DACSEIS

IST-2000-26057

Workpackage 4

Evaluation of Software for Variance Estimation in Complex Surveys

Deliverables 4.1 and 4.2

2004

List of contributors:

Seppo Laaksonen, Pauli Ollila, Kaja Sõstra; Statistics Finland
Yves Berger; University of Southampton
Harm-Jan Boonstra, Jan van den Brakel; Statistics Netherlands
Anthony Davison, Sylvain Sardy; EPFL;
Kersten Magg, Ralf Münnich, Dominik Ohly, University of Tübingen

Main responsibility:

Pauli Ollila; Statistics Finland

IST–2000–26057–DACSEIS

The DACSEIS research project is financially supported within the IST programme of the European Commission. Research activities take place in close collaboration with Eurostat.

`http://europa.eu.int/comm/eurostat/research/`

`http://www.cordis.lu/ist/`

`http://www.dacseis.de/`

Preface

One task of the DACSEIS project is to evaluate statistical software taking survey features into account. This document contains four main parts. *Criteria for Reviewing Software* includes detailed criteria for the evaluation partners on studying theoretical, computational, documentational and other aspects, *Introduction of Software* gives an overview of the different software currently available and briefly describe the features of them. *Descriptions of Software* includes more detailed information about eleven software and their theoretical and technical properties. *Evaluation of Software* includes studies about the theoretical and the practical solutions and use of eight software in different survey situations. Furthermore, the software are tested with the data sets from three DACSEIS universes representing European surveys in order to collect experiences of the software (e.g. speed, memory, accuracy, usability, difficulties, limitations).

Pauli Ollila Statistics Finland

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The quality of survey data is considered being increasingly important today when surveys are more and more international, and because surveys designs are changing especially due to new data collection tools. Overall, surveys may be affected by worsening accuracy problems, also due to new error sources. Various complex methods for bias reduction and efficiency improvement are used in the research institutes. The statistical systems at the pan-European level require harmonisation and comparability for the socio-economic data of different National Statistical Institutes. The importance of this issue can be seen in the activity of Eurostat in recent years. Task Force on Variance Estimation has compiled some reports about the variance estimation practices of national statistical institutes, including information on the used software. The EU-financed SUPCOM project Model Quality Report in Business Statistics (Davies and Smith, 1999) black is another example of that activity.

In the field of variance estimation, the research work conducted within the DACSEIS project has impact by describing the current situation on different branches of survey methodology, developing new methods for these purposes and testing the old and new methods with data universes based on several national surveys. In this context, the statistical software taking survey features into account are of great interest. Workpackage 4 of the DACSEIS project was dedicated to the evaluation of this kind of software, taking into account especially the European practices in statistics making in National Statistical Institutes. The evaluation should provide an overview on the current situation of the software with studies on theoretical and practical aspects available. One vital reason of this evaluation is to give the end-user sufficient information to make a decision regarding his or her own needs.

It is obvious that all error sources cannot be studied when the software evaluation is concerned, e.g. non-sampling errors as measurement errors, editing errors and frame errors, although these may be considerable in many surveys. Thus the quality issues here concentrate on sampling error, variance due to missingness when adjustments by reweighting have been done and variance after imputation. Furthermore, there should be a clear structure what the aspects to be evaluated are and how the evaluation is conducted. Finally, the experiences when using different data sets based on real surveys would provide valuable information of the software. In this context the testing process should serve the research work conducted in other DACSEIS workpackages.

The structure of the deliverable contains four main parts: *criteria for reviewing software, introduction of software, descriptions of software and comparative software evaluation.* The *criteria part* defines what aspects are studied and what limitations are set in order to keep the evaluation task feasible. There are detailed advice for studying theoretical, computational, documentational and other aspects. The standard form as a tool for describing the properties of the software is introduced. The criteria part was written before the evaluation process, and originally it should have appeared as a separate deliverable. However, for better readability and technical reasons it was decided to be joined together with the second deliverable containing the actual software evaluation.

Introduction of Software gives an overview of the different software currently available and briefly describe the features of them. It serves as a tool for the reader to acquaint oneself to different software. *Description of Software* includes more detailed information about eleven software. Every software has a description on different aspects of software: *theoretical aspects, computational aspects, documentation* and *other aspects*. Furthermore, there are tables of the existing and non-existing theoretical properties, technical details and contact information for every software.

The *evaluation part* contains comparisons of different theoretical and computational aspects eight different software (SAS, SPSS, Stata, Bascula, Clan, Genesees, Poulpe, Sudaan, WesVar). R and S-Plus software weren't actually designed for any survey problems, and here they were excluded. At the beginning of the DACSEIS project the predecessor of Genesees, i.e. GSSE, was not available for purchase in general, so it was excluded as well. At first, the general evaluation part includes an overview on the theoretical features included in the software with some discussion about the applicability of these methods. The special requirements for survey information provided by the user are considered to be important, e.g. the adaptation of design and auxiliary information into the software system. Some mistakes appearing occasionally in survey practices (e.g. unusual weights, one unit in a stratum, wrong information) are studied with a test pattern. The interface and output examples with comments illustrate the usability of the software.

Another part of the evaluation contains testing with the three sample data sets selected from the DACSEIS Universes (see DACSEIS deliverables D3.1 and D3.2) representing Finnish Labour Force Survey, Swiss Household Budget Survey and German Microcensus. The estimation tasks conducted in the evaluation are in harmony with the simulation tasks in other workpackages of DACSEIS. There are tables and descriptions for the estimates and variance estimates of the Horvitz-Thompson and calibration/GREG estimators at the predefined levels of study. Also memory allocation as well as speed are studied. These descriptions (especially for Finnish Labour Force Survey) include many specific computational and practical aspects of the software. Furthermore, there is a simple speed test for R and SAS-based software in order to compare the potential of R with some existing applications in SAS. Finally, a general overview of the imputation procedure in SAS, Proc MI, is given including some tests with the Swiss Household Budget Survey data.

Chapter 2

Criteria for Reviewing Software

2.1 Main Principles

This chapter covers the criteria for the evaluation of the software packages available for variance estimation. These aspects (in practice stated at the first phase of the project) were followed during the evaluation process in workpackage 4 as closely as possible. On the other hand, these criteria as such were also evaluated during the testing process and, consequently, some revision was conducted especially when the methodology included in workpackage 1 was concerned. Note that in practice these criteria were available for the evaluation partners of the DACSEIS project at the first phase of the project. *These criteria were followed when both the general evaluation of software and the particular testing with the DACSEIS data sets was conducted. The DACSEIS standard form (see Section 2.2.4) was a tool for evaluating the properties of the software. However, some other properties outside the structure of the standard form were described as well.*

Users of the evaluation results. As the principal aim, the DACSEIS project concentrates on providing a European quality management system for socio-economic data in the national statistical sectors drawn from complex survey sampling. From that point of view it is important that *the needs of statisticians from these sectors are carefully taken into account in the evaluation of the software.* A statistician, who is responsible for providing results from a survey and who has some knowledge of the methodology, is defined here as an *ordinary user* of the software. On the other hand, some features in some software are designed especially for *a sophisticated user* with a longer experience of both the software and the methodology. These features must be considered to some extent as well.

Evaluators. It is considered important to have several testing partners within the DACSEIS project in order to have an extensive and balanced view on the software. The principle of having at least two testing partners for one software necessitates that as well. The evaluation organisations are *Statistics Finland* (main evaluator), *Statistics Netherlands, Swiss Federal Institute of Technology, University of Southampton* and *University of Tübingen.* The main persons carrying out the evaluation in these institutes should be users with good knowledge of the topic, but experiences and comments from ordinary users will be aggregated as well.

General evaluation and testing (examining the software as it is). This part includes the study of the properties available in the software as well as testing those properties with various data sets (local, DACSEIS, artificial) ranging from simple to complex. The *manuals, help menus* as well as *documents and previous studies of the topic* (see Chapter 3) will be exploited extensively for this work. The basic study is carried out in the framework given in Section 2.2. For helping the study of the aspects the *standard form* (see Section 2.2.4) was developed. The nature of this part is descriptive; together with the standard form information there will be memos in a more written style. See Section 2.3 for more details.

Specific testing with the DACSEIS data files. Considering the principal aim of the project, this is the most important task of testing. The sample files are selected from the pseudo-universe data sets (representing different European surveys) created under workpackage 3. Finally they are uploaded onto the DACSEIS website in order to be downloaded by the testing partners. *The files should represent some variety of different properties appearing in surveys in order to attain a good scope of the situations usually met by the ordinary user.* Furthermore, the software testing operations to be carried out with the DACSEIS sample files needs to be in harmony with the methodology used in the DACSEIS simulations based on workpackage 1. Thus the range of testing aspects is much smaller than in general testing. The numerical comparisons are conducted via this part of testing. For every software there will be a *process description* about the operations carried out during testing. This should help the reader to understand the nature of the software together with specific requirements set for the user by the software. See Section 2.4 for the descriptions of the sample data sets and their properties together with a detailed testing plan.

2.2 Aspects to be taken into Account

2.2.1 Theoretical Aspects

Quality components to be included in the evaluation. There are several quality components in surveys, and only some of them can actually be covered in this study of software. The most obvious aspect of this evaluation is the accuracy of the procedures and results. However, these tests cannot include explicitly such non-sampling errors as measurement errors and editing errors, although these may be considerable in many surveys. Frame errors, especially undercoverage, will also be excluded. As in workpackage 1 with the methodology for the simulation tests, the quality issues here focus on *sampling error, variance due to missingness when adjustments by reweighting have been done* and *variance after imputation.* For the final tests it is important that these properties can be estimated by using the features included in the software, i.e. *they are available for the user without any specific programming/processing in the environment where the software is executing* (e.g. in SAS). The programming work for these methods as such will be carried out elsewhere in some other workpackages.

Types of variance estimates. Generally the variance estimation in survey software is conducted with *analytical (direct)* and *approximate solutions* based on the calculations at the unit level. In recent years various *resampling methods* have also been introduced for different variance estimation purposes in some software.

Types of parameters. The choice of different parameters to be estimated usually vary from the *fixed-definition style* to the *flexible definition of functions*. In practice, not all parameters are available for estimation in all software.

Types of estimators for which variance estimates may be computed. It is essential that *estimation methods utilising auxiliary information* are also included in testing processes. The most common practices from this branch are *ratio* and *regression estimation*, *post-stratification* and *calibration*.

Types of sampling designs supported. A major issue in survey software evaluation is *applicability to different* (and sometimes quite complex) *sampling designs*. Some examples are *pre-stratification, PPS sampling, cluster sampling, multistage sampling, with and without replacement sampling, sampling in two or more phases, rotating schemes*.

Other features. Apart from these aspects there are some other theoretical features that will be studied, such as detection and treatment of outliers, small area / domain solutions.

2.2.2 Computational Aspects

Technical properties. Descriptions of the technical solutions, concerning such as interface, platform(s) of software, import/export of files (including input of auxiliary data), types of input and output files (content and understandability of metadata [1]) and scalability[2] (See: `http://search390.techtarget.com/sDefinition/0,,sid10{\_}gci212940,00.html`) are considered to be important. The capacity and efficiency issues (speed and memory requirements) are of special interest.

Ease of practical operations concerning the research process. These operations include data preparation for weights and auxiliary information as well as software processing for some theoretical operations required for the survey. Furthermore, diagnostics (graphics, tables to help users), checks (whether the program warns when something is wrong), built-in routines (regression, bootstrapping, imputation) appear more or less in different software.

User impressions. Carrying out different theoretical and practical operations concerning the survey gives experiences that provide the basis for this subjective evaluation. User-friendliness, ease of programming, problems with algorithms and bugs appearing in the software are examples of user impressions.

[1] Metadata are data that define and describe other data.

[2] In information technology, scalability refers to two features: 1) The ability of a computer application or product (hardware or software) to continue to function well as it (or its context) is changed in size or volume in order to meet a user need, and 2) The ability of a computer to not only function well in the rescaled situation, but to take actually full advantage of it.

2.2.3 Documentation and Other Aspects

The documentation concerning the software is an aspect important for users from the beginner to the specialist. The manual and its quality, special papers existing, specification of metadata, help menus and the website are aspects to be taken into account together with evaluation of the examples in these tools. Other topics are such as how general vs. country/institute-specific the software is, how well the maintenance is guaranteed, how the software will be developed further, the price and the hotline (help desk) available.

2.2.4 Standard Form for Collecting Basic Information of the Aspects

At the early phase of the project a *standard form* was developed in order to give a framework for collecting basic information of the aspects defined in this chapter. Every evaluation partner received this form for the subsequent evaluation work of their software to be tested. As a preliminary operation the main idea of this form was to check the existence of the theoretical aspects and to describe the other aspects in a more written style. Note that the categorisation refers to the themes presented in 2.2.1, 2.2.2 and 2.2.3. These sections served as guidelines to the descriptions made by the evaluators. However, the reports of the evaluation partners contained much more than only the basic information in the standard form, providing valuable information for the more detailed study of the software.

Theoretical aspects:

Types of variance estimates: *direct calculation, Taylor linearisation, jackknife, bootstrap, balanced repeated replication method, other.*

Types of parameters:*total, mean, ratio, percentages, quantiles, confidence intervals, other.*

Types of estimators for which variance estimates may be computed: *ratio estimation, post-stratification, calibration, other.*

Types of sampling designs supported: *stratification, PPS sampling, cluster sampling, multistage sampling, with / without replacement sampling, rotating schemes, other.*

Other features: *detection and treatment of outliers, small area / domain solutions, other.*

Computational aspects: *user friendliness, interface, platforms of software, speed, importing and exporting files, types of input and output files, diagnostics, checks, quality of simulation algorithm, ease of programming, memory requirements, problems in algorithms - bugs, built-in routines (regression, bootstrapping, imputation), other problems met.*

Documentation and other aspects: *manual and its quality, special papers, specification of metadata, help menus, website, how informative and correct the possible examples are, available in the above-mentioned tools, other documentation aspect, how general vs. country/institute-specific the software is, how well its maintenance is guaranteed, how the software may be further developed, price, hotline (help desk), other aspects.*

2.3 Instructions for General Evaluation and Testing

2.3.1 Some Practical Points

Section 2.3 provides instructions for general evaluation and testing with data in practice. However, the main theoretical, methodological and computational comparisons between the software are conducted with the DACSEIS sample data sets, and for that purpose a detailed plan is defined (see Section 2.4).

Working order. Small-scale tests with some simple test data (provided by the software or from other projects) are conducted before embarking on tests with more complex data from specific national/international projects and with the universe files of DACSEIS. Especially with unfamiliar software we see it is essential to adapt first "to the system" by using the most ordinary methods and practices. However, the DACSEIS data sets are the most important where the actual results are concerned (see Section 2.4). The emphasis in general testing is on the theoretical and computational aspects.

Strategy for reporting. Many points in the evaluation are qualitative and partially subjective, as well as dependent on the experience with software and variance estimation. As a strategy it is suggested that the tester should write down running comments during the testing even if he or she is not definitely sure whether a certain point is relevant. It is better to make too many and contradictory than too few and logical comments. Later, these comments may be reviewed and discussed in the evaluation group. Descriptions from general testing together with the information from the standard form (see Section 2.2.4) make the basis for the software evaluation report, but as stated before, the main comparisons are conducted with the DACSEIS data sets.

2.3.2 Studying Theoretical Aspects

At the early stage of the evaluation, the standard form (see Section 2.2.4) sets the framework for collecting basic information about the theoretical aspects, for example (see Section 2.2.1). This section describes how to study these aspects in a deeper way. At the first stage it may be wise not to conduct data testing with every theoretical feature of the software in detail. Nevertheless, some basic theoretical requirements should be fulfilled. Furthermore, some methods in the software may not be necessary but may prove beneficial.

Testing sampling designs with data. The sampling designs to be tested vary from simple to complex, including stratification, clusters, different stages, unequal probabilities, perhaps different phases and rotation. The designs behind the DACSEIS data sets provide some complexity, but otherwise some additional data are required. The participants might have some national data or artificial data for that purpose. One should pay attention to the fact how sufficiently theoretically those different (complex) sampling designs are dealt with in the software.

Adaptation of design information into the system. *How is the design information transferred into the software, and is this transfer theoretically sufficient:* role of the weights, nature of the design: with/without replacement, draw/list sequential selection procedures, design identification, i.e. strata, clusters, phases, size variables. This point (together with other points) is later studied in more detail in the DACSEIS data set testing with the process descriptions (see Section 2.4) to be made for every software in order to help the reader to understand the nature of the software together with specific requirements set for the user by the software.

Parameter and estimator study. There are two main alternatives for parameter and estimator definitions, i.e. *a fixed set of parameters and specific estimators* for which the estimates and the variance estimates are calculated, or a situation where *the form of the estimator can be constructed fairly freely.*

Availability of variance estimation methods using auxiliary information in the estimation phase. Some estimation methods will not be solved simply by using certain forms of construction. One may have an additional requirement for calibration and post-stratification. The use of auxiliary information (e.g. ratio and regression estimators, post-stratification, calibration) in estimation is an essential feature in survey sampling. However, it is not common for variance estimation software to have options for taking these methods into account. If such an option is available, a small test with appropriate data is conducted.

Basic variance estimation methods. As known, the variance estimation principles for nonlinear estimators can be divided into two categories: mathematical approximation of the variance and reusing the sample. At this stage we concentrate on nonlinear parameters and their estimators; nonlinear estimators modified with auxiliary information are dealt with later. The techniques used in each software package will be studied to some extent.

2.3.3 "Clean Data" Assumption vs. Problems in Data and Estimation

"Clean data" assumption. For the first tests to be conducted with different data sets one should omit all problems caused by the data: no coverage problems, the data are considered as the sample without unit non-response, no imputation made, only variables with full response included, no outliers and erroneous values assumed, no small subgroups for results. At least in this situation the software should produce right results according to the theory. The purpose of this testing process is not to check whether every result is calculated correctly. Nevertheless, some checks must be made and it is reasonable to adopt this "clean data" assumption at first. The point estimates in all results should be the same as when produced with standard software with the same weighting. Variance estimates under different designs can be checked at least in the case of linear estimators, and one could conduct parallel program runs with all the software in question with the same data. Results within strata and within some other sub-populations should be checked as well. This phase of testing need not be long.

Unit non-response. Practically all survey data contain some unit non-response. Usually, the researchers follow the *ignorability principle*, that is, they use the respondents' data as if it were obtained from the original sample selected under the chosen sampling design. Consequently, they ignore the terms due to unit non-response in variance estimation. However, if this aspect has been taken into account in a certain software package, the technique should be tested. The use of some adjustments (e.g. post-stratification, calibration) is strictly connected to their testing (see previous sections), the others may have different practices. If a software package provides more or less correct variance estimates under response propensity modelling, for example, this should be tested and documented.

Imputation and problems occurring in other operations/"non-clean" situations. Excluding unit non-response, these "nuisances" are considered at this stage. If any options are available for these problems in the software we are going to evaluate they can be tested with data (real or artificial) containing the problem.

2.3.4 Studying Computational, Informational and Stylistic Aspects

The first tests introduce us to the use of the software and the tests give us some impressions on how the following previously presented ordinary operations can be done: *principles of operations* (programming; menu or screen driven), *data processing, solutions for weights, designs, estimators* and *auxiliary information, some simplest options for results*. In general, some hints of user-friendliness and easiness of the software are already available. The style of output may be more or less appropriate for different purposes. Furthermore, the user may need some information about the processes while conducting his or her work. What is the quality of this information?

The *speed of the software* is not an easy task to be evaluated, mainly due to different computers, platforms and interface in question. Some software give information about the computation time for different processes, some do not. One possibility is to write down CPU (or real) time results if available, and for other cases just to roughly measure or categorise the amount of time used for computation. In the end some comparison results in a same (rough) scale should be produced. The *memory requirements* are an easier task, at least where the possible auxiliary data, working data and output data are concerned.

2.4 Detailed Plan for Testing with the DACSEIS Data Files

2.4.1 Principles and General Descriptions

Main ideas. Choosing three European surveys as the basis of testing emphasises the real-life situation familiar to many ordinary users of the software. The samples to be tested are selected from the DACSEIS universes created in workpackage 3 for simulations to be carried out for testing the methodology appearing in workpackage 1. These universes are based on the real survey data provided by the partners in the project (see deliverable D3.1 and D3.2 *"Monte-Carlo Simulation Study of European Surveys"* for details). The creation of these universes also includes different non-response models from which one or two were selected for testing purposes. The nature of the testing process with the DACSEIS data files differs from general evaluation and testing by having a *detailed, restricted plan* to be followed by the partners. The *main comparisons* of the theoretical and computational results are carried out in this context.

Harmonisation with workpackage 1. Deliverable D1.1 (*Data Quality in Complex Surveys*) gives an overview to the methodology on the measurement of accuracy on estimates from complex surveys. In the project it was decided that these methodological results should have a vital role in specified data testing of the methodology available in the survey software as well. Thus, the harmonisation of workpackage 1 and workpackage 4 was carried out after the finalisation of the first part of workpackage 1. In practice, the persons responsible for the forthcoming simulations in workpackage 1 and test processing in workpackage 4 decided in detail the surveys to be tested, the information included in the data, and the survey-based and methodological properties and aspects to be taken into account. These decisions were transferred into a task list, which should be followed by the partners during testing.

Comparison of results. The comparisons based on testing between different software include *numerical results* (estimates and variance estimates) concerning the survey and the operations conducted in testing, *computational results* including speed and memory resources, and finally the *ease of conducting the estimation process* in the software (the pros and cons of operations needed in different phases of the process).

Process descriptions. For every software there will be a *process description* (following a framework created before testing, see Section 2.4.2) about the operations carried out during testing. This should help the reader to understand the nature of the software together with specific requirements set for the user by the software. Furthermore, it is a tool for a general comparison between the software in this testing section. The principles of the process description are presented in Section 2.4.2.

General experiences and special situations. In addition to the results and the process descriptions, an overview on the general experiences as well as some special (unexpected) situations during testing is expected from the partners.

2.4.2 Process Description

Considering a descriptive survey, the final outcome will usually be tables with estimates and standard errors / coefficients of variation / confidence intervals of the estimators classified to the levels of study chosen by the researcher. Before this is obtained, we carry out several theoretical operations to follow the methodology decided to be used and the requirements set to the study. *The way in which these operations occur in the flow of processing differ from one software to another.* For each software in testing there will be a *process description* about the operations carried out during testing. This should help the reader to understand the nature of the software together with specific requirements set for the user by the software. Based on preliminary knowledge about the software, a framework is developed in order to help this description work. However, when compared with the framework, if some major differences occur in the structure of the process of the software, the tester can naturally adjust the framework to the new situation. The speed (CPU, real or approximate time) and memory results (kilobytes/megabytes for data appearing in the processing) are listed at these different phases of processing. Note that not all of these actions appear in every software.

External calculations of additional variables. Some software require variables which may not be in the original data (identification variables, variables for the design, terms for weight calculation, replicate weights [for some resampling procedures], dummy variables in some situations). If calculation of additional variables is not possible inside the tested software (i.e. Bascula) these calculations should be done using some other more general software (SAS, SPSS) before importing survey data into the system.

Metadata preparation. In some cases pre-definitions of the variables are required in the system in which the software works (e.g. Bascula works in the Blaise system. The Blaise data model must be prepared for reading survey data into Blaise).

Importing survey data into the system. There are different ways of importing the survey data into the system, provided that the data are not made in the system.

Selecting and/or defining variables and weights. Depending on the software, the variables in question are either selected via menus or defined in the software or its backing system.

Importing/processing population information. Information from the population level (e.g. stratum/cluster frequencies, marginals of auxiliary information) is vital for design-based variance calculations as well as for the case of utilising auxiliary information in estimation and variance estimation. The requirements of the software vary to some extent.

Specifying weighting model / estimation function and variance estimation method. The weighting model or estimation function must be defined in some software (possibly with auxiliary information). Optional variance estimation methods can be chosen in some software.

Computing weights / replicate weights. The phase when the weights are computed (if not already existing) varies from one software to another. Some systems create the weights internally, either automatically (based on the design information available) or with the request of the user.

Computing estimates and variance estimates. Before actual computing the user must indicate the software, the parameters (in some cases the estimation functions), the estimator(s) to be used and the levels of study (categorisation). Furthermore, the software require more or less the following: identification of the weight, some information about the design and the population (stratum, cluster, unit identification, frequencies at different levels of sampling, marginals of auxiliary information).

Chapter 3

Introduction of Software

3.1 Background

When choosing the software to be evaluated, it was decided that the needs and practices of European statistical offices were of special concern, especially the branch of social statistics. This decision was in harmony with the general principles of the DACSEIS project. At the beginning of the project a questionnaire was sent to European statistical offices, collecting vital information on household and individual surveys, their basic structures, sampling designs and schemes, estimation methods, variance estimation methods and the software that were in use in those surveys. Another important source of information about the variance estimation practices at the European level was the Eurostat-financed SUPCOM project Model Quality Report in Business Statistics (Davies and Smith, 1999). In all, the decision was based on the frequency of the software among the European statistical offices, the suitability of the software in the context of the European social surveys and their practices, the special features available in the software for more advanced estimation methods and whether the software provides an environment for further methodological development. The primary focus was decided to be *on the needs of an ordinary user*, who can be a statistician of a statistics department responsible for providing results from a survey. Regarding multiple imputation there has been increasing development work on both estimation and variance estimation, including software for that purpose. However, it was decided that those software were not in the evaluation process.

A significant source of information for software evaluation in DACSEIS was the previous evaluation work carried out in recent years. At the European level a substantial force of this branch is Eurostat with its projects concerning quality issues. The *Task Force on Variance Estimation* has compiled some reports about the variance estimation practices of national statistical institutes, including information on the used software, such as the report *Variance Estimation Methods in the European Union* (Eurostat, 2003), progress reports for the Task Force and for the Structural Business Statistics. The second volume of the SUPCOM project report, *Comparison of Variance Estimation Software and Methods* (Smith *et al*, 1998), provided a good basis for further evaluation work. Another report, *Strengths and Limitations of Using Sudaan, Stata and WesVarPC for Computing Variances from NCES Data Sets* (Broeme and Rust, 2000) included software comparison with calculations based on the data sets of the U.S. National Education Longitudinal Survey. Recent papers dealing with variance estimation using survey software are such as CARLSON (1998), BROGAN (1998), HORTEN and LIPSITZ (2001) and TOMPKINS and SILLER (2001). Considering statistical journals, both the *Journal of Official Statistics* and *Survey Statistician* have provided a possibility for software reviews, see e.g. Survey Statistician 43, 44, 46, 47 and 48. Since 2001 Statistics Canada has published the *Imputation Bulletin* twice a year with some software evaluation as well. When the Internet is concerned, a substantial source of information of survey software is the website maintained by Survey Research Methods Section, i.e. `http://fas-www.harvard.edu/$\sim$stats/survey-soft`. Another site (by UCLA Academic Technology Services) dealing with the practical side of the topic is `http://www.ats.ucla.edu/stat/seminars/svy{\_}intro/default.htm`.

3.2 General Software (SAS, S-Plus, R, SPSS, Stata)

The software chosen for evaluation were divided into two categories: *general software* and *specialised software*. The general software were defined to be developed for broader use than only for survey sampling problems (which might be dealt with in a specific part of the software). *SAS*, *Stata* and *SPSS* currently have calculation features which take the sampling design into account. In addition, *S-Plus* and *R* give an environment for an advanced methodologist to make corresponding calculations by himself/herself; however, these two software were not included in the detailed evaluation with calculations using DACSEIS sample data files, because they are not suitable as such (without programming or other construction efforts) for an ordinary user.

SAS is a very versatile commercial package for all kinds of needs connected to data in an electronic form. There are both menu-based and programmable ways of working. Programs can be built with a large variety of procedures and operations for the data. Furthermore, macro language, application building system and matrix language can be applied for very specific purposes. There are many analytical and graphical tools either in a procedure or menu form; the latter alternative is a better choice for less experienced users. There are several (in some cases user or organisation-made) programs and macros for different purposes.

SAS includes fixed procedures for sample selection and survey analysis (descriptive statistics and frequency tables with corresponding standard errors and confidence intervals; regression model and logistic model tools taking the design into account). There are many designs for sampling, but only basic designs (stratification, pps sampling, cluster sampling, two-phase sampling with primary sampling unit variation) for calculations. The SAS survey procedures *do not include* any estimation methods utilising auxiliary information (e.g. GREG, calibration, non-response adjustments). SAS is using the Taylor approximation for variance estimation of non-linear estimators.

S-Plus and **R** are languages and environments for statistical computing and graphics, based on the S language. The term “environment” is intended to characterize R or S-Plus as fully planned and coherent systems, rather than incremental accretions of very specific and inflexible tools, as sometimes occurs with other data analysis software. They are integrated suites of software facilities for data manipulation, calculation and graphical display. R is free of charge. S-Plus is commercial software with menus and dialog boxes providing a framework for easier programming and processing.

S-Plus and R do not provide any in-built survey tools, but the user communities provide some solutions for survey problems as well, e.g. the R 'survey' package for some simple survey problems and packages for resampling methods.

SPSS (Superior Performing Statistical Software) is a modular, tightly integrated, commercial product family including procedures with a variety of statistical methodology and data processing properties. It is a menu-controlled system with easy and quick analysis facilities, having separate windows for different purposes (e.g. data, variable properties, analysis, programming). Analysis routines can be saved or programmed with a specific language (Sax Basic). Special tools for different purposes can be added to the software, e.g. data mining, database analysis and market research, as well as missing value analysis and complex samples.

SPSS Release 12 has the module "SPSS Complex Samples", which includes four procedures offering the possibility to draw and to analyse a stratified, pps, clustered or multistage sample (with primary sampling unit variation). The calculations are concentrated on some simple descriptive statistics and cross-tabulations with standard errors, design effects, confidence intervals and hypothesis tests. SPSS as such *do not include* any estimation methods utilising auxiliary information (e.g. GREG, calibration, non-response adjustments). However, there is a calibration tool *g-Calib* provided by Statistics Belgium for more advanced estimation. SPSS is using the Taylor approximation for variance estimation of non-linear estimators.

Stata is a general software package for a broad scope of standard statistical analyses, from simple descriptive operations to linear regression, logistic regression, Poisson regression, estimation of probit models, test of independence of two-way tables, estimation and testing hypotheses of linear combinations of parameters. This commercial package is widely used in the statistical community. It is mainly command-driven with a few menus. Data interaction is conducted with text data or Stata data files. There are a variety of graphical and analytical tools.

Stata offers the possibility to account for stratified, pps, cluster or multistage (with primary sampling unit variation) sampling design features for estimation and variance estimation of some simple descriptive statistics and tabulations. Furthermore, for modelling and testing there is a family of survey commands taking the design features into account. The Stata survey commands *do not include* any estimation methods utilising auxiliary information (e.g. GREG, calibration, non-response adjustments). Stata is using the Taylor approximation for variance estimation of non-linear estimators.

3.3 Specialised Software (Bascula, Clan, Genesees, Sudaan, Poulpe, WesVar)

The *specialised software* were defined as concentrating primarily on survey estimation with specific variance estimation abilities. *Bascula, Clan, Poulpe, Sudaan* and *WesVar* were chosen to that group. In addition, it was then decided that *GSSE* would be described at the general level without a detailed evaluation including calculations, albeit quite recently its name has changed to *Genesees*, having many new features with a possibility for a free acquisition of the product from Statistics Italy (with registration).

Bascula is a software package for calculating weights for all units in a sample using auxiliary information. The interface is menu-based, and it is part of the Blaise System. Blaise is an integrated system for survey processing, focusing on data entry/editing, (telephone) interviewing, electronic data capture, management of surveys, manipulation of data and metadata.

The methods for utilising auxiliary information in weighting and variance estimation include post-stratification, linear weighting for GREG estimation and multiplicative weighting for calibration.Bascula can use the computed weights to estimate population totals, means and ratios as well as standard errors based on Taylor linearisation and/or balanced repeated replication (BRR). For the purpose of variance estimation stratification, pps sampling, cluster sampling and multi-phase sampling (with primary sampling unit variation) are supported.

Clan is a non-commercial SAS-based macro package, originally developed for different variance estimation problems at Statistics Sweden. The operations are carried out with a few different macro expressions in SAS programming. These macros can be run among the normal SAS programs, when the library of the core macros of Clan is defined.

Clan is a very versatile tool for both complex designs and various methods concerning estimation and non-response correction methods. A variety of sampling designs can be dealt with in the program: stratification, pps sampling (also Pareto πps), cluster sampling, multi-stage sampling (with primary sampling unit variation), two-phase sampling, rotating scheme, network sampling and subsampling of non-respondents. The estimation methods include GREG estimation, calibration (with a possibility for weight construction), calibration for non-response, response homogeneity groups and partially overlapping domain totals. All kinds of functions of totals can be defined as the parameters to be estimated. The variance for the estimators of these parameters is estimated with the Taylor linearisation method.

Genesees is non-commercial software for variance estimation, developed in the Italian National Institute of Statistics (ISTAT) originally for the needs of business and household surveys. It can be considered as an update of the software GSSE with some enhancements concentrating especially on the interface. It works in the SAS environment, and the interface is based on screens for selecting input variables and parameters and some other definitions. Although the interface is in Italian, the expressions in the interface are explained in detail in the Users' guide.

Genesees is based on the theory of calibration estimation utilising the possibility to estimate the variance with the Taylor linearisation for the GREG estimators. Furthermore, Genesees deals with stratified multistage sampling (with primary sampling unit variation) and pps sampling. The parameters to be estimated are limited to totals, means, proportions and ratios. Other auxiliary information methods than calibration/GREG are not available in Genesees.

Poulpe is a non-commercial software for variance estimation, originally created for the needs of the French Household Survey and the French Business Survey (the main developer Jean-Claude Deville). The software, interacting via screens, utilises the SAS environment, based on several macros created for calculation purposes. On the other hand, the user can calculate with macros straight in the program flow without screens. The interface and most of the documentation is in French, which may be a limitation for some user groups.

Poulpe can deal with a large variety of sampling designs, estimation methods and non-response adjustments within a unified theoretical structure. Stratification, cluster sampling, multi-stage sampling (with accurate variance estimation at each stage), unequal probability sampling (with and without replacement), systematic sampling, rotating schemes and multi-phase sampling can be dealt with Poulpe. There are advanced estimation methods available for variance estimation: calibration, ratio estimator, post-stratification. The unit non-response can also be treated in Poulpe by considering an additional phase (Poisson sampling) in the sampling design. The flexible 'function of totals' technique enables the definitions of rather complex parameters as well. The variance estimation is based on the Taylor linearisation of these functions of totals.

Sudaan, a commercial package for statistical analysis of correlated data, is either an SAS-based module or a separate system. It is created and further developed by RTI. The main importance is on the analysis tools; the variance estimation practices come along with these aims. There are many procedures for different analytical needs (linear models, logistic regression, log-linear regression, modelling categorical outcomes, proportional hazards modelling for failure time outcomes), all taking the sampling design into account. A procedure-driven programming language looks familiar to the SAS users.

The sampling designs available for calculations include stratification, cluster sampling, pps sampling (with and without replacement), multi-stage sampling (with both primary and secondary sampling unit variation included). The variance estimation methods include both the Taylor linearisation as well as sample reuse methods (Balanced Repeated Replications, Delete-1 Jackknife and Replicate Weight Jackknife). Excluding post-stratification, the methods utilising auxiliary information at the estimation phase are not available in the procedures. There are four different alternatives of the design effect to be chosen. Standardisation and contrasting subgroup estimators are specific features available in descriptive procedures. Sudaan calculates also the standard error for the median and other quartiles.

WesVar is a commercial software package for computing estimates and replicate estimates that properly reflect complex sampling and estimation procedures. This menu-driven system provides a user-friendly tool for various variance estimation tasks.

WesVar is flexible and can be used with complex sampling designs including multistage, stratified and unequal probability samples. The replicate variance estimates can reflect many types of estimation schemes, such as non-response adjustment, post-stratification, raking and ratio estimation. WesVar can calculate means and proportions with their variance estimates. It is also easy to use WesVar to compute variance estimates for complex functions of estimates, including ratios, differences of ratios and log-odds ratios. WesVar calculates standard errors and confidence intervals for the estimates specified by user and calculates chi-square tests of independence for two-way tables of weighted frequencies. WesVar computes estimated coefficients for linear and logistic regression models and tests the significance of subsets of linear combinations of parameter estimates.

3.4 A Brief Overview of Other Software

In addition to those eleven software described in Sections 3.2 and 3.3, there are several other software for variance estimation in different survey situations.

AM Software is a statistical software package for analysing data from complex samples, especially large-scale assessments, developed by the American Institutes for Research. It is a menu-based system working in Windows, including Taylor linearisation for all and jackknife, bootstrap and balanced repeated replications for some analysis. The software can be downloaded for free from the website of the Institute: `http://am.air.org/`.

CENVAR is a screen-oriented, menu-driven software package for the calculation of reliability measures for sample designs ranging from simple random samples of elements to more complex stratified, multistage cluster designs, developed by the US Bureau of Census. It is free software (based on the PC CARP software), downloadable from `http://www.census.gov/ipc/www/imps/download.htm`. For more information on CENVAR, see `http://www.census.gov/ipc/www/imps/cv.htm`.

Epi-Infois Windows-based, both menu and command-driven software designed for the global community of public health practitioners and researchers, developed by the Centers for Disease Control and the World Health Organization. When using its GSAMPLE module, the user can calculate for data with stratified sampling, with or without clustering; multistage samples; unequal-probability (e.g. pps) samples; however, samples are considered to be selected with replacement. For more information, see `http://www.cdc.gov/epiinfo/about.htm`. The software can be downloaded for free from `http://www.cdc.gov/epiinfo/EIvendor.htm`.

PC CARP is software for calculating estimates and standard errors in complex surveys, developed at the Center for Survey Statistics & Methodology at the Iowa State University. The software is a standalone program with text menus and it functions in a Windows/DOS PC. It deals only with data in ASCII form. More information can be obtained from `http://cssm.iastate.edu/software/pccarp.html`. In principle, PC CARP is **not** free software, but for further details of the current policy contact the organisation.

Generalized Estimation System (GES) is an SAS-based application utilising the screen features of SAS/AF, developed originally for the needs of Statistics Canada. It deals with various complex designs and allows the use of auxiliary information for different estimation purposes. It utilises the generalised regression estimator theory as the basis of calculations. In principle, GES is **not** free software, but contact the methodological department of Statistics Canada for further details of the current distribution policy.

IVEwareis an SAS-based application for complex designs with the possibility for multiple imputation, developed at the University of Michigan. One can use several analytical procedures of SAS under IVEware. The software applies the jackknife and the Taylor approximation approaches for variance estimation. See `http://www.isr.umich.edu/src/smp/ive/` for further information and downloads (the software is free).

VPLX is standalone software for calculating the standard errors of the means, proportions and totals (also subclasses), developed by Dr Robert Fay from the U.S. Bureau of Census. The program takes the stratification and clusters into account. Variance estimation is based on replication methods (jackknife primarily, balanced repeated replication, random groups with weight modifications by the user). The software is free. Versions for Windows 3.1, 95, NT, 2000 and Unix are available for a download, see `http://www.census.gov/sdms/www/vwelcome.html`.

Chapter 4

Descriptions of Software

4.1 General Software

4.1.1 SAS

Theoretical aspects. Regarding survey sampling, there is one procedure for selecting samples, i.e. SURVEYSELECT, and four procedures for calculating results: SURVEYMEANS, SURVEYREG, SURVEYFREQ and SURVEYLOGISTIC (the last two only in version 9). The SURVEYMEANS procedure produces estimates of survey population totals and means with standard error and confidence interval calculation. A ratio of two means is included in SURVEYMEANS as well, but at least in version 8 only for estimation at the population level; domains must be dealt with a specific SRATSUB macro (obtainable from `http://ftp.sas.com/techsup/download/stat/sratsub.html`). SURVEYREG performs regression analysis for sample survey data, fitting linear models and computing regression coefficients and the covariance matrix. For one to n-way frequency and crosstabulation tables there is the SURVEYFREQ procedure, with corresponding variance estimation methods. It also allows design-based tests of association between variables, and for 2x2 tables the risk differences, odds ratios, relative risks and their confidence limits. SURVEYLOGISTIC performs logistic regression for categorical responses in sample survey data.

The procedures take both stratification and clusters into account. However, only the first stage variation of primary sampling units is considered in calculations. The designs with unequal probability sampling (PPS) can be defined to the procedures, but the variance estimation is in this case based on the with-replacement assumption. The method of Taylor approximation is used in variance estimation. Preliminary data preparations are needed in order to provide the stratum/cluster frequencies at the population level to the procedure. A variable including the weight information as well as stratum/cluster identification variables are required. The procedures as such do not include any features for utilising auxiliary information at the estimation phase, e.g. GREG and post-stratification. Note that outside these procedures the SAS programming language provides possibilities for all kinds of specific methods concerning variance estimation in different situations. Many applications made for more advanced calculations (primarily done by using macro language) are available in SAS user networks. On the other hand, the existing SAS procedures do not extend to more challenging designs (two-phase sampling, rotating scheme), there are only a few parameters to be estimated in its descriptive procedures (however, in analytical procedures the model parameters are considered), and there is no in-built tool for utilising auxiliary information in estimation.

Table 4.1: Theoretical Properties of SAS Survey Procedures

<table>
<tr><td colspan="7">Sampling designs</td></tr>
<tr><td>stratification</td><td>pps sampling</td><td>cluster sampling</td><td>two-stage sampling</td><td>three stages or more</td><td>Rotating Scheme</td><td>two-phase sampling</td></tr>
</table>

<table>
<tr><td colspan="3">Parameter definition</td></tr>
<tr><td>fixed in procedures</td><td>function of totals</td><td>specific function facility</td></tr>
</table>

<table>
<tr><td colspan="6">Variance estimation methods</td></tr>
<tr><td rowspan="2">analytic</td><td rowspan="2">taylor</td><td colspan="3">sample reuse</td><td rowspan="2">linearised jackknife</td></tr>
<tr><td>brr</td><td>jack</td><td>boot</td></tr>
</table>

<table>
<tr><td colspan="5">Specific estimation methods taken into account in variance estimation</td></tr>
<tr><td>poststratification</td><td>ratio estimator</td><td>GREG</td><td>calibration</td><td>response homogeneity groups</td></tr>
</table>

Existing property: bold text in shaded area
Non-existing property: italic text in white area

Computational aspects. The menu-based interaction in SAS suits better less experienced users, but in general the programming language of SAS (conducted in an editor) gives possibilities for very versatile data processing and further analytic and diagnostic operations. Graphical studies can be conducted fairly easily both by using menus and programming. User-constructed applications can be built with the macro facility of SAS or with the AF module providing tools for that purpose. However, these alternatives are much more demanding to be carried out than the basic operations with the ordinary SAS language. Both menu and program-based facilities for importing and exporting data sets of different types are available (see the table below). There are many platforms for SAS, but the current versions available in them vary, for example for Macintosh there is only version 6.12 for SAS (in January 2004 the latest version of SAS is 9).

The survey procedures are available only in programming. These procedures, following quite an easy syntax to be used, are comfortable for basic users having rather simple problems, sampling designs and estimators. Essential information is introduced in the procedures via statements and options, and the output includes the basic results needed for these estimation needs. As in-built structures, the procedures carry out the calculations rather fast. However, the macro applications needed for the domains in the ratio procedure, for example, slow down the calculations substantially. Some primary calculations based on the population are needed in order to get the required stratum frequencies (overall frequency) and sampling weights.

Table 4.2: Technical Details of SAS

January 2004	**SAS 9.0**
platforms/environments	OS/390 (SAS v9.0), CMS (v8.2), VSE (v6.08X), OS/2 (v8.2), Windows NT, XP, 2000 (v9.0), UNIX (v9.0), Alpha (v9.0), VAX (v8.2), Macintosh (v6.12)
interface	programming facility, menus and screens
data import	Excel 4, 5, 7, 97, 2000; Access 97, 2000; dBase; Lotus 1-2-3 versions 1, 3, 4; ASCII
data export	Excel 4, 5, 7, 97, 2000; Access 97, 2000; dBase; Lotus 1-2-3 versions 1, 3, 4; ASCII
requirements	for SAS in Windows minimum 128 Mb for improved performance, minimum 128 Mb of swap file space, SVGA Monitor (800x600 or higher), Base SAS requires 195 Mb of disk space, other parts vary from less than 1 to 133 Mb.

Documentation. Being very versatile software, the variety of SAS manuals is substantial, and along with the version updates the manuals follow fairly quickly. In addition to the ordinary manuals dealing with the different parts of SAS there are several technical manuals concentrating on specific topics. The online help of SAS and the tutorials assist the user in detail with problems and new practices. It is possible to use a search engine in the SAS help. A support website of SAS provides much general information on SAS as well as documents, programs and macro examples for different situations and methods (for example, data analysis sample programs on `http://ftp.sas.com/techsup/download/stat/`). Such advice and solutions can be found both from the SAS support and from the different user groups of SAS (e.g. an international organisation SUGI and the local organisations in many countries).

Other aspects. In general, SAS provides a good environment for all kinds of needs connected to data in an electronic form. Being multipurpose commercial software with a very large number of users around the world, the existence of SAS is by no means threatened. Its programming language and application tools give possibilities for conducting very advanced methods in practice. The survey procedures are quite recently launched (with some problems at the early stages of development, e.g. in domain estimation), but it seems that there will be further improvements coming concerning survey issues in SAS. However, at this stage of development the survey procedures in SAS might be used in production primarily in some usual sampling design situations estimating the most common parameters. Furthermore, some basic analysis and modelling can be conducted taking the sampling design into account. The price of this multifunctional software varies depending on the parts to be included in the system required by the user, but in general SAS is considered to be fairly expensive when compared with its competitors.

Table 4.3: General Information on SAS

January 2004	**SAS 9.0**	
Organisation, contact information	SAS Worldwide Headquarters SAS Institute Inc. 100 SAS Campus Drive Cary, NC 27513-2414 USA Phone: (919) 677-8000 Fax: (919) 677-4444	SAS International Headquarters SAS Institute GmbH P.O. Box 105340 Neuenheimer Landstr. 28-30 D-69043 Heidelberg GERMANY Phone : (49) 6221-4160 Fax : (49) 6221-474850
Website	`www.sas.com` support.sas.com	
Details for purchasing the product	Contact the office of your country (the list of offices available on the website). SAS has a large number of different parts designed for various purposes. The price depends much on the combination of the properties chosen. Furthermore, the prices vary between countries and the pricing policy is different for educational organisations. A single-user version with the Base SAS will be a few hundred Euros, and every additional product will increase the price. For organisations with hundreds of users with several SAS products available the price will easily rise to tens of thousands of Euros yearly.	

4.1.2 S-Plus and R

Theoretical aspects. The S language, of which R and S-Plus are two versions, provides an unparalleled range of general statistical tools, such as linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, data mining, imputation, simulation, bootstrap and other resampling routines. S-Plus & R *do not have built-in sample survey facilities as such*, but they provide tools for advanced programming concerning different problems in survey sampling. There is a recently developed R package 'survey' for survey sampling (LUMLEY, 2003) available on CRAN (`http://cran.ch.r-project.org`), which is still under development. Currently it can handle stratified and cluster sampling and raking, but not non-response. The variance estimation techniques available are delete-one jackknife, balanced repeated replication and Taylor linearization. There are also other packages available for survey estimation purposes in R, e.g. 'brrweights' for computing replicate weights and 'PostStratify' for post-stratifying a replicate weight survey, and the number of survey packages seems to increase. The majority of the computing routines programmed in other workpackages (especially the Monte Carlo simulation part) of the DACSEIS project are written in R.

Computational aspects. R and S-Plus are simple for those used to a high-level programming language. S-Plus on Windows has a very user-friendly 'menus and dialog boxes interface'. R does not, and although there are means to program one, in C, Tcl/Tk or Java, they are laborious. The interface depends on the version and platform. Both command line and graphical interfaces are available. Version of R and S-Plus exist for most of the main software platforms, though there is no version of S-Plus for the Macintosh.

Table 4.4: Theoretical Properties of R Survey Packages

Sampling designs						
stratification	*pps sampling*	**cluster sampling**	*two-stage sampling*	*three stages or more*	*Rotating Scheme*	*two-phase sampling*

Parameter definition		
fixed in commands (svy)	*function of totals*	*specific function facility*

Variance estimation methods					
analytic	**taylor**	**sample reuse**			*linearised jackknife*
		brr	**jack**	*boot*	

Specific estimation methods taken into account in variance estimation				
poststratification	*ratio estimator*	*GREG*	**calibration (raking)**	*response homogeneity groups*

Existing property: bold text in shaded area
Non-existing property: italic text in white area

The speed, that of a high-level language, can be improved by programming in C, C++, or Fortran, and using the interface to native code. For computationally-intensive tasks, code in these languages can be linked and called at run time. Advanced users can write C code to manipulate R objects directly. S is a language with three current implementations or “engines”, the “old S engine” (S version 3; S-PLUS 3.x and 4.x), the “new S engine” (S version 4; S-PLUS 5.x and above), and R. Given this understanding, asking for “the differences between R and S” really amounts to asking for the specifics of the implementation of the S language.

Data can be imported from most available formats, such as SAS or ASCII, and can be output in a variety of format, including special S-Plus and/or R dump files. There are excellent graphical tools for diagnostics. Also diagnostic checks are available if the code is well programmed. The quality of simulation algorithms is excellent. R is much more tolerant of badly-written code that can make S-Plus crawl.

R is much better than S-Plus when the memory requirements are concerned. S-Plus’s internal structure makes it rather memory-hungry. When R is started up it grabs a very large piece of memory and uses it to store the objects internally. R performs its own memory management of this piece of memory. S stores all objects as separate files in a director. R includes an effective data handling and storage facility, a suite of operators for calculations on arrays, a large collection of intermediate tools for data analysis, graphical facilities for data analysis and display either on-screen or on hardcopy. R is much smaller than S-Plus and runs on less powerful machines. On Windows it fits on four floppies and runs on an 8Mb Windows 95 machine. Once identified, bugs are rapidly removed from R by its expert panel. Bugs and awkward implementations in S-Plus are less rapidly fixed, and some remained in place for many years.

R, like S, is designed around a true computer language, and it allows users to add additional functionality by defining new functions. Much of the system is itself written in the

R dialect of S, which makes it easy for users to follow the algorithmic choices made. R can be considered as an implementation of S. There are some important differences, but much code written for S runs unaltered under R.

Table 4.5: Technical Details of S-Plus and R

January 2004	**S-Plus 6.2**	**R 1.8.1**
platforms/environments	Windows NT 4.0, 2000, XP Professional, Unix (Linux, Solaris)	Windows 95/98/ME/NT4/2000/XP, Macintosh OS X (10.2.x and above, for some older versions R 1.7.1), Linux, most Unix systems
interface	menus and dialog boxes	command line and graphical interfaces
data import	most available formats, such as SAS or ASCII	most available formats, such as SAS or ASCII
data export	a variety of format, including special S-Plus and/or R dump files	a variety of format, including special S-Plus and/or R dump files
requirements	256 Mb of free disc space for complete installation, about 100 Mb of swap space/virtual memory, Pentium II/233 processor with 96 Mb of RAM, SuperVGA or other Windows compatible graphics card and monitor	at least 8 Mb Windows 95 machine

Documentation. Both S-Plus and R have excellent documentation. The language has elements of object-oriented languages, and some metadata can be included using the corresponding tools. The help menus are excellent for both, using a web-browser interface in the case of R. There is a website for the history of the S language as well as websites for R and S-Plus. The routines written for different specific purposes are of varying quality, depending on who has written the package. Most packages are excellent; a few are less so. There are numerous very useful books on S programming, in particular VENABLES and RIPLEY (1999).

Other aspects. R and S-Plus are widely used in the statistical community. R is supported by a very active mailing-list and an international user community of experts, from whom free advice is rapidly available if necessary. S-Plus is commercial software, but it too is internationally known, and it is widely used in commercial environments. Both packages have a large user community, and it is unlikely that they will disappear in the foreseeable future. Both R and S-Plus are under continual development, with new tools and routines appearing constantly. R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It can be downloaded freely from the web site. On the other hand, S-Plus is commercial. R version 1.8.1 was released in November 2003.

Table 4.6: General Information on S-Plus and R

January 2004	**S-Plus 6.2**	**R 1.8.1**
Organisation, contact information	**Insightful Switzerland** Christoph Merian-Ring 11 4153 Reinach Switzerland Tel: +41 61 717 9340 Fax: +41 61 717 9341 info.ch@insightful.com **Insightful France** 7, rue Auber 31000 Toulouse France Tel: +33 0 5 62 27 70 60 Fax: +33 0 5 62 27 70 61 info.fr@insightful.com **Insightful UK** 5th Floor Network House Basing View Basingstoke, Hampshire RG21 4HG Tel: +44 (0) 1256 339800 Fax: +44 (0) 1256 339839 info.uk@insightful.com	**The R Foundation for Statistical Computing** *c/o Institut für Statistik und Wahrscheinlichkeitstheorie* *Technische Universität Wien* *Wiedner Hauptstraße 8-10/1071* *1040 Vienna, Austria* *Tel: (+43 1) 58801 10715* *Fax: (+43 1) 58801 10798* R-foundation@R-project.org

Websites	`http://www.insightful.com/`
	`http://cm.bell-labs.com/cm/ms/departments/sia/S/`
	(history of S language)
		`http://www.r-project.org`
		`http://stat.ethz.ch/CRAN/welcome.html`

Details for purchasing the product	Contact the distributor of your country (the list of distributors with contact information available on the website) or the international headquarters (see above). The price depends on the type of the organisation to use the product (much cheaper for educational organisations). Furthermore, choosing standalone or network solutions affects the price as well as the number of users. For universities one-user version costs usually less than 1000 Euros, but the commercial licences may cost a few thousand Euros.	R is free. The source code can be downloaded from the website of the r-project. The website provides advice (in the form of documents) for installation and use of R.

4.1.3 SPSS

Theoretical aspects. SPSS includes procedures for descriptive analysis, numerical prediction, group identification and forecasting. Regarding survey sampling, SPSS Release 12 has the add-on module "SPSS Complex Samples", which includes four procedures offering the possibility to draw and to analyse (with help of the module obtained or already existing) a stratified, clustered or multistage sample:

- Complex Samples Plan (CSPLAN) for specifying a sampling scheme and defining the plan file used by the following procedures
- Complex Samples Selection (CSSELECT) for obtaining the defined sample from a population
- Complex Samples Descriptives (CSDESCRIPTIVES) for estimating means, sums and ratios of variables, computing standard errors, design effects, confidence intervals and hypothesis tests for the drawn sample
- Complex Samples Tabulate (CSTABULATE) for displaying one-way frequency tables or two-way cross-tabulations, related to the above-mentioned descriptive statistics which can be requested by subgroups, too

The analysis of a sample presupposes the definition of a weighting variable (initial inclusion probabilities).

If values are missing from data, "SPSS Missing Value Analysis" may find some relationship between the missing values and other variables. The procedure can estimate what the value would be if the data were not missing. It can also estimate the mean, covariance matrix and correlation matrix, via regression or the EM algorithm. However, the available methods for replacing missing values in the data sheet (Series Mean, Mean/ Median of nearby points, Linear interpolation, Linear Trend at a point) are only useful for metric variables. This replacing (imputation) is not taken into account in variance estimation.

A general calibration methodology to calculate the extrapolation coefficients with SPSS has been designed by Statistics Belgium. A general calibration methodology to calculate the extrapolation coefficients with SPSS has been designed by Statistics Belgium. This programme, called "g-CALIB", allows calibrating hierarchically nested element and cluster level auxiliary information, either separately or simultaneously. g-CALIB requires about 4.1 MB hard disk space. The programme is easy to handle. It offers a clear and meaningful subdivided main window for specifying input data, parameter settings and the calibration model. The calibration constraints are entered in lines one by one in a separate window. User-defined calibration jobs can be saved. A detailed documentation (pdf-file) including screen shots and hints for output analysis is given to the user. Illustrative data can be generated with a special Microsoft Excel-file too. The simulated elements are transformed to SPSS by an additional routine, which can be integrated into a so-called "extended calibration job" as a "before calibration syntax". "After calibration syntaxes" can be integrated as well into a job. However, these operations are not yet taken into account in variance estimation. At present (January 2004) version 2 of g-CALIB is near its finalisation, including new theoretical features.

The GREG is currently not implemented and needs to be added, either by self-programmed code or by applying CSREGRESSION additionally. This also holds for more sophisticated weighting and imputation techniques. Bootstrap methods may be implemented via the Output Management System (OMS) for knowledgeable users.

Table 4.7: Theoretical Properties of SPSS

Sampling designs						
strati-fication	**pps sampling**	**cluster sampling**	**two-stage sampling**	**three stages or more**	*Rotating Scheme*	*two-phase sampling*

Parameter definition		
fixed in commands (means, totals, proportion, ratios)	*function of totals*	*specific function facility*

Variance estimation methods					
analytic	**taylor**	*sample reuse*			*linearised jackknife*
		brr	*jack*	*boot*	

Specific estimation methods taken into account in variance estimation				
poststratification	*ratio estimator*	*GREG*	*calibration*	*response homogeneity groups*

Existing property: bold text in shaded area
Non-existing property: italic text in white area

Computational aspects. SPSS is essentially menu-controlled. Data manipulation or analysis can be run and obtained by following the drop-down menu or by calling a macro. Some changes can be made on the fly during table creations. Individual format assignments can be got easily and special analysis routines can be saved or programmed as a module with Sax Basic. It is possible to record and save the menu commands as a macro as in MS Excel in order to automate the analysis process. There are separate windows for data, variable properties, analysis output and a special syntax window for SPSS's command language. SPSS graphical user interface enables a preview of tables (see SPSS tables). It is possible to add subtotals or change variable type or categories on the fly.

The SPSS command language can be compared with Microsoft's Visual Basic for Application for the Office family. The time needed for executing the commands depends strongly on whether the data are imported or called as an SPSS file. If the imported observations are saved as an SPSS file before recoding data and running the CSTABULATE procedure, calculations will take less time, particularly when applying to large data files.

The import and export facilities are exceptional. There are drivers for many ODBC compliant databases. A Database Wizard guides through the process of accessing data. SPSS output files can be saved in most commonly used formats, such as HTML, Text, Excel, Word/RTF and XML. There are many general diagnostic tools, but not especially for survey sampling. Employing SmartViewer Web Server with report OLAP interactive tables, graphs and charts can be presented in the web. Memory requirements are relatively low (see the table below).

Table 4.8: Technical Details of SPSS

January 2004	**SPSS 12.0**
platforms/environments	Microsoft Windows 98/2000/Me/Xp and NT4.0 **(SPSS 12.0)** Macintosh OS X **(SPSS 11.0)** Microsoft® Windows NT Server, Microsoft Windows 2000 Server, Windows Server 2003, SunSM Solaris 2.7 and above, IBM AIX 4.3.3 and above, and Red Hat Linux 8.0 **(SPSS Server)**.
interface	menu controlled, possible to use programming language as well
data import	SPSS, Systat, Excel, Lotus, SYLK, dBase, SAS, ASCII
data export	SPSS, ASCII, Text, Excel, 1-2-3, SYLK, SAS, HTML, Word/RTF, and XML
requirements (minimum)	For SPSS Base 12.0 Pentium class processor, 128 Mb RAM and SVGA monitor. SPSS SmartViewer Web Server requires a 500 MHz processor, 512 Mb RAM minimum and additional 700 Mb hard drive disk. Certainly, for practical use a more sophisticated system is strongly recommended. For example, the installation of the software package (SPSS Base 12.0, Advanced Models, Categories, Complex Samples, Conjoint, Exact Tests, Maps, Missing Value Analysis, Regression Models, Tables, and Trends) needs 232 Mb hard drive space plus (optionally) 51.8 Mb for the "SPSS Data Access Pack for Windows".

Documentation. SPSS has a selection of detailed and comprehensible documentation addressed to both professionals and users with less statistical skills. Metadata (characteristics) of variables are listed in a separate window in SPSS. The online help includes tutorials with a plenty of examples combined with step- by-step instructions, a chart advisor, a result coach and a statistical glossary. MS Internet Explorer and Adobe Acrobat Reader (included on SPSS CD) are required for working with the online documentation/ tutorials. In addition, case studies are available from the homepage. A website for SPSS exists (see *Contacts* below). In SPSS many examples with practical background are available to the user to improve data analysis skills with the program.

Other aspects. SPSS is widely used in business and educational establishments. SPSS Software is available in English, Japanese, French, German, Italian, Spanish, Chinese, Polish, Korean and Russian. The package has a large user community worldwide (see for instance the SPSS homepage). Both the rapid development of the SPSS product family in the last years and a big circle of users ask for an easy to apply but powerful program, suggest that SPSS will be used in future as well. For hotline and service centre for various countries see www.spss.com/worldwide. The current development on estimation with survey data makes it an alternative for producing (at the moment mostly descriptive) results taking the sampling design into account.

Table 4.9: General Information on SPSS

January 2004	**SPSS 12.0**	
Organisation, contact information	Corporate headquarters SPSS Inc. 233 S. Wacker Drive, 11th floor Chicago, IL 60606 Phone: 1 (312) 651-3000 Fax: 1 (312) 651-3668	Sales Department SPSS Inc. 233 S. Wacker Drive, 11th floor Chicago, IL 60606-6307 Phone: 1 (800) 543-2185 Fax: 1 (800) 841-0064
Website	www.spss.com www.spss.com/worldwide (offices) information about g-calib: http://www.statbel.fgov.be/studies/cal{_}en.asp	
Details for purchasing the product	Contact the office of your country (the list of offices with contact information available on the website). SPSS is commercial software including several different facilities for different purposes. Program prices depend on the countries concerning instructor, campus or students licences. SPSS offers one-year maintenance as well. For example, a campus licence may cost around 100 Euros but the price can be much more when dealing with private corporations and more advanced package combinations. Download of a 30-day demo version of SPSS Base 11.5 (56.7 Mb) is possible after registration.	

4.1.4 Stata

Theoretical aspects. As indicated, there are several different statistical analysis tools available in Stata. There are operations from simple point and variance estimation to more complex model calculations taking the survey aspects into consideration and conducted with commands beginning with **svy**. Stata can account for the following characteristics of the sampling design:

- unequal inclusion probabilities,
- stratification,
- clustering,
- without replacement sampling.

The different characteristics of the sampling design can be defined in Stata using a specific command. After this sample design characteristics are set, Stata automatically accounts for these sampling features in the survey commands. For the estimation of (sub)population parameters such as means, proportions, ratios and totals there are separate commands available. Stata always calculates covariances between different variables or different subpopulations.

Variance estimation procedures are based on direct methods or Taylor linearisations. Resample methods available in Stata 8 are the bootstrap and the jackknife. These procedures, however, assume independent observations. Resampling methods specifically for complex sample survey designs are not available. Stata calculates the design effect. There are no regression or calibration estimators as such in Stata. Consequently, it is not possible to use auxiliary information directly in the different estimation procedures. However, auxiliary information can be incorporated indirectly in the estimation procedure by means of the Stata program language. In Stata it is possible to conduct imputation as well.

Table 4.10: Theoretical Properties of Stata

<table>
<tr><td colspan="6">Sampling designs</td></tr>
<tr><td>strati-fication</td><td>pps sampling</td><td>cluster sampling</td><td>two-stage sampling</td><td>three stages or more</td><td>Rotating Scheme</td><td>two-phase sampling</td></tr>
<tr><td colspan="6">Parameter definition</td></tr>
<tr><td colspan="2">fixed in menus, commands</td><td colspan="2">function of totals</td><td colspan="2">specific function facility</td></tr>
<tr><td colspan="6">Variance estimation methods</td></tr>
<tr><td rowspan="2">analytic</td><td rowspan="2">taylor</td><td colspan="3">sample reuse</td><td rowspan="2">linearised jackknife</td></tr>
<tr><td>brr</td><td>jack*</td><td>boot*</td></tr>
<tr><td colspan="6">Specific estimation methods taken into account in variance estimation</td></tr>
<tr><td>poststratification</td><td>ratio estimator</td><td>GREG</td><td>calibration</td><td>response homogeneity groups</td></tr>
</table>

* only simple random sampling with replacement.
Existing property: bold text in shaded area
Non-existing property: italic text in white area

Computational aspects. Stata is mainly command-driven, but a few actions are available through the menu. The various commands are easily found in the Stata documentation. Learning Stata commands is easy due to their consistent structure. The platforms of software are Macintosh, Unix and Windows with cross-platform compatibility. The speed of the software is generally fast since Stata stores the whole dataset in memory. The estimation of large tables quickly becomes slow because the full covariance matrix is computed along with the estimates. Stata has its own binary data format but text data files can be easily read in. An optional package Stat/Transfer is available, which allows conversion of many popular formats.

The types of output files are text output and Stata data files, except for graphics Postscript and Windows Metafiles. For diagnostics there are extensive graphical capabilities. Some obvious checks are performed, such as whether the finite population correction is the same for all records within a stratum. There is no warning if the inclusion weights are incompatible with the finite population correction. Interactive sessions are easily converted into programs. The speed of Stata is fast due to storing the whole dataset in memory. It can use virtual memory although especially under Windows this results in reduced performance. For large tables calculating covariances between different variables or different subpopulations the speed of Stata reduces considerably. When dealing with very large data sets it depends on the magnitude class of Stata (*Small Stata / Intercooled Stata / Stata SE*) in use to make the software carry out the operation needed. When size problems occur, in some cases one can adjust an option in Stata to continue the work.

Table 4.11: Technical Details of Stata

January 2004	**Stata 8**
platforms/environments	Windows XP, 2000, NT, ME, 98; Power Macintosh (OS X 10.1 or later); Alpha AXP running Digital Unix; HP-9000 with HP-UX, Intel Pentium with Linux (32-bit); RS/6000 running AIX; SGI running Irix 6.5; SPARC running Solaris (64-bit and 32-bit). Linux 64
interface	mainly command-driven, but some operations can be conducted by using menus as well
data import*	Stata data files, text files (different ASCII forms), fda data (SAS XPORT)
data export*	Stata data files, ASCII, fda data (SAS XPORT)
requirements (minimum)	Pentium or compatible, 128 Mb memory, 20 Mb hard disk

* an optional package Stat/Transfer is available (allows conversion of many popular formats).

Documentation. There is a very comprehensive set of manuals for users at different skill levels, also including background information on statistical methods. Additions are published bi-monthly in the Stata Technical Bulletin. On-line help includes descriptions of all Stata commands and general help on how to use Stata. Additional information can be found on the website. Many examples of the use of Stata are provided in the Reference Manual.

Other aspects. Stata is a general purpose statistical package widely used in the statistical community. Updates including bug fixes are made available monthly and can be installed over the web. Stata is continually being developed. Prices differ significantly between different categories of users. Fast e-mail support service can be obtained. There is a wide support network, both from Stata developers and users community (mailing list). In general, with its versatile properties usable with complex designs as well it might be suitable especially for analytical purposes.

Table 4.12: General Information on Stata

January 2004	**Stata 8**
Organisation, contact information	**StataCorp LP** 4905 Lakeway Drive College Station, Texas 77845 USA Telephone: **800-782-8272**(that's 800-STATAPC, U.S.), **800-248-8272**(Canada), **979-696-4600**(Worldwide) Hours are 8 to 5, Central Time, Monday through Friday Fax: **979-696-4601**(Worldwide) Email: stata@stata.com technical support: tech-support@stata.com.
Website	`www.stata.com`
Details for purchasing the product	Contact the local distributor (see the website for the list of them) or order online on the website. Two main types of Stata are available for purchase: Stata SE (largest) and Intercooled Stata (Small Stata is for student use). The prices depend on the type and whether it is aimed for corporate or educational plans. Single user prices vary from a few hundred Euros to more than one thousand Euros. Network solutions are available as well.

4.2 Specialised Software

4.2.1 Bascula

Theoretical aspects. Bascula has been developed as a user-friendly tool for computing weights using auxiliary information. Starting from sampling weights based on the sampling design (also called inclusion or design weights) and incorporating the auxiliary information, the weighting package calculates one set of final (or adjustment) weights at the record level. These final weights can be expressed as the product of sampling weights and correction weights. Using auxiliary information implies that all estimates will be consistent with the known population totals; the precision of estimators is often improved; possible bias due to non-response or coverage errors is often reduced.

Bascula offers various weighting methods: post-stratification, ratio estimation, linear weighting based on the general regression estimator, and multiplicative weighting based on iterative proportional fitting. For surveys based on cluster sampling where all elements within a cluster are observed, it is possible to use linear weighting to calculate element weights that are equal within clusters. In this way, inconsistencies between the outcomes derived from element weighting and cluster weighting are avoided. In combination with linear weighting it is possible to apply a bounding algorithm, which may be used to force correction weights within a certain interval, which helps to prevent the occurrence of negative weights.

The choice of the weighting method depends on the available auxiliary information. If there are only categorical auxiliary variables and complete cross-classification of all auxiliary variables is available, the choice will be for post-stratification. However, if the numbers of sample elements in some strata are very small (less than 10), there is a risk of unstable estimators of population characteristics. In such a situation the user can decide to continue with marginal information and to choose between linear and multiplicative weighting. If the weighting model consists of only one continuous variable, the choice is between the ratio estimator and linear weighting. If the weighting model involves more than one continuous variable, then linear weighting is the only option. For consistent weighting (cluster sampling) the only option will be linear consistent weighting.

The computation of variances requires information about the sampling design. Bascula can handle the following designs:

- Stratified single-stage sampling where primary sampling units (PSUs) are selected using simple random sampling;
- Stratified two-stage sampling where both PSUs and secondary sampling units (SSUs) are selected by simple random sampling;
- Stratified multi-stage sampling where PSUs are selected (possibly with unequal probabilities) with replacement.

For sampling designs not supported, it may sometimes be reasonable to adopt one of these three designs as an approximation. If necessary, Bascula carries out the following preparations in order to be able to estimate variances with respect to the implemented designs: strata collapse by forming pseudo-strata if in some strata only one PSU has been sampled; when PSUs are selected with certainty (recognisable by a primary sampling fraction equal to 1), they should be considered as a separate stratum. As a consequence, each SSU within that PSU is viewed as a PSU. Bascula will do so automatically provided that these PSUs are at the end of their original stratum.

Bascula supports two methods for estimating variances. First, an algorithm based on balanced repeated replication (BRR) has been implemented. Half samples (resamples) from the full (or parent) sample are formed and a set of final weights for each resample is determined. Variance estimation is based on the estimates of target variables from resamples. The second way to estimate variances is based on a Taylor linearised expression for the variance estimator associated with the general regression estimator. This method cannot be used in combination with multiplicative weighting.

Table 4.13: Theoretical Properties of Bascula

Sampling designs						
stratification	**pps sampling**	**cluster sampling**	**two-stage sampling**	**three stages or more**	*Rotating Scheme*	*two-phase sampling*

Parameter definition		
fixed in menus, commands	*function of totals*	*specific function facility*

Variance estimation methods					
analytic	**taylor**	**sample reuse**			*linearised jackknife*
		brr	*jack*	*boot*	

Specific estimation methods taken into account in variance estimation				
poststratification	**ratio estimator**	**GREG (linear weighting)**	**calibration (multiplicative weighting)**	*response homogeneity groups*

Existing property: bold text in shaded area
Non-existing property: italic text in white area

Computational aspects. Bascula is menu-controlled. Bascula is built around a setup that contains the information needed to run a weighting session. A weighting session may be started from scratch by entering all necessary information step by step. New tab sheets appear in the main window after a step has been completed. A setup can be saved as a weighting input file with a file extension '.wif'. Instead of starting from scratch it is also possible to open an existing setup in Bascula for further processing. Bascula uses a setup in which all information needed to run a weighting session can be stored for later use. It holds all input parameters, notably the name of the sample file, population totals for auxiliary variables and the weighting scheme/model. The sample data file can be of either ASCII or Blaise format. Bascula is capable of retrieving all the relevant information about the sample data directly from the Blaise meta file.

The weighting and estimation process consists of eight steps:

- specification of a sample data file;
- selecting variables and assigning roles to them;
- defining population tables;
- specifying a weighting model;
- model reduction;
- specifying a weighting method and setting other parameters;
- weighting;
- estimation.

Various design/estimator combinations (i.e. strategies) can be handled. The outcomes can be written to an ASCII file for further processing. Weights can also be written back to the sample file. More specific information about a weighting session is stored in a log file.

Bascula is also available as a software component. The interface of this component, referred to as the Bascula Application Programming Interface (API), provides access to all of Bascula's functionality. The API can be used to automate weighting and estimation processes. Another possibility for carrying out a weighting session is by using a Manipula setup. Manipula is part of the Blaise package. It performs manipulations on a sample file via a set of instructions subdivided into sections.

Table 4.14: Technical Details of Bascula

January 2004	**Bascula 4.1**
platforms/environments	suitable for Windows 95, 98, ME, 2000, XP and Windows NT 4.0
interface	Menu-controlled
data import	ASCII or Blaise format
data export	ASCII or Blaise format
requirements	Pentium II or faster processor and a minimum of 16 Mb of system memory (32 Mb recommended).

Documentation. The Bascula 4.0 Reference Manual includes information about weighting and variance estimating methods employed by the software, a description on how to use Bascula for weighting and estimation, and more detailed information about some theoretical aspects. The use of Bascula is described clearly step by step and illustrated with informative figures. Throughout the manual a simple example based on fictitious data is used to demonstrate Bascula's capabilities. Two example files are included in the software package. The manual is available on the Bascula Website. Help menus are sometimes too brief. It is recommendable to use the Reference Manual.

Other aspects. Bascula has been developed primarily to calculate final weights. Bascula is a flexible tool for using auxiliary information, defining sample design and choosing between different weighting procedures. However, Bascula is capable of estimating population totals, means and ratios with corresponding variances (Taylor and/or BRR). Possibilities to define output tables are quite restricted. Bascula is a suitable software package for calculating survey weights especially if the survey is conducted using the Blaise system. The basic estimates and variance estimates can be computed inside Bascula to check the results of weighting.

Table 4.15: General Information on Bascula

January 2004	**Bascula 4.1**	Distributor
Organisation, contact information	Statistics Netherlands P.O. Box 4000 2270 JM Voorburg The Netherlands Fax + 31 70 3375969	Westat, Inc. Address Blaise services - RE 330S 1650 Research Boulevard Rockville, MD 20850-3129 USA Fax: +301 294-2040 Blaise@Westat.com
Website	`www.cbs.nl/en/service/blaise/bascula.htm` `www.westat.com/blaise`	
Details for purchasing the product	Contact Statistics Netherlands or Westat for purchasing Blaise. Note that Bascula is only a part of the Blaise system, which includes a wide range of different properties for survey purposes. Blaise licences are offered under two plans. The *corporate licence plan* is designed for organisations with ongoing CAI work, usually with high usage patterns (more than 100 application users). The *regular licence plan* is designed for a wide variety of other circumstances. As an example, the price per concurrent developer per year for the basic system is slightly less than 2,000 Euros.	

4.2.2 Clan

Theoretical aspects. The estimation methods in Clan are based on the theory of inclusion probabilities in weighting in different sampling strategies (following the footsteps of SÄRNDAL *et al.*, 1992), and variance estimation is a straightforward application of the Taylor linearisation method within these strategies. In practice, Clan takes only without-replacement schemes into account. The parameters to be estimated are functions of totals utilising Woodruff's (WOODRUFF, 1971) transformation method for variance estimation. Stratification, clusters, two-phase sampling and rotation can be defined for calculations. Sampling with unequal probabilities can be dealt with in Clan (pps sampling, Pareto πps sampling). Only the first stage variation is taken into account in variance estimation in multistage sampling. However, the manual presents an indirect way to introduce the second stage variation to the variance as well. A speciality of Clan is the ability for calculations under network sampling (familiar from surveys concerning households and their members). The approach of model-assisted estimation is essential in Clan, dealing with different methods (e.g. post-stratification, ratio, regression estimation and response homogeneity groups) under the framework of the general regression estimation theory (GREG). Clan also has a feature for calculating calibration weights in its macros; its variance estimation is based on the GREG theory as well. In addition, the calibration weights calculated elsewhere can be brought into the system. In all of these cases the data containing information about the marginal distributions and parameters of the population used in calibration must be available for variance estimation purposes of Clan. There is also a calibration tool for non-response, and subsampling of non-respondents with a proper calculation procedure is available as well. Clan calculates results for partially overlapping domain totals too.

Table 4.16: Theoretical Properties of Clan

Sampling designs						
stratification	**pps sampling**	**cluster sampling**	**two-stage sampling**	**three stages or more**	**Rotating Scheme**	**two-phase sampling**

Parameter definition		
fixed in menus / commands / procedures	**function of totals**	*specific function facility*

Variance estimation methods					
analytic	**taylor**	*sample reuse*			*linearised jackknife*
		brr	*jack*	*boot*	

Specific estimation methods taken into account in variance estimation				
poststratification	**ratio estimator**	**GREG**	**calibration**	**response homogeneity groups**

Existing property: bold text in shaded area
Non-existing property: italic text in white area

Computational aspects. Clan functions in the SAS system, so the platform, import and export files and primary memory requirements follow the properties of SAS. Nevertheless, the data sets for Clan must be in the SAS form. The Clan macros are utilised within the ordinary programming of the user. There are two main macros to be used: %FUNCTION and %CLAN. The %FUNCTION macro requires all the necessary information about the parameters to be estimated (expressed as functions of totals: adding, subtracting, multiplying, dividing available), whether using the Horvitz-Thompson estimator or the GREG estimator (defined for each total separately), all the estimators of the tables to be printed at the end of processing and the data set including auxiliary information from the population level with additional definitions for calculation purposes. The %CLAN macro includes all the design information (frequency variables, identification variables), data for calculation, table dimension definitions, allowing frequency checks, and additional weighting. The basic weights are calculated by Clan based on the frequency information and design/estimation definitions provided by the user.

The installation of Clan is easy, just to put a macro library (in earlier version an SAS program) to a known path and refer to it at the beginning of the program utilising Clan. Only the base SAS is required for Clan to function. Using the properties of the macro language, Clan is relatively fast in basic estimation processes, but the calculation of calibration weights takes some time even with moderate-sized data sets. The outcomes of the calculations in Clan are in the form of SAS data sets (both estimation results and calibration weights). The PRINT procedure can be utilised for obtaining visible outputs.

The structure of the table of results is fully defined by the user; the definition of the estimators in rows and columns can be a bit laborious if the user is not familiar with the macro language loops.

The way of conducting the estimation in Clan is rather logical but not very user-friendly. The nature of the macro language, numerous definitions and some prior preparations in estimation situations require much studying of the manual for a less experienced user with limited theoretical knowledge. On the other hand, with its versatile properties Clan is fairly easily suitable to the production of statistics with additional complexities/improvements in the sampling design/estimation methods, provided that the methodology behind the operations is known well enough by the researcher carrying out the operations. In fact, the origin of the program stems from special problems and methodologies appearing in some statistics at Statistics Sweden.

Table 4.17: Technical Details of Clan

January 2004	**Clan 3.1**
platforms/environments	working in SAS system
interface	utilising predefined macros within programming in SAS system
data import	facilities provided in SAS
data export	facilities provided in SAS
requirements	SAS language and macro facility at least

Documentation. The main information source of Clan is the users' manual, which can be obtained either in printed or electronic form. The manual consists of theoretical information as well as clarification about the syntax used in Clan macros. There are several well-explained examples of different cases in which Clan can be used (ratio estimator, GREG, two-stage sampling, subsampling of nonrespondents, two-phase sampling, network sampling). In addition to the manual, there are only some brief additional papers available. There are no help menus, website or hotline for search of information.

Other aspects. Being developed by two researchers (Claes Andersson, Lennart Nordberg) of Statistics Sweden along with their main duties, the future of Clan is somewhat uncertain. However, the software is used in a few statistical offices as well as other research organisations with survey problems. The applicability to complex survey designs and to the use of auxiliary information in various situations makes it a good tool for users with strong theoretical and practical knowledge of surveys. Unfortunately for less experienced users the threshold to use the software might be too high.

Table 4.18: General Information on Clan

January 2004	**Clan 3.1**
Organisation, contact information	Claes Andersson Statistics Sweden Klostergatan 23, SE-701 89 Örebro, Sweden claes.andersson@scb.se
Website	no website
Details for purchasing the product	Clan itself is free of charge, but it requires the SAS system. It is recommendable to contact Statistics Sweden for purchasing the software and obtaining the latest material and information available.

4.2.3 Genesees

Theoretical aspects. The methodological background of Genesees is based on the general class of calibration estimators and the theory behind it (Deville and Särndal, 1992). Within this framework the practices of GREG estimation are available, as well as ratio raking and generalised raking, for example. Variance estimation is based on the Taylor's linearisation method. The parameters to be estimated are totals, averages, frequencies, proportions, ratios and ratios of ratios. As in other specialised software, stratification and clusters are included. Genesees allows both sampling with and without replacement in its variance calculations, as well as unequal probability sampling (with-replacement assumption). A special feature available in Genesees is the breakdown of strata (collapsing) for strata with small response frequencies (the user can choose the frequency level). This problem is especially notable when there is only one primary sampling unit in one or more strata. The system creates so-called super strata for proper variance estimation. Genesees also provides inter-class correlation valued within the design clusters of one stage and valued within the primary units for designs of two or multistages. Together with estimates and their standard errors and confidence intervals the user can obtain the design effect and the efficiency of estimation as additional statistics. The speciality of Genesees is the synthetic presentation of sample errors, based on approximation methods that notably ease the calculation of the sampling errors when dealing with complex designs and a large amount of calculations for non-linear functions of the data samples. The theoretical alternatives for the synthetic presentation are regressive models and the effect of the sampling design. There has been some development work on a prototype of Genesees to include the linearised jackknife approach for variance estimation (see Falorsi *et al.*, 2002).

Computational aspects. The interface opens strictly as an SAS-based screen/menu application. The user must provide the information needed for estimation in four specific screens for that purpose (target variables, design variables, variables related to the estimator, variables in domain estimation). Other screens serve the calculation processes, some studies and savings of data. The structure of the menus is logical when the whole estimation process is concerned. However, all the definitions must be included as variables in the input data, and the amount of information Genesees asks is large, including many mandatory variables. For some simple design and estimation situations some mandatory

Table 4.19: Theoretical Properties of Genesees

Sampling designs						
stratification	**pps sampling**	**cluster sampling**	**two-stage sampling**	**three stages or more**	*Rotating Scheme*	*two-phase sampling*

Parameter definition		
fixed in menus (means, totals, proportions, ratios)	*function of totals*	*specific function facility*

Variance estimation methods					
analytic	**taylor**	*sample reuse*			**(linearised jackknife)***
		brr	*jack*	*boot*	

Specific estimation methods taken into account in variance estimation				
poststratification	**ratio estimator**	**GREG**	**calibration**	**response homogeneity groups**

* in a prototype of GENESEES.
Existing property: bold text in shaded area
Non-existing property: italic text in white area

definitions seem to be pointless from the theoretical point of view. These preliminary preparations in SAS might be rather laborious especially for the less-experienced user. The requirement of a character form for some variables is slightly odd. One cannot proceed without the variable requirements being fulfilled. The user can check the variable definitions made so far from a specific menu. The program makes many different checks concerning the design, estimation and data.

Although the interface is fairly user-friendly (when ruling out the fact that it is in Italian), some inconsistencies during using the software reveal that Genesees is not a finished product. The menus on the top of the screen vanish occasionally (they can be found with the left mouse button), and SAS menus appear from time to time. Going back to the previous screen is possible only from one menu option, and some screens (Log, Results Viewer) must be closed by using SAS menus. Error messages in Italian create a challenge for users not understanding the language. The speed depends on the definitions and the complexity of the design/estimation scheme, but with a simple design and simple estimation the results can be obtained rather quickly. The output form is clear and informative, covering the basic results and many additional features that Genesees provides. The program creates many output data sets for different purposes (a data set of the input parameters, errors revealed in the input data set, data sets containing information on estimates and sampling errors, data sets containing information on stratification and the sample, data sets containing information on estimate and sampling errors useful for successive processing and output files); saving them is conducted separately from the actual estimation process by using a specific menu.

Table 4.20: Technical Details of Genesees

January 2004	**Genesees 1.0**
platforms/environments	working in SAS system (version 8 at least)
interface	screen/menu-driven SAS application
data import	facilities provided in SAS
data export	facilities provided in SAS
requirements	SAS language and macro facility, SAS IML, SAS AF, SAS STAT, SAS GRAPH Space needed for installation 4 Mb on the fixed disc, at least 64 Mb of memory

Documentation. The main source of information in Genesees is the manual in English. The structure of the manual is good and logical, providing the necessary information for conducting the calculations with Genesees. The Italian expressions appearing in menus and screens are carefully explained. These properties are needed in order to succeed in the calculations. The theoretical part of the manual is quite convincing, including good descriptions of the methodology available in Genesees. The help menu helps only the Italian-understanding users. There are some papers and abstracts by the developers of Genesees/GSSE available concerning different aspects of the software. There is no website for Genesees.

Other aspects. The versatile theoretical properties of Genesees provide a good platform for calculations utilising different kinds of advanced methods of survey sampling. However, the interface in Italian as well as the laborious preparation work and numerous definitions before estimation might influence how rapidly the users will adopt the software to their data processing. Its features are especially at the estimation part; for simple situations Genesees is too complex in its practices when compared with the SAS survey procedures, for example.

Table 4.21: General Information on Genesees

January 2004	**Genesees 1.0**
Organisation, contact information	ISTAT Servizio Studi Metodologici Via Depretis, 74/B Rome I-00184 Italy Fax: 00 39 0646733221 pagliuca@istat.it, falorsi@istat.it
Website	no website
Details for purchasing the product	Contact Statistics Italy. The software itself is free of charge, but it requires the SAS software. It is requested that the researcher ordering the software should register. The software will not work without code information provided by ISTAT.

4.2.4 Poulpe

Theoretical aspects. The estimation philosophy of Poulpe is based on the use of auxiliary information via generalised regression estimation or calibration. Poulpe can deal with a large variety of sampling designs, estimation methods and non-response adjustments within a unified theoretical structure. Due to its exceptional way of describing the sampling design (defined as a 'tree'), it can deal with very complex designs, including stratification, cluster sampling and multistage sampling. Further properties of the software provide variance estimation for unequal probability sampling, systematic sampling, rotating schemes and multiphase sampling. This requires prior data and format arrangements as well as good knowledge of the algorithms involved for variance estimation. As parameters it deals with all kinds of functions of totals (adding, subtracting, multiplying and dividing as the basic functions). Variance estimation in Poulpe is based either on direct calculation or on Taylor approximation (linearisation method). The design effect can be requested as well. Poulpe allows the use of advanced estimation methods (ratio estimator, post-stratification, calibration). It is possible to use calibration weights (e.g. using the CALMAR macro from INSEE) for point estimation. Poulpe takes the calibration weights into account for variance estimation. It takes the unit non-response into account by considering an additional phase (Poisson sampling) in the sampling design.

Table 4.22: Theoretical Properties of Poulpe

Sampling designs						
stratification	**pps sampling**	**cluster sampling**	**two-stage sampling**	**three stages or more**	**Rotating Scheme**	**two-phase sampling**

Parameter definition		
fixed in menus / commands / procedures	**function of totals**	*specific function facility*

Variance estimation methods					
analytic	**taylor**	*sample reuse*			*linearised jackknife*
		brr	*jack*	*boot*	

Specific estimation methods taken into account in variance estimation				
poststratification	**ratio estimator**	**GREG**	**calibration**	**response homogeneity groups**

Existing property: bold text in shaded area
Non-existing property: italic text in white area

Computational aspects. Poulpe is screen-driven macro-based software used in the SAS environment. In principle, it guides the user through the process required for sufficient estimation and variance estimation; the user can manage without programming. The nature of dealing with the sampling design requires several preliminary definitions. In the sample dataset, the cluster membership must be set up according to a specific format (expressed with pairs of letters). Two input files (SAS data sets) need to be created to specify the structure of the sampling design. The first input file gives information about the stages, phases, strata, clusters, size variable, ranks and calibration variables. The second input files indicates the sizes of the stages, phases, strata and clusters. If those obligatory preparations for estimation are not carried out properly, the software will not allow the program to proceed until sufficient corrections are made. In general, creating the sampling tree as well as many compulsory definitions in the data set to be used may cause considerable difficulties for the ordinary user not very familiar with sampling theory. In that sense Poulpe is not user-friendly.

The screens appearing during the process are in French. The final outcome after filling the information required on the screens is the SAS programme for computing the variance estimates. The user can omit the screens by putting the predefined macros with sufficient information straight into the program flow in SAS. This requires good knowledge about the structure of the macros. The current version of Poulpe (released in 2003) works with SAS 8. For importing and exporting files it uses the possibilities provided by SAS (see Section 4.1.1). The output files are in a common SAS output form. In order to guide the user there are many error codes for different situations during program processing. Those codes are in French. One could use the tools SAS provides for diagnostics, but as such Poulpe does not provide much for that purpose. The speed depends on the number of stages, the type of surveys (stratified surveys are time consuming and require more disk space), the number of sampled units and the number of variance estimates required. In general it is estimated that the disk space required in Poulpe is about six times the size of the SAS sample dataset.

Table 4.23: Technical Details of Poulpe

January 2004	**Poulpe for SAS 8**
platforms/environments	working in SAS system
interface	screen-driven, but it is possible to use predefined macros in SAS programming as well
data import	facilities provided in SAS
data export	facilities provided in SAS
requirements	SAS language and macro facility at least

Documentation. There are many detailed documents in electronic form describing Poulpe theoretically, structurally and practically with several examples. However, the few documents existing in English do not cover the topic sufficiently for advanced use of the software for the non-French speaking user. The benefit of using the tree structure for describing the sampling design, i.e. the possibility to deal with very varying kinds of sampling designs, causes a rather complex system to be followed with several different aspects to be taken into account. Due to this reason together with other advanced features introduced in the software, the variety of documentation is somewhat disintegrated, and the threshold for understanding the software and its properties may be high for the ordinary user. There are no help menus available for Poulpe. It has no website and no hotline/help desk.

Other aspects. Where sampling theory is concerned, Poulpe is a very advanced software for variance estimation, covering many different sampling situations and the use of auxiliary information under a unified theoretical framework. The *unité méthodes de sondages* of INSEE (French National Statistical Institute) is in charge of its maintenance, but in the last few years the development work has been occasional. At the moment, the size of the Poulpe user group is small, consisting mainly of French-speaking users. In its current form the software is too complex for an ordinary user. For larger popularity the software should be improved from the practical point of view: the interface and documentation in English as well, a larger, properly unified Users' guide, some development for the screen system, and possibly a tool for preliminary preparations of the data.

Table 4.24: General Information on Poulpe

January 2004	**Poulpe for SAS 8**
Organisation, contact information	INSEE, *unité méthodes de sondages*, 18, boulevard Adolphe Pinard. 75675 Paris France Phone +33(0) 320.628.910 (Bernard Weytens) or +33(0) 141.176.072 (Jean-Noel Petit) bernard.weytens@insee.fr jean-noel.petit@insee.fr
Website	no website
Details for purchasing the product	Poulpe is free of charge, but it requires SAS software. It is recommendable to contact INSEE for purchasing the macro packages and obtaining the latest material and information available.

4.2.5 Sudaan

Theoretical aspects. Sudaan includes several procedures for estimation in correlated data. They produce results for both descriptive and analytical purposes. The procedures of a more descriptive nature are CROSSTAB (weighted frequency and percentage distributions for one-way and multi-way tabulations with standard errors), DESCRIPT (descriptive statistics for analysis variables, including means, totals, percentages, geometric means, medians and other quantiles and their standard errors) and RATIO (ratio estimates and their standard errors). Other procedures, clearly for analytic purposes, are LOGISTIC (RLOGIST in SAS-callable Sudaan, logistic regression models), LOGLINK (log-linear regression; count data not in the form of proportions), MULTILOG (modelling categorical outcomes with more than two categories), REGRESS (linear models) and SURVIVAL (proportional hazards modelling for failure time outcomes). The procedures include many theoretical features as options for further studies and testing. Sudaan can take several different sampling designs into account (as options of the procedures and statements). As in many other software, stratification, clusters and unequal probabilities (with replacement) can be defined in the procedure, but in addition, it is possible to include in the design unequal probabilities without replacement. Furthermore, here variance estimation with Taylor linearisation also reaches lower unit levels than the primary sampling units, i.e. the first stage clusters. The replication methods available in Sudaan are the delete-1 jackknife, the replicate weight jackknife and the balanced repeated replications. However, the assumption about these methods is that the first stage units are selected with replacement. Excluding post-stratification, the methods utilising auxiliary information at the estimation phase are not available in the procedures. There are four different alternatives of the design effect to be chosen. Standardisation and contrasting subgroup estimators are specific features available in descriptive procedures.

Table 4.25: Theoretical Properties of Sudaan

Sampling designs						
strati-fication	**pps sampling**	**cluster sampling**	**two-stage sampling**	**three stages or more**	*Rotating Scheme*	*two-phase sampling*

Parameter definition		
fixed in procedures	*function of totals*	*specific function facility*

Variance estimation methods					
analytic	**taylor**	**sample reuse**			*linearised jackknife*
		brr	**jack**	*boot*	

Specific estimation methods taken into account in variance estimation				
poststrati-fication	*ratio estimator*	*GREG*	*calibration*	*response homo-geneity groups*

Existing property: bold text in shaded area
Non-existing property: italic text in white area

Computational aspects. There are two alternatives for Sudaan: SAS-callable Sudaan and the standalone version. The programming syntax of the Sudaan procedures resembles the procedures in SAS, making the program rather easy for the SAS users. The information is given in statements and options of the procedure as in SAS procedures. The various data import forms (ASCII, SASXPORT, SUDAAN, SUDXPORT, SPSS) are important especially for the standalone version, requiring some extra definitions (in specific data sets) for reading ASCII. Technically Sudaan does not require very much from the computer: for SAS-callable Sudaan version 8.0 Windows 95, the 486 machine, 8 Mb of RAM and 40 Mb of hard disk are the minimum requirements.

The hierarchy levels of strata and clusters (NEST statement) can be changed from the default in the procedure with specific definitions, provided that corresponding population frequency variables are also given. For domain tabulations (SUBGROUP statement) the variables should be continuously numerated from 1 to the number of the groups, otherwise empty rows/columns will appear in tabulations. A recoding feature is provided in Sudaan for help in these cases. Furthermore, the number of classes in the tabulation variables must be provided in the LEVELS statement. This quite a rigid way of dealing with subgroups should be improved in the forthcoming versions. For prints and output data sets there are many detailed definitions available in the options of the procedures. Sudaan controls the data and required operations in the context of the sampling design: for example, there are checks for frequencies given by the user and small-cell warnings for tabulations. In many cases the execution of the procedure halts (with error messages) if there are discrepancies between the defined design and the reality in the data sets.

Table 4.26: Technical Details of Sudaan

January 2004	**Sudaan 8.0.2**
platforms/environments	Windows NT/2000/XP (release 8.0.2 separately for SAS versions 8 and 9) Windows 95/98/ME and NT/2000/XP, SUN workstations under Solaris (release 8.0 for SAS v. 8) Windows 95/98/ME and NT/2000/XP and DOS (Standalone release 8.0)
interface	additional procedures within SAS system (SAS-callable) or a standalone module for programming with procedures
data import	Standalone: ASCII, SAS 6.04 (except SUN/Solaris), SASXPORT, SUDAAN, SUDXPORT, SPSS SAS-callable: SAS (versions supported by SAS), SUDAAN, SUDXPORT, In the SAS-callable version one can preliminarily use the import possibilities of SAS.
data export	Standalone: ASCII, SASXPORT, SUDAAN, SUDXPORT, SPSS (except SUN/Solaris) SAS-callable: SAS (versions supported by SAS), SUDAAN, SUDXPORT, In the SAS-callable version one can preliminarily use the export possibilities of SAS.
requirements	For individual PC (Sudaan version 8.0 for SAS version 8 for Windows): 486 or better PC Compatible, 8 Mb of RAM (16 MB RAM suggested), 40 Mb of hard disk, Windows 95 and above

Documentation. The primary sources of information are two volumes of the users' manual and the online help with indexes and search engines appearing in SAS when SAS-callable Sudaan is installed. A majority of information in the online help can be found in the manuals. Together with the basic examples of the procedures the manuals provide additional examples of useful analytical practices within the descriptive procedures (e.g. standardisation and contrasting subgroup estimators). Theoretical solutions are well explained in the manuals for that purpose. An active website exists for further information about Sudaan.

Other aspects. The status of Sudaan is strong especially in those user groups that appreciate the versatile assortment of different analytical tools for complex surveys under cluster-correlated data. Dealing with theoretical problems is very detailed, and the scientific development and application work has been continuous since the early 1970s. However, purchasing the software only for simple designs and descriptive purposes is not reasonable, because for SAS users there are the survey procedures already available.

Table 4.27: General Information on Sudaan

January 2004	**Sudaan 8.0.2**
Organisation, contact information	Research Triangle Institute SUDAAN Statistical Software Center PO Box 12194 3040 Cornwallis Rd. Research Triangle Park, NC 27709-2194 USA Phone: 919-541-6602 Fax: 919-541-7431 SUDAAN@rti.org
Website	`www.rti.org/sudaan`
Details for purchasing the product	Contact Research Triangle Institute for purchasing Sudaan. It is also possible to make an order of the software online on the website of Sudaan. Both SAS-callable Sudaan and standalone Sudaan are of the same price. Pricing differs for academic, governmental and commercial purposes, as well as for an individual PC or a LAN PC. The range of prices is approximately from 700 to 1,400 Euros for the new first user. A two-year student version can be obtained with less than 100 Euros.

4.2.6 WesVar

Theoretical aspects. WesVar supports both balanced repeated and jackknife approaches for calculating replicate weights and provides the following replication methods:

- Balanced repeated replication (BRR) for designs with two primary sampling units (PSUs) per stratum;
- Fay's BRR variant;
- Jackknife for unstratified designs (JK1);
- Jackknife for stratified designs with two PSUs per stratum (JK2);
- Jackknife for stratified designs with two or more PSUs per stratum (JKn).

Other methods of replication such as the bootstrap can be handled in WesVar, but the user must input the replicate weights and factors appropriate for that method.

Replication methods are implemented in WesVar using the following steps:

Step 1 Divide the sample into subsample replicates that mirror the design of the sample by specifying the variance strata (VarStrat) and PSU (VarUnit) variables.

Step 2 Calculate weights for each replicate, using the same procedures used for the full-sample weight. The replicate weights are attached to the WesVar data file. WesVar supports weighting-class non-response adjustments, post-stratification and raking adjustments.

Step 3 Calculate replicate estimates for each of the replicates using the same methods used for the full-sample estimate. This computation is done by WesVar Table Requests and Regression Requests.

Step 4 Estimate the variance of the full-sample estimate, using the resulting full-sample and replicate estimates. The outputs of Table Requests and Regression Requests reflect this computation.

A maximum of 512 replicates can be created for the BRR and Fay's method using the Hadamard matrices stored in WesVar. A maximum of 9,999 replicates can be created for the jackknife replication methods. If a sample has a larger number of strata the user has to compute replicate weights outside WesVar. It provides diagnostics whether replicates have a large effect on variances.

Table 4.28: Technical Details of WesVar

Sampling designs						
stratification	**pps sampling**	**cluster sampling**	**two-stage sampling**	**three stages or more**	**Rotating Scheme**	*two-phase sampling*

Parameter definition		
fixed in menus, commands, procedures	*function of totals*	**specific function facility**

Variance estimation methods					
analytic	**taylor**	**sample reuse**			*linearised jackknife*
		brr	**jack**	**boot***	

Specific estimation methods taken into account in variance estimation				
poststratification	**ratio estimator**	*GREG*	**calibration (raking)**	**response homogeneity groups**

* require some extra formatting steps

Documentation. WesVar 4.0 Users' Guide includes information about installing, opening and operating, detailed examples, in-depth information on how to specify requests to produce estimates and their sampling errors. In addition, technical appendices describe replication methods and how the estimates are produced. The Users' Guide is well organised, clear and informative. Examples are thoroughly explained step by step with illustrations. It is easy for a new user to follow these directions.

Other aspects. WesVar is especially developed to estimate precision of survey estimates under complex sampling design and/or for complex functions of estimates. Methods used for computing replicate weights and estimating variances are highly computer-intensive. Memory requirements grow exponentially depending on the sample size and sample design. WesVar is recommendable for small-scale complex surveys with a relatively small number of primary sampling units.

Table 4.29: General Information on WesVar

January 2004	**WesVar 4.2**
Organisation, contact information	Westat 1650 Research Boulevard Rockville, MD 20850 USA phone: 301-294-2006, 301-294-2040 wesvar@westat.com
Website	`www.westat.com/wesvar`
Details for purchasing the product	Contact Westat for purchasing the product. The prices are cheaper for academic organisations (first user around 300 Euros) than for commercial or government organisations (first user around 400 Euros). Student version about 20 Euros. It is possible to download a 30-day demo version from the website.

Chapter 5

Evaluation of Software

5.1 Introduction

The evaluation of the software follow the framework set in Chapter 2. First of all, the general evaluation part includes an overview on the theoretical features included in the software with some discussion about the applicability of these methods (*Parameters and Sampling Designs in Software, Variance Estimation Methods in Software, Auxiliary Information and Specific Methods in Estimation in Software*). The special requirements for survey information provided by the user are considered to be important (*Adaptation of Design and Auxiliary Information into the Software System*). The mistakes appearing occasionally in survey practices were studied to some extent. A string of small tests including various problematic survey situations was conducted in the software (*Reacting Problems in Data and Estimation*). The interface and output examples with comments illustrate the usability of the software (*Interface and Style of Output*). Finally, concluding material about the survey processes of the software is presented (*Comparing Survey Processes in Different Software*).

Another part of the evaluation chapter is the testing with the three sample data sets selected from the DACSEIS Universes representing Finnish Labour Force Survey, Swiss Household Budget Survey and German Microcensus. Two first sections describe the background and the framework parallel to the simulation tasks in some other workpackages of DACSEIS. These three data sets are dealt with separately, although the experiences about the software are mainly the same for all data sets. *Note that the section of the Finnish Labour Force Survey includes many specific computational and practical aspects of the software. It was decided that the computational usability of the software with comparison could be described best in the context of real situation, as a continuous process.* The speed and the use of memory are of great interest here.

The software that are evaluated in this section are ***SAS***, ***SPSS***, ***Stata***, ***Bascula***, ***Clan***, ***Poulpe***, ***Sudaan*** and ***WesVar***. The description part included ***R***, ***S-Plus*** and ***Genesees*** as well, but they were excluded from the final, more detailed study. Although having potential (especially ***R***) for very advanced program construction regarding survey sampling, ***R*** and ***S-Plus*** software weren't actually designed for any survey problems. At the beginning of the DACSEIS project the predecessor of ***Genesees***, i.e. ***GSSE*** (developed by Statistics Italy), was not available for purchase in general, so it was excluded from the tests. Note that regarding ***Poulpe*** in data testing the programmer of this simulation had severe difficulties with the manuals in French and the material in English didn't provide enough information for a sufficient study of the macros. In that sense the evaluation is unfair for ***Poulpe***.

However, in Section 5.3.6 there is a simple speed test for ***R*** and ***SAS***-based software in order to compare the potential of ***R*** with some existing applications in ***SAS***. Furthermore, a general overview of the imputation procedure in SAS, Proc MI, is given in Section 5.3.7 including some tests with the Swiss Household Budget Survey data.

5.2 General Evaluation of the Software

5.2.1 Parameters and Sampling Designs in Software

Usually, the main aim of the survey is to produce fairly accurate estimates of one or more parameters of the target population. Furthermore, the survey requires a well-defined sampling design in order to follow the principle of probability sampling. Simple linear parameters (total, mean, proportion) can be estimated following the theory of the chosen sampling design. For nonlinear parameter functions estimation and variance estimation require more complex operations.

Parameters. All of the general software (***SAS***, ***SPSS***, ***Stata***) together with ***Bascula*** and ***Sudaan*** have the desired parameters fixed, i.e. the user cannot define them freely. The common survey parameters (totals, frequencies, means, proportions, ratios) are included in all of these software. In ***Stata*** a variety of analysis tools can be used with the survey data. Correspondingly, in ***Sudaan*** several analysis procedures can produce estimates for models. In these cases the standard error and corresponding confidence interval are calculated for the estimators of the model coefficients (together with t- and p-values).

Another possibility is to give the parameter as a function of totals (i.e. linear parameters), i.e. $\theta = f(\tau_1, \tau_2, \ldots, \tau_r)$, where $\tau_1, \tau_2, \ldots, \tau_r$ are population totals. These kind of functions can be treated by the software with the Taylor linearisation method (see Section 4.2.2) without any additional programming. ***Clan*** and ***Poulpe*** follow this principle. Finally, the parameter definition of ***WesVar*** is not limited to the functions of totals. For example, absolute value, exponential function, logarithm functions, median and quantile can be introduced in the parameter function. ***WesVar*** uses resampling estimates for variance estimation, and no theoretical derivations are needed even for very complex functions. ***Sudaan*** calculates the variance of the median and quantiles with the Woodruff method (See Section 5.2.2).

Sampling designs. Sampling design options available in different software usually follow the same theoretical properties, but a few exceptions exist. The formulae below are the analytic solutions familiar from the sampling literature, but they apply also for resampling methods at least for simpler cases, provided that the replicate weights are constructed correctly. See Section 5.2.2 for the principles of resampling methods and the Taylor linearisation method. Note that the formulae below include terms n and n_h: *in theory*

Table 5.1: Parameter Definition in Different Software

Software	fixed in menus / procedures / commands	function of totals	specific function facility
SAS	**FIP**		
R (survey)	**FIC**		
SPSS	**FIC**		
Stata	**FIMC**		
Bascula	**FIMC**		
Clan		**FT**	
Genesees		**FT**	
Poulpe		**FT**	
Sudaan	**FIP**		
WesVar			**FF**

they refer to the sample size and the sample size in the stratum h, but *in practice* they usually refer to the number of respondents and the number of respondents in the stratum h, assuming that the response probability is a constant either at the population level or at the stratum level.

Stratification. If pointed out in the software that the stratification exists and the stratum variable is defined, all the software calculate the variance in each stratum separately (in ***WesVar*** for stratification with more than two units in strata it requires the '*Jkn*' option, if the replicate weights are to be calculated internally). The estimator is

$$\widehat{\theta} = \sum \widehat{\theta}_h \ ,$$

where $\widehat{\theta}_h$ is the estimator in the stratum h. Correspondingly, due to the independence the variance estimator of the estimator is

$$\widehat{\mathrm{var}}(\widehat{\theta}) = \sum \widehat{\mathrm{var}}(\widehat{\theta}_h) \ .$$

However, if the stratum structure is not suitable for the BRR method (an alternative for Taylor's linearisation in ***Bascula***, see Section 5.2.2), ***Bascula*** artificially creates strata in order to make the calculations possible. This may cause some bias in variance estimation. ***WesVar*** does not allow calculations of methods not suitable for the existing stratum structure.

Simple random sampling without replacement. Simple random sampling without replacement can be considered as a basic survey design, and it is natural that all the software include this method in estimation. The estimator of the total is of the form

$$\widehat{\tau} = \sum \frac{N y_i}{n} = N \overline{y} \ ,$$

and the variance of the estimator is

$$\widehat{\mathrm{var}}(\widehat{\tau}) = \frac{N^2}{n}(1-f)\frac{(y_i - \overline{y})^2}{n-1} \ ,$$

Table 5.2: Sampling Designs in Different Software

Software	stratification	pps sampling	cluster sampling	two-stage sampling	three stages or more	rotating scheme	two-phase sampling	design effect*
SAS	**STR**	**PPS**	**CLU**	**2-ST**				
R (survey)	**STR**		**CLU**					
SPSS	**STR**	**PPS**	**CLU**	**2-ST**	**3-STM**			**DEFF**
Stata	**STR**	**PPS**	**CLU**	**2-ST**	**3-STM**			**DEFF**
Bascula	**STR**	**PPS**	**CLU**	**2-ST**	**3-STM**			
Clan	**STR**	**PPS**	**CLU**	**2-ST**	**3-STM**	**ROT**	**2-PH**	
Genesees	**STR**	**PPS**	**CLU**	**2-ST**	**3-STM**			**DEFF**
Poulpe	**STR**	**PPS**	**CLU**	**2-ST**	**3-STM**	**ROT**	**2-PH**	**DEFF**
Sudaan	**STR**	**PPS**	**CLU**	**2-ST**	**3-STM**			**DEFF**
WesVar	**STR**	**PPS**	**CLU**	**2-ST**	**3-STM**	**ROT**		**DEFF**

* available either automatically or with an option / a simple definition.

where we have the sampling rate $f = n/N$. In all software the variance-diminishing contribution of the sampling rate f (f_h for stratification) must be introduced in the software in some way: providing the population (stratum) size (***SAS***, ***SPSS***, ***Stata***, ***Bascula***, ***Poulpe***, ***Sudaan***), providing the sampling rate (in strata)(***SAS***, ***Stata***, ***Sudaan***), providing both the population (stratum) size and the number of respondents (in strata) (***Clan***), providing the finite population correction $(1 - f)$ (or $(1 - f_h)$ for stratification) (***WesVar***). ***Sudaan*** requires a separate option in order to recognise the design (WOR or for stratification SRSWOR) and in ***Poulpe*** the design must be identified in the sampling tree data (the design denoted by '*SAS*').

Sampling with replacement. With-replacement sampling is a basic assumption in ***WesVar***. ***Sudaan*** provides it in some design options (WR, STRWR, SRS). Then no population size is needed (in the 'SRS' option no weight is allowed at all). In the other software the with-replacement aspect must be brought into the system somehow (e.g. $f = 1/N$). ***Clan*** is not designed for sampling with replacement.

Systematic sampling. Usually variance estimation in systematic sampling is dealt with by using the variance estimator of simple random sampling without replacement. Only

Poulpe provides an alternative for this practice, i.e.

$$\widehat{\mathrm{var}}\left(\widehat{\tau}\right) = \frac{N^2}{n}\left(1-f\right)\frac{\sum_{i=1}^{n-1}\left(y_i - y_{i+1}\right)^2}{2(n-1)} .$$

The level (or levels) where systematic sampling is carried out must be defined in the 'sampling tree' data of ***Poulpe*** (the design denoted by '*SYS*').

Sampling with probabilities proportional to size. When sampling with probabilities proportional to size (PPS) is conducted in a with-replacement manner by using the estimator of the total

$$\widehat{\tau} = \sum \frac{y_i}{np_i} ,$$

with the number of respondents n and the probability p_i indicating the size of the unit i. For this estimator there is a simple variance estimator available, i.e

$$\widehat{\mathrm{var}}(\widehat{\tau}) = \sum \frac{\{y_i/(np_i) - \widehat{\tau}\}^2}{n-1} .$$

The software uses the weight given for estimation, i.e. $1/np_i$. When unequal probability sampling without replacement is considered, usually the with-replacement approximation is used for variance estimation, but ***Sudaan*** has an option for more exact variance estimation in that case (denoted by '*UNEQWOR*', available for the first stage only). It uses the Yates-Grundy-Sen variance estimator

$$\widehat{\mathrm{var}}(\widehat{\tau}) = \sum_{i=1}^{n}\sum_{i>j}^{n}\left(\frac{\pi_i\pi_j - \pi_{ij}}{\pi_{ij}}\right)\left(\frac{y_i}{\pi_i} - \frac{y_j}{\pi_j}\right)^2 ,$$

where π_i and π_j are the inclusion probabilities for units i and j, and π_{ij} is the joint inclusion probability for the pair of units i and j. The joint inclusion probability matrix must be provided separately ('*JOINTPROB*' statement), so there might be difficulties with the matrix when a large survey data is dealt with. ***Poulpe*** provides unequal probability sampling without replacement as well (denoted by '*PPT*' in the sampling tree) and it avoids the use of the joint inclusion probabilities with the Hájek variance estimator

$$\widehat{\mathrm{var}}(\widehat{\tau}) = \frac{n}{n-1}\sum\left(\frac{y_i}{\pi_i} - \widehat{R}\right)^2(1-\pi_i)$$

where $\widehat{R} = \sum(1-\pi_i)y_i/(\widehat{d}\pi_i)$ and $\widehat{d} = \sum(1-\pi_i)$.

Cluster sampling. The idea of (one-stage) cluster sampling is to obtain the estimates for each cluster (based on the elements of the clusters) and to calculate the variance of the estimator based on the cluster estimates. All the previous formulae apply when y_i is replaced by τ_i, which is the total obtained from cluster i, and the variables (N and n) represent frequencies of clusters. Provided that the indicator of clusters is given, the software automatically calculate the PSU level variance.

Two-stage sampling. Here we have sampling also at the second stage. Most of the software (***SAS***, ***SPSS***, ***Stata***, ***Bascula***, ***Clan***, ***WesVar***) omit the second-stage variation from the calculations. Usually the relative bias of this first-stage variance estimator is considered to be unimportant. However, the estimator and its weight *should include both stages* in order to obtain correct estimates. In ***Clan*** one can introduce the second-stage variation into the calculations with additional computational efforts. In ***Sudaan*** and ***Poulpe*** the exact variance estimator is used. Let s_I denote the sample of primary sampling units (PSU) and π_{Ii} denote the first phase inclusion probability. Within each PSU i, a sample s_{IIi} is selected. The variance estimator is

$$\widehat{\mathrm{V}}\left(\widehat{\tau}\right) = \widehat{\mathrm{V}}_{PSU} + \widehat{\mathrm{V}}_{SSU} = \widehat{\mathrm{V}}_I + \sum_{i \in s_I} \frac{\widehat{\mathrm{V}}_{IIi}}{\pi_{Ii}},$$

where $\widehat{\mathrm{V}}_I = \mathrm{V}\left(\sum_{s_I} \frac{\widehat{\tau}_i}{\pi_i}\right)$ and $\widehat{\mathrm{V}}_{IIi} = \mathrm{V}\left(\sum_{s_{IIi}} \frac{\widehat{\tau}_{ik}}{\pi_{ik}}\right)$.

Three-stage sampling or more. When considering only the primary sampling units and their variation, the situation does not change from the two-stage sampling case. If calculated exactly (***Poulpe***) the software follow the principle of variance components, e.g. for three stage sampling

$$\widehat{\mathrm{V}}\left(\widehat{\tau}\right) = \widehat{\mathrm{V}}_{PSU} + \widehat{\mathrm{V}}_{SSU} + \widehat{\mathrm{V}}_{TSU} = \widehat{\mathrm{V}}_I + \sum_{i \in s_I} \frac{\widehat{\mathrm{V}}_{IIi}}{\pi_{Ii}} + \sum_{i \in s_I} \frac{1}{\pi_{Ii}} \sum_{k \in s_{IIi}} \frac{\widehat{\mathrm{V}}_{IIIik}}{\pi_{IIk|i}}.$$

Two-phase sampling. Two-phase sampling occurs occasionally in situations where a deeper study is needed for a subgroup (or subgroups) of the original sample. An important special case is the additional study of nonrespondents (e.g. a brief phone interview with only the most essential topics). Finally one can make an estimator (and variance estimator) taking the information of nonrespondents into account by following the probability structure how the second-phase sample was formed. As in two-stage sampling, the idea is to decompose the variance for the first-phase estimator $\widehat{\tau}_1 = \sum_{k \in s_1} \frac{y_k}{\pi_{1,k}}$ and the second-phase estimator $\widehat{\tau}_2 = \sum_{k \in s_2} \frac{y_k}{\pi_{1,k}\pi_{2,k}}$, i.e.

$$\widehat{\mathrm{var}}\left(\widehat{\tau}\right) = \widehat{\mathrm{var}}_1\left(\widehat{\tau}_1; \pi_{1,1}, \ldots, \pi_{1,n}\right) + \widehat{\mathrm{var}}_2\left(\widehat{\tau}_2; \pi_{2,1}, \ldots, \pi_{2,n}\right).$$

The inclusion probabilities of unit k, i.e. $\pi_{1,k}$ and $\pi_{2,k}$ refer to the first and the second phase, correspondingly.

In ***Clan*** the second-phase target group is identified and the corresponding frequency information for the target group is given (options '*NSHG*' and '*NRHG*'). In addition to simple random sampling without replacement, ***Poulpe*** provides also an alternative of Poisson sampling for the treatment of non-response.

Rotating scheme. ***Clan***, ***Poulpe*** and ***WesVar*** have the feature of taking the rotating scheme into account in variance estimation. The partial change of units belonging to the sample in different time points could have effect in estimation. For example, if the independently selected panels A and B are studied at time 1 and the corresponding panels B and C at time 2 (perhaps following a different design structure when compared with time 1) we can deal with them as separate components. A simple approach for the estimation of the total (presentation following the structure in ***Clan***) is

$$\widehat{\tau}_{ab}^{(1)} = \alpha \cdot \widehat{\tau}(A)_{ab}^{(1)} + (1-\alpha) \cdot \widehat{\tau}(B)_{ab}^{(1)}$$

where $\widehat{\tau}(A)^{(1)}_{ab}$ and $\widehat{\tau}(B)^{(1)}_{ab}$ are the ordinary Horvitz-Thompson estimators of the total $\widehat{\tau}_{ab}$ based on the data of panel A and B respectively. The choice of the panel weight α is essential, for example $\alpha = 1/2$ for A and B with equal sizes and designs, and $\alpha = n_A/(n_A + n_B)$, where n_A and n_B are the sample sizes for panel A and B respectively. Analoguously, for time 2 we might have

$$\widehat{\tau}^{(2)}_{bc} = \beta \cdot \widehat{\tau}(B)^{(2)}_{bc} + (1-\beta) \cdot \widehat{\tau}(C)^{(2)}_{bc} \ .$$

The variance is estimated with

$$\begin{aligned} \mathrm{V}\left(\widehat{\tau}^{(1)}_{ab}\right) &= \alpha^2 \mathrm{V}\left(\widehat{\tau}(A)^{(1)}_{ab}\right) + (1-\alpha)^2 \mathrm{V}\left(\widehat{\tau}(B)^{(1)}_{ab}\right) \\ &= \mathrm{V}\left(\alpha\widehat{\tau}(A)^{(1)}_{ab}\right) + \mathrm{V}\left((1-\alpha)\widehat{\tau}(B)^{(1)}_{ab}\right) \end{aligned}$$

and correspondingly

$$\begin{aligned} \mathrm{V}\left(\widehat{\tau}^{(2)}_{bc}\right) &= \beta^2 \mathrm{V}\left(\widehat{\tau}(B)^{(2)}_{bc}\right) + (1-\beta)^2 \mathrm{V}\left(\widehat{\tau}(C)^{(2)}_{bc}\right) \\ &= \mathrm{V}\left(\beta\widehat{\tau}(B)^{(2)}_{bc}\right) + \mathrm{V}\left((1-\beta)\widehat{\tau}(C)^{(2)}_{bc}\right) \end{aligned}$$

Design effect. The principle of the design effect can be seen in

$$\widehat{\mathrm{Deff}}(\widehat{\tau}) = \frac{\widehat{\mathrm{var}}\,(\widehat{\tau})}{\widehat{\mathrm{var}}\,(\widehat{\tau}_{SRS})} \ ,$$

i.e. the ratio of the variance under the actual sampling design to the variance under simple random sampling. Note that the denominator is for simple random sampling without replacement in ***SPSS, Stata*** and ***Poulpe***, but in ***Sudaan, WesVar*** the supposed design is simple random sampling with replacement. The definition of this term is not without problems: the direct simple random sampling variance form

$$\widehat{\mathrm{var}}(\widehat{\tau}) = \frac{N^2}{n}(1-f)\frac{\sum(y_i - \overline{y})^2}{n-1}$$

may be badly biased if strongly unequal inclusion probabilities are involved in the actual sampling design. ***Sudaan*** provides four alternatives for calculating a variance estimate for simple random sampling with replacement: 1) the formula above with $f = 1/N$ (for subgroups of study it is assumed that the sample sizes are not fixed, 2) as in point 1 but subgroups fixed, 3) model based simple random sampling variance and 4) simple random sampling variance reflecting to the varying inclusion probabilities used in the design. The similar alternative as point 4 is found in ***Poulpe*** (for simple random sampling without replacement). It provides an unbiased alternative taking the actual design into account, i.e.

$$\widehat{\mathrm{var}}(\widehat{\tau}) = \frac{\widehat{N}}{n}\left(1 - \frac{n-1}{\widehat{N}-1}\right)\sum_s \frac{1}{\pi_k}\left(y_k - \frac{\widehat{\tau}}{\widehat{N}}\right)^2$$

where $\widehat{N} = \sum_s 1/\pi_k$. For two-phase sampling, π_k is the product of the probability at the first-phase and at the second phase. ***WesVar*** follows that principle (incorporating inclusion probabilities and an estimate of population size) as well in the case of simple random sampling with replacement.

5.2.2 Variance Estimation Methods in Software

For linear estimators, there are analytic variance estimation methods available in the sampling literature (see Section 4.2.1). Otherwise, variance estimation methods for nonlinear estimators can be divided into two main classes: *traditional approximative methods* (Taylor's linearisation, using inverse cumulative density functions for quantiles) and *resampling (replication) methods* (e.g. Balanced Repeated Replications, Jackknife, Bootstrap). A recent approach utilising the properties of both branches is the *linearised jackknife* (YUNG and RAO, 1996), where 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. However, only a prototype of one software (*Genesees*) has this feature. The following table shows which methods are available in the software under study.

Table 5.3: Variance Estimation Methods in Different Software

Software	**analytic**	**taylor, inverse cumulative density functions**	**Sample reuse**			**linearised jackknife**
			balanced repeated replication	**jackknife**	**bootstrap**	
SAS	**ANA**	**TAY**				
R (survey)	**ANA**	**TAY**	**BRR**	**JACK**		
SPSS	**ANA**	**TAY**				
Stata	**ANA**	**TAY**		**JACK**1)	**BOOT**1)	
Bascula	**ANA**	**TAY**	**BRR**			
Clan	**ANA**	**TAY**				
Genesees	**ANA**	**TAY**				**(LJ)**2)
Poulpe	**ANA**	**TAY**				
Sudaan	**ANA**	**TAY, ICDF**	**BRR**	**JACK**		
WesVar	**ANA**		**BRR**	**JACK**	**BOOT**3	

1) only for the i.i.d. case (no sampling design definition);
2) only in a prototype of Genesees
3) replicate weights must be provided by the user

Taylor linearisation method. The principle of this method is to make a linear approximation for the nonlinear estimator and to use the variance estimate for this approximation to estimate the variance of the estimate itself. The estimator of the variance is of the form

$$\begin{aligned}\widehat{\text{var}}(\widehat{\theta}) &= \widehat{\text{var}}\left(\sum_{j=1}^{r} f_j'(\tau_1,\ldots,\tau_r)\widehat{\tau}_j\right)\\ &= \sum_{i=1}^{r}\sum_{j=1}^{r} f_i'(\widehat{\tau}_1,\ldots,\widehat{\tau}_r) f_j'(\widehat{\tau}_1,\ldots,\widehat{\tau}_r)\widehat{C}(\widehat{\tau}_i,\widehat{\tau}_j)\end{aligned}$$

where $f'_j(\widehat{\tau}_1,\ldots,\widehat{\tau}_r)$ and $\widehat{C}(\widehat{\tau}_i,\widehat{\tau}_j)$ are the sample estimators of $f'_j(\tau_1,\ldots,\tau_r)=\dfrac{\partial f(\tau_1,\ldots,\tau_r)}{\partial\tau_j}$ and $C(\widehat{\tau}_i,\widehat{\tau}_j)$, which is the covariance between $\widehat{\tau}_i$ and $\widehat{\tau}_j$. The partial derivatives $f'_1,\ldots,f'_j,\ldots,f'_r$ usually depend on the unknown $(\tau_1,\ldots,\tau_r)$.

Most of the linearisation-based software (***SAS***, ***SPSS***, ***Stata***, ***Bascula***, ***Sudaan***) have only in-built calculation structures for some pre-defined estimators. However, the software using the function of totals in their calculation system (***Clan*** and ***Poulpe***), provide automatically a linearisation of the estimator function, i.e. $\widehat{\theta}=f(\widehat{\tau}_1,\ldots,\widehat{\tau}_r)$ given by the user, for variance estimation purposes. By using Woodruff's (Woodruff, 1971) transformation $\widehat{z}_i=\sum_{j=1}^{r} f'_j(\widehat{\tau}_1,\ldots,\widehat{\tau}_r)y_{ji}$, where y_{ji} is the variable j of the observation i, the calculation is significantly simplified. Further, for rational functions we have

$$f(\tau_1,\ldots,\tau_r)=G\left(g_1(\tau_1,\ldots,\tau_r),g_2(\tau_1,\ldots,\tau_r)\right)\ ,$$

and the Woodruff transformation is possible separately for functions g_1 and g_2. In ***Clan*** and ***Poulpe*** these transformations are conducted in a stepwise fashion in order to take into account all the functions described by the user. Addition ('*ADD*'), subtraction ('*SUB*'), multiplication ('*MULT*') or division ('*DIV*') of two totals or functions of totals can be defined in both ***Clan*** and ***Poulpe***.

Inverse cumulative density function for quantiles. This method (see e.g. Francisco and Fuller, 1991), based on the functions of the cumulative distribution function of a variable, is available in ***Sudaan***. Usually the method appears in the context of estimating the confidence interval of the estimator of the median. The cumulative distribution function of the population U is

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

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

$$\widehat{F}(y)=\sum_{k\in s}\frac{z_{k,y}}{\pi_k}\Bigg/\sum_{k\in s}\frac{1}{\pi_k}\ .$$

The final result is the confidence interval

$$\left[\widehat{F}^{-1}\left(0.5-t_{0.025}(d)\sqrt{\widehat{\mathrm{V}}(F(M))}\right),\widehat{F}^{-1}\left(0.5+t_{0.025}(d)\sqrt{\widehat{\mathrm{V}}(F(M))}\right)\right],$$

where $F(M)$ is the cumulative density function of the median. ***Sudaan*** calculates the standard error from which we calculate the standard error of the estimator of the median based on the difference of the upper and lower terms of the confidence, i.e.

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

for the 95 % level of confidence and d degrees of freedom for the t-value of the confidence interval. Note that the interval is usually asymmetric around the median, and therefore the standard error is an approximation. On the other hand, 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. For other quantiles 0.5 is changed to the corresponding value. Note that ***WesVar*** deals with quantiles in the context of replication methods, and this approach is not needed in that software.

Replication methods. The variation between the replicate estimates and full-sample estimates is used in order to estimate the variance for the full sample. In practice the software uses a matrix suitable for the replication method in question. The matrix can be brought into the system as a separate data file (***WesVar***, ***Sudaan***) or the software creates it (***Bascula*** for BRR, ***Sudaan*** for Jackknife, ***WesVar*** for BRR, unstratified Jackknife and stratified Jackknife). This matrix can include some correction terms reflecting to the specific nature of the method. The 'two-units-per-stratum' methods (*Balanced Repeated Replication, Fay's adjusted BRR, Jackknife with two-unit samples*) follow a specific matrix structure (Hadamard matrix). ***Bascula*** allows the BRR method for other designs as well by creating artificial strata. ***Sudaan*** requires the matrix for the BRR method. ***WesVar*** does not allow calculations of methods not suitable for the existing stratum structure. Note that ***WesVar*** can deal with e.g. the bootstrap method as well, if the user provides the proper matrix and the correction coefficients required by the method. ***Stata*** deals with the jackknife and the bootstrap only for the i.i.d. case; in that case ***Stata*** allows no sampling design definition.

For all the methods, the variance estimator can be expressed in a general form (presentation following the structure in ***WesVar,*** applicable for others as well)

$$\widehat{\mathrm{var}}(\widehat{\theta}) = c \sum_{g=1}^{G} f_g h_g (\widehat{\theta}_{(g)} - \widehat{\theta})^2 \ ,$$

where

$\widehat{\theta}$ is the full-sample estimate of parameter θ,

$\widehat{\theta}_{(g)}$ is the estimate of θ based on the observations included in the gth replicate,

G is the total number of replicates formed,

c is a constant that depends on the replication method.

f_g is JKn factor (JK = Jackknife)

h_g is finite population correction factor.

The last two terms (f_g, h_g) appear only in the stratification with more than two units per stratum.

5.2.3 Auxiliary Information and Special Methods in Estimation in Software

A common practice in surveys is to improve estimation with auxiliary information, usually from the population level. In addition, almost always surveys include non-response; this problem can be treated with various methods. These two aspects are more or less present in the software for surveys. In this section we deal with *post-stratification, ratio estimator, GREG estimator, response homogeneity groups, calibration and specific non-response treatments* and their effect to variance estimation in software. The following table reveals that in practice only specialised software have properties which take these aspects into account.

Table 5.4: Auxiliary Information and Special Methods in Estimation in Different Software

Software	**post-stratification**	**ratio estimator**	**GREG estimator***	**calibration**	**response homogeneity groups**
SAS					
R (survey)				**CAL****	
SPSS					
Stata					
Bascula	**PS**	**RAT**	**GREG*****	**CAL****	
Clan	**PS**	**RAT**	**GREG**	**CAL**	**RHG**
Genesees	**PS**	**RAT**	**GREG**	**CAL**	**RHG**
Poulpe	**PS**	**RAT**	**GREG**	**CAL**	**RHG**
Sudaan	**PS**				
WesVar	**PS**	**RAT**		**CAL****	**RHG**

* the software should have a feature for GREG estimation; a modelling tool in general is not considered here, ** raking, *** linear weighting

GREG estimator. The general regression estimator (see Särndal *et al.*, 1992) can be in fact considered as a family of estimators, depending on the model assumptions which are chosen. It is well beyond the scope of this deliverable to provide a detailed overview on model assisted estimation. In this context only the basic idea is presented, mainly following the structure of presentation in ***Clan***. The GREG estimator of the total is

$$\widehat{\tau} = \widehat{\tau}_{y\pi} + \sum_{j=1}^{J} \widehat{B}_j(\tau_{x_j} - \widehat{\tau}_{x_j\pi})$$

where $\widehat{\tau}_{y\pi} = \sum_r \frac{y_k}{\pi_k \widehat{\theta}_k}$ (summation over respondents r) is the Horvitz-Thompson estimator of τ_y, $\widehat{\tau}_{x_j\pi} = \sum_r \frac{x_{jk}}{\pi_k \widehat{\theta}_k}$ is the Horvitz-Thompson estimator of τ_{x_j}, and $\widehat{B}_1, \widehat{B}_2, \ldots, \widehat{B}_J$ are the components of the vector $\widehat{\mathbf{B}} = \left(\sum_r \frac{\mathbf{x}_k \mathbf{x}'_k q_k}{\pi_k \widehat{\theta}_k}\right)^{-1} \sum_r \frac{\mathbf{x}_k y_k q_k}{\pi_k \widehat{\theta}_k}$ (weighted least squares estimator), where we have the value $\mathbf{x}_k = (x_{1k}, \ldots, x_{jk}, \ldots, x_{Jk})\prime$ of the vector $\boldsymbol{x}$ of length J. Note that there is an estimator of the response probability for the unit k, i.e. $\widehat{\theta}_k$ (see the corresponding section below). It is assumed that the study variable y and the auxiliary variables $\mathbf{x}$ follow a linear model

$$\begin{cases} \mathrm{E}_\xi(Y_k) & = \mathbf{x}'_k b \\ \mathrm{V}_\xi(Y_k) & = \sigma_k^2 \ = \sigma^2/q_k \end{cases}$$

where E_ξ and V_ξ denote expected value and variance with respect to the model ξ while β and σ_k^2 are usually unknown model parameters and $q_k (> 0)$ is known for every i. The essence here is that the researcher can choose in what form these terms appear in the model. Two familiar examples are the *ratio model* for the ratio estimator, i.e.

$$\begin{cases} \mathrm{E}_\xi(Y_k) & = \beta x_k \\ \mathrm{V}_\xi(Y_k) & = \sigma^2 x_k \end{cases}$$

and the *group mean model* for the poststratified estimator, i.e.

$$\begin{cases} \mathrm{E}_\xi(Y_k) = \beta_g \\ \mathrm{V}_\xi(Y_k) = \sigma_g^2 \end{cases}$$

where $g = 1, \ldots,$ G is the indicator of the nonoverlapping groups of the population. The variance of the GREG estimator is the variance of the residuals, $e_k = y_k - \mathbf{x}'_k \widehat{\mathbf{B}}$.

The user can adjust the model directly in ***Clan*** and ***Poulpe***. ***Clan*** provides also a tool for weight calculation. One of the key properties of ***Bascula*** is the weighting process, allowing the user to create a weighting model for various purposes. The *'linear weighting'* model can be considered as a method producing GREG estimates. Both in ***Clan*** and ***Bascula*** the user can define bounds for weights as well. This is sometimes preferred when the model would produce e.g. negative weights. The GREG weight is of the form (presentation style from ***Clan***)

$$w_k = \left(1 + (\tau_x - \widehat{\tau}_x)' \left(\sum\nolimits_r \frac{\mathbf{x}_k \mathbf{x}'_k q_k}{\pi_k \widehat{\theta}_k}\right)^{-1} \mathbf{x}_k q_k\right) \times \frac{1}{\pi_k \widehat{\theta}_k} = \frac{g_k}{\pi_k \widehat{\theta}_k}.$$

Note that the non-response can be somewhat adjusted via the term $\widehat{\theta}_k$, which is important when e.g. the response homogeneity group method is used. Normally we have $\widehat{\theta}_k = m_h/n_h$, where m_h is the number of respondents in stratum h and n_h is the sample size in stratum h. All three software can deal with two-level models, e.g. including both household and person level information. For correct variance estimation in the software the user should provide the same auxiliary information on which the model is based.

Post-stratification. The GREG estimator has post-stratification as a special case, but in practice all the software including this feature have a simple way to conduct it. The requirements are the population frequencies for nonoverlapping post-strata (either in a separate data or included in the survey data) and a variable indicating these post-strata. If the sampling design includes stratification the post-strata frequencies should be obtained for every stratum. For the case when stratification already exist the estimator is

$$\widehat{\tau} = \sum_{h=1}^{H} \sum_{g=1}^{G_h} \frac{N_{hg}}{n_{hg}} \sum_{s_{hg}} y_i \ .$$

The use of post-stratification often improves estimation and it can be considered as a method of correcting the bias due to non-response, provided that we have the number of respondents in hg as a denominator.

Ratio estimator. A simple estimation method is to adjust the estimator of the total with a correcting ratio concerning an auxiliary variable. The ratio between the total of variable x and its estimate reveals the magnitude of underestimation or overestimation, which may be beneficial for the estimation of the total of y, if there is enough positive correlation between x and y. The ratio estimator is of the form

$$\widehat{\tau}_{rat} = \frac{\tau_x}{\widehat{\tau}_x}\widehat{\tau}_y \ .$$

As indicated earlier, this can be seen as a special case of the GREG estimator. This estimator can be directly defined in ***Bascula*** (option in a menu), ***Clan*** (using GREG feature), ***Poulpe*** (using GREG feature) and ***WesVar*** (option in a screen). However, including a ratio as a parameter to be estimated, ***SAS***, ***Stata***, ***SPSS*** and ***Sudaan*** can be used indirectly in order to achieve the ratio estimator: one should simply define the denominator variable of the ratio as $x^* = x/\tau_x$.

Calibration. The theory of calibration has been under increasing research in recent years. The idea is to introduce the population distributions and/or parameters in the weight modelling process in such a way that some conditions (usually dealing with a distance measure) can be fulfilled. In more complex situations some iterations are needed in order to obtain the weights. The calibration theory is not presented in this context. The outcome (often suitable for the production of the official statistics) is that when estimating the calibration variables from the survey data with calibration weights, we get always the right parameters and/or distributions of the population.

The most common approach in calibration is raking with marginal totals (a simple example of raking is the ratio estimator). In addition, the GREG weights can be interpreted as an outcome of calibration as well (fulfilling the ”right parameters/distributions” condition). ***Bascula*** (multiplicative weighting), ***Clan*** (GREG based) and ***WesVar*** (raking), have both a structure for calibration weight creation and for variance estimation taking the calibration into account. ***Poulpe*** deals only with variance estimation for calibration; a SAS macro called ***Calmar*** provides the calibration weights. ***Clan*** allows the external calibration weights as well in variance estimation. As in other adjustment methods, the auxiliary information which is utilised in the creation of the model must be provided by the user (also for the case when the weights are imported to the system). Here we briefly present two tools for calibration (in addition to ***Clan*** and ***Bascula***)

Response homogeneity groups and other non-response corrections. The idea of response homogeneity groups is to assume that the non-response probabilities θ_{k}. are formed in a more complex way than having only $\widehat{\theta}_k = m_h/n_h$, where m_h is the number of respondents in stratum h and n_h is the sample size in stratum h. The units in these groups should be internally as similar as possible, when the non-response is concerned. Note that here, differing from the post-stratification, the population level information is not used for group hg. The estimator is then

$$\widehat{\tau} = \sum_{h=1}^{H} \frac{N_h}{n_h} \sum_{g=1}^{G_h} \frac{n_{hg}}{m_{hg}} \sum_{k=1}^{m_{hg}} y_{hgk} \ .$$

The method appears in ***Clan***, ***Poulpe*** and ***WesVar***, and it requires both the sample size and the number of respondents in each group hg, and naturally the group identification variable (only the identification in ***WesVar***).

One could also use a logit model in order to estimate $\widehat{\theta}_k$. As seen in the GREG context, the basic sampling weight consists of the function of the inclusion probability and the estimator of the response probability, i.e. $(\widehat{\theta}_k \pi_k)^{-1}$. In ***Poulpe*** two additional methods are available for non-response correction.

1) The sampling weights $(\widehat{\theta}_k \pi_k)^{-1}$ are transformed into calibration (Calmar) weights (w_k) such that $\sum_{k \in r} w_k \mathbf{x}_k = \boldsymbol{\tau}_x$, where r is the sample of respondent. The variance estimators of $\widehat{\tau} = \sum_{k \in r} w_k y_k$ used is $\widehat{\text{var}}\,(\widehat{\tau}) = \widehat{\text{var}}\left(\sum_{k \in r} w_k u_k\right)$ where $u_k = y_k - \mathbf{x}_k^{'} \widehat{\mathbf{B}}$. Then a two-phase variance estimator is implemented (the second phase is a Poisson sampling design).

2) This method allows for an implicit correction of the non-response. The sampling weights $(\widehat{\theta}_k \pi_k)^{-1}$ are transformed into calibration (Calmar) weights (w_k) such that $\sum_{k \in r} w_k \mathbf{x}_k = \tau_x$. As far as the variance of $\widehat{\tau}$ is concerned, it is assumed that $p_k = r/n$ and a two-phase variance estimator is implemented (the second phase is a Poisson sampling design).

5.2.4 Adaptation of Design into the Software System

Identifying the strata and stratum sizes / sampling rates. All of the software expect that the observed data to be used *should have an identification variable for the strata*. The identification of strata takes place in procedures (***SAS, Sudaan***), in a macro (***Clan***), in a command '*svyset*' (***Stata***), in a screen '*create weights*' (***WesVar***), in a design data (***SPSS***), in a setup ('*sampling design identifiers*') containing design and variable information (***Bascula***) or in a design structure ('tree') (***Poulpe***). Furthermore, in order to calculate the finite population correction for stratified / non-stratified simple random sampling without replacement and in some cases to check the weights provided by the user, the software needs the stratum size(s). Some software accept the sampling rate(s) as well. Thus in practice *the user must always calculate / obtain the stratum sizes (sampling rates) in advance*, and in almost all cases these calculations must be merged into the observation data set.

* In ***SAS*** a separate option defining a data set containing the stratum sizes must be given, although the data set including observations can be given in that option as well (provided that the stratum size information is merged into the data set beforehand).

* ***SPSS*** requires a creation of a specific data set for the designs with an '*analysis preparation wizard*'. The stratum variable is given by the user in the design data; the variable should be in the observation data set as well.

* In ***Stata*** the user gives the variable name containing either the sampling rates (when values are smaller or equal to one) or the stratum sizes (when values are greater or equal to the sample size) in the observation data set. Note that this definition option

in the *svyset* command has the name *'fpc'*, though the alternatives which can be given are not the real finite population correction $(N_h - n_h)/N_h$.

* In ***Bascula*** a setup containing design and variable information must be created, and it includes also a menu for defining the 'population cell totals' ('*population tables*' screen), which must be inserted or imported from another source (the import can be carried out in that same menu).

* In ***Clan*** the variable containing the stratum sizes should be given in a macro.

* In ***Poulpe*** a sampling structure ('tree') containing all the stratification and cluster variables (together with information about the design and levels of study) is created. This data set is utilised throughout the process. In addition the stratum variable appears in some macros.

* In ***Sudaan*** the stratum size variable is given in a separate statement (*TOTCNT*). Because ***Sudaan*** allows multilevel stratification and/or clustering and it takes this feature into account in variance estimation, the order of the stratum/cluster size variables must correspond to the order of the stratum/cluster variables in the '*NEST*' statement.

* ***WesVar*** does not require the stratum sizes; the finite population correction can be dealt with by defining it in the '*attach factors*' screen.

An example of the sampling tree in Poulpe for FLFS

```
data v8.tree;
   set v8.treebasis;

      NSUP='AA'; NINF='AB'; TYPTIR='TOT'; output;
      NSUP='AB'; NINF='BA'; TYPTIR='EXH';
          ENTI = 'DUMMY STRATUM'; ENTAG = 'DUMMY STRATUM';
          output;
      NSUP='BA'; NINF='CB'; TYPTIR='SAS'; ENTI = ' ';
      ENTAG = 'DUMMY STRATUM' ; output;
       if nsup=' ' then delete;

run;
```

Identifying the clusters and number of clusters. The principles dealing with clusters follow strictly those given above concerning strata, when only one stratum variable and one cluster variable are given. In ***SAS***, ***SPSS***, ***Stata***, ***Bascula*** and ***WesVar*** the rest of the cluster levels may have impact in weights, but only the variance at the primary sampling unit level is taken into account in variance estimation. ***Sudaan*** calculates also the variance impact of the lower level clusters, so the frequency identification is needed at all levels. ***Poulpe*** follows the defined design structure through all levels and it takes that into account in variance estimation. ***Clan*** offers an indirect way of dealing with lower level clusters in variance estimation via additional weight terms together with some data processing.

5.2.5 Adaptation Auxiliary Information and Weights into the Software System

Identifying the post-strata and model variables, population parameters and marginal distributions. In ***Bascula*** the post-stratum and model information is defined in a weighting model. The population-level distributions for post-stratification and modelling can be given in the '*population tables*' screen (in the same way as the stratum sizes). ***Clan*** requires a separate data set including the variable names for post-stratification/calibration, the marginal distributions and the population parameters (if used in models) in a specific form in order to achieve correct variance estimation. Naturally these auxiliary variables must be found from the observation data set. Only post-stratification is available in ***Sudaan***. Post-stratification variable names (existing in the observation data set) are given in the procedure in the '*POSTVAR*' statement. The '*POSTWGT*' statement includes the marginal frequencies from the population level, e.g. when two post-stratum variables have 2 and 3 classes, the '*POSTWGT*' statement should have 6 numbers representing the joint marginal distribution of these two variables. ***WesVar*** has a specific screen for the definition of post-stratification. The post-stratum sizes together with a link variable for post-strata must be exactly expressed in a separate ASCII file, which can be read by the software. Correspondingly, raking has a specific screen for definitions.

Table 5.5: An Example of Marginal Distribution Data for Calibration of FLFS in Clan.

Obs	MAR1	MAR2	MAR3	MAR4	MAR5	MAR6	VAR	N
1	660495	660634	766775	829621	546590	435885	AGE10	6
2	1942052	1957948	.	.	.	.	GEN2	2
3	1055731	1374099	518263	520502	412525	18880	REG2	6

Weighting. Most of the software require that the weight to be used in estimation must be prepared in advance. Some software allow / assume more than one weight component.

* In the ***SAS*** survey procedures the name of the pre-prepared weight to be used must be given in the '*WEIGHT*' statement. The preliminary weight preparation is possible in the ***SAS*** system.

* In ***SPSS*** the weight must be in the observation data set, and the weight variable name is given when defining the design data. The preliminary weight preparation is possible in ***SPSS***.

* ***Stata*** requires the weight information to be in the observation data set, and the variable name is given in the '*svyset pweight*' command. The preliminary weight preparation is possible in ***Stata***.

* In ***Bascula*** the user creates a weighting model, i.e. what corrections shall be conducted to the inclusion weight in order to reach the final weight. The operations available for corrections are post-stratification, ratio, linear and multiplicative weighting. The inclusion weight variable is prepared beforehand (not possible in ***Bascula***), and the names of the correction weight and the final weight are given in ***Bascula*** for placing the corresponding weight values in them.

* ***Clan*** does not require the weights to be given. It uses the variables describing stratum sizes and the number of responded PSUs for weighting. If response homogeneity groups (RHG) are needed for non-response correction, also the sample sizes in the strata, the RHG identification variable and the variable containing respondents in RHGs are required. Apart from stratified simple random sampling without replacement of PSUs, one can introduce additional weights (including further unequalities in probabilities, e.g. concerning network sampling) via a separate weight option.

* ***Poulpe*** requires externally prepared weights to be given in the survey data. A specific macro ***Calmar*** is created for calibration purposes.

* The ***Sudaan*** procedures have the '*WEIGHT*' statement in order to specify the variable whose values are the analysis weights to be used in computing estimates. The statement can be omitted, if replicate weights are given for the BRR or the jackknife methods; then ***Sudaan*** computes the overall weight as the average of the replicate weights on each record. For the i.i.d. sampling situation (option SRS) the weights are not allowed. The preliminary weight preparation must be done outside ***Sudaan*** (e.g. in ***SAS,*** when using the SAS-callable version).

* ***WesVar*** requires one variable as a '*full-sample weight*' when creating a ***WesVar*** data file for estimation. Note that if post-stratification or raking are due to be taken into account in variance estimation, these operations must be conducted in ***WesVar***, not beforehand included in the full-sample weight. The preliminary weight preparation must be done outside ***WesVar***.

Replicate weights. All three software with replication methods (***Bascula***, ***Sudaan*** and ***WesVar***) can calculate the replicate weights. However, in ***WesVar*** and ***Sudaan*** replicate weights can be imported to the system (in ***Sudaan*** the '*REPDATA*' definition either for BRR or Jackknife). When the number of replicate weights exceeds 10 000, ***WesVar*** requires external resampling weights.

5.2.6 Reacting Problems in Data and Estimation

Unit non-response. Practically all survey data include unit non-response. It is a common practice that the researcher decides how the problem is dealt with at least when creating weights and additional estimation methods. The software do not recognize the unit non-response as such, because the systems cannot know what is the situation before carrying out the survey.

Item non-response. The missing values in some variables of the survey data observations have direct effect in estimation and analysis. When using the weights adjusted to the unit non-response, e.g. the total of the variable with item non-response might be underestimated in some software. The frequency tables with two or more dimensions might be problematic when including item non-response. If the class 'missing' is allowed in tabulations, then the results are easily interpreted. The way how the software react to classifications omitting the 'missing' class vary. The most common alternative is to exclude from the output any statistics for subgroup defined by a missing value on one of the categorical table variable. If any analysis variable or any variable used to define a computed statistics is missing, the default is to exclude the entire record from the request (in all software). A partial exclusion is another alternative, i.e. the use of all available data. Missing values are determined on a table-by-table basis, thus the cases used to compute statistics may vary across frequency or crosstabulation tables (in ***SPSS*** and ***Stata***). The procedure for imputation of item non-response, *PROC MI* in ***SAS***, is briefly described and tested with SHBS in Section 5.3.7. The following table shows the different ways of dealing with the estimator of the total when item non-response occurs (Poulpe not included in the calculations).

Test data for item non-response, weight problems and wrong stratum test

```
data testdata;
     input Dummy Id Stratum X Y Weight WeightMissing Weight0
          WeightNegative Weight0_1 StratumSize
             NumberOfRespondents
          WrongStratum WrongTotal;
     cards;
1 1 1  3  6  5  .  0 -2 0.7  20 4 1 20
1 2 1 25 12  5  5  5  5   5  20 4 1 20
1 3 1  .  7  5  5  5  5   5  20 4 1 20
1 4 1 12  5  5  5  5  5   5  20 4 1 20
1 1 2  1 44 25  .  0 -1 0.2 125 5 1  4
1 2 2  . 42 25 25 25 25  25 125 5 2  4
1 3 2  . 19 25 25 25 25  25 125 5 2  4
1 4 2 29  4 25 25 25 25  25 125 5 2  4
1 5 2 12  4 25 25 25 25  25 125 5 2  4
;
run;
```

Problems with weights. The software should clearly react to missing and zero weights. The calibration theory allows the weights to be even negative. Most of the software don't accept this kind of weights. The weights between zero and one appear in model assisted estimation as well as in PPS sampling with replacement. The following tables show the strategies of different software with problematic weight values (***Clan*** and ***Bascula*** are not included here, because they do not use external weights in the normal stratified simple random sampling situation; they create the weight internally.).

Table 5.6: Item non-response in the estimator of the total

Software	Overall result		Stratum 1		Stratum 2		Messages
	total, x	s.e.	total, x	s.e.	total, x	s.e.	
SAS	1250	604.6	200	85.7	1050	598.5	
SPSS	1250	609.9	200	88.3	1050	603.5	
Stata	1250	609.9	200	88.3	1050	603.5	(*) Some variables contain missing values
Bascula	1250	696.4	200	100.4	1050	689.1	
Clan	1250	696.4	200	100.4	1050	689.1	
Sudaan	1250	696.4	200	100.4	1050	689.1	
WesVar	1250	692.9	200	100.9	1050	685.5	Warning: 3 observations were excluded from the preceding table. These observations were excluded because they contained one or more requested variables with missing values.

Table 5.7: Missing weight values for the estimator of the total (variable WeightMissing)

Software	Overall result		Stratum 1		Stratum 2		Messages
	total, y	s.e.	total, y	s.e.	total, y	s.e.	
SAS	1845	883.6	120	28.8	1725	883.1	Log: Due to nonpositive weights, 2 observation(s) were deleted. Output: Number of Obs with Nonpositive weights 2
SPSS	1845	883.6	120	28.8	1725	883.1	
Stata	1845	883.6	120	28.8	1725	883.1	
Sudaan	1845	883.6	120	28.8	1725	883.1	
WesVar	1845	946.6	120	44.4	1725	945.6	one or more records have zero, missing or negative weight(s)

Table 5.8: Zero weight values for the estimator of the total (variable Weight0)

Software	Overall result		Stratum 1		Stratum 2		Messages
	total, y	s.e.	total, y	s.e.	total, y	s.e.	
SAS	1845	879.9	120	27.9	1725	879.4	Log: Due to nonpositive weights, 2 observation(s) were deleted. Output: Number of Obs with Nonpositive weights 2
SPSS	1845	883.6	120	28.8	1725	883.1	
Stata	1845	951.6	120	44.4	1725	950.6	
Sudaan	1845	883.6	120	28.8	1725	883.1	Number of observations skipped 2 (WEIGHT variable nonpositive)
WesVar	1845	946.6	120	44.4	1725	945.6	

Table 5.9: Negative weight values for the estimator of the total (variable WeightNegative)

Software	Overall result		Stratum 1		Stratum 2		Messages
	total, y	s.e.	total, y	s.e.	total, y	s.e.	
SAS	1845	879.9	120	27.9	1725	879.4	Log: Due to nonpositive weights, 2 observation(s) were deleted. Output: Number of Obs with Nonpositive weights 2
SPSS	1845	883.6	120	28.8	1725	883.1	
Stata	-	-	-	-	-	-	note: negative pweight(s) negative weights encountered
Sudaan	1845	883.6	120	28.8	1725	883.1	Number of observations skipped 2 (WEIGHT variable nonpositive)
WesVar	1789	942.2	108	47.9	1681	941.0	one or more records have zero, missing or negative weight(s)

Table 5.10: Weight values between 0 and 1 for the estimator of the total (variable Weight0_1)

Software	Overall result		Stratum 1		Stratum 2		Messages
	total, y	s.e.	total, y	s.e.	total, y	s.e.	
SAS	1858	947.7	124	41.4	1734	946.7	
SPSS	1858	947.7	124	41.4	1734	946.7	
Stata	1858	947.7	124	41.4	1734	946.7	
Sudaan	1858	947.7	124	41.4	1734	946.7	
WesVar	1858	947.7	124	41.4	1734	946.7	

User providing contradictory design information. The design information provided by the user (e.g. weights, stratum sizes, stratum identification, cluster sizes, cluster identification, sampling rates, number of respondents, finite population correction) may be in contradiction with each other. Three cases are studied here: 1) when the weights do not sum up to stratum sizes although the design requires; 2) when there is more than one weight within strata although there should not (here testing with a wrong stratum code); 3) when the given stratum size is smaller than the number of observations in the stratum, the software may react or may not. The following tables give results for the second and the third case, the first is dealt with in Table 5.10 for weight values between 0 and 1.

Table 5.11: Wrong stratum code (variable WrongStratum)

Software	Overall result		WrongStratum 1		WrongStratum 2		Messages
	total, y	s.e.	total, y	s.e.	total, y	s.e.	
SAS	2975	920.5	1250	920.5	1725	0	
SPSS	2975	920.5	1250	920.5	1725	0	Population size values are unequal or missing of st. 1 The first valid value found will be used.
Stata	-	-	-	-	-	-	fpc for all observations within a stratum must be the same
Bascula	-	-	-	-	-	-	The PSU fraction within stratum 1 (line 5) is not constant.
Clan	-	-	-	-	-	-	E R R O R NRESP > NPOP
Sudaan	-	-	-	-	-	-	Old and new TOTCNT are unequal
WesVar	-	-	-	-	-	-	stratum 2 has non-consecutive varunits for METHOD=JKn

Table 5.12: Stratum size smaller than size of sample in stratum

Software	Overall result		Stratum 1		Stratum 2		Messages
	total, y	s.e.	total, y	s.e.	total, y	s.e.	
SAS	-	-	-	-	-	-	Log: Population total 4 for stratum 2 in data set is smaller than the sample size 5.
SPSS	2975	27.8	150	27.8	-	-	
Stata	-	-	-	-	-	-	fpc must be <=1 if a rate, or >= no. sampled PSUs per stratum if PSU totals
Bascula	-	-	-	-	-	-	The sample total of cell (2) of model term Stratum exceeds the population total.
Clan	-	-	-	-	-	-	E R R O R NRESP > NPOP
Sudaan	-	-	-	-	-	-	Population count (4) is less than sample size (5)
WesVar	2975	1074.8	150	27.8	2825	1074.4	

Strata and domains with no PSUs or one PSU. The case of no sampling units in a stratum must be corrected by the user in order to have the expansion done to the level of the whole population. Usually this is taken into account in weights by collapsing strata (see the GMC test in Section 5.3.5) and providing the corrected information (weights, stratum identifications and frequencies) to the system. Only systems which also allow the original inclusion probabilities without non-response (***Bascula***, ***Clan***, ***Poulpe***) may notice the missing stratum. The one-PSU case is reflected in various ways, as it can be seen in the following table.

Table 5.13: One PSU in a stratum

Software	Overall result		Stratum 1		Stratum 2		Messages
	total, y	s.e.	total, y	s.e.	total, y	s.e.	
SAS	2945	1074.4	120	0.0	2825	1074.4	Log: Only one observation in a stratum for variable(s) Y. The variance of Y in that stratum is estimated by zero. Only one observation in a stratum in domain STRATUM for variable(s) Y. The variance of Y in that stratum is estimated by zero.
SPSS	2945	1074.4	120	0.0	2825	1074.4	
Stata	-	-	-	-	-	-	Stratum with only one PSU detected
Bascula	2945	1049.3	120	0.0	2825	1049.3	The sample file contains strata with just a single PSU. For variance estimation purposes Bascula has joined these strata together with neighbouring strata, a total of 1 times.
Clan	2945	1074.4	120	0.0	2825	1074.4	
Sudaan	-	-	-	-	-	-	Sample size is 1 but TOTCNT variable is 20
WesVar	-	-	-	-	-	-	stratum 1 has only 1 varunits for METHOD=JKn

Outliers. The outliers are not recognised by the software in the survey context. However, ***WesVar*** has an outlier test, which finds replicates, that are contributing a large percentage to the overall variance of the estimate. A studentized range test is used to assess the effect of the replicates on the variance. The studentized range is given as

$$q = \frac{1}{s}\left[\max\left(\widehat{\theta}_{(1)}, \ldots, \widehat{\theta}_{(G)}\right) - \min\left(\widehat{\theta}_{(1)}, \ldots, \widehat{\theta}_{(G)}\right)\right] ,$$

where

$\widehat{\theta}_{(g)}$ is the estimate from replicate $g(g = 1, \ldots, G)$,

s is the standard deviation of the replicate estimates.

For critical values c_1 and c_2, if inequality $c_1 < q < c_2$ is not satisfied, the maximum contribution to the variable by a replicate is given in the output and is approximated as

$$100\frac{\left(\widehat{\theta}_{(\max)} - \widehat{\theta}\right)}{Gs^2} ,$$

where

$$\widehat{\theta}_{(\max)} = \max\left(\widehat{\theta}_{(1)}, \ldots, \widehat{\theta}_{(G)}\right) ,$$

$\widehat{\theta}$ is the full-sample estimate,

G is the number of replicates.

5.2.7 Interface and Style of Output

Five types of interface are presented here, i.e. ***SAS***, ***SPSS***, ***Stata***, ***Bascula*** and ***WesVar***. ***Clan***, ***Poulpe*** and ***Sudaan*** function in the ***SAS*** environment.

SAS. The interface of SAS is based on three main screens (extended program editor, log window, output window). It is possible adjust the setting of the windows. The menus provide various properties for use; however, the survey procedures are not present in menus. A very useful feature for ***SAS*** programming is the syntax check based on colours representing different operations in the extended program editor. Erroneous programming can easily be seen as unusual colouring of letters on the editor.

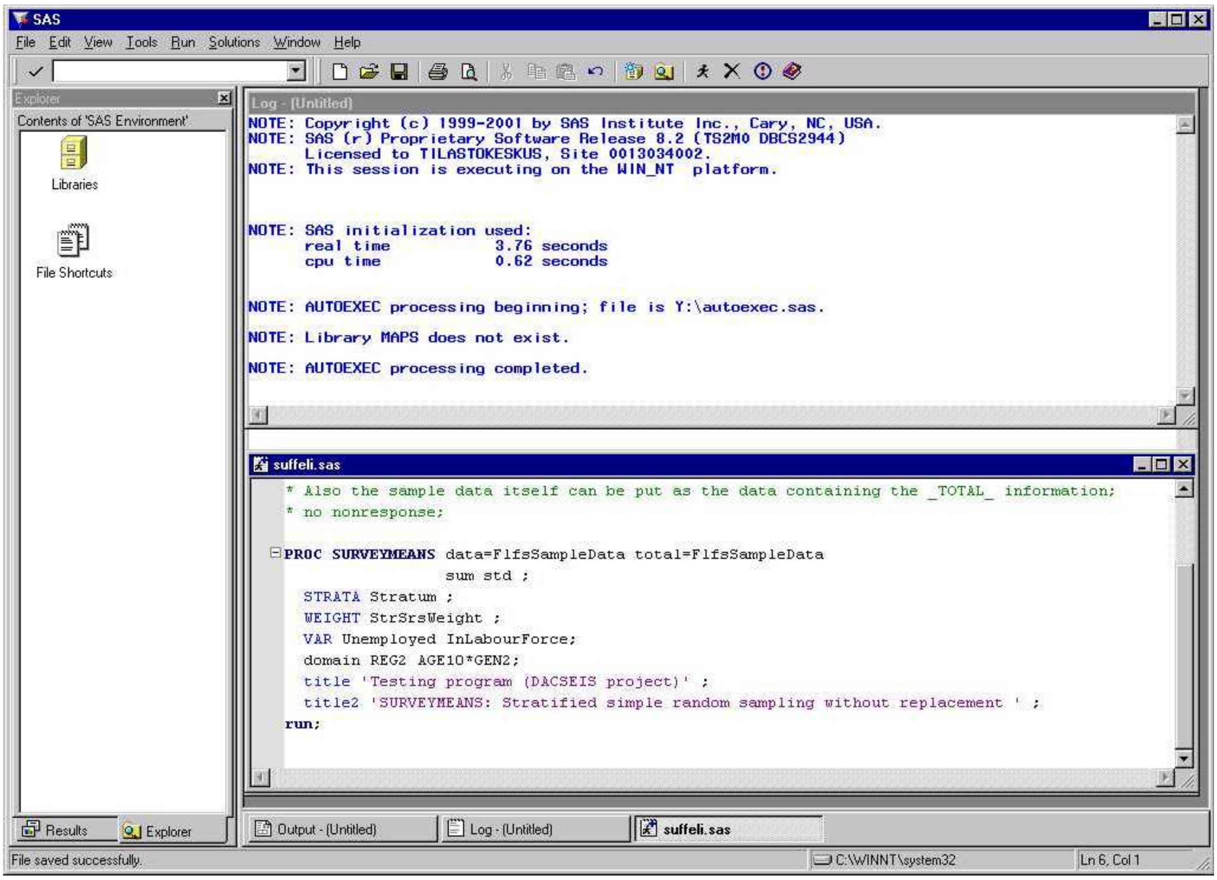


Figure 5.1: SAS interface with a program for FLFS

SPSS. Traditionally the main use of ***SPSS*** is based on menus, but recently the programming option has gained popularity, especially among users with repetitive tasks of similar kind. The data set is easily available, and results can be achieved with a few easy choices on menus. There is enough guidance for definitions of survey estimation, and the creation of the analysis plan (needed for estimation taking the sampling design into account) is rather easy.

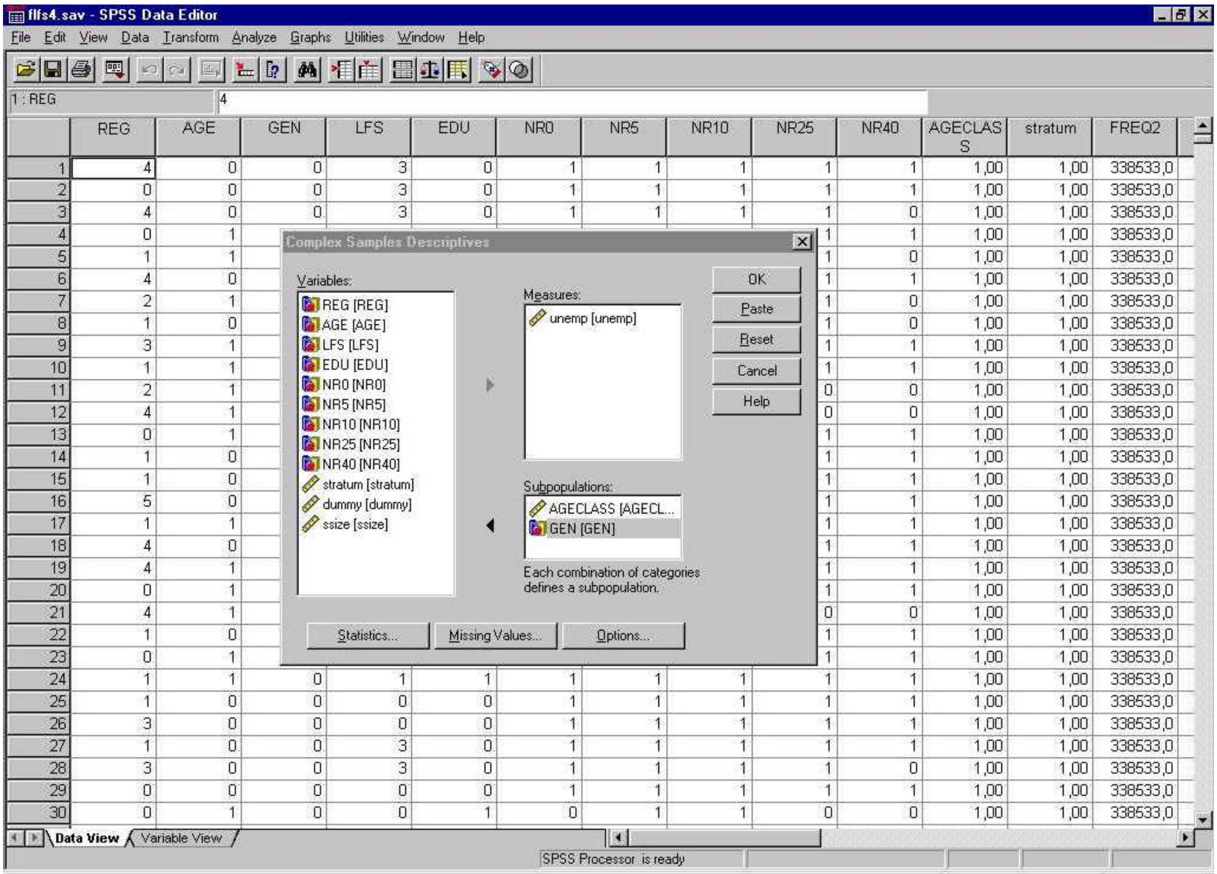


Figure 5.2: SPSS interface with menus

Stata. Two main alternatives exist for operations in ***Stata***: menus and commands. It seems that more experienced users apply the commands (given on a specific *'Stata command'* screen), but the menus are easy to be used as well. The *'variables'* screen is handy for definition operations. The *'Stata results'* screen provides both messages and output. Occasionally this mixed information might not be the best alternative as an output.

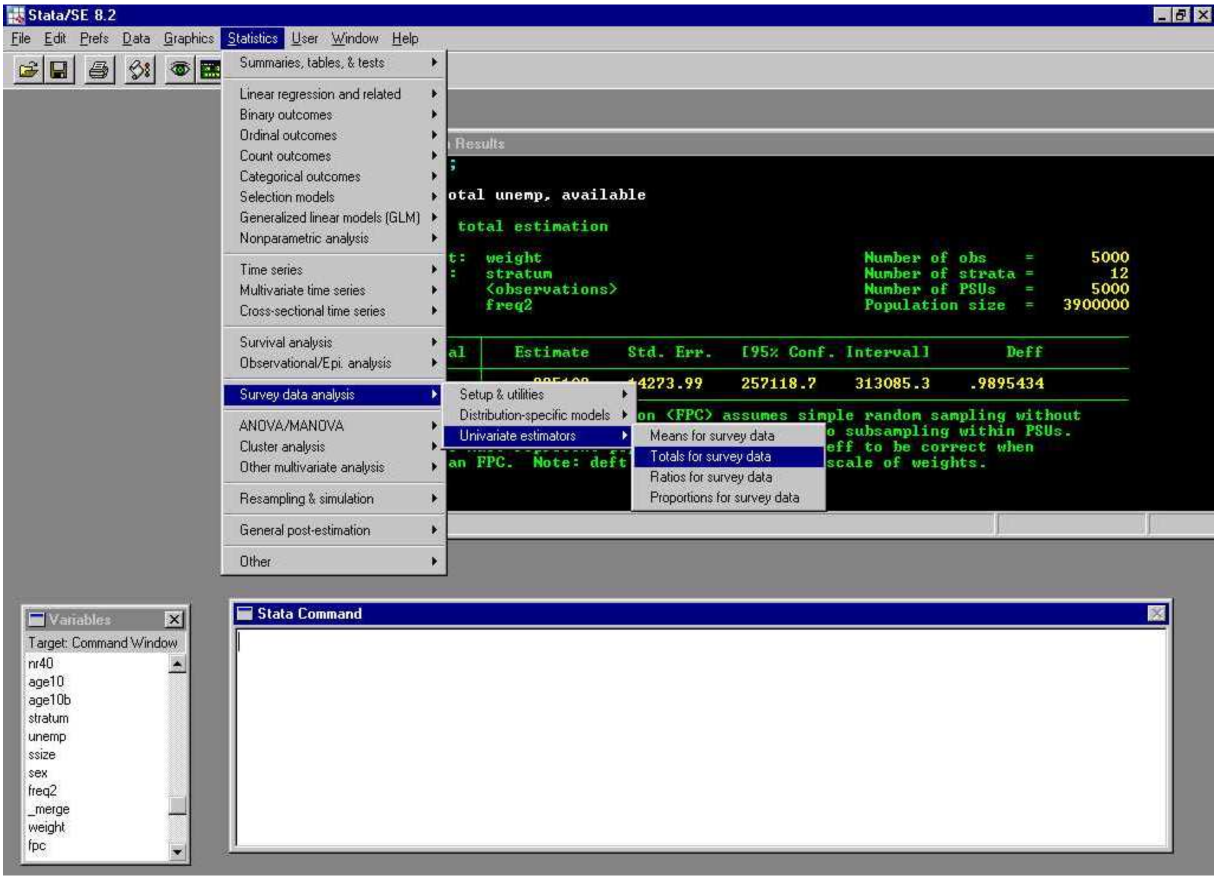


Figure 5.3: Stata interface with menus and screens

Bascula. ***Bascula*** functions in the ***Blaise*** environment as a survey estimation tool, which can be chosen on a menu. It starts from the survey data set in ***Blaise*** form, prepared earlier. The screens are in a logical order (it is easy to go back). Some screens appear only after some processes have been conducted (weighting model, weighting method). The interface serves the survey process well, although the non-advanced first-time user might be a bit confused at first. The given options of the process can be saved as a weighting model (.wif).

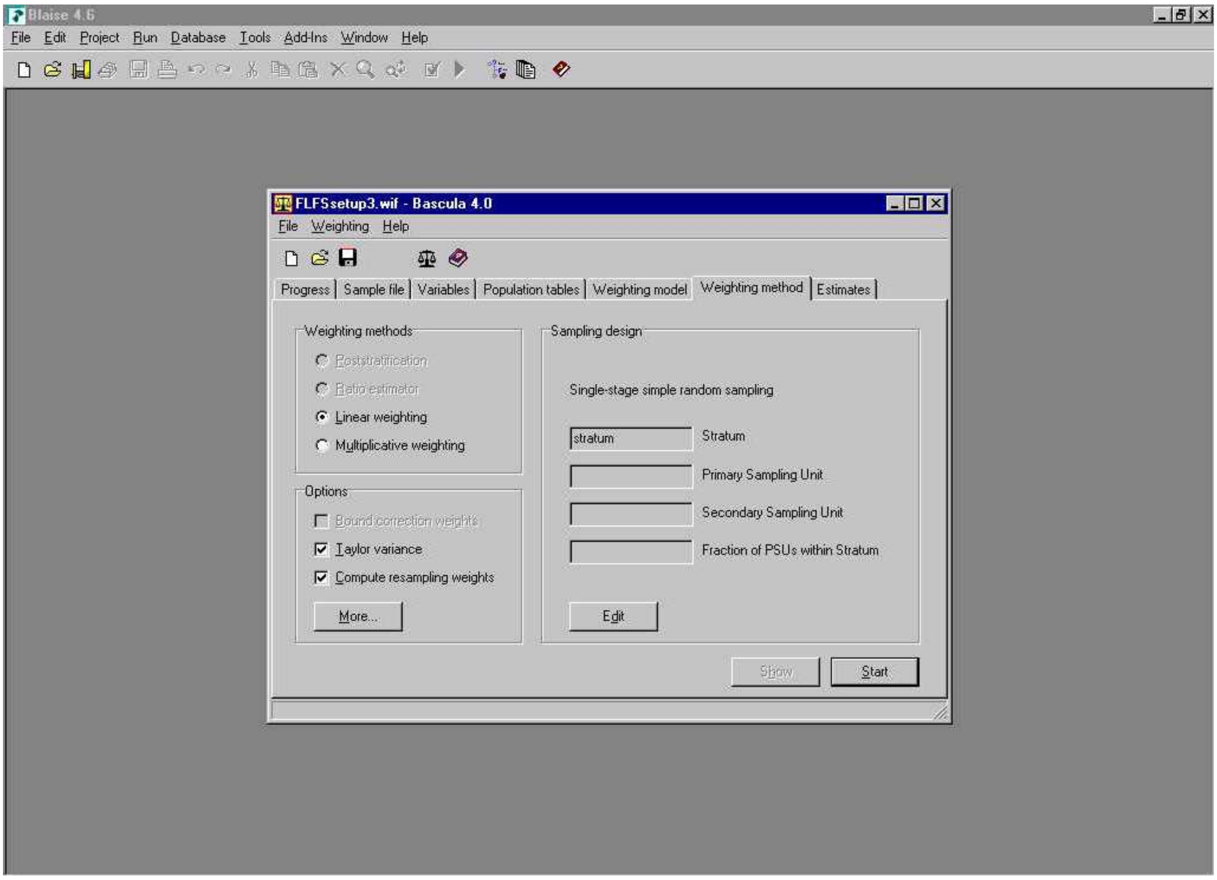


Figure 5.4: Bascula interface with menus/screens

WesVar. Screens, menus and buttons guide the user of ***WesVar*** in the definition and estimation process. ***WesVar*** includes two main parts: 1) the data and the definitions, 2) the workbook, which includes information about what shall be estimated at what levels of study. The logic of screens is a bit confusing at first. The change of windows (e.g. the move from the data file screen to the workbook screen) can be done in practice by using the window menu. Without knowing that the user might be 'lost' for a while. The 'job queue' window is the only place where the user can see whether the software is estimating or not.

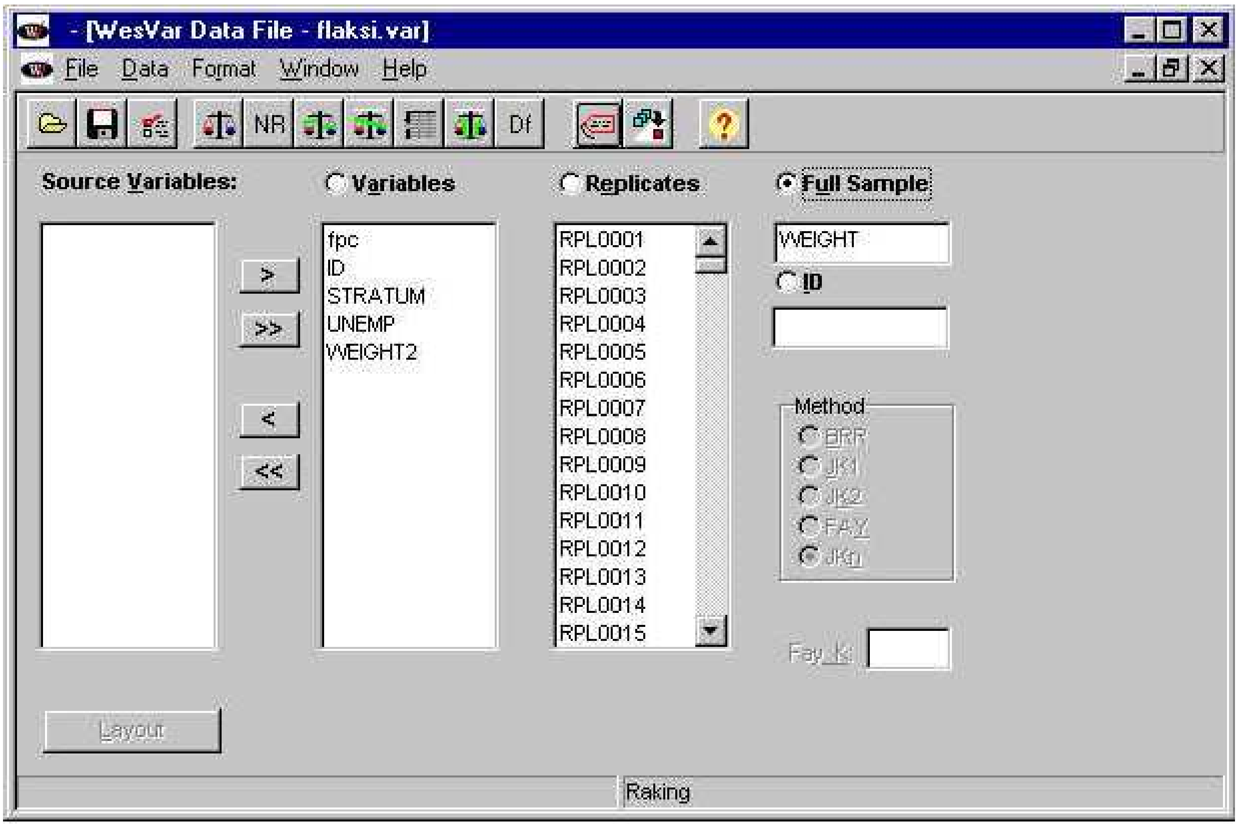


Figure 5.5: WesVar interface with menus/screens

These output examples are from the context of the DACSEIS data testing with FLFS. The comments can be found after the string of Figures.

SURVEYMEANS: Stratified simple random sampling without replacement

The SURVEYMEANS Procedure

Data Summary

Number of Strata	12
Number of Observations	5000
Sum of Weights	3900000

Statistics

Variable	Sum	Std Dev
UNEMPLOYED	285102	14274
INLABOURFORCE	2550312	20781

Domain Analysis: REG2

REG2	Variable	Sum	Std Dev
1	UNEMPLOYED	48974	6169.562832
	INLABOURFORCE	766288	21456
2	UNEMPLOYED	115648	9330.931043
	INLABOURFORCE	886461	22486
3	UNEMPLOYED	44053	5836.388389
	INLABOURFORCE	308796	14761
4	UNEMPLOYED	32570	5000.060784
	INLABOURFORCE	306982	14714
5	UNEMPLOYED	43857	5814.962323
	INLABOURFORCE	275541	14063
6	UNEMPLOYED	0	0
	INLABOURFORCE	6243.671512	2206.055546

Domain Analysis: AGE10*GEN2

AGE10	GEN2	Variable	Sum	Std Dev

Figure 5.6: SAS Surveymeans Output with FLFS estimation

Testing program (DACSEIS project)
CLAN: Stratified simple random sampling without replacement

Obs	ROW	COL	PTOT	STOT	PRATE	SRATE	PGTOT	SGTOT	PGRATE	SGRATE
1	1	1	48974.43	6169.56	0.06391	0.007843	49301.38	6105.23	0.06391	0.007843
2	2	1	115648.22	9330.93	0.13046	0.009960	116380.25	9109.06	0.13043	0.009958
3	3	1	44052.66	5836.39	0.14266	0.017618	42434.89	5413.97	0.14263	0.017618
4	4	1	32569.70	5000.06	0.10610	0.015484	33794.68	5037.47	0.10608	0.015481
5	5	1	43857.04	5814.96	0.15917	0.019447	41847.74	5289.81	0.15912	0.019449
6	6	1	0.00	0.00	0.00000	0.000000	0.00	0.00	0.00000	0.000000
7	7	1	285102.04	14273.99	0.11179	0.005489	283758.93	14171.12	0.11123	0.005443

Testing program (DACSEIS project)
CLAN: Stratified simple random sampling without replacement

Obs	ROW	COL	PTOT	STOT	PRATE	SRATE	PGTOT	SGTOT	PGRATE	SGRATE
1	1	1	38046.14	5123.23	0.24873	0.030814	37822.12	5084.03	0.24767	0.030657
2	2	1	35327.50	5028.29	0.23529	0.031039	34836.22	4958.60	0.23281	0.030769
3	3	1	32864.00	4708.63	0.10427	0.014883	32732.94	4686.67	0.10399	0.014835
4	4	1	26915.53	4422.57	0.10366	0.016841	26879.38	4414.50	0.10348	0.016802
5	5	1	34377.46	4951.10	0.09483	0.013606	34234.40	4926.09	0.09444	0.013540
6	6	1	36714.73	5201.58	0.11166	0.015696	36456.81	5165.46	0.11072	0.015571
7	7	1	21421.82	4017.31	0.05882	0.010986	21383.71	4011.65	0.05865	0.010958
8	8	1	34751.69	5013.68	0.09756	0.013977	34686.74	5004.63	0.09736	0.013950
9	9	1	14585.48	3172.97	0.11905	0.025002	14572.78	3169.39	0.11846	0.024882
10	10	1	10097.69	2751.75	0.08609	0.022844	10153.84	2767.22	0.08614	0.022858
11	11	1	0.00	0.00	0.00000	0.000000	0.00	0.00	0.00000	0.000000
12	12	1	0.00	0.00	0.00000	0.000000	0.00	0.00	0.00000	0.000000
13	13	1	285102.04	14273.99	0.11179	0.005489	283758.93	14171.12	0.11123	0.005443

Figure 5.7: Clan/SAS Output with FLFS estimation (printing 'dut' SAS data set)

Fichiers alloués dans l étape ESTIF1

Obs	FICHIERS	LIBRAIRI	TABLE
1	Données appelées en entrée	D:\DACSEIS\POULPE	R1CLF

Etape ESTIF1
résultats des calculs d'estimateurs
pour les variables fonctions demandées

Obs	VUNEMP	VINLABF	VUN_R1	VUN_R2	VUN_R3	VUN_R4
1	203746871.01	431845726.14	38063505.54	87066274.13	34063429.43	25000607.84

Obs	VUN_R5	VUN_R6	VRATE	VRATE_R1	VRATE_R2	VRATE_R3	VRATE_R4
1	33813786.82	0	.000030126	.000061507	.000099208	.000310396	.000239752

Obs	VRATE_R5	VRATE_R6	UNEMP	INLABF	UN_R1	UN_R2	UN_R3	UN_R4	UN_R5
1	.000378181	0	285102	2550312	48974.43	115648.2	44052.66	32569.7	43857.04

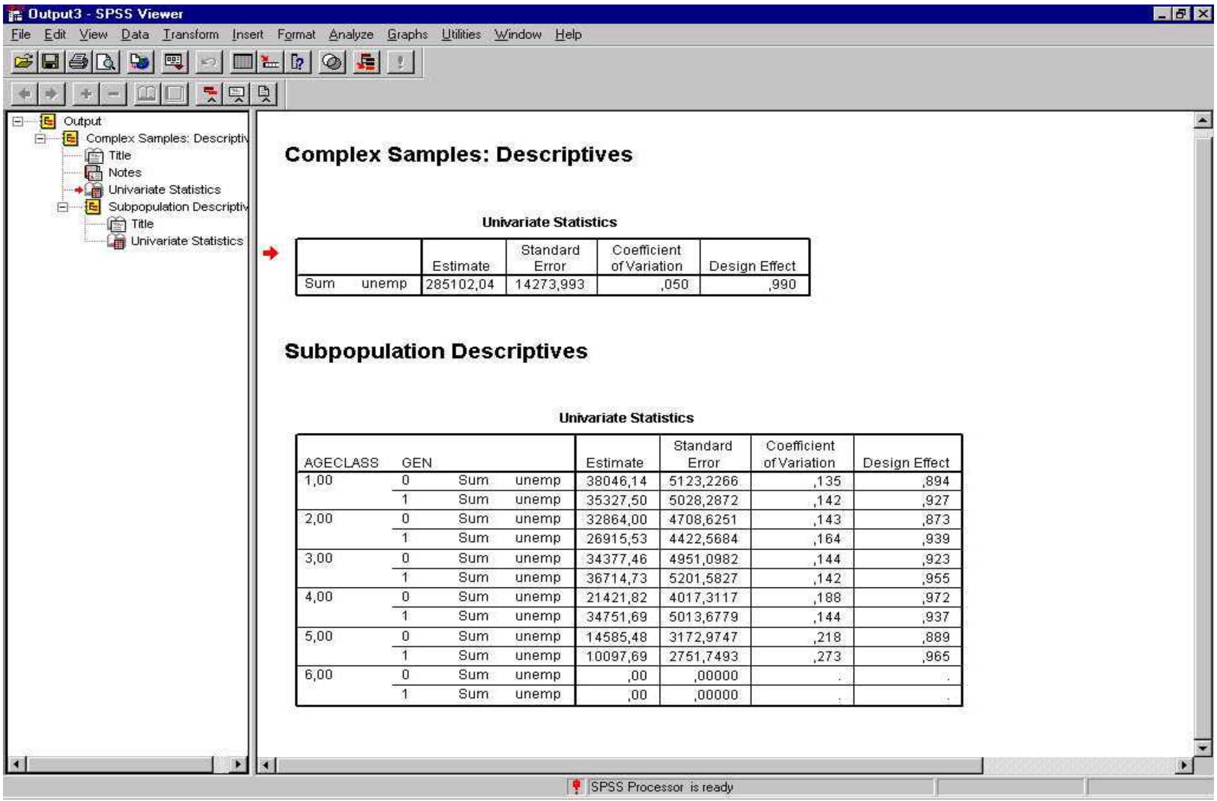
Obs	UN_R6	RATE	RATE_R1	RATE_R2	RATE_R3	RATE_R4	RATE_R5	RATE_R6
1	0	0.11179	0.063911	0.13046	0.14266	0.10610	0.15917	0

```
**********************
* Etape CHARLIS      *
**********************
```

Figure 5.8: Poulpe/SAS Output with FLFS Estimation (Variances)

```
Number of observations read    :   5000    Weighted count :   3900000
Denominator degrees of freedom :   4988


Date: 05-19-2004                 Research Triangle Institute                  Page  : 1
Time: 11:04:47                     The DESCRIPT Procedure                     Table : 1

Variance Estimation Method: Taylor Series (STRWOR)
by: Variable, REG2.

-----------------------------------------------------------------------------------------
Variable
   REG2                     Sample Size      Weighted Size           Total       SE Total
-----------------------------------------------------------------------------------------
UNEMPLOYED
   Total                           5000         3900000.00       285102.04       14273.99
   1                               1343         1048821.89        48974.43        6169.56
   2                               1749         1365253.94       115648.22        9330.93
   3                                691          538013.37        44052.66        5836.39
   4                                644          501603.42        32569.70        5000.06
   5                                555          432255.29        43857.04        5814.96
   6                                 18           14052.10            0.00           0.00
-----------------------------------------------------------------------------------------


Date: 05-19-2004                 Research Triangle Institute                  Page  : 2
Time: 11:04:47                     The DESCRIPT Procedure                     Table : 1

Variance Estimation Method: Taylor Series (STRWOR)
by: Variable, REG2.

-----------------------------------------------------------------------------------------
Variable
   REG2                            Mean            SE Mean    DEFF Mean #1   DEFF Total #1
-----------------------------------------------------------------------------------------
UNEMPLOYED
   Total                           0.07               0.00            0.99            0.99
   1                               0.05               0.01            1.01            1.01
```

Figure 5.9: Sudaan/SAS Output with FLFS estimation



Complex Samples: Descriptives

Univariate Statistics

		Estimate	Standard Error	Coefficient of Variation	Design Effect
Sum	unemp	285102,04	14273,993	,050	,990

Subpopulation Descriptives

Univariate Statistics

AGECLASS	GEN			Estimate	Standard Error	Coefficient of Variation	Design Effect
1,00	0	Sum	unemp	38046,14	5123,2266	,135	,894
	1	Sum	unemp	35327,50	5028,2872	,142	,927
2,00	0	Sum	unemp	32864,00	4708,6251	,143	,873
	1	Sum	unemp	26915,53	4422,5684	,164	,939
3,00	0	Sum	unemp	34377,46	4951,0982	,144	,923
	1	Sum	unemp	36714,73	5201,5827	,142	,955
4,00	0	Sum	unemp	21421,82	4017,3117	,188	,972
	1	Sum	unemp	34751,69	5013,6779	,144	,937
5,00	0	Sum	unemp	14585,48	3172,9747	,218	,889
	1	Sum	unemp	10097,69	2751,7493	,273	,965
6,00	0	Sum	unemp	,00	,00000	.	.
	1	Sum	unemp	,00	,00000	.	.

Figure 5.10: SPSS Output with FLFS estimation

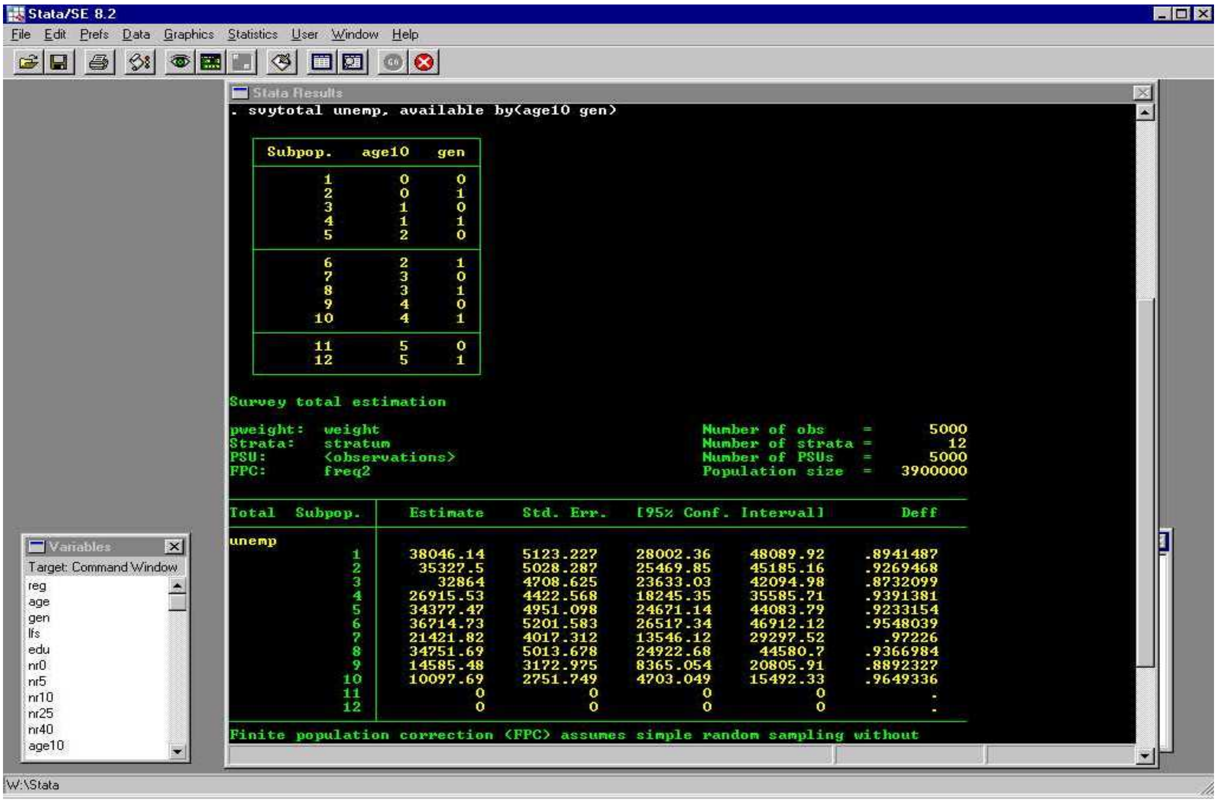


Figure 5.11: Stata Output with FLFS estimation

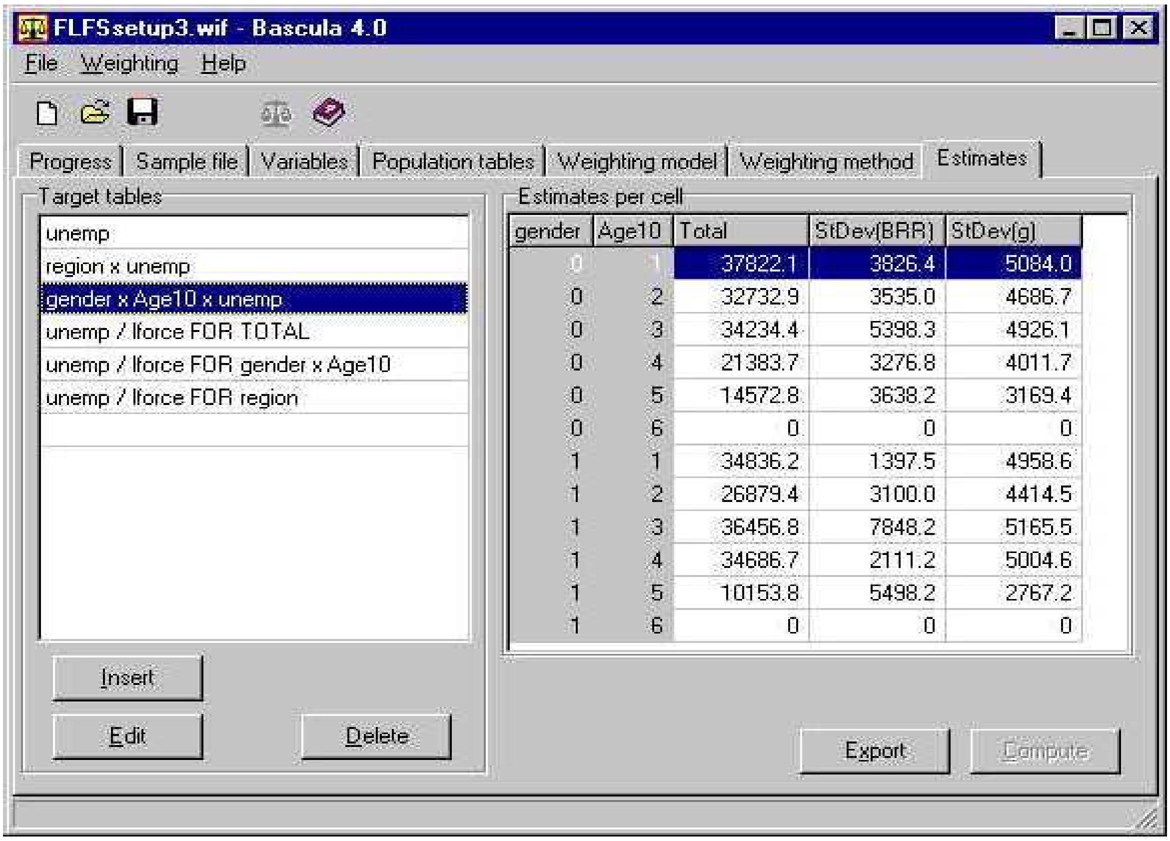
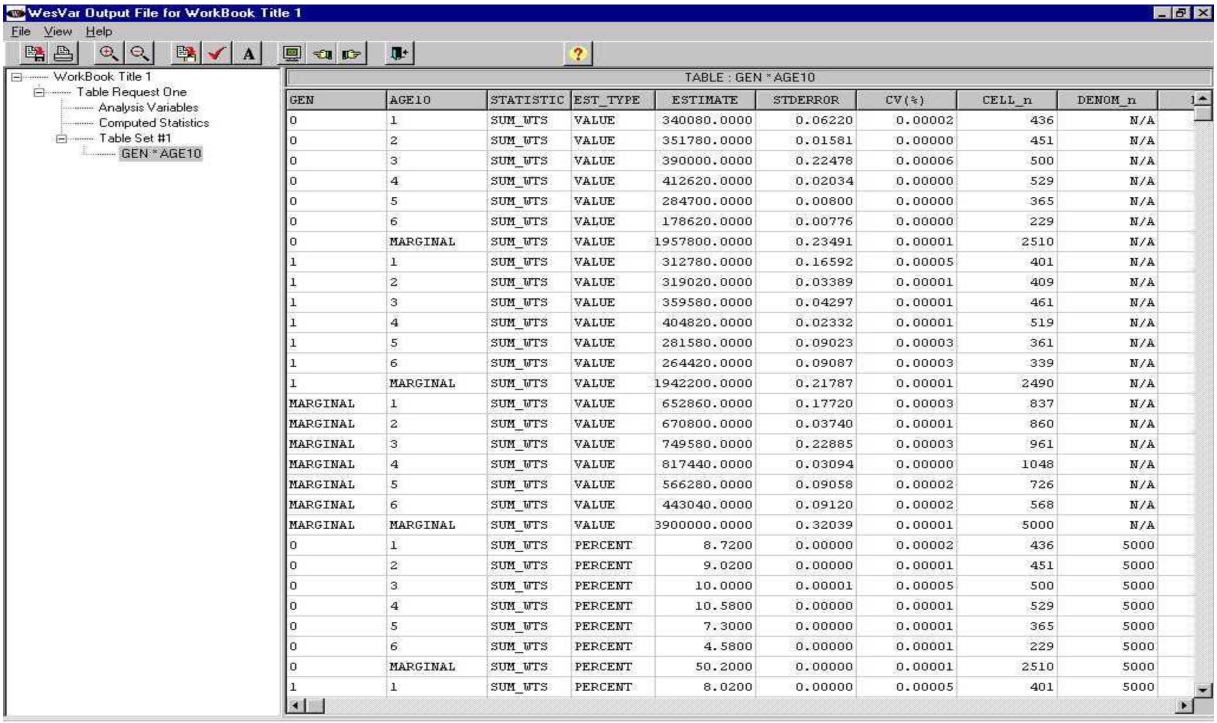


Figure 5.12: Bascula Output with FLFS estimation


WesVar Output File for WorkBook Title 1

File View Help

WorkBook Title 1
Table Request One
Analysis Variables
Computed Statistics
Table Set #1
GEN * AGE10

TABLE : GEN * AGE10

GEN	AGE10	STATISTIC	EST_TYPE	ESTIMATE	STDERROR	CV(%)	CELL_n	DENOM_n
0	1	SUM_WTS	VALUE	340080.0000	0.06220	0.00002	436	N/A
0	2	SUM_WTS	VALUE	351780.0000	0.01581	0.00000	451	N/A
0	3	SUM_WTS	VALUE	390000.0000	0.22478	0.00006	500	N/A
0	4	SUM_WTS	VALUE	412620.0000	0.02034	0.00000	529	N/A
0	5	SUM_WTS	VALUE	284700.0000	0.00800	0.00000	365	N/A
0	6	SUM_WTS	VALUE	178620.0000	0.00776	0.00000	229	N/A
0	MARGINAL	SUM_WTS	VALUE	1957800.0000	0.23491	0.00001	2510	N/A
1	1	SUM_WTS	VALUE	312780.0000	0.16592	0.00005	401	N/A
1	2	SUM_WTS	VALUE	319020.0000	0.03389	0.00001	409	N/A
1	3	SUM_WTS	VALUE	359580.0000	0.04297	0.00001	461	N/A
1	4	SUM_WTS	VALUE	404820.0000	0.02332	0.00001	519	N/A
1	5	SUM_WTS	VALUE	281580.0000	0.09023	0.00003	361	N/A
1	6	SUM_WTS	VALUE	264420.0000	0.09087	0.00003	339	N/A
1	MARGINAL	SUM_WTS	VALUE	1942200.0000	0.21787	0.00001	2490	N/A
MARGINAL	1	SUM_WTS	VALUE	652860.0000	0.17720	0.00003	837	N/A
MARGINAL	2	SUM_WTS	VALUE	670800.0000	0.03740	0.00001	860	N/A
MARGINAL	3	SUM_WTS	VALUE	749580.0000	0.22885	0.00003	961	N/A
MARGINAL	4	SUM_WTS	VALUE	817440.0000	0.03094	0.00000	1048	N/A
MARGINAL	5	SUM_WTS	VALUE	566280.0000	0.09058	0.00002	726	N/A
MARGINAL	6	SUM_WTS	VALUE	443040.0000	0.09120	0.00002	568	N/A
MARGINAL	MARGINAL	SUM_WTS	VALUE	3900000.0000	0.32039	0.00001	5000	N/A
0	1	SUM_WTS	PERCENT	8.7200	0.00000	0.00002	436	5000
0	2	SUM_WTS	PERCENT	9.0200	0.00000	0.00001	451	5000
0	3	SUM_WTS	PERCENT	10.0000	0.00001	0.00005	500	5000
0	4	SUM_WTS	PERCENT	10.5800	0.00000	0.00001	529	5000
0	5	SUM_WTS	PERCENT	7.3000	0.00000	0.00001	365	5000
0	6	SUM_WTS	PERCENT	4.5800	0.00000	0.00001	229	5000
0	MARGINAL	SUM_WTS	PERCENT	50.2000	0.00000	0.00001	2510	5000
1	1	SUM_WTS	PERCENT	8.0200	0.00000	0.00005	401	5000

Figure 5.13: WesVar Output with FLFS estimation

The informativeness of the outputs vary, mainly because of the amount of properties available in the software. ***Clan*** provides only very basic information in a SAS data set *'dut'*, in practice the estimates and their standard errors at the requested levels of study. ***Bascula*** follows the same principle (in many cases providing two alternative standard errors, i.e. Taylor linearisation and BRR). However, the output is clearly Blaise-oriented. The output of ***Bascula*** can be exported as a text file. ***Poulpe*** provides a large amount of information on the output screen of ***SAS***. Several results and checks are included, and this makes the study of the results somewhat complicated. The headings and messages in French cause difficulties for non-French speaking researchers. Both ***SAS*** and ***Sudaan*** have quite a fixed pattern for output, and the style of ***Sudaan*** is a bit old-fashioned, though informative. ***SPSS*** provides an output form which is compact and easy to be edited. ***Stata*** shows the results on the *'Stata results'* screen. The black background may be disturbing for some users. ***WesVar*** provides a lot of information as a default. However, in practice many of these results are not necessary for the user.

5.2.8 Comparing Survey Processes in Different Software

Following the structure presented in Section 2.4.2, the survey processes of both three general software and five specialised software are described in a comparable table form, providing the process as a continuation emphasising the phases which are vital for successful estimation and variance estimation.

Table 5.14: Survey Processes in Three General Software

SURVEY PROCESSES IN THREE GENERAL SOFTWARE¤		
Importing: ***survey data, population level information*** ¶ If not entered in the software, the survey data must be imported in some format recognisable for the software. If not included in the survey data, the stratum sizes and other population level information must be imported as well.¤		
SAS. The survey data set is imported by using the '*import*' menu, in a procedure ('*Import*'), or by reading an ASCII data in a data phase in the program code. Furthermore, the population data can be entered into a new SAS file as well or it can be created within a '*Data*' phase in programming. The population data is merged into the survey data.¤	*SPSS.* The survey data set is imported by using the '*open/data*' menu or in the program code with '*import*'. Furthermore, the population data can be entered into a new SPSS file as well. The population data is merged into the survey data.¤	*Stata.* The survey data set (in ASCII or '*SAS Xport*') is imported by using the '*import*' menu or e.g. with the commands '*insheet*' or '*infile*'. For importing many other data formats the '*StatTransfer*' package is recommended. Furthermore, the population data can be entered into a new Stata file as well. The population data is merged into the survey data.¤
Comment: In general the data import from other systems succeeds rather well in these software. The most demanding form is ASCII with various delimiters and record structures. In all software merging data sets requires sorting, which takes time in large data.¤		
Survey data processing: ***weight variable, classifying, recoding*** ¶ Some data preparations are needed before the estimation process: the weight is created, some new classification variables are created and recoding of variables is conducted¤		
SAS. The location of the survey data set (i.e. a SAS library) is defined. Generating survey data summary statistics with a procedure ('*Summary*', '*Means*'). Merging the summary data into the survey data ('*Data*' phase). Calculating weight with the terms available in the '*Data*' phase in programming. A new variable *_TOTAL_* is created for stratum sizes (required by the survey procedures, *_RATE_* if sampling rates are provided). ¤	*SPSS.* The SPSS data set to be used is opened. Generating survey data summary statistics with the menu '*descriptive statistics*' or in the program code ('*Frequencies*'). Merging the summary data into the survey data. Calculating weights with the terms available.¤	*Stata.* The Stata data set to be used is opened. Generating survey data summary statistics into the data set by either using a menu ('*create or change variables*') or commands (e.g. '*generate*', '*extended generate*'). Calculating weights with the terms available. ¤
Comment: Stata skips here the sometimes time-consuming merging phase; it generates the summary statistics straight into the data set. Calculations, classifications and recoding are carried out in different ways. In SAS the functions are given in the '*Data*' phases of the program. In Stata a large variety of different functions are available in menus. In SPSS calculations can be done either by using menus or programming.¤		
Preliminary sampling design definition¶ The menu-based systems require preliminary definitions of the sampling design before estimation and variance estimation.¤		
SAS. No preliminary definition of the sampling design.¤	*SPSS.* An analysis plan data (variables for strata, clusters, weight, stage label) is created with the menu '*prepare for analysis*' or in the program code ('*Csplan analysis*'). This data can be used in subsequent analysis and in other sessions as well.¤	*Stata.* Defining the sampling design (variables for stratum, weight, fpc, psu) with the menu '*set variables for survey data*' or by using the command '*svyset*'. This definition is the basis for estimations during the session until a new definition is set. ¤
Comment: For surveys repeatedly conducted the possibility to directly use the previous design structure is an advantage (SPSS plan data, SAS programming). ¤		
Computing estimates and variance estimates at the requested levels of study¶ Together with the estimators and the levels of study the user can define some properties concerning the data, missing values and the output of the results.¤		
SAS. The survey procedures (e.g. '*Surveymeans*') have the necessary design information for calculations given in the statements and options. ¤	*SPSS.* Frequencies, descriptive statistics, crosstabulations and ratios can be calculated either via the '*complex samples*' menu or by programming (commands with a '*CS*' beginning). ¤	*Stata.* Defining the parameters to be estimated together with the levels of study (menu '*univariate estimators*' or commands '*svytotal*', '*svymean*', '*svyratio*', '*svyprop*') and executing the menu/command. ¤
Comment: All software provide the possibility for data reduction, inclusion / exclusion of observations containing missing values and many statistics concerning estimation (e.g. standard error, confidence interval). The SAS procedures have most statistics, but the design effect is not available. On the other hand, Stata does not provide the coefficient of variation in its survey part. ¤		

Table 5.15: Survey Processes in Three SAS-Based Specialised Software

SURVEY PROCESSES IN THREE SAS-BASED SPECIALISED SOFTWARE		
Importing/Processing/Calculating additional variables: ***survey data, population level information*** This phase is conducted in the SAS system.		
SAS (for Clan, Poulpe, Sudaan). The survey data set is imported by using the '*import*' menu, in a procedure ('*Import*'), or by reading an ASCII data in a data phase in the program code. Furthermore, the population data can be entered into a new SAS file as well or it can be created within a '*Data*' phase in programming. The population data is merged into the survey data. For Clan and Poulpe the marginal distributions and parameters of the population must be calculated as a separate SAS file, if calibration will be conducted.		
Survey data processing: ***weight variable, classifying, recoding*** Some data preparations are needed before the estimation process: the weight is created, some new classification variables are created and recoding of variables is conducted		
SAS (for Clan, Poulpe, Sudaan). The location of the survey data set (i.e. a SAS library) is defined. Generating survey data summary statistics with a procedure ('*Summary*', '*Means*'). Merging the summary data into the survey data ('*Data*' phase). Calculating weight with the terms available in the '*Data*' phase in programming. Note that for Clan, Poulpe and Sudaan one can utilise also frequency information from the sample level (including non-respondents) for non-response correction.		
Clan. The weights are created in Clan based on the summary statistics, so no preliminary weight calculation is needed. Classifying and recoding of variables are conducted in the SAS system, but obtaining subgroup results in Clan requires some program rows in a Clan macro %AUXVAR.	***Poulpe.*** These are prepared in the SAS system. Obtaining subgroup results in Poulpe requires making dichotomous 0/1 variables for every subgroup.	***Sudaan.*** There are tools for classifying and recoding in Sudaan, but the weights must be prepared in the SAS system.
Preliminary sampling design specification Some software require preliminary preparations describing the sampling design before the actual estimation procedure or macro.		
Clan: Estimators for some more complex design situations (e.g. two-phase sampling and rotational design) can be defined in the %AUXVAR macro.	***Poulpe:*** A specific sampling tree data with fixed-name variables is created in order to give detailed information how every phase of estimation and its variance estimation is calculated in Poulpe. This mechanism enables the large variety of different designs at many levels available for estimation in Poulpe.	***Sudaan:*** All specifications in the procedures.
Specifying weighting model and estimation function The weighting model or estimation function must be defined in some software (possibly with auxiliary information).		
Clan: The estimation function and the name of the pre-defined data set including the marginal information for calibration is given in the %AUXVAR macro. The (chain of) estimation functions utilise four macro functions (ADD, SUB, MUL, DIV) for achieving the function of totals. Two macro alternatives are available for estimation: %TOT for Horvitz-Thompson estimation and %GREG for GREG estimation. Also parameters concerning the GREG estimator to be created can be given here as well as additional options for non-response treatment.	***Poulpe:*** The marginal data set is renamed to a form known by Poulpe ("GEO"). The (chain of) estimation functions utilise four macro functions (ADD, SUB, MUL, DIV) for achieving the function of totals (in the %FONCTION macro). A lot of additional options concerning the design, estimation and methodology can be defined in this phase (macro %LANCEV2).	***Sudaan:*** All specifications in the procedures.
Computing weights, estimates and variance estimates at the requested levels of study (with sampling design / variance estimation method specification) Together with the estimators and the levels of study the user can define some properties concerning the data, missing values and the output of the results.		
Clan: The macro %CLAN conducts the final calculations of the weights, estimates and standard errors. The stratum/cluster identification variables and summary statistics variables from different levels (population, sample, respondents) are given in the macro for basic weighting. Furthermore, the macro utilises the definitions given in the %AUXVAR macro for the calibration and/or the calculations.	***Poulpe:*** This phase utilises the definitions given before. The librarys where the data sets to be used are located are defined. The name of the data set, the weights, the variables to be dealt with as well as the levels of study (dichotomous variables) are given (macro %CHARLIS).	***Sudaan:*** The variety of procedures include the fixed structure of design definition and level of study statements: NEST for stratification and clustering, TOTCNT for totals in strata and clusters, SAMCNT for sample sizes in strata and clusters, SUBGROUP for domains, LEVELS for the number of domain groups in each domain variable, POSTVAR and POSTWGT for post-stratification. The sampling design and the variance estimation method are given in the option DESIGN; the jackknife and the BRR are assuming with-replacement sampling. Furthermore, the analytical procedures include a lot of method specific options, statements and functions.

Table 5.16: Survey Processes in Bascula and WesVar

SURVEY PROCESSES IN BASCULA AND WESVAR	
Processing data / Calculating additional variables: ***survey data, population level information*** This phase is conducted in some other system (e.g. SAS, SPSS, Stata).	
For both Bascula and WesVar need the basic identification variables for stratification and clustering. WesVar needs the identification variable for the observation as well. Bascula needs also the original inclusion weight, and WesVar the sampling weight.	
Metadata preparation In some cases pre-definitions of the variables are required in the system in which the software works.	
Bascula: The Blaise system must be informed what kind of variables will be imported in a form of a metadata.	***WesVar***: No metadata preparation.
Importing survey data into the system There are different ways of importing the survey data into the system, provided that the data are not made in the system.	
Bascula: Importing of the ASCII data is conducted in the Blaise system by using the metadata definitions and the Manipula tool for the import.	***WesVar***: The data can be imported in ASCII and in some SPSS and SAS forms. For ASCII the layout must be given.
Selecting and/or defining variables and weights Depending on the software, the variables in question are either selected via menus or defined in the software or its backing system.	
Bascula: The variables to be used in Bascula are defined on the Variables screen (e.g. categorical variables, inclusion weight).	***WesVar***: In the WesVar Data File section the user defines the variables to be used and the weight variable.
Importing population information Information from the population level (e.g. stratum/cluster frequencies, marginals of auxiliary information) is vital for design-based variance calculations as well as for the case of utilising auxiliary information in estimation and variance estimation.	
Bascula: The population information can either be written manually or imported from a data set on the Population Tables screen. There can be several population tables.	***WesVar***: The finite population correction is added on a separate menu. The system needs the population totals only for post-stratification.
Specifying weighting model, post-stratification, non-response adjustment and variance estimation method The weighting model or estimation function must be defined in some software (possibly with auxiliary information).	
Bascula: The weighting model can be given on the Weighting Model screen. After checking the contents of the variables in question Bascula allows the specification of the method on the Weighting Method screen (post-stratification, ratio estimator, linear weighting, multiplicative weighting). The identification variables of the design and the model are given as well as the variance estimation method (Taylor/BRR).	***WesVar***: There are five main variance estimation methods available in the WesVar Data File section (BRR, non-stratified Jackknife-1, two-units per stratum Jackknife without and with Fay's correction and stratified Jackknife-1). Non-response adjustments and post-stratification are given in the WesVar Data File Section as well (defining the categorisation to be used and other features).
Computing weights and/or replicate weights The phase when the weights are computed (if not already existing) varies from one software to another. Some systems create the weights internally, either automatically (based on the design information available) or with the request of the user.	
Bascula: The system checks whether the data and the definitions allow weighting (e.g. some inconsistencies cause the program to warn the user and stop here). When everything is ok the final weights are calculated.	***WesVar***: The sampling weight variable is prepared in advance and given as a variable. WesVar can create the replicate weights for the jackknife and the BRR following the method instructions given by the user.
Defining the estimation function, computing estimates and variance estimates at the requested levels of study Together with the estimators and the levels of study the user can define some properties concerning the data, missing values and the output of the results.	
Bascula: The variables and the requested levels of study are given on the Estimates screen. Several tables ("target tables" in Bascula) can be given.	***WesVar***: In the WesVar Workbook section the user can define the estimates to be calculated and the levels of study. The structure in the Workbook provides a lot of alternatives for varying output needs.

5.3 Testing with the Sample Data Sets Selected from the DACSEIS Universe

5.3.1 Descriptions of Samples, Their Designs and Parameters to Be Studied

Surveys chosen for testing. It was decided that the *Finnish Labour Force Survey (FLFS)*, the *Swiss Household Budget Survey (SHBS)* and the *German Microcensus (GM)* were the surveys selected for testing. The design of the FLFS with a categorical study variable (*Status in labour force*) is simple as such, but one small region together with non-response provides some challenge for variance estimation. The SHBS sample (selected from the pseudo-universe of households) includes two continuous study variables (household income and expenditure) within a rather simple design. The socio-economic status together with the type of household as levels of study would provide interesting domain results. The one-stage stratified area sample of the German Microcensus (study variable *Employment*) is a challenge for variance estimation in the software, e.g. having strata with only one cluster. The very large sample size of the survey (831,004 persons, divided into 16 federal states) pushes the software to the extreme. Descriptions of three different samples are presented below.

Finnish Labour Force Survey

* Variables available in the sample:

  Region (6),

  Ageclass (12), *Agegroups* (from 15-19 to 70-74),

  Gender (2),

  Status in labour force (8: employed, unemployed, conscripts, students, disabled, pensioners, persons performing domestic work, others),

  Education (7: no answer, upper secondary education, post-secondary non-tertiary education, 5B-programmes, 5A-programmes, second stage of tertiary education, level unknown).

* Parameters to be estimated: *total number of unemployed, unemployment rate*

* Stratification[1]: *joint Agegroups (0,1), (2,3), . . . , (10,11) cross Gender.*

* Levels of sampling: *persons*

* Sample size: $5,000$

German Microcensus

* Variables available in the sample:

  Federal state (16: Schleswig - Holstein, Hamburg, Niedersachsen, Bremen, Nordrhein - Westfalen, Hessen, Rheinland - Pfalz, Baden - Württemberg, Bayern, Saarland, Berlin, Brandenburg, Mecklenburg - Vorpommern, Sachsen, Sachsen - Anhalt, Thüringen),

  Regional class (214 in all),

  House size class(5: small buildings, medium-sized buildings, huge buildings, community accommodation / institutions, new buildings),

  Sample area (internal number within each fed/reg/house size combination), *Household number* (internal number within each fed/reg/house size combination),

[1]In the original survey post-stratification.

Household size from 1 to 75,

Age from 0 to 95+,

Gender (2),

Ethnicity (3: German, EU foreigner, Non-EU foreigner),

Duration of job seeking (4: missing or non-seeking, up to 6 months, 6 to 12 months, more than 12 months),

Employment (3: employed labour force, unemployed labour force, non labour force),

Reported at the job centre (2: employed, unemployed).

* Parameters to be estimated: *total number of unemployed, unemployment rate*
* Stratification: *Federal state* cross *Regional class* cross *House size class*
* Levels of sampling: *groups of dwellings, households, persons (each group of dwellings as a whole)*
* Sample size total: $831,004$ (from $6,791$ to $173,654$ in Federal states)

Swiss Household Budget Survey

* Variables available in the sample:

  Region (7: Plateau central, Region lemanique, Zurich, Suisse du Nord-Ouest, Suisse orientale, Suisse centrale, Tessin),

  Income,

  Expenditure,

  Type of household (8: person alone, mother/father alone + 1 child, mother/father alone + 2 or more children, couple, couple with 1 child, couple with 2 children, couple with 3 or more children, other),

  Socio-economic status (6: other, salaried, independent [without farmer], farmer, unemployed, pensioner)
* Parameters to be estimated: *mean income, mean expenditure*
* Stratification: *Region*
* Levels of sampling: *households*
* Sample size: $15,434$

5.3.2 Choices for Testing: Estimators, Auxiliary Information, Variance Estimation Methods, Non-response Mechanisms and Levels of Study

Section 2.4.2 describes the starting point for the testing process. Here we decide the estimators and the methodologies behind them that will be tested in situations including a full sample and some non-response. Some levels of study are defined as well. The decisions were made in collaboration with the experts of the corresponding surveys and the group responsible for the methodological collection in workpackage 1 in order to mirror the real-life situation as well as possible.

Properties tested when existing. Some methods are always tested when they appear in the software. In some software there are *several possible methods available for variance estimation.* All these methods should be tested if the design in question allows it. For the *correction of non-response there may be a method* outside the ignorability principle (see Section 2.3.3), post-stratification and calibration (in practice the response homogeneity group method). This should be tested as well. At least one software includes *imputation (and variance estimation for it)* in its system. This method available for imputation is a target for testing.

<u>Finnish Labour Force Survey</u>

* Types of estimators: Horvitz-Thompson estimator, calibration estimator
* Auxiliary information: *Region*, *Gender*, joint *Agegroups*
* Non-response mechanism: non-response at the original level, non-response of 5%
* Levels of study: *Region*, *Gender* cross *joint Agegroups*

<u>German Microcensus</u>

* Types of estimators: Horvitz-Thompson estimator, general regression estimator
* Auxiliary information: *Regional class, Gender* cross *Nationality (German / non-German)*
* Non-response mechanism: Non-response of 5%
* Levels of study: *5-year Agegroups (from 0-5 up to 65+)* cross *Gender*

<u>Swiss Household Budget Survey</u>

* Types of estimators: Horvitz-Thompson estimator, general regression estimator
* Auxiliary information: *Region* cross *Socio-economic status* (for software testing)
* Non-response mechanism: non-response at the original level, non-response of 5%
* Levels of study: *Socio-economic status, Type of household*

There are data sets for each survey including necessary population-level marginal information for stratification and estimation with auxiliary information.

5.3.3 FLFS Results with Speed and Memory Comparisons

Finnish Labour Force Survey. Originally the sample size in the Finnish Labour Force Survey (FLFS) is about twice larger than in this situation. In general the required classification should not cause any substantial difficulties, except the Åland islands, which are somewhat overrepresented in the original survey (not here). A natural phenomenon is the number of unemployed in the age group 65 - 74, due to the low number of persons in labour force in that group. *Note that this section containing FLFS processing includes the main practical experiences of the software; the parts of processing with other two data (SHBS and GMC) include especially data specific experiences.*

Table 5.17: Results of Finnish Labour Force Survey (no non-response)

		Horvitz-Thompson estimator			**Calibration** (Clan, Poulpe and Bascula)		
	Size	**Total** *Rate %*	**Total, standard error**	*Rate, standard error*	**Total** *Rate %*	**Total, standard error**	*Rate, standard error*
All	5000	**285102** *11.17*	**14274** (tb,we), **13977** (bb), **14283** (sj)	*0.55 (tb,sj,we), 0.53 (bb)*	**283759** *11.12*	**14171** (cl, bt, po), **13524** (bb)	*0.54 (cl, bt, po), 0.51 (bb)*
Uusimaa*	1343	**48974** *6.39*	**6170** (tb,we), **6635** (bb)	*0.78 (tb,we), 0.79 (sj), 0.81 (bb)*	**49301** *6.39*	**6105** (cl, bt, po) , **6233** (bb)	*0.78 (cl, bt, po), 0.81 (bb)*
Southern Finland	1749	**115648** *13.05*	**9331** (tb,we), **10052** (bb)	*1.00 (tb,sj,we), 0.98 (bb)*	**116380** *13.04*	**9109** (cl, bt, po) , **9088** (bb)	*1.00 (cl, bt, po), 0.98 (bb)*
Eastern Finland	691	**44053** *14.27*	**5836** (tb,we), **5378** (bb)	*1.76 (tb), 1.77 (sj,we), 1.5 (bb)*	**42435** *14.26*	**5414** (cl, bt, po) , **4526** (bb)	*1.76 (cl, bt, po), 1.5 (bb)*
Mid-Finland	644	**32570** *10.61*	**5000** (tb,we), **4100** (bb)	*1.55 (tb,sj,we), 1.2 (bb)*	**33795** *10.61*	**5037** (cl, bt, po) , **4164** (bb)	*1.55 (cl, bt, po), 1.2 (bb)*
Northern Finland	555	**43857** *15.92*	**5815** (tb,we), **6259** (bb)	*1.94 (tb), 1.95 (sj,we), 2.0 (bb)*	**41848** *15.91*	**5289** (cl, bt, po) , **5568** (bb)	*1.94 (cl, bt, po), 2.0 (bb)*
Åland isl.	18	**0** *0.00*	**0** (all)	*0.00 (all)*	**0** *0.00*	**0** (all)	*0.00 (all)*
M, 15-24	436	**38046** *24.87*	**5123** (tb,we), **3880** (bb)	*3.08 (tb), 3.09 (sj), 1.7 (bb)*	**37822** *24.76*	**5084** (cl, bt, po) , **3826** (bb)	*3.07 (cl, bt, po), 1.7 (bb)*
M, 25-34	451	**32864** *10.43*	**4709** (tb,we), **3736** (bb)	*1.49 (tb,sj), 1.2 (bb)*	**32733** *10.40*	**4687** (cl, bt, po) , **3535** (bb)	*1.48 (cl, bt, po), 1.1(bb)*
M, 35-44	500	**34377** *9.48*	**4951** (tb,we), **5521** (bb)	*1.36 (tb,sj), 1.6 (bb)*	**34234** *9.44*	**4926** (cl, bt, po) , **5398** (bb)	*1.35 (cl, bt, po), 1.5 (bb)*
M, 45-54	529	**21422** *5.88*	**4017** (tb,we), **3272** (bb)	*1.10 (tb,sj), 0.91 (bb)*	**21384** *5.87*	**4012** (cl, bt, po) , **3277** (bb)	*1.10 (cl, bt, po), 0.90 (bb)*
M, 55-64	365	**14585** *11.91*	**3173** (tb,we), **3649** (bb)	*2.50 (tb), 2.51(sj), 2.6 (bb)*	**14573** *11.85*	**3169** (cl, bt, po) , **3638** (bb)	*2.49 (cl, bt, po), 2.6 (bb)*
M, 65-74	229	**0** *0.00*	**0** (all)	*0.00 (all)*	**0** *0.00*	**0** (all)	*0.00 (all)*
W, 15-24	401	**35328** *23.53*	**5028** (tb,we), **1607** (bb)	*3.10 (tb), 3.11 (sj), 1.7 (bb)*	**34836** *23.28*	**4959** (cl, bt, po) , **1397** (bb)	*3.08 (cl, bt, po), 1.5 (bb)*
W, 25-34	409	**26916** *10.37*	**4423** (tb,we), **3168** (bb)	*1.68 (tb), 1.69 (sj), 1.3 (bb)*	**26879** *10.35*	**4415** (cl, bt, po) , **3100** (bb)	*1.68 (cl, bt, po), 1.3 (bb)*
W, 35-44	461	**36715** *11.17*	**5202** (tb,we), **8039** (bb)	*1.57 (tb,sj), 2.5 (bb)*	**36457** *11.07*	**5165** (cl, bt, po) , **7848** (bb)	*1.56 (cl, bt, po), 2.4 (bb)*
W, 45-54	519	**34752** *9.76*	**5014** (tb,we), **2235** (bb)	*1.40 (tb,sj), 0.69 (bb)*	**34687** *9.74*	**5005** (cl, bt, po) , **2111** (bb)	*1.40 (cl, bt, po), 0.67 (bb)*
W, 55-64	361	**10098** *8.61*	**2752** (tb,we), **5441** (bb)	*2.28 (tb), 2.29 (sj), 3.5 (bb)*	**10154** *8.61*	**2767** (cl, bt, po) , **5498** (bb)	*2.29 (cl, bt, po), 3.5 (bb)*
W, 65-74	339	**0** *0.00*	**0** (all)	*0.00 (all)*	**0** *0.00*	**0** (all)	*0.00 (all)*

* *Uusimaa region* contains the capital area. Abbreviations: tb = based on Taylor approximation (SAS, SPSS, Stata, Bascula/Taylor, Poulpe, Sudaan/Taylor), bb = Bascula/BRR, bt = Bascula/Taylor, sj = Sudaan/jackknife, we = WesVar (Jkn), cl = Clan.

The results as such are not a surprise. When having the region as one of the calibration criteria, it is obvious that we have better results when studying the regions separately. On the other hand, the model structure includes the stratification variables (gender, agegroup) as well, so the increase of efficiency (i.e. standard error reduction) is not substantial for the calibration estimator. Those 18 observations selected from the FLFS universe of DACSEIS, representing the Åland islands are all employed.

When the variance estimation methodologies are concerned, all the variance estimates based on *the Taylor linearisation (**SAS, SPSS, Stata, Bascula***/Taylor, ***Poulpe, Sudaan***/Taylor) are the same. The results of the *jackknife* in ***Sudaan*** are equal to the situation where simple random sampling with replacement is conducted in every stratum. When the finite population correction is provided together with proper replicate weights (either created by the software or imported) in ***WesVar***, we should get the analytical variance estimate in the linear case. In general, here the *jackknife with stratification* of ***WesVar*** (JKn option) provides exactly the same standard errors as the Taylor linearisation for the total of unemployed. For the unemployment rate the difference (when existing) is very marginal. The principle of artificially adjusting the sampling design to fit the *Balanced Repeated Replication* method in ***Bascula*** produces rather unbalanced standard error results when compared with other methods.

The data including non-response patterns following the original level (11.6 %) and the 5 % level were studied as well. In practice the FLFS data set was reduced by using indicator variables NR0 (original level) and NR5. Correspondingly, new weights had to be created in order to reflect to the non-response. Here the response probability estimator was $\widehat{\theta}_k = m_h/n_h$.

Table 5.18: Overall results of Finnish Labour Force Survey for three non-response patterns

	Horvitz-Thompson estimator			**Calibration** (Clan, Poulpe and Bascula)		
	Total *Rate*	**Total, standard error**	*Rate, standard error*	**Total** *Rate*	**Total, standard error**	*Rate, standard error*
No non-response	**285102** *11.17*	**14274** (tb,we), **13977** (bb), **14283** (sj)	*0.55 (tb ,sj ,we), 0.53 (bb)*	**283759** *11.12*	**14171** (cl, bt, po) , **13524** (bb)	*0.54 (cl, bt, po), 0.51 (bb)*
5 % non-response level	**276282** *10.80*	**14402** (tb, we), **14411** (sj)	*0.55 (tb ,sj ,we),*	**274174** *10.71*	**14261** (cl, bt, po) ,	*0.55 (cl, bt, po),*
Original non-response pattern (11.6 %)	**263130** *10.25*	**14645** (tb, we), **14653** (sj)	*0.56 (tb, sj, we),*	**259921** *10.10*	**14449** (cl, bt, po) ,	*0.55 (cl, bt, po),*

Considering all results, the estimates tended to get lower when the non-response increased. It reveals the phenomenon that can be noticed in the real labour force survey as well: the non-response group has more unemployed than the response group. Both the strata as response homogeneity groups as such and the calibration surely have bias reducing effect, but they do not cover some categories reflecting to the unemployment (e.g. the socio-economic status).

Preparations. The text files containing the survey data and the population totals are not sufficient as such for estimation in the FLFS case. In order to carry out the survey calculations in a proper manner, the software need additional variables for the FLFS data: the number of respondents by stratum, the basic weight and/or the identification variable. A new age group must be created. ***SAS***, ***SPSS*** and ***Stata*** can deal with data processing; ***Sudaan***, ***WesVar*** and ***Bascula*** have a recoding facility for categorical variables. ***Clan***, ***Poulpe*** and ***Sudaan*** function in the ***SAS*** environment. Thus the preliminary calculations must be conducted outside ***WesVar*** and ***Bascula***, and then another system for data processing is required. Here these calculations are carried out in ***SAS***.

Only ***Bascula*** and ***Clan*** have the ability for creating survey weights. For other software the calculation must be done by the user. In ***SAS, SPSS*** and ***Stata*** the number of respondents must be calculated, however in ***Stata*** the result can be directed strictly into the survey data. In ***SAS*** and ***SPSS*** a merging process of a separate result file must be conducted. Furthermore, the data set of population totals must be merged as well.

SAS, ***SPSS***, ***Stata*** and ***WesVar*** read files. The process is not very difficult for the text form used in the FLFS files. However, ***WesVar*** needs the prior calculations and in this case the data set was of the SAS V6 form. ***Bascula*** needs the Blaise system for reading files. The preparation for this process (a metadata) is rather laborious. It requires exact definitions of the file to be read and what kind of variables are going to be created (including also the range of values for every variable). The *Import* procedure was used in ***SAS*** for importing the text data. It is likely that a direct reading in a data phase in ***SAS*** could read the data quicker, although programming would take some time then.

Estimation. There are two main approaches when preparations and calculations in the software are concerned: *programming* and *providing information by using menus and screens*. In survey situations ***SAS***, ***Clan***, and ***Sudaan*** are clearly program-oriented; ***WesVar*** and ***Bascula*** are menu/screen based; ***Stata*** follows either menus or commands given on a screen; ***Poulpe*** and ***SPSS*** belong to both classes. In the FLFS case ***Poulpe*** macros were utilised instead of screens and ***SPSS*** was used in both ways.

SAS Surveymeans was easy to be programmed. An oddity appearing in the log was the warning of some one-unit strata, though there were no such strata with the chosen levels of study. For the domains for the unemployment rate the SRATSUB macro had to be used: the survey procedures could not deal with it properly. The macro required a little more time to study.

Sudaan has a slightly more rigid syntax than ***SAS***. The *'Levels'* statement for domains requires the number of classes in each domain variable. This should be dealt with in the software, not provided by the user. Furthermore, in FLFS the categorical variables had a value 0, which cannot be used in domain studies in ***Sudaan***. As a result all these variables had to be recoded. The output of ***Sudaan*** includes a lot of unnecessary information and the adjustments of the output form require some cumbersome statements.

In FLFS we used two macros of ***Clan***: %AUXVAR for auxiliary information, estimator structure (including GREG) and preparations for tabulations, and %CLAN for data and design information with table dimensions. It is clear that the macro programming of

Clan is clearly for the advanced users; although logical and well explained in the manual, it is much too complex for simple tasks as a basic Horvitz-Thompson estimator and its standard error. For calibration the marginal distributions had to be in a predefined form in a ***SAS*** data, and this preparation required some extra data steps in **SAS**. The macros produced several temporary data sets during the estimation process. The outcome was a ***SAS*** data 'Dut' where the estimates and the standard errors could be found.

It was decided that for the DACSEIS sample data sets the macros of ***Poulpe*** were used instead of screens (which are the main tools of providing information in ***Poulpe***), mainly because of simulation reasons (easy to be put into a program). It is possible that the simulations do too much in this context: the programmer of this simulation had severe difficulties with the manuals in French and the material in English didn't provide enough information for a sufficient study of the macros. In that sense the evaluation is unfair. Nevertheless, the results are the same as for other Taylor-based software. ***Poulpe*** carries out a large number of data phases and procedures during execution, perhaps as a preparation for many different theoretical properties available in it. As a versatile software, the macros of ***Poulpe*** have a long string of parameters (originally for screens), and only few of them need to be filled in the FLFS case. The output has a lot of information in it, including checks and general messages. The style of the output is not very finalised: one has to search the right place for the variance estimates (*not* providing standard errors). The domain estimation is very laborious: one has to do a separate indicator variable for every domain beforehand.

SPSS has a new feature '*Complex Samples*' available both in a menu and in programming. It is rather easy to describe the analysis plan for FLFS using menus, giving the stratum identification, the weight variable, the size variable for strata and choosing the probability structure (here '*equal WOR*'). The analysis plan is saved as an ***SPSS*** '*csaplan*' file, which can be used later. The plan must exist or be created before any calculations can be done. The estimator and the levels of study with some options can be chosen in separate menus for descriptives (total) and ratios. The total and the desired accuracy measures must be chosen from a separate screen (mean and standard error are defaults). Note that a better way for more laborious tasks is to make a program having these requests (as in the GMC case). The output is compact and it can be edited, copied and saved easily.

Also ***Stata*** has the menu-based operation system together with the command facility. The survey settings can be easily done with a *'svyset'* command or by using a menu '*set variables for a survey data*'. The system recalls the settings of the end of the previous session. The calculations and the level of study definitions are simple by using corresponding menus for different parameters. The output has only the relevant (user defined) information in a compact form. The output includes an explanation of the principle of the design effect, if that option is chosen.

Bascula can be activated from one menu of ***Blaise***. It produces a separate screen for operations, and in the beginning the user can choose whether to use an existing setup (.wif data set) or create a new setup. Because ***Bascula*** is a software especially concentrating on weighting, that part is emphasised in the menus of ***Bascula***. The population totals for stratification and calibration must be imported onto one screen. The user can also copy the distribution from another application, e.g. ***Excel***. The weighting model is requested before further calculations can be carried out. ***Bascula*** checks the model and existing auxiliary information. Then the weighting method (post-stratification, ratio estimator, linear weighting, multiplicative weighting) and the variance estimation method (Taylor, BRR) can be chosen. The weights and replicate weights are calculated after that. The request of the estimates together with the levels of study must be provided on a separate screen, and finally the results can be computed. The results have a clear output and they can be exported into a separate file (.btf data set). In all the process of finally getting estimates took some time for the first-time user, but when learning the logic of the program, the task became easier.

WesVar has a two-part structure in its calculations: 1) operations for a ***WesVar*** data file, 2) creating or carrying out ***WesVar*** workbook requests. When creating the ***WesVar*** data (.var data set) it is important that all the variables that might be used are included. There must be an identification variable for the primary sampling units, i.e. in the FLFS case the persons. A nuisance concerning the PSU identification variable is that it must begin from the value 1 in every stratum, otherwise ***WesVar*** does not proceed. The creation of the replication weights takes time, which is surprising when compared with the fast speed of the jackknife with stratification in ***Sudaan***. Attaching the finite population correction vector caused some problems. The import of the predefined fpc correction terms produced commas as decimal separators (dots in the text file), which were in contradiction with the jackknife weights with dots. After finalising the data file definitions the user must get back to the general screen and create or choose a workbook, which then opens another screen. This is not self-evident for the first-timer. One or more table requests can be defined in the workbook. The workbook has many facilities and properties. One can define the functions to be calculated (a variable name must be provided). However, the total is automatically calculated for the analysis variables chosen in the workbook. When the user submits the table request, there is no visible indication whether the program is running or not. One must choose the *'job queue'* window for knowing the situation of the task. For viewing the output must be opened from a 'button' on the screen. The output includes some information not necessary in practice (e.g. all the replicate weights) and the user must scroll to the end of the output before the results which are requested can be found. ***WesVar*** is considerably slow with FLFS when compared with the other software.

Speed. The software emphasise different aspects in their calculations: ***SAS***, ***SPSS*** and ***Stata*** are developed for simple common survey situations, including only the basic calculations; ***Sudaan*** has a structure prepared for more advanced calculations; ***Bascula*** concentrates especially on proper weighting; ***Clan*** and ***Poulpe*** are very advanced in both dealing with the sampling designs and the estimation methods; ***WesVar*** emphasises easiness and general applicability of resampling methods at the expense of speed. The running times of the FLFS tasks follow this categorisation. SAS, ***SPSS*** and ***Stata*** produce the results fast with the original survey tools of the software. However, the *SRATSUB* macro in ***SAS*** increases the total time. ***Clan*** carries out some operations inside ***SAS*** automatically, perhaps including processes not needed for simple tasks. Nevertheless, the tasks including calibration were performed almost as fast as ***SAS, SPSS*** and ***Stata***. It is likely that ***Poulpe*** prepares the system for complex calculations as well, at least the high number of data phases and procedures during execution might indicate that. Some checks and processing of the weighting system of ***Bascula*** increase the run time, but not significantly in FLFS. ***WesVar*** is not at its best when large non-clustered data are concerned. The creation and use of numerous replicate weights slows down the process significantly. The test computer included following properties: *Intel Pentium 3, 1 GHz, 0.26 Gigabytes (GB), MS Windows NT.*

Table 5.19: Comparing Speed of Different Software with FLFS

Software	Preparations	Estimation	Total time
SAS	Preparations in SAS: importing survey data and population totals (proc import), new classification and recoding, new marginals (summary & data phase), sample size information, sorting and merging the data sets, creation of weights **3.97 sec** .	- Procedure Surveymeans **0.20 sec**. - Macro SRATSUB **11 sec.**	**15.17 sec**
Clan		- marginals for calibration (creation and proper form) **0.15 sec.** - some sorts and data phases within macro, **1.59 sec.**	**5.31 sec.**
Poulpe		A lot of data phases and procedures within macros, **appr. 26 sec.**	**appr. 30 sec**
Sudaan		Procedure Descript **0.26 sec**. (**0.26 sec** for jackknife)	**4.23 sec.**
SPSS	importing data and population totals, age classification, sample frequencies for weights, merging totals and sample frequencies, calculating weights **appr 2 sec.**	- *defining the analysis plan* - *choosing the estimates and levels of study* - executing the requests for the total < **1 sec** - for the ratio < **1 sec**	**appr. 3 - 5 sec.**
Stata	importing data and pop. totals, age class, merging totals to the sample data,calculating sample frequency into the sample data **appr 2 sec**	- *defining the design (svyset)* - *choosing the estimates and levels of study* - executing for total < **1 sec** - executing for ratio < **1 sec**	**appr. 2 - 4 sec.**
Bascula	in SAS: importing survey data and population totals, in SAS: new variables, basic inclusion weights, exporting the data to ASCII, *metadata preparation in Blaise for reading,* reading ASCII file into Blaise using metadata **7 sec**	- *s electing and defining variables and sample design in Bascula:* - *copying population totals to Bascula* - specifying weighting model, checks **1 sec** , computing weights - computing estimates **2 sec**	**appr 10 sec,** Bascula **5 sec**
WesVar	in SAS: importing survey data and population totals, in SAS: new variables, weights, including sd2-type SAS data into WesVar, selecting variables and creating a WesVar data **6 sec.**	- creating 5000 replicate weights **7 min 40 sec.** - attaching finite population correction information < **1 sec** - *creating a WesVar workbook for tabulations* - submitting the workbook requests **2 min 43 sec.**	**appr 10 min 30 sec,** WesVar **10 min 25 sec**

The test computer: *Intel Pentium 3, 1 GHz, 0.26 GB, MS Windows NT.*

Memory. In most software it is difficult to follow what data processes are going on during execution and how the memory capacity is utilised. The final outcome of the preparation and estimation process of the software are the data sets remaining on the hard disk. This study describes the sizes of these data sets after conducting the FLFS calculations. It reveals that ***SPSS*** and ***Stata*** have smallest data sets (0.2 Mb). ***SAS Surveymeans*** and **Sudaan** do only the calculation and the remaining data sets are from the joint processing in ***SAS***, requiring about five times more space than in ***SPSS*** and ***Stata***. The temporary files of ***Clan*** increase the amount of memory to nearly 2 Mbs. The Blaise data file as such in ***Bascula*** is less than the ***SAS*** file, but the setup files add the total memory allocation to the level of 2.2 Mbs. The large amount of data sets in ***Poulpe*** reserve a lot of space (10 Mb). The need of a replicate weight matrix to be included in the ***WesVar*** survey data inflates the size to 441 Mb.

Table 5.20: Comparing Memory Allocation of Different Software with FLFS

<table>
<tr><th>Software</th><th>Joint SAS data sets</th><th>Software specific data sets</th></tr>
<tr><td>SAS</td><td rowspan="4">- survey data (sas7bdat-type) 961 kb
- data of new totals (sas7bdat) 5 kb
- sample size data (sas7bdat) 5 kb</td><td>- none</td></tr>
<tr><td>Clan</td><td>- marginal data sets 34 kb
- temporary SAS data sets: err2, _cl, _db0, _ds0, _xv1 1921 kb
- data containing results: dut 9 kb</td></tr>
<tr><td>Poulpe</td><td>- marginal data sets 34 kb
- a lot of temporary data sets created and deleted during execution
- p1_aa, p1_ab, p1_ba, p1jou, r1res, r1clp, r1lis, p1pii, treez, fsamp, geo, tree 9985 kb</td></tr>
<tr><td>Sudaan</td><td>- none</td></tr>
</table>

	Data sets
SPSS	SPSS data set **202 kb,** auxiliary information **1 kb** , analysis plan **1 kb**
Stata	Stata data set **204 kb** , auxiliary information **1 kb**
Bascula	Blaise prepared datamodel (.bmi) **4 kb,** Blaise data base (.bdb) **547 kb**, setups (.blg, .rfw, .wga) **1707 kb**
WesVar	WesVar data file (.var) **441 930 kb,** Workbooks **4 kb**

5.3.4 SHBS Results with Speed and Memory Comparisons

Swiss Household Budget Survey. The continuous variables of the Swiss Household Budget Survey (expenditure and income) were the main reasons for taking this survey in the evaluation. However, the software performed the calculations with no problem, except ***WesVar***, which cannot create replicate weights for data sets containing more than 10000 primary sampling units.

Table 5.21: Results of Swiss Household Budget Survey (no non-response)

		Horvitz-Thompson estimator			**Model assisted estimation** (Clan, Poulpe, Bascula)		
	Size	**Mean I** *Mean E*	**Mean Income, standard error**	*Mean Exp., standard error*	**Mean I** *Mean E*	**Mean Income, standard error**	*Mean Exp., standard error*
All	15434	**8353** *7442*	**44** (tb), **45** (bb)	*45 (tb), 50 (bb)*	**8327** *7420*	**40** (cl, bt, po), **49** (bb)	*42 (cl, bt, po), 48 (bb)*
Other	385	**4953** *5538*	**186** (tb, sj), **197** (bb)	*219 (tb), 220 (sj), 214 (bb)*	**4918** *5509*	**181** (cl, bt, po), **197** (bb)	*214 (cl, bt, po), 214 (bb)*
Salaried	10301	**9273** *7957*	**51** (tb, sj), **66** (bb)	*54 (tb), 56 (bb)*	**9243** *7936*	**48** (cl, bt, po), **66** (bb)	*52 (cl, bt, po), 56 (bb)*
Independent (without farmer)	1337	**8791** *9142*	**192** (tb, sj), **210** (bb)	*216 (tb), 217 (sj), 230 (bb)*	**8776** *9116*	**192** (cl, bt, po), **210** (bb)	*213 (cl, bt, po), 230 (bb)*
farmer	296	**5888** *5718*	**236** (tb), **237** (sj), **212** (bb)	*146 (tb), 141 (bb)*	**5890** *5740*	**234** (cl, bt, po), **212** (bb)	*145 (cl, bt, po), 141 (bb)*
unemployed	240	**4104** *4837*	**139** (tb, sj), **146** (bb)	*231 (tb), 232 (sj), 259 (bb)*	**4100** *4838*	**137** (cl, bt, po), **146** (bb)	*230 (cl, bt, po), 259 (bb)*
pensioner	2875	**5907** *5456*	**90** (tb, sj, bb)	*88 (tb), 70 (bb)*	**5898** *5443*	**88** (cl, bt, po), **90** (bb)	*86 (cl, bt, po), 70 (bb)*
person alone	4197	**5336** *4843*	**57** (tb, sj), **75** (bb)	*48 (tb), 50 (bb)*	**5339** *4843*	**57** (cl, bt, po), **71** (bb)	*48 (cl, bt, po, bb)*
mother/father alone + 1 child	365	**5991** *5243*	**151** (tb), **152** (sj), **141** (bb)	*140 (tb), 141 (sj), 110 (bb)*	**6003** *5289*	**150** (cl, bt, po), **141** (bb)	*140 (cl, bt, po), 108 (bb)*
mother/father alone + 2 or more children	277	**6946** *6128*	**227** (tb), **228** (sj), **242** (bb)	*218 (tb), 219 (sj), 220 (bb)*	**6767** *6060*	**194** (cl, bt, po), **239** (bb)	*189 (cl, bt, po), 221*
couple	4809	**9369** *8209*	**85** (tb, sj), **82** (bb)	*95 (tb), 96 (sj), 93 (bb)*	**9362** *8203*	**84** (cl, bt, po), **78** (bb)	*94 (cl, bt, po), 90 (bb)*
couple with 1 child	1767	**10040** *9102*	**144** (tb), **145** (sj), **158** (bb)	*174 (tb), 220 (bb)*	**10056** *9099*	**147** (cl, bt, po), **157** (bb)	*170 (cl, bt, po), 219 (bb)*
couple with 2 children	2386	**9935** *8885*	**95** (tb, sj), **107** (bb)	*89 (tb), 101 (bb)*	**9922** *8881*	**94** (cl, bt, po), **108** (bb)	*89 (cl, bt, po), 103 (bb)*
couple with 3 or more children	1015	**10015** *9245*	**153** (tb), **154** (sj), **209** (bb)	*144 (tb), 183 (bb)*	**10003** *9229*	**151** (cl, bt, po), **215** (bb)	*140 (cl, bt, po), 184 (bb)*
other	618	**9302** *7719*	**242** (tb), **243** (sj), **299** (bb)	*263 (tb), 264 (sj), 261*	**9389** *7802*	**239** (cl, bt, po), **290** (bb)	*274 (cl, bt, po), 259 (bb)*

Abbreviations: tb = based on Taylor approximation (SAS, SPSS, Stata, Bascula/Taylor, Poulpe, Sudaan/Taylor), bb = Bascula/BRR, bt = Bascula/Taylor, sj = Sudaan/jackknife, we = WesVar (Jkn), cl = Clan.

Although we have two continuous variables here in order to estimate the mean for both, the results based on modelling were quite close to the Horvitz-Thompson estimates. Again the Balanced Repeated Replications of Bascula provided rather varying standard errors.

Speed. The software behave quite similarly as in the FLFS case, but having slightly longer times. The effect of the SRATSUB macro of ***SAS*** can be seen again. The slow performance of ***Stata*** is a surprise.

Table 5.22: Comparing Speed of Different Software with SHBS

Software	**Preparations**	**Estimation**	**Total time**
SAS	Preparations in SAS:	- Procedure Surveymeans **0.29 sec.** - Macro SRATSUB **16 sec** .	21.40 sec
Clan	importing survey data and population totals (proc import), new classification and recoding, new marginals (summary & data phase), sample size information 0.01 sec sorting and merging the data sets, creation of weights **5.11 sec.**	- marginals for calibration (creation and proper form), some sorts and data phases within macro, **10 sec.**	15.11 sec.
Poulpe		A lot of data phases and procedures within macros, appr. **1 min. 1 sec** .	appr. 1 min 6 sec
Sudaan		Procedure Descript **0.42 sec** . (**0.78 sec** for jackknife)	5.53 sec.
SPSS	importing data and population totals, age classification , sample frequencies for weights, merging totals and sample frequencies, calculating weights **appr. 3 sec.**	- *defining the analysis plan* - *choosing the estimates and levels of study* - executing the requests for two means **appr. 3 sec**	appr. 6 sec.
Stata	importing data and pop. totals age class, merging totals to the sample data, calculating sample frequency into the sample data **appr 2 sec**	- *defining the design (svyset)* - *choosing the estimates and levels of study* - executing the requests for two means **appr. 16 sec.**	appr. 18 sec.
Bascula	in SAS: importing survey data and population totals, in SAS: new variables, basic inclusion weights, exporting the data to ASCII, *metadata preparation in Blaise for reading,* reading ASCII file into Blaise using metadata **appr. 7 sec**	- *selecting and defining variables and sample design in Bascula:* - *copying population totals to Bascula* - specifying weighting model **appr. 1 sec.** - checks, computing weights **appr. 18 sec.** - computing estimates **appr. 7 sec.**	appr 33 sec, Bascula 28 sec

The test computer: *Intel Pentium 3, 1 GHz, 0.26 Gb, MS Windows NT.*

Memory. The software with modelling tasks have most of the memory allocation as before. The setup files of Bascula (especially rfw) are rather large, when compared with the FLFS case.

Table 5.23: Comparing Memory Allocation of Different Software with SHBS

Software	Joint SAS data sets	Software specific data sets
SAS	- survey data (sas7bdat-type) **2065 kb**	- none
Clan	- data of new totals (sas7bdat) **5 kb** - sample size data (sas7bdat) **5 kb**	- marginal data sets **47 kb** - temporary SAS data sets: err2, _cl, _db0, _ds0, _xv1 **5467 kb** - data containing results: dut **9 kb**
Poulpe		- marginal data sets **47 kb** - a lot of temporary data sets created and deleted during execution - p1_aa, p1_ab, p1_ba, p1jou, r1res, r1clp, r1lis, p1pii, treez, fsamp, geo, tree **33 100 kb**
Sudaan		- none

	Data sets
SPSS	SPSS data set **1644 kb,** auxiliary information **1 kb** , analysis plan **1 kb**
Stata	Stata data set **1754 kb,** auxiliary information **1 kb**
Bascula	Blaise prepared datamodel (.bmi) **4 kb,** Blaise data base (.bdb) **1930 kb** , setups (.blg, .rfw, .wga) **14 936 kb**

5.3.5 GMC Results with Speed and Memory Comparisons

German Microcensus. The German Microcensus with its highly stratified cluster sampling and large amounts of observations provides a challenging task for the software. Some of the software refuse the situation appearing to some extent in GMC, i.e. one PSU in a stratum. Thus many strata had to be collapsed to the neighbouring stratum, and this changed the original design structure. ***WesVar*** couldn't deal with the data of this magnitude: the largest region was Schleswig-Holstein with 29140 records, 35 strata and 1680 primary sampling units. Some technical problems appeared constantly while testing ***Poulpe*** with the multilevel GMC, and due to comprehension problems with the manuals and messages in French the calculations of GMC did not succeed. ***Bascula*** did not give an alternative for Balanced Repeated Replication with the whole GMC data. ***Stata*** needed the memory extension ('set memory') up to 200 megabytes.

Table 5.24: Results of German Microcensus (no non-response)

		Horvitz-Thompson estimator			**Model assisted estimation**		
	Size	**Total** *Rate*	**Total, standard error**	*Rate, standard error*	**Total** *Rate*	**Total, standard error**	*Rate, standard error*
		3910441 *9.66*	**20737**	*0.00*	**39063112** 9.65	**19374**	*0.00*
M, 0 - 5	19617	**0** .	**0**	.	**0** .	**0**	.
M, 6 - 10	23748	**0** .	**0**	.	**0** .	**0**	.
M, 11 - 15	23404	**0** .	**0**	.	**0** .	**0**	.
M, 16 - 20	22490	**67973** *8.82*	**2600** (tb)	*0.32* (tb)	**67881** *8.88*	**2567**	*0.32*
M, 21 - 25	23321	**190360** *10.90*	**4365** (tb)	*0.24* (tb)	**189541** *10.86*	**4307**	*0.23*
M, 26 - 30	31881	**245064** *9.06*	**4962** (tb)	*0.18* (tb)	**244187** *9.03*	**4896**	*0.17*
M, 31 - 35	35200	**241902** *7.21*	**4932** (tb)	*0.14* (tb)	**241343** *7.19*	**4852**	*0.14*
M, 36 - 40	32044	**225603** *7.31*	**4713** (tb)	*0.15* (tb)	**225617** *7.30*	**4649**	*0.15*
M, 41 - 45	29242	**203294** *7.21*	**4524** (tb)	*0.15* (tb)	**203560** *7.22*	**4473**	*0.15*
M, 46 - 50	26776	**185091** *7.25*	**4346** (tb)	*0.16* (tb)	**184228** *7.22*	**4269**	*0.16*
M, 51 - 55	25709	**197064** *8.35*	**4438** (tb)	*0.18* (tb)	**196707** *8.35*	**4387**	*0.18*
M, 56 - 60	32216	**393606** *15.99*	**6287** (tb)	*0.24* (tb)	**392887** *15.98*	**6209**	*0.23*
M, 61 - 65	23720	**67983** *9.52*	**2621** (tb)	*0.35* (tb)	**68038** *9.55*	**2611**	*0.35*
M, 66 -	50944	**3051** *1.34*	**544** (tb)	*0.24* (tb)	**3092** *1.36*	**547**	*0.24*
N, 0 - 5	18939	**0** .	**0**	.	**0** .	**0**	.
N, 6 - 10	22648	**0** .	**0**	.	**0** .	**0**	.
N, 11 - 15	22237	**0** .	**0**	.	**0** .	**0**	.
N, 16 - 20	22045	**59520** *9.94*	**2473** (tb)	*0.39* (tb)	**59836** *9.99*	**2474**	*0.39*
N, 21 - 25	22899	**154025** *9.99*	**3900** (tb)	*0.24* (tb)	**153927** *9.99*	**3868**	*0.24*
N, 26 - 30	31008	**220229** *9.39*	**4701** (tb)	*0.19* (tb)	**220073** *9.38*	**4643**	*0.19*
N, 31 - 35	34269	**256837** *9.99*	**5048** (tb)	*0.19* (tb)	**256513** *9.97*	**4972**	*0.18*
N, 36 - 40	31501	**234262** *9.82*	**4820** (tb)	*0.19* (tb)	**234869** *9.84*	**4759**	*0.19*
N, 41 - 45	29140	**211948** *9.21*	**4611** (tb)	*0.19* (tb)	**211846** ***9.21***	**4556**	*0.19*
N, 46 - 50	26970	**199273** *9.72*	**4477** (tb)	*0.21* (tb)	**198937** *9.70*	**4432**	*0.21*
N, 51 - 55	25964	**204170** *11.47*	**4529** (tb)	*0.24* (tb)	**204764** *11.51*	**4503**	*0.24*
N, 56 - 60	32071	**323150** *19.30*	**5693** (tb)	*0.31* (tb)	**322521** *19.29*	**5645**	*0.30*
N, 61 - 65	25047	**21538** *7.50*	**1480** (tb)	*0.50* (tb)	**21440** *7.48*	**1464**	*0.49*
N, 66 -	85954	**4497** *3.09*	**667** (tb)	*0.45* (tb)	**4501** *3.10*	**664**	*0.45*

Abbreviations: tb = based on Taylor approximation (SAS, SPSS, Stata, Bascula/Taylor, Poulpe, Sudaan/Taylor), bb = Bascula/BRR, bt = Bascula/Taylor, sj = Sudaan/jackknife, we = WesVar (Jkn), cl = Clan.

Due to the vast amount of observations both the Horvitz-Thompson estimator and the GREG estimator give rather similar results.

Speed. The preparations last very long with GMC, including various data and merging phases together with collapsing strata. Note that both ***Bascula*** and ***Clan*** had specific operations for the GREG estimator which took much more time than normal Horvitz-Thompson estimation would have taken. In general ***SAS***, ***Sudaan***, ***SPSS*** and ***Stata*** performed the estimation task fairly quickly.

Table 5.25: Comparing Speed of Different Software with GMC

Software	**Preparations**	**Estimation**	**Total time**
SAS	DUE TO STRATUM COLLAPSING ALL THE PREPARATIONS WERE MADE IN SAS: - importing survey data and population totals ec (proc import), collapsing strata, new classification and recoding, new marginals (summary & data phase), sample size information, sorting and merging the data sets, creation of weights **16 min 38 sec.**	Procedure Surveymeans 2 min 37 sec.	19 min 15 sec
Clan		- marginals for calibration (creation and proper form), some sorts and data phases within macro, 29 min 59 sec.	46 min 37 sec
Sudaan		Procedure Descript 1 min 21 sec. (2 min 1 sec. for jackknife).	17 min 59 sec
SPSS		- *defining the analysis plan* - *choosing the estimates and levels of study* - executing the requests for the total **appr. 52 sec** - for the ratio **appr. 24 sec**	17 min 54 sec
Stata		- *defining the design (svyset)* - *choosing the estimates and levels of study* - executing for total **appr. 44 sec** - executing for ratio **appr. 1 min 31 sec**	18 min 43 sec
Bascula		- *selecting and defining variables and sample design in Bascula:* - *copying population totals to Bascula* - specifying weighting model **appr. 1 min 33 sec.** - checks, computing weights **appr. 3 min 16 sec.** - computing estimates **appr. 19 min 58 sec.**	41 min 25 sec

The test computer: *Intel Pentium 3, 1 GHz, 0.26 GB, MS Windows NT.*

Memory. The size of the data set in ***SAS***, ***SPSS*** and ***Stata*** is quite the same, and less than in ***Bascula***. Furthermore, the setups reserve a lot of space in ***Bascula***. The large temporary data sets of ***Clan*** are probably outcomes of modelling operations.

Table 5.26: Comparing Memory Allocation of Different Software with GMC

<table>
<tr><th>Software</th><th>Joint SAS data sets</th><th>Software specific data sets</th></tr>
<tr><td>SAS</td><td>- survey data (sas7bdat-type) 107 127 kb</td><td>- none</td></tr>
<tr><td>Clan</td><td rowspan="2">- data of new totals (sas7bdat) 5 kb
- sample size data (sas7bdat) 5 kb</td><td>- marginal data sets 37 kb
- temporary SAS data sets: err2, _cl, _db0, _ds0, _xv1 338 537 kb
- data containing results: dut 9 kb</td></tr>
<tr><td>Sudaan</td><td>- none</td></tr>
<tr><td></td><td colspan="2">Data sets</td></tr>
<tr><td>SPSS</td><td colspan="2">SPSS data set 105 768 kb, auxiliary information 1 kb , analysis plan 1 kb</td></tr>
<tr><td>Stata</td><td colspan="2">Stata data set 107 127 kb , auxiliary information 1 kb</td></tr>
<tr><td>Bascula</td><td colspan="2">Blaise prepared datamodel (.bmi) 4 kb, Blaise data base (.bdb) 129 845 kb , setups (.blg, .rfw, .wga) 30 058 kb</td></tr>
</table>

5.3.6 Comparing Speed between R and SAS-based Software

R provides a promising platform for creating various applications also regarding survey methods and variance estimation. Here we compare the speed of two systems with the German Microcensus data sets. The SAS-based software in question are SAS (Surveymeans), Sudaan (Descript) and Clan (macro). The complexity of the original design and estimation is somewhat simplified here: we conduct stratified element-level Horvitz-Thompson estimation for the total number of unemployed for the full sample without non-response. In order to provide a situation as fair as possible for both systems, all sessions were executed separately for every data (if conducted at the same session after another large task, the preparation takes more time). All other applications were closed down while executing. The computer included following properties: *Intel Pentium 3, 1 GHz, 0.26 GB, MS Windows NT.*

Sorting and merging seem to be a time-consuming part in SAS and correspondingly in R reading and appending data sets take time. In this test the preparations are faster with R, but the estimation part is quicker with SAS. SAS Surveymeans is quite efficient, but the threshold seen between data 14 and data 15 is remarkable. It seems that the SAS data 15 is not fully included in RAM and thus reading-writing between RAM and hard disk takes much more time than the real SAS operations. Therefore results with 15 and 16 do not show the real speed of SAS but mostly the speed of the hard disk drive. The problems of R indicate reaching the memory limits of the computer.

It is a little bit unfair to compare the advanced software (Sudaan, Clan) in this simple situation. As a default Sudaan calculates more than SAS and especially this R code.

Table 5.27: Comparing Speed between R and SAS- based Software

BL's merged together (order from the smallest to the largest)	**Size of data set (MB)**	**Units**	**Preparation time (seconds)**		**Estimation time (seconds)**			
			R	**SAS**	**R**	**SAS**	**Sudaan**	**Clan**
1: bre	SAS: 1.2 R: 0.6	6791	1.24	2.12	0.23	0.15	0.25	1.66
5: bre, sal, mvp, ham, thn	SAS: 14.5 R: 7.0	79525	15.20	22.28	0.78	1.06	1.35	11.93
10: bre, sal, mvp, ham, thn, bra, saa, swh, ber, rlp	SAS: 43.7 R: 21.2	240378	47.86	75.07	1.81	1.59	2.26	52.34
14: bre, sal, mvp, ham, thn, bra, saa, swh, ber, rlp, sac, hes, nie, baw	SAS: 96.3 R: 46.7	529611	139.68	173.67	20.75	1.78	4.45	200.76
15: bre, sal, mvp, ham, thn, bra, saa, swh, ber, rlp, sac, hes, nie, baw, bay	SAS: 119.5	657350	*	280.26	*	14.25	35.20	331.09
16: bre, sal, mvp, ham, thn, bra, saa, swh, ber, rlp, sac, hes, nie, baw, bay, nrw	SAS: 151.1	831004	**	349.18	**	18.21	101.78	426.50

* R data 15: *"Error: cannot allocate vector of size 2068 Kb, In addition: Warning message: Reached total allocation of 254 Mb: see help(memory.size) "*
** R data 16: *Application failure (a message by "Dr. Watson" of the computer)*

Furthermore, Clan is clearly created for model-assisted estimation with auxiliary data and there is also the ability for calibration. Its syntax provides some additional data sets and conducts sorting, which are the main reason for the slower performance. If the test design and levels of study were more complex here, the results might be weaker for R.

5.3.7 Procedure MI for Imputation in SAS: An Overview and Testing with Swiss Household Budget Survey

The experimental MI procedure performs multiple imputation of missing data. Multiple imputation inference involves three phases:

1. The missing data are filled in m times to generate m complete data sets.

2. The m complete data sets are analyzed using standard statistical analyses.

3. The results from the m complete data sets are combined to produce inferential results.

The new MI procedure creates multiply imputed data sets for incomplete multivariate data. It uses methods that incorporate appropriate variability across the m imputations. The method of choice depends on the patterns of missingness. For data sets with monotone missing patterns, either a parametric regression method that assumes multivariate normality or a nonparametric method that uses propensity scores is appropriate.

For data sets with arbitrary missing patterns, a Markov Chain Monte Carlo (MCMC) method (SCHAFER, 1997) that assumes multivariate normality is used to impute all missing values or just enough missing values to make the imputed data sets have monotone missing patterns.

The new MIANALYZE procedure can be used to generate valid statistical inferences about these parameters by combining results from the m analyses. These two procedures are available in experimental form in Release 8.2 of the SAS System.

Thorough descriptions of procedures MI and MIANALYZE can be found in documents:

http://support.sas.com/rnd/app/papers/miv802.pdf

http://support.sas.com/rnd/app/papers/mianalyzev802.pdf

Swiss HBS data was used for testing MI procedure. Despite the partial missing assumption complete missingness for INC (income) variable was used according to original item missingness variable (INR0). Five (optional) datasets with imputed values were generated during the run of proc MI:

```
proc mi data=SHBS_INR0 seed=37851 out=outmi;
var exp inc_nr0;
run;
```

```
                              The MI Procedure

                             Model Information

          Data Set                                  WORK.SHBS_INR0
          Method                                    MCMC
          Multiple Imputation Chain                 Single Chain
          Initial Estimates for MCMC                EM Posterior Mode
          Start                                     Starting Value
          Prior                                     Jeffreys
          Number of Imputations                     5
          Number of Burn-in Iterations              200
          Number of Iterations                      100
          Seed for random number generator          37851

                            Missing Data Patterns

                                                     ---------Group Means--------
   Group    EXP    INC_nr0        Freq     Percent           EXP          INC_nr0

       1     X      X            14575       94.43    7409.961523     8351.339088
       2     X      .              859        5.57    7702.948856               .

                         EM (Posterior Mode) Estimates

                _TYPE_     _NAME_              EXP         INC_nr0

                MEAN                   7426.268127     8361.847180
                COV        EXP            28562260        18405743
                COV        INC_nr0        18405743        30337243

                  Multiple Imputation Variance Information

                                                                Relative      Fraction
               -----------------Variance-----------------       Increase       Missing
Variable         Between        Within         Total      DF   in Variance   Information

INC_nr0        79.700774   1969.130143   2064.771072  1654.7      0.048570      0.047342

                  Multiple Imputation Parameter Estimates

       Variable          Mean     Std Error    95% Confidence Limits         DF

       INC_nr0    8367.832831     45.439752     8278.707     8456.958    1654.7
```

Figure 5.14: Output of Proc MI

The "Model Information" table describes the method used in the multiple imputation process. By default, the procedure uses the Markov Chain Monte Carlo (MCMC) method with a single chain to create five imputations. The MI procedure takes 200 burn-in iterations before the first imputation and 100 iterations between imputations.

The "Missing Data Patterns" table lists distinct missing data patterns with corresponding frequencies and percents. Here, an "X" means that the variable is observed in the corresponding group and a "." means that the variable is missing. The table also displays group-specific variable means. The table shows that item non-response is 5.6% for variable income. Mean expenditure is slightly higher for the group where income variable is missing.

The "Multiple Imputation Variance Information" table displays the between-imputation variance, within-imputation variance, and total variance for combining complete-data inferences. It also displays the degrees of freedom for the total variance. The relative increase in variance due to missing values was 4.9%.

The "Multiple Imputation Parameter Estimates" table displays the estimated mean and standard error of the mean for each variable. The inferences are based on the t distribution. The table also displays a 95% confidence interval for the mean.

Point estimates based on m imputations are calculated as follows:

$$\overline{\mu} = \frac{1}{m} \sum_{i=1}^{m} \mu_i \text{ , where}$$

μ_i is the point estimate (mean) of ith imputed dataset.

Within-imputation variance is estimated as the average of variances of the m imputed datasets:

$$W = \frac{1}{m} \sum_{i=1}^{m} v_i \text{ , where}$$

v_i is the estimate of variance in the ith dataset.

Between-imputation variance B is calculated as follows:

$$B = \frac{1}{m-1} \sum_{i=1}^{m} (\mu_i - \overline{\mu})^2$$

Then the variance estimate associated with $\overline{\mu}$ is the total variance

$$\mathrm{V} = W + \left(1 + \frac{1}{m}\right) B$$

Results of proc MI were computed by region, household type, and socio-economic status. Results were compared with results obtained from dataset without item non-response. Overall mean income was slightly overestimated using multiple imputation procedure (by 0.18%). Procedure MI overestimated also standard error of mean by 4.0%: Figure below shows that mean income of 3^{rd} socio-economic status (farmers) is overestimated by 8%. Also mean income of unemployed (4) and others (0) are overestimated by 3–4%. We can see significant overestimates of the standard error for above mentioned socioeconomic groups too. These three socio-economic groups are relatively small compared to salaried, pensioners and entrepreneurs. Significant overestimates of standard error we can see for single parent families with one child (household type 1). For all other observed groups the relative bias of standard error is below 10%.

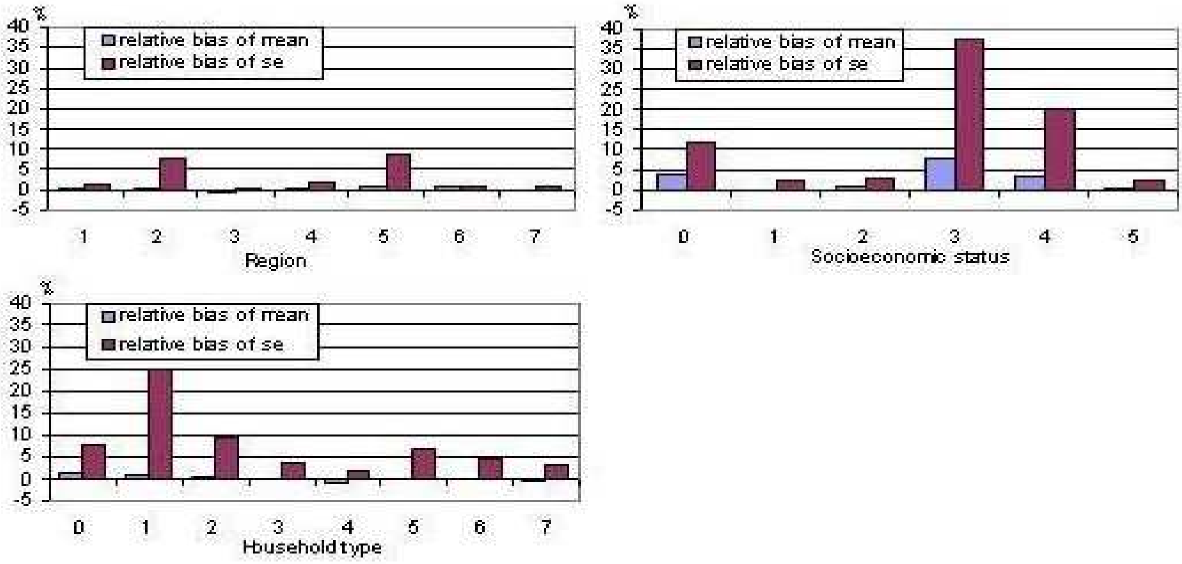


Figure 5.15: Relative bias of mean income and its standard error obtained using MI procedure compared to the mean income and standard error computed from complete dataset by region, household type and socioeconomic status

Table 5.28: Point estimates and variance estimates for mean income based on 5 imputed datasets by region, household type and socio-economic status (item non-response mechanism INR0).

		Point and variance estimates based on 5 imputed datasets					Results without item non-response	
		Mean income	W	B	V	se	Mean income	se
	Total	8367.8	1969.1	79.7	2064.8	45.4	8352.8	43.7
Region								
1	Plateau central	7912.2	5995.1	78.0	6088.8	78.0	7903.6	76.9
2	Region lemanique	8382.2	13337.8	619.4	14081.0	118.7	8354.4	110.4
3	Zurich	9052.0	12363.3	223.3	12631.3	112.4	9101.9	112.0
4	Suisse du Nord-Ouest	9013.8	19444.4	154.7	19630.0	140.1	8999.9	137.6
5	Suisse orientale	7794.1	8760.2	1052.6	10023.3	100.1	7739.3	91.9
6	Suisse centrale	8364.6	28365.0	816.1	29344.3	171.3	8312.7	169.5
7	Tessin	7772.9	28042.1	523.6	28670.4	169.3	7767.7	168.2
Type								
0	person alone	5415.5	3464.6	233.9	3745.3	61.2	5336.3	56.8
1	mother/father alone + 1child	6061.4	25295.6	8570.8	35580.6	188.6	5991.2	150.8
2	mother/father alone + 2 or more children	6996.2	54080.8	6764.5	62198.2	249.4	6946.1	227.6
3	couple	9365.5	7655.9	161.4	7849.6	88.6	9369.4	85.6
4	couple with 1 child	9956.6	20910.4	601.7	21632.5	147.1	10040.3	144.4
5	couple with 2 children	9959.2	9620.2	557.3	10288.9	101.4	9934.9	95.0
6	couple with 3 or more children	10004.0	23874.6	1316.1	25453.9	159.5	10015.2	152.8
7	other	9257.7	56091.0	3452.8	60234.4	245.4	9301.6	237.9
Status								
0	other	5150.2	41417.2	3303.8	45381.7	213.0	4953.0	190.5
1	salaried	9260.7	2665.4	39.1	2712.4	52.1	9273.5	50.8
2	independent (without farmer)	8875.3	37731.2	1086.5	39035.0	197.6	8790.9	192.2
3	farmer	6336.3	68696.6	32615.5	107835.2	328.4	5887.7	238.5
4	unemployed	4237.2	25549.6	356.3	25977.2	161.2	4103.8	134.4
5	pensioner	5917.6	8486.0	84.0	8586.9	92.7	5907.4	90.5

Chapter 6

Summary

Workpackage 4 of the DACSEIS project is dedicated to the evaluation of the statistical software taking survey features into account, emphasising especially the European practices in statistics making in National Statistical Institutes. The evaluation provides an overview on the current situation of the software with studies on theoretical and practical aspects available. One vital reason of this evaluation is to give the end-user sufficient information to make a decision regarding his or her own needs. The results of the workpackage can be found in deliverable D4.1 and D4.2 (Evaluation of Software for Variance Estimation in Complex Surveys).

It was important to have several testing partners within the DACSEIS project in order to have an extensive and balanced view on the software. The principle of having at least two testing partners for one software necessitated that as well. The evaluation organisations were *Statistics Finland* (main evaluator), *Statistics Netherlands*, *Swiss Federal Institute of Technology*, *University of Southampton* and *University of Tübingen*. At the first phase of the project the evaluation criteria were described. These criteria defined what aspects are studied and what limitations are set in order to keep the evaluation task feasible. The criteria part included detailed advice for the evaluation partners on studying theoretical, computational, documentational and other aspects.

The software chosen for evaluation were divided into two categories: *general software* and *specialised software*. The general software were defined to be developed for broader use than only for survey sampling problems (which might be dealt with in a specific part of the software). *SAS*, *Stata* and *SPSS* currently have calculation features which take the sampling design into account. In addition, *S-Plus* and *R* give an environment for an advanced methodologist to make corresponding calculations by himself/herself; however, these two software were not included in the detailed evaluation with calculations using DACSEIS sample data files, because they are not suitable as such (without programming or other construction efforts) for an ordinary user. The *specialised software* were defined as concentrating primarily on survey estimation with specific variance estimation abilities. *Bascula, Clan, Poulpe, Sudaan* and *WesVar* were chosen to that group. In addition, it was then decided that *GSSE* would be described at the general level without a detailed evaluation including calculations, albeit quite recently its name has changed to *Genesees*, having many new features with a possibility for a free acquisition of the product from Statistics Italy (with registration).

SAS is a very versatile commercial package for all kinds of needs connected to data in an electronic form. There are both menu-based and programmable ways of working. Programs can be built with a large variety of procedures and operations for the data. Furthermore, macro language, application building system and matrix language can be applied for very specific purposes. There are many analytical and graphical tools either in a procedure or menu form; the latter alternative is a better choice for less experienced users. There are several (in some cases user or organisation-made) programs and macros for different purposes.

SAS includes fixed procedures for sample selection and survey analysis (descriptive statistics and frequency tables with corresponding standard errors and confidence intervals; regression model and logistic model tools taking the design into account). There are many designs for sampling, but only basic designs (stratification, pps sampling, cluster sampling, two-phase sampling with primary sampling unit variation) for calculations. The SAS survey procedures *do not include* any estimation methods utilising auxiliary information (e.g. GREG, calibration, non-response adjustments). SAS is using the Taylor approximation for variance estimation of non-linear estimators.

S-Plus and **R** are languages and environments for statistical computing and graphics, based on the S language. The term "environment" is intended to characterize R or S-Plus as fully planned and coherent systems, rather than incremental accretions of very specific and inflexible tools, as sometimes occurs with other data analysis software. They are integrated suites of software facilities for data manipulation, calculation and graphical display. R is free of charge. S-Plus is commercial software with menus and dialog boxes providing a framework for easier programming and processing.

S-Plus and R do not provide any in-built survey tools, but the user communities provide some solutions for survey problems as well, e.g. the R 'survey' package for some simple survey problems and packages for resampling methods.

SPSS (Superior Performing Statistical Software) is a modular, tightly integrated, commercial product family including procedures with a variety of statistical methodology and data processing properties. It is a menu-controlled system with easy and quick analysis facilities, having separate windows for different purposes (e.g. data, variable properties, analysis, programming). Analysis routines can be saved or programmed with a specific language (Sax Basic). Special tools for different purposes can be added to the software, e.g. data mining, database analysis and market research, as well as missing value analysis and complex samples.

SPSS Release 12 has the module "SPSS Complex Samples", which includes four procedures offering the possibility to draw and to analyse a stratified, pps, clustered or multistage sample (with primary sampling unit variation). The calculations are concentrated on some simple descriptive statistics and cross-tabulations with standard errors, design effects, confidence intervals and hypothesis tests. SPSS as such *do not include* any estimation methods utilising auxiliary information (e.g. GREG, calibration, non-response adjustments). However, there is a calibration tool *g-Calib* provided by Statistics Belgium for more advanced estimation. SPSS is using the Taylor approximation for variance estimation of non-linear estimators.

Stata is a general software package for a broad scope of standard statistical analyses, from simple descriptive operations to linear regression, logistic regression, Poisson regression, estimation of probit models, test of independence of two-way tables, estimation and testing hypotheses of linear combinations of parameters. This commercial package is widely used in the statistical community. It is mainly command-driven with a few menus. Data interaction is conducted with text data or Stata data files. There are a variety of graphical and analytical tools.

Stata offers the possibility to account for stratified, pps, cluster or multistage (with primary sampling unit variation) sampling design features for estimation and variance estimation of some simple descriptive statistics and tabulations. Furthermore, for modelling and testing there is a family of survey commands taking the design features into account. The Stata survey commands *do not include* any estimation methods utilising auxiliary information (e.g. GREG, calibration, non-response adjustments). Stata is using the Taylor approximation for variance estimation of non-linear estimators.

Bascula is a software package for calculating weights for all units in a sample using auxiliary information. The interface is menu-based, and it is part of the Blaise System. Blaise is an integrated system for survey processing, focusing on data entry/editing, (telephone) interviewing, electronic data capture, management of surveys, manipulation of data and metadata.

The methods for utilising auxiliary information in weighting and variance estimation include post-stratification, linear weighting for GREG estimation and multiplicative weighting for calibration.Bascula can use the computed weights to estimate population totals, means and ratios as well as standard errors based on Taylor linearisation and/or balanced repeated replication (BRR). For the purpose of variance estimation stratification, pps sampling, cluster sampling and multi-phase sampling (with primary sampling unit variation) are supported.

Clan is a non-commercial SAS-based macro package, originally developed for different variance estimation problems at Statistics Sweden. The operations are carried out with a few different macro expressions in SAS programming. These macros can be run among the normal SAS programs, when the library of the core macros of Clan is defined.

Clan is a very versatile tool for both complex designs and various methods concerning estimation and non-response correction methods. A variety of sampling designs can be dealt with in the program: stratification, pps sampling (also Pareto πps), cluster sampling, multi-stage sampling (with primary sampling unit variation), two-phase sampling, rotating scheme, network sampling and subsampling of non-respondents. The estimation methods include GREG estimation, calibration (with a possibility for weight construction), calibration for non-response, response homogeneity groups and partially overlapping domain totals. All kinds of functions of totals can be defined as the parameters to be estimated. The variance for the estimators of these parameters is estimated with the Taylor linearisation method.

Genesees is non-commercial software for variance estimation, developed in the Italian National Institute of Statistics (ISTAT) originally for the needs of business and household surveys. It can be considered as an update of the software GSSE with some enhancements concentrating especially on the interface. It works in the SAS environment, and the interface is based on screens for selecting input variables and parameters and some other definitions. Although the interface is in Italian, the expressions in the interface are explained in detail in the Users' guide.

Genesees is based on the theory of calibration estimation utilising the possibility to estimate the variance with the Taylor linearisation for the GREG estimators. Furthermore, Genesees deals with stratified multistage sampling (with primary sampling unit variation) and pps sampling. The parameters to be estimated are limited to totals, means, proportions and ratios. Other auxiliary information methods than calibration/GREG are not available in Genesees.

Poulpe is a non-commercial software for variance estimation, originally created for the needs of the French Household Survey and the French Business Survey (the main developer Jean-Claude Deville). The software, interacting via screens, utilises the SAS environment, based on several macros created for calculation purposes. On the other hand, the user can calculate with macros straight in the program flow without screens. The interface and most of the documentation is in French, which may be a limitation for some user groups.

Poulpe can deal with a large variety of sampling designs, estimation methods and non-response adjustments within a unified theoretical structure. Stratification, cluster sampling, multi-stage sampling (with accurate variance estimation at each stage), unequal probability sampling (with and without replacement), systematic sampling, rotating schemes and multi-phase sampling can be dealt with Poulpe. There are advanced estimation methods available for variance estimation: calibration, ratio estimator, post-stratification. The unit non-response can also be treated in Poulpe by considering an additional phase (Poisson sampling) in the sampling design. The flexible 'function of totals' technique enables the definitions of rather complex parameters as well. The variance estimation is based on the Taylor linearisation of these functions of totals.

Sudaan, a commercial package for statistical analysis of correlated data, is either an SAS-based module or a separate system. It is created and further developed by RTI. The main importance is on the analysis tools; the variance estimation practices come along with these aims. There are many procedures for different analytical needs (linear models, logistic regression, log-linear regression, modelling categorical outcomes, proportional hazards modelling for failure time outcomes), all taking the sampling design into account. A procedure-driven programming language looks familiar to the SAS users.

The sampling designs available for calculations include stratification, cluster sampling, pps sampling (with and without replacement), multi-stage sampling (with both primary and secondary sampling unit variation included). The variance estimation methods include both the Taylor linearisation as well as sample reuse methods (Balanced Repeated Replications, Delete-1 Jackknife and Replicate Weight Jackknife). Excluding post-stratification, the methods utilising auxiliary information at the estimation phase are not available in the procedures. There are four different alternatives of the design effect to be chosen. Standardisation and contrasting subgroup estimators are specific features available in descriptive procedures. Sudaan calculates also the standard error for the median and other quartiles.

WesVar is a commercial software package for computing estimates and replicate estimates that properly reflect complex sampling and estimation procedures. This menu-driven system provides a user-friendly tool for various variance estimation tasks.

WesVar is flexible and can be used with complex sampling designs including multistage, stratified and unequal probability samples. The replicate variance estimates can reflect many types of estimation schemes, such as non-response adjustment, post-stratification, raking and ratio estimation. WesVar can calculate means and proportions with their variance estimates. It is also easy to use WesVar to compute variance estimates for complex functions of estimates, including ratios, differences of ratios and log-odds ratios. WesVar calculates standard errors and confidence intervals for the estimates specified by user and calculates chi-square tests of independence for two-way tables of weighted frequencies. WesVar computes estimated coefficients for linear and logistic regression models and tests the significance of subsets of linear combinations of parameter estimates.

Test with problems in data and estimation. A small artificial data set was used in order to study the performance of the software (SAS, SPSS, Stata, Bascula, Clan, Sudaan, WesVar) in different 'non-clean' data situations. The test with *non-imputed item non-response* with weights adjusted to unit non-response provided four different ways of calculating the standard error. It is not self-evident what is the right way to calculate standard error in this case.

Another set of tests provided some *missing, zero and negative weights* and some *weights between zero and one* (Bascula and Clan were not included, because they create the weights based on given information). All the software provided the same standard error results except WesVar for the case of some missing weights. Two software showed a warning about the situation (SAS, WesVar). For zero weights there were four different standard error values, and two warnings (SAS, Sudaan). Stata refused to calculate with negative weights, for others there were three different standard error values. All except SPSS provided a warning. Weights between zero and one did not have any effect to the calculations: all the results were the same and there were no messages.

The third set of tests included contradictory design information. At first, the wrong stratum code was given, and in this case SAS and SPSS gave results. All software except SAS gave a warning. Secondly, one stratum size smaller than the sample size was given. WesVar does not utilise stratum sizes, and thus results were given in this case. SPSS gave results as well, though utilising the wrong information. All the others refused to calculate; they had also a message about the situation.

Finally, a stratum with one primary sampling unit was created. Stata, Sudaan and WesVar did not provide any results in this case. The standard error results of Bascula differed from the other results, due to the stratum collapsing conducted by Bascula. SPSS and Clan did not give any warning.

Experiences from the DACSEIS data set testing. *Three sample data were selected* from the DACSEIS universes representing Finnish Labour Force Survey (FLFS), Swiss Household Budget Survey (SHBS) and German Microcensus (GMC). Those data were used for software evaluation purposes, following the design and estimation requirements in harmony with the structure of the simulation studies. However, the testing procedure included only such methods *which were available in the software* and *produced proper variance estimates taking the survey design into account.* Thus e.g. the imputation methods were not tested, because there were no software to deal with them in that sense. Two estimators were used for every survey: a *Horvitz-Thompson* estimator and a *GREG/calibration estimator* utilising auxiliary information from the population level. Eight software were tested with the DACSEIS data sets (SAS, SPSS, Stata, Bascula, Clan, Poulpe, Sudaan and WesVar). One computer in Statistics Finland was used as a reference machine where all the software were used.

When all the data sets and the unit non-response patterns were considered, every software provided the same estimates of the parameters to be studied, so in that sense the software are equal. The variance estimators based on the Taylor linearisation in various software were at the same level as well. Furthermore, the resampling variance estimators were equal or nearly equal to the linearisation estimators, except the artificial Balanced Repeated Replication in Bascula, which gave variance estimates differing greatly from others (note that the Taylor-based variance estimates of Bascula were parallel with the results of other software). The parameters to be estimated here were rather simple (*total of unemployed and unemployment rate for FLFS and GMC; mean income and mean expenditure for SHBS*), so there should be rather marginal differences between the variance estimation methods with these large data sets. The differences between the software came in three main criteria: *1) how advanced the software is regarding the designs and the estimation methods, 2) what properties there are available serving the tasks needed in surveys, 3) how fast the software is and how the large data sets can be dealt with.*

Concerning the test situation, the sampling designs in question (stratified simple random sampling without replacement, stratified cluster sampling) were available in all software. Every software produces the *Horvitz-Thompson estimator* with a variance estimator for these parameters to be estimated (provided that the size of the data set is not too big). The *GREG estimators* with proper variance estimation could be obtained only in Bascula, Clan and Poulpe. In addition to these three software containing also calibration there was a raking method in WesVar for performing a *calibration estimator*.

One important factor is how the data preparations (e.g. stratum sizes, number of respondents in strata, finite population corrections, weights, classifications and/or recoding) can be conducted. The required preparations in full were possible only in SAS, SPSS and Stata. Recoding and classification were available in Bascula, Sudaan and WesVar. Otherwise the preparations had to be done in another environment (we used SAS in this test pattern).

The GMC data sets provided a good test for the speed and memory use of the software. Several strata including only one primary sampling unit required the procedure of strata collapsing before calculations (some software refuse to calculate in the one PSU case). For even the smallest regions the replication weight production for primary sampling units in WesVar caused considerably longer processing times than in other software. As a limit, WesVar could deal with Schleswig-Holstein with 29140 records, 35 strata and 1680 primary sampling units. Correspondingly, the largest data set with a default setting Stata was Niedersachsen with 74334 records. However, the memory of Stata can be extended in the session with a specific statement and larger data sets may be run respectively. In general, Stata performed the calculations rather fast. SAS-based software, Bascula and SPSS could be utilised for the full GMC data (831004 observations). SAS (proc surveymeans), SPSS and Sudaan conducted the task sufficiently, although SAS (surveymeans) had memory difficulties with some domain categorisations, at least in the test computer. The advanced software doing additional calculations for more complex demands required more time.

The replicate weights made the WesVar data sets very large even at the level of a few thousand observations (e.g. for 5000 observations of FLFS 442 Mb). Poulpe has a structure which is prepared for very complex design and estimation processes; it produced a large number of different data sets and in total they reserved a lot of space, though much less than WesVar. In Bascula the survey data set in the Blaise form was rather large. Also Clan made additional data sets for estimation purposes, but the amount of memory they reserved was much less than in Poulpe. In practice the SAS survey data set is bigger than the corresponding Stata data set and the SPSS data set. Proc Surveymeans of SAS and Sudaan did not produce any additional data sets with these tasks.

Speed test with R and SAS-based software. In addition to the tests pattern for three DACSEIS data, a simplified test was conducted with the GMC data sets (in all 831 004 observations) in order to test the speed of R coding and SAS-related survey software (Proc Surveymeans, Sudaan, Clan). The task was reduced to Horvitz-Thompson estimation at the general level as if the sample was selected as element sampling. The results with different data combinations revealed that with this task R was faster in preparations and SAS in estimation. Sudaan (Descript procedure) was slightly slower than SAS Surveymeans. Clan conducts many additional data phases in SAS and thus its calculations took longer than for the others. Still the performance of Clan was quite fair.

Testing a SAS procedure for multiple imputation. The new SAS procedure MI creates multiply imputed data sets for incomplete multivariate data. It was tested with the Swiss Household Budget Survey data with the income variable containing 5.6 % of item non-response. The standard error of mean income was computed based on five imputed datasets by region, household type, and socio-economic status. Results were compared with results obtained from dataset without item non-response. At the overall level the standard error of the mean income was overestimated by 4.0 %. Some notable overestimates of standard errors could be found in some socio-economic levels of study, e.g. the farmers as the largest (8 %). However, those groups were relatively small, and for other classifications the bias of the standard error was not large.

Recommendations for the end-users with comments. *SAS* survey procedures, *SPSS* 'complex samples' and *Stata* 'survey data analysis' can be recommended for simple survey tasks. *Stata* (when compared with the other two) has most advanced modelling tools taking the sampling design into account. *SAS, SPSS* and *Stata* can deal with hundreds of thousands of observations sufficiently. All three software can be considered user-friendly, the required structures are rather easy to be conducted in menus (SPSS, Stata) or in programming (SAS, SPSS).

When there are various needs for data analysis of complex surveys, *Sudaan* is a good alternative. It deals with more complex sampling designs than *SAS*, *SPSS* and *Stata*, and *Sudaan* has a much wider range of analysis tools for surveys. Also large tasks were performed rather easily in *Sudaan*. However, *Sudaan* requires some preliminary preparations (e.g. weights, stratum sizes) for the data to be used in calculations: another system is needed for that process. The most natural choice is *SAS*, because there is a *SAS*-callable version of *Sudaan*. *Sudaan* follows the structure of SAS programming.

WesVar emphasises easiness and general applicability of resampling methods at the expense of speed. The software is very versatile (including e.g. poststratification and raking), and also functions of great complexity can be included in the estimation process. However, *WesVar* is not at its best with large scale surveys with a lot (several thousands) of primary sampling units. In that kind of cases both speed and memory comparison results were poor when compared with others. As in *Sudaan*, also *WesVar* requires some preliminary preparations for the data, which cannot strictly be done in *WesVar*. Despite some minor inconviniencies while using *WesVar*, the software is user-friendly with its menus/screens.

Bascula is a part of *Blaise* (an integrated system for survey processing), and it might not be reasonable to purchase *Blaise* only for the use of *Bascula*. When having *Blaise* available, *Bascula* provides an advanced weighting tool (linear or multiplicative weighting) with abilities for proper variance estimation based on Taylor's linearisation. The Balanced Repeated Replications method of *Bascula* provides rather unstable variance estimates. However, the calculations can be conducted only for the total, the mean and the ratio. The speed of *Bascula* is not as fast as in *SAS* or *SPSS*. *Bascula* as well requires external preparations for the survey data to be dealt with. When the basic order of the weight and estimate calculations of *Bascula* is understood, the operations can be carried out quite easily.

Clan and *Poulpe* are free-of-charge macro packages, which can be used in *SAS*. They are the most advanced software in comparison, when the sampling designs and especially the estimation methods based on auxiliary information are concerned. In addition to the normal sampling designs, also e.g. network sampling, rotational designs and two-phase sampling can be dealt with. Also additional features for non-response treatment exist in these software. Weights based on both calibration and GREG estimation can be applied if the auxiliary information used in modelling is provided by the user. *Clan* also calculates weights based on the theory of the GREG estimator. *Poulpe* does not create weights: there is another macro package, *Calmar*, available for calibration. The speciality of *Poulpe* is the definition of a sampling tree, in which all the levels of sampling can be expressed in such a way that the system calculates e.g. the stratified multilevel designs with various selection methods accurately. Both software are clearly meant for sophisticated users; the level of theoretical knowledge needed for operations is quite high. Although logical and rather tight, the program code required in *Clan* is not easy to be understood unless the user is experienced with SAS code and to some extent macro language as well (although the well-prepared manual helps). *Poulpe* is constructed in French and there are only few documents in English (not sufficiently). This is a considerable disadvantage of *Poulpe* regarding a large-scale use in Europe. Furthermore, its structure should be developed to a more user-friendly direction. Now the variety of different properties and aspects makes *Poulpe* rather difficult to be understood as a whole. Both software calculate more than the ordinary survey software, thus the processing times are longer, especially in *Poulpe*.

Future aspects of software development. It is obvious that in the near future the commercial general software SAS, SPSS and Stata will mainly maintain the current, rather basic level of variance estimation. If not requested by a substantial portion of users, the companies won't have interest for applying very advanced methods dealing with e.g. model-assisted estimation, calibration, non-response treatment (including response propensity modelling), imputation and other methodologies presented e.g. in other work-packages of the DACSEIS project. Members of the user community of non-commercial R have very recently produced a few promising packages for some advanced methods of surveys. R provides a good platform for development in this branch as well, but the lack of proper interface for non-experienced users is a restricting factor.

There has been some outstanding work by Sudaan in recent decades providing detailed tools for analysis under complex surveys. The inclusion of resampling variance estimation methods is a sign of broadening the scope of the methodology. In general, the use of resampling methods is still rather limited in the software, although those methods could deal with estimators with considerable complexity without any specific effort by the user, as e.g. WesVar can. The construction of the replicate matrices together with speed and memory aspects is a major topic for future development of resampling methods in the software. A new approach in variance estimation, i.e. the jackknife linearisation has provided promising results in tests with some survey data. Thus a special interest is focused on the prototype of Genesees including this method, and whether this could be used more generally.

Model-assisted estimation and calibration have been features of four software: Bascula, Clan, Genesees and Poulpe. Bascula and Genesees have a rather limited scope of parameters, so the development of these methods reaches its peak in Clan and Poulpe. In general, the methodological framework of Poulpe is based on a few simple theoretical rules: with them we can trace the right variance expression whether we are dealing with stratified multistage sampling, multiphase sampling, rotational design, nonresponse effect etc. Clan has also abilities to this kind of situations, and some additional methods are developed for the practical problems appearing in the context of National Statistical Institutes. Both software have a structure which could be expanded to a new methodology. Unfortunately the use of Clan and especially Poulpe is rather difficult, and in that sense there should be efforts for make these programs much user friendlier.

An ideal situation could be a synthesis of the good properties of these software: the easy interface, the speed and the data processing facilities of SAS, SPSS and Stata, the flexibility and the simplicity of R and S-Plus, the methodology in the analysis context in Sudaan, the easiness of resampling with complex estimators in WesVar, the linearised jackknife of Genesees, the easy weight construction in Bascula, a large scale of different methods in Clan for various problems appearing in survey practices, a unified theoretical approach of Poulpe for most of the methods appearing in the survey theory. Moreover, an end-user needs many such methods that are not currently available in these software, including variance estimation of response propensity modelling and various alternatives for imputation variances. Of course a user would be more happy if other accuracy imperfections such as due to frame errors and measurement errors would be included in the same environment. The commercial companies might not conduct such development work, unless there is a clear need for this kind of specialised and general software. But who is willing to finance such a development work? This being the case, in a near future an end-user needs more than one environment for his/her different survey estimation tasks. It should be also noted that some significant and workable developments have been performed more or less as a hobby, and hence it is not guaranteed that all will be maintained well in future. This also suggests that a National Statistical Institute end-user will work with more than one environment.

References

Broeme, P. and Rust, K. (2000): *Strengths and Limitations of using SUDAAN. Stata und WesVarPC for computing Variances from NCES Data Sets.* Working paper. National Center for Education Statistics.

Brogan, D. (1998): *Pitfalls of using standard statistical software packages for sample survey data.* Encyclopedia of Biostatistics, Wiley.

Carlson, B. L. (1998): *Software for statistical analysis of sample survey data.* Encyclopedia of Biostatistics, Wiley.

Davies, P. and Smith, P. (1999): *Model Quality Report in Business Statistics.* SUPCOM Project 97-06. Volume II: Comparison of Variance Estimation Software and Methods.

Deville, J. C. and Särndal, C. E. (1992): Calibration estimators in survey sampling. *Journal of the American Statistical Association* **87**, 376–382.

Eurostat (2003): *Variance estimation methods in the European Union. Monographs of Official Statistics.* Eurostat.

Falorsi, S., Righi, P. and Solari, F. (2002): A prototype of GENEralised software for sampling estimates and errors in surveys GENESEES in the presence of unit non-response. In *Proceedings of Conference.* COMPSTAT.

Francisco, C. A. and Fuller, W. A. (1991): Quantiles estimation with a complex survey design. *Annals of Statistics* **19**, 454–469.

Horten, N. J. and Lipsitz, S. R. (2001): Statistical computing software reviews. *The American Statistician* **55**, 244–254.

Lumley, T. (2003): Analysis of complex survey samples. Technical report, Department of Biostatistics, University of Washington.

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

Schafer, J. L. (1997): *Analysis of Incomplete Multivariate Data.* Chapman & Hall, London.

Smith, P., Chambers, R., Underwood, C. and Full, S. (1998): Comparison of variance estimation software for sample surveys with particular application to business surveys. NTTS Conference Paper, Sorrento (Italy). `http://europa.eu.int/en/comm/eurostat/research/conferences/ntts-98/pape%rs/cp/080c.pdf`.

Tompkins, L. and Siller, A. (2001): Analyzing complex sample survey data: A new beginning. Proceedings of NESUG01 Conference.

Venables, W. N. and Ripley, B. D. (1999): *Modern applied statistics with S-Plus.* Third edition. Springer Verlag.

Woodruff, R. S. (1971): A simple method for approximating the variance of a complicated estimate. *Journal of the American Statistical Association* **66**, 411–414.

Yung, W. and Rao, J. N. K. (1996): Jackknife lineraization variance estimators under startified multi-stage sampling. *Survey Methodology* **22**, 22–31.