DEUTSCHE
BUNDESBANK
EUROSYSTEM

Panel on household
finances

2. PHF Scientific Data User Workshop (Part II)

Version 2.0

Martin Eisele (Statistics Department)

Junyi Zhu (Research Center)

Nov 23rd 2013

Structure

- **Recapitulation: Rubin's rule for multiple imputation and sampling design**

- **Overview of steps in data work for a research project**

- **Working with PHF:**
  - ➢ Data preparation: long and median runs
  - ➢ Data management
  - ➢ Estimation

- **Further reference**

Rubin’s Rule for multiple imputation

- The overall estimate is the average of individual estimates from each implicate.

- The total variance is a kind of sum of within-imputation and between-imputation variance.

- If you forget the formula, here is a reference from Joseph L. Schafer: http://sites.stat.psu.edu/~jls/mifaq.html#howto.

Sampling design

- **Since PHF is not collected via simple random sampling, we need to account for the sampling design information to correctly obtain point estimate and its variance estimate. Particularly,**

  - Ignoring the clustering can underestimate the standard error.

  - Ignoring the stratification can overestimate the standard error.

Sampling design (con’t)

- Taylor series linearization (TSL) can approximate the variance estimate of any statistic which is a function of mean and total by the variance estimate for a linear expression of means and totals. We have the formulas as the design-based estimator for the latter under common complex survey designs.

- Replicate weights (bootstrap) are produced by adjusting the sampling weights to fit some estimation models. Since the estimation is derived from a data resampled from the original one, we can obtain many replicates of such adjusted weights. The variation across the estimates using these replicate weights is taken as the final variance estimate.

Sampling design (con't)

- **Here are the design variables required for each estimate in PHF:**
  - Point estimate: sampling weight (exhoch_hh)
  - Variance estimate via Taylor series linearization (TSL): sampling weight (exhoch_hh), strata (stratum) and primary sampling unit (p_nr).
  - Variance estimate via replicate weight: sampling weight (exhoch_hh) and replicate weight (wr*).

- **If you are not familiar to these concepts, here is a reference from Statistical Consulting Group, UCLA: http://www.ats.ucla.edu/stat/stata/seminars/applied_svy_stata11/default.htm.**

How does Stata handle them?

- Since PHF is a survey data multiply imputed with complex sampling design, it is suggested to resort to two packages in Stata:

  - Mulitple imputation – mi
  - Complex design – svy

- The rest of the slides and examplary do files will illustrate their application for PHF.

How does Stata handle them (con’t)?

Panel on household finances

- **Generally, you are recommended to update to version 11.2 if you have a Stata version 11.x or to version 12.1 if you have a Stata version 12.x.**

- **To perform mi estimate when accounting for design features, there is heterogeneity across different versions in Stata:**

How does Stata handle them (con't)?

Version	Solution 1: Load modified mi estimate program when considering replicate weights	Solution 2: use vceok option when considering replicate weights	Solution 3: TSL without considering replicate weights	Calculate the median and other quantiles*
11.2	Yes	No	Yes	Yes
12.0	No	No	Yes	No
12.1	No	Yes	Yes	Yes

***This can be achieved via an ECB written command compatible only with replicate weight (bootstrap) setting. By default, TSL setting in STATA does not support the calculation of any percentile. Since STATA 12.0 can only use TSL to take care of sampling design for mi estimate, there is no way for it to calculate these statistics when design-based variance estimate is required. But it is still possible if the only interest is to obtain the point estimate for these statistics (e.g. by mi estimate without svy setting).**

General advice on data preparation/management

Is it necessary to grasp the content between the slide 11 and 25 on data preparation and management?

- No. If you can do all of the cleaning, recoding, merging, variable generation and other data management work before estimation (this seems almost always possible), then simply skim over the following description on data preparation and management and go directly to slide 25.

Overview of steps in data work for a research project

Panel on household finances

1. **Retrieve all the potential variables in the long run you will use and run phf_dtprep.do to form a merged data for future use and save the list of identified imputed variables (which is required to be registered in stata mi environment). <- *phf_dtprep.do***

2. **Determine the list of working variables in the median run and run phf_working to set up the mi svy environment (you might only pick up a subset of the whole pool of identified imputed variables achieved in the step above). <- *phf_working.do***

3. **Use the mi svy set data created in step 2 for the short run. <- *phf_working.do***

Working with PHF: Data preparation (phf_dtprep.do) - Tasks

- Add the zero implicate which Stata needs simply as the original/unimputed data .

- Identify the imputed variables which will be saved so that they can be registered via mi package in the future.

- Merge the H, P and W data for the future use.

Working with PHF: Data preparation (phf_dtprep.do) – Parameters

Panel on household finances

- **It is advisable to maintain only the variables you might work with.**
  - Two reasons:
    1. It can take quite a long time to run the preparation if all the variables are kept.
    2. The merged data with everything can be memory demanding for some users.

- **Do not take only a subsample (e.g. only the males) which will disable Stata to do correct subpopulation analysis.**

- **For the replicate weights, you had better to keep at least two of them even if you do not care the variance estimate using them. This can save you some time if you do not want to bother with this aspect of setting in the code.**

Working with PHF: Data preparation (phf_dtprep.do) – Identify the imputed variables

Panel on household finances

- **Achieved by the large foreach block in phf_dtprep.do.**

- **Here are the interpretation for the three logical conditions used to determine whether the variable is imputed:**

  - `sd'>0 & `sd' <. - there are nonmissing varying values for some observation.

  - `count‘<6 - there are missing values for some observation (this can cover the cases when all the rest of nonmissing values are constant or all are missing, which are not picked up by first condition).

  - `var'fl>=2000 - there are values with imputation flag.

- **IMPORTANT STEP: copy the variable lists stored in the local macros IMPUTEDVARS_H and IMPUTEDVARS_P to two declaration statements for the global macros impv_h and impv_p in wkvar.do which will be used by phf_working.do (these lists are printed in the log file)!**

Working with PHF: Data preparation (phf_dtprep.do) – Data merging and long format

- We domonstrate a joint data in the long format with all the h variables attached to the row with pid=1 in the p data.

- Merge of h and p data is done by merge 1:1 because pid=1 in h data (see the result of tab _merge, missing).

- The next merge with w data is done by merge m:1 with dropping those _merge=2 because w data contains the households not appearing in the final sample.

Working with PHF: Data preparation (phf_dtprep.do) – Data merging and long format (con’t)

- **With the long format, you can perform both h and p levels of regression style analysis without reshaping efforts (examples will be shown later on):**

  - the analysis on h level: always insert a qualifier ‘if pid==1’; produce the regressors with the estimate over the household members (e.g. mean income) contained at least on the row with pid=1.

  - the analysis on p level: spread the h level variables to the members of each household whenever these variables will be used as regressor (after the merge, h variables are missing on the positions with pid>1).

Working with PHF: Working variables (wkvar.do)

- **Specify the analysis and ancillary variables which will be included by phf_working.do.**

- **The global macros impv_h and impv_p should not be changed in the median run, which include the whole pool of the identified imputed variables from h and p data.**

Working with PHF: Further data preparation (phf_working.do)

Panel on household finances

- **Determine if the analysis variables belong to the identified imputed variables.**
- **Mi set/import:**

```
mi import flong, m(impid) id(caseid pid) clear
```

m – implicate id; id – pid is also included so that persons can be uniquely identified

Working with PHF: Further data preparation (phf_working.do; con’t)

- **We focus on flong format. If you intend to have different format (e.g. wide, flongsep...), please consult stata manual in order to make the corresponding adjustments.**

- **When you use a mi command, do not always expect you can achieve the same result as the data with same structure but not mi set. Stata might apply some invisible "corrections" on the back whenever it is mi set.**

- **You are suggested to read these references (from Social Science Computing Cooperative, UW-Madison) on the particular cautions you should bear:** **http://www.ssc.wisc.edu/sscc/pubs/stata_mi_manage.htm and http://www.ssc.wisc.edu/sscc/pubs/stata_mi_estimate.htm.**

Working with PHF: Further data preparation (phf_working.do; con’t)

Panel on household finances

- **mi varying:**
  - Verify the variables are registered properly, esp. imputed variables.
  - Inspect those identified as Unregister xxx .
  - By default, those listed as Unregistered super/varying (variation occurs among the values when m>0 and this obs. has a missing value when m=0) are further registered.
- **For the details on this block of code as well as the conversion to wide format, please refer to Stata manual.**

Working with PHF: Further data preparation (phf_working.do; con’t)

Panel on household finances

- **mi svy set:**

  - ➢mi svyset [pw=exhoch_hh], bsrweight(wr0001-wr0100) vce(bootstrap)

    - ✓Set the appropriate design features and use bootstrapped replicate weights to calculate the variance estimate.

  - ➢mi svyset p_nr [pw=exhoch_hh], strata(stratum)

    - ✓Turn on TSL setting instead if there is the restriction from version 12.0; this can also be an alternative for variance estimate which can be rather fast.

Working with PHF: Data management (phf_working.do) - flong format and pattern of missing value

- **The subset with m=0 represents the original data before imputation. All the rest is exactly the long format data before mi import.**

- **The pattern of the observation with value missing when m=0:**

  - Usually the value is missing when m=0 and nonmissing when m>0 (often they vary), which is a typical pattern for an imputed observation.

  - Often the observation has values also all missing when m>0, which is the case when this variable is filtered due to the head variable. This is not imputed.

  - Sometimes the values are partially missing when m>0. This can happen when the head variable is also imputed and some of implicates are imputed to filter this branch variable (true missing). Here is an example:

```
sort impid
br impid caseid pid hb1010 dhb2400 if caseid==254
```

Working with PHF: Data management (phf_working.do) – variable generation

- **Usually the recommended mi xeq command can be almost always substituted by regular commands in flong format.**

- **mi passive is also recommended by stata manual to generate new variables. However, this prefix is more appropriate when the variable generated depends on only the current observation (reference: http://www.stata.com/statalist/archive/2010-11/msg01134.html).**

- **Do not alter the variables registered as imputed (e.g. using replace command) because STATA might apply some automatic corrections later on (e.g. it will use a nonmissing value in m=0 to update the others in m>0 when they are not same; however, this can be possible: e.g. , we intend to replace an imputed variable by a sum with another imputed one, and for one case, this variable is observed and thus nonmissing in m=0 and the other is unobserved and imputed. Instead, we should use mi passive to generate this sum).**

Working with PHF: Data management (phf_working.do) – variable generation (con’t)

- **An aggregate statistics is created over the members within each household:**

  ```
  by caseid impid, sort: egen dpg0200_a=mean(dpg0200)
  ```

- **H variable is spread over all the persons within each household:**

  ```
  by caseid impid, sort: egen hb0900_e=total(hb0900*(pid==1)), missing
  ```

Working with PHF: a quick approach to start estimation

- **In most cases, you do not have to fully follow what are presented so far. Instead, these are the steps required to set up before estimation (listed by the order you should implement):**

1. Add a zero implicate (contained in phf_dtprep.do)
2. Run the mi import command (described on slide 18; contained in phf_working.do)
3. Run the mi svyset command (described on slide 21 ; contained in phf_working.do)

- **However, these slides above address some generally applicable tips which can be useful even if you take now a quick approach: slides 13, 15, 16, 19, 22 and 24.**

Working with PHF: Estimation (phf_working.do) – General advices

- **Two important options for mi estimate:**

  - ➢By default, mi estimate will first verfiy if the same observations are used for the estimation in each imputation and issue the error message if not. However, this kind of variation can be often legitimate in PHF: some observations can be truely missing induced by the imputation of the header variable. Therefore, you are generally suggested to turn on the option esampvaryok.

  - ➢Turn on the option vceok to enable the calculation using the replicate weight if the version is 12.1.

Working with PHF: Estimation (phf_working.do) – General advices (con’t)

- **The general advices on variance estimate:**

  - when the command is supported by svy and mi estimate: run a fast estimate via TSL or replicate weight method with a few weights to obtain an approximate sense for statistical inference; and have a verification run in the end to include all the replicate weights when the model investigation is over.

  - when point estimate is the main concern: run without svy prefix but including a weight specification, e.g. mi estimate: mean ra0300 [pw=exhoch_hh].

Working with PHF: Estimation (phf_working.do) – Common estimates

Panel on household finances

- **Descriptive statistics**

  mi estimate, esampvaryok `opvce': svy: mean ra0300

  mi estimate, esampvaryok `opvce': svy: medianize ra0300 (incompatible with v 12.0)

  mi estimate, esampvaryok `opvce': svy: mean hb0900 if pid==1 (h level)

  ➢other descriptive statistics supported by mi estimate: proportion, ratio and total

- **All regression commands supported:** [**http://www.stata.com/help.cgi?mi_estimation#estimation_command**](http://www.stata.com/help.cgi?mi_estimation#estimation_command)

Working with PHF: Estimation (phf_working.do) – Common estimates (con’t)

- **how about ignoring mi or svy:**
  - common starting point for someone unfamiliar with mi and svy (what is the difference?)

```
mean hb0900 if pid==1 & impid>0 [pw=exhoch_hh]
```

  - or even take only one impicate

```
mean hb0900 if pid==1 & impid==1 [pw=exhoch_hh]
```

  - then let us ignore only svy

```
mi estimate: mean hb0900 if pid==1 [pw=exhoch_hh]
```

Working with PHF: Estimation (phf_working.do) – Percentiles

- **Percentile is not supported by mi estimate by default. To obtain the point estimates, we can code by ourselves:**

```
forvalues i=1(1)5 {
    capture drop hb0900_`i'pt
    pctile hb0900_`i'pt = hb0900 [pw=exhoch_hh] if impid==`i' & pid==1, n(100)
}
cap drop hb0900pt
gen hb0900pt = (hb0900_1pt+hb0900_2pt+hb0900_3pt+hb0900_4pt+hb0900_5pt)/5000
```

Working with PHF: Estimation (phf_working.do) – Subpopulation analysis

- **It is advised to use subpop and/or over option when subpopulation analysis is the interest (e.g. average age of FKP in the renter household).**

```
mi estimate, esampvaryok `opvce' : svy, subpop(renter): mean ra0300
mi estimate, esampvaryok  `opvce': svy: mean ra0300, over(renter)
```

- **Instead, using IF or IN qualifier to delete the ineligible cases can distort the variance estimate since it ignores the variance of the eligibility of each observation. This randomness is introduced by sample-to-sample uncertainty conditional on the sample design.**

Working with PHF: Estimation (phf_working.do) – Compatibility between mi estimate and other commands

Panel on household finances

- **How can mi estimate support any estimate command?**

  - Mi estimate searches a couple of saved results in e() before it can run. This link describes what they are: http://www.stata.com/help.cgi?program_properties#mi.

Working with PHF: Estimation (phf_working.do) – Compatibility between mi estimate and other commands (con’t)

Panel on household finances

- Generally, there are two cases to make the command survive with mi estimate: a. save/create the required results in e() if they are not produced by the command by default and/or we want to replace the default ones (e.g. e(b) and e(v)); or b. do nothing if the command produce them by default and we do not want to change them. These two approaches are described by under the subtitles “Combining point estimates and their variances”(case a) and “An example using suest”(case b) respectively through this link: http://www.stata.com/support/faqs/statistics/combine-results-with-multiply-imputed-data/.

- It seems some command provides a shortcut approach for the case a: there is a post option which can save a version of the results required (esp. e(b) and e(V)).

Working with PHF: Estimation (phf_working.do) – Compatibility between mi estimate and other commands (con't)

Panel on household finances

- How can any postestimation command support mi estimate?
- The trouble is they might search a couple of saved results in e() which is not produced by mi estimate by default.
- There are similar solutions for mi estimate, esp. the post option.

Working with PHF: Estimation (phf_working.do) – Compatibility between mi estimate and other commands (example)

- **margins is one command which is not supported by mi estimate by default…**

```
*program to calculate the average marginal effect
capture program drop mimargins
program mimargins, eclass properties(mi)
  svy bootstrap: logit renter c.hb0100 if pid==1
  margins , dydx(hb0100) post
end

* cmdok tells stata to support a command which is not mi estimatible
mi estimate, esampvaryok `opvce' cmdok post: mimargins 1

ereturn list

eststo out1
esttab out1
```

Working with PHF: Estimation (phf_working.do) – Compatibility between mi estimate and other commands (example; con’t)

Panel on household finances

- **Experiments:**

  - delete the post option for margins. The outcome is that the rest of program will only use the result from svy: logit since there is no e(b) and e(V) from margins.

  - delete post option in mi estimate then we will not see e(b) and e(V) in the returned results.

Working with PHF: Estimation (phf_working.do) – Compatibility between mi estimate and other commands (general approach)

- We might want to produce some mi estimator which can be beyond the scope of mi estimate (e.g. the estimate command does not save the results we want anywhere...). The second part of a "by hand" calculation of R square in this article from Statistical Consulting Group, UCLA, can be generalized: http://www.ats.ucla.edu/stat/stata/faq/mi_r_squared.htm .

Further references

- **If you are interested in flexibly generating statistics over household members, here is a link:**

  http://www.stata.com/support/faqs/data-management/creating-variables-recording-properties/

- **If you are a SAS user, this paper can be your starting point:**

  Berglund, Patricia, (2010). An Introduction to Multiple Imputation of Complex Sample Data Using SAS v9.2, SAS Global Forum 2010, Paper 265-2010. Paper can be downloaded from here.

DEUTSCHE BUNDESBANK
EUROSYSTEM

Thank you for your attention !

Panel on household finances