Deliverable Number: 2.11

Deliverable Title: Guidelines for computing longitudinal weights and sampling documentation

Work Package: 2 (Representing the population)

Deliverable type: Report

Dissemination status: Public

Submitted by: SHARE ERIC

Authors: Michael Bergmann, Giuseppe De Luca.

Date Submitted: July, 2018

www.seriss.eu   @SERISS_EU

SERISS (Synergies for Europe's Research Infrastructures in the Social Sciences) aims to exploit synergies, foster collaboration and develop shared standards between Europe's social science infrastructures in order to better equip these infrastructures to play a major role in addressing Europe's grand societal challenges and ensure that European policymaking is built on a solid base of the highest-quality socio-economic evidence.

The four year project (2015-19) is a collaboration between the three leading European Research Infrastructures in the social sciences - the European Social Survey (ESS ERIC), the Survey of Health Ageing and Retirement in Europe (SHARE ERIC) and the Consortium of European Social Science Data Archives (CESSDA AS) - and organisations representing the Generations and Gender Programme (GGP), European Values Study (EVS) and the Wage Indicator Survey.

Work focuses on three key areas: Addressing key challenges for cross-national data collection, breaking down barriers between social science infrastructures and embracing the future of the social sciences.

Guidelines for computing longitudinal weights and sampling documentation

July 2018

Abstract

This report describes the Stata programs that are provided to users of the Survey of Health, Ageing and Retirement in Europe (SHARE) in order to compute their own calibrated longitudinal weights (part 1). Since the basic units of analysis can be either individuals or households, we illustrate the computation of both calibrated longitudinal individual weights for inference to the target population of individuals and calibrated longitudinal household weights for inference to the target population of households. In addition, we describe standards in the documentation of the sampling process and the resulting gross samples that can be seen as a blueprint for large-scale cross-national surveys (part 2).

Contents

Introduction

The focus of this deliverable of the Synergies for Europe's Research Infrastructures in the Social Sciences (SERISS) project, Work Package 2 (Representing the population), Task 2.3 (Weighting for complex survey design) is twofold. First, we address a typical problem for panel surveys which provide longitudinal weights to compensate for problems of sample attrition: The disproportionate increase in the number of possible sets of weights, the more waves have been conducted so far. As a consequence it gets more and more impractical to release all possible weights along with the actual data and the panel study hence has to choose the "right" weights. Those sets of weights might be appropriate for most but not all users. Therefore, we give detailed explanations how users can compute their own calibrated longitudinal weights by using the sreweight Stata command implemented by Pacifco (2014). Second, we describe standards in the documentation of the sampling process and the resulting gross samples that can be seen as a blueprint for large-scale cross-national surveys. Essentially, this report hence gives guidelines for users as well as managers of other surveys on how to a) construct longitudinal weights that fulfill the specific needs of users' own analyses and b) illustrate possibilities of carefully documenting the often complex process of sampling. Thereby, the latter is indispensable for the former, which is why we present the two aspects together here.

Regarding the first part of this report, we provide examples using data and documentation tools from the Survey on Health, Ageing and Retirement in Europe (SHARE), where calibration is used to compensate for problems of attrition in the longitudinal samples of different waves. The longitudinal dimension of the SHARE data, as well as their multi-national nature, allow us to illustrate the great generality of the calibration approach. We focus on the computation of the calibrated longitudinal weights between the first two waves, but the general proceeding can be easily adapted to any combination of subsamples/waves. The following guidelines and provided programs are thus of great help also for users of other panel surveys, such as the Generations and Gender Programme (GGP). Since the basic units of analysis can be either individuals or households, calibrated longitudinal weights are computed at the individual level for inference to the target population of individuals and at the household level for inference to the target population of households. Both types of calibrated weights allow us to adjust the original design weights so that weighted survey estimates match the known population totals (the so-called calibration margins) for a given set of control variables. As discussed in SERISS deliverable 2.9 (Rossetti 2017), calibration margins for the target populations investigated by SHARE are taken from the regional demographic statistics given by Eurostat. This external source of data contains population figures and number

of deaths by year, NUTS1 regional area, gender and age. The rationale behind the calibration adjustment is that by ensuring consistency between the sample and the population distributions of these benchmark variables, the calibrated weights will also perform well when applied to other study variables of interest.

We describe the protocol that is followed in SHARE to harmonize and document the sampling procedures, that benefited considerably through the cooperation with the other surveys within this SERISS workpackage. Several steps are involved in this second part of the report: Firstly, each country drawing a baseline or refreshment sample is initially required to provide a Sample Design Form (SDF) containing a complete description of both the chosen sampling frame and the associated sampling design. This sampling proposal has, secondly, to be evaluated and approved by the SHARE Central coordination before countries are allowed to draw the sample according to the approved sampling design process. Finally, each country provides a gross sample file containing the list of selected households, the associated sampling frame information needed for the computation of selection probabilities (e.g. household-level and population-level information about stratification and clustering), the household-level information about regional codes, and (if any) additional auxiliary variables that could be used for ex-post compensation of non-sampling errors. Both, the SDF as well as the gross sample file, are provided along this report and can be adapted to the specific needs of other surveys.

The remainder of the report is organized as follows. In the first part, Section 1 illustrates how to extract individual and household level SHARE databases for longitudinal studies, while Section 2 describes the Stata code for constructing the underlying vectors of calibration margins. Sections 3 and 4 show, respectively, the computation of calibrated longitudinal individual and household weights. Section 5 offers some conclusions on the computation of longitudinal weights that account for problem of attrition in longitudinal surveys. In second part, Section 6 gives an overview about the typical sampling process in SHARE that consists of the main data collection as well as two rounds of preparatory data collection to test the instrument as well as all necessary operational sequences. Section 7 and 8 then describe innovative tools that are used to document the various sampling designs and the corresponding gross samples in the SHARE countries drawing a refreshment sample. Finally, Section 9 offers some conclusions on the documentation of the sampling process.

Part 1: Guidelines for computing own longitudinal weights

1 Longitudinal SHARE data

SHARE data can be used for a variety of cross-sectional and longitudinal studies. For the purposes of cross-sectional studies (see D2.10 for the computation of cross-sectional weights), the target population in each country consists of persons of 50 years or older at a particular point in time and their possibly younger spouses/partners, who speak (one of) the official language(s) of the country (regardless of nationality and citizenship) and who do not live either abroad or in institutions such as prisons and hospitals during the entire fieldwork period (Bergmann et al. 2017). For longitudinal studies, the target population is typically defined as the cross-sectional target population at the beginning of a time reference period that survives up to the end of the period considered (see, e.g., Lynn, 2009). These target populations could also be defined in terms of households as all households with at least one member belonging to the cross-sectional/longitudinal target population of individuals. This section shows how to extract longitudinal samples of individuals and households for representing these alternative variants of the SHARE target population.

1.1 Individual longitudinal data

Our first example concerns the sample of individuals interviewed in the first two waves of SHARE. We must pay specific attention to correctly select the longitudinal sample of respondents that were present in both waves (balanced sample). The underlying Stata code is as follows:

```
. *---------------------------------------------------------------------------
. * Extract individual data for longitudinal sample
. *---------------------------------------------------------------------------
. local SHARE "C:\DATA\SHARE"
. local wi = 1 // initial wave //
. local wf = 2 // final wave //
. *---------------------------------------------------------------------------
. global w = "l_`wi´_`wf´"
. * Identify longitudinal (balanced) sample
. forvalues wj=`wi´(1)`wf´ {
  2.          qui use "data/sharew`wj´_rel6-0-0_cv_r", clear
  3.          qui keep if interview==1
  4.          qui keep mergeid
  5.          if `wj´>`wi´ qui merge 1:1 mergeid using temp_balanced_ii,keep(3) nogen
  6.          sort mergeid
  7.          qui save temp_balanced_ii,replace
  8. }
.
. * Merge longitudinal sample
. qui use "data/sharew`wi´_rel6-0-0_cv_r", clear
. keep mergeid hhid1 country gender yrbirth interview
. qui gen age_w1 = 2004 - yrbirth if yrbirth>0
```

```
. qui keep if interview==1

. drop interview

. qui merge 1:1 mergeid using "data/sharewX_rel6-0-0_gv_longitudinal_weights_w1w2",        ///
>        keepus(dw_w1 cliw_b) assert(1 3) nogen

. qui merge 1:1 mergeid using "data/sharew`wi´_rel6-0-0_gv_housing",                       ///
>        keepus(nuts1_2003) assert(3) nogen

. qui merge 1:1 mergeid using temp_balanced_ii

. gen balanced = _merge==3

. cap lab drop balanced

. lab define balanced 0 "not all waves" 1 "all waves"

. lab value balanced balanced

. tab country balanced,m
```

Country identifier	balanced not all w	all waves	Total
Austria	437	1,132	1,569
Germany	1,398	1,599	2,997
Sweden	959	2,090	3,049
Netherlands	1,167	1,801	2,968
Spain	824	1,492	2,316
Italy	782	1,771	2,553
France	1,090	2,032	3,122
Denmark	443	1,263	1,706
Greece	447	2,450	2,897
Switzerland	267	730	997
Belgium	942	2,868	3,810
Israel	762	1,688	2,450
Total	9,518	20,916	30,434

```
. drop if balanced==0
(9,518 observations deleted)

. drop _merge balanced

.

. * Save data

. qui compress

. sort mergeid

. qui saveold mydata_long_ind, replace

.

. describe
Contains data from mydata_long_ind.dta
  obs:        20,916
  vars:            9                            12 Jan 2018 12:55
  size:     1,505,952                           (_dta has notes)
```

variable name	storage type	display format	value label	variable label
mergeid	str12	%12s		Person identifier (fix across modules and waves)
hhid1	str11	%11s		Household identifier (wave 1)
country	byte	%14.0g	country	Country identifier
gender	byte	%10.0f	gender	Male or female
yrbirth	int	%10.0f	dkrf	Year of birth
age_w1	int	%9.0g		
dw_w1	double	%10.0g		Design weight - wave 1
cliw_b	double	%10.0g		Calibrated longitudinal individual weight - panel: 1_2
nuts1_2003	str27	%27s		NUTS level 1: nomenclature of territorial units for statistics

```
Sorted by: mergeid

. list mergeid hhid1 country gender yrbirth in 1/5, sepby(hhid1) noobs
```

mergeid	hhid1	country	gender	yrbirth
AT-000327-01	AT-000327-A	Austria	Male	1952
AT-000327-02	AT-000327-A	Austria	Female	1955
AT-001816-02	AT-001816-A	Austria	Male	1948
AT-002132-01	AT-002132-A	Austria	Female	1933
AT-004234-01	AT-004234-A	Austria	Male	1950

The release 6.0.0 of the SHARE data includes 20,916 respondents that were present in the first two waves, with national samples ranging from a minimum size of 730 observations in Switzerland and a maximum size of 2,868 observations in Belgium. In total, there are 9,518 respondents interviewed in wave 1 but not in wave 2 that have been dropped from our longitudinal sample. The database `mydata_long_ind` contains information on the individual and household identifiers, the country indicator, basic demographic characteristics of the respondents (gender, year of birth, age at the time of the wave 1 interview, and NUTS level 1), the design weights in wave 1, and the calibrated longitudinal individual weights for the balanced panel from wave 1 to wave 2. Additional information on the design weights for SHARE can be found in Klevmarken et al. (2005), Lynn et al. (2013), De Luca et al. (2015) and Bergmann et al. (2017). In Section 3, we shall illustrate how to reproduce the calibrated weights `cliw_b`.

1.2 Household longitudinal data

In addition to individual level information, SHARE collects household level data about consumption, income and wealth. For this type of variables, we are often interested in constructing a sample of households. In this section we show how to extract the balanced panel of households in the first two waves of the SHARE panel, that is all households with at least one member belonging to the corresponding longitudinal sample of individuals. Our Stata code for this type of database is as follows:

```
. *---------------------------------------------------------------------
. * Extract household data for longitudinal sample (e.g. from wave 1 to wave 2)
. *---------------------------------------------------------------------
. local SHARE "C:\DATA\SHARE"
. local wi = 1 // initial wave //
. local wf = 2 // final wave   //
. global w = "l_`wi'_`wf'"
. * Identify longitudinal (balanced) sample
.
. forvalues wj=`wi'(1)`wf' {
  2.        qui use "data/sharew`wj'_rel6-0-0_cv_r", clear
```

```
  3.          gen str9 hhmergeid = mergeid
  4.          keep hhmergeid
  5.          bys hhmergeid:keep if _n==1
  6.          if `wj´>`wi´ qui merge 1:1 hhmergeid using temp_balanced_hh,keep(3) nogen
  7.          sort hhmergeid
  8.          qui save temp_balanced_hh,replace
  9. }
.
. * Merge longitudinal sample
.
. use "data/sharewX_rel6-0-0_gv_longitudinal_weights_w1w2",replace
. gen str9 hhmergeid = mergeid
. qui bys hhmergeid: keep if _n==1
. keep hhmergeid dw_w1 clhw_b
. qui save temp_long,replace
. qui use "data/sharew`wi´_rel6-0-0_cv_r", clear
. gen str9 hhmergeid = mergeid
. keep hhmergeid hhid1 country gender yrbirth interview
. qui gen age_w1 = 2004 - yrbirth if yrbirth>0
. bys hhid1: gen member = _n
. rename gender gender_
. rename yrbirth yrbirth_
. rename age_w1 age_w1_
. rename interview interview_
```

```
. * Reshape the data at the household level
.
. reshape wide gender_ yrbirth_ age_ interview_, i(hhid1) j(member)
(note: j = 1 2 3 4 5 6 7 8 9 10)
Data                                   long   ->   wide
───────────────────────────────────────────────────────────────
Number of obs.                        43993   ->   20809
Number of variables                       8   ->      43
j variable (10 values)               member   ->   (dropped)
xij variables:
                                    gender_   ->   gender_1 gender_2 ... gender_10
                                   yrbirth_   ->   yrbirth_1 yrbirth_2 ... yrbirth_10
                                     age_w1_  ->   age_w1_1 age_w1_2 ... age_w1_10
                                  interview_  ->   interview_1 interview_2 ... interview_10
───────────────────────────────────────────────────────────────
. qui merge 1:1 hhmergeid using temp_long,                         ///
>        keepus(dw_w1 clhw_b) assert(1 3) nogen
. qui merge 1:m hhid1 using "data/sharew`wi´_rel6-0-0_gv_housing", ///
>        keepus(nuts1_2003) assert(3) nogen
. qui bys hhid1: keep if _n==1
. qui merge 1:1 hhmergeid using temp_balanced_hh,assert(1 3)
. gen balanced = _merge==3
. cap lab drop balanced
. lab define balanced 0 "not all waves" 1 "all waves"
. lab value balanced balanced
. noi tab country balanced,m
     Country │       balanced
  identifier │ not all w  all waves │     Total
─────────────┼──────────────────────┼──────────
     Austria │       273        900 │     1,173
     Germany │       874      1,119 │     1,993
      Sweden │       517      1,620 │     2,137
 Netherlands │       606      1,340 │     1,946
       Spain │       467      1,219 │     1,686
       Italy │       505      1,267 │     1,772
      France │       601      1,452 │     2,053
     Denmark │       238        937 │     1,175
      Greece │       252      1,729 │     1,981
 Switzerland │       153        553 │       706
     Belgium │       519      2,000 │     2,519
      Israel │       353      1,315 │     1,668
─────────────┼──────────────────────┼──────────
       Total │     5,358     15,451 │    20,809
. drop if balanced==0
(5,358 observations deleted)
. drop _merge balanced
.
. * Save data
.
. qui compress
. sort hhid1
. qui saveold mydata_long_hhs, replace
```

```
. describe

Contains data from mydata_long_hhs.dta
  obs:         15,451
  vars:           46                            12 Jan 2018 12:55
  size:      1,792,316                          (_dta has notes)
─────────────────────────────────────────────────────────────────────────────
              storage   display    value
variable name   type    format     label     variable label
─────────────────────────────────────────────────────────────────────────────
hhid1         str11     %11s                  Household identifier (wave 1)
gender_1      byte      %10.0f     gender     1 gender_
yrbirth_1     int       %10.0f     dkrf       1 yrbirth_
interview_1   byte      %21.0g     interview  1 interview_
age_w1_1      int       %9.0g                 1 age_w1_

gender_2      byte      %10.0f     gender     2 gender_
yrbirth_2     int       %10.0f     dkrf       2 yrbirth_
interview_2   byte      %21.0g     interview  2 interview_
age_w1_2      byte      %9.0g                 2 age_w1_

gender_3      byte      %10.0f     gender     3 gender_
yrbirth_3     int       %10.0f     dkrf       3 yrbirth_
interview_3   byte      %21.0g     interview  3 interview_
age_w1_3      int       %9.0g                 3 age_w1_

gender_4      byte      %10.0f     gender     4 gender_
yrbirth_4     int       %10.0f     dkrf       4 yrbirth_
interview_4   byte      %21.0g     interview  4 interview_
age_w1_4      byte      %9.0g                 4 age_w1_

gender_5      byte      %10.0f     gender     5 gender_
yrbirth_5     int       %10.0f     dkrf       5 yrbirth_
interview_5   byte      %21.0g     interview  5 interview_
age_w1_5      byte      %9.0g                 5 age_w1_

gender_6      byte      %10.0f     gender     6 gender_
yrbirth_6     int       %10.0f     dkrf       6 yrbirth_
interview_6   byte      %21.0g     interview  6 interview_
age_w1_6      byte      %9.0g                 6 age_w1_

gender_7      byte      %10.0f     gender     7 gender_
yrbirth_7     int       %10.0f     dkrf       7 yrbirth_
interview_7   byte      %21.0g     interview  7 interview_
age_w1_7      byte      %9.0g                 7 age_w1_

gender_8      byte      %10.0f     gender     8 gender_
yrbirth_8     int       %10.0f     dkrf       8 yrbirth_
interview_8   byte      %21.0g     interview  8 interview_
age_w1_8      byte      %9.0g                 8 age_w1_

gender_9      byte      %10.0f     gender     9 gender_
yrbirth_9     int       %10.0f     dkrf       9 yrbirth_
interview_9   byte      %21.0g     interview  9 interview_
age_w1_9      byte      %9.0g                 9 age_w1_

gender_10     byte      %10.0f     gender     10 gender_
yrbirth_10    int       %10.0f     dkrf       10 yrbirth_
interview_10  byte      %21.0g     interview  10 interview_
age_w1_10     byte      %9.0g                 10 age_w1_

country       byte      %14.0g     country    Country identifier
hhmergeid     str9      %9s
dw_w1         double    %10.0g                Design weight - wave 1
clhw_b        double    %10.0g                Calibrated longitudinal household weight - panel: 1_2
nuts1_2003    str27     %27s                  NUTS level 1: code of territorial units for statistics
─────────────────────────────────────────────────────────────────────────────
```

```
Sorted by: hhid1
. list hhid1 country gender_1 gender_2 yrbirth_1 yrbirth_2 in 1/3, noobs

        hhid1    country   gender_1   gender_2   yrbirt~1   yrbirt~2

     AT-000327-A   Austria       Male     Female       1952       1955
     AT-001816-A   Austria     Female       Male       1943       1948
     AT-002132-A   Austria     Female       Male       1933       1966
```

The release 6.0.0 of the SHARE data includes 15,451 households with at least one member present both in wave 1 and in wave 2 (balanced sample), with national samples ranging from a minimum size of 553 observations in Switzerland and a maximum size of 2,000 observations in Belgium. The remaining 5,358 households of wave 1 are excluded from the database `mydata_long_hhs` because none of their members was interviewed in wave 2. To reproduce the calibrated longitudinal household weights `clhw_b`, we have stored the information about gender, year of birth, age and interview status of all household members in a wide format. The construction of this type of calibrated weights is discussed in Section 4.

2 Calibration margins

This section shows how to construct the vector of calibration margins using external information on population figures and number of deaths stored in the database `margins_nuts1.dta`. For detailed information about this database, we refer the reader to the SERISS Deliverable 2.9. Below we focus on the do-file `CalMar`, which allows creating vectors of population margins for gender and age groups for a specific country. The longitudinal population is obtained by subtracting from the population in the initial year the number of deaths in the following years up to the final one. This do-file creates one scalar and three vectors. The scalar contains the national target population related to the chosen age groups and years (for example population 50+ in 2004). The first two vectors contain a breakdown of the total population by gender-age group (i.e. males and females in the chosen age groups) and NUTS1 regional area, respectively. Stacking together the gender-age group vector and the vector obtained by excluding the first component of the NUTS1 vector yields the vector of calibration margins used to construct the SHARE calibrated weights. The macros needed to initialize the do-file `CalMar` are the country label (macro `cc`), the country number (macro `cc_num`), the reference year (macro `pop_time`), the final year (macro `mort_time`), the number of age groups (macro `age_groups`), and the lower (macro `age_thr_low`) and upper (macro `age_thr_upp`) thresholds of the age groups. The macro `mort_time` (final year) is only used

for computing longitudinal weights, i.e. for population in reference year surviving in final year. For cross-sectional weights this macro must be set to zero. The macro `w` defines simply a label for the wave of interest.

```
. *--------------------------------------------------------------------------------------------
. * Set local macros
. *--------------------------------------------------------------------------------------------
. local cc            ${cc}              // country
. local cc_num        ${cc_num}          // country number
. local pop_time      ${pop_time}        // reference year
. local mort_time     ${mort_time}       // final year
. local w             ${w}
. local age_groups    ${age_groups}      // number of age groups
. local age_thr_low   ${age_thr_low}     // lower thresholds of age groups
. local age_thr_upp   ${age_thr_upp}     // upper thresholds of age groups
. *------------------------------------------------------------------------------
. * Vector of calibration margins from margins_nuts1.dta
. *------------------------------------------------------------------------------
. qui use "margins_nuts1", clear
. qui keep if country=="`cc´"
.
. gen age_mort = age - (year - `pop_time´)
. local age_min: word `age_groups´ of `age_thr_low´
. local age_max: word 1           of `age_thr_upp´
. qui drop if age<`age_min´
. assert age>=`age_min´ & age<=`age_max´
.
. * Joint age-sex classification
. local rname ""
. local t=1
. matrix `cc´_w`w´_P=0
. cap matrix drop `cc´_w`w´_P_AGE_THR
. cap matrix drop `cc´_w`w´_P_SA
. forvalues ss=0(1)1 {
   2.   if `ss´==0 local slab "M"
   3.   if `ss´==1 local slab "F"
   4.   forvalues aa=1(1)`age_groups´ {
   5.     local age_upp: word `aa´ of `age_thr_upp´
   6.     local age_low: word `aa´ of `age_thr_low´
   7.     qui sum pop    if year==`pop_time´       ///
>                         & sex==`ss´              ///
>                         & (age>=`age_low´ & age<=`age_upp´)
   8.     local marg_`t´=r(sum)
   9.     qui sum deaths  if year>=`pop_time´      ///
>                         & year<`mort_time´       ///
>                         & sex==`ss´              ///
>                           & (age_mort>=`age_low´ & age_mort<=`age_upp´)
  10.     local marg_`t´=`marg_`t´´-r(sum)
  11.     assert `marg_`t´´>0
  12.     matrix `cc´_w`w´_P = `cc´_w`w´_P + `marg_`t´´
  13.     matrix `cc´_w`w´_P_SA = nullmat(`cc´_w`w´_P_SA) \ (`marg_`t´´)
  14.     if `aa´==1 local rname "`rname´ `slab´-`age_low´+"
  15.     else       local rname "`rname´ `slab´-`age_low´-`age_upp´"
  16.     if `ss´==0 matrix `cc´_w`w´_P_AGE_THR=nullmat(`cc´_w`w´_P_AGE_THR)\(`age_low´,`age_upp´)
  17.     local t=`t´+1
  18.   }
  19. }
```

```
.  matrix coln `cc´_w`w´_P_SA      =POP
.  matrix rown `cc´_w`w´_P_SA      =`rname´
.  matrix coln `cc´_w`w´_P         =POP
.  matrix rown `cc´_w`w´_P         =TOT
.  matrix coln `cc´_w`w´_P_AGE_THR ="age_thr_low age_thr_upp"
.  matrix list `cc´_w`w´_P
.  matrix list `cc´_w`w´_P_SA
.
.  * NUTS1 classification
.  qui tab nuts1
.  local nreg=r(r)
.  if `nreg´>1 {
.    cap matrix drop `cc´_w`w´_P_NUTS1
.    local rname ""
.    local t=1
.    encode nuts1, gen(REG)
.    forvalue nn=1(1)`nreg´ {
  2.      local nn_lab: label REG `nn´
  3.      qui sum pop    if year==`pop_time´        ///
>                       & sex==2                    ///
>                       & nuts1=="`nn_lab´"
  4.      local marg_`t´=r(sum)
  5.      qui sum deaths  if year>=`pop_time´       ///
>                      & year<`mort_time´           ///
>                      & sex==2                     ///
>                      & (age_mort>=`age_min´)      ///
>                      & nuts1=="`nn_lab´"
  6.      local marg_`t´=`marg_`t´´-r(sum)
  7.      assert `marg_`t´´>0
  8.      matrix `cc´_w`w´_P_NUTS1 = nullmat(`cc´_w`w´_P_NUTS1)\(`marg_`t´´)
  9.      local rname "`rname´ `nn_lab´"
 10.      local t=`t´+1
 11.    }
.    matrix coln `cc´_w`w´_P_NUTS1 =POP
.    matrix rown `cc´_w`w´_P_NUTS1 =`rname´
.    matrix `cc´_w`w´_P_N          =`cc´_w`w´_P_NUTS1[2..`nreg´,1]
.    matrix `cc´_w`w´_P_MARG       =`cc´_w`w´_P_SA \ `cc´_w`w´_P_N
. }
. else {
.    matrix      `cc´_w`w´_P_NUTS1 =`cc´_w`w´_P
.    matrix      `cc´_w`w´_P_MARG  =`cc´_w`w´_P_SA
. }
. matrix list `cc´_w`w´_P_NUTS1
. matrix list `cc´_w`w´_P_MARG
. *------------------------------------------------------------------------
```

3   Calibrated longitudinal individual weights

In this section we show how to reproduce the calibrated longitudinal individual weights (i.e. the variable cliw_b) for the balanced panel of respondents in a given country. Without loss of generality we focus on the German respondents in waves 1 and 2 by setting the following macros

```
. *------------------------------------------------------------------------
. * Select country (e.g. DE), initial wave (e.g. w1) and final wave (e.g. w2)
. *------------------------------------------------------------------------
. global wi 1                         // initial wave
```

```
. global wf 2                        // final wave
. global cc "DE"                     // country label
. global cc_num "12"                 // country number
. global pop_time 2004               // reference year
. global mort_time 2006              // final year
. global w "l_`wi´_`wf´"
. global age_groups 4                // number of age groups
. global age_thr_low "80 70 60 50"   // lower thresholds of age groups
. global age_thr_upp "89 79 69 59"   // upper thresholds of age groups
. *-------------------------------------------------------------------------
```

The remainder of our code can be easily adapted to the other countries and waves by changing the values of these macros.

Next, we run the do-file `CalMar` to define the vector of calibration margins for the population in the reference year that survives up to the final year. As discussed in the previous section, this is done by running the do-file `CalMar` which subtracts the number of deaths between initial and final year from the target population in the initial year.

```
. *-------------------------------------------------------------------------
. * Run CalMar.do
. *-------------------------------------------------------------------------
. noi run CalMar.do
symmetric DE_wl_1_2_P[1,1]
         POP
TOT  28724982

DE_wl_1_2_P_SA[8,1]
            POP
  M-80+     720789
M-70-79   2444070
M-60-69   4879081
M-50-59   4891028
  F-80+   1978719
F-70-79   3575138
F-60-69   5298005
F-50-59   4938152

DE_wl_1_2_P_NUTS1[16,1]
         POP
DE1  3553863
DE2  4189658
DE3  1147983
DE4   921566
DE5   243895
DE6   587565
DE7  2112105
DE8   608756
DE9  2776002
DEA  6235713
DEB  1413099
DEC   390135
DED  1687864
DEE   964298
DEF  1011053
DEG   881427
```

13

```
DE_wl_1_2_P_MARG[23,1]
               POP
   M-80+    720789
 M-70-79   2444070
 M-60-69   4879081
 M-50-59   4891028
    F-80+   1978719
 F-70-79   3575138
 F-60-69   5298005
 F-50-59   4938152
      DE2   4189658
      DE3   1147983
      DE4    921566
      DE5    243895
      DE6    587565
      DE7   2112105
      DE8    608756
      DE9   2776002
      DEA   6235713
      DEB   1413099
      DEC    390135
      DED   1687864
      DEE    964298
      DEF   1011053
      DEG    881427
```

In the computation of calibrated longitudinal weights, we want to exclude the subset of calibration margins for regional NUTS1 area by setting the DE_wl_1_2_P_MARG vector equal to the vector DE_wl_1_2_P_SA. This can be easily implemented by the following code.

```
. *-------------------------------------------------------------------------
. * No NUTS1 for longitudinal weights
. *-------------------------------------------------------------------------
. matrix ${cc}_w${w}_P_MARG = ${cc}_w${w}_P_SA
. noi mat li ${cc}_w${w}_P_MARG
DE_wl_1_2_P_MARG[8,1]
               POP
   M-80+    720789
 M-70-79   2444070
 M-60-69   4879081
 M-50-59   4891028
    F-80+   1978719
 F-70-79   3575138
 F-60-69   5298005
 F-50-59   4938152
. *-------------------------------------------------------------------------
```

The dimensions of these vectors are stored in a set of local macros because they correspond to the number of calibration equations.

```
. *-------------------------------------------------------------------------
. * Number of calibration equations
. *-------------------------------------------------------------------------
. mata: st_matrix("C1",rows(st_matrix("${cc}_w${w}_P_SA")))
. local C1 = C1[1,1]
. mata: st_matrix("C",rows(st_matrix("${cc}_w${w}_P_MARG")))
. local C = C[1,1]
```

```
. local C2 = `C´ - `C1´
. local nag = `C1´ / 2
. assert `C1´==8
. assert `C2´==0
. *------------------------------------------------------------------------
```

Next we load the individual level longitudinal database and select the DE subsample:

```
. *------------------------------------------------------------------------
. * Get local macros
. *------------------------------------------------------------------------
. local wi          ${wi}                // initial wave
. local wf          ${wf}                // initial wave
. local cc          ${cc}                // country label
. local cc_num      ${cc_num}            // country number
. local w           ${w}
. *------------------------------------------------------------------------
. * Load my SHARE dataset and select the country-data
. *------------------------------------------------------------------------
. qui use mydata_long_ind, clear
. qui keep if country==`cc_num´
. *------------------------------------------------------------------------
```

Our set of calibration variables consists of age and gender. We need to ensure that calibrated weights are missing when the calibration variables contain one missing values due to item nonresponse (unless we impute these missing values), and for all respondents aged less than 50 years because these persons do not belong to the target population of interest. Based on these criteria, we find that calibrated weights will be missing for 30 observations.

```
. *------------------------------------------------------------------------
. * Calibration variables
. *------------------------------------------------------------------------
. sum age gender dw_w`wi´
    Variable │        Obs        Mean    Std. Dev.       Min        Max
    ─────────┼──────────────────────────────────────────────────────────
      age_w1 │      1,599     63.4459    9.098648         33         94
      gender │      1,599    1.543465    .4982631          1          2
       dw_w1 │      1,599    4888.512    1738.689   1987.101   9525.167
. *------------------------------------------------------------------------
. * Binary indicator for missing weights
. *------------------------------------------------------------------------
. qui gen nowi=(dw_w1==.|gender==.|age==.|age<50)
. noi tab nowi, mis
       nowi │      Freq.     Percent        Cum.
    ────────┼───────────────────────────────────
          0 │      1,569       98.12       98.12
          1 │         30        1.88      100.00
    ────────┼───────────────────────────────────
      Total │      1,599      100.00
. *------------------------------------------------------------------------
```

In the following code we generate the binary indicators `xi_1-xi_8` for the 8 gender-age groups. The list of these indicators is stored in the local macro `list_Cvar`. Although the local macro C2 is equal to zero, we keep the code as general as possible.

```
. *-------------------------------------------------------------------------
. * Binary indicators for calibration groups
. *-------------------------------------------------------------------------
. local t = 1
. forvalues ss=1(1)2 {
2.    forvalues aa=1(1)`nag´ {
3.       local lb = ${cc}_w${w}_P_AGE_THR[`aa´,1]
4.       local ub = ${cc}_w${w}_P_AGE_THR[`aa´,2]
5.       if `aa´==1 qui gen xi_`t´=(age_w`wi´>=`lb´)*(age_w`wi´!=.   )*(gender==`ss´) if nowi!=1
6.       else       qui gen xi_`t´=(age_w`wi´>=`lb´)*(age_w`wi´<=`ub´)*(gender==`ss´) if nowi!=1
7.       local t = `t´ + 1
8.    }
9. }
. forvalues i=1(1)`C2´ {
2.    local i2 = `C1´ + `i´
3.    qui gen xi_`i2´ = (region==`i´ & age_w`wi´>=50 & age_w`wi´!=. & gender!=.) if nowi!=1
4. }
. list mergeid gender age_w1 xi_1-xi_8 if _n<=5, noobs

      mergeid   gender   age_w1   xi_1   xi_2   xi_3   xi_4   xi_5   xi_6   xi_7   xi_8

   DE-000132-01   Female       51      0      0      0      0      0      0      0      1
   DE-001381-01   Female       51      0      0      0      0      0      0      0      1
   DE-002106-01     Male       74      0      1      0      0      0      0      0      0
   DE-002106-02   Female       65      0      0      0      0      0      0      1      0
   DE-002173-01     Male       68      0      0      1      0      0      0      0      0

. *-----------------------------------------------------------------------
```

At this stage we have everything we need for reproducing the SHARE calibrated longitudinal weights by using the `sreweight` command. Below we show the syntax of this command using a logit specification of the distance function with lower bound $l = 0.01$ and upper bound $u = 10$. We allow the maximum number of iteration to go up to 200 by the `niter` option. As original design weights (the option `sweight`), we use the design weights of the wave when the longitudinal sample was originally selected (wave 1).

```
. *-------------------------------------------------------------------------
. * Compute calibrated weights (distance function: DS - case 6)
. *-------------------------------------------------------------------------
. local list_CVar ""
. forvalues i=1(1)`C´ {
2.    local list_CVar `list_CVar´ xi_`i´
3. }
. sreweight `list_CVar´ if nowi!=1 & dw_w`wi´!=.,        ///
>        nweight(my_wgt) sweight(dw_w`wi´)              ///
>        total(${cc}_w${w}_P_MARG)                     ///
>        dfunction(ds) upbound(10) lowbound(.01)        ///
>        niter(200)
Note: missing values encountered. Rows with missing values are not included in the calibration procedure
Iteration 1
```

```
  (output omitted)
Iteration 171 - Converged
Survey and calibrated totals

      Variable │        Original            New
    ───────────┼───────────────────────────────
          xi_1 │          105068         720789
          xi_2 │          588938        2444070
          xi_3 │         1453059        4879081
          xi_4 │         1230270        4891028
          xi_5 │          459720        1978719
          xi_6 │          808439        3575138
          xi_7 │         1516538        5298005
          xi_8 │         1418732        4938152
    ───────────┴───────────────────────────────

Note: type-ds distance function used
Current bounds: upper=10 - lower=.01
. *-------------------------------------------------------------------------
```

In this example, convergence required 171 iterations. The new calibrated weights are stored in the variable `my_wgt` and the associated vector of Lagrange multipliers is available in the output vector `r(lm)`. Below, we compare our calibrated longitudinal weights `my_wgt` with both the original design weights `dw_w1` and the calibrated longitudinal weights `cliw_b` available in the release 6.0.0 of the SHARE data.

```
. *-------------------------------------------------------------------------
. * Comparison between calibrated and design weights
. *-------------------------------------------------------------------------
. gen my_wgt_f=(my_wgt==.)
. bysort hhid`wi´: gen double hh=(_n==1)
. table my_wgt_f, c(count hh  sum my_wgt) row format(%9.0f)

    ─────────────┬─────────────────────────
      my_wgt_f   │     N(hh)   sum(my_wgt)
    ─────────────┼─────────────────────────
             0   │     1,569      28724982
             1   │        30             0
             │
         Total   │     1,599      28724982
    ─────────────┴─────────────────────────

. noi compare my_wgt cliw_b
    ───────────── difference ─────────────
    count       minimum      average      maximum
    ──────────────────────────────────────────────────────────────────────
    my_wgt=cliw_b            1569
    ───────────
    jointly defined         1569            0            0            0
    jointly missing           30
    ───────────
    total                   1599
. *-------------------------------------------------------------------------
```

The calibrated weights `my_wgt` coincide exactly with the calibrated weights `cliw_b` available in the SHARE data. These weights contains 30 missing values and 1,569 non-missing observations. The sum over all sample units matches exactly the size of the target population.

4 Calibrated longitudinal household weights

In this section, we consider the calibrated longitudinal household weights `clhw_b`. The main difference with respect to the calibrated longitudinal individual weights is that each household member must receive an identical calibrated weight that depends on the household design weight and the vectors of calibration variables of all 50+ household members. The setting of the macros to select the country, the initial and the final wave, as well as the specification of calibration margins, defined at the individual level, is similar to the previous section.

```
. *-----------------------------------------------------------------------
. * Select country (e.g. DE), initial wave (e.g. w1) and final wave (e.g. w2)
. *-----------------------------------------------------------------------
. global wi 1                          // initial wave //
. global wf 2                          // final wave //
. global cc "DE"                       // country label //
. global cc_num "12"                   // country number //
. global pop_time 2004                 // reference year //
. global mort_time 2006                // final year //
. global w "l_`wi´_`wf´"
. global age_groups 4                  // number of age groups
. global age_thr_low "80 70 60 50"     // lower thresholds of age groups
. global age_thr_upp "89 79 69 59"     // upper thresholds of age groups
. *-----------------------------------------------------------------------
. * Run CalMar.do
. *-----------------------------------------------------------------------
. run CalMar.do
. *-----------------------------------------------------------------------
. * Number of calibration equations
. *-----------------------------------------------------------------------
. * No NUTS1 for longitudinal weights
. matrix ${cc}_w${w}_P_MARG = ${cc}_w${w}_P_SA
. noi mat li ${cc}_w${w}_P_MARG
  (output omitted)
. mata: st_matrix("C1",rows(st_matrix("${cc}_w${w}_P_SA")))
. local C1 = C1[1,1]
. mata: st_matrix("C",rows(st_matrix("${cc}_w${w}_P_MARG")))
. local C = C[1,1]
. local C2 = `C´ - `C1´
. local nag = `C1´ / 2
. assert `C1´==8
. assert `C2´==0
. *-----------------------------------------------------------------------
. * Load my SHARE dataset and select the country-data
. *-----------------------------------------------------------------------
. qui use mydata_long_hhs, clear
. qui keep if country==`cc_num´
. *-----------------------------------------------------------------------
```

In the household level data, the binary indicator for missing calibrated weights is equal to 1 if the design weight `dw_w1` is missing or there exist no household member aged 50 years or older at

the time of the wave 1 interview. We find that calibrated longitudinal household weights will be missing for 29 household.

```
. *---------------------------------------------------------------------------
. * Binary indicator for missing weights
. *---------------------------------------------------------------------------
. gen n_elig_w`wi´=0
. forvalues i=1(1)10 {
2.    replace n_elig_w`wi´ = n_elig_w`wi´  +(age_w`wi´_`i´>=50 & age_w`wi´_`i´!=.)
3. }
. gen nowh = (dw_w`wi´==.|n_elig_w`wi´==0)
. noi tab nowh, mis
      nowh |      Freq.     Percent        Cum.
-----------+-----------------------------------
         0 |      1,090       97.41       97.41
         1 |         29        2.59      100.00
-----------+-----------------------------------
     Total |      1,119      100.00
. *---------------------------------------------------------------------------
```

Then, we generate a set of variables (i.e. the variables xh_1-xh_8) counting the number of household members that belong to each gender-age class calibration group. The set of calibration variables is stored in the macro list_CVar. As before, we keep the code as general as possible, although in this example the local macro C2 is equal to zero because we do not use the NUTS1 regional areas as additional calibration margins.

```
. *---------------------------------------------------------------------------
. * Indicators for calibration groups
. *---------------------------------------------------------------------------
. local X="age_w`wi´_"
. local Y="gender_"
. local list_CVar ""
. local t = 1
. forvalues ss=1(1)2 {
2.  forvalues aa=1(1)`nag´ {
3.     gen xh_`t´ = 0
4.     local list_CVar `list_CVar´ xh_`t´
5.     local lb = ${cc}_w${w}_P_AGE_THR[`aa´,1]
6.     local ub = ${cc}_w${w}_P_AGE_THR[`aa´,2]
7.     forvalues i=1(1)10 {
8.       if `aa´==1 replace xh_`t´ = xh_`t´ + ((`X´`i´>=`lb´)*(`X´`i´!=.)*(`Y´`i´==`ss´))
9.       else        replace xh_`t´ = xh_`t´ + ((`X´`i´>=`lb´)*(`X´`i´<=`ub´)*(`Y´`i´==`ss´))
10.    }
11.    local t = `t´ + 1
12. }
13.}
. forvalues r=1(1)`C2´ {
.    gen xh_`t´ = 0
2.  local list_CVar `list_CVar´ xh_`t´
3.  forvalues i=1(1)10 {
4.     replace xh_`t´ = xh_`t´ + (region==`r´ & `Y´`i´!=. & `X´`i´>=50 & `X´`i´!=.) if region>0
5.  }
6.  local t = `t´ + 1
7.}
```

```
. sum xh*

    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+------------------------------------------------------------
        xh_1 |      1,119    .0330652    .1788869         0          1
        xh_2 |      1,119    .1599643    .3667368         0          1
        xh_3 |      1,119    .3458445    .4758555         0          1
        xh_4 |      1,119    .2689902    .4436331         0          1
        xh_5 |      1,119    .0866845     .281498         0          1
-------------+------------------------------------------------------------
        xh_6 |      1,119    .1483467    .3605978         0          2
        xh_7 |      1,119    .3547811    .4805259         0          2
        xh_8 |      1,119    .3252904    .4686927         0          1

. list hhid1      gender_1 age_w`wi´_1             ///
>                          gender_2 age_w`wi´_2             ///
>                          gender_3 age_w`wi´_3             ///
>                          gender_4 age_w`wi´_4             ///
>                          xh_1-xh_8                        ///
>                          if hhid1=="DE-100831-A", noobs
```

hhid1 DE-100831-A	gender_1 Female	age_w1_1 58	gender_2 Male	age_w1_2 53	gender_3 Female	age_w1_3 86	gender_4 Female

age_w1_4 76	xh_1 0	xh_2 0	xh_3 0	xh_4 1	xh_5 1	xh_6 1	xh_7 0	xh_8 1

```
. *-----------------------------------------------------------------------
```

Now, we are ready to compute the calibrated individual household weights available in the SHARE database using a logit specification of the distance function with lower bound $l = 0.5$ and upper bound $u = 15$. Up to 152 iterations are needed to achieve convergence. The resulting weights my_wgt_hh are exactly equal to the calibrated longitudinal household weights clhw_b available in the release 6.0.0 of the SHARE data.

```
. *-----------------------------------------------------------------------
. * Compute calibrated weights (distance function: DS - case 6)
. *-----------------------------------------------------------------------
.         sreweight `list_CVar´ if nowh!=1,              ///
>                 nweight(my_wgt_hh) sweight(dw_w`wi´)    ///
>                 total(${cc}_w${w}_P_MARG)               ///
>                 dfunction(ds) upbound(15) lowbound(.5)  ///
>                 niter(200)
Note: missing values encountered. Rows with missing values are not included in the calibration proce
> dure
Iteration 1
  (output omitted)
Iteration 152 - Converged

Survey and calibrated totals

    Variable |      Original           New
-------------+----------------------------------
        xh_1 |        136073        720789
        xh_2 |        715785       2444070
        xh_3 |       1690376       4879081
        xh_4 |       1458319       4891028
        xh_5 |        516020       1978719
        xh_6 |        861385       3575138
        xh_7 |       1787257       5298005
        xh_8 |       1658252       4938152
-------------+----------------------------------
```

```
Note: type-ds distance function used
Current bounds: upper=15 - lower=.5
. *-----------------------------------------------------------------------------
. * Comparison between calibrated and design weights
. *-----------------------------------------------------------------------------
. gen my_wgt_hh_f=(my_wgt_hh==.)
. gen double hh=1
. table my_wgt_hh_f, c(count hh sum my_wgt_hh) row format(%9.0f)
```

my_wgt_hh _f	N(hh)	sum(my_wgt~h)
0	1,090	16895749
1	29	0
Total	1,119	16895749

```
.
. noi compare my_wgt_hh clhw_b
```

count	difference minimum	average	maximum
my_wgt_hh=clhw_b	1090		
jointly defined	1090	0	0	0
jointly missing	29		
total	1119		

5 Conclusions

In the first part of this report, we have provided an overview of the Stata programs available to compute calibrated weights that account for problems of attrition in longitudinal surveys. The intuitive idea of the calibration approach is to adjust the original design weights so that weighted survey estimates match the known population totals (the so-called calibration margins) for a given set of control variables including number of deaths by year, NUTS1 regional area, gender and age. Given such information from Eurostat, calibrated weights can be computed easily through the sreweight command implemented by Pacifico (2014), which is fast and includes several options for controlling the key features of the underlying optimization problem. In this report, we have illustrated the use of this Stata command by providing a variety of examples in the context of the SHARE data. Our Stata do-files can be easily adapted to compute calibrated longitudinal weights for any combination of subsamples/waves. It thus addresses a typical problem for panel surveys, namely the difficulty to release all possible weights along with the actual data. Users can hence compute appropriate weights that are most suitable for their specific analyses. The same approach can also be extended to other panel surveys such as the Generations and Gender Programme (GGP).

Part 2: Guidelines for sampling documentation

6 Overview of the sampling process

Probability sampling is a central prerequisite for data representativeness and hence high data quality. Like the other surveys within SERISS too, SHARE is putting a great emphasis on accurate and sound sampling procedures in order to achieve the highest possible data quality and representativeness of the target population. Against this background, each wave of the SHARE study consists of the main data collection as well as two rounds of preparatory data collection (i.e. pretest and field rehearsal) roughly six and twelve months before that. These two test rounds are important to actually test the instrument as well as all necessary operational sequences. By this, problems can be detected early enough to react rapidly and to make necessary adjustments.

Producing a gross sample that is fielded in the respective data collection can be described as a four step procedure with some specific features in each case (see Figure 1). Firstly, in the development phase, the respective country team and field agency conducting data collection develop a specific sampling design in close cooperation with SHARE Central and the respective persons responsible for the coordination of sampling. This incorporates deciding on a sampling frame and specifying the concrete sampling procedure. Secondly, the sample must be drawn by the country team/field agency (or through a third party, e.g. an institution hosting a national register) and subsequently processed to produce a gross sample file. In a third step, SHARE Central checks that the gross sample conforms to SHARE standards (e.g. unique household/person identifiers over waves, regional codes according to the official Eurostat nomenclatura). In the last step, SHARE Central prepares the gross sample data for distribution through the Sample Control software (Sample CTRL), which are then combined with the respective addresses uploaded by the country teams.

7 Documentation of the sampling design

To guarantee consistent and uniform sampling quality that meets the high standards in SHARE, it is required to document the specific details in the so called Sample Design Form (SDF). This is a structured online survey tool provided by SHARE Central to gather all sample information in a harmonized way from all countries (a screenshot of the SDF can be seen in Figure 2). By this, the systematic evaluation of the different and often complex sampling designs in countries

with a baseline or refreshment sample can be facilitated.[1] Survey agencies/country teams are asked to describe in detail their sampling design and give information to the following key areas: target population, sampling frame and selection of survey unit, as well as design specifics, such as stratification, clustering and availability of auxiliary information to improve design weights. A document containing all questions in the survey tool is provided along with this report and can be adapted to other surveys in SERISS, too (see Annex 1).

The SDF serves several purposes: Firstly, it is necessary that SHARE Central understands and approves the design to ensure that the sample complies with comparable quality standards across all participating countries, while considering the specific situation in each country. Therefore, the SDF is a tool to communicate and discuss the development and implementation of a sampling design between the country team/survey agency and SHARE Central. Because this is a serious task and hence should take some time, it is strongly advised to start the sampling design process as early as possible, but at the same time not too early either, since that might lead to more incorrect or outdated addresses (for example if the sample is obtained one year in advance of fieldwork). The main challenge thus is to find a balance in this respect. This applies in particular for countries where the application process for ordering a probability sample is known to be difficult and tedious. Secondly, The SDF is used to inform the process of quality checking the gross sample file provided by the country teams/survey agencies, before releasing it for data collection via the Sample CTRL software. After fieldwork has ended, the SDF and the gross sample are the basis for understanding the specific sampling procedure in order to calculate design weights for data analysis. And lastly, it will also provide information for end user documentation on sample design elements necessary to conduct proper analyses (e.g. correcting standard errors for cluster design effects).

8 Preparation of the gross sample file

For conducting the actual fieldwork, a gross sample file is needed that includes all sampled individuals or households which can be assigned to interviewers via the Sample CTRL software. In order for this process to work, each country that draws a baseline/refreshment sample needs to deliver the gross sample file to SHARE Central at least four weeks before fieldwork starts (see Figure 1 for a timeline of the sampling process). The file format and necessary fields are defined in the Gross

---

[1] Differences between baseline and refreshment samples are negligible regarding the documentation of the sampling process, but not regarding the weighting strategy. First, these two types of sample are likely to have different inclusion probabilities and hence different design weights. Second, as discussed in Section 1, respondents belonging to the refreshment sample are excluded from the balanced sample of individuals and households used to construct the calibrated longitudinal weights.

Sample File Template (GSFT, see below for a detailed description) that is provided by SHARE Central and might differ depending on the specific sampling design. The Excel template is provided along with this report, too (see Annex 2). After receiving the gross sample file, SHARE Central verifies that it contains the necessary information for all sampled cases. Also some double checks of the sampling process will be conducted as far as that is possible based on GSFT and SDF. Due to data protection reasons addresses and last names must not be included in the gross sample file delivered to SHARE Central. The transfer process of all other information is done via a secure transfer server (https://share.mpg.de).

SHARE also requires using so called batches that are random subsamples of the entire gross sample considering the specific sample design and are introduced sequentially depending on progress of fieldwork. Only after a batch has been exhausted, the next one will be opened and fielded. The reasons for using batches are two-fold: First, it gives more control over fieldwork progress and hence stimulates good performance as batches that are not started can be excluded from the denominator of the response rate calculation.[2] Second, it preserves sample quality in case fieldwork results are better than expected and the target number of interviews is attained before the end of fieldwork. Without batches, countries are obliged to fully exhaust the complete gross sample, since stopping halfway would result in a nonrandom selection of respondents. Using batches, in contrast, allows to stop fieldwork before the complete sample has been exhausted, thus saving costs.

The batches must be random subsamples (without replacement) of the final list of units (individuals or households) in the gross sample. They should be stratified by cluster, i.e. the number of units from each cluster in a batch should be proportional to the cluster size. For example if the gross sample size is 1000 and five percent of the units are in cluster 1, each batch should also contain five percent of the cases from cluster 1. If, for example, six batches are assumed with the first one containing half the gross sample (500 units) and the other five batches ten percent each (100 units), then batch 1 should contain 25 units from cluster 1 (500×0.05) and each subsequent batch should contain five units (100×0.05). As a rule of thumb, SHARE advises to use at least 50 percent of the gross sample for the first batch. In addition, it is important that the country team leader has full control over the timing when a new batch is opened, after checking the number of successful interviews and the response rate achieved so far. On the one hand, new batches should only be activated after all households in the previous batch have been assigned a final code (i.e. interview,

---

[2] Response rate definitions in SHARE pertain to: The American Association for Public Opinion Research. 2016. Standard Definitions: Final Dispositions of Case Codes and Outcome Rates for Surveys. 9th edition. AAPOR. URL: http://www.aapor.org/AAPOR_Main/media/publications/Standard-Definitions20169theditionfinal.pdf.

refusal, or non-contact).[3] On the other hand, there must be enough remaining fieldwork time to fully exhaust all cases in the new batch. Therefore, the country team should closely communicate with SHARE Central to find the best solution for all involved partners. In case of an agreement, the country team/survey agency will receive a correction file that is executed on the Sample CTRL software and opens another (previously inactive) batch that than can be assigned to interviewers.

Figure 3 shows the Excel template that has to be filled by the country teams/survey agencies. Each row corresponds to one sampling unit, i.e. either a target person when personal information on age is available (e.g. from a personal register) or a household when no individual information is available and the sampled households have to be screened for age eligibility. For household samples only the yellow fields have to be filled. Additionally, white fields have to be filled for person samples. If the birth date is not available from the sampling frame the dotted fields "age" or "age_category" can be included instead. Grey fields are mandatory (if applicable) for main samples only; for pretest and field rehearsal samples these columns can be left empty.

The following list provides additional information about the various fields in the GSFT:

- `pidcom` and `hhidcom` - these are the person and household identifiers that will be used for all further internal data processing. It is therefore vital that these are unique and consistent. The first two characters of both identifiers are the country code of your country. For `pidcom` the last two digits are a person counter within the household (i.e. "01" for all new samples). For `hidcom` the last character should always be an "A" for all new samples (i.e. main data collection, pretest and field rehearsal). The six digit number in the middle has to be generated to be unique within a country across all past and current waves and sample types as well.

- `first_name`, `gender`, `month`, `year`, `age`, and `age_category` - these are individual characteristics of a sample from a person register. In order to identify the sampled individual when contacting the household at least `first_name`, `gender` and `year` have to be provided and are therefore mandatory for person samples. The column including the birth `year` may be substituted with `age` or `age_category` if the former is unavailable in the sampling frame.

- `replicate` and `REPLICATEACTIVE` - these indicators define the fieldwork batches that have to be used at least in the main data collection (see above for information on how to create

---

[3] SHARE standards require that at least six contact attempts are made in each household (from which two must be done on weekday daytime, two on weekday evening, and two at the weekend) before the final code "non-contact" can be assigned (see Annex 1 "Specifications" of SHARE model contract).

batches). The batches should be numbered starting with "1" not "0". `REPLICATEACTIVE` should be set to "1" for the batch that is active at the start of fieldwork.

- `linkage` - this indicates if a country will link the survey data with administrative records (e.g. pension insurance data) in the main data collection. If record linkage is done, the column should be filled with a "1" and "0" otherwise.

- `nuts3`, `lau1`, `lau2` - these regional classifications are only mandatory in the main sample. `lau1` can only be provided if this classification is available in the respective country.

- `psu`, `psu_stratum`, `psu_popsize`, `ssu`, `ssu_stratum`, `ssu_popsize`, `final_stratum` - these indicators describe the probability sampling design. Since probability sampling is only mandatory in the main sample, these fields only need to be filled for the main data collection, too. The fields for `psu` and `ssu` are indicators for the first and second sampling stage clusters respectively. Depending on the specific sampling design a certain country might only have one clustering stage (or even none). In that case the `ssu` specific fields would stay empty. If stratification is used at one or both cluster sampling stages, the stratum of the specific `psu` or `ssu` in `psu_stratum` and/or `ssu_stratum` has to be indicated. Furthermore, it is also needed to provide the size of the target population within each `psu` and/or `ssu` if this is relevant for a country's sampling design. If stratification is used in the final sampling stage, i.e. individual persons or households within clusters, the stratum in the `final_stratum` field needs to be indicated. Taken together, these fields should cover the necessary information to describe most sampling designs encountered in SHARE. If a sampling design cannot be described with these standard fields in the GSFT additional fields should be provided.

9 Conclusions

In the second part of this report we have described standards in the documentation of the sampling process and the resulting gross samples that can be seen as a blueprint for large-scale cross-national surveys. By requiring a complete description of both the chosen sampling frame and the associated sampling design from the country teams/survey agencies, which then has to be evaluated and approved by the SHARE Central coordination, a consistent and uniform sampling quality is guaranteed that meets the high standards in SHARE. Together with a harmonized gross sample file that includes all sampled individuals or households, it can be verified that all necessary information is available for understanding the specific sampling procedure in order to calculate appropriate design

weights for data analysis (see part 1, but also SERISS Deliverable D2.10). Taken together, this facilitates the systematic evaluation and documentation of the different and often complex sampling designs in countries with a baseline or refreshment sample.

References

Bergmann, M., De Luca, G., and Scherpenzeel, A. (2017). Sample design and weighting strategies in SHARE wave 6. In Malter F. and A. Börsch-Supan (ed.), *SHARE wave 6: Panel innovations and collecting dried blood spots* (pp. 77–93). Munich: MEA, Max Planck Institute for Social Law and Social Policy.

De Luca, G., Rossetti, C., and Malter, F. (2015). Sample design and weighting strategies in SHARE wave 5. *SHARE Wave 5: Innovations & Methodology* (pp. 75–84). Munich: MEA, Max Planck Institute for Social Law and Social Policy.

Klevmarken, A., Hesselius, P., and Swensson, B. (2005). The SHARE sampling procedures and calibrated designs weights. In Börsch-Supan A. and H. Jürges (ed.), *The Survey of Health, Aging, and Retirement in Europe Methodology* (pp. 28–69). Mannheim Research Institute for the Economics of Aging (MEA).

Lynn, P. (2009). Methods for longitudinal surveys. In P. Lynn (Ed.), *Methodology of longitudinal surveys* (pp. 1-19). Chichester: Wiley.

Lynn, P., De Luca, G., Ganninger, M., and H ader, S. (2013). Sample design in SHARE wave four. In Malter F. and A. Börsch-Supan A. (ed.), *SHARE wave 4 Innovations & Methodology* (pp. 74–123). Munich: MEA, MaxPlanckInstitute for Social Law and Social Policy.

Pacifico, D. (2014). sreweight: A Stata command to reweight survey data to external totals. *Stata Journal* 14: 4–21.

Rossetti, C. (2017). Database containing necessary information for computation of population margins. Deliverable 2.9 of the SERISS project funded under the European Union's Horizon 2020 research and innovation programme GA No: 654221.
Available at: www.seriss.eu/resources/deliverables.

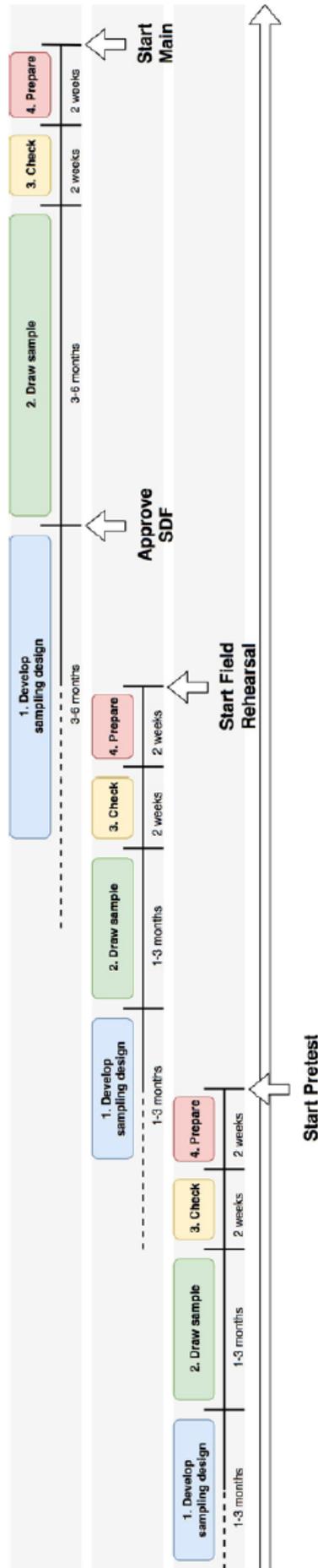Figure 1: Sampling process time line

Figure 2: Screenshot Sampling Design Form (SDF)

Figure 3: Screenshot Gross Sample File Template (GSFT)

pidcom	hhidcom	first_name	gender	month	year	age	age_category	replicate
person ID	household ID	first name (of drawn respondent)	gender (1=male, 2=female)	month of birth (unknown =999)	year of birth (unknown =99999)	age (if year is unavailable)	age category (if year/age are unavailable)	batch number
CC-485306-01	CC-485306-A	Willem	2	999	99999		50 to 54	1
CC-485774-01	CC-485774-A	Gerritjan	2	6	1959			1
CC-485836-01	CC-485836-A	Maria	1	5	1923			2
CC-487351-01	CC-487351-A	Sarah	2	2	99999	60		2
CC-494747-01	CC-494747-A	Philipp	1	5	1924			3

Annex 1

Sampling Design Form (SDF) questionnaire

Annex 2

Gross Sample File Template (GSTF)