Tools for the analysis
of 'income mobility'
in Stata

Philippe Van Kerm

`philippe.vankerm@ceps.lu`

CEPS/INSTEAD, G.-D. Luxembourg

2nd Dutch Stata Users Meeting, Maastricht – 2002/05/23

1. Income mobility analysis?

- **Quantifying the movement over time of given recipient units through the distribution of economic well-being**

- Recipient units are typically individuals, households or dynasties

- Most often comparing 2 time periods (but not necessarily)

- Well-being usually gauged by income, earnings, or expenditure (but could apply to occupational or educational indicators)

2.   Income mobility analysis? (ctd.)

- A great variety of normative or statistical arguments have been proposed about what *is* mobility

- When is there more of it in A than in B? What is relevant? Face value income? Income shares? Ranks?

- The 'controversy' translates in a large number of measures and indices

- I've written code for computing only a fraction of existing indices

3. Data sets

- Repeated observations on income for a (generally large) number to recipient units

- Unlike the `-xt-` commands, the modules apply to 'wide' data

- Data are usually obtained from large scale surveys:

  - 'complex' sampling scheme (stratified, multistage sampling)...
  - ... and unequal sampling probability
  - The modules allow for weights but the major current caveat of the modules is that no standard error estimates are provided

- E.g. the *Consortium of Household Panels for European Socio-economic Research* (CHER). The CHER project offers a "database for longitudinal household studies harmonizing micro datasets from a variety of independent national panels and from the European Community Household Panel." (See `http://cher.ceps.lu`)

4.   The `incomob` package

- A collection of seven short ado files (`imobking`, `imobshor`, `imobhart`, `imobcdw`, `imobarj`, `imobflds` and `imobfok`)

- Each ado file computes a single mobility index (from a specific family)

- The indices are restricted to '2 periods analyses'

- The generic syntax is: `imobstg` *inc0 inc1 [weight]* [, *options*]

- No standard error estimates are provided but the adopted format makes the commands amenable to tractable resampling-based inference

`incomob` **examples**

```
.  use incalllu.dta , clear

.    imobfok eqinc98 eqinc99 [aw=wght98]
Fields-Ok (Economica 1999) [logs -- non-directional]:  0.157

.    imobfok eqinc98 eqinc99 [aw=wght98] , natural f(%6.0f)
Fields-Ok (JET 1996) [natural unit -- non-directional]: 161935

.    bysort agr98: imobfok eqinc98 eqinc99 [aw=wght98]


_____

-> agr98 = 0
Fields-Ok (Economica 1999) [logs -- non-directional]:  0.230


_____

-> agr98 = 30
Fields-Ok (Economica 1999) [logs -- non-directional]:  0.152


_____

-> agr98 = 60
Fields-Ok (Economica 1999) [logs -- non-directional]:  0.102
```

```
.    imobcdw eqinc98 eqinc99 [aw=wght98]
Chakravarty-Dutta-Weymark (1985) measure (GE(0)):  0.010

.    bysort agr98: imobcdw eqinc98 eqinc99 [aw=wght98]


_____
-> agr98 = 0
Chakravarty-Dutta-Weymark (1985) measure (GE(0)):  0.015


_____
-> agr98 = 30
Chakravarty-Dutta-Weymark (1985) measure (GE(0)):  0.007


_____
-> agr98 = 60
Chakravarty-Dutta-Weymark (1985) measure (GE(0)):  0.011
```

A side-effect: _gclsort

- _gclsort is an egen function to sort a single variable in a dataset
- sort in increasing order or according to a specified key variable
- (needed for imobking)
- available on the SSC-IDEAS archive

5.   The `shormob` and `fokmob` ado-files

- `shormob` and `fokmob` are two other commands for estimation for the Shorrocks (JET 1978) and Fields-Ok (JET 1996) families of indices

- These measures are available in `incomob` (as `imobshor` and `imobfok`) but added functionality are provided with `shormob` and `fokmob`, i.e. more results are computed and displayed by default

- The other-side of the coin is that the commands tend to be slower (and thence less amenable to resampling-based inference)

- `shormob` allows any number of time periods (i.e. more than 2)

fokmob examples

```
.    fokmob eqinc98 eqinc99 [aw=wght98]

Fields & Ok [1996] 'absolute' mobility indices:

        Total Absolute Mobility:   9.54e+08
                    6.65e+08 due to transfer
                    2.89e+08 due to growth
        Per capita Mobility:       161934.57
                    112902.48 due to transfer
                    49032.08 due to growth
        Percentage Mobility:       16.87%
                    11.76% due to transfer
                     5.11% due to growth


.    fokmob eqinc98 eqinc99 [aw=wght98] , log

Fields & Ok [1999] 'relative' mobility indices:

        Relative Mobility:             0.157
                      0.105 due to transfer
                      0.052 due to growth
```

shormob examples

```
.  shormob eqinc95 eqinc96 eqinc97 eqinc98 eqinc99 eqinc00 /*
     */ [aw=wght95] , ge(-1 0 1 2) atkinson (1 2 3)

Shorrocks [1978]'s mobility index:

  Using GE(-1)         inequality index : 0.19108
  Using GE(0)          inequality index : 0.16196
  Using GE(1)          inequality index : 0.17187
  Using GE(2)          inequality index : 0.22437
  Using Atkinson(1)    inequality index : 0.15496
  Using Atkinson(2)    inequality index : 0.16176
  Using Atkinson(3)    inequality index : 0.20375

.    shormob eqinc95 eqinc96 eqinc97 eqinc98 eqinc99 eqinc00 /*
        */ [aw=wght95] , plot ge(0 1) /*
        */ c(ll[-]) ... and other graph options ...
```

Luxembourg 95-00

Hungary 92-97

UK 91-96

UK 94-99

6. **matrxmob**

- Another type of approach: the transition matrix

- `matrxmob` estimates transition matrices and provides some summary measures

7.   On the 'to do' list

- Statistical inference (taking complex survey design into account à la $\texttt{svy}$ commands)

- Many other measures have been advocated and could be compiled

- Providing functionality for use in 'long' format?