocc4)+(coefXZ[11]*hdedu2)+
       (coefXZ[12]*hdedu3)+(coefXZ[13]*hdedu4)+
       (coefXZ[14]*reg2)+(coefXZ[15]*reg3)+(coefXZ[16]*hht2)

# The INFHS population data
Rdata <- cbind(Y,Rdata)

# Stratification
# h   1    2    3    4    5    6
# Nh 120  371  522  641  608 2355
# nh 100  150  125  75   100   50

# Re-ranking the order of strata
# for sampling process by 'survey' package
# with a different rank: 6 2 5 4 3 1
Nh <- c(2355,371,608,641,522,120)
nh <- c(50,150,100,75,125,100)
N  <- sum(Nh)
n  <- sum(nh)

loops <- 10000 # Replication number

numCoef   <- nrow(coef.pop)
coefNames <- c("(Intercept)","headage2","headage3",
               "headage4","headage5","headage6","headsex2",
               "headocc2","headocc3","headocc4","headed2",
               "headed3","headed4","region2","region3",
               "hhtype2")
uCoef1  <- matrix(0,loops,numCoef);colnames(uCoef1) <- coefNames
wCoef1  <- matrix(0,loops,numCoef);colnames(wCoef1) <- coefNames
```

```
uCoef   <- matrix(0,loops,numCoef);colnames(uCoef)  <- coefNames
wCoef   <- matrix(0,loops,numCoef);colnames(wCoef)  <- coefNames
uSE     <- matrix(0,loops,numCoef);colnames(uSE)    <- coefNames
wSE     <- matrix(0,loops,numCoef);colnames(wSE)    <- coefNames
noCoefs <- matrix(0,loops,2)
colnames(noCoefs)<- c("p1","p2")

# For CV
avgmspe.u  <- c()
avgmspe.w  <- c()
avgmspew.u <- c()
avgmspew.w <- c()

# All results
result <- matrix(0,loops,10);colnames(result)<-c("case","vVu",
          "vVw","Vu","Vw","smVu","smVw","uCV","wCV","Vote")

# Simulation performance
set.seed(10000)
for(i in 1:loops){

    # Generate finite population data
    Pop <- pop.fn(Rdata,i)

    # Select a sample
    Sam <- sam.fn(Pop,strataY,Nh,nh,i)

    # Dummy variables
    headsex2  <- (Sam$headsex==2)*1
    headage2  <- (Sam$headage==2)*1
    headage3  <- (Sam$headage==3)*1
    headage4  <- (Sam$headage==4)*1
    headage5  <- (Sam$headage==5)*1
    headage6  <- (Sam$headage==6)*1
    headocc2  <- (Sam$headocc==2)*1
    headocc3  <- (Sam$headocc==3)*1
    headocc4  <- (Sam$headocc==4)*1
    headed2   <- (Sam$headed==2)*1
    headed3   <- (Sam$headed==3)*1
    headed4   <- (Sam$headed==4)*1
    religion2 <- (Sam$religion==2)*1
    religion3 <- (Sam$religion==3)*1
    region2   <- (Sam$region==2)*1
    region3   <- (Sam$region==3)*1
    hhtype2   <- (Sam$hhtype==2)*1

    Sam    <- cbind(Sam,headage2,headage3,headage4,headage5,
              headage6,headsex2,headocc2,headocc3,headocc4,
```

```
            headed2,headed3,headed4,religion2,religion3,
            region2,region3,hhtype2)

    # Design identification
    desSam <- svydesign(id=~1,strata=~strataY,
                weights=~weight,data=Sam)

    # Model fit and model choice
    uMod1  <- lm(Y~headage2+headage3+headage4+headage5+
              headage6+headsex2+headocc2+headocc3+
              headocc4+headed2+headed3+headed4+region2+
              region3+hhtype2,Sam)
    wMod1  <- svyglm(Y~headage2+headage3+headage4+headage5+
              headage6+headsex2+headocc2+headocc3+
              headocc4+headed2+headed3+headed4+region2+region3+
              hhtype2,design=desSam)
    uMod2  <- uSelect(uMod1,Sam)
    wMod2  <- wSelect(wMod1,desSam)

    # Store estimators
    uCoef1[i,] <- coefStore(numCoef,uMod1$coefficients,coefNames)
    wCoef1[i,] <- coefStore(numCoef,wMod1$coefficients,coefNames)
    uCoef[i,]  <- coefStore(numCoef,uMod2$coefficients,coefNames)
    wCoef[i,]  <- coefStore(numCoef,wMod2$coefficients,coefNames)
    uSE[i,]    <- coefStore(numCoef,SE(uMod2),coefNames)
    wSE[i,]    <- coefStore(numCoef,SE(wMod2),coefNames)
    noCoefs[i,1] <- length(uMod2$coefficients)
    noCoefs[i,2] <- length(wMod2$coefficients)

    # Model choice decision
    uName <- names(uMod2$coefficients)
    wName <- names(wMod2$coefficients)

    if(setequal(uName,wName)==TRUE)
        {result[i,] <- c(1,9,9,9,9,9,9,9,9,9)}
    else{
        # Vuong
        result[i,1] <- 2
        result[i,2] <- Vtest(uMod2,wMod2,Sam)
        result[i,3] <- wVtest(uMod2,wMod2,Sam)
        result[i,4] <- Vchoice(result[i,2])
        result[i,5] <- Vchoice(result[i,3])
        result[i,6] <- smVchoice(result[i,4],noCoefs[i,1],
                        noCoefs[i,2])
        result[i,7] <- smVchoice(result[i,5],noCoefs[i,1],
                        noCoefs[i,2])

        # CV
```

```
        result[i,8:9] <- cvOperation(uMod2,wMod2,5,Sam,i)

        # Voting system
        result[i,10] <- voting(sum(result[i,6:9]),i)
    }
    print(round(c(i,result[i,]),2))
}  # End of loop

# Evaluation the results only in case of
# non-equivalent regressors
idx     <- which(result[,1]==2)
uCoef2  <- uCoef[idx,]
wCoef2  <- wCoef[idx,]
uSE2    <- uSE[idx,]
wSE2    <- wSE[idx,]
result2 <- result[idx,]

# Print results
paste("Vuong.NW"); table(result2[,4])
paste("Vuong.W"); table(result2[,5])
paste("CV.NW"); table(result2[,8])
paste("CV.W"); table(result2[,9])
paste("Voting System"); table(result2[,10])
allResults2 <- printResults(coef.pop,numCoef,result2,
               uCoef2,wCoef2,uSE2,wSE2,noCoefs)
```

B.1.2 Simulation of Missing Stratification Information

Here functions used for creating finite population data, selecting a sample and operating CV are the same as employed in the previous subsection.

```
# Load Packages
library(MASS)
library(lpSolve)
library(sampling)
library(survey)
library(lattice)

# Regression coefficient parameters of
#the target population model
intercept<- 2
coefZ    <- as.matrix(c(4, 4, 4, 4, 3))
coefX    <- as.matrix(c(1, -1,  -2, 2, 1, 2 , 3, 1.5, 1.5, 1))
coefXZ   <- as.matrix(c(intercept,coefZ,coefX))
coef.pop <- as.matrix(c(intercept,coefZ,coefX))

# Import the real data set (INFHS)
```

```
hh.data  <-  as.data.frame(read.table("E:/ForPreeya/nfhs2.txt",
             header=T, sep = "\t", fill = T))

# Delete unused variables
Rdata <- hh.data[,-c(1:3,8,12:16)]

# Dealing missing data(9) with the mode
nr <- nrow(Rdata)
    for(i in 1:nr){
        if(Rdata$headage[i]==9) {Rdata$headage[i] <-6}
        if(Rdata$headocc[i]==9) {Rdata$headocc[i] <-3}
        if(Rdata$religion[i]==9){Rdata$religion[i]<-1}
    }

# Dummy variables
attach(Rdata)
reg2    <- (region==2)*1
reg3    <- (region==3)*1
hht2    <- (hhtype==2)*1
hdsex2  <- (headsex==2)*1
hdage2  <- (headage==2)*1
hdage3  <- (headage==3)*1
hdage4  <- (headage==4)*1
hdage5  <- (headage==5)*1
hdage6  <- (headage==6)*1
hdocc2  <- (headocc==2)*1
hdocc3  <- (headocc==3)*1
hdocc4  <- (headocc==4)*1
hdedu2  <- (headed==2)*1
hdedu3  <- (headed==3)*1
hdedu4  <- (headed==4)*1
rel2    <- (religion==2)*1
rel3    <- (religion==3)*1
detach(Rdata)

# Create Y
Nsp <- nrow(Rdata)
Y   <- coefXZ[1]+(coefXZ[2]*hdage2)+(coefXZ[3]*hdage3)+
       (coefXZ[4]*hdage4)+(coefXZ[5]*hdage5)+(coefXZ[6]*hdage6)+
       (coefXZ[7]*hdsex2)+(coefXZ[8]*hdocc2)+(coefXZ[9]*hdocc3)+
       (coefXZ[10]*hdocc4)+(coefXZ[11]*hdedu2)+
       (coefXZ[12]*hdedu3)+(coefXZ[13]*hdedu4)+
       (coefXZ[14]*reg2)+(coefXZ[15]*reg3)+(coefXZ[16]*hht2)

# The INFHS population data
Rdata <- cbind(Y,Rdata)

# Stratification
```

```
Nh  <- c(2355,371,608,641,522,120)
nh  <- c(50,150,100,75,125,100)
N   <- sum(Nh)
n   <- sum(nh)

# Replication number
loops <- 10000

# Number of parameters
numCoef <- nrow(coef.pop)

coefNames <- c("(Intercept)","headage2","headage3",
               "headage4","headage5","headage6","headsex2",
               "headocc2","headocc3","headocc4","headed2",
               "headed3","headed4","region2","region3",
               "hhtype2")

uCoef1  <- matrix(0,loops,numCoef);colnames(uCoef1) <-coefNames
wCoef1  <- matrix(0,loops,numCoef);colnames(wCoef1) <-coefNames
uCoef   <- matrix(0,loops,numCoef);colnames(uCoef)  <-coefNames
wCoef   <- matrix(0,loops,numCoef);colnames(wCoef)  <-coefNames
uSE     <- matrix(0,loops,numCoef);colnames(uSE)    <-coefNames
wSE     <- matrix(0,loops,numCoef);colnames(wSE)    <-coefNames
noCoefs <- matrix(0,loops,2)
colnames(noCoefs)<- c("p1","p2")

# For CV
avgmspe.u  <- c()
avgmspe.w  <- c()
avgmspew.u <- c()
avgmspew.w <- c()

# All results
result          <- matrix(0,loops,10)
colnames(result)<- c("case","vVu","vVw","Vu","Vw","smVu",
                     "smVw","uCV","wCV","Vote")

# Simulation performance
set.seed(10000)
for(i in 1:loops){

    # Generate finite population data and select a sample
    Pop <- pop.fn(Rdata,i)
    Sam <- sam.fn(Pop,strataY,Nh,nh,i)

    # Dummy variables
    headsex2  <- (Sam$headsex==2)*1
    headage2  <- (Sam$headage==2)*1
```

```
headage3  <- (Sam$headage==3)*1
headage4  <- (Sam$headage==4)*1
headage5  <- (Sam$headage==5)*1
headage6  <- (Sam$headage==6)*1
headocc2  <- (Sam$headocc==2)*1
headocc3  <- (Sam$headocc==3)*1
headocc4  <- (Sam$headocc==4)*1
headed2   <- (Sam$headed==2)*1
headed3   <- (Sam$headed==3)*1
headed4   <- (Sam$headed==4)*1
religion2 <- (Sam$religion==2)*1
religion3 <- (Sam$religion==3)*1
region2   <- (Sam$region==2)*1
region3   <- (Sam$region==3)*1
hhtype2   <- (Sam$hhtype==2)*1

Sam       <- cbind(Sam,headsex2,headocc2,headocc3,
             headocc4,headed2,headed3,headed4,
             religion2,religion3,region2,region3,hhtype2)

# Design identification
desSam    <- svydesign(id=~1,strata=~strataY,
             weights=~weight,data=Sam)

# Model fit and model choice
uMod1   <- lm(Y~headsex2+headocc2+headocc3+headocc4+
           headed2+headed3+headed4+region2+region3+
           hhtype2,Sam)
wMod1   <- svyglm(Y~headsex2+headocc2+headocc3+headocc4+
           headed2+headed3+headed4+region2+region3+hhtype2,
           design=desSam)
uMod2   <- uSelect(uMod1,Sam)
wMod2   <- wSelect(wMod1,desSam)

# Store estimators
uCoef1[i,] <- coefStore(numCoef,uMod1$coefficients,coefNames)
wCoef1[i,] <- coefStore(numCoef,wMod1$coefficients,coefNames)
uCoef[i,]  <- coefStore(numCoef,uMod2$coefficients,coefNames)
wCoef[i,]  <- coefStore(numCoef,wMod2$coefficients,coefNames)

uSE[i,]    <- coefStore(numCoef,SE(uMod2),coefNames)
wSE[i,]    <- coefStore(numCoef,SE(wMod2),coefNames)
noCoefs[i,1] <- length(uMod2$coefficients)
noCoefs[i,2] <- length(wMod2$coefficients)

# Model choice decision
uName <- names(uMod2$coefficients)
wName <- names(wMod2$coefficients)
```

```
    if(setequal(uName,wName)==TRUE)
        {result[i,] <- c(1,9,9,9,9,9,9,9,9,9)}
    else{
        # Vuong
        result[i,1] <- 2
        result[i,2] <- Vtest(uMod2,wMod2,Sam)
        result[i,3] <- wVtest(uMod2,wMod2,Sam)
        result[i,4] <- Vchoice(result[i,2])
        result[i,5] <- Vchoice(result[i,3])
        result[i,6] <- smVchoice(result[i,4],noCoefs[i,1],
                        noCoefs[i,2])
        result[i,7] <- smVchoice(result[i,5],noCoefs[i,1],
                        noCoefs[i,2])

        # CV
        result[i,8:9] <- cvOperation(uMod2,wMod2,5,Sam,i)

        # Voting system
        result[i,10] <- voting(sum(result[i,6:9]),i)
    }
    print(round(c(i,result[i,]),2))
}  # End of loop

# Evaluation the results only in case of
# non-equivalent regressors
idx     <- which(result[,1]==2)
uCoef2  <- uCoef[idx,]
wCoef2  <- wCoef[idx,]
uSE2    <- uSE[idx,]
wSE2    <- wSE[idx,]
result2 <- result[idx,]

# Print results
paste("Vuong.NW"); table(result2[,4])
paste("Vuong.W") ; table(result2[,5])
paste("CV.NW")   ; table(result2[,8])
paste("CV.W")    ; table(result2[,9])
paste("Voting System"); table(result2[,10])
allResults2 <- printResults(coef.pop,numCoef,result2,
               uCoef2,wCoef2,uSE2,wSE2,noCoefs)
```

B.1.3 Simulation of Response-Based Sampling

```
# Finite population data stratified by the target variable Y
pop.fn  <- function(INFdat,N.h,n.h,loops){
```

```
    N   <- sum(N.h);n <- sum(n.h)
    w   <- N.h/n.h
    noStr   <- length(N.h)
    strataY <- rep(0,N)
    scalW   <- rep(0,N)
    count   <- rep(0,noStr)
    id <- 1

    while((count[1]<N.h[1])|(count[2]<N.h[2])|
          (count[3]<N.h[3])|(count[4]<N.h[4])){
        e <- rnorm(1,0,3)
        if((e<=-1)&&(count[1]<N.h[1])){
            INFdat$Y[id] <- INFdat$Y[id] + e
            strataY[id] <- 1
            scalW[id] <- (n/N)*w[1]
            count[1] <- count[1] + 1
            id <- id + 1
        }else{if((e>-1)&&(e<=0)&&(count[2]<N.h[2])){
            INFdat$Y[id] <- INFdat$Y[id] + e
            strataY[id] <- 2
            scalW[id] <- (n/N)*w[2]
            count[2] <- count[2] + 1
            id <- id + 1
            }else{if((e>=0)&&(e<1)&&(count[3]<N.h[3])){
                INFdat$Y[id] <- INFdat$Y[id] + e
                strataY[id] <- 3
                scalW[id] <- (n/N)*w[3]
                count[3] <- count[3] + 1
                id <- id + 1
                }else{if(count[4]<N.h[4]){
                    INFdat$Y[id] <- INFdat$Y[id] + e
                    strataY[id] <- 4
                    scalW[id] <- (n/N)*w[4]
                    count[4] <- count[4] + 1
                    id <- id + 1
                    }
                }
            }
        }
    }
    pop <- cbind(strataY,scalW,INFdat)
    return(pop)
}

# Construction of sampling weights
sam.fn <- function(pop,str,N.h,n.h,loops){
    strY    <-  strata(pop,stratanames=c("strataY"),
                size=n.h,method="srswor")
```

```
    sam     <-  pop[strY$ID_unit,]
    s       <- as.vector(sam$str)
    weight  <- c()
    for(i in 1:sum(n.h)){
        if(s[i]==1) {weight[i]<-N.h[1]/n.h[1]} else
        if(s[i]==2) {weight[i]<-N.h[2]/n.h[2]} else
        if(s[i]==3) {weight[i]<-N.h[3]/n.h[3]} else
                    {weight[i]<-N.h[4]/n.h[4]}
    }
    sam <- as.data.frame(cbind(weight,sam))
    return(sam)
}

# CV operation
cvOperation   <- function(mod1,mod2,folds,sample,loops){
    Ss <- as.vector(round(table(sample$strataY)/folds,0))
    sSize     <- matrix(data=0,nrow=5,ncol=12)
    sSize[1,] <- Ss
    sSize[2,] <- Ss
    sSize[3,] <- Ss
    sSize[4,] <- Ss
    sSize[5,] <- Ss

    f <- getFolds(folds,sample,sSize)

    # Create variables to storage MSPE
    mspe.u  <<- c()
    mspe.w  <<- c()
    mspew.u <<- c()
    mspew.w <<- c()

    for (i in 1:folds){ # Separate sample into 5 folds

        # A testing set
        sampF  <- sample[f[i,],]# Sample of the i.th fold

        # A training set
        samp_F <- sample[fRest(folds,i,f),]
        des_F  <- svydesign(id=~1,strata=~strataY,
                  weights=~scalW,data=samp_F

        # Fit two models
        uUfitCV <- Ufit(mod1,samp_F)
        wWfitCV <- Wfit(mod2,des_F)

        # Predict the testing set
        uPred <<- predict(uUfitCV,newdata=sampF)
        wPred <<- predict(wWfitCV,newdata=sampF)
```

```
        # Calculate MSPE
        mspe.u[i]  <<- MSPE(sampF$Y,uPred)
        mspe.w[i]  <<- MSPE(sampF$Y,wPred)
        mspew.u[i] <<- MSPEw(sampF$Y,uPred,sampF$weight)
        mspew.w[i] <<- MSPEw(sampF$Y,wPred,sampF$weight)
    }
    # Averge MSPE
    avgmspe.u[loops] <<- mean(mspe.u)
    avgmspe.w[loops] <<- mean(mspe.w)
    avgmspew.u[loops]<<- mean(mspew.u)
    avgmspew.w[loops]<<- mean(mspew.w)
    # Choose a final model
    uEndM <- endMod(avgmspe.u[loops],avgmspe.w[loops])
    wEndM <- endMod(avgmspew.u[loops],avgmspew.w[loops])
    return(c(uEndM,wEndM))
}

rm(list = ls(all = TRUE))
# Load packages
library(MASS)
library(lpSolve)
library(sampling)
library(survey)
library(lattice)

# Regression coefficient parameters of the target population model
intercept<- 2
coefZ    <- as.matrix(c(4, 4, 4, 4, 3))
coefX    <- as.matrix(c(1, -1,  -2, 2, 1, 2 , 3, 1.5, 1.5, 1))
coefXZ   <- as.matrix(c(intercept,coefZ,coefX))
coef.pop <- as.matrix(c(intercept,coefZ,coefX))

# Import the real data set (INFHS)
hh.data  <-  as.data.frame(read.table("E:/ForPreeya/nfhs2.txt",
             header=T, sep = "\t", fill = T))

# Delete unused variables
Rdata <- hh.data[,-c(1:3,8,12:16)]

# Deal missing data(9) with the mode
nr <- nrow(Rdata)
for(i in 1:nr){
    if(Rdata$headage[i]==9) {Rdata$headage[i] <-6}
    if(Rdata$headocc[i]==9) {Rdata$headocc[i] <-3}
    if(Rdata$religion[i]==9){Rdata$religion[i]<-1}
}
```

```
# Dummy variables
attach(Rdata)
reg2   <- (region==2)*1
reg3   <- (region==3)*1
hht2   <- (hhtype==2)*1
hdsex2 <- (headsex==2)*1
hdage2 <- (headage==2)*1
hdage3 <- (headage==3)*1
hdage4 <- (headage==4)*1
hdage5 <- (headage==5)*1
hdage6 <- (headage==6)*1
hdocc2 <- (headocc==2)*1
hdocc3 <- (headocc==3)*1
hdocc4 <- (headocc==4)*1
hdedu2 <- (headed==2)*1
hdedu3 <- (headed==3)*1
hdedu4 <- (headed==4)*1
rel2   <- (religion==2)*1
rel3   <- (religion==3)*1
detach(Rdata)

# Create Y
Nsp <- nrow(Rdata)
Y   <- coefXZ[1]+(coefXZ[2]*hdage2)+(coefXZ[3]*hdage3)+
       (coefXZ[4]*hdage4)+(coefXZ[5]*hdage5)+
       (coefXZ[6]*hdage6)+(coefXZ[7]*hdsex2)+
       (coefXZ[8]*hdocc2)+(coefXZ[9]*hdocc3)+
       (coefXZ[10]*hdocc4)+(coefXZ[11]*hdedu2)+
       (coefXZ[12]*hdedu3)+(coefXZ[13]*hdedu4)+
       (coefXZ[14]*reg2)+(coefXZ[15]*reg3)+
       (coefXZ[16]*hht2)

# INFHS population data
Rdata <- cbind(Y,Rdata)

# Stratification
# h 1 2 3 4
Nh  <- c(1500,1300,1200,617)
nh  <- c(50,100,200,250)
N   <- sum(Nh)
n   <- sum(nh)

# Replication number
loops <- 10000

# Number of regression coefficient parameters
numCoef   <- nrow(coef.pop)
```

```
coefNames <- c("(Intercept)","headage2","headage3",
               "headage4","headage5","headage6",
               "headsex2","headocc2","headocc3",
               "headocc4","headed2", "headed3",
               "headed4","region2","region3","hhtype2")

uCoef1  <- matrix(0,loops,numCoef);colnames(uCoef1) <-  coefNames
wCoef1  <- matrix(0,loops,numCoef);colnames(wCoef1) <-  coefNames
uCoef   <- matrix(0,loops,numCoef);colnames(uCoef)  <-  coefNames
wCoef   <- matrix(0,loops,numCoef);colnames(wCoef)  <-  coefNames
uSE     <- matrix(0,loops,numCoef);colnames(uSE)    <-  coefNames
wSE     <- matrix(0,loops,numCoef);colnames(wSE)    <-  coefNames
noCoefs <- matrix(0,loops,2)
colnames(noCoefs) <- c("p1","p2")


# For CV
avgmspe.u  <- c()
avgmspe.w  <- c()
avgmspew.u <- c()
avgmspew.w <- c()

# All results
result          <- matrix(0,loops,10)
colnames(result)<- c("case","vVu","vVw","Vu","Vw","smVu",
                     "smVw","uCV","wCV","Vote")

# Simulation performance
set.seed(10000)
for(i in 1:loops){
    # Generate finite population data and select a sample
    Pop <- pop.fn(Rdata,Nh,i)
    Sam <- sam.fn(Pop,strataY,Nh,nh,i)

    # Dummy variables
    headage2  <- (Sam$headage==2)*1
    headage3  <- (Sam$headage==3)*1
    headage4  <- (Sam$headage==4)*1
    headage5  <- (Sam$headage==5)*1
    headage6  <- (Sam$headage==6)*1
    headsex2  <- (Sam$headsex==2)*1
    headocc2  <- (Sam$headocc==2)*1
    headocc3  <- (Sam$headocc==3)*1
    headocc4  <- (Sam$headocc==4)*1
    headed2   <- (Sam$headed==2)*1
    headed3   <- (Sam$headed==3)*1
    headed4   <- (Sam$headed==4)*1
    religion2 <- (Sam$religion==2)*1
```

```
religion3 <- (Sam$religion==3)*1
region2   <- (Sam$region==2)*1
region3   <- (Sam$region==3)*1
hhtype2   <- (Sam$hhtype==2)*1

Sam  <- cbind(Sam,headage2,headage3,headage4,headage5,
        headage6,headsex2,headocc2,headocc3,headocc4,
        headed2,headed3,headed4,region2,region3,hhtype2)

# Design identification
desSam <- svydesign(id=~1,strata=~strataY,
          weights=~scalW,data=Sam)
desRep <- as.svrepdesign(desSam)

# Model fit and model choice
uMod1  <- lm(Y~headage2+headage3+headage4+headage5+headage6+
          headsex2+headocc2+headocc3+headocc4+headed2+
          headed3+headed4+region2+region3+hhtype2,Sam)
wMod1  <- svyglm(Y~headage2+headage3+headage4+headage5+
          headage6+headsex2+headocc2+headocc3+headocc4+
          headed2+headed3+headed4+region2+region3+hhtype2,
          design=desSam)
uMod2  <- uSelect(uMod1,Sam)
wMod2  <- wSelect(wMod1,desSam)

# Store estimators
uCoef1[i,]   <- coefStore(numCoef,uMod1$coefficients,coefNames)
wCoef1[i,]   <- coefStore(numCoef,wMod1$coefficients,coefNames)
uCoef[i,]    <- coefStore(numCoef,uMod2$coefficients,coefNames)
wCoef[i,]    <- coefStore(numCoef,wMod2$coefficients,coefNames)

uSE[i,]      <- coefStore(numCoef,SE(uMod2),coefNames)
wSE[i,]      <- coefStore(numCoef,SE(wMod2),coefNames)
noCoefs[i,1] <- length(uMod2$coefficients)
noCoefs[i,2] <- length(wMod2$coefficients)

# Model choice decision
uName <- names(uMod2$coefficients)
wName <- names(wMod2$coefficients)

if(setequal(uName,wName)==TRUE)
    {result[i,] <- c(1,9,9,9,9,9,9,9,9,9)}
else{
    # Vuong
    result[i,1] <- 2
    result[i,2] <- Vtest(uMod2,wMod2,Sam)
    result[i,3] <- wVtest(uMod2,wMod2,Sam)
    result[i,4] <- Vchoice(result[i,2])
```

```
        result[i,5] <- Vchoice(result[i,3])
        result[i,6] <- smVchoice(result[i,4],noCoefs[i,1],
                        noCoefs[i,2])
        result[i,7] <- smVchoice(result[i,5],noCoefs[i,1],
                        noCoefs[i,2])

        # CV
        result[i,8:9] <- cvOperation(uMod2,wMod2,5,Sam,i)

        # Voting system
        result[i,10] <- voting(sum(result[i,6:9]),i)
    }
    print(round(c(i,result[i,]),2))
}  # End of loop

# Evaluation the results only in case of
# non-equivalent regressors
idx     <- which(result[,1]==2)
uCoef2  <- uCoef[idx,]
wCoef2  <- wCoef[idx,]
uSE2    <- uSE[idx,]
wSE2    <- wSE[idx,]
result2 <- result[idx,]

# Print results
paste("Vuong.NW"); table(result2[,4])
paste("Vuong.W") ; table(result2[,5])
paste("CV.NW")   ; table(result2[,8])
paste("CV.W")    ; table(result2[,9])
paste("Voting System"); table(result2[,10])
allResults2 <- printResults(coef.pop,numCoef,
               result2,uCoef2,wCoef2,
               uSE2,wSE2,noCoefs)
```

B.1.4 Simulation of Modeling Bias

```
# Load packages
library(survey)
library(MASS)
library(lpSolve)
library(sampling)

# Population model : y = 1 + x + z + ep
para    <- as.matrix(c(1,1,1))
N       <- 5000
n       <- 1000
```

```
mu      <- matrix(c(0,0),2,1)
rho     <- 0.1
Sigma   <- matrix(c(1,rho,rho,1),2,2)

set.seed(1000)
xz  <- mvrnorm(N,mu,Sigma,empirical=F)
x   <- I(xz[,1]>0)*1
z   <- I(xz[,2]>0)*1

# Generate population data
pop.fn  <- function(loops){
    ep  <- rnorm(N,0,1)
    y   <- para[1] + para[2]*x + para[3]*z + ep
    pop <- cbind(y,x,z)
}

# Fit logistic regression: E(Z|X) in population data
    logis <- glm(z~x,family=binomial("logit"))
    summary(logis)
    pred  <- fitted(logis);head(pred)
    if(pred[1] < pred[2])
    {(P0 <- pred[1]) && (P1 <- pred[2])}
    if(pred[1] > pred[2])
    {(P1 <- pred[1]) && (P0 <- pred[2])}

# Select a sample by the Z variable
samp.fn <- function(loops){
    Nh  <- table(z);    nh <- c(600,400)
    stratZ <- strata(pop,stratanames=c("z"),
              size=c(nh,nh),method="srswor")
    samp   <- as.data.frame(pop[stratZ$ID_unit,])
    weight <- c()
    for(i in 1:n){
        if(samp$z[i]==0) {weight[i] <- Nh[1]/nh[1]}
                    else {weight[i] <- Nh[2]/nh[2]}
    }
    samp   <<- cbind(samp,weight)
    dessamp<<- svydesign(id=~1,strata=~z,weights=weight,
               data=samp)
}

# Simulation performance

N0   <- length(subset(x,x==0))
N1   <- length(subset(x,x==1))
noParas <- 2
sampColnames <- c("a.hat","b.hat")
```

```
# Replication number
loops   <- 1000

# Storage variables
uCoef           <-  matrix(0,loops,noParas)
colnames(uCoef) <-  sampColnames
wCoef           <-  matrix(0,loops,noParas)
colnames(wCoef) <-  sampColnames
sumwixi         <-  c()

# Simulation performance
set.seed(10002)
for(i in 1:loops){

    # Generate finite population data and select a sample
    pop.fn(i)
    samp.fn(i)

    # Model fit
    ufit <- lm(y~x,samp)
    wfit <- svyglm(y~x,dessamp)

    # Store estimators
    uCoef[i,]   <- coef(ufit)
    wCoef[i,]   <- coef(wfit)
}

# Incomplete model parameters
a.star <- para[1] + para[3]*P0
b.star <- para[2] + P1 - P0

# - Intercept term
# Bias of the UMS from the complete population model
uBias.ahatP <- (1/loops)*sum(uCoef[,1]-para[1])

# Bias of the UMS from the incomplete model
uBias.ahatE <- (1/loops)*sum(uCoef[,1]-a.star)

# Bias of the WMS from the complete population model
wBias.ahatP <- (1/loops)*sum(wCoef[,1]-para[1])

# Bias of the WMS from the incomplete model
wBias.ahatE <- (1/loops)*sum(wCoef[,1]-a.star)

intEst <- matrix(round(c(uBias.ahatP,uBias.ahatE,
          wBias.ahatP,wBias.ahatE),3),2,2)
rownames(intEst)<- c("Pop.model","Sam.model")
colnames(intEst)<- c("ahat.u","ahat.w")
```

```
print(intEst)

# - Slope
# Bias of the UMS from the complete population model
uBias.bhatP <- (1/loops)*sum(uCoef[,2]-para[2])

# Bias of the UMS from the incomplete model
uBias.bhatE <- (1/loops)*sum(uCoef[,2]-b.star)

# Bias of the WMS from the complete population model
wBias.bhatP <- (1/loops)*sum(wCoef[,2]-para[2])

# Bias of the WMS from the incomplete model
wBias.bhatE <- (1/loops)*sum(wCoef[,2]-b.star)

slopeEst <- matrix(round(c(uBias.bhatP,uBias.bhatE,
            wBias.bhatP,wBias.bhatE),3),2,2)
rownames(slopeEst)<-c("Pop.model","Sam.model")
colnames(slopeEst)<- c("bhat.u","bhat.w")
print(slopeEst)
```

B.2 Simulation of Logistic Regression Model

The following functions are shared for all three scenarios.

```
# Unweighted backward elimination
flag <- 0
uSelect <- function(model,sample) {
    # Covariate matrix
    V  <- as.data.frame(model.matrix(model))
    v  <- names(V)

    # Delete "Intercept" before checking p-values
    v  <- v[-1]
    p.value <-  summary(model)$coefficients[,"Pr(>|t|)"]

    # Delete p.value of "Intercept"
    p.value <-  p.value[-1]

    while (max(p.value) >= 0.05) {
        # Find a position of the highest p.value
        for (i in 1:length(p.value)){
            if (max(p.value) == p.value[i])
                {id <- i}
        }
        # Delete the variable with the highest p-value
        v  <- v[-id]
```

```
        if (length(v) == 0){
            flag <<- 1
            print ("flag is on")
            break
        }

        txt <- "Y~"
        # Refit the model
        for (i in 1:length(v)){
        txt <- paste(txt,"+",v[i])
        }
        model  <-  glm(txt,family=quasibinomial,sample)
        V  <- as.data.frame(model.matrix(model))
        v  <- names(V)
        v  <- v[-1]
        p.value <-summary(model)$coefficients[,"Pr(>|t|)"]
        p.value <-  p.value[-1]
    }
    return(model)
}

# Weighted backward elimination
flag <- 0
wSelect <- function(model,design) {
    # Covariate matrix
    V  <- as.data.frame(model.matrix(model))
    v  <- names(V)

    # Delete "Intercept" before checking p-values
    v  <- v[-1]
    p.value <-  summary(model)$coefficients[,"Pr(>|t|)"]

    # Delete p.value of "Intercept"
    p.value <-  p.value[-1]

    while (max(p.value) >= 0.05) {
        # Find a position of the highest p.value
        for (i in 1:length(p.value)){
            if (max(p.value) == p.value[i])
                {id <- i}
        }
        # Delete the variable with the highest p-value
        v  <- v[-id]

        if (length(v) == 0){
            flag <<- 1
            print ("flag is on")
            break
```

```
        }

        txt <- "Y~"
        # Refit the model
        for (i in 1:length(v)){
            txt <- paste(txt,"+",v[i])
        }
        model <- svyglm(txt,family=quasibinomial,design)
        V  <- as.data.frame(model.matrix(model))
        v  <- names(V)
        v  <- v[-1]
        p.value <-summary(model)$coefficients[,"Pr(>|t|)"]
        p.value <-  p.value[-1]
    }
    return(model)
}

# Inverse logit function
invlogit <- function(x) {1/(1+exp(-x))}

# Unweighted Vuong statistic
Vtest <- function(mod1,mod2,sample){
    y       <- sample$Y
    p1      <- mod1$fitted.values
    p2      <- mod2$fitted.values
    logLf   <- c()
    logLg   <- c()
    d       <- 0
    v1      <- 0
    v2      <- 0
    for(i in 1:n){
        logLf[i] <- (y[i]*p1[i]) - log(1+exp(p1[i]))
        logLg[i] <- (y[i]*p2[i]) - log(1+exp(p2[i]))
        d   <- d + (logLf[i]-logLg[i])
        v1  <- v1+(y[i]*(p1[i]-p2[i])-log(1+exp(p1[i]))+
                log(1+exp(p2[i])))^2
        v2  <- v2+(y[i]*(p1[i]-p2[i])-log(1+exp(p1[i]))+
                log(1+exp(p2[i])))
    }
    LR      <- (1/n)*d
    varhat  <- ((1/n)*v1)-((1/n)*v2)^2
    zLR     <- LR/sqrt(varhat/n)
    return(zLR)
}

# Weighted Vuong statistic
wVtest <- function(mod1,mod2,sample){
    w       <- as.vector(sample$weight)
```

```
    sum.w   <- sum(w)
    y       <- sample$Y
    p1      <- mod1$fitted.values
    p2      <- mod2$fitted.values
    logLf   <- c()
    logLg   <- c()
    d       <- 0
    v1      <- 0
    v2      <- 0
    sum.wi2 <- 0
    for(i in 1:n){
        logLf[i] <- w[i]*((y[i]*p1[i])-log(1+exp(p1[i])))
        logLg[i] <- w[i]*((y[i]*p2[i])-log(1+exp(p2[i])))
        d   <- d + (logLf[i]-logLg[i])
        v1  <- v1+(y[i]*(p1[i]-p2[i])-log(1+exp(p1[i]))+
               log(1+exp(p2[i])))^2
        v2  <- v2+(y[i]*(p1[i]-p2[i])-log(1+exp(p1[i]))+
               log(1+exp(p2[i])))
        sum.wi2 <- sum.wi2 + w[i]^2
    }
    LR      <- (1/sum.w)*(sum(logLf-logLg))
    varhat  <- (sum.wi2/sum.w^2)*(((1/n)*v1)-((1/n)*v2)^2)
    zLR     <- LR/sqrt(varhat)
    return(zLR)
}

# Unweighted model fit
Ufit <- function(Model,sample){
    XZs         <- model.matrix(Model)
    namesXZs    <- colnames(XZs)
    txt <- "Y~"
    for(i in 2:length(namesXZs)){
        txt <- paste(txt,"+",namesXZs[i])
    }
    fit <-  glm(txt,family=binomial,sample)
    return(fit)
}

# Weighted model fit
Wfit <- function(Model,Design){
    XZs  <- model.matrix(Model)
    namesXZs  <- colnames(XZs)
    txt <- "Y~"
    for(i in 2:length(namesXZs)){
        txt <- paste(txt,"+",namesXZs[i])
    }
    Wfit    <-  svyglm(txt,family=quasibinomial,Design)
    return(Wfit)
```

```
}

# Unweighted prediction error
PE  <- function(y,yhat) {
    n1  <- c()
    for(i in 1:length(y)){
        n1[i]<- ifelse(y[i]==yhat[i],0,1)
    }
    pe  <- sum(n1)/length(y)
    return(pe)
}

# Weighted prediction error
PEw <- function(y,yhat,wfold) {
    w1 <- c()
    for(i in 1:length(y)){
        w1[i] <- ifelse(y[i]==yhat[i],0,wfold[i])
    }
    pe  <- sum(w1)/sum(wfold)
    return(pe)
}

# CV choice
endMod <- function(u,w) {
    e <- c()
    if(u < w) {e <- 1}# 1 =="unweighted"
    else{e <- 2}      # 2 == "weighted"
    return(e)
}

# CV operation
cvOperation <- function(mod1,mod2,folds,sample,loops){
    Ss <- as.vector(round(table(sample$strataY)/folds,0))
    sSize     <- matrix(data=0,nrow=5,ncol=12)
    sSize[1,] <- Ss
    sSize[2,] <- Ss
    sSize[3,] <- Ss
    sSize[4,] <- Ss
    sSize[5,] <- Ss

    f <- getFolds(folds,sample,sSize)

    # Creating variables to storage PE
    pe.u  <<- c()
    pe.w  <<- c()
    pew.u <<- c()
    pew.w <<- c()
```

```
    # Separate sample data into 5 folds
    for (i in 1:folds){
        # A testing set
        sampF  <- sample[f[i,],]

        # A training set
        samp_F <- sample[fRest(folds,i,f),]

        # Design identification
        des_F  <- svydesign(id=~1,strata=~strataY,
                  weights=~weight,data=samp_F)

        # Fit two models
        uUfitCV <- Ufit(mod1,samp_F)
        wWfitCV <- Wfit(mod2,des_F)

        # Predict the testing set
        uPred <- invlogit(predict(uUfitCV,newdata=sampF))
        wPred <- invlogit(predict(wWfitCV,newdata=sampF))
        uYhat <- ifelse(uPred<0.5,0,1)
        wYhat <- ifelse(wPred<0.5,0,1)

        # Calculate PE
        pe.u[i]  <- PE(sampF$Y,uYhat)
        pe.w[i]  <- PE(sampF$Y,wYhat)
        pew.u[i] <- PEw(sampF$Y,uYhat,sampF$weight)
        pew.w[i] <- PEw(sampF$Y,wYhat,sampF$weight)
    }
    # Average PE
    avgpe.u[loops] <- mean(pe.u)
    avgpe.w[loops] <- mean(pe.w)
    avgpew.u[loops]<- mean(pew.u)
    avgpew.w[loops]<- mean(pew.w)
    # Choose a final model
    uEndM <- endMod(avgpe.u[loops],avgpe.w[loops])
    wEndM <- endMod(avgpew.u[loops],avgpew.w[loops])
    return(c(uEndM,wEndM))
}
```

B.2.1 Simulation of Non-Informative Sampling

```
# Load packages
library(MASS)
library(lpSolve)
library(sampling)
library(survey)
```

```
library(lattice)

# Regression coefficient parameters of
# the target population model
intercept<- -1
coefZ    <- as.matrix(c(3, 3, 3, 3, 3))
coefX    <- as.matrix(c(0.5,-0.5,-1,0.5,0.5,1,2,0.5,0.5,0.5))
coefXZ   <- as.matrix(c(intercept,coefZ,coefX))
coef.pop <- as.matrix(c(intercept,coefZ,coefX))

# Import the real data set (INFHS)
hh.data<-as.data.frame(read.table("E:/ForPreeya/nfhs2.txt",
         header=T, sep = "\t", fill = T))

# Delete unused variables
Rdata<- hh.data[,-c(1:3,8,12:16)]

# Deal missing data(9) with the mode
nr <- nrow(Rdata)
for(i in 1:nr){
    if(Rdata$headage[i]==9){Rdata$headage[i]<-6}
    if(Rdata$headocc[i]==9){Rdata$headocc[i]<-3}
    if(Rdata$religion[i]==9){Rdata$religion[i]<-1}
}

# Dummy variables
attach(Rdata)
reg2   <- (region==2)*1
reg3   <- (region==3)*1
hht2   <- (hhtype==2)*1
hdsex2 <- (headsex==2)*1
hdage2 <- (headage==2)*1
hdage3 <- (headage==3)*1
hdage4 <- (headage==4)*1
hdage5 <- (headage==5)*1
hdage6 <- (headage==6)*1
hdocc2 <- (headocc==2)*1
hdocc3 <- (headocc==3)*1
hdocc4 <- (headocc==4)*1
hdedu2 <- (headed==2)*1
hdedu3 <- (headed==3)*1
hdedu4 <- (headed==4)*1
rel2   <- (religion==2)*1
rel3   <- (religion==3)*1
detach(Rdata)

# Create Y
Nsp <- nrow(Rdata)
```

```
Y   <- 1/(1+(exp(-(coefXZ[1]+(coefXZ[2]*hdage2)+
       (coefXZ[3]*hdage3)+(coefXZ[4]*hdage4)+
       (coefXZ[5]*hdage5)+(coefXZ[6]*hdage6)+
       (coefXZ[7]*hdsex2)+(coefXZ[8]*hdocc2)+
       (coefXZ[9]*hdocc3)+(coefXZ[10]*hdocc4)+
       (coefXZ[11]*hdedu2)+(coefXZ[12]*hdedu3)+
       (coefXZ[13]*hdedu4)+(coefXZ[14]*reg2)+
       (coefXZ[15]*reg3)+(coefXZ[16]*hht2)))))

# The INFHS population data
Rdata <- cbind(Y,Rdata)

# Finite population data stratified
# by design variable (headage)
pop.fn <- function(INFdat,loops){
    Nf <- nrow(INFdat)
    u  <- c()
    strataY <- c()

    for(i in 1:Nf){
        if(INFdat$headage[i]==1){(strataY[i] <- 1)}
        if(INFdat$headage[i]==2){(strataY[i] <- 2)}
        if(INFdat$headage[i]==3){(strataY[i] <- 3)}
        if(INFdat$headage[i]==4){(strataY[i] <- 4)}
        if(INFdat$headage[i]==5){(strataY[i] <- 5)}
        if(INFdat$headage[i]==6){(strataY[i] <- 6)}
        u[i] <- runif(1)
        INFdat$Y[i] <- ifelse(u[i]<INFdat$Y[i],1,0)
    }
    dat <- cbind(strataY,INFdat)
    return(dat)
}

# Construction of sample weights
sam.fn <- function(pop,str,N.h,n.h,loops){
    strY  <-  strata(pop,stratanames=c("strataY"),
              size=n.h,method="srswor")
    sam  <-  pop[strY$ID_unit,]
    s    <- as.vector(sam$str)
    weight <- c(); scalW <- c()
    for(i in 1:sum(n.h)){
        if(s[i]==1) {(weight[i]<-N.h[6]/n.h[6])} else
        if(s[i]==2) {(weight[i]<-N.h[2]/n.h[2])} else
        if(s[i]==3) {(weight[i]<-N.h[5]/n.h[5])} else
        if(s[i]==4) {(weight[i]<-N.h[4]/n.h[4])} else
        if(s[i]==5) {(weight[i]<-N.h[3]/n.h[3])} else
                    {(weight[i]<-N.h[1]/n.h[1])}
    }
```

```
    sam <- as.data.frame(cbind(weight,scalW,sam))
    return(sam)
}

# Stratification
Nh <- c(2355,371,608,641,522,120)
nh <- c(50,150,100,75,125,100)
N  <- sum(Nh)
n <- sum(nh)

# Replication number
loops <- 10000

# Number of regression coefficient parameters
numCoef   <- nrow(coef.pop)

coefNames <- c("(Intercept)","headage2","headage3",
               "headage4","headage5","headage6",
               "headsex2","headocc2","headocc3",
               "headocc4","headed2","headed3",
               "headed4","region2","region3","hhtype2")
uCoef1    <- matrix(0,loops,numCoef);colnames(uCoef1) <-coefNames
wCoef1    <- matrix(0,loops,numCoef);colnames(wCoef1) <-coefNames
uCoef     <- matrix(0,loops,numCoef);colnames(uCoef)  <-coefNames
wCoef     <- matrix(0,loops,numCoef);colnames(wCoef)  <-coefNames
uSE       <- matrix(0,loops,numCoef);colnames(uSE)    <-coefNames
wSE       <- matrix(0,loops,numCoef);colnames(wSE)    <-coefNames
noCoefs   <- matrix(0,loops,2);colnames(noCoefs)<- c("p1","p2")

# For CV
avgpe.u  <- c()
avgpe.w  <- c()
avgpew.u <- c()
avgpew.w <- c()

# All results
result <- matrix(0,loops,10)
colnames(result)<- c("case","vVu","vVw","Vu","Vw","smVu",
                     "smVw","uCV","wCV","Vote")

# Simulation performance
set.seed(10000)
for(i in 1:loops){

    # Generate finite population data and select a sample
    Pop <- pop.fn(Rdata,i)
    Sam <- sam.fn(Pop,strataY,Nh,nh,i)
```

```
# Dummy variables
headsex2  <- (Sam$headsex==2)*1
headage2  <- (Sam$headage==2)*1
headage3  <- (Sam$headage==3)*1
headage4  <- (Sam$headage==4)*1
headage5  <- (Sam$headage==5)*1
headage6  <- (Sam$headage==6)*1
headocc2  <- (Sam$headocc==2)*1
headocc3  <- (Sam$headocc==3)*1
headocc4  <- (Sam$headocc==4)*1
headed2   <- (Sam$headed==2)*1
headed3   <- (Sam$headed==3)*1
headed4   <- (Sam$headed==4)*1
religion2 <- (Sam$religion==2)*1
religion3 <- (Sam$religion==3)*1
region2   <- (Sam$region==2)*1
region3   <- (Sam$region==3)*1
hhtype2   <- (Sam$hhtype==2)*1

Sam <- cbind(Sam,headage2,headage3,headage4,
    headage5,headage6,headsex2,headocc2,headocc3,
    headocc4,headed2,headed3,headed4,religion2,
    religion3,region2,region3,hhtype2)

# Design identification
desSam <- svydesign(id=~1,strata=~strataY,
          weights=~weight,data=Sam)

# Model fit and model choice
uMod1 <- glm(Y~headage2+headage3+headage4+headage5+
         headage6+headsex2+headocc2+headocc3+headocc4+
         headed2+headed3+headed4+region2+region3+hhtype2,
         family=quasibinomial,Sam)
wMod1 <-  svyglm(Y~headage2+headage3+headage4+headage5+
         headage6+headsex2+headocc2+headocc3+headocc4+
         headed2+headed3+headed4+region2+region3+hhtype2,
         family=quasibinomial,design=desSam)
uMod2 <- uSelect(uMod1,Sam)
wMod2 <- wSelect(wMod1,desSam)

# Store estimators
uCoef1[i,] <- coefStore(numCoef,uMod1$coefficients,coefNames)
wCoef1[i,] <- coefStore(numCoef,wMod1$coefficients,coefNames)
uCoef[i,]  <- coefStore(numCoef,uMod2$coefficients,coefNames)
wCoef[i,]  <- coefStore(numCoef,wMod2$coefficients,coefNames)

uSE[i,]    <- coefStore(numCoef,SE(uMod2),coefNames)
wSE[i,]    <- coefStore(numCoef,SE(wMod2),coefNames)
```

```
    noCoefs[i,1] <- length(uMod2$coefficients)
    noCoefs[i,2] <- length(wMod2$coefficients)

    # Model choice decision
    uName <- names(uMod2$coefficients)
    wName <- names(wMod2$coefficients)

    if(setequal(uName,wName)==TRUE)
        {result[i,] <- c(1,9,9,9,9,9,9,9,9,9)}
    else{
        # Vuong
        result[i,1] <- 2
        result[i,2] <- Vtest(uMod2,wMod2,Sam)
        result[i,3] <- wVtest(uMod2,wMod2,Sam)
        result[i,4] <- Vchoice(result[i,2])
        result[i,5] <- Vchoice(result[i,3])
        result[i,6] <- smVchoice(result[i,4],
                       noCoefs[i,1],noCoefs[i,2])
        result[i,7] <- smVchoice(result[i,5],
                       noCoefs[i,1],noCoefs[i,2])

        # CV
        result[i,8:9] <- cvOperation(uMod2,wMod2,5,Sam,i)

        # Voting system
        result[i,10] <- voting(sum(result[i,6:9]),i)
    }
    print(round(c(i,result[i,]),2))
}  # End loop

# Evaluatig the results only in case of un-equivalent regressors
idx     <- which(result[,1]==2)
uCoef2  <- uCoef[idx,]
wCoef2  <- wCoef[idx,]
uSE2    <- uSE[idx,]
wSE2    <- wSE[idx,]
result2 <- result[idx,]

# Printing the results
paste("Vuong.NW"); table(result2[,4])
paste("Vuong.W"); table(result2[,5])
paste("CV.NW"); table(result2[,8])
paste("CV.W"); table(result2[,9])
paste("Voting System"); table(result2[,10])
allResults2 <- printResults(coef.pop,numCoef,result2,
               uCoef2,wCoef2,uSE2,wSE2,noCoefs)
```

B.2.2 Simulation of Missing Stratification Information

Functions used for creating finite population data, selecting a sample and operating CV are the same as the functions employed in the NIS in B.1.1.

```
rm(list = ls())
# Load packages
library(MASS)
library(lpSolve)
library(sampling)
library(survey)
library(lattice)

# Regression coefficient parameters of the target population model
intercept<- -1
coefZ    <- as.matrix(c(3, 3, 3, 3, 3))
coefX    <- as.matrix(c(0.5, -0.5, -1, 0.5, 0.5,
            1, 2, 0.5, 0.5, 0.5))
coefXZ   <- as.matrix(c(intercept,coefZ,coefX))
coef.pop <- as.matrix(c(intercept,coefX))

# Import the real data set (INFHS)
hh.data  <-  as.data.frame(read.table("E:/ForPreeya/nfhs2.txt",
             header=T, sep = "\t", fill = T))

# Delete unused variables
Rdata <- hh.data[,-c(1:3,8,12:16)]

# Deal missing data (9) with the mode
nr <- nrow(Rdata)
for(i in 1:nr){
    if(Rdata$headage[i]==9) {Rdata$headage[i]<-6}
    if(Rdata$headocc[i]==9) {Rdata$headocc[i]<-3}
    if(Rdata$religion[i]==9){Rdata$religion[i]<-1}
}

# Dummy variables
attach(Rdata)
reg2    <- (region==2)*1
reg3    <- (region==3)*1
hht2    <- (hhtype==2)*1
hdsex2  <- (headsex==2)*1
hdage2  <- (headage==2)*1
hdage3  <- (headage==3)*1
hdage4  <- (headage==4)*1
hdage5  <- (headage==5)*1
hdage6  <- (headage==6)*1
hdocc2  <- (headocc==2)*1
hdocc3  <- (headocc==3)*1
```

```
hdocc4  <- (headocc==4)*1
hdedu2  <- (headed==2)*1
hdedu3  <- (headed==3)*1
hdedu4  <- (headed==4)*1
rel2    <- (religion==2)*1
rel3    <- (religion==3)*1
detach(Rdata)

# Create Y
Nsp <- nrow(Rdata)
Y   <- 1/(1+(exp(-(coefXZ[1]+(coefXZ[2]*hdage2)+
       (coefXZ[3]*hdage3)+(coefXZ[4]*hdage4)+
       (coefXZ[5]*hdage5)+(coefXZ[6]*hdage6)+
       (coefXZ[7]*hdsex2)+(coefXZ[8]*hdocc2)+
       (coefXZ[9]*hdocc3)+(coefXZ[10]*hdocc4)+
       (coefXZ[11]*hdedu2)+(coefXZ[12]*hdedu3)+
       (coefXZ[13]*hdedu4)+(coefXZ[14]*reg2)+
       (coefXZ[15]*reg3)+(coefXZ[16]*hht2)))))

# The INFHS population data
Rdata <- cbind(Y,Rdata)

# Stratification
Nh  <- c(2355,371,608,641,522,120)
nh  <- c(50,150,100,75,125,100)
N   <- sum(Nh)
n   <- sum(nh)

# Replication number
loops   <- 10000

nPop    <- matrix(0,loops,2);colnames(nPop)<-c("N1","N0")

# Number of regression coefficient parameters
numCoef <- nrow(coef.pop)

coefNames<- c("(Intercept)","headsex2","headocc2",
             "headocc3","headocc4","headed2","headed3",
             "headed4","region2","region3","hhtype2")
uCoef1  <- matrix(0,loops,numCoef);colnames(uCoef1) <-coefNames
wCoef1  <- matrix(0,loops,numCoef);colnames(wCoef1) <-coefNames
uCoef   <- matrix(0,loops,numCoef);colnames(uCoef)  <-coefNames
wCoef   <- matrix(0,loops,numCoef);colnames(wCoef)  <-coefNames
uSE     <- matrix(0,loops,numCoef);colnames(uSE)    <-coefNames
wSE     <- matrix(0,loops,numCoef);colnames(wSE)    <-coefNames
noCoefs <- matrix(0,loops,2)
colnames(noCoefs)<- c("p1","p2")
```

```
# For CV
avgpe.u  <- c()
avgpe.w  <- c()
avgpew.u <- c()
avgpew.w <- c()

# All results
result <- matrix(0,loops,10)
colnames(result) <- c("case","vVu","vVw","Vu","Vw",
                      "smVu","smVw","uCV","wCV","Vote")

# Simulation performance
set.seed(10000)
for(i in 1:loops){

    # Generate finite population data and select a sample
    Pop <- pop.fn(Rdata,i)
    Sam <- sam.fn(Pop,strataY,Nh,nh,i)
    nPop[i,]<- table(Pop$Y)

    # Dummy variables
    headsex2  <- (Sam$headsex==2)*1
    headage2  <- (Sam$headage==2)*1
    headage3  <- (Sam$headage==3)*1
    headage4  <- (Sam$headage==4)*1
    headage5  <- (Sam$headage==5)*1
    headage6  <- (Sam$headage==6)*1
    headocc2  <- (Sam$headocc==2)*1
    headocc3  <- (Sam$headocc==3)*1
    headocc4  <- (Sam$headocc==4)*1
    headed2   <- (Sam$headed==2)*1
    headed3   <- (Sam$headed==3)*1
    headed4   <- (Sam$headed==4)*1
    religion2 <- (Sam$religion==2)*1
    religion3 <- (Sam$religion==3)*1
    region2   <- (Sam$region==2)*1
    region3   <- (Sam$region==3)*1
    hhtype2   <- (Sam$hhtype==2)*1
    Sam       <- cbind(Sam,headsex2,headocc2,headocc3,
                   headocc4,headed2,headed3,headed4,
                   religion2,religion3,
                   region2,region3,hhtype2)
    desSam    <- svydesign(id=~1,strata=~strataY,
                   weights=~weight,data=Sam)

    # Model fit and model choice
    uMod1  <- glm(Y~headsex2+headocc2+headocc3+headocc4+
              headed2+headed3+headed4+region2+region3+
```

```
            hhtype2,family=quasibinomial(),Sam)
    wMod1  <- svyglm(Y~headsex2+headocc2+headocc3+headocc4+
            headed2+headed3+headed4+region2+region3+
            hhtype2,family=quasibinomial(),design=desSam)
    uMod2  <- uSelect(uMod1,Sam)
    wMod2  <- wSelect(wMod1,desSam)

    # Store estimators
    uCoef1[i,]  <- coefStore(numCoef,uMod1$coefficients,coefNames)
    wCoef1[i,]  <- coefStore(numCoef,wMod1$coefficients,coefNames)
    uCoef[i,]   <- coefStore(numCoef,uMod2$coefficients,coefNames)
    wCoef[i,]   <- coefStore(numCoef,wMod2$coefficients,coefNames)

    uSE[i,]     <- coefStore(numCoef,SE(uMod2),coefNames)
    wSE[i,]     <- coefStore(numCoef,SE(wMod2),coefNames)
    noCoefs[i,1] <- length(uMod2$coefficients)
    noCoefs[i,2] <- length(wMod2$coefficients)

    # Model choice decision
    uName <- names(uMod2$coefficients)
    wName <- names(wMod2$coefficients)

    if(setequal(uName,wName)==TRUE)
        {result[i,] <- c(1,9,9,9,9,9,9,9,9,9)}
    else{
        # Vuong
        result[i,1] <- 2
        result[i,2] <- Vtest(uMod2,wMod2,Sam)
        result[i,3] <- wVtest(uMod2,wMod2,Sam)
        result[i,4] <- Vchoice(result[i,2])
        result[i,5] <- Vchoice(result[i,3])
        result[i,6] <- smVchoice(result[i,4],
                       noCoefs[i,1],noCoefs[i,2])
        result[i,7] <- smVchoice(result[i,5],
                       noCoefs[i,1],noCoefs[i,2])

        # CV
        result[i,8:9] <- cvOperation(uMod2,wMod2,5,Sam,i)

        # Voting system
        result[i,10] <- voting(sum(result[i,6:9]),i)
    }
    print(round(c(i,result[i,]),2))
}  # End of loops

# Evaluation the results only in case of
# un-equivalent regressors
idx     <- which(result[,1]==2)
```

```
uCoef2  <- uCoef[idx,]
wCoef2  <- wCoef[idx,]
uSE2    <- uSE[idx,]
wSE2    <- wSE[idx,]
result2 <- result[idx,]

# Print results
paste("Vuong.NW"); table(result2[,4])
paste("Vuong.W"); table(result2[,5])
paste("CV.NW"); table(result2[,8])
paste("CV.W"); table(result2[,9])
paste("Voting System"); table(result2[,10])
allResults2 <- printResults(coef.pop,numCoef,
               result2,uCoef2,wCoef2,
               uSE2,wSE2,noCoefs)
```

B.2.3 Simulation of Response-Based Sampling

```
rm(list = ls())
# Load packages
library(MASS)
library(lpSolve)
library(sampling)
library(survey)
library(lattice)

# Regression coefficient parameters of
# the target population model
intercept<- -1
coefZ    <- as.matrix(c(3, 3, 3, 3, 3))
coefX    <- as.matrix(c(0.5, -0.5, -1, 0.5,
            0.5, 1, 2, 0.5, 0.5, 0.5))
coefXZ   <- as.matrix(c(intercept,coefZ,coefX))
coef.pop <- as.matrix(c(intercept,coefZ,coefX))

# Import the real data set (INFHS)
hh.data  <-  as.data.frame(read.table("F:/ForPreeya/
             nfhs2.txt",header=T,sep="\t",fill = T))

# Delete unused variables
Rdata    <- hh.data[,-c(1:3,8,12:16)]

# Deal missing data (9) with the mode
nr <- nrow(Rdata)
for(i in 1:nr){
    if(Rdata$headage[i]==9) {Rdata$headage[i]<-6}
```

```
    if(Rdata$headocc[i]==9) {Rdata$headocc[i]<-3}
    if(Rdata$religion[i]==9){Rdata$religion[i]<-1}
}

# Dummy variables
attach(Rdata)
reg2    <- (region==2)*1
reg3    <- (region==3)*1
hht2    <- (hhtype==2)*1
hdsex2  <- (headsex==2)*1
hdage2  <- (headage==2)*1
hdage3  <- (headage==3)*1
hdage4  <- (headage==4)*1
hdage5  <- (headage==5)*1
hdage6  <- (headage==6)*1
hdocc2  <- (headocc==2)*1
hdocc3  <- (headocc==3)*1
hdocc4  <- (headocc==4)*1
hdedu2  <- (headed==2)*1
hdedu3  <- (headed==3)*1
hdedu4  <- (headed==4)*1
rel2    <- (religion==2)*1
rel3    <- (religion==3)*1
detach(Rdata)

# Creating Pr(Y=1|X)
Nsp <- nrow(Rdata)
PrY <- 1/(1+(exp(-(coefXZ[1]+(coefXZ[2]*hdage2)+
       (coefXZ[3]*hdage3)+(coefXZ[4]*hdage4)+
       (coefXZ[5]*hdage5)+(coefXZ[6]*hdage6)+
       (coefXZ[7]*hdsex2)+(coefXZ[8]*hdocc2)+
       (coefXZ[9]*hdocc3)+(coefXZ[10]*hdocc4)+
       (coefXZ[11]*hdedu2)+(coefXZ[12]*hdedu3)+
       (coefXZ[13]*hdedu4)+(coefXZ[14]*reg2)+
       (coefXZ[15]*reg3)+(coefXZ[16]*hht2)))))

# Finite Population data stratified
# by the target variable Y
pop.fn <- function(INFdat,N.h,n.h,loops){
    N  <- sum(N.h)
    n  <- sum(n.h)

    noStr   <- length(N.h)
    strataY <- rep(0,N)
    scalW   <- rep(0,N)
    count   <- rep(0,noStr)
    id <- 1
    while((count[1]<N.h[1])|(count[2]<N.h[2])){
```

```
        u <- runif(1)
        if((u<=PrY[id])&&(count[1]<N.h[1])){
            INFdat$Y[id] <- 1
            strataY[id] <- 1
            count[1] <- count[1] + 1
            id <- id + 1
        }else{if((u>PrY[id])&&(count[2]<N.h[2])){
            INFdat$Y[id] <- 0
            strataY[id] <- 2
            count[2] <- count[2] + 1
            id <- id + 1}
        }
    }
    pop <- cbind(strataY,INFdat)
    return(pop)
}

# Construction sample weights
sam.fn  <-  function(pop,str,N.h,n.h,loops){
    strY<-  strata(pop,stratanames=c("strataY"),
            size=n.h,method="srswor")
    sam <-  pop[strY$ID_unit,]
    N   <-  sum(N.h); n <- sum(n.h)
    s   <-  as.vector(sam$str)
    weight <- c()
    for(i in 1:sum(n.h)){
        if(s[i]==1) {(weight[i] <- N.h[1]/n.h[1])}
        else        {(weight[i] <- N.h[2]/n.h[2])}
    }
    sam <- as.data.frame(cbind(weight,scalW,sam))
    return(sam)
}

# Stratification
Nh  <- c(2000,2617)
nh  <- c(400,200)
N   <- sum(Nh)
n   <- sum(nh)

# Replication number
loops <- 10000

# Number of regression coefficient parameters
numCoef <- nrow(coef.pop)

coefNames <- c("(Intercept)","headage2","headage3",
              "headage4","headage5","headage6",
              "headsex2","headocc2","headocc3",
```

```
              "headocc4","headed2","headed3",
              "headed4","region2","region3","hhtype2")
uCoef1  <- matrix(0,loops,numCoef);colnames(uCoef1)<-coefNames
wCoef1  <- matrix(0,loops,numCoef);colnames(wCoef1)<-coefNames
uCoef   <- matrix(0,loops,numCoef);colnames(uCoef) <-coefNames
wCoef   <- matrix(0,loops,numCoef);colnames(wCoef) <-coefNames
uSE     <- matrix(0,loops,numCoef);colnames(uSE)   <-coefNames
wSE     <- matrix(0,loops,numCoef);colnames(wSE)   <-coefNames
noCoefs <- matrix(0,loops,2);colnames(noCoefs) <- c("p1","p2")

# For CV
avgpe.u  <- c()
avgpe.w  <- c()
avgpew.u <- c()
avgpew.w <- c()

# All results
result <- matrix(0,loops,10)
colnames(result) <- c("case","vVu","vVw","Vu","Vw",
                      "smVu","smVw","uCV","wCV","Vote")

# Simulation performance
set.seed(10000)
for(i in 1:loops){

    # Create finite population data and select a sample
    Pop <- pop.fn(Rdata,Nh,nh,i)
    Sam <- sam.fn(Pop,strataY,Nh,nh,i)

    # Dummy variables
    headsex2  <- (Sam$headsex==2)*1
    headage2  <- (Sam$headage==2)*1
    headage3  <- (Sam$headage==3)*1
    headage4  <- (Sam$headage==4)*1
    headage5  <- (Sam$headage==5)*1
    headage6  <- (Sam$headage==6)*1
    headocc2  <- (Sam$headocc==2)*1
    headocc3  <- (Sam$headocc==3)*1
    headocc4  <- (Sam$headocc==4)*1
    headed2   <- (Sam$headed==2)*1
    headed3   <- (Sam$headed==3)*1
    headed4   <- (Sam$headed==4)*1
    religion2 <- (Sam$religion==2)*1
    religion3 <- (Sam$religion==3)*1
    region2   <- (Sam$region==2)*1
    region3   <- (Sam$region==3)*1
    hhtype2   <- (Sam$hhtype==2)*1
```

```
Sam    <- cbind(Sam,headage2,headage3,headage4,headage5,
          headage6,headsex2,headocc2,headocc3,headocc4,
          headed2,headed3,headed4,religion2,religion3,
          region2,region3,hhtype2)

# Design identification
desSam <- svydesign(id=~1,strata=~strataY,
          weights=~weight,data=Sam)

# Model fit and model choice
uMod1 <- glm(Y~headage2+headage3+headage4+headage5+
         headage6+headsex2+headocc2+headocc3+headocc4+
         headed2+headed3+headed4+region2+region3+hhtype2,
         family=quasibinomial,Sam)
wMod1 <- svyglm(Y~headage2+headage3+headage4+headage5+
         headage6+headsex2+headocc2+headocc3+headocc4+
         headed2+headed3+headed4+region2+region3+hhtype2,
         family=quasibinomial,design=desSam)
uMod2 <- uSelect(uMod1,Sam)
wMod2 <- wSelect(wMod1,desSam)

# Store estimators
uCoef1[i,] <- coefStore(numCoef,uMod1$coefficients,coefNames)
wCoef1[i,] <- coefStore(numCoef,wMod1$coefficients,coefNames)
uCoef[i,]  <- coefStore(numCoef,uMod2$coefficients,coefNames)
wCoef[i,]  <- coefStore(numCoef,wMod2$coefficients,coefNames)

uSE[i,]    <- coefStore(numCoef,SE(uMod2),coefNames)
wSE[i,]    <- coefStore(numCoef,SE(wMod2),coefNames)
noCoefs[i,1] <- length(uMod2$coefficients)
noCoefs[i,2] <- length(wMod2$coefficients)

# Model choice decision
uName <- names(uMod2$coefficients)
wName <- names(wMod2$coefficients)

if(setequal(uName,wName)==TRUE)
    {result[i,] <- c(1,9,9,9,9,9,9,9,9,9)}
else{
    # Vuong
    result[i,1] <- 2
    result[i,2] <- Vtest(uMod2,wMod2,Sam)
    result[i,3] <- wVtest(uMod2,wMod2,Sam)
    result[i,4] <- Vchoice(result[i,2])
    result[i,5] <- Vchoice(result[i,3])
    result[i,6] <- smVchoice(result[i,4],
                   noCoefs[i,1],noCoefs[i,2])
    result[i,7] <- smVchoice(result[i,5],
```

```
                        noCoefs[i,1],noCoefs[i,2])

        # CV
        result[i,8:9] <- cvOperation(uMod2,wMod2,5,Sam,i)

        # Voting system
        result[i,10] <- voting(sum(result[i,6:9]),i)
    }
    print(round(c(i,result[i,]),2))
}  # End of loops

# Evaluation the results only in case of un-equivalent regressors
idx     <- which(result[,1]==2)
uCoef2  <- uCoef[idx,]
wCoef2  <- wCoef[idx,]
uSE2    <- uSE[idx,]
wSE2    <- wSE[idx,]
result2 <- result[idx,]

# Print results
paste("Vuong.NW"); table(result2[,4])
paste("Vuong.W") ; table(result2[,5])
paste("CV.NW")   ; table(result2[,8])
paste("CV.W")    ; table(result2[,9])
paste("Voting System"); table(result2[,10])
allResults2 <- printResults(coef.pop,numCoef,result2,
               uCoef2,wCoef2,uSE2,wSE2,noCoefs)
```

B.2.4 Simulation of Modeling Bias

```
library(survey)
library(MASS)
library(lpSolve)
library(sampling)

# Pop.model : logit Pr(y=1|X,Z) = 1 + x + z
para <- as.matrix(c(1,1,1))
N    <- 5000; n <- 1000
mu   <- matrix(c(0,0),2,1)
rho  <- 0.1
Sigma <- matrix(c(1,rho,rho,1),2,2)
set.seed(1000)
xz  <- mvrnorm(N,mu,Sigma,empirical=F)
x   <- I(xz[,1]>0)*1
z   <- I(xz[,2]>0)*1
n.x <-table(x)
```

```
n.xz<-table(x,z)

# Inverse logit function
invlogit <- function(x) {1/(1+exp(-x))}

# Fit logistic regression: E(Z|X) in population data
logis <- glm(z~x,family=binomial("logit"));summary(logis)
pred  <- fitted(logis);head(pred)
if(pred[1] < pred[2]) {(P0 <- pred[1]) && (P1 <- pred[2])}
if(pred[1] > pred[2]) {(P1 <- pred[1]) && (P0 <- pred[2])}

# f(x)
fx <- function(x) {
    1+((exp(para[3])-1)/(1+exp(para[1]+para[3]+(para[2]*x))))
    *(P0+((P1-P0)*x))}

# Generate population data
pop.fn  <- function(loops){
    prob<- invlogit(para[1] + para[2]*x + para[3]*z)
    u   <- runif(N,0,1)
    y   <- ifelse(u < prob,1,0)
    pop<<-cbind(y,x,z)
}

# Select a sample
samp.fn <- function(loops){
    Nh <- table(z); nh <- c(600,400)
    stratZ <- strata(pop,stratanames=c("z"),
              size=c(nh,nh),method="srswor")
    samp   <- as.data.frame(pop[stratZ$ID_unit,])
    weight <- c()
    for(i in 1:n){
        if(samp$z[i]==0) {weight[i] <- Nh[1]/nh[1]}
        else {weight[i] <- Nh[2]/nh[2]}
    }
    samp   <<- cbind(samp,weight)
    dessamp<<- svydesign(id=~1,strata=~z,weights=weight,data=samp)
}

# Simulation
N0   <- length(subset(x,x==0))
N1   <- length(subset(x,x==1))

noParas <- 2
sampColnames <- c("a.hat","b.hat")

loops   <- 1000
```

```
# Stored variables
uCoef   <- matrix(0,loops,noParas)
colnames(uCoef)<-sampColnames
wCoef   <- matrix(0,loops,noParas)
colnames(wCoef)<-sampColnames
sumwixi <- c()
uCoef   <- matrix(0,loops,noParas)
colnames(uCoef)<-sampColnames
wCoef   <- matrix(0,loops,noParas)
colnames(wCoef)<-sampColnames
results <- matrix(0,2,4)
colnames(results)<-c("au","bu","aw","bw")
rownames(results)<-c("P1","P2")
auXZ    <- c()
buXZ    <- c()
auX     <- c()
buX     <- c()
awXZ    <- c()
bwXZ    <- c()
awX     <- c()
bwX     <- c()

set.seed(10002)
for(i in 1:loops){
    # Generate finite population data and select a sample
    pop.fn(i)
    samp.fn(i)

    # Model fit
    ufit <- glm(y~x,family=quasibinomial,samp)
    wfit <- svyglm(y~x,family=quasibinomial,dessamp)

    # Store estimators
    uCoef[i,] <- coef(ufit)
    wCoef[i,] <- coef(wfit)

    # Expectation of Beta.hat
    auX[i]<- para[1]+sum(fx(samp$x)*invlogit(para[1]+
            (para[2]*samp$x))-invlogit(para[1]+
            (para[2]*samp$x)))/sum(para[1]+
            (para[2]*samp$x))
    buX[i]<- para[2]+sum(samp$x*(fx(samp$x)*
            invlogit(para[1]+(para[2]*samp$x))-
            invlogit(para[1]+(para[2]*samp$x))))/
            sum(samp$x^2*(para[1]+(para[2]*samp$x)))
    awX[i]<- para[1]+sum(samp$weight*(fx(samp$x)*
            invlogit(para[1]+(para[2]*samp$x))-
            invlogit(para[1]+(para[2]*samp$x))))/
```

```
            sum(samp$weight*(samp$x^2*(para[1]+
            (para[2]*samp$x))))
    bwX[i]<- para[2]+sum(samp$weight*samp$x*
            (fx(samp$x)*invlogit(para[1]+(para[2]*
            samp$x))-invlogit(para[1]+(para[2]*
            samp$x))))/sum(samp$weight*samp$x^2*
            (samp$x^2*(para[1]+(para[2]*samp$x))))
    print(i)
}

# Expectation of estimators
results[1,1] <- (1/loops)*sum(uCoef[,1]-para[1])
results[1,2] <- (1/loops)*sum(uCoef[,2]-para[2])
results[2,1] <- (1/loops)*sum(uCoef[,1]-auX)
results[2,2] <- (1/loops)*sum(uCoef[,2]-buX)
results[1,3] <- (1/loops)*sum(wCoef[,1]-para[1])
results[1,4] <- (1/loops)*sum(wCoef[,2]-para[2])
results[2,3] <- (1/loops)*sum(wCoef[,1]-awX)
results[2,4] <- (1/loops)*sum(wCoef[,2]-awX)
print(results)
```

B.3 Simulation of Sampling Ignorability

The following functions are shared for all three scenarios.

```
# Pesaran's test
PS <- function(mod,sample,loop){
    Xnw<- mod$model[,-1]
    Xnw<- as.matrix(cbind(rep(1,n),Xnw))
    wt <- sample$weight
    # Creating new.Xw==weight*Xnw
    XnwStar <- as.matrix(apply(Xnw,2,'*',wt))
    dataStar<- as.data.frame(cbind(sample$Y,XnwStar))
    fitStar <- lm("V1~headage2+headage3+headage4+
              headage5+headage6+headsex2+headocc2+
              headocc3+headocc4+headed2+headed3+
              headed4+region2+region3+hhtype2",
              data=dataStar)

    # Pesaran statistic
    p <- length(mod$coefficients)
    r1 <- as.matrix(mod$residuals)
    r2 <- as.matrix(fitStar$residuals)
    f1 <- as.matrix(mod$fitted.values)
    e1 <- (1/(n-p))*t(r1)%*%r1
    e2 <- (1/(n-p))*t(r2)%*%r2
    In <- as.matrix(diag(rep(1,n)))
```

```
    quaXnw <- Xnw%*%(solve(t(Xnw)%*%Xnw))%*%t(Xnw)
    quaXnwStar <- XnwStar%*%(solve(t(XnwStar)
                  %*%XnwStar))%*%t(XnwStar)

    A   <- In - (2*quaXnwStar%*%quaXnw) +
           (quaXnw%*%quaXnwStar%*%quaXnw)
    trA <- sum(diag(A))
    e21 <- (1/(n-p))*((t(f1)%*%A%*%f1)+(e1*trA))

    # Variance
    Mx    <- In - quaXnw
    Mstar <- In - quaXnwStar
    res21 <- Mstar%*%f1
    res211<- Mx%*%res21
    e211  <- t(res211)%*%res211

    # Statistic value
    Test <- (n/2)*log(e2/e21)
    Var  <- (e1/e21^2)*e211
    T    <- Test/sqrt(Var);T
    return(round(T,2))
}

# DuMouchel & Duncan's test
DD <- function(mod,sample,loop){
    # Creating matrix X & Z = WX
    # where W is a diagonal matrix of sample weights
    X <- as.data.frame(mod$model[,-1])
    W <- sample$weight
    Z <- as.matrix(apply(X,2,'*',W))
    colnames(Z)<-paste("newZ", 1:ncol(Z), sep = "")

    # Creating a new dataframe
    dataXY  <- as.data.frame(cbind(sample$Y,X))
    names(dataXY)[1]    <- paste("Y")
    dataXZ  <- as.data.frame(cbind(X,Z))
    dataYXZ <- as.data.frame(cbind(sample$Y,dataXZ))
    names(dataYXZ)[1]   <- paste("Y")

    # Creating "text" for fitting models
    nX      <- names(X)
    txtX    <- "Y~"
    for (i in 1:length(nX)){
        txtX <- paste(txtX,"+",nX[i])
    }
    v       <- names(dataXZ)
    txtXZ   <- "Y~"
    for (i in 1:length(v)){
```

```
        txtXZ <- paste(txtXZ,"+",v[i])
    }

    # Fit models
    fitX  <<- lm(txtX,dataXY)
    fitXZ <<- lm(txtXZ,dataYXZ)

    # Number of regression coefficients
    p   <- length(fitX$coefficients)

    # Number of sum of squares
    nSSfitX  <- nrow(anova(fitX)["Sum Sq"])
    nSSfitXZ <- nrow(anova(fitXZ)["Sum Sq"])

    # Calculations
    ssX  <<- sum(anova(fitX)["Sum Sq"]) -
             anova(fitX)["Sum Sq"][nSSfitX,]
    ssXZ <<- sum(anova(fitXZ)["Sum Sq"])-
             anova(fitXZ)["Sum Sq"][nSSfitXZ,]
    ssW  <<- ssXZ - ssX
    ssT  <<- sum(anova(fitX)["Sum Sq"])
    ssE  <<- ssT - ssX - ssW
    MSW  <<- ssW/p
    MSE  <<- ssE/(n-(2*p))

    # F-statistic value
    fW   <<- MSW/MSE

    # F-table value ; sig.level = 0.05
    alpha  <- 0.05
    Fw.tab <- qf(1-alpha, p, n-(2*p))

    result <- round(c(fW,Fw.tab),2)
    return(result)
}

# Evaluate estimators obtained from each test
printResults <- function(cfPop,noCfPop,finalEf,uCf,wCf,noCfs){
    # DuMouchel & Duncan's test
    endDD <- finalCoef(finalEf[,3],uCf,wCf,noCfPop)

    # Pesaran's test
    endPS <- finalCoef(finalEf[,2],uCf,wCf,noCfPop)

    # Evaluating final coefs of each method
    # Evaluating of direct choice
    uRB     <-RB(cfPop,noCfPop,uCf)
    wRB     <-RB(cfPop,noCfPop,wCf)
```

```
    uRV     <-RV(cfPop,noCfPop,uCf)
    wRV     <-RV(cfPop,noCfPop,wCf)
    uRRMSE  <-RRMSE(cfPop,noCfPop,uCf)
    wRRMSE  <-RRMSE(cfPop,noCfPop,wCf)

    # DuMouchel & Duncan
    RB.DD       <- RB(cfPop,noCfPop,endDD)
    RRMSE.DD    <- RRMSE(cfPop,noCfPop,endDD)
    RV.DD       <- RV(cfPop,noCfPop,endDD)

    # Pesaran
    RB.PS       <- RB(cfPop,noCfPop,endPS)
    RRMSE.PS    <- RRMSE(cfPop,noCfPop,endPS)
    RV.PS       <- RV(cfPop,noCfPop,endPS)

    # Results
    allRB       <- round(cbind(uRB,wRB,RB.DD,RB.PS),3)
    allRV       <- round(cbind(uRV,wRV,RV.DD,RV.PS),3)
    allRRMSE <- round(cbind(uRRMSE,wRRMSE,RRMSE.DD,RRMSE.PS),3)
    print(allRB);print(allRV);print(allRRMSE)
}
```

B.3.1 Simulation of Non-Informative Sampling

Here we note that functions 'pop.fn' and 'sam.fn' as employed in Subsection B.1.1 are again used for this case.

```
rm(list = ls(all = TRUE))
# Loading Packages
library(MASS)
library(lpSolve)
library(sampling)
library(survey)
library(lattice)

# Regression coefficient parameters of the target population model
intercept<- 2
coefZ    <- as.matrix(c(4, 4, 4, 4, 3))
coefX    <- as.matrix(c(1, -1, -2, 2, 1, 2 , 3, 1.5, 1.5, 1))
coefXZ   <- as.matrix(c(intercept,coefZ,coefX))
coef.pop <- as.matrix(c(intercept,coefZ,coefX))

# Import the real data set (INFHS)
hh.data  <-  as.data.frame(read.table("E:/ForPreeya/nfhs2.txt",
             header=T, sep = "\t", fill = T))

# Delete unused variables
```

```
Rdata <- hh.data[,-c(1:3,8,12:16)]

# Deal missing data (9) with the mode
nr <- nrow(Rdata)
for(i in 1:nr){
    if(Rdata$headage[i]==9){Rdata$headage[i]<-6}
    if(Rdata$headocc[i]==9){Rdata$headocc[i]<-3}
    if(Rdata$religion[i]==9){Rdata$religion[i]<-1}
}

# Dummy variables
attach(Rdata)
reg2   <- (region==2)*1
reg3   <- (region==3)*1
hht2   <- (hhtype==2)*1
hdsex2 <- (headsex==2)*1
hdage2 <- (headage==2)*1
hdage3 <- (headage==3)*1
hdage4 <- (headage==4)*1
hdage5 <- (headage==5)*1
hdage6 <- (headage==6)*1
hdocc2 <- (headocc==2)*1
hdocc3 <- (headocc==3)*1
hdocc4 <- (headocc==4)*1
hdedu2 <- (headed==2)*1
hdedu3 <- (headed==3)*1
hdedu4 <- (headed==4)*1
rel2   <- (religion==2)*1
rel3   <- (religion==3)*1
detach(Rdata)

# Create Y
Nsp <- nrow(Rdata)
Y   <- coefXZ[1]+(coefXZ[2]*hdage2)+(coefXZ[3]*hdage3)+
       (coefXZ[4]*hdage4)+(coefXZ[5]*hdage5)+
       (coefXZ[6]*hdage6)+(coefXZ[7]*hdsex2)+
       (coefXZ[8]*hdocc2)+(coefXZ[9]*hdocc3)+
       (coefXZ[10]*hdocc4)+(coefXZ[11]*hdedu2)+
       (coefXZ[12]*hdedu3)+(coefXZ[13]*hdedu4)+
       (coefXZ[14]*reg2)+(coefXZ[15]*reg3)+
       (coefXZ[16]*hht2)

# The INFHS population data
Rdata <- cbind(Y,Rdata)

# Stratification
Nh <- c(2355,371,608,641,522,120)
nh <- c(50,150,100,75,125,100)
```

```
N  <- sum(Nh)
n  <- sum(nh)

# Replication number
loops <- 10000

# Number of regression coefficient parameters
numCoef   <- nrow(coef.pop)
coefNames <- c("(Intercept)","headage2","headage3",
               "headage4","headage5","headage6",
               "headsex2","headocc2","headocc3",
               "headocc4","headed2","headed3",
               "headed4","region2","region3","hhtype2")

# Store estimators
uCoef1  <- matrix(0,loops,numCoef);colnames(uCoef1)<-coefNames
wCoef1  <- matrix(0,loops,numCoef);colnames(wCoef1)<-coefNames
uCoef   <- matrix(0,loops,numCoef);colnames(uCoef) <-coefNames
wCoef   <- matrix(0,loops,numCoef);colnames(wCoef) <-coefNames

# Final values obtained from the two tests
finalVal            <- matrix(0,loops,4)
colnames(finalVal)  <- c("case","PS","fCal","Ftable")

# Final results
finalEff            <- matrix(0,loops,3)
colnames(finalEff)  <- c("Case","Weffect.PS""Weffect.DD")

set.seed(10000)
for(i in 1:loops){

    # Generate finite population data and select a sample
    Pop <- pop.fn(Rdata,i)
    Sam <- sam.fn(Pop,strataY,Nh,nh,i)

    # Dummy variables
    headsex2  <- (Sam$headsex==2)*1
    headage2  <- (Sam$headage==2)*1
    headage3  <- (Sam$headage==3)*1
    headage4  <- (Sam$headage==4)*1
    headage5  <- (Sam$headage==5)*1
    headage6  <- (Sam$headage==6)*1
    headocc2  <- (Sam$headocc==2)*1
    headocc3  <- (Sam$headocc==3)*1
    headocc4  <- (Sam$headocc==4)*1
    headed2   <- (Sam$headed==2)*1
    headed3   <- (Sam$headed==3)*1
    headed4   <- (Sam$headed==4)*1
```

```
religion2 <- (Sam$religion==2)*1
religion3 <- (Sam$religion==3)*1
region2   <- (Sam$region==2)*1
region3   <- (Sam$region==3)*1
hhtype2   <- (Sam$hhtype==2)*1

Sam    <- cbind(Sam,headage2,headage3,headage4,
          headage5,headage6,headsex2,headocc2,
          headocc3,headocc4,headed2,headed3,
          headed4,religion2,religion3,region2,
          region3,hhtype2)

# Design identification
desSam <- svydesign(id=~1,strata=~strataY,
            weights=~weight,data=Sam)

# Model fit and model choice
uMod1  <- lm(Y~headage2+headage3+headage4+
           headage5+headage6+headsex2+headocc2+
           headocc3+headocc4+headed2+headed3+
           headed4+region2+region3+hhtype2,Sam)
wMod1 <- svyglm(Y~headage2+headage3+headage4+
           headage5+headage6+headsex2+headocc2+
           headocc3+headocc4+headed2+headed3+
           headed4+region2+region3+hhtype2,
           design=desSam)
uMod2   <- uSelect(uMod1,Sam)
wMod2   <- wSelect(wMod1,desSam)

# Store estimators
uCoef1[i,] <- coefStore(numCoef,uMod1$coefficients,coefNames)
wCoef1[i,] <- coefStore(numCoef,wMod1$coefficients,coefNames)
uCoef[i,]  <- coefStore(numCoef,uMod2$coefficients,coefNames)
wCoef[i,]  <- coefStore(numCoef,wMod2$coefficients,coefNames)

# Test effect of sample weights
uName <- names(uMod2$coefficients)
wName <- names(wMod2$coefficients)

if(setequal(uName,wName)==TRUE)
{
    finalVal[i,]  <- c(1,PS(uMod2,Sam,i),DD(uMod2,Sam,i))
    finalEff[i,1] <- finalVal[i,1] # The i.th case

    # 1==no weighted effect; 2==weighted effect exists
    finalEff[i,1] <- ifelse(abs(finalVal[i,2])
                     <=1.96,1,2)         # PS
    finalEff[i,2] <- ifelse(abs(finalVal[i,3])
```

```
                          <=finalVal[i,4],1,2)# DD
    }
}  # End of loop

# Evaluate the results only in case of
# equivalent regressors
idx     <- which(finalEff[,1]==1)
uCoef2  <- uCoef[idx,]
wCoef2  <- wCoef[idx,]
result  <- finalEff[idx,]

# Print results
table(result[,2])# The choice of PS
table(result[,3])# The choice of DD
allResults <- printResults(coef.pop,numCoef,
              result,uCoef2,wCoef2,noCoefs)
```

B.3.2 Simulation of Missing Stratification Information

Once again, we note that functions 'pop.fn' and 'sam.fn' as employed in Subsection B.1.1 are used for this case.

```
rm(list = ls(all = TRUE))
# Loading Packages
library(MASS)
library(lpSolve)
library(sampling)
library(survey)
library(lattice)

# Regression coefficient parameters of
# the target population model
intercept<- 2
coefZ    <- as.matrix(c(4, 4, 4, 4, 3))
coefX    <- as.matrix(c(1,-1,-2,2,1,2,3,1.5,1.5,1))
coefXZ   <- as.matrix(c(intercept,coefZ,coefX))
coef.pop <- as.matrix(c(intercept,coefX))

# Importing the real data set (INFHS)
hh.data <- as.data.frame(read.table("E:/ForPreeya/
           nfhs2.txt",header=T,sep="\t",fill = T))

# Deleting the useless variables
Rdata<- hh.data[,-c(1:3,8,12:16)]

# Dealing with missing data (9)
nr <- nrow(Rdata)
```

```
for(i in 1:nr){
    if(Rdata$headage[i]==9){Rdata$headage[i]<-6}
    if(Rdata$headocc[i]==9){Rdata$headocc[i]<-3}
    if(Rdata$religion[i]==9){Rdata$religion[i]<-1}
}

# Dummy variables
attach(Rdata)
reg2   <- (region==2)*1
reg3   <- (region==3)*1
hht2   <- (hhtype==2)*1
hdsex2 <- (headsex==2)*1
hdage2 <- (headage==2)*1
hdage3 <- (headage==3)*1
hdage4 <- (headage==4)*1
hdage5 <- (headage==5)*1
hdage6 <- (headage==6)*1
hdocc2 <- (headocc==2)*1
hdocc3 <- (headocc==3)*1
hdocc4 <- (headocc==4)*1
hdedu2 <- (headed==2)*1
hdedu3 <- (headed==3)*1
hdedu4 <- (headed==4)*1
rel2   <- (religion==2)*1
rel3   <- (religion==3)*1
detach(Rdata)

# Creat Y
Nsp <- nrow(Rdata)
Y   <- coefXZ[1]+(coefXZ[2]*hdage2)+(coefXZ[3]*hdage3)+
       (coefXZ[4]*hdage4)+(coefXZ[5]*hdage5)+
       (coefXZ[6]*hdage6)+(coefXZ[7]*hdsex2)+
       (coefXZ[8]*hdocc2)+(coefXZ[9]*hdocc3)+
       (coefXZ[10]*hdocc4)+(coefXZ[11]*hdedu2)+
       (coefXZ[12]*hdedu3)+(coefXZ[13]*hdedu4)+
       (coefXZ[14]*reg2)+(coefXZ[15]*reg3)+
       (coefXZ[16]*hht2)

# The INFHS population data
Rdata <- cbind(Y,Rdata)

# Stratification
Nh <- c(2355,371,608,641,522,120)
nh <- c(50,150,100,75,125,100)
N  <- sum(Nh)
n  <- sum(nh)

# Replication number
```

```
loops <- 10000

# Number regression coefficient parameters
numCoef   <- nrow(coef.pop)
coefNames <- c("(Intercept)","headsex2","headocc2",
               "headocc3","headocc4","headed2",
               "headed3","headed4","region2",
               "region3","hhtype2")
uCoef1  <- matrix(0,loops,numCoef);colnames(uCoef1)<- coefNames
wCoef1  <- matrix(0,loops,numCoef);colnames(wCoef1)<- coefNames
uCoef   <- matrix(0,loops,numCoef);colnames(uCoef) <- coefNames
wCoef   <- matrix(0,loops,numCoef);colnames(wCoef) <- coefNames

# Final values of two tests
finalVal <- matrix(0,loops,4)
colnames(finalVal) <- c("case","PS","fCal","Ftable")

# Final results of effect of sample weights
finalEff <- matrix(0,loops,3)
colnames(finalEff) <- c("Case","Weffect.PS","Weffect.DD")

# Simulation performance
set.seed(10000)
for(i in 1:loops){

    # Generate finite population data and select a sample
    Pop <- pop.fn(Rdata,i)
    Sam <- sam.fn(Pop,strataY,Nh,nh,i)

    # Dummy variables
    headsex2    <- (Sam$headsex==2)*1
    headage2    <- (Sam$headage==2)*1
    headage3    <- (Sam$headage==3)*1
    headage4    <- (Sam$headage==4)*1
    headage5    <- (Sam$headage==5)*1
    headage6    <- (Sam$headage==6)*1
    headocc2    <- (Sam$headocc==2)*1
    headocc3    <- (Sam$headocc==3)*1
    headocc4    <- (Sam$headocc==4)*1
    headed2     <- (Sam$headed==2)*1
    headed3     <- (Sam$headed==3)*1
    headed4     <- (Sam$headed==4)*1
    religion2   <- (Sam$religion==2)*1
    religion3   <- (Sam$religion==3)*1
    region2     <- (Sam$region==2)*1
    region3     <- (Sam$region==3)*1
    hhtype2     <- (Sam$hhtype==2)*1
```

```
    Sam    <- cbind(Sam,headage2,headage3,headage4,
              headage5,headage6,headsex2,headocc2,
              headocc3,headocc4,headed2,headed3,
              headed4,religion2,religion3,region2,
              region3,hhtype2)

    # Design identification
    desSam <- svydesign(id=~1,strata=~strataY,
              weights=~weight,data=Sam)

    # Model fit and model choice
    uMod1  <- lm(Y~headsex2+headocc2+headocc3+
               headocc4+headed2+headed3+headed4+
               region2+region3+hhtype2,Sam)
    wMod1   <- svyglm(Y~headsex2+headocc2+headocc3+
               headocc4+headed2+headed3+headed4+
               region2+region3+hhtype2,design=desSam)
    uMod2   <- uSelect(uMod1,Sam)
    wMod2   <- wSelect(wMod1,desSam)

    # Store estimators
    uCoef1[i,] <- coefStore(numCoef,uMod1$coefficients,coefNames)
    wCoef1[i,] <- coefStore(numCoef,wMod1$coefficients,coefNames)
    uCoef[i,]  <- coefStore(numCoef,uMod2$coefficients,coefNames)
    wCoef[i,]  <- coefStore(numCoef,wMod2$coefficients,coefNames)

    # Test effect of sample weights
    uName <- names(uMod2$coefficients)
    wName <- names(wMod2$coefficients)

    if(setequal(uName,wName)==TRUE)
    {
        finalVal[i,]<- c(1,PS(uMod2,Sam,i),DD(uMod2,Sam,i))
        finalEff[i,1]<-finalVal[i,1] # The i.th case
        # 1==no weighted effect; 2==weighted effect exists
        finalEff[i,2]<-ifelse(finalVal[i,2]
                        <1.96,1,2)         # PS
        finalEff[i,3]<-ifelse(finalVal[i,3]
                        <finalVal[i,4],1,2)# DD
    }
    print(c(i,finalEff[i,]))
}  # End of loop

# Evaluate the results only in case of
# equivalent regressors
idx     <- which(finalEff[,1]==1)
uCoef2  <- uCoef[idx,]
wCoef2  <- wCoef[idx,]
```

```
result  <- finalEff[idx,]

# Print results
table(result[,2])# The choice of PS
table(result[,3])# The choice of DD
allResults <- printResults(coef.pop,numCoef,
              result,uCoef2,wCoef2,noCoefs)
```

B.3.3 Simulation of Response-Based Sampling

Noting that functions 'pop.fn' and 'sam.fn' employed in this case are the same functions as used for the RBS in Subsection B.1.3.

```
rm(list = ls(all = TRUE))
# Loading Packages
library(MASS)
library(lpSolve)
library(sampling)
library(survey)
library(lattice)

# Regression coefficient parameters of
# the target population model
intercept<- 2
coefZ    <- as.matrix(c(4, 4, 4, 4, 3))
coefX    <- as.matrix(c(1,-1,-2,2,1,2,3,1.5,1.5,1))
coefXZ   <- as.matrix(c(intercept,coefZ,coefX))
coef.pop <- as.matrix(c(intercept,coefZ,coefX))

# Import the real data set (INFHS)
hh.data <- as.data.frame(read.table("E:/ForPreeya/
           nfhs2.txt",header=T,sep="\t",fill = T))

# Delete unused variables
Rdata <- hh.data[,-c(1:3,8,12:16)]

# Deal missing data (9) with the mode
nr <- nrow(Rdata)
for(i in 1:nr){
    if(Rdata$headage[i]==9) {Rdata$headage[i]<-6}
    if(Rdata$headocc[i]==9) {Rdata$headocc[i]<-3}
    if(Rdata$religion[i]==9){Rdata$religion[i]<-1}
}

# Dummy variables
attach(Rdata)
reg2   <- (region==2)*1
```

```
reg3   <- (region==3)*1
hht2   <- (hhtype==2)*1
hdsex2 <- (headsex==2)*1
hdage2 <- (headage==2)*1
hdage3 <- (headage==3)*1
hdage4 <- (headage==4)*1
hdage5 <- (headage==5)*1
hdage6 <- (headage==6)*1
hdocc2 <- (headocc==2)*1
hdocc3 <- (headocc==3)*1
hdocc4 <- (headocc==4)*1
hdedu2 <- (headed==2)*1
hdedu3 <- (headed==3)*1
hdedu4 <- (headed==4)*1
rel2   <- (religion==2)*1
rel3   <- (religion==3)*1
detach(Rdata)

# Create Y
Nsp <- nrow(Rdata)
Y   <- coefXZ[1]+(coefXZ[2]*hdage2)+(coefXZ[3]*hdage3)+
       (coefXZ[4]*hdage4)+(coefXZ[5]*hdage5)+
       (coefXZ[6]*hdage6)+(coefXZ[7]*hdsex2)+
       (coefXZ[8]*hdocc2)+(coefXZ[9]*hdocc3)+
       (coefXZ[10]*hdocc4)+(coefXZ[11]*hdedu2)+
       (coefXZ[12]*hdedu3)+(coefXZ[13]*hdedu4)+
       (coefXZ[14]*reg2)+(coefXZ[15]*reg3)+
       (coefXZ[16]*hht2)

# The INFHS population data
Rdata <- cbind(Y,Rdata)

# Stratification
Nh  <- c(1500,1300,1200,617)
nh  <- c(50,100,200,250)
N   <- sum(Nh)
n   <- sum(nh)

# Replication number
loops <- 10000

# Number of regression coefficient parameters
numCoef   <- nrow(coef.pop)
coefNames <- c("(Intercept)","headage2","headage3",
               "headage4","headage5","headage6",
               "headsex2","headocc2","headocc3",
               "headocc4","headed2","headed3",
               "headed4","region2","region3","hhtype2")
```

```
# Storing estimators
uCoef1  <- matrix(0,loops,numCoef);colnames(uCoef1) <- coefNames
wCoef1  <- matrix(0,loops,numCoef);colnames(wCoef1) <- coefNames
uCoef   <- matrix(0,loops,numCoef);colnames(uCoef)  <- coefNames
wCoef   <- matrix(0,loops,numCoef);colnames(wCoef)  <- coefNames

# Final values of two tests
finalVal            <- matrix(0,loops,4)
colnames(finalVal)  <- c("case","PS","fCal","Ftable")

# Final results of effect of sample weights
finalEff            <- matrix(0,loops,3)
colnames(finalEff)  <- c("Case","Weffect.PS","Weffect.DD")

# Simulation performance
set.seed(10000)
for(i in 1:loops){

    # Generate finite population data and select a sample
    Pop <- pop.fn(Rdata,Nh,i)
    Sam <- sam.fn(Pop,strataY,Nh,nh,i)

    # Dummy variables
    headsex2    <- (Sam$headsex==2)*1
    headage2    <- (Sam$headage==2)*1
    headage3    <- (Sam$headage==3)*1
    headage4    <- (Sam$headage==4)*1
    headage5    <- (Sam$headage==5)*1
    headage6    <- (Sam$headage==6)*1
    headocc2    <- (Sam$headocc==2)*1
    headocc3    <- (Sam$headocc==3)*1
    headocc4    <- (Sam$headocc==4)*1
    headed2     <- (Sam$headed==2)*1
    headed3     <- (Sam$headed==3)*1
    headed4     <- (Sam$headed==4)*1
    religion2   <- (Sam$religion==2)*1
    religion3   <- (Sam$religion==3)*1
    region2     <- (Sam$region==2)*1
    region3     <- (Sam$region==3)*1
    hhtype2     <- (Sam$hhtype==2)*1

    Sam <- cbind(Sam,headage2,headage3,headage4,
           headage5,headage6,headsex2,headocc2,
           headocc3,headocc4,headed2,headed3,
           headed4,religion2,religion3,region2,
           region3,hhtype2)
```

```
    # Design identification
    desSam <- svydesign(id=~1,strata=~strataY,
              weights=~weight,data=Sam)

    # Model fit and model choice
    uMod1  <- lm(Y~headage2+headage3+headage4+
              headage5+headage6+headsex2+
              headocc2+headocc3+headocc4+
              headed2+headed3+headed4+region2+
              region3+hhtype2,Sam)
    wMod1  <- svyglm(Y~headage2+headage3+
              headage4+headage5+headage6+
              headsex2+headocc2+headocc3+
              headocc4+headed2+headed3+
              headed4+region2+region3+
              hhtype2,design=desSam)
    uMod2  <- uSelect(uMod1,Sam)
    wMod2  <- wSelect(wMod1,desSam)

    # Store estimators
    uCoef1[i,]  <- coefStore(numCoef,
                   uMod1$coefficients,coefNames)
    wCoef1[i,]  <- coefStore(numCoef,
                   wMod1$coefficients,coefNames)
    uCoef[i,]   <- coefStore(numCoef,
                   uMod2$coefficients,coefNames)
    wCoef[i,]   <- coefStore(numCoef,
                   wMod2$coefficients,coefNames)

    # Test effect of sample weights
    uName <- names(uMod2$coefficients)
    wName <- names(wMod2$coefficients)

    if(setequal(uName,wName)==TRUE)
    {
        finalVal[i,]<- c(1,PS(uMod2,Sam,i),
                         DD(uMod2,Sam,i))
        finalEff[i,1]<-finalVal[i,1]#The i.th case
        # 1==no weighted effect
        # 2==weighted effect exists
        finalEff[i,2]<-ifelse(finalVal[i,2]
                         <1.96,1,2) # PS
        finalEff[i,3]<-ifelse(finalVal[i,3]
                         <finalVal[i,4],1,2)# DD
    }
    print(c(i,finalEff[i,]))
}  # End of loop
```

```
# Evaluate the results only in case of
# equivalent regressors
idx     <- which(finalEff[,1]==1)
uCoef2  <- uCoef[idx,]
wCoef2  <- wCoef[idx,]
result  <- finalEff[idx,]

# Print results
table(result[,2])# The choice of PS
table(result[,3])# The choice of DD
allResults <- printResults(coef.pop,numCoef,
              result,uCoef2,wCoef2,noCoefs)
```

Bibliography

Anthony, M. & Holden, S. B. (1998). Cross-Validation for binary classification by real-valued functions: theoretical analysis. *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, 218 - 229. doi:10.1145/279943.279987

Arlot, S., & Celisse, A. (2010). A survey of cross-validation procedures for model selection. *Statistics Surveys, 4*(1), 40 - 79. doi:10.1214/09-SS054

Binder, D. A. (1983). On the variances of asymptotically normal estimatiors from complex surveys. *International Statistical Review, 51*(3), 279-292.

Binder, D. A., & Roberts, G. (2009). Design- and Model-Based Inference for Model Parameters. In D. Pfeffermann, & C. R. Rao (Eds.), *Handbook of statistics 29 Vol. 29B Sample Surveys: Inference and Analysis* (pp. 33-54). UK: North Holland.

Brewer, K. R. W. (1999). Design-based or prediction-based inference? Stratified random vs stratified balanced sampling. *International Statistical Review, 67*(1), 35-47.

Brewer, K. R. W, & Mellor, R.W. (1973). The effect of sample structure on analytical surveys. *Austratian Journal of Statistics, 15*(3), 145-152.

Brown, B. Wm. , & Hollander, M. (1977). *Statistics : A Biomedical Introduction*. New York: John Willey & Sons.

Everitt, B. S. (2002). *Cambridge Dictionary of Statistics*. West Nyack, NY, USA: Cambridge University Press.

Casella, G., & Berger, R. L. (2002). *Statistical Inference*. Pacific Grove, CA : Thomson Learning.

Chambers, R. L. (1996). Robust Case-Weighting for Multipurpose Establishment Surveys. *Journal of Official Statistics, 12*(1), 3-32.

Chambers, R. L. (2003). *Analysis of Complex Survey Data: Course Note*. Wollongong: University of Wollongong.

Chambers, R. L., Dorfman, A. H., & Sverchkov, M. Y. (2003). Nonparametric regression with complex survey data. In R. L. Chambers & C. J. Skinner (Eds.), *Analysis of Survey Data* (pp. 151-174). Chichester, UK: John Wiley & Sons.

Chambless, L. E., & Boyle, K. E. (1985). Maximum likelihood methods for complex sample data: Logistic regression and discrete proportional hazards models. *Communications in Statistics-Theory and Methods, 14*(6), 1377-1392.

Cumberland, W. G., & Royal, R. M. (1981). Prediction Models in Unequal Probability Sampling. *Journal of the Royal Statistical Society. Series B, 43*(3), 353-367.

Davidson, R., & MacKinnon, J. G. (1980). On a simple procedure for testing non-nested regression models. *Economics Letters(5)*(1), 45-48.

Dawid, A. P. (1979). Conditional independence in statistical theory. *Journal of Royal Statistical Society. Series B, 41*(1), 1-31.

Droge, B. (1999). Asymptotic optimality of full cross-validation for selecting linear regression models. *Statistics & Probability Letters, 44*(4), 351-357.

DuMouchel, W.H., & Duncan, G.L. (1983). Using sample survey weights in multiple regression analysis of stratified samples. *Journal of the American Statistical Association, 78*(383), 535-543.

Eideh, A. A. H, & Nathan, G. (2006). Fitting time series models for longitudinal data under informative sampling, *Journal of Statistical Planning and Inference, 136*(9), 3052-3069.

Faraway, J. J. (2005). *Linear model with R*. London : Chapman & Hall.

Feng, C. J., Yu, Z, Kingi, U., & Baig, M. P. (2005). Threefold vs. Fivefold Cross Validation in One-Hidden-Layer and Two-Hidden-Layer Predictive Neural Network Modeling of Machining Surface Roughness Data. *Journal of Manufacturing Systems, 24*(2), 93-107.

Fletcher, D., & Dixon, P. M. (2012). Modelling data from different sites, times or Studies: weighted vs. unweighted regression. *Methods in Ecology and Evoluation, 3*(1), 168-176.

Fuller, W. A. (1984). Least Squares and Related Analyses for Complex Survey Designs. *Survey Methodology, 10*(1), 97-125.

Geuna, S. (2000). Appreciating the Difference Between Design-Based and Model-Based Sampling Strategies in Quantitative Morphology of the Nervous System. *The Journal of Comparative Neurology, 427*(3), 333-339.

Granger, C. W. J., King, M. L., & White, H. (1995). Comments on testing economic theories and the use of model selection criteria. *Journal of Econometrics, 67*(1), 173-187. doi:10.1016/0304-4076(94)01632-A

Graubard, B. I., & Korn, E. L. (1993). Hypothesis Testing With Complex Survey Data: The Use of Classical Quadratic Test StatisticsWith Particular Reference to Regression Problems. *Journal of the American Statistical Association, 88*(422), 629-641.

Gregoire, T. G. (1998). Design-based and model-based inference in survey sampling: appreciating the difference. *Canadian Journal of Forest Research, 28*(10), 1429-1447.

Han, J., & Kamber, M. (2006). *Data Mining: Concepts and Techniques* (2nd ed.). San Francisco: Elsevier Science.

Hansen, M. H., Madow, W. G. & Tepping, B. J. (1983). An evaluation of model dependent and probability-sampling inferences in sample surveys. *Journal of the Anerican Statistical Association, 78*(384), 776-793.

Herzberg, A. M., & Tsukanov, A. V. (1986). A note on modifications of the jackknife criterion for model selection. *Utilitas Mathematics*, 29, 209-216.

Herringa, S. G, West, B. T & Berglund, P. A. (2010). *Applied survey data analysis*. Boca Raton: Taylor & Francis.

Holt, D., Smith, M. F., & Winter, P. D. (1980). Regression Analysis of Data from Complex Surveys. *Royal Statistical Society. Series A, 143*(4), 474-487.

Hosmer, D. W., & Lemeshow, S. (1989). *Applied logistic regression*. New York: John Willey & Sons.

Hossain, M. Z., & Bhatti, M. I. (2003). Recent Development in Econometric Analysis of Model Selection. *Managerial Finance, 29*(7), 90-108.

Iachan, R. (1984). Sampling strategies, robustness and efficiency: The state of the art. *International Statistical Review, 52*(2), 209-218.

Johnson,N.L & Kotz, S. (1970). *Continuous univariate distributions*. Hougton Mifflin; New York.

Kalton, G., & Piesse, A. (2011). Survey research methods in evaluation and case-control studies. *Statistics in Medicine, 26*(8), 1675-1687. doi:10.1002/sim.2796

Kantardzic, M. (2011). *Data Mining: Concepts, Models, Methods, and Algorithms*. Retrieved from http://ieeexplore.ieee.org.ezproxy.uow.edu.au/xpl/ebooks/bookPdfWithBanner.jsp?fileName=6105632.pdf &bkn=6105606

Kish, L. (1965). *Survey sampling*. London: Wiley.

Korn, E. L, & Graubard, B. I. (1995). Analysis of large health surveys: Accounting for the sampling design. *Journal of Royal Statistical Society. Series A (Statistic in Society), 158*(2), 263-295.

Kott, P. S (1991). A Model-Based Look at Linear Regression with Survey Data. *The American Statistician, 45*(2), 107-112.

Kullback, S., & Leibler, R. A. (1951). On Information and Sufficiency. *The Annals of Mathematical Statistics, 22*(1), 79-86.

Lahiri , P. (Eds.) (2001). *Model Selection*. Beachwood, Ohio: Institute of Mathematical Statistics.

Lee, E. S., Forthofer, R. N., & Lorimor, R. J. (1986). Analysis of Complex Sample Survey Data. *Sociological Methods & Research, 15*(1-2), 69-100. doi:10.1177/0049124186015001007

Li, J., & Valliant, R. (2011). Detecting Groups of Influential Observations in Linear Regression Using Survey Data Adapting The Forward Search Method. *Pakistan Journal of Statistics, 27*(4), 507-528.

Linhart, H., & Zucchini, W. (1986). *Model Selection*. New York: John Willey & Sons.

Little, R. J. (1991). Inference with survey weights. *Journal of Official Statistics, 7*(4), 405-424.

Little, R. J. (2004). To model or not to Model? Comparing modes of inference for finite population sampling. *Journal of the American Statistical Association, 99*(466), 546-556.

Lohr, S. L. (1999). *Sampling: Design and Analysis*. Pacific Grove, Calif.; London: Duxbury Press.

Lohr, S. L, & Liu, J. (1994). A Comparison of Weighted and Unweighted Analyses in the National Crime Victimization Survey. *Journal of Quantitative Criminology, 10*(4), 343-360.

Lumley, T. (2010). *Complex surveys: a guide to analysis using R*. New York: John Willey & Sons.

Molina, E. A., Smith, T. M. F, & Sugden, R. A. (2001). Modelling Overdispersion for Complex Survey Data. *International Statistical Review, 69*(3), 373-384.

Nedyalkova, D., & Tillė, Y. (2008). Optimal sampling and estimation strategies under the linear model. *Biometrika, 95*(3), 521-537.

Nordberg, L. (1989). Generalized Linear Modeling of Sample Survey Data. *Journal of Official Statistics, 5*(3), 223-239.

Pesaran, M. H. (1974). On the General Problem of Model Selection. *The Review of Economic Studies, 41*(2), 153-171.

Pesaran, M. H. (1987). Global and partial non-nested hypotheses and asymptotic local power. *Econometric Theory, 3*(1), 69-97.

Pfeffermann, D. (1993). The role of sampling weights when modeling survey data. *International Statistical Review, 61*(2), 317-337.

Pfeffermann, D. (1996). The use of sampling weights for survey data analysis. *Statistical Methods in Medical Research, 5*(3), 239-261.

Pfeffermann, D., Krieger, A. M., & Rinott, Y. (1998). Parametric distributions of complex survey data under informative probability sampling. *Statistica Sinica, 8*(4), 1087-1114.

Pfeffermann, D. (2009). Introduction to Part 6. In D. Pfeffermann, & C. R. Rao (Eds.), *Handbook of statistics 29 Vol. 29B Sample Surveys: Inference and Analysis* (pp. 423-453). UK: North Holland.

Pfeffermann, D., & Holmes, D. J. (1985). Robustness Considerations in the Choice of Methods of Inference for Regression Analysis of Survey Data. *Journal of the Royal Statistical Society. Series A, 148*(3), 468-278.

Pfeffermann, D., & Nathan, G. (1981). Regression Analysis of Data from a Cluster Sample. *Journal of the American Statistical Association, 76*(375), 681-689.

Pfeffermann, D., & Sverchkov, M. Y. (1999). Parametric and Semi-Parametric Estimation of Regression Models Fitted to Survey Data. *Sankhya, series B, 61*(1), 166-186.

Pfeffermann, D., & Sverchkov, M. Y. (2003). Fitting Generalized Linear Models Under Informative Sampling. In R. L. Chambers & C. J. Skinner (Eds.), *Analysis of Survey Data* (pp. 175-195). Chichester, UK: John Wiley & Sons.

Pfeffermann, D., & Sverchkov, M. (2009). Inference under Informative Sampling. In D. Pfeffermann, & C. R. Rao (Eds.), *Handbook of statistics 29 Vol. 29B Sample Surveys: Inference and Analysis* (pp. 455-487). UK: North Holland.

Picard, R. R., & Cook, R. D. (1984). Cross-Validation of Regression Models. *Journal of the American Statistical Association, 79*(387), 575-583.

Prentice, R. L., & Pyke, R. (1979). Logistic disease incidence models and case-control studies. *Biometrika, 66*(3), 403-411.

Reid, N. (2010). Likelihood inference. *Wiley Interdisciplinary Reviews: Computational Statistics, 2*(5), 517-525.

Reid, N. (2012). Likelihood inference in complex settings. *The Canadian Journal of Statistics, 40*(4), 731-744.

Reiter, J. P., Zanutto, E. L., & Hunter, L. W. (2005). Analytical Modeling in Complex Surveys of Work Practices. *Industrial and Labor Relations Review, 59*(1), 82-100.

Royall, R. M. (1970). On finite population sampling theory under certain linear regression models. *Biometrika, 57*(2), 377-387.

Royall, R. M. (1976). The linear least squares prediction approach to two-stage sampling. *Journal of the American Statistical Association, 71*(355), 657-664.

Royall, R. M. & Cumberland, W. G. (1981). The finite population linear regression estimator and estimators of its variance. *Journal of the American Statistical Association, 76*(376), 924-930.

Shao, J. (1993). Linear Model Selection by Cross-Validation. *Journal of the American Statistical Association, 88*(422), 486-494.

Smith, T. M. F. (1976). The foundations of survey dappling: A review. *Journal of the Royal Statistical Society. Series A, 139*(2), 183-204.

Smith, T. M. F. (1984). Present position and potential developments: Some personal views: Sample surveys. *Journal of the Royal Statistical Society. Series A, 147*(2), 208-221.

Sterba, S. K. (2009). Alternative model-based and design-based frameworks for inference from samples to populations: from polarization to integration. *Multivariate Behavioral Research, 44*(6), 711-740. doi:10.1080/00273170903333574

Stone, M. (1974). Cross-Validation Choice and Assessment of Statistical Predictions. *Journal of the American Statistical Association, 35*(2), 111-147.

Sugden, R. A., & Smith, T. M. F. (1984). Ignorable and Informative Designs in Surveys Sampling Inference. *Biometrika, 71*(3), 495-506.

Thompson, M. E. (1997). *Theory of sample surveys*. London; Melbourne: Chapman & Hall.

Van Loon, A. Jeanne M., Tijhuis, M., Picavet, H. Susan J., Surtees, Paul G., & Ormel, J. (2003). Survey Non-response in the Netherlands: Effects on Prevalence Estimates and Associations. *Annals of Epidemiology, 13*(2), 105-110. doi:10.1016/S1047-2797(02)00257-0

Vuong, Q. H. (1989). Likelihood Ratio Tests for Model Selection and Non-Nested Hypothese. *Econometrica, 57*(2), 307-333.

Wheeler, D. C, VanHorn, J. E., & Paskett, E. (2008). A Comparison of Design-Based and Model-Based Analysis of Sample Surveys in Geography. *The Professional Geographer, 60*(4), 466-477.

White, H. (1982). Maximum likelihood extimation of misspecified models. *Journal of Economics, 50*(1), 1-25.

Winship, C., & Radbill, L. (1994). Sampling Weights and Regression Analysis. *Sociological Methods and Research, 23*(2), 230-257. doi:10.1177/0049124194023002004

Witter, I. H., & Frank, E. (2000). *Data mining : practical machine learning tools and techniques with Java implementations*. San Francisco: Morgan Kaufmann.

Wu, Y. Y., & Fuller, A. (2005). Preliminary Testing Procedures for regression with survey samples. *In Proceedings of the Survey Research Method Section, American Statistical Association*, 3683-3688.

Zhang, P. (1993). Model selection via multifold cross validation. *The Annals of Statistics, 21*(1), 299-313.

Zucchini, W. (2000). An Introduction to Model Selection. *Journal of Mathematical Psychology, 44*(1), 41-61. doi:10.1006/jmps.1999.1276