Centre for Longitudinal Studies,

Institute of Education,

University of London.

The Millenium Cohort Study

User Guide to Analysing MCS Data Using STATA

Sosthenes C. Ketende
Tel: 020 7612 6899
Email: s.ketende@ioe.ac.uk

Elizabeth M. Jones
Tel: 020 7911 5566
Email: e.jones@ioe.ac.uk

December, 2011

Centre for Longitudinal Studies
Institute of Education, University of London
20 Bedford Way
London
WC1H 0AL

website: www.cls.ioe.ac.uk

ISBN 978-1-906929-35-0

The Centre for Longitudinal Studies (CLS) is a department within the Faculty of Policy and Society of the Institute of Education, University of London. The department houses an ESRC Resource Centre devoted to the collection, management and analysis of large-scale longitudinal data. It is the home of three internationally-renowned birth cohort studies: the 1958 National Child Development Study (NCDS), the 1970 British Cohort Study (BCS) and the Millennium Cohort Study (MCS).

Contents

List of Figures

1 Introduction to Stata

This section will cover a short introduction to `Stata` mainly aimed at those using `Stata` for the first time.

1.1 Main `Stata` window

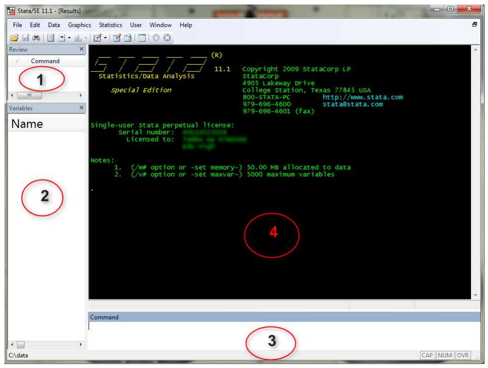

Figure 1: Stata main window

Following is a description of each window:

1. **Review**
   - Here you will see all the issued or excuted `Stata` commands. You can re-issue commands just by double clicking them in this window.

2. **Variables**
   - Variables of the dataset in memory are displayed here. It can be resized to the right to see variable labels.

3. **Command**
   - This where you input all `Stata` commands.

4. **Results**
   - All results of the issued commands are displayed here

1.2 Menu bar

Figure 2: Stata menu bar

Hover your mouse pointer over the menu bar to see what each button does. You can browse the data in memory using the `Data Browser`, edit data using the `Data Editor`, create a new do-file using `New Do-file Editor`, manage log files with `Log Begin/Close/Suspend/Resume`, print the contents of the results window with `Print Results`, save your data file with `Save`, or open a new data file with `Open (use)`.

1.3 Commands for getting started

A few basic commands are useful for keeping `Stata` up to date and for getting ready to work with data files.

- **Update**
  - Make sure that your Stata is up to date with changes and add-ons by typing `update query` into the command window. Stata will let you know if there are any updates that you do not yet have. You can then click on links in the results window to get updates that you need. Updates come out frequently, so be sure to check regularly.

- **Help**
  - If there is a command that you are not sure how to use, you can get help on it by typing `help [command]` into the command window. This will give you detailed information on the command, including correct format, available command options, and examples.

- **Clear**
  - Before you start working with a new data file, you should first enter `clear` into the command window. This will remove any data from memory; you cannot open a new data file in `Stata` when you already have one in memory.

- **Set memory**
  - The MCS data files are very large and are usually too big for the standard amount of memory in `Stata` to handle. Before starting to work with MCS data files, you will have to allocate more memory. You can do this with the command `set memory 512m`. If your computer does not have enough memory it this will not work, and you will have to reduce the memory amount. If this happens, try to find the largest amount that your computer will allow you to allocate.

- **Version control** At the beginning of a do file, you should specify the version of Stata you are using at the time, using the command `version [ver]`, e.g., `version 10.1`. This will allow you to continue to use your do file even on later versions of Stata.

1.4 Do-file editor

You can enter commands directly into the command window, but you can enter only one command at a time this way and your commands will not be saved. You can create files of commands, called do files in `Stata`, using the do-file editor. To start the do-file editor, click the `New Do-file Editor` button on the menu bar or simply hit Ctrl+8.

The do-file editor has its own menu and menu bar. Hover over the buttons in the menu bar to see what they do. You can use the buttons to open an existing do file, save your file, and run all or part of your do file. To execute the entire file, you can click the `Do` button, select `Do` under `Tools` in the menu, or hit Ctrl+D. You can execute commands from your cursor location to the end of the file by clicking `Do to Bottom` under `Tools` or hitting Ctrl+Shift+D. To execute specific commands, highlight the line(s) you wish to execute and select `Do Selection` under `Tools` or click the `Do` button, which will appear as the `Do Selected Lines` button when you have something highlighted.

2 Preparing data for analysis

In the following section we will introduce you to the first steps of analysing data.

2.1 Starting a do file

We will work with the MCS data using a do file rather than by entering commands directly into the command window. Start a new do file, as explained above in Section 1.4. Save your do file as **mcsuser.do**.

You should start your do file with the commands below preceded by the dots (don't include the dots when you enter the commands). Most of these commands have been described above. To see the results of your commands, highlight the lines you want to execute and click the `Do Selected Lines` button.

```
.
. version 11.1
. clear all
. set more off
. set memory 712m
Current memory allocation

                    current                                memory usage
    settable          value     description                 (1M = 1024k)
    --------------------------------------------------------------------
    set maxvar        20000     max. variables allowed           7.631M
    set memory         712M     max. data space                712.000M
    set matsize        8000     max. RHS vars in models        488.953M
```

```
                                                                   1,208.584M
. set maxvar 28000
Current memory allocation
                    current                                        memory usage
    settable          value     description                        (1M = 1024k)
    ---------------------------------------------------------------------------
    set maxvar        28000     max. variables allowed                10.683M
    set memory         712M     max. data space                      712.000M
    set matsize        8000     max. RHS vars in models              488.953M
                                                                  -----------
                                                                   1,211.635M
. set matsize 6000
Current memory allocation
                    current                                        memory usage
    settable          value     description                        (1M = 1024k)
    ---------------------------------------------------------------------------
    set maxvar        28000     max. variables allowed                10.683M
    set memory         712M     max. data space                      712.000M
    set matsize        6000     max. RHS vars in models              275.162M
                                                                  -----------
                                                                     997.845M
```

The results of the set commands above show how much memory has been allocated for variables, the data, and the data matrix. The total amount of memory allocated is about 1GB. If your computer does not have this much free memory available, you will get an error message.

The following command will change the current `Stata` working directory to an MCS4 directory on drive F:. Please note that the drive letter and directory on your computer maybe different.

```
.
. cd F:\mcs4
 cd F:\mcs4
```

2.2 Merging datasets from different MCS sweeps

MCS datasets are stored cross-sectionally, i.e., data from each sweep are stored separately. Additionally, often there are datasets from other questionnaires in the same sweep which are stored separately from the main interview datasets. Thus, merging files for analysis is sometimes unavoidable. We will merge datasets from sweeps 1, 2, 3 and 4 onto the dataset which holds survey design variables such as sample weights.
We start by using the family level dataset i.e make it the active file(in computer memory). Next merge all the others using the `merge` command as follows:

```
. use mcs_longitudinal_family_file.dta, clear
.
. * this file contains negative weights for non-productive cases.
. * Please change these to system missing before you start your analysis as follows:
.
. foreach var of varlist *wt1 *wt2*{
```

```
  2. replace `var´=. if `var´<0
  3. }
(692 real changes made, 692 to missing)
(3654 real changes made, 3654 to missing)
(3998 real changes made, 3998 to missing)
(5387 real changes made, 5387 to missing)
(692 real changes made, 692 to missing)
(3654 real changes made, 3654 to missing)
(3998 real changes made, 3998 to missing)
(5387 real changes made, 5387 to missing)
.

.
. merge m:m mcsid using mcs1_parent_interview
    Result                           # of obs.
    -----------------------------------------
    not matched                           692
        from master                       692  (_merge==1)
        from using                          0  (_merge==2)

    matched                            18,552  (_merge==3)
    -----------------------------------------

. drop _merge

. merge m:m mcsid using mcs2_parent_interview
    Result                           # of obs.
    -----------------------------------------
    not matched                         3,654
        from master                     3,654  (_merge==1)
        from using                          0  (_merge==2)

    matched                            15,590  (_merge==3)
    -----------------------------------------

. drop _merge

. merge m:m mcsid using mcs3_parent_interview
    Result                           # of obs.
    -----------------------------------------
    not matched                         3,998
        from master                     3,998  (_merge==1)
        from using                          0  (_merge==2)

    matched                            15,246  (_merge==3)
    -----------------------------------------


. drop _merge

. merge m:m mcsid using mcs4_parent_interview

    Result                           # of obs.
    -----------------------------------------
    not matched                         5,387
        from master                     5,387  (_merge==1)
        from using                          0  (_merge==2)

    matched                            13,857  (_merge==3)
    -----------------------------------------

. drop _merge

. merge m:m mcsid using mcs3_child_assessment_data
    Result                           # of obs.
    -----------------------------------------
    not matched                         3,998
        from master                     3,998  (_merge==1)
        from using                          0  (_merge==2)

    matched                            15,460  (_merge==3)
    -----------------------------------------
```

```
. drop _merge
```

3 Generating variables

In this section we will use the following `Stata` commands: `generate`, `replace`, `label variable`, `label define`, `label values` and `codebook`. Let's begin by generating the **f**inite **p**opulation **c**orrection factor (fpc) if it is not in your data file.

```
. generate Nh2=5289 if pttype2==1
. replace  Nh2=1853 if pttype2==2
. replace  Nh2=169  if pttype2==3
. replace  Nh2=345  if pttype2==4
. replace  Nh2=274  if pttype2==5
. replace  Nh2=709  if pttype2==6
. replace  Nh2=409  if pttype2==7
. replace  Nh2=258  if pttype2==8
. replace  Nh2=242  if pttype2==9
```

Another useful command is `codebook`. It is useful in providing quick, basic information about variables.

```
. codebook  amvote00  adgmai00 cmvote00  cdgmai00
----------------------------------------------------------------------
amvote00              s1 main voted in last election
----------------------------------------------------------------------

                 type:  numeric (byte)
                label:  amvote00

                range:  [-9,2]                       units:  1
        unique values:  5                        missing .:  692/19244

           tabulation:  Freq.   Numeric  Label
                           18        -9  refusal
                           38        -8  don´t know
                           38        -1  not applicable
                         9318         1  yes
                         9140         2  no
                          692         .
----------------------------------------------------------------------
adgmai00              s1 dv main respondent age at interview (grouped)
----------------------------------------------------------------------

                 type:  numeric (byte)
                label:  adgmai00

                range:  [-2,4]                       units:  1
        unique values:  5                        missing .:  692/19244

           tabulation:  Freq.   Numeric  Label
                           10        -2  not known
                         1068         1  14 to 19
                         8208         2  20 to 29
                         8632         3  30 to 39
                          634         4  40 plus
                          692         .

----------------------------------------------------------------------
```

```
cmvote00              s3 main: whether voted in general election
--------------------------------------------------------------------

                  type:  numeric (byte)
                 label:  cmvote00

                 range:  [-9,2]                       units:  1
         unique values:  5                        missing .:  3998/19244

            tabulation:  Freq.   Numeric  Label
                             9        -9  refusal
                            55        -8  don´t know
                            74        -1  not applicable
                          9027         1  yes
                          6081         2  no
                          3998         .
--------------------------------------------------------------------
cdgmai00              s3 dv main respondent age at interview (grouped)
--------------------------------------------------------------------

                  type:  numeric (byte)
                 label:  cdgmai00

                 range:  [1,4]                        units:  1
         unique values:  4                        missing .:  3998/19244

            tabulation:  Freq.   Numeric  Label
                             5         1  16 to 19
                          3723         2  20 to 29
                          8753         3  30 to 39
                          2765         4  40 plus
                          3998         .
```

So far we have been using the entire MCS 1 to 4 datasets as created by merging all the parental interview files together. This file occupies a lot of computer memory. We are now going to select and keep only the variables we need for the analysis and save the resulting file as **mcsuserwkshp.dta**. **Please do not email or take outside of this room a copy or part of this file. Use UKDA procedures to acquire datasets**.

```
. keep   mcsid sentry country pttype2 sptn00 weight1 weight2
Nh2 aaoutc00 baoutc00 *aoutc00   aovwt1 aovwt2 bovwt1 bovwt2
*ovwt* *ovwt* amvote00  adgmai00 adm06e00 adnvqm00 adrelp00
admwrk00  bmvote00  bdgmai00 bdm06e00  bdrelp00 bdmwrk00
bdmsam00 cmvote00  cdgmai00 cdm06e00 cdnvqm00 cdrelp00
cdmwrk00 cdmsam00 *bmin*  dhcsexa0 ddmbmi00 *rrso*  *clsl*
*acti* *loil* *lolm* *seho* *plfr* *fapa* *tvho* *tvrm* *cnum*
*smus*

. save mcsuserwkshp, replace
```

4 Weighting - Recap

Different analyses require the use of different weights as you have heard today in earlier sessions. The table below sets out when various kinds of available weights can be used.

4.1 Sampling Weights

Type of Analysis	Weight to be Used
UK country specific level analyses	Weight1
Whole of UK-level analysis	Weight2
UK country specific level analyses within Ward type	No weight*

* because the sample is self-weighting.

4.2 Attrition /non-response weights

Type of Analysis	Wave (sweep)	Weight to be Used
UK country specific level analyses	S1	aovwt1
Whole of UK-level analysis	S1	aovwt2
UK country specific level analyses	S2	bovwt1
Whole of UK-level analysis	S2	bovwt2
GB only analysis i.e. excluding NI	S2	bovwtgb
UK country specific level analyses	S3	covwt1
Whole of UK-level analysis	S3	covwt2
GB only analysis i.e. excluding NI	S3	covwtgb
UK country specific level analyses	S4	dovwt1
Whole of UK-level analysis	S4	dovwt2
GB only analysis i.e. excluding NI	S4	dovwtgb
UK country specific level analyses within Ward type	All waves	No weight*

* because the sample is self-weighting.

4.3 Qn: I have wave t outcome but wave t-1 predictors, which weight should I use?

You often will be working with data from more than one sweep. Which weight should you use in that situation? If you have, for example, an outcome variable from MCS 4, but predictor variables from MCS 3, MCS 2 and MCS 1, you should use an MCS 4 weight. This is because the sample that you are using will be restricted to families who took part in MCS 4.

4.4 Definitions

Stratification. MSC is stratified by design. There 9 different strata with all UK countries having two strata i.e. Advantaged and disadvantaged. England has one more strata for Ethnic minorities. The stratum variable is called pttype2.

Clustering. MCS is also clustered at ward level. Wards were the primary sampling unit. A few wards were combined into one making what is often referred to as super-wards. The ward variable is called sptn00.

Finite Population Correction factor (fpc). When the size of the sample becomes a large fraction of the size of the population we use something called a finite population correction factor (fpc). The finite population correction factor measures how much extra precision we achieve when the sample size become close to the population size.

4.5 Cross-sectional analyses

Even when you are doing cross-sectional analyses of a single sweep (other than the first) of MCS data, you may find that you need to merge together multiple sweeps. This is because some questions were asked only of respondents who did not answer the question at the last sweep, or those whose answers have changed since the last sweep.

For example, at MCS 3 respondents were asked whether they had earned any new qualifications, and if they had were asked what those were. With only MCS 3 data, you will have qualifications for those who obtain new ones or those who did not answer at MCS 2, but you will not have any data on qualifications for those who did not obtain any new ones since MCS 2. To have data on qualifications for everyone, you will have to merge on MCS 2 data.

5 Setting data for analysis

The two key `Stata` commands needed here are `svyset` and `svydes`. We will set up sweep 1 data for whole of UK analyses using `svyset` as follows:

```
. svyset sptn00 [pweight=aovwt2], strata(pttype2) fpc(Nh2)
      pweight: aovwt2
          VCE: linearized
  Single unit: missing
     Strata 1: pttype2
         SU 1: sptn00
        FPC 1: Nh2
```

Let's now describe the data using `svydes` and see some survey design settings and values and, if correct, continue with the analysis.

```
. noi svydes
Survey: Describing stage 1 sampling units
      pweight: aovwt2
          VCE: linearized
  Single unit: missing
     Strata 1: pttype2
         SU 1: sptn00
        FPC 1: Nh2
                                         #Obs per Unit
                                  ----------------------------
Stratum   #Units     #Obs      min       mean      max
-------- -------- -------- -------- -------- --------
       1      110     4617        7     42.0      131
       2       71     4522        9     63.7      142
```

```
        3          19        2394          45       126.0         403
        4          23         832          13        36.2         120
        5          50        1928          12        38.6         238
        6          32        1145          14        35.8          88
        7          30        1191          14        39.7          93
        8          23         723          14        31.4          73
        9          40        1200          12        30.0          74
 ----------  ----------  ----------  ----------  ----------  ----------
        9         398       18552           7        46.6         403
                              692 = #Obs with missing values in the
                       ----------   survey characteristcs
                            19244
```

All seems to be ok. We have 9 strata with sample units (wards) ranging from 19 (England Ethnic) to 110 (England Advantaged). Take note of the 692 cases with missing values in the survey characteristics and the minimum, mean and maximum number of families per stratum.

After `svyset`, all subsequent analyses will use the same design features. You don't have to re-issue it each time you are running a new estimation command unless you are changing survey design features such as the weight variable.

6 Data analysis

In this section we will carry out survey data analyses using `Stata` 11.1.

6.1 Descriptive analyses

To demonstrate how MCS data can be analysed correctly using `Stata` we will look at cross-sectional predictors of voting in previous election using sweep 1 variables.

6.1.1 MCS1 predictor variables of voting in previous election

The following is a tabulation of our main outcome (dependent) variable, **amvote00**. We have three options `percent obs` and `format` which produce percentages and observed samples, and format estimates to 3 decimal places respectively. To obtain weighted sample sizes use `count` instead of the `obs` option.

```
. svy:tab amvote00, percent obs format(%9.3g)
(running tabulate on estimation sample)

Number of strata   =         9                  Number of obs      =     18552
Number of PSUs     =       398                  Population size    = 18552.967
                                                Design df          =       389

----------------------------------
s1 main    |
voted in   |
last       |
election   | percentages          obs
```

```
             |
   refusal   |     .0505        18
  don´t kn   |      .152        38
  not appl   |      .148        38
       yes   |        51      9318
        no   |      48.7      9140
             |
     Total   |       100     18552

  Key:  percentages  =  cell percentages
        obs          =  number of observations
```

The output above shows that we have some cleaning work to do. Let's replace the first three values with `Stata` missing values.

```
. replace amvote00=. if inlist(amvote00,-9,-8,-1 )
(94 real changes made, 94 to missing)
```

We also need to have the variable in a binary 1/0 format for logistic regression.

```
. replace amvote00=0 if amvote00==2
(9140 real changes made)
```

And let's correct the value labels to 0=No, 1=Yes. Remember that the value label has 1=Yes already... we only have to add 0=No.

```
. label define amvote00 0 No, add
```

Let's do a similar replacement of values with `Stata` missing as above on other variables all at once. Please use this command carefully to avoid unintended results. Check all the variables using a command such as `codebook` first before using it.

```
. foreach var of varlist adgmai00 adm06e00 adnvqm00 adrelp00 admwrk00{
  replace `var´=. if `var´<0
   }
(10 real changes made, 10 to missing)
(51 real changes made, 51 to missing)
(0 real changes made)
(3194 real changes made, 3194 to missing)
(0 real changes made)
```

Shown below is a cross tabulation between the new (clean) dependent variable and main respondent's age (grouped). We requested row percentages by using the `row` option. To obtain column percentages use the `column` option.

```
. svy:tab  adgmai00 amvote00, row percent obs format(%9.3g)
(running tabulate on estimation sample)

Number of strata   =         9                  Number of obs      =     18448
Number of PSUs     =       398                  Population size    = 18480.811
                                                Design df          =       389

s1 dv      |
main       |
responden  |
t age at   |  s1 main voted in
interview  |    last election
(grouped)  |    No    yes  Total
           |
```

```
 14 to 19 |     86     14    100
          |    909    149   1058
          |
 20 to 29 |   58.3   41.7    100
          |   4681   3491   8172
          |
 30 to 39 |   38.5   61.5    100
          |   3349   5239   8588
          |
  40 plus |   31.9   68.1    100
          |    194    436    630
          |
    Total |   48.8   51.2    100
          |   9133   9315  18448

  Key:  row percentages
        number of observations
```

6.1.2 MCS3 predictors of voting in previous election

Since we are now switching from using sweep 1 variables to sweep 3 variables, the first thing we have to do is change the weight from sweep 1 to sweep 3. This is done by running the `svyset` command as we did with sweep 1 data but this time changing the `pweight` from aovwt2 to covwt2. If at any time you would like to know your survey design settings such which weight you are using, just use the `svydes` command.

```
. svyset sptn00 [pweight=covwt2], strata(pttype2) fpc(Nh2)

      pweight: covwt2
          VCE: linearized
  Single unit: missing
     Strata 1: pttype2
         SU 1: sptn00
        FPC 1: Nh2

. svydes

Survey: Describing stage 1 sampling units

      pweight: covwt2
          VCE: linearized
  Single unit: missing
     Strata 1: pttype2
         SU 1: sptn00
        FPC 1: Nh2
```

			#Obs per Unit		
Stratum	#Units	#Obs	min	mean	max
1	110	4069	6	37.0	105
2	71	3759	9	52.9	120
3	19	1889	34	99.4	306
4	23	669	11	29.1	98
5	50	1512	9	30.2	174
6	32	917	11	28.7	70
7	30	897	10	29.9	65
8	23	594	11	25.8	64
9	40	940	8	23.5	50
9	398	15246	6	38.3	306

```
        3998 = #Obs with missing values in the
      --------   survey characteristcs
       19244
```

The 3,998 cases were un-productive including those not issued at sweep 3 of the MCS.

Below is a tabulation of the voting variable at sweep 3. As you can see, we have to do some cleaning and change it to 1/0 variable before running regression models.

```
. svy:tab cmvote00, percent obs format(%9.3g)
(running tabulate on estimation sample)

Number of strata   =         9                  Number of obs      =     15246
Number of PSUs     =       398                  Population size    = 15604.448
                                                Design df          =       389

----------------------------------------
s3 main:  |
whether   |
voted in  |
general   |
election  |  percentages           obs
----------+-----------------------------
  refusal |        .0579             9
 don´t kn |         .356            55
 not appl |         .407            74
      yes |         58.2          9027
       no |           41          6081
          |
    Total |          100         15246
----------------------------------------
  Key:  percentages  =  cell percentages
        obs          =  number of observations

. replace cmvote00=. if inlist(cmvote00,-9,-8,-1 )
(138 real changes made, 138 to missing)

. replace cmvote00=0 if cmvote00==2
(6081 real changes made)

. label define cmvote00 0 No, add
```

The same tabulation as above, but on a cleaned voting variable.

```
. svy:tab cmvote00, percent obs format(%9.3g)
(running tabulate on estimation sample)

Number of strata   =         9                  Number of obs      =     15108
Number of PSUs     =       398                  Population size    = 15476.328
                                                Design df          =       389

----------------------------------------
s3 main:  |
whether   |
voted in  |
general   |
election  |  percentages           obs
----------+-----------------------------
       No |         41.3          6081
      yes |         58.7          9027
          |
```

```
       Total |          100        15108
-------------+--------------------------
  Key:  percentages  =  cell percentages
        obs          =  number of observations
```

Next we will have once again to clean predictor variables as we did for sweep 1 data. But first, a tabulation of one variable, **cdrelp00**, shows that there are cases coded as “not applicable” which in fact are single parents. So we need to create a new value label for this group and make the necessary changes before issuing a global replace command to all predictor variables.

```
. replace cdrelp00=3 if cdrelp00==-1
(3021 real changes made)

. label define cdrelp00 3 single, modify

. foreach var of varlist cdgmai00 cdm06e00 cdnvqm00 cdrelp00 cdmwrk00{
          replace `var´=. if `var´<=0
          }
(0 real changes made)
(246 real changes made, 246 to missing)
(0 real changes made)
(60 real changes made, 60 to missing)
(12 real changes made, 12 to missing)
```

See a cross tabulation between the cleaned variable and age of main respondent.

```
. svy:tab  cdgmai00 cmvote00, row percent obs format(%9.3g)
(running tabulate on estimation sample)

Number of strata   =         9                Number of obs      =     15108
Number of PSUs     =       398                Population size    = 15476.325
                                              Design df          =       389

-------------------------------
s3 dv      |
main       |
responden  |  s3 main: whether
t age at   |  voted in general
interview  |      election
(grouped)  |    No    yes  Total
-----------+-------------------
 16 to 19  |   100      0    100
           |     5      0      5
           |
 20 to 29  |  64.7   35.3    100
           |  2226   1445   3671
           |
 30 to 39  |  36.1   63.9    100
           |  3059   5628   8687
           |
  40 plus  |  28.3   71.7    100
           |   791   1954   2745
           |
    Total  |  41.3   58.7    100
           |  6081   9027  15108
-------------------------------
  Key:  row percentages
        number of observations

  Pearson:
    Uncorrected   chi2(3)         = 1094.4604
    Design-based  F(2.92, 1136.22)=  240.2143     P = 0.0000
```

Notice the lowest age group where there are only five cases, as expected. We will add these cases to the next age group. You may wish to change the value label to reflect the fact that age group 20-29 is now 16-29.

```
. replace cdgmai00=2 if cdgmai00==1
(5 real changes made)
```

7 Multivariate analysis: Healthy diet in children

We will now carry out a multivariate analysis using an outcome variable from MCS 4. The outcome we are going to use is whether the main respondent reports controlling the cohort member's diet in order to make it healthier. We will use predictor variables from MCS 3.

7.1 Creating the MCS 4 outcome variable

The outcome variable was generated as follows (this variable is already in your file so you do not need to create it):

```
. forvalues i=1/4{
  2. generate dreason`i´=0 if daoutc00==1
  3. replace dreason`i´=1 if (dmrrsoaa==`i´|dmrrsoab==`i´|dmrrsoac==`i´|dmrrsoa
> d==`i´|dmrrsoae==`i´|dmrrsoaf==`i´|dmrrsoag==`i´|dmrrsoah==`i´|dmrrsoba==`i´|
> dmrrsobb==`i´|dmrrsobc==`i´|dmrrsobd==`i´|dmrrsobe==`i´|dmrrsobf==`i´|dmrrsob
> g==`i´|dmrrsobh==`i´|dmrrsoca==`i´|dmrrsocb==`i´|dmrrsocc==`i´|dmrrsocd==`i´|
> dmrrsoce==`i´|dmrrsocf==`i´|dmrrsocg==`i´|dmrrsoch==`i´)
  4. }
(5387 missing values generated)
(5287 real changes made)
(5387 missing values generated)
(533 real changes made)
(5387 missing values generated)
(717 real changes made)
(5387 missing values generated)
(1252 real changes made)

.
. label var       dreason1        "DV S4 Healthy/balanced diet"

.
. tab1 dreason1

-> tabulation of dreason1
```

DV S4 Healthy/balanced diet	Freq.	Percent	Cum.
0	8,570	61.85	61.85
1	5,287	38.15	100.00
Total	13,857	100.00	

7.2 MCS 3 predictors of healthy diet in children at MCS 4

We reset the survey set-up by using sweep 4 non-response adjusted weight.

```
. svyset sptn00 [pweight=dovwt2], strata(pttype2) fpc(Nh2)

      pweight: dovwt2
          VCE: linearized
  Single unit: missing
     Strata 1: pttype2
         SU 1: sptn00
        FPC 1: Nh2
```

It is important to choose a reasonable reference category in any analysis dealing with categorical data. By default, `Stata` uses the group with the lowest integer for reference. You can change this very easily by declaring just before the regression command which group is to be the reference group for a given variable in STATA 10. In STATA 11 this is specified for each variable in the variable list.

Shown below is an example of how to specify the reference group in STATA 11, where we have selected a category with value 96 for the educational qualification variable to be the reference group. Category 96 are main respondents without NVQ qualifications.

In the `logit` command below, the prefix `i.` indicates that a variable is categorical, and STATA will automatically create dummy variable for it. The `b` followed by a number in the prefix indicates which category you would like to be the reference group. If you leave out the `b` and number, STATA will use the lowest value as the reference group.

```
. svy:logit dreason1  ib4.cdgmai00 ib6.cdm06e00 ib96.cdnvqm00 ib3.cdrelp00 ib2.
> cdmwrk00
(running logit on estimation sample)

Survey: Logistic regression

Number of strata   =         9                Number of obs      =     12941
Number of PSUs     =       398                Population size    = 12961.133
                                              Design df          =       389
                                              F(  16,    374)    =     34.31
                                              Prob > F           =    0.0000
```

dreason1	Coef.	Linearized Std. Err.	t	P>|t|	[95% Conf.	Interval]
cdgmai00						
2	-.2789476	.0726775	-3.84	0.000	-.4218374	-.1360577
3	.0096113	.056795	0.17	0.866	-.1020523	.1212749
cdm06e00						
1	.7970259	.2218618	3.59	0.000	.3608275	1.233224
2	1.111088	.3078125	3.61	0.000	.5059037	1.716272
3	.6464354	.3079056	2.10	0.036	.0410679	1.251803
4	.4458498	.2495537	1.79	0.075	-.0447931	.9364926
5	.8143247	.2631857	3.09	0.002	.2968802	1.331769
cdnvqm00						
1	.191847	.1173472	1.63	0.103	-.038867	.4225611
2	.6399257	.0944488	6.78	0.000	.4542316	.8256197
3	.9190234	.0966886	9.50	0.000	.7289259	1.109121
4	1.222054	.0913943	13.37	0.000	1.042366	1.401743
5	1.36479	.1138911	11.98	0.000	1.140871	1.588709
95	.1287386	.1744159	0.74	0.461	-.2141772	.4716543

```
    cdrelp00 |
          1  |   .0453954   .0616026     0.74   0.462    -.0757204    .1665112
          2  |  -.0608961   .0743182    -0.82   0.413    -.2070117    .0852195
             |
  1.cdmwrk00 |   .0991952    .050164     1.98   0.049     .0005688    .1978217
       _cons |  -2.069142   .2290705    -9.03   0.000    -2.519513   -1.618771
--------------------------------------------------------------------------------
```

To get odds ratios instead of coefficients is easily done by issuing `svy:logit, or` after the regression above. See the results below.

```
. svy:logit, or
Survey: Logistic regression
Number of strata   =         9                  Number of obs      =     12941
Number of PSUs     =       398                  Population size    = 12961.133
                                                Design df          =       389
                                                F(  16,    374)    =     34.31
                                                Prob > F           =    0.0000

--------------------------------------------------------------------------------
             |             Linearized
    dreason1 | Odds Ratio   Std. Err.       t    P>|t|      [95% Conf. Interval]
-------------+------------------------------------------------------------------
    cdgmai00 |
          2  |   .7565796   .0549863    -3.84   0.000     .6558407    .8727922
          3  |   1.009658   .0573435     0.17   0.866     .9029824    1.128935
             |
    cdm06e00 |
          1  |   2.218932   .4922963     3.59   0.000     1.434516    3.432278
          2  |   3.037662   .9350303     3.61   0.000     1.658484     5.56375
          3  |   1.908725   .5877071     2.10   0.036     1.041923    3.496641
          4  |   1.561817   .3897572     1.79   0.075     .9561953    2.551018
          5  |   2.257651   .5941814     3.09   0.002     1.345654    3.787739
             |
    cdnvqm00 |
          1  |   1.211485   .1421644     1.63   0.103     .9618786    1.525865
          2  |    1.89634   .1791071     6.78   0.000     1.574963    2.283295
          3  |   2.506841   .2423829     9.50   0.000     2.072853    3.031692
          4  |   3.394153   .3102064    13.37   0.000     2.835918    4.062274
          5  |   3.914899    .445872    11.98   0.000     3.129492    4.897421
         95  |   1.137393   .1983794     0.74   0.461     .8072053    1.602643
             |
    cdrelp00 |
          1  |   1.046442   .0644636     0.74   0.462     .9270754    1.181177
          2  |    .940921   .0699275    -0.82   0.413     .8130102    1.088956
             |
  1.cdmwrk00 |   1.104282   .0553952     1.98   0.049     1.000569    1.218745
--------------------------------------------------------------------------------
```

7.3 Categorical variable's overall significance in a model

Sometimes it is necessary to test a categorical variable to see whether it should be in your substantive model or not. This might be an important check if say a 6 category variable has most categories non-significant in comparison to the reference group. The test is done following a regression estimation, as shown below.

```
. svyset sptn00 [pweight=dovwt2], strata(pttype2) fpc(Nh2)
```

```
      pweight: dovwt2
          VCE: linearized
  Single unit: missing
     Strata 1: pttype2
         SU 1: sptn00
        FPC 1: Nh2
. * selecting a reference age group e.t.c
.
. svy:logit dreason1  ib4.cdgmai00 ib6.cdm06e00 ib96.cdnvqm00 ib3.cdrelp00 ib2.
> cdmwrk00
(running logit on estimation sample)

Survey: Logistic regression

Number of strata   =         9                Number of obs      =     12941
Number of PSUs     =       398                Population size    = 12961.133
                                              Design df          =       389
                                              F(  16,    374)    =     34.31
                                              Prob > F           =    0.0000

------------------------------------------------------------------------------
             |             Linearized
    dreason1 |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
    cdgmai00 |
          2  |  -.2789476    .0726775    -3.84   0.000    -.4218374   -.1360577
          3  |   .0096113     .056795     0.17   0.866    -.1020523    .1212749
             |
    cdm06e00 |
          1  |   .7970259    .2218618     3.59   0.000     .3608275    1.233224
          2  |   1.111088    .3078125     3.61   0.000     .5059037    1.716272
          3  |   .6464354    .3079056     2.10   0.036     .0410679    1.251803
          4  |   .4458498    .2495537     1.79   0.075    -.0447931    .9364926
          5  |   .8143247    .2631857     3.09   0.002     .2968802    1.331769
             |
    cdnvqm00 |
          1  |    .191847    .1173472     1.63   0.103     -.038867    .4225611
          2  |   .6399257    .0944488     6.78   0.000     .4542316    .8256197
          3  |   .9190234    .0966886     9.50   0.000     .7289259    1.109121
          4  |   1.222054    .0913943    13.37   0.000     1.042366    1.401743
          5  |    1.36479    .1138911    11.98   0.000     1.140871    1.588709
         95  |   .1287386    .1744159     0.74   0.461    -.2141772    .4716543
             |
    cdrelp00 |
          1  |   .0453954    .0616026     0.74   0.462    -.0757204    .1665112
          2  |  -.0608961    .0743182    -0.82   0.413    -.2070117    .0852195
             |
  1.cdmwrk00 |   .0991952     .050164     1.98   0.049     .0005688    .1978217
       _cons |  -2.069142    .2290705    -9.03   0.000    -2.519513   -1.618771
------------------------------------------------------------------------------
```

Here is how the test is done on each variable in the model above. Notice the i. at the beginning of the variable name.

```
. testparm i.cdgmai00

Adjusted Wald test

 ( 1)  [dreason1]2.cdgmai00 = 0
 ( 2)  [dreason1]3.cdgmai00 = 0

       F(  2,   388) =   12.82
            Prob > F =    0.0000

. testparm i.cdm06e00

Adjusted Wald test

 ( 1)  [dreason1]1.cdm06e00 = 0
 ( 2)  [dreason1]2.cdm06e00 = 0
 ( 3)  [dreason1]3.cdm06e00 = 0
 ( 4)  [dreason1]4.cdm06e00 = 0
```

```
 ( 5)  [dreason1]5.cdm06e00 = 0

       F(  5,   385) =    4.22
            Prob > F =    0.0010

. testparm i.cdnvqm00

Adjusted Wald test

 ( 1)  [dreason1]1.cdnvqm00 = 0
 ( 2)  [dreason1]2.cdnvqm00 = 0
 ( 3)  [dreason1]3.cdnvqm00 = 0
 ( 4)  [dreason1]4.cdnvqm00 = 0
 ( 5)  [dreason1]5.cdnvqm00 = 0
 ( 6)  [dreason1]95.cdnvqm00 = 0

       F(  6,   384) =   52.11
            Prob > F =    0.0000

. testparm i.cdrelp00

Adjusted Wald test

 ( 1)  [dreason1]1.cdrelp00 = 0
 ( 2)  [dreason1]2.cdrelp00 = 0

       F(  2,   388) =    1.67
            Prob > F =    0.1896

. testparm i.cdmwrk00

Adjusted Wald test

 ( 1)  [dreason1]1.cdmwrk00 = 0

       F(  1,   389) =    3.91
            Prob > F =    0.0487
```

8 Country specific analyses

So far we have been analysing data for the whole of the UK. Let's shift to a one country only analysis using Scotland as an example. We will first use the `preserve` command to store the current dataset, then we will use the `keep` command to retain cases in Scotland (at sweep 1) and then set the data by changing the weight to **dovwt1**. We will then analyse the data and after running our model `restore` the data to its original 19244 cases in the active dataset.

```
. preserve

. keep if country==3
(16908 observations deleted)


. svyset sptn00 [pweight=dovwt1], strata(pttype2) fpc(Nh2)

      pweight: dovwt1
          VCE: linearized
  Single unit: missing
     Strata 1: pttype2
         SU 1: sptn00
        FPC 1: Nh2

. * selecting a reference age group e.t.c
.
. svy:logit dreason1  ib4.cdgmai00 ib96.cdnvqm00 cbmin3 ib1.creason1 ib3.cdrelp
> 00 ib2.cdmwrk00
(running logit on estimation sample)

Survey: Logistic regression
```

```
Number of strata   =         2                Number of obs      =      1527
Number of PSUs     =        62                Population size    =  1513.276
                                              Design df          =        60
                                              F(  13,     48)    =     13.46
                                              Prob > F           =    0.0000

------------------------------------------------------------------------------
             |             Linearized
    dreason1 |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
    cdgmai00 |
          2  |  -.6224473   .2081793    -2.99   0.004    -1.038868   -.2060268
          3  |  -.2089541   .1502767    -1.39   0.170    -.5095522     .091644
             |
    cdnvqm00 |
          1  |   .0929641   .4552641     0.20   0.839    -.8176997    1.003628
          2  |   .4043576   .2684337     1.51   0.137    -.1325897    .9413049
          3  |   .6892166   .3175955     2.17   0.034      .053931    1.324502
          4  |    1.23882   .2668383     4.64   0.000     .7050635    1.772576
          5  |   1.386433   .2763046     5.02   0.000     .8337415    1.939125
         95  |  -.1810645   .5721976    -0.32   0.753     -1.32563     .963501
             |
      cbmin3 |  -.0426083   .0299436    -1.42   0.160    -.1025044    .0172877
  0.creason1 |  -1.224552   .1266301    -9.67   0.000     -1.47785   -.9712541
             |
    cdrelp00 |
          1  |  -.2754715   .1886051    -1.46   0.149    -.6527379    .1017949
          2  |  -.4995988   .2148613    -2.33   0.023    -.9293854   -.0698122
             |
  1.cdmwrk00 |   -.134159   .1462681    -0.92   0.363    -.4267388    .1584208
       _cons |   .7364158   .5830483     1.26   0.211    -.4298544    1.902686
------------------------------------------------------------------------------

. * testing whether variable as a whole significant in the model
. testparm i.cdgmai*

Adjusted Wald test

 ( 1)  [dreason1]2.cdgmai00 = 0
 ( 2)  [dreason1]3.cdgmai00 = 0

       F(  2,    59) =    4.40
            Prob > F =    0.0166

. testparm i.cdnvqm00*

Adjusted Wald test

 ( 1)  [dreason1]1.cdnvqm00 = 0
 ( 2)  [dreason1]2.cdnvqm00 = 0
 ( 3)  [dreason1]3.cdnvqm00 = 0
 ( 4)  [dreason1]4.cdnvqm00 = 0
 ( 5)  [dreason1]5.cdnvqm00 = 0
 ( 6)  [dreason1]95.cdnvqm00 = 0

       F(  6,    55) =   12.23
            Prob > F =    0.0000

. testparm c.cbmin3

Adjusted Wald test

 ( 1)  [dreason1]cbmin3 = 0

       F(  1,    60) =    2.02
            Prob > F =    0.1599

. testparm i.creason*

Adjusted Wald test

 ( 1)  [dreason1]0.creason1 = 0

       F(  1,    60) =   93.51
            Prob > F =    0.0000

. testparm i.cdrelp*
```

```
Adjusted Wald test
 ( 1)  [dreason1]1.cdrelp00 = 0
 ( 2)  [dreason1]2.cdrelp00 = 0
       F(  2,    59) =    2.66
            Prob > F =    0.0783
. testparm i.cdmwrk*
Adjusted Wald test
 ( 1)  [dreason1]1.cdmwrk00 = 0
       F(  1,    60) =    0.84
            Prob > F =    0.3627
. restore
```

9 Sub group analyses

Sub group analyses require a very careful approach in how `Stata` estimation commands are issued. To demonstrate how this is done, let's repeat the healthy diet analysis at sweep 4 but restrict the analysis to main respondent parents who smoked at MCS 4.

9.1 A case of one sub group: Smoking parents

We first have to create a variable 0/1 to identify main respondent parents who smoked before we analyse the data. Take note of how the regression command is written. We also need to remember to reset the survey commands to use `dovwt2`, as we had it set to `dovwt1` for the previous country-specific analysis.

```
. generate dsmoking=0 if daoutc00==1
(5387 missing values generated)
. foreach var of varlist dmsmus0*{
  2. replace dsmoking=1 if inlist( `var´,2,3,4,5,6,95)
  3. }
(3721 real changes made)
(0 real changes made)
(0 real changes made)
(0 real changes made)
(0 real changes made)
. label define dsmoking 0 "Non Smoker" 1 "Smoker"
. label values dsmoking dsmoking
. label var dsmoking "S4 DV Smoking status"

. xi:svy,subpop(dsmoking):logit dreason1  ib4.cdgmai00   ib96.cdnvqm00 cbmin3 i
> b1.creason1  ib2.dhcsexa0
(running logit on estimation sample)
Survey: Logistic regression
Number of strata   =         9                 Number of obs      =      13580
Number of PSUs     =       398                 Population size    = 13537.628
                                               Subpop. no. of obs =       3444
                                               Subpop. size       =  3624.108
                                               Design df          =        389
                                               F(  11,    379)    =      21.28
                                               Prob > F           =     0.0000
```

dreason1	Coef.	Linearized Std. Err.	t	P>|t|	[95% Conf. Interval]	
cdgmai00						
2	-.2534616	.1489121	-1.70	0.090	-.5462348	.0393116
3	-.0685646	.1519179	-0.45	0.652	-.3672474	.2301183
cdnvqm00						
1	.1314756	.1716974	0.77	0.444	-.2060955	.4690467
2	.5334646	.146616	3.64	0.000	.2452057	.8217235
3	.7496407	.162587	4.61	0.000	.4299815	1.0693
4	1.007501	.1547579	6.51	0.000	.703235	1.311768
5	1.187366	.2375777	5.00	0.000	.7202687	1.654463
95	-.1577631	.3649066	-0.43	0.666	-.8751991	.5596729
cbmin3	.0542649	.0263425	2.06	0.040	.0024734	.1060564
0.creason1	-1.000716	.0894952	-11.18	0.000	-1.176671	-.8247611
1.dhcsexa0	-.0716753	.0840008	-0.85	0.394	-.2368276	.093477
_cons	-1.423169	.4611524	-3.09	0.002	-2.329832	-.5165062

```
.
. testparm i.cdgmai*

Adjusted Wald test

 ( 1)  [dreason1]2.cdgmai00 = 0
 ( 2)  [dreason1]3.cdgmai00 = 0

       F(  2,   388) =    2.27
            Prob > F =    0.1047

.
. testparm i.cdnvqm00*

Adjusted Wald test

 ( 1)  [dreason1]1.cdnvqm00 = 0
 ( 2)  [dreason1]2.cdnvqm00 = 0
 ( 3)  [dreason1]3.cdnvqm00 = 0
 ( 4)  [dreason1]4.cdnvqm00 = 0
 ( 5)  [dreason1]5.cdnvqm00 = 0
 ( 6)  [dreason1]95.cdnvqm00 = 0

       F(  6,   384) =   11.48
            Prob > F =    0.0000

. testparm  c.cbmin3

Adjusted Wald test

 ( 1)  [dreason1]cbmin3 = 0

       F(  1,   389) =    4.24
            Prob > F =    0.0401

. testparm i.creason*

Adjusted Wald test

 ( 1)  [dreason1]0.creason1 = 0

       F(  1,   389) =  125.03
            Prob > F =    0.0000

. testparm i.dhcsexa*

Adjusted Wald test

 ( 1)  [dreason1]1.dhcsexa0 = 0

       F(  1,   389) =    0.73
            Prob > F =    0.3940
```

9.2 A case of two sub groups: Smoking parents of daughters

Why not just use `if` **dsmoking**==1? Try it and see what you get! Keep an eye on the **Population size** and **Subpop. size** in your output.

You can use `if` correctly with the sub-population command as shown below. The analysis now is on main respondent parents who smoked at MCS 4 and have a cohort member daughter.

```
. svy , subpop(dsmoking if dhcsexa0==2):logit dreason1  ib4.cdgmai00   ib96.cdn
> vqm00 cbmin3 ib1.creason1
(running logit on estimation sample)

Survey: Logistic regression

Number of strata   =         9                 Number of obs      =     13725
Number of PSUs     =       398                 Population size    = 13711.963
                                               Subpop. no. of obs =      1641
                                               Subpop. size       =  1694.042
                                               Design df          =       389
                                               F(  10,    380)    =     12.46
                                               Prob > F           =    0.0000

------------------------------------------------------------------------------
             |             Linearized
    dreason1 |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
    cdgmai00 |
          2  |  -.2582754   .2125007    -1.22   0.225    -.6760689    .1595182
          3  |  -.1009425   .2243948    -0.45   0.653    -.5421209    .3402359
             |
    cdnvqm00 |
          1  |  -.0044342    .234091    -0.02   0.985    -.4646761    .4558077
          2  |   .3199584   .1700085     1.88   0.061    -.0142921    .6542089
          3  |   .3018655   .2348921     1.29   0.200    -.1599515    .7636825
          4  |   .7405968   .2259109     3.28   0.001     .2964377    1.184756
          5  |   1.036269   .3387527     3.06   0.002     .3702534    1.702284
         95  |  -1.971469    .607662    -3.24   0.001    -3.166182   -.7767565
             |
      cbmin3 |   .0913746   .0317786     2.88   0.004     .0288954    .1538539
  0.creason1 |  -1.046339   .1308719    -8.00   0.000    -1.303644   -.7890343
       _cons |  -1.770171    .550442    -3.22   0.001    -2.852384    -.687957
------------------------------------------------------------------------------

. testparm i.cdgmai*

Adjusted Wald test

 ( 1)  [dreason1]2.cdgmai00 = 0
 ( 2)  [dreason1]3.cdgmai00 = 0

       F(  2,   388) =    0.94
            Prob > F =    0.3911

.
. testparm i.cdnvqm00*

Adjusted Wald test

 ( 1)  [dreason1]1.cdnvqm00 = 0
 ( 2)  [dreason1]2.cdnvqm00 = 0
 ( 3)  [dreason1]3.cdnvqm00 = 0
 ( 4)  [dreason1]4.cdnvqm00 = 0
 ( 5)  [dreason1]5.cdnvqm00 = 0
 ( 6)  [dreason1]95.cdnvqm00 = 0

       F(  6,   384) =    5.98
            Prob > F =    0.0000

. testparm  c.cbmin3
```

```
Adjusted Wald test
 ( 1)  [dreason1]cbmin3 = 0
       F(  1,   389) =    8.27
            Prob > F =    0.0043
. testparm i.creason*
Adjusted Wald test
 ( 1)  [dreason1]0.creason1 = 0
       F(  1,   389) =   63.92
            Prob > F =    0.0000
.
```

10 The effect of unit non-response weights on estimates

Unit non-response is when an MCS family doesn't participate in a particular sweep. To account for unit non-response, weights that are inverses of the predicted probability of participating in a sweep were estimated and combined with the sampling weights. The resulting overall weights are what we have used in all the analyses so far.

For all weight variable names, 1 indicates weights for country-specific analyses and 2 indicates weights for analyses combining all the UK countries.

- aovwt1/2 are the overall weights for sweep 1.
- bovwt1/2 are the overall weights for sweep 2.
- covwt1/2 are the overall weights for sweep 3.
- dovwt1/2 are the overall weights for sweep 4.

The variable names for the sampling weights are weight1 and weight2.

We will now run the same analysis using the overall weight and the sampling weight and look for differences in the results.

The results below are a repeat of an analysis from above, using the sweep 4 overall weight.

```
.
. svyset sptn00 [pweight=dovwt2], strata(pttype2) fpc(Nh2)
      pweight: dovwt2
          VCE: linearized
  Single unit: missing
     Strata 1: pttype2
         SU 1: sptn00
        FPC 1: Nh2
.
```

```
. xi:svy,subpop(dsmoking):logit dreason1  ib4.cdgmai00   ib96.cdnvqm00 cbmin3 i
> b1.creason1  ib2.dhcsexa0
(running logit on estimation sample)

Survey: Logistic regression

Number of strata   =         9                  Number of obs      =      13580
Number of PSUs     =       398                  Population size    = 13537.628
                                                Subpop. no. of obs =       3444
                                                Subpop. size       =  3624.108
                                                Design df          =        389
                                                F(  11,    379)    =      21.28
                                                Prob > F           =     0.0000

------------------------------------------------------------------------------
             |             Linearized
    dreason1 |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
    cdgmai00 |
          2  |  -.2534616   .1489121    -1.70   0.090    -.5462348    .0393116
          3  |  -.0685646   .1519179    -0.45   0.652    -.3672474    .2301183
             |
    cdnvqm00 |
          1  |   .1314756   .1716974     0.77   0.444    -.2060955    .4690467
          2  |   .5334646    .146616     3.64   0.000     .2452057    .8217235
          3  |   .7496407    .162587     4.61   0.000     .4299815      1.0693
          4  |   1.007501   .1547579     6.51   0.000      .703235    1.311768
          5  |   1.187366   .2375777     5.00   0.000     .7202687    1.654463
         95  |  -.1577631   .3649066    -0.43   0.666    -.8751991    .5596729
             |
      cbmin3 |   .0542649   .0263425     2.06   0.040     .0024734    .1060564
  0.creason1 |  -1.000716   .0894952   -11.18   0.000    -1.176671   -.8247611
  1.dhcsexa0 |  -.0716753   .0840008    -0.85   0.394    -.2368276     .093477
       _cons |  -1.423169   .4611524    -3.09   0.002    -2.329832   -.5165062
------------------------------------------------------------------------------

.
. testparm i.cdgmai*

Adjusted Wald test

 ( 1)  [dreason1]2.cdgmai00 = 0
 ( 2)  [dreason1]3.cdgmai00 = 0

       F(  2,   388) =    2.27
            Prob > F =    0.1047

.
. testparm i.cdnvqm00*

Adjusted Wald test

 ( 1)  [dreason1]1.cdnvqm00 = 0
 ( 2)  [dreason1]2.cdnvqm00 = 0
 ( 3)  [dreason1]3.cdnvqm00 = 0
 ( 4)  [dreason1]4.cdnvqm00 = 0
 ( 5)  [dreason1]5.cdnvqm00 = 0
 ( 6)  [dreason1]95.cdnvqm00 = 0

       F(  6,   384) =   11.48
            Prob > F =    0.0000

. testparm  c.cbmin3

Adjusted Wald test

 ( 1)  [dreason1]cbmin3 = 0

       F(  1,   389) =    4.24
            Prob > F =    0.0401

. testparm i.creason*

Adjusted Wald test

 ( 1)  [dreason1]0.creason1 = 0

       F(  1,   389) =  125.03
```

```
           Prob > F =    0.0000
. testparm i.dhcsexa*
Adjusted Wald test
 ( 1)  [dreason1]1.dhcsexa0 = 0
       F(  1,   389) =    0.73
            Prob > F =    0.3940
.
```

And below are the results for the same analysis, but run with the sampling weight.

```
. svyset sptn00 [pweight=weight2], strata(pttype2) fpc(Nh2)
      pweight: weight2
          VCE: linearized
  Single unit: missing
     Strata 1: pttype2
         SU 1: sptn00
        FPC 1: Nh2
.
. svy:logit dreason1  ib4.cdgmai00 ib6.cdm06e00 ib96.cdnvqm00 cbmin3 ib1.creaso
> n1 ib1.csmoking ib2.dhcsexa0
(running logit on estimation sample)
Survey: Logistic regression
Number of strata   =         9                Number of obs      =     12791
Number of PSUs     =       398                Population size    =  13374.64
                                              Design df          =       389
                                              F(  17,    373)    =     65.10
                                              Prob > F           =    0.0000
```

dreason1	Coef.	Linearized Std. Err.	t	P>|t|	[95% Conf.	Interval]
cdgmai00						
2	-.211307	.0770379	-2.74	0.006	-.3627697	-.0598442
3	.0182851	.0610225	0.30	0.765	-.1016902	.1382603
cdm06e00						
1	.6840007	.2219888	3.08	0.002	.2475528	1.120449
2	.9073818	.3289123	2.76	0.006	.2607136	1.55405
3	.5986148	.2860974	2.09	0.037	.0361241	1.161105
4	.3975458	.2450103	1.62	0.105	-.0841643	.879256
5	.7659634	.2726265	2.81	0.005	.2299575	1.301969
cdnvqm00						
1	.1145311	.11545	0.99	0.322	-.112453	.3415151
2	.5250855	.0900058	5.83	0.000	.3481268	.7020441
3	.7618788	.1009432	7.55	0.000	.5634162	.9603414
4	.9635261	.0956209	10.08	0.000	.7755277	1.151524
5	1.087051	.1138406	9.55	0.000	.8632306	1.31087
95	.0460645	.19177	0.24	0.810	-.330971	.4230999
cbmin3	.056783	.0127207	4.46	0.000	.0317732	.0817929
0.creason1	-1.052498	.0482946	-21.79	0.000	-1.147449	-.9575468
0.csmoking	.1489219	.0522412	2.85	0.005	.0462115	.2516322
1.dhcsexa0	-.0884602	.0426914	-2.07	0.039	-.1723951	-.0045254
_cons	-2.172269	.3196376	-6.80	0.000	-2.800702	-1.543836

```
.
. testparm i.cdgmai*
Adjusted Wald test
```

```
 ( 1)  [dreason1]2.cdgmai00 = 0
 ( 2)  [dreason1]3.cdgmai00 = 0

       F(  2,   388) =    7.59
            Prob > F =    0.0006

. testparm i.cdm06e00*

Adjusted Wald test

 ( 1)  [dreason1]1.cdm06e00 = 0
 ( 2)  [dreason1]2.cdm06e00 = 0
 ( 3)  [dreason1]3.cdm06e00 = 0
 ( 4)  [dreason1]4.cdm06e00 = 0
 ( 5)  [dreason1]5.cdm06e00 = 0

       F(  5,   385) =    2.83
            Prob > F =    0.0159

. testparm i.cdnvqm00*

Adjusted Wald test

 ( 1)  [dreason1]1.cdnvqm00 = 0
 ( 2)  [dreason1]2.cdnvqm00 = 0
 ( 3)  [dreason1]3.cdnvqm00 = 0
 ( 4)  [dreason1]4.cdnvqm00 = 0
 ( 5)  [dreason1]5.cdnvqm00 = 0
 ( 6)  [dreason1]95.cdnvqm00 = 0

       F(  6,   384) =   28.77
            Prob > F =    0.0000

. testparm  c.cbmin3

Adjusted Wald test

 ( 1)  [dreason1]cbmin3 = 0

       F(  1,   389) =   19.93
            Prob > F =    0.0000

. testparm i.creason*

Adjusted Wald test

 ( 1)  [dreason1]0.creason1 = 0

       F(  1,   389) =  474.95
            Prob > F =    0.0000

. testparm i.csmoking*

Adjusted Wald test

 ( 1)  [dreason1]0.csmoking = 0

       F(  1,   389) =    8.13
            Prob > F =    0.0046

. testparm i.dhcsexa*

Adjusted Wald test

 ( 1)  [dreason1]1.dhcsexa0 = 0

       F(  1,   389) =    4.29
            Prob > F =    0.0389

.
```

Compare the two sets of results.

11 Discussion

How would you combine variables from parental interview data (which is what we have been using) and household grid or cohort member level data where there may be more than one record in the dataset per family?

In what situations would you merge:

- household or child level data on parental interview data?
- parental interview on household or child level data ?

12 Conclusion

There are a few issues to remember when analysing MCS data. Some of the issues which were ignored today are:

- we used data from main respondents only
- majority of main respondents are female
- a main respondent at sweep 4 might different from sweep 3
- there are other possible predictors not in the parental interview file