CSB
WORKING PAPER

centreforsocialpolicy.eu March 2013 No 13 / 02

The EU-SILC sample design variables: critical review and recommendations

Tim Goedemé

University of Antwerp
Herman Deleeck Centre for Social Policy
Sint-Jacobstraat 2
B – 2000 Antwerp
fax +32(0)3 265 57 98

The EU-SILC sample design variables: critical review and recommendations*

Tim Goedemé

Working Paper No. 13 / 02

March 2013

ABSTRACT

The EU Statistics on Income and Living Conditions (EU-SILC) are the principal data source for analysing the social situation in Europe. Given that EU-SILC is based on a representative sample in each participating country, estimates based on EU-SILC are subject to sampling variance. One of the principal determinants of the sampling variance is the sample design that has been used for drawing the sample. Therefore, standard errors, significance tests and confidence intervals should be computed taking the sample design as much as possible into account. For doing so, good sample design variables are an indispensable starting point. In this paper, I review the quality of sample design information in the EU-SILC dataset and formulate recommendations for data producers about how to improve the quality of sample design variables and for data users about how to make optimal use of the information that is already available in the EU-SILC UDB.

Keywords: EU-SILC, sample design, sample design variables, sampling variance, standard error

Corresponding author:
Tim Goedemé
Herman Deleeck Centre for Social Policy (CSB)
Faculty of Political and Social Sciences
University of Antwerp and Net-SILC2
Tel: +32 3 265 55 55
Email: tim.goedeme@ua.ac.be

* This paper has been presented during the Workshop on standard error estimation and other related sampling issues in EU-SILC, organized in the context of the EU-funded "Net-SILC2" project, Eurostat, Luxembourg, 29-30 March 2012. Comments and suggestions by the participants of this workshop and by Guillaume Osier and Yves Berger are gratefully acknowledged. This work has received financial support from the second Network for the analysis of EU-SILC (Net-SILC2), funded by Eurostat. The European Commission bears no responsibility for the analyses and conclusions, which are solely those of the author.

1. Introduction

Random samples constitute a powerful tool to gain insight into the social and living conditions of millions of people while keeping costs relatively low. However, given that sample-based surveys contain information of only a limited number of households, the point estimates produced on the basis of the data are (among others) subject to sampling variance. Fortunately, statistical theory and methods offer well-grounded instruments to estimate the sampling variance of point estimates. In fact, without an indication of the sampling variance, a point estimate based on a sample is pointless. Standard errors, confidence intervals and p-values are the commonly used indicators of the random error of point estimates, caused by sampling variance.

In order to accurately estimate standard errors and confidence intervals of point estimates based on samples, it is necessary to take account of the sample design, weighting schemes, imputation and the characteristics of the indicator one is interested in (e.g. Eurostat, 2002; Heeringa et al., 2010; Wolter, 2007). If these elements are ignored, standard errors may be severely over-or (more frequently) under-estimated. The same is true for estimated design effects, the estimation of which is required to determine the minimum necessary nominal sample size. Among others, this results in the need for proper sample design variables to be included in the dataset. In this paper I evaluate the sample design variables included in the EU-SILC dataset which aim to identify primary sampling units (PSUs), primary strata, and the order of selection. The sample design variables are evaluated from two perspectives: from the perspective of data providers (national statistical institutes (NSIs), Eurostat) and data users (everyone who uses EU-SILC micro data for learning something about the social situation in the European Union or any of its regions). Given that for many countries there are still difficulties with the available sample design variables, relatively detailed recommendations on the construction of sample design variables are formulated. In the annex, a Stata do-file is included which provides detailed information on how EU-SILC users could best make use of the available sample design information in the EU-SILC User Database (UDB) when estimating standard errors and confidence intervals.

The recommendations formulated in this report build on earlier work of Verma et al. (2010) and Goedemé (2010, 2013). The discussion and examples are based on an analysis of the cross-sectional datasets (EU-SILC UDB 2005-2009 and EU-SILC 2008 data available to Eurostat). Possibly, other problems do exist for the longitudinal dataset. However, in the current paper, the analysis and recommendations are limited to the cross-sectional datasets.

The starting point is the assumption that data users may want to use EU-SILC for any kind of analysis of any variable included in the UDB. Therefore, the included sample design variables must be as precise as possible and cannot be constructed such that they may be roughly adequate for one kind analysis, and much more imprecise for another. In addition, it is assumed that most data users lack the time and resources to make a detailed study of the sample design of every EU-SILC country and to appropriately adapt the sample design variables as they are currently provided in EU-SILC. In other words, it is assumed that data users would benefit most from variables which do not need any further manipulation in order to take account of the sample design when estimating standard errors and confidence intervals. These variables could be provided in various forms: either in their original form, either in some form of 'computational strata and PSUs', or in the form of replicate weights, to be provided alongside the regular EU-SILC UDB. However, as a starting point, data providers of the national statistical institutes should deliver accurate sample design variables which reflect as closely as possible the real sample design.

This paper is structured as follows. In section two, I shortly discuss the ultimate cluster approach, from which it follows that good sample design variables related to the first stage of the sample design suffice in most cases for taking the sample design into account when estimating the sampling variance. Subsequently, recommendations are formulated for three different versions of the sample design variables. In section 3, I elaborate on the most important version of these, the 'original' sample design variables, which should reflect for each country as closely as possible the real sample design. In the following section, I discuss how from these 'original' sample design variables 'computational strata and PSUs' could be formed, which are constructed such that they can be used for estimating the sampling variance. Finally, in section 5, I discuss in what form sample design variables should and could be provided in the EU-SILC UDB. In addition, I highlight the main features of Stata do-files which have been produced to make optimal use of the available information in the UDB for constructing computational strata and PSU variables. An example of these do-files is included in the annex. Section 6 summarizes the main points of discussion and concludes.

2. The ultimate cluster method

As is explained by Osier (2012), the ultimate cluster approach simplifies substantially variance calculations (Kalton, 1979; Heeringa et al., 2010: 67-68; Wolter, 2007: 33). In addition, it considerably reduces the necessary sample design information in the dataset. With the ultimate cluster approach, it is assumed that the aggregate of selected ultimate sampling units of each PSU included in the sample forms an ultimate

cluster in the sampling process. As a result, the necessary sample design information in the dataset is limited to the first stage of the sample design: the only thing a researcher should know is to which primary strata and PSU each ultimate sampling unit in the sample belongs. Information on other stages of the sample design become irrelevant. Therefore, in this paper I focus on the four variables which should enable the identification of the primary strata and PSUs in the EU-SILC dataset. If more precise variance calculations are to be done, or in cases that the ultimate cluster method could result in biased variance estimates, the formulated guiding principles and recommendations should also be applied to the variables which identify the sample design at successive stages of the sample selection process.

3. The original sample design variables

In EU-SILC, the following sample design variables are available:

- DB050: primary strata (not included in the EU-SILC UDB)
- DB060: primary sampling units
- DB062: secondary sampling units
- DB070: order of selection of primary sampling units

In many cases, without further manipulation, these variables cannot be used for variance estimations. However, what is crucial, is that they should enable any user to construct sample design variables which can be used in variance calculations. Unfortunately, this is not always the case.

It is important to stress that other sample design variables may be relevant as well. More in particular, this is the case when weights have been calibrated. Since this could substantially reduce the sampling variance (especially if calibration variables are strongly correlated with EU-SILC variables of interest), it is highly recommended that the EU-SILC UDB would also contain the pre-calibration weight (cf. DB080) as well as the calibration variables. Having said that, I will limit the discussion in this paper to the four variables listed above.

In what follows, I will discuss one by one the main principles which should guide the construction of the original sample design variables (and which have been found to be violated for one or more countries). It is a matter of choice whether the original sample design variables or the 'computational sample design variables' are transmitted by NSIs to Eurostat. However, if correct 'original' sample design variables are transmitted to Eurostat, the burden for Eurostat to transform these into 'computational strata and PSUs' should be relatively limited and would result in a consistent coding of the sample design variables across countries.

3.1. The dataset should include sample design variables

In some cases the sample design variables are missing. First and foremost, the stratification variable DB050 is not included in the EU-SILC UDB. Second, for a number of countries also the PSU variable is missing. This is for instance the case of Belgium (for some years / releases) and Germany, and in some countries where dwellings have been selected at the first stage (e.g. Austria rotational panels before 2009) and for split-off households in Latvia (corrected as of the 2009 EU-SILC UDB, version 2). Furthermore, in the case of self-representing PSUs, information is needed about stratification at the second stage of the sample design (see below). Currently, such a variable is completely lacking. In addition, for many countries, sample design variables are missing for rotational panels selected in previous waves (a problem up to wave 2007).

It is highly recommended that the sample design variables are completed, also for earlier waves of EU-SILC.

3.2. All sample design variables should reflect the situation at the moment of selection

For variance estimations, information is needed about the sample selection process. As a result, sample design variables should include codes of (primary) sampling units and (primary) strata which refer to the moment of selection. Currently, this principle is not always respected. For instance, for some countries DB050 has to be used jointly with DB040 in order to reconstruct all primary strata (Spain, France, EU-SILC 2008). However, DB040 contains information on the region where a respondent lives at the moment of interview. Given the panel character of EU-SILC, households may move from one region to another between the moment of selection and the moment of interview. Among others, this results in PSUs being 'split' across various strata. In other cases, DB060 sometimes contains separate PSU codes for households which 'moved out' of the original region which coincided with the PSU (United Kingdom). However, also in these cases PSU codes should remain the same and reflect the moment of selection.

3.3. Each selected PSU should receive a unique identifier

If PSUs are sampled with replacement, multiple hits could occur. Even though considering these multiple hits as a single PSU should not bias

variance estimates, it is preferable that for every hit a PSU receives a separate identifier (potentially this is a problem for the Belgian EU-SILC data, see also below, same holds for Latvia). There are two reasons for this: only by doing so the correct number of degrees of freedom can be obtained, and it makes it easier for outsiders to relate the sample design information in the dataset to the sample design description in the national quality reports.

3.4. If a stratum contains a single PSU in the sample, the reason for this situation should be indicated in the dataset

If a stratum contains a single primary sampling unit, the within-stratum variance cannot be estimated. The remedy depends on how this single PSU came about. There are two possibilities.

First, it could be that a stratum contains a single PSU because within this stratum only one PSU has been selected among various PSUs in the stratum population, or because only one PSU among various selected PSUs contains respondents. In each of the latter two cases strata should be collapsed in order to estimate the sampling variance. For doing so, similar strata should be collapsed. Given the multi-purpose nature of EU-SILC, the criteria for defining which strata are most similar are not so easy to determine. However, it is crucial that for this purpose, no information from the sample itself is used, but rather from the sampling frame (or any other source). If available, average income seems a worthy candidate because of its correlation with many of the variables of interest included in EU-SILC. In these cases of single PSUs, it is a matter of discussion whether NSIs directly collapse strata or whether they include a flag indicating which strata should be collapsed.

Second, single PSUs could be self-representing PSUs. Self-representing PSUs (or certainty PSUs) are PSUs included in the sample with a probability of selection equal to 1. Several countries include certainty PSUs (e.g. France, Italy, the United Kingdom (Northern Ireland)). For variance estimations, certainty PSUs should be considered a stratum rather than a PSU. Special care should be paid in the case of systematic samples. If PSUs are selected with probabilities proportionate to size, and the interval which is used for systematic sampling is smaller than the size measure of some PSUs, these PSUs are certainty PSUs and should be treated the same as self-representing PSUs in non-systematic samples.

If self-representing PSUs are treated as regular PSUs instead of strata, this could result in a serious over-estimation of the sampling variance. Therefore, the original PSU variable should be accompanied by a flag variable which clearly indicates whether PSUs are self-representing or not. In addition, if this procedure is followed, in the case of self-representing

PSUs a variable is needed to identify strata at the subsequent stage of the sample selection (if applicable), as well as a variable to identify the secondary sampling units (i.e., the first stage at which the probability of selection is less than 1). Alternatively, self-representing PSUs and their substrata at the second stage of the sample selection scheme should immediately be coded as primary strata (in variable DB050), and the sampling units at the subsequent stage of the sample design as primary sampling units (in variable DB060). The procedure to be applied is a matter of discussion. The advantage of using a flag variable, is the closer correspondence of the sample design variables to the sample design description in the national quality reports. The advantage of the alternative procedure, is that the need for a variable identifying secondary sampling units and secondary strata is avoided.

In any case, it would be helpful if a detailed description of strata containing a single PSU would be included in the national quality reports.

3.5. Sample design variable codes should remain consistent across (rotational) panels and waves

This statement is self-evident in the case of the longitudinal dataset. However, it is also true for various cross-sections. If researchers have to estimate differences from one cross-section to another, they have to take the covariance between the various cross-sections into account. Two different sources of covariance exist in EU-SILC.

First and foremost, a covariance between various waves is likely to exist due to the panel character of EU-SILC: part of the respondents of both waves are the same. For most countries, such covariance is limited to waves within an interval of 4 years. However, in several countries panel rotation is spread across a longer time period (France, Norway), or a pure panel is implemented (Luxembourg). Given the ultimate cluster method, in countries with a multi stage sample design, it is not necessary to be able to merge data files on the household level. Instead, at least PSU codes should be consistent over time, such that covariance at this level across waves can be accounted for. However, in countries for which clustering is limited to the household level, for accurate variance estimation household IDs should remain consistent over time.

Second, in some countries, like Belgium and Spain, PSUs have been drawn for the entire duration of EU-SILC. As a result, even if households are rotated out after four waves, some covariance can exist even for waves separated by more than 4 years. Consequently, for countries with

this type of design, PSU codes should remained fixed for the entire duration of EU-SILC.

3.6. Systematic samples: order of selection

If PSUs are drawn using systematic selection, special attention must be paid to implicit stratification if the PSUs on the sampling frame are sorted on known variables. Given that direct variance estimation of a systematic sample is not possible, an approximation has to be used. If PSUs are drawn with equal probability of selection, Wolter (2007) suggests to use the order of selection as a starting point for picking up implicit stratification. In other words, a variable is needed which indicates the order of selection within each explicit stratum. In principle, this information is provided in variable DB070. However, currently it is not very clear whether this information is accurate for all countries and whether the variable refers for all countries to the first stage of the sample design.

If PSUs are drawn with unequal probabilities of selection, Wolter (2007: 335-353) is less conclusive as to which variance estimation formula should be preferred for general purposes. In this case, assuming a random sample with replacement of PSUs with unequal probabilities of selection may be preferable, especially if one is interested in estimating a confidence interval rather than the standard error. However, Verma et al. (2010) seem to suggest that also in this case the order of selection should be used for defining computational strata. Probably, further research is needed which explicitly performs simulations based on examples from EU-SILC. In any case, including the order of selection in the dataset allows data users to take implicit stratification into account when they judge this is most appropriate for their analysis. Please remember also that one should be careful with certainty PSUs in the case of systematic sampling with probabilities proportional to size (see above).

Independently of whether PSUs have been drawn with equal or unequal probabilities of selection, it would be helpful if the order of selection starts with 1 for each explicit stratum, or otherwise would contain clear 'breaks' between different explicit strata[1]. Doing so, would easily allow to form computational strata separately for each explicit stratum without running the risk that computational strata are formed which unintentionally contain PSUs of two different explicit strata.

[1] This is the preferred option for UDB users, as DB050 is lacking in the UDB.

3.7. Challenges of the rotational panel design

3.7.1.When the sample design is not changed across rotational panels

If the rotational design is implemented at a level below the PSU level, i.e. the PSUs included in the sample remain fixed for the entire duration of EU-SILC, no difficulties arise for estimating the sampling variance of the cross-sectional dataset, as the first stage of the sample is exactly the same for all rotational panels. Similarly, if PSUs rotate in and out the sample, and respondents are properly weighted for cross-sectional estimation, no difficulties should arise. In this case, every rotational panel could be interpreted as a new round of draws following the same sample design, the only difference being the long time interval. In both cases, PSU and primary strata codes should be consistent over time and across rotational panels.

If PSUs rotate in and out of the sample and are sampled with systematic selection and the sampling frame is not randomly ordered, the situation is more complex. Also in this case, the selection of various rotational panels could be interpreted as repeated sampling from the same population. However, two different approaches are possible, and, at least in my understanding, it is not very clear which option should be preferred.

The easiest solution would be to disregard the fact that PSUs have been drawn systematically and to assume that a simple random sample of PSUs has been drawn (as has been suggested in the Swedish national quality report). In this case, the sample is interpreted as a repeated sample from the same population, and if strata and PSU codes are consistently recorded across rotational panels, no difficulties should arise. As noted above, this is the more conservative and in some cases preferable approach, especially if PSUs are sampled with a probability of selection equal to their size (as is the case for most countries).

However, if there are efficiency gains due to the systematic sampling of PSUs from an ordered sampling frame, it may be useful to exploit these, especially with regard to the monitoring of the Europe 2020 poverty and social exclusion target (the more precise the measurement, the better the target could be monitored). To pick up any implicit stratification, it is necessary that computational strata can be formed while taking account of the order of selection of all PSUs of all rotational panels together. In other words, it may be the case that a computational stratum contains one PSU from sampling year n, and one PSU from sampling year n-1, n-2 or n-3. This is only possible if across years the sampling frame remains ordered in the same way, and if variable DB070 contains a consistent numbering of the order of selection taking all PSUs together. Table 1 illustrates the numbering that should be applied. If the order of PSUs on the sampling frame changes substantially from year to year (but the sorting variables

remain the same), the various rotational panels cannot be considered to be repeated samples from the same population and the question can be asked whether in that case an unbiased estimation of the sampling variance is still possible for the cross-section, while taking account of the order of selection.

Table 1. Hypothetical construction of DB070, for each rotational panel, PSUs with an interval of 5 are selected with a random starting point

PSU order on sampling frame	rotational panel / sample								DB070
	year n-3		year n-2		year n-1		year n		
	starting point	order of selection	starting point	order of selection	starting point	order of selection	starting point	order of selection	
1	x	1							1
2							x	1	2
3			x	1					3
4									
5					x	1			4
6		2							5
7								2	6
8				2					7
9									
10						2			8
11		3							9
12								3	10
13				3					11
14									
15						3			12
16		4							13
17								4	14
18				4					15
19									
20						4			16
…									

Source: Author's compilation.

3.7.2.When sample designs differ for some rotational panels

Some countries have changed their sample design over time. Examples include Austria, Hungary and Norway. Some introduced a multi stage design including stratification (Austria), others abandoned a multi stage design (Norway) and in Hungary the sample design changes for every rotational panel (Eurostat, 2011: 6). What should be done if the sample design of various rotational panels within the same cross-sectional dataset differ?

In principle all households included in the cross-sectional data files are part of the same population and could have been selected for each rotational panel (except for the relatively few cases including the newly born, persons who died and those who migrated in one of the years covered by the rotational panels included in the cross-sectional database). So, if the variance is estimated using a replication-based procedure, the

households should have a non-zero probability of being selected for each of the implemented sample selection schemes which all together resulted in the cross-sectional sample. In other words, for each of the rotational panel designs, sample design information should be made available for all respondents included in the cross-sectional database, independently of the rotational panel they belong to. For instance, if a rotational panel is selected in year n-2 following a simple random sample design and a rotational panel is selected in year n-1 as a stratified sample, appropriate strata codes should also be produced for the rotational panel initiated in year n-2, preferably in a separate variable. As a result, if the sample design changes, for every change, a new set of sample design variables is needed with information for all cases included in the database and a flag indicating which cases have actually been selected under the particular sample design. This method runs into difficulties if the definition of PSUs changes from one year to another. For instance, if one rotational panel uses a non-clustered design (with clustering above the household level) and another a clustered design, clusters cannot be re-composed for households selected under the non-clustered design (as other households belonging to the same PSU are not included in the original sample). It would be useful if countries like Austria and Hungary could clarify how they handle these difficulties when estimating the sampling variance.

3.7.3.Conclusion: keep complexity manageable

A point estimate without an estimate of precision is pointless. Hence, sample designs and sample design information in the dataset should be such that the sampling variance of point estimates can be estimated with a reasonable degree of approximation. In some cases, there are good reasons to introduce some degree of complexity in the sample design: multiple stages of selection may reduce interviewing costs and (implicit) stratification may increase the precision of estimates. However, when samples consist of multiple panels, complexity should not be unnecessarily inflated. Especially, changes in the first stage of the sample design should be avoided (that is, changes in the definition of primary strata and PSUs).

Therefore, the simplicity of the sample design should be an important aim in the future, in order to facilitate relatively accurate and efficient estimations of the sampling variance.

Examples of keeping the complexity of complex sample designs with manageable limits include two stage sample designs, with a large number of sufficiently small PSUs selected once and for all such that the sampled fraction of PSUs is relatively small, that it is easy to keep PSU codes consistent across time and rotational panels, and that data do not need to

be merged at the individual level in order to estimate the sampling variance of a difference between various cross-sections in a straightforward way.

4. Computational strata and PSUs

If the original sample design variables are recorded as described above, in some cases some further manipulations are necessary to estimate the sampling variance. In this section, I shortly discuss the manipulations which are necessary to compute sample design variables which may directly be used for variance estimation purposes, that is, computational strata and PSUs.

4.1. Define one PSU variable and one primary stratum variable

As a starting point, the PSU variable is equal to DB060. If, however, DB060 is not filled because households or persons have been selected at the first stage of the sample design, household ID numbers must be used. If DB060 refers to a self-representing PSU, for the households included in this PSU, the new PSU variable is equal to DB062 or household ID.

Similarly, for defining the primary stratum variable, one starts from DB050. If this variable is not filled, a unique country code is used instead. In the case of self-representing PSUs, unique strata codes are assigned to these PSUs and included as such in the primary stratum variable. At the same time, if not done so already by NSIs, primary strata containing one PSU are collapsed on the basis of information provided by NSIs.

It would be helpful if all PSU and stratum codes were unique across all countries and – as stressed above – consistent over time (i.e. across various waves of EU-SILC). Only by doing so, it is possible to estimate the sampling variance for aggregates of countries (e.g. the EU-27, NMS-10, EU-15) and for differences between two EU-SILC waves. In principle, it suffices that PSU codes are unique within each stratum, but by making them unique across all strata, it is easier to check the effect of stratification on estimated standard errors.

4.2. Systematic samples

In the case of systematic samples (at the first stage of the sample design), special care has to be paid to variance estimation. In their simplest form, they could be interpreted as random samples of PSUs. However, this would in some cases result in an overestimation of the

variance, due to the neglect of implicit stratification. More advanced estimators are available which aim to pick up the efficiency introduced by implicit stratification. The simplest method of these orders all PSUs in the same order as their original order of selection, and defines strata such that each stratum contains two PSUs which originally have been selected one after the other. This procedure should be applied for each explicit stratum. Please note that further research about the desirability of taking the order of selection into account is necessary, especially if PSUs are selected with unequal probabilities of selection (see above).

Once the computational strata and PSUs are constructed as described above, they are ready for variance estimation purposes. This is especially so if the sampling variance is estimated using a linearization based approach or the bootstrap. If the Jackknife Repeated Replication (JRR) is preferred, some further modifications may be advisable in order to reduce the computational burden. Verma et al. (2010: 30-40) discuss in detail which further modifications to the sample design variables can be implemented in order to facilitate variance estimation using the JRR approach.

4.3. A proposal

Annex 1 includes some proposed changes to the document "EU-SILC 065" which describes the EU-SILC target variables. The description is based on the document of the "2013 operation (Version September 2012)". I am convinced that by introducing relatively small changes and explaining somewhat better how the variables should be recorded, it will be much easier to derive correct computational primary strata and PSUs from the sample design variables included in the EU-SILC dataset. Errors will be avoided (both in the national statistical institutes and at the level of Eurostat), and especially the flag variables will contain more information such that the user of the data will know what to do for the estimation of the sampling variance. For all variables, it is stressed that codes should be unique across all EU-SILC survey years and rotational panels, such that it is possible to estimate the sampling variance of changes over time.

- The biggest change is implemented in variable DB050, which contains the values for identifying the primary strata. More in particular, I suggest to treat self-representing PSUs as if they were strata and to collapse strata immediately if they contain only one PSU as a result of the selection process. The flag variable is changed such that self-representing PSUs and collapsed strata are still identifiable in the data.
- The treatment of self-representing PSUs as if they were strata for DB050, implies that in DB060 secondary sampling units should be included as if they were PSUs, or that the variable is set to missing if these secondary sampling units are households. If PSU codes are

unique across rotational panels and time and remain consistent for various cross sections of EU-SILC, there is no need to know whether PSUs remain fixed over time or not. However, especially since that, currently, PSU codes are not consistent over time, it would be useful to include in the flag variable information on whether entire PSUs rotate in and out of the sample or whether rotation is implemented within PSUs instead of at the level of PSUs.

- In the case of DB070, the biggest change relates to the flag variable, which – in the case this proposal is implemented – would contain the necessary information about the selection process to decide on whether or not one would have to take account of the order of selection.

If these changes are implemented in the data, one would still need to integrate information of DB060 with household IDs in the case DB060 is missing (while taking care that DB060 and household ID codes are not overlapping). In addition, if it is opportune to take implicit stratification into account, DB050 should be combined with DB070 to compute the correct 'computational strata'. Finally, all computational strata would need to be made unique across the entire dataset if one wants to compute the sampling variance of statistics at the level of groups of countries. In other words, there remains a substantial role for Eurostat / the data user for generating sampling design variables that can be used in the process of variance estimation. Nonetheless, by implementing these changes data users will finally have sufficient information at their disposal for correctly doing so. Furthermore, the steps in the process that would benefit most from a single approach across all countries would be performed by Eurostat, while the steps that include very detailed information on the sample design and information that is only available on the sampling frame remain the responsibility of the national statistical institutes.

Apart from the changes in the recording of variables in the dataset, more information on the sample design is also needed in the national quality reports. This relates especially to the treatment of single PSUs (self-representing or not), but is also true in general. More in particular a clear overview of the definition and number of PSUs, SSUs, … as well as primary, secondary, … strata, would be very helpful, especially if it includes explicit information on how this information is recorded in the EU-SILC sample design variables.

5. Sample design variables for the UDB

In this section I will first discuss options for the future. In a second part I will shortly elaborate on how EU-SILC UDB users can make optimal use of the available sample design information in the UDB to generate computational strata and PSU variables.

5.1. Current problems and options for the future

Apart from the problems mentioned previously, currently there are three problems which EU-SILC UDB users have to face. First, the stratification variable is missing (i.e. DB050). Consequently, standard errors are likely to be (somewhat) over-estimated. However, the lack of DB050 in the UDB results for some countries in a second problem. That is, in some cases DB070 (the order of selection) and DB060 (PSUs) are not uniquely defined across strata (e.g. the United Kingdom respectively Slovenia and Poland). Especially in the case of DB060 this leads to problems, as PSUs with a similar code are collapsed across strata. Third, UDB users are not able to merge various waves of EU-SILC. Therefore, they cannot accurately estimate changes over time using the EU-SILC UDB. The main reason for this lack of information, are disclosure risks, i.e., the risk that households or persons can be identified in the dataset. There are three different satisfactory solutions for this problem.

In the ideal case, the EU-SILC UDB includes the computational strata and PSUs as defined above. This would enable UDB users to correctly estimate standard errors, taking as much as possible the sample design into account. Furthermore, it provides full flexibility as to the estimation procedure that is used for variance estimation. This is useful, as the suitable approach to variance estimation sometimes depends on the type of analysis and as it enables data users to estimate the contribution of various elements of the sample design to the total design effect. If the computational strata and PSU variables are included in their original form, standard estimation procedures of the common statistical packages could be applied (these are usually based on linearization). Moreover, given that the original computational strata and PSU variables are the starting point for variance estimation and must be computed in any case, direct provision of these variables means the smallest burden on NSIs and Eurostat.

Given the importance of sample design effects on estimated standard errors and the additional burden on Eurostat and/or NSIs to make alternatives available, any deviation from providing the original computational strata and PSUs in the EU-SILC UDB should be based on a scientific analysis of the real disclosure risk that would be associated with the provision of the complete original computational strata and PSU variables in the UDB.

Two valuable alternative strategies are available which could substantially reduce the disclosure risk: the provision of replicate weights and the application of random groupings to form larger strata and PSUs (Verma et al., 2010; Heeringa et al., 2010: 103-104). Each of these approaches has its strengths and weaknesses. Here, I will limit the discussion to the main arguments:

Strengths of providing replicate weights:

- Replicate weights could be developed with full sample design information available to NSIs, while taking account of calibration. In principle, these replicate weights could be constructed with limited approximations and simplifications.
- Replication based methods are flexible in the sense that they can be applied for any statistic, even in cases that no analytically derived variance formulae are available.
- Minimised disclosure risks, for the non-specialist much harder to identify strata and PSUs than with direct information on the basis of sample design variables. In addition, it is possible to introduce small 'errors' such that the identification of strata and PSUs becomes even harder.

Weaknesses of replicate weights:

- One variance estimation method must be chosen (in practice JRR or bootstrap), which should be adequate for a wide range of indicators.
- An extra dataset with replicate weights, to be provided alongside the UDB, is needed, which should include a sufficiently high number of replicate weights
- In order to take account of the covariance between cross-sectional datasets, different sets of replicate weights are needed for comparing results of various cross-sections (waves), rapidly increasing the burden on NSIs and/or Eurostat.
- The production of replicate weights assumes that sufficient knowledge and resources are available within NSIs and Eurostat to take this extra burden on board. If Eurostat is to construct the replicate weights (in order to relieve the NSIs and to apply a common procedure), Eurostat should have access to all the necessary information (e.g. also with regard to calibration).

Strengths of randomly collapsing strata and PSUs:

- Smaller disclosure risk compared to original computational strata and PSU variables
- Gives UDB users full control over the approach to variance estimation
- No extra dataset with replicate weights is necessary
- Relatively limited burden on NSIs and Eurostat

Weaknesses of randomly collapsing strata and PSUs:

- Sufficient knowledge and resources have to be available to NSIs and/or Eurostat. Collapsing strata and PSUs should be done with sufficient care in order not to introduce bias in variance estimation. The collapsed strata codes should be consistent across various cross sections.
- Somewhat larger disclosure risk than in the case of replicate weights (but not necessarily, also here small errors could be introduced to impede the identification of strata and PSUs in the UDB).
- More approximations are needed than in the case of replicate weights, separate variables need to be provided in order to be able to account for calibration.

5.2. Making optimal use of the available information in the UDB

Given the current state of affairs, Stata do-files have been written which make optimal use of the available information in the EU-SILC UDB to construct computational strata and PSUs which can directly be used for variance estimation purposes. An example for the EU-SILC 2009 UDB, version 2 is included in Annex 2. In what follows, I will shortly discuss the main problems this do-file tries to solve. Due to limited information on the quality of DB070, preference has been given to a more conservative approach. In other words, it has been assumed that systematic samples have been selected non-systematically.

The general principle is the following: whenever there is regional stratification, DB040 (i.e. NUTS 1) is used as a stratification variable. If DB060 is available, this variable is used as the PSU variable, otherwise household ID is used, while ensuring that in countries where strata contain both DB060 and household ID codes, every PSU number is unique, independently of its origin. For countries without regional stratification, or for which variable DB040 is missing, the country code is used instead, to ensure that variance estimates can be produced for any aggregation of countries in the dataset. Every stratum and every PSU receives a unique number for the entire dataset. The latter is not only useful for estimating the sampling variance for aggregates of countries,

but also for separately estimating the effect of stratification on the standard error.

In addition, several country-specific modifications to the sample design variables are implemented. For the last cross-sectional UDBs, DB060 is missing for Belgium. However, DB070 contains the order of selection for every PSU, which as a result can be used as a PSU variable instead. For the Czech Republic and Slovenia, PSU codes have to be made unique across rotational panels (DB075). The same applies to Latvia, if one wants to treat multiple hits on the same PSU as separate PSUs in the variance estimation process. For three countries, self-representing PSUs can be identified. In France, the 53 PSUs with information on secondary sampling units (DB062) and with the largest weighted number of households are assumed to be self-representing (i.e. they are considered strata and DB062 is used as PSU variable). In the case of Italy and the United Kingdom, DB060 codes which figure in at least three rotational panels are assumed to refer to self-representing PSUs (a sure strategy for the United Kingdom, but less so for Italy). In the latter two countries, household IDs are used as PSU codes.

Given that DB040 refers to the situation at the time of the interview, rather than the moment of selection, some PSUs are split across DB040 due to households that have moved from one region to another. For Belgium, the Czech Republic, Spain, France, Italy and Romania these households can be identified and be re-allocated with a reasonable degree of certainty to the correct stratum (i.e. the region inhabited by the majority of households of the PSU to which they belong). Unfortunately, in the case of Poland and Slovenia this strategy cannot be applied given that DB060 codes are not unique across strata, for which DB040 is a very rough proxy. As a result, in the UDB households belonging to different strata but with the same PSU code are treated as one PSU. If the main concern for this lack of information is disclosure risk, a sound procedure for aggregating PSUs should be used, such that UDB users can be sure their variance estimates are not biased.

Finally, if sample design variables are not completed for earlier waves of EU-SILC, it should be checked whether making use of the available sample design information (only for the newest rotational panel) would result in more accurate estimates than simply assuming that EU-SILC consists of a simple random sample of households.

6. Conclusion

Random samples are a powerful tool to learn something about millions of people on the basis of information for several thousands of households, involving relatively limited costs. Given that not the entire population is

included in the survey, estimates are confronted with – among others – sampling variance. Fortunately, this sampling variance can be estimated if good sample design variables are available and proper software is used.

Currently, the available sample design variables are in many cases not directly usable for variance estimation purposes. In some cases they are completely lacking (especially in the UDB), in others, self-representing PSUs are not identifiable. Furthermore, in some cases stratum and PSU codes are not consistent across various rotational panels. It is problematic that PSU and stratum codes are not consistent across various EU-SILC waves, such that the sampling variance of the difference between two EU-SILC waves cannot be estimated. This is problematic, as for many countries an accurate monitoring of the Europe 2020 poverty and social exclusion target is not possible with EU-SILC due to its relatively limited effective sample size. Any efficiency gain that could result from the covariance between several cross-sections would be useful in this respect. Annex 1 includes a concrete (and modest) proposal to improve the quality of the sample design information in the EU-SILC dataset.

In addition to good sample design variables, accurate information and documentation of the sample design (and changes over time) is necessary, not only for better understanding the sample design variables in the dataset and checking their quality, but also to enable researchers to understand dependencies in the data they have to take into account when estimating standard errors, such as in the case of fixed PSUs. Currently, some national quality reports are not publicly available on Circa, whereas others do not provide sufficient detail on the sample design. For instance, from the national quality reports it is not for all countries very clear whether PSUs have been selected for once and for all or whether they rotate in and out of the sample. In addition, it would be very useful if national quality reports would also discuss how the sample design variables have been encoded, what their relation is to the implemented sample design, and which number of strata and PSUs one should be able to identify in the dataset. Furthermore, additional information should be provided when strata contain only one PSU. If self-representing PSUs are included, a clear overview of the number of secondary strata and sampling units included in the dataset would be useful. Otherwise, data users can only guess to what extent sample design variables are accurate.

Finally, disclosure risks should be discussed on a scientific basis. If disclosure risks are too high, two strategies can be followed. One is to provide replicate weights. The other is to provide computational strata and PSUs in the UDB which are a correct aggregation of the original strata and PSUs. At the very least, countries should be encouraged to make DB060 codes and DB070 codes unique across strata, such that even without variable DB050, UDB users can correctly identify PSUs and the order of selection in the UDB. In order to facilitate the estimation of the sampling variance while taking account of the sample design, Stata do-files have

been produced to construct computational strata and PSU variables which make optimal use of the available sample design information in the UDB (an example is included in Annex 2). It would be useful to extend the range of do-files to other waves of EU-SILC and to do a similar exercise for the longitudinal datasets. In addition, it would be helpful if the do-files would be translated to other software packages. However, it should be stressed that this approach is not ideal. The direct provision of adequate computational strata and PSUs in the EU-SILC UDB is highly preferable.

7. References

Eurostat (2002), *Monographs of official statistics. Variance estimation methods in the European Union*, Luxembourg: Office for Official Publications of the European Communities, 63p.

Eurostat (2011), *2009 Comparative EU Intermediate Quality Report, version 3*, Luxembourg: European Commission, 93p.

Goedemé, T. (2010), *The construction and use of sample design variables in EU-SILC. A user's perspective*, Report prepared for Eurostat, Antwerp: Herman Deleeck Centre for Social Policy, University of Antwerp, 16p.

Goedemé, T. (2013), 'How much Confidence can we have in EU-SILC? Complex Sample Designs and the Standard Error of the Europe 2020 Poverty Indicators' in *Social Indicators Research,* 110(1): 89-110. doi: 10.1007/s11205-011-9918-2.

Heeringa, S. G., West, B. T., and Berglund, P. A. (2010), *Applied Survey Data Analysis*, Boca Raton: Chapman & Hall/CRC, 467p.

Kalton, G. (1979), 'Ultimate Cluster Sampling' in *Journal of the Royal Statistical Society. Series A (General),* 142(2): 210-222.

Osier, G. (2012), *The linearization approach implemented by Eurostat for the first wave of EU-SILC: what could be done from the second wave onwards?*, Paper presented during the Workshop on standard error estimation and other related sampling issues in EU-SILC, organized in the context of the EU-funded "Net-SILC2" project, Eurostat, Luxembourg, 29-30 March 2012.

Verma, V., Betti, G., and Gagliardi, F. (2010), *An assessment of survey errors in EU-SILC*, Eurostat Methodologies and Working Papers, Luxembourg: Eurostat, 70p.

Wolter, K. M. (2007), *Introduction to Variance Estimation*, New York: Springer, 447p.

8. Annex 1: Proposed changes to 'EU-SILC 065'

The text that follows is based on the EU-SILC 065 document of the 2013 operation (version of September 2012). The proposed changes are highlighted.

DB050: Primary strata
[Primary strata as used in the selection of the sample] *BASIC DATA (Basic household data including degree of urbanisation)* *Cross-sectional and longitudinal* *Reference period: at selection* *Unit: household* *Mode of collection: frame, register or sample design*

Values
1 - 99999

Flags	
1	Primary stratum
2	Self-representing PSU
3	Collapsed stratum due to single PSU (only for stratum with single PSU)
-2	not applicable (no stratification)

DB050 provides an identification code for the strata in case the target population (or a part thereof) is stratified at the first stage of the sample design. Stratifying a population means dividing it into non-overlapping subpopulations, called strata. Independent samples are then selected within each stratum. DB050 refers only to explicit strata, in the case of systematic sampling of PSUs, implicit stratification will be accounted for through the use of DB070.

In order to facilitate the computation of the standard errors for the common EU indicators, for the equivalised disposable income, for the unadjusted gender pay gap and for a list of income components, countries should[2] fill in this variable (in the case of stratification) for ALL panels and waves in the file, and not only the first one of the sub-sample (being the year of the selection of the concerned household). The recorded information, however, always refers to the situation at the time of the selection of the concerned household.

The above definition applies also to the new-entries from the second wave onwards.

[2] Agreement during the Living Conditions Working Group meeting in June 2009.

All primary strata receive a unique value which remains the same for the entire duration of EU-SILC (make sure the value is consistent for all EU-SILC waves).
The information in DB050 should enable the identification of ALL explicit primary strata, a combination with other variables (such as DB040) may not be necessary to identify all strata.
In the case of self-representing PSUs (that is, PSUs that are selected with a probability of 1), a separate, unique, value is assigned to DB050 for its identification and the flag variable receives code 2.
If strata consist of only 1 PSU selected among a larger number of PSUs in the population, or if it consists of only one PSU (among a larger number of PSUs) with respondents, primary strata have to be collapsed such that every stratum consists of at least two PSUs. For doing so, strata should be grouped with strata that are most similar in terms of the variables of interest for the analysis of EU-SILC. The decision of which strata are collapsed should be based on information that is available on the sampling frame. Preferably, strata similar in terms of average income are collapsed. If this information is not available, the following information is used, ordered from most preferred to least preferred: [average income, rate of employment, unemployment rate, degree of urbanisation, average age of the population].

DB060: PSU-1 (first stage)
DB062: PSU-2 (second stage)

[PSU-1 (first stage) as used in the selection of the sample]
[PSU-2 (second stage) as used in the selection of the sample]
BASIC DATA (Basic household data including degree of urbanisation)
Cross-sectional and longitudinal
Reference period: at selection
Unit: household
Mode of collection: frame, register or sample design

Values
1 - 9999 PSU (see below the required format)

Flags	
1	Fixed across time
2	Rotates in and out of the sample
-2	not applicable

If direct-element sampling is either impossible (lack of sampling frame) or its implementation too expensive (the population is widely distributed geographically), multi-stage selections can be done. Firstly, the population is divided into disjoint sub-populations, called **primary sampling units (PSUs)**. A sample of PSUs is then selected (first–stage sampling). Secondly, each sampled PSU is divided itself into disjoint sub-populations,

called **secondary sampling units (SSUs).** SSUs are then independently drawn from each PSU (second-stage sampling) and so on….
DB060 (DB062) provides identification codes for the selected PSUs (SSUs). Every selected PSU (SSU) should receive a value that is unique across all PSUs (SSUs) that have ever been selected in EU-SILC, and which remains the same for the entire duration of EU-SILC. In the case that the same PSU (SSU) is selected several times ('multiple hits'), the PSU (SSU) receives a unique value for every hit. The flag variable indicates whether PSUs rotate in and out of the sample, or whether they are fixed for the entire duration of EU-SILC.
In case there is at least a third stage of selection, additional variables DB06i (i≥3) shall be transmitted as identification numbers for the units sampled at stage i.(except for households, which are identified by the variable DB030, and for strata, identified by DB050). In the particular situation where more than one household can share the same dwelling, dwellings must be regarded as clusters of households and then coded accordingly, as the units that are selected at the ultimate stage. In order to facilitate the computation of the standard errors for the common EU indicators, for the equivalised disposable income, for the unadjusted gender pay gap and for a list of income components, countries should[3] fill in this (these) variable(s) (in the case of clustering) for ALL waves in the file, and not only the first one of the sub-sample (being the year of the selection of the concerned household). The recorded information, however, **always refers to the situation at the time of the selection** of the concerned household.
The above definition applies also to the new-entries from the second wave onwards.
In the case of self-representing PSUs, secondary sampling units should be treated as if they were primary sampling units and receive a unique code in variable DB060 (except if these are households, in which case DB060 is not applicable). The identification of the self-representing units themselves is implemented in variable DB050.

DB070: Order of selection of PSU

[Order of selection of PSU as used in the selection of the sample]
BASIC DATA (Basic household data including degree of urbanisation)
Cross-sectional and longitudinal
Reference period: at selection
Unit: household
Mode of collection: frame, register or sample design

[3] Agreement during the Living Conditions Working Group meeting in June 2009.

Values
1 - 9999 order of selection of PSU (see below the required format)
Flags
-2 not applicable Or a combination of two digits: First digit: fixed or changing order of selection 1 order on sampling frame is fixed for all EU-SILC survey years 2 order on sampling frame may change over time Second digit: probability of selection of PSUs 1 PSUs have an equal probability of selection (within explicit strata) 2 PSUs have an unequal probability of selection (within explicit strata) e.g. the order of PSUs on the sampling frame remains fixed for the entire duration of EU-SILC and PSUs are selected with a probability equal to their size: the flag is equal to 12

If primary sampling units (or households in case of direct-element sampling) are selected systematically, DB070 contains the rank of selection of those units. This information is important for variance estimation purposes as a systematic drawing from a judiciously ordered sampling frame may substantially decrease sampling errors. If systematic selections have been performed at other sampling stages, additional variables DB07(i-1), that is the order of the selection of the units of stage i (i>1), shall be transmitted too.

In order to facilitate the computation of the standard errors for the common EU indicators, for the equivalised disposable income, for the unadjusted gender pay gap and for a list of income components, countries should[4] fill in this (these) variable(s) (in the case of systematic selection) for ALL waves in the file, and not only the first one of the sub-sample (being the year of the selection of the concerned household). The recorded information, however, always refers to the situation at the time of the selection of the concerned household.

The above definition applies also to the new-entries from the second wave onwards.

[4] Agreement during the Living Conditions Working Group meeting in June 2009.

9. Annex 2: Stata do-file for constructing sample design variables for the cross-sectional EU-SILC 2009 UDB, version 2.

The do-file for this and other years will be made available on the internet after a final check (http://www.ua.ac.be/tim.goedeme). Please cite this paper and Goedemé (2011) when using these do-files. The do-files have to be run on the EU-SILC D-file, before it is merged to the other EU-SILC files.

```
<<code for loading the D-file>>

clear
set more off

foreach var of varlist _all {
	local newname = upper("`var'")
	cap rename `var' `newname'
}

*0. Preparation
***************

*generate country and hid variable

cap rename DB020 country
cap rename COUNTRY country

cap drop countryNR
encode country, gen(countryNR)

cap rename DB030 hid
cap rename HID hid

****************************************
* store all country labels in a global *
****************************************
 //please note that the following code is extracted from my vallab command, available at my homepage
 //a similar command, with somewhat different output, is provided by "levelsof"

local varlist country
sort `varlist'
tempvar tesje
qui: gen `tesje'=1 if `varlist'[_n]!=`varlist'[_n-1]
sort `tesje' `varlist'
qui: count if `tesje'==1
local nrvalues=r(N)
```

```
global countries
local counter=1
while `counter'<=`nrvalues' {
	local value1=`varlist'[`counter']
	local value2=`varlist'[`counter'-1]
	if "`value1'"!="`value2'" {
		global countries ${countries} `value1'
	}
	local counter=`counter'+1
}
global ncountries=wordcount("${countries}")
display "${countries}"
display "number of countries in datafile: " $ncountries


**************************************************
*Special cases that have to be handled before rest*
**************************************************
cap drop psutest
gen psutest=DB060

*1. Austria
**********

*In the case of AT, DB060 is partially missing, but this corresponds to sample design:
*AT: DB060 when available, otherwise hid
*DB060 is unique across strata, but currently no re-grouping is required, as it is the first wave a two-stage sample design has been implemented

*2. Belgium
**********
*DB060 is missing, but DB070 contains the order of selection of PSUs and can be used as a PSU variable instead.
replace psutest=DB070 if country=="BE"


*3. Czech Republic: DB060 not uniqe across panels
************************************************
replace psutest=DB060*10+DB075 if country=="CZ"

*4. France: self-representing PSUs
*********************************
*In France 53 PSUs are self-representing and for them DB062 should be filled
*In principle they refer to urban regions of more than 100,000 inhabitants.
```

*Also urban regions with between 20,000 and 100,000 inhabitants are sampled in several stages, with DB062 filled.

*As a result, the 53 self-representing PSUs should be the biggest ones (weighted number of households)

```
cap drop poppsu
cap drop groups

bysort country DB060: egen poppsu=sum(DB090)

replace poppsu=. if country!="FR" | (country=="FR" & DB062==.)
gsort -poppsu, gen(groups)

*tab DB060 if groups<=53
*Be careful: some DB062 have same code as some DB060!

replace psutest=DB062+0.1 if groups<=53 & country=="FR"

*5. Italy
*********

*Two stage sample design, rotation at PSU level. Large municipalities are self-representing and remain always in the sample.
*-> detect DB060 appearing in at least three out of four panels DB075, assume these are self-representing
*-> DB062 is filled, but if made unique by DB060, simply acts as a household identifier (as many hid as unique DB062)

cap drop tester
cap drop npanels

sort country DB060 DB075
gen tester=.
replace tester=1 if DB060[_n]==DB060[_n-1] & DB075[_n]!=DB075[_n-1]

bysort country DB060: egen npanels=sum(tester)

sort country DB060
ta npanels if country=="IT" & DB060[_n]!=DB060[_n-1]
ta npanels if country=="IT"
replace psutest=. if npanels>=2 & country=="IT"


cap drop tester
gen tester=DB060 if npanels>=2 & country=="IT"
cap drop groupsit
```

```
gsort tester, gen(groupsit)


*6. Latvia
**********


*1. make DB060 unique across DB075
*PSUs are drawn separately for each rotational panel, but PSU codes are not unique across DB075 in the case of multiple hits,
***so they should be made unique across DB075 (in principle not doing so should not bias variance estimates).

replace psutest=DB060*10+DB075 if country=="LV"



*2. Allocate split-off households randomly to PSUs of same rotational panel

*In the case of LV, DB060 is missing for 47 households. These are split-off households for which the orginal PSU is not given.
*Missing PSU codes could be randomly filled (alternatively, they could be dropped):
***If PSUcodes are randomly assigned, care is needed as PSUs are re-drawn for every panel. As a result, split-off households should be grouped with PSUs of the correct rotational panel.
***--> *since version 2 of EU-SILC 2009 this is no longer a problem, so can be ignored.


ta country if country=="LV" & DB060==.
local missinglv=r(r)

if `missinglv'!=0 {
	qui: tab DB075 if country=="LV", matrow(LVvals75)
	local nrows=rowsof(LVvals75)
	local vals75
	forvalues x=1/`nrows' {
		local value=el(LVvals75, `x', 1)
		local vals75 `vals75' `value'
	}
	local psuLV psuLV
	cap drop psuLV
	gen `psuLV'=.
	set seed 0001
	foreach panel of local vals75 {
		di "panel no. `panel'"
```

```
            cap mat drop mat075
            qui: tab psutest if country=="LV" & DB060!=. &
DB075==`panel', matrow(mat075)

            local uni060lv=r(r)

            di "No. of PSUs in panel: `uni060lv'"

            if `uni060lv'>1 {
                  replace     `psuLV'=1+int((`uni060lv')*runiform())     if
country=="LV" & DB060==. & DB075==`panel'
                  replace     psutest=el(mat075,     `psuLV',     1)     if
country=="LV" & DB060==. & DB075==`panel'
            }
      }
      sort country DB075 psutest
      list DB060 psutest DB075 if country=="LV" & DB060==.
}

*7. Slovenia
************
*Most probably, DB060 codes are not unique across DB075.

replace psutest=DB060*10+DB075 if country=="SI"

*8. United Kingdom
******************
*1. Northern Ireland is a self-representing PSU
*** The self-representing PSU (Northern Ireland) is recognisable as the
PSU with the largest number of households, the only PSU which appears in
the 4 rotational panels, the PSU with the largest number of households
and the only PSU with missing values for DB070.
*** self-representing PSU is itself a stratum & PSUs within this stratum
are households

cap drop cons
gen cons=1 if country=="UK"
cap drop nrpsu
bysort country DB060: egen nrpsu=total(cons==1) if country=="UK"

sum nrpsu if country=="UK"
local max=r(max)

ta npanels if country=="UK" & DB060[_n]!=DB060[_n-1]

ta nrpsu npanels if country=="UK"
```

```
sum DB060 if npanels==3 & country=="UK"
local test1=r(min)
sum DB060 if nrpsu==`max' & country=="UK"
local test2=r(min)
sum DB060 if DB070==. & country=="UK"
local test3=r(min)

if `test1'!=`test2' | `test1'!=`test3' {
	di in red "There is a problem with finding Northern Ireland, please mail this error to tim.goedeme@ua.ac.be"
	exit
}
else replace psutest=. if npanels==3 & country=="UK"

*2 if households move to another postcode sector, they form a new DB060 code.
***That is why the number of PSUs is higher than those reported
*-> unfortunately, there are too many PSUs (DB060) which contain only 1 household, otherwise they could be randomly merged with other PSUs...
*bysort country DB060: egen nhid=count(hid)
*sort country DB060
*ta nhid if country=="UK" & DB060[_n]!=DB060[_n-1]


**********************************
*Prepare Stratification variable*
**********************************

global stratcs AT BE BG CZ ES FR GR IT PL RO

cap drop region0
gen region0=""
foreach ctry of global stratcs {
	replace region0=DB040 if country=="`ctry'"
}
replace region0="ES80" if DB040=="ES63"|DB040=="ES64" //Melilla (ES64) and Ceuta (ES63) must be grouped together as they are part of the same stratum.

cap drop region1
encode region0, gen(region1)
replace region1=0 if region1==.
sum region1
local min=r(max)

replace region1=groups+`min' if country=="FR" & groups<=53
sum region1
local min=r(max)
```

```
replace region1=groupsit+`min' if country=="IT" & npanels>=2

sum region1
local minimum=r(max)
local maximum=10

while `maximum'<=`minimum' {
local maximum=`maximum'*10
}
cap drop strata0
gen strata0=countryNR*`maximum'+region1

sum strata0 if country=="UK"
local stratum=r(max)+2
replace strata0=`stratum' if country=="UK" & psutest==.

sum strata0


**********************
*Prepare PSU variable*
**********************

sum hid
local minimum1=r(max)
local maximum1=10
while `maximum1'<=`minimum1' {
	local maximum1=`maximum1'*10
}

sum psutest
local minimum2=r(max)
local maximum2=10

while `maximum2'<=`minimum2' {
local maximum2=`maximum2'*10
}

cap drop psu0
gen double psu0=.
replace psu0=strata0*`maximum2'+hid/`maximum1'
replace psu0=strata0*`maximum2'+psutest if psutest!=.


sum psu0


**********************
```

```
*RE-grouping of PSUs*
**********
*(AT), BE, CZ, ES, FR, HU(?), IT, RO: re-group split PSUs!

* In the case of several countries, stratification by DB040 causes PSUs to be split across regions because
*of households moving between moment of selection and moment of interview. Hence, households that have moved, should be re-allocated to the correct stratum

***please note that in the case of Poland, PSU codes are not unique across strata and therefore should split after stratification, re-grouping would do more harm than good
*** in other countries for which DB060!=., no households have moved between moment of selection and moment of interview.

set more off
global countrypsu BE CZ ES FR IT RO

cap drop checker
gen checker=.

sort country psutest hid

cap drop nocheck
gen nocheck=1 if psutest==. | (country=="FR" & groups<=53)

replace checker=0 if psu0[_n-1]!=psu0[_n] & psutest[_n-1]!=psutest[_n] | nocheck==1
replace checker=0 if psu0[_n-1]==psu0[_n] & psutest[_n-1]==psutest[_n] & nocheck!=1
replace checker=1 if psu0[_n-1]!=psu0[_n] & psutest[_n-1]==psutest[_n] & nocheck!=1
replace checker=2 if psu0[_n-1]==psu0[_n] & psutest[_n-1]!=psutest[_n] & nocheck!=1

*reset checker to 0 if PSUs must be split across strata

foreach ctry of global countries {
	di "`ctry'", _continue
	replace checker=0 if country=="`ctry'" & strpos("${countrypsu}", "`ctry'")==0
}
sort country psu0
foreach ctry of global countrypsu {
	tab country checker if country=="`ctry'" & psu0[_n]!=psu0[_n-1]
}
set more off
```

```
cap drop strata1
gen strata1=strata0

foreach ctry of global countrypsu {
	global psu`ctry'
	di "`ctry'"
	tab psutest if country=="`ctry'" & checker==1, matrow(psu`ctry')
// if you don't want the output, change to qui: tab etc.
	local rows=rowsof(psu`ctry')
	forvalues x=1/`rows' {
		local nr=el(psu`ctry', `x',1)
		global psu`ctry' ${psu`ctry'} `nr'
	}
	di "${psu`ctry'}"
}


foreach ctry of global countrypsu {
	di "`ctry'"

	foreach psu of global psu`ctry' {
		local check1
		local check2
		local check3


		tab psutest strata0 if country=="`ctry'" & psutest==`psu',
matcell(freq1) matcol(stratname) // if you don't want this output, change
to qui: tab
		local cols=r(c)
		forvalues y=1/`cols' {
			local check1=el(freq1, 1, `y')
			if `y'<`cols' {
				local check2 `check2' `check1',
			}
			if `y'==`cols' {
				local check2 `check2' `check1'
			}
		}
		local check3=max(`check2')

		forvalues y=1/`cols' {
			if el(freq1, 1, `y')==`check3' {
			replace	strata1=el(stratname,	1,	`y')	if
(country=="`ctry'" & psutest==`psu')
			di "`ctry' `psu': "el(stratname, 1, `y')
```

```
                continue, break
                }
        }

    }
}

qui: sum psutest
local minimum2=r(max)
local maximum2=10
while `maximum2'<=`minimum2' {
local maximum2=`maximum2'*10
}
cap drop psu1
gen double psu1=psu0
replace psu1=strata1*`maximum2'+psutest if psutest!=.

**************
*Finalisation*
**************
drop countryNR psutest poppsu groups npanels tester groupsit cons nrpsu region0 region1 strata0 psu0 checker nocheck


*1. Check sample designs on the basis of the re-constructed sample design variables

local vals 1
foreach x of local vals {
    svyset psu`x' [pw=DB090], strata(strata`x')

    cap mat drop svy`x'
    preserve
    foreach ctry of global countries {
        cap restore, preserve
        di "********************"
        di "`ctry'"
        di "********************"

        keep if country=="`ctry'"

        cap drop single`ctry'
        svydes if country=="`ctry'"
        local nsingle=r(N_single)
        local misstrat=r(N_mstrata)
        local mispsu=r(N_munits)
        local misobs=r(N_miss)
        local nstrats=r(N_strata)
```

```
            local npsu=r(N_units)
            local nobs=r(N)
            mat svy`x'=(nullmat(svy`x') \ `nsingle', `misstrat', `mispsu',
`misobs', `nstrats', `npsu', `nobs')
            cap drop single`ctry'
      }
      restore
      mat rownames svy`x'=${countries}
      mat colnames svy`x'=nsingle misstrat mispsu misobs nstrats npsu
nobs
      mat li svy`x'
}

*Example: population shares by degree of urbanisation in Belgium

svyset hid [pw=DB090]
svy: prop DB100 if country=="BE" // if condition instead of subpop option
is allowed as BE is a stratum
svyset hid [pw=DB090], strata(strata1)
svy: prop DB100 if country=="BE"
svyset psu1 [pw=DB090], strata(strata1)
svy: prop DB100 if country=="BE"

*2. Save D-file
rename country DB020
rename hid DB030

compress

<<code to save the D-file with the two new variables>>
```