TUTKIMUKSIA FORSKNINGSRAPPORTER RESEARCH REPORTS 240

A THEORETICAL OVERVIEW FOR VARIANCE ESTIMATION IN SAMPLING THEORY WITH SOME NEW TECHNIQUES FOR COMPLEX ESTIMATORS

Pauli Ollila

Tilastokeskus
Statistikcentralen
Statistics Finland

TUTKIMUKSIA FORSKNINGSRAPPORTER RESEARCH REPORTS 240

A THEORETICAL OVERVIEW FOR VARIANCE ESTIMATION IN SAMPLING THEORY WITH SOME NEW TECHNIQUES FOR COMPLEX ESTIMATORS

Pauli Ollila

Tilastokeskus
Statistikcentralen
Statistics Finland

Editorial Board of the Research Report Series
The Scientific Advisory Board of Statistics Finland

Chief Editor
Head of Research
Timo Alanko

Cover design
Maija Sohlman

Layout
Pauli Ollila

© Statistics Finland 2004

ISSN 0355–2071
ISBN 952–467–319–3

Yliopistopaino

Preface

After my master's thesis, around 1993, I was both intrigued and puzzled by the variety of sample reuse methods, and – to be frank – variance estimation in general. The bootstrap methods seemed to widen the scope of variance estimation methods. The numerous different methods especially in sampling theory, using subsamples, combinations of subsamples, resamples with replacement, nonindependent random groups, combinations of random groups, etc., sometimes with weight or resampling design adjustments, indicated that there was a clear lack of a unified theoretical framework. When utilising the realised sample with a specific method, we mostly end up with the same theoretical basis as in sampling theory, but this time in the resampling context. In addition, the methodology of resample selection dependent on the previous resample outcomes (e.g. random grouping) needed some further theoretical development. My licentiate thesis from 1996 was an attempt to form such a framework to some extent. I continued the research by developing a method to improve variance estimation in the case of complex estimators. However, it was in the 1999 Baltic-Nordic Workshop of Survey Sampling when I first learned about the way of expressing the sampling design as a multivariate distribution as well as expressing the samples as vectors. This work of Docent Imbi Traat from the University of Tartu provided answers to things remaining not fully finalised in the licentiate thesis. Later, I had the privilege to have Imbi Traat as a supervisor of my work for the doctoral thesis. Her accurate and critical evaluation work during the last three years provided irreplaceable help when otherwise I probably would still be striving to finalise the thesis.

Now the thesis is complete. Firstly, it provides a theoretical overview of variance estimation, which takes into account different approaches of sampling, extends the concept of probability sampling to selecting several samples and resamples, and presents the existing methodology and its different aspects in the expanded context. Secondly, the thesis provides new theoretical results concerning correction coefficients for unbiased variance estimation (by using cumulants and k-statistics) and the sampling distributions in two-phase sampling. Thirdly, new methods for variance estimation are presented: post-design vector approach in variance estimation, variance estimation based on two-phase resample spaces and alternatively on two different resample spaces, and the methods based on the decomposition of the variance utilising sample pair probabilities.

I would like to thank several people for the scientific and other support during the years of preparing the thesis. First of all, Docent Imbi Traat will have my sincere thanks for her valuable contribution. I also wish to thank three very influential persons of the Finnish sampling scheme (and also internationally well-known), i.e. Professor Erkki Pahkinen, Professor Risto Lehtonen and Professor Seppo Laaksonen for their scientific advice, promotion of my work, and both practical and mental support during the different and sometimes difficult phases of writing. Professor Hannu Niemi from the Statistical Department of the University of Helsinki has been the key person for the continuation of my work, both encouraging and arranging practical things, such as some work periods dedicated to the thesis at the department. Furthermore, I thank Professor Gunnar Kulldorff from the University of Umeå for his important contribution to the tradition of the Baltic-Nordic Workshops on Survey Sampling. This active and discussing environment has been an essential part in my scientific progress; it is also certain that its fruitful scientific influence will be seen in the Baltic and Nordic countries in the forthcoming decades. I also thank the referees, Professor Anthony Davison from the Swiss Federal Institute of Technology and Professor Aleksandras Plikusas from the University of Vilnius for their valuable contribution in the last phase of the thesis, pointing out the important aspects and problematic issues and parts in the work. Bethany Smith did very important work by improving my sometimes quite insufficient written English in the first version of the thesis. I thank both Statistics Finland and the Statistical Department of the University of Helsinki that have been supportive by arranging financing for periods dedicated to research work and writing of the thesis. Finally, I wish to express my sincere thanks to my wife Anne and our daughter Mikaela, for the patience, support and understanding during the years of the preparation of the thesis.

Helsinki, August 2004

Pauli Ollila

CONTENTS

1. INTRODUCTION

In order to understand the problems of variance estimation in survey sampling and the basis of this thesis, a brief look at the history and the current situation of probability sampling, traditional variance estimation methods, and sample reuse methods is needed. Although probability sampling from a finite population as a theoretical statistical operation was already recognised in the scientific world at the beginning of the 20th century (see e.g. Fisher 1930, Neyman 1934), it was only in the late 1930's when it gradually became better known as a method suitable for many purposes, for example surveys from different populations (see e.g. Neyman 1938, Mahalanobis 1939). The exceptional nature of the finite population and the probability theory connected to it led to many developments in the field of sampling theory, the main achievements being the definition of sampling design, the development of design-based estimation theory, and estimation methods utilising auxiliary information from sources other than the survey itself. These early developments are covered in the essential books on sampling theory – Yates (1949), Deming (1950), Cochran (1953), Hansen et al. (1953) and Sukhatme (1954).

The increasing interest in sampling and its applications in different contexts raised the question of the quality of these methods. One crucial topic was how much the results we obtain with our estimators and selection methods vary – how big is the variance of the estimator. For some simple (e.g. linear) estimators there were analytical unbiased variance estimators available, but for more complex estimators special efforts had to be made. One sample as such was not a sufficient basis for getting enough information about the variation of the estimator. The choice for a smooth nonlinear estimator was the Taylor linearisation method, familiar from the field of classical statistics. Here the problem of one sample was omitted by approximating the variance. The estimator was linearised and the traditional sampling theory for variance estimation of that linearised estimator was used. On the other hand, the non-smooth statistics, especially quantiles, were of interest as well. Woodruff (1952) presented a simple method for estimating of the confidence interval of the median estimator. The principle can be applied with other quantiles as well, excluding extreme order statistics such as the minimum and the maximum. The idea of finding the variance of the median estimator was in fact discussed very little in the literature (e.g. Rao and Wu 1987, Francisco and Fuller 1991).

In this thesis the most attention is paid to variance estimation methods based on the reuse of the sample. Some simple variance estimation methods of this kind were developed in the middle of the 20th century. One of the first examples of subsampling was presented in Mahalanobis (1946a). The method of dependent random groups (Hansen et al. 1953, Wolter 1985) randomly divided the sample into several groups, and the estimates calculated separately from these groups were utilised in variance estimation. The jackknife method, developed in general by Quenouille (1949), was not originally developed for variance estimation in survey sampling, but for bias reduction purposes in the field of classical statistical theory. The idea of the method was to divide a sample into A separate non-overlapping random groups and to utilise the units included in every combination of $A-1$ random groups for the calculation of estimates. The variance and bias estimation was based on these estimates. The extension to survey sampling was carried out by Durbin (1959). The most popular alternative of this method is still the $n-1$ jackknife, where every without-replacement subsample of size $n-1$ is created and then utilised for variance estimation purposes.

There was growing interest in the reuse of the sample, one crucial example being the pseudo-replication method by McCarthy (1969), also known as Balanced Half-Samples and Balanced Repeated Replications. It was based on the use of matrices with some balance properties for some extreme sampling situations with an extensive stratification and two units (usually clusters) selected in every stratum. At the end of the 1970's powerful development of resampling methods in classical statistics began. Efron's work (e.g. 1979, 1982) was the foundation of bootstrap theory, including resampling with replacement from an i.i.d. sample, usually of the same size as the original sample. The main idea was to independently repeat the resampling procedure a large number of times in order to get many estimates of a parameter. The estimates were used also for variance estimation, as well as confidence interval or bias calculation. Very quickly these thoughts were applied in the sampling context as well. However, the topic was much more complex in the finite population case, and presented numerous questions such as: how to react to different sampling methods with possibly unequal probabilities; what kind of effect the stratification has, especially with small stratum sizes; how to deal with a conceptual difference between the original sample and the resamples due to the different sample/resample spaces; what kind of resampling scheme should be used when the original sample is selected with or without replacement.

Several approaches were introduced in order to apply bootstrap ideas in the finite population sampling context. Gross (1980) presented a method creating a *pseudopopulation* by "copying" the sample of size n_a m times to create a set of approximately the same size as the original population and with some conditions. From this resamples of size n_b (usually near n_a) were selected. This approach was further developed by Bickel and Freedman (1984), Chao and Lo (1985), Sitter (1992a), Booth et al. (1994). McCarthy and Snowden (1985) first presented the subsampling approach (*Bootstrap With Replacement, BWR*) for the finite population case. The procedure by Sitter (1992b) was based on m resamples selected independently without replacement. These resamples of size n_{b0}, usually the nearest integer value to n_a / m, were combined together, and an estimate was calculated from that combined set of observations (size $n_b = n_{b0}\, m$). Rao and Wu (1988) developed a *scaling method for smooth functions of population means* using with replacement resamples with smaller size than n_a, i.e. the original sample size. The method was further developed by Rao et al. (1992) by *rescaling the survey weights*. This method is applicable for non-smooth functions as well.

Today, the development of computers and statistical software has created a favourable situation for resampling methods in everyday practice. Vast multiple reuse of the information obtained in a survey is possible. On the other hand, the current situation of finite population resampling theory is somewhat sparse. No unified treatment of resampling exists. This holds especially for complex sampling designs and complex estimators. Furthermore, the existing methods usually work unsatisfactorily with small sample sizes and smaller populations/strata. Therefore, variance estimation in a finite population still needs further improvements and new solutions. In this thesis some ideas are presented and studied.

Chapter 2 of the thesis is devoted to three different approaches to defining the sample (ordered set, design vector, subset) and, correspondingly, to the different approaches of sampling design. Some description of sampling practices at different levels (population, sample, set of samples, set of resamples) is also given. In Chapter 3 the basic ideas of variance estimation are given, along with an overview of the existing variance estimation methods for complex estimators. These chapters are necessary for the reader to see the origin of the existing methods and especially to understand the essence of the new variance estimation methods presented in this work. In fact, none of the available resampling methods take into account the special features of different estimators. Often, the corrected conditional variance of the resampling estimator (conditioned by the sample) is used for variance estimation, but the current correction principle is insensitive to the different behaviour of the conditional variance of different estimators. Thus one of the key ideas of this thesis is to provide alternative methods for more detailed correction in the variance estimators.

Some new theoretical results and variance estimation techniques for *complex estimators* are presented in this thesis. The examples of the estimators presented in this thesis are from the field of survey sampling, and one could find much more complex estimators from the other areas of statistics. However, if we exclude results for some specific estimators, the new methods developed for resampling are presented in a general way. Usually the simplest sampling designs are assumed – simple random sampling without replacement (SI) and with replacement (SIR). In Chapter 4 the main principle of correction is dealt with theoretically. The primary finding is that by using cumulants and k-statistics one can derive estimator-specific correction coefficients for different sampling/resampling combinations. *It has been suggested that the correction coefficient of nonlinear estimators depends also on the population distribution of the y-variable.* The analytical form of the correction coefficient are derived for the estimator of the population variance. The results are analysed with respect to different population distributions and different sample/resample sizes. This approach provides tools for evaluating existing variance estimation methods as well. It has been shown that the traditional linear case correction coefficient does not work properly for nonlinear estimators. It causes a bias, e.g. under SIR sampling the traditional jackknife method is positively biased, producing variance estimates that are too high on average. Limits of the resample sizes causing minimal bias of the variance estimator are derived for observed situations.

The novel variance estimation method in Chapter 5 is based on the post-design vector. Adjustments are often required in order to correct the difference between the original and resampling designs as well as scale and weight differences. Here this problem is solved by artificially creating (in practice unevenly expanding) the resampling design vector with some conditions based on the theoretical properties of the occurrence counts for the different sampled units. The introduced post-design vector method serves e.g. as an alternative to randomisation, avoiding the need for two resample sizes.

The variance estimation methods in Chapter 6 (also novel) include an estimator-dependent nonlinearity part in the correction. The first method of bias reduction utilises information from resampling in two phases. Resampling with two resample sizes is a shortcut method for two-phase resampling. Finally, theoretically it is possible to define a correction in order to achieve the minimal mean square error of the variance estimator. That theorem is utilised for variance estimation purposes as well.

In Chapter 7 the theory of two-phase sampling is utilised for developing new results when combining multinomial first phase design and simple random sampling without replacement in the second phase. Furthermore, some variance estimators are based on the use of sample pair probabilities. The idea is to use decomposed formulae of the variance and the conditional variance and to adjust the sample/resample pair probabilities in the respective formulae. In simple random sampling without replacement the sample pair probabilities reduce to the combinatorial terms, and the sample pair probabilities for sampling and two-phase sampling (i.e. resampling situation) are then used. The first variance estimator uses the theory of conditionality and the second adjusts the resample pair probabilities based on the numbers of joint units of the pairs.

In Chapter 8 simulation results for two small data sets are presented including most of the existing variance estimation methods as well as the new methods from Chapters 5-7. The tables include relative biases and mean square errors for the variance estimators. The results are summarised at the end of the chapter. The program code for this simulation is given in Appendix C.

2. PROBABILITY SAMPLING

2.1. Basic Definitions

The *population* is denoted by $U = \{u_1, u_2, \ldots, u_N\}$ including N identifiable *elements*. Another frequently used option is to identify the population with indices, i.e. $U = \{1, 2, \ldots, N\}$. An N x q matrix $\underline{X} = [\underline{x}_1\ \underline{x}_2\ \ldots\ \underline{x}_N]'$, with $\underline{x}_i = [x_{i,1}\ x_{i,2}\ \ldots\ x_{i,q}]$ representing the values of q variables of the population element u_i, is called the *population data matrix.*

Table 2.1. Population and Its Data Matrix

U	$\underline{X}$			
u_1	$x_{1,1}$	$x_{1,2}$	...	$x_{1,q}$
u_2	$x_{2,1}$	$x_{2,2}$	...	$x_{2,q}$
...	...	...	...	...
u_N	$x_{N,1}$	$x_{N,2}$	...	$x_{N,q}$

A function of the matrix $\underline{X}$, i.e. $\theta = g(\underline{X})$, is called a *parameter of the population.* In some sources, e.g. Cassel et al. (1977) the matrix $\underline{X}$ is considered as a parameter of the population and then $\theta = g(\underline{X})$ is called a *parametric function.*

There are several definitions for the concept of a *"sample"* in the sampling literature, see e.g. Godambe (1955) and (1970), Hanurav (1966), Cassel et al. (1977). The broadest possible definition is: *a sample is a collection of elements involving only population elements, and it may involve zero elements.* In some contexts in the text the elements of the sample are called the *observations.*

Let us denote the sample by $os = (u_{(1)}, u_{(2)}, \ldots, u_{(n)})$ where $u_{(1)}$ is the element appearing first in the sample, $u_{(2)}$ the second etc. This form of presentation is called an *ordered sample,* see e.g. Godambe (1955), Cassel et al. (1977), Särndal et al. (1992). The sample *os* contains *the selection order of elements.* The element at the i^{th} draw may be any population element $u_{(i)} = u_j \in U$ where $j = 1, ..., N$. For example, (6, 1, 3, 7, 1) is a sample provided that the population includes such elements. The sample *os* has a size of n, i.e. the number of draws, and it may include *repeats* of the population elements. Here we bring also another ordered form for a sample, $(u_{<1>}, u_{<2>}, \ldots, u_{<n>})$, where $u_{<i>}$ appears earlier in the population frame than $u_{<i+1>}$ or $u_{<i>} = u_{<i+1>}$. This is called the *ordered-by-population sample.* This presentation, if used, is more common for a sample with distinct units, that is, where units in the sample do not repeat, i.e. $u_{<i>} \neq u_{<i+1>}$. In many cases only *distinct population elements in the sample* are used for estimation (see e.g. Basu 1958, Raj and Chamis 1958, and Cassel et al. 1977). The corresponding set is $s_d = \{d_1, d_2, \ldots, d_\nu\}$ with $d_1, d_2, \ldots, d_\nu$ being ν distinct elements from a population.

Traat (2000) described samples in the form of a vector, i.e. $\underline{k} = (k_1, k_2, \ldots, k_N)$ where k_i $(i = 1, \ldots, N)$ is the number of times the element u_i appears in the sample. Here we call this vector a *vector sample*, and it is a point in the N-dimensional space of non-negative integers. Different presentations of a sample are illustrated in Table 2.2.

Table 2.2. Different Presentations of a Sample for $U = \{1, 2, 3, 4\}$ and $n = 3$

Ordered Sample			Ordered-by-Population Sample			Vector Sample				Set of Distinct Population Elements			Size of the Set
$u_{(1)}$	$u_{(2)}$	$u_{(3)}$	$u_{<1>}$	$u_{<2>}$	$u_{<3>}$	k_1	k_2	k_3	k_4	d_1	d_2	d_3	r
1	1	1	1	1	1	3	0	0	0	1	-	-	1
2	2	2	2	2	2	0	3	0	0	2	-	-	
3	3	3	3	3	3	0	0	3	0	3	-	-	
4	4	4	4	4	4	0	0	0	3	4	-	-	
1	1	2	1	1	2	2	1	0	0	1	2	-	2
1	2	1											
2	1	1											
1	2	2	1	2	2	1	2	0	0				
2	1	2											
2	2	1											
1	1	3	1	1	3	2	0	1	0	1	3	-	
1	3	1											
3	1	1											
•••			•••			•••				•••			
3	4	4	3	4	4	0	0	1	2	3	4	-	
4	3	4											
4	4	3											
1	2	3	1	2	3	1	1	1	0	1	2	3	3
1	3	2											
2	1	3											
2	3	1											
3	1	2											
3	2	1											
•••			•••			•••				•••			
2	3	4	2	3	4	0	1	1	1	2	3	4	
2	4	3											
3	2	4											
3	4	2											
4	2	3											
4	3	2											

Repeated situations in the table are left blank. Here we have $n^N = 3^4 = 81$ different ordered samples. For each ordered-by-population sample, as well for each vector sample there are $n! / \prod_{i=1}^{N} k_i!$ different ordered samples, e.g. for the vector sample (2, 1, 0, 0) we have 3! / (2! 1! 0! 0!) = 6 / 2 = 3 ordered samples. It can be seen that the ordered-by-population sample and the vector sample are in fact different forms of the same thing: the first being the collection of elements form, and the second the vector form. The drawing order of elements is lost in both of them. The set of distinct elements omits the count information; only appearance in the sample is taken into account.

In Table 2.2 we saw the possible samples with $n = 3$ for a population of size $N = 4$. In general, if for a fixed N the sample size n is allowed to be any non-negative integer, we have an *infinite set of ordered samples*. This set is called here a *sample space* $\mathcal{S}$.

2.2. Defining Sampling Design

Sampling design. For the population $U = \{1, 2, \ldots, N\}$ we have the infinite sample space $\mathcal{S}$. A sample is denoted by *os* (ordered sample). Let us define a function $p(\bullet)$ on $\mathcal{S}$ such that $p(os) \geq 0$ for all $os \in \mathcal{S}$ and $\sum_{os \in \mathcal{S}} p(os) = 1$. This function gives the probability $p(os)$ of selecting the sample *os* from $\mathcal{S}$ and is called a *sampling design*. The sample *os* is an outcome of the random quantity *OS*. The probability distribution of *OS* is defined by $p(\bullet)$ which is considered to be the broadest definition of the sampling design.

Reduction of the sample space. In practice, the infinite sample space is too wide for specified types of sampling designs. For them a *restricted sample space* is more appropriate, i.e. such samples that will not have a positive probability under the sampling design being considered are excluded. The reduction is performed by restricting sample size or values of a vector sample, by forming disjoint groups of the population for sampling, by other methods. The *sample size n* can be restricted to an *interval* $[n_L, n_U]$, *a number of different sample sizes* (here z distinct samples) $n_1, \ldots, n_z$ may be defined, or a *fixed sample size* can be used. The number of times each element appears in the sample can be seen from the *vector sample*, and usually we have some limitations on the possible counts of elements. Making disjoint groups in the population is a practice appearing in some sampling designs. For example in *stratified sampling* the selection process is conducted independently in each disjoint group, but in *cluster sampling* some groups from the set of all disjoint groups are selected and then the units in the selected group are selected as a whole. The sample space is considerably restricted with these sampling designs.

Sampling scheme. In order for a sample to be selected there must be some rules about how to include elements in the sample so that the probabilities of the sampling design are fulfilled. These rules are called a *sampling scheme*.

Some sampling schemes and restriction principles. The selection probabilities can be defined for each element of the population or for some groups of elements ("clusters"). Depending on the situation, the elements of the population or these groups of elements can be considered as *sampling units*. It is typical for sampling schemes that the *probabilities used in selection can change* during the sampling process. If we select each element of the sample independently of others from the whole population we have a *with-replacement sample* (WR sample). On the other hand, the draws of elements can be such that the outcome of the draw depends on previous draws. When we exclude the previously chosen elements from the population elements before the next draw, we have a *without-replacement sample* (WOR sample). Correspondingly, when these sampling schemes use equal unit probabilities within a draw we have *simple random sampling with replacement* (SIR), and *simple random sampling without replacement* (SI), though in the latter case the probabilities change from draw to draw.

The *sampling design*, the *sampling scheme* and the *restricted sample space* together form an entity needed for the work of the researcher. The notion of the restricted sample space is considered to be important in this work, though the sampling design in fact defines the restricted sample space. When studying variance estimation based on sample reuse, we find a *variety of ways to construct the design, the scheme and the restrictions of the sample space for resampling*. In that context they serve as tools for variance estimation. Next we present a few of the most common designs and reduction principles of the sample space, and in addition some examples of practices appearing only in variance estimation. They are considered to be important for understanding the subsequent chapters. For a more general view on sampling designs, a classification of different designs with equal or unequal selection probabilities can be found in Cassel et al. (1977). See also Brewer and Hanif (1983) for a variety of unequal probability designs, and for some other designs see Traat et al. (2000), Rosén (1997) and Kröger et al. (1999).

Design vector approach. Traat (2000), and also Traat et al. (2000), present the sampling situation in terms of vectors and emphasise the distributional nature of the sampling design. The vector $\underline{I} = (I_1, I_2, \ldots, I_N)$ is a *random design vector* where I_i represents the number of selections for the unit $i \in U$. The *realisation of the design vector* $\underline{I}$, denoted by $\underline{k} = (k_1, k_2, \ldots, k_N)$, is a point in the N-dimensional space of non-negative integers $\underline{k} \in \mathrm{N}^N$. The distribution connected to these points, i.e. the multivariate distribution of the vector $\underline{I}$, is called the sampling design. The probability function of $\underline{I}$ is denoted by $p(\underline{k})$, i.e.

$$p(\underline{k}) = \Pr\{\underline{I} = \underline{k}\}, \qquad \sum_{\underline{k}} p(\underline{k}) = 1, \quad \underline{k} \in \mathrm{N}^N,$$

where $\underline{I} = \underline{k}$ means $I_i = k_i$ for all i, $k_i \in \{0, 1, \ldots\}$. The sums $\sum_{i=1}^{N} I_i$ and $\sum_{i=1}^{N} k_i$ describe the sizes of the random sample size and its realisation, respectively.

Traat et al. (2000) deal with several different sampling designs based on distributional theory. Here three common designs are presented for use in the subsequent theoretical developments. The first one is *SI design*, or in terms of the traditional distributions, *simple multivariate Bernoulli distribution*. Its probability function is

$$p(\underline{k}) = \binom{N}{n}^{-1}, \tag{2.2.1.}$$

when $k_i \in \{0,1\}$ and $\sum_{i=1}^{N} k_i = n$; $p(\underline{k}) = 0$ otherwise.

Sampling with selection probabilities proportional to some size variable can be easily applied for with-replacement sampling design. The with-replacement sampling design (in the language of distributions a multinomial distribution, see also Johnson et al. 1997) is of the form

$$p(\underline{k}) = n! \prod_{i=1}^{N} \frac{p_i^{k_i}}{k_i!}, \tag{2.2.2.}$$

where $k_i \in \{0, 1, \ldots, n\}$ and $\sum_{i=1}^{N} k_i = n$; $p(\underline{k}) = 0$ otherwise. Here this design is called a *multinomial design* and denoted by $M(n; p_1, \ldots, p_N)$. Its special case with equal single-draw probabilities is *SIR sampling*, which may be called *simple multinomial design* (Traat 2000, Brewer and Hanif 1983), and is of the form

$$p(\underline{k}) = \frac{n!}{N^n \prod_{i=1}^{N} k_i!}, \tag{2.2.3.}$$

where $k_i \geq 0$ and $\sum_{i=1}^{N} k_i = n$; $p(\underline{k}) = 0$ otherwise.

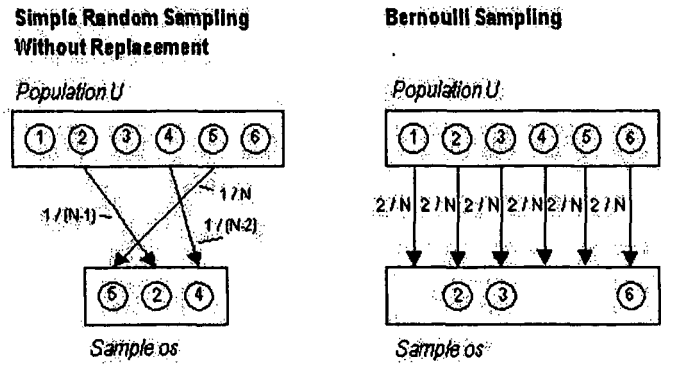
When studying the vector sample, we see that WOR sampling of elements has the vector sample restriction $k_i \in \{0,1\}$, where $i \in U$. In addition, some resampling designs specified for variance estimation purposes may have this limitation as well. For example the *pseudopopulation approach* (e.g. Gross 1980, Bickel and Freedman 1984, Sitter 1992a, Booth et al. 1994) in its simplest form (repeating the population m times and selecting a WOR sample from that pseudopopulation) is a case of setting possible counts for unit i, $k_i \in \{0, 1, \ldots, m\}$.

Subset approach. Familiar from Särndal et al. (1992), the *subset approach* differs from the broadest definition of the sample space. Here we have a finite, restricted sample space S_{sub} including the *subsets of the population U* (from an empty set to the population itself). Thus it includes only WOR samples. The size of this set of samples is 2^N. A sample is denoted by s. The function $p(\bullet)$ gives the probability $p(s)$ of selecting the sample s from S_{sub}. This function is a *sampling design*. The sample s is an outcome of the set-valued random variable S. The probability distribution of S is defined by $p(\bullet)$ with conditions $p(s) > 0$ for all $S \in S_{sub}$ and $\sum_{s \in S_{sub}} p(s) = 1$.

2.3. *Sampling from a Population or Sample*

Basic ideas. The ordinary practice in sample surveys is *sampling from the population.* It forms the basis for satisfactory estimation of population parameters. The variety of selection methods is created due to either practical reasons or the efficiency of the estimation or both. Figure 2.1 shows two examples of different sampling designs and realised samples.

Figure 2.1. Two Examples of Sampling from the Population, $N = 6$

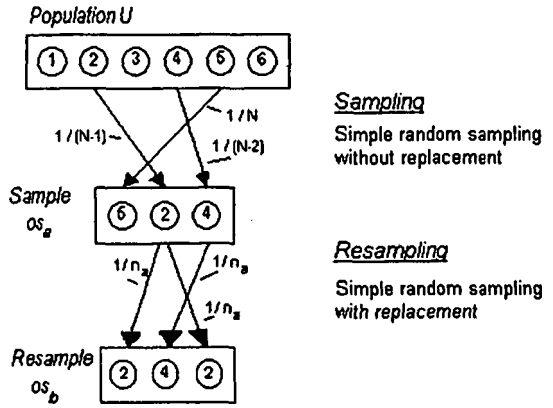


In SI sampling we have a drawing probability that changes every time a unit is selected and then eliminated from the next selection. In Bernoulli sampling selection is decided individually for each unit, starting from the first. The selection probability remains constant due to the independence of the selections; the consequence is a random sample size (in this example the expected sample size is 2).

Sampling from a sample may occur as an ingredient of the sampling scheme designed for drawing a sample from a population. This is the case with classical two-phase sampling or double sampling (originally Neyman 1938, see also Cochran 1977, Särndal et al. 1992). However, the special needs concerning mainly variance, confidence interval and occasionally bias estimation require the reuse of the sample. Then repeated sampling from the sample becomes a useful tool. *Resampling* as a concept is applied for the second phase.

Figure 2.2 illustrates an example of resampling. The sample and the resample are shown in terms of ordered samples.

Figure 2.2. Two-Phase Sampling (SI sampling and SIR resampling)



Some resampling methods define probabilities in such a way that it is beneficial to express resampling with the design vector approach. Then a *resample* $\underline{k}_b$ is selected either from the sample $\underline{k}_a$ or its modification $\underline{k}_{a,mod}$. The modification can be defined as $\underline{k}_{a,mod} = (qk_1, \ldots, qk_N)$, where q is a positive integer and k_i (for every $i \in U$) is an outcome of the vector $\underline{k}_a$. In other words the modification contains only units from the sample $\underline{k}_a$, but they can appear once or more. The most common example of this modification is the *pseudopopulation* or *empirical population* (e.g. Gross 1980, Bickel and Freedman 1984, Sitter 1992a, Booth et al. 1994) for a specific bootstrap application. The *resampling design* is denoted by $p(\underline{k}_b \mid \underline{k}_a)$ or if necessary, by $p(\underline{k}_b \mid \underline{k}_{a,mod})$. Index a refers to the first phase and b to the second phase. Sample sizes in phases are denoted by n_a and n_b, respectively.

Some resampling designs. In principle all designs used in ordinary surveys for drawing a sample can be used in resampling situations as well, but it is wise to choose a resampling design that won't introduce unnecessary bias in variance or confidence interval estimators. The first resampling designs were introduced in the late 1940's. In the infinite population context Quenouille (1949) presented the *jackknife method*, which can be interpreted as a without-replacement resampling design with a resample size of $n-1$. However, in practice these resamples are not chosen randomly, but instead the whole resample space is used for variance estimation (see Section 3.4). Durbin (1959) considered first the jackknife method in the finite population framework. Resampling without replacement with a resample size of $n-2$ or lower appears more rarely in practice.

The *bootstrap approach* for independent and identically distributed (i.i.d.) observations (Efron 1979 and 1982) includes independent selections of with-replacement resamples of size n. This practice was soon adopted for various purposes in classical statistics. However, it is not common to use a resample size other than n in the i.i.d. situation; recently the topic of subsampling has been dealt with in e.g. Politis and Romano (1994), Bickel et al. (1997) and the book by Politis et al. (1999). They are studying the properties of this practice as well in terms of classical statistical theory, and also without-replacement resampling. McCarthy and Snowden (1985) presented a with-replacement resampling method (*Bootstrap With Replacement, BWR*) as second-phase sampling for the finite population case.

In addition, there are some resampling designs which do not appear in ordinary sampling situations. These special designs are created mainly for variance estimation purposes. For example, they may allow some repetitions of observations in the resample $\underline{k}_b$, but not in a strict with-replacement sampling manner. The resample is defined as a vector $\underline{k}_b = (q_1 k_{b1}, ..., q_N k_{bN})$, where $q_i \geq 0$ is an integer, and k_{bi} is the number of appearances of the unit i in the resample under this special resampling design.

The resampling design by Sitter (1992b) is based on m resamples selected independently without replacement. These resamples of size n_{b0}, usually equal to the nearest integer value to n_a / m, are combined together, and an estimate is calculated from that combined set of observations (size $n_b = n_{b0} m$). This yields an upper bound m to the count number of unit i. When $m = 1$ we naturally ge SI resampling, and correspondingly when $n_{b0} = 1$ the resampling design yields SIR sampling.

Another resampling method (Gross 1980, Bickel and Freedman 1984, Chao and Lo 1985, Sitter 1992a, Booth et al. 1994) first creates a pseudopopulation by "copying" m times the sample of size n_a to a set of approximately the same size as the original population, with some conditions. From that a sample of size n_b is selected. This resampling design can be expressed in terms of the vector approach. For example with SI resampling we have a multivariate hypergeometric distribution for resamples (in terms of the set approach expressed in Chao and Lo 1985)

$$p(\underline{k}_b \mid \underline{k}_{a,\mathrm{mod}}) = \frac{\prod_{i=1}^{N} \binom{m}{k_{bi}}}{\binom{mn_a}{n_b}}, \qquad (2.3.1.)$$

where $\underline{k}_b = (k_{b1}, \ldots, k_{bN})$, N is the pseudopopulation size, and the value k_{bi} is always zero if the unit i is not in the sample $\underline{k}_a$.

In Shao and Tu (1995) a theoretical overview of the jackknife and bootstrap methods is given as well as a chapter on finite population resampling. In Davison and Hinkley (1997) there is a thorough presentation about bootstrap methods and their application. The asymptotic properties of these methods have also been studied in the finite population case; see e.g. Krewski and Rao (1981), Bickel and Freedman (1984), Rao and Wu (1985 and 1988), Sitter (1992b) and Booth et al. (1994). Sitter (1992a) and Presnell and Booth (1994) have conducted general comparisons of these methods.

2.4. *Sampling from a Set of Samples or Resamples*

Sampling from the population can be considered as a special case of *sampling from the set of all samples*, which is called *metasampling* (Ollila 1996). Selecting several samples from the set of samples is quite rare in practical surveys, but for variance estimation an example is the *independent random groups method* (see Wolter 1985). This practice has also been called *interpenetrating samples* in Mahalanobis (1939 and 1946b), *replicated sampling* in Deming (1956) and *random group method* in Hansen et al. (1953). However, when a simulation is involved, this practice can be interpreted as a Monte Carlo method.

When *sampling from the set of all resamples* is considered, we end up with well-known methods like the *bootstrap replications* and some *random group* and *jackknife* procedures (see Section 3.5). The target is either to estimate the conditional variance (possibly modified or scale corrected), or to provide a simple applicable method for variance estimation. *The concept of metasampling is useful in the resampling context where some of the methods have restrictions, which is impossible to handle in terms of ordinary sampling design terminology* (see Section 3.5).

Following principles of ordered sample theory, we define a *metasample* $os^* = (s_1, \ldots, s_A)$, where $s_1, \ldots, s_A$ are samples from the set of samples. The *metasampling design* $p(os^*)$ defines the probability distribution of the metasample os^*. Correspondingly, sampling from the set of resamples conditioned by s_a has the metasampling design $p(os^* \mid s_a)$. The *metasample space* $\mathbf{S}^*$ is a set of metasamples with $p(os^*) > 0$ (for the conditional situation $p(os^* \mid s_a) > 0$).

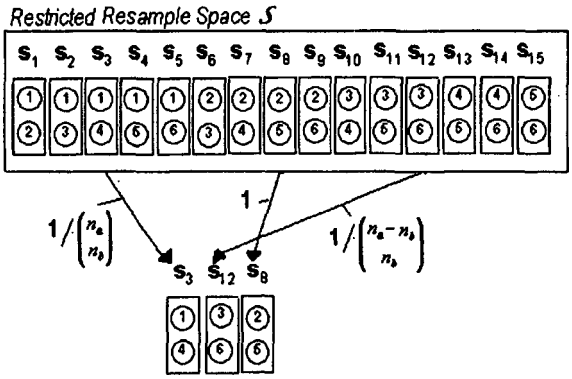
As noted before repeated sampling from the population is a special case of the metasampling design, in which $p(os^*) = p(s_1) \ldots p(s_A)$. The same holds in the resampling context. Next we give an example in which the metasampling design is such that the next selection of a resample *depends on the previous one*. This holds for instance for the *dependent random groups method* (in some texts the *nonindependent random groups method*, Hansen et al. 1953, see Wolter 1985). It is carried out by dividing observations of the sample into A disjoint groups of size n_b ($n_b + 1$ is possible for some groups as well, if $An_b < n_a$). The partition must follow the nature of the original sampling design, e.g. the random groups must be of a systematic form, if the sampling design is systematic. In our example we define $An_b = n_a$.

In metasampling terms a metasample of size A is drawn from the set-type resample space, in which the number of different resamples is $C = \binom{n_a}{n_b}$. Metasampling is carried out in the following way: first we draw resample s_{b1} of size n_b, then only those resamples with no joint units with s_{b1} are allowed for selection, and correspondingly at the subsequent phases all units from the previous resamples are not allowed in the resamples which could be selected. The metasampling design based on the random groups method with equal subsample sizes n_b and $n_a = An_b$ is of the form (Ollila 1996)

$$p(os^* \mid s_a) = \frac{1}{\prod_{g=1}^{A} \binom{n_a - (g-1)n_b}{n_b}}, \qquad (2.4.1.)$$

and in this example every existing os^* has an equal probability. In Figure 2.3 we have a realisation of this example with $n_a = 6$, $n_b = 2$ and $A = 3$. Sample s_3 is selected first, sample s_{12} second and sample s_8 third. The probabilities follow (2.4.1). Note that no population elements appear more than once in these three resamples.

Figure 2.3. Selecting a Metasample with Disjoint Units from the Space of Set-type-resamples



Here $\{s_3, s_{12}, s_8\}$ is a metasample of size 3.

2.5. *Estimator and Its Properties*

The general notation of a *parameter* is θ. Correspondingly, the *estimator of a parameter* is denoted by $\hat{\theta}$, and it is defined in the sample space which is defined by the sampling design $p(\underline{k})$. The *estimate* $\hat{\theta}_{\underline{k}}$ has a subscript referring to a specific sample $\underline{k}$. The estimator $\hat{\theta}$ may have values (i.e. estimates) for all the samples in the sample space. However, it is possible that for one or more samples it may *not* have values. This can happen due to the design, e.g. a with-replacement sample (u_3, u_3, u_3) won't produce a correlation coefficient. It can also happen because the data are such that the estimate cannot be calculated, e.g. for a correlation coefficient elements u_1, u_2, u_3 have the y values 5, 5, 5.

Below we define the most frequently used properties of estimators, appearing later in the thesis. They are so-called design-based properties, evaluated with respect to the sampling design.

Expectation of the estimator

$$E(\hat{\theta}) = \sum_{\underline{k}} p(\underline{k})\hat{\theta}_{\underline{k}} \,. \tag{2.5.1}$$

Bias of the estimator

$$B(\hat{\theta}) = E(\hat{\theta}) - \theta \,. \tag{2.5.2}$$

Variance of the estimator

$$\begin{aligned} V(\hat{\theta}) &= \sum_{\underline{k}} p(\underline{k})(\hat{\theta}_{\underline{k}} - E(\hat{\theta}))^2 \\ &= E(\hat{\theta}^2) - [E(\hat{\theta})]^2. \end{aligned} \tag{2.5.3}$$

Mean square error of the estimator

$$\begin{aligned} MSE(\hat{\theta}) &= \sum_{\underline{k}} p(\underline{k})(\hat{\theta}_{\underline{k}} - \theta)^2 \\ &= V(\hat{\theta}) + [B(\hat{\theta})]^2. \end{aligned} \tag{2.5.4}$$

Standard deviation of the estimator

$$S(\hat{\theta}) = \sqrt{V(\hat{\theta})} \,. \tag{2.5.5}$$

Relative standard deviation of the estimator

$$RS(\hat{\theta}) = S(\hat{\theta}) / E(\hat{\theta}) \,. \tag{2.5.6}$$

Relative bias of the estimator

$$RB(\hat{\theta}) = B(\hat{\theta}) / \theta \,. \tag{2.5.7}$$

The most frequently estimated parameter in sample surveys is a total $t = \sum_U y_i$. An unbiased estimator for it is

$$\hat{t} = \sum_U \frac{I_i y_i}{E(I_i)}, \qquad (2.5.8.)$$

which is a Horvitz-Thompson estimator for WOR-designs (then $E(I_i) = P(I_i = 1) = \pi_i$). The estimator will be called linear if it is linear in I_i. Thus, estimators of the type (2.5.8) and their linear functions are *linear estimators*.

3. MAIN IDEAS FOR VARIANCE ESTIMATION

3.1. Several Independent Samples Available

If it is decided that several independent samples will be selected from the population with the same design, we get the variance estimator of the estimator $\hat{\theta}$ very easily, i.e.

$$\hat{V}(\hat{\theta}) = \sum_{g=1}^{A} \frac{(\hat{\theta}_g - \bar{\hat{\theta}})^2}{A-1}, \qquad (3.1.1.)$$

where A is the number of independent samples, $\hat{\theta}_g$ is the estimate on sample g, and

$$\bar{\hat{\theta}} = \sum_{g=1}^{A} \frac{\hat{\theta}_g}{A}. \qquad (3.1.2.)$$

Due to the independence between the $\hat{\theta}_g$'s, this estimator is unbiased for the true variance $E_a[\hat{V}(\hat{\theta})] = V(\hat{\theta})$.

On the other hand, having several samples available we can use the estimator (3.1.2) for estimating θ. Then we get the variance estimator for that (see e.g. Wolter 1985, page 33), i.e.

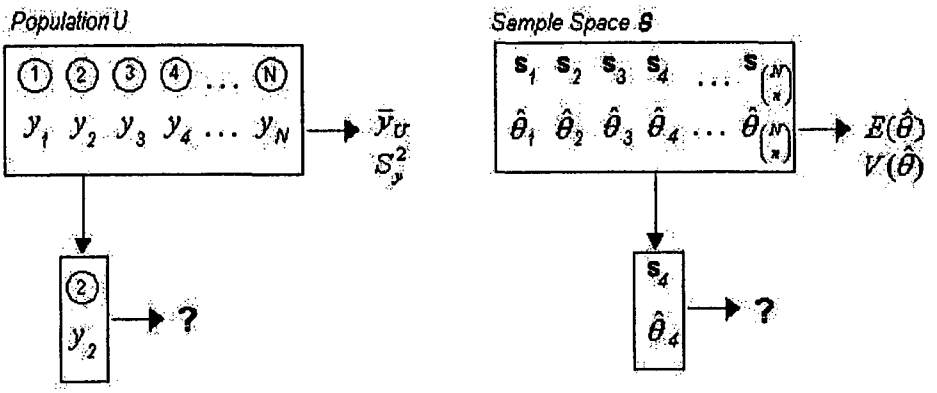
$$\hat{V}(\bar{\hat{\theta}}) = \sum_{g=1}^{A} \frac{(\hat{\theta}_g - \bar{\hat{\theta}})^2}{A(A-1)}. \qquad (3.1.3.)$$

However, this procedure is not common in sample surveys, mainly due to impracticalities and small numbers of available independent samples, so that using $\hat{\theta}_g$ for variance estimation causes instability in some cases.

3.2. One-Sample Problem

When we estimate the parameter $\theta = g(\underline{X})$, i.e. the function of the population data matrix $\underline{X}$, we can almost always rely on one sample. This self-evident fact together with well-chosen sampling designs and estimation methods creates the basis for unbiased or approximately unbiased estimation of most of the usual parameters. If we are left with only one unit in the sample, we get no proper estimator for many parameters. One of them is the population variance, i.e. S_y^2. If we are left with only one sample it may seem difficult to find the variance of an estimator.

Figure 3.1. One-unit / One-sample Problem in Estimation



In Figure 3.1 $\hat{\theta}$ is evaluated in the restricted sample space (fixed size n WOR samples). Usually we have only one sample available for estimation, here s_4. A crucial issue is how to estimate the variance with one unit, i.e. one-sample.

3.3. *Solution 1: Units of the Sample*

The first alternative is to use the *units of the original sample* for variance estimation (without any resampling). We call it variance estimation at the unit level. The parameter function $\theta = g(\underline{X})$ may be such that for some designs and estimators an *analytical unbiased variance estimator* can be obtained (e.g. the π *estimator* for the designs with all second order inclusion probabilities being positive, see e.g. Särndal et al. 1992). On the other hand, for that same parameter θ other designs and estimators can be such that *only a biased variance estimator* is available (e.g. the *generalised regression estimator*, see Särndal et al. 1992). The parameter function may be too complex to get an unbiased variance estimator based on the sample units. For example, for the regression estimator the solution is based on the *linearisation technique*, familiar from mathematics and classical statistical theory (see e.g. Wolter 1985 and Särndal et al. 1992). This technique is also widely used for nonlinear "smooth" functions of totals $\theta = f(t_1,...,t_q)$.

Let us consider the first-order Taylor approximation of the estimator $\hat{\theta}$, i.e.

$$\hat{\theta}_0 = \theta + \sum_{j=1}^{q} a_j(\hat{t}_j - t_j) \qquad (3.3.1.)$$

with $\hat{t}_j$ being the π estimator of the total t_j, and the factors being partial derivatives, $a_j = \frac{\partial f}{\partial \hat{t}_j}$ at the point $\underline{\hat{t}} = \underline{t}$. The variance of $\hat{\theta}_0$, which equals the variance of $\sum_{j=1}^{q} a_j(\hat{t}_j - t_j)$, is considered to be a good approximation of the real variance of the estimator, i.e. $V(\hat{\theta}_0) \approx V(\hat{\theta})$. Thus the analytical variance estimator based on Taylor's approximation is simply

$$\hat{V}(\hat{\theta}) = \hat{V}(\hat{\theta}_0) \qquad (3.3.2.)$$

This technique is utilised also in the *jackknife linearisation method*, where the jackknife variance estimator based on resampling is converted into a simpler form by using mathematical derivations in the weight structure. See Section 3.8 for further details.

In addition, there are some other analytic methods (applied often in confidence interval estimation as well), e.g. the *Woodruff method* for quantiles (Woodruff 1952, see discussion and further developments e.g. in Maritz and Jarrett 1978, Gross 1980, Rao et al. 1990, and Francisco and Fuller 1991). The method, based on the functions of the cumulative distribution function of a variable, usually appears in the context of median estimation. Briefly, the median of the variable y is defined as $M = F^{-1}(0.5)$, where the cumulative distribution function of the population U is

$$F(y) = \frac{1}{N} \sum_{k \in U} z_{k,y}, \qquad (3.3.3.)$$

where $z_{k,y} = I(y_k \leq y)$ is an indicator function with respect to the condition $y_k \leq y$. The estimator of this function is

$$\hat{F}(y) = \sum_{s} \frac{z_{k,y}}{\pi_k} \Big/ \sum_{s} \frac{1}{\pi_k}. \qquad (3.3.4.)$$

Now, let us consider the estimator of the median

$$\hat{M} = \hat{F}^{-1}(0.5)\,, \tag{3.3.5.}$$

where $\hat{F}^{-1}$ is an inverse function of (3.3.4). For large samples it is justified to assume that $\hat{F}(M) \cong 0.5$. For the π estimator we have

$$V(\hat{F}(M)) \cong N^{-2} \sum_{k} \sum_{l} (\pi_{kl} - \pi_k \pi_l) \left(\frac{z_{k,M} - 0.5}{\pi_k} \right) \left(\frac{z_{l,M} - 0.5}{\pi_l} \right) \tag{3.3.6.}$$

and its estimator is

$$\hat{V}(\hat{F}(M)) = \hat{N}^{-2} \sum_{k} \sum_{l} \frac{\pi_{kl} - \pi_k \pi_l}{\pi_{kl}} \left(\frac{z_{k,\hat{M}} - 0.5}{\pi_k} \right) \left(\frac{z_{l,\hat{M}} - 0.5}{\pi_l} \right). \tag{3.3.7.}$$

Now it is easy to put down confidence intervals for $F(M)$, and by inverting this, we get the confidence interval for the median M:

$$\left[\hat{F}^{-1}\left(0.5 - t_{0.025}(d) \sqrt{\hat{V}(F(M))} \right),\ \hat{F}^{-1}\left(0.5 + t_{0.025}(d) \sqrt{\hat{V}(F(M))} \right) \right] \tag{3.3.8.}$$

from which we calculate the standard error of the estimator of the median by using

$$\sqrt{\hat{V}(\hat{M})} = \frac{\left[\hat{F}^{-1}\left(0.5 + t_{0.025}(d) \sqrt{\hat{V}(F(M))} \right) - \hat{F}^{-1}\left(0.5 - t_{0.025}(d) \sqrt{\hat{V}(F(M))} \right) \right]}{2 t_{0.025}(d)} \tag{3.3.9.}$$

for the 95 % level of confidence and d degrees of freedom for the t-value of the confidence interval. Note that the interval (3.3.8) is usually asymmetric around the median, and therefore (3.3.9) is an approximation. This calculation technique of the standard error is mentioned e.g. in Rao and Wu (1988, for the 95 % interval) and Francisco and Fuller (1991). Justly one can argue that calculating a standard error based on the method developed for (asymmetric) estimation of the confidence interval undermines the original idea: dividing the difference by 2 somewhat simplifies the situation. However, there is a need for a standard error estimator outside the calculation of the confidence interval, e.g. the coefficient of variation in percentages.

Conditional variance of the resample estimator. A special feature of the estimation problem in sampling theory is the possibility to choose the probabilities $p(\underline{k})$ (i.e. the sampling design) and to define the (restricted) sample space based on the units with these positive probabilities. Furthermore, the estimator may be constructed such that it utilises some auxiliary information from the population level. These possibilities apply to the resampling situation as well.

Let us assume that the resampling design – the (restricted) resample space conditioned by the sample – and the resample estimator $\hat{\theta}_b$ are chosen. Then for any given sample $\underline{k}_a$ the $\hat{\theta}_b$ can be evaluated in the whole resample space. A variance estimator may then be the *conditional variance of the resample estimator* $\hat{\theta}_b$ *given* $\underline{k}_a$, i.e.

$$V(\hat{\theta}_b \mid \underline{k}_a) = \sum_{\underline{k}_b \mid \underline{k}_a} p(\underline{k}_b \mid \underline{k}_a)(\hat{\theta}_{\underline{k}_b} - E(\hat{\theta}_b \mid \underline{k}_a))^2 , \qquad (3.4.1.)$$

where $\hat{\theta}_{\underline{k}_b}$ is the estimator $\hat{\theta}_b$ evaluated on the resample $\underline{k}_b$, $\underline{k}_b \mid \underline{k}_a$ is a possible resample which can be taken from $\underline{k}_a$, summation takes place over all such resamples, $p(\underline{k}_b \mid \underline{k}_a)$ is a resampling design, and

$$E(\hat{\theta}_b \mid \underline{k}_a) = \sum_{\underline{k}_b \mid \underline{k}_a} p(\underline{k}_b \mid \underline{k}_a)\hat{\theta}_{\underline{k}_b} \qquad (3.4.2.)$$

is the *conditional expectation of the estimator* $\hat{\theta}_b$ *given* $\underline{k}_a$.

The *expected conditional variance* is needed for studies of the variance estimators, and its form is

$$E_a[V(\hat{\theta}_b \mid \underline{I}_a)] = \sum_{\underline{k}_a} p(\underline{k}_a) \sum_{\underline{k}_b \mid \underline{k}_a} p(\underline{k}_b \mid \underline{k}_a)(\hat{\theta}_{\underline{k}_b} - E(\hat{\theta}_b \mid \underline{k}_a))^2. \qquad (3.4.3.)$$

In some cases the *conditional mean square error of the estimator* given $\underline{k}_a$,

$$MSE(\hat{\theta}_b \mid \underline{k}_a) = \sum_{\underline{k}_b \mid \underline{k}_a} p(\underline{k}_b \mid \underline{k}_a)(\hat{\theta}_{\underline{k}_b} - \hat{\theta}_a)^2 , \qquad (3.4.4.)$$

is used for variance estimation purposes.

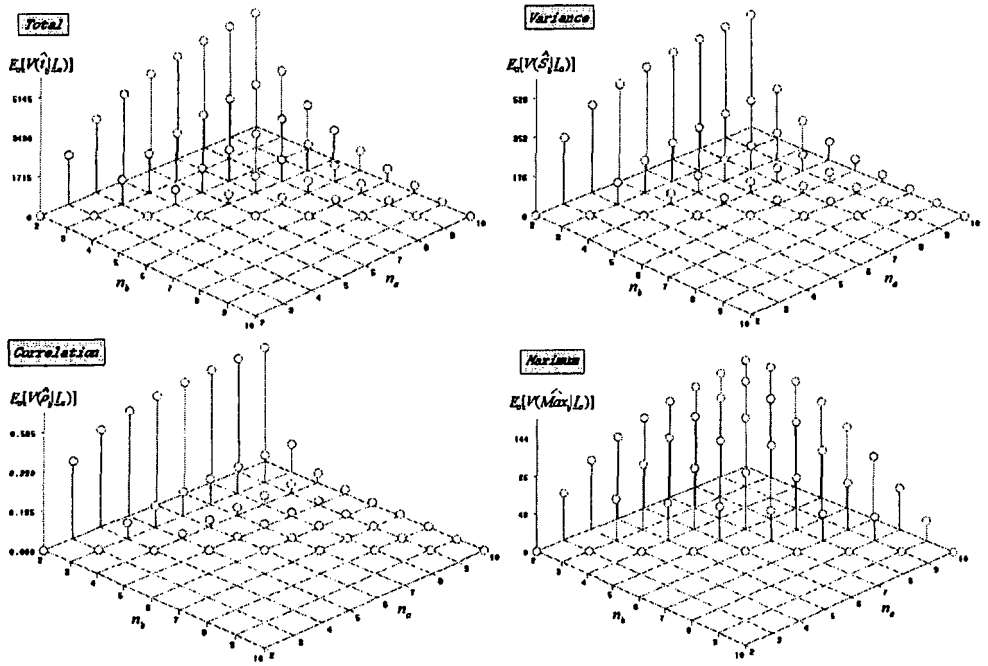
Basic questions arise:

1. What is the quality of the conditional variance as an estimator?
2. How can it be corrected or improved?
3. What are the right choices for the resampling design and the resample estimator?

No general answers can be found from the sampling literature for all parameters and estimators and sampling designs at the same time.

Graphical study of conditional variances. The following figures illustrate the issue of how the expected conditional variance approximates the variance of different estimators. The estimators of four population parameters (total, correlation coefficient, variance, maximum) are considered. The population size is $N = 10$. The data in this section are $\underline{y}$ = (15, 19, 10, 7, 11, 28, 18, 16, 9, 45) and $\underline{x}$ = (9, 6, 3, 4, 5, 13, 10, 11, 7, 17). The sampling design is SI with size n_a and the resampling design is SI with size n_b. The expected conditional variance is calculated for every possible pair (n_a, n_b) and displayed as a vertical line on the figures. The case when $n_a = N$ displays the variance of the estimator. This quantity is the subject of the estimation. The graphical software used here (SAS®) constructs the scale of the figure by using the maximal value in the data; thus a shape comparison is possible here. Since conditional variance works well (with minor corrections) for variance estimation of linear estimators then the shape of the figures needs to be compared with the one for the total. The figures show the different speeds of the decrease (for fixed n_a the expected conditional variance decreases to zero as n_b approaches n_a).

Figure 3.2. Expected Conditional Variances for Estimators of Four Parameters



Comparing complex situations with the linear case (total) the expected conditional variance of the estimator of the population variance seems to be the nearest in this case. In the case of the correlation we have more decrease when n_b is increasing and less decrease when n_a is decreasing. For the maximum a strange result can be seen: no decrease of the conditional variance when n_b is increasing, e.g. for values n_a from 7 to 10 and n_b from 2 to 4.

Under SI design, the shape of the pattern is the same for every linear estimator (including Taylor's approximation for the function of totals). The expressions of the expected conditional variance depend on $(1/n_b - 1/n_a)$. With well-known SI formulae we have for the mean $E_a[V(\bar{y}_b \mid \underline{I}_a)] = (1/n_b - 1/n_a)S_y^2$, for the total $E_a[V(\hat{t}_b \mid \underline{I}_a)] = N^2(1/n_b - 1/n_a)S_y^2$, and for the Taylor's approximation estimator $E_a[V(\hat{\theta}_{b0} \mid \underline{I}_a)] = (1/n_b - 1/n_a)S_e^2$. The variance of the variable y, i.e. S_y^2 (or of the derived variable, S_e^2), is then a constant and it gives the scale of the pattern. *It must be emphasised here that the same does not hold for nonlinear estimators.* Varying data can cause surprisingly strong effects to the shape of the pattern; the case of the maximum seems to be especially sensitive.

3.5. *Solution 3: Metasample from the Resample Space*

The independent selection of resamples (a special case of *metasampling* in terms of Section 2.4) can be considered as a *bootstrap replication method* in a broad sense. Selecting a metasample os^* means that the original resampling design $p(s_b \mid s_a)$ is used at each draw of a resample s_b. Then, following the principle in (3.1.1), the unbiased estimator of the conditional variance is $\hat{V}(\hat{\theta}_b \mid s_a) = \sum_{s_b \in os^*} \frac{(\hat{\theta}_{s_b} - \bar{\hat{\theta}})^2}{A-1}$, where $\bar{\hat{\theta}} = \sum_{s_b \in os^*} \hat{\theta}_{s_b} / A$ and A is the number of resamples s_b in the metasample os^*. The size A must usually be large for good estimation. The corresponding estimator for the conditional mean square error is

$$\hat{MSE}(\hat{\theta}_b \mid s_a) = \sum_{s_b \in os^*} \frac{(\hat{\theta}_{s_b} - \hat{\theta}_a)^2}{A-1}, \qquad (3.5.1.)$$

where $\hat{\theta}_a$ is the estimate calculated on s_a.

However, we can select some of the resamples from the resample space with some selection conditions as well. The resulting metasample is used for variance estimation purposes, like in the *dependent random group method* explained in more detail in Section 2.4. The corresponding metasampling design can be seen in (2.4.1). The variance estimator is of the form

$$\hat{V}(\hat{\theta}_a) = \sum_{s_i \in os^*} \frac{(\hat{\theta}_{s_i} - \bar{\hat{\theta}})^2}{A(A-1)}, \qquad (3.5.2.)$$

where $\bar{\hat{\theta}} = \sum_{s_i \in os^*} \frac{\hat{\theta}_{s_i}}{A}$. Note that this variance estimator *does not* coincide with the corresponding independent random group estimator in (3.1.1), as here the expression has an additional A in the denominator. Again the alternative is a more conservative variance estimator, obtained when replacing $\bar{\hat{\theta}}$ by $\hat{\theta}_a$ above.

Norlén and Waller (1979) provide an extension with several random groupings from which the variance estimates are obtained. With repetitions it makes the variance estimator more stable. Then the variance estimator is an average

$$\hat{V}(\hat{\theta}_a) = \sum_{r=1}^{R} \frac{\hat{V}_r}{R}, \qquad (3.5.3.)$$

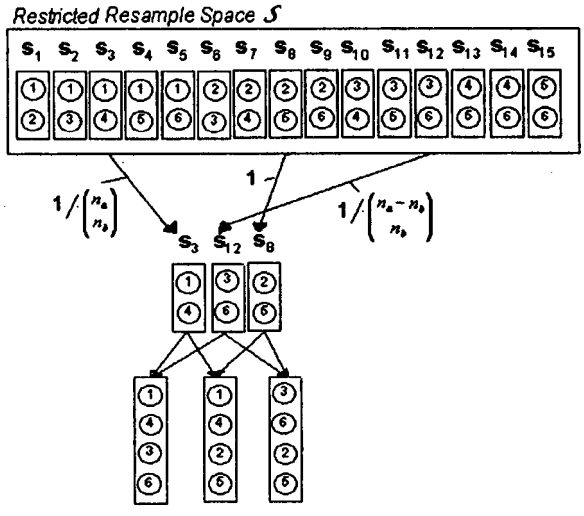
where $\hat{V}_r$ is the variance estimator from the rth random grouping, and R is the number of random groupings. This is an estimator of the *expectation over metasamples*

$$E^*[\hat{V}(\hat{\theta}_a) \mid s_a] = \sum_{os^* \in S^* \mid s_a} p(os^* \mid s_a) \hat{V}_{os^*}(\hat{\theta}_a), \qquad (3.5.4.)$$

where $\mathbf{S}^* \mid s_a$ is the set of different metasamples conditioned by the sample s_a and $\hat{V}_{os^*}(\hat{\theta}_a)$ is the variance estimator based on sample os^*. The expectation over metasamples is used in simulations in Chapter 8.

The *jackknife method* requires the random groups method as a basis, except in the $n-1$ situation. The random groups method uses the estimates from each group, but the jackknife method goes a step further: it combines the information of $A-1$ random groups creating a set of the observations of these random groups, and this procedure is conducted for each combination of $A-1$ random groups. In Ollila (1996) this set belongs to a more general definition of a *combined sample*. When $An_b = n_a$, it is evident that these combined samples, s_{comb}, of size $(A-1)n_b$ are outcomes from the resample space $\mathbf{S}_b \mid s_a$ of resample size $(A-1)n_b$. An example of the process of combining the random groups can be seen in Figure 3.3.

Figure 3.3. Selecting a Metasample with Disjoint Units from the Space of Set-type Resamples with Combined Samples for the Jackknife



The metasampling design based on the general jackknife method with equal jackknife resample sizes n_b is (Ollila 1996)

$$p(os^* \mid s_a) = \frac{1}{\prod_{g=1}^{A} \binom{n_a - (g-1)n_{rg}}{n_{rg}}}, \qquad (3.5.5.)$$

where the random group subsample size $n_{rg} = n_a - n_b$, i.e. it equals the design of the random groups method in (2.4.1). This is justified as we consider one jackknife resample as a remainder of the sample when the chosen random group is selected. It is well known that the general jackknife requires the use of random groups in order to be conducted properly. From the combined resample s_{comb} we get $\hat{\theta}_{(-s_b)}$ ("$-s_b$" referring to the exclusion of the resample s_b) and the variance estimator can be formed, i.e. here the second form appearing in Wolter (1985, page 166)

$$\hat{V}(\hat{\theta}_a) = \frac{A-1}{A} \sum_{s_b \in os^*} (\hat{\theta}_{(-s_b)} - \bar{\hat{\theta}}_{comb})^2, \qquad (3.5.6.)$$

where $\bar{\hat{\theta}}_{comb} = \frac{1}{A} \sum_{s_b \in os^*} \hat{\theta}_{(-s_b)}$.

The *balanced half-samples method* (McCarthy 1969) is not presented here in any more detail. It is based on the utilisation of matrices with some balance properties and it has been quite popular in some extreme sampling situations with extensive stratification and two units (usually clusters) selected in every stratum.

3.6. *Linear Case Coefficient*

Almost every resampling method presented in the sampling literature requires for the linear estimator that the resampling variance estimator $\hat{V}_{res}(\bar{y}_a)$ and the analytical variance estimator based on one sample, $\hat{V}(\bar{y}_a)$, be equal; that is, the *linear case condition* is

$$\hat{V}_{res}(\bar{y}_a) = \hat{V}(\bar{y}_a). \qquad (3.6.1.)$$

This condition for the linear case is constructed into the resampling strategy in different ways in the existing resampling methods. We can *adjust the variables* that are included in the estimator (e.g. Rao and Wu's variable rescaling bootstrap 1988). The *survey weights* (i.e. w_i defined for $i = 1, \ldots, N$ according to the first-phase design) *can be altered* for the same reason (e.g. weight rescaling bootstrap by Rao et al. 1992). There can be a *scaling coefficient* for the correction of the conditional variance (e.g. Wolter 1985, McCarthy and Snowden 1985). The *resampling design can be chosen in such a way that no rescaling is needed:* by using *usual resampling designs* (e.g. McCarthy and Snowden's bootstrap with replacement 1985), by using *"combined" resamples* (e.g. combining m independent without-replacement resamples, Sitter 1992b) or by using a *"pseudopopulation"* as a basis for the resampling (Gross 1980, Bickel and Freedman 1984, Chao and Lo 1985, Sitter 1992a, Booth et al. 1994 bootstrap without replacement). It is quite a common practice to ignore the correction or some part of it (e.g. the finite population correction part), mainly due to large samples and resamples.

A scaling coefficient is usually used for the correction of the conditional variance,

$$\hat{Q}_{lin} = \frac{\hat{V}(\bar{y}_a)}{V(\bar{y}_b \mid \underline{k}_a)} \qquad (3.6.2.)$$

which is a constant in the linear case and does not depend on the variable y. This coefficient is also called a *linear case coefficient*. Let us study different sampling and resampling designs.

Table 3.1. Variances, Variance estimators and Conditional Variances in the Linear Case with Some Designs

Variance $V(\bar{y}_a)$	**Variance estimator** $\hat{V}(\bar{y}_a)$	**Conditional variance** $V(\bar{y}_b \mid \underline{k}_a)$
SI sampling $\frac{(N-n_a)}{Nn_a}\frac{\sum_{i\in U}(y_i-\bar{y}_U)^2}{N-1}$	SI sampling $\frac{(N-n_a)}{Nn_a}\frac{\sum_{i\in s_a}(y_i-\bar{y}_{s_a})^2}{n_a-1}$	SI resampling $\frac{(n_a-n_b)}{n_an_b}\frac{\sum_{i\in s_a}(y_i-\bar{y}_{s_a})^2}{n_a-1}$
SIR sampling $\frac{\sum_{i\in U}(y_i-\bar{y}_U)^2}{Nn_a}$	SIR sampling $\frac{\sum_{i\in s_a}(y_i-\bar{y}_{s_a})^2}{n_a(n_a-1)}$	SIR resampling $\frac{\sum_{i\in s_a}(y_i-\bar{y}_{s_a})^2}{n_an_b} =$ $\frac{(n_a-1)}{n_an_b}\frac{\sum_{i\in s_a}(y_i-\bar{y}_{s_a})^2}{n_a-1}$
Multinomial sampling $M(n; p_1, \ldots, p_N)$ $\frac{1}{N^2n_a}\sum_{i\in U}p_i\left(\frac{y_i}{p_i}-t_U\right)^2$	Multinomial sampling $M(n; p_1, \ldots, p_N)$ $\frac{1}{N^2n_a(n_a-1)}\sum_{i\in s_a}\left(\frac{y_i}{p_i}-\hat{t}\right)^2$ where $\hat{t}=\sum_{i\in s_a}\frac{y_i}{n_ap_i}$	SI resampling, variable $y_i/(Np_i)$ $\frac{(n_a-n_b)}{N^2n_an_b}\frac{\sum_{i\in s_a}\left(\frac{y_i}{p_i}-\hat{t}\right)^2}{n_a-1}$ SIR resampling, variable $y_i/(Np_i)$ $\frac{1}{N^2n_an_b}\frac{\sum_{i\in s_a}\left(\frac{y_i}{p_i}-\hat{t}\right)^2}{n_a-1}$

The variable $y_i/(Np_i)$ is needed here in order to make $\hat{Q}_{lin}$ independent from p_i and y_i. The coefficients $\hat{Q}_{lin}=\frac{\hat{V}(\bar{y}_a)}{V(\bar{y}_b \mid \underline{k}_a)}$ for different combinations of sampling and resampling designs are as follows.

Table 3.2. Linear Case Coefficients $\hat{Q}_{lin}$ for Some Design Combinations

SI / SI	$\frac{(N-n_a)n_b}{N(n_a-n_b)}$
SI / SIR	$\frac{(N-n_a)n_b}{N(n_a-1)}$
Multinomial / SI	$\frac{n_b}{n_a-n_b}$
Multinomial / SIR	$\frac{n_b}{n_a-1}$

We recall that SIR is a special case of multinomial.

In many cases (e.g. combinations with SI and SIR sampling in the first and second phases) we get

$$\hat{Q}_{lin} = \frac{V(\bar{y}_a)}{E_a[V(\bar{y}_b \mid I_a)]}, \tag{3.6.3.}$$

i.e. in the denominator is the expected conditional variance (expectation is taken over the first phase design). This is justified by using formulae in Table 3.1 for SI and SIR cases and by having $E_{SI}\left[\frac{\sum_{i\in s_a}(y_i-\bar{y}_{s_a})^2}{n_a-1}\right]=\frac{\sum_{i\in U}(y_i-\bar{y}_U)^2}{N-1}$ and $E_{SIR}\left[\frac{\sum_{i\in s_a}(y_i-\bar{y}_{s_a})^2}{n_a-1}\right]=\frac{\sum_{i\in U}(y_i-\bar{y}_U)^2}{N}$. The coefficient $\hat{Q}_{lin}$ is the basis for many estimation problems with resampling methods. Methods avoiding a scaling coefficient in the variance estimator use $\hat{Q}_{lin}$ in other resampling design adjustments. Thus the correction methods given in Section 3.7 are all based on the condition $\hat{V}_{res}(\bar{y}_a)=\hat{V}(\bar{y}_a)$, i.e. the resampling variance estimator equals the analytical design-based variance estimator. However, as seen in Figure 3.2, the conditional variance of a nonlinear estimator *may not behave in a similar manner* as the conditional variance of the linear estimator. This phenomenon occurs especially strongly in small populations and samples. In practice we encounter this situation when there are small strata in the sampling design. With larger populations and sample sizes the importance of scale and design corrections diminishes, and then the main issue is the resampling design (in practice the jackknife or the bootstrap) and its effect on variance estimation.

Whatever the correction method is, by using the linear case condition *we ignore the estimator-specific nonlinearity part in the corrections*. The bias caused by this can be considerable with some variance estimators in small populations, as seen in simulations in Chapter 8. One of the key ideas of this thesis is to *provide alternative methods for more detailed correction in the variance estimators*.

A computational example about the linear case condition. Next we study how the linear case condition suits variance estimation of different estimators $\hat{\theta}_a$ with a specific sample size. As the estimator $\hat{\theta}_a$ we consider estimators for *total, ratio, correlation, variance, median* and *maximum*.

The unpredictable behaviour of the conditional variance in small populations is one of the main criticisms against the linear case correction principle. Let us assume that the variance we want to estimate is for the estimator under *SI sampling* with size 5, i.e. $V(\hat{\theta}_{a,SI})$, for the population of size N = 10 (same data as before). Two resampling designs, SI and SIR, are considered. Figure 3.4 shows the expected conditional variances for different estimators. The true $V(\hat{\theta}_{a,SI})$ can be found on SI-graphs at n_a = 10, n_b = 5. The black, larger dots show the *sample size* n_b which should be selected from the first-phase sample of size n_a in order to get an unbiased estimator of the variance, i.e. $E_a[V(\hat{\theta}_b \mid \underline{I}_a)] = V(\hat{\theta}_{a,SI})$ without any scaling.

For the total we calculate the appropriate resample size n_b from the condition $\hat{V}(\bar{y}_a) = V(\bar{y}_b \mid \underline{k}_a)$. For SI resampling

$$\left(\frac{N-n_a}{Nn_a}\right)\frac{\sum_{i\in s_a}(y_i-\bar{y}_{s_a})^2}{n_a-1} = \left(\frac{n_a-n_b}{n_a n_b}\right)\frac{\sum_{i\in s_a}(y_i-\bar{y}_{s_a})^2}{n_a-1}$$

gives the solution

$$n_b = \frac{Nn_a}{2N-n_a},$$

and for SIR resampling solving

$$\left(\frac{N-n_a}{Nn_a}\right)\frac{\sum_{i\in s_a}(y_i-\bar{y}_{s_a})^2}{n_a-1} = \frac{\sum_{i\in s_a}(y_i-\bar{y}_{s_a})^2}{n_a n_b}$$

gives

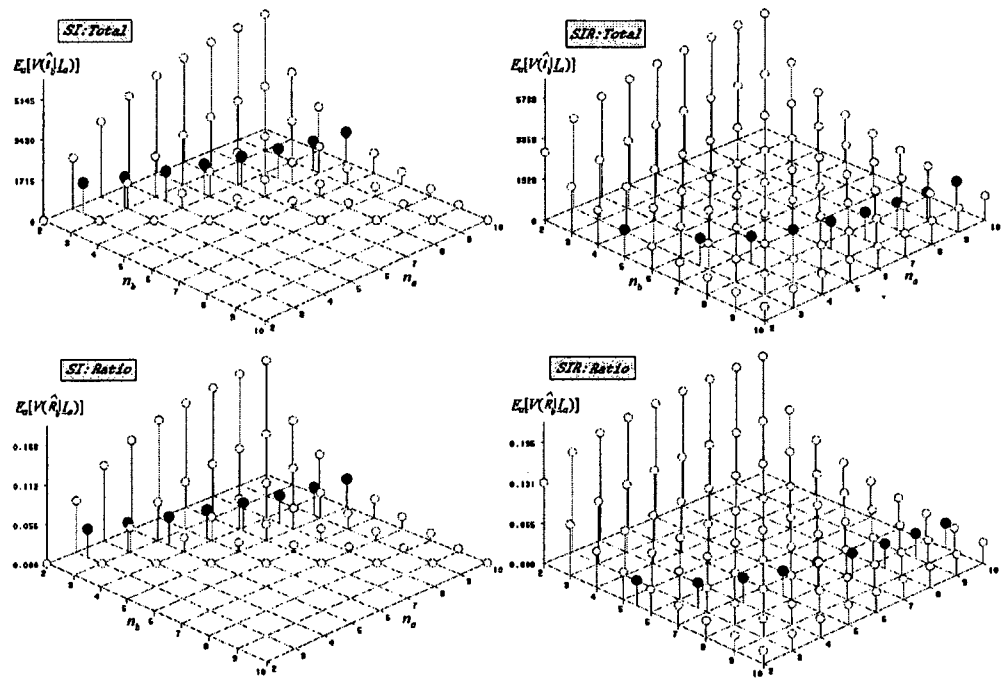
$$n_b = \frac{N(n_a-1)}{N-n_a}.$$

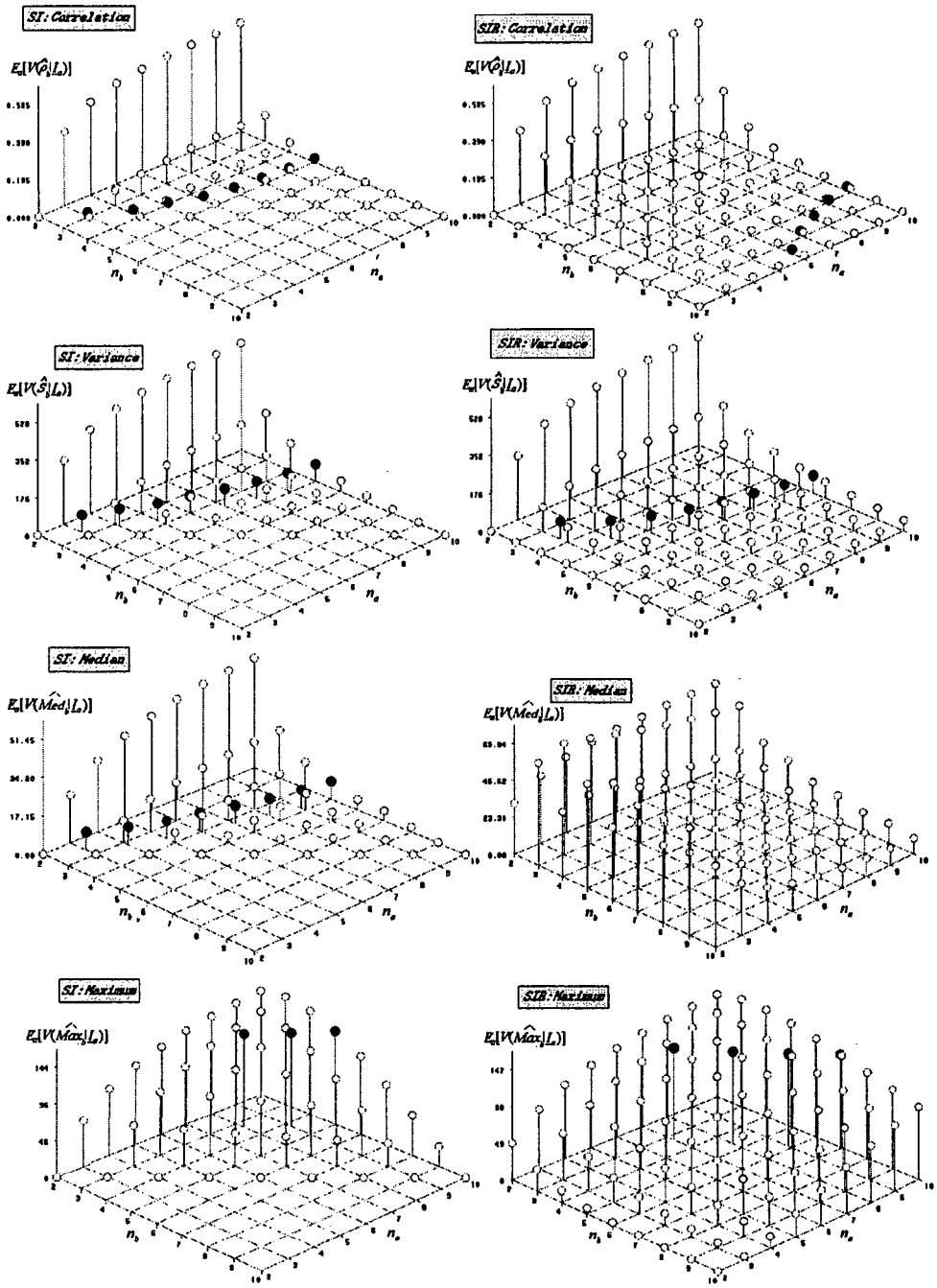
For five other estimators with SI and SIR resampling designs the sample size n_b has been calculated by

$$n_b = n_{b1} + \frac{E_a[V(\hat{\theta}_{b1} \mid \underline{I}_a)] - V(\hat{\theta}_{a,SI})}{E_a[V(\hat{\theta}_{b1} \mid \underline{I}_a)] - E_a[V(\hat{\theta}_{b2} \mid \underline{I}_a)]},$$

where $\hat{\theta}_{b1}$ is the estimator with the resample size n_{b1} and $\hat{\theta}_{b2}$ is the estimator with the resample size $n_{b2} = n_{b1} + 1$. Here we assume that $E_a[V(\hat{\theta}_{b1} \mid \underline{I}_a)] > V(\hat{\theta}_{a,SI}) > E_a[V(\hat{\theta}_{b2} \mid \underline{I}_a)]$ or in rare cases $E_a[V(\hat{\theta}_{b2} \mid \underline{I}_a)] > V(\hat{\theta}_{a,SI}) > E_a[V(\hat{\theta}_{b1} \mid \underline{I}_a)]$. Note that the non-integer resample size n_b appearing in these calculations and in Figures 3.4 and 3.5 is the result of the requirement for $E_a[V(\hat{\theta}_b \mid \underline{I}_a)] = V(\hat{\theta}_{a,SI})$. In practice only integers n_{b1} and n_{b2} can be used. When comparing the black dot line between the total and the other estimators $\hat{\theta}_a$ under SI and SIR we see how much the resample size provided by the assumption $\hat{V}(\bar{y}_a) = V(\bar{y}_b \mid \underline{k}_a)$ differs from the real resample size needed to estimate $V(\hat{\theta}_{a,SI})$. When studying Figure 3.4 it is evident that in SI sampling n_b does not differ very much from one estimator to another (except the correlation and the maximum), but in SIR sampling the with-replacement nature of the process together with small sample sizes causes more oddities to the expected conditional variances (e.g. the correlation, the median and the maximum).

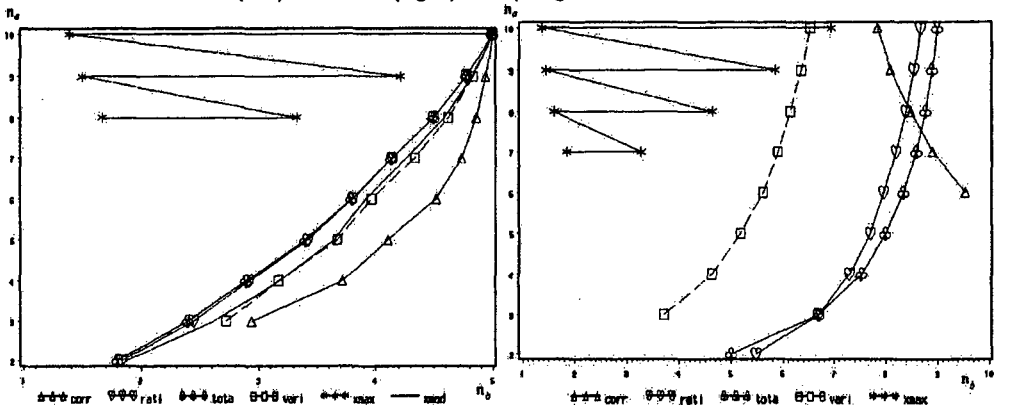
Figure 3.4. Expectations of the Conditional Variances for Different n_a and n_b




SI: Correlation
SIR: Correlation
SI: Variance
SIR: Variance
SI: Median
SIR: Median
SI: Maximum
SIR: Maximum

The axis for n_b is cut at ten, the size of the population (but naturally it is possible to select with-replacement samples greater than ten). *In practice only one first-phase sample is given*, e.g. of size $n_a = 5$. Then it is clear that *only those conditional variances* which are based on that sample *are available for variance estimation.* Here we look through all n_a, n_b combinations since it is important to see the different shapes of the surfaces for different estimators and different resampling designs. Let us study the resample sizes needed for unbiased variance estimation (without scaling) in more detail.

Figure 3.5. Resample Sizes Needed for Unbiased Variance Estimation for Different Estimators under SI (left) and SIR (right) sampling



In Figure 3.5 we see that in SI resampling the necessary n_b are closest to each other for the total and the ratio. The n_b differs most strikingly with the maximum, having in fact two values to yield $E_a[V(\hat{\theta}_b \mid \underline{I}_a)] = V(\hat{\theta}_{a,SI})$. Furthermore, the curve for correlation has less change in n_b than the curves of the four other estimators.

When estimating the variance of the SI estimator with SIR resampling, we see that generally there must be more than 5 units to be selected in order to achieve the same level of variance. The actual variances in this case are revealed in Figure 3.4 on SI-pictures in the case where n_a = 10 in Figure 3.4. Note that the SIR line with n_a = 10 does not coincide with the corresponding SI line with n_a = 10. In Figures 3.4 and 3.5 for the SIR case, if we exclude the ratio and the total, we see that the shape of the curve differs from one estimator to another rather clearly. The correlation coefficient has a curve with a different direction than the others (with n_a increasing n_b decreases) except the maximum with its two-solution graph. The median is a problematic estimator due to the different practice in odd and even cases, e.g. in Figure 3.4 in SIR we have for some $n_{b1} < n_{b2}$ the unusual result $E_a[V(\hat{\theta}_{b1} \mid \underline{I}_a)] < E_a[V(\hat{\theta}_{b2} \mid \underline{I}_a)]$.

External scaling. If in these variance estimations the scaling process is conducted outside the conditional variance $V(\hat{\theta}_b \mid \underline{k}_a)$, we call this process *external scaling*. This practice appears in some contexts (e.g. the jackknife, see Wolter 1985, the bootstrap e.g. McCarthy and Snowden 1985). The variance estimator is then

$$\hat{V}(\hat{\theta}_a) = \hat{Q}_{lin} V(\hat{\theta}_b \mid \underline{k}_a) . \qquad (3.7.1.)$$

From the formula it is obvious that the linear case condition (formula 3.6.1) holds. In many cases the set of all resamples is far too large for calculation of the conditional variance $V(\hat{\theta}_b \mid \underline{k}_a)$ and so its estimator is needed, that being $\hat{V}(\hat{\theta}_b \mid \underline{k}_a) = \sum_{s_b \in os^*} \frac{(\hat{\theta}_{s_b} - \bar{\hat{\theta}})^2}{A-1}$, where $\bar{\hat{\theta}} = \sum_{s_b \in os^*} \frac{\hat{\theta}_{s_b}}{A}$.

Internal scaling. Another possibility is to *adjust variables* on which the parameter θ is based (Rao and Wu, 1988, for the parameters of the form $\theta = f(t_1, \ldots, t_q)$, which are sometimes called "smooth parameters") or to adjust survey weights (i.e. first-phase sampling design weights) connected to these variables (Rao et al. 1992, applicable for non-smooth parameters as well). This process is called here *internal scaling*, because these scale-corrected variables or weights are used in the estimator, denoted by $\tilde{\theta}$. Scaling of variables also appears in other methods, including the dependent random groups method and the jackknife, when the finite population correction is taken into account (see Wolter 1985). For SI sampling this correction is $\left(\frac{1}{n_a} - \frac{1}{N}\right)$.

In the method of Rao and Wu (1988) every variable is rescaled in the following way. For example, the variable y is rescaled with

$$\tilde{y}_i = \sqrt{\hat{Q}_{lin}}\, y_i + (1 - \sqrt{\hat{Q}_{lin}})\bar{y}_a . \qquad (3.7.2.)$$

Correspondingly, the estimator of the mean in the second phase of SI or SIR sampling is

$$\tilde{y}_b = \sqrt{\hat{Q}_{lin}}\, \bar{y}_b + (1 - \sqrt{\hat{Q}_{lin}})\bar{y}_a . \qquad (3.7.3.)$$

All the variables needed for the estimation of the parameter are rescaled, i.e. for example for the ratio we use the estimator $\hat{R}_{y,x} = \hat{t}_{\tilde{y},b} / \hat{t}_{\tilde{x},b}$. The estimator of $V(\hat{\theta}_a)$ is then the conditional variance

$$\hat{V}(\hat{\theta}_a) = V(\tilde{\theta}_b \mid \underline{k}_a) \tag{3.7.4.}$$

or its estimator with independent resamples $\hat{V}(\tilde{\theta}_b \mid \underline{k}_a) = \sum_{s_b \in os^*} \frac{(\tilde{\theta}_{s_b} - \bar{\tilde{\theta}})^2}{A-1}$, where $\bar{\tilde{\theta}} = \sum_{s_b \in os^*} \frac{\tilde{\theta}_{s_b}}{A}$.

The weight rescaling method based on Rao et al. (1992) transforms the survey weights (i.e. the first-phase design weights) as follows:

$$\tilde{w}_{ai} = (1 - \sqrt{\hat{Q}_{lin}}) w_{ai} + \sqrt{\hat{Q}_{lin}} k_{bi} w_{bi}, \tag{3.7.5.}$$

where w_{ai} is the first-phase design weight for the total, w_{bi} is the resampling weight of the total and k_{bi} is the indicator, i.e. how many times the unit i appears in the resample. This weight is defined for every unit in the sample because it requires the contribution of the first-phase weighting (i.e. w_{ai}) for rescaling.

It can be seen that (3.7.3) can be converted into a weighted form with weights (3.7.5). Expressing the means $\bar{y}_b$ and $\bar{y}_a$ by their weighted forms we have

$$\begin{aligned}
\tilde{y}_b &= \sqrt{\hat{Q}_{lin}} \bar{y}_b + (1 - \sqrt{\hat{Q}_{lin}}) \bar{y}_a \\
&= \sum_{i=1}^{N} \left[k_{bi} \sqrt{\hat{Q}_{lin}} w_{bi} y_i + (1 - \sqrt{\hat{Q}_{lin}}) w_{ai} y_i \right] / N \\
&= \sum_{i=1}^{N} \tilde{w}_{ai} y_i / N.
\end{aligned}$$

The weight rescaling method can also be applied for quantiles, even when we have a complex design.

In Ollila (1996) internal scaling was studied for some nonlinear estimators. For example, for the estimator of the variance S_y^2 the rescaled estimator (in terms of variable rescaling) is

$$\tilde{S}_{y,b}^2 = \frac{1}{n_b - 1}\sum_{i=1}^{n_b}(\tilde{y}_i - \tilde{y}_b)^2 = \frac{1}{n_b - 1}\sum_{i=1}^{n_b}(\sqrt{\hat{Q}_{lin}}\, y_i - \sqrt{\hat{Q}_{lin}}\,\bar{y}_b)^2$$
$$= \hat{Q}_{lin}\hat{S}_{y,b}^2 ,$$

where $\hat{S}_{y,b}^2 = \frac{1}{n_b - 1}\sum_{i=1}^{n_b}(y_i - \bar{y}_b)^2$. Its conditional variance is

$$V(\tilde{S}_{y,b}^2 \mid \underline{k}_a) = \hat{Q}_{lin}^2 V(\hat{S}_{y,b}^2 \mid \underline{k}_a), \qquad (3.7.6.)$$

which is of a different order than the result obtained with the external scaling method, i.e. $\hat{Q}_{lin} V(\hat{\theta}_b \mid \underline{k}_a)$.

On the other hand, for the correlation coefficient we have

$$\tilde{\rho}_{yx,b} = \frac{\tilde{S}_{yx,b}}{\tilde{S}_{y,b}\tilde{S}_{x,b}} = \frac{\hat{Q}_{lin}\hat{S}_{yx,b}}{\sqrt{\hat{Q}_{lin}}\hat{S}_{y,b}\sqrt{\hat{Q}_{lin}}\hat{S}_{x,b}} = \frac{\hat{S}_{yx,b}}{\hat{S}_{y,b}\hat{S}_{x,b}} = \hat{\rho}_{yx,b}, \qquad (3.7.7.)$$

i.e. there is no scaling effect in this case.

Resampling design adjustments and randomisation. One purpose of some variance estimation methods is to avoid any scaling terms, i.e. to base the calculations purely on the resamples obtained in the reuse process. The resampling design is adjusted according to the first-phase sampling design by using some criterion. These kinds of resampling methods are strict resampling (i.e. using no adjustments in the resampling design), resample combinations, or creation of an artificial population. Their specific feature is that either the resample size, or the repetition number (m) for artificial samples or populations, or both are adjusted in order to fulfil the criterion. In practice these numbers are found from the linear case condition,

$$V(\bar{y}_b \mid \underline{k}_a) = \hat{V}(\bar{y}_a),$$

(see Section 3.6).

Thus, using those numbers the variance estimator of any estimator $\hat{\theta}_a$ is simply the conditional variance

$$\hat{V}(\hat{\theta}_a) = V(\hat{\theta}_b \mid \underline{k}_a) \tag{3.7.8.}$$

or its estimator $\hat{V}(\hat{\theta}_b \mid \underline{k}_a) = \sum_{s_b \in os^*} \frac{(\hat{\theta}_{s_b} - \bar{\hat{\theta}})^2}{A-1}$, where $\bar{\hat{\theta}} = \sum_{\underline{s}_b \in os^*} \hat{\theta}_{s_b} / A$.

In Section 2.3 some resampling designs were described: strict resampling with or without replacement, resampling from the pseudopopulation, and making combined resamples from some small resamples. Furthermore, in Sitter's *Mirror-Match Method* (1992a, 1992b) the resampling design should resemble the original design as much as possible. Then the resampling fraction must also be the same as in the original sampling phase, i.e. $n_b / n_a = n_a / N$.

However, the numbers n_b or m found from $V(\bar{y}_b \mid \underline{k}_a) = \hat{V}(\bar{y}_a)$ are usually non-integers. In order to resolve that problem the *randomisation method* was introduced (see Bickel and Freedman 1984, Sitter 1992a and 1992b).

Let us write

$$E_a[V(\bar{y}_b \mid \underline{I}_a)] = G_p f(y_U), \tag{3.7.9.}$$

where the function of the variable y in the population $f(y_U)$ (in practice the variance) is a constant for all n_b, and the term G_p includes the size or the repetition measures or both from the fixed design p.

Table 3.3. Some Examples of G_p and $f(y_U)$ in Different Situations

Resampling design	$V(\bar{y}_b \mid \underline{k}_a)$	$E_a[V(\bar{y}_b \mid \underline{I}_a)] = G_p f(y_U)$	
		G_p	$f(y_U)$ (expectation over 1st phase design)
SI	$\frac{(n_a - n_b)}{n_a n_b} \frac{\sum_{i \in s_a} (y_i - \bar{y}_a)^2}{n_a - 1}$	$\frac{(n_a - n_b)}{n_a n_b}$	$f_{SIR}(y_U) = \frac{\sum_{i \in U} (y_i - \bar{y}_U)^2}{N}$ $f_{SI}(y_U) = \frac{\sum_{i \in U} (y_i - \bar{y}_U)^2}{N-1}$
SIR	$\frac{(n_a - 1)}{n_a n_b} \frac{\sum_{i \in s_a} (y_i - \bar{y}_a)^2}{n_a - 1}$	$\frac{(n_a - 1)}{n_a n_b}$	
SI with n_b repeated m times independently, combining units (Sitter 1992b)	$\frac{(n_a - n_b)}{n_a m n_b} \frac{\sum_{i \in s_a} (y_i - \bar{y}_a)^2}{n_a - 1}$	$\frac{(n_a - n_b)}{n_a m n_b}$	
SI, variable y_i /(Np_i)	$\frac{(n_a - n_b)}{N^2 n_a n_b} \frac{\sum_{i \in s_a} \left(\frac{y_i}{p_i} - \hat{t} \right)^2}{n_a - 1}$	$\frac{(n_a - n_b)}{N^2 n_a n_b}$	$f_{MN}(y_U) = \sum_{i \in U} p_i \left(\frac{y_i}{p_i} - t_U \right)^2$
SIR, variable y_i /(Np_i)	$\frac{1}{N^2 n_a n_b} \frac{\sum_{i \in s_a} \left(\frac{y_i}{p_i} - \hat{t} \right)^2}{n_a - 1}$	$\frac{1}{N^2 n_a n_b}$	

Here $\hat{t} = \sum_{i \in s_a} \frac{y_i}{n_a p_i}$ and MN refers to the general multinomial sampling design. Let us define p_{exact} as the resampling design including size or repetition measures or both (which are non-integers), p_u as the resampling design with the rounded-up integer value(s) of these measures, and p_l is the resampling design with the rounded down integer value(s). The aim of the randomisation is to reach

$$E_a[P(p_l)V(\bar{y}_{b,l} \mid \underline{I}_a) + (1 - P(p_l))V(\bar{y}_{b,u} \mid \underline{I}_a)] = E_a[V(\bar{y}_{b,exact} \mid \underline{I}_a)],$$

where $P(p_l)$ and $P(p_u)$ are the mixing probabilities between lower and upper designs, $0 < P(p_l) < 1$, and $P(p_u) = 1 - P(p_l)$. When decomposed with $f(y_U)$ and G_p we get

$$P(p_l)G_l + [1 - P(p_l)]G_u = G_{exact}$$

and solving for $P(p_l)$ yields

$$P(p_l) = \frac{G_{exact} - G_u}{G_l - G_u}. \qquad (3.7.10.)$$

Again, results obtained for linear estimators can be expanded to any estimator. The variance estimator for any estimator $\hat{\theta}_a$ is defined in the form

$$\hat{V}(\hat{\theta}_a) = P(p_l)V(\hat{\theta}_{b,l} \mid \underline{k}_a) + (1 - P(p_l))V(\hat{\theta}_{b,u} \mid \underline{k}_a). \qquad (3.7.11.)$$

An example of this randomisation principle is found in Section 5.4. In addition, Sitter (1992b) presents an alternative for the randomisation which involves using two non-integer measures and conducting the randomisation in two phases.

However, there is a disadvantage in the randomisation method: in some rare cases the mixing probabilities can either exceed 1 or be negative or both (see Sitter 1992b). On the other hand, variable rescaling is not applicable, because then $\hat{Q}_{lin} > 1$. The only way to find an estimate of $\hat{V}(\overline{y}_a)$ is to use the externally scaled variance estimator. McCarthy and Snowden (1985) mentioned this possibility for the bootstrap without replacement method.) In Chapter 5 an alternative method for randomisation (allowing also $\hat{Q}_{lin} > 1$) is presented, i.e. the post-design vector method.

3.8. *Linearising the Variance Estimator*

In Yung and Rao (1996) the *jackknife linearisation method* was introduced. The aim is to obtain a variance estimator computationally simpler than the jackknife variance estimator and yet to get values close to those obtained by the jackknife method. The main idea is to linearise the jackknife variance estimator and then to integrate the linearisation process into the weight structure in order to ensure simple computation of results. In practice, we transfer the jackknife calculation from the resample space level to the sample unit level by applying theoretical derivations in the case of the estimator of a smooth parameter. The variance estimator is of the form $\hat{V}(\hat{\theta}) = \frac{1}{n(n-1)} \sum_{i=1}^{n} l_{i,\hat{\theta}}^2$, where n is the size of the original sample and the value $l_{i,\hat{\theta}}$ is an outcome of the linearisation process that depends on the estimator $\hat{\theta}$. Yung and Rao (1996) provided results for a post-stratified estimator, a generalised regression estimator, and an estimator of the ratio. Canty and Davison (1999) studied different resampling methods in the context of the Labour Force Survey, and they also gave some examples of the applications of the jackknife linearisation method. Although this method uses the idea of the jackknife, the term $l_{i,\hat{\theta}}$ has to be created separately for every estimator and this may lead to rather cumbersome derivations.

3.9. *Summary of the Main Variance Estimation Methods*

The following table presents the main methods of variance estimation in the context of finite population sampling. The concepts are described in Chapters 2 and 3. In the resampling procedures the basis of the variance estimator is assumed to be the conditional variance or its metasampling estimator, except for the random groups and the general jackknife methods, where no conditional variance is used.

Table 3.4. Main Variance Estimation Methods

	Method	Main idea / **Resample size** / *Scaling approach*			
Units of the sample	Taylor	Linearise the estimator in order to make the calculations at the unit level in terms of linear estimators and (possibly) modified variables.			
	Woodruff	Use the properties of the inverse cumulative sampling distribution of a variable.			
Strict resampling	Jackknife $n-1$	Use all SI resamples of size $n-1$. This is a special case of the jackknife method.			*The scaling can be either internal, external or a mixture of both. In internal scaling the variability of the weights or the variables is adjusted to the scale following the linear case. In external scaling there is a coefficient adjusting the conditional variance or the bootstrap estimator of it.*
	SIR	Use all SIR resamples. This forms the basis for the traditional bootstrap method.	**The resample size is usually chosen near the original sample size *n*. Another option is a size near the resample size fulfilling the variance equality condition in the linear case, denoted here by n_b.**		
	SI	Use all SI resamples.			
Conditional metasample	Dependent random groups	Divide a sample into random groups and utilising the group information for variance estimation.	**The resample size is an outcome of the random group division.**		
	Jackknife (not $n-1$)	Combine the random group units for variance estimation.	**The resample size is an outcome of the combination process.**		
Modified resampling design	Randomisation with n_b	Select with $n_{b,l}$ (n_b rounded down) and $n_{b,u}$ (n_b rounded up) following a specific probability structure.		**The resample size is an outcome of the conditions set by the method.**	*There are no scaling terms. Adjusting the resampling design in order to fulfil the linear case criterion.*
	Mirror-Match Method with randomisation	Combine subsamples independently for each resample to be utilised in variance estimation.			
	BWO with randomisation	Create a pseudopopulation (possibly of varying size) for the selection process, with resample sizes $n_{b,l}$ and $n_{b,l}$ and a probability structure.			
Linearising the variance estimator	Linearised jackknife	Find the unit-level weight structure that is a consequence from the linearisation of the jackknife estimator.			

4. USING CUMULANTS AND k-STATISTICS FOR VARIANCE OF VARIANCE

4.1. Some History

The theory of moments, cumulants and *k*-statistics was first introduced by Fisher (1930). Based on Dwyer's (1938) work (from Wishart 1952), Dressel (1940) introduced the *L*-statistics specified for the finite population case. Kendall's book (1943) covers and develops these issues further. Irwin and Kendall (1944) provided sampling moments of moments for a finite population. Tukey (1950) emphasised in his article the usefulness of *k*-statistics, and he gave some detailed examples of different *k*-statistics utilising the theory of symmetric sums. Wishart (1952) gave the moment coefficients of the k-statistics to the sixth order, as well as some formulae for the seventh and eighth orders. McCullagh's book (1987) gives a modern overview on cumulants and *k*-statistics. This theory was applied in the context of the π estimator only rather recently. In Meister and Traat (1997) the moments and cumulants of the π estimator and of inclusion indicators are considered. Efforts to utilise the current capability of computers for automatic cumulant calculations, also in the finite population environment, have been demonstrated in Bellhouse (2001) using the program SymSS (Stafford and Bellhouse 1997, Bellhouse et al. 1997).

Population cumulants and moments have been important estimation objects in classical statistics. The same cannot be said for sampling theory. Only the second order cumulant – population variance – has received some attention. Here we consider its estimation with resampling theory using some known results on cumulants and *k*-statistics. We derive the scale coefficient Q and its estimators. *The result demonstrates that when more complex population parameters than simple totals are being estimated, the population distribution effects in the scale coefficient Q.*

4.2. Definitions

Most of the material on moments and cumulants is based on sections in Kendall and Stuart (1969). Other sources are stated separately. The *moments* describe properties of the distribution. In the *finite population case* it is assumed that each value y_i, $i = 1, ..., N$, has the probability mass $1/N$. The finite population moments are moments of this discrete distribution. Thus, the *r-order moment* is

$$\mu'_r = \frac{1}{N}\sum_{i=1}^{N} y_i^r , \qquad (4.2.1.)$$

and the *central r-order moment* is

$$\mu_r = \frac{1}{N}\sum_{i=1}^{N}(y_i - \mu'_1)^r \,. \tag{4.2.2.}$$

The *cumulants* κ_r present an alternative for the description of the distribution, and they are defined by

$$\ln\phi(t) = \sum_{n=0}^{\infty}\kappa_n \frac{(it)^n}{n!}\,, \tag{4.2.3.}$$

where $\phi(t) = \frac{1}{N}\sum_{i=1}^{N} e^{ity_i}$ is the finite population characteristic function. The general relationships between first moments and cumulants to either direction (up to the order 10) can be found e.g. in Kendall and Stuart (1969).

Table 4.1. Moments and Cumulants in Terms of Each Other Up to the 4th Order

$\mu'_1 = \kappa_1$	$\mu'_2 = \kappa_2 + \kappa_1^2$	$\mu'_3 = \kappa_3 + 3\kappa_2\kappa_1 + \kappa_1^3$	$\mu'_4 = \kappa_4 + 4\kappa_3\kappa_1 + 3\kappa_2^2 + 6\kappa_2\kappa_1^2 + \kappa_1^4$
$\mu_1 = 0$	$\mu_2 = \kappa_2$	$\mu_3 = \kappa_3$	$\mu_4 = \kappa_4 + 3\kappa_2^2$
$\kappa_1 = \mu'_1$	$\kappa_2 = \mu'_2 - \mu'^2_1$	$\kappa_3 = \mu'_3 - 3\mu'_2\mu'_1 + 2\mu'^3_1$	$\kappa_4 = \mu'_4 - 4\mu'_3\mu'_1 - 3\mu'^2_2 + 12\mu'_2\mu'^2_1 - 6\mu'^4_1$

For the sample we have the *r-order moment*

$$m'_r = \frac{1}{n}\sum_{j=1}^{n} y_j^r \tag{4.2.4.}$$

and the *central r-order moment* (*mean-moment*)

$$m_r = \frac{1}{n}\sum_{i=1}^{n}(y_i - m'_1)^r \tag{4.2.5.}$$

which can be alternatively written as

$$m_r = \frac{1}{n}\sum_{i=1}^{n}\left[\sum_{q=0}^{r}\binom{r}{q} y_i^{r-q}(-m'_1)^q\right] = \sum_{q=0}^{r}\binom{r}{q}(-m'_1)^q m'_{r-q}.$$

A function of the observations $y_1, \ldots, y_n$ is called *symmetric* if it depends explicitly on every y_i and its value is unchanged if we interchange any two y's. The *power sums*, $s_r = \sum y_i^r$, are examples of symmetric functions.

Another type of symmetric function is a *sum of products,* like the expression $\sum y_i^2 y_j y_k^3 y_l$, which includes four *different* suffixes with the summation taking place over all of them. Here the number of terms in the sum is $n(n-1)(n-2)(n-3)$.

The definition of the *k*-statistic in general is very simple. The *k*-statistic of order *r*, denoted k_r, is defined as a sample function whose expectation with respect to the sampling distribution is the *r*-order population cumulant:

$$E(k_r) = \kappa_r .$$

For i.i.d. sampling, and so also for SIR sampling, the statistic k_r is a symmetric function of the observations

$$k_r = \sum\sum (y_1^{r_1} y_2^{r_2} \ldots y_{z_1}^{r_1} y_{z_1+1}^{r_2} \ldots y_{z_1+z_2}^{r_2} \ldots y_{z_1+\ldots+z_s}^{r_s}) A(r_1^{z_1} \ldots r_s^{z_s})$$

where the second summation extends over all the ways of assigning the $z_1 + z_2 + \ldots + z_s$ subscripts (including permutations) from the n available, the first summation extends over all partitions of the number r, $(r_1^{z_1} r_2^{z_2} \ldots r_s^{z_s})$, and $A(r_1^{z_1} r_2^{z_2} \ldots r_s^{z_s})$ is a number that depends on the partition (Kendall and Stuart 1969). This approach won't provide us with a simple way to resolve *k*-statistics for different orders *r*.

Kendall and Stuart (1969) provide the *k*-statistics under i.i.d sampling up to the eighth order in terms of symmetric products and sums, and up to the fourth order in terms of sample moments. Some examples of the *k*-statistics:

$$k_1 = m'_1 = \frac{1}{n}\sum_{i=1}^{n} y_i \qquad (4.2.6.)$$

$$k_2 = \frac{n}{n-1} m_2 = \frac{n}{n-1}\left[m'_2 - m'^2_1\right] = \frac{n}{n-1}(\overline{y^2} - \bar{y}^2) = \frac{1}{n-1}\sum_{i=1}^{n}(y_i - \bar{y})^2 \qquad (4.2.7.)$$

$$k_3 = \frac{n^2}{(n-1)(n-2)} m_3 \qquad (4.2.8.)$$

$$k_4 = \frac{n^2}{(n-1)(n-2)(n-3)}\{(n+1)m_4 - 3(n-1)m_2^2\} \qquad (4.2.9.)$$

Let us denote by K_r the same function as k_r, but defined at the population level. For example K_1 *through* K_4 are defined by (4.2.6) - (4.2.9) with n replaced by N and sums taken over the population instead of the sample; thus sample moments are replaced by population moments. Then for SI sampling there is the property

$$E(k_r) = K_r .$$

Note that there is a difference between K_r functions and cumulants.

Table 4.2. Cumulants and K_r Functions in the Population to the 4th Order

$\kappa_1 = \mu'_1$	$K_1 = \mu'_1$
$\kappa_2 = \mu'_2 - \mu'^2_1$	$K_2 = \frac{N}{N-1}\left[\mu'_2 - \mu'^2_1\right]$
$\kappa_3 = \mu_3$	$K_3 = \frac{N^2}{(N-1)(N-2)}\mu_3$
$\kappa_4 = \mu_4 - 3\mu_2^2$	$K_4 = \frac{N^2}{(N-1)(N-2)(N-3)}\{(N+1)\mu_4 - 3(N-1)\mu_2^2\}$

By definition the k-statistic is a random variable and it can be characterised by classical distributional characteristics – moments and cumulants. The *cumulants of k-statistics* have been studied extensively in the classical i.i.d. sampling case. In general, for the expectations of products of k-statistics Kendall and Stuart present a combinatorial method using known properties of cumulants. Further, they provide tables of the cumulants of the most important k-statistics. Since the variance of the estimator $\hat{\theta}$ is its second cumulant, it is essential to express the estimator as a symmetric function in terms of one or more k-statistics, and then to apply the rules for finding cumulants of k-statistics.

Kendall and Stuart do not provide any similar method for finding the moments and cumulants of a finite population. However, the following rule from the book is applicable for these purposes: if there is some symmetric function whose expectation under SIR sampling may be expressed *linearly* in terms of cumulants, say

$$E_{SIR}(f) = \sum a_j K_j ,$$

we may write down the analogous relation for SI sampling in terms of K's:

$$E_{SI}(f) = \sum a_j K_j .$$

We will apply these properties for estimating the variance of the population variance with the help of resampling under the SIR / SI case.

4.3. *Theoretical Correction Coefficient for Estimation of Population Variance*

The concept of a true correction coefficient. In the resampling situation the conditional variance $V(\hat{\theta}_b \mid \underline{k}_a)$ is used for estimating the variance $V(\hat{\theta}_a)$, and usually it needs some correction. Earlier we considered a/the linear case correction coefficient. Here we introduce a true correction coefficient which works for any estimator

$$Q_E = \frac{V(\hat{\theta}_a)}{E_a[V(\hat{\theta}_b \mid \underline{I}_a)]}. \qquad (4.3.1.)$$

With this coefficient the variance estimator is unbiased, i.e. $Q_E E_a[V(\hat{\theta}_b \mid \underline{I}_a)] = V(\hat{\theta}_a)$.

In simple situations (e.g. the linear estimator) this coefficient does not depend on the y-variable in the population. Here we consider the case in which distributional characteristics of the y-variable (its cumulants) are involved in Q_E. We let the sampling and resampling designs be simple – SI and SIR – but we consider more complex estimator here.

Let us consider the finite population variance, i.e. the second cumulant κ_2. Its unbiased estimator under SIR sampling is the second k-statistic k_2, given in (4.2.7). Under SI sampling k_2 is an unbiased estimator for K_2 (which in sampling theory is ordinarily denoted by S_y^2). According to Table 4.2 it is almost unbiased for κ_2 as well: $E_{SI}(k_2) = K_2 = \frac{N}{N-1}\kappa_2$. We derive the coefficient $Q_E = \frac{V(k_2)}{E_a[V(k_{2b} \mid \underline{k}_a)]}$ for two sampling/resampling combinations: SI/SI, and SIR/SI.

Variance of k_2. The variance of k_2 under SI sampling is (Kendall and Stuart 1969)

$$V_{SI}(k_2) = \frac{(N-n_a)(Nn_a - n_a - N - 1)}{n_a(n_a-1)N(N+1)}K_4 + \frac{2(N-n_a)}{(n_a-1)(N+1)}K_2^2 \equiv A_{SI}K_4 + B_{SI}K_2^2. \qquad (4.3.2.)$$

where K_4 and K_2 are given in Table 4.2, and A_{SI}, B_{SI} are notations for their coefficients. Correspondingly, the variance of k_2 under SIR sampling is (Kendall and Stuart 1969)

$$V_{SIR}(k_2) = \frac{1}{n_a}\kappa_4 + \frac{2}{n_a - 1}\kappa_2^2 \equiv A_{SIR}\kappa_4 + B_{SIR}\kappa_2^2, \qquad (4.3.3.)$$

where κ_2 and κ_4 are the population cumulants (see Table 4.1)

Conditional variance of the resampling estimator of k_2. The resampling estimator of k_2 is the second k-statistic k_{2b} or its modification in the second-phase sample. In SI resampling we use k_{2b} which is unbiased for k_2; its conditional variance is analogically to (4.3.2)

$$V_{SI}(k_{2b} \mid \underline{k}_a) = \frac{(n_a - n_b)(n_a n_b - n_b - n_a - 1)}{n_b(n_b - 1)n_a(n_a + 1)} k_4 + \frac{2(n_a - n_b)}{(n_b - 1)(n_a + 1)} k_2^2 \qquad (4.3.4.)$$
$$\equiv C_{SI} k_4 + D_{SI} k_2^2,$$

where k_4 is the fourth k-statistic (4.2.9) of the first-phase sample.

In SIR resampling for unbiased estimation of k_2 we use $k'_{2b} = \frac{n_a}{(n_a - 1)} k_{2b}$, and the conditional variance of k'_{2b} is analogically to (4.3.3)

$$V_{SIR}(k'_{2b} \mid \underline{k}_a) = \frac{n_a^2}{(n_a - 1)^2} \left(\frac{\kappa_{4a}}{n_b} + \frac{2\kappa_{2a}^2}{n_b - 1} \right)$$

where κ_{4a}, κ_{2a} are cumulants at the sample level. Using their relations with k-statistics (based on Table 4.2) we get

$$V_{SIR}(k'_{2b} \mid \underline{k}_a) = \frac{(n_a - 2)(n_a - 3)}{n_b(n_a + 1)(n_a - 1)} k_4 + \left(\frac{2}{(n_b - 1)} - \frac{6}{n_b(n_a + 1)} \right) k_2^2 \qquad (4.3.5.)$$
$$\equiv C_{SIR} k_4 + D_{SIR} k_2^2.$$

Expectation of conditional variance. The expectation of the conditional variance in SI/SI is

$$E_{a,SI}[V_{SI}(k_{2b} \mid \underline{I}_a)] = C_{SI} K_4 + D_{SI}[V_{SI}(k_2) + K_2^2]$$
$$= C_{SI} K_4 + D_{SI}[A_{SI} K_4 + (B_{SI} + 1) K_2^2] \qquad (4.3.6.)$$
$$= (C_{SI} + D_{SI} A_{SI}) K_4 + D_{SI}(B_{SI} + 1) K_2^2$$

and correspondingly for SI/SIR, SIR/SI and SIR/SIR we have

$$E_{a,SI}[V_{SIR}(k'_{2b} \mid \underline{I}_a)] = (C_{SIR} + D_{SIR} A_{SI}) K_4 + D_{SIR}(B_{SI} + 1) K_2^2 \qquad (4.3.7.)$$

$$E_{a,SIR}[V_{SI}(k_{2b} \mid \underline{I}_a)] = (C_{SI} + D_{SI} A_{SIR}) K_4 + D_{SI}(B_{SIR} + 1) K_2^2 \qquad (4.3.8.)$$

$$E_{a,SIR}[V_{SIR}(k'_{2b} \mid \underline{I}_a)] = (C_{SIR} + D_{SIR} A_{SIR}) K_4 + D_{SIR}(B_{SIR} + 1) K_2^2. \qquad (4.3.9.)$$

Correction coefficients for SIR/SI and SI/SI. By using the formula for the correction coefficient (4.3.1) and appropriate combinations of (4.3.2), (4.3.3) and (4.3.6) – (4.3.9) we can get the true correction coefficients for SI/SI, SI/SIR, SIR/SI and SIR/SIR. Our aim is to express the true correction coefficients as a function of $\hat{Q}_{lin}$ (see Table 3.2).

In the SIR/SI case we have $\hat{Q}_{lin} = \frac{n_b}{n_a - n_b}$. For this we first develop (4.3.8):

$$E_{a,SIR}[V_{SI}(k_{2b} \mid \underline{I}_a)] = \left(\frac{(n_a - n_b)(n_a n_b - n_b - n_a - 1)}{n_b(n_b - 1)n_a(n_a + 1)} + \frac{2(n_a - n_b)}{(n_b - 1)(n_a + 1)} \frac{1}{n_a} \right) \kappa_4$$
$$+ \frac{2(n_a - n_b)}{(n_b - 1)(n_a + 1)} \left(\frac{2}{(n_a - 1)} + 1 \right) \kappa_2^2$$
$$= \frac{(n_a - n_b)}{n_a n_b} \left[\kappa_4 + \frac{2n_a n_b}{(n_b - 1)(n_a - 1)} \kappa_2^2 \right]$$

Now we get

$$Q_E = \frac{\frac{1}{n_a} \left[\kappa_4 + \frac{2n_a}{(n_a - 1)} \kappa_2^2 \right]}{\frac{(n_a - n_b)}{n_a n_b} \left[\kappa_4 + \frac{2n_a n_b}{(n_b - 1)(n_a - 1)} \kappa_2^2 \right]}$$

and after some development the coefficient is of the form

$$Q_E = \hat{Q}_{lin} \frac{(n_a - 1)\kappa_4 + 2n_a \kappa_2^2}{(n_a - 1)\kappa_4 + 2n_a \left(\frac{n_b}{n_b - 1} \right) \kappa_2^2}. \qquad (4.3.10.)$$

We note that the only difference between the numerator and the denominator comes from the term $\frac{n_b}{n_b - 1}$ revealing *that the true coefficient Q_E is always smaller than the linear case coefficient.* The error caused by the use of the linear case coefficient gets worse with smaller resample sizes. Thus for the estimator of the variance of k_2 under SIR sampling, the SI resample should be the largest possible, i.e. $n_b = n_a - 1$, for the linear coefficient to work the best. In Appendix B there is a SAS® program providing this example.

The important message in (4.3.10) is that the correction coefficient of a nonlinear estimator depends also on the population distribution of the y-variable (through its cumulants). Denoting the normalised cumulant of the population by

$$\kappa_4' = \frac{\kappa_4}{\kappa_2^2},$$

we can present (4.3.10) in the form:

$$Q_E = \hat{Q}_{lin}\left(1 - \frac{1}{[\kappa_4'(n_a-1)(n_b-1)/(2n_a)+n_b]}\right). \qquad (4.3.11.)$$

For the infinite normal population (also for large finite populations produced by normal superpopulations) $\kappa_4' = 0$, giving

$$Q_E = \hat{Q}_{lin}\left(1 - \frac{1}{n_b}\right).$$

For the minimal value $\kappa_4' = -2$ [1], we have

$$Q_E = \hat{Q}_{lin}\left(1 - \frac{n_a}{(n_a+n_b-1)}\right).$$

For the maximal possible resample size, $n_b = n_a - 1$, which is the normal jackknife case, we see that

$$Q_E = \hat{Q}_{lin}\left(1 - \frac{1}{2}\frac{n_a}{(n_a-1)}\right),$$

showing that for populations with κ_4' near -2, the resample variance estimator needs only about half the correction that $\hat{Q}_{lin}$ does. As seen from (4.3.11) $\hat{Q}_{lin}$ works better for populations with big κ_4'.

---

[1] Since $E(\sum_{i=0}^{n} a_i(X-EX)^i)^2 = E\sum_{i=0}^{n}\sum_{j=0}^{n} a_i a_j (X-EX)^{i+j} = \sum_{i=0}^{n}\sum_{j=0}^{n} a_i a_j \mu_{i+j} \geq 0$, we find that the matrix of the quadratic form is nonnegative definite. For n = 3, $\begin{vmatrix} 1 & 0 & \mu_2 \\ 0 & \mu_2 & \mu_3 \\ \mu_2 & \mu_3 & \mu_4 \end{vmatrix} \geq 0.$

Since $\mu_3^2 \geq 0$, consequently $\mu_4\mu_2 - \mu_2^3 \geq 0$, or $\frac{\mu_4}{\mu_2^2} \geq 1$. This results for cumulants $\frac{\kappa_4}{\kappa_2^2} \geq -2$.

For SI/SI we use the decomposition of the two-phase variance

$$V_*(k_{2b}) = E_{a,SI}[V_{SI}(k_{2b} \mid \underline{I}_a)] + V_{a,SI}[E_{SI}(k_{2b} \mid \underline{I}_a)] .$$

Since under SI k_{2b} is unbiased for k_2, we get

$$E_{a,SI}[V_{SI}(k_{2b} \mid \underline{I}_a)] = V_*(k_{2b}) - V(k_2) .$$

Now the correction coefficient can be developed into the form

$$Q_E = \frac{V_{SI}(k_2)}{E_{a,SI}[V_{SI}(k_{2b} \mid \underline{I}_a)]} = \left(\frac{V_*(k_{2b})}{V(k_2)} - 1 \right)^{-1}, \qquad (4.3.12.)$$

where $V(k_2) = V_{a,SI}(k_a)$. We know that two-phase SI/SI with sample sizes n_a and n_b in the two phases is a SI design with sample size n_b. Consequently we can evaluate both variances in (4.3.12) by using the expression (4.3.2). To find $V_*(k_{2b})$ we replace each n_a by n_b in the expression.

After simplification we get

$$\frac{V_*(k_{2b})}{V(k_2)} = \frac{(n_a - 1)(1 - n_b/N)(1 - 1/N - 1/n_b - 1/(Nn_b))K_4 + 2K_2^2}{(n_b - 1)(1 - n_a/N)(1 - 1/N - 1/n_a - 1/(Nn_a))K_4 + 2K_2^2}. \qquad (4.3.13.)$$

The variance ratio depends on the population through κ_4 / κ_2^2. We can express it through cumulants:

$$K_4' = \frac{K_4}{K_2^2} = \frac{(N-1)(N+1)}{(N-2)(N-3)} \left[\kappa_4' + \frac{6}{N+1} \right], \qquad (4.3.14.)$$

where $\kappa_4' = \kappa_4 / \kappa_2^2$ is the normalised fourth cumulant of the population. We see that

$$\lim_{N \to \infty} K_4' = \kappa_4' . \qquad (4.3.15.)$$

Based on (4.3.12) - (4.3.15) it can be seen that

$$\lim_{N\to\infty} Q_E = \left[\frac{(n_a-1)(1-1/n_b)\kappa_4' + 2}{(n_b-1)(1-1/n_a)\kappa_4' + 2} - 1\right]^{-1}. \qquad (4.3.16.)$$

Alternatively,

$$\lim_{N\to\infty} Q_E = \frac{(n_a-1)(n_b-1)\kappa_4'/n_a + 2}{(n_a-1)(n_b-1)\kappa_4'/(n_b n_a)}. \qquad (4.3.17.)$$

The linear case coefficient for SI/SI is $\hat{Q}_{lin} = \frac{(N-n_a)n_b}{N(n_a-n_b)}$. Taking the limit, we get

$$\hat{Q}_{lin}^{\infty} = \lim_{N\to\infty} \hat{Q}_{lin} = \frac{n_b}{(n_a-n_b)}. \qquad (4.3.18.)$$

With (4.3.18) we have for (4.3.17)

$$\lim_{N\to\infty} Q_E = \hat{Q}_{lin}^{\infty} \frac{(n_a-1)(n_b-1)\kappa_4' + 2n_a}{(n_a-1)(n_b-1)\kappa_4'}$$

$$= \hat{Q}_{lin}^{\infty}\left[1 + \frac{2n_a}{(n_a-1)(n_b-1)\kappa_4'}\right].$$

Here it is easy to see that the accuracy of the correction depends on κ_4'.

In principle this kind of study can be applied for other single- or multi-variate estimators (e.g. the covariance) which provide products of *k*-statistics when the first-phase and second-phase designs allow that process. However, the formulae may be more complex and the results may not be as clear as this one. For estimators having functions of *k*-statistics *not* in the form of products, e.g. the ratio or the correlation coefficient, one might consider the use of linearisation in order to transform the coefficient Q_E into a form with which this kind of study can be conducted.

5. A CORRECTION METHOD BASED ON POST-DESIGN VECTORS

The adjustments needed to estimate the variance of the estimator with the resampling variance are either rescaling procedures or resampling design adjustments. For the latter alternative a randomisation process (i.e. conducting resampling with two different resample sizes following some predetermined probability structure) is often conducted.

In this chapter we introduce a method that *utilises the theory of the design vector approach*, presented in Section 2.2. The aim is to *change the resampling design vector* $\underline{k}_b = (k_{b1}, k_{b2}, \ldots, k_{bN})$ in such a way that it corrects the resampling variance. This method avoids the *need for randomisation*, the *problems of internal scaling* (like the requirement $\hat{Q}_{lin} \leq 1$ and bad performance for some estimators, e.g. correlation) and the *correction with external scaling, which is of a rather general nature*. Basically, the *expansion of the k_i values in the vector* (i.e. the counts of how many times the population element appears in the sample) is conducted in a manner which is easy to carry out in practice.

The chapter includes the definition of the *occurrence counts*, which are essential for the method. The theoretical properties of the occurrence counts are studied, and later they are used when the *post-design vector* is defined. The *expansion principle* of the post-design vector is demonstrated with examples. All this theory is utilised when the *variance estimator based on post-design vectors* is created. Finally, some examples show the method in practice as well as a comparison between this method and the randomisation method.

5.1. Occurrence Counts

Definitions. From Section 2.2 we know that the random *design vector* $\underline{I} = (I_1, I_2, \ldots, I_N)$ and its realisation $\underline{k} = (k_1, k_2, \ldots, k_N)$ with the probabilities $p(\underline{k}) = Pr\{\underline{I} = \underline{k}\}$ describe the *sampling design* in terms of the design vector approach (Traat 2000). The vector $\underline{k}$ shows for each element in the population the number of times it appears in the sample. The set of N-dimensional points $\underline{k}$ with $p(\underline{k}) > 0$ is the *sample space*. In the most general case the sample space is an N-dimensional space on non-negative integers $k_i \in \{0, 1, \ldots\}$, or one can make some restrictions to the sample space (see Section 2.2).

Here we want to study a specific partition of the sample space created by *occurrence counts* c_g, where c_g shows how many population units are selected g times, $g \in \{0, 1, \ldots, n\}$. Then

$$\sum_{g=0}^{n} c_g g = n . \qquad (5.1.1.)$$

For WOR sampling we have only $c_0 = N - n$ and $c_1 = n$.

For a specific $\underline{k}$ vector there are $n!/\prod_{i=1}^{N} k_i!$ different drawing orders of elements under WR designs of size n, as can be seen from the formula for the general multinomial design in Section 2.2.

Let us decompose the sample space of a general multinomial design $M(n; p_1, \ldots, p_N)$ by the occurrence counts. In fact, this holds for all designs for which $I_i \in \{0, 1, \ldots, n\}$ with $\sum_{i=1}^{N} I_i = n$.

Table 5.1. Occurrence counts under the multinomial design ($N = 4$, $n = 3$)

k_1	k_2	k_3	k_4	drawing orders	c_0	c_1	c_2	c_3
3	0	0	0	1	**3**	**0**	**0**	**1**
0	3	0	0	1				
0	0	3	0	1				
0	0	0	3	1				
2	1	0	0	3	**2**	**1**	**1**	**0**
1	2	0	0	3				
2	0	1	0	3				
1	0	2	0	3				
2	0	0	1	3				
1	0	0	2	3				
0	2	1	0	3				
0	1	2	0	3				
0	2	0	1	3				
0	1	0	2	3				
0	0	2	1	3				
0	0	1	2	3				
1	1	1	0	6	**1**	**3**	**0**	**0**
1	1	0	1	6				
1	0	1	1	6				
0	1	1	1	6				

In the table all possible samples $\underline{k} = (k_1, k_2, k_3, k_4)$ are given. In fixed-size WOR $(n = 3)$ sampling only the last four $\underline{k}$ vectors can occur. The sum of the *"drawing orders"* column is 64, which is the number of all possible ordered samples, N^n (see Section 2.2 for definition). The number of different $\underline{k}$ vectors under fixed c_g values is $\dfrac{N!}{\prod_{g=0}^{n} c_g!}$.

Results for the linear case. We now prove some results for the estimator of the population mean under multinomial sampling and SIR resampling. Let $\hat{y}_i = y_i/(Np_i)$, where p_i is the single-draw probability for the multinomial design in the first phase. We study the following estimator:

$$\bar{\hat{y}}_b = \frac{1}{n_b} \sum_{i=1}^{N} k_{bi} \hat{y}_i \tag{5.1.2.}$$

where $\underline{k}_b = (k_{b1}, \ldots, k_{bN})$ is the second phase sample. We study the behaviour of $\bar{\hat{y}}_b$ when it is conditioned by fixed c_g values and the first phase sample $\underline{k}_a = (k_{a1}, \ldots, k_{aN})$. For the SIR resampling design we have

$$p(\underline{k}_b \,|\, c_0, \ldots, c_{n_b}; \underline{k}_a) = 1 \Big/ \frac{n_a!}{\prod_{g=0}^{n_b} c_g!}, \tag{5.1.3.}$$

i.e. it is a constant when c_g values are fixed. The expectation of $\bar{\hat{y}}_b$ with respect to this conditional design is

$$E(\bar{\hat{y}}_b \,|\, c_0, \ldots, c_{n_b}; \underline{k}_a) = \sum_{\underline{k}_b | c_0, \ldots, c_{n_b}; \underline{k}_a} \bar{\hat{y}}_b \; p(\underline{k}_b \,|\, c_0, \ldots, c_{n_b}; \underline{k}_a).$$

Replacing $\bar{\hat{y}}_b$ by (5.1.2) and the probabilities by (5.1.3), we get

$$E(\bar{\hat{y}}_b \,|\, c_0, \ldots, c_{n_b}; \underline{k}_a) = \frac{1}{n_b} \frac{\prod_{g=0}^{n_b} c_g!}{n_a!} \sum_{\underline{k}_b | c_0, \ldots, c_{n_b}; \underline{k}_a} \sum_{i=1}^{N} k_{bi} \hat{y}_i. \tag{5.1.4.}$$

Changing the summation order in (5.1.4) we start by summing over the first coordinates of all $\underline{k}_b$ vectors with given condition, and then sum over the second coordinates etc. We notice (see also Table 5.1) that summation in the given $(c_0, \ldots, c_{n_b})$ class yields a constant. Denoting this constant by q we have

$$E(\bar{\hat{y}}_b \mid c_0, \ldots, c_{n_b}; \underline{k}_a) = \frac{1}{n_b} \frac{\prod_{g=0}^{n_b} c_g!}{n_a!} q \sum_{i=1}^{N} k_{ai} \hat{y}_i \,. \qquad (5.1.5.)$$

To find q we consider the first coordinate k_{b1}. Its possible values are 0, 1, ..., n_b. For fixed $k_{b1} = k$ there are many such $\underline{k}_b$ vectors. Letting $\underline{k}_b^- = (k_{b2}, \ldots, k_{bN})$ we can find the number of different $\underline{k}_b^-$ vectors for fixed $k_{b1} = k$ and fixed c_g values: $\frac{(n_a - 1)!}{(c_k - 1)! \prod_{g=0 \mid g \neq k}^{n_b} c_g!}$. Now,

$$q = \sum_{k=0}^{n_b} \frac{(n_a - 1)!}{(c_k - 1)! \prod_{g=0 \mid g \neq k}^{n_b} c_g!} k = \frac{(n_a - 1)!}{\prod_{g=0}^{n_b} c_g!} \sum_{k=0}^{n_b} c_k k = \frac{(n_a - 1)! n_b}{\prod_{g=0}^{n_b} c_g!} \,.$$

Replacing q in (5.1.5)

$$E(\bar{\hat{y}}_b \mid c_0, \ldots, c_{n_b}; \underline{k}_a) = \frac{\sum_{i=1}^{N} k_{ai} \hat{y}_i}{n_a} = \bar{\hat{y}}_a \,. \qquad (5.1.6.)$$

We noticed that the second phase mean (5.1.2) is unbiased for the first phase mean (5.1.6) in the occurrence count classes under SIR resampling design. The results are also valid for SIR sampling in the first phase when we change $\hat{y}_i$ to y_i, because $1/(N p_i) = 1$. For pairs (k, k') the coefficient is $q = \frac{(n_a - 2)!}{\prod_{g=0}^{n_b} \prod_{g'=0}^{n_b} c_{g,g'}!} \sum_{k=0}^{n_b} \sum_{k' \neq k}^{n_b} c_{k,k'} k k'$.

It is of interest to see the form of the variance of the estimator $\overline{\hat{y}}_b$ conditioned by values c_s and the vector $\underline{k}_a$. This variance is

$$V(\overline{\hat{y}}_b \mid c_0, \ldots, c_{n_b}; \underline{k}_a) = E(\overline{\hat{y}}_b^2 \mid c_0, \ldots, c_{n_b}; \underline{k}_a) - \overline{\hat{y}}_a^2 . \qquad (5.1.7.)$$

Let us study the parts separately. The form of the first term on the right in (5.1.7) is

$$E(\overline{\hat{y}}_b^2 \mid c_0, \ldots, c_n; \underline{k}_a) = E(\sum_{i=1}^{N} k_{bi}^2 \hat{y}_i^2 / n_b^2 \mid c_0, \ldots, c_{n_b}; \underline{k}_a)$$
$$+ E(\sum_{i=1}^{N} \sum_{j \neq i}^{N} k_{bi} k_{bj} \hat{y}_i \hat{y}_j / n_b^2 \mid c_0, \ldots, c_{n_b}; \underline{k}_a).$$

Using the notations

$$\overline{\hat{y}_a^2} = \frac{1}{n_a} \sum_{i=1}^{N} k_{ai} \hat{y}_{ai}^2$$

$$\overline{\hat{y}\,\hat{y}'}_a = \frac{1}{n_a(n_a - 1)} \sum_{i=1}^{N} \sum_{j \neq i}^{N} k_{ai} k_{aj} \hat{y}_i \hat{y}_j \qquad (5.1.8.)$$

we get

$$E(\overline{\hat{y}}_b^2 \mid c_0, \ldots, c_{n_b}; \underline{k}_a) = \frac{\sum_{k=0}^{n_b} c_k k^2}{n_b^2} \overline{\hat{y}_a^2} + \frac{\sum_{k=0}^{n_b} \sum_{k' \neq k}^{n_b} c_{k,k'} k\, k'}{n_b^2} \overline{\hat{y}\,\hat{y}'}_a .$$

The occurrence count c_k simply shows the number of appearances of the value k in the resample vector k_b. Thus the summation $\sum_{i=1}^{N} k_{bi}^2$ is in fact the summation of different k^2 values together with the occurrence counts of k, i.e. $\sum_{k=0}^{n_b} c_k k^2$. Correspondingly we have

$$\sum_{k=0}^{n_b} \sum_{k' \neq k}^{n_b} c_{k,k'} k\, k' = \sum_{k=0}^{n_b} \sum_{k'=0}^{n_b} c_{k,k'} k\, k' - \sum_{k=0}^{n_b} c_k k^2 = \sum_{k=0}^{n_b} c_k k \sum_{k'=0}^{n_b} c_{k'} k' - \sum_{k=0}^{n_b} c_k k^2 = n_b^2 - \sum_{k=0}^{n_b} c_k k^2$$

and finally we get

$$E(\overline{\hat{y}}_b^2 \mid c_0, \ldots, c_{n_b}; \underline{k}_a) = \frac{\sum_{i=1}^{N} k_{bi}^2}{n_b^2} \overline{\hat{y}_a^2} + \left(1 - \frac{\sum_{i=1}^{N} k_{bi}^2}{n_b^2}\right) \overline{\hat{y}\,\hat{y}'}_a. \qquad (5.1.9.)$$

The second term of (5.1.7) is

$$\overline{\hat{y}}_a^2 = \frac{1}{n_a} \overline{\hat{y}_a^2} + \left(1 - \frac{1}{n_a}\right) \overline{\hat{y}\,\hat{y}'}_a \qquad (5.1.10.)$$

and if we substitute the terms (5.1.9) and (5.1.10) into (5.1.7), we get

$$V(\overline{\hat{y}}_b \mid c_0, \ldots, c_{n_b}; \underline{k}_a) = \left(\frac{\sum_{i=1}^{N} k_{bi}^2}{n_b^2} - \frac{1}{n_a}\right) S_{\hat{y}a}^2, \qquad (5.1.11.)$$

where $\hat{S}_{\hat{y}}^2 = \overline{\hat{y}_a^2} - \overline{\hat{y}\,\hat{y}'}_a$ is the sample variance of the variable $\hat{y}_i = y_i/(N p_i)$ in the first phase sample. In the special case of $\hat{y}_i = y_i$ we get S_{ya}^2. When k can take on only the values 0 or 1 (SI resampling), we have $1/n_b - 1/n_a$ in the parenthesis. Let us study (5.1.11) in more detail.

5.2. *Post-Design Vector*

Definitions. As explained before, the vector $\underline{k}$ represents the realised sample. However, nothing prevents us from *modifying* the sample afterwards, by increasing some $k_j > 0$ of the vector $\underline{k}$ in order to emphasise some units. This means that the frequency of these units in the sample will be increased. Naturally these actions affect estimation and its accuracy.

Let us consider a realised sample $\underline{k}$ of size n. We will expand the values of the sample $\underline{k}$ with the vector $\underline{d} = (d_1, \ldots, d_N)$ with $d_i \geq 1$ (not necessary an integer). The expanded vector is $\underline{k}^* = (d_1 k_1, \ldots, d_N k_N)$. This vector $\underline{k}^*$ is called the *post-design vector*, and the new sample size is

$$n^* = \sum_{i=1}^{N} d_i k_i. \qquad (5.2.1.)$$

In the present context the post-design vector is a *tool for resampling*, i.e. the realised resample $\underline{k}_b$ is converted into $\underline{k}_b^*$ *for variance estimation purposes.*

Carrying out the expansion. There are many different ways of constructing a post-design vector (in practice $\underline{d}$ is defined for *selected* units, because $k_i = 0$ for others). For example, for sampling with fixed size n, we can set $d_j = 1$ for one selected unit j, and for others multiply the value by q, i.e. the new occurrence count values are $c_0^* = N - n, c_r^* = q(c_r - 1) + 1$, and for others qc_r. Furthermore, less than $n-1$ units may be chosen for the expansion or the k-value of the fixed unit may be more than one.

In general, the principle by which the units with altered k-values are chosen should be independent from the units in the population. This principle can be preserved by for example defining the first $n-1$ units chosen in the *ordered realised sample* as the targets.

Examples. Let us assume a population of four elements and the realised sample $\underline{k}$ = (1, 1, 0, 1). When (say) the first and fourth k-values increase, it is evident that e.g. the sample $\underline{k}$ = (1000, 1, 0, 1000) is close to the sample $\underline{k}$ = (1, 0, 0, 1) as far as the usual estimators of most of the parameters are concerned, i.e. the "importance" of the second unit decreases. This can be reasoned from the fact that WOR samples form a subset of WR samples. The occurrence counts for WOR sampling are $c_0 = N - n$ and $c_1 = n$. For suitable sample sizes n in WR sampling it is possible to find samples where $c_0 = N - n$ and $c_2 = n / 2$ or $c_0 = N - n$ and $c_3 = n / 3$ etc.. Then these samples are "double", "triple" etc. the lower size samples. For example $\underline{k} = (2,2,0,0,0)$ is a sample with n = 4 in a population N = 5. It includes the units 1 and 2 twice each, and in this case most of the estimators $\hat{\theta}$ ("smooth functions", or in other words "functions of the totals", and others) will give the same value both with the vector $\underline{k} = (2,2,0,0,0)$ and with $\underline{k} = (1,1,0,0,0)$. The extreme is the rare case with $c_0 = N - n$ and $c_n = 1$, where the same unit is selected n times. Some estimators are not defined in such a case, e.g. the correlation coefficient. The other vectors are "unbalanced" samples from the population, i.e. there is more than one non-zero outcome for $c_1, \ldots, c_n$.

With-replacement (WR) samples: The expansion is a multiplied value of k_i, e.g. for k_i = 2 we can have 4, 6, 8, etc.. If we decide to leave one sampling unit without the count expansion, we just choose one value j with $k_j > 0$. There is no expansion if $k_j = 1$, and for greater values of k_j we expand the value $k_j - 1$ and add the remaining one in the end, e.g. for k_j = 3 we get 5, 7, 9, etc. The following example is a realised sample where the units 1, 2 and 4 are chosen, with the vector (2, 2, 0, 1).

Table 5.2. Examples of Expansions in WR Samples

n'	k_1	k_2	k_3	k_4	c_0	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8	...
5	2	2	0	1	1	1	2	0	0	0	-	-	-	
9	4	3	0	2	1	0	1	1	1	0	0	0	0	
13	6	4	0	3	1	0	0	1	1	0	1	0	0	...
17	8	5	0	4	1	0	0	0	1	1	0	0	1	...

Here the unit 2 has the expansion restriction described earlier. Note that the selection of this unit must be based on the realised sample units (possibly including repetitions), not on the distinct units of the sample, e.g. in this example one unit from the set {1, 1, 2, 2, 4} is chosen, not from the set {1, 2, 4} (4 over-represented).

Without replacement (WOR) samples: The following example is a realised sample where the units 1, 2 and 4 are chosen.

Table 5.3. Examples of Expansions in WOR Samples

n'	k_1	k_2	k_3	k_4	c_0	c_1	c_2	c_3	c_4
3	1	1	0	1	1	3	0	0	-
5	2	1	0	2	1	1	2	0	0
7	3	1	0	3	1	1	0	2	0
9	4	1	0	4	1	1	0	0	2

The application of this method is not in point estimation, because the creation of an imbalance in the linear case and in most of the non-linear estimators increases variance and makes the estimator less effective. *Its primary application can be found in the context of resampling for variance estimation.*

5.3. *Variance Estimation with Post-Design Vector Estimators*

Background. We wish to estimate the variance $V(\hat{\theta}_a)$ of the estimator $\hat{\theta}_a$ (of size n_a). If we want to use resampling for variance estimation we have to decide how to correct *differences between the original design and the resampling design.* Note that even if SI sampling is used in both phases the designs are different due to different sample sizes, and this also causes *differences between the original sample space and the resample space.* The most common correction method is to use the linear case coefficient $\hat{Q}_{lin} = \hat{V}(\bar{y}_a)/V(\bar{y}_b \mid \underline{k}_a)$, provided that the coefficient $\hat{Q}_{lin}$ and the study variable y are independent.

One strategy is to adjust the resampling design so that the criterion $\hat{Q}_{lin} = 1$ is fulfilled, i.e. the conditional variance $V(\hat{\theta}_b \mid \underline{k}_a)$ or its bootstrap estimator will be the variance estimator of $V(\hat{\theta}_a)$. In his articles Sitter (1992a, 1992b) deals with different methods that *avoid* rescaling terms in the variance estimator formula, in the variables or in the weights. The main idea is that the resampling design has to include the necessary adjustments in order to achieve the linear case criterion. Whatever the chosen resampling scheme is, often it is enough to find a proper resample size n_b (almost always a non-integer). The integer values around n_b, i.e. $n_{b,u}$ and $n_{b,l}$ will be part of the resampling scheme.

In order to conduct the resampling scheme with a non-integer sample size *the randomisation process* is needed. Randomisation is also used for other non-integer terms, such as subsample sizes, repetition terms for the resample size, and pairs of both terms. In practice, we assign probabilities for different outcomes of the integer terms, e.g. $p(n_{b,l})$, $p(n_{b,u}) = 1 - p(n_{b,l})$, and we use them in the bootstrap selections.

Below we propose another method that *avoids the randomisation process* by using a *suitably chosen post-design vector in resampling*. The result of the method is that the *resample size is fixed* and the conditional variance $V(\hat{\theta}_b^* \mid \underline{k}_a)$ or its bootstrap estimator $\hat{V}(\hat{\theta}_b^* \mid \underline{k}_a) = \sum_{s_b \in os^*} \frac{(\hat{\theta}_{s_b}^* - \overline{\hat{\theta}^*})^2}{A-1}$ is the desired variance estimator, where $\hat{\theta}_{s_b}^*$ is the estimator using $\underline{k}^* = (d_1 k_{b1}, \ldots, d_N k_{bN})$ and $\overline{\hat{\theta}_b^*} = \sum_{s_b \in os^*} \hat{\theta}_{s_b}^* / A$.

Principles of the method. The aim is to create a situation where *the conditional variance of the post-design estimator* $\overline{y}_b^*$ *using* $\underline{k}^* = (d_1 k_{b1}, \ldots, d_N k_{bN})$ *equals the conditional variance of the resample estimator* $\overline{y}_b$ *with the (possibly non-integer) sample size* n_b, i.e. reaching the linear case condition by ensuring that

$$V(\overline{y}_b^* \mid \underline{k}_a) = V(\overline{y}_b \mid \underline{k}_a, n_b). \qquad (5.3.1.)$$

This is equivalent to the situation $V(\overline{y}_b^* \mid \underline{k}_a) = \hat{V}(\overline{y}_b)$, where $\hat{V}(\overline{y}_b)$ is the variance estimator based on the first-phase sampling design. With these two conditional variances it is possible to find the factors $\underline{d} = (d_1, \ldots, d_N)$ which are needed in order to to create a post-design vector structure $\underline{k}_b^* = (d_1 k_{b1}, \ldots, d_N k_{bN})$ that will make (5.3.1) true. Then we have the expanded vector $\underline{k}_b^*$ based on the principles chosen for the method.

For example, with $d_j = 1$ and $d_i = q$, $i \neq j$ in WOR sampling we get

$$\underline{k}_b^* = (k_{b1}^*, \ldots, k_{bN}^*) = (k_{b1}q, \ \ldots, k_{bj-1}q, 1, k_{bj+1}q \ \ldots, k_{bN}q), \tag{5.3.2.}$$

where j is the selected resample unit with the expansion restriction. The size $n_{b,u}$ is taken as the *rounded-up* n_b. The occurrence counts are: $c_0 = N - n_{b,u}$, $c_1 = 1$, $c_q = n_{b,u} - 1$.

This is one solution for constructing the variance estimator

$$\hat{V}(\hat{\theta}_a) = V(\hat{\theta}_b^* \mid \underline{k}_a). \tag{5.3.3.}$$

Note that for WOR resampling the situation $n_{b,u} = n_a$ is acceptable here, because then there are $\dfrac{n_a!}{\prod_{g=0}^{n_a} c_g!} = \dfrac{n_a!}{1!(n_a-1)!} = n_a$ estimates. As an example, for the case $N = 6$, $n_a = 4$, $n_{b,u} = 4$, $\underline{k}_b = \underline{k}_a = (0,1,1,0,1,1)$ we still have (0, 1, q, 0, q, q), (0, q, 1, 0, q, q), (0, q, q, 0, 1, q) and (0, q, q, 0, q, 1).

Since the parameter is *a smooth function,* i.e. $\theta = f(t_{U,1}, \ldots, t_{U,z})$, the expansion factor q (or factors) can be placed straightforwardly into the estimator calculation; e.g. for $n_b = 3$ and $q = 2.6$ we sum up for every variable (here y) $\hat{t}_y^* = N(2.6y_{(1)} + 2.6y_{(2)} + y_{(3)})/6.2$ where $y_{(1)}$ is the value of y of the first selected unit, etc. In addition, the quantiles are formed from data where each unit has a frequency $k_i^* = d_i k_i$ in the expanded vector. Although the formula of the factor q may be complicated as such, *this factor is the same for variance estimation of the estimators of all parameters.* In addition the weight structure can easily *be adjusted for programming purposes.*

Example: SI/SIR sampling with one q-factor. Here we have a situation in which the original design is SI and the resampling design is SIR. We want to obtain the exact sample size that fulfils the linear case condition, i.e.

$$\frac{1}{n_b}\left(\frac{n_a - 1}{n_a}\right) = \frac{N - n_a}{N n_a}. \tag{5.3.4.}$$

Solving for the resample size in (5.3.4) provides us with

$$n_b = \frac{N(n_a - 1)}{N - n_a}. \tag{5.3.5.}$$

Usually n_b is not an integer. This solution resembles McCarthy and Snowden's (1985) *Bootstrap With Replacement* method (BWR); see also Sitter (1992b).

Let us consider a resample $\underline{k}_b$ of size $n_{b,u}$ with $c_0 = N - n_{b,u}$, $c_1, c_2, \ldots, c_{n_{b,u}}$. In this case one of the selected units in the resample is not to be expanded by the q-factor. For the vector $\underline{k}_b$ we have $k_{bi}^* \in \{0, 1, q\}$ and the artificial resample size is

$$n_b^* = \sum_{i=1}^{N} k_{bi}^* = (n_{b,u} - 1)q + 1 . \tag{5.3.6.}$$

Correspondingly, the term needed for the variance expression is

$$\sum_{i=1}^{N} k_{bi}^{*2} = (n_{b,u} - 1)q^2 + 1 \cdot 1^2 + (N - n_{b,u}) \cdot 0^2 = (n_{b,u} - 1)q^2 + 1 . \tag{5.3.7.}$$

We know from (5.1.13) that in this case $V(\bar{y}_b \,|\, c_0, \ldots, c_{n_{b,u}}; \underline{k}_a) = \left(\frac{\sum_{i=1}^{N} k_{bi}^2}{n_{b,u}^2} - \frac{1}{n_a} \right) \hat{S}_y^2$. For some non-integer value n_b with $n_{b,u} - 1 < n_b < n_{b,u}$ the condition

$$\frac{1}{n_b} - \frac{1}{n_a} = \frac{\sum_{i=1}^{N} k_{bi}^{*2}}{n_b^{*2}} - \frac{1}{n_a}$$

finally has the form

$$\frac{1}{n_b} = \frac{\sum_{i=1}^{N} k_{bi}^{*2}}{n_b^{*2}} . \tag{5.3.8.}$$

This is needed for the variance estimation method based on the post-design estimator. The artificial post-design vector is created by adjusting the expanding factor q for the required resample size n_b in (5.3.8) with (5.3.6) and (5.3.7), i.e.

$$\frac{1}{n_b} = \frac{[(n_{b,u} - 1)q^2 + 1]}{[(n_{b,u} - 1)q + 1]^2} \tag{5.3.9.}$$

Solving (5.3.9) with respect to q it follows that

$$q = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A} \tag{5.3.10.}$$

where

$$A = (n_{b,u} - 1)^2 - n_b(n_{b,u} - 1) \tag{5.3.11.}$$

$$B = 2(n_{b,u} - 1), \tag{5.3.12.}$$

$$C = 1 - n_b . \tag{5.3.13.}$$

The larger root is selected. The post-design vector

$$\underline{k}^* = (k_1 q, \ ..., \ (k_j - 1)q + 1, ..., k_N q)$$

represents a sample which because of the suitable q-expansion provides the situation $V(\bar{y}_b^* \mid \underline{k}_a) = \hat{V}(\bar{y}_a)$. In practice we expand *the first* $n_b - 1$ *units selected in the resample with q,* and *the last unit of the resample* remains *unexpanded.* Note that the sample element appearing in the last unit may occur elsewhere in the resample due to WR sampling in the second phase.

5.4. *Post-Design Vector Method as an Alternative to Randomisation: an Example*

Randomisation. Following Section 3.7, if we consider the case where we have the exact non-integer resample size n_b (solving for n_b from $\hat{Q}_{lin} = 1$ – see Table 3.2 for some examples of $\hat{Q}_{lin}$) and only the resample sizes vary (i.e. $n_{b,l} < n_b < n_{b,u}$), then the randomisation principle following (3.7.10) gives the probability

$$p(n_{b,l}) = \frac{G_{n_b} - G_{n_{b,u}}}{G_{n_{b,l}} - G_{n_{b,u}}}, \tag{5.4.1.}$$

where the G-terms are not dependent on the variable y (see Section 3.7 for a detailed explanation). For example, for the case N = 15, n_a = 7 and with the SI design in both phases, the solution of the exact sample size is

$$n_b = \frac{N n_a}{2N - n_a} = \frac{15 \cdot 7}{2 \cdot 15 - 7} = 4.5652174 .$$

Because the second phase is SI sampling, for $E_a[V(\bar{y}_b \mid \underline{k}_a)] = G_{n_b} f_{SI}(y_U)$ we have $G_{n_b} = \frac{1}{n_b} - \frac{1}{n_a}$ and $f_{SI}(y) = S_y^2$. With this we get

$$p(n_{b,l}) = \frac{G_{n_b} - G_{n_{b,u}}}{G_{n_{b,l}} - G_{n_{b,u}}} = \frac{\frac{1}{n_b} - \frac{1}{n_{b,u}}}{\frac{1}{n_{b,l}} - \frac{1}{n_{b,u}}} = \frac{0.219 - 0.2}{0.25 - 0.2} = 0.38$$

and

$$p(n_{b,u}) = 1 - p(n_{b,l}) = 0.62 .$$

We use the *variance estimator* (3.7.11)

$$\hat{V}(\hat{\theta}_a) = p(n_{b,l}) V(\hat{\theta}_{b,l} \mid \underline{k}_a) + [1 - p(n_{b,l})] V(\hat{\theta}_{b,u} \mid \underline{k}_a)$$

or its bootstrap approximation $\sum_{s_b \in os^*} \frac{(\hat{\theta}_{s_b} - \bar{\hat{\theta}})^2}{A-1}$ with each $\hat{\theta}_{s_b}$ based on either $n_{b,u}$ or $n_{b,l}$ units, depending on the outcome of the randomisation. *In practice we select resamples of size* 4 *with probability* 0.38 *and resamples of size* 5 *with probability* 0.62.

Post-design vector method. In this case *the randomisation can be avoided by using the post-design vector method.* Our example with $N = 15$, $n_a = 7$ gives

$$A = n_{b,l}^{\ 2} - n_b n_{b,l} = 16 - 4.5652174 * 4 = -2.2608696$$
$$B = 2n_{b,l} = 2 * 4 = 8 ,$$
$$C = 1 - n_b = 1 - 4.5652174 = -3.5652174 .$$

and then we get from (5.3.10)

$$q = \frac{-8 \pm \sqrt{64 - 32.2419665}}{-4.5217392}$$

with the larger solution $q = 3.015527$.

We can check that $\frac{n_{b,l}q^2+1}{[n_{b,l}q+1]^2}=0.2190476$ and $\frac{1}{n_b}=0.2190476$ are equal as the theory expects. The variance estimator is thus $\hat{V}(\hat{\theta}_a)=V(\hat{\theta}_b^*|\underline{k}_a)$ and its bootstrap approximation is $\hat{V}(\hat{\theta}_b^*|\underline{k}_a)=\sum_{s_b\in os^*}\frac{(\hat{\theta}_{s_b}^*-\overline{\hat{\theta}^*})^2}{A-1}$. In practice we *select only resamples of size* 5 and multiply the first four selected units by an *expanding constant* 3.015527.

Let us present an example of a realisation of a resampling design vector with selection order (7, 2, 6, 5, 4):

$$k_b^* = \{0, 3.015527, 0, 1, 3.015527, 3.015527, 3.015527\}.$$

Correspondingly the estimator of the mean *for this vector* $\underline{k}^*$ is

$$\overline{y}_b^* = \sum_{i=1}^{N} k_{bi}^* y_i \Big/ n_b^* = [3.015527\cdot(y_7+y_2+y_6+y_5)+y_4]/13.062108$$

and the correlation coefficient is $\hat{\rho}_b^* = \left(\overline{x_b^* y_b^*}-\overline{x_b^*}\,\overline{y_b^*}\right)\Big/\left[\left(\overline{x_b^{*2}}-\overline{x_b^*}^2\right)\left(\overline{y_b^{*2}}-\overline{y_b^*}^2\right)\right]$. Further, the conditional variance $V(\hat{\rho}_b^*|\underline{k}_a)$ can be used as an estimator of $\hat{V}(\hat{\rho}_a)$ and if there is not enough computer capacity to deal with the conditional resample space as a whole we can use the bootstrap estimator of the conditional variance, i.e. $\hat{V}(\hat{\rho}_b^*|\underline{k}_a)=\frac{\sum_{s_b\in os^*}(\hat{\rho}_{s_b}^*-\overline{\hat{\rho}^*})^2}{A-1}$.

5.5. *Summary*

The *design vector* provides information about the number of appearances in the sample for each element in the population. This chapter the *occurrence counts*, which are the values that include information on how many zeroes, ones, twos etc. there are in the design vector. These occurrence counts and their theoretical properties (studied in this chapter as well) are essential when creating the variance estimation method based on the *post-design vector*. The adjustments that are required in order to correct the difference between the original and resampling designs as well as scale and weight differences are performed here by *artificially recreating (in practice unevenly expanding) the resampling design vector* with some conditions based on the theoretical properties of the occurrence counts. This method can serve for example as an alternative to randomisation. The chapter provides the principles of expansion with some examples and properties. Finally, the *variance estimator based on the post-design vector* is presented, with two examples.

6. ESTIMATOR-DEPENDENT CORRECTION BASED ON RESAMPLING

The main criticism against the linear case condition used in practically all traditional resampling methods is the lack of estimator-specific correction when estimating the variance, especially when populations (strata) and sample sizes are small. If we are going to reduce the bias of the variance estimator, then one question to consider is *how to introduce the "nonlinearity" part that is not included in the linear case correction.* In other words, how is it possible to *get some estimator-dependent information into the correction.* In this chapter the correcting information is borrowed from the available resample spaces conditioned by the realised sample $\underline{k}_a$, and in some cases from the second-phase resample spaces conditioned by the resample $\underline{k}_b$. The scaling method here is external. In all, this chapter provides three different correction methods for variance estimation. Further, the problems related to resample sizes in this situation are dealt with to some extent.

6.1. *Using Resampling for Bias Correction*

What would be the best coefficient to use in order to correct the conditional variance $V(\hat{\theta}_b \mid \underline{k}_a)$ as far as the bias of the estimator is concerned? The optimal choice is

$$Q = \frac{V(\hat{\theta}_a)}{V(\hat{\theta}_b \mid \underline{k}_a)}, \tag{6.1.1.}$$

where we always get the correct parameter $QV(\hat{\theta}_b \mid \underline{k}_a) = V(\hat{\theta}_a)$, i.e. there is no variation. This coefficient is theoretical and accurate in its correction, so it is not an estimator. A more general alternative, familiar from (4.3.1), is $Q_E = \frac{V(\hat{\theta}_a)}{E_a[V(\hat{\theta}_b \mid \underline{I}_a)]}$, which contains a constant in the denominator as well. This is a theoretical coefficient, and with this coefficient the estimator is unbiased, i.e. $Q_E E_a[V(\hat{\theta}_b \mid \underline{I}_a)] = V(\hat{\theta}_a)$.

The linear case correction $\hat{Q}_{lin} = \dfrac{\hat{V}(\bar{y}_a)}{V(\bar{y}_b \mid \underline{k}_a)}$ from (3.6.1) is widely used, and the coefficient $\hat{Q}_{lin}$ does not depend on the study variable y. The bias of the variance estimator based on the linear case correction can be expressed as

$$B_a[\hat{V}_{lin}(\hat{\theta})] = \hat{Q}_{lin} E_a[V(\hat{\theta}_b \mid I_a)] - Q_E E_a[V(\hat{\theta}_b \mid I_a)] \qquad (6.1.2.)$$

$$= \left[\frac{V(\bar{y}_a)}{E_a[V(\bar{y}_b \mid \underline{I}_a)]} - \frac{V(\hat{\theta}_a)}{E_a[V(\hat{\theta}_b \mid \underline{I}_a)]} \right] E_a[V(\hat{\theta}_b \mid I_a)]. \qquad (6.1.3.)$$

Resampling in two phases. Can we estimate with resampling the theoretical coefficient $Q_E = \dfrac{V(\hat{\theta}_a)}{E_a[V(\hat{\theta}_b \mid \underline{k}_a)]}$ from (4.3.1) and produce an unbiased variance estimator? As we know from (3.6.1), *in the linear case* we have $Q_E = \hat{Q}_{lin} = \dfrac{\hat{V}(\bar{y}_a)}{V(\bar{y}_b \mid \underline{k}_a)}$ with many first and second phase designs. Unfortunately, there is no strict alternative $\hat{Q}_E = \dfrac{\hat{V}(\hat{\theta}_a)}{V(\hat{\theta}_b \mid \underline{k}_a)}$, because in fact the numerator $\hat{V}(\hat{\theta}_a)$ is the goal of our efforts. Taylor's linearisation method (see Section 3.3) will not help either, because in that case both numerator and denominator are terms from the *linear(ised) case* producing $\hat{Q}_{lin}$ as well.

The first correction solution is to conduct resampling in two phases. The estimator of the coefficient is then of the form

$$\hat{Q}_E = \frac{V(\hat{\theta}_b \mid \underline{k}_a)}{E_b[V(\hat{\theta}_c \mid \underline{I}_b) \mid \underline{k}_a]}. \qquad (6.1.4.)$$

In general, the correction method using second-phase resampling is laborious and impractical, unless there is a computer program designed for that purpose. However, (6.1.4) leads us to a *shortcut assumption* that avoids complex second-phase resampling. That procedure is presented in the next section.

Resampling with two different resample sizes. Let us assume that

$$V_b[E(\hat{\theta}_c \mid \underline{I}_b) \mid \underline{k}_a] = V(\hat{\theta}_b \mid \underline{k}_a) , \tag{6.1.5.}$$

so that in practice the bias $B(\hat{\theta}_c \mid \underline{k}_b) = E(\hat{\theta}_c \mid \underline{k}_b) - \hat{\theta}_b$ is considered to be non-existant. Furthermore, let the resample sizes be $n_b > n_c$. If the first and second resampling designs are the same (e.g. SIR / SIR), this provides us with

$$E_b[V(\hat{\theta}_c \mid \underline{I}_b) \mid \underline{k}_a] = V(\hat{\theta}_c \mid \underline{k}_a) - V_b[E(\hat{\theta}_c \mid \underline{I}_b) \mid \underline{k}_a] . \tag{6.1.6.}$$

Then the coefficient in (6.1.4) has the form

$$\hat{Q}_E = \frac{\hat{Q}_{lin,ac} V(\hat{\theta}_b \mid \underline{k}_a)}{V(\hat{\theta}_c \mid \underline{k}_a) - V_b[E(\hat{\theta}_c \mid \underline{I}_b) \mid \underline{k}_a]} . \tag{6.1.7.}$$

The assumption in (6.1.5) and the form of the coefficient in (6.1.7) give us another coefficient

$$\hat{Q}_E = \frac{\hat{Q}_{lin,ac} V(\hat{\theta}_b \mid \underline{k}_a)}{V(\hat{\theta}_c \mid \underline{k}_a) - V(\hat{\theta}_b \mid \underline{k}_a)} \tag{6.1.8.}$$

$$= \hat{Q}_{lin,ac} \Big/ \left(\frac{V(\hat{\theta}_c \mid \underline{k}_a)}{V(\hat{\theta}_b \mid \underline{k}_a)} - 1 \right), \tag{6.1.9.}$$

with conditions from (6.1.6). This procedure avoids second-phase resampling, and now it is enough to have resampling with two different resample sizes. When we study the coefficient (6.1.9) it can be seen that $V(\hat{\theta}_c \mid \underline{k}_a) > V(\hat{\theta}_b \mid \underline{k}_a)$ is a requirement for the coefficient to behave properly. This may not be achieved with some complex estimators, see e.g. the case of the maximum in simulations in Chapter 8.

Deciding on resample sizes. Some questions still remain open: 1) what is the resample size n_b? 2) what is the second phase resample / second resample size n_c? 3) what is the second phase resampling design? In simulations in Chapter 8 we have either a) $n_b = n_a - 1$ & $n_c = n_a - 2$ or b) $n_{b,I}$ is the integer part of n_b (not necessary an integer) fulfilling $\hat{Q}_{lin} = 1$ & $n_c = n_{b,I} - 1$. Other options are presented next.

Let us assume the sample size n_b (not necessary an integer) is such that we have the situation $\hat{Q}_{lin} = 1$ (see Section 3.6). Then with this sample size we can solve for n_c in

$$\frac{V(\bar{y}_b \mid \underline{k}_a)}{E_b[V(\bar{y}_c \mid \underline{I}_b) \mid \underline{k}_a]} = 1 . \tag{6.1.10.}$$

For example, with SI sampling in both phases we get $\left(\frac{1}{n_b} - \frac{1}{n_a}\right) \Big/ \left(\frac{1}{n_c} - \frac{1}{n_b}\right) = 1$ which gives us $n_c = \frac{2n_a - n_b}{n_a n_b}$. Finally calculate (6.1.7) with n_b and n_c by using either randomisation or post-design vectors, if the sizes are not integers. The second-phase resampling design can be such that it allows calculation of $n_{c,0}$ independently of the study variable y.

When we have chosen $\hat{Q}_{lin} \neq 1$ (e.g. in practice the $n - 1$ *jackknife*), the conditional variance $V(\hat{\theta}_b \mid \underline{k}_a)$ is *not* an estimator of $V(\hat{\theta}_a)$. One alternative is to use the form $\hat{V}(\hat{\theta}_a) = \hat{Q}_{lin} V(\hat{\theta}_b \mid \underline{k}_a)$. Correspondingly, an estimator of $V(\hat{\theta}_b \mid \underline{k}_a)$ could then be

$$\hat{V}(\hat{\theta}_b \mid \underline{k}_a) = \hat{Q}_{lin,bc} E_b[V(\hat{\theta}_c \mid \underline{I}_a) \mid \underline{k}_a] , \tag{6.1.11.}$$

where $\hat{Q}_{lin,bc} = \frac{V(\bar{y}_b \mid \underline{k}_a)}{E_b[V(\bar{y}_c \mid \underline{I}_b) \mid \underline{k}_a]}$.

In the end, the coefficient would be

$$\hat{Q}_E = \frac{\hat{V}(\hat{\theta}_a)}{\hat{V}(\hat{\theta}_b \mid \underline{k}_a)} = \frac{\hat{Q}_{lin} V(\hat{\theta}_b \mid \underline{k}_a)}{\hat{Q}_{lin,bc} E_b[V(\hat{\theta}_c \mid \underline{I}_b) \mid \underline{k}_a]} = \frac{\hat{Q}_{lin,ac} V(\hat{\theta}_b \mid \underline{k}_a)}{E_b[V(\hat{\theta}_c \mid \underline{I}_b) \mid \underline{k}_a]}, \tag{6.1.12.}$$

where $\hat{Q}_{lin,ac} = \frac{\hat{V}(\bar{y}_a)}{E_b[V(\bar{y}_c \mid \underline{I}_b) \mid \underline{k}_a]}$.

In any case the problem of finding n_c is still left. The choice in (6.1.10) with $\hat{Q}_{lin} = 1$ and $\hat{Q}_{lin,bc} = 1$ leads to a special case of (6.1.12). A more general alternative is to set

$$\hat{Q}_{lin,ac} = \frac{\hat{Q}_{lin}}{\hat{Q}_{lin,bc}} = 1 \tag{6.1.13.}$$

i.e. to remove the scaling effect caused by $\hat{Q}_{lin}$.

6.2. *Minimal MSE Coefficient and Its Estimator*

Let us study the variance of the variance estimator using a correcting coefficient $\hat{Q}$, i.e.

$$V_a[\hat{V}(\hat{\theta}_a)] = \hat{Q}^2 V_a[V(\hat{\theta}_b \mid \underline{I}_a)].$$

Let us assume that $\hat{Q}$ is chosen freely; in other words we do not apply the linear case correction to this situation. It is evident that if the coefficient decreases the variance of the variance estimator will decrease. The disadvantage of this diminishing coefficient is its expanding effect on the bias of the variance estimator. Therefore the *mean square error of the variance estimator* might be the best way to study its overall efficiency.

Let us study the parts of the mean square error of the variance estimator in more detail, including the unknown coefficient $\hat{Q}$. We have

$$\begin{aligned} MSE_a[\hat{V}(\hat{\theta}_a)] &= V_a[\hat{V}(\hat{\theta}_a)] + [E[\hat{V}(\hat{\theta}_a)] - V(\hat{\theta}_a)]^2 \\ &= \hat{Q}^2 V_a[V(\hat{\theta}_b \mid \underline{I}_a)] + [\hat{Q} E_a[V(\hat{\theta}_b \mid \underline{I}_a)] - V(\hat{\theta}_a)]^2 \end{aligned}$$

and the final form is then

$$\begin{aligned} MSE[\hat{V}(\hat{\theta}_a)] = \hat{Q}^2 V_a[V(\hat{\theta}_b \mid \underline{I}_a)] + \hat{Q}^2 E_a[V(\hat{\theta}_b \mid \underline{I}_a)]^2 \\ - 2\hat{Q} E_a[V(\hat{\theta}_b \mid \underline{I}_a)] V(\hat{\theta}_a) + V(\hat{\theta}_a)^2. \end{aligned} \tag{6.2.1.}$$

When we take the derivative with respect to $\hat{Q}$ in (6.2.1) and set it equal to zero, we get

$$\begin{aligned} & 2\hat{Q} V_a[V(\hat{\theta}_b \mid \underline{I}_a)] + 2\hat{Q} E_a[V(\hat{\theta}_b \mid \underline{I}_a)]^2 - 2E_a[V(\hat{\theta}_b \mid \underline{I}_a)] V(\hat{\theta}_a) = 0 \\ \Leftrightarrow \quad & \hat{Q}\{V_a[V(\hat{\theta}_b \mid \underline{I}_a)] + E_a[V(\hat{\theta}_b \mid \underline{I}_a)]^2\} = E_a[V(\hat{\theta}_b \mid \underline{I}_a)] V(\hat{\theta}_a), \end{aligned}$$

and then the theoretical coefficient producing the minimal mean square error of the variance estimator is of the form

$$Q_{\min} = \frac{E_a[V(\hat{\theta}_b \mid \underline{I}_a)] V(\hat{\theta}_b)}{V_a[V(\hat{\theta}_b \mid \underline{I}_a)] + E_a[V(\hat{\theta}_b \mid \underline{I}_a)]^2} \tag{6.2.2.}$$

or alternatively

$$Q_{\min} = \frac{V(\hat{\theta}_a)}{\dfrac{V_a[V(\hat{\theta}_b \mid \underline{I}_a)]}{E_a[V(\hat{\theta}_b \mid \underline{I}_a)]} + E_a[V(\hat{\theta}_b \mid \underline{I}_a)]}. \tag{6.2.3.}$$

In terms of Q_E in (4.3.1) the formula (6.2.3) is then

$$Q_{\min} = \frac{Q_E}{\frac{V_a[V(\hat{\theta}_b \mid \underline{I}_a)]}{\{E_a[V(\hat{\theta}_b \mid \underline{I}_a)]\}^2} + 1} = \frac{Q_E}{\frac{E_a[V(\hat{\theta}_b \mid \underline{I}_a)^2] - \{E_a[V(\hat{\theta}_b \mid \underline{I}_a)]\}^2}{\{E_a[V(\hat{\theta}_b \mid \underline{I}_a)]\}^2} + 1}$$

and finally we have

$$Q_{\min} = \frac{\{E_a[V(\hat{\theta}_b \mid \underline{I}_a)]\}^2 \cdot Q_E}{E_a[V(\hat{\theta}_b \mid \underline{I}_a)^2]}. \qquad (6.2.4.)$$

It can be seen from (6.2.4) that $Q_{\min} \leq Q_E$ and hence the theoretical variance estimator with $Q_{\min}$ is *always* smaller than or equal to the theoretical unbiased variance estimator. Furthermore, the bias of the variance estimator with $Q_{\min}$ is always negative or zero.

As in Section 6.1, we can try to use resampling in two phases for the estimator of the MSE minimising coefficient (6.2.4),

$$\hat{Q}_{\min} = \frac{\{E_b[V(\hat{\theta}_c \mid \underline{I}_b) \mid \underline{k}_a]\}^2 \cdot \hat{Q}_{lin}}{E_b[V(\hat{\theta}_c \mid \underline{I}_b)^2 \mid \underline{k}_a]}. \qquad (6.2.5.)$$

In the simulations in Chapter 8 we have the same choices of resample sizes as in Section 6.1.

6.3. *Summary*

In this chapter we provide alternatives to the linear case coefficient $\hat{Q}_{lin}$ used in almost all resampling variance estimation methods. The idea is to introduce an *estimator-dependent nonlinearity part* to the correction. The first variance estimation method for estimating the unbiased expectation coefficient Q_E uses information from *resampling in two phases*. There are different principles for choosing the resample sizes; here two of them are presented. A more practical shortcut for this method is *resampling with two resample sizes* with the assumption $V_b[E(\hat{\theta} \mid \underline{I}_b) \mid \underline{k}_a] = V(\hat{\theta}_b \mid \underline{k}_a)$. However, this assumption may not be valid in very complex cases. Another way of dealing with the problem of variance estimation is to *estimate the theoretical minimal mean square error coefficient* Q_{min}. As in the first estimation method, resampling in two phases is also used in this case for estimation purposes. As the theory indicates, we obtain lower coefficient values than for the $\hat{Q}_{lin}$ coefficient. Chapter 8 provides some promising results for the $\hat{Q}_E$ variance estimators in particular.

7. TWO-PHASE SAMPLING DESIGN IN VARIANCE ESTIMATION

When we are using resampling for estimating $V(\hat{\theta}_a)$ – the variance of $\hat{\theta}_a$ with respect to the first phase – our main problem is the *incorrect scale of estimates* $\hat{\theta}_b$ provided within the resampling design, which differs from the first-phase sampling design. The estimates are conditional to the realised sample. Early in this chapter some results are presented for two-phase sampling and distributions. Later the *sample pairs* and their *probability distributions* at both sampling and resampling levels are considered in detail. The categorising criterion for these pairs is the *number of distinct observations*.

The scale problem is dealt with in two different ways here. The idea of the first method is to assume that $E(\hat{\theta}_b \mid \underline{k}_a) = \hat{\theta}_a$ and then *use the conditionality theory* for the construction of the variance estimator. Another solution is to *adjust the resample pair probabilities* (considered as weights here) by using conditions from the linear case. For this purpose some results are presented for the decomposed variance of the mean, i.e. the linear case. The idea is to find a function of the n_b resample pair probabilities that fulfills the conditions of the n_a resample pair probabilities. Finally, the resamples of size n_b are utilised with these weight conditions in order to reach an unbiased variance estimator in the linear case.

7.1. Unbiasedness Assumption of the Resampling Estimator

In two-phase sampling situations we use in the second phase the estimator $\hat{\theta}_b$, whose variance with respect to the two phases $V_*(\cdot)$ can be expressed as

$$V_*(\hat{\theta}_b) = V_a[E(\hat{\theta}_b \mid \underline{I}_a)] + E_a[V(\hat{\theta}_b \mid \underline{I}_a)] . \qquad (7.1.1.)$$

Let us assume that $\hat{\theta}_b$ in the second phase is unbiased for $\hat{\theta}_a$, i.e.

$$E(\hat{\theta}_b \mid \underline{k}_a) = \hat{\theta}_a . \qquad (7.1.2.)$$

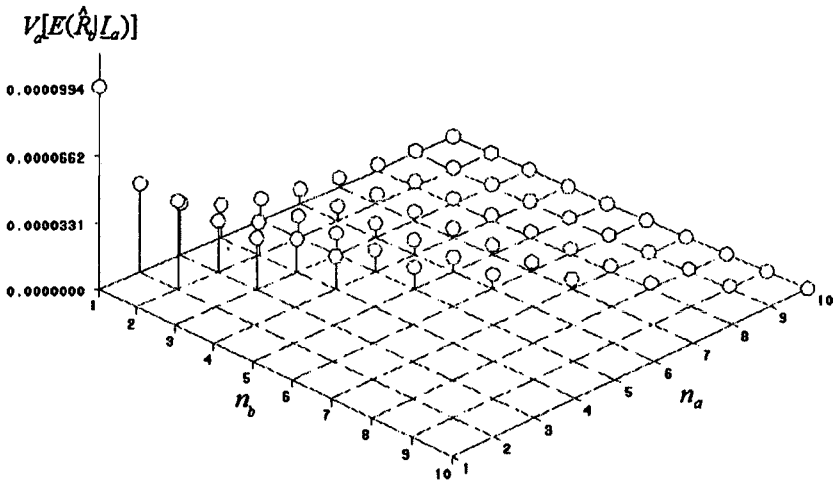
Then our target of study can be expressed from (7.1.1) as

$$V_a(\hat{\theta}_b) = V_*(\hat{\theta}_b) - E_a[V(\hat{\theta}_b \mid \underline{I}_a)] . \qquad (7.1.3.)$$

To estimate $V_a(\hat{\theta}_a)$ we need to estimate both terms in (7.1.3). The second term is estimated without bias by the conditional variance $V(\hat{\theta}_b \mid \underline{I}_a)$, which, in turn, can be well estimated by repeated sampling from the sample at hand, $\underline{k}_a$.

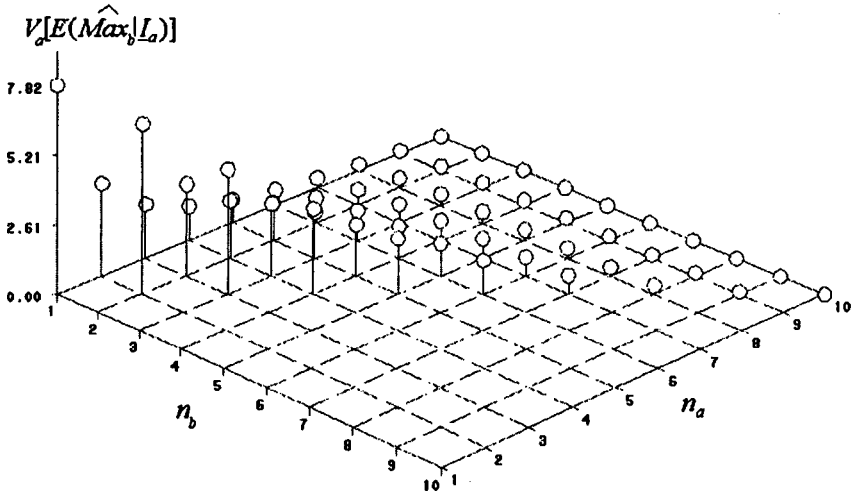
Consequently, the problem is in estimating $V_*(\hat{\theta}_b)$. Some possibilities are studied below. We point out that even if (7.1.2) does not hold, $V_a[E(\hat{\theta}_b \mid \underline{I}_a)]$ approximates $V_a(\hat{\theta}_a)$ well. An example is the ratio in SI sampling, shown in Figure 7.1.

Figure 7.1. Variance of the Conditional Expectation of the Estimator of the Ratio ($N = 10$)



It can easily be seen in the figure that when n_b varies from 1 to n_a, $V_a[E(\hat{R}_b \mid \underline{I}_a)]$ remains constant. The real variance $V(\hat{R}_a)$ can be found in the case $n_a = n_b$. On the other hand, the non-smooth functions may still be rather difficult to deal with, see for example the behaviour of the maximum in SI sampling in Figure 7.2.

Figure 7.2. Variance of the Conditional Expectation of the Estimator of the Maximum ($N = 10$)



The variance $V_a[E(\hat{Max}_b \mid \underline{I}_a)]$ is not stable when n_b varies from 1 to n_a.

However, the only variance available in the second phase is $V(\hat{\theta}_b \mid \underline{I}_a)$. From (7.1.1) it is easy to see that $V_*(\hat{\theta}_b) \geq E_a[V(\hat{\theta}_b \mid \underline{I}_a)]$. Thus $V(\hat{\theta}_b \mid \underline{I}_a)$ is not applicable as such for estimating $V_*(\hat{\theta}_b)$ without bias. The idea is to alter $V(\hat{\theta}_b \mid \underline{I}_a)$ (denoted here by V_{alter}) so that $V_*(\hat{\theta}_b) = E_a(V_{alter})$. This is performed via the *decomposition of the variance* and the *sample pair probability distributions*. First the basis of two-phase sampling is presented along with some new results.

7.2. On the Two-phase Sampling Design with New Results

In Traat et al. (2001) the two-phase design is studied using the vector approach. This approach allows us to combine WOR and WR designs simultaneously in the phases of the two-phase design. The two-phase design probability $p^*(\cdot)$ is expressed as

$$p^*(\underline{k}_b) = \sum_{\underline{k}_a(\underline{k}_b)} p(\underline{k}_b \mid \underline{k}_a) p(\underline{k}_a) = E_a[p(\underline{k}_b \mid \underline{I}_a)] \qquad (7.2.1.)$$

where $\sum_{\underline{k}_a(\underline{k}_b)}$ means that the summation is taken over such $\underline{k}_a$ from which one can select $\underline{k}_b$.
An interesting result we present here concerns SI sampling in the second phase.

Let the two-phase design be composed with an arbitrary size n_a and design $p(\underline{k}_a)$ in the first phase and an SI-design in the second phase. We intend to perform SI sampling on the ordered sample so that the repeats are handled as separate sampling units. For example, if unit $i \in U$ appears 3 times in the first phase sample ($k_{ai} = 3$), then after SI sampling it can occur 0, 1, 2 or 3 times in the final sample (provided $n_b \geq 3$). As a result SI sampling in the ordered sample is the same as the hypergeometric sample. Therefore the second-phase design has the form

$$p(\underline{k}_b \mid \underline{k}_a) = \prod_{i=1}^{N} \binom{k_{ai}}{k_{bi}} \Big/ \binom{n_a}{n_b}.$$

Then

$$p^*(\underline{k}_b) = \binom{n_a}{n_b}^{-1} \sum_{\underline{k}_a(\underline{k}_b)} \prod_{i=1}^{N} \binom{k_{ai}}{k_{bi}} p(\underline{k}_a), \qquad (7.2.2.)$$

where the sum in (7.2.2) is the inclusion probability of a specified sample $\underline{k}_b$ in the first-phase sample $\underline{k}_a$. Note that for WOR designs ($k_{ai} \in \{0,1\}$) the combinatorial term vanishes, as $\prod_{i=1}^{N} \binom{k_{ai}}{k_{bi}} = 1$.

Letting

$$\sum_{\underline{k}_a(\underline{k}_b)} \prod_{i=1}^{N} \binom{k_{ai}}{k_{bi}} p(\underline{k}_a) = p(\underline{k}_b)\,, \tag{7.2.3.}$$

we have

$$p^*(\underline{k}_b) = \binom{n_a}{n_b}^{-1} p(\underline{k}_b)\,. \tag{7.2.4.}$$

Note that $p(\underline{k}_b)$ here is not a sampling design on samples $\underline{k}_b$.

Let us look at an example of the various probabilities involved. Let $N = 5$, $n_a = 4$ and $n_b = 3$. Let the sampling design in the first phase be WOR, with all of its samples and their probabilities. The sample space with $p(\underline{k}_a) > 0$ is denoted by $\aleph_a$. Table 7.1 presents this situation.

Table 7.1. First-Phase Sampling Design

$\aleph_a$	U					$p(\underline{k}_a)$
	1	2	3	4	5	
$\underline{k}_{a,1}$	1	1	1	1	0	0.1
$\underline{k}_{a,2}$	1	1	1	0	1	0.2
$\underline{k}_{a,3}$	1	1	0	1	1	0.3
$\underline{k}_{a,4}$	1	0	1	1	1	0.3
$\underline{k}_{a,5}$	0	1	1	1	1	0.1
sum						1

All possible samples $\underline{k}_b$ of the two-phase design, their inclusion probabilities $p(\underline{k}_b)$ in the first-phase sample, their design probabilities $p^*(\underline{k}_b)$, and the sample space $\aleph_b^*$ for $p^*(\underline{k}_b) > 0$ are given in Table 7.2.

Table 7.2. Two-Phase Sampling Design

$\aleph_b^*$	U					$p(\underline{k}_b)$	$p^*(\underline{k}_b)$
	1	2	3	4	5		
$\underline{k}_{b,1}$	1	1	1	0	0	0.3	3 / 40
$\underline{k}_{b,2}$	1	1	0	1	0	0.4	4 / 40
$\underline{k}_{b,3}$	1	1	0	0	1	0.5	5 / 40
$\underline{k}_{b,4}$	1	0	1	1	0	0.4	4 / 40
$\underline{k}_{b,5}$	1	0	1	0	1	0.5	5 / 40
$\underline{k}_{b,6}$	1	0	0	1	1	0.6	6 / 40
$\underline{k}_{b,7}$	0	1	1	1	0	0.2	2 / 40
$\underline{k}_{b,8}$	0	1	1	0	1	0.3	3 / 40
$\underline{k}_{b,9}$	0	1	0	1	1	0.4	4 / 40
$\underline{k}_{b,10}$	0	0	1	1	1	0.4	4 / 40
sum						4	1

Probabilities $p(\underline{k}_b)$ are based on (7.2.3) and $p^*(\underline{k}_b)$ on (7.2.4), where $\binom{n_a}{n_b}^{-1} = \binom{4}{3}^{-1} = \frac{1}{4}$.

The sample space for $p^*(\underline{k}_b)$ consists of all possible 3-unit samples which can be taken from 4 (there are 10 altogether). Thus $p^*(\underline{k}_b)$ can also be thought of as a one-phase design on the samples of size 3 from U.

As a new result we show below that multinomial design in the first phase and SI design in the second phase composes a two-phase design which is again multinomial. Let $p(\underline{k}_a)$ be $M(n_a; p_1, \ldots, p_N)$, i.e. a classical WR design with sample size n_a and selection probabilities p_i. Then

$$p(\underline{k}_a) = n_a! \prod_{i=1}^{N} \frac{p_i^{k_{ai}}}{k_{ai}!}$$

for $\sum_{i=1}^{N} k_{ai} = n_a$.

Let $p(\underline{k}_b \mid \underline{k}_a)$ be SI with sample size $n_b \le n_a$

$$p(\underline{k}_b \mid \underline{k}_a) = \binom{n_a}{n_b}^{-1} \prod_{i=1}^{N} \binom{k_{ai}}{k_{bi}},$$

where $\sum_{i=1}^{N} k_{ai} = n_a$ and $\sum_{i=1}^{N} k_{bi} = n_b$. Now the two-phase design (7.2.2) takes the form

$$p^*(\underline{k}_b) = \binom{n_a}{n_b}^{-1} n_a! \sum_{\underline{k}_a(\underline{k}_b)} \prod_{i=1}^{N} \binom{k_{ai}}{k_{bi}} \frac{p_i^{k_{ai}}}{k_{ai}!},$$

and after developing the combinatorial terms we have

$$p^*(\underline{k}_b) = n_b!(n_a - n_b)! \sum_{\underline{k}_a(\underline{k}_b)} \prod_{i=1}^{N} \frac{p_i^{k_{ai}}}{k_{bi}!(k_{ai} - k_{bi})!}.$$

Writing $p_i^{k_{ai}} = p_i^{k_{bi}} p_i^{k_{ai}-k_{bi}}$ and taking the terms not dependent on $\underline{k}_a$ out of the summation sign we have

$$p^*(\underline{k}_b) = n_b! \prod_{i=1}^{N} \frac{p_i^{k_{ai}}}{k_{bi}!} \sum_{\underline{k}_a(\underline{k}_b)} (n_a - n_b)! \prod_{i=1}^{N} \frac{p_i^{k_{ai}-k_{bi}}}{(k_{ai} - k_{bi})!}. \qquad (7.2.5.)$$

Noting that the sum over all such $\underline{k}_a$ from which $\underline{k}_b$ can be selected means alternatively $\sum_{\underline{k}_a(\underline{k}_b)}(\cdot) = \sum_{\underline{k}_a - \underline{k}_b \ge 0}(\cdot)$, and denoting $\underline{k}_a - \underline{k}_b = \underline{k}_c$ with elements $k_{ci} = k_{ai} - k_{bi}$, we can see that the sum in (7.2.5) runs over all of the multinomial probabilities, and is thus equal to 1:

$$\sum_{\underline{k}_c} n_c! \prod_{i=1}^{N} \frac{p_i^{k_{ci}}}{k_{ci}!} = 1,$$

where $n_c = \sum_{i=1}^{N} k_{ci} = \sum_{i=1}^{N} (k_{ai} - k_{bi}) = n_a - n_b$.

Thus we have proved that the two-phase design is multinomial $M(n_a; p_1, \ldots, p_N)$ with

$$p^*(\underline{k}_b) = n_b! \prod_{i=1}^{N} \frac{p_i^{k_{bi}}}{k_{bi}!}$$

for $\sum_{i=1}^{N} k_{bi} = n_b$. In the special case when we have SIR with n_a in the first phase and SI with n_b in the second phase we have a two-phase SIR design with n_b.

7.3. Decomposition of Variance and Sample Pair Probabilities

Definitions. In Section 7.1 the two-phase design is studied in terms of the vector approach, and here that study is extended for sample pairs. The design-based variance of the estimator $\hat{\theta}$

$$V(\hat{\theta}) = E(\hat{\theta}^2) - [E(\hat{\theta})]^2$$

can be expressed in the form

$$V(\hat{\theta}) = \sum_{\underline{k}} \hat{\theta}_{\underline{k}}^2 p(\underline{k}) - \sum_{\underline{k},\underline{k}'} \hat{\theta}_{\underline{k}} \hat{\theta}_{\underline{k}'} p(\underline{k} \,\&\, \underline{k}') \,. \tag{7.3.1.}$$

Here $\hat{\theta}_{\underline{k}}$ denotes the estimate based on the sample $\underline{k}$. Correspondingly, we have

$$V(\hat{\theta}_b \mid \underline{k}_a) = \sum_{\underline{k}_b | \underline{k}_a} \hat{\theta}_{\underline{k}_b}^2 p(\underline{k}_b \mid \underline{k}_a) - \sum_{\underline{k}_b, \underline{k}_b' | \underline{k}_a} \hat{\theta}_{\underline{k}_b} \hat{\theta}_{\underline{k}_b'} p(\underline{k}_b \,\&\, \underline{k}_b' \mid \underline{k}_a) \tag{7.3.2.}$$

for the conditional variance.

The idea of the decomposition approach can be found when we are comparing the two-phase sampling probability $p^*(\underline{k}_b \,\&\, \underline{k}_b')$ and the expectation

$$E_a[p(\underline{k}_b \,\&\, \underline{k}_b' \mid \underline{I}_a)] = \sum_{\underline{k}_a(\underline{k}_b)} p(\underline{k}_a) p(\underline{k}_b \,\&\, \underline{k}_b' \mid \underline{k}_a) \,, \tag{7.3.3.}$$

where the summation over $\underline{k}_a(\underline{k}_b)$ is explained in (7.2.1). We note that they are *not* equal to (7.2.1) with $p^*(\underline{k}_b) = E_a[p(\underline{k}_b \mid \underline{I}_a)]$.

SI sampling. Let us now study *SI sampling and resampling.* We know that in this case

$$p^*(\underline{k}_b) = \binom{N - n_b}{n_a - n_b} \binom{N}{n_a}^{-1} \binom{n_a}{n_b}^{-1} = \binom{N}{n_b}^{-1} = p(\underline{k}_b) \,.$$

Consequently variances under these designs are also equal: $V_*(\hat{\theta}_b) = V(\hat{\theta}_b)$.

From the decomposition of the variance (7.3.1) we get that sample pair probabilities are equal too, i.e. $p^*(\underline{k}_b \,\&\, \underline{k}_b') = p(\underline{k}_b \,\&\, \underline{k}_b') = \binom{N}{n_b}^{-2}$. Furthermore, we have $p(\underline{k}_b \,\&\, \underline{k}_b' \,|\, \underline{k}_a) = \binom{n_a}{n_b}^{-2}$ for second-phase SI sampling pairs based on $\underline{k}_a$. Finally, from (7.3.3) the expectation of the conditional probability gives

$$E_a[p(\underline{k}_b \,\&\, \underline{k}_b' \,|\, \underline{I}_a)] = \binom{N-d}{n_a-d}\binom{N}{n_a}^{-1}\binom{n_a}{n_b}^{-2}, \qquad (7.3.4.)$$

where d is the *number of distinct observations* in the sample pair $\underline{k}_b \,\&\, \underline{k}_b'$. As an example we consider the population with $N = 7$ elements, $U = \{1, 2, 3, 4, 5, 6, 7\}$. We select a SI sample $\underline{k}_a$ of size $n_a = 5$. This is an element from the sample space $\aleph_a$, which includes all vector samples of size n_a (21 elements altogether). Then we create all second-phase SI samples $\underline{k}_b$ of size $n_b = 2$ from $\underline{k}_a$, i.e. the resample space $\aleph_{b|\underline{k}_a}$ with size 10. If we take the union $U(\aleph_{b|\underline{k}_a}) = \bigcup_{\underline{k}_a \in \aleph_a} \aleph_{b|\underline{k}_a}$ over all samples in $\aleph_a$, it is evident that no sample in $U(\aleph_{b|\underline{k}_a})$ equals any sample in $\aleph_a$ simply due to the different sample sizes. However, with SI sampling in both phases we have $U(\aleph_{b|\underline{k}_a}) = \aleph_b$ (21 two-element sets from U). In this case $U(\aleph_{b|\underline{k}_a} \times \aleph_{b|\underline{k}_a}) = \aleph_b \times \aleph_b$, i.e. the product spaces are the same. In fact, in the SI case even the estimates $\hat{\theta}_b$ in the space $\aleph_b$ and in the space $U(\aleph_{b|\underline{k}_a})$ are the same. We still know that $V(\hat{\theta}_b) \geq E_a[V(\hat{\theta}_b \,|\, \underline{I}_a)]$. There are $\binom{7}{2}^2 = 441$ pairs of samples in the space $\aleph_b \times \aleph_b$. Correspondingly, there are $\binom{7}{5}\binom{5}{2}^2 = 2100$ resample pairs altogether, if all first-phase samples are considered. *In this resample-pair set some pairs appear more frequently than others.* For example, for (1,2) & (1,2) there are $\binom{7-2}{5-2} = 10$ samples $\underline{k}_a$ that include the units 1 and 2; for (1,2) & (1,3) there are $\binom{7-3}{5-3} = 6$ $\underline{k}_a$*'s* that include the

units 1, 2 and 3; for (1,2) & (3,4) there are $\binom{7-4}{5-4}=3$ $\underline{k}_a$'s that include the units 1, 2, 3 and 4.

Thus the key issue here is the *number of distinct observations* of the pair. *These different distributions of* $p(\underline{k}_b \,\&\, \underline{k}_b')$ and $E_a[p(\underline{k}_b \,\&\, \underline{k}_b' \,|\, \underline{I}_a)]$ *are the reason that* $V_*(\hat{\theta}_b) \geq E_a[V(\hat{\theta}_b)\,|\,\underline{I}_a)]$ *in SI sampling/resampling, i.e.* $[E(\hat{\theta}_b)]^2 \leq E_a[E(\hat{\theta}_b\,|\,\underline{I}_a)^2]$, *because in this case* $E(\hat{\theta}_b^2) = E_a[E(\hat{\theta}_b^2\,|\,\underline{I}_a)]$. However, even with very simple designs we encounter many complexities, and when allowing repetitions of the units or unequal probabilities these distributions provide even more difficulties.

7.4. *Variance Estimator for Two-Phase Sampling with* n_b

Let us assume that $U(\aleph_{b|\underline{k}_a}) = \aleph_b$, i.e. the union of all resample spaces with n_b equals the sample space with n_b. This requirement is not fulfilled for example when WR resampling is carried out from a WOR sample or we have $2n_b > n_a$. In the latter case there are not enough distinct observations in the sample $\underline{k}_a$ for estimating some pairs $\underline{k}_b \,\&\, \underline{k}_b'$ in the sample space, i.e. the pairs with $d > n_a$ cannot be estimated strictly. Further, let us assume that the estimates are equal, i.e. $U(\vartheta_{\hat{\theta}_b|\underline{k}_a}) = \vartheta_{\hat{\theta}_b}$. Under these assumptions the estimator of $V_*(\hat{\theta}_b)$ is then

$$\hat{V}_*(\hat{\theta}_b) = \sum_{\underline{k}_b|\underline{k}_a} \hat{\theta}_{\underline{k}_b}^2\, p(\underline{k}_b\,|\,\underline{k}_a)\frac{p^*(\underline{k}_b)}{E_a[p(\underline{k}_b\,|\,\underline{k}_a)]} - \sum_{\underline{k}_b,\underline{k}_b'|\underline{k}_a} \hat{\theta}_{\underline{k}_b}\hat{\theta}_{\underline{k}_b'}\, p(\underline{k}_b \,\&\, \underline{k}_b'\,|\,\underline{k}_a)\frac{p^*(\underline{k}_b \,\&\, \underline{k}_b')}{E_a[p(\underline{k}_b \,\&\, \underline{k}_b'\,|\,\underline{k}_a)]}, \tag{7.4.1.}$$

where * refers to two-phase sampling. For example, in the SI / SI case we have $E_a[\hat{V}_*(\hat{\theta}_b)] = V_*(\hat{\theta}_b)$, so the variance estimator is unbiased. Let us study the linear case with the estimator of the population mean in the SI / SI case with $N = 7$, $n_a = 4$, and $n_b = 2$. Let the values of the variable y in the sample $\underline{k}_a$ be 1, 3, 4 and 6. Then we have $\bar{y}_{\underline{k}_a} = 3.5$, $\overline{y^2}_{\underline{k}_a} = 12.25$, $\overline{y^2}_{\underline{k}_a} = 15.5$, $\overline{yy'}_{\underline{k}_a} = 11.166667$ and $\hat{S}_y^2 = \overline{y^2}_{\underline{k}_a} - \overline{yy'}_{\underline{k}_a} = 4.333333$. The analytic form of the variance estimator with n_b is

$$\hat{V}(\bar{y}_b) = \left(\frac{1}{n_b} - \frac{1}{N}\right)\hat{S}_y^2 = \left(\frac{1}{2} - \frac{1}{7}\right)\cdot 4.333333 = 1.548. \tag{7.4.2.}$$

Correspondingly, we have the variance estimator (7.4.1), where $\frac{p^*(\underline{k}_b)}{E_a[p(\underline{k}_b \mid \underline{k}_a)]} = 1$,

$p(\underline{k}_b \mid \underline{k}_a) = 1/6$, $p(\underline{k}_b \,\&\, \underline{k}_b' \mid \underline{k}_a) = 1/36$, $p^*(\underline{k}_b \,\&\, \underline{k}_b') = 1/441$, and

$$E_a[p(\underline{k}_b \,\&\, \underline{k}_b' \mid \underline{k}_a)] = \binom{7-d}{4-d} \Big/ \binom{7}{4}\binom{4}{2}^2,$$

where d is the number of distinct units in the pair $\underline{k}_b \,\&\, \underline{k}_b'$. Thus

- $E_a[p(\underline{k}_b \,\&\, \underline{k}_b' \mid \underline{k}_a)] = 1/1260$ for $d = 4$,
- $E_a[p(\underline{k}_b \,\&\, \underline{k}_b' \mid \underline{k}_a)] = 4/1260$ for $d = 3$, and
- $E_a[p(\underline{k}_b \,\&\, \underline{k}_b' \mid \underline{k}_a)] = 10/1260$ for $d = 2$.

Finally, the sums $Z_d = \sum_{\underline{k}_b \,\&\, \underline{k}_b' \mid d} \bar{y}_{\underline{k}_b} \bar{y}_{\underline{k}_b'}$ in different groups of distinct units are

- $Z_2 = 80$,
- $Z_3 = 294$,
- $Z_4 = 67$.

Now we get the variance estimate

$$\hat{V}_*(\bar{y}_b) = 80 \cdot \frac{1}{6} - \left[80 \cdot \frac{1}{36} \cdot \frac{1/441}{10/1260} + 294 \cdot \frac{1}{36} \cdot \frac{1/441}{4/1260} + 67 \cdot \frac{1}{36} \cdot \frac{1/441}{1/1260} \right] = 1.548.$$

Consequently, the estimator (7.4.1) produces the same value as the unbiased estimator given in (7.4.2). In practice, the probabilities $p^*(\underline{k}_b \,\&\, \underline{k}_b')$ and $E_a[p(\underline{k}_b \,\&\, \underline{k}_b' \mid \underline{k}_a)]$ are very laborious to calculate for all pairs $\underline{k}_b \,\&\, \underline{k}_b'$, unless the designs are rather simple, such as the combinatorial probabilities which occur in the case of SI sampling.

7.5. *Variance Estimator Based on Unbiasedness Assumption*

For resamples with $2n_b \le n_a$ let us use the estimator of $V_*(\hat{\theta}_b)$ from (7.4.1) together with the unbiasedness assumption (7.1.2) to develop an estimator for the variance in (7.1.3),

$$\hat{V}(\hat{\theta}) = \hat{V}_*(\hat{\theta}_b) - V(\hat{\theta}_b \mid \underline{k}_a).$$

Inserting (7.4.1) and (7.3.2) we have

$$\begin{aligned}\hat{V}(\hat{\theta}_a) = &\sum_{\underline{k}_b \mid \underline{k}_a} \hat{\theta}^2_{\underline{k}_b} p(\underline{k}_b \mid \underline{k}_a)\left(\frac{p^*(\underline{k}_b)}{E_a[p(\underline{k}_b \mid \underline{k}_a)]} - 1\right) \\ &- \sum_{\underline{k}_b, \underline{k}_b' \mid \underline{k}_a} \hat{\theta}_{\underline{k}_b} \hat{\theta}_{\underline{k}_b'} p(\underline{k}_b \,\&\, \underline{k}_b' \mid \underline{k}_a)\left(\frac{p^*(\underline{k}_b \,\&\, \underline{k}_b')}{E_a[p(\underline{k}_b \,\&\, \underline{k}_b' \mid \underline{k}_a)]} - 1\right)\end{aligned} \qquad (7.5.1.)$$

Finally, if $p^*(\underline{k}_b) = E_a[p(\underline{k}_b \mid \underline{k}_a)]$ we have the form

$$\hat{V}(\hat{\theta}_a) = \sum_{\underline{k}_b, \underline{k}_b' \mid \underline{k}_a} \hat{\theta}_{\underline{k}_b} \hat{\theta}_{\underline{k}_b'} p(\underline{k}_b \,\&\, \underline{k}_b' \mid \underline{k}_a)\left(1 - \frac{p^*(\underline{k}_b \,\&\, \underline{k}_b')}{E_a[p(\underline{k}_b \,\&\, \underline{k}_b' \mid \underline{k}_a)]}\right) \qquad (7.5.2.)$$

When the variance estimator $\hat{V}_*(\hat{\theta}_b)$ is unbiased (e.g. SI/SI), taking the expectation of the estimator

$$\begin{aligned}E_a[\hat{V}(\hat{\theta}_a)] &= E_a[\hat{V}_*(\hat{\theta}_b)] - E_a[V(\hat{\theta}_b \mid \underline{I}_a)] \\ &= V_*(\hat{\theta}_b) - E_a[V(\hat{\theta}_b \mid \underline{I}_a)],\end{aligned}$$

and using (7.1.1) we get

$$E_a[\hat{V}(\hat{\theta}_a)] = V_a[E(\hat{\theta}_b \mid \underline{I}_a)], \qquad (7.5.3.)$$

and this shows that we have an unbiased estimator of $V_a[E(\hat{\theta}_b \mid \underline{I}_a)]$. Thus the validity of the assumption $E(\hat{\theta}_b \mid \underline{k}_a) = \hat{\theta}_a$ is essential. In Chapter 8 the simulations reveal that for the population variance this approach creates an unbiased variance estimator, even for small population sizes.

7.6. *Variance Estimator Based on Weight Adjustments*

Decomposed variance taking the number of joint observations into account. The *number of joint observations*, r_a, in the sample pair $(\underline{k}_a, \underline{k}_a')$ is the key property of the variance estimator we develop below. We could use the number of distinct observations, $d = 2n_a - r_a$, as before, but for the sake of simplicity in the formulae, we concentrate here on r_a. The probability distribution of the number of joint observations is given by $p(r_a) = \sum_{\underline{k}_a, \underline{k}_a' | r_a} p(\underline{k}_a, \underline{k}_a')$, where the summation takes place over such pairs $(\underline{k}_a, \underline{k}_a')$ which have r_a joint units. For example, with SI sampling we have $p(r_a) = \binom{N}{r_a}\binom{N-r_a}{n_a-r_a}\Big/\binom{N}{n_a}^2$. From the variance expression (7.3.1) we now get

$$V(\hat{\theta}_a) = E(\hat{\theta}_a^2) - \sum_{r_a=\max(0,2n_a-N)}^{n_a} p(r_a) \cdot E(\hat{\theta}_{\underline{k}_a} \hat{\theta}_{\underline{k}_a'} \mid r_a), \qquad (7.6.1.)$$

where

$$E(\hat{\theta}_{\underline{k}_a} \hat{\theta}_{\underline{k}_a'} \mid r_a) = \sum_{\underline{k}_a, \underline{k}_a' | r_a} \hat{\theta}_{\underline{k}_a} \hat{\theta}_{\underline{k}_a'} p(\underline{k}_a, \underline{k}_a'). \qquad (7.6.2.)$$

We will first look at the decomposed variance of the sample mean (i.e. the linear case) is of interest here. This form is used later when creating the criterion for variance estimators based on the decomposed variance. Let us study the conditional expectation in (7.6.2) for the linear estimator under SI sampling, i.e. $E(\bar{y}\,\bar{y}' | r_a)$. Here the set approach is used as it is more convenient. Thus we have a sample pair s and s'. Writing $\bar{y} = \frac{1}{n}\left[\sum_{i \in s, i \notin s'} y_i + \sum_{i \in s, s'} y_i\right]$ and $\bar{y}' = \frac{1}{n}\left[\sum_{i \in s', i \notin s} y_i + \sum_{i \in s', s} y_i\right]$ we find that the product of two means has the structure

$$\bar{y}\,\bar{y}' = \frac{1}{n^2}\left[\sum_{i \in s,s'} y_i^2 + \sum_{i \neq l \in s,s'} y_i y_l + \sum_{i \in s,s'} \sum_{j \in s, j \notin s'} y_i y_j + \sum_{i \in s,s'} \sum_{k \in s', k \notin s} y_i y_k + \sum_{j \in s, j \notin s'} y_j \sum_{k \in s', k \notin s} y_k\right]. \qquad (7.6.3.)$$

The expectation (7.6.2) is then (for notational simplicity we use r instead of r_a)

$$E(\bar{y}\,\bar{y}'\mid r)=\frac{1}{n^2}\Big(\sum_{i\in s,s'}E(y_i^2\mid r)+\sum_{i\in s,s'}\sum_{l\in s,s'}E(y_iy_l\mid r)$$
$$+\sum_{i\in s,s'}\sum_{j\in s,\,j\notin s'}E(y_iy_j\mid r)+\sum_{i\in s,s'}\sum_{k\notin s',k\notin s}E(y_iy_k\mid r)+\sum_{j\in s,\,j\notin s'}\sum_{k\in s',k\notin s}E(y_jy_k\mid r)\Big)$$
$$=\frac{1}{n^2}\Big[r\,\overline{y_U^2}+\big(r(r-1)+r(n-r)+r(n-r)+(n-r)^2\big)\overline{y\,y'}_U\Big]$$

where y is a variable in the population with values y_i that have probabilities $1\,/\,N$, $\overline{y_U^2}=\sum_{i\in U}y_i^2/N$, $\overline{y\,y'}_U=\sum_{i\neq j\in U}y_iy_j/[N(N-1)]$, and r is the size of the union of s and s', i.e. the *number of joint observations*. Finally we get the result

$$E(\bar{y}\,\bar{y}'\mid r)=\frac{r}{n^2}\,\overline{y_U^2}+\left(1-\frac{r}{n^2}\right)\overline{y\,y'}_U\,. \tag{7.6.4.}$$

We now have

$$V(\bar{y})=\left[\frac{1}{n}\,\overline{y_U^2}+\left(1-\frac{1}{n}\right)\overline{y\,y'}_U\right]-\sum_r p(r)\left[\frac{r}{n^2}\,\overline{y_U^2}+\left(1-\frac{r}{n^2}\right)\overline{y\,y'}_U\right]$$
$$=\left[\frac{1}{n}-\frac{\sum_r p(r)\,r}{n^2}\right]\overline{y_U^2}+\left[\left(1-\frac{1}{n}\right)-\sum_r p(r)\left(1-\frac{r}{n^2}\right)\right]\overline{y\,y'}_U$$

and the final form is

$$V(\bar{y})=\frac{1}{n}\left[1-\frac{\sum_r p(r)\,r}{n}\right]\left(\overline{y_U^2}-\overline{y\,y'}_U\right). \tag{7.6.5.}$$

Here $\sum_r p(r)\,r$ is the expected number of joint units, which under SI becomes n^2/N.

Weight adjustments with two alternatives. The variance of any estimator $\hat{\theta}_a$ in terms of joint units is given in (7.6.1). For the conditional variance we have analogously

$$V(\hat{\theta}_b\mid \underline{k}_a)=E(\hat{\theta}_b^2\mid \underline{k}_a)-\sum_{r_b=\max(0,2n_b-n_a)}^{n_b}p(r_b\mid \underline{k}_a)\cdot E(\hat{\theta}_{\underline{k}_b}\hat{\theta}_{\underline{k}_b'}\mid r_b,\underline{k}_a)\,. \tag{7.6.6.}$$

We want to adjust (7.6.6.) so that it estimates (7.6.1).

In order to estimate the first term in (7.6.1) we can replace $E(\hat{\theta}_b^2 \mid \underline{k}_a)$ in (7.6.6) with $\hat{\theta}_a^2$, but how can we estimate the second term in (7.6.1)? We will try to adjust the factor $p(r_b \mid \underline{k}_a)$ in (7.6.6). Let us define

$$\tilde{p}(r_b \mid \underline{k}_a) = \sum_{r_a=\max(0,2n_a-N)}^{n_a} b(r_a, r_b \mid \underline{k}_a) \cdot p(r_a), \qquad (7.6.7.)$$

where $b(r_a, r_b \mid \underline{k}_a)$ is such that for the linear estimator $\bar{y}_b$ we have

$$V(\bar{y}_b \mid \underline{k}_a) = E(\bar{y}_b^2 \mid \underline{k}_a) - \sum_{r_b=\max(0,2n_b-n_a)}^{n_b} \tilde{p}(r_b \mid \underline{k}_a) \cdot E(\bar{y}_{\underline{k}_b} \bar{y}_{\underline{k}_b} \mid r_b, \underline{k}_a) = \hat{V}(\bar{y}_a).$$

Let us now study the relationship between two pairs of variables, n_a, r_a and n_b, r_b. We find this relationship from the requirement that $E_a[E(\bar{y}_{\underline{k}_b} \bar{y}_{\underline{k}_b} \mid r_b, \underline{I})] = E(\bar{y}_{\underline{k}_a} \bar{y}_{\underline{k}_a} \mid r_a)$, which is the second term in (7.6.1). For some designs we get an equation that does not depend on the data, e.g. in the case of the variance S_y^2, and we can get some solutions for n_b and r_b.

For example, the two-phase sampling situation with SI/SI and one-phase SI sampling produce for $E_a[E(\bar{y}_{\underline{k}_b} \bar{y}_{\underline{k}_b} \mid \underline{I}_a, r_b)] = E(\bar{y}_{\underline{k}_a} \bar{y}_{\underline{k}_a} \mid r_a)$ an equation based on (7.6.4); that is,

$$\frac{r_a}{n_b^2} E_a[\overline{y_{\underline{I}_a}^2}] + \left(1 - \frac{r_a}{n_b^2}\right) E[\overline{y\, y'}_{\underline{I}_a}] = \frac{r_a}{n_a^2} \overline{y_U^2} + \left(1 - \frac{r_a}{n_a^2}\right) \overline{y\, y'}_U$$

$$\Leftrightarrow \qquad \frac{r_b}{n_b^2} = \frac{r_a}{n_a^2}$$

and finally we have

$$r_b = \frac{r_a n_b^2}{n_a^2}. \qquad (7.6.8.)$$

The term r_b can also be used for nonlinear estimators, as the next example shows. Let us study the case $N = 7$ and $n_a = 4$ for which the probabilities $p(r)$ are as follows.

Table 7.3. The Probabilities of Joint Units $p(r)$ in SI/SI Sampling with $N = 7$, $n_a = 4$ and Various n_b

n_b	N, n_a	0	1	2	3	4
2	7, 4	0.167	0.667	0.167	-	-
3	7, 4	-	-	0.75	0.25	-
4	7, 4	-	-	-	-	1
4	7, 7	-	0.114	0.514	0.343	0.029

Questions: Which is the right n_b? What would be suitable reweighting for different values of $p(\underline{r})$?

The non-integer r's that are solutions of (7.6.8) are taken into account by using probabilities $b(r_a, r_{b,lower}) = r_{b,upper} - r_b$, where $r_{b,lower}$ is the greatest integer less than r_b and $r_{b,upper} = r_{b,lower} + 1$. The process of finding the right n_b is as follows:

- We have $\hat{\theta}_a^2$ as an unbiased estimator for the case $\hat{E}(\hat{\theta}_{\underline{k}_a} \hat{\theta}_{\underline{k}_a} \mid r_a = 4) = \hat{\theta}_{\underline{k}_a}^2$.
- In order to estimate $\hat{E}(\hat{\theta}_{\underline{k}_a} \hat{\theta}_{\underline{k}_a} \mid r_a = 3)$, we first choose $n_b = 3$ from which it follows that $r_b = r_a n_b^2 / n_a^2 = 3*3^2 / 4^2 = 1.6875$, which unfortunately cannot be achieved due to the minimum $r_b = 2$. Thus $n_b = 3$ is rejected.
- The next alternative for $\hat{E}(\hat{\theta}_{\underline{k}_a} \hat{\theta}_{\underline{k}_a} \mid r_a = 3)$ is $n_b = 2$, and this choice gives the following results:

Table 7.4. Solutions for r_b and $b(r_a, r_{b,lower})$

Conditional expectation	$r_b = r_a n_b^2 / n_a^2$	$r_{b,lower}$; $r_{b,upper}$	$b(r_a, r_{b,lower})$
$E(\hat{\theta}_{\underline{k}_a} \hat{\theta}_{\underline{k}_a} \mid r_a = 3)$	$3 \cdot 2^2 / 4^2 = 0.75$	0; 1	0.25
$E(\hat{\theta}_{\underline{k}_a} \hat{\theta}_{\underline{k}_a} \mid r_a = 2)$	$2 \cdot 2^2 / 4^2 = 0.5$	0; 1	0.5
$E(\hat{\theta}_{\underline{k}_a} \hat{\theta}_{\underline{k}_a} \mid r_a = 1)$	$1 \cdot 2^2 / 4^2 = 0.25$	0; 1	0.75

Thus, in practice we use $E(\hat{\theta}_{\underline{k}_b} \hat{\theta}_{\underline{k}_b} \mid r_b = 0, \underline{k}_a)$, $E(\hat{\theta}_{\underline{k}_b} \hat{\theta}_{\underline{k}_b} \mid r_b = 1, \underline{k}_a)$ and $\hat{\theta}_a^2$. The following table presents the terms needed for variance estimation.

Table 7.5. Terms Needed for Variance Estimation

$n_b = 4$	$p(r_a)$
$\hat{\theta}^2_{\underline{k}_a}$	$1 \cdot 0.0285715 = 0.0285715$
$n_b = 2$	$\tilde{p}(r_b) = \sum_{r_a=\max(0,2n_a-N)}^{n_a-1} b(r_a, r_b) \cdot p(r_a)$
$E(\hat{\theta}_{\underline{k}_b} \hat{\theta}_{\underline{k}_b} \mid r_b = 1, \underline{k}_a)$	$0.75 \cdot 0.114285 + 0.5 \cdot 0.514285 + 0.25 \cdot 0.3428571 = 0.5428571$
$E(\hat{\theta}_{\underline{k}_b} \hat{\theta}_{\underline{k}_b} \mid r_b = 0, \underline{k}_a)$	$0.25 \cdot 0.114285 + 0.5 \cdot 0.514285 + 0.75 \cdot 0.3428571 = 0.4285714$

Note that the term $p(r_a)$ is the probability of the number of joint observations from (7.6.1), i.e. the decomposed expression of $V(\hat{\theta}_a)$. The variance estimator is of the form

$$\hat{V}(\hat{\theta}_a) = [1 - p(r_a)]\hat{\theta}_a^2 - \sum_{r_b=\max(0,2n_b-n_a)}^{n_b} \tilde{p}(r_b) \cdot E(\hat{\theta}_{\underline{k}_b} \hat{\theta}_{\underline{k}_b} \mid r_b, \underline{k}_a) \qquad (7.6.9.)$$

and using the terms in Table 7.3 we get

$$\hat{V}(\hat{\theta}_a) = \hat{\theta}^2_{\underline{k}_a} - 0.0285715 \cdot \hat{\theta}^2_{\underline{k}_a} - 0.5428571 \cdot E(\hat{\theta}_{\underline{k}_b} \hat{\theta}_{\underline{k}_b} \mid r_b = 1, \underline{k}_a)$$
$$- 0.4285714 \cdot E(\hat{\theta}_{\underline{k}_b} \hat{\theta}_{\underline{k}_b} \mid r_b = 0, \underline{k}_a).$$

Let us take the same linear case example with the estimator of the population mean as was examined previously in this section ($N = 7$, $n_a = 4$, $n_b = 2$). Let the variable y values in the sample $\underline{k}_a$ be 1, 3, 4 and 6. Then we have $\bar{y}_{\underline{k}_a} = 3.5$, $\bar{y}^2_{\underline{k}_a} = 12.25$, $\overline{y^2}_{\underline{k}_a} = 15.5$, $\overline{yy'}_{\underline{k}_a} = 11.166667$ and $\hat{S}^2_y = \overline{y^2}_{\underline{k}_a} - \overline{yy'}_{\underline{k}_a} = 4.333333$. By using the analytic form of the variance estimator of $\bar{y}_{\underline{k}_a}$ we get

$$\hat{V}(\bar{y}_{\underline{k}_a}) = \left(\frac{1}{n_a} - \frac{1}{N}\right)\hat{S}^2_y = 0.1071429 \cdot 4.333333 = 0.46429.$$

From (7.6.4) we have $E(\bar{y}\,\bar{y}'|\,r)=\frac{r}{n^2}E(y^2)+\left(1-\frac{r}{n^2}\right)E(y\,y')$, and when it is adjusted to this situation, the estimator is of the form

$$\hat{V}(\bar{y}_{\underline{k}_a})=\bar{y}^2_{\underline{k}_a}-p(r_a=n_a)\bar{y}^2_{\underline{k}_a}-\sum_{r_b=\max(0,2n_b-n_a)}^{n_b}\tilde{p}(r_b)\cdot\left[\frac{r_b}{n_b^2}\overline{y^2}_{\underline{k}_a}+\left(1-\frac{r_b}{n_b^2}\right)\overline{y\,y'}_{\underline{k}_a}\right]$$

$$=\bar{y}^2_{\underline{k}_a}-p(r_a=4)\bar{y}^2_{\underline{k}_a}-\tilde{p}(r_b=1)\left[\frac{1}{4}\overline{y^2}_{\underline{k}_a}+\frac{3}{4}\overline{y\,y'}_{\underline{k}_a}\right]-\tilde{p}(r_b=0)\overline{y\,y'}_{\underline{k}_a}$$

$$=12.25-0.0285715\cdot 12.25-0.5428471\cdot\left[\frac{1}{4}\cdot 15.5+\frac{3}{4}\cdot 11.166667\right]-0.4285714\cdot 11.166667$$

$$=12.25-0.35-6.65-4.7857137=0.46429$$

which is the same as the analytical solution of the variance estimator.

However, the different scale of $\hat{\theta}_a^2$ (of size n_a) and the rest of the terms (of size n_b) in the estimator (7.6.9) can add some additional bias to the variance estimator. This was noticed in some preliminary tests of the variance estimator. When $\hat{\theta}_a^2$ is replaced with $[E(\hat{\theta}_b\,|\,\underline{k}_a)]^2$ we obtain less biased and more stable results than with (7.6.9). Note that the replacement term is *not* $E(\hat{\theta}_b^2\,|\,\underline{k}_a)$ from the decomposition of the conditional variance in (7.6.6). This is due to the assumption that $E(\hat{\theta}_b\,|\,\underline{k}_a)\approx\hat{\theta}_a$. Thus the *variance estimator* is of the form

$$\hat{V}(\hat{\theta}_a)=(1-p(r_a))[E(\hat{\theta}_b\,|\,\underline{k}_a)]^2-\sum_{r_b=\max(0,2n_b-n_a)}^{n_b}\tilde{p}(r_b)\cdot E(\hat{\theta}_{\underline{k}_b}\hat{\theta}_{\underline{k}_b}\,|\,r_b,\underline{k}_a)\,. \quad (7.6.10.)$$

This variance estimator is used in the simulations in Chapter 8 instead of (7.6.9).

8. *SIMULATIONS*

In this thesis an overview of several existing variance estimation methods is given. In addition some new methods are presented, such as

* post-design vector method
* two-phase resampling correction method for bias reduction
* two-size resampling correction method for bias reduction
* correction based on the MSE criterion with two-phase resampling
* decomposition method with unbiasedness assumption
* decomposition method with weight adjustments

In this chapter we conduct two simulations with real data by using *exact* calculations of relative properties of these old and new variance estimators. Section 8.1 includes descriptions and the main reasons for the choices of the sampling design and of the estimators for which the variances are to be estimated. Some theoretical peculiarities appearing in the calculations of some methods and estimators are pointed out. In Section 8.2 we present 23 different variance estimation methods and two main strategies for the selection of the resample size. Descriptions of three relative properties of the variance estimators can be found in Section 8.3. The programming principles as well as the structure of the simulation program are presented in Section 8.4. The reasons for the use of small-size populations are characterised in Section 8.5. The descriptions of two data sets, simulation results and their interpretations can be found in Section 8.6.

8.1. *Sampling Design and Estimators*

This simulation is restricted to the situation where simple random sampling without replacement is the chosen sampling design. There are four reasons for this:

1) in practice all of the variance estimators presented in this thesis can be calculated under this design;

2) the task of calculating *exact properties of the estimators* can be carried out under this design without any major difficulties;

3) good results at the simplest level of sampling may indicate better variance estimation in more complex situations as well, although this hypothesis is by no means straightforward;

4) the main efforts in this thesis were focused on variance estimation of complex estimators but not necessarily under complex sampling designs.

Here we are dealing with small population sizes. Then we avoid the situation where most of the traditional variance estimation methods tend to unify asymptotically. With low sample and resample sizes we put the methods to a difficult test – in many cases the differences can be clearly seen. On the other hand, overall conclusions cannot be drawn, because the results may in some cases vary quite significantly depending on the values of the x and y variables and their correlation. Still, the results may reveal some aspects of the nature of different methods.

The parameters and their corresponding estimators for which the variances are to be estimated are:

- Variance of variable x in the population: $\hat{S}_x^2 = \sum_{k \in r} \frac{(x_i - \bar{x}_r)^2}{n-1}$
- Ratio of the totals of variables x and y: $\hat{R} = \hat{t}_x / \hat{t}_y$
- Median of variable x: $\hat{Md}_x = \hat{F}_x^{-1}(0.5)$
- Maximum of variable x: $\hat{Max}_x = \hat{F}_x^{-1}(1)$

Note that for even samples (and resamples) the median is not the mean of the two middle values, but the lower of these two values.

Taking a look at the non-resampling variance estimation methods for these estimators, we see that the first two are smooth statistics and so can be dealt with by using Taylor's linearisation method. There is a variance estimation method for the median based on Woodruff's confidence interval method (Woodruff 1952, Rao and Wu 1988, see Section 3.3). For some small samples the method produced the same observed value for both lower and upper confidence interval values. In order to introduce some variation the method had to be adjusted by taking the next value as the upper term of the calculations. However, the Woodruff method is not applicable in the case of the maximum, simply because the estimated variance of $\hat{F}_x^{-1}(1)$ in (3.3.3) becomes zero.

There are some peculiarities in the calculations of different resampling variance estimation methods, such as

- For the case of the maximum, the internal scaling and post-design vector methods just produce the conditional variance of the estimator for the resample size n_b.
- The internal scaling method produces negative weights with some combinations of n_a and n_b (see Section 3.7).

- The variance estimation methods based on the decomposition approach can sometimes lead to negative variance estimates.

- The two-size resampling method based on the assumption $V_b[E(\hat{\theta}_c \mid \underline{I}_b) \mid \underline{k}_a] = V(\hat{\theta}_b \mid \underline{k}_a)$ from (6.1.5) and the decomposition method with $\hat{V}_a[E(\hat{\theta}_b \mid \underline{I}_a)] = \hat{V}(\hat{\theta}_b) - V(\hat{\theta}_b \mid \underline{k}_a)$ from (7.3.1) may give some strange results in some cases when the assumptions are not valid, especially as far as the median and the maximum are concerned.

- With the median it may happen that $V(\hat{Md}_{n_b-1} \mid \underline{k}_a) < V(\hat{Md}_{n_b} \mid \underline{k}_a)$ due to the different practices in the calculation of the median with odd and even sample sizes, especially when we are dealing with such low sample sizes; it is obvious that some resampling variance estimation methods suffer because of this. For example, the jackknife $n-1$ method may produce worse results than subsampling with $n-2$ simply due to the "odd sample size vs. even resample size" case or vice versa.

8.2. *Variance Estimators to Be Studied*

The simulation includes most of the variance estimation methods presented in this thesis. The techniques of the methods are clear as such, but some choices had to be made, i.e. resample sizes and numbers of random groups. Two principles were taken into account:

1) resamples as near as possible to the original sample size (familiar from the traditional jackknife and bootstrap practices), and

2) resamples near to the resample size fulfilling the linear case criterion, denoted here as n_{adj} (familiar from the bootstrap methods with adjusted resampling designs, e.g. the BWO method).

Note that in this simulation the dependent random groups method and the "not $n-1$" jackknife method are based on random groups of equal size; for example in the case $n_a = 7$ only the three $n_{b0} = 2$ random groups are taken into account in the calculations. Furthermore, the results of these methods are based on the conditional expectations over different groupings, so the variance of the variance estimator *does not* include the variation due to groupings within a sample. Table 8.1 presents all the variance estimators to be studied in the simulations.

Table 8.1. Variance Estimators to Be Studied

Variance estimation method	Scale adjustment	Formula e
1) Approximation based on Taylor's linearisation	-	3.3.2
2) Woodruff's method for the median	-	3.3.9
3) Jackknife $n-1$	external $\hat{Q}_{lin}$	3.7.1
4) Jackknife $n-1$	internal $\hat{Q}_{lin}$	3.7.4
5) Jackknife two out of three random groups	external $\hat{Q}_{lin}$	3.5.6
6) Two random groups	external $\hat{Q}_{lin}$	3.5.2
7) Three random groups	external $\hat{Q}_{lin}$	3.5.2
8) Bootstrap with replacement $n-1$	external $\hat{Q}_{lin}$	3.7.1
9) Bootstrap with replacement $n-1$	internal $\hat{Q}_{lin}$	3.7.4
10) Subsampling with adjusted subsample size $n_{adj,low}$	external $\hat{Q}_{lin}$	3.7.1
11) Subsampling with adjusted subsample size $n_{adj,low}$	internal $\hat{Q}_{lin}$	3.7.4
12) Subsampling based on adjusted subsample size n_{adj}	randomisation	3.7.11
13) Subsampling based on adjusted subsample size n_{adj}	post-design vector	5.3.3
14) Pseudopopulation-based variance estimation	randomisation	3.7.11
15) Two-phase resampling, SI, $n_{adj,low}$, $n_{adj,low}-1$	external $\hat{Q}_E$	3.7.1 & 6.1.4
16) Two-phase resampling, SI, $n-1$, $n-2$	external $\hat{Q}_E$	3.7.1 & 6.1.4
17) Two-phase resampling with MSE criterion, $n_{adj,low}$, $n_{adj,low}$ -1	external $\hat{Q}_{min}$	3.7.1 & 6.2.5
18) Two-phase resampling with MSE criterion, $n-1$, $n-2$	external $\hat{Q}_{min}$	3.7.1 & 6.2.5
19) Resampling with two sizes, $n_{adj,low}$, $n_{adj,low}-1$	external $\hat{Q}_E$	3.7.1 & 6.1.8
20) Resampling with two sizes, $n-1$, $n-2$	external $\hat{Q}_E$	3.7.1 & 6.1.8
21) Decomposition approach, unbiasedness assumption, $n_b = 2$	resample pair terms reweighted	7.5.2
22) Decomposition approach, unbiasedness assumption, $n_b = 3$	resample pair terms reweighted	7.5.2
23) Decomposition approach with weight adjustments, maximal n_b available	resample pair terms reweighted	7.6.10

Here n_{adj} is the resample size (usually a non-integer), which fulfils the linear case condition. Correspondingly $n_{adj,low}$ is the integer part of n_{adj}. The results for the random groups method and the jackknife two out of three random groups method are based on the *mean over all possible variance estimates* within a sample $\underline{k}_a$ (see Appendix A for the definition). Note that in normal practice *only one random grouping* is used for variance estimation.

8.3. *Properties of the Variance Estimators to Be Studied*

Due to the varying scales of the estimators in question, it is reasonable to study relative properties of the variance estimators. These properties are:

- Relative bias of the variance estimator:

$$RB_a[\hat{V}(\hat{\theta})] = \frac{E_a[\hat{V}(\hat{\theta})] - V(\hat{\theta})}{V(\hat{\theta})}$$

- Relative standard error of the variance estimator:

$$RSE_a[\hat{V}(\hat{\theta})] = \frac{\sqrt{V_a[\hat{V}(\hat{\theta})]}}{V(\hat{\theta})}$$

- Relative square root of the MSE of the variance estimator:

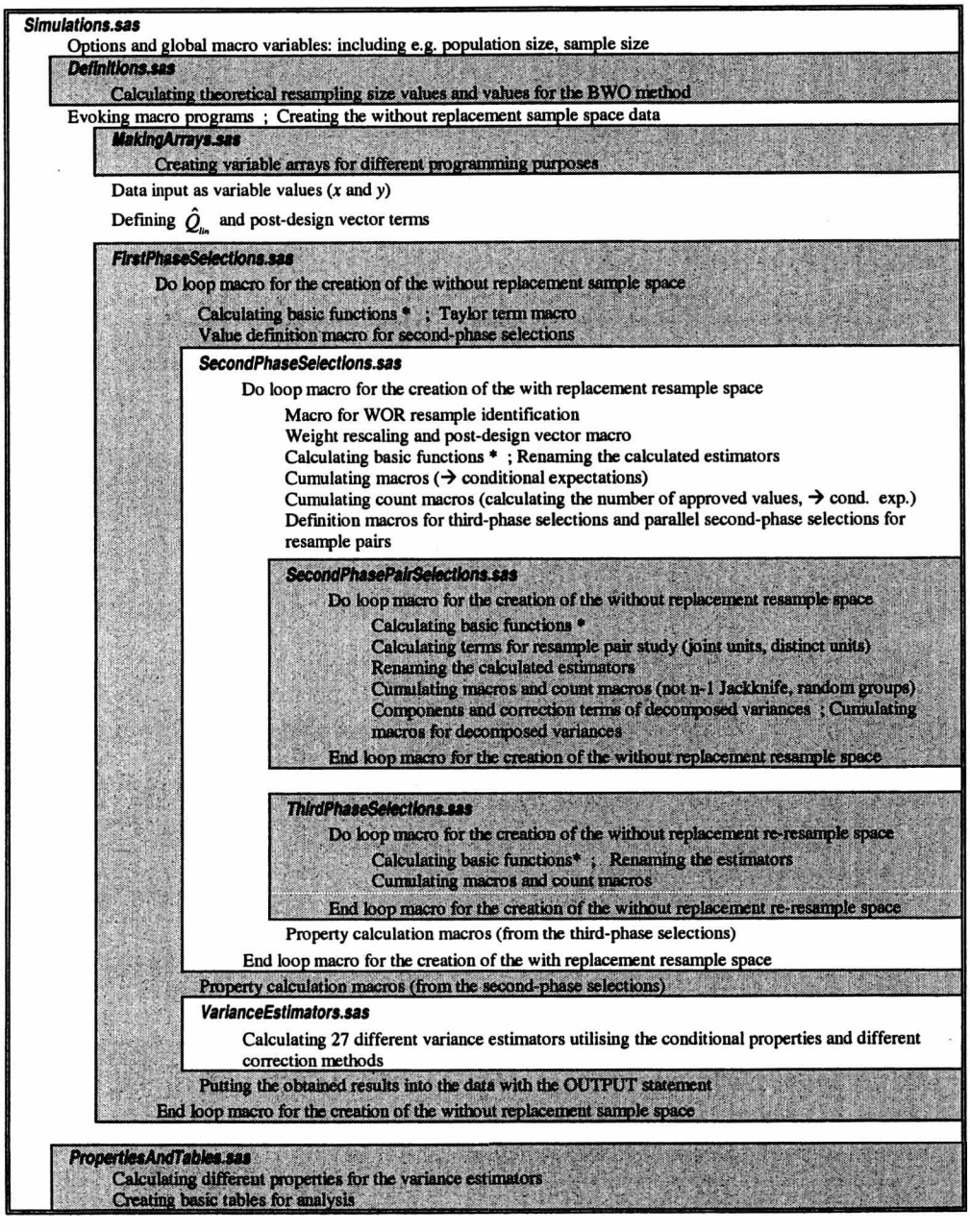
$$RSRM_a[\hat{V}(\hat{\theta})] = \frac{\sqrt{V_a[\hat{V}(\hat{\theta})] + \left[B_a[\hat{V}(\hat{\theta})]\right]^2}}{V(\hat{\theta})}$$

Note that these properties are adjusted when there are nonexisting estimator values, as these are excluded from the probability calculations of the sampling design. In the programmes there is a counter that saves the number of valid values of the estimator.

8.4. *Structure of the Simulation Program*

Table 8.2 shows the main structure of the simulation program. The idea is to use the SAS® macro language in order to perform the routines that are repeated at different levels using the same macros and to make the process independent of the data and sample size choices as much as possible. The checks were conducted with the linear case estimator (the total) for which we get the analytical variance estimator as a result, except for the MSE criterion variance estimators. Furthermore, the creation of the sample and resample spaces was controlled using counters and runtime prints. For some methods the results from the program were compared with the theoretical calculations done by hand, and for the Taylor approximations the results were verified with a CLAN macro package provided by Statistics Sweden.

Table 8.2. Structure of the Simulation Program



* Basic functions: *variable sum, order number, totals, ratios, variance, covariance, correlation, median and maximum.*

8.5. *Data*

When carrying out simulations, it is possible to use real data as well as artificial data. The latter is usually based on some superpopulation model. Although tests were conducted using both types of data , only the results obtained using real data are presented here. We are dealing with small populations for which the properties of the estimators are calculated *exactly*. The program is based on the design vector approach, and provides these properties for *both with- and without replacement resampling designs*. The superpopulation model may not be clearly present in a small population, and thus it may be more appropriate to use data from real situations. When stratification is extensive, this phenomenon occurs from time to time. Further, the program code in Appendix C allows tests for other small populations, if there is a need for some additional studies. Here it is considered important to have at least one population with clearly skewed variable value distribution.

8.6. *Simulation Results*

One stratum of the environmental expenditure survey. The first set of data is taken from the Finnish environmental expenditure survey carried out in 1999, and includes the register variables "gross product" (x) and "number of personnel" (y) for the enterprises in one stratum. The number of units in the stratum is nine. The distribution of the gross product is quite skewed and there is some correlation between the number of personnel and the gross product, as expected. One value of the "number of personnel" variable has been changed slightly in order to avoid repetition.

Here is the data from the environmental expenditure survey:

- x (1000 *FiM*): 447, 12866, 62960, 13767, 90, 20066, 1645, 33832, 7574
- y: 2, 8, 5, 9, 1, 10, 3, 4, 6

where population size $N = 9$ and sample size $n_a = 6$.

Table 8.3. Results from the Environmental Expenditure Survey – Register Variables from One Stratum (%)

	Variance $\hat{S}_x^2$			Ratio $\hat{R}_{xy}$			Median $\hat{Md}_x$			Maximum $\hat{Max}_x$		
	RB	RSE	RSRM	RB	RSE	RSRM	RB	RSE	RSRM	RB	RSE	RSRM
TAY / WOOD	-33.6	49.2	59.6	1.0	62.0	62.0	-85.5	15.3	86.8	-	-	-
JACK n–1, ext	3.8	76.9	77.0	7.9	72.7	73.1	-3.7	109.2	109.2	-15.6	64.5	66.4
JACK n–1, int	14.1	84.5	85.7	11.2	77.6	78.4	-42.2	65.5	77.9	-	-	-
JACK 2/3 rg, ext	3.8	76.9	77.0	16.7	70.2	72.2	-11.4	58.0	59.2	-58.5	25.8	64.0
RG2, ext	16.5	82.3	84.0	29.5	69.2	75.3	-61.0	20.2	64.2	-75.6	13.5	76.8
RG3, ext	5.2	77.4	77.6	13.2	73.1	74.2	-30.5	43.8	53.3	-37.6	42.7	56.9
BWR n–1, ext	-41.9	41.5	59.0	17.1	66.1	68.3	104.7	102.4	146.4	-61.0	24.4	65.7
BWR n–1, int	-39.6	44.2	59.3	4.5	61.3	61.4	46.5	90.0	101.3	-	-	-
SUB, n_{adj}, ext	6.3	77.9	78.1	14.7	74.0	75.5	-25.3	46.5	52.9	-36.9	42.8	56.5
SUB, n_{adj}, int	1.3	74.4	74.4	10.1	70.3	71.0	-63.0	41.9	75.7	-	-	-
SUB, n_{adj} rand	5.5	77.5	77.7	12.5	73.6	74.6	-18.1	56.1	58.9	-29.8	49.8	58.1
SUB, n_{adj} post-dv	-32.4	49.6	60.0	10.1	71.1	71.8	-42.2	65.5	77.9	-	-	-
BWO, rand	-14.0	63.0	64.5	11.8	70.3	71.3	18.2	84.4	86.3	-35.8	46.0	58.3
2PR, Q_E, n_{adj}, n_{adj}–1	-2.6	74.3	74.4	1.8	71.4	71.4	-35.7	54.5	65.1	-7.9	71.7	72.2
2PR, Q_E, n–1, n–2	-0.2	75.3	75.3	-0.2	70.8	70.8	95.2	320.5	334.3	13.2	99.0	99.9
2PR, MSE, n_{adj}, n_{adj}–1	-33.8	49.6	60.0	-24.1	54.9	60.0	-62.5	23.8	66.9		28.5	65.8
2PR, MSE, n–1, n–2	-16.5	62.7	64.8	-11.0	63.3	64.2	-45.4	56.1	72.2	-32.7	54.1	63.2
2SR, n_{adj}, n_{adj}–1	-2.6	74.3	74.4	-0.2	74.4	74.4	118.8	464.0	479.0	113.8	233.7	260.0
2SR, n–1, n–2	-0.2	75.3	75.3	-1.8	72.0	72.0	-260	794.0	835.5	55.6	152.9	162.7
DEC, n_b=2	0.0	75.4	75.4	19.8	64.9	67.8	-69.6	16.0	71.4	-76.3	13.3	77.4
DEC, n_b=3	0.0	75.4	75.4	12.4	68.0	69.2	-22.2	51.1	55.7	-59.3	25.7	64.7
DEL, max n_b	4.8	77.3	77.4	16.7	70.4	72.4	-15.6	56.0	58.2	-58.5	25.9	63.9

All the result tables contain the same methods in the same order as in Table 8.1. Explanations of the abbreviations: TAY = Taylor's approximation method; WOOD = Woodruff's method for quantiles; JACK= Jackknife method; ext = external scaling; int = internal scaling; 2/3 rg = two out of three random groups jackknife; RG2 = two random groups; RG3 = three random groups; BWR = bootstrap with replacement, i.e. resampling with replacement; SUB = subsampling, i.e. resampling without replacement; n_{adj} = resample size fulfilling the linear case condition; $n_{adj,I}$ = integer part of n_{adj}; rand = randomisation; post-dv = post-design vector, BWO = bootstrap without replacement with a pseudopopulation; SI = simple random sampling without replacement; SIR = simple random sampling with replacement; 2PR = two-phase resampling; Q_E = correction with estimated Q_E; MSE = correction with MSE criterion; 2SR = two-size resampling; DEC = decomposition approach with unbiasedness assumption; DEL = decomposition approach with adjusted linear-case weighting; max n_b = maximal n_b available. The five best methods are indicated by grey shading and the five worst methods are indicated by black shading.

When dealing with the relative bias of the variance estimator of the variance $\hat{S}_x^2$, one can see that the *Taylor approximation* is not sufficient at this level, as it clearly gives a downward biased variance estimator. The *externally scaled jackknife estimator* is quite good, which is not a surprise, because $\hat{S}_x^2$ may have similar properties in the SI/SI case as in the SIR/SI case, where $n_b = n_a - 1$ gives the best result for externally scaled variance estimation when smaller subsample sizes are compared with it (see Chapter 4). The *internally scaled jackknife method* scales the variance estimator more than the external alternative. The *with-replacement resampling designs* cause severe underestimation of the variance with both types of scaling. *The post design vector* clearly loses to its competitor, *randomised subsampling*. Both *two-phase resampling methods* and their shortcuts, *two-size resampling methods,* give very accurate results, and this may indicate that the nonlinearity part included in the correction reduces the bias of the variance estimators. The *MSE correction* overreacts downwards to some extent. The *decomposition approach with the unbiasedness assumption* provides *totally unbiased* variance estimators for $\hat{S}_x^2$. Some additional tests reveal that this phenomenon seems to be independent of the data and holds with populations smaller than 10. The *decomposition approach with adjusted weighting with a maximal* n_b also performs well.

In general the variance of the variance estimator depends on the scaling factor used in its calculation, especially when the external scaling method is used. We may increase the variance when correcting upwards. On the other hand, downwardly biased variance estimators in general (including other methods as well) tend to have a small variance (as here for example with *Taylor* and *bootstrap with replacement*). The question is how much bias we can bear in these cases. Looking at the relative standard error of the variance estimator of the variance $\hat{S}_x^2$, we can see that most of the not-so-biased variance estimators are concentrated between 70 % and 80 %, and even the most inaccurate are not far away from that interval.

Correspondingly, the relative square root of the mean square error of the variance estimator of the variance $\hat{S}_x^2$ provides bad results for severely downward-biased estimators. The main goal of MSE correction, i.e. minimising the MSE of the variance estimator, is not fully achieved, although the results are almost as good as those of the five best methods.

The ratio $\hat{R}_{xy}$ is in many cases a rather stable estimator, and this can be seen in both the bias and the variance of the variance estimators: the bias is quite low for nearly all variance estimators and the relative standard error varies between 65 % and 75 % in most cases. The *Taylor approximation* performs very well in this case with respect to all of the criteria being considered. Again the *two-phase resampling* and *two-size resampling* methods give some of the best results, as far as the bias is concerned. In general *bootstrap with replacement with internal scaling* seems to have good properties.

The estimation of the median $\hat{Md}_x$ is affected by two problems that cause difficulties for some variance estimation methods. As mentioned before, the different practices for odd and even sample and resample sizes may have some effect with such small sample and resample sizes. With this data, such a phenomenon is especially likely to occur when an odd resample size is used for estimation. Furthermore, some methods are based on conditionality rules (*two-size resampling, decomposition with unbiasedness assumption*), and the assumptions may not be valid for quantiles, at least with small sample and resample sizes. Both properties can be seen especially clearly in *two-size resampling*, where the case "$n-1, n-2$" provides a *negative* expectation of the variance estimator. Therefore this kind of variance estimator may be giving unexpected results when estimating the quantiles. For the median we have the *Woodruff method*, which unfortunately is in this case severely biased downward. The relative biases for the *two jackknife methods*, *subsampling with randomisation*, the *BWO method* and the two *decomposition methods* can be considered to be rather small.

The maximum $\hat{Max}_x$ is an extreme estimator that usually gives unpredictable results. There is no Woodruff method for the maximum, and furthermore we will not present any results for the internal rescaling and post-design vector methods, because they do not correct the conditional variance of the maximum at all (the maximum with the weight adjustment equals the maximum without the weight adjustment). As far as the bias of the variance estimator is concerned, both of the two-phase resampling methods are the best, followed by the $n-1$ jackknife and subsampling randomisation. Otherwise the results resemble the median case.

In summarizing the results for this set of data, it must be noted that the overall performance of the *BWO method* is rather stable with respect to both the bias and the variance, even though it does not produce the best results in any category. However, the *jackknife $n-1$ method* is even better in that sense. Where the bias is concerned, *two-phase resampling* is the best method, except in the case of the median. For the variance and the ratio *two-size resampling* works nearly as well as *two-phase resampling*. The results of the with-replacement methods (bootstrap with replacement) are not very good, at least with this data set and sample size. The *post-design vector approach* is in general less accurate than subsampling randomisation. The results of the *decomposition methods* vary from one estimator to another.

Unemployment figures, municipalities of South Karelia, Finland, January 2001. The second data set includes the unemployment figures of the municipalities in the district of South Karelia, Finland from January 2001. These figures have been provided by the Finnish Ministry of Labour. The number of municipalities is $N = 14$ and the chosen sample size here is $n_a = 7$.

Table 8.4. Unemployment figures from South Karelia in January 2001

Municipality	Number of unemployed	Number of persons in the labour force
Imatra	2444	14505
Joutseno	809	5292
Lappeenranta	4453	27692
Lemi	192	1501
Luumäki	319	2298
Parikkala	404	2017
Rautjärvi	368	2091
Ruokolahti	395	2810
Saari	129	644
Savitaipale	289	1853
Suomenniemi	53	330
Taipalsaari	280	2278
Uukuniemi	50	225
Ylämaa	83	692

Table 8.5. Results from the District of South Karelia Unemployment Figures (%)

	Variance $\hat{S}_x^2$			Ratio $\hat{R}_{xy}$			Median $\hat{Md}_x$			Maximum $\hat{Max}_x$		
	RB	RSE	RSRM	RB	RSE	RSRM	RB	RSE	RSRM	RB	RSE	RSRM
TAY / WOOD	-29.4	68.3	74.4	-22.7	81.7	84.8	-84.8	20.3	87.2	-	-	-
JACK n–1, ext	1.6	98.3	98.3	35.4	129.6	134.4	71.7	229.0	240.0	-11.2	86.4	87.1
JACK n–1, int	24.0	120.0	122.3	210743	8567263	8569855	177.9	458.6	491.9	-	-	-
JACK 2/3 rg, ext	3.1	99.1	99.3	50.8	85.3	99.3	835.8	1450	1674	-70.4	23.1	74.1
RG2, ext	8.6	102.3	102.7	89.6	83.9	122.8	174.3	360.2	400.1	-84.9	11.4	85.7
RG3, ext	3.2	99.4	99.4	39.1	110.2	117.0	120.2	369.5	388.5	-32.5	60.0	68.2
BWR n–1, ext	-41.4	55.9	69.6	45.2	81.2	93.0	1285	1460	1945	-64.7	28.8	70.8
BWR n–1, int	-38.5	59.1	70.5	-6.5	79.7	80.0	2449	2783	3707	-	-	-
SUB, n_{sub}, ext	3.3	99.2	99.3	55.4	96.9	111.7	61.3	150.6	162.6	-51.8	39.3	65.0
SUB, n_{sub}, int	1.0	97.2	97.2	12.9	90.3	91.2	-16.1	144.2	145.1	-	-	-
SUB, n_{sub} rand	2.7	98.9	98.9	48.6	103.4	114.2	103.8	286.5	304.7	-40.5	50.9	65.1
SUB, n_{sub} post-dv	-26.8	70.5	75.4	36.4	99.5	106.0	88.5	318.0	330.1	-	-	-
BWO, rand	-23.7	73.3	77.0	37.2	92.7	99.9	374.4	590.5	699.2	-48.7	44.7	66.1
2PR, Q_F, n_{sub}, n_{sub}–1	-1.1	97.1	97.1	29.9	108.3	112.4	-8.7	94.2	94.6	-23.3	67.9	71.8
2PR, Q_F, n–1, n–2	0.5	97.8	97.8	31.9	150.3	153.7	180.3	466.1	499.7	17.6	125.6	126.8
2PR, MSE, n_{sub}, n_{sub}–1	-43.0	55.6	70.3	-12.4	65.5	66.7	-40.3	49.0	63.4	-72.5	22.5	75.9
2PR, MSE, n–1, n–2	-14.0	83.8	84.9	9.1	115.4	115.8	3.6	136.3	136.4	-26.7	75.0	79.6
2SR, n_{sub}, n_{sub}–1	-1.1	97.1	97.1	54.6	185.0	192	-502	14553	14562	1710	3723	4097
2SR, n–1, n–2	0.5	97.8	97.8	42.6	175.3	180	-1679	34380	34421	61.4	182.2	192.3
DEC, n_b=2	0.2	97.7	97.7	54.3	75.9	93.3	41.6	154.2	159.8	-85.2	11.3	85.9
DEC, n_b=3	0.1	97.7	97.7	27.0	82.4	86.7	511.2	925.0	1057	-71.0	22.9	74.6
DEL, max n_b, $\hat{\theta}_a^2$	2.5	99.2	99.2	1988	3115	3695	-1077	2078	2340	261.6	267.7	374.3
DEL, max n_b, $E(\hat{\theta}_b \mid \underline{k}_a)^2$	2.5	99.2	99.2	32.9	83.2	89.4	606	1080	1239	-70.8	23.0	74.4

Abbreviations in this table are the same as in Table 8.3.

The results for the bias of the variance estimator of the variance $\hat{S}_x^2$ seem to fall into two groups: low bias and high downward bias. More than half of the methods fall into the former group. The *decomposition approach* still provides good results, although absolute unbiasedness is not reached this time. *Two-phase resampling with Q_E correction* performs well again, and *subsampling with internal correction* also seems to be well-suited to this data. The *bootstrap with replacement method* is the most downwardly biased, followed by one of the *MSE correction methods* and *Taylor's approximation.* Here the variance of the variance estimator of the variance $\hat{S}_x^2$ is quite tightly connected to the bias of the variance estimator: high negative bias causes low variance. The same rule holds even for the mean square estimator of the variance estimator of the variance, even though the squared bias term is included in the calculations.

The ratio $\hat{R}_{xy}$ (i.e. the unemployment rate) produces much more varied results here than for the first set of data. Again for this ratio we see that the *Taylor approximation* works very well with each criterion. Furthermore, the *MSE correction* as well as the *BWR method with internal scaling* combine low bias, variance and MSE. The *jackknife n – 1 with internal scaling* is extremely poor in this case and the reason is simple: the coefficient $\hat{Q}_{lin,SI/SI} = 2.57$ exceeds the limit of 1 and causes severe inconsistency in the weighting structure. Again the variance and the mean square error seem to follow the bias of the variance estimator.

The results for the median $\hat{Md}_x$ show rather large differences between the methods. Here both of the *two-phase resampling methods* (Q_E and MSE correction) are very efficient with respect to all of the criteria. In addition, the *internal rescaling for subsampling* and the $n_b = 2$ *decomposition method* have low biases. As before, *two-size resampling with the conditionality assumption* is not suitable for variance estimation of the quantiles, at least with these data. The *Woodruff method* is strongly negatively biased, as with the previous data. Note that for some data sets other than these two this method provided less negative bias, so no straight conclusions can be drawn.

The variance of the maximum $\hat{Max}_x$ is mostly underestimated. The *two-phase resampling methods* with Q_E have a bias-correcting effect. Further, the *externally scaled n – 1 jackknife* and the *random group 3 method* work well. As for the median, *two-size resampling with the unbiasedness assumption* is poor here as well.

As a summary for this second data set, we note the good performance of both *two-phase sampling methods* (Q_E and MSE correction), followed by the *internally scaled subsampling method.* The *randomisation methods, jackknife n – 1 with external scaling* and the *random groups 3 method* give rather moderate results in general. The *post-design vector approach* is better than *subsampling with randomisation* in two cases out of three, as far as the bias is concerned. The *decomposition approach with the unbiasedness assumption* seems to be quite good in general. The *Taylor* and *Woodruff methods* provide downwardly biased results, as expected. The *bootstrap with replacement method* is not successful, except in the case of the ratio. Again the shortcut method of *two-size resampling* is not valid with quantiles, giving negative expectations.

8.7. *Summary of the Performances of the Variance Estimators*

The simulation program measured three accuracy properties (*relative bias, relative standard error, relative square root of the mean square error*) for 23 different variance estimation methods. The data used in the simulation were two register variables (gross product, number of personnel) from one stratum in the environmental expenditure survey and some labour force figures (number of unemployed, number of persons in the labour force) from South Karelia, Finland in January 2001.

The scaling factor used in either strict internal or external correction or in resampling design adjustments tends to affect the variance of the variance estimator to some extent. Increasing the factor often increases the variance, and vice versa, though the rate of change varies depending on the method and the estimator.

The variance estimator based on *Taylor's linearisation method* was tested with two estimators: the ratio $\hat{R}_{xy}$ and the variance $\hat{S}_x^2$. With these two data sets the linearisation method for the ratio worked well, especially with the enterprise data. On the other hand, the variance estimator of $\hat{S}_x^2$ was clearly downwardly biased.

The *Woodruff method for the median* seemed to underestimate the variance considerably. However, some other tests revealed much less negative bias with some data, so no final conclusions can be drawn. The method may be sensitive to the data in question, and the small sample size could bring some uncertainty as well. Again the variance of the variance estimator was small when compared to others, due to the large negative bias.

The problems with *internal correction* are its sensitivity to the structure of the estimator and its tendency to give unexpected results for some sample/resample size combinations. These can be seen clearly in certain cases, for example with the second data set *the jackknife with internal correction* presents enormous positive bias due to the fact that $\hat{Q}_{lin,SI/SI} = 2.57$ (exceeding the limit 1). On the other hand, the results for the ratio from the *bootstrap with sampling with replacement with internal correction* are good for both data sets, though in general the figures are not convincing in favour of that method. The best performance in the internal correction category can be found when we choose the *subsample size* $n_{adj,I}$ that produces a $\hat{Q}_{lin,SI/SI}$ as near as possible to one (from the lower side).

The *externally corrected jackknife n - 1 method* gives good overall results where the bias is concerned, especially for the maximum. When the variance and the mean square error are examined, the results are moderate. For *resampling with replacement* the *external correction* seems to be less accurate, though in some cases the variances are among the lowest. *Subsampling* with $n_{adj,I}$ and *external correction* provides less precise variance results, and the bias results are not among the top five in any case.

The methods based on random groups, i.e. *two random groups*, *three random groups* and the *jackknife with two out of three random groups*, are in fact less efficient than in the calculation used here, because here we use $E^*[\hat{V}(\hat{\theta}_n) \mid \underline{k}_n]$, the mean of variance estimators based on all possible random groupings in one sample $\underline{k}_a$. All of these random group methods are quite unsuccessful, except in the case of the maximum, where some good results can be found using the three random groups method.

Two *randomisation methods* are studied here: *subsampling with randomisation* and the *bootstrap without replacement method* based on the *pseudopopulation* (the *BWO* method). The theory indicates that subsampling with randomisation and subsampling with external correction and $n_{adj,i}$ should be quite similar to each other, and this is the case for both data sets. The bias is lower for subsampling with randomisation in seven cases out of eight. The performance of these methods is better for the enterprise data than for the municipality data, producing relatively low bias and MSE in the quantile cases. The BWO method seems to be rather stable as far as the overall results are concerned, with no values among the best or the worst. Only the variance estimation of the median estimator for the second data set had quite a high bias value.

The *post-design vector method* (see Chapter 4) serves as competition for the randomisation methods. As with the BWO method, the results are quite stable and moderate. When compared with subsampling with randomisation, the bias figures are even, three against three. However, for the estimator $\hat{S}_x^2$ the bias was significantly higher in the post-design vector case.

Further utilisation of resample spaces in the correction (see Chapter 6) is carried out in three of the methods: *two-phase resampling* with Q_E *correction, two-size resampling* with Q_E *correction* and *two-phase resampling* with Q_{MSE} *correction. Two-phase resampling* with Q_E *correction* proved to be bias-reducing in most of the cases, giving accurate variance estimation results even for such complex estimators as the median and the maximum. Only the $n-1, n-2$ alternative suffered from the technical difference of calculating the median in the odd and even sample/resample size cases (see details of this in previous sections of this chapter). *Two-size resampling* with Q_E *correction* provided equally strong results for $\hat{S}_x^2$ as those of two-phase resampling, so the shortcut principle works well in this case. The results for the ratio were a bit less accurate but still fair. For the median and the maximum this method produced either negative or very large variance estimates. Based on these results, it is clear that two-size resampling with this assumption cannot be recommended for quantiles. The aim of *two-phase resampling* with Q_{MSE} *correction* was to affect the MSE of the variance estimator, and in fact this phenomenon could be seen in the analysis of municipalities in the second data set. With some estimators the bias was also among the best. With respect to the first data set the main concern was the negative bias caused by the downward correction effect build into the method. In general, these three methods were based on the simple external correction principle, which is sensitive to the value of $\hat{Q}$ being used: once the coefficient is diminished, the variance will be diminished as well. It is still unclear whether the resample space information could give even better correction with some other scaling principle.

The *decomposition variance estimators with sample pair probabilities* (see Chapter 7) produced both good and bad results. The surprising effect with the first data set (and in general all the data with $N < 10$) was that the bias of the *variance estimator* of $\hat{S}_x^2$ with the *unbiasedness assumption* vanished completely. Possibly there is a theoretical property causing this phenomenon; no explanation could be given at this stage. The *variance estimators with the weight adjustment* also performed well in the case of $\hat{S}_x^2$. For the other estimators, the variance estimators with the unbiasedness assumption as well as the weight-adjusted variance estimator had some good results on both the bias and the variance criteria (except in the case of the maximum), but in general the results were less satisfying. Some other simulations revealed a few inconsistencies in these three variance estimators, so we cannot give a straightforward recommendation about the use of these methods.

9. CONCLUSIONS

The first purpose of this thesis was to give a *theoretical overview* of variance estimation in sampling theory. The principles of *probability sampling* (Chapter 2) provide the basis for both estimation and variance estimation. Three main definitions of sampling design (ordered samples, design vectors, subsets) are utilised later in different variance estimation contexts. Levels of sampling are important when resampling variance estimation methods are being considered. Here the distinction between sampling from the population, sampling from the sample, sampling from the set of samples and sampling from the set of resamples is essential in order to describe different variance estimation practices later on. The *main ideas for variance estimation* are presented in Chapter 3, which concentrates on the one-sample case: how to utilise the information based on one sample in variance estimation. Three solutions are available for variance estimation: using units of the sample, using the resample space, and using a metasample from the resample space. The criterion commonly used for the correction of scale differences, the linear case criterion, is dealt with in detail. The different existing methods for scale correction are presented here as well. This overview forms the basis for some new theoretical results and variance estimation methods presented in Chapters 4 through 8.

The *theory of scale differences* in resampling variance estimation is studied in Chapter 4 in terms of *k*-statistics and cumulants. The study is carried out in the case of the estimator of the population variance, $\hat{S}_y^2$. Here the correction coefficient is derived for two cases of sampling/resampling, namely SIR/SI and SI/SI. One can conclude that the coefficient which is usually used for resampling variance estimation of $\hat{S}_y^2$ is always too big: in this case the error is the lowest when the resample size $n_b = n_a - 1$ (e.g. jackknife $n - 1$). The results of Chapter 4 are important, since they show that for *nonlinear estimators the correction of the resampling variance estimator has to depend on the population distribution of the study variable.*

The *post-design vector method* (Chapter 5) utilises the principles of the more recently developed distribution theory framework for sampling designs by artificially changing the original resampling design vector structure. It serves as an alternative to the randomisation method needed in some variance estimation methods based on resampling.

The further utilisation of resample spaces at different levels is carried out in different scaling methods presented in Chapter 6 (*two-phase resampling method with Q_E correction, two-size resampling method with Q_E correction* and *two-phase resampling method with Q_{MSE} correction*). The idea is to bring estimator-specific information into the correction phase, either to reduce the bias of the variance estimator or to achieve the minimal mean square error of the variance estimator.

The structures of the variance and the conditional variance are studied and utilised for variance estimation in Chapter 7. Within *two-phase sampling theory* it is proved that a multinomial design in the first phase and a SI design in the second phase comprise a two-phase design which is again multinomial. *In the context of sample/resample pair probabilities* and the *variances conditioned by the number of joint elements* in the sample/resample pair some resampling variance estimators are constructed. With these operations we try to theoretically remove some bias-causing terms from the variance estimators based on resampling.

After carefully studying their theoretical and practical properties as well as on the results from the simulation study we may conclude that the main purposes of the new variance estimation methods are in many cases fulfilled, although some drawbacks remain.

Post-design vector. Based on the results, no statement can be made about whether the post-design vector method or the randomisation method is better. However, only one form of expansion of the vector was studied in the simulation. In general, it remains to be seen whether there are some other application areas for these kind of design vector adjustments, e.g. from the field of modelling.

Two-phase resampling with Q_E correction. The most convincing results come for from the *two-phase resampling* method with Q_E correction. The simulations show low bias figures in nearly every case, and it is obvious that the method brings some beneficial estimator-specific information into the variance estimation process. There may be some other principles for choosing the resample size that work even better. However, the method is not very practical unless there is a computer program already available for that purpose.

Two-size resampling with Q_E correction. The *two-size resampling method* with Q_E correction is applicable to at least the smooth statistics used in the simulation (ratio and variance), but it is clear that this shortcut version of the two-phase resampling method will not work for quantiles. The properties of this method should be studied in more detail with other smooth statistics.

Two-phase resampling method with Q_{MSE}. The variance of the variance estimator calculated using the *two-phase resampling* method with Q_{MSE} correction is often low, according to the simulations. However, the main concern is the clear negative bias occurring in many cases. It seems that the method often corrects downwards too strongly.

Decomposition variance estimator with unbiasedness assumption. The *decomposition variance estimator with the unbiasedness assumption* seems to have a currently unknown theoretical property that removes the bias of the variance estimator, at least in the case of the variance $\hat{S}_x^2$ with some small sample/resample sizes. In general the method is not among the best. Since it is cumbersome to calculate and applicable only to very simple designs it serves merely as a theoretical construction, which may reveal some new properties about the process of variance estimation with resampling.

Decomposition variance estimator with weight adjustments. The *decomposition variance estimator with weight adjustments* is an even more complex method, and does not bring any further benefit when compared with the method based on the unbiasedness assumption.

It should be noted that the bias and MSE correction methods as well as the decomposition method might be simplified by using the technique of linearising the variance estimator (Yung and Rao 1996); being quite a recent idea, this alternative was not included in the studies of the new methods in this thesis. Furthermore, the new correction methods were carried out by using a simple external correction, which also affects the variance of the variance estimator. Further studies should be done to investigate the possibilities of some other new correction principles.

The simulations and the examples in the thesis included only small populations and the sampling designs were without stratification. In the case of stratification, when we conduct variance estimation independently in each stratum, the methods presented in the thesis apply in these strata separately. However, in large sample situations with numerous strata some of these correction methods might be cumbersome to be conducted. When dealing with one-stage cluster sampling, the methods can be applied as at the element level, but for two-stage sampling the methods in the form presented in this thesis are not applicable. For unequal probability sampling there are some solutions presented in Chapters 4 and 5. In general, more tests of the applicability of these methods in larger scale surveys as well more exploration of the theoretical (e.g. asymptotic) properties is needed.

References

Basu, D. (1958). On Sampling with and without Replacement, *Sankhyã*, **20**, 287 – 294.

Bellhouse, D. R. (2001). The Central Limit Theorem Under Simple Random Sampling. *The American Statistician*, **55**, 352 – 357.

Bellhouse, D. R., Philips, R. and Stafford, J. E. (1997). Symbolic Operators for Multiple Sums. *Computational Statistics and Data Analysis*, **24**, 443 – 454.

Bickel, P. J. and Freedman, D. A. (1984). Asymptotic Normality and the Bootstrap in Stratified Sampling. *The Annals of Statistics*, **12**, 470 – 482.

Bickel, P. J., Götze, F. and van Zwet, W. R. (1997). Resampling Fewer Than n Observations: Gains, Losses, and Remedies for Losses. *Statistica Sinica*, **7**, 1 – 31.

Booth, J. G., Butler, R. W. and Hall, P. (1994). Bootstrap Methods for Finite Populations. *Journal of the American Statistical Association*, **89**, 1282 – 1289.

Brewer, K. R. W. and Hanif, M. (1983). *Sampling with Unequal Probabilities*. Springer-Verlag, New York.

Canty, A. J. and Davison, A. C. (1999). Resampling-based Variance Estimation for Labour Force Surveys. *The Statistician*, **48**, 379 – 391.

Cassel, C. M., Särndal, C. E. and Wretman, J. H. (1977). *Foundations of Inference in Survey Sampling*. Wiley, New York.

Chao, M. and Lo, S. (1985). A Bootstrap Method for Finite Population. *Sankhyã*, **A47**, 399 – 405.

Cochran, W. G. (1953). *Sampling Techniques (First Edition)*. Wiley, New York.

Cochran, W. G. (1977). *Sampling Techniques*. Wiley, New York.

Davison, A. C. and Hinkley, D. V. (1997). *Bootstrap Methods and Their Application*. Cambridge University Press.

Deming, W. E. (1950). *Some Theory of Sampling*. Wiley, New York.

Deming, W. E. (1956). On Simplifications of Sampling Design through Replication with Equal Probabilities and without Stages. *Journal of the American Statistical Association*, **51**, 24 – 53.

Dressel, P. L. (1940). Statistical Semivariants and Their Estimates with Particular Emphasis on Their Relation to Algebraic Invariants. Annals of Mathematical Statistics, **11**, 33 – 57.

Durbin, J. (1959). A Note on the Application of Quenouille's Method of Bias Reduction to the Estimation of Ratios. *Biometrika*, **46**, 477 – 480.

Dwyer, P. S. (1938). Combined Expansions of Products of Symmetric Power Sums and of Sums of Symmetric Power Products with Application to Sampling. *Annals of Mathematical Statistics*, **9**, 1 – 47.

Efron, B. (1979). Bootstrap Methods: Another Look at the Jackknife. *The Annals of Statistics*, **7**, 1 – 26.

Efron, B. (1982). *The Jackknife, the Bootstrap and Other Resampling Plans*. SIAM Monograph no. 38, Philadelphia.

Fisher, R. A. (1930). Moments and Product Moments of Sampling Distributions. *Proceedings of the London Mathematical Society*, **30**, 199 – 238.

Francisco, C. A. and Fuller, W. A. (1991). Quantiles Estimation with a Complex Survey Design. *Annals of Statistics*, **19**, 454 – 469.

Godambe, V. P. (1955). A Unified Theory of Sampling from Finite Populations. *Journal of the Royal Statistical Society*, **B17**, 269 – 278.

Godambe, V. P. (1970). Foundations of Survey Sampling. *The American Statistician*, **24**, 33 – 38.

Gross, S. (1980). Median Estimation in Sample Surveys, in *Proceedings of the Survey Research Methods Section*, American Statistical Association, 181 – 184.

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

Hanurav, T. V. (1966). Some Aspects of Unified Sampling Theory. *Sankhyā*, **A24**, 429 – 436.

Irwin, J. O. and Kendall, M. G. (1944). Sampling Moments of Moments for a Finite Population. *Annals of Eugenics*, **12**, 138 – 142.

Johnson, N. L., Kotz, S. and Balakrishnan, N. (1997). *Discrete Multivariate Distributions*. Wiley, New York.

Kendall, M. G. (1943). *The Advanced Theory of Statistics*. Vol. 1, Charles Griffin, London.

Kendall, M. G. and Stuart, A. (1969). *The Advanced Theory of Statistics, Vol.* 1. Charles Griffin, London.

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.

Kröger, H., Särndal, C-E. and Teikari, I. (1999). Poisson Mixture Sampling: A Family of Designs for Coordinated Selection Using Permanent Random Numbers, *Survey Methodology*, **25**, No 1, 3 – 11.

Mahalanobis, P. C. (1939). A Sample Survey of the Acreage under Jute in Bengal. *Sankhyā*, **4**, 511 – 531.

Mahalanobis, P. C. (1946a). Report on the Bihar Crop Survey: Rabi Season 1943 – 1944. *Sankhyā*, **7**, 269 – 280.

Mahalanobis, P. C. (1946b). Recent Experiments in Statistical Sampling in the Indian Statistical Institute. *Journal of the Royal Statistical Society*, **109**, 325 – 370.

Maritz, J. S. and Jarrett, R. G. (1978). A Note on Estimating the Variance of the Sample Median. *Journal of the American Statistical Association*, **73**, 194 – 196.

Meister, K. and Traat, I. (1997). On the Design-based Distribution of Some Estimators in Survey Sampling. *Theory of Stochastic Processes*, **3**, 324 – 329.

McCarthy, P. J. (1969). Pseudo-replication: Half-samples. *Review of the International Statistical Institute*, **37**, 239 – 264.

McCarthy, P. J. and Snowden, C. B. (1985). *The Bootstrap and Finite Population Sampling.* Vital and Health Statistics, Ser. 2, 95, Public Health Service Publication 85-1369, U.S. Government Printing Office, Washington.

McCullagh, P. (1987). *Tensor Methods in Statistics.* Chapman & Hall, London.

Norlén, U. and Waller, T. (1979). Estimation in a Complex Survey – Experiences from a Survey of Buildings with Regard to Energy Usage. Statistisk Tidskrift, **17**, 109 – 124.

Neyman, J. (1934). On the Two Different Aspects of the Representative Method: the Method of Stratified Sampling and the Method of Purposive Selection. *Journal of the Royal Statistical Society*, **97**, 558 – 606.

Neyman, J. (1938). Contribution to the Theory of Sampling Human Populations. *Journal of the American Statistical Association*, **33**, 101 – 116.

Ollila, P. K. (1996). *Variance Estimation under the Framework of Metasampling.* Unpublished Licentiate Thesis, Department of Statistics, University of Helsinki.

Politis, D. N. and Romano, J. P. (1994). Large Sample Confidence Regions Based on Subsamples under Minimal Assumptions. *Annals of Statistics*, **22**, 2031 – 2050.

Politis, D. N., Romano, J. P. and Wolf, M. (1999). *Subsampling.* Springer, New York.

Quenouille, M. H. (1949). Approximate Tests of Correlation in Time Series. *Journal of the Royal Statistical Society*, **B11**, 68 – 84.

Presnell, B. and Booth, J. G. (1994). *Resampling Methods for Sample Surveys.* Technical Report No. 470, Department of Statistics, University of Florida.

Raj, D. and Khamis, S. H. (1958). Some Remarks on Sampling with Replacement. *Annals of Mathematical Statistics*, **29**, 550 – 557.

Rao, J. N. K. and Wu, C. F. J. (1985). Inference from Stratified Samples: Second-Order Analysis of Three Methods for Nonlinear 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., Wu, C. F. J. and Yue, K. (1992). Some Recent Work on Resampling Methods for Complex Surveys. *Survey Methodology*, **18**, n:o 2, 209 - 217.

Rosén, B. (1997). On Sampling with Probability Proportional to Size. *Journal of Statistical Planning and Inference*, **62**, 159 – 191.

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

Sitter, R. R. (1992a). Comparing Three Bootstrap Methods for Survey Data. *The Canadian Journal of Statistics*, **20**, n:o 2, 135 - 154.

Sitter, R. R. (1992b). A Resampling Procedure for Complex Survey Data. *Journal of the American Statistical Association*, **87**, 755 - 765.

Stafford, J. E. and Bellhouse, D. R. (1997). A Computer Algebra for Sample Survey Theory. *Survey Methodology*, **23**, 3 – 10.

Sukhatme, P. V. (1954). *Sampling Theory of Surveys with Applications*. Iowa State College Press, Ames.

Traat, I. (2000). Sampling Design as a Multivariate Distribution. In *New Trends in Probability and Statistics*, 5, T. Kollo et al. (eds) 195 - 207.

Traat, I., Bondesson, L. and Meister, K. (2000). *Distribution Theory for Sampling Designs*. Research Report No. 2. Department of Mathematical Statistics. Umeå University.

Traat, I., Meister, K., and Sõstra, K. (2001). Statistical Inference in Sampling Theory. *Theory of Stochastic Processes*, **7**, 301 – 316.

Tukey, J. W. (1950). Some Sampling Simplified. *Journal of the American Statistical Association*, **45**, 501 – 519.

Särndal, C - E., Swensson, B. and Wretman, J. (1992). *Model Assisted Survey Sampling*, Springer-Verlag, New York.

Wishart, J. (1952). Moment Coefficients of the k-statistics in Samples from a Finite Population. *Biometrika*, **39**, 1 – 13.

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

Woodruff, R.S. (1952). Confidence Intervals for Medians and Other Position Measures. *Journal of the American Statistical Association*, **78**, 879 – 884.

Yates, F. (1949). *Sampling Methods for Censuses and Surveys*. Charles Griffin and Company, London.

Yung, W. and Rao, J. N. K. (1996). Jackknife Linearization Variance Estimators Under Stratified Multi-Stage Sampling. *Survey Methodology*, **22**, n:o 1, 22 – 31.

Appendix A: Expectations of the Random Group and Jackknife Variance Estimators

The dependent random group method with $n_a = n_b A$ takes the expectation (denoted by E^*) over all variance estimates within $\underline{k}_a$, which is

$$E^*[\hat{V}(\hat{\theta}_a)\,|\,\underline{k}_a] = \frac{E^*(\overline{\hat{\theta}_b^2}\,|\,\underline{k}_a) - E^*(\overline{\hat{\theta}_b}^2\,|\,\underline{k}_a)}{A-1}$$

$$= \left(\frac{1}{A-1}\right)\left[\left(1-\frac{1}{A}\right)E(\hat{\theta}_b^2\,|\,\underline{k}_a) - \frac{1}{A}\sum_{\underline{k}_b,\underline{k}_b'|r=0,\underline{k}_a} \frac{\hat{\theta}_{\underline{k}_b}\hat{\theta}_{\underline{k}_b'}}{\binom{n_a}{n_b}\binom{n_a-n_b}{n_b}}\right]$$

$$= \frac{1}{A}\left[E(\hat{\theta}_b^2\,|\,\underline{k}_a) - \frac{1}{A-1}E(\hat{\theta}_{\underline{k}_b}\hat{\theta}_{\underline{k}_b'}\,|\,r=0,\underline{k}_a)\right]$$

where r is the number of joint elements in the pair $\underline{k}_b, \underline{k}_b'$ following the decomposition approach in Section 7.3.2. In terms of joint elements the expectation of the jackknife variance estimator in SI sampling is

$$E^*[\hat{V}(\hat{\theta}_a)\,|\,\underline{k}_a] = (A-1)\left[E^*(\overline{\hat{\theta}_{comb}^2}\,|\,\underline{k}_a) - E^*(\overline{\hat{\theta}_{comb}}^2\,|\,\underline{k}_a)\right]$$

$$= (A-1)\left[\left(1-\frac{1}{A}\right)E(\hat{\theta}_{An_b}^2\,|\,\underline{k}_a) - \frac{1}{A}\sum_{\underline{k}_b,\underline{k}_b'|r=(A-2)n_b} \frac{\hat{\theta}_{\underline{k}_b}\hat{\theta}_{\underline{k}_b'}}{\binom{n_a}{(A-1)n_b}\binom{(A-1)n_b}{(A-2)n_b}}\right]$$

$$= \frac{(A-1)^2}{A}E(\hat{\theta}_b^2\,|\,\underline{k}_a) - \frac{A-1}{A}E(\hat{\theta}_{\underline{k}_b}\hat{\theta}_{\underline{k}_b'}\,|\,r=(A-2)n_b,\underline{k}_a).$$

Appendix B: Program for Example in Section 4.3

```
/* Program demonstrating the k2 statistics example in Section 4.3 */
* ==== Creating  data ==== ;
data ExampleData;
 input x1 x2 x3 x4 x5 x6;
 cards;
1 4 8 9 13 14
;
run;
* ===== Processing the parameters of the population and creating the SIR sample space
=====;
data SIRSampleSpace ;
 set ExampleData ;
 * ------ Definitions ------ ;
 array x(6) x1-x6;
PopulationSize=6;  SampleSize=4;  ResampleSize=3;
M=(x1+x2+x3+x4+x5+x6)/PopulationSize; * Mean;
Vx=(x1**2+x2**2+x3**2+x4**2+x5**2+x6**2)/PopulationSize - M**2; * Variance of x ;
M4=((x1-M)**4+(x2-M)**4+(x3-M)**4+(x4-M)**4+(x5-M)**4+(x6-M)**4)/PopulationSize; *4th
Central Moment;
M2_2=(((x1-M)**2+(x2-M)**2+(x3-M)**2+(x4-M)**2+(x5-M)**2+(x6-
M)**2)/PopulationSize)**2;*2ndCM squared;
kappa4=M4-3*M2_2; * 4th Cumulant;
kappa2_2=M2_2; * 2nd Cumulant squared;
Vk2 = kappa4/SampleSize + 2*kappa2_2/(SampleSize-1); * Variance of k2 ;
EVk2b =(1/ResampleSize - 1/SampleSize)*kappa4 + 2*(1/(ResampleSize-1) - 1/(SampleSize-
1))*kappa2_2;
* Expectation (SIR) of Conditional Variance of k2b (SI) ;
Qlin= ResampleSize/(SampleSize-ResampleSize); * Linear Case Coefficient Qlin;
QEdefin=Vk2/EVk2b; * Real Coefficient QE by definition;
QEalt=Qlin*((SampleSize-1)*kappa4+2*SampleSize*kappa2_2)/((SampleSize -  1)*kappa4 +
    2*SampleSize*kappa2_2* ResampleSize /  (ResampleSize-1)); * Real Coefficient QE in
terms of Qlin;
* ------- Creating SIR sample space for n=4 --------;
 do i1=1 to 6; do i2=1 to 6; do i3=1 to 6; do i4=1 to 6;
  sm=(x(i1)+x(i2)+x(i3)+x(i4))/SampleSize;   * Sample Mean;
  sm4=((x(i1)-sm)**4+(x(i2)-sm)**4+(x(i3)-sm)**4+(x(i4)-sm)**4)/SampleSize; * 4th
Central Sample Mom;
  sm2_2=(((x(i1)-sm)**2+(x(i2)-sm)**2+(x(i3)-sm)**2+(x(i4)-sm)**2)/SampleSize)**2;
*Square of 2nd CSM;
  s2=SampleSize*((x(i1)**2+x(i2)**2+x(i3)**2+x(i4)**2)/SampleSize -
    ((x(i1)+x(i2)+x(i3)+x(i4))/SampleSize)**2)/(SampleSize-1); * Estimator of Variance
of Variable x;
  k4=SampleSize**2*((SampleSize+1)*sm4-3*(SampleSize-1)*sm2_2) /((SampleSize-
1)*(SampleSize-
    2)*(SampleSize-3)); * Fourth k-statistics;
  k2_2=SampleSize**2*sm2_2/((SampleSize-1)**2);    * Square of Second k-statistics;
  Vk2b=(SampleSize-ResampleSize)*(SampleSize*ResampleSize-ResampleSize-SampleSize-1)*k4
      /(ResampleSize*(ResampleSize-1) * SampleSize*(SampleSize+1)) +2*(SampleSize-
        ResampleSize)*k2_2/((ResampleSize-1)*(SampleSize+1)); * Conditional variance
of  k2b;
   output;
 end; end; end; end;
run;
options nocenter;
proc means data=SIRSampleSpace mean var;
  var s2 Vx Vk2 Vk2b EVk2b QEdefin QEalt;
  title1 's2 is unbiased estimator of Vx';
  title2 'Theoretical Vk2 equals variance of s2';
  title3 'Expectation of Vk2b equals theoretical EVk2b';
  title4 'Theoretical QE equals alternate expression of QE';
run;
```

Appendix C: Simulation Program Package

```
* ============ Simulations.sas ============;
options nocenter;
* ---------- Globally valid macro variables ----------;
%global CapN n_a parm num ;
%let CapN=9; * Size of Population;  %let n_a=6;  * Size of Sample;
%let parm=7; * Number of Parameters;
%let num=26; * Number of Variance Estimators (Taylor/Woodruff together);
%let DataId=2; * Selection of the data (1: Municipalities, 2: Enterprises);

* ------ Defining the second and third phase resample sizes with conditions of the
linear case --- ;
%inc 'd:\vaikkari02\Test\Definitions.sas';
%BEGINNI;
* ---------- Evoking macro programs ----------;
* Utilising the macro values from BEGINNI ;
%inc 'd:\vaikkari02\Test\MacroSums.sas'; * Specific sum macros for speeding up
calculations;
%inc 'd:\vaikkari02\Test\Macros.sas'; * Macros used in different sas-programs;
%inc 'd:\vaikkari02\Test\MakingArrays.sas'; * Creation of arrays and some value
preparations;
%inc 'd:\vaikkari02\Test\FirstPhaseSelections.sas';
            * First phase selections with calculations of estimators and resampling
properties;
%inc 'd:\vaikkari02\Test\SecondPhaseSelections.sas';
            * Second phase selections with calculations of estimators :
%inc 'd:\vaikkari02\Test\SecondPhasePairSelections.sas';
            * Second phase pair selections for decomposition, random groups and
general jackknife;
%inc 'd:\vaikkari02\Test\ThirdPhaseSelections.sas';
            * Third phase selections with calculations of estimators ;
%inc 'd:\vaikkari02\Test\VarianceEstimators.sas'; * Calculating 27 different variance
estimators ;

* --- METAMACR is the "meta"macro for other macros to be run at the end of the program -
-;
%macro METAMACR;
* ---------- Basic data DATANAME ----------;
* Note: in order to avoid resource limitations, we put the data to a specified library;
%global DATANAME;
libname POlib 'c:\POtilapais'; %let DATANAME= POlib.pohja ;
data &DATANAME ;
* Defining the basic arrays and creating some coefficients for the data
(MakingArrays.sas);
%ARRMACR ;
* ---------- Data identification ----------;
* 1: Unemployment Data (x: number of unemployed, y: labour force / 100 (for the ratio
to be in %);
* 2: Enterprise Data (x: gross product, y: number of personnel);

%if &DataId=1 %then %do;
x1=2444;  x2=809;  x3=4453;  x4=192;  x5=319;  x6=404;  x7=368;  x8=395;  x9=129;
x10=289 ; x11=53;  x12=280;  x13=50;  x14=83 ;
y1=145.05;y2=52.92;y3=276.92;y4=15.01;y5=22.98;y6=20.17;y7=20.91;y8=28.10;y9=6.44;y10=1
8.53;y11=3.30;y12=22.78;y13=2.25;y14=6.92 ;
%end;
%else %if &DataId=2 %then %do;
x1=447;x2=12866;x3=62960;x4=13767;x5=90;x6=20066;x7=1645;x8=33832;x9=7574;
y1=2;  y2=8 ;   y3=5 ;   y4=9 ;   y5=1 ;y6=10 ;  y7=3 ;  y8=4 ;   y9=6 ;
%end;

* The population with two variables is sorted by x, where x is the variable where the
order statistics are calculated; * Three design vector levels are prepared (i for
sampling, j for resampling and jj for second-phase resampling);
%BEGISORT ;
* This macro includes all the selection phases;
%SELECT1;
```

```
run;

* This macro presents the results;
%inc 'd:\vaikkari02\Test\PropertiesAndTables.sas';
%RESULTS ;
%mend METAMACR;
* Submitting the metamacro;
%METAMACR ;

* ======= Definitions.sas ========;
* Preparations for other macros and later calculations in the program ;
%MACRO BEGINNI;
* In SAS it is easier to make the calculations in the data phase than at the macro
level;
%global BWO_n BWO_mlow BWO_mup alan_a LMCO UMCO linb linblow linbhigh;
data temporary;
  * the resample size required by the linear case with lower and upper integers;
  lin_b=1/(2/&n_a - 1/&CapN) ;   lin_blow=int(lin_b);   lin_bhigh=ceil(lin_b);
  * to the macro level;
  call symput('linb',lin_b); call symput('linblow',lin_blow); call
symput('linbhigh',lin_bhigh);
  * BWO: resample size (BWOn) and multiplication size (BWOm), n rounded to the integer
value;
  BWOn = int(&n_a - (1 - &n_a / &CapN)) ;  BWOm = &CapN*BWOn / &n_a**2 ;
  * If BWOm is one or below, the procedure is repeated, one less resample size and a
new m ;
  if BWOm<=1 then do;  BWOn=BWOn-1;  BWOm=&CapN*(1-(1/&n_a - 1/&CapN))/&n_a ;   end;
  * Lower and upper values of m;
  BWOml=int(BWOm); BWOmu=ceil(BWOm);
  * Probabilities for randomisation;
  bp_low=((1-&n_a/&CapN)/(&n_a*(&n_a-1)) - BWOmu*(1-BWOn/(&n_a*BWOmu)) /
(BWOn*(&n_a*BWOmu-1)))
          / (BWOml*(1-BWOn/(&n_a*BWOml)) / (BWOn*(&n_a*BWOml-1))
            - BWOmu*(1-BWOn/(&n_a*BWOmu)) / (BWOn*(&n_a*BWOmu-1)));
  bp_up=1-bp_low;
  * to the macro variables;
  call symput('BWO_n',BWOn);  call symput('BWO_mlow',BWOml);  call
symput('BWO_mup',BWOmu);
run;

data temporary2;
  * Maximum number of random groups;
  maxrg=int(&n_a/2);  call symput('max_rg',maxrg);
  * the resample size of the second resampling phase with respect to the original
resample size;
  lin_c=1/((1/&linb - 1/&n_a)**2/(1/&n_a - 1/&CapN) + 1/&linb);  call
symput('linc',lin_c);
  * Woodruff lower and upper confidence interval terms;
  MTERM=(&CapN-&n_a)*0.25/(&n_a*(&CapN-1));  LMC=0.5-(1.96*MTERM);
UMC=0.5+(1.96*MTERM);
  call symput('LMCO',lmc); call symput('UMCO',umc);
run;

* ----------------- Some macro variable information ----------------------;

%global h o n_1 nb_1 olow;
* for some constructed variables including macro variable information only one
  letter long macro variable is suitable;
%let o=&linblow; %let h=&linbhigh;
* giving values for future use;
%let n_1=%eval(&n_a -1); %let zii=%eval(&n_1 -1); %let oz=%eval(&o - 1);
* specific construction terms;
%let olow=_&oz ; %let nb_1 = _&zii;

%MEND BEGINNI ;

* ================ MakingArrays.sas ============== ;
```

```
* Defining the arrays with the packages created in Macros.sas;

%MACRO ARRMACR;

* f-beginning for a wr sample, g-beginning for a wor sample:
  variable places f1e1, f1es1, f1eq1, f1v1, f1la1 ….;
* e = expectation of variable, es = expectation of squared variable, eq = square of
expectation ;
* v = variance of variable, la = number of accepted values of variable;
* number of parameters in "parm";
* Input info: 1) dimension of population, 2) number of parameters,
              3) --> beginnings of variables ;

%ARRPROP(&CapN,&parm,f,g,ff,gg,frs,grs,fbl,fbu,gbl,gbu,dg,grg,gj2_,sif,sifb);
%ARRPROP(&CapN,&parm,frf,fde,gde,.,.,.,.,.,.,.,.,.,.,.,.,.,.,.,.);

* References: f,g --> first-phase sampling properties,
              ff,gg --> second-phase sampling properties,
              rs,rf --> rescaling functions
                        bl,bu --> BWO randomisation lower and upper integers
              rg --> random groups,
              j2_ --> jackknife with two out of three random groups
              dg --> decomposition approach, group term
              sif --> decomposition approach, final properties (theory of
conditionality)
              sifb --> decomposition approach, final properties (weight adjustments)
              de --> post-design vector
;
* For second-phase resampling;
%ARRTWO(&n_a,&parm,fzu); %ARRTWO(&n_a,&parm,gzu);

* Linear case coefficients;
 array Qlwor(&n_a);  array Qlwr(&n_a);

* Internal rescaling: general weighting term for all sample units;
 array worsa(&n_a); array wrsa(&n_a);

* Internal rescaling: resample-specific weighting term ;
 array worsb(&n_a); array wrsb(&n_a);

* Variables in dimension of population; * x,y as study variables;
* i, j, jj, dj design vector variables (sample, resample, second phase resample, second
resample for sample pairs);
* alu variables for order information; * zi dimension of first phase counter;
* rb and brb for counter terms in decomposition;
* prton is probability term for r (decomposition, number of joint units in resample
pair);
* Input info: 1) dimension of population, 2) --> beginnings of variables ;

%ARR(&CapN,x,y,i,j,jj,dj,jalu,jjalu,ialu,zi,rb,brb,prton,.,.);

* minimum length for vector variables (speeding up calculations);
length i1-i&CapN 3. j1-j&CapN 3. jj1-jj&CapN 3.  dj1-dj&CapN 3. ;

* Calculating coefficients for simple random resampling with replacement (wr) and
without replacement (wor);

  do dw=1 to &n_a;
  * Linear case coefficients (Q1) ;
    Qlwor(dw)=(1/&n_a - 1/&CapN)/(1/dw - 1/&n_a);  Qlwr(dw)=(1/&n_a - 1/&CapN)/(1/dw -
1/(&n_a*dw));
  * Internal rescaling: general weighting term for all sample units  ;
    worsa(dw)=((1-sqrt(Qlwor(dw)))*&CapN/&n_a) ;  wrsa(dw)=((1-
sqrt(Qlwr(dw)))*&CapN/&n_a) ;
  * Internal rescaling: resample-specific weighting term ;
    worsb(dw)=(sqrt(Qlwor(dw))*&CapN/dw) ;  wrsb(dw)=(sqrt(Qlwr(dw))*&CapN/dw);
  end;
```

```
* Preparations for the design vector calculations;
  ades= &linblow **2 - &linb * &linblow;  bdes= 2 * &linblow; cdes= 1 - &linb ;
  desvco1= (-bdes-sqrt(bdes**2-4*ades*cdes)) / (2*ades); desvco2= (-bdes+sqrt(bdes**2-
4*ades*cdes)) / (2*ades); desvcoef=max(desvco1,desvco2);

%mend ARRMACR;

* ============== FirstPhaseSelections.sas ==============;
* Defining the sample space (without replacement);

%MACRO SELECT1;
ifa=&CapN;
* -------- Macro creating do loops for without replacement sampling -------- ;
* Input info: 1) dimension of population, 2) sample size ;
%DO_SFI(&CapN,&n_a);

* Real time information ; put 'POPULATION ' ifa 'SAMPLE ' i1-i&CapN;

* ---------- macro calculating sums ----------;
* utilising array variables i, x ja y for calculations of sums, sums of squares and
product of two variables
  and sum of i;
* a referring to first phase sample; * s describing sum;
* producing variables asx, asy, asxq, asyq, asxy, an ;
* Input info: 1) dimension of population, 2) phase, 3) vector identification,
             4) variable 1, 5) variable 2, 6) end of frequency variable;
%SUMPAC(&CapN,a,i,x,y,n);

* ---------- order numbers for x ----------;
* Input info: 1) size of population, 2) inclusion variable, 3) sample size;
%ORDPA(&CapN,i,an);

* ---------- estimators of totals ----------;
* expanding weight N/na, statistics to be expanded: asx, asy, asxq, asyq, asxy;
* producing variables atx, aty, atxq, atyq, atxy ;
* Input info: 1) dimension of population, 2) phase, 3) variable 1, 4) variable 2,
             5) end of frequency variable;
%TOTPAC(&CapN,a,x,y,n);

* ---------- estimator of ratio ----------;
* atx / aty and its square; * producing variables arxy ja arxyq ;
if aty>0 then do;
   arxy=atx/aty; arxyq=arxy**2;
end; else do;
   arxy=.; arxyq=.;
end;

* ------ estimating variance, covariance and correlation ----------;
* denominator n-1;
* producing variables avy, avx, acvxy, acrxy;
* Input info: 1) dimension of population, 2) phase, 3) variable 1, 4) variable 2, 5)
end of frequency
             variable;
if an>1 then do; %DISPAC(&CapN,a,x,y,n); end;

* ------ term for taylor approximation of variance S2x ------;
* Input info: 1) dimension of population, 2) id-vector variable, 3) sum of x in sample,
             4) variance of x in sample, 5) sample size;

%VATEXSUM(&CapN,i,asx,avx,&n_a);

* ----- medians and maximums -------;
* Input info: 1) dimension of population, 2) phase, 3) variable 1, 4) variable 2, 5)
end of frequency variable;
%QUAPA(&CapN,a,x,y,n);
```

```
* ---------- GIVING VALUES ----------;
* Preparing variables of phase 2; * Variables afle1, afleq1, afles1, aflv1 etc. which
get value . ;
* f for with replacement, g for without replacement;
* Input info: 1) sample size, 2) value, 3) denoting the function (f or g),
              4) number of estimated parameters;

%VALPAC(&n_a,.,f,&parm); %VALPAC(&n_a,.,g,&parm); %VALPAC(&n_a,.,dg,&parm);
%VALPAC(&n_a,.,grg,&parm); %VALPAC(&n_a,.,gj2_,&parm); %VALPAC(&n_a,.,sif,&parm);
%VALPAC(&n_a,.,sifb,&parm); %VALPAC(&n_a,.,frs,&parm); %VALPAC(&n_a,.,frf,&parm);
%VALPAC(&n_a,.,grs,&parm); %VALPAC(&n_a,.,fbl,&parm); %VALPAC(&n_a,.,fbu,&parm);
%VALPAC(&n_a,.,fde,&parm); %VALPAC(&n_a,.,gde,&parm); %VALPASPE(&n_a,.,fzu,&parm);
%VALPASPE(&n_a,.,gzu,&parm);

* References: f,g --> first-phase sampling properties,
              rs,rf --> rescaling functions
                        bl,bu --> BWO randomisation lower and upper integers
              rg --> random groups,
              j2_ --> jackknife with two out of three random groups
              dg --> decomposition approach, group term
              sif --> decomposition approach, final properties (theory of
conditionality)
              sifb --> decomposition approach, final properties (weight adjustments)
              de --> post-design vector
              zu --> second-phase sampling level
;

* ------------- Second-phase sampling ------------------- ;
* from program SecondPhaseSelections.SAS;

%SELECT2 ;

* ---------- CALCULATING PROPERTIES OF ESTIMATORS ----------;
* In SELECT2 macro we calculate sum of estimates (with CUMU macros)
  and number of existing values (with COUN macros). With them
  we calculate here: expectation of estimator, expectation of square of estimator,
  square of expectation, variance of estimator; * Variables fle1 fles1 fleq1 flv1 etc.;
* Input info: 1) sample size, 2) number of estimated parameters,
              3) ---->  WR/WOR functions to be calculated;

%PACK1(&n_a,&parm,f,g,grs,fbl,fbu,grg,gj2_,frs,frf,gde,fde,.,.,.,.,.,.,.,.,.,.,.);
%PACKSIF(&n_a,&parm,sif) ; %PACKSPE(&n_a,&parm,gzu);

* ----------------- VARIANCE ESTIMATORS ---------------------;
%ESTIMA;

* Resample size terms used in two-phase and two-size resampling;
n_aeab=&linb; n_aeac=&linc;
* Taylor calculation for correlation not conducted;
tayg2=.;

output &DATANAME;

   %END_S(&n_a);

%mend SELECT1;

* ============ SecondPhaseSelections.sas ============;
* Defining the resample space (with replacement) for every sample;
%MACRO SELECT2;

* ------- Macro creating do loops for with replacement sampling ------- ;
* Input info: 1) dimension of population, 2) sample size, 3) sampling level indicator
letter ;
%DO_SWR(&CapN,&n_a,b);

* Here checking, whether without replacement sample;
```

```
* also calculating number of twos for the post-design vector (practical need);
%SEPAR(&CapN);

* ---------- macro calculating sums ------------;
* utilising array variables j, x ja y for calculations of sums, sums of squares and
product of two variables and sum of j;
* b referring to second phase sample; * s describing sum;
* producing variables bsx, bsy, bsxq, bsyq, bsxy, bn ;
* Input info: 1) dimension of population, 2) phase, 3) vector identification,
              4) variable 1, 5) variable 2, 6) end of frequency variable;
%SUMPAC(&CapN,b,j,x,y,n);

* ---------- Rao, Wu & Yue weight rescaling --------- ;
if bn>0 then do;
* sum variables for without and with replacement cases;
mesuwor=0; mesuwr=0;
* indicator variable for finding median location ;
* getting value 0 when 0.5 is exceeded with sum variable;
meworst=1; mewrst=1;
* no values for medians in the beginning;
bmexrs=.; bmexrf=.;

do bii=1 to &CapN;
      if worb=1 then do;
      if meworst then mesuwor=mesuwor+i(bii)*((1-sqrt(Qlwor(bn)))*&CapN/&n_a
         +j(bii)*sqrt(Qlwor(bn))*&CapN/bn)/&CapN ;
        if mesuwor>=0.5 and meworst then do; bmexrs=x(bii); meworst=0; end;
      end;
    if mewrst then mesuwr=mesuwr+i(bii)*((1-
                  sqrt(Qlwr(bn)))*&CapN/&n_a+j(bii)*sqrt(Qlwr(bn))*&CapN/bn)/&CapN ;
      if mesuwr>=0.5 and mewrst then do; bmexrf=x(bii); mewrst=0; end;
end;

* ---------- Macros calculating sums (rescaling) ----------;
* See previous macro for details;
* Input info: 1) dimension of population, 2) phase, 3) vector identification,
              4) variable 1, 5) variable 2, 6) rescaling code,
              7) end of frequency variable;

%SUMPAWO(&CapN,b,j,x,y,rs,n); %SUMPAWR(&CapN,b,j,x,y,rf,n);

* Order numbers for variable x;
%ORDPAC(&CapN,j,bn);

* Principle for post-design vector approach: we catch a wr sample of size
  (upper limit for randomisation + 1) with a vector e.g. 120110
  (including one 2, three 1:s and two 0:s), and all the corresponding
  combinations. We weight the 2 alternative with 1 and the 1:s with
  the design vector expanding factor q. The result is a wor sample
  (here 110110 with weights q10qq0). Thus the reason to use this
  specific wr sample is purely technical in order to go through all
  the possible samples with this weighting;

PostCond=(bn=&linbhigh+1 and numoftwo=1 and numofone=bn-2);

if PostCond then do;
  %SUMPDE;
   podeve=0; podin=1;
   do bii=1 to &CapN;
      podeve=podeve+i(bii)*((j(bii)=2) + (j(bii)=1)*desvcoef)/bnde ;
        if podeve>=0.5 and podin then do; bmexde=x(bii); podin=0; end;
   end;

end;

* ---------- estimators of totals ----------;
* expanding weight N/nb, statistics to be expanded: bsx, bsy, bsxq, bsyq, bsxy;
```

```
* producing variables btx, bty, btxq, btyq, btxy ;
* Input info: 1) dimension of population, 2) phase, 3) variable 1, 4) variable 2,
              5) end of frequency variable;
%TOTPAC(&CapN,b,x,y,n);
* post-design vector case;
if PostCond then do;   %TOTPAC(&CapN,b,xde,yde,nde);  end;

* Sum values of rescaled variables equal total values (weighting inside);
btxrs=bsxrs; btyrs=bsyrs; btxrsq=bsxrsq; btyrsq=bsyrsq; btxrsyrs=bsxrsyrs;
btxrf=bsxrf; btyrf=bsyrf; btxrfq=bsxrfq; btyrfq=bsyrfq; btxrfyrf=bsxrfyrf;

* ---------- estimator of ratio ----------------;

if bty>0 then brxy=btx/bty;
if btyrs>0 then brxrsyrs=btxrs/btyrs; if btyrf>0 then brxrfyrf=btxrf/btyrf;
if PostCond then do;   if btyde>0 then brxdeyde=btxde/btyde; end;

* squaring estimators;

btx2=btx**2; bty2=bty**2; btxrs2=btxrs**2; btyrs2=btyrs**2; btxrf2=btxrf**2;
btyrf2=btyrf**2;
if PostCond then do;   btxde2=btxde**2;   btyde2=btyde**2; end;
brxy2=brxy**2; brxrsyrs2=brxrsyrs**2; brxrfyrf2=brxrfyrf**2;
if PostCond then do;   brxdeyde2=brxdeyde**2; end;

* ------ estimating variance, covariance and correlation ----------;
* denominator nb-1;
* BottB from DO_SWR: includes the number of permutations of current
  WR resample vector;
if bottb>1 then do;
* variables bvy, bvx, bcvxy, bcrxy;
%DISPAC(&CapN,b,x,y,n); %DISPAC(&CapN,b,xrs,yrs,n); %DISPAC(&CapN,b,xrf,yrf,n);
if PostCond then do;   %DISPAC(&CapN,b,xde,yde,nde);  end;
end; else do;
bvy=.; bvx=.; bvyq=.; bvxq=.; bcvxy=.; bcrxy=.; bcrxyq=.; bvyrs=.; bvxrs=.; bvyrsq=.;
bvxrsq=.;
bcvxrsy=.; bcrxrsyrs=.; bcrxrsyrsq=.; bvyrf=.; bvxrf=.; bvyrfq=.; bvxrfq=.; bcvxrfy=.;
bcrxrfyrf=.; bcrxrfyrfq=.; bvyrf=.; bvxrf=.; bvyrfq=.; bvxrfq=.; bcvxrfy=.;
bcrxrfyrf=.; bcrxrfyrfq=.;
end;

* ------ medians and maximums ----------;
* Input info: 1) dimension of population, 2) phase, 3) variable 1,
              4) variable 2, 5) end of frequency variable;
%QUAPAC(&CapN,b,x,y,n);

* replacing the estimates to unified form (f beginning);
* for rescaling maximum (6) is not considered;

f1=bvx; f2=bcrxy; f3=brxy; f4=btx; f5=bmex;  f6=bmax; f7=bvy;
frs1=bvxrs; frs2=bcrxrsyrs; frs3=brxrsyrs; frs4=btxrs; frs5=bmexrs;  frs6=.;
frs7=bvyrs;
frf1=bvxrf; frf2=bcrxrfyrf; frf3=brxrfyrf; frf4=btxrf; frf5=bmexrf;  frf6=.;
frf7=bvyrf;

if PostCond then do;
  fde1=bvxde; fde2=bcrxdeyde; fde3=brxdeyde; fde4=btxde; fde5=bmexde;  fde6=.;
fde7=bvyde;
end;

* CUMULATING FOR PROPERTY CALCULATIONS;
* Building blocks for PACK macro; * CUMU for variables, COUN for counts;
* CUMU variables f1e1-f1e6 f1es1-f1es6 etc., COUN variables f11a1-f11a6 etc.;

%CUMU1(&n_a,&parm,f); %CUMU1rs(&n_a,&parm,frf); %CUMU2(&n_a,&parm,g);
%CUMU2rs(&n_a,&parm,grs);
if PostCond then do;   %CUMU2de(&n_a,&parm,gde); end;
```

```
if bn>1 then %CUMUBOL(&n_a,&parm,fbl); if bn>1 then %CUMUBOU(&n_a,&parm,fbu);
%COUN1(&n_a,&parm,f); %COUN1rs(&n_a,&parm,frf); %COUN2(&n_a,&parm,g);
%COUN2rs(&n_a,&parm,grs);
if PostCond then do;   %COUN2de(&n_a,&parm); end;
if bn>1 then %COUNBOL(&n_a,&parm,fbl); if bn>1 then %COUNBOU(&n_a,&parm,fbu);

* ------ SELECTION FOR SAMPLE PAIRS ----------;
* from SecondPhasePairSelections.sas ;
* only for without replacement resamples (worb=1);
* Random groups, jackknife 2 out of 3, decomposition;
if worb=1 then do; haly=0; %VAL(&CapN,0,dj); %SELECT2add; end;

* ---------------- GIVING VALUES ----------------;
* Preparing variables of phase 3;
* Variables bf1e1, bf1eq1, bf1es1, bf1v1 etc. which get value . ;
* f for with replacement, g for without replacement;
* Input info: 1) sample size, 2) value, 3) denoting the function (f or g),
              4) number of estimated parameters;
%VALPAC(bn,.,ff,&parm); %VALPAC(bn,.,gg,&parm);

* -- SAMPLING SECOND RESAMPLE i.e. THIRD PHASE SAMPLING --;
* only for without replacement resamples (worb=1);
if worb=1 then do;   %SELECT3; %PACK3(bn,&parm,gg);  end;

* In practice we need only gg_v_ and its square;
%do vew=1 %to %eval(&n_a-2);
   gzu1_&vew =gg1v&vew ; gzu2_&vew =gg2v&vew ; gzu3_&vew =gg3v&vew ;
   gzu4_&vew =gg4v&vew ; gzu5_&vew =gg5v&vew ; gzu6_&vew =gg6v&vew ; gzu7_&vew
=gg7v&vew ;
%end;

%CUMUext2(&n_a,&parm,gzu); %COUNext2(&n_a,&parm,gzu);

end; * end for bn>0;

%END_S(&CapN);

%MEND SELECT2;

* ============== SecondPhasePairSelections.sas ==============;

%MACRO SELECT2add;

* ------------- Selecting Second Resample of Sample Pair --------------;
%DO_SFAD(&CapN);

* ------------- macro calculating sums ---------------;
* utilising array variables dj, x ja y for calculations of sums, sums of squares and
product of two variables and sum of dj;
* db referring to second phase sample; * s describing sum;
* producing variables dbsx, dbsy, dbsxq, dbsyq, dbsxy, dbn ;
* Input info: 1) dimension of population, 2) phase, 3) vector identification,
              4) variable 1, 5) variable 2, 6) end of frequency variable;
%SUMPAC(&CapN,db,dj,x,y,n);
if dbn>0 then do;
* Order numbers;
%ORDPAC(&CapN,dj,dbn);
* ------------- estimators of totals ------------------;
* expanding weight N/nb, statistics to be expanded: dbsx, dbsy, dbsxq, dbsyq, dbsxy;
* producing variables dbtx, dbty, dbtxq, dbtyq, dbtxy ;
* Input info: 1) dimension of population, 2) phase, 3) variable 1, 4) variable 2,
              5) end of frequency variable;
%TOTPAC(&CapN,db,x,y,n);
if dbty>0 then dbrxy=dbtx/dbty;
* squaring;
dbtx2=dbtx**2; dbty2=dbty**2; dbrxy2=dbrxy**2;
```

```
* ------ estimating variance, covariance and correlation ----------;
* denominator nb-1;
* variables dbvy, dbvx, dbcvxy, dbcrxy;
if dbn>1 then do; %DISPAC(&CapN,db,x,y,n); end;
else do; dbvy=.; dbvx=.; dbvyq=.; dbvxq=.; dbcvxy=.; dbcrxy=.; dbcrxyq=.; end;

* Order Statistics;
%QUAPAC(&CapN,db,x,y,n);

* ---------- NUMBER OF JOINT ELEMENTS ----------;

numsim=0;
do sii=1 to &CapN;  numsim=numsim+((j(sii)=dj(sii)) & (j(sii)=1)); end;

* replacing the estimates to unified form (wor samples ---> g);
dg1=dbvx; dg2=dbcrxy; dg3=dbrxy; dg4=dbtx; dg5=dbmex;  dg6=dbmax; dg7=dbvy;

* Calculating for random groups;
if numsim=0 then do;   %CUMUD2(&n_a,&parm,grg);  %COUND2(&n_a,&parm,grg);  end;

* Calculating for jackknife 2 out of 3;
if &n_a>=6 and numsim=&n_a-4 then do; %CUMUD2(&n_a,&parm,gj2_);
%COUND2(&n_a,&parm,gj2_); end;

* ---------- DECOMPOSITION CALCULATIONS ----------;

maxlimit=(2*bn-&n_a>0)*(2*bn-&n_a); maxcapit=(2*&n_a-&CapN>0)*(2*&n_a-&CapN);

* the limit for the decomposition weight adjustment method; * indice do not begin from
zero!;

 if n_adebn=. and maxlimit<=(&n_a-1)*(bn**2)/(&n_a**2) then do;
     do wix=maxcapit to &n_a;
          if wix<&n_a then do;
          rb(wix+1)=wix*(bn**2)/(&n_a**2); * rb value;
             if rb(wix+1)<maxlimit then haly=1;
            brb(wix+1)=ceil(rb(wix+1))-rb(wix+1); * probability of lower r alternative;
            ruba=rb(wix+1);
          end;
          term5=gamma(&n_a+1)/(gamma(wix+1)*gamma(&n_a-wix+1));
          term6=gamma(&CapN-&n_a+1)/(gamma((&CapN-&n_a)-(&n_a-wix)+1)*gamma(&n_a-
wix+1));
          * probability based on combinatory terms; prton(wix+1)=term6*term5*term4;
          if haly=0 then do; n_adebn=bn; haly=1; end;
     end;
   end;
  term1=(gamma(bn+1)*gamma(&n_a-bn+1)/gamma(&n_a+1))**2; * 1 / ( na nb )2 ;
  term2=(gamma(bn+1)*gamma(&CapN-bn+1)/gamma(&CapN+1))**2; * 1 / ( N nb )2;
  term4=gamma(&n_a+1)*gamma(&CapN-&n_a+1)/gamma(&CapN+1); * 1 / (N na) ;

  do tii = maxlimit to bn;
    * number of distinct elements;
    distinct=tii+2*(bn-tii);
    otherterm=0;
    if numsim=tii and maxlimit<=tii<=bn then do;
          term3c=gamma(&CapN-distinct+1)/(gamma(&CapN-&n_a+1)*gamma(&n_a-distinct+1));
          pstar=term3c*term1*term4;                    * ( N-d  na-d ) ;
          wholeterm=term1*(1-term2/pstar);
            do wix=maxcapit to &n_a-1;
                maxavailable=
            (rb(wix+1)=int(rb(wix+1)))*(rb(wix+1)=numsim)*prton(wix+1)+
                (rb(wix+1)>int(rb(wix+1)))*(ceil(rb(wix+1))=numsim)*((1-
brb(wix+1))*prton(wix+1))+

(rb(wix+1)>int(rb(wix+1)))*(int(rb(wix+1))=numsim)*(brb(wix+1)*prton(wix+1));
               otherterm=otherterm+maxavailable;
               lower_rb=int(rb(wix+1)); upper_rb=ceil(rb(wix+1));
```

```
          brbvalue=brb(wix+1); probused=prton(wix+1);
           end;
           * denominator of sum of pairs for expectation;
         term7=gamma(&n_a-bn+1)/(gamma((&n_a-bn)-(bn-numsim)+1)*gamma(bn-numsim+1));
         term8=gamma(bn+1)/(gamma(bn-numsim+1)*gamma(numsim+1));
           kurt=sqrt(term1)/(term7*term8);
           otherterm=otherterm*sqrt(term1)/(term7*term8);
       %CUMUDG2(&n_a,&parm,sif);
   end;
  end;

end; * end for condition bn>0;

%END_Sfad(&CapN);

%MEND SELECT2add;

* ============ ThirdPhaseSelections.sas ============;
* Defining the second phase resample space (without replacement) for every resample;


%MACRO SELECT3;

%DO_SIII(&CapN,bn,c);

* ---------- macro calculating sums ----------;
* utilising array variables jj, x ja y for calculations of sums, sums of squares and
product of two variables and sum of jj;
* c referring to third phase sample; * s describing sum; * producing variables csx,
csy, csxq, csyq, csxy, cn ;
* Input info: 1) dimension of population, 2) phase, 3) vector identification, 4)
variable 1, 5) variable 2, 6) end of frequency variable;

%SUMPAC(&CapN,c,jj,x,y,n);

* ORDER NUMBERS;
%ORDPACB(&CapN,jj,cn);

* ---------- estimators of totals ----------;
* expanding weight N/nc, statistics to be expanded: csx, csy, csxq, csyq, csxy; *
producing variables ctx, cty, ctxq, ctyq, ctxy ;
* Input info: 1) dimension of population, 2) phase, 3) variable 1, 4) variable 2,
             5) end of frequency variable;

if cn>0 then do; %TOTPAC(&CapN,c,x,y,n);  end;

if cty>0 then crxy=ctx/cty;

* squaring estimators;
ctx2=ctx**2;  cty2=cty**2; crxy2=crxy**2;

* ------ estimating variance, covariance and correlation ----------;
* denominator nc-1;
if bottc>1 and cn>1 then do; %DISPAC(&CapN,c,x,y,n);  end;
else do; cvy=.; cvx=.; cvyq=.; cvxq=.; ccvxy=.; ccrxy=.; ccrxyq=.; end;
* ------ medians and maximums ---------;
* Input info: 1) dimension of population, 2) phase, 3) variable 1,
             4) variable 2, 5) end of frequency variable;
if cn>0 then do; %QUAPACC(&CapN,c,x,y,n);  end;
ff1=cvx; ff2=ccrxy; ff3=crxy; ff4=ctx; ff5=cmex;  ff6=cmax; ff7=cvy;

* CUMULATING FOR PROPERTY CALCULATIONS;
* Building blocks for PACK macro; * CUMU for variables, COUN for counts;
* CUMU variables ff1e1-ff1e6 ff1es1-ff1es6 etc. COUN variables ff11a1-ff11a6 etc.;

if cn>0 then do;  %CUMUc2(bn,&parm,gg);  %COUNc2(bn,&parm,gg); end;
```

```
%END_S(&CapN);

%MEND SELECT3;

* ========= VarianceEstimators.sas =========;

* 1) Woodruff method for median;
* 2) External linear case scaling coefficient ;
* 3) Bias correction with two resample sizes;
* 4) Bias correction in two phases;
* 5) MSE reducing variance estimator;
* 6) Strict randomisation for resamples;
* 7) Bootstrap without replacement with randomisation;
* 8) Post-design vector approach;
* 9) Taylor s linearisation;
* 10) Dependent random groups method;
* 11) Jackknife 2 out of 3;
* 12) Decomposition with conditionality assumption;
* 13) Decomposition with weight adjustments;

%macro estima;

* +++++ 1) VARIANCE ESTIMATOR (MEDIAN): Woodruff +++++;
tayg5=((wohigh-wolow)/(2*1.96))**2;

%do iv=1 %to %eval(&n_a-1) ;
* +++++ 2) VARIANCE ESTIMATOR: External linear case scaling coefficient +++++;
Q1gab=(1/&n_a - 1/&CapN)/(1/&iv - 1/&n_a); Q1fab=(1/&n_a - 1/&CapN)/(1/&iv -
1/(&n_a*&iv));
extg1_&iv = Q1gab * g1v&iv; extg2_&iv= Q1gab * g2v&iv ; extg3_&iv = Q1gab * g3v&iv;
extg4_&iv= Q1gab * g4v&iv ; extg5_&iv = Q1gab * g5v&iv; extg6_&iv= Q1gab * g6v&iv ;
extf1_&iv = Q1fab * f1v&iv; extf2_&iv= Q1fab * f2v&iv ; extf3_&iv = Q1fab * f3v&iv;
extf4_&iv= Q1fab * f4v&iv ; extf5_&iv = Q1fab * f5v&iv; extf6_&iv= Q1fab * f6v&iv ;

%do iv2=1 %to %eval(&iv-1);

Q1gac=(1/&n_a - 1/&CapN)*(1/&iv2 - 1/&iv)/((1/&iv - 1/&n_a)**2);
%let vi=_ ; %let e=e; %let eq=eq; %let es=es;

* ++++ 3) VARIANCE ESTIMATOR: Bias correction with two resample sizes ++++;
if &iv>1 then bc2si1_&iv&vi&iv2= Q1gac * g1v&iv / (g1v&iv2/g1v&iv - 1) ;
if &iv>1 then bc2si2_&iv&vi&iv2= Q1gac * g2v&iv / (g2v&iv2/g2v&iv - 1) ;
bc2si3_&iv&vi&iv2= Q1gac * g3v&iv / (g3v&iv2/g3v&iv - 1) ;
bc2si4_&iv&vi&iv2= Q1gac * g4v&iv / (g4v&iv2/g4v&iv - 1) ;
bc2si5_&iv&vi&iv2= Q1gac * g5v&iv / (g5v&iv2/g5v&iv - 1) ;
bc2si6_&iv&vi&iv2= Q1gac * g6v&iv / (g6v&iv2/g6v&iv - 1) ;

* ++++ 4) VARIANCE ESTIMATOR: Bias correction in two phases;
if &iv>1 then twp1_&iv&vi&iv2 = Q1gac * ( g1v&iv )**2 / gzu1_&iv2&e&iv ;
if &iv>1 then twp2_&iv&vi&iv2 = Q1gac * ( g2v&iv )**2 / gzu2_&iv2&e&iv ;
twp3_&iv&vi&iv2 = Q1gac * ( g3v&iv )**2 / gzu3_&iv2&e&iv ;
twp4_&iv&vi&iv2 = Q1gac * ( g4v&iv )**2 / gzu4_&iv2&e&iv ;
twp5_&iv&vi&iv2 = Q1gac * ( g5v&iv )**2 / gzu5_&iv2&e&iv ;
twp6_&iv&vi&iv2 = Q1gac * ( g6v&iv )**2 / gzu6_&iv2&e&iv ;

* ++++ 5) VARIANCE ESTIMATOR: MSE reducing variance estimator +++;
if &iv>1 then mim1_&iv&vi&iv2 = Q1gab * g1v&iv * gzu1_&iv2&eq&iv / gzu1_&iv2&es&iv ;
if &iv>1 then mim2_&iv&vi&iv2 = Q1gab * g2v&iv * gzu2_&iv2&eq&iv / gzu2_&iv2&es&iv ;
mim3_&iv&vi&iv2 = Q1gab * g3v&iv * gzu3_&iv2&eq&iv / gzu3_&iv2&es&iv ;
mim4_&iv&vi&iv2 = Q1gab * g4v&iv * gzu4_&iv2&eq&iv / gzu4_&iv2&es&iv ;
mim5_&iv&vi&iv2 = Q1gab * g5v&iv * gzu5_&iv2&eq&iv / gzu5_&iv2&es&iv ;
mim6_&iv&vi&iv2 = Q1gab * g6v&iv * gzu6_&iv2&eq&iv / gzu6_&iv2&es&iv ;

%end;

* ++++ 6) VARIANCE ESTIMATOR: Strict randomisation for resamples ++++++;
p_low=(1/&linb - 1/&linbhigh)/(1/&linblow - 1/&linbhigh); p_upp=1-p_low;
```

```
%let z=&linblow; %let r=&linbhigh;

rd1=p_low * g1v&z + p_upp * g1v&r ; rd2=p_low * g2v&z + p_upp * g2v&r ;
rd3=p_low * g3v&z + p_upp * g3v&r ; rd4=p_low * g4v&z + p_upp * g4v&r ;
rd5=p_low * g5v&z + p_upp * g5v&r ; rd6=p_low * g6v&z + p_upp * g6v&r ;

* ++++ 7) VARIANCE ESTIMATOR: Bootstrap without replacement with randomisation ++++;
bp_low=((1-&n_a/&CapN)/(&n_a*(&n_a-1))
        - &BWO_mup*(1-&BWO_n/(&n_a*&BWO_mup)) / (&BWO_n*(&n_a*&BWO_mup-1)))
       / (&BWO_mlow*(1-&BWO_n/(&n_a*&BWO_mlow)) / (&BWO_n*(&n_a*&BWO_mlow-1))
           - &BWO_mup*(1-&BWO_n/(&n_a*&BWO_mup)) / (&BWO_n*(&n_a*&BWO_mup-1)));
bp_up=1-bp_low;

%let c=&BWO_n;

bwo1=bp_low * fbl1v&c + bp_up * fbu1v&c ; bwo2=bp_low * fbl2v&c + bp_up * fbu2v&c ;
bwo3=bp_low * fbl3v&c + bp_up * fbu3v&c ; bwo4=bp_low * fbl4v&c + bp_up * fbu4v&c ;
bwo5=bp_low * fbl5v&c + bp_up * fbu5v&c ; bwo6=bp_low * fbl6v&c + bp_up * fbu6v&c ;

* ++++ 8) VARIANCE ESTIMATOR: Post-design vector approach ++++++;
dve1=gde1v&r;  dve2=gde2v&r;  dve3=gde3v&r;  dve4=gde4v&r;  dve5=gde5v&r; dve6=.;

%end;

* +++++ 9) VARIANCE ESTIMATOR: Taylor s linearisation ++++++ ;
tayg3 = (1/(aty/&CapN)**2)*(1/&n_a - 1/&CapN)*(avx+avy*arxy**2-2*arxy*acvxy);
takorj = (&n_a/(&n_a-1))**2; tayg1 = takorj*(1/&n_a - 1/&CapN)*vate/(&n_a-1); tayg4=.;
tayg6=.;

* ++++ 10) VARIANCE ESTIMATOR: Dependent random groups method ++++ ;
* If division is even, then exact calculations, otherwise number of
  groups is not constant;

%do zif=2 %to &max_rg;
     korjke=(&CapN-&n_a)*&zif/(&CapN*&n_a);
     rg1_gr&zif = korjke*(g1es&zif - grg1e&zif) ; rg2_gr&zif = korjke*(g2es&zif -
grg2e&zif) ;
     rg3_gr&zif = korjke*(g3es&zif - grg3e&zif) ; rg4_gr&zif = korjke*(g4es&zif -
grg4e&zif) ;
     rg5_gr&zif = korjke*(g5es&zif - grg5e&zif) ; rg6_gr&zif = korjke*(g6es&zif -
grg6e&zif) ;
%end;

* ++++ 11) VARIANCE ESTIMATOR: JACKKNIFE 2 out of 3 ++++ ;
* If division is even, then exact calculations, otherwise number of
  groups is not constant;

     %let orson=%eval(&n_a-2);
     korjkej2=(&CapN-&n_a)*&orson**2/(&CapN*&n_a*(&n_a-&orson));
     j1_gr_2 = korjkej2*(g1es&orson - gj2_1e&orson) ;
     j2_gr_2 = korjkej2*(g2es&orson - gj2_2e&orson) ;
     j3_gr_2 = korjkej2*(g3es&orson - gj2_3e&orson) ;
     j4_gr_2 = korjkej2*(g4es&orson - gj2_4e&orson) ;
     j5_gr_2 = korjkej2*(g5es&orson - gj2_5e&orson) ;
     j6_gr_2 = korjkej2*(g6es&orson - gj2_6e&orson) ;

* ++++ 12) VARIANCE ESTIMATOR: Decomposition with conditionality assumption ++++;

%do iv=1 %to %eval(&n_a-1) ;
       deco1&vi&iv = sif1e&iv; deco2&vi&iv = sif2e&iv; deco3&vi&iv = sif3e&iv;
       deco4&vi&iv = sif4e&iv; deco5&vi&iv = sif5e&iv; deco6&vi&iv = sif6e&iv;
* ++++ 13) VARIANCE ESTIMATOR: Decomposition with weight adjustments ++++;
   if n_adebn=&iv then do;
     decb1 = (1-prton(&n_a+1))*(avx**2)-sifb1e&iv; decb2 = (1-
prton(&n_a+1))*(acrxy**2)-sifb2e&iv;
```

```
      decb3 = (1-prton(&n_a+1))*(arxy**2)-sifb3e&iv; decb4 = (1-
prton(&n_a+1))*(atx**2)-sifb4e&iv;
      decb5 = (1-prton(&n_a+1))*(amex**2)-sifb5e&iv; decb6 = (1-
prton(&n_a+1))*(amax**2)-sifb6e&iv;
      decc1 = (1-prton(&n_a+1))*(g1eq&iv)-sifb1e&iv; decc2 = (1-
prton(&n_a+1))*(g2eq&iv)-sifb2e&iv;
      decc3 = (1-prton(&n_a+1))*(g3eq&iv)-sifb3e&iv; decc4 = (1-
prton(&n_a+1))*(g4eq&iv)-sifb4e&iv;
      decc5 = (1-prton(&n_a+1))*(g5eq&iv)-sifb5e&iv; decc6 = (1-
prton(&n_a+1))*(g6eq&iv)-sifb6e&iv;
   end;

%end;

%mend;

* ============ PropertiesAndTables.sas ====================;
* Includes the ratio variance estimator;

%macro RESULTS;

* Index getting through six different estimators;

%do zig=1 %to 6;
%let v=v; %let vi=_; %let vigr2=_gr2;%let vigr3=_gr3;%let vigrv2=_gr_2;
%let ev=ev; %let vv=vv; %let d2=_2; %let d3=_3; %let mdv=mdv; %let q1_=q1_ ; %let
q3_=q3_ ;
%let vahe1=0; %let vahe2=0;
%if &n_a>5 %then %do;  %let sma1=0; %let sma2=0; %end;
%else %do; %let sma1=1; %let sma2=3; %end;  %let numba=%eval(&num-&vahe2-&sma2);

* ---------- EXPECTATIONS AND VARIANCES FOR VARIANCE ESTIMATORS ETC. ----------;

proc univariate data=&DATANAME vardef=n noprint;
  class n_aeab n_aeac; * these variables are brought "through" the procedure;
  var
      avx acrxy arxy atx avy amex amax       /* estimators (7)  */
      avxq acrxyq arxyq atxq avyq     /* squares of estimators */
      /* resample sizes: two principles floor(nlin) and na-1 */
        g&zig&v&o g&zig&v&n_1 f&zig&v&n_1       /* normal conditional variances */
        grs&zig&v&o grs&zig&v&n_1  frf&zig&v&n_1        /* internal scaling */
      extg&zig&vi&o extg&zig&vi&n_1  extf&zig&vi&n_1  /* external scaling */
      tayg&zig       /* Taylor */
      bc2si&zig&vi&n_1&nb_1           /* two-size bias correction, na-1 and na-2 */
        bc2si&zig&vi&o&olow           /* two-size bias correction, ceil(nlin) and
floor(nlin) */
      rd&zig              /* strict randomisation */
      bwo&zig             /* BWO with randomisation */
      rg&zig&vigr2       /* two random groups */
        dve&zig           /* post-design vector */
      deco&zig&d2         /* decomposition with conditionality, nb =2 */
        decb&zig          /* decomposition with weight adjustments, theta2 */
        decc&zig          /* decomposition with weight adjustments, Ethetab2 */
   %if &n_a>5 %then %do;
      deco&zig&d3         /* decomposition with conditionality, nb =3 */
      rg&zig&vigr3        /* three random groups */
        j&zig&vigrv2      /* jackknife 2 out of 3 */
   %end;
        twp&zig&vi&n_1&nb_1           /* Two-phase bias correction, na-1 ja na-2 */
        twp&zig&vi&o&olow             /* Two-phase bias correction, ceil(nlin) and
floor(nlin) */
        mim&zig&vi&n_1&nb_1           /* MinMSE, na-1 ja na-2 */
        mim&zig&vi&o&olow             /* MinMSE, ceil(nlin) and floor(nlin) */
;
  output out=glis&zig
      /* calculating means */
      mean= ag1e ag2e ag3e ag4e ag7e ag5e ag6e ag1qe ag2qe ag3qe ag4qe ag7qe
```

```
            g&zig&ev&o g&zig&ev&n_1 f&zig&ev&n_1  grs&zig&ev&o grs&zig&ev&n_1
frf&zig&ev&n_1
           mextg&zig&vi&o mextg&zig&vi&n_1 mextf&zig&vi&n_1 mtayg&zig
mbc2si&zig&vi&n_1&nb_1
            mbc2si&zig&vi&o&olow mrd&zig mbwo&zig mrg&zig&vigr2 mdve&zig mdeco&zig&d2
           mdecb&zig mdecc&zig
           %if &n_a>5 %then %do;
             mdeco&zig&d3 mrg&zig&vigr3 mj&zig&vigrv2
           %end;
            mtwp&zig&vi&n_1&nb_1 mtwp&zig&vi&o&olow mmim&zig&vi&n_1&nb_1
mmim&zig&vi&o&olow
      /* calculating variances */
      var= ag1v ag2v ag3v ag4v ag7v ag5v ag6v ag1qv ag2qv ag3qv ag4qv ag7qv
           g&zig&vv&o g&zig&vv&n_1 f&zig&vv&n_1 grs&zig&vv&o grs&zig&vv&n_1
frf&zig&vv&n_1
          vextg&zig&vi&o vextg&zig&vi&n_1 vextf&zig&vi&n_1 vtayg&zig
vbc2si&zig&vi&n_1&nb_1
            vbc2si&zig&vi&o&olow vrd&zig vbwo&zig vrg&zig&vigr2 vdve&zig vdeco&zig&d2
          vdecb&zig vdecc&zig
          %if &n_a>5 %then %do;
            vdeco&zig&d3 vrg&zig&vigr3 vj&zig&vigrv2
          %end;
            vtwp&zig&vi&n_1&nb_1 vtwp&zig&vi&o&olow vmim&zig&vi&n_1&nb_1
vmim&zig&vi&o&olow
;
run;

* ---------- Making Ratio variance estimator ------------;
data alkub&zig(keep=n_aeab raeb&zig&vi&o raeb&zig&vi&n_1 rafb&zig&vi&n_1);
 merge &DATANAME(keep=n_aeab g&zig&v&o g&zig&v&n_1 f&zig&v&n_1
                      g7v&o g4v&o g7e&o g4e&o  g7v&n_1 g4v&n_1 g7e&n_1 g4e&n_1
                      f7v&n_1 f4v&n_1 f7e&n_1 f4e&n_1)
       glis&zig(keep=n_aeab ag7v ag4v ag7e ag4e);
   by n_aeab ;
   raeb&zig&vi&o=ag4v*g&zig&v&o/g4v&o; raeb&zig&vi&n_1=ag4v*g&zig&v&n_1/g4v&n_1;
   rafb&zig&vi&n_1=ag4v*f&zig&v&n_1/f4v&n_1;
run;

proc univariate data=alkub&zig vardef=n noprint;
  class n_aeab;
  var raeb&zig&vi&o raeb&zig&vi&n_1 rafb&zig&vi&n_1 ;
  output out = fiba&zig(drop=_type_ _freq_)
         mean = mraeb&zig&vi&o mraeb&zig&vi&n_1 mrafb&zig&vi&n_1
          var =  vraeb&zig&vi&o vraeb&zig&vi&n_1 vrafb&zig&vi&n_1
;
run;

data glisb&zig;
 merge glis&zig fiba&zig;
   by n_aeab;
run;

/* Getting the results of the mean to the final form */

data bias&zig(keep=bias1-bias&numba kana1-kana&numba)
     relbias&zig(keep=relbias1-relbias&numba );
 set glisb&zig;
   array kana(&numba);
   array old(*)
         grs&zig&ev&o grs&zig&ev&n_1 frf&zig&ev&n_1 mraeb&zig&vi&o mraeb&zig&vi&n_1
         mrafb&zig&vi&n_1 mbc2si&zig&vi&o&olow mbc2si&zig&vi&n_1&nb_1 mtayg&zig
         mrd&zig mbwo&zig mextg&zig&vi&o mextg&zig&vi&n_1 mextf&zig&vi&n_1
mrg&zig&vigr2
            mdve&zig mdeco&zig&d2 mdecb&zig mdecc&zig
   %if &n_a>5 %then %do;
         mdeco&zig&d3 mrg&zig&vigr3 mj&zig&vigrv2
   %end;
```

```
        mtwp&zig&vi&o&olow mtwp&zig&vi&n_1&nb_1 mmim&zig&vi&o&olow
mmim&zig&vi&n_1&nb_1 ;
      array bias(&numba); array relbias(&numba);
      do i=1 to &numba;
         kana(i)=old(i);  bias(i)=old(i)-ag&zig&v;  relbias(i)=100*bias(i)/ag&zig&v;
      end;
      output bias&zig; output relbias&zig;
run;

proc transpose data=bias&zig out=siis&zig;
  var bias1-bias&numba ; run;
proc transpose data=bias&zig out=bliv&zig;
  var kana1-kana&numba ; run;
proc transpose data=relbias&zig out=relsiis&zig;
  var relbias1-relbias&numba ; run;

/* Getting the results of the variance to the final form */
data variance&zig(keep=variance1-variance&numba scavar1-scavar&numba);
 set glisb&zig;
   array old(*)
         grs&zig&vv&o grs&zig&vv&n_1 frf&zig&vv&n_1 vraeb&zig&vi&o vraeb&zig&vi&n_1
         vrafb&zig&vi&n_1 vbc2si&zig&vi&o&olow vbc2si&zig&vi&n_1&nb_1 vtayg&zig
         vrd&zig vbwo&zig vextg&zig&vi&o vextg&zig&vi&n_1 vextf&zig&vi&n_1
vrg&zig&vigr2
              vdve&zig vdeco&zig&d2 vdecb&zig vdecc&zig
   %if &n_a>5 %then %do;
         vdeco&zig&d3 vrg&zig&vigr3 vj&zig&vigrv2
   %end;
         vtwp&zig&vi&o&olow vtwp&zig&vi&n_1&nb_1 vmim&zig&vi&o&olow
vmim&zig&vi&n_1&nb_1 ;
      array variance(&numba); array scavar(&numba);
      do i=1 to &numba ;
       variance(i)=old(i);  scavar(i)=variance(i)/(ag&zig&v**2);
       vaari=variance(i); scarpia=scavar(i); agitaattori=ag&zig&v**2;
      end;
run;

proc transpose data=variance&zig out=kriik&zig;
  var scavar1-scavar&numba ; run;

* ---------------- Preparing the printouts ------------;

data siis&zig;
 set siis&zig;
   jarnum=_n_;
if _n_=1 then do;  descr='weight rescaling              '; descr2='SI,Lnlin     '; un=7;
end;
if _n_=2 then do;  descr='weight rescaling              '; descr2='SI,n-1       '; un=3;
end;
if _n_=3 then do;  descr='weight rescaling              '; descr2='SIR,n-1      '; un=5;
end;
if _n_=4 then do;  descr='ratio estimator of variance  '; descr2='SI,Lnlin,Vy  ';
un=13-&sma2; end;
if _n_=5 then do;  descr='ratio estimator of variance  '; descr2='SI,n-1,Vy    ';
un=14-&sma2; end;
if _n_=6 then do;  descr='ratio estimator of variance  '; descr2='SIR,n-1,Vy   ';
un=15-&sma2; end;
if _n_=7 then do;descr='two size resampling correct. '; descr2='Lnlin,Lnlin-1';un=18-
&vahe1-&sma2; end;
if _n_=8 then do; descr='two size resampling correct. '; descr2='n-1, n-2     '; un=19-
&vahe1-&sma2; end;
if _n_=9 then do;
          %if &zig<5 %then %do;
                descr='taylors linearisation        ';  descr2=' ' ;
          %end;
          %else %if &WNOTE=0 %then %do;
                descr='Woodruffs method             ';  descr2=' ' ;
```

```
          %end; %else %do;
                descr='Woodruffs method              '; descr2='ADJUSTED!';
          %end;
               un=1; end;
if _n_=10 then do; descr='randomisation for a resample '; descr2=' ' ; un=11-&sma2;
end;
if _n_=11 then do; descr='BWO with randomisation       '; descr2=' ' ; un=12-&sma2;
end;
if _n_=12 then do; descr='external correction          '; descr2='SI,Lnlin'; un=6; end;
if _n_=13 then do; descr='external correction          '; descr2='SI,n-1 ' ; un=2; end;
if _n_=14 then do; descr='external correction          '; descr2='SIR,n-1 ' ; un=4;
end;
if _n_=15 then do; descr='two equal-sized random groups'; descr2=' ' ; un=9-&sma1; end;
if _n_=16 then do; descr='design vector (rand. for res)'; descr2=' ' ; un=22-&vahe2-
&sma2; end;
if _n_=17 then do; descr='decomposition 2              '; descr2=' ' ; un=23-&vahe2-
&sma2; end;
if _n_=18 then do; descr='decomposition, weight,thetaa2'; descr2=' ' ; un=24-&vahe2-
&sma2; end;
if _n_=19 then do; descr='decomposition,wgt , ethetab2 '; descr2=' ' ; un=25-&vahe2-
&sma2; end;
%if &n_a>5 %then %do;
 if _n_=20 then do; descr='decomposition 3               '; descr2=' ' ; un=26-&vahe2;
end;
 if _n_=21 then do; descr='jackknife n-2';                  descr2=' ' ; un=8; end;
 if _n_=22 then do; descr='three equal-sized random gro '; descr2=' ' ; un=10; end;
%end;
if _n_=23-&sma2 then do; descr='two-phase resampling correct.'; descr2='Lnlin,Lnlin-1';
un=16-&sma2; end;
if _n_=24-&sma2 then do; descr='two-phase resampling correct.'; descr2='n-1, n-2      ';
un=17-&sma2; end;
if _n_=25-&sma2 then do; descr='minimal MSE                  '; descr2='Lnlin,Lnlin-1';
un=20-&sma2; end;
if _n_=26-&sma2 then do; descr='minimal MSE                  '; descr2='(n-1, n-2)    ';
un=21-&sma2; end;
rename col1=Bias;
run;

data relsiis&zig;
 set relsiis&zig;
   jarnum=_n_; rename col1=RelativeBias;
run;

data kriik&zig;
 set kriik&zig;
   jarnum=_n_; rename col1=RelativeVariance;
run;

data together&zig;
 merge siis&zig relsiis&zig kriik&zig ;
   by jarnum;
   Relative_sqrt_MSE =100*sqrt(RelativeVariance+(RelativeBias/100)**2);
   Relative_SE=100*sqrt(RelativeVariance);    label RelativeBias='Relative Bias (%)';
run;

proc sort data=together&zig;
 by un;
run;

options nocenter ls=150;
proc print data=together&zig noobs label;
  var descr descr2 RelativeBias Relative_SE Relative_sqrt_MSE  ;
 run;
%end;

%mend RESULTS ;
```

```
/* ==================== Macros.sas ========================

Here it is decided to explain only the purpose of each macro, a detailed description
for each
action would be too big an effort.

1) ARRAY MACROS (making array variables for the data)
- ARR      (making arrays of specified dimension for the given variable "beginnings"
[max 15])
- ARRPROP (making arrays for the property variables for estimators)
- ARRTWO   (making arrays for the property variables for estimators for two-phase
           resampling estimators ;

2) FUNCTION MACROS
- COR      (calculating the correlation with the given statistics)
- COVA     (calculating the covariance with the given statistics)
- TOTA     (calculating the total based on the given sum and the number of observations)
- VARI     (calculating the variance based on the given statistics)
- DISPAC (package calculating dispersion statistics [cor, cova, vari])
- TOTPAC (package calculating different totals)

3) LOOP MACROS
- DOSFI   (calculating beginnings of do loops for without replacement samples)
- DOSFAD (calculating beginnings of do loops for with replacement resamples in
resample pairs)
- DOSWR   (calculating beginnings of do loops for with replacement resamples)
- DOSIII (calculating beginnings of do loops for second-phase without replacement
resamples)

4) END MACROS
- ENDS     (ends first, second, and third phase loops)
- ENDSFAD (ends for sample pair selection loops)

5) ESTIMATE CUMULATING MACROS
- CUMU1     (Constructing sums of estimates (WR resamples))
- CUMU1RS   (Constructing sums of estimates for internal rescaling purposes (wr))
- CUMU2     (Constructing sums of estimates (WOR resamples))
- CUMUD2    (Constructing sums of estimates (WOR resamples) for sample pairs)
- CUMUDG2   (Constructing sums of estimates (WOR resamples) for sample pairs, weight
adjustments)
- CUMUEXT2 (Constructing sums of estimates (WOR resamples) for concluding results from
third-phase sampling)
- CUMUBOL   (Constructing sums of estimates (WOR resamples) for BWO lower limit)
- CUMUBOU   (Constructing sums of estimates (WOR resamples) for BWO upper limit)
- CUMU2RS   (Constructing sums of estimates for internal rescaling purposes (wor))
- CUMU2DE   (Constructing sums of estimates (WOR resamples) for post-design vector
approach)
- CUMUC2    (Property sum calculations from third-phase sampling)

6) NUMBER OF VALID ESTIMATE VALUES MACROS

- COUN1     (Constructing number of valid estimate values (WR resamples))
- COUN1RS   (Constructing number of valid estimate values (WR resamples), internal
rescaling)
- COUN2     (Constructing number of valid estimate values (WOR resamples))
- COUND2    (Constructing number of valid estimate values (WOR resamples), sample
pairs)
- COUNEXT2 (Constructing number of ... (WOR resamples) for concluding results from
third-phase sampling)
- COUNBOL   (Constructing number of valid estimate values (WOR resamples), for BWO
lower limit)
- COUNBOU   (Constructing number of valid estimate values (WOR resamples), for BWO
upper limit)
- COUN2RS   (Constructing number of valid estimate values (WOR resamples), internal
rescaling)
- COUN2DE   (Constructing number of valid estimate values (WOR resamples), for post-
design vector approach)
```

```
- COUNC2   (Property calculations for third-phase sampling, number of valid values)

7) VALUE ASSIGNING MACROS
- VAL
- VALPAC
- VALPAGE
- VALPASPE
- VALPACC

8) ORDERING MACROS
- ORDPA    (Giving order numbers of x for observations, first-phase selections)
- ORDPAC   (Giving order numbers of x for observations, first-phase selections)
- ORDPACB (Giving order numbers of x for observations, first-phase selections)

9) PROPERTY CALCULATING MACROS
- PACK1    (Calculating properties of conditional estimators, based on CUMU and COUN
macros)
- PACKSIF (Calculating properties of conditional estimators, based on CUMU and COUN
macros (decomposition))
- PACK3    (Calculating properties of conditional estimators, based on CUMU and COUN
macros (concluding third phase))
- PACKSPE (Calculating properties of conditional estimators, based on CUMU and COUN
macros (third-phase based estimators))

10) MACRO IDENTIFYING WOR SAMPLE
- SEPAR (Identifying without replacement sample in second-phase sampling (wr);

11) SUM MACROS
- VATEXSUM (Term for taylor approximation of variance S2x);
- SUMPAC    (Summing up different terms)
- SUMPAWOR (Summing up different terms (internal wor rescaling))
- SUMPAWR   (Summing up different terms (internal wr rescaling))
- SUMPDE    (Summing up post-design vector terms)

12) QUANTILE MACROS
- QUAPA     (Quantile calculations (first phase))
- QUAPAC    (Quantile calculations (second phase))
- QUAPACC   (Quantile calculations (third phase))

13) POPULATION SORT MACRO
- BEGISORT (Population of two variables sorted by x)

*/

* ------------- 1) ARRAY MACROS -----------------;

* Making arrays of specified dimension for the given "beginnings" [max 15];
* in MakingArrays.sas;
* Input info: 1) dimension of population, 2) -> 16)  beginnings of variables ;
%macro ARR(big,wa,wb,wc,wd,we,wf,wg,wh,wi,wj,wk,wl,wm,wn,wo);
   %let one=1;
   %do zm=1 %to 15;
      %if &zm=1 %then %do;  %let war = &wa ; %end;  %if &zm=2 %then %do;  %let war =
&wb ; %end;
        %if &zm=3 %then %do;  %let war = &wc ; %end;  %if &zm=4 %then %do;  %let war =
&wd ; %end;
        %if &zm=5 %then %do;  %let war = &we ; %end;  %if &zm=6 %then %do;  %let war =
&wf ; %end;
        %if &zm=7 %then %do;  %let war = &wg ; %end;  %if &zm=8 %then %do;  %let war =
&wh ; %end;
        %if &zm=9 %then %do;  %let war = &wi ; %end;  %if &zm=10 %then %do; %let war =
&wj ; %end;
        %if &zm=11 %then %do; %let war = &wk ; %end;  %if &zm=12 %then %do; %let war =
&wl ; %end;
        %if &zm=13 %then %do; %let war = &wm ; %end;  %if &zm=14 %then %do; %let war =
&wn ; %end;
        %if &zm=15 %then %do; %let war = &wo ; %end;
```

```
      %if &war^=. %then %do;
           array &war(&big) &war&one-&war&big ;
      %end;
   %end;
%mend ARR;

* Making arrays for the property variables for estimators
  (expectation of the estimator, expectation of the squared estimator,
   square of the expectation, variance of the estimator, count variable);
* in MakingArrays.sas;
* Input info: 1) dimension of population, 2) number of parameters,
  3) -> 17) beginnings of variables   ;
%macro ARRPROP(big,nu,c1,c2,c3,c4,c5,c6,c7,c8,c9,c10,c11,c12,c13,c14,c15);

   %let one=1; %let e=e; %let eq=eq; %let es=es ; %let v=v; %let la=la;
   %do zq=1 %to 15;
     %let warre=&&c&zq;
     %if &warre ^= . %then %do;
      %do zm=1 %to &nu;
           array &warre&zm&e(&big) &warre&zm&e&one-&warre&zm&e&big ;
           array &warre&zm&es(&big) &warre&zm&es&one-&warre&zm&es&big ;
           array &warre&zm&eq(&big) &warre&zm&eq&one-&warre&zm&eq&big ;
           array &warre&zm&v(&big) &warre&zm&v&one-&warre&zm&v&big ;
              array &warre&zm&la(&big) &warre&zm&la&one-&warre&zm&la&big ;
      %end;
       %end;
   %end;

%mend;

* Making arrays for the property variables for estimators for two-phase  resampling
estimators ;
* in MakingArrays.sas ;
* Input info: 1) dimension of population, 2) number of parameters, 3) beginning of
variable ;

%macro ARRTWO(big,nu,fi);
    %let yx=1; %let e=e; %let eq=eq; %let es=es ; %let v=v; %let la=la; %let w=_;
    %do zm=1 %to %eval(&nu); %do zik=1 %to %eval(&big-2);
           array &fi&zm&w&zik&e(&big) &fi&zm&w&zik&e&yx-&fi&zm&w&zik&e&big ;
           array &fi&zm&w&zik&es(&big) &fi&zm&w&zik&es&yx-&fi&zm&w&zik&es&big ;
           array &fi&zm&w&zik&eq(&big) &fi&zm&w&zik&eq&yx-&fi&zm&w&zik&eq&big ;
           array &fi&zm&w&zik&v(&big) &fi&zm&w&zik&v&yx-&fi&zm&w&zik&v&big ;
              array &fi&zm&w&zik&la(&big) &fi&zm&w&zik&la&yx-&fi&zm&w&zik&la&big;
    %end; %end;
%mend ARRTWO;

* ------------------- 2) FUNCTION MACROS ------------------------;

* Calculating the correlation with the given statistics; * in MACRO DISPAC ;
* Input info: 1) name of variable, 2) covariance variable,
              3) variance of y variable, 4) variance of x variable;
%macro COR(nam,covxy,vary,varx);
       &nam = &covxy/(sqrt(&vary)*sqrt(&varx));
%mend;

* Calculating the covariance with the given statistics; * in MACRO DISPAC ;
* Input info: 1) name of variable, 2) size of population, 3) sample size,
              4) total of xy product, 5) y total, 6) x total ;
%macro COVA(nam,big,sma,xytot,ytot,xtot);
       &nam = &sma * (&xytot/&big - &ytot*&xtot/(&big**2))/(&sma-1);
%mend;

* Calculating the total with the given statistics; * in MACRO TOTPAC ;
* Input info: 1) name of variable, 2) size of population, 3) sample size, 4) sum
variable ;
%macro TOTA(nam,big,sma,vari);
```

```
      &nam = &big * &vari / &sma ;
%mend;

* Calculating the variance of the variable with the given statistics;
* in MACRO DISPAC;
* Input info: 1) name of variable, 2) size of population,
              3) name of total of squared variable, 4) name of square of variable
total ;
%macro VARI(nam,big,sma,y2to,yto2);
       &nam = &sma * (&y2to / &big - &yto2 / (&big**2))/(&sma-1) ;
%mend;

* Package calculating dispersion statistics [cor, cova, vari];
* in FirstPhaseSelections.sas, SecondPhaseSelections.sas, ThirdPhaseSelections.sas,
    SecondPhasePairSelections.sas ;
* Input info: 1) size of population, 2) beginning of variable,
              3) variable 1, 4) variable 2, 5) end of count variable;
%macro DISPAC(big,be,x,y,co);
      %let q=q; %let ka=2; %let v=v; %let cv=cv; %let cr=cr; %let t=t;
      if &be&co>1 then do;
        %VARI(&be&v&x,&big,&be&co,&be&t&x&q,((&be&t&x)**2));
        %VARI(&be&v&y,&big,&be&co,&be&t&y&q,((&be&t&y)**2));
        &be&v&x&q=&be&v&x**2; &be&v&y&q=&be&v&y**2;
        %COVA(&be&cv&x&y,&big,&be&co,&be&t&x&y,&be&t&y,&be&t&x);
        if &be&v&y >0 and &be&v&x >0 then do;
%COR(&be&cr&x&y,&be&cv&x&y,&be&v&y,&be&v&x) ; end;
        else &be&cr&x&y =.; &be&cr&x&y&q = &be&cr&x&y**2; end;
%mend;

* Package calculating different totals (for x, y, x**2, y**2, xy);
* in FirstPhaseSelections.sas, SecondPhaseSelections.sas, ThirdPhaseSelections.sas,
    SecondPhasePairSelections.sas ;
* Input info: 1) size of population, 2) beginning of variable,
              3) variable 1, 4) variable 2, 5) end of count variable;
%macro TOTPAC(big,be,x,y,co);
      %let q=q; %let t=t; %let s=s;
          %TOTA(&be&t&x,&big,&be&co,&be&s&x); %TOTA(&be&t&y,&big,&be&co,&be&s&y);
          %TOTA(&be&t&x&q,&big,&be&co,&be&s&x&q);
%TOTA(&be&t&y&q,&big,&be&co,&be&s&y&q);
          %TOTA(&be&t&x&y,&big,&be&co,&be&s&x&y);
%mend;

* ---------- LOOP MACROS ----------;

* Calculating beginnings of do loops for without replacement samples;
* in FirstPhaseSelections.sas;
* Input info: 1) size of population, 2) sample size;
%macro DO_SFI(iso,siva);
   dif=&siva;
   do zi1 = 1 to &iso - &siva + 1;
   i(zi1)=1;  if zi1 > 1 then i(prev1)=0;  prev1=zi1;
   %do qw=2 %to &siva;
       katto= &iso - &siva + &qw ;
     %let ede=%eval(&qw-1);
     do zi&qw = zi&ede + 1 to katto ;
       i(zi&qw)=1; if (zi&qw > &qw) and (prev&qw ne zi&qw) then i(prev&qw)=0;
prev&qw=zi&qw;
   %end;
%mend DO_SFI;


* Calculating beginnings of do loops for with replacement resamples in sample pairs ;
* in SecondPhasePairSelections.sas ;
* Input info: 1) size of population;
%macro DO_SFAD(iso);
  dben=0; ranni=1;
  do dj1 = i1 to 0 by -1;
```

```
   dben=dben+dj1; if dben=bn then ranni=0;
   %do qw=2 %to &iso;
       do dj&qw = ranni*i&qw to 0 by -1;
      dben=dben+dj&qw; if dben=bn then ranni=0;
   %end;
     if dben=bn then do;
%mend DO_SFAD;

* Calculating beginnings of do loops for with replacement resamples ;
* in SecondPhaseSelections.sas ;
* Input info: 1) size of population, 2) sample size, 3) sampling level indicator
letter;
%macro DO_SWR(iso,pien,wb);
   nq1=&pien ;
   %do qw=1 %to &iso;
       %let qwb=%eval(&qw+1);
         hig = nq&qw * i&qw ;
         do j&qw = hig to 0 by -1;
          %if &qw<&iso %then %do;  nq&qwb = nq&qw - j&qw ;  %end;
            if &qw = &iso then do;
           bot=1; jsu=0; bwo_loid=1; bwo_upid=1; bwosul=0; bwotulol=1; bwosuu=0;
bwotulou=1;
           do wp=1 to &iso;
                jsu=sum(jsu,j(wp)); bot=bot/gamma(j(wp) + 1);
              if bwo_loid=1 and j(wp)<=&BWO_mlow and bwosul<=&BWO_n then do;
                bwosul=bwosul+j(wp);

bwotulol=bwotulol*gamma(&BWO_mlow+1)/(gamma(j(wp)+1)*gamma((&BWO_mlow-j(wp))+1));
                    if wp=&iso then do; if bwosul=&BWO_n then bwo_loid=1; else
bwo_loid=0; end;
                  end; else bwo_loid=0;
             if bwo_upid=1 and j(wp)<=&BWO_mup and bwosuu<=&BWO_n then do;
                bwosuu=bwosuu+j(wp);
               bwotulou=bwotulou*gamma(&BWO_mup+1)/(gamma(j(wp)+1)*gamma((&BWO_mup-
j(wp))+1));
                if wp=&iso then do; if bwosuu=&BWO_n then bwo_upid=1; else
bwo_upid=0; end;
                end; else bwo_upid=0;
              end;

              if jsu>0 then bott&wb=gamma(jsu + 1) * bot;
              if bwosul>0 and bwo_loid=1 then botlo=gamma(bwosul + 1)  ; else
botlo=.;
              if bwosul>0 and bwo_loid=1 then botlox=botlo*bwotulol; else botlox=.;
              if bwosuu>0 and bwo_upid=1 then botup=gamma(bwosuu + 1)  ; else
botup=.;
              if bwosuu>0 and bwo_upid=1 then botupx=botup*bwotulou; else botupx=.;
            end;
%end;
%mend DO_SWR;

* Calculating beginnings of do loops for second-phase without replacement resamples ;
* in ThirdPhaseSelections.sas ;
* Input info: 1) size of population, 2) sample size, 3) sampling level indicator
letter;
%macro DO_SIII(iiso,ppien,wwb);
   nnq1=&ppien - 1; nowor=0;
   %do qqw=1 %to &iiso;
       %let qqwb=%eval(&qqw+1);
         hhig = (nnq&qqw>0) * (j&qqw=1) ;
         do jj&qqw = hhig to 0 by -1;
            %if &qqw = &iiso %then %do;
          %end;
          %if &qqw<&iiso %then %do; nnq&qqwb = nnq&qqw - jj&qqw ; %end;
            if &qqw = &iiso then do; botc=1; jjsu=0;
           do wwp=1 to &iiso; jjsu=sum(jjsu,jj(wwp)); botc=botc/gamma(jj(wwp) + 1);
end;
```

```
            if jjsu>0 then bott&wwb=gamma(jjsu + 1) * botc;
          end;
   %end;
%mend DO_SIII;

* ------------------ END MACROS ---------------------;
* making ends of do-loops ;
* in FirstPhaseSelections.sas, SecondPhaseSelections.sas, ThirdPhaseSelections.sas;
* Input info: 1) size of population;
%macro END_S(iso);
   %do qw=1 %to &iso; end; %end;
%mend END_S;

* in SecondPhasePairSelections.sas;
* Input info: 1) size of population;
%macro END_SFad(siva);
   end; %do qw=1 %to &siva; dben=dben-dj(&siva-&qw+1); if &qw=1 then ranni=1; end;
%end;
%mend END_SFad;

* ----------------- ESTIMATE CUMULATING MACROS ------------------;
* Constructing sums of estimates (WR resamples) ;
* including number of permutations of different wr samples (from bottb);
* in SecondPhaseSelections.sas;
* Input info: 1) size of population, 2) number of parameters to be estimated,
              3) general function letter (e.g. f or g);
%macro CUMU1(big,nu,c1);
    %let zq=1; %let warre=&&c&zq; %let murre=&warre;
      %if &warre ^= . %then %do;
          %let zm=.;
          * expectation and expectation of square;
          %let e=e(qm); %let es=es(qm);
          %do zm=1 %to &nu ;
              %let waffe=&murre&zm&e ; %let waffes=&murre&zm&es ;
              zumu=&zm;
              do qm=1 to &big -1;
               bing=f&zm ;
              if bing ne . and bn=qm then do;
               &waffe=sum(&waffe, bottb * bing ); &waffes=sum(&waffes, bottb *
bing**2);
               end;
              end;
          %end;
       %end;
%mend;

* Constructing sums of estimates for internal rescaling purposes (wr);
* including number of permutations of different wr samples (from bottb);
* in SecondPhaseSelections.sas;
* Input info:1)size of population, 2)number of parameters to be estimated, 3)general
function letter;
%macro CUMU1rs(big,nu,c1);
  %let zq=1; %let warre=&&c&zq; %let murre=&warre;
      %if &warre ^= . %then %do;
          %let zm=.; %let e=e(qm); %let es=es(qm);
          %do zm=1 %to &nu ;
              %let waffe=&murre&zm&e; %let waffes=&murre&zm&es ;
              zumu=&zm;
              do qm=1 to &big-1;
               bing=frf&zm ;
              if bing ne . and bn=qm then do;
               &waffe=sum(&waffe, bottb * bing ); &waffes=sum(&waffes, bottb *
bing**2);
               end;
              end;
          %end;
       %end;
```

```
%mend;

* Constructing sums of estimates (WOR resamples);
* in SecondPhaseSelections.sas;
* Input info: 1)size of population, 2) number of parameters to be estimated, general
function letter;
%macro CUMU2(big,nu,c1);
  %let zq=1; %let warre=&&c&zq; %let murre=&warre;
      %if &warre ^= . %then %do;
          %let zm=.; %let e=e(qm); %let es=es(qm);
          %do zm=1 %to &nu ;
              %let waffe=&murre&zm&e ; %let waffes=&murre&zm&es ;
              zumu=&zm;
              do qm=1 to &big-1;
               bing=f&zm ;
              if bing ne . and bn=qm then do;
               if worb=1 then &waffe=sum(&waffe, bottb * bing );
              if worb=1 then &waffes=sum(&waffes, bottb * bing**2);
               end;
              end;
          %end;
       %end;
%mend;

* Constructing sums of estimates (WOR resamples) for sample pairs;
* in SecondPhasePairSelections.sas;
* Input info: 1)size of population, 2) number of parameters to be estimated, general
function letter;
%macro CUMUD2(big,nu,c1);
  %let zq=1; %let warre=&&c&zq; %let murre=&warre;
      %if &warre ^= . %then %do;
          %let zm=.; %let e=e(qm); %let es=es(qm);
          %do zm=1 %to &nu ;
              %let waffe=&murre&zm&e ; %let waffes=&murre&zm&es ;
              zumu=&zm;
              do qm=1 to &big-1;
               sbing=dg&zm ; zing=f&zm ;
              if bing ne . and bn=qm then do;
               &waffe=sum(&waffe,  sbing * zing ); &waffes=sum(&waffes, (sbing *
zing)**2 );
               end;
              end;
          %end;
       %end;
%mend;

* Constructing sums of estimates (WOR resamples) for sample pairs;
* Specific macro for decomposition with weight adjustments;
* in SecondPhasePairSelections.sas;
* Input info: 1)size of population, 2) number of parameters to be estimated, general
function letter;
%macro CUMUDG2(big,nu,c1);
  %let zq=1; %let b=b;
    %let warre=&&c&zq; %let murre=&warre; %let murrb=&warre&b;
      %if &warre ^= . %then %do;
          %let zm=.; %let e=e(qm); %let es=es(qm);
          %do zm=1 %to &nu ;
              %if &zm=1 %then %let miu=avx; %if &zm=2 %then %let miu=acrxy;
              %if &zm=3 %then %let miu=arxy; %if &zm=4 %then %let miu=atx;
              %if &zm=5 %then %let miu=amex; %if &zm=6 %then %let miu=amax;
              %let waffe =&murre&zm&e ; %let waffes=&murre&zm&es ;
              %let waffb =&murrb&zm&e ; %let waffbs=&murrb&zm&es ;
              zumu=&zm;
              do qm=1 to &big-1;
               sbing=dg&zm ; zing=f&zm ;
           if bing ne . and bn=qm then do;
               &waffe=sum(&waffe, wholeterm*sbing * zing );
```

```
                &waffes=sum(&waffes, wholeterm*(sbing * zing)**2 );
                &waffb=sum(&waffb, otherterm*sbing*zing) ;
                &waffbs=sum(&waffbs, otherterm*(sbing*zing) **2) ;
              halsua=sbing*zing;
                end;
               end;
           %end;
       %end;
       otherterm=0;
%mend;

* Constructing sums of estimates (WOR resamples) for concluding results from third-
phase sampling;
* in SecondPhaseSelections.sas;
* Input info: 1)size of population, 2) number of parameters to be estimated, general
function letter;
%macro CUMUext2(big,nu,c1);
  %let zq=1;
    %let warre=&&c&zq; %let murre=&warre;
      %if &warre ^= . %then %do;
          %let zm=.; %let w=_; %let e=e(qm); %let es=es(qm);
          %do zm=1 %to &nu ;
             %do zis=1 %to &big-2;
              %let waffe=&murre&zm&w&zis&e ; %let waffes=&murre&zm&w&zis&es ;
              zumu=&zm;
              do qm=1 to &big-1;
               bing=gzu&zm&w&zis ; ua=&zm; uc=&zis;
             if bing ne . and bn=qm then do;
               if worb=1 then &waffe=sum(&waffe, bottb * bing );
               if worb=1 then &waffes=sum(&waffes, bottb * bing**2);
           end;
             end;
             %end;
          %end;
       %end;

%mend;

* Constructing sums of estimates (WOR resamples) for BWO lower limit;
* in SecondPhaseSelections.sas;
* Input info: 1)size of population, 2) number of parameters to be estimated, general
function letter;
%macro CUMUbol(big,nu,c1);
  %let zq=1;
    %let warre=&&c&zq; %let murre=&warre;
      %if &warre ^= . %then %do;
          %let zm=.; %let e=e(qm); %let es=es(qm);
          %do zm=1 %to &nu ;
              %let waffe=&murre&zm&e ; %let waffes=&murre&zm&es ;
              zumu=&zm;
              do qm=1 to &big -1;
               bing=f&zm ;
             if bing ne . and bn=qm then do;
               if botlox ne . then &waffe=sum(&waffe, botlox * bing );
               if botlox ne . then &waffes=sum(&waffes, botlox * bing**2);
               end;
              end;
          %end;
       %end;
%mend;

* Constructing sums of estimates (WOR resamples) for BWO lower limit;
* in SecondPhaseSelections.sas;
* Input info: 1)size of population, 2) number of parameters to be estimated, general
function letter;
%macro CUMUbou(big,nu,c1);
  %let zq=1;
```

```
    %let warre=&&c&zq; %let murre=&warre;
      %if &warre ^= . %then %do;
          %let zm=.; %let e=e(qm); %let es=es(qm);
          %do zm=1 %to &nu ;
              %let waffe=&murre&zm&e ; %let waffes=&murre&zm&es ;
              zumu=&zm;
              do qm=1 to &big -1;
               bing=f&zm ;
              if bing ne . and bn=qm then do;
               if botupx ne . then &waffe=sum(&waffe, botupx * bing );
               if botupx ne . then &waffes=sum(&waffes, botupx * bing**2);
               end;
              end;
          %end;
       %end;
%mend;

* Constructing sums of estimates for internal rescaling purposes (wor);
* in SecondPhaseSelections.sas;
* Input info: 1)size of population, 2) number of parameters to be estimated, general
function letter;
%macro CUMU2rs(big,nu,c1);
  %let zq=1;
    %let warre=&&c&zq; %let murre=&warre;
      %if &warre ^= . %then %do;
          %let zm=.; %let e=e(qm); %let es=es(qm);
          %do zm=1 %to &nu ;
              %let waffe=&murre&zm&e ; %let waffes=&murre&zm&es ;
              zumu=&zm;
              do qm=1 to &big-1;
               bing=frs&zm ;
              if bing ne . and bn=qm then do;
 if worb=1 then &waffe=sum(&waffe, bottb * bing ); if worb=1 then
&waffes=sum(&waffes,bottb*bing**2);
               end;
              end;
          %end;
       %end;
%mend;

* Constructing sums of estimates (WOR resamples) for post-design vector approach;
* in SecondPhaseSelections.sas;
* Input info: 1)size of population, 2) number of parameters to be estimated, general
function letter;
%macro CUMU2de(big,nu,c1);
  %let zq=1;  %let warre=&&c&zq; %let murre=&warre;
      %if &warre ^= . %then %do;
          %let zm=.;  %let e=e(&linbhigh); %let es=es(&linbhigh);
          %do zm=1 %to &nu ;
              %let waffe=&murre&zm&e ; %let waffes=&murre&zm&es ;
              zumu=&zm; bing=fde&zm ;
               * Note qm+1 as the resample size;
              if bing ne . then do;
               &waffe=sum(&waffe, bottb * bing ); &waffes=sum(&waffes, bottb *
bing**2);
               end;
          %end;
       %end;
%mend;

* Property sum calculations from third-phase sampling;
* in ThirdPhaseSelections.sas;
* Input info: 1)size of population, 2) number of parameters to be estimated, general
function letter;
%macro CUMUc2(big,nu,c1);
  %let zq=1; %let warre=&&c&zq; %let murre=&warre;
      %if &warre ^= . %then %do;
```

```
          %let zm=.; %let e=e(qm); %let es=es(qm);
          %do zm=1 %to &nu ;
              %let waffe=&murre&zm&e ; %let waffes=&murre&zm&es ;
              zumu=&zm;
              do qm=1 to &big-1;
               cbing=ff&zm ;
              if cbing ne . and cn=qm then do;
               &waffe=sum(&waffe, bottc * cbing ); &waffes=sum(&waffes, bottc *
cbing**2);
               end;
              end;
          %end;
       %end;
%mend;

* ---------- COUNTER MACROS FOR EXPECTATIONS ----------;

* Constructing number of valid estimate values (WR resamples);
* in SecondPhaseSelections.sas;
* Input info: 1)size of population, 2) number of parameters to be estimated, general
function letter;
%macro COUN1(big,nu,c1);
  %let la=la(qm); %let zq=1;  %let warre=&&c&zq; %let murre=&warre;
    %if &warre ^= . %then %do;
    %do zm=1 %to &nu ; zumu=&zm;
        do qm=1 to &big-1 ;
         if f&zm ne . and bn=qm then do;
        &murre&zm&la=sum(&murre&zm&la,bottb);
         end;
      end;
    %end;
  %end;
%mend;

* Constructing number of valid estimate values (WR resamples), internal rescaling;
* in SecondPhaseSelections.sas;
* Input info: 1)size of population, 2) number of parameters to be estimated, general
function letter;
%macro COUN1rs(big,nu,c1);
  %let la=la(qm); %let zq=1; %let warre=&&c&zq; %let murre=&warre;
    %if &warre ^= . %then %do;
    %do zm=1 %to &nu ; zumu=&zm;
        do qm=1 to &big-1 ;
         if frf&zm ne . and bn=qm then do; &murre&zm&la=sum(&murre&zm&la,bottb); end;
      end;
    %end;
  %end;

%mend;

* Constructing number of valid estimate values (WOR resamples);
* in SecondPhaseSelections.sas;
* Input info: 1)size of population, 2) number of parameters to be estimated, general
function letter;
%macro COUN2(big,nu,c1);
  %let la=la(qm); %let zq=1; %let warre=&&c&zq; %let murre=&warre;
    %if &warre ^= . %then %do;
    %do zm=1 %to &nu ; zumu=&zm;
        do qm=1 to &big-1 ;
         if f&zm ne . and bn=qm then do;
         if worb=1 then &murre&zm&la=sum(&murre&zm&la,bottb);
         end;
      end;
    %end;
   %end;

%mend;
```

```
* Constructing number of valid estimate values (WOR resamples), sample pairs;
* in SecondPhaseSelections.sas;
%macro COUND2(big,nu,c1);
  %let la=la(qm);
  %let zq=1;
    %let warre=&&c&zq; %let murre=&warre;
    %if &warre ^= . %then %do;
    %do zm=1 %to &nu ; zumu=&zm;
        do qm=1 to &big-1 ;
         if dg&zm ne . and bn=qm then do; &murre&zm&la=sum(&murre&zm&la,1); end;
      end;
    %end;
   %end;

%mend;

* Constructing number of valid estimate values (WOR resamples) for concluding results
from third-phase sampling;
* in SecondPhaseSelections.sas;
* Input info: 1) size of population, 2) number of parameters to be estimated, general
function letter;
%macro COUNext2(big,nu,c1);
  %let la=la(qm); %let w=_; %let zq=1; %let warre=&&c&zq; %let murre=&warre;
    %if &warre ^= . %then %do;
    %do zm=1 %to &nu ; zumu=&zm;
       %do zis=1 %to &big-2;
        do qm=1 to &big-1 ;
         if gzu&zm&w&zis ne . and bn=qm then do;
         if worb=1 then &murre&zm&w&zis&la=sum(&murre&zm&w&zis&la,bottb);
         end;
      end;
       %end;
    %end;
   %end;

%mend;

* Constructing number of valid estimate values (WOR resamples), for BWO lower limit;
* in SecondPhaseSelections.sas;
* Input info: 1) size of population, 2) number of parameters to be estimated, general
function letter;
%macro COUNbo1(big,nu,c1);
  %let la=la(qm); %let zq=1; %let warre=&&c&zq; %let murre=&warre;
    %if &warre ^= . %then %do;
    %do zm=1 %to &nu ; zumu=&zm;
        do qm=1 to &big-1 ;
         if f&zm ne . and bn=qm then do;
         if botlox ne .  then &murre&zm&la=sum(&murre&zm&la,botlox);
         end;
      end;
    %end;
   %end;
%mend;

* Constructing number of valid estimate values (WOR resamples), for BWO upper limit;
* in SecondPhaseSelections.sas;
%macro COUNbou(big,nu,c1);
  %let la=la(qm); %let zq=1; %let warre=&&c&zq; %let murre=&warre;
    %if &warre ^= . %then %do;
    %do zm=1 %to &nu ; zumu=&zm;
        do qm=1 to &big-1 ;
         if f&zm ne . and bn=qm then do;
         if botupx ne . then &murre&zm&la=sum(&murre&zm&la,botupx);
         end;
      end;
    %end;
```

```
   %end;
%mend;

* Constructing number of valid estimate values (WOR resamples), internal rescaling;
* in SecondPhaseSelections.sas;
* Input info: 1) size of population, 2) number of parameters to be estimated, general
function letter;
%macro COUN2rs(big,nu,c1);
  %let la=la(qm); %let zq=1;  %let warre=&&c&zq;       %let murre=&warre;
    %if &warre ^= . %then %do;
    %do zm=1 %to &nu ; zumu=&zm;
        do qm=1 to &big-1 ;
         if frs&zm ne . and bn=qm then do;
         if worb=1 then &murre&zm&la=sum(&murre&zm&la,bottb);
         end;
      end;
    %end;
   %end;

%mend;

* Constructing number of valid estimate values (WOR resamples), for post-design vector
approach;
* in SecondPhaseSelections.sas;
* Input info: 1) size of population, 2) number of parameters to be estimated, general
function letter;
%macro COUN2de(big,nu);
  king=&linbhigh;
  %let laz=la(king); %let murre=gde;
    %do zm=1 %to &nu ; zumu=&zm;
         if fde&zm ne . then do;
         &murre&zm&laz=sum(&murre&zm&laz,bottb);
         end;
    %end;

%mend;

* Property calculations for third-phase sampling, number of valid values;
* in ThirdPhaseSelections.sas;
* Input info: 1) size of population, 2) number of parameters to be estimated, general
function letter;
%macro COUNc2(big,nu,c1);
  %let la=la(qm); %let zq=1 ; %let warre=&&c&zq; %let murre=&warre;
    %if &warre ^= . %then %do;
    %do zm=1 %to &nu ; zumu=&zm;
        do qm=1 to &big-1 ;
         if ff&zm ne . and cn=qm then do;
         &murre&zm&la=sum(&murre&zm&la,bottc);
         end;
      end;
    %end;
   %end;
%mend;

* ---------- VALUE ASSIGNING MACROS ----------;
* In SecondPhaseSelections.sas;
* Input info: 1) Size of population, 2) value to be assigned, 3) beginning of variable
;

%macro VAL(big,val,wa);
        %do qw=1 %to &big;  &wa&&qw=&val ;  %end;
%mend VAL;

* in FirstPhaseSelections.sas, SecondPhaseSelections.sas, MACRO VALPACC;
* Input info: 1) Size of population, 2) value to be assigned, 3) beginning of
variable,
              4) number of parameters to be estimated;
```

```
%macro VALPAC(dime,valu,fi,nu);
   %let yx=1; %let e=e(ze); %let eq=eq(ze); %let es=es(ze) ; %let v=v(ze); %let
la=la(ze);
     %do zm=1 %to &nu;
       do ze=1 to &dime;
         &fi&zm&e=&valu; &fi&zm&es=&valu ; &fi&zm&eq=&valu ; &fi&zm&v=&valu ;
&fi&zm&la=&valu ;
       end;
     %end;
%mend VALPAC;

* Input info: 1) size of population, 2) value to be assigned, 3) beginning of
variable,
              4) number of parameters to be estimated;

%macro VALPAge(dime,valu,fi,nu);

   %let yx=1; %let e=e; %let eq=eq; %let es=es ; %let v=v; %let la=la; %let
b=&linbhigh; %let w=_ ;
     %do zm=1 %to &nu;
       %do ze=1 %to &dime;
           &fi&zm&e&b = &valu ;  &fi&zm&es&b = &valu ;
             &fi&zm&eq&b = &valu ; &fi&zm&v&b = &valu ;  &fi&zm&la&b = &valu ;
       %end;
     %end;


%mend VALPAge;

* Input info: 1) size of population, 2) value to be assigned, 3) beginning of
variable,
              4) number of parameters to be estimated;

%macro VALPASPE(dime,valu,fi,nu);

   %let yx=1; %let e=e(ze); %let eq=eq(ze); %let es=es(ze) ; %let v=v(ze);  %let
la=la(ze); %let w=_;
     %do zm=1 %to &nu;
        %do zis=1 %to %eval(&dime-2);
       do ze=1 to &dime;
           &fi&zm&w&zis&e = &valu ;  &fi&zm&w&zis&es = &valu ;
             &fi&zm&w&zis&eq = &valu ; &fi&zm&w&zis&v = &valu ; &fi&zm&w&zis&la =
&valu ;
       end;
        %end;
     %end;

%mend VALPASPE;

* Values for a large amount of different variables;
* Input info: 1)size of population, 2) value to be assigned, 3) number of parameters
to be estimated,
              4) --> beginnings of variables;

%macro
VALPACC(big,va,nu,c1,c2,c3,c4,c5,c6,c7,c8,c9,c10,c11,c12,c13,c14,c15,c16,c17,c18,c19,c
20);

   %do zq=1 %to 20;
      %let warre=&&c&zq;
      %if &warre ^= . %then %do;  %VALPAC(&big,&va,&warre,&nu); %end;
   %end;

%mend;
```

```
* ------------------ ORDERING MACROS ------------------;
* Giving order numbers of x for observations, for first-phase sampling;
* Input info: 1) size of population, 2) inclusion variable, 3) sample size;
%macro ORDPA(big,inde,co);
          %let rrz=(rz);
          ianu=0;
              do rz=1 to &big;
             if &inde&rrz>0 then do;
                 do ez=1 to &inde&rrz; ianu=sum(ianu,1); ialu(ianu)=rz; end;
             end;
          end;
              do kuf=&co+1 to &CapN; ialu(kuf)=.; end;
%mend ORDPA;

* Giving order numbers of x for observations, for second-phase sampling ;
* Input info: 1) size of population, 2) inclusion variable, 3) sample size;
%macro ORDPAC(big,inde,co);
          %let rrz=(rz);
          janu=0;
              do rz=1 to &big;
             if &inde&rrz>0 then do;
                  do ez=1 to &inde&rrz; janu=sum(janu,1); jalu(janu)=rz; end;
             end;
          end;
              do kuf=&co+1 to &CapN; jalu(kuf)=.; end;

%mend ORDPAC;

* Giving order numbers of x for observations, for third-phase sampling;
* Input info: 1) size of population, 2) inclusion variable, 3) sample size;
%macro ORDPACB(bbig,iinde,cco);
          %let rrrz=(rrz);
          jjanu=0;
             do rrz=1 to &bbig;
             if 0 < &iinde&rrrz <= &bbig then do;
                 do eez=1 to &iinde&rrrz; jjanu=sum(jjanu,1); jjalu(jjanu)=rrz; end;
             end;
          end;
             do kkuf=&cco+1 to &CapN; jjalu(kkuf)=.; end;
%mend ORDPACB;

* ------------------ PROPERTY CALCULATING MACROS ------------------;
* Calculating properties of conditional estimators, based on CUMU and COUN macros  ;
* in FirstPhaseSelections.sas ;
* Input info: 1) size of population, 2) numbero of parameters to be estimated, 3)-->
beginnings of variables ;
%macro
PACK1(big,nu,c1,c2,c3,c4,c5,c6,c7,c8,c9,c10,c11,c12,c13,c14,c15,c16,c17,c18,c19,c20);
  %let la=la(qm); %let e=e(qm) ; %let es=es(qm) ; %let eq=eq(qm); %let v=v(qm);
  %do zq=1 %to 20;
   %let warre=&&c&zq; %let murre=&warre;
   %if &warre ^= . %then %do;
   %do zm=1 %to &nu ;
     zum=&zm; zequ=&zq;
     do qm=1 to &big - 1;
          if &murre&zm&la>0 then &murre&zm&e =&murre&zm&e/&murre&zm&la;
          if &murre&zm&la>0 then &murre&zm&es=&murre&zm&es/&murre&zm&la;
        if &murre&zm&la>0 then &murre&zm&eq=&murre&zm&e*&murre&zm&e ;
       if &murre&zm&la>0 then &murre&zm&v =&murre&zm&es-&murre&zm&eq ;
     end;
   %end;
  %end;
  %end;
%mend PACK1;

* Calculating properties of conditional estimators, based on CUMU and COUN macros
(decomposition)  ;
```

```
* in FirstPhaseSelections.sas ;
%macro PACKSIF(big,nu,c1);
  %let la=la(qm); %let e=e(qm) ; %let es=es(qm) ; %let eq=eq(qm); %let v=v(qm); %let
zq=1 ; %let b=b;
   %let warre=&&c&zq; %let murre=&warre; %let murrb=&warre&b;
   %if &warre ^= . %then %do;
   %do zm=1 %to &nu ;
     zum=&zm; zequ=&zq;
     do qm=1 to &big - 1;
        &murre&zm&eq=&murre&zm&e*&murre&zm&e ; &murre&zm&v =&murre&zm&es-&murre&zm&eq
;
        &murrb&zm&eq=&murrb&zm&e*&murrb&zm&e ; &murrb&zm&v =&murrb&zm&es-&murrb&zm&eq
;
     end;
   %end;
  %end;

%mend PACKSIF;

* Calculating properties of conditional estimators, based on CUMU and COUN macros
(concluding third phase)  ; * in SecondPhaseSelections.sas ;
%macro PACK3(big,nu,c1);
  %let la=la(qm); %let e=e(qm) ; %let es=es(qm) ; %let eq=eq(qm); %let v=v(qm); %let
zq=1;
   %let warre=&&c&zq; %let murre=&warre;
   %if &warre ^= . %then %do;
   %do zm=1 %to &nu ;
     zum=&zm; zequ=&zq;
     do qm=1 to &big - 1;
          if &murre&zm&la>0 then &murre&zm&e =&murre&zm&e/&murre&zm&la;
          if &murre&zm&la>0 then &murre&zm&es=&murre&zm&es/&murre&zm&la;
        if &murre&zm&la>0 then &murre&zm&eq=&murre&zm&e*&murre&zm&e ;
        if &murre&zm&la>0 then &murre&zm&v =&murre&zm&es-&murre&zm&eq ;
     end;
   %end;
  %end;
%mend PACK3;

* Calculating properties of conditional estimators, based on CUMU and COUN macros
(third-phase based
  estimators); * in FirstPhaseSelections.sas ;
%macro PACKSPE(big,nu,c1);
  %let la=la(qm); %let e=e(qm) ; %let es=es(qm) ; %let eq=eq(qm); %let v=v(qm);  %let
zq=1;
   %let warre=&&c&zq;  %let murre=&warre;  %let w=_;
   %if &warre ^= . %then %do;
   %do zm=1 %to &nu ;
     zum=&zm; zequ=&zq;
       %do zik=1 %to %eval(&big-2);
         sika=&zik;
       do qm=1 to &big - 1;
          if &murre&zm&w&zik&la>0 then &murre&zm&w&zik&e
=&murre&zm&w&zik&e/&murre&zm&w&zik&la;
            if &murre&zm&w&zik&la>0 then
&murre&zm&w&zik&es=&murre&zm&w&zik&es/&murre&zm&w&zik&la;
        if &murre&zm&w&zik&la>0 then
&murre&zm&w&zik&eq=&murre&zm&w&zik&e*&murre&zm&w&zik&e ;
        if &murre&zm&w&zik&la>0 then &murre&zm&w&zik&v =&murre&zm&w&zik&es-
&murre&zm&w&zik&eq ;
       end;
       %end;
   %end;
  %end;
%mend PACKSPE;


* ---------- MACRO IDENTIFYING WOR SAMPLE ---------- ;
```

```
* in SecondPhaseSelections.sas;
* Input info: 1) size of population;
%macro SEPAR(big);
      %let bim=%eval(&big-1);
        worb=(
          %do iba=1 %to &bim; (j&iba in (0,1)) &   %end;
          (j&big in (0,1))) ;
      %let bim=%eval(&big-1);
      numoftwo=(
          %do iba=1 %to &bim; (j&iba = 2) +  %end;
          (j&big = 2)) ;
        numofone=(
          %do iba=1 %to &bim; (j&iba = 1) +  %end;
          (j&big = 1)) ;

%mend;


* ---------------------- SUM MACROS -------------------;
* Term for taylor approximation of variance S2x ; * in FirstPhaseSelections.sas;
* Input info: 1) dimension of population, 2) id-vector variable,
              3) sum of x in sample, 4) variance of x in sample, 5) sample size;
%macro VATEXSUM(iso,co,kesk,varm,oko);
       vate=
   %do qw=1 %to &iso-1; &co&qw*((x&qw-&kesk/&oko)**2 -  (&oko-1)* &varm / &oko )**2 +
%end;
       &co&iso*((x&iso-&kesk/&oko)**2 - (&oko-1)* &varm / &oko )**2;
%mend VATEXSUM;

* Summing up different terms;
* in FirstPhaseSelections.sas, SecondPhaseSelections.sas,
SecondPhasePairSelections.sas,
     ThirdPhaseSelections.sas;
* Input info: 1) dimension of population, 2) phase, 3) vector identification,
              4) variable 1, 5) variable 2, 6) end of frequency variable;
%macro SUMPAC(big,be,co,x,y,com);
   %let q=q; %let ka=2; %let s=s;
   %let koppi = &co&big ; &be&com = &&ms&koppi ; %let kopx = &x&co&big ; &be&s&x =
&&ms&kopx ;
   %let kopy = &y&co&big ; &be&s&y = &&ms&kopy ; %let kopxq= &x&q&co&big ; &be&s&x&q =
&&ms&kopxq ;
   %let kopyq = &y&q&co&big ; &be&s&y&q = &&ms&kopyq ; %let kopxyq = &x&y&co&big ;
   &be&s&x&y = &&ms&kopxyq ; &be&s&x&ka = &be&s&x**2;&be&s&y&ka = &be&s&y**2;
%mend;

* Summing up different terms (internal wor rescaling); * in SecondPhaseSelections;
* Input info: 1) dimension of population, 2) phase, 3) vector identification,
              4) variable 1, 5) variable 2, 6) extra ending (rs, rf), 7) end of
frequency variable;

%macro SUMPAWO(big,be,co,x,y,paa,com);
   %let q=q; %let ka=2; %let s=s;
       &be&s&x&paa =
        %do qw=1 %to &big-1; i&qw*(worsa(bn)+ &co&qw * worsb(bn))*&x&qw +  %end;
       i&big*(worsa(bn)+ &co&big * worsb(bn))* &x&big ;

       &be&s&y&paa =
        %do qw=1 %to &big-1; i&qw*(worsa(bn)+ &co&qw * worsb(bn))*&y&qw +   %end;
       i&big*(worsa(bn)+ &co&big * worsb(bn))* &y&big ;
       &be&s&x&paa&q =
        %do qw=1 %to &big-1;  i&qw*(worsa(bn)+ &co&qw * worsb(bn))*&x&qw*&x&qw  +
%end;
       i&big*(worsa(bn)+ &co&big * worsb(bn))* &x&big*&x&big ;
       &be&s&y&paa&q =
        %do qw=1 %to &big-1; i&qw*(worsa(bn)+ &co&qw * worsb(bn))*&y&qw*&y&qw +
%end;
       i&big*(worsa(bn)+ &co&big * worsb(bn))* &y&big*&y&big ;
```

```
      &be&s&x&paa&y&paa =
       %do qw=1 %to &big-1; i&qw*(worsa(bn)+ &co&qw * worsb(bn))*&x&qw*&y&qw +
%end;
      i&big*(worsa(bn)+ &co&big * worsb(bn))* &x&big*&y&big ;
      &be&s&x&paa&ka = &be&s&x&paa**2;  &be&s&y&paa&ka = &be&s&y&paa**2;
%mend;

* Summing up different terms (internal wr rescaling);* in SecondPhaseSelections;
* Input info: 1) dimension of population, 2) phase, 3) vector identification,
              4) variable 1, 5) variable 2, 6) extra ending (rs, rf), 7) end of
frequency variable;

%macro SUMPAWR(big,be,co,x,y,paa,com);

   %let q=q; %let ka=2; %let s=s;
      &be&s&x&paa =
       %do qw=1 %to &big-1; i&qw*(wrsa(bn)+ &co&qw * wrsb(bn))*&x&qw +  %end;
      i&big*(wrsa(bn)+ &co&big * wrsb(bn))* &x&big ;
      &be&s&y&paa =
       %do qw=1 %to &big-1;  i&qw*(wrsa(bn)+ &co&qw * wrsb(bn))*&y&qw +  %end;
      i&big*(wrsa(bn)+ &co&big * wrsb(bn))* &y&big ;
      &be&s&x&paa&q =
       %do qw=1 %to &big-1;  i&qw*(wrsa(bn)+ &co&qw * wrsb(bn))*&x&qw*&x&qw  +  %end;
      i&big*(wrsa(bn)+ &co&big * wrsb(bn))* &x&big*&x&big ;
      &be&s&y&paa&q =
       %do qw=1 %to &big-1;  i&qw*(wrsa(bn)+ &co&qw * wrsb(bn))*&y&qw*&y&qw +  %end;
      i&big*(wrsa(bn)+ &co&big * wrsb(bn))* &y&big*&y&big ;
      &be&s&x&paa&y&paa =
       %do qw=1 %to &big-1; i&qw*(wrsa(bn)+ &co&qw * wrsb(bn))*&x&qw*&y&qw +  %end;
      i&big*(wrsa(bn)+ &co&big * wrsb(bn))* &x&big*&y&big ;
      &be&s&x&paa&ka = &be&s&x&paa**2;  &be&s&y&paa&ka = &be&s&y&paa**2;
%mend;

* Summing up post-design vector terms; * in SecondPhaseSelections;
%macro SUMPDE;
   %let q=q; %let ka=2; %let s=s;
      bnde =
       %do qw=1 %to &CapN-1; (j&qw=2) + (j&qw=1)*desvcoef +   %end;
      (j&CapN=2) + (j&CapN=1)*desvcoef;
      bsyde =
       %do qw=1 %to &CapN-1; (j&qw=2) * y&qw + (j&qw=1)*j&qw*desvcoef*y&qw +   %end;
       (j&CapN=2) * y&CapN +  (j&CapN=1)*desvcoef*y&CapN ;
      bsxde =
      %do qw=1 %to &CapN-1; (j&qw=2) * x&qw + (j&qw=1)*j&qw*desvcoef*x&qw +  %end;
       (j&CapN=2) * x&CapN + (j&CapN=1)*desvcoef*x&CapN ;
      bsxdeq =
       %do qw=1 %to &CapN-1; (j&qw=2) * x&qw*x&qw + (j&qw=1)*j&qw*desvcoef*x&qw*x&qw
+   %end;
       (j&CapN=2) * x&CapN*x&CapN + (j&CapN=1)*desvcoef*x&CapN*x&CapN ;
      bsydeq =
       %do qw=1 %to &CapN-1; (j&qw=2) * y&qw*y&qw + (j&qw=1)*j&qw*desvcoef*y&qw*y&qw
+   %end;
       (j&CapN=2) * y&CapN*y&CapN + (j&CapN=1)*desvcoef*y&CapN*y&CapN ;
      bsxdeyde =
      %do qw=1 %to &CapN-1; (j&qw=2) * y&qw * x&qw + (j&qw=1)*j&qw*desvcoef*y&qw*x&qw
+   %end;
       (j&CapN=2) * y&CapN*x&CapN + (j&CapN=1)*desvcoef*y&CapN*x&CapN ;
      bsxde2 = bsxde**2; bsyde2 = bsyde**2;
%mend;

* ----------------- QUANTILE MACROS --------------;

* Quantile calculations (first phase); * in FirstPhaseSelections.sas;
* Input info: 1) dimension of population, 2) phase, 3) variable 1,
              4) variable 2, 5) end of frequency variable;

%macro QUAPA(big,be,x,y,co);
```

```
      %GLOBAL WNOTE ; %let me=me; %let ma=ma; surf = &be&&co;
       if surf > 0 then do;
            *WOODRUFF;
            bim=&LMCO; bom=&UMCO; ikka=surf*&LMCO; ikky=surf*&UMCO;
         if int(ikka)=int(ikky) then do;
            * here we inform the tabulation about the adjustment;
           woodnote=1; ikky=ikky+1;
            end;
            else woodnote=0;
        call symput('WNOTE',woodnote); wolow=&x(ialu(int(ikka)));
wohigh=&x(ialu(int(ikky)));
       if int(surf/2)=surf/2 then do;
            ialua=ialu(surf/2); ialuy=ialu((surf/2)+1);
            alanu=&x(ialua); ylanu=&x(ialuy); &be&me&x=alanu; meddi=&be&me&x ;
        end;
       else do;
        ialua=.;ialuy=.;alanu=.;ylanu=.;medja=(surf+1)/2; &be&me&x=&x(ialu(medja));
meddi=&be&me&x;
       end;
      end;
      else do; &be&me&x=.; meddi=.; end;
      if surf>0 then do;
         &be&ma&x=&x(ialu(surf)); maksi=&be&ma&x;
        end; else do;
         &be&ma&x=.; maksi=.;
        end;
%mend;

* Quantile calculations (second phase); * in SecondPhaseSelections.sas;
* Input info: 1)dimension of population, 2)phase, 3)variable 1, 4)variable 2, 5)end of
frequency variable;

%macro QUAPAC(big,be,x,y,co);
      %let me=me; %let ma=ma;
      surf = &be&&co;
       if surf > 0 then do;
       if int(surf/2)=surf/2 then do;
            jalua=jalu(surf/2); jaluy=jalu((surf/2)+1); alanu=&x(jalua);
ylanu=&x(jaluy);
            &be&me&x=alanu; meddi=&be&me&x ;
        end;
       else do;
            jalua=.; jaluy=.; alanu=.; ylanu=.; medja=(surf+1)/2;
&be&me&x=&x(jalu(medja));
         meddi=&be&me&x ;
          end;
      end;
      else do; &be&me&x=.; meddi=.; end;
      if surf>0 then do;
         &be&ma&x=&x(jalu(surf)); maksi=&be&ma&x;
        end; else do;
         &be&ma&x=.; maksi=.;
        end;
%mend;
* Quantile calculations (third phase); * in ThirdPhaseSelections.sas;
* Input info: 1) dimension of population, 2) phase, 3) variable 1,
            4) variable 2, 5) end of frequency variable;
%macro QUAPACC(big,be,x,y,co);
      %let me=me; %let ma=ma;
      surf = &be&&co;
       if surf > 0 then do;
       if int(surf/2)=surf/2 then do;
            jjalua=jjalu(surf/2); jjaluy=jjalu((surf/2)+1);
            alanu=&x(jjalua); ylanu=&x(jjaluy); &be&me&x=alanu; meddi=&be&me&x ;
        end;
       else do;
            jjalua=.; jjaluy=.; alanu=.; ylanu=.; medja=(surf+1)/2;
```

```
          &be&me&x=&x(jjalu(medja)); meddi=&be&me&x ;
            end;
        end;
          else do; &be&me&x=.; meddi=.; end;
        if surf>0 then do; &be&ma&x=&x(jjalu(surf)); maksi=&be&ma&x; end;
        else do; &be&ma&x=.; maksi=.; end;
%mend;
* ------------------- POPULATION SORT MACRO -------------------;
* Population of two variables sorted by x; * In Simulations.sas;
%macro BEGISORT;
do dw=1 to &CapN-1;
 * getting zeroes to index variables;
 i(dw)=0; j(dw)=0; jj(dw)=0;
 do dw2=dw+1 to &CapN;
   if x(dw2)<x(dw) then do;
     valix=x(dw2); x(dw2)=x(dw); x(dw)=valix; valiy=y(dw2); y(dw2)=y(dw); y(dw)=valiy;
   end; end; end;
* further zeroes;
i(&CapN)=0; j(&CapN)=0; jj(&CapN)=0;
drop valix valiy;
%mend BEGISORT;

* ================= MacroSums.sas ==================;
* This print includes only a part of the program. One should make these functions also
for letter combinations j, dj, jj, xi, xj, xjj, xdj, yi, yj, yjj, ydj, xqi, xqj, xqjj,
xqdj, yqi, yqj, yqjj, yqdj, xyi, xyj, xyjj and xydj ;
* In addition, one should make the corresponding functions from 20 down to 2 (if
requiring population size more than 20, it is possible to do so);
* Using these functions fastens the program;

* examples;
%let msi20=sum(i1,i2,i3,i4,i5,i6,i7,i8,i9,i10,i11,i12,i13,i14,i15,i16,i17,i18,i19,i20)
;
%let msyj7=sum(y1*j1,y2*j2,y3*j3,y4*j4,y5*j5,y6*j6,y7*j7) ;
%let msxqdj4=sum(x1*x1*dj1,x2*x2*dj2,x3*x3*dj3,x4*x4*dj4) ;
%let msxyj6=sum(y1*x1*j1,y2*x2*j2,y3*x3*j3,y4*x4*j4,y5*x5*j5,y6*x6*j6) ;

%global
 msj2  msj3  msj4  msj5  msj6  msj7  msj8  msj9  msj10  msj11 msj12 msj13 msj14 msj15
msj16 msj17 msj18 msj19 msj20
 msjj2  msjj3  msjj4  msjj5  msjj6  msjj7  msjj8  msjj9  msjj10  msjj11 msjj12 msjj13
msjj14 msjj15 msjj16 msjj17 msjj18 msjj19 msjj20
 msi2  msi3  msi4  msi5  msi6  msi7  msi8  msi9  msi10  msi11 msi12 msi13 msi14 msi15
msi16 msi17 msi18 msi19 msi20
 msdj2  msdj3  msdj4  msdj5  msdj6  msdj7  msdj8  msdj9  msdj10  msdj11 msdj12 msdj13
msdj14 msdj15 msdj16 msdj17 msdj18 msdj19 msdj20

 msxj2  msxj3  msxj4  msxj5  msxj6  msxj7  msxj8  msxj9  msxj10  msxj11 msxj12 msxj13
msxj14 msxj15 msxj16 msxj17 msxj18 msxj19 msxj20
 msxjj2  msxjj3  msxjj4  msxjj5  msxjj6  msxjj7  msxjj8  msxjj9  msxjj10  msxjj11
msxjj12 msxjj13 msxjj14 msxjj15 msxjj16 msxjj17 msxjj18 msxjj19 msxjj20
 msxi2  msxi3  msxi4  msxi5  msxi6  msxi7  msxi8  msxi9  msxi10  msxi11 msxi12 msxi13
msxi14 msxi15 msxi16 msxi17 msxi18 msxi19 msxi20
 msxdj2  msxdj3  msxdj4  msxdj5  msxdj6  msxdj7  msxdj8  msxdj9  msxdj10  msxdj11
msxdj12 msxdj13 msxdj14 msxdj15 msxdj16 msxdj17 msxdj18 msxdj19 msxdj20

 msyj2  msyj3  msyj4  msyj5  msyj6  msyj7  msyj8  msyj9  msyj10  msyj11 msxy12 msyj13
msyj14 msyj15 msyj16 msyj17 msyj18 msyj19 msyj20
 msyjj2  msyjj3  msyjj4  msyjj5  msyjj6  msyjj7  msyjj8  msyjj9  msyjj10  msyjj11
msyjj12 msyjj13 msyjj14 msyjj15 msyjj16 msyjj17 msyjj18 msyjj19 msyjj20
 msyi2  msyi3  msyi4  msyi5  msyi6  msyi7  msyi8  msyi9  msyi10  msyi11 msyi12 msyi13
msyi14 msyi15 msyi16 msyi17 msyi18 msyi19 msyi20
 msydj2  msydj3  msydj4  msydj5  msydj6  msydj7  msydj8  msydj9  msydj10  msydj11
msydj12 msydj13 msydj14 msydj15 msydj16 msydj17 msydj18 msydj19 msydj20

 msxqj2  msxqj3  msxqj4  msxqj5  msxqj6  msxqj7  msxqj8  msxqj9  msxqj10  msxqj11
msxqj12 msxqj13 msxqj14 msxqj15 msxqj16 msxqj17 msxqj18 msxqj19 msxqj20
```

```
 msxqjj2  msxqjj3  msxqjj4  msxqjj5  msxqjj6  msxqjj7  msxqjj8  msxqjj9  msxqjj10
msxqjj11 msxqjj12 msxqjj13 msxqjj14 msxqjj15 msxqjj16 msxqjj17 msxqjj18 msxqjj19
msxqjj20
 msxqi2  msxqi3  msxqi4  msxqi5  msxqi6  msxqi7  msxqi8  msxqi9  msxqi10  msxqi11
msxqi12 msxqi13 msxqi14 msxqi15 msxqi16 msxqi17 msxqi18 msxqi19 msxqi20
 msxqdj2  msxqdj3  msxqdj4  msxqdj5  msxqdj6  msxqdj7  msxqdj8  msxqdj9  msxqdj10
msxqdj11 msxqdj12 msxqdj13 msxqdj14 msxqdj15 msxqdj16 msxqdj17 msxqdj18 msxqdj19
msxqdj20

 msyqj2  msyqj3  msyqj4  msyqj5  msyqj6  msyqj7  msyqj8  msyqj9  msyqj10  msyqj11
msyqj12 msyqj13 msyqj14 msyqj15 msyqj16 msyqj17 msyqj18 msyqj19 msyqj20
 msyqjj2  msyqjj3  msyqjj4  msyqjj5  msyqjj6  msyqjj7  msyqjj8  msyqjj9  msyqjj10
msyqjj11 msyqjj12 msyqjj13 msyqjj14 msyqjj15 msyqjj16 msyqjj17 msyqjj18 msyqjj19
msyqjj20
 msyqi2  msyqi3  msyqi4  msyqi5  msxqi6  msyqi7  msyqi8  msyqi9  msyqi10  msyqi11
msyqi12 msyqi13 msyqi14 msxqi15 msyqi16 msyqi17 msyqi18 msyqi19 msyqi20
 msyqdj2  msyqdj3  msyqdj4  msyqdj5  msyqdj6  msyqdj7  msyqdj8  msyqdj9  msyqdj10
msyqdj11 msyqdj12 msyqdj13 msyqdj14 msyqdj15 msyqdj16 msyqdj17 msyqdj18 msyqdj19
msyqdj20

 msxyj2  msxyj3  msxyj4  msxyj5  msxyj6  msxyj7  msxyj8  msxyj9  msxyj10  msxyj11
msxyj12 msxyj13 msxyj14 msxyj15 msxyj16 msxyj17 msxyj18 msxyj19 msxyj20
 msxyjj2  msxyjj3  msxyjj4  msxyjj5  msxyjj6  msxyjj7  msxyjj8  msxyjj9  msxyjj10
msxyjj11 msxyjj12 msxyjj13 msxyjj14 msxyjj15 msxyjj16 msxyjj17 msxyjj18 msxyjj19
msxyjj20
 msxyi2  msxyi3  msxyi4  msxyi5  msxyi6  msxyi7  msxyi8  msxyi9  msxyi10  msxyi11
msxyi12 msxyi13 msxyi14 msxyi15 msxyi16 msxyi17 msxyi18 msxyi19 msxyi20
 msxydj2  msxydj3  msxydj4  msxydj5  msxydj6  msxydj7  msxydj8  msxydj9  msxydj10
msxydj11 msxydj12 msxydj13 msxydj14 msxydj15 msxydj16 msxydj17 msxydj18 msxydj19
msxydj20

;
```

TUTKIMUKSIA-SARJA
RESEARCH REPORTS SERIES

Tilastokeskus on julkaissut Tutkimuksia v. 1966 alkaen, v. 1990 lähtien ovat ilmestyneet seuraavat:

164. **Henry Takala,** Kunnat ja kuntainliitot kansantalouden tilinpidossa. Tammikuu 1990. 60 s.

165. **Jarmo Hyrkkö,** Palkansaajien ansiotasoindeksi 1985=100. Tammikuu 1990. 66 s.

166. **Pekka Rytkönen,** Siivouspalvelu, ympäristöhuolto ja pesulapalvelu 1980-luvulla. Tammikuu 1990. 70 s.

167. **Jukka Muukkonen,** Luonnonvaratilinpito kestävän kehityksen kuvaajana. 1990. 119 s.

168. **Juha-Pekka Ollila,** Tieliikenteen tavarankuljetus 1980-luvulla. Helmikuu 1990. 45 s.

169. **Tuovi Allén – Seppo Laaksonen – Päivi Keinänen – Seija Ilmakunnas,** Palkkaa työstä ja sukupuolesta. Huhtikuu 1990. 90 s.

170. **Ari Tyrkkö,** Asuinolotiedot väestölaskennassa ja kotitaloustiedustelussa. Huhtikuu 1990. 63 s.

171. **Hannu Isoaho – Osmo Kivinen – Risto Rinne,** Nuorten koulutus ja kotitausta. Toukokuu 1990. 115 s.

171b. **Hannu Isoaho – Osmo Kivinen – Risto Rinne,** Education and the family background of the young in Finland. 1990. 115 pp.

172. **Tapani Valkonen – Tuija Martelin – Arja Rimpelä,** Eriarvoisuus kuoleman edessä. Sosioekonomiset kuolleisuuserot Suomessa 1971–85. Kesäkuu 1990. 145 s.

173. **Jukka Muukkonen,** Sustainable development and natural resource accounting. August 1990. 96 pp.

174. **Iiris Niemi – Hannu Pääkkönen,** Time use changes in Finland in the 1980s. August 1990. 118 pp.

175. **Väinö Kannisto,** Mortality of the elderly in late 19th and early 20th century Finland. August 1990. 50 pp.

176. **Tapani Valkonen – Tuija Martelin – Arja Rimpelä,** Socio-economic mortality differences in Finland 1971–85. December 1990. 108 pp.

177. **Jaana Lähteenmaa – Lasse Siurala,** Nuoret ja muutos. Tammikuu 1991. 211 s.

178. **Tuomo Martikainen – Risto Yrjönen,** Vaalit, puolueet ja yhteiskunnan muutos. Maaliskuu 1991. 120 s.

179. **Seppo Laaksonen,** Comparative Adjustments for Missingness in Short-term Panels. April 1991. 74 pp.

180. **Ágnes Babarczy – István Harcsa – Hannu Pääkkönen,** Time use trends in Finland and in Hungary. April 1991. 72 pp.

181. **Timo Matala,** Asumisen tuki 1988. Kesäkuu 1991. 64 s.

182. **Iiris Niemi – Parsla Eglite – Algimantas Mitrikas – V.D. Patrushev – Hannu Pääkkönen,** Time Use in Finland, Latvia, Lithuania and Russia. July 1991. 80 pp.

183. **Iiris Niemi – Hannu Pääkkönen,** Vuotuinen ajankäyttö. Joulukuu 1992. 83 s.

183b **Iiris Niemi – Hannu Pääkkönen – Veli Rajaniemi – Seppo Laaksonen – Jarmo Lauri,** Vuotuinen ajankäyttö. Ajankäyttötutkimuksen 1987–88 taulukot. Elokuu 1991. 116 s.

184. **Ari Leppälahti – Mikael Åkerblom,** Industrial Innovation in Finland. August 1991. 82 pp.

185. **Maarit Säynevirta,** Indeksiteoria ja ansiotasoindeksi. Lokakuu 1991. 95 s.

186. **Ari Tyrkkö,** Ahtaasti asuvat. Syyskuu 1991. 134 s.

187. **Tuomo Martikainen – Risto Yrjönen,** Voting, parties and social change in Finland. October 1991. 108 pp.

188. **Timo Kolu,** Työelämän laatu 1977–1990. Työn ja hyvinvoinnin koettuja muutoksia. Tammikuu 1992. 194 s.

189. **Anna-Maija Lehto,** Työelämän laatu ja tasa-arvo. Tammikuu 1992. 196 s.

190. **Tuovi Allén – Päivi Keinänen – Seppo Laaksonen – Seija Ilmakunnas,** Wage from Work and Gender. A Study on Wage Differentials in Finland in 1985. 88 pp.

191. **Kirsti Ahlqvist,** Kodinomistajaksi velalla. Maaliskuu 1992. 98 s.

192. **Matti Simpanen – Irja Blomqvist,** Aikuiskoulutukseen osallistuminen. Aikuiskoulutustutkimus 1990. Toukokuu 1992. 135 s.

193. **Leena M. Kirjavainen – Bistra Anachkova – Seppo Laaksonen – Iiris Niemi – Hannu Pääkkönen** – Zahari Staikov, Housework Time in Bulgaria and Finland. June 1992. 131 pp.

194. **Pekka Haapala – Seppo Kouvonen,** Kuntasektorin työvoimakustannukset. Kesäkuu 1992. 70 s.

195. **Pirkko Aulin-Ahmavaara,** The Productivity of a Nation. November 1992. 72 pp.

196. **Tuula Melkas,** Valtion ja markkinoiden tuolla puolen. Kanssaihmisten apu Suomessa 1980-luvun lopulla. Joulukuu 1992. 150 s.

197. **Fjalar Finnäs,** Formation of unions and families in Finnish cohorts born 1938–67. April 1993. 58 pp.

198. **Antti Siikanen – Ari Tyrkkö,** Koti – Talous – Asuntomarkkinat. Kesäkuu 1993. 167 s.

199. **Timo Matala,** Asumisen tuki ja aravavuokralaiset. Kesäkuu 1993. 84 s.

200. **Arja Kinnunen,** Kuluttajahintaindeksi 1990=100. Menetelmät ja käytäntö. Elokuu 1993. 89 s.

201. **Matti Simpanen,** Aikuiskoulutus ja työelämä. Aikuiskoulutustutkimus 1990. Syyskuu 1993. 150 s.

202. **Martti Puohiniemi,** Suomalaisten arvot ja tulevaisuus. Lokakuu 1993. 100 s.

203. **Juha Kivinen – Ari Mäkinen,** Suomen elintarvike- ja metallituoteteollisuuden rakenteen, kannattavuuden ja suhdannevaihteluiden yhteys; ekonometrinen analyysi vuosilta 1974 – 1990. Marraskuu 1993. 92 s.

204. **Juha Nurmela,** Kotitalouksien energian kokonaiskulutus 1990. Marraskuu 1993. 108 s.

205a. **Georg Luther,** Suomen tilastotoimen historia vuoteen 1970. Joulukuu 1993. 382 s.

205b. **Georg Luther,** Statistikens historia i Finland till 1970. December 1993. 380 s.

206. **Riitta Harala – Eva Hänninen-Salmelin – Kaisa Kauppinen-Toropainen – Päivi Keinänen – Tuulikki Petäjäniemi – Sinikka Vanhala,** Naiset huipulla. Huhtikuu 1994. 64 s.

207. **Wangqiu Song,** Hedoninen regressioanalyysi kuluttajahintaindeksissä. Huhtikuu 1994. 100 s.

208. **Anne Koponen,** Työolot ja ammattillinen aikuiskoulutus 1990. Toukokuu 1994. 118 s.

209. **Fjalar Finnäs,** Language Shifts and Migration. May 1994. 37 pp.

210. **Erkki Pahkinen – Veijo Ritola,** Suhdannekäänne ja taloudelliset aikasarjat. Kesäkuu 1994. 200 s.

211. **Riitta Harala – Eva Hänninen-Salmelin – Kaisa Kauppinen-Toropainen – Päivi Keinänen – Tuulikki Petäjäniemi – Sinikka Vanhala,** Women at the Top. July 1994. 66 pp.

212. **Olavi Lehtoranta,** Teollisuuden tuottavuuskehityksen mittaminen toimialatasolla. Tammikuu 1995. 73 s.

213. **Kristiina Manderbacka,** Terveydentilan mittarit. Syyskuu 1995. 121 s.

214. **Andres Vikat,** Perheellistyminen Virossa ja Suomessa. Joulukuu 1995. 52 s.

215. **Mika Maliranta,** Suomen tehdasteollisuuden tuottavuus. Helmikuu 1996. 189 s.

216. **Juha Nurmela,** Kotitaloudet ja energia vuonna 2015. Huhtikuu 1996. 285 s.

217. **Rauno Sairinen,** Suomalaiset ja ympäristöpolitiikka. Elokuu 1996. 179 s.

218. **Johanna Moisander,** Attitudes and Ecologically Responsible Consumption. August 1996. 159 pp.

219. **Seppo Laaksonen** (ed.), International Perspectives on Nonresponse. Proceedings of the Sixth International Workshop on Household Survey Nonresponse. December 1996. 240 pp.

220. **Jukka Hoffrén,** Metsien ekologisen laadun mittaaminen. Elokuu 1996. 79 s.

221. **Jarmo Rusanen – Arvo Naukkarinen** – Alfred Colpaert – Toivo Muilu, Differences in the Spatial Structure of the Population Between Finland and Sweden in 1995 – a GIS viewpoint. March 1997. 46 pp.

222. **Anna-Maija Lehto,** Työolot tutkimuskohteena. Marraskuu 1996. 289 s.

223. **Seppo Laaksonen** (ed.), The Evolution of Firms and Industries. June 1997. 505 pp.

224. **Jukka Hoffrén,** Finnish Forest Resource Accounting and Ecological Sustainability. June 1997. 132 pp.

225. **Eero Tanskanen,** Suomalaiset ja ympäristö kansainvälisestä näkökulmasta. Elokuu 1997. 153 s.

226. **Jukka Hoffrén,** Talous hyvinvoinnin ja ympäristöhaittojen tuottajana –Suomen ekotehokkuuden mittaaminen. Toukokuu 1999. 154 s.

227. **Sirpa Kolehmainen,** Naisten ja miesten työt. Työmarkkinoiden segregoituminen Suomessa 1970–1990. Lokakuu 1999. 321 s.

228. **Seppo Paananen,** Suomalaisuuden armoilla. Ulkomaalaisten työnhakijoiden luokittelu. Lokakuu 1999. 152 s.

229. **Jukka Hoffrén,** Measuring the Eco-efficiency of the Finnish Economy. October 1999. 80 pp.

230. **Anna-Maija Lehto – Noora Järnefelt** (toim.), Jaksaen ja joustaen. Artikkeleita työolotutkimuksesta. Joulukuu 2000. 264 s.

231. **Kari Djerf,** Properties of some estimators under unit nonresponse. January 2001. 76 pp.

232. **Ismo Teikari,** Poisson mixture sampling in controlling the distribution of response burden in longitudinal and cross section business surveys. March 2001. 120 pp.

233. **Jukka Hoffrén,** Measuring the Eco-efficiency of Welfare Generation in a National Economy. The Case of Finland. November 2001. 199 pp.

234. **Pia Pulkkinen,** "Vähän enemmän arvoinen" Tutkimus tasa-arvokokemuksista työpaikoilla. Tammikuu 2002. 154 s.

235. **Noora Järnefelt – Anna-Maija Lehto,** Työhulluja vai hulluja töitä? Tutkimus kiirekokemuksista työpaikoilla. Huhtikuu 2002. 130 s.

236. **Markku Heiskanen,** Väkivalta, pelko, turvattomuus. Surveytutkimusten näkökulmia suomalaisten turvallisuuteen. Huhtikuu 2002. 323 s.

237. **Tuula Melkas,** Sosiaalisesta muodosta toiseen. Suomalaisten yksityiselämän sosiaalisuuden tarkastelua vuosilta 1986 ja 1994. Huhtikuu 2003. 195 s.

238. **Rune Höglund – Markus Jäntti – Gunnar Rosenqvist (eds.),** Statistics, econometrics and society: Essays in honour of Leif Nordberg. April 2003. 260 pp.

239. **Johanna Laiho – Tarja Nieminen (toim.),** Terveys 2000 -tutkimus. Aikuisväestön haastatteluaineiston tilastollinen laatu. Otanta-asetelma, tiedonkeruu, vastauskato ja estimointi- ja analyysiasetelma. Maaliskuu 2004. 95 s.

240. **Pauli Ollila,** A Theoretical Overview for Variance Estimation in Sampling Theory with Some New Techniques for Complex Estimators. September 2004. 151 pp.

The Research Reports series describes the Finnish society in the light of up-to-date research results. Scientific studies carried out at Statistics Finland or those based on the data sets of Statistics Finland are published in the series

When making judgements of the quality of survey sampling, especially with complex estimators, the method of variance estimation is of importance. Along with the linearisation approach, there are many methods based on sample reuse. The variety of these methods, especially in sampling theory, does not have any unified theoretical framework. The first part of this thesis is a theoretical overview of variance estimation, covering the foundations of the sampling theory and the current methodology of variance estimation. In addition, new methods and theoretical results are provided. The cumulants and k-statistics are utilised to study the theoretical correction coefficient of unbiased variance estimation. The post-design vectors are used for the scale adjustment. There are correction methods utilising two-phase resample spaces and alternatively two resample spaces. New sampling distribution results concerning without-replacement and with-replacement designs in two-phase sampling situations are presented. A variance decomposition approach utilising sample pair probabilities is given with variance estimators. Finally, the old and new methods are tested with two real-life data sets.

Tilastokeskus, myyntipalvelu
PL 4C
00022 TILASTOKESKUS
puh. (09) 1734 2011
faksi (09) 1734 2500
myynti@tilastokeskus.fi
www.tilastokeskus.fi

Statistikcentralen, försäljning
PB 4C
00022 STATISTIKCENTRALEN
tfn (09) 1734 2011
fax. (09) 1734 2500
myynti@stat.fi
www.stat.fi

Statistics Finland, Sales Services
P.O.Box 4C
FIN-00022 STATISTICS FINLAND
Tel. +358 9 1734 2011
Fax +358-9-1734 2500
myynti@stat.fi
www.stat.fi

ISSN 0355–2071
= Research
ISBN 952–467–319–3
Product number 89038
CD