HRS | HEALTH AND RETIREMENT STUDY
A Longitudinal Study of Health, Retirement, and Aging
Sponsored by the National Institute on Aging

An Elementary Cookbook of Data Management using HRS Data with SPSS, SAS and Stata Examples

Documentation Report

Marita A. Servais

Survey Research Center
Institute for Social Research
University of Michigan
Ann Arbor, Michigan

June 2004

Funding

The Health and Retirement Study is funded by a grant from the National Institute on Aging (U01 AG009740) with supplemental support from the Social Security Administration. HRS is conducted by the University of Michigan.

Suggested Citation

Servais, M. A. (2004). An Elementary Cookbook of Data Management using HRS Data with SPSS, SAS and Stata Examples. University of Michigan.
https://hrs.isr.umich.edu/publications/biblio/5611

Table of Contents

Overview of Merging

Many analyses require variables that appear in separate files. Sometimes you will need to obtain variables from files at different levels that contain different numbers of records. Before you can do your analysis work, the files will need to be merged in an appropriate manner. Prior to doing any data management you should ask yourself several types of questions.

- What are the **variables** of interest? Identifying the variables needed for an analysis allows you to subset files to include only the necessary variables, weights, and identification variables. Smaller files are more manageable. If your analysis will use variables from more than one wave, identifying *comparable variables* in each wave and understanding any subtle differences in the variables is essential to successful analysis.
- What should be the **level** of the analysis file – will it have one record per respondent, one record per household or what?
- What **identification variables** will be required to merge the various files that contain variables needed for your analysis?
- What **type of merge** will be required? Will the merge be a one-to-one matching of records, e.g., respondent-to-respondent, or a one-to-many, e.g., household-to-respondent, matching? How do you want to handle cases that do not have a matching record in one or more of the input files?

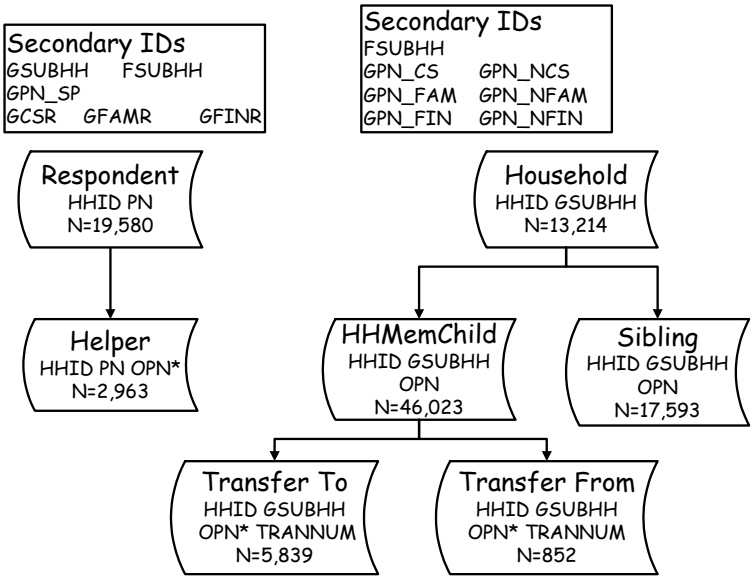
Primary and Secondary Identification Variables

In order to merge records from different files, you need to specify appropriate identification variables. Primary identifiers are the variables that uniquely identify a record in a file. Secondary identifiers are included to allow merging with files from other levels.

For example, the primary identification variables for respondent-level files from any wave are HHID and PN. The primary identification variables for household-level files are HHID and a wave-specific sub-household identifier, e.g., HHID and GSUBHH for 2000 household-level files. GSUBHH is a secondary identifier included on 2000 respondent-level files to allow merging with 2000 household-level files.

The primary identifiers and number of records (Ns) for the 2000 core data are illustrated graphically below[1]. Secondary identifiers for respondent based files (respondent and helper-level files) and for household based files (household, household member or child, sibling, transfer to and transfer from-level files) are also provided. The 1995, 1996, 1998 and 2002 core files have, for the most part, similar identification schemes. See the individual wave's Data Documentation document for specifics (from the HRS Web site http://hrsonline.isr.umich.edu/ → Documentation → Data Descriptions).

[1] The OPN designated with an asterisk has non-person values, e.g., "038" meaning "ALL CHILDREN EQUALLY". See codebooks for specifics.



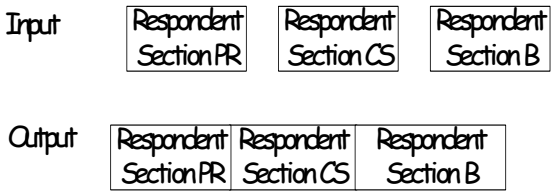
HHID and the wave-specific sub-household identifiers can be used to link cross-sectional household data with the cross-sectional respondent data. The wave-specific sub-household identifiers for core and exit files are listed below.

Sub-sample	Year	SUBHH - Core	SUBHH - Exit
HRS - Wave 1	1992	ASUBHH	--
AHEAD - Wave 1	1993	BSUBHH	--
HRS - Wave 2	1994	CSUBHH	--
AHEAD - Wave 2	1995	DSUBHH	NSUBHH
HRS - Wave 3	1996	ESUBHH	PSUBHH
HRS/AHEAD/CODA/WB	1998	FSUBHH	FSUBHH
HRS/AHEAD/CODA/WB	2000	GSUBHH	RSUBHH
HRS/AHEAD/CODA/WB	2002	HSUBHH	SSUBHH

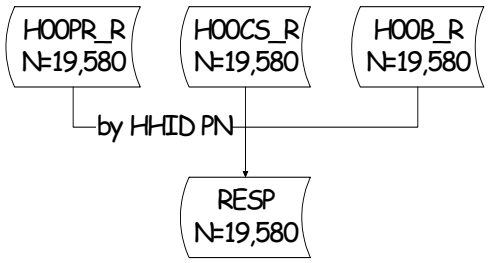
Cross-Sectional Merging

1. Merging Two or More Respondent-level Files

To create a respondent-level file with data from two or more respondent-level files, merge the respondent-level files using HHID and PN. This is a one-to-one match. Each input file has the same number of records and each record in each file will match a record in the other file(s).



For example, for 2000, each input file will contain 19,580 records. A respondent-level output file with 19,580 respondent records will result.



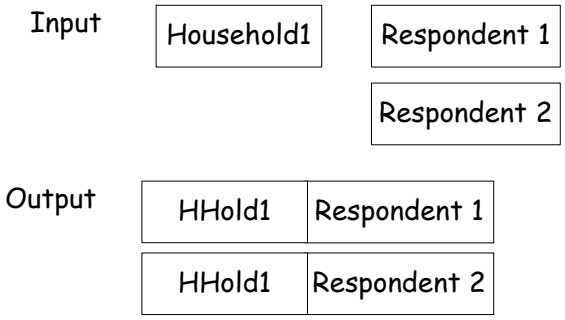
2. Merging Two or More Household-level Files

Similarly, to create a household-level file with data from two or more household-level files, merge the household-level files using HHID and nSUBHH where nSUBHH is the current-wave SUBHH. This is a one-to-one match. Each input file has the same number of records.

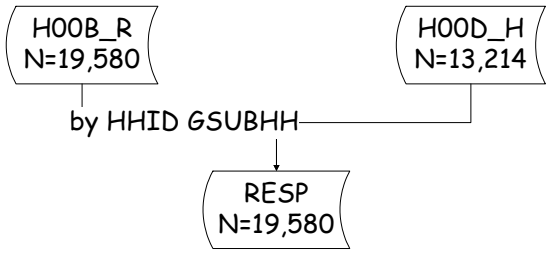
For 2000 use HHID and GSUBHH to merge 2000 household files with each other. Each input file will contain 13,214 records. A household-level output file with 13,214 household records will result.

3. Merging a Respondent-level File with a Household-level File: Creating a Respondent-level File with Data from a Household-level File

To create a respondent-level file including data from a household-level file, merge the respondent-level file(s) and the household-level file(s) using HHID and nSUBHH where nSUBHH is the current wave SUBHH. This is a one-to-many match (one household-to-many, up to two, respondents).



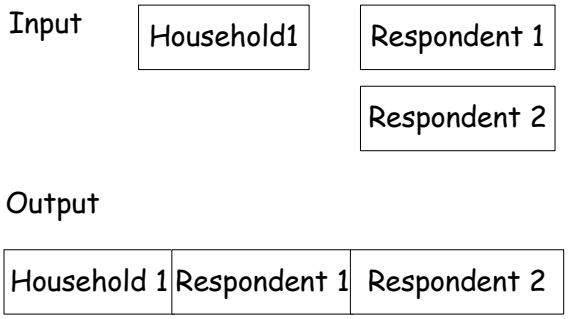
For 2000, use HHID and GUSBHH for merging. Household-level input files contain 13,214 records; respondent-level input files contain 19,580 records. A respondent-level output file with 19,580 respondent records will result.



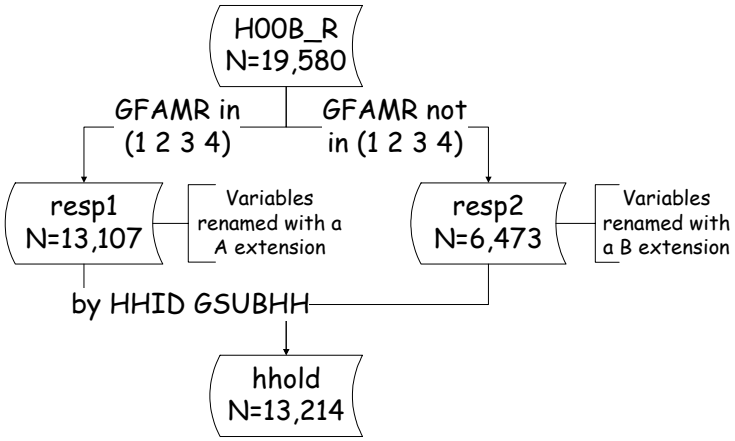
Since you are creating a respondent-level output file, be sure to keep PN, as well as HHID and GSUBHH, in the output file even though PN is not used for the merging.

4. Creating a Household-level File with Respondent Data from up to Two Respondents

To create a household-level file including variables from both respondents requires several steps.



First, separate the respondent-level records into two groups, one for the family respondent, the first person, and one for the non-family respondent, the second person.



We suggest you use GFAMR - 2000 WHETHER FAMILY RESPONDENT to create the two groups of respondents. The first group will contain 13,107 records (107 households did not have a family respondent); the second group will contain 6,473 records.

If you choose to use another variable to create two respondent groups, be sure each respondent group does not have more than one person from any one household (otherwise data will be lost).

Second, rename all variables for the second respondent group (except HHID and GSUBHH) to avoid overlap in the combined file.

Third, merge the two respondent files and the household-level file using HHID and GSUBHH.

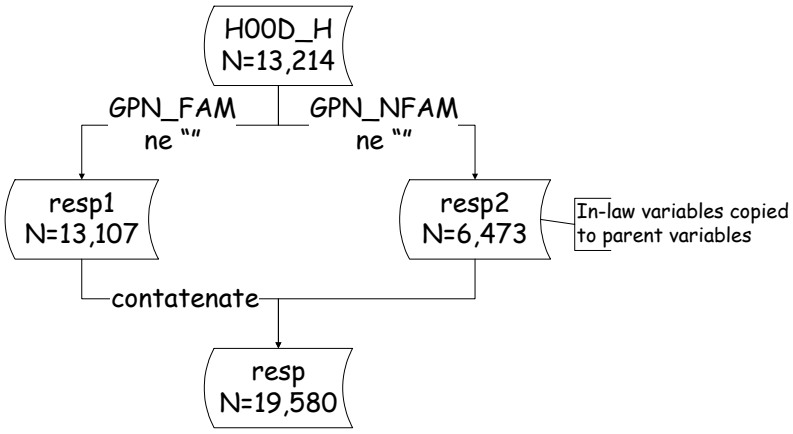
Although the respondent groups contain fewer than 13,214 records, this is a one-to-one match. A household-level file with 13,214 household records results. The household-level output file includes household variables, a set of variables for the first respondent and a set of variables for the second respondent.

We recommend that you keep PN for the first and second respondent even though PN is not used for the merging and the output file is a household-level file.

5. Creating a Respondent-level File with Parent Data from a Household-level File

Questions about parents of both the respondent and the respondent's spouse or partner were asked of the family respondent. This information is distributed at a household-level. You may wish to create a respondent-level file with information about the respondent's parents, whether the respondent was the family respondent or not. In order to do this, you will have to obtain the information about the parents from the household record and merge the variables to the proper respondent record.

In the example below, the file "famr" will have 13,107 observations one for each family respondent. The file "nfamr" will have 6,473 observations one for each non-family respondent. The combined file "resp" will have 19,580 observations one for each respondent.



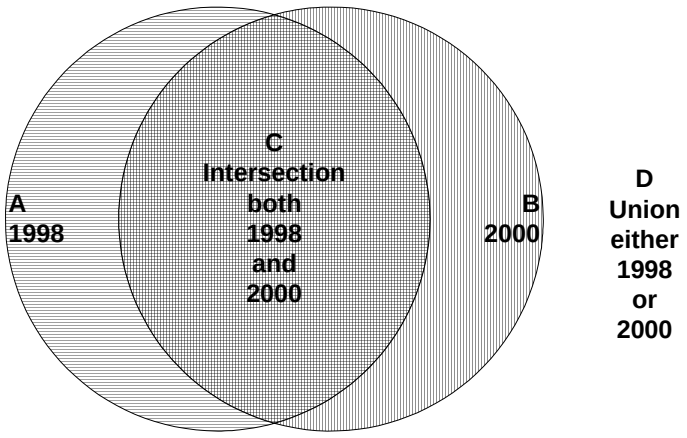
Longitudinal Merging

6. Merging 1998 Respondent-level Files with 2000 Respondent-level Files

Respondent records from all waves and from the tracker file may be linked by HHID and PN. When matching files on the same level, e.g., respondent records that do not each contain the same number of records, e.g., a respondent may have provided an interview at t-1 but not at t-2, you need to determine how you want to handle cases that do not have a matching record in one or more of the input files. For example when merging 1998 and 2000 core respondent-level files, you should decide whether you wish the output file to contain

A) records for respondents with records in the 1998 file, N=21384,
B) records for respondents with records in the 2000 file, N=19580,
C) the intersection, records only for respondents with records in both the 1998 and 2000 files, N=18858, or
D) the union, records for respondents with a record in either of the input files, N=22106.

The number of records in the output file will depend on which option you choose. The example code illustrates the intersection of the 1998 and 2000 files, option C, and includes comments to indicate how to specify the other options.



If the union of records, option D, were chosen, the combined file would have 22,106 records, 18,858 with information from both 1998 and 2000, 2,526 with information only from 1998, and 722 cases with information only from 2000.

Listing of contributions to combined file 1998 & 2000 (union)

in98	in00	Frequency	Cumulative Frequency
0	1	722	722
1	0	2526	3248
1	1	18858	22106

Example Code

Sample code in three languages, SPSS, SAS and Stata is provided below. We hope that this will be of help to you as you consider your own analytic needs. The examples below are consistent with the examples described and illustrated in a general way above. You will have to provide your directory names and will typically want to specify files and variables appropriate for your particular analysis.

SPSS Examples

1. SPSS Merging Two or More Respondent-level Files

```
/*-----------------------------------------------------------------------
/*      sort each input dataset by HHID PN
/*      subset variables

GET FILE 'c:\hrs2000\spss\h00a_r.sav'
        /keep=hhid pn g1051 g1053.
SORT CASES BY hhid pn.
save outfile='c:\temp\h00a_r.sav'.

GET FILE 'c:\hrs2000\spss\h00b_r.sav'
        /keep=hhid pn g1229 g1233.
SORT CASES BY hhid pn.
save outfile='c:\temp\h00b_r.sav'.

GET FILE 'c:\hrs2000\spss\h00c_r.sav'
        /keep=hhid pn g1654 g1655.
SORT CASES BY hhid pn.
save outfile='c:\temp\h00c_r.sav'.

/*-----------------------------------------------------------------------
/*      merge three datasets by HHID PN to create a respondent file
/*      this is a one-to-one merge

MATCH FILES
        /FILE='c:\temp\h00a_r.sav'
        /FILE='c:\temp\h00b_r.sav'
        /FILE='c:\temp\h00c_r.sav'
        /BY hhid pn.
EXECUTE.

save outfile='c:\hrs2000\spss\resp.sav'.
```

2. SPSS Merging Two or More Household-level Files

```
/*-----------------------------------------------------------------------
/*      sort each input dataset by HHID GSUBHH
/*      subset variables

GET FILE 'c:\hrs2000\spss\h00f_h.sav'
        /keep=hhid gsubhh g3060 g3061.
SORT CASES BY hhid gsubhh.
save outfile='c:\temp\h00f_h.sav'.
```

```
GET FILE 'c:\hrs2000\spss\h00j_h.sav'
        /keep=hhid gsubhh g5068.
SORT CASES BY hhid gsubhh.
save outfile='c:\temp\h00j_h.sav'.

/*-------------------------------------------------------------------
/*      merge two datasets by HHID GSUBHH to create a household file
/*      this is a one-to-one merge

MATCH FILES
        /FILE='c:\temp\h00f_h.sav'
        /FILE='c:\temp\h00j_h.sav'
        /BY hhid gsubhh.
EXECUTE.

save outfile='c:\hrs2000\spss\hhold.sav'.
```

3. SPSS Merging a Respondent-level File with a Household-level File: Creating a Respondent-level File with Data from a Household-level File

```
/*-------------------------------------------------------------------
/*      sort each input dataset by HHID GSUBHH
/*      subset variables

GET FILE 'c:\hrs2000\spss\h00a_r.sav'
        /keep=hhid gsubhh pn g1051 g1053.
SORT CASES BY hhid gsubhh.
save outfile='c:\temp\h00a_r.sav'.

GET FILE 'c:\hrs2000\spss\h00j_h.sav'
        /keep=hhid gsubhh g5068.
SORT CASES BY hhid gsubhh.
save outfile='c:\temp\h00j_h.sav'.

/*-------------------------------------------------------------------
/*      merge two datasets by HHID GSUBHH to create a respondent file
/*      this is a one-to-many merge

MATCH FILES
        /FILE='c:\temp\h00a_r.sav'
        /TABLE='c:\temp\h00j_h.sav'
        /BY hhid gsubhh.
EXECUTE.

save outfile='c:\hrs2000\spss\resp.sav'.
```

4. SPSS Creating a Household-level File with Respondent Data from up to Two Respondents

```
/*-----------------------------------------------------------------------
/*      file of family respondents

GET FILE 'c:\hrs2000\spss\h00a_r.sav'
        /keep=hhid gsubhh gfamr pn g1051 g1053.
USE ALL.
SELECT IF(gfamr  = 1 or gfamr = 2 or gfamr = 3 or gfamr=4).
sort cases by hhid gsubhh.
rename variables  gfamr=gfamra.
rename variables  pn=pna.
rename variables  g1051=g1051a.
rename variables  g1053=g1053a.
EXECUTE .

save outfile='c:\temp\h00a1.sav'.

/*-----------------------------------------------------------------------
/*      file of non-family respondents

GET FILE 'c:\hrs2000\spss\h00a_r.sav'
        /keep=hhid gsubhh gfamr pn g1051 g1053.
sort cases by hhid gsubhh.
USE ALL.
SELECT IF(gfamr  <> 1 and gfamr <> 2 and gfamr <> 3 and gfamr<>4).
rename variables  gfamr=gfamrb.
rename variables  pn=pnb.
rename variables  g1051=g1051b.
rename variables  g1053=g1053b.
EXECUTE .
save outfile='c:\temp\h00a2.sav'.

/*-----------------------------------------------------------------------
/*      variables from a household-level file

GET FILE 'c:\hrs2000\spss\h00j_h.sav'
        /keep=hhid gsubhh g5068.
SORT CASES BY hhid gsubhh.
save outfile='c:\temp\h00j.sav'.

/*-----------------------------------------------------------------------
/*      household-level file with information from two respondents

MATCH FILES
        /FILE='c:\temp\h00a1.sav'
```

```
        /FILE='c:\temp\h00a2.sav'
        /FILE="c:\temp\h00j.sav"
        /BY hhid gsubhh.
EXECUTE.

save outfile='c:\hrs2000\spss\hhold.sav'.
```

5. SPSS Creating a Respondent-level File with Parent Data from a Household-level File

```
/*-----------------------------------------------------------------------
/*      select parent variables from household file for family r

/*      keep id variables and parent variables of interest
GET FILE 'c:\hrs2000\spss\h00d_h.sav'
        /keep=hhid gpn_fam g2122 g2123.

/*      assign person number for family r
STRING pn (A3).
COMPUTE pn = gpn_fam .
VARIABLE LABELS pn 'Person Number' .

USE ALL.
/*      output records for households with family R
SELECT IF(pn <>").
EXECUTE.
save outfile='c:\temp\famr.sav' /drop= gpn_fam.

/*-----------------------------------------------------------------------
/*      select parent variables from household file for non-family r

/* keep id variables and parent variables of interest
GET FILE 'c:\hrs2000\spss\h00d_h.sav'
        /keep=hhid gpn_nfam g2309 g2310.

/*      assign person number for family r
STRING pn (A3).
COMPUTE pn = gpn_nfam .
VARIABLE LABELS pn 'Person Number' .

/*      copy in-law variables in to output variable location
rename variables g2309=g2122.
rename variables g2310=g2123.
USE ALL.

/*      output records for households with non-family R
SELECT IF(pn <>").
```

```
EXECUTE .

save outfile='c:\temp\nfamr.sav' /drop= gpn_nfam.

/*-----------------------------------------------------------------------
/*      concatenate files
ADD FILES
        /FILE='c:\temp\nfamr.sav'
        /FILE='c:\temp\famr.sav'.
EXECUTE.

save outfile='c:\hrs2000\spss\resp.sav'.
```

6. SPSS Merging 1998 and 2000 Respondent-level Files -- Intersection

```
/*-----------------------------------------------------------------------
/* subsetting keeping the variables of interest and ID variables
GET FILE "V:\LIBRARY\1998hrs\Final\core\built\spss\H98B_R.sav"
/keep=HHID PN FSUBHH F1109 F1116 F1129 F1146 F1156 F1156A F1176 F1194
F1189.
SORT CASES BY HHID PN.
SAVE OUTFILE='c:\temp\B98.sav'.

GET FILE "V:\LIBRARY\2000hrs\Final\core\built\spss\H00B_R.sav"
/keep=HHID PN GSUBHH G1238 G1245 G1262 G1279 G1289 G1309 G1322 G1327.
SORT CASES BY HHID PN.
SAVE OUTFILE='c:\temp\B00.sav'.

/*-----------------------------------------------------------------------
/* the merge
MATCH FILES
 /FILE ="c:\temp\B98.sav"
 /IN=a
 /FILE="c:\temp\B00.sav"
 /IN=b
 /BY HHID PN

/*      create merge file
/*      in SPSS a merge control variable is only created if you use the /IN= option
/*      There are two values for the merge control variable:
/*              0 - the record is not in common with the other data set
/*              1 - the record is common with the other data set

/*      the merge control variables are kept after the merge

/*      Example A in 1998;
```

```
/*      SELECT if (a=1).

/*      Example B in 2000;
/*      SELECT if (b=1).

/*      Example C in 1998 and 2000 -- intersection;
/*      SELECT if (a=1 and b=1).

/*      Example D in 1998 or 2000 -- union;
/*      default for SPSS - no merge control criteria or;
/*      SELECT if (a=1 or b=1).

SELECT if (a=1 and b=1).
SAVE OUTFILE='c:\temp\B0098.sav' /drop=a b.
```

SAS Examples

1. SAS Merging Two or More Respondent-level Files

```
*-------------------------------------------------------------------------;
*       sort each input dataset by HHID PN
        subset variables;

proc sort
        data=in.h00a_r
        out=h00a_r(keep=hhid pn g1051 g1053);
        by hhid pn;
run;

proc sort
        data=in.h00b_r
        out=h00b_r(keep=hhid pn g1229 g1233);
        by hhid pn;
run;

proc sort
        data=in.h00c_r
        out=h00c_r(keep=hhid pn g1654 g1655);
        by hhid pn;
run;

*-------------------------------------------------------------------------;
*       merge three datasets by HHID PN to create a respondent file
        this is a one-to-one merge;
data resp;
        merge h00a_r h00b_r h00c_r;
        by hhid pn;
run;
```

2. SAS Merging Two or More Household-level Files

```
*-------------------------------------------------------------------------;
*       sort each input dataset by HHID GSUBHH
        subset variables;
proc sort
        data=in.h00f_h
        out=h00f_h(keep=hhid gsubhh g3060 g3061);
        by hhid gsubhh;
run;

proc sort
data=in.h00j_h
```

```
        out=h00j_h(keep=hhid gsubhh g5068);
        by hhid gsubhh;
run;

*-------------------------------------------------------------------------;
*       merge two datasets by HHID GSUBHH to create a household file
        this is a one-to-one merge;
data hhold;
        merge h00f_h h00j_h;
        by hhid gsubhh;
run;
```

3. SAS Merging a Respondent-level File with a Household-level File: Creating a Respondent-level File with Data from a Household-level File

```
*-------------------------------------------------------------------------;
*       sort each input dataset by HHID GSUBHH
        subset variables;

proc sort
        data=in.h00a_r
        out=h00a_r(keep=hhid gsubhh pn g1051 g1053);
        by hhid gsubhh;
run;
proc sort
        data=in.h00j_h
        out=h00j_h(keep=hhid gsubhh g5068);
        by hhid gsubhh;
run;
*-------------------------------------------------------------------------;
*       merge two datasets by HHID GSUBHH to create a respondent file
        this is a one-to-many merge;
data resp;
        merge h00a_r h00j_h;
        by hhid gsubhh;
run;
```

4. SAS Creating a Household-level File with Respondent Data from up to Two Respondents

```
*-------------------------------------------------------------------------;
*       file of family respondents;
proc sort
        data=in.h00a_r(keep=hhid gsubhh gfamr pn g1051 g1053
        rename=(gfamr=gfamra pn=pna g1051=g1051a g1053=g1053a)
        where=(gfamra in(1 2 3 4)))
        out=h00a1;
```

```
        by hhid gsubhh;
run;

*-------------------------------------------------------------------------;
*       file of non-family respondents;
proc sort
        data=in.h00a_r(keep=hhid gsubhh gfamr pn g1051 g1053
        rename=(gfamr=gfamrb pn=pnb g1051=g1051b g1053=g1053b)
        where=(gfamrb not in(1 2 3 4)))
        out=h00a2;
        by hhid gsubhh;
run;

*-------------------------------------------------------------------------;
*       variables from a household-level file;
proc sort
        data=in.h00j_h(keep=hhid gsubhh g5068)
        out=h00j;
        by hhid gsubhh;
run;

*-------------------------------------------------------------------------;
*       household-level file with information from two respondents;
data hhold;
        merge
        h00a1 h00a2 h00j;
        by hhid gsubhh;
run;
```

5. SAS Creating a Respondent-level File with Parent Data from a Household-level File

```
*-------------------------------------------------------------------------;
*       select parent variables from household file for family r;
data famr;
        set in.h00d_h;
        *       keep id variables;
        keep hhid pn;
        *       keep parent variables of interest;
        keep g2122 g2123;
        *       assign person number for family r;
        pn=gpn_fam;
        attrib pn label='PERSON NUMBER' format=$char3.;
        *       output records for households with family R;
        if pn ne '';
run;
```

```
*-------------------------------------------------------------------------;
*       select parent variables from household file for non-family r;
data nfamr;
        set in.h00d_h;
        *       keep id variables;
        keep hhid pn;
        *       keep parent variables;
        keep g2122 g2123;
        *       copy in-law variables in to output variable location;
        g2122=g2309;
        g2123=g2310;
        *       assign person number for family r;
        pn=gpn_nfam;
        attrib pn label='PERSON NUMBER' format=$char3.;
        *       output records for households with non-family R;
        if pn ne '';
run;

*-------------------------------------------------------------------------;
*       concatenate files;
data resp;
        set famr nfamr;
run;
```

6. SAS Merging 1998 and 2000 Respondent-level Files -- Intersection

```
*-------------------------------------------------------------------------;
* subsetting keeping the variables of interest and ID variables;
data b00;
        set in00.h00b_r;
        keep HHID PN GSUBHH G1238 G1245 G1262 G1279 G1289 G1309 G1322
G1327 ;
run;
proc sort data=b00;
        by hhid pn;
run;

data b98;
        set in98.h98b_r;
        keep HHID PN FSUBHH F1109 F1116 F1129 F1146 F1156 F1156A F1176
F1194 F1189;
run;
proc sort data=b98;
        by hhid pn;
run;
```

```
*-------------------------------------------------------------------------;
/*
        create merged file

        In SAS a merge control variable is only created if you use the IN= option
        There are only two values for the merge control variable:
                0 - the record is not in common with the other data set
                1 - the record is common with the other data set
        the merge control variables are temporary variables which are not kept in the
        output dataset you may save them by assigning their value to permanent variable

        Example A in 1998 (N=21384) -- All respondents present in 1998
        if a=1; or if a;

        Example B in 2000 (N=19580) -- All respondents present in 2000
        if b=1; or if b;

        Example C in 1998 and 2000 (N=18858) -- Intersection
        if a=1 and b=1; or if a and b;

        Example D in 1998 or 2000 (N=22106) -- Union
        this is the default action or it may be explicity specified
        if a=1 or b=1; or if a or b;
*/

data b0098c;
        merge b98(in=a) b00(in=b);
        by hhid pn;
        in98=a;
        in00=b;
        if a and b;
run;
```

Stata Examples

1. Stata Merging Two or More Respondent-level Files

```
set prefix "hrs2000"
* This is an optional statement
* Make sure all the data files are in directory "hrs2000"

use HHID PN G1229 G1233 using h00b_r
sort HHID PN
save b_r, replace

use HHID PN G1654 G1655 using h00c_r
sort HHID PN
save c_r, replace

*       merge three datasets by HHID PN to create a respondent file
*    this is a one-to-one merge

use HHID PN G1051 G1053 using h00a_r
sort HHID PN
merge HHID PN using b_r
drop _m
sort HHID PN
merge HHID PN using c_r
drop _m
sort HHID PN
save resp, replace
```

2. Stata Merging Two or More Household-level Files

```
*-------------------------------------------------------------------------
*       sort each input dataset by HHID GSUBHH
*       subset variables

use HHID GSUBHH G3060 G3061 using h00f_h
sort HHID GSUBHH
save f_h, replace

use HHID GSUBHH G5068 using h00j_h
sort HHID GSUBHH
merge HHID GSUBHH using f_h
drop _m
sort HHID GSUBHH
save hhold, replace
```

3. Stata Merging a Respondent-level File with a Household-level File: Creating a Respondent-level File with Data from a Household-level File

```
*-------------------------------------------------------------------------
*       sort each input dataset by HHID GSUBHH
*       subset variables

use HHID GSUBHH G5068 using h00j_h
sort HHID GSUBHH
save j_h, replace

*-------------------------------------------------------------------------
*       merge two datasets by HHID GSUBHH to create a respondent file
*       this is a one-to-many merge

use HHID PN GSUBHH G1051 G1053 using h00a_r
sort HHID GSUBHH
merge HHID GSUBHH using j_h
tab _m
drop _m
save resp, replace
```

4. Stata Creating a Household-level File with Respondent Data from up to Two Respondents

```
*-------------------------------------------------------------------------
*       variables from a household-level file

use HHID GSUBHH G5068 using h00j_h
sort HHID GSUBHH
save j_h, replace

*-------------------------------------------------------------------------
*       file of family respondents

use HHID GSUBHH GFAMR PN G1051 G1053 using h00a_r
keep if GFAMR==1|GFAMR==2|GFAMR==3|GFAMR==4
rename GFAMR GFAMRa
rename PN PNa
rename G1051 G1051a
rename G1053 G1053a
sort HHID GSUBHH
save a_r1, replace

*-------------------------------------------------------------------------
*       start with file of non-family respondents and merge to create
```

```
*-------------------------------------------------------------------------
*       household-level file with information from two respondents;

use HHID GSUBHH GFAMR PN G1051 G1053 using h00a_r
keep if GFAMR~=1 & GFAMR~=2 & GFAMR~=3 & GFAMR~=4
rename GFAMR GFAMRb
rename PN PNb
rename G1051 G1051b
rename G1053 G1053b
sort HHID GSUBHH
merge HHID GSUBHH using a_r1
tab _m
drop _m
sort HHID GSUBHH
merge HHID GSUBHH using j_h
tab _m
drop _m
sort HHID GSUBHH
save hhold, replace
```

5. Stata Creating a Respondent-level File with Parent Data from a Household-level File

```
*-------------------------------------------------------------------------
*       select parent variables from household file for family r
use h00d_h, clear

* assign person number for family r
gen str3 PN=GPN_FAM
label var PN "PERSON NUMBER"

* keep ID variables and parent variables of interest
keep HHID PN G2122 G2123

*       output records for households with family R
keep if PN~=""
save fam1, replace

* select parent variables from household file for non-family r
use h00d_h, clear
drop G2122 G2123

*-------------------------------------------------------------------------
*       select parent variables from household file for non-family r
*       assign person number for family r
gen str3 PN=GPN_NFAM
```

```
* copy in-law variables in to output variable location
gen G2122=G2309
gen G2123=G2310

* keep ID variables and parent variables of interest
keep HHID PN G2122 G2123

*       output records for households with non-family R
keep if PN~=""
save fam2, replace

*-------------------------------------------------------------------------
* concatenate  files
use fam1, clear
append using fam2
save resp, replace
```

6. Stata Merging 1998 and 2000 Respondent-level Files -- Intersection

```
*-------------------------------------------------------------------------
* subsetting keeping the variables of interest and ID variables
use H98B_R, clear
keep HHID PN FSUBHH F1109 F1116 F1129 F1146 F1156 F1156A F1176 F1194
F1189
sort HHID PN
save B98, replace

use H00B_R, clear
keep HHID PN GSUBHH G1238 G1245 G1262 G1279 G1289 G1309 G1322 G1327
sort HHID PN
save B00, replace

*-------------------------------------------------------------------------
*       the merge
use B98
sort HHID PN
merge HHID PN using B00
sort HHID PN

*       create merge file
*       in Stata when a merge is done a merge variable, _merge, is created
*       the values of _merge are :
*               1 - record only in the first data set
*               2 - record only in the second data set
*               3 - record in common between datsets (intersection)
```

```
*       The default merge is the union of the two datasets

* only retain records that appear in both input files
keep if _merge==3

keep HHID PN FSUBHH GSUBHH
*       drop PNR PNSP HOLD _merge

save B0098, replace
```