CONFIDENTIALITY AND VARIANCE ESTIMATION IN COMPLEX SURVEYS

by

Wen Wilson Lu
M.Sc., Simon Fraser University, 2000

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
in the Department
of
Statistics and Actuarial Science

SIMON FRASER UNIVERSITY
August 2004

APPROVAL

Name: Wen Wilson Lu

Degree: Doctor of Philosophy

Title of thesis: Confidentiality and Variance Estimation in Complex Surveys

Examining Committee: Dr. Charmaine Dean
Chair

Dr. Randy R. Sitter,
Senior Supervisor

Dr. Boxin Tang

Dr. Derek Bingham

Dr. Joan Hu

Dr. Jiahua Chen,
External Examiner
Department of Statistics and Actuarial Science
University of Waterloo

Date Approved: _____

Abstract

A variance estimator in a large survey based on jackknife or balanced repeated replication typically requires a large number of replicates and replicate weights. Reducing the number of replicates has important advantages for computation and for limiting the risk of data disclosure in public use data files. In the first part of this thesis, we propose algorithms adapted from scheduling theory to reduce the number of replicates. The algorithms are simple and efficient and can be adapted to easily account for analytic domains. An important concern with combining strata is that the resulting variance estimators may be inconsistent. We establish conditions for the consistency of the variance estimators and give bounds on attained precision of the variance estimators that are linked to the consistency conditions. The algorithms are applied to both a real sample survey and to samples from simulated populations, and the algorithms perform very well in attaining variance estimators with precision levels close to the upper bounds.

Another important issue in survey sampling is the conflict of interest between information sharing and disclosure control. Statistical agencies routinely release microdata for public use with stratum and/or cluster indicators suppressed for confidentiality. For the purpose of variance estimation, pseudo-cluster indicators are sometimes produced for use in linearization methods or replication weights for use in resampling methods. If care is not taken these can be used to (partially) reconstruct the stratum and/or cluster indicators and thus inadvertently break confidentiality. In the second part of this thesis, we will demonstrate the dangers and adapt algorithms used from scheduling theory and elsewhere to attempt to reduce this danger.

Acknowledgments

I would like to take this chance to thank my supervisor Dr. Randy Sitter for his invaluable encouragement and guidance during the past six years. To me he is not only a supervisor but also a big brother and a good friend. Without his unselfish and generous support, I cannot imagine myself being where I am right now.

I would also like to thank two of Randy's former students, Changbao and Derek. Whenever I need any assistance, academic or in real life, they are always there for me. Thanks Changbao. Thanks Derek.

I thank all the faculty members in our department for consistently giving me good advice and supporting me.

I would like to thank Sylvia, Sadika and other staff members of the department for their kindness and help.

Many thanks to my friends and fellow students Simon, Michael, May, Monica, Jason L., Jason N., Jason S., Crystal, Chunfang, Sandy, Pritam, Steve, and many more.

I would like to thank Mike Brick, Sylvia Dohrmann, Leyla Mohadjer, Inho Park, Jay Clark and Lee Harding from Westat for their kind help.

And to my wife, I give my deepest love and appreciation. Thanks for all the years supporting me and putting up with me. Those are the best memories of my life.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

This thesis consists of two related problems raised in survey contexts. The first considers issues around replication-based variance estimates in stratified multi-stage sampling with many strata and/or many psu's. The methods are demonstrated and evaluated on the 1995 National Health Interview Survey (NHIS). The second considers the case where the number of strata and/or psu's are small and is motivated by issues encountered in variance estimation for the Health and Nutrition Examination Survey (NHANES). In both cases issues of performance of resulting variance estimates and issues of maintaining confidentiality in publically released micro-data files are of importance.

Replication is commonly used to estimate standard errors from complex surveys. In addition to having desirable statistical properties (Rao and Wu, 1985; Krewski and Rao, 1981; Shao and Tu, 1995), replication methods easily accommodate adjustments such as nonresponse, poststratification, raking, or generalized regression weighting adjustments, provided the replicate estimates can be computed from replicate weights. However, the full set of replicates for a large survey often numbers in the thousands, making the computing time required by end-users for some iterative procedures still of practical concern (e.g., see Cohen, 1997; Valliant, 1996; Rao and Shao, 1996). Thus, fast and easy methods for reducing the number of replicates without greatly sacrificing the performance of the resulting variance estimators are of practical interest. As these methods typically result in groups of primary sampling units (psu's) from

different strata being deleted (or not) together, they have the additional and often more important advantage of limiting data disclosure risks in public use data files (Yung, 1997).

We consider stratified multi-stage sampling with n_h psu's sampled from each stratum. Though the methods and algorithms are applicable to general n_h, we focus primarily on the common special case, $n_h = 2$, to simplify the presentation. The replication methods most appropriate for this common survey design are the jackknife and balanced repeated replication (BRR). However, a drawback is too many replicates are needed. Although several methods have been proposed to reduce the number of replicates for these schemes, the implementation is often *ad hoc* and not always direct. In this thesis, we adapt simple but fast algorithms from scheduling theory in parallel-processor computer networks to reduce the number of replicates and yield efficient variance estimators. We then extend these to account for estimators of key analytic domains.

McCarthy (1966) suggests partially balanced repeated replication (PBRR) as an efficient way of reducing the number of replicates in the BRR. Lee (1972, 1973) shows the efficiency of PBRR could be improved by ordering the strata before applying the partial balancing and proposes algorithms to do so (see Wolter, 1985, pp. 125-130). Kalton (1977) describes a combined strata technique that adds flexibility in the implementation of PBRR and also makes the method applicable to the jackknife. Rust (1986) further develops PBRR and the combined strata method and describes the relationship between them. Other methods include the grouped jackknife and the delete-d jackknife. These last two, however, are often more appropriate when a large number of psu's are sampled within each stratum (see Shao and Tu, 1995 for details).

Since methods such as PBRR often result in a loss in precision for the variance estimate, some practitioners may be reluctant to employ them. However, this concern can be unjustified if the proper method is applied, as we demonstrate in the sequel. The primary reason for estimating standard errors of survey estimates is to compute confidence intervals and tests of significance. For this purpose, a variance estimator should be independent of the estimator and have a central χ^2 distribution with either

a known number of degrees of freedom, r, or with r large enough so that the normal approximation is reasonable. Generally, 30 degrees of freedom will retain good efficiency for 90 or 95 percent confidence intervals. Thus, in a survey with a potentially large number of replicates based on jackknife or BRR, reducing the number of replicates and the degrees of freedom of the variance estimator may have a negligible effect on the width and coverage of resulting confidence intervals.

Rust (1986) uses the Satterthwaite approximation to propose the degrees of freedom for a variance estimator as $r = 2\mathrm{E}(v)/\mathrm{Var}(v)$, where v is a variance estimator, which in this setting is essentially determined by the variance of the variance estimator. In a stratified sample it depends on the relative size of the strata, the within-stratum variance, and the within-stratum kurtosis of the characteristic (see Hansen, Hurwitz and Madow, 1953, Ch. 10). For example, in a proportionate stratified simple random sample from a normally distributed population with equal stratum variances, the traditional approximation is that r equals the number of psu's minus the number of strata. However, deviations from these conditions can result in much lower values of r.

On the other hand, the loss in precision of the variance estimator may be substantial in some cases, especially if the method of reducing the number is not carefully considered (Valliant, 1996; Rao and Shao, 1996). Thus, guidance on appropriate methods of reducing the number of replicates is needed, and simple algorithms that enable replicate designers to avoid large reductions in r are especially important.

Reduced replicate variance estimators are most needed for public use files, where users with different computing facilities analyze the survey data and avoiding disclosure risks is an important issue. Domain estimates and contrasts of estimates across domains are often the primary focus of analysis and sometimes are the main reason the survey is conducted. A simple procedure for reducing the number of replicates is more of an issue for domains and the literature in this area is more limited (see Nixon et al., 1998; DiGaetano et al., 1998). The algorithms we present allow replicate designers to explicitly ensure the value of r for estimates from key domains are not substantially reduced.

In Chapter 2, we introduce basic notation in surveys and establish the framework

for stratified multi-stage sampling. Then we review two major approaches for variance estimation within our framework: linearization and replication, including jackknife, BRR and bootstrap. We also present procedures for obtaining modified jackknife or BRR estimators using some grouping schemes in an effort to reduce the number of replicates and hence improve computational efficiency.

In Chapter 3, we develop conditions under which the resulting combined strata variance estimators are consistent, propose algorithms to design reduced replicate schemes, without regard to domains. Some theoretical upper- and lower-bounds for attainable degrees of freedom are derived, examined and connected to the consistency of the resulting variance estimators and most proofs are provided.

In Chapter 4, we extend the algorithms proposed in Chapter 3 to handle the multi-domain situations. More theoretical evaluation on the attainable degrees of freedom is accomplished with regard to domains. We also apply the proposed algorithms to data from the 1995 NHIS and some artificial populations based upon it, and perform a limited simulation study.

In recent years, statistical agencies have noticed increasing demand from a variety of external users for the data they collect. Among the typical users are policy makers, who need up-to-date social and economic statistics to help them make key decisions, and academic researchers requiring more detailed data at the micro level to conduct their own statistical analyses. Unfortunately, the potential risk of disclosing individual information is real if little care has been taken towards confidentiality concerns whenever a data file is publicly released.

In Chapter 5, we review recent accomplishments on disclosure control in general and then examine the issue as it relates to variance estimation motivated by a real survey. We will develop a simple method for breaking confidentiality by using only the design and replicate weights, without knowledge of what replicate method was used, thus emphasizing the extent of the practical problem. In the case where there are enough strata and/or psu's to use the methods developed in Chapters 3 and 4, the resulting replicate weights do a good job of masking the strata and psu identifiers. However, in some cases there are not enough psu's to sacrifice degrees of freedom and new methods are needed. This is the problem faced in NHANES.

Motivated by this practical problem, we propose some algorithmic approaches in an effort to minimize the risk of disclosure. At the end of the chapter, we present an application of the proposed approaches to NHANES, including an altered/masked (to retain confidentiality) version of what was actually used in the recently released data. The methodology and programs are currently being used at Westat Inc.

Chapter 6 discusses future research related to extending the practical and theoretical developments in Chapters 3 and 4 to more complex survey designs and a broader class of estimators of interest. We go on to consider a generalization of the original problem and present some preliminary results on what could be termed as approximately balanced orthogonal multi-arrays (Sitter, 1993) or balanced bootstraps (see Davison, Hinkley, and Schechtman, 1986; Efron, 1990; Graham et al., 1990; Nigam and Rao, 1996).

Chapter 2

Replication Methods for Variance Estimation

Replication is commonly used to estimate standard errors from complex surveys. In addition to having desirable statistical properties (Rao and Wu, 1985; Krewski and Rao, 1981; Shao and Tu, 1995), replication methods easily accommodate adjustments such as nonresponse, poststratification, raking, or generalized regression weighting adjustments, provided the replicate estimates can be computed from replicate weights. However, the full set of replicates for a large survey often numbers in the thousands, making the computing time required by end-users for some iterative procedures still of practical concern (e.g., see Cohen, 1997; Valliant, 1996; Rao and Shao, 1996). Thus, fast and easy methods for reducing the number of replicates without greatly sacrificing the performance of the resulting variance estimators are of practical interest.

We consider stratified multi-stage sampling. To simplify presentation, we focus primarily on the common special case of two primary sampling units per stratum with discussion on the extension to the more general case. The replication methods most appropriate in this case are the jackknife and balanced repeated replication (BRR).

In this chapter, we introduce basic notation in surveys and establish the framework for stratified multi-stage sampling. Then we present two major approaches for variance estimation within our framework: linearization and replication, including the jackknife, BRR and bootstrap. We also present procedures for obtaining modified

jackknife or BRR estimators using some grouping scheme in an effort to reduce the number of replicates and hence improve computational efficiency.

2.1 Stratified Multi-Stage Sampling

In stratified multi-stage sampling, a finite population of N primary sampling units (psu's) or clusters, is partitioned into L nonoverlapping strata of $N_1, N_2, \cdots, N_L$ psu's, respectively, where $N_1 + \cdots + N_L = N$. Each psu consists of secondary sampling units (ssu's). In two-stage sampling, ssu's are ultimate units. More generally, each ssu consists of a set of ultimate units in multi-stage sampling. The total number of ultimate units within stratum h is $M_h = \sum_{i=1}^{N_h} N_{hi}$, where N_{hi} is the number of ultimate units in the i-th psu within stratum h. Denote $M = \sum_{h=1}^{L} M_h = \sum_{h=1}^{L} \sum_{i=1}^{N_h} N_{hi}$, the total number of ultimate units in the finite population.

2.1.1 Sampling Scheme

Suppose that in stratum h, n_h clusters are selected from the N_h clusters without replacement with inclusion probability π_{hi}, $\sum_{i=1}^{N_h} \pi_{hi} = n_h$. The total sample size at the psu level is $n = n_1 + \cdots + n_L$. Within each selected cluster, say the (hi)-th psu, n_{hi} ultimate units are selected from the N_{hi} ultimate units according to some sampling plan. Note that neither the number of stages nor the sampling plan used after the first stage of sampling is specified. The total number of sampled ultimate units is $m = m_1 + \cdots + m_L$, where $m_h = \sum_{i=1}^{n_h} n_{hi}$, the number of sampled ultimate units within stratum h. A selected sample of ultimate units is then denoted by the index set

$$\mathcal{S} = \{(hil) : l = 1, \cdots, n_{hi}; i = 1, \cdots, n_h; h = 1, \cdots, L\},$$

where the subscripts h, i and l refer to stratum, psu within stratum and ultimate unit within psu, respectively. Note that $\mathcal{S}$ is a subset of

$$\mathcal{U} = \{(hil) : l = 1, \cdots, N_{hi}; i = 1, \cdots, N_h; h = 1, \cdots, L\},$$

the index set of all ultimate units in the finite population.

2.1.2 Characteristic of Interest and Point Estimator

For a measurement, or possibly a vector of measurements, of some characteristic $\mathbf{y}$, let $\mathbf{y}_{hil}$ denote the value of $\mathbf{y}$ attached to an ultimate sample unit $(hil) \in \mathcal{S}$ and $\mathbf{Y}_{hil}$ the value attached to an ultimate population unit in $\mathcal{U}$, respectively. Note that the difference between the two indices is that the former one is random and the latter one is not even though they are displayed identically. Denote the stratified population mean of characteristic $\mathbf{y}$ as

$$\bar{\mathbf{Y}} = \frac{\mathbf{Y}}{M} = \frac{1}{M}\sum_{h=1}^{L}\sum_{i=1}^{N_h}\sum_{l=1}^{N_{hi}}\mathbf{Y}_{hil} = \frac{1}{M}\sum_{h=1}^{L}\sum_{i=1}^{N_h}\mathbf{Y}_{hi} = \sum_{h=1}^{L}W_h\bar{\mathbf{Y}}_h, \tag{2.1}$$

where $\mathbf{Y}_{hi} = \sum_{l}^{N_{hi}}\mathbf{Y}_{hil}$, $\bar{\mathbf{Y}}_h = \sum_{i}^{N_h}\mathbf{Y}_{hi}/M_h$, and $W_h = M_h/M$. Denote $\hat{\mathbf{Y}}_{hi}$ as an unbiased estimator of total $\mathbf{Y}_{hi}$ for the (hi)-th selected psu based on sampling at the second and subsequent stages. Then an unbiased estimator of the stratum total $\mathbf{Y}_h$ is given by $\hat{\mathbf{Y}}_h = \sum_{i=1}^{n_h}\hat{\mathbf{Y}}_{hi}/\pi_{hi}$. Thus, the unbiased estimator of $\bar{\mathbf{Y}}$ is given as

$$\begin{aligned}\bar{\mathbf{y}} &= \sum_{h=1}^{L}W_h\bar{\mathbf{y}}_h = \sum_{h=1}^{L}\sum_{i=1}^{n_h}W_h\mathbf{y}_{hi}/n_h \\ &= \sum_{(hil)\in\mathcal{S}}w_{hil}\mathbf{y}_{hil}/M = \hat{\mathbf{Y}}/M,\end{aligned} \tag{2.2}$$

where $\bar{\mathbf{y}}_h = \sum_{i=1}^{n_h}\mathbf{y}_{hi}/n_h$, $\mathbf{y}_{hi} = \hat{\mathbf{Y}}_{hi}/(M_h\pi_{hi}/n_h)$, $\hat{\mathbf{Y}} = \sum_{(hil)\in\mathcal{S}}w_{hil}\mathbf{y}_{hil}$, and w_{hil} is dependent on how $\hat{\mathbf{Y}}_{hi}$ is formed. In this thesis, we will typically consider the parameter of interest to be of the form $\theta = \varphi(\bar{\mathbf{Y}})$, expressed as a "smooth" function of the population mean $\bar{\mathbf{Y}}$, and a reasonable (approximately unbiased and consistent) estimator to be $\hat{\theta} = \varphi(\bar{\mathbf{y}})$.

2.1.3 Sampling Weights

In equation (2.2), we express the sample mean $\bar{\mathbf{y}}$ as a weighted average of the individual sampling units. The sampling weight w_{hil} can be thought of as the number of individuals in the population represented by the sampled unit (hil). Sampling weights are calculated as the inverse of selection probabilities and hence are determined by the

sampling design. Ideally, if every sampled unit has a response to the survey, the base weights, w_{hil}, would be sufficient to estimate the population values. However, every survey has some level of nonresponse and hence some sort of weighting adjustment is needed to account for the appearance of nonresponse. One should note that all methods for adjusting nonresponse are necessarily model-based with the assumption that the nonrespondents behave similar to the respondents or at least are related to respondents in some way. Typically, a nonresponse adjustment procedure consists of the following steps:

1) Determine what characteristics that are related to response propensity.

2) Separate the data into classes (or cells) defined by these characteristics.

3) Inflate the weights in each cell by the factor

$$\frac{\text{Total weight of all cases in the cell}}{\text{Total weight of responders in the cell}}.$$

In addition, it is quite common in surveys that certain demographic subgroups, such as age, sex and/or race, are over- or under-covered and poststratification is employed to control the weighted sample total to known population totals for these groups. The ultimate weights used in point estimation and variance estimation are expressed as a multiplication of base weights and different weighting adjustment factors.

2.1.4 Domain Estimation

In large surveys, it is sometimes desirable to obtain separate estimates of the same characteristic of interest for subpopulations, or domains. Suppose there are D domains. For $d = 1, \cdots, D$, let $\mathcal{U}_d$ be the subset of $\mathcal{U}$ that are in domain d with $\mathcal{U} = \bigcup_{d=1}^{D} \mathcal{U}_d$, and for a given sample $\mathcal{S}$, let $\mathcal{S}_d$ be the subset of $\mathcal{S}$ that are in domain d with $\mathcal{S} = \bigcup_{d=1}^{D} \mathcal{S}_d$. Let $M_{(d)}$ be the number of ultimate population units in $\mathcal{U}_d$ and $m_{(d)}$ be the number of ultimate sample units in $\mathcal{S}_d$. Suppose we are interested in estimating the mean of a scalar characteristic z for domain d, $\bar{Z}_d = \sum_{(hil)\in\mathcal{U}_d} z_{hil}/M_{(d)}$.

A natural estimator of $\bar{Z}_d$ would be

$$\bar{z}_d = \frac{\sum_{(hil)\in\mathcal{S}_d} w_{hil} z_{hil}}{\sum_{(hil)\in\mathcal{S}_d} w_{hil}}.$$

An equivalent way of estimating z for all domains is to, for a selected sample $\mathcal{S}$, introduce a new sequence of indicator variables $\mathbf{t} = (t_1, \cdots, t_D)$ and $\mathbf{x} = (x_1, \cdots, x_D)$, where

$$t_{hil(d)} = \begin{cases} 1 & \text{if sample unit } (hil) \in \mathcal{S}_d, \\ 0 & \text{if sample unit } (hil) \notin \mathcal{S}_d, \end{cases} \tag{2.3}$$

and $x_{hil(d)} = t_{hil(d)} z_{hil}$, for $d = 1, \cdots, D$. It is easy to see that $\bar{Z}_d$ is well estimated by the ratio estimator

$$\bar{z}_d \;=\; \frac{\sum_{(hil)\in\mathcal{S}} w_{hil} x_{hil(d)}/M}{\sum_{(hil)\in\mathcal{S}} w_{hil} t_{hil(d)}/M} = \frac{\bar{x}_d}{\bar{t}_d}.$$

Without loss of generality, we will use the vector notation for the characteristic of interest $\mathbf{y}$ to include both cases of multiple characteristics and domains throughout this thesis.

2.1.5 Variance Estimator of the Sample Mean

Though clusters are typically selected without replacement, for variance estimation it is still a common practice to treat the sample as if the first stage clusters are drawn with replacement. The incentive for this approximation is to considerably simplify the calculation, especially when unequal probability sampling is applied at the first stage. Generally this approximation leads to overestimation of variance, but the bias is negligible if the first stage sampling fractions n_h/N_h are small. Therefore, throughout this thesis, we will use the with replacement approximation when considering variance estimation.

The variance-covariance matrix of the stratified sample mean, $\bar{\mathbf{y}}$, is given by

$$V(\bar{\mathbf{y}}) = \sum_{h=1}^{L} W_h^2 \mathbf{\Gamma}_h / n_h,$$

where $\mathbf{\Gamma}_h = E(\mathbf{y}_{hi} - \bar{\mathbf{Y}}_h)(\mathbf{y}_{hi} - \bar{\mathbf{Y}}_h)'$ is the within-stratum variance of cluster totals. The usual unbiased estimator of $V(\bar{\mathbf{y}})$ is given by

$$v(\bar{\mathbf{y}}) = \sum_{h=1}^{L} W_h^2 \hat{\mathbf{\Gamma}}_h / n_h, \tag{2.4}$$

where $\hat{\mathbf{\Gamma}}_h = \sum_{i=1}^{n_h} (\mathbf{y}_{hi} - \bar{\mathbf{y}}_h)(\mathbf{y}_{hi} - \bar{\mathbf{y}}_h)'/(n_h - 1)$ is an unbiased estimator of $\mathbf{\Gamma}_h$.

Two approaches for obtaining the variance estimator of a nonlinear statistic $\hat{\theta} = \varphi(\bar{\mathbf{y}})$, linearization and replication, are introduced in the following sections.

2.1.6 Asymptotic Framework

So far we have only discussed the population with finite size, even though the total number of ultimate population units, M, may be very large. To be able to present some existing and proposed asymptotic results for the variance estimators, we constitute an asymptotic framework with the assumption that the finite population under study is embedded in a series of increasing finite populations $\mathcal{U}_k$, $k = 1, 2, \cdots$, drawn from an infinite superpopulation. This embedding can give us properties such as consistency and asymptotic normality. All population quantities L, M, N, N_h, N_{hi}, $\mathbf{Y}_{hil}$ and θ and sample quantities m, n, n_h, n_{hi} and $\mathbf{y}_{hil}$ depend on the population index k. As k increases, the corresponding finite population $\mathcal{U}_k$ becomes larger and larger and so do these population and sample quantities. Thus, as $k \to \infty$, we have $L_k \to \infty$, $N_k \to \infty$ and $n_k \to \infty$. However, for simplicity of presentation, k will be suppressed throughout this thesis and all limiting processes including $n, N \to \infty$ $(n/N \to a < 1)$ will be understood to be as the result of $k \to \infty$.

2.2 The Linearization Method

Using a Taylor Series expansion, we have

$$\hat{\theta} - \theta = \varphi(\bar{\mathbf{y}}) - \varphi(\bar{\mathbf{Y}}) \approx (\bar{\mathbf{y}} - \bar{\mathbf{Y}})' \nabla \varphi(\bar{\mathbf{Y}}),$$

where $\nabla\varphi(\mathbf{t}) = (\varphi_1(\mathbf{t}), \cdots, \varphi_p(\mathbf{t}))$, $\varphi_k(\mathbf{t}) = \partial\varphi(\mathbf{t})/\partial t_k$ with $\mathbf{t} = (t_1, \cdots, t_p)'$. This linear approximation leads to the well known "linearization" variance estimator (sometimes termed the delta-method)

$$v_L(\hat{\theta}) = \sum_{h=1}^{L} \nabla\varphi(\bar{\mathbf{y}})'\hat{\mathbf{\Gamma}}_h\nabla\varphi(\bar{\mathbf{y}})/n_h. \tag{2.5}$$

2.3 The Jackknife

With any form of replication, subsamples are repeatedly selected from the full sample, the statistic of interest is computed for each subsample, and the variability among the subsample or replicate estimates is used to estimate the variance of the full sample statistic.

The jackknife is first introduced by Quenouille (1949) as a method to estimate and consequently reduce the bias of an estimator. It has become a more valuable tool since Tukey (1958) demonstrated that the jackknife can also be used to construct variance estimators. For the jackknife in a stratified multi-stage sampling setting, we form $n = \sum_h n_h$ replicates. Each replicate is formed by deleting one of the n sampled psu's at a time, say the j-th psu within stratum k. In survey sampling, for the ease of implementing variance estimation, the deletion of a psu is often accomplished by creating a vector of adjusted weights for each replicate, called replicate weights. The (kj)-th replicate estimate $\hat{\theta}^{(kj)} = \varphi(\bar{\mathbf{y}}^{(kj)})$ and $\bar{\mathbf{y}}^{(kj)}$ are calculated as in (2.2) but with w_{hil} replaced by replicate weights $w_{hil(kj)} = b_{kj}w_{hil}$, where

$$b_{kj} = \begin{cases} 1 & \text{if } k \neq h, \\ 0 & \text{if } k = h \text{ and } j = i, \\ \frac{n_h}{n_h - 1} & \text{if } k = h \text{ but } j \neq i. \end{cases} \tag{2.6}$$

The usual jackknife estimator (JKn) of $V(\hat{\theta})$ is given by

$$v_{JKn}(\hat{\theta}) = \sum_{k=1}^{L} \frac{n_k - 1}{n_k} \sum_{j=1}^{n_k} (\hat{\theta}^{(kj)} - \hat{\theta})^2. \tag{2.7}$$

When $n_h = 2$ for all h, the jackknife variance estimator requires $2L$ replicates and simplifies to $v_{JKn} = \sum_{k=1}^{2L}(\hat{\theta}^{(k)} - \hat{\theta})^2$.

An alternative jackknife variance estimator that requires only L replicates is the *drop all but one* jackknife (Rust and Rao, 1996). Here only one replicate is formed from each stratum by randomly selecting one psu, say the j-th psu within stratum k, and deleting all other psu's from the stratum. We again calculate the replicate estimate $\hat{\theta}^{(k)} = \varphi(\bar{\mathbf{y}}^{(k)})$ and $\bar{\mathbf{y}}^{(k)}$ as in (2.2) but with w_{hil} replaced by

$$w_{hil(k)} = \begin{cases} w_{hil} & \text{if } k \neq h, \\ 0 & \text{if } k = h \text{ and } j = i, \\ n_h w_{hil} & \text{if } k = h \text{ but } j \neq i. \end{cases}$$

When $n_h = 2$, this jackknife, denoted as JK2, randomly deletes one in each stratum, yielding

$$v_{JK2}(\hat{\theta}) = \sum_{k=1}^{L}(\hat{\theta}^{(k)} - \hat{\theta})^2. \tag{2.8}$$

The asymptotic properties of jackknife variance estimators, along with the linearization and BRR variance estimators, for smooth functions of a sample mean have been established by Krewski and Rao (1981) and Rao and Wu (1988). However, it is well known that the usual *delete-1* jackknife variance estimator for sample quantiles is inconsistent.

In practice, it is quite common that a set of replicate weights will be generated by statistical agencies using some replication method and will then be released along with the whole sample and associated sampling weights in a public use microdata file. The motivation for supplying replicate weights is to make the end users not only be able to obtain the estimates for functions of characteristics they are interested in, but also have the freedom to estimate the associated variation with ease.

2.4 Balanced Repeated Replication

McCarthy (1966) introduced and developed the balanced repeated replication (BRR) method as a variance estimator in stratified sampling. To describe the basic idea of

the BRR, we assume that $n_h = 2$ for all h as this is originally where the BRR was applied. We then form a set of balanced half-samples that drop one sampled psu in each stratum simultaneously and double the weights on the remaining psu's. The set of R balanced half-samples can be defined by an $R \times L$ matrix $[\delta_h^r]$, $1 \leq r \leq R$ and $1 \leq h \leq L$, where

$$\delta_h^r = \begin{cases} +1 & \text{if the first sample psu in stratum } h \text{ is selected in the } r\text{-th half-sample,} \\ -1 & \text{otherwise,} \end{cases} \tag{2.9}$$

and $\sum_{r=1}^{R} \delta_h^r \delta_{h'}^r = 0$ for all $h \neq h' = 1, \cdots, L$. Such an $R \times L$ matrix can be obtained from any $R \times R$ Hadamard matrix given that R is a multiple of 4 and no less than $L + 1$. Based on $[\delta_h^r]_{R \times L}$, we calculate replicate estimate $\hat{\theta}^{(r)} = \varphi(\bar{\mathbf{y}}^{(r)})$ and $\bar{\mathbf{y}}^{(r)}$ as in (2.2) with w_{hil} replaced by

$$w_{hil(r)} = w_{hil}[1 + (-1)^{i+1}\delta_h^r],$$

where $i = 1, 2$, $h = 1, \cdots, L$ and $r = 1, \cdots, R$. The BRR estimator of $V(\hat{\theta})$ is given by

$$v_{BRR}(\hat{\theta}) = \frac{1}{R} \sum_{r=1}^{R} (\hat{\theta}^{(r)} - \hat{\theta})^2. \tag{2.10}$$

Extensions of the BRR for $n_h > 2$ and unequal exist but involve complicated constructions as the matrix needed must satisfy mixed-level orthogonality and balance constraints (see Gurney & Jewett, 1975; Gupta & Nigam, 1987; Wu, 1991; Sitter, 1993; and under the term *balanced bootstrap*, Nigam and Rao, 1996). It should be noted that, unlike the jackknife, the BRR variance estimator for sample quantiles is consistent (Shao and Wu, 1992).

2.5 Fay's BRR

A generalization of the BRR method, called Fay's BRR method (Dippo, Fay and Morganstein, 1984), can be similarly defined. By introducing the Fay's factor $0 \leq$

$\varepsilon < 1$, we calculate $\hat{\theta}^{(r)} = \varphi(\bar{\mathbf{y}}^{(r)})$ and $\bar{\mathbf{y}}^{(r)}$ as in (2.2) with w_{hil} replaced by

$$w_{hil(r)} = w_{hil}[1 + (-1)^{i+1}\delta_h^r(1-\epsilon)],$$

where δ_h^r is defined as in (2.9). Note that when ε equals zero we get back the regular BRR. The Fay's BRR estimator of $V(\hat{\theta})$ is given by

$$v_{BRR-F}(\hat{\theta}) = \frac{1}{R(1-\epsilon)^2}\sum_{r=1}^{R}(\hat{\theta}^{(r)} - \hat{\theta})^2. \tag{2.11}$$

2.6 The Bootstrap

Sitter (1992) summarizes various bootstrap methods for variance estimation in complex surveys (see also Gross 1980, Bickel and Freedman 1984, McCarthy and Snowden 1985, and Rao and Wu 1988). We only introduce the rescaling method (Rao and Wu, 1988) here. The basic idea of this method is to draw a resample vector with replacement from the original sample, rescale each resampled unit and apply the original estimator to the rescaled vector. For $b = 1, \cdots, B$, the steps for obtaining bootstrap replicate $\hat{\theta}^{(b)}$ are as follows:

1) For stratum h, randomly select m_h^* clusters from the n_h original sample clusters.

2) Let $m_{hi(b)}$ be the number of times the (hi)-th cluster is resampled for replicate b, where $\sum_i m_{hi(b)} = m_h^*$. Define the bootstrap weights for replicate b as

$$w_{hil(b)} = \left[\left\{1 - \left(\frac{m_h^*}{n_h - 1}\right)^{1/2}\right\} + \left(\frac{m_h^*}{n_h - 1}\right)^{1/2}\frac{n_h}{m_h^*}m_{hi(b)}\right]w_{hil}. \tag{2.12}$$

3) Calculate the b-th bootstrap estimate $\hat{\theta}^{(b)} = \varphi(\bar{\mathbf{y}}^{(b)})$ and $\bar{\mathbf{y}}^{(b)}$ as in (2.2) but with w_{hil} replaced by $w_{hil(b)}$.

The bootstrap variance estimator for $\hat{\theta}$ is then given by

$$v_{BOOT}(\hat{\theta}) = \frac{1}{B}\sum_{b=1}^{B}\left(\hat{\theta}^{(b)} - \hat{\theta}^{(\cdot)}\right)^2, \tag{2.13}$$

where $\hat{\theta}^{(\cdot)} = (1/B)\sum_b \hat{\theta}^{(b)}$ (or $\hat{\theta}$). Note that we usually control the resample size of m_h^* to be no greater than $n_h - 1$ so the corresponding bootstrap weights, $w_{hil(b)}$, will all be positive.

2.7 The Combined Strata Grouped Jackknife

The combined strata technique, among several methods intended to reduce computational cost when the number of strata is large, is proposed by Kalton (1977) and can be applied to either jackknife or BRR. For the combined strata grouped jackknife, the L strata are merged to form G combined strata according to some grouping scheme. Denote the index set for the g-th combined stratum as L_g, $g = 1, \cdots, G$. Again we assume that $n_h = 2$ for all h. We then form $2G$ replicates, with two replicates from each combined stratum. For combined stratum g, the first replicate is formed by deleting one sample psu from each stratum $h \in L_g$ simultaneously; the second replicate is formed by deleting the other half of the psu's. We then calculate the replicate estimate $\hat{\theta}^{(gj)} = \varphi(\bar{\mathbf{y}}^{(gj)})$ and $\bar{\mathbf{y}}^{(gj)}$ as in (2.2) but with w_{hil} replaced by

$$w_{hil(gj)} = \begin{cases} w_{hil} & \text{if } h \notin L_g, \\ 0 & \text{if } h \in L_g \text{ and } j = i, \\ 2w_{hil} & \text{if } h \in L_g \text{ but } j \neq i, \end{cases}$$

where $j = 1, 2$, $g = 1, \cdots, G$. Then the combined strata grouped jackknife estimator of $V(\hat{\theta})$ is given by

$$v_{cJKn}(\hat{\theta}) = \sum_{g=1}^{G} \sum_{j=1}^{2} (\hat{\theta}^{(gj)} - \hat{\theta})^2/2. \tag{2.14}$$

The *drop all but one* combined strata grouped jackknife in this case is easy to obtain by forming only one replicate from each combined stratum,

$$v_{cJK2}(\hat{\theta}) = \sum_{g=1}^{G} (\hat{\theta}^{(g)} - \hat{\theta})^2, \tag{2.15}$$

where $\hat{\theta}^{(g)} = \varphi(\bar{\mathbf{y}}^{(g)})$ and $\bar{\mathbf{y}}^{(g)}$ calculated as in (2.2) with w_{hil} replaced by

$$w_{hil(g)} = \begin{cases} w_{hil} & \text{if } h \notin L_g, \\ 0 & \text{if } h \in L_g \text{ and } j = i, \\ 2w_{hil} & \text{if } h \in L_g \text{ but } j \neq i. \end{cases}$$

There is little literature discussing the asymptotic properties for the combined strata grouped jackknife. In this thesis, we will establish its consistency under certain conditions.

2.8 The Combined Strata Grouped BRR

For the combined strata grouped BRR, similar to the combined strata grouped jackknife introduced in the previous section, the L strata are partitioned into G combined strata using some grouping scheme with the combined strata denoted as an index set L_g, $g = 1, \cdots, G$. We obtain an $R \times G$ matrix $[\delta_g^r]$ from columns of any $R \times R$ Hadamard matrix, provided R is a multiple of 4 and greater than G. We then use this matrix to form a set of R balanced replicates by defining

$$\delta_g^r = \begin{cases} +1 & \text{if the 1st psu in each stratum } h \in L_g \text{ is selected in the } r\text{-th replicate,} \\ -1 & \text{if the 2nd psu in each stratum } h \in L_g \text{ is selected in the } r\text{-th replicate} \end{cases}$$

and replacing sampling weights w_{hil} with replicate weights

$$w_{hil(r)} = \begin{cases} w_{hil}[1 + (-1)^i]\delta_g^r & \text{if } h \in L_g, \\ w_{hil}[1 + (-1)^{i+1}]\delta_g^r & \text{otherwise.} \end{cases}$$

We calculate replicate estimate $\hat{\theta}^{(r)} = \varphi(\bar{\mathbf{y}}^{(r)})$ and obtain the combined strata grouped BRR estimator of $V(\hat{\theta})$,

$$v_{cBRR}(\hat{\theta}) = \frac{1}{R}\sum_{r=1}^{R}(\hat{\theta}^{(r)} - \hat{\theta})^2. \tag{2.16}$$

In fact, the combined strata technique proposed by Kalton (1977) originates from the idea of an earlier approach, *partially balanced repeated replication* (PBRR) (McCarthy, 1966) for reducing the number of replicates with the use of BRR. When each group has the same size, Rust (1984) shows that for every PBRR procedure there is a combined strata BRR procedure with an identical pattern of half-sample assignment values and variance estimator. In this thesis, we consider the combined strata BRR as

it is a generalization which allows the possibility of different numbers of strata in each combined stratum and hence could potentially increase the precision of the resulting variance estimator for a given number of replicates.

2.9 Some Practical Considerations

Statistical agencies often release microdata along with sampling weights and replicate weights in the form showed in Table 2.1. This will become more important as we consider confidentiality and disclosure issues in subsequent chapters.

Table 2.1: Format for publicly released data

Sample Index	Variable Values	Sampling Weights	Replicate Weights			
1	$\mathbf{y_1}$	w_1	$w_{1(1)}$	$w_{1(2)}$	$\cdots$	$w_{1(R)}$
2	$\mathbf{y_2}$	w_2	$w_{2(1)}$	$w_{2(2)}$	$\cdots$	$w_{2(R)}$
	$\cdots$	$\cdots$		$\cdots\cdots$		
m	$\mathbf{y_m}$	w_m	$w_{m(1)}$	$w_{m(2)}$	$\cdots$	$w_{m(R)}$

For combined strata methods, either jackknife or BRR, the application is not as easy as described in the previous sections if the n_h are not equal. The only exception is JK2 as its *delete-all-but-one* mechanism makes it fairly easy to implement the procedure of simultaneously deleting psu's per stratum within the same group. A common scenario in real survey situations is that n_h's are unequal and/or greater than 2. If they are all even, we can randomly divide the n_h psu's in stratum h into $n_h/2$ groups with 2 psu's within each group and then apply the proposed combining strata methods by treating the constructed $n_h/2$ groups as variance strata to form replicates. Otherwise, this can be approximately done. In addition, there exist some elegant methods of forming balanced replicates for general n_h. Sitter (1993) proposes an approach to obtain balanced orthogonal multi-arrays in an effort to increase the

efficiency of BRR variance estimator.

A similar approach, currently used in practice for the unstratified multi-stage sampling, is to split each sample psu into two approximately equal-sized pseudo-psu's, and then group the obtained pseudo-psu's into variance strata with two pseudo-psu's within each stratum. The main purpose of this approach is to suppress the original psu indicators from the end users for confidentiality concerns, which might be severe if the number of original sample psu's is small.

For important psu's, such as the largest metropolitan areas, it is highly preferable to include them into any sample with certainty. This essentially means that for such a psu, the psu itself is a stratum and the ssu's are psu's. These are sometimes termed self-representing (SR) strata. In this case, the ssu's are randomly paired and the SR strata are treated as a larger set of 2-psu strata for variance estimation purposes.

The impact of either approach on variance estimators varies depending on what kind of sampling scheme is undertaken and which replication method is employed. But in general, we will restrict to $n_h = 2$ for simplicity of presentation.

Chapter 3

Algorithms for Grouping Schemes with Single Domain

In large surveys, reducing the number of replicates and hence associated degrees of freedom of the variance estimator may have negligible effect on the width and coverage of resulting confidence intervals provided the reduced number of degrees of freedom is still large enough so that the normal approximation is reasonable. Several methods have been proposed to reduce the number of replicates for these schemes. McCarthy (1966) suggests partially balanced repeated replication (PBRR). Kalton (1977) describes a combining strata technique that generalizes the PBRR and that also makes the method applicable to the jackknife. Other methods such as the grouping and the delete-d jackknife are appropriate when many psu's are sampled within a small number of strata (Shao and Tu, 1995). In practice, the methods used are often *ad hoc*.

The loss in precision of the combined strata variance estimator can be substantial in some cases (see Valliant, 1996; Rao and Shao, 1996). To examine this concern, Rust (1986) uses the Satterthwaite approximation to express the degrees of freedom for a variance estimator as $r = 2[\mathrm{E}(v)]^2/\mathrm{Var}(v)$, where v is the variance estimator. This is essentially determined by the variance of the variance estimator. The traditional approximation that r equals the number of psu's minus the number of strata can seriously overstate the value of r, especially when strata are combined (see section

3.1).

Reduced replicate variance estimators are most needed for public use files, where domain estimates and contrasts of estimates across domains are often the primary focus of analysis. The literature for this problem is more limited (see Nixon et al., 1998; DiGaetano et al., 1998). In this thesis, we first establish the consistency of variance estimators based on combining strata under certain conditions and then we adapt fast but simple algorithms from scheduling theory in parallel-processor computer networks to group strata so as to yield efficient and consistent combined strata variance estimators. We compare these to existing algorithms available for the simplest situation, and then extend the proposed algorithms to account for estimators of key analytic domains.

These methods typically result in groups of psu's from different strata being deleted (or not) together, and therefore have the additional and often more important advantage of limiting data disclosure risks in public use data files (Yung, 1997).

In this chapter, section 3.1 develops conditions under which the resulting combined strata variance estimators are consistent. Section 3.2 proposes algorithms to design reduced replicate schemes, without regard to domains. In section 3.3, some theoretical upper- and lower-bounds for attainable degrees of freedom are derived, examined and connected to the consistency of the resulting variance estimators, while most proofs are relegated to section 3.4.

3.1 Effective Degrees of Freedom and Consistency

Rust (1986) derives the variance of a combined strata jackknife variance estimator of a scalar linear statistic from a stratified sample with two psu's sampled with replacement in each stratum as

$$V[v_c(\bar{y})] = \frac{1}{8}\sum_{h=1}^{L} W_h^4 \sigma_h^4 (\beta_h - 3) + \frac{1}{2}\sum_{g=1}^{G}\left(\sum_{h\in L_g} W_h^2 \sigma_h^2\right)^2 , \tag{3.1}$$

where $\sigma_h^2 = \mathrm{E}(y_{hi} - \bar{Y}_h)^2$ and $\beta_h = \mathrm{E}(y_{hi} - \bar{Y}_h)^4/\sigma_h^4$ are respectively the population variance and kurtosis in stratum h for the cluster estimates. A combined stratum is

denoted by g and contains L_g original strata. In single stage samples, σ_h^2 and β_h are the population variance and kurtosis for the variable being estimated. In multi-stage samples, these quantities are the corresponding parameters of the distribution of the cluster estimators, and thus depend on the sample design within clusters as well as the element variance of the variable (see Kish, 1965; pp. 289-291). Based on (3.1), Rust (1986) suggests the following as an approximation to the effective degrees of freedom,

$$df[v_c(\bar{y})] \doteq \frac{2V^2(\bar{y})}{V[v_c(\bar{y})]} \doteq \frac{\left(\sum_{h=1}^{L} W_h^2\sigma_h^2\right)^2}{(1/4)\sum_{h=1}^{L} W_h^4\sigma_h^4(\beta_h-3)+\sum_{g=1}^{G}\left(\sum_{h\in L_g} W_h^2\sigma_h^2\right)^2}. \quad (3.2)$$

The accuracy of this approximation depends on the asymptotic normality of $\bar{y}$ and hence how well the distribution of its variance estimator $v_c(\bar{y})$, after appropriate scaling, can be approximated by a chi-square distribution. In addition, the normality of within-stratum cluster estimate y_{hi}, (corresponding to $\beta_h = 3$), will also help increase the overall attainable degrees of freedom in equation (3.2), as will be discussed in detail in section 3.3. Note that we mainly use $df[v_c(\bar{y})]$ as a reference measurement in an effort to demonstrate that the proposed algorithms outperform their competitors by a noticeable margin. Also, a moderate bias of $df[v_c(\bar{y})]$ will have very limited impact in the resulting confidence interval of $\bar{y}$.

In both (3.1) and (3.2),

$$\sum_{g=1}^{G}\left(\sum_{h\in L_g} W_h^2\sigma_h^2\right)^2 = \sum_{h=1}^{L} W_h^4\sigma_h^4 + \sum_{g=1}^{G}\sum_{h\neq h'\in L_g} W_h^2 W_{h'}^2\sigma_h^2\sigma_{h'}^2,$$

and the second term of the righthand side is the increase in the variance of the variance estimator due to combining strata. Lee (1972) points out that different PBRR schemes could affect the precision of the variance estimator if $a_h = W_h^2\sigma_h^2/n_h$ are not equal for all strata.

Rust (1984) and Nixon et al. (1998) discuss the properties of the combined strata variance estimator. In general, they claim any grouping procedure will perform reasonably and generate consistent variance estimators if any single combined stratum

is not dominantly larger than the others. We formalize their claims in Theorem 3.1 below. First, note that the consistency of the regular jackknife and BRR variance estimators are established in Krewski and Rao (1981) under regularity conditions:

C1) $\sum_{h=1}^{L} W_h E|\mathbf{y}_{hi} - \bar{\mathbf{Y}}_h|^{2+\delta} = O(1)$ for some $\delta > 0$;

C2) $\max_{1\le h\le L} n_h = O(1)$;

C3) $\max_{1\le h\le L} W_h = O(n^{-1})$;

C4) $\sum_{h=1}^{L} W_h \mathbf{\Gamma}_h^2/n_h \to \mathbf{\Gamma}$ (positive definite);

C5) $\bar{\mathbf{Y}} \to \boldsymbol{\mu} = (\mu_1, ..., \mu_p)'$; and

C6) The first derivative $\varphi_k(\cdot)$ of $\varphi(\cdot)$ are continuous in a neighborhood of $\boldsymbol{\mu}$.

Let $\mathbf{a}_h = (a_{h1}, ..., a_{hp})'$, where $a_{hk} = W_h^2\sigma_{hk}^2/n_h$ and σ_{hk}^2 is the k-th diagonal element of $\mathbf{\Gamma}_h$ and, for any given grouping scheme, let

$$S_{gk} = \sum_{h\in L_g} a_{hk} \quad \text{and} \quad \bar{S}_{\cdot k} = \sum_{g=1}^{G} S_{gk}/G = \sum_{h=1}^{L} a_{hk}/G.$$

We formalize the Nixon et al. (1998) claims in the following theorem.

Theorem 3.1 *Under conditions C1-C6, for any combining strata procedure satisfying*

$$\max_{1\le g\le G} S_{gk}/\bar{S}_{\cdot k} = O(1), \quad k = 1, \cdots, p, \tag{3.3}$$

the resulting combined strata grouped variance estimator of $\hat{\theta} = \varphi(\bar{\mathbf{y}})$, $v_c(\hat{\theta})$, *either for the jackknife or BRR, is consistent, i.e.,*

$$n\{v_c(\hat{\theta}) - V(\hat{\theta})\} \xrightarrow{p} 0 \quad (G \to \infty). \tag{3.4}$$

Proof. See section 3.4.

This theorem supports Rust (1984)'s deduction that an optimal combining procedure is the one that results in combined strata that are equal in terms of $S_{gk} = \sum_{h\in L_g} a_{hk}$ for $g = 1, ..., G$. In practice such groupings are done in some *ad hoc* fashion while attempting to balance multiple goals. However, there are three algorithms

given in Wolter (1985, pg. 128) applicable to the simplest situation of scalar y, a_h, and $\hat{\theta} = \bar{y}$, where S_{gk} are simplified as S_g. These algorithms restrict the search to groupings with approximately equal number of variance strata in each combined stratum while forming combined strata with approximately equal S_g. One should notice that although Theorem 3.1 itself accommodates the more complex multi-domain situation, it is not clear what kind of combining strategy is most beneficial in resulting effective degrees of freedom. We will discuss this issue in more detail in chapter 4.

Whether for the BRR or jackknife, one can rewrite the first algorithm in Wolter's book as

Semi-Ascending Order Arrangement(SAOA) Algorithm (Lee, 1972, 1973):

1. Arrange the L strata in ascending order of a_h.
2. Re-arrange the last $L/2$ ($(L-1)/2$ if L is odd) strata in descending order of a_h.
3. Form G groups of size $\eta = L/G$ as follows:
   assign stratum $g, g+G, ..., g+(\eta-1)G$ to the g-th group for $g = 1, ..., G$.

The other two algorithms are similar in spirit and will not be discussed further. The idea is simple minded. It rests on the principle that if one orders a_h from smallest to largest and groups by taking every G-th item then the total in each group will be similar.

All three procedures force each combined stratum to have an approximately equal number of original strata. Rust (1986) developed theory without this restriction and provided some practical guidelines for constructing combined strata to equalize the S_g, but did not pursue algorithmic development to complement. He made the practical recommendation to construct combined strata primarily based on the values of W_h, with some precautions. In making this recommendation he noted that W_h are known, whereas σ_h^2 usually cannot be estimated accurately and differ by variable. Furthermore, W_h often have greater relative variability than σ_h^2 and thus ensuring that $\sum_h W_h^2/n_h$ are nearly equal ensures $\sum_h W_h^2\sigma_{hk}^2/n_h$ are nearly equal for $k = 1, ..., p$. Rust also suggests that keeping the group sizes approximately equal makes the method

more robust for domain estimates and estimates for a variety of characteristics from the survey which were not considered when forming groups. Intuitively, avoiding combinations that place a large number of strata in one (or a few) combined strata should reduce the likelihood of having a very unequal distribution of the A_g for a particular domain or characteristic.

3.2 Algorithms Adapted from Scheduling Theory

The problem of strata grouping is essentially a maximization problem over possible groupings. If we let Ω be the set of all possible groupings of the L original variance strata into G groups, and let $df(\gamma)$ be the effective degrees of freedom (or in general any measure of the quality of the variance estimator) resulting from a particular grouping $\gamma = \{\gamma_1, \cdots, \gamma_G\} \in \Omega$, then we wish to find the particular grouping $\gamma^* \in \Omega$ such that

$$df(\gamma^*) = \max_{\gamma \in \Omega} df(\gamma). \tag{3.5}$$

In principle, one could search over all $\gamma \in \Omega$ to find the optimal solution, however, this is infeasible in most practical situations due to the huge number of possible groupings.

A class of algorithms which has not been considered in this setting were originally used to solve the famous multiprocessing problem in scheduling theory. This problem can be formalized using notation paralleling our grouping problem as

Definition 3.1 *(***Multiprocessing Problem***) Given a finite set of L computation jobs, a nonnegative duration time a_h for each $h = 1, \cdots, L$, and a number $G \geq 2$ of processors, the goal is to obtain a grouping $\gamma^* = \{\gamma_1^*, \cdots, \gamma_G^*\} \in \Omega$ of the L jobs, such that*

$$C_{max}(\gamma^*) = \min_{\gamma \in \Omega} C_{max}(\gamma), \tag{3.6}$$

where $C_{max}(\gamma) = \max\{S_g; 1 \leq g \leq G\}$.

In other words, we wish the final completion time, $C_{max}(\gamma)$, to be minimized.

A good grouping scheme for problem (3.6) will also be a reasonable choice for problem (3.5), considering that a minimized deviation of **the maximum** S_g from the

average group size, $\sum_{h=1}^{L} a_h/G$, implies small deviation of S_g for **any** group g from $\sum_{h=1}^{L} a_h/G$. As L becomes large both should tend to zero.

The first algorithm we propose is an adaptation of the so called Longest Processing Time Algorithm (A_{LPT}) in scheduling theory (see Parker, 1995, pg. 88) to directly address (3.5). As will be demonstrated later, A_{LPT} can also be easily extended to handle the more complex multiple-domain situation where Lee's algorithms and the others have difficulties. The idea of A_{LPT} is to first arrange all jobs (variance strata) in decreasing order (longest processing time or LPT order), and then, whenever a processor (group) becomes available, assign the first available job (unassigned variance stratum), consecutively, from largest to smallest, to some group so as to minimize C_{max} for the currently assigned jobs.

In (3.5) we have L variance strata with $a_1, \cdots, a_L$ ($a_h = W_h^2\sigma_h^2/n_h$) to be assigned to G groups. For $h = 1, \cdots, L$ and $j = 1, \cdots, G$, let

$$x_{hj} = \begin{cases} 1 & \text{if the } h\text{-th stratum is in the } j\text{-th group,} \\ 0 & \text{otherwise.} \end{cases}$$

Clearly, $\sum_{j=1}^{G} x_{hj} = 1$ for all h, as each variance stratum will be assigned to exactly one group.

A_{LPT} Algorithm:

1. For $j = 1, \cdots, G$, let $x_{jj} = 1$ and $S_j^{(G)} = a_j$. Let $h = G$.
2. If $h < L$, let $j_{h+1} = \underset{1 \le j \le G}{\operatorname{argmin}} S_j^{(h)}$ and let $x_{h+1,j_{h+1}} = 1$; otherwise stop.
3. Let $h = h + 1$. Repeat Step 2.

Denote, for each h,

$$S_j^{(h)} = \sum_{i=1}^{h} a_i x_{ij} \quad \text{and} \quad \bar{S}^{(h)} = \sum_{j=1}^{G} S_j^{(h)}/G = \sum_{i=1}^{h} a_i/G,$$

and $df^{(h)}$ as the effective degrees of freedom obtained by allocating the largest h variance strata, $a_1, \cdots, a_h$, to G groups. From (3.2) we know that

$$df^{(h)} = \frac{c_0^{(h)}}{c_1^{(h)} + \sum_{j=1}^{G} (S_j^{(h)})^2}, \tag{3.7}$$

where $c_0^{(h)} = (\sum_{i=1}^{h} a_i)^2$ and $c_1^{(h)} = (1/4)\sum_{i=1}^{h} a_i^2(\beta_i - 3)$. If we denote $df^{(h+1),j}$ as the degrees of freedom computed by assigning the next stratum (a_{h+1}) to group j given the previous grouping scheme, or letting $x_{h+1,j} = 1$, we get

$$\begin{aligned} df^{(h+1),j} &= \frac{c_0^{(h+1)}}{c_1^{(h+1)} + \sum_{i=1}^{G}(S_i^{(h)} + a_{h+1}x_{h+1,i})^2} \\ &= \frac{c_0^{(h+1)}}{c_1^{(h+1)} + \sum_{i=1}^{G}(S_i^{(h)})^2 + 2a_{h+1}S_j^{(h)} + a_{h+1}^2}. \end{aligned} \tag{3.8}$$

To adapt the A_{LPT} strategy to problem (3.5) we let

$$df^{(h+1)} = \max_{1 \le j \le G} df^{(h+1),j}.$$

As h approaches $L-1$, we obtain the final effective degrees of freedom, $df = df^{(L)}$. More formally,

Algorithm 1:

1. Let $x_{jj} = 1$ and $S_j^{(G)} = a_j$, $j = 1, \cdots, G$. Let $h = G$.
2. If $h < L$, let $j_{h+1} = \underset{1 \le j \le G}{\operatorname{argmax}}\, df^{(h+1),j}$ and let $x_{h+1,j_{h+1}} = 1$; otherwise stop.
3. Let $h = h + 1$. Repeat Step 2.

Algorithm 1 is simple and extremely fast. Its basic strategy is to take the stratum with the largest a_h from all remaining strata and place it in the group which maximizes df. In fact, in the single-domain case, Algorithm 1 is equivalent to the A_{LPT} algorithm in the sense that the procedure of maximizing the left hand side of expression (3.8) is the same as finding the minimum $S_j^{(h)}$ and allocating a_{h+1} to it. However, the simplicity in concept allows us to easily extend Algorithm 1 to the multi-domain case by replacing the objective function, $df^{(h),j}$, with a more complex functional form, as will be described in detail in chapter 4. In addition, the algorithm can also be modified to meet the equal number of strata per group restriction, if desired. Let $L^* = L/G$ be the number of strata required in each group. We only need to change Step 2 of Algorithm 1 so that only groups with less than L^* strata are allowed to accommodate

the stratum to be assigned, i.e.

Algorithm 2:

1. Let $x_{jj} = 1$, $d_j = L_* - 1$ and $S_j^{(G)} = a_j$, $j = 1, \cdots, G$, $L^* = L/G$. Let $h = G$.
2. Let $I = \{1 \leq j \leq G : d_j > 0\}$. If $h < L$, let $j_{h+1} = \underset{j \in I}{\operatorname{argmax}}\, df^{(h+1),j}$, $x_{h+1,j_{h+1}} = 1$ and $d_{j_{h+1}} = d_{j_{h+1}} - 1$; otherwise stop.
3. Let $h = h + 1$. Repeat Step 2.

3.3 Some Theoretical Evaluations

One motivation for using a combined strata grouped variance estimator is to save computational resources for the end-users by using fewer replicates. That is to say, given the number of replicates, one wants efficiency. We therefore establish a series of upper- and lower-bounds for attainable degrees of freedom for the purpose of comparing the performance of algorithms. More than that, as the upper- and lower-bounds can be expressed as functions of the numbers of original and combined strata as well as some measurement describing the behavior of the variance strata, we can also use them to help us determine how many replicates are needed to achieve the required precision even before applying any of these grouping algorithms, and thus reduce any exploratory work in determining the number of groups to consider.

An upper-bound for the degrees of freedom that results from (3.2) directly is given in the following lemma. It is practically important in many situations, as the upper-bound is often nearly attainable by the proposed algorithms.

Lemma 3.1 *For a complex sampling design with known stratum quantities W_h, σ_h^2 and $\beta_h \geq 3$, $h = 1, \cdots, L$, the degrees of freedom for a combined strata grouped variance estimator obtained by applying a combining strata scheme and using (3.2) satisfy*

$$df[v_c(\bar{y})] \leq \min \left\{ G, \frac{\left(\sum_{h=1}^{L} W_h^2 \sigma_h^2\right)^2}{\sum_{h=1}^{L} W_h^4 \sigma_h^4} \right\}.$$

Proof. Notice that $\sum_{g=1}^{G}\sum_{h\in L_g} W_h^2\sigma_h^2 = \sum_{h=1}^{L} W_h^2\sigma_h^2$, and thus

$$\sum_{g=1}^{G}\left(\sum_{h\in L_g} W_h^2\sigma_h^2\right)^2 \geq \frac{1}{G}\left(\sum_{h=1}^{L} W_h^2\sigma_h^2\right)^2.$$

Furthermore, $\sum_{h=1}^{L} W_h^4\sigma_h^4(\beta_h - 3) \geq 0$ if $\beta_h \geq 3$ for all $h = 1, \cdots, L$. Therefore, from (3.2) we have

$$df(v_c(\bar{y})) \leq \frac{\left(\sum_{h=1}^{L} W_h^2\sigma_h^2\right)^2}{\sum_{g=1}^{G}\left(\sum_{h\in L_g} W_h^2\sigma_h^2\right)^2} \leq \frac{\left(\sum_{h=1}^{L} W_h^2\sigma_h^2\right)^2}{\frac{1}{G}\left(\sum_{h=1}^{L} W_h^2\sigma_h^2\right)^2} = G.$$

Noting that $\sum_{g=1}^{G}\left(\sum_{h\in L_g} W_h^2\sigma_h^2\right)^2 \geq \sum_{g=1}^{G}\sum_{h\in L_g} W_h^4\sigma_h^4 = \sum_{h=1}^{L} W_h^4\sigma_h^4$, it follows that $df(v_c(\bar{y})) \leq \left(\sum_{h=1}^{L} W_h^2\sigma_h^2\right)^2 / \sum_{h=1}^{L} W_h^4\sigma_h^4$, and the result follows.

Note that the second term inside the minimum in Lemma 3.1 is the degrees of freedom of the full jackknife when $\beta_h = 3$. It seems obvious that the attainable degrees of freedom cannot exceed the number of replicates, G. However, there are cases when the assumption of $\beta_h \geq 3$ does not always hold in a realized finite population, the resulting degrees of freedom calculated from formula (3.2) could then be slightly larger than the number of groups. This is of little practical concern as we primarily use degrees of freedom as a reference criterion for judging the quality of candidate grouping schemes.

Lower-bounds can be of interest as they are associated with individual grouping schemes and thus show how much we may avoid losing by applying a good one. They also give some insight into what aspects of the situation impact an algorithm's performance.

Lemma 3.2 *For variance strata $a_1, \cdots, a_L$, the degrees of freedom obtained from Algorithm 1 satisfy*

$$df \geq \frac{G}{1 + [G\sum_{i=1}^{G}(a_i - \bar{a}_G)^2 + (G-1)\sum_{i=G+1}^{L} a_i^2]/[L^2\bar{a}^2]},$$

where $\bar{a} = \sum_{i=1}^{L} a_i/L$ and $\bar{a}_G = \sum_{i=1}^{G} a_i/G$.

Proof. See section 3.4.

Note that the second term of the denominator in the above lower bound, depending on the behavior of a_h's, will be negligible for most practical situations and has at most an order of $O(1)$. Hence, Lemma 3.2 demonstrates two points:

1. The attained degrees of freedom by applying Algorithm 1 will be the same magnitude as the number of groups, a natural upper bound for *df* provided in Lemma 3.1;

2. The consistency of the corresponding variance estimator is also essentially assured as the ratio $V[v_{cj}(\bar{y})]/Var^2(\bar{y})$, the inverse of *df*, will go to zero as the number of groups becomes large.

3.4 Theorem Proofs

3.4.1 Proof of Theorem 3.1

We will outline the proof for v_{cj} using JK2. The proof parallels that of Krewski and Rao's (1981) proof for the delete-1 jackknife and BRR without combining strata. The proof for JKn and BRR are similar in spirit and thus are not presented.

We first prove that the theorem holds for the linear case, that is, $n\{v_c(\bar{\mathbf{y}}) - V(\bar{\mathbf{y}})\} \xrightarrow{p} \mathbf{0}$ as $G \to \infty$. Denote

$$\bar{\mathbf{y}}^{(g)} = \bar{\mathbf{y}} + \sum_{h \in L_g} W_h(\mathbf{y}_{hi} - \bar{\mathbf{y}}_h) = \bar{\mathbf{y}} + \sum_{h \in L_g} \mathbf{z}_h,$$

where $\mathbf{y}_{hi}$ represents a randomly chosen unit from stratum h, $\mathbf{z}_h = W_h(\mathbf{y}_{hi} - \bar{\mathbf{y}}_h)$, and

$$(\bar{\mathbf{y}}^{(g)} - \bar{\mathbf{y}})(\bar{\mathbf{y}}^{(g)} - \bar{\mathbf{y}})' = \sum_{h \in L_g} \mathbf{z}_h \cdot \sum_{h \in L_g} \mathbf{z}_h' = \sum_{h \in L_g} \mathbf{z}_h \mathbf{z}_h' + \sum_{h \neq h' \in L_g} \mathbf{z}_h \mathbf{z}_{h'}'.$$

Thus, the combined strata grouped JK2 variance estimator can be expressed as

$$\begin{aligned} v_c(\bar{\mathbf{y}}) &= \sum_{g=1}^{G} (\bar{\mathbf{y}}^{(g)} - \bar{\mathbf{y}})(\bar{\mathbf{y}}^{(g)} - \bar{\mathbf{y}})' = \sum_{g=1}^{G} \sum_{h \in L_g} \mathbf{z}_h \mathbf{z}_h' + \sum_{g=1}^{G} \sum_{h \neq h' \in L_g} \mathbf{z}_h \mathbf{z}_{h'}' \\ &= \mathbf{v}_u + \mathbf{d}, \end{aligned}$$

where $\mathbf{v}_u = \sum_{h=1}^{L} \mathbf{z}_h \mathbf{z}_h'$ is an unbiased and consistent estimator of $V(\bar{\mathbf{y}})$. Thus, we need only show that $n\mathbf{d} \to \mathbf{0}$, where

$$\mathbf{d} = \sum_{g=1}^{G} \sum_{h \neq h'}^{L_g} \mathbf{z}_h \mathbf{z}_{h'}' = \left[\sum_{g=1}^{G} \sum_{h \neq h'}^{L_g} W_h W_{h'} \left(y_{hij} - \bar{y}_{hj} \right) \left(y_{h'i'k} - \bar{y}_{h'k} \right) \right]_{p \times p}.$$

Since $\bar{\mathbf{y}} - \bar{\mathbf{Y}} \to \mathbf{0}$ in probability, it suffices to show that, for any $\varepsilon > 0$ and $j, k = 1, \cdots, p$,

$$A = P\left\{ n \left| \sum_{g=1}^{G} \sum_{h \neq h'}^{L_g} W_h W_{h'} \left(y_{hij} - \bar{Y}_{hj} \right) \left(y_{h'i'k} - \bar{Y}_{h'k} \right) \right| \geq \varepsilon \right\} \xrightarrow{p} 0, \tag{3.9}$$

as $G \to \infty$. Applying Chebyshev's Inequality on the left hand side of (3.9), we have

$$\begin{aligned} A &\leq \frac{n^2}{\varepsilon^2} E \left[\sum_{g=1}^{G} \sum_{h \neq h'}^{L_g} W_h W_{h'} \left(y_{hij} - \bar{Y}_{hj} \right) \left(y_{h'i'k} - \bar{Y}_{h'k} \right) \right]^2 \\ &= \frac{n^2}{\varepsilon^2} \sum_{g=1}^{G} \sum_{h \neq h'}^{L_g} W_h^2 W_{h'}^2 E \left(y_{hij} - \bar{Y}_{hj} \right)^2 E \left(y_{h'i'k} - \bar{Y}_{h'k} \right)^2 \\ &= \frac{n^2}{\varepsilon^2} \sum_{g=1}^{G} \sum_{h \neq h'}^{L_g} n_h n_{h'} a_{hj} a_{h'k} \\ &\leq \frac{G}{\varepsilon^2} \left(\max_{1 \leq h \leq L} n_h \right)^2 \left(n^2 \bar{S}_{\cdot j} \bar{S}_{\cdot k} \right) \left(\frac{\max_g S_{gj}}{\bar{S}_{\cdot j}} \right) \left(\frac{\max_g S_{gk}}{\bar{S}_{\cdot k}} \right) = O\left(\frac{1}{G} \right), \end{aligned}$$

under conditions C2, C4 and (3.3), and using the fact that $\bar{S}_{\cdot j} = O(1/(LG))$. Then (3.9) follows.

Now we consider $\hat{\theta} = \varphi(\bar{\mathbf{y}})$. Only the case $p = 1$ is considered in detail; extension to $p > 1$ is relatively straightforward. Let $X_{h(g)} = LW_h(y_{h(g)} - \bar{Y}_h)$. Under C1 and C3, it follows that $L^{-1} \sum_{h=1}^{L} E|X_{h(g)}|^{2+\delta} = O(1)$. Since

$$\bar{X}_{(g)} = L^{-1} \sum_{h} X_{h(g)} = \sum_{h} W_h (y_{h(g)} - \bar{Y}_h) = \bar{y}^{(g)} - \bar{Y}$$

and $E(\bar{X}_{(g)}) = 0$, by the law of large numbers for independent and non-identically distributed random variables (see Krewski and Rao, 1981, Lemma 3.2 and related references therein), for $\Delta > 0$,

$$P\{|\bar{y}^{(g)} - \bar{Y}| \geq \Delta/2\} = P\{|\bar{X}_{(g)} - E(\bar{X}_{(g)})| \geq \Delta/2\} = O(L^{-(1+\delta/2)}).$$

Thus,

$$\begin{aligned} P\{\max_{1\le g\le G}|\bar{y}^{(g)}-\bar{Y}|\ge\Delta/2\} &\le \sum_g P\{|\bar{y}^{(g)}-\bar{Y}|\ge\Delta/2\} \\ &= G\cdot O(L^{-(1+\delta/2)})\to 0 \end{aligned}$$

as $G, L \to \infty$ ($G \le L$). Since $\bar{y} - \bar{Y} \to 0$ and $\bar{Y} \to \mu$, we have

$$P\{all\ \bar{y}^{(g)},\ \bar{y} \in (\mu-\Delta, \mu+\Delta)\ simultaneously\} \to 1. \tag{3.10}$$

For all $\bar{y}^{(g)}$, $\bar{y}$ in $I = (\mu - \Delta, \mu + \Delta)$, a Taylor expansion can be used to express

$$\hat{\theta}^{(g)} = \hat{\theta} + (\bar{y}^{(g)} - \bar{y})\varphi'(\xi^{(g)}),$$

where $\xi^{(g)}$ lies between $\bar{y}^{(g)}$ and $\bar{y}$. Let $\phi(t) = \varphi'(t) - \varphi'(\mu)$. Because of its continuity at $t = \mu$, for any $\epsilon > 0$, there exists some $\Delta_\epsilon > 0$ such that, for any $t \in (\mu - \Delta_\epsilon, \mu + \Delta_\epsilon)$, $|\phi(t)| < \epsilon$. Thus, by (3.10)

$$P\{\max_{1\le g\le G}|\phi(\xi^{(g)})|<\epsilon\} \ge P\{all\ \bar{y}^{(g)},\ \bar{y}\in I_\epsilon\ simultaneously\} \to 1,$$

that is, the $\max_g |\phi(\xi^{(g)})| \to 0$. Therefore,

$$\begin{aligned} n\cdot v_{cj}(\hat{\theta}) &= n\sum_{g=1}^{G}(\hat{\theta}^{(g)}-\hat{\theta})^2 = n\sum_{g=1}^{G}(\bar{y}^{(g)}-\bar{y})^2\varphi'(\xi^{(g)})^2 \\ &= n\sum_{g=1}^{G}(\bar{y}^{(g)}-\bar{y})^2\left[\varphi'(\mu)^2+\phi(\xi^{(g)})^2+2\varphi'(\mu)\phi(\xi^{(g)})\right] \\ &= n\cdot v_{cj}(\bar{y})\varphi'(\mu)^2 + \text{remainder}. \end{aligned} \tag{3.11}$$

The first term on the right-hand side $\to \sigma^2 = \varphi'(\mu)^2 V(\bar{y})$ in probability, while the remainder goes to zero since $\max_g |\phi(\xi^{(g)})| \to 0$. This completes the proof.

3.4.2 Proof of Lemma 3.2

Using the notation of section 3.2,

$$\bar{S}^{(h+1)} = \frac{1}{G}\sum_{j=1}^{G} S_j^{(h+1)} = \frac{1}{G}\sum_{j=1}^{G}\left(S_j^{(h)} + a_{h+1}x_{h+1,j}\right) = \bar{S}^{(h)} + \frac{a_{h+1}}{G}.$$

Hence, for $h = G, \cdots, L-1$,

$$\begin{aligned}
\sum_{j=1}^{G}\left(S_j^{(h+1)} - \bar{S}^{(h+1)}\right)^2 &= \sum_{j=1}^{G}\left[S_j^{(h)} + a_{h+1}x_{h+1,j} - \left(\bar{S}^{(h)} + \frac{a_{h+1}}{G}\right)\right]^2 \\
&= \sum_{j=1}^{G}\left[\left(S_j^{(h)} - \bar{S}^{(h)}\right) + a_{h+1}\left(x_{h+1,j} - \frac{1}{G}\right)\right]^2 \\
&= \sum_{j=1}^{G}\left(S_j^{(h)} - \bar{S}^{(h)}\right)^2 + 2a_{h+1}\sum_{j=1}^{G}x_{h+1,j}\left(S_j^{(h)} - \bar{S}^{(h)}\right) + a_{h+1}^2\sum_{j=1}^{G}\left(x_{h+1,j} - \frac{1}{G}\right)^2 \\
&= \sum_{j=1}^{G}\left(S_j^{(h)} - \bar{S}^{(h)}\right)^2 + 2a_{h+1}\left(S_{j_{h+1}}^{(h)} - \bar{S}^{(h)}\right) + \frac{G-1}{G}a_{h+1}^2 \\
&\leq \sum_{j=1}^{G}\left(S_j^{(h)} - \bar{S}^{(h)}\right)^2 + \frac{G-1}{G}a_{h+1}^2,
\end{aligned}$$

as j_{h+1} is determined such that $S_{j_{h+1}}^{(h)} = \min_j S_j^{(h)} \leq \bar{S}^{(h)}$. By induction,

$$\begin{aligned}
\sum_{j=1}^{G}(S_j^{(L)} - \bar{S}^{(L)})^2 &\leq \sum_{j=1}^{G}(S_j^{(G)} - \bar{S}^{(G)})^2 + \frac{G-1}{G}\sum_{i=G+1}^{L}a_i^2 \\
&= \sum_{i=1}^{G}(a_i - \bar{a}_G)^2 + \frac{G-1}{G}\sum_{i=G+1}^{L}a_i^2.
\end{aligned}$$

Therefore,

$$\begin{aligned}
df &= \frac{\left(\sum_{j=1}^{G}S_j^{(L)}\right)^2}{\sum_{j=1}^{G}S_j^{(L)2}} = \frac{G^2\bar{S}^{(L)2}}{\sum_{j=1}^{G}\left[S_j^{(L)} - \bar{S}^{(L)}\right]^2 + G\bar{S}^{(L)2}} \\
&= \frac{G}{1 + \sum_{j=1}^{G}\left[S_j^{(L)} - \bar{S}^{(L)}\right]^2/(L^2\bar{a}^2/G)} \\
&\geq \frac{G}{1 + \left[G\sum_{i=1}^{G}(a_i - \bar{a}_G)^2 + (G-1)\sum_{i=G+1}^{L}a_i^2\right]/(L^2\bar{a}^2)}. \qquad (3.12)
\end{aligned}$$

This completes the proof.

Chapter 4

Algorithms for Grouping Schemes with Multiple Domains

Most sample surveys are multi-purpose and reliable estimation for domains is an important goal. Designing reduced replicate schemes that perform well across a range of measured characteristics and/or domains is a more difficult problem. To handle estimators from different domains, or more generally a $p \times 1$ vector of characteristics, we must consider $\mathbf{a}_h = (a_{h1}, ..., a_{hp})'$, where $a_{hk} = W_h^2 \sigma_{hk}^2 / n_h$ is specified for characteristic or domain k. As previously discussed, the σ_{hk}^2 often have smaller relative variability across strata than the W_h^2 do when considering different characteristics and thus a practical solution for dealing with multiple characteristics is to apply the methods of section 2 with the σ_{hk}^2 replaced with 1. However, when considering different domains such a practical solution may not work well if only a portion of the variance strata is represented from a particular domain. In this situation, the W_h distributions are different for different domains. Nevertheless, if information is available and a need determined, multiple characteristics and/or domains can be accommodated. For simplicity, from now on we focus only on domains and will treat the population as if it consists of p domains with the value of a_{hk} different for each domain. The degrees of freedom for a variance estimator for the k-th estimator is still given by (3.1). For the multivariate situation, the problem is to determine an appropriate formulation of the univariate maximization problem described above.

In practice, combining strata in such a way as to account for multiple domains is a difficult problem that is often handled using *ad hoc* procedures. When the rationale for a combining scheme is given, it may involve some judgement about the relative importance of the precision of variance estimators for different domains. For example, if region is an important domain, then it might be reasonable to combine strata so strata from the same region are not placed in the same combined stratum, to the extent possible, so that each deletion removes units from multiple domains. If estimates for other domains such as urbanity and race of the respondent are also required, the procedure for combining is more complex and guidelines are difficult to describe. For some examples see DiGaetaeno et al. (1998), Nixon et al. (1998), Little et al. (1997), and Parsons, Chan and Curtin (1990).

4.1 Proposed Algorithms

The difficulty lies in trying to jointly reduce $V(v_c(\bar{y}_k))$ for all k simultaneously. One way to approach the problem is to merely state it as an optimization problem in a similar way to (3.1). That is, let Ω_G be the set of all possible groupings of the L original variance strata into G groups, and let $df_k(\gamma)$ be the effective degrees of freedom for domain k resulting from a particular grouping $\gamma \in \Omega$. Then we wish to find the particular grouping γ^* such that

$$df_k(\gamma^*) = \max_{\gamma \in \Omega} df_k(\gamma) \tag{4.1}$$

for all k simultaneously. This is an intractable problem in most situations, so one might replace it with

$$f(\mathbf{df}(\gamma^*)) = \max_{\gamma \in \Omega} f(\mathbf{df}(\gamma)), \tag{4.2}$$

where $\mathbf{df} = (df_1, ..., df_p)'$ and $f(\cdot)$ is some function which measures a type of average degree of freedom over the p domains. The simplest examples might be $f(\mathbf{df}) = \sum_k df_k/p$ or more generally, $f(\mathbf{df}) = \sum_k w_k df_k$ where $w_1, ..., w_k$ are weights representing the relative importance of the domains, or perhaps $f(\mathbf{df}) = \min\{df_1, \cdots, df_p\}$,

in an effort to keep the degrees of freedom needed for **each** key domain above a reasonable level, say 30.

Once $f(\cdot)$ has been defined, denote

$$S_{jk}^{(h)} = \sum_{i=1}^{h} a_{ik}x_{ij} \quad \text{and} \quad \bar{S}_{\cdot k}^{(h)} = \sum_{j=1}^{G} S_{jk}^{(h)}/G.$$

Furthermore, denote

$$\mathbf{df}^{(h)} = \left[\frac{G^2 \cdot (\bar{S}_{\cdot k}^{(h)})^2}{\sum_{i=1}^{G}(S_{ik}^{(h)})^2}\right]'_{p\times 1} \quad \text{and} \quad \mathbf{df}^{(h+1),j} = \left[\frac{G^2 \cdot (\bar{S}_{\cdot k}^{(h+1)})^2}{\sum_{i=1}^{G}(S_{ik}^{(h)})^2 + 2a_{h+1,k}S_{jk}^{(h)} + a_{h+1,k}^2}\right]'_{p\times 1}.$$

Then one only needs to modify Step 2 of Algorithms 1 and 2 in section 3.2 to obtain the following algorithms:

Algorithm 3:

1. Let $x_{jj} = 1$ and $S_{jk}^{(G)} = a_{jk}$, $j = 1, \cdots, G$. Let $h = G$.
2. If $h < L$, let $j_{h+1} = \underset{1\le j\le G}{\text{argmax}}\ f(\mathbf{df}^{(h+1),j})$ and let $x_{h+1,j_{h+1}} = 1$; otherwise stop.
3. Let $h = h + 1$. Repeat Step 2.

Algorithm 4:

1. Let $x_{jj} = 1$, $d_j = \eta - 1$ and $S_{jk}^{(G)} = a_{jk}$, $j = 1, \cdots, G$. Let $h = G$.
2. Let $I = \{1 \le j \le G : d_j > 0\}$. If $h < L$, let $j_{h+1} = \underset{j\in I}{\text{argmax}}\ f(\mathbf{df}^{(h+1),j})$, $x_{h+1,j_{h+1}} = 1$ and $d_{j_{h+1}} = d_{j_{h+1}} - 1$; otherwise stop.
3. Let $h = h + 1$. Repeat Step 2.

Notice that the ordering of variance strata in this situation is a bit more complex, as well. Empirical investigations suggest that ranking strata entry in terms of the value of $f(a_{h1}, \cdots, a_{hp})$, $h = 1, \cdots, L$, performs fairly well. One can use a similar approach to extend SAOA by ordering the strata via $f(a_{h1}, \cdots, a_{hp})$. We will see in section 4 that this extension of SAOA does not perform very well, however.

4.2 Some Theoretical Results for Multiple Domains

Firstly, the upper-bound given in Lemma 1 can be easily extended to multiple domains. Assume that $UB_1, \cdots, UB_p$ are upper-bounds for the vector of degrees of freedom, $\mathbf{df} = (df_1, \cdots, df_p)'$, respectively. Then $f(UB_1, \cdots, UB_p)$ will be an upper-bound of $f(\mathbf{df})$ given that $f(\cdot)$ is monotone. In fact, it is shown in our simulation that the attained $f(\mathbf{df})$ is quite close to this upper-bound, demonstrating that the proposed algorithms work very well. On the other hand, the lower-bound of $f(\mathbf{df})$ cannot be obtained in the same way, as the grouping for different domains conflict. The following lemmas take into account the correlations among domains and give lower-bounds for $f(\mathbf{df})$ for two basic forms of $f(\cdot)$.

Lemma 4.1 *For variance strata* $\{a_{hk} : h = 1, \cdots, L, ;\ k = 1, \cdots, p\}$*, the resulting average degrees of freedom over all domains obtained by using Algorithm 3 satisfies (up to second order)*

$$\begin{aligned} f(\mathbf{df}) &= \frac{1}{p}\sum_k df_k \approx G + \frac{G}{p}\sum_{k=1}^{p}(z_k^{(L)2} - z_k^{(L)}) \\ &\geq G + \frac{1}{p}\sum_{k=1}^{p}\left\{\frac{u_k^2}{G}\left[\sum_{j=1}^{G}(a_{jk} - \bar{a}_{\cdot k}^G)^2\right]^2 - b_k^{(L-G+1)}\sum_{j=1}^{G}(a_{jk} - \bar{a}_{\cdot k}^G)^2 + c_k^{(L-G+1)}\right\} \end{aligned}$$

where $b_k^{(1)} = u_k$, $c_k^{(1)} = 0$ *and, for* $g = 1, \cdots, L-G$, $b_k^{(g+1)} = b_k^{(g)} - 2(G+1)u_k^2 a_{h+1,k}^2/G^2$ *and* $c_k^{(g+1)} = c_k^{(g)} + (G-1)^2 u_k^2 a_{h+1,k}^4/G^3 - G - 1 b_k^{(g)} a_{h+1,k}^2/G$.

Proof. See section 4.6.

Lemma 4.2 *For variance strata* $\{a_{hk} : h = 1, \cdots, L,\ k = 1, \cdots, p\}$*, the resulting minimum degrees of freedom from all domains obtained by using Algorithm 3 satisfies*

$$\begin{aligned} f(\mathbf{df}) &= \min_{1\leq k\leq p} df_k = \min_{1\leq k\leq p}\frac{(\sum_{j=1}^{G} S_{jk})^2}{\sum_{j=1}^{G} S_{jk}^2} \\ &\geq \frac{G}{1 + \max_{1\leq k\leq p}\{u_k[\sum_{j=1}^{G}(a_{jk} - \bar{a}_{\cdot k}^G)^2 + \frac{G-1}{G}\sum_{i=G+1}^{L} a_{ik}^2]\}/G}. \end{aligned}$$

where $\bar{a}_{\cdot k}^G = \sum_{i=1}^{G} a_{ik}/G$, $k = 1, \cdots, p$.

Proof. See section 4.6.

4.3 Application and Empirical Investigation

In this section we apply the algorithms presented above to data from the 1995 National Health Interview Survey (NHIS) and some artificial populations based upon it, and perform a limited simulation study.

4.3.1 The NHIS Survey

The NHIS is an annual household survey that profiles the health characteristics of the civilian noninstitutionalized population of the United States. The National Center for Health Statistics (NCHS) is responsible for the survey, and the Census Bureau collects the data. The NHIS uses a relatively typical multistage, stratified sample design. For the 1995-2004 NHIS design, the United States was partitioned into about 2,000 psu's, which are individual counties, groups of adjacent counties or metropolitan areas. All psu's were assigned to either self-representing (SR) strata or nonself-representing (NSR) strata. As is discussed in section 2.9, a SR stratum contains only one psu, meaning that the psu will be drawn into any sample with certainty. On the other hand, a NSR stratum contains more than one psu and will have two psu's selected in an annual sample.

4.3.2 Current Replication Methods in NHIS

The NHIS public release file documentation does not contain the complete stratum and psu identifiers because of confidentiality concerns. Even if the complete identifiers were released, the full jackknife variance estimator would result in thousands of sets of replicate weights. The documentation instead gives two approximate methods for estimating variances from the design data on the file (for the rationale for these methods see Parsons and Casady, 1986; Parsons, Chan, and Curtin, 1990). The first method (Method 1) has 187 variance strata, each with exactly two psu's. The second method (Method 2) has many more strata. They suggest Method 2 can be used with linearization variance estimation software. The main difference between two methods involves handling self-representing (SR) psu's. In Method 2, the SR

psu's are partitioned into substrata based on race/ethnicity, with the substrata used as variance strata. Each secondary sampling unit (ssu) in a SR unit is treated as a separate psu rather than pairing them into pseudo-psu's as is done in Method 1. Nixon et al. (1998) discuss these methods and introduce Method 3. Method 3 attempts to produce stable variance estimates for national and domain estimates while reducing the number of replicates. In this method, 70 variance strata were created using the combined strata approach and allocating the existing variance strata in Method 2 roughly proportionate to the population total within each of the four regions (a key domain). This is an example of an *ad hoc* approach that explicitly considers domains. The ad hoc nature requires a high level of expertise and intuition, and a great deal of time and effort.

The variance strata for the three methods were evaluated for national estimates, and for estimates of the following domains: region, poverty status, metropolitan status, and race/ethnicity. Nixon et al. (1998) shows that all three methods provide variance estimates with adequate degrees of freedom for national estimates using v_{jk2}, although Method 1 (M1) has 187 replicates, Method 2 (M2) has 2,167 replicates, and Method 3 (M3) has only 70 replicates. For a 95 percent confidence interval, the appropriate t-value for Method 2 is 1.97, and for methods 1 and 3 it is 1.99. They further show that Method 1 performs poorly for many of the domains, whereas the other two methods generally do well. Even though Method 3 only has 70 replicates, it produces at least 30 degrees of freedom for each domain (See Table 4.1, reproduced in part from Table 2 of Nixon, et al., 1998). Note that we treat the 2,167 variance strata of Method 2 as true strata which makes Method 2 essentially the full jackknife for our comparisons. The values for Method 2 in Table 4.1 were recalculated and differ slightly from those presented in Nixon et al. (1998).

4.3.3 A Quick Comparison of Different Methods

Before doing any indepth comparison of methods, we applied Algorithm 3 (A3), using the sum of degrees of freedom across all domains, to this situation and obtained the

second to last column in Table 4.1. The algorithm takes only a few seconds of real-time on a laptop. Algorithm 4 performed similarly (discussed later). We should note that Lee's SAOA, even when ordered on the sum of degrees of freedom, did not even out perform Method 3, and did very poorly relative to proposed Algorithm 3 (details later).

Table 4.1: Method Comparison: Estimated df's with upper bound

Domain	M1	M2	M3	A3	Upper Bound for M3 & A3
National	72	488	66	70	70
Region 1	7	167	61	66	70
Region 2	24	122	57	67	70
Region 3	51	139	59	69	70
Region 4	12	103	56	61	70
Poverty	53	152	52	64	70
Nonwhite	26	163	55	69	70
Hispanic	10	52	32	38	52
MSA	48	470	63	69	70
Non-MSA	60	59	42	58	59
Number of Replicates	187	2,167	70	70	

4.3.4 Some Hypothetical Populations

To make a more indepth comparison of methods we consider three populations. Population 2 is the NHIS public release files used in Table 4.1. Populations 1 and 3 are based upon the NHIS keeping the same domain weights and number of variance strata, but altering the W_h values to consider the impact of various patterns. Populations 1 and 3 used $W_h^* = W_h^{1.1}$ and $W_h^{**} = W_h^{0.75}$ to mimic more and less extreme situations, respectively. Figure 4.1 presents the distribution of variance strata $a_h = W_h^2\sigma_h^2/n_h$ for all three populations. The shapes are reasonably typical.

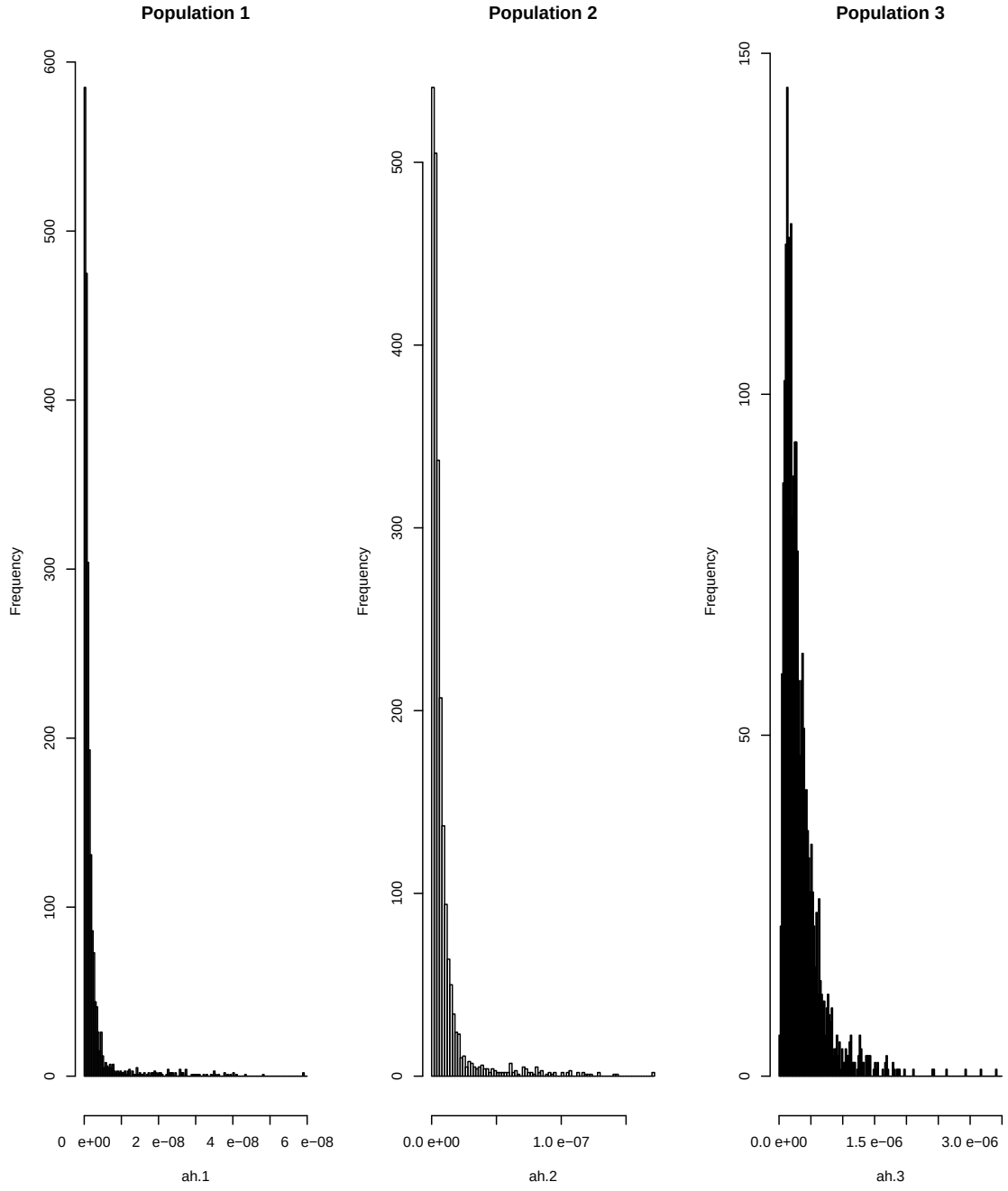


Figure 4.1: Distribution of a_h's for three populations

4.3.5 Comparison of Algorithm Performance

We first apply Lee's SAOA and Algorithm 1 to the National values, using $G = 50$ and 70 groups, and examine the performance on the domains to see the potential impact of ignoring key domains when creating grouped replicates. The results are summarized in Table 4.2, rounded to the nearest integer for ease of presentation.

Table 4.2: Attained df's for SAOA, Proposed Algorithms 1 and Method 3

Domain	Population 1				Population 2 (NHIS, 1995)					Population 3			
	SAOA		A1		SAOA		A1		**M3**	SAOA		A1	
	$G = 50$	70	50	70	50	70	50	70	**70**	50	70	50	70
National	49	68	50	70	49	69	50	70	**66**	50	70	50	70
Region 1	37	41	36	46	41	50	37	51	**61**	45	60	47	59
Region 2	39	39	34	45	37	46	37	49	**57**	46	58	43	55
Region 3	42	50	40	53	41	52	43	56	**59**	45	63	46	62
Region 4	31	39	31	36	36	42	35	42	**56**	43	57	44	58
Poverty	34	45	36	47	39	53	38	52	**52**	46	63	47	63
Nonwhite	39	49	41	44	40	51	40	50	**55**	44	60	44	61
Hispanic	17	19	17	19	22	32	26	32	**32**	45	60	45	60
MSA	46	63	47	64	48	66	48	67	**63**	50	69	50	69
Non-MSA	34	45	33	38	35	44	32	40	**42**	36	41	30	41
Avg(df)	37	46	37	46	39	51	39	51	**54**	45	60	45	60
Min(df)	17	19	17	19	22	32	26	32	**32**	36	41	30	41

The last column under Population 2 repeats the results for Method 3 of Nixon et al. (1998) from Table 4.1, only for Population 2 and $G = 70$ (for ease of comparison). We see that using SAOA or Algorithm 1, and ignoring domains yields poor performance relative to Method 3 of Nixon et al. (1998) for Population 2, for some domains.

To make SAOA comparable to proposed Algorithm 3, we apply SAOA to the ordered average df's, or $f(\mathbf{df}) = \sum_k df_k/p$ in the simplest attempt to account for domains, denoted SAOA_2. The performance of both SAOA_2 and Algorithm 3 on the same three populations are presented in Table 4.3 for group size $G = 50$.

As can be seen, Algorithm 3 outperforms both SAOA_2 and Method 3 by a wide margin, on average about 10 df's greater with $G = 70$. In addition, the attained

degrees of freedom for Algorithm 3 are fairly close to the theoretical upper bounds, denoted UB(A3) in Table 4.3, indicating that there is little motivation to look for more sophisticated algorithms. The results for proposed Algorithms 2 and 4 were not included in Table 2 as they were essentially equivalent to those of Algorithms 1 and 3, respectively.

Table 4.3: Attained df's for $SAOA_2$, Proposed Algorithms 3 and its Upper Bound

Domain	Population 1			Population 2			Population 3		
	$SAOA_2$	A3	UB(A3)	$SAOA_2$	A3	UB(A3)	$SAOA_2$	A3	UB(A3)
				$G = 50$					
National	49	50	50	49	50	50	50	50	50
Region 1	38	46	50	38	49	50	45	50	50
Region 2	35	49	50	39	50	50	43	50	50
Region 3	39	50	50	40	50	50	45	50	50
Region 4	33	43	50	33	48	50	42	50	50
Poverty	38	44	50	41	48	50	46	50	50
Nonwhite	38	50	50	41	50	50	45	50	50
Hispanic	17	19	23	26	33	50	44	50	50
MSA	45	50	50	47	50	50	50	50	50
Non-MSA	34	47	50	38	48	50	43	49	50
Avg(df)	37	45	47	39	48	50	45	50	50
Min(df)	17	19	23	26	33	50	42	49	50

This brings up the issue of Rust's (1986) recommendation to create approximately equal sized groups. Intuitively, avoiding combinations that place a large number of strata in one (or a few) combined strata should reduce the likelihood of having a very unequal distribution of the A_g for a particular domain or characteristic which has not been controlled for. The essentially equivalent performance of Algorithms 1 and 3 to Algorithms 2 and 4 seems to contradict this intuition. The explanation is related to the nature of the algorithms. Since Algorithms 1 and 3 assign the largest a_h's to groups first and add them to groups with the smallest current total, the algorithms tend to automatically create groups of approximately equal size. For example, A1 applied to the full population with $G = 70$ had 12%, 78% and 10% of

groups with sizes 30, 31 and 32, respectively. Also, the groups will clearly tend to have similar a_h distributions with some large, medium and small sizes. This should help in terms of maintaining confidentiality by making it more difficult to reconstruct the psu indicators from resulting replicate weights, since psu's of likely disparate make-up are deleted together. We discuss such issues in more detail in Chapter 5.

4.4 Simulation Study

To evaluate the performance of variance estimators and confidence intervals resulting from grouping with Algorithm 3, we perform a simulation study to compare it to the full jackknife applied to all 2,167 strata. For each of the three populations, we generated $S = 10,000$ independent stratified simple random samples with replacement with equal sample size for each stratum, $n_h = 2$.

A series of quantities are introduced as the criteria for evaluating the performance of the resulting variance estimators after grouping via Algorithm 3 as compared to the full jackknife. For a variance estimator v, its relative bias was measured by

$$\text{rel. bias} = \frac{\sum_s v_s/S - \text{MSE}}{\text{MSE}}.$$

The precision of a variance estimator v is measured by its relative instability, defined by

$$\text{rel. instab} = \frac{[\sum_s (v_s - \text{MSE})^2/S]^{1/2}}{\text{MSE}}.$$

The confidence intervals are also compared in terms of their error rates in lower (L) and upper (U) tails, corresponding to 5% nominal error rate in each tail, and standardized lengths. The error rates and standardized lengths were calculated as

$$\text{error rate in the lower tail} = (\text{no. of samples with} \quad \theta < \theta_{1s})/S,$$

$$\text{error rate in the upper tail} = (\text{no. of samples with} \quad \theta > \theta_{2s})/S,$$

and

$$\text{standardized length} = \frac{\sum_s l_s/S}{2z_{\alpha/2}\sqrt{\text{MSE}}},$$

respectively. The results of the simulation study are reported in Table 4.4. We see that reducing the number of replicates from over 2000 to 50 or 70 does not impact the performance of resulting variance estimates and confidence intervals greatly.

4.5 Summary

In the first part of this thesis, we have formalized conditions under which combined strata grouped jackknife and balanced repeated replications schemes will lead to consistent variance estimators in stratified multi-stage surveys. We then construct algorithms based on those used in scheduling theory for multi-processor computer networks to develop such replication-based variance estimation strategies that reduce the number of sets of replicate weights with least impact on resulting degrees of freedom overall and for some key analytical domains. This reduces the computational burden on less sophisticated end-users of public release data sets.

The nature of the proposed algorithms ensures the resulting jackknife and/or balanced repeated replication variance estimators have good performance. As well, they provide a set of publicly released replicate weights which provide protection against disclosure of psu and strata identifiers and thus maintain confidentiality.

A few things to note. First, there are many existing algorithms for grouping items in many varying contexts that could in principle be adapted to solve this problem (e.g. more sophisticated algorithms from scheduling theory, simulated annealing and also evolutionary and/or genetic algorithms for structured data). However, the performance of the simple algorithms given here suggests that in this context a more complicated strategy is not necessary. Second, there are situations where the number of strata and psu's is not large enough for grouping to be a viable strategy as there are insufficient degrees of freedom to sacrifice any, and yet confidentiality is still important. In these cases alternate strategies for creating replicate weights are necessary. This will be considered in Chapter 5.

Table 4.4: Lower(L) and upper(U) tail error rates for Alg. 3 and full jackknife

Domain	Method	Group	5% tail error rate			Rel.Bias	Rel.Instab	St.Len.
			L	U	L+U			
National	Algo. 3	30	5.1	5.1	10.2	3.85	0.28	1.01
		50	5.6	5.1	10.7	4.05	0.22	1.01
		70	5.1	5.5	10.6	4.68	0.19	1.02
	Full jack.	-	5.2	5.1	10.3	4.03	0.08	1.02
Region 1	Algo. 3	30	4.7	7.0	11.7	-1.48	0.25	0.98
		50	4.6	6.6	11.2	0.21	0.20	1.00
		70	4.7	6.0	10.7	0.77	0.17	1.00
	Full jack.	-	4.0	6.6	10.6	0.24	0.11	1.00
Region 2	Algo. 3	30	5.6	6.0	11.6	-0.01	0.27	0.99
		50	4.9	5.7	10.6	-0.44	0.20	0.99
		70	4.8	5.3	10.1	-0.15	0.18	1.00
	Full jack.	-	4.5	5.3	9.8	-0.07	0.13	1.00
Region 3	Algo. 3	30	4.7	5.3	10.0	2.50	0.27	1.00
		50	4.6	5.2	9.8	1.65	0.20	1.00
		70	4.8	5.0	9.8	1.75	0.17	1.01
	Full jack.	-	4.5	5.0	9.5	1.86	0.12	1.01
Region 4	Algo. 3	30	5.1	6.1	11.2	-0.89	0.25	0.99
		50	5.4	5.5	10.9	-0.53	0.20	0.99
		70	5.4	5.5	10.9	-1.01	0.18	0.99
	Full jack.	-	5.1	5.3	10.4	-0.75	0.14	0.99
Poverty	Algo. 3	30	5.9	5.5	11.4	0.82	0.25	1.00
		50	6.1	4.7	10.8	1.86	0.21	1.00
		70	5.9	4.7	10.6	1.87	0.18	1.01
	Full jack.	-	5.7	4.7	10.4	1.52	0.11	1.01
Nonwhite	Algo. 3	30	6.2	5.0	11.2	-3.16	0.26	0.98
		50	6.4	5.0	11.4	-2.36	0.20	0.98
		70	5.4	4.6	10.0	-2.46	0.16	0.98
	Full jack.	-	5.9	4.7	10.6	-2.60	0.11	0.99
Hispanic	Algo. 3	30	5.3	6.0	11.3	3.94	0.30	1.01
		50	4.5	5.8	10.3	3.79	0.26	1.01
		70	4.5	5.4	9.9	4.39	0.24	1.02
	Full jack.	-	4.3	4.9	9.2	3.95	0.20	1.02
MSA	Algo. 3	30	5.6	5.3	10.9	2.55	0.27	1.00
		50	5.5	5.1	10.6	2.06	0.21	1.01
		70	5.0	5.2	10.2	2.85	0.18	1.01
	Full jack.	-	5.3	5.2	10.5	2.29	0.07	1.01
Non-MSA	Algo. 3	30	6.0	5.9	11.9	-4.85	0.26	0.97
		50	6.2	5.7	11.9	-3.98	0.21	0.97
		70	6.0	5.5	11.5	-4.11	0.18	0.97
	Full jack.	-	6.1	5.5	11.6	-4.11	0.18	0.98

4.6 Theorem Proofs

4.6.1 Proof of Lemma 4.1

For $h = G, \cdots, L-1$, $k = 1, \cdots, D$ and $j = 1, \cdots, G$, let $T_{jk}^{(h)} = S_{jk}^{(h)} - \bar{S}_{\cdot k}^{(h)}$. Clearly, $\sum_{j=1}^{G} T_{jk}^{(h)} = 0$ and $T_{jk}^{(h+1)} = T_{jk}^{(h)} + a_{h+1,k}(x_{h+1,j} - 1/G)$. Thus,

$$\begin{aligned}
\left\{\sum_{j=1}^{G}\left[T_{jk}^{(h+1)}\right]^2\right\}^2 &= \left\{\sum_{j=1}^{G}\left[T_{jk}^{(h)} + a_{h+1,k}(x_{h+1,j} - 1/G)\right]^2\right\}^2 \\
&= \left\{\sum_{j=1}^{G}\left[\left(T_{jk}^{(h)}\right)^2 + 2T_{jk}^{(h)} a_{h+1,k}\left(x_{h+1,j} - 1/G\right) + a_{h+1,k}^2\left(x_{h+1,j} - 1/G\right)^2\right]\right\}^2 \\
&= \left\{\sum_{j=1}^{G}\left(T_{jk}^{(h)}\right)^2 + 2a_{h+1,k}T_{j_{h+1},k} + \frac{G-1}{G}a_{h+1,k}^2\right\}^2 \\
&= \left[\sum_{j=1}^{G}\left(T_{jk}^{(h)}\right)^2\right]^2 + 4a_{h+1,k}^2 T_{j_{h+1},k}^2 + \frac{(G-1)^2}{G^2}a_{h+1,k}^4 \\
&\quad + 4\left[a_{h+1,k}\sum_{j=1}^{G}\left(T_{jk}^{(h)}\right)^2 + \frac{G-1}{G}a_{h+1,k}^3\right]T_{j_{h+1},k} \\
&\quad + \frac{2(G-1)}{G}a_{h+1,k}^2\sum_{j=1}^{G}\left(T_{jk}^{(h)}\right)^2.
\end{aligned}$$

It then follows that for $h = G, \cdots, L-1$ and $g = L - h$,

$$
\begin{aligned}
&\sum_{k=1}^{D}\left\{\frac{u_k^2}{G}\left[\sum_{j=1}^{G}\left(T_{jk}^{(h+1)}\right)^2\right]^2-b_k^{(g)}\sum_{j=1}^{G}\left(T_{jk}^{(h+1)}\right)^2+c_k^{(g)}\right\}\\
&= \sum_{k=1}^{D}\left\{\frac{u_k^2}{G}\left[\sum_{j=1}^{G}\left(T_{jk}^{(h)}\right)^2\right]^2-\left[b_k^{(g)}-\frac{2(G-1)}{G^2}u_k^2a_{h+1,k}^2\right]\sum_{j=1}^{G}\left(T_{jk}^{(h)}\right)^2\right.\\
&\quad \left.+\left[c_k^{(g)}+\frac{(G-1)^2}{G^3}u_k^2a_{h+1,k}^4-\frac{G-1}{G}b_k^{(g)}a_{h+1,k}^2\right]\right\}+\sum_{k=1}^{D}\left\{\frac{4u_k^2a_{h+1,k}^2}{G}\left(T_{j'k}^{(h)}\right)^2\right.\\
&\quad \left.+\left[\frac{4u_k^2a_{h+1,k}}{G}\sum_{j=1}^{G}\left(T_{jk}^{(h)}\right)^2+\frac{G-1}{G^2}u_k^2a_{h+1,k}^3-2b_k^{(g)}a_{h+1,k}\right]T_{j'k}^{(h)}\right\}_{j'=j_{h+1}}\\
&= \sum_{k=1}^{D}\left\{\frac{u_k^2}{G}\left[\sum_{j=1}^{G}\left(T_{jk}^{(h)}\right)^2\right]^2-\left[b_k^{(g)}-\frac{2(G-1)}{G^2}u_k^2a_{h+1,k}^2\right]\sum_{j=1}^{G}\left(T_{jk}^{(h)}\right)^2+c_k^{(g+1)}\right\}+f_{j_{h+1}}^{(h)}\\
&\geq \sum_{k=1}^{D}\left\{\frac{u_k^2}{G}\left[\sum_{j=1}^{G}\left(T_{jk}^{(h)}\right)^2\right]^2-\left[b_k^{(g)}-\frac{2(G-1)}{G^2}u_k^2a_{h+1,k}^2\right]\sum_{j=1}^{G}\left(T_{jk}^{(h)}\right)^2+c_k^{(g+1)}\right\}+\frac{1}{G}\sum_{j=1}^{G}f_j^{(h)}\\
&= \sum_{k=1}^{D}\left\{\frac{u_k^2}{G}\left[\sum_{j=1}^{G}\left(T_{jk}^{(h)}\right)^2\right]^2-\left[b_k^{(g)}-\frac{2(G-1)}{G^2}u_k^2a_{h+1,k}^2-\frac{4u_k^2a_{h+1,k}^2}{G^2}\right]\sum_{j=1}^{G}\left(T_{jk}^{(h)}\right)^2+c_k^{(g+1)}\right\}\\
&= \sum_{k=1}^{D}\left\{\frac{u_k^2}{G}\left[\sum_{j=1}^{G}\left(T_{jk}^{(h)}\right)^2\right]^2-b_k^{(g+1)}\sum_{j=1}^{G}\left(T_{jk}^{(h)}\right)^2+c_k^{(g+1)}\right\}
\end{aligned}
$$

where $f_j^{(h)}$ is defined in Step 2 of Algorithm 2.

By induction, we have

$$
\begin{aligned}
G\sum_{k=1}^{D}\left[(z_k^{(L)})^2-z_k^{(L)}\right] &= \sum_{k=1}^{D}\left\{\frac{u_k^2}{G}\left[\sum_{j=1}^{G}\left(T_{jk}^{(L)}\right)^2\right]^2-b_k^{(1)}\sum_{j=1}^{G}\left(T_{jk}^{(L)}\right)^2+c_k^{(1)}\right\}\\
&\geq \sum_{k=1}^{D}\left\{\frac{u_k^2}{G}\left[\sum_{j=1}^{G}\left(T_{jk}^{(G)}\right)^2\right]^2-b_k^{(L-G+1)}\sum_{j=1}^{G}\left(T_{jk}^{(G)}\right)^2+c_k^{(L-G+1)}\right\}\\
&= \sum_{k=1}^{D}\left\{\frac{u_k^2}{G}\left[\sum_{j=1}^{G}(a_{jk}-\bar{a}_{\cdot k}^{G})^2\right]^2-b_k^{(L-G+1)}\sum_{j=1}^{G}(a_{jk}-\bar{a}_{\cdot k}^{G})^2+c_k^{(L-G+1)}\right\}.
\end{aligned}
$$

The result follows.

4.6.2 Proof of Lemma 4.2

For $h = G, \cdots, L-1$ and $k = 1, \cdots, D$,

$$\begin{aligned}
&\max_{1\le k\le D} \sum_{j=1}^{G} u_k \left(S_{jk}^{(h+1)} - \bar{S}_{\cdot k}^{(h+1)}\right)^2 \\
&\quad \le \max_{1\le k\le D}\left[\sum_{j=1}^{G} u_k \left(S_{jk}^{(h)} - \bar{S}_{\cdot k}^{(h)}\right)^2 + \frac{G-1}{G} u_k a_{h+1,k}^2\right] \\
&\qquad + \max_{1\le k\le D}\left[2u_k a_{h+1,k}\left(S_{j_{h+1},k}^{(h)} - \bar{S}_{\cdot k}^{(h)}\right)\right] \\
&\quad = \max_{1\le k\le D}\left[\sum_{j=1}^{G} u_k \left(S_{jk}^{(h)} - \bar{S}_{\cdot k}^{(h)}\right)^2 + \frac{G-1}{G} u_k a_{h+1,k}^2\right] \\
&\qquad + 2\min_{1\le j\le G}\left\{\max_{1\le k\le D}\left[u_k a_{h+1,k}\left(S_{jk}^{(h)} - \bar{S}_{\cdot k}^{(h)}\right)\right]\right\} \\
&\quad \le \max_{1\le k\le D}\left[\sum_{j=1}^{G} u_k \left(S_{jk}^{(h)} - \bar{S}_{\cdot k}^{(h)}\right)^2 + \frac{G-1}{G} u_k a_{h+1,k}^2\right].
\end{aligned}$$

By induction,

$$\max_{1\le k\le D} \sum_{j=1}^{G} u_k \left(S_{jk}^{(L)} - \bar{S}_{\cdot k}^{(L)}\right)^2 \le \max_{1\le k\le D}\left[\sum_{j=1}^{G} u_k \left(a_{jk} - \bar{a}_{\cdot k}^{G}\right)^2 + \frac{G-1}{G}\sum_{i=G+1}^{L} u_k a_{ik}^2\right].$$

Therefore, we have

$$\begin{aligned}
df_{\min} &= \min_{1\le k\le D} df_k = \min_{1\le k\le D} \frac{\left(\sum_{j=1}^{G} S_{jk}\right)^2}{\sum_{j=1}^{G} S_{jk}^2} \\
&= \frac{G}{1 + \max_{1\le k\le D}\left[\sum_{j=1}^{G} u_k \left(S_{jk}^{(L)} - \bar{S}_{\cdot k}^{(L)}\right)^2\right]/G} \\
&\ge \frac{G}{1 + \max_{1\le k\le D}\left\{u_k\left[\sum_{j=1}^{G}\left(a_{jk} - \bar{a}_{\cdot k}^{G}\right)^2 + \frac{G-1}{G}\sum_{i=G+1}^{L} a_{ik}^2\right]\right\}/G}.
\end{aligned}$$

This completes the proof.

Chapter 5

Disclosure Control and Variance Estimation

In recent years, statistical agencies have seen a noticeably increasing demand from a variety of external users for the data they collect. Among the typical users are policy makers, who need up-to-date social and economic statistics to help them make key decisions, and academic researchers requiring more detailed data at the micro level to conduct their own statistical analyses. Unfortunately, the potential risk of disclosing individual information will also increase dramatically if little care has been taken towards confidentiality concerns whenever a data file is publicly released. In this chapter, we will review recent accomplishments on disclosure control in general. We will then examine the issue as it relates to variance estimation motivated by a real survey. We will develop a simple method for breaking confidentiality by using only the design and replicate weights, without knowledge of what replicate method was used, thus emphasizing the extent of the practical problem. We will propose some algorithmic approaches in an effort to minimize the risk of disclosure. At the end of the chapter, we will present an application of the proposed approaches to a real survey.

5.1 Basic Concepts of Disclosure Control

In President's Commission on Federal Statistics (1971), confidentiality is explained as: it is prohibited to release data in a manner that would allow public identification of the respondent or would in any way be harmful to him. Disclosure occurs when confidential information is revealed. There are three types of disclosure: identity disclosure, attribute disclosure and inferential disclosure. In this thesis, only identity disclosure is considered because our major concern is to limit the disclosure of confidential information through public use data files, where identification is generally regarded as disclosure. In principle, information that directly or indirectly reveals the identity of the respondents has to be suppressed. As a basic practice, strata and psu identifiers are not released as they are not needed for many point estimates provided the design weights are available. In addition, a number of basic data manipulation techniques can be used to accomplish the goal of disclosure control:

1) *Top and/or bottom coding some key variables.* The tails of continuous distributions for ordinal variables or the end categories for categorical variables have much higher risks to be identified as fewer cases fall into those regions. Top and/or bottom coding, or collapsing those regions reduce such risks.

2) *Collapsing response categories (global coding).* This technique is a generalization of top/bottom coding as it collapses any adjacent regions which have rare observations and subsequently high identification risks.

3) *Locally suppressing information of some variables.* For some variables blocking values of outliers from external use is the only feasible way to protect confidentiality. Subsequent information loss may be partially compensated by imputation at later stages.

More sophisticated and systematic approaches on disclosure control will be discussed in detail throughout this chapter.

5.2 Methods on Disclosure Control in General

In this section, we review recent literature on disclosure control methods appropriate to microdata.

5.2.1 Additive Noise Methods

Fuller (1993) considers a variety of masking methods by adding error to data elements prior to release. These fall generally within the class of measurement error methods. Kim and Winkler (1997) presents a two-stage disclosure limitation strategy, applied to matched CPS-IRS data. Moore (1996a) provides a critical examination of the degree of confidentiality protection and analytic usefulness provided by the Kim and Winkler (1997) method. Winkler (1998) compares the effectiveness of a number of competing disclosure limitation methodologies to preserve both confidentiality and analytic usefulness. The methods considered include the additive-noise and swapping techniques of Kim and Winkler (1997) and the additive-noise approach of Fuller (1993). Duncan and Mukherjee (1998) derives an optimal disclosure limitation strategy for statistical databases - i.e., micro-databases which respond to queries with aggregate statistics. Evans et al. (1998) presents an additive-noise method for disclosure limitation which is appropriate to establishment tabular data. Pursey (1999) discusses the disclosure control methods developed and implemented by Statistics Canada to release a Public Use Microdata File of financial data from small businesses.

5.2.2 Multiple Imputation and Related Methods

Rubin (1993) is the first paper to suggest the use of multiple imputation techniques for disclosure limitation for microdata analyses. His radical suggestion - to release only synthetic data generated from actual data by multiple imputation - is motivated by the increase in the demand for public use microdata, and increasing concern about the confidentiality of such data. Later Fienberg (1994) proposes a method of confidentiality protection in the spirit of Rubin (1993). Whereas Rubin (1993) suggests generating synthetic microdata sets by multiple imputation, Fienberg (1994)

suggests generating synthetic microdata by bootstrap methods. This method retains many of the desirable properties of Rubin's (1993) proposal - namely disclosure risk is reduced because only synthetic data are released, and the resultant microdata can be analyzed using standard statistical methods. In a series of articles, Kennickell (1991, 1997, 1998, 2000) describes the Federal Reserve Imputation Technique Zeta (FRITZ), used for both missing value imputation and disclosure limitation in the Survey of Consumer Finances (SCF). The SCF is a triennial survey administered by the Federal Reserve Board to collect detailed information on all household assets and liabilities.

5.2.3 Data Swapping Methods

Moore (1996b) presents a brief overview of data swapping techniques for disclosure limitation, a more sophisticated technique than found elsewhere in the literature and an algorithm for a controlled data swap based on the rank-based proximity swap of Greenberg (1987). Moore (1996c) also suggests modifications to the Confidentiality Edit, the data-swapping procedure used for disclosure limitation in the 1990 Decennial Census and presents two measures of the degree of distortion induced by the swap, and an algorithm to minimize this distortion. Takemura (2002) proposes local recoding and record swapping based on the optimum matching of the records, where pairs of close records are formed and observed values are recoded or swapped within each pair.

5.3 Disclosure Control and Variance Estimation

All disclosure limitation methods introduced in the previous section focus on data disclosure via viewing only the design weights and the data when providing record level microdata for public use. There is little literature available on how to protect confidentiality when statistical analysis is being conducted by the end users and valid variance estimation is needed, for user-constructed estimators. Mayda et al. (1997) examines the relationship between variance estimation and confidentiality protection

in surveys with complex designs. In consideration of the Canadian National Population Health Survey (NPHS), a longitudinal survey with a multi-stage clustered design, they present two concerns which occurred in the process of releasing a microdata file for public use from such a complex survey: 1) specific design information such as stratum and cluster identifiers should be removed from the data due to the extremely detailed level of geographic information they represented; 2) providing cluster information could allow users to reconstitute households, increasing the probability of identifying individuals. However, it is easy to see that, without knowing the stratum and cluster identifiers, the external users will not be able to correctly compute variances using a linearization method. This reflects another aspect of the conflict between providing high quality data and protecting confidentiality. They propose an approach to resolve the conflict. Specifically, strata and clusters are collapsed to form "super-strata" and "super-clusters". Only the super-strata and super-cluster identifiers are included in the public use file, which allows researchers to obtain unbiased variance estimates under certain conditions while protecting confidentiality. This approach is similar in spirit to the replication reduction proposed in the previous chapters for creating replicate weights and if done using the methods proposed there, as we have already demonstrated, when the number of strata is large, the loss in precision of variance estimates could be limited. However, in practice there are sometimes only a small number of strata available for the purpose of variance estimation, in which case the collapsing strata strategy will be inappropriate as the loss of degrees of freedom in this case could greatly affect the precision of variance estimates. This was the case in the National Health and Nutrition Examination Survey (NHANES) which will be described and considered in the next section.

Because of these concerns, it is very tempting for statistical agencies to generate and release a set of replicate weights along with the raw data to the end users without providing the stratum and psu identifiers. Hopefully, the end users can utilize the replicate weights to obtain variance estimates and yet are unable to discover any confidential information. That is, release the data much like in Table 2.1. This is not so, as has been pointed out for specific replication methods. In light of this, Yung (1997) proposes an approach that constructs a set of average bootstrap replicate weights in

an effort to obtain valid variance estimates from the public use microdata files while still respecting confidentiality constraints. However, as we will demonstrate, the replicate weights, thanks to their specific deletion structure, no matter which replication method is applied, can still be used to reconstruct the stratum and psu identifiers. In the next section, we will discuss the connection between replicate weights and stratum/psu identifiers, examine existing methods for reconstructing these identifiers from replicate weights that are specific to a particular replication method and then develop a simple clustering approach applicable to any replication method.

5.4 Replicate Weights and Stratum/psu Identifiers

The connection between replicate weights and stratum/psu identifiers is embedded in the way the replicate weights are originally generated. We create Table 5.1 by calculating the ratios of replicate weights and design weights for all R sets of replicates.

Table 5.1: The Replicate Weights/Design Weights Ratio

Sample	Replicate			
	1	2	$\cdots$	R
1	$w_{1(1)}/w_1$	$w_{1(2)}/w_1$	$\cdots$	$w_{1(R)}/w_1$
2	$w_{2(1)}/w_2$	$w_{2(2)}/w_2$	$\cdots$	$w_{2(R)}/w_2$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
m	$w_{m(1)}/w_m$	$w_{m(2)}/w_m$	$\cdots$	$w_{m(R)}/w_m$

5.4.1 Jackknife Replicate Weights

For jackknife replicate weights formed by (2.6), the r-th column of weight ratios from Table 5.1, corresponding to the r-th replicate, will only consist of three different values: 0, 1 and a positive constant, say $c_r(> 1)$. We can easily conclude the following:

1) all the sample units associated with 0 are from the same psu; and

2) all the sample units associated with c_r are from the same stratum as the psu's in 1).

This means that we can easily determine all cluster (psu) and stratum identifiers by examining all R sets of replicate weight/sampling weight ratios, respectively.

5.4.2 BRR Replicate Weights

For BRR replicate weights generated by (2.10), it seems more difficult to reconstruct the stratum and/or psu identifiers as each BRR replicate consists of half of the sampled psu's from each stratum. In other words, there will be replicate weights of zero from all strata and half of the sampled psu's simultaneously. However, Shah (2001) presents an algorithm which will accurately reconstruct stratum and psu identifiers provided that no adjustments to weights such as non-response, post-stratification or other calibration to known totals are made. The basic idea of Shah's approach is to modify Table 5.1 and create a new table (see Table 5.2) with its cells, $\delta_{s(r)}$, defined as

$$\delta_{s(r)} = \begin{cases} 1 & \text{if } w_{s(r)} > 0, \\ 0 & \text{otherwise,} \end{cases}$$

where $w_{s(r)}$ is the corresponding replicate weight introduced in Table 2.1.

Table 5.2: The Matrix Representation of the Indicator Variable

Sample	Replicate			
	1	2	$\cdots$	R
1	$\delta_{1(1)}$	$\delta_{1(2)}$	$\cdots$	$\delta_{1(R)}$
2	$\delta_{2(1)}$	$\delta_{2(2)}$	$\cdots$	$\delta_{2(R)}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
m	$\delta_{m(1)}$	$\delta_{m(2)}$	$\cdots$	$\delta_{m(R)}$

We can easily conclude the following

1) if any two rows of Table 5.2 are identical, then the two corresponding sample units are from the same psu; and

2) if any two rows of Table 5.2 are complementary to each other, meaning that the sum of two rows is a vector of 1's, the two corresponding units are from the same stratum.

5.4.3 Bootstrap Replicate Weights

For bootstrap replicate weights generated by (2.12), it seems even more difficult to reconstruct the stratum and/or psu identifiers than the case of BRR or jackknife. Yung (1997) considers a simple case of $m_h^* = n_h - 1$ and claims that the cluster membership can be identified in this case. In fact, for any bootstrap replicate b, at least one cluster from each stratum, say the i-th cluster within each stratum h, will have bootstrap final weights equal to zero, that is, $w_{hil(b)} = (n_h/(n_h-1))m_{hi(b)}w_{hil} = 0$ when $m_{hi(b)} = 0$. By introducing an indicator variable identical to the one in Table 5.2, Yung (1997) argues that the cluster identifiers can be obtained by examining all B bootstrap replicate weights using a similar approach as that of Shah (2001). Yung (1997) then goes on to propose a mean bootstrap method in an effort to hide the cluster identifiers from the end users. The key idea is to repeat the resampling procedure for each bootstrap replicate enough times, say S times. Denote the number of times the (hi)-th cluster is drawn as $m_{hi(b)s}$, where $s = 1, \cdots, S$. Then let $m^*_{hi(b)} = \sum_s m_{hi(b)s}/S$ and use it to replace the previous $m_{hi(b)}$ in (2.12). Therefore, all the final bootstrap weights will be nonzero provided at least some $m_{hi(b)s}$ are positive for each replicate b.

5.4.4 Using Clustering to Reconstruct Psu Identifiers

So far we have discussed existing methods for breaking confidentiality through replicate weights provided that we know which replication method is applied to generate such replicate weights. We will demonstrate that at least the psu identifiers can be

easily reconstructed even if we do not have that information. An important fact is that, no matter which replication method is applied, the relative change of design weights for any replicate will be the same for all sampled units within the same psu, or, in other words, the values of elements in the same column of Table 5.1 will be the same if they are from the same psu. Although we cannot uniquely identify all sampled units of a specific psu in one replicate, we are able to accomplish this by viewing all R sets of replicate weights. In fact, one can argue that two rows in Table 5.1 are identical if and only if the associated sampled units are from the same psu, provided the number of replicates is comparable to the number of clusters. Note that Yung's mean bootstrap method will not help protect confidentiality based on this argument because the averaged number of times a cluster is sampled from the r-th replicate, even greater than zero, is still a constant subject to a fixed replicate and cluster combination, resulting in the same value in Table 5.2 for sampled units from the same cluster. In fact, it is even worse. By averaging the number of times a cluster is sampled in the same replicate, Yung's approach inflates the chance of differentiating one cluster from the others, meaning that we need a much smaller number of replicates to identify cluster memberships.

Another important issue is that in practice, the replicate weights will likely be adjusted for nonresponse and poststratification, making the psu reconstruction procedure a little more complex. Inspired by this, one might consider whether or not the additive-noise technique, described in the previous section, would also help protect confidentiality when applied to replicate weights.

Combining all these thoughts, we will view the problem a bit differently. Treat the rows in Table 5.1 as points in an R dimensional space, and the m sampled units will shrink to replicates of n distinct points in that space provided that no weight adjustments have been made. If we add some noise or perturbation to the original replicate weights, including nonresponse and poststratification adjustments as special cases, it is very likely that we will still observe n distinguishable clusters in the R dimensional space as the magnitude of the perturbations should be small relative to that of the distance between any two originally distinct points. If this is not so, then intuitively the magnitude of the added variability due to these random perturbations

will be comparable to the variability measured by the replication method.

Knowing little about clustering algorithms other than that they exist, we explored Splus for such and found a command called "hclust" which stands for hierarchical clustering. This function takes multi-variate data and forms cluster trees. We did not try any thing more sophisticated to make the point that even someone with rudimentary skills and tools could do this. We applied this algorithm to a 3 year set of NHANES data which had had Fay's BRR replicate weights created. Reweighting had been done as had some collapsing of a few psu's so as to be able to apply the BRR, and one large psu was split into 3. Using only the adjusted design weights and the 24 sets of replication weights the program assigned the 5,000 sampled units to psu's with only a 6% error rate. By closely examining the 6% sampled units with the misspecified psu identifiers from reconstruction, we learned that it was caused by splitting the large psu into 3 psu's.

We also considered the introduction of a random noise ε to replicate weights. We can express the perturbed replicate weights as

$$w^*_{hil(r)} = w_{hil(r)}(1 + \varepsilon_{hil}) = b_{hil(r)} w_{hil}(1 + \varepsilon_{hil}), \tag{5.1}$$

where $b_{hil(r)} = w_{hil(r)}/w_{hil}$ are the elements in Table 5.1 and ε_{hil} are identically distributed with mean 0 and variance σ^2. For simplicity, we let $\varepsilon_{hil} \sim U(-\Delta, \Delta)$ for $\Delta = 0.1, 0.2, 0.3, 0.4, 0.5$, and applied it to the same 5,000 sampled units as above. The resulting percentages of misspecified psu identifiers were 6%, 5%, 6%, 26% and 50%, respectively. This result means that, unless we perturb the weights by at least $35\% \sim 40\%$, we will not be able to effectively protect the original psu identifiers. We consider this further in Chapter 6.

To illustrate our point that Yung's method not only fails to add confidentiality protection but in fact makes it much easier to reconstruct the original cluster identifiers, we designed the following simulation study: 1) create 100 sets of bootstrap weights, each of which was the average of 20, precisely as was done in the Yung (1997) paper; 2) then apply the method using only 2000 of the weights and only 2, 3, 4, and 5 of the replicates. It turns out that the error rate for assigning units to original psu's is 2.5% using 2 replicates and 0 using 3 or more replicates, respectively. We repeat the

simulation without averaging weights, that is using the ordinary bootstrap method. The error rates are 47.5%, 28%, 5.5% and 1.5% using 2 to 5 replicates, respectively. The clustering algorithm performs very well in terms of reconstructing original psu's in both cases.

In summary, it is evident that the replicate weights, no matter how they are created, with or without weight adjustments, can be used to reconstruct the original psu identifiers quite easily, which essentially eliminates statistical agencies' hope to provide the replicate weights in the publicly released data file and motivates the research interests in new approaches to disclosure control on variance estimation.

Another point to make is that the stratum identifiers will be harder to reconstruct, when little is known about the replication method. We will discuss possible ways of reconstructing stratum identifiers in the future research section of this thesis. One should note, however, that confidentiality of stratum indicators may be less important in many cases. For example, as we will describe in the next section, for the NHANES survey the psu's are counties or metropolitan areas. Thus psu census data is available from other sources.

5.5 Disclosure Control in the NHANES Survey

After the previous detailed consideration of the connection between replicate weights and stratum/psu identifiers, we realize that supplying replicate weights with publicly released data sets is not a solution to confidentiality protection. In this section, we review some currently used techniques for disclosure control in the National Health and Nutrition Examination Surveys (NHANES) and evaluate the resulting effect on variance estimation.

5.5.1 The NHANES Survey

NHANES is a continuous, ongoing, annual survey of the noninstitutionalized civilian population of the U.S. To meet the objectives of the Survey Integration Plan of the Department of Health and Human Services (DHHS), the NHANES 1999 to 2001

surveys are being linked to the 1995 NHIS at the primary sampling unit (PSU) level as well as the content (i.e., questions and questionnaire sections) level. Starting with the 2002 survey, NHANES will be linked to the NHIS at the content level only. Each single year and any combination of consecutive years comprise a nationally representative sample of the U.S. population. This design will facilitate potential linkage to other health and nutrition surveys that provide yearly estimates and will allow aggregate level national estimates from NHANES each year.

A four-stage sample is selected for NHANES. Within each of the selected psu's, an average of 24 segments are selected and a subsample of the households within these segments are selected and screened. Within the screened households, members of particular race/ethnicity-sex-age subdomains are identified as potential sampled persons; all other members of the household are excluded.

5.5.2 Psu-Splitting

Table 5.3: Baseline Replication Design

Certainty status	PSU	Replicate				
		1	2	···	26	27
Noncertainty psu's	A	×				
	B		×			
	···					
Certainty psu's	Z1,Z2 1^{st} seg				×	
	Z1,Z2 2^{nd} seg					×
	Z1,Z2 3^{rd} seg				×	
	Z1,Z2 4^{th} seg					×
	···					

Because no explicit stratification was used to select the psu's from the two panels of NHIS and because of the small number of psu's in the sample, the delete-1 jackknife was used to create replicates for variance estimation for the analysis of the NHANES

1999-2000 data; for noncertainty psu's, the psu is the variance unit, and for the certainty psu's, two variance units were formed by alternating segments. Table 5.3 (reproduced from Figure 1 of Dohrmann et al., 2002) depicts the creation of replicates in the baseline design. The × area denotes that in creating the given replicate, the particular psu or the particular segment was dropped.

Various methods for splitting each psu into two creating a total of 52 pseudo-psu's to which the delete-1 jackknife could be applied were considered and the impact on the performance of the resulting jackknife variance estimates and on disclosure of original psu indicators was examined (see Dohrmann et al., 2002, for more detail). The final chosen method for the 1999-2000 NHANES release (termed the clustered-split psu alternative in Dohrmann et al.) entailed ordering the ssu's on minority density and then assigning the first half within a psu to one pseudo-psu and the second half to another, as depicted in Table 5.4 (reproduced from Figure 3 of Dohrmann et al., 2002). Due to the ordering on minority density one expects that the resulting pseudo-psu's formed from this method will not have the same characteristics as the full psu. In addition, the order of the replicates is then scrambled to further ensure confidentiality.

There is little practical difference in terms of confidentiality between supplying the end-user the 52 sets of jackknife replicate weights or giving the pseudo-psu indicators, as one can easily obtain one from the other via the method of the previous section. The questions are: i) is it now easy to re-match units to original psu's; and ii) will the resulting jackknife variance estimate still perform reasonably well on various characteristics.

The conclusions in Dohrmann et al. (2002) are mixed. The protection of confidentiality seemed more satisfactory than did the performance of the resulting variance estimator. On the 70 characteristics investigated, the jackknife variance estimates were on average 20% low and showed a rather striking pattern when compared to the baseline design effect (see Figure 7 of Dohrmann et al., 2002). We will only consider the first aspect in this thesis. For the second aspect, the pattern is related to the properties of design effects (Inho Park (2004), unpublished Westat report) and will not be discussed, except to say that the pattern is to be expected and a better plot for evaluation would be to plot the estimated standard errors before and after splitting.

Table 5.4: Clustered-split PSU Replication Design

Certainty status	PSU	Replicate				
		1	2	$\cdots$	51	52
Noncertainty psu's	A 1^{st} seg	$\times$				
	A 2^{nd} seg	$\times$				
	$\cdots$	$\times$				
	A 12^{th} seg	$\times$				
	A 13^{th} seg		$\times$			
	A 14^{th} seg		$\times$			
	$\cdots$		$\times$			
	A 24^{th} seg		$\times$			
	$\cdots$					
Certainty psu's	Z1,Z2 1^{st} seg				$\times$	
	Z1,Z2 2^{nd} seg					$\times$
	Z1,Z2 3^{rd} seg				$\times$	
	Z1,Z2 4^{th} seg					$\times$
	$\cdots$					

5.5.3 The Evaluation of Psu-Splitting Effect

Before examining the various aspects of psu-splitting, we will discuss the method of variance estimation decided upon for the 2001-2002 NHANES and subsequent releases, as a decision to change the basic methodology was made. Instead of using a delete-1 jackknife, the noncertainty psu's were paired to form variance strata and the ssu's within certainty psu's were similarly treated to form L variance strata. If this pairing is done randomly, it is not difficult to show that, for the purposes of variance estimation, treating the design as if it were a stratified multistage design with 2 psu's per stratum selected with replacement and applying a BRR or jackknife will yield unbiased variance estimates for linear estimators provided the first-stage sampling fraction is negligible (see Rao and Shao, 1996; and Rust and Rao, 1996 for related

discussion).

With this in mind, to examine the observed underestimation using psu-splitting, we will consider a general stratified multi-stage design with $n_h = 2$ psu's selected with probabilities p_{hi} with replacement, where $h = 1, ..., L$ denote the strata and $i = 1, 2$ indicate the sampled psu's. Let $\hat{Y} = \sum_{hik} w_{hik} y_{hik}$ be a linear unbiased estimator of population total Y, where $k \in G_{hi}$ and G_{hi} represents the set of sampled ultimate units.

Consider the BRR. To construct a BRR variance estimator for $\hat{Y}$, we need to create R sets of replicate weights. The r-th set of replicate weights are defined as $w_{h1k(r)} = (1 + \delta_{rh}) w_{h1k}$ and $w_{h2k(r)} = (1 - \delta_{rh}) w_{h2k}$, for $h = 1, ..., L$ and $k \in G_{hi}$, where the δ_{rh} is the element in the r-th row and h-th column of a matrix of ones and negative ones with orthogonal columns, as was described in Chapter 2. Typically, this matrix is comprised of a set of L columns from an $R \times R$ Hadamard matrix with $L \leq R \leq L + 3$. In this case, half the replicate weights in each replicate are twice the original weight and half are zero. Often in practice, the more general Fay's BRR method is applied instead of BRR with regard to weight adjustment (Judkins, 1990).

Replacing w_{hik} by the Fay BRR weights $w_{hik(r)}$ in $\hat{Y}$ we get $\hat{Y}_{(r)}$ and the BRR variance estimator is given by

$$v_{BRR-F} = \frac{1}{R(1-\epsilon)^2} \sum_{r=1}^{R} (\hat{Y}_{(r)} - \hat{Y})^2. \tag{5.2}$$

We can write $\hat{Y} = \sum_{h=1}^{L} \bar{r}_h$, where $\bar{r}_h = (r_{h1} + r_{h2})/2$ and $r_{hi} = \sum_{k \in G_{hi}} 2 w_{hik} y_{hik}$. It is then not difficult to show that (see section 5.9)

$$v_{BRR-F} = v(\hat{Y}) = \sum_{h=1}^{L} \left(\frac{r_{h1} - r_{h2}}{2} \right)^2. \tag{5.3}$$

In fact, it is not difficult to show that the usual linearization variance estimator and the JK2 jackknife variance estimator are equal to v_{BRR-F} in this case. Thus, we will evaluate the impact of psu-splitting on $v(\hat{Y})$ in (5.3) without further specifying variance estimation methods the end-user might choose.

To perform psu-splitting in this context, the hi-th psu is split into two sets of ultimate units, $G_{hi,1}$ and $G_{hi,2}$ say, where $G_{hi,1} \cup G_{hi,2} = G_{hi}$. One from each of

these pairs are placed together to form a new variance strata of two pseudo-psu's, thus creating two new variance strata from each original variance stratum. Then $r_{hi} = \sum_{k \in G_{hi,1}} n_h w_{hik} y_{hik} + \sum_{k \in G_{hi,2}} n_h w_{hik} y_{hik} = r_{1,hi} + r_{2,hi}$, and $\hat{Y} = \sum_{h=1}^{L} [(r_{1,h1} + r_{2,h2}) + (r_{2,h1} + r_{1,h2})]$ in obvious notation, and applying Fay's BRR to these new variance strata and pseudo-psu's yields

$$\begin{aligned} v^*(\hat{Y}) &= \sum_{h=1}^{L} \left[\left(\frac{r_{1,h1} - r_{2,h2}}{2} \right)^2 + \left(\frac{r_{1,h2} - r_{2,h1}}{2} \right)^2 \right] \\ &= v(\hat{Y}) - (1/2) \sum_h (r_{1,h1} - r_{2,h2})(r_{2,h1} - r_{1,h2}). \end{aligned} \tag{5.4}$$

The performance of the split-psu replication method will depend upon the second term in (5.4) which depends upon how the psu's are split. One could attempt to split the psu's so as to make the second term on the right hand side of (5.4) as small as possible, or at least positive to be conservative. A similar result can be obtained for the delete-1 jackknife used in the 1999-2000 NHANES release and partially explains the under-estimation reported in Dohrmann et al. (2002). More careful consideration of the second term has the potential to reduce the problem. However, the NHANES survey, as with most other surveys, has many characteristics of interest and it would be difficult to ensure this generally for all of them. Even if it were possible, it may not be best to create more pseudo-psu's and/or variance strata than there were in the original survey as the end-user may then be over-confident as to the degrees of freedom of the resulting variance estimator.

An obvious alternative is to recombine the split psu's with each other. That is, let $G^*_{h1} = G_{h1,1} \cup G_{h2,1}$ and $G^*_{h2} = G_{h1,2} \cup G_{h2,2}$ be the two recombined pseudo-psu's for stratum h and apply Fay's BRR. It is not difficult to show that the result is

$$v^*(\hat{Y}) = v(\hat{Y}) - \sum_h (r_{1,h1} - r_{2,h2})(r_{2,h1} - r_{1,h2}). \tag{5.5}$$

The proof of (5.5) is given in section 5.9. Thus, a similar strategy for splitting the psu's would be recommended before recombining.

The key point to realize is that this strategy of psu splitting and recombining is merely one method of changing ssu assignment to psu's. If the total of the ssu's

swapped between $h1$ and $h2$ are equal, i.e., $r_{1,h2} = r_{2,h1}$, the variance estimator will be unchanged. However, as with psu-splitting, it will be difficult to ensure this holds on a large number of characteristics. This idea can be generalized to include swapping ssu's or in the case of multi-stages even ultimate units between different psu's, when constructing the replicate weights or pseudo-psu's for the purpose of variance estimation. By doing so, we may be able to do less swapping and have a better chance to limit the possibility of data disclosure while disturbing the variance estimator less.

5.6 Proposed Approaches

From the previous section we have learned that psu splitting and recombining can be interpreted as a special case of an ssu swapping approach, which still has a lot of potential for improvement. We summarize that a good ssu swapping algorithm should meet the following criteria:

1. Since one of our major goals is to hide the original psu indicators from the end users, a considerable portion of ssu's should be switched from each original psu, making it unidentifiable for any cluster analysis of ssu patterns. Furthermore, in any formed pseudo-psu, the number of ssu's from any original psu should not be inordinately large.

2. Another goal is to limit the resulting bias of the variance estimator. As demonstrated in the previous section, we could achieve this goal by maintaining approximately equal r_j, $j = 1, \cdots, n$, for the pseudo-psu's.

Based on these criteria, we present two types of approaches: one is a two stage approach. In the first stage we pair like segments together from different psu's; at stage two we select a user-specified proportion of those paired segments and swap them between psu's. The second type is to sequentially search for the most like pair of segments from different psu's available at the current step and swap them; repeat this procedure until a required proportion of segments has been swapped. This is more like the grouping algorithms of Chapters 3 and 4.

5.6.1 Match-and-Swap Approach

One approach to swapping ssu's is to adapt methods for local record swapping which have been proposed for the purpose of disclosure control on the micro data set itself, as described in section 5.2.3. For example, Takemura (2002) suggests using an algorithm due to Edmonds (1965), or an approximation to it, to pair elements in the data on the basis of a distance measure between records (vector of y characteristics). Then elements of the so paired records could be swapped. This could be quite easily adapted to our problem as follows. We add the psu indicator as one of the components of the record. We then adjust the distance measure so that if two records have the same psu indicator value, i.e. they are from the same psu, the measure of distance becomes extremely large. This prohibits records from the same psu being paired together. We then apply the pairing algorithm. Once pairs have been formed, we choose $\alpha\%$ of the pairs and switch their psu indicator.

There are a number of possible algorithms for *matching*. We chose to use a publicly available implementation of a version of Edmonds' algorithm called WMATCH (see Gabow, 1973). With the *matching* obtained and Criterion 2 met, now the question is which pairs should be switched in order to satisfy Criterion 1. We propose a linear programming approach aiming to balance the proportion of switched pairs of ssu's over all psu's. Let n_i be the number of ssu's in the i-th psu, $i = 1, \cdots, n$, and $n_0 = \sum n_i/2$ be the number of matched pairs. We then let $(a_{i1}, a_{i2}), i = 1, \cdots, n_0$ denote the matched pairs, (p_{i1}, p_{i2}) be their psu indicators and $d_1, \cdots, d_{n_0}$ their corresponding distance measures. Then solve the linear programming (LP) problem,

$$\begin{aligned} &\min_{x_j \in \{0,1\}} d_1 \cdot x_1 + \cdots + d_{n_0} \cdot x_{n_0} \\ s.t. \quad &\sum_{j \in P_i} x_j \geq \alpha \cdot n_i, \quad i = 1, \cdots, n_0. \end{aligned} \tag{5.6}$$

The optimization procedure will be accomplished through indicator variables x_j, $j = 1, \cdots, n_0$, which determine whether the matched pairs of ssu's are being switched or not. It is easy to recognize that any feasible solution will satisfy the requirement of $\alpha\%$ switching proportion for all psu's and the optimal solution will further minimize the overall distance of switched pairs of ssu's.

The outline of steps for the entire algorithm is given as follows:

Algorithm 1

Step 1. Define a distance measure;

Step 2. Apply a matching algorithm to pair ssu's from different psu's;

Step 3. Solve the LP problem (5.6) to obtain $\alpha\%$ switching.

The following points should be noted:

1. In the previous description, we assume that all ssu's can be paired (called complete *matching*) in a matching algorithm. This is not quite true in practice, because: (a) The computational time can be extremely large. Thus, we take Takemura's suggestion to consider only the K nearest neighbors of each ssu for matching, which implies that a complete *matching* may not exist; and (b) the free source code (WMATCH) we obtained usually cannot achieve complete *matching*, especially when only K nearest neighbors of each ssu are included.

2. If, in practice the required switching percentage $\alpha\%$ is small, we may not need a complete matching. There will be enough matched pairs to choose an adequate switching. From our experience, if the ratio $n_0/\sum n_i$ is no less than α, it is almost assured that enough switches can be made. Note that n_0 here, as the total number of matched pairs, will be less than $\sum n_i/2$ if we do not have a complete matching.

3. If a complete matching is needed, we can still modify the proposed algorithm to meet the requirement. In fact, we can replace Step 2 with:

   Step 2.1 Obtain a list of all ssu's associated with K nearest neighbors and then apply a *matching* algorithm to the list to obtain the *matching*.

   Step 2.2 If the obtained *matching* is complete, go to Step 3. Otherwise, reconstruct a list of all unmatched ssu's with $K' \leq K$ nearest neighbors among them.

Step 2.3 Apply the *matching* algorithm to the current list and repeat Step 2.2.

One could use other matching strategies/algorithms. In fact, when we described the general match-and-swap approach to methodologists at Westat Inc., for practical reasons, they chose to use software based on record-linkage to match and to randomly choose matched pairs for swapping. This will be described in detail in section 5.8.

5.6.2 Sequential Swapping Approach

The performance of the match-and-swap approach is heavily dependent on how well the matching is accomplished. When the number of ssu's is large, or if we want to extend this approach to record level swapping, we may face severe computational burden at the matching stage. In addition, the lack of flexibility at the swapping stage also makes the match-and-swap approach less than attractive. For example, if an $\alpha\%$ swapping is needed, we may like to have the proportion of ssu's from each psu to be swapped to as many other psu's as possible and subsequently reduce the risk of identifying certain swapping patterns between any two psu's. However, such a goal is hard to achieve if the match-and-swap approach is applied as the number of candidate pairs of ssu's is limited even when a complete matching is available. Thus, we also propose a sequential swapping algorithm in an effort to resolve these concerns.

An idea of sequential swapping is that, instead of swapping a proportion of matched pairs of ssu's at the same time, we establish a rule to determine the best pair of ssu's for swapping under some optimality criterion at the current step and then repeat the swapping steps until enough ssu's have been swapped while the criteria mentioned above are all satisfied. This is more in the spirit of the algorithms used for grouping in Chapters 3 and 4. The question is, how to find a good rule in order to swap ssu's between psu's. Firstly, to satisfy the requirement for the $\alpha\%$ swapping for each psu, we would like to choose the psu with the least current swapping percentage (i.e. the highest percentage of original ssu's remaining) as one of the two psu's to be swapped and set up our stopping criterion as: no psu still retains more than $(1-\alpha)\%$ of its original ssu's. Then, we choose the best pair of ssu's out of all possible pairs which consist of one from the chosen psu's and the other from any other psu in terms of

distance measure for the pairs. Denote $l_i = \lfloor \alpha * n_i \rfloor + 1$ as the minimum number of ssu's to be swapped from psu i and $u_{ij} = \lfloor \beta * l_i \rfloor$ as the maximum number of ssu's to be swapped from psu j to psu i, where $\lfloor \cdot \rfloor$ stands for the operation of truncating a number to the largest integer less than or equal to it and β is a tuning parameter which prevents the swapping rate between any two psu's from being inordinately large.

The steps for this algorithm are given as follows:

Algorithm 2

Step 1 Preparation

Step 1.1 Compute the distance D_{ij} for any pair (i, j) of ssu's;

Step 1.2 Calculate l_i, u_{ij}, $i, j = 1, \cdots, n$;

Step 1.3 Let $L_i = 0$ and $U_{ij} = u_{ij}$, $i, j = 1, \cdots, n$;

Step 2 Determine the first psu indicator as $i_0 = arg\min_{1 \le i \le n} L_i$:

Step 2.1 Among other psu's in $J = \{j \neq i_0 : U_{i_0 j} > 0\}$, choose j_0 such that the distance between the pair of ssu's from psu i_0 and j_0, respectively, is minimized.

Step 2.2 Remove both ssu's from the current list;

Step 2.3 Let $L_{i_0} = L_{i_0} + 1$, $L_{j_0} = L_{j_0} + 1$ and $U_{i_0 j_0} = U_{i_0 j_0} - 1$.

Step 3 If $L_i \ge l_i$ for all $i = 1, \cdots, n$, stop; otherwise, repeat step 2.

Algorithm 2 is simple in concept and very fast because there are only a limited number of comparisons needed for each ssu pair swapping step. Its simplicity also makes it very flexible to accommodate different constraints or requirements if needed. These properties are preferable in practice because a real data swapping procedure takes a lot of time and effort and needs to be repeated many times in order to obtain a satisfactory final swapping result, since typically only some of the characteristics are used for matching, and one must examine the resulting performance of variance estimators on all characteristics. In the next section, we will use a small simulation study to demonstrate these properties.

5.7 A Small Simulation Study

Because the proposed approaches are motivated by real survey problems, we find the best place to apply the proposed algorithms and evaluate their performance is the NHANES survey, where the problems originally occurred. However, this involves some sensitive information and we have to be very careful with our evaluation procedure. Because of our collaborations with Westat Inc., we are able to obtain the NHANES raw data under strict confidentiality agreements. One major difficulty is that we are not allowed to take the data away from Westat's computers, not even to a laptop in the same room for just a temporary simulation run. On the other hand, because of the software licensing issue, Westat methodologists are unwilling to use any software from out of the company, including the free matching package WMATCH we recommended in our proposed Algorithm 1. Restricted by these practical concerns, we design our algorithm performance evaluation procedure in two stages. In this section, we apply the proposed algorithms to the NHANES 1999-2000 Sample Person Demographics File, currently released for public use, to compare the speed, similarity of swapped units and flexibility for both algorithms, which we consider as factors that help protect confidentiality. In the next section, we compare the results using the matching software AutoMatch, available at Westat, originally used for finding matching records, in Algorithm 1 so that we can apply both algorithms to the raw data and compare their performance in variance estimation.

In this section, we choose three variables for our simulation. The variables are: age in years (RIDAGEYR), race/ethnicity (RIDRETH1) and gender (RIAGENDR). RIAGENDR is coded as 1 or 2. RIDRETH1 is coded as 1, 2, 3 or 4. The value of RIDAGEYR ranges from 0 to 85. There are 9,965 individual records in this data set, each of which is associated with a psu identifier numbered from 1 to 52. Our goal is to apply the proposed approaches for swapping a certain percentage of individual records, instead of ssu's, between different psu's without noticeably changing values or categories of the variables.

5.7.1 Distance Measure

Using the selected variables, denoted as a, r and g, respectively, we define the distance between any pair of records, (i, j), as:

$$D_{ij} = \begin{cases} 1.0 & \text{if } i \text{ and } j \text{ are from the same psu;} \\ \frac{5I(r_i,r_j)+5I(g_i,g_j)+|a_i-a_j|/5}{30} & \text{otherwise,} \end{cases}$$

where $I(a, b) = 1$ if $a = b$ and 0 otherwise. The distance will range from 0 to 1. The smaller the distance is, the more similar the two corresponding records are to each other. Therefore, with the way we define the distance, we like to obtain a *matching* or swap records associated with small distances in our simulation study. Because we only use 3 variables in our simulation, two of which are categorical with the other one discrete, perfect matches are often possible, as shown in Table 5.7.

5.7.2 Results for the Match-and-Swap Approach

Our first step is to use WMATCH to obtain a desired *matching*. However, in our first pass, we were only able to obtain a partial *matching* in which the percentage of matched pairs of records, γ, ranged from 16% to 65%, depending on the number of nearest neighbors K chosen. Consequently, the condition of α% switches for each psu could not be met for some large values of α. Table 5.5 summarizes approximate com-

Table 5.5: Elapsed CPU time (in seconds) and achieved average distance

k	γ	α				
		0.1	0.2	0.3	0.4	0.5
5	0.16	600 (0.0)	NA	NA	NA	NA
10	0.26	1200 (0.0)	1200 (0.0)	NA	NA	NA
20	0.43	1800 (0.0)	1800 (0.0)	NA	NA	NA
40	0.65	3600 (0.0)	3600 (0.0)	3600 (0.0)	3600 (0.0)	NA

putation times and achieved average distances in which NA's represent unattainable

switches in the above sense. As discussed previously, we modified our algorithm to obtain a complete matching. Table 5.6 shows the improved results.

Table 5.6: Elapsed CPU time (in seconds) and achieved average distance

K	γ	α				
		0.1	0.2	0.3	0.4	0.5
5	1.00	600 (0.0)	1200 (0.0)	1800 (0.0)	2400 (0.0)	3000 (0.0)
10	1.00	1200 (0.0)	1200 (0.0)	1800 (0.0)	2400 (0.0)	3000 (0.0)
20	1.00	1800 (0.0)	1800 (0.0)	1800 (0.0)	2400 (0.0)	3000 (0.0)
40	1.00	3600 (0.0)	3600 (0.0)	3600 (0.0)	3600 (0.0)	4800 (0.0)

5.7.3 Results for the Sequential Swapping Approach

For the sequential swapping approach, we have better control during the swapping process in terms of the source of swapped ssu's in any specific psu. In addition to $\alpha\%$, the required proportion of ssu's to be swapped in any psu, we define another quantity, $\beta\%$, as the upper limit of ssu's swapped from any other psu to the target psu. By introducing β, we tend to monitor the component for each formed pseudo-psu such that the swapped ssu's in it are from a variety of original psu's. This will be beneficial for confidentiality concerns. The result for each combination of α and β levels is attainable in our simulation. Table 5.7 provides Algorithm 2's performance under different conditions:

5.7.4 Summarizing Results

The above results show that the minimum distance has been attained in all cases, meaning that both approaches perform well in reducing the bias of the variance estimator for the demographic variables chosen as swapping criterion. Furthermore, we notice that even though we introduce another control factor β in Algorithm 2, it is

Table 5.7: Elapsed CPU time (in seconds) and achieved average distance

$\beta\backslash\alpha$	0.1	0.2	0.3	0.4	0.5
0.1	489.55 (0.0)	899.20 (0.0)	1268.50 (0.0)	1588.19 (0.0)	1866.99 (0.0)
0.2	488.43 (0.0)	902.61 (0.0)	1271.81 (0.0)	1594.62 (0.0)	1866.50 (0.0)
0.3	491.95 (0.0)	902.82 (0.0)	1271.36 (0.0)	1592.89 (0.0)	1868.17 (0.0)
0.4	492.91 (0.0)	901.70 (0.0)	1271.81 (0.0)	1595.43 (0.0)	1868.11 (0.0)

still faster than Algorithm 1. From the simulation result, we find no obvious patterns between any two pseudo-psu's, which is favorable for confidentiality protection. It also shows that the second approach is fairly flexible for meeting most practical requirements.

5.8 Application to NHANES and Evaluation

As described previously, we are able to apply both proposed algorithms to the 1999-2003 NHANES data at segment (ssu) level with some modification to Algorithm 1 due to practical concerns. Once the segments are swapped, the SUDAAN program for calculating variance by Taylor series is used and the results are evaluated. Westat methodologists used an altered Algorithm 1 in the 2001-2003 NHANES release. Because the software Westat methodologists used to obtain the matching of segment pairs is based on the record linkage technique, we summarize this matching strategy before applying our swapping procedures to the NHANES data.

5.8.1 Matching Adapted from Record Linkage Technique

A modification to the match-and-swap approach at the matching stage is to apply probability-based record linkage techniques to identify optimal swapping partners. The theory for record linkage given by Fellegi and Sunter (1969), and Winkler (1995) discusses the implementation and parameter estimation. In the basic setup, the match

weight assigned to a record pair is derived from a likelihood ratio that accounts for the closeness of the matching fields being compared. We use r for a record pair, v for a field (or variable) compared where there are $v = 1, ..., V$ fields. The weight of a record pair w_r is:

$$w_r = \log_2 \left[\frac{\prod_{v=1}^{V} m_v^{z_{rv}} (1 - m_v)^{1-z_{rv}}}{\prod_{v=1}^{V} u_v^{z_{rv}} (1 - u_v)^{1-z_{rv}}} \right] \tag{5.7}$$

where $m_v = P(\text{field } v \text{ agrees in pair } r | r \in M)$, M is the set of matches, $u_v = P(\text{field } v \text{ agrees in pair } r | r \in U)$, U is the set of non-matches, and $z_{rv} = 1$ if field v agrees and 0 otherwise.

The match weight w_r can be interpreted as a type of log-odds or log-likelihood ratio. By taking the anti-log of w_r, we have

$$2^{w_r} = \frac{L(z_r | r \in M)}{L(z_r | r \in U)} = LR(z_r),$$

where z_r is the vector of 0's and 1's for disagreements and agreements of the component fields in pair r. $L(z_r | r \in M) = \prod_{v=1}^{V} m_v^{z_{rv}} (1 - m_v)^{1-z_{rv}}$ is the likelihood of a particular configuration of agreement and disagreement outcomes among the fields given that the pair is a true match, and $L(z_r | r \in U) = \prod_{v=1}^{V} u_v^{z_{rv}} (1 - u_v)^{1-z_{rv}}$ is the likelihood of the same configuration given that the pair is a true non-match. The transformed weight, a likelihood ratio $LR(z_r)$, is a measure of the strength of evidence that a pair is a match. In general, a likelihood ratio greater than 1 is evidence that the pair is more likely to be a correct match than a non-match, while a likelihood ratio less than 1 indicates the opposite.

Due to in house access to the software, Westat methodologists used the software AutoMatch (MatchWare Technologies, Inc., 1996) for implementation (see Winglee et al., 2000; and Gomatam et al., 2002; for applications with this package). This software requires the user to estimate the conditional matching probabilities m_v and u_v for each matching field and calculates the log-odds weights for all possible record pairs. It then determines the optimal set of pairs by taking the set with the greatest sum of weights. An iterative procedure can also be used to refine the values of the conditional matching probabilities.

In summary, the modified algorithm includes two steps:

Algorithm 1a

Step 1. Apply record linkage techniques to conduct complete matching of the segments. Matching uses constraints to prohibit the pairing of segments from the same psu, and apply a weight threshold to avoid poor matches (segments with no good matching partners are not swapped).

Step 2. Sample a fixed percentage of the matched segments within each psu for swapping. Sampling controls the maximum number of segments for swapping (i.e., the swapping rate) per psu.

5.8.2 Conditional Matching Probabilities

For all proposed approaches, we use six variables describing various demographic characteristics (for example, a percentage of the segment of a particular race or ethnicity) to determine segment pairs, denoted as x_1, x_2, x_3, x_4, x_5 and x_6. To obtain the necessary parameters for the modified match-and-swap approach, AutoMatch was run several times to refine the m_v and u_v values that should be used for each field (see Table 5.8). The first four fields ranged from 0 to 100, and were matched numerically. An extra parameter, d, was included that allowed the weight to be prorated if it differed by a certain amount. For example, if the values for x_1 differed between two segments by 1 percent, the weight for that pair would be slightly less than the full agreement weight, rather than the full disagreement weight. Only if the difference was over 10 percent would the pair be given the full disagreement weight. The last two fields had much smaller ranges, and were matched by comparing the percentage difference between the segment pairs. Again, an extra parameter was used to allow for small disagreements between the pairs. For example, if one segment had a value of 2.0 for x_5 and another had 2.1, that field would have a 5 percent difference, and would get weight between the agreement and disagreement weights, instead of the full disagreement weight.

Table 5.8: Conditional Matching Probabilities

Field	m_v	u_v	d	Agree Weight	Disagree Weight
x_1	0.90	0.015	10	5.86	-3.45
x_2	0.70	0.010	10	6.09	-1.68
x_3	0.80	0.015	5	5.58	-2.32
x_4	0.95	0.015	20	6.06	-4.47
x_5	0.40	0.045	10	3.13	-0.67
x_6	0.90	0.010	20	6.49	-3.29

5.8.3 Distance Measure

For the use of Algorithm 2, a distance measure is defined using these demographic variables. Note that all variables have been standardized so that they range from 0.0 to 1.0. We also penalize the distance between segments from the same stand (psu) in an effort to avoid such pairing. The distance between any two segments i and j is then defined as:

$$D(i,j) = \begin{cases} 10.0 & \text{if } i \text{ and } j \text{ are in the same psu,} \\ \sum_{k=1}^{6}(x_{ik} - x_{jk})^2 & \text{if } i \text{ and } j \text{ are not in the same psu.} \end{cases}$$

Thus, the distance will range from 0.0 to 6.0 for segments from different stands. The smaller the distance between two segments is, the more alike they are. Again, we would like to swap segments associated with small distances in our simulation study.

5.8.4 Swapping Procedures

Another practical restriction is that segments should be swapped between certain pairs of psu's. More specifically, only segments from psu's with high disclosure risk need to be swapped to the rest of psu's with low disclosure risk, essentially forming the following swapping procedures:

Procedure 1: use Automatch to match segments ignoring the psu structure; randomly select a segments from each of b psu's which are considered as having the highest disclosure risk; swap selected segments with their best available pairings which are not in any of the b high disclosure risk psu's.

Procedure 2: sequentially select the best available pair of segments between high risk psu's and low risk psu's and swap them until a segments from each of the high risk psu's have been swapped.

5.8.5 Evaluation

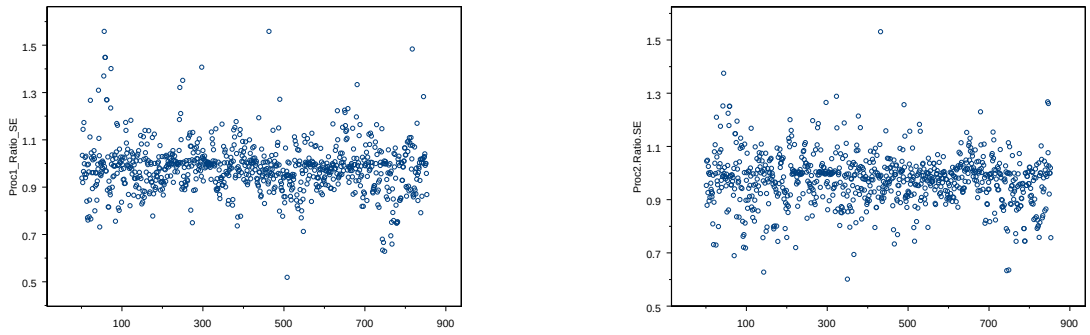
After applying two swapping procedures to the segment level NHANES data, SUDAAN is run to obtain the point estimates, standard errors and design effects of all variables for each swapping procedure. Several descriptive statisics are calculated for each procedure's standard error and design effect relative to those of baseline design. Ideally, these values would be close to 1, meaning that the standard error and design effect will not be greatly affected by the swapping procedures. Table 5.9 shows the descriptive statistics by procedure for the standard error ratio and the design effect ratio, respectively.

Table 5.9: Ratio of SEs and DEFFs by Method to Baseline Design

Swapping	Descriptive Statistics						
Procedure	Mean	Std.Dev	Kurtosis	Skewness	Min	Median	Max
			Ratio of Standard Errors				
1	0.983	0.106	4.747	0.801	0.519	0.986	1.559
2	0.975	0.096	2.777	0.076	0.602	0.985	1.531
			Ratio of Design Effects				
1	0.974	0.219	8.641	1.778	0.269	0.971	2.430
2	0.959	0.188	4.987	0.812	0.362	0.969	2.345

To help visualize the potential pattern caused by either swapping procedure, we plot the standard errors and design effects of all variables obtained by either swapping procedure against those from baseline design in Figure 5.1 and 5.2, respectively.

Figure 5.1: Distribution of Ratios of Standard Errors by Procedure



In summary, both procedures perform well, with procedure 2 generating lower variation and skewness among all variables. Also note that procedure 2 runs much faster (within seconds for ssu level swapping) than procedure 1 and has a great deal of flexibility to accommodate the necessary adjustments which may occur after an initial swapping. For example, we could easily include some variables that did not perform well in our distance measure and observe the impact immediately. However, for procedure 1 it will not be as easy because we have to re-evaluate the conditional matching probabilities.

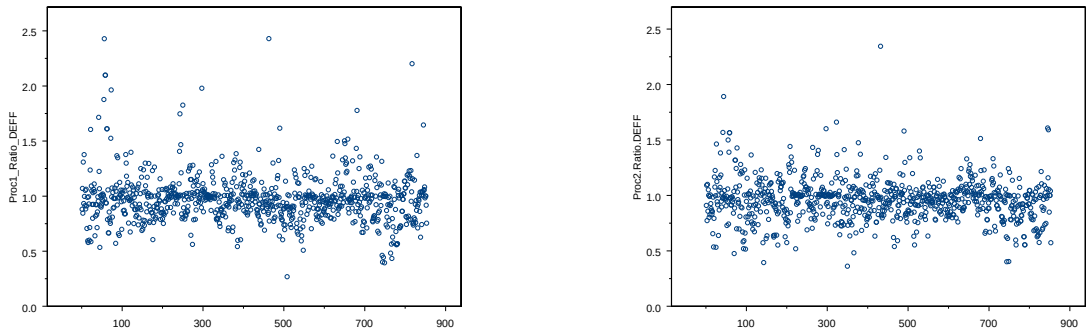
5.9 Proof of (5.3) and (5.5)

5.9.1 Proof of (5.3)

The r-th replicate estimate is given by

$$\hat{Y}_{(r)} = \sum_{h=1}^{L} \left\{ \frac{1}{2}\Big[1 + \delta_{rh}(1-\varepsilon)\Big] r_{h1} + \frac{1}{2}\Big[1 - \delta_{rh}(1-\varepsilon)\Big] r_{h2} \right\}, \tag{5.8}$$

Figure 5.2: Distribution of Ratios of Design Effects by Procedure



where $\delta_{rh} = +1$ or -1 depending on whether the first or the second psu of the h-th stratum is in the r-th half sample such that $\sum_{r=1}^{R} \delta_{rh} = 0$ for all h and

$$\sum_{r=1}^{R} \delta_{rh}\delta_{rh'} = 0 \quad \text{for any } h \neq h'. \tag{5.9}$$

Since $\hat{Y}_{(r)} - \hat{Y} = (1-\varepsilon)\sum_{h=1}^{L} \delta_{rh}(r_{h1} - r_{h2})/2$ and $\delta_{rh}^2 \equiv 1$ for any h, r, we have

$$\begin{aligned}
v_{BRR-F}(\hat{Y}) &= \frac{1}{R(1-\epsilon)^2}\sum_{r=1}^{R}(\hat{Y}_{(r)} - \hat{Y})^2 \\
&= \frac{1}{4R}\sum_{r=1}^{R}\left[\sum_{h=1}^{L}\delta_{rh}(r_{h1} - r_{h2})\right]^2 \\
&= \frac{1}{4R}\sum_{r=1}^{R}\left[\sum_{h=1}^{L}(r_{h1} - r_{h2})^2 + \sum_{h\neq h'}^{L}\delta_{rh}\delta_{rh'}(r_{h1} - r_{h2})(r_{h'1} - r_{h'2})\right] \\
&= \frac{1}{4}\sum_{h=1}^{L}(r_{h1} - r_{h2})^2 + \frac{1}{4R}\sum_{h\neq h'}^{L}(r_{h1} - r_{h2})(r_{h'1} - r_{h'2})\sum_{r=1}^{R}\delta_{rh}\delta_{rh'}.
\end{aligned}$$

The result (5.3) follows from (5.9).

5.9.2 Proof of (5.5)

By definition, we have $r_{hi} = r_{1,hi} + r_{2,hi}$ and $r^*_{hi} = r^*_{1,hi} + r^*_{2,hi}$ for any h and i. Thus, observing $r^*_{h1} - r^*_{h2} = r_{h1} - r_{h2} + 2(r_{1,h2} - r_{2,h1})$ and $r_{h1} - r_{h2} + r_{1,h2} - r_{2,h1} = r_{1,h1} - r_{2,h2}$, we have

$$\begin{aligned} v^*(\hat{Y}) &= \sum_{h=1}^{L} \left(\frac{r^*_{h1} - r^*_{h2}}{2} \right)^2 \\ &= \sum_{h=1}^{L} \left[\left(\frac{r_{h1} - r_{h2}}{2} \right) + (r_{1,h2} - r_{2,h1}) \right]^2 \\ &= \sum_{h=1}^{L} \left(\frac{r_{h1} - r_{h2}}{2} \right)^2 + \sum_{h=1}^{L} (r_{1,h2} - r_{2,h1})(r_{h1} - r_{h2} + r_{1,h2} - r_{2,h1}) \\ &= v(\hat{Y}) - \sum_{h} (r_{1,h1} - r_{2,h2})(r_{2,h1} - r_{1,h2}), \end{aligned}$$

which completes the proof.

Chapter 6

Future Research and Concluding Remarks

6.1 Consistency of Replication Based Variance Estimators

In this thesis, we proved the consistency of the JK2 combined strata grouped variance estimator when $\hat{\theta}$ is a smooth function of means and there are 2 psu's per stratum, though a similar proof is available for JKn and the BRR. Rao and Shao (1996, Theorem 3) consider a different problem, where there are only a few strata with many psu's in each stratum. In their situation, the psu's within a stratum are randomly paired into pseudo strata and then a large BRR applied. There is no grouping of strata. This is much like the way that SR strata were handled in Chapter 4. They establish the consistency for smooth functions of means and show that the method performs well for quantiles in a simulation study, as the minimum n_h gets large. The general problem is exemplified by the NHIS survey of Chapter 4 where there are many strata with small number of psu's and a few strata with many psu's (SR), and one randomly pairs the psu's in the large (SR) strata to form a much larger set of pseudo strata and then applies the combined grouped BRR to this set consisting of the original strata with small number of psu's and the pseudo strata created from the

large psu's. Though we ignored this aspect in our evaluations, as the original data was unavailable, we expect that the method will perform well in this case, given the results in our simulations, in Rao and Shao's simulations and the Rao and Shao proof of consistency in their context. In fact, under reasonably straight-forward conditions, it is quite clear that the method will remain consistent for smooth function of means. For example, assume that there are a small number of large strata (SR) and a large number of strata with 2 psu's. The Rao and Shao (1996) proof of consistency applies to the group of large strata while the Krewski and Rao (1981) proof applies to the large number of 2 psu strata, which will imply consistency, without grouping of strata. Our Theorem 1 will then imply that grouping of strata will go to the same limit and thus the desired consistency results. For SR strata this is a reasonable asymptotic framework as the SR strata is really a psu sampled with certainty and its "psu's" are really the ssu's within the certainty psu, the number of which are often large, i.e. $n_h \to \infty$. We hope to develop a general set of conditions under which this variance estimator will be consistent, using BRR, JK2 and JKn.

In addition, it is likely that the theoretical results on the consistency of the BRR for non-smooth estimators such as quantiles should be extendable to the above general method of combined group BRR after random pairing of psu's in a few large strata. Rao and Shao's simulations on quantiles for their situation showed good performance. We hope to develop consistency results for the BRR in this situation and relate the performance back to the method of grouping into combined strata.

Also, it may be possible to develop a methodology that will work for non-smooth estimators using JK2 and JKn. It is well-known that the delete-1 jackknife is not consistent for non-smooth estimators. However, in the combined strata approach more than one psu is deleted in each replicate. This means there is potential to use the delete-d jackknife theory of Shao and Wu (1989) to establish consistency. We would like to investigate this, as well as, the connection to how one should do the grouping and can the same grouping simultaneously establish consistency for smooth and non-smooth estimators.

6.2 Replicate Weight Perturbation

As described in Chapter 5, consider introducing a random noise ε to replicate weights and express the perturbed replicate weights as

$$w^*_{hil(r)} = w_{hil(r)}(1+\varepsilon_{hil}) = b_{hil(r)}w_{hil}(1+\varepsilon_{hil}).$$

Denote the replicate estimate as $\bar{y}^*_{(r)} = \sum_{(hil)\in s} w^*_{hil(r)}y_{hil}/M$. The variance estimator of $\bar{y}$ after perturbation is then given as

$$\begin{aligned} v^* &= c_R\sum_{r=1}^{R}(\bar{y}^*_{(r)}-\bar{y})^2 \\ &= c_R\sum_{r=1}^{R}\left[(\bar{y}^*_{(r)}-\bar{y}_{(r)})+(\bar{y}_{(r)}-\bar{y})\right]^2 \\ &= v + c_R\sum_{r=1}^{R}(\bar{y}^*_{(r)}-\bar{y}_{(r)})^2 + 2c_R\sum_{r=1}^{R}(\bar{y}^*_{(r)}-\bar{y}_{(r)})(\bar{y}_{(r)}-\bar{y}), \end{aligned} \tag{6.1}$$

where $v = c_R\sum_{r=1}^{R}(\bar{y}_{(r)}-\bar{y})^2$ is the usual replicate variance estimator and c_R varies depending on which replication method is employed. Denote E_ϵ as the expectation with respect to iid random variables ε_{hil}. We have

$$E_\epsilon(v^*) = v + c_R\sum_{r=1}^{R}E_\epsilon(\bar{y}^*_{(r)}-\bar{y}_{(r)})^2 \tag{6.2}$$

as $E_\epsilon(\bar{y}^*_{(r)}-\bar{y}_{(r)}) = \sum_{(hil)\in s}E_\epsilon(\varepsilon_{hil})w_{hil(r)}y_{hil}/M = 0$. Furthermore, the last term is

$$\begin{aligned} c_R\sum_{r=1}^{R}E_\epsilon(\bar{y}^*_{(r)}-\bar{y}_{(r)})^2 &= c_R\sum_{r=1}^{R}E_\epsilon\left[\sum_{(hil)\in s}(w^*_{hil(r)}-w_{hil(r)})y_{hil}/M\right]^2 \\ &= \frac{c_R}{M^2}\sum_{r=1}^{R}E_\epsilon\left[\sum_{(hil)\in s}b_{hil(r)}w_{hil}\varepsilon_{hil}y_{hil}\right]^2 \\ &= \frac{\sigma^2}{M^2}\sum_{(hil)\in s}w^2_{hil}y^2_{hil}\left[c_R\sum_{r=1}^{R}b^2_{hil(r)}\right]. \end{aligned} \tag{6.3}$$

Since $c_R\sum_{r=1}^{R}b^2_{hil(r)}$ is a term of order $O(1)$ and $\sum_{(hil)\in s}w^2_{hil}y^2_{hil}/M^2$ should have the same order as the replication variance estimator v under some conditions, the

left hand side of (6.3) will not disappear unless σ^2 goes to zero, hence generating a noneligible bias in variance estimation. On the other hand, as we showed in our small simulation in section 5.4.4, the clustering algorithm will reconstruct psu identifiers with high accuracy for any moderately small σ^2. We feel this issue is interesting and is worth a more indepth investigation. In the future, we would like to explore the connection between the variance of added noise and the resulting bias of the variance estimator, both through simulation and theoretical development. We would also like to try different distributions on ε in our simulation. The outcome for this research direction will be either discovering a new method for masking psu identifiers from replicate weights or demonstrating that there is no feasible way to do so. We expect the latter.

6.3 General Approaches to Replicate Weight Construction

This thesis considers some specific solutions to a problem in replication based variance estimation which can be more generally characterized as follows.

We wish to construct a set of replicate weights as depicted in Table 2.1 where R is not too large and the resulting variance estimator is consistent. Chapters 3 and 4 use the approach of grouping strata and applying existing replication methods simultaneously to all strata in a group. For example, the BRR when so applied can be viewed as assigning every stratum in a group to the same column of a Hadamard matrix. This has advantages and disadvantages depending upon the context. For example, if confidentiality is an issue, as it often is, then grouping strata will help mask suppressed psu and strata identifiers. On the other hand, the variance of the resulting variance estimator will increase. Chapters 3 and 4 discuss method to limit this impact. If, however, confidentiality is not of concern, this can be viewed as a method of reducing end-user effort. If so, in general, we could consider how to construct two-way arrays with small R satisfying a set of conditions which ensure desirable statistical properties of the resulting variance estimator. They should be able to handle general

n_h and ensure consistency, and efficiency of resulting variance estimators of smooth and non-smooth functions of means. Meanwhile, the construction procedure should be easy and fast to implement.

There already exists such a set of conditions for BRR-type methods which in part satisfy these requirements, as described in Sitter's (1993) definition of a *balanced orthogonal multi-array* (BOMA). However, they are difficult to construct with small R. When you have one, you can in principle classify multiple strata to the same column, to get a combined strata BOMA. Because of the difficulty in constructing such mixed-level BOMA's, a better approach might be to sacrifice the orthogonality or balance to get an approximate BOMA that still ensures consistency and may be more efficient in terms of variance of the variance estimator than combining strata, which essentially can be viewed as allowing columns in the array to be completely confounded.

Consider a simple example of a matrix (see Table 6.1) that could be used for this purpose.

Table 6.1: Construction of a 14×8 Array

	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
1	−	−	−	+	+	+	−	+
2	+	−	−	−	−	+	+	+
3	−	+	−	−	+	−	+	+
4	+	+	−	+	−	−	−	+
5	−	−	+	+	−	−	+	+
6	+	−	+	−	+	−	−	+
7	−	+	+	−	−	+	−	+
8	+	+	+	−	−	−	+	−
9	−	+	+	+	+	−	−	−
10	+	−	+	+	−	+	−	−
11	−	−	+	−	+	+	+	−
12	+	+	−	−	+	+	−	−
13	−	+	−	+	−	+	+	−
14	+	−	−	+	+	−	+	−

This matrix was obtained by taking the 8 run factorial, folding it over and then

removing the row of +'s and the row of −'s. Note that the 14 sets represented by keeping the row numbers corresponding to +'s in each row form a resolvable BIBD. Now, if we let each row define a resample by keeping two copies of each y_j which has a + below it, then each row defines a resample of size n. For example, row 2 defines the resample $\{y_1, y_1, y_6, y_6, y_7, y_7, y_8, y_8\}$. Thus, this matrix defines 14 resamples that are balanced in such a way that if the estimator were $\hat{\theta} = \bar{y}$, then $(1/14)\sum_{r=1}^{14}(\hat{\theta}_r^* - \hat{\theta})^2 = s^2/n$, where $\hat{\theta}_r^*$ is calculated from the r-th replicate.

First, consider the iid case, where we wish to avoid any rescaling. Let $X = \{x_{ij}\}$ be an $R \times n$ matrix consisting of 1's and 0's which replace all the +'s and −'s in Table 6.1, respectively. Let $\alpha_i = \sum_{j=1}^{n} x_{ij}$, the number of 1's in row i. Then the matrix must satisfy three conditions:

1) $\sum_{i=1}^{R} x_{ij}/\alpha_i = R/n$ for each $j = 1, ..., n$;

2) $\sum_{i=1}^{R} x_{ij}/\alpha_i^2 = 2R/n^2$ for each $j = 1, ..., n$;

3) $\sum_{i=1}^{R} x_{ij}x_{ik}/\alpha_i^2 = R(n-2)/[n^2(n-1)]$ for each $j \neq k = 1, ..., n$.

In fact, in the example of Table 6.1, conditions 1 and 2 are equivalent as a balanced array condition. Condition 3 is a BIBD- type condition. This set of conditions mimic the BOMA definition in Sitter (1993).

The fold-over approach can be used to construct arrays satisfying BOMA conditions with $R = 2(n-1)$ for n, a multiple of 4, by using folded-over Hadamard matrices. However, because the number of columns, which represents the number of units, has to be a multiple of 4 for such an array, we still need to answer the following question: is it possible to apply an array obtained by using the fold-over approach to the cases where the number of sampled units is not constrained to a multiple of 4? This actually leads to a more general question: if there is no easy way to construct an array perfectly satisfying this set of conditions, can we satisfy them approximately? By approximately, we mean to establish a criterion to evaluate how closely these conditions can be satisfied for an array. On the other hand, we would also like to know how good an approximation we need, to retain the consistency of the resulting variance estimator. We try to answer both questions by considering an objective function

such as

$$g(X) = \sum_j \left\{ (A_j - 1)^2 + (B_j - 1)^2 + \sum_{k \neq j} [(n-1)C_{jk} + 1]^2 \right\},$$

where $A_j = \sum_{i=1}^R w_{ij}/R$, $B_j = \sum_{i=1}^R (w_{ij} - 1)^2/R$, $C_{jk} = \sum_{i=1}^R (w_{ij} - 1)(w_{ik} - 1)/R$, and $w_{ij} = n\delta_{ij}/\alpha_i$. If the three conditions are satisfied, $g(x) = 0$. By defining such a function and having some construction ideas, we have made some modest progress on this more general problem, though with the restriction to the iid case. This could be classified as a balanced bootstrap (see Davison, Hinkley, and Schechtman, 1986; Efron, 1990; Graham et al., 1990; Nigam and Rao, 1996). The first step is to introduce the following lemma.

Lemma 6.1 *For $n + t = n' = 4m$, $0 \leq t \leq 3$, $R = 2n' - 2$, the two-way array $X(R, n)$, built by folding an $n' \times n'$ Hadamard matrix, and then removing t columns, satisfies the BOMA conditions exactly for $t = 0$ or 1 and the following nearly BOMA conditions for $t = 2$ or 3.*

1) $\sum_i^R \frac{x_{ij}}{\alpha_i} = \frac{R}{n} + O(\frac{1}{n^2})$, $j = 1, \ldots, n$;

2) $\sum_i^R \frac{x_{ij}}{\alpha_i^2} = \frac{R\tilde{\alpha}}{n} + O(\frac{1}{n^3})$, $j = 1, \ldots, n$;

3) $\sum_i^R \frac{x_{ij}x_{ik}}{\alpha_i^2} = \frac{R(1-\tilde{\alpha})}{n(n-1)} + O(\frac{1}{n^3})$, $j \neq k = 1, \ldots, n$.

The proof of Lemma 6.1 is given in section 6.4. Naturally, the next step would be to investigate whether or not, with the BOMA conditions approximately satisfied, the resulting variance estimator retains its consistency and has satisfactory performance.

We realize that the extension to stratified multi-stage sampling with general n_h is still quite challenging, including both algorithmic approaches to find arrays and theoretical consideration. If all strata are large and of the same size, then one could use an array constructed as in Lemma 6.1 and its complement constructing a multi-array via Kronecker products as shown in the example below reproduced from Sitter (1993, Example 5.1).

Example 5.1: Let $L = 7$ and $p = 4$. Then

$$B = \begin{pmatrix} + & + & + & + & + & + & + \\ - & + & - & + & - & + & - \\ - & - & + & + & - & - & + \\ + & - & - & + & + & - & - \\ + & + & + & - & - & - & - \\ - & + & - & - & + & - & + \\ - & - & + & - & + & + & - \\ + & - & - & - & - & + & + \end{pmatrix} \quad \text{and} \quad C = \begin{pmatrix} + & - & + & - \\ + & - & - & + \\ + & + & - & - \end{pmatrix}. \tag{6.4}$$

$p = 4$ units per stratum in $R = 24$ replicates. These were obtained using the Hadamard matrices given in Wolter (1985) p. 322. So $A = B \otimes C$ is obtained by replacing the +'s and −'s in B with $+C$ and $-C$. Note that C has $p = 4$ columns and that C and $-C$ together could be obtained via the fold-over method applied to the 4 run factorial design matrix. If we label the columns 1 through 4, then each row

Table 6.2: A BOMA$(24, 4^7; 2^7)$

			h			
1	2	3	4	5	6	7
(1,3)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)
(1,4)	(1,4)	(1,4)	(1,4)	(1,4)	(1,4)	(1,4)
(1,2)	(1,2)	(1,2)	(1,2)	(1,2)	(1,2)	(1,2)
(2,4)	(1,3)	(2,4)	(1,3)	(2,4)	(1,3)	(2,4)
(2,3)	(1,4)	(2,3)	(1,4)	(2,3)	(1,4)	(2,3)
(3,4)	(1,2)	(3,4)	(1,2)	(3,4)	(1,2)	(3,4)
(2,4)	(2,4)	(1,3)	(1,3)	(2,4)	(2,4)	(1,3)
(2,3)	(2,3)	(1,4)	(1,4)	(2,3)	(2,3)	(1,4)
(3,4)	(3,4)	(1,2)	(1,2)	(3,4)	(3,4)	(1,2)
(1,3)	(2,4)	(2,4)	(1,3)	(1,3)	(2,4)	(2,4)
(1,4)	(2,3)	(2,3)	(1,4)	(1,4)	(2,3)	(2,3)
(1,2)	(3,4)	(3,4)	(1,2)	(1,2)	(3,4)	(3,4)
(1,3)	(1,3)	(1,3)	(2,4)	(2,4)	(2,4)	(2,4)
(1,4)	(1,4)	(1,4)	(2,3)	(2,3)	(2,3)	(2,3)
(1,2)	(1,2)	(1,2)	(3,4)	(3,4)	(3,4)	(3,4)
(2,4)	(1,3)	(2,4)	(2,4)	(1,3)	(2,4)	(1,3)
(2,3)	(1,4)	(2,3)	(2,3)	(1,4)	(2,3)	(1,4)
(3,4)	(1,2)	(3,4)	(3,4)	(1,2)	(3,4)	(1,2)
(2,4)	(2,4)	(1,3)	(2,4)	(1,3)	(1,3)	(2,4)
(2,3)	(2,3)	(1,4)	(2,3)	(1,4)	(1,4)	(2,3)
(3,4)	(3,4)	(1,2)	(3,4)	(1,2)	(1,2)	(3,4)
(1,3)	(2,4)	(2,4)	(2,4)	(2,4)	(1,3)	(1,3)
(1,4)	(2,3)	(2,3)	(2,3)	(2,3)	(1,4)	(1,4)
(1,2)	(3,4)	(3,4)	(3,4)	(3,4)	(1,2)	(1,2)

of C can be used to define a 2-subset of the 4 units by keeping the units with a + sign

in their column. So the 3 rows of C become the 3 subsets $(1,3)$, $(1,4)$, and $(1,2)$, and similarly the 3 rows of $-C$ become the 3 subsets $(2,4)$, $(2,3)$, and $(3,4)$. Doing this throughout A we obtain the BOMA$(24, 4^7; 2^7)$ given in Table 6.2, reproduced from Sitter (1993, Table 1), which gives a balanced half-sample technique for $L = 7$ strata with

6.4 Proof of Lemma 6.1

It is easy to examine that all conditions hold exactly for the case of $t = 0, 1$. So do the conditions 1 and 2 for the case of $t = 2$. Here we will only show that all conditions hold when $t = 3$, which is the case where we delete three columns from an original array with $4m$ columns, obtained by using the fold-over approach, to form $X(R, n)$.

Note that $\alpha_i = \sum_j x_{ij}$, $i = 1, \cdots, R$, only take 4 different values, ranging from $2m-3$ to $2m$. Without loss of generality, we assume that the 3 columns to be deleted have the patterns as shown in Table 6.3 and then we introduce the following notation:

$$\alpha'_1 = \alpha_i, \quad i = 1, \cdots, m-1, \quad \cdots, \quad \alpha'_8 = \alpha_i, \quad i = 7m-1, \cdots, 8m-2,$$

$$\begin{aligned} S_{1j} &= \sum_{i=1}^{m-1} x_{ij}, \quad \cdots, \quad S_{8j} = \sum_{i=7m-1}^{8m-2} x_{ij} \quad j = 1, \cdots, n \quad \text{and} \\ T_{1jk} &= \sum_{i=1}^{m-1} x_{ij}x_{ik}, \quad \cdots, \quad T_{8jk} = \sum_{i=7m-1}^{8m-2} x_{ij}x_{ik} \quad j \neq k = 1, \cdots, n. \end{aligned}$$

We see that the $S_{i'j}$'s and $T_{i'jk}$'s satisfies, from the orthogonality of Hadamard matrices, the equations,

$$\begin{aligned} &S_{1j} + S_{2j} = m-1, \quad S_{3j} + S_{4j} = m, \quad S_{5j} + S_{6j} = m, \quad S_{7j} + S_{8j} = m, \\ &S_{1j} + S_{3j} = m-1, \quad S_{2j} + S_{4j} = m, \quad S_{5j} + S_{7j} = m, \quad S_{6j} + S_{8j} = m, \end{aligned}$$

and

$$T_{1jk} + T_{2jk} + T_{2jk} + T_{4jk} = m-1, \quad T_{5jk} + T_{6jk} + T_{7jk} + T_{8jk} = m.$$

Table 6.3: Patterns of Deleted Columns ($n' = 4m$, $R = 2n' - 2 = 8m - 2$)

Row	C1	C2	C3	α_i	Row	C1	C2	C3	α_i
1	+	+	+	$2m-3$	$4m$	−	−	−	$2m$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
$m-1$	+	+	+	$2m-3$	$5m-2$	−	−	−	$2m$
m	+	+	−	$2m-2$	$5m-1$	−	−	+	$2m-1$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
$2m-1$	+	+	−	$2m-2$	$6m-2$	−	−	+	$2m-1$
$2m$	+	−	+	$2m-2$	$6m-1$	−	+	−	$2m-1$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
$3m-1$	+	−	+	$2m-2$	$7m-2$	−	+	−	$2m-1$
$3m$	+	−	−	$2m-1$	$7m-1$	−	+	+	$2m-2$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
$4m-1$	+	−	−	$2m-1$	$8m-2$	−	+	+	$2m-2$

1) From Table 6.3, we know

$$\alpha'_{i'} = \begin{cases} 2m-3, & i'=1, \\ 2m-2, & i'=2,3,8, \\ 2m-1, & i'=4,6,7, \\ 2m, & i'=5, \end{cases}$$

and some related terms can be expressed as

$$\frac{1}{2m-t} = \frac{1}{2m} + \frac{t}{4m^2} + O(\frac{1}{m^3})$$

and

$$\begin{aligned} \frac{1}{(2m-t)^2} &= \left[\frac{1}{2m} + \frac{t}{4m^2} + O(\frac{1}{m^3})\right]^2 \\ &= \frac{1}{4m^2} + \frac{t}{4m^3} + O(\frac{1}{m^4}). \end{aligned}$$

Therefore,

$$\begin{aligned} \sum_{i=1}^{R} \frac{x_{ij}}{\alpha_i} &= \sum_{i'=1}^{8} \frac{S_{i'j}}{\alpha'_{i'}} \\ &= \frac{\sum_1^8 S_{i'j}}{2m} + \frac{3S_{1j} + 2(S_{2j} + S_{3j} + S_{8j}) + S_{4j} + S_{6j} + S_{7j}}{4m^2} + O(\frac{1}{m^2}) \\ &= \frac{4m-1}{2m} + \frac{\sum_1^4 S_{i'j} + \sum_1^8 S_{i'j} + S_{1j} - S_{4j} - S_{5j} + S_{8j}}{4m^2} + O(\frac{1}{m^2}) \\ &= \frac{4m-1}{2m} + \frac{6m-3}{4m^2} + O(\frac{1}{m^2}) \\ &= 2 + \frac{1}{m} + O(\frac{1}{m^2}). \end{aligned}$$

On the other hand,

$$\begin{aligned} \frac{R}{n} &= \frac{8m-2}{4m-3} = 2 + \frac{4}{4m-3} \\ &= 2 + \frac{1}{m} + O(\frac{1}{m^2}), \end{aligned}$$

or

$$\sum_{i=1}^{R} \frac{x_{ij}}{\alpha_i} = \frac{R}{n} + O(\frac{1}{n^2}). \tag{6.5}$$

2) Since

$$\begin{aligned} R\tilde{\alpha} &= \sum_{i}^{R} \frac{1}{\alpha_i} = \frac{m-1}{2m-3} + \frac{m-1}{2m} + \frac{3m}{2m-1} + \frac{3m}{2m-2} \\ &= \frac{8m-2}{2m} + \frac{3(m-1)+3m+6m}{4m^2} + O(\frac{1}{m^2}) \\ &= 4 + \frac{2}{m} + O(\frac{1}{m^2}), \end{aligned}$$

we have

$$\begin{aligned} \frac{R\tilde{\alpha}}{n} &= \left[\frac{1}{4m} + \frac{3}{16m^2} + O(\frac{1}{m^3})\right]\left[4 + \frac{2}{m} + O(\frac{1}{m^2})\right] \\ &= \frac{1}{m} + \frac{5}{4m^2} + O(\frac{1}{m^3}). \end{aligned}$$

On the other hand,

$$\begin{aligned} \sum_{i=1}^{R} \frac{x_{ij}}{\alpha_i^2} &= \sum_{k=1}^{8} \frac{S_{kj}}{\alpha'^2_k} \\ &= \frac{4m-1}{4m^2} + \frac{6m-3}{4m^3} + O(\frac{1}{m^3}) \\ &= \frac{1}{m} + \frac{5}{4m^2} + O(\frac{1}{m^3}) \\ &= \frac{R\tilde{\alpha}}{n} + O(\frac{1}{n^3}). \end{aligned} \tag{6.6}$$

3) Since

$$\begin{aligned} \frac{R(1-\tilde{\alpha})}{n(n-1)} &= \frac{1}{n-1}\left(\frac{R}{n} - \frac{R\tilde{\alpha}}{n}\right) \\ &= \left[\frac{1}{4m} + \frac{1}{4m^2} + O(\frac{1}{m^3})\right]\left[2 + \frac{1}{m} + O(\frac{1}{m^2}) - \frac{1}{m} - \frac{5}{4m^2}\right] \\ &= \frac{1}{2m} + \frac{1}{2m^2} + O(\frac{1}{m^3}), \end{aligned}$$

we have

$$\begin{aligned}
\sum_{i=1}^{R} \frac{x_{ij}x_{ik}}{\alpha_i^2} &= \sum_{i'=1}^{8} \frac{T_{i'jk}}{\alpha'^2_{i'}} \\
&= \frac{\sum_1^8 T_{i'jk}}{4m^2} + \frac{\sum_1^4 T_{i'jk} + \sum_1^8 T_{i'jk} + T_{1jk} - T_{5jk} - T_{4jk} + T_{8jk}}{4m^3} + O(\frac{1}{m^3}) \\
&= \frac{2m-1}{4m^2} + \frac{(3m-2) + (m-1-2S_{5j}) - (m-2S_{8j})}{4m^3} + O(\frac{1}{m^3}) \\
&= \frac{1}{2m} + \frac{1}{2m^2} + O(\frac{1}{m^3}) \\
&= \frac{R(1-\tilde{\alpha})}{n(n-1)} + O(\frac{1}{n^3}). \qquad (6.7)
\end{aligned}$$

Bibliography

Abowd, J.M., and Woodcock, S.D. (2002). "Appendix to disclosure limitations in longitudinal linked data." In Doyle, P., Lane, J., Theeuwes, J., and Zayatz, L. (Eds), *Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies*. Elsevier.

Bickel, P.J., and Freedman, D.A. (1984). "Asymptotic normality and the bootstrap in stratified sampling." *Annals of Statistics*, 12, 470-482.

Cohen, S.B. (1997). "An evaluation of alternative PC-based software packages developed for the analysis of complex survey data." *The American Statistician*, 51, 285-292.

Davison, A.C., Hinkley, D.V. and Schechtman, E. (1986). "Efficient bootstrap simulation." *Biometrika*, 73, 555-566.

DiGaetano, R., Brick, M.J., and Cervantes, I.F. (1998). "Preserving degrees of freedom in a multi-mode, multi site survey." In the *Proceedings of the American Statistical Association, Survey Research Methods Section*, 475-480.

Dippo, C.S., Fay, R.E. and Morganstein, D.H. (1984). "Computing variances from complex samples with replicate weights." In the *Proceedings of the American Statistical Association, Section on Survey Research Methods*, 489-494.

Dohrmann, S., Curtin, L.R., Mohadjer, L., Montaquila, J., and L,T. (2002). "National health and nutrition examination survey limiting the risk of data disclosure using replication techniques in variance estimation." In the *Proceedings*

of the American Statistical Association, Section on Survey Research Methods, 807-812.

Duncan, G.T., and Mukherjee, S. (1998). "Optimal disclosure limitation strategy in statistical data- bases: Deterring tracker attacks through additive noise." *Heinz School of Public Policy and Management*, Working Paper No. 1998-15.

Edmonds, J. (1965). "Maximum matching and a polyhedron with (0,1) vertices." *Journal of Research of the National Bureau of Standards*, B69, 125-130.

Efron, B. (1990). "More efficient bootstrap simulations." *Journal of the American Statistical Association*, 85, 79-89.

Evans, T., Zayatz, L., and Slanta, J. (1998). "Using noise for disclosure limitation of establishment tabular data." *Journal of Official Statistics*, 14, 537-551.

Felligi, I.P. and Sunter, A.B. (1969). "A theory for record linkage." *Journal of the American Statistical Association*, 64, 1183-1210.

Fienberg, S.E. (1994). "A radical proposal for the provision of micro-data samples and the preservation of condentiality." *Carnegie Mellon University Department of Statistics*, Technical Report No. 611.

Fuller, W.A. (1993). "Masking procedures for microdata disclosure limitation." *Journal of Official Statistics*, 9, 383-406.

Gabow, H.N. (1973), *Implementation of Algorithms for Maximum Matching on Non-bipartite Graphs*. unpublished Ph.D thesis, Stanford University, Department of Computer Science.

Gomatam, S., Carter, R., Ariet, A., and Mitchell, G. (2002). "An empirical companion of record linkage procedures." *Statistics in Medicine*, 21, 1485-1496.

Graham, R.L., Hinkley, D.V., John, P.W.M. and Shi, S. (1990). "Balanced design of bootstrap simulations." *Journal of the Royal Statistical Society Series B*, 52, 185-202.

Greenberg, B. (1987). "Rank swapping for masking ordinal microdata." U.S. Census Bureau, unpublished manuscript.

Gross, S. (1980). "Median estimation in sample surveys." In the *Proceedings of the Section on Survey Research Methods, American Statistical Association*, 181-184.

Gupta, V.K., and Nigam, A.K. (1987). "Mixed orthogonal arrays for variance estimation with unequal numbers of primary selections per stratum." *Biometrika*, 74, 735–742.

Gurney, M., and Jewett, R. (1975). "Constructing orthogonal replications for variance estimation." *Journal of the American Statistical Association*, 70, 819–821.

Hansen, M.H., Hurwitz, W.N. and Madow, W.G. (1953). *Sample Survey Methods and Theory: Vol. I.* New York: Wiley.

Judkins, D.R. (1990). "Fay's method for variance estimation." *Journal of Official Statistics*, 6, 223-239.

Kalton, G. (1977). "Practical methods for estimating survey sampling errors." *Bulletin of the International Statistical Institute*, 47(3), 495-514.

Kennickell (1991)Kennickell, A.B. (1991). "Imputation of the 1989 Survey of Consumer Finances: Stochastic relaxation and multiple imputation." *SCF Working Paper*, prepared for the Annual Meetings of the Amercian Statistical Association, Atlanta, Georgia, August 1991.

Kennickell, A.B. (1997). "Multiple imputation and disclosure protection: The case of the 1995 Survey of Consumer Finances." *SCF Working Paper.*

Kennickell, A.B. (1998). "Multiple imputation in the Survey of Consumer Finances." *SCF Working Paper*, prepared for the August 1998 Joint Statistical Meetings, Dallas, TX.

Kennickell, A.B. (2000). "Wealth measurement in the Survey of Consumer Finances: Methodology and directions for future research." *SCF Working Paper*, prepared for the May 2000 annual meetings of the American Association for Public Opinion Research, Portland, Oregon.

Kim, J.J., and Winkler, W.E. (1997). "Masking microdata les." *U.S. Census Bureau Research Report*, No. RR97/03.

Kish, L. (1965). *Survey Sampling*. New York: Wiley.

Krewski, D., and Rao, J.N.K. (1981). "Inference from stratified samples: properties of the linearization, jackknife, and balanced repeated replication methods." *Annals of Statistics*, 9, 1010-1019.

Kovar, J.G., Rao, J.N.K., and Wu, C.F.J. (1988). "Bootstrap and other methods to measure errors in survey estimates." *Canadian Journal of Statistics*, 16, Supplement, 25–45.

Lee, K.H. (1972). "The use of partially balanced designs for half-sample replication method of variance estimation." *Journal of the American Statistical Association*, 67, 324-334.

Lee, K.H. (1973),. "Using partially balanced designs for the half-sample method of variance estimation." *Journal of the American Statistical Association*, 68, 612-614.

Little, R.J.A., Lewitzky, S., Heeringa, S., Lepkowski, J., and Kessler, R.C. (1997). "Assessment of weighting methodology for the national comorbidity survey." *American Journal of Epidemiology*, 146, 439-449.

MatchWare Technologies, Inc. (1996). *AutoMatch: Generalized Record Linkage System User's Manual*. Silver Spring, MD: Matchware Technologies, Inc.

Mayda, J., Mohl, C., and Tambay, J. (1997). "Variance estimation and condentiality: They are related!" Unpublished Manuscript, Statistics Canada.

McCarthy, P.J. (1966). "Replication: an approach to the analysis of data from complex surveys." *Vital and Health Statistics* (Ser. 2, No. 14), Washington, DC: U.S. Government Printing Office.

McCarthy, P.J., and Snowden, C.B. (1985). "The bootstrap and finite population sampling." *Vital and Health Statistics*(Ser. 2, No. 95), Public Health Service Publication 85–1369, Washington, DC: U.S. Government Printing Office.

Moore, Jr., R.A. (1996a). "Analysis of the Kim-Winkler algorithm for masking microdata les - how much masking is necessary and sufficient? Conjectures for the development of a controllable algorithm." *U.S. Census Bureau Research Report*, No. RR96/05.

Moore, Jr., R.A. (1996b). "Controlled data-swapping techniques for masking public use microdata sets." *U.S. Census Bureau Research Report*, No. RR96/04.

Moore, Jr., R.A. (1996c). "Preliminary recommendations for disclosure limitation for the 2000 Census: Improving the 1990 condentiality edit procedure." *U.S. Census Bureau Statstical Research Report*, No. RR96/06.

Nigam, A.K., and Rao, J.N.K. (1996). "On balanced bootstrap for stratified multistage samples." *Statistica Sinica*, 6, 199-214.

Nixon, M.G., Brick, M.J., Kalton, G., and Lee, H. (1998). "Alternative variance estimation methods for the NHIS." In the *Proceedings of the American Statistical Association, Survey Research Methods Section*, 326-331.

Park, I. (2004). Confidential Westat Internal Report.

Parker, R.G. (1995). *Deterministic Scheduling Theory*. New York: Chapman & Hall USA.

Parsons, V.L., and Casady, R.J. (1986). "Variance estimation and the redesigned national health interview survey." In the *Proceedings of the American Statistical Association, Survey Research Methods Section*, 406-441.

Parsons, V.L., Chan, J., and Curtin, L.R. (1990). "Analytic limitations to current national health surveys." In the *Proceedings of the American Statistical Association, Survey Research Methods Section*, 736-741.

Pursey, S. (1999). "Disclosure control methods in the public release of a microdata file of small businesses." Presented at the Joint ECE/Eurostat Work Session on Statisical Data Confidentiality, Thessaloniki, Greece, Working Paper No. 5.

Quenouille, M.H. (1949). "Problems in plane sampling." *Annals of Mathematical Statistics*, 20, 355-375.

Rao, J.N.K., and Wu, C.F.J. (1985). "Inference from stratified samples: second order analysis of three methods for non-linear statistics." *Journal of the American Statistical Association*, 80, 620-630.

Rao, J.N.K., and Wu, C.F.J. (1988). "Resampling inference with complex survey data." *Journal of the American Statistical Association*, 83, 231-241.

Rao, J.N.K., and Shao, J. (1996). "On balanced half-sample variance estimation in stratified random sampling." *Journal of the American Statistical Association*, 91, 343-348.

Rubin, D.B. (1993). "Discussion of statistical disclosure limitation." *Journal of Official Statistics*, 9, 461-468.

Rust, K.F. (1984). *Techniques for estimating variances for sampling surveys.* Unpublished Ph.D. dissertation, University of Michigan, Department of Biostatistics.

Rust, K.F. (1986). "Efficient replicated variance estimation." In the *Proceedings of the American Statistical Association, Survey Research Methods Section*, 81-87.

Rust, K., and Rao, J.N.K. (1996). "Variance estimation for complex sample surveys using replication techniques." *Statistical Methods in Medical Research*, 5, 3, 283-310.

Shah (2001). "A method to create pseudo strata and PSU's based on BRR weights." Unpublished Manuscript, Research Triangle Institute.

Shao, J., and Tu, D. (1995). *The Jackknife and Bootstrap*, New York: Springer-Verlag.

Shao, J., and Wu, C.F.J. (1992). "Asymptotic properties of the balanced repeated replication method for sample quantiles." *Annals of Statistics*, 20, 1571-1593.

Sitter, R.R. (1992). "Comparing three bootstrap methods for survey data." *Canadian Journal of Statistics*, 20, 135-154.

Sitter, R.R. (1993). "Balanced repeated replications based on orthogonal multi-arrays." *Biometrika*, 80, 211-221.

Takemura, A. (2002). "Local recoding and record swapping by maximum weight matching for disclosure control of microdata sets." *Journal of Official Statistics*, 18, 275-289.

Tukey, J.W. (1958). "Bias and confidence in not-quite large samples." *Annals of Mathematical Statistics*, 29, 614.

Valliant, R. (1996). "Limitations of balanced half-sampling." *Journal of Official Statistics*, 12, 225-240.

Winglee, M., Valliant, R., Brick, J.M., and Machlin, S. (2000). "Probability matching of medical events." *Journal of Economic and Social Measurement*, 26, 129-140.

Winkler, W.E. (1995). "Matching and record linkage." In Cox, B.G., et al. (Eds.), *Business Survey Methods*, 355-384. New York: J. Wiley.

Winkler, W.E. (1997). "Views on the production and use of condential micro-data." *U.S. Census Bureau Research Report*, No. RR97/01.

Winkler, W.E. (1998). "Producing public-use les that are analytically valid and condential." *U.S. Census Bureau Research Report*, No. RR98/02.

Wolter, K.M. (1985). *Introduction to Variance Estimation*. New York: Springer-Verlag.

Wu, C.F.J. (1991). "Balanced repeated replications based on mixed orthogonal arrays." *Biometrika* 78, 181-188.

Yung, W. (1997). "Variance estimation for public use files under confidentiality constraints." In the *Proceedings of the American Statistical Association, Section on Survey Research Methods*, 434-439.