9 User guide

9.1 Introduction

As we have seen in the previous chapters, the HFCS data are characterized by special features that must be taken into account when analyzing the data. The data are multiply imputed and contain survey weights and replicate weights. The HFCS data are also stored in several files, due to the structure of the survey. These files differ in terms of the data level (household or individual), the number of implicates (i.e. each implicate is a separate file) and the type of data, depending on whether the data were collected or constructed (derived variables, i.e. aggregated variables, and replicate weights vs. survey variables). This chapter[1] provides Stata®[2] code that users can employ step by step to account for all of these features.[3] Some extracts of the code were provided by Sébastien Pérez-Duarte[4] (ECB) and have been slightly altered and expanded for release here. The ECB is expected to also make several program codes available in fall 2016 in addition to publishing the dataset. In this chapter Stata® program code is contained in the blue boxes. It can be copied into the Stata® command window,[5] but must be run in the sequence outlined below (altering the sequence and/or including other lines may make the code corrupt). Additionally, the online appendix contains a do-file "`user_guide.do`" with all the steps that are laid out below.[6] This chapter first explains how to merge the separate files, then describes one way to set up the structure for imputations and survey information. Finally, some examples of simple estimation commands and how they are used are provided.

9.2 Merging the data files

The core HFCS data, which contain all internationally agreed variables, consist of the five multiply imputed samples or implicates at the household level (files H1–H5), the corresponding samples at the individual level (files P1–P5) and the corresponding set of aggregated variables[7] (files D1–D5). Before creating a new dataset containing all these files, users must specify the path to the datasets and the folder containing the do-files on their computers. The variables used for merging are the household identifier "sa0010," the implicate number "im0100" and the country identifier "sa0100."

[1] *The authors refrain from making a judgment about which programs to use and with which settings. In particular if the size of the subsamples varies in each iteration, the estimation of discontinuous estimators does not comply with the assumptions of the results evidenced in literature (e.g. Little and Rubin, 2002). It is the responsibility of users to check whether individual estimation commands are valid and adequate under particular conditions.*

[2] *The codes were written for Stata® version 13.1 and are not valid for previous Stata® versions.*

[3] *Any changes and improvements made to the code are continuously updated in the online appendix. Any adjustments made since the release of the first wave of the HFCS were included in this program code.*

[4] *Principal Economist Statistician in the Statistics Development/Coordination Division of the ECB.*

[5] *Due to the way Stata® handles line breaks, they may need to be deleted if the program code is copied by hand.*

[6] *The two macros containing the individual path to the data and the additional do-files must be specified before execution. Given the size and structure of the data and depending on software and hardware specifications, executing the do-file may require a long time.*

[7] *The ECB is expected to make the definitions of the aggregated variables and the datasets available in fall 2016.*

```
*********************************************************************
***Merging the files of the HFCS data
*********************************************************************

*Set macro for the path to the data (must be specified by the user)
global hfcsdata="path to the appropriate folder where the data are stored"

*Set macro for the path to the do-files (must be specified by the user)
global hfcsdofile="path to the appropriate folder where the do-files are stored"

*Set working directory
cd "$hfcsdata"

*Merging the p and h files together (wide format)
forvalues i=1(1)5 {
  use "$hfcsdata\P`i'.dta", clear
  drop id hid survey
  foreach var of varlist  sa0010- fra0500 {
    local `var'lab: variable label `var'
  }
  reshape wide ra0?0* fra0?0* ra0020 fra0020 ra0030 fra0030 ra0040 fra0040 p* fp* , ///
  i(sa0010 sa0100) j( ra0010)
  foreach j of varlist ra* fra* p* fp* {
    local last2car=substr("`j'", `=length("`j'")-1', 1)
    local last1car=substr("`j'", length("`j'"), 1)
    if "`last2car'"=="1" {
      local firstcar=substr("`j'",1, `=length("`j'")-2')
      rename `j' `firstcar'_`last2car'`last1car'
      label variable `firstcar'_`last2car'`last1car' ///
  "``firstcar'lab' - `last2car'`last1car'"
    }
    else {
      local firstcar=substr("`j'",1, `=length("`j'")-1')
      rename `j' `firstcar'_`last1car'
      label variable `firstcar'_`last1car' "``firstcar'lab'-`last1car'"
    }
}
  save "$hfcsdata\P`i'_temp.dta", replace
  clear
  use "$hfcsdata\H`i'.dta", clear
  merge 1:1 sa0010 sa0100 im0100 using "$hfcsdata\P`i'_temp.dta", nogen
  save "$hfcsdata\M`i'.dta", replace
  erase "$hfcsdata\P`i'_temp.dta"
}

*Merging the core with the derived variables
forvalues i=1(1)5 {
  use "$hfcsdata\M`i'.dta", clear
  merge 1:1 sa0010 im0100 sa0100 using "$hfcsdata\D`i'.dta"
  save "$hfcsdata\temp`i'.dta", replace
}
```

```
*Merging the implicates together[1]
use "$hfcsdata\temp1.dta", clear
forvalues j=2(1)5 {
  append using "$hfcsdata\temp`j'.dta"
}

*Drop unnecessary variables and labels
drop _merge
label drop _merge

*Save the HFCS data
save "$hfcsdata\hfcs.dta", replace
```

1 The temporary files are kept for configuring the multiply imputed data and are only erased following this procedure

So-called M-files are created by reshaping the P-files (using the `reshape` command), including the appropriate naming of the P-file variables, and by merging the resulting dataset with the H-files. They are provided in wide format[8] (i.e. one line of the data matrix contains information on a given household and the information on each individual within a household is included in a separate variable). Merged with the D-files, these M-files yield the entire HFCS dataset in the "hfcs.dta" file.

9.3 Multiple imputations

The next step is to import both the original data and the imputed values into Stata®'s `mi` (i.e. `mi estimate` commands for appropriate use of the multiple imputation structure). As the original data are not part of the HFCS data files, we have to construct them from the information about whether observations vary across implicates (indicating multiple imputation and, hence, missing values) and from the information about missing values taken from the flags.[9] Finally, original and imputed data must be imported and registered. Users should take note of the `"IMPUTEDVARS"` macro in the program code below, which contains a string listing all imputed variables once the corresponding loop has been executed. Moreover, the aggregated variables are registered as having been passively imputed. If registration was successful, running the `mi varying` command should yield only a few variables (e.g. the implicate number "im0100") and the flags as "unregistered varying."

[8] *It is also possible to merge the data files in "long" format using an almost identical code without needing to reshape the personal files.*

[9] *All missing values (including "Don't know", "No answer" and skip patterns) are set to "." and are paired with specific flags reflecting different types of missing values (e.g. skipped observations are flagged with a "0"). Flag variables have the same variable name, but their names are preceded by an "f."*

```
*********************************************************************
***Preparing the data for mi import
*********************************************************************

*Create the zero implicate to simulate the original data
*Use one implicate of the data
use "$hfcsdata\temp1.dta", clear
*Replace the implicate number by "0" to simulate the original data
replace im0100=0
*Append all other implicates
append using "$hfcsdata\hfcs.dta"

*For some reason string variables do not play well with mi commands and need to
be encoded into numeric variables
foreach var of varlist hb* hc* hd* hg* hh* hi* pa* pe* pf* pg* ra* sa0100 sb1000 {
  capture confirm numeric variable `var'
    if _rc {
    rename `var' `var'_string
    encode `var'_string, gen(`var')
    drop `var'_string
    }
}

*Set as soft missing (".") in im0100==0 all values varying, and also those whose
flags set them as imputed
global IMPUTEDVARS=""
foreach var of varlist hb* hc* hd* hg* hh* hi* pa* pe* pf* pg* ra* {
  capture confirm numeric variable `var'
  if !_rc {
    tempvar sd count
    quietly bysort sa0100 sa0010 : egen `sd'=sd(`var')
    quietly bysort sa0100 sa0010 : egen `count'=count(`var')
    quietly count if ( (`sd'>0 & `sd' <. ) | `count'<6 | (f`var'>4000 & f`var'<5000) ///
    ) & im0100==0
    if r(N)>0 global IMPUTEDVARS "$IMPUTEDVARS `var'"
    quietly replace `var'=. if ( (`sd'>0 & `sd' <. ) | `count'<6 | (f`var'>4000 & ///
    f`var'<5000) ) & im0100==0
    drop `sd' `count'
    disp ".", _continue
  }
}

*Here we need to set all derived variables for im0100==0 missing because it is
passively imputed
foreach var of varlist d* {
replace `var'=. if im0100==0
}

*Drop unnecessary variables
drop id _merge

*Save the HFCS data
save "$hfcsdata\hfcs.dta", replace

*Erase temporary files that will not be needed anymore
forvalues i=1(1)5 {
  erase "$hfcsdata\temp`i'.dta"
}
```

```
*********************************************************************
*****Import as multiply imputed data
*********************************************************************

*Import the imputation structure of the data into Stata
mi import flong, m(im0100) id(sa0100 sa0010) clear

*Register the variables that are imputed
mi register imputed $IMPUTEDVARS

*Register derived variables as passively imputed
mi register passive d*

*Check whether all imputed variables are registered
mi varying

*Save the HFCS-data with mi structure
save "$hfcsdata\hfcs.dta", replace
```

9.4 Survey variables

Having configured the data as multiply imputed, we can designate the data as complex survey data, identify variables that contain information about the survey design and specify the default method for variance estimation. In our case, all this information is contained in the final survey weights (hw0010) and in the 1,000 sets of replicate weights (wr0001–wr1000), which are provided in a separate file and hence have to be merged with the data first.

```
*********************************************************************
***Setting up Complex Survey Design
*********************************************************************

*Encode country indicator
use "$hfcsdata\W.dta", clear
rename sa0100 sa0100_string
encode sa0100_string, gen(sa0100)
drop sa0100_string
save "$hfcsdata\Wtemp.dta", replace
*Using the HFCS data with mi structure
use "$hfcsdata\hfcs.dta", clear

*Merging the data with replicate weights
merge m:1 sa0100 sa0010 using "$hfcsdata\Wtemp.dta"

*Drop unnecessary variable and files
drop _merge
erase "$hfcsdata\Wtemp.dta"

*Setting the appropriate survey structure using replicate weights
mi svyset [pw=hw0010], bsrweight(wr0001-wr1000) vce(bootstrap)

*Save the HFCS-data with mi svyset structure
save "$hfcsdata\hfcs.dta", replace
```

9.5 Standard estimation procedures

The data are now ready to be analyzed in Stata®. After writing `mi estimate: svy:` followed by the estimation command in question Stata® will provide correct estimates and standard errors, taking into account both the multiple imputation framework and the replicate weights.[10] The `esampvaryok` option can be useful when the sample size varies across implicates due to imputations.[11] Stata® versions below Stata® 12 do not allow the use of replicate weights together with multiply imputed data. For Stata® 12 or higher, the option `vceok` (used after the `mi estimate` command, e.g. "`mi estimate, vceok:…`") can be used as a workaround. It should be noted that in order to calculate the correct variance for subsamples of households (see second example in the following program code), Stata® requires a dummy variable for each of these subsamples combined with the use of the option for subpopulations (i.e. "`…svy, subpop(dummy)…`").[12] Alternatively, it is possible to use the option `over(variable)` for certain estimation commands (see last example in the following program code).

```
*********************************************************************
***Using Standard Estimation Procedures
*********************************************************************

*Using the HFCS-data with mi svyset structure
use "$hfcsdata\hfcs.dta", clear

*Mean of current value of primary housing unit
mi estimate, esampvaryok vceok: svy: mean hb0900

*Mean of current value of primary housing unit for part owner of the primary
housing unit
gen partowner=(hb0300==2)
mi estimate, esampvaryok vceok: svy, subpop(partowner): mean hb0900
```

[10] *A correct point estimate of statistics can be carried out on the basis of the final survey weights. Replicate weights are needed to calculate a variance estimator.*

[11] *Rubin's combination rules (see e.g. Little and Rubin, 2002) were derived on the assumption that the same set of observations is used in each imputed data set. Thus they may not necessarily apply when the sets of observations used in the data analysis differ. This is why* `mi estimate` *generates an error when this happens. When the subsets used in each complete data analysis differ relatively little, the conventional formulas may still be applicable. In this case, users can choose to use the* `esampvaryok` *option or find a better way to deal with the violation of the assumption of Rubin's combination rules described above. To our knowledge, this issue has not yet been addressed in literature.*

[12] *The use of an if-condition does not account for the uncertainty of the subsample size and therefore yields incorrect variance estimators.*

```
*Proportions of owner/renter of primary housing unit
mi estimate, esampvaryok vceok: svy: proportion hb0300

*Ratio of current to acquisition value of primary housing unit
mi estimate, esampvaryok vceok: svy: ratio hb0900 hb0800

*Regression of current value of primary housing on acquisition value and year of
acquisition
mi estimate, esampvaryok vceok: svy: regress hb0900 hb0800 hb0700

*Average level deposits according to gender of the first person
mi estimate, esampvaryok vceok: svy: mean  da2101, over(ra0200_1)
```

9.6 Additional estimation procedures

To calculate medians or other quantiles, we use a different Stata® package, called `medianize`, which was developed by the ECB (the respective do-file can be found in the online appendix). It must be used with caution since it is not yet a standard feature of Stata®; so far it has been tested only in limited environments. Other Stata® features used are the `tabstat` command and analytical weights.

```
**********************************************************************
***Including Additional Estimation Procedures
**********************************************************************

*ECB-written command to calculate medians (and some other quantile statistics),
which should be run before the estimation command
capture program drop medianize
do "$hfcsdofile\medianize.do"

*Median of amount still owned in the first loan collateralized with primary housing
unit
mi estimate, esampvaryok vceok: svy: medianize hb1701
*Median of amount still owned in the first loan collateralized with primary housing
unit over gender of first person
mi estimate, esampvaryok vceok: svy: medianize hb1701, over(ra0200_1)

*10th percentile of amount still owned in the first loan collateralized with primary
housing unit over gender of first person
mi estimate, esampvaryok vceok: svy: medianize hb1701, over(ra0200_1) stat(p10)
```

9.7 Online appendix

The online appendix contains the Stata® code described above and the do-files necessary to estimate certain quantiles. The Stata® code in the online appendix will be updated as required, to include program codes for other HFCS-relevant topics. Every additional do-file will be supported with the corresponding documentation.