Dynamic Analysis of Work-Life Histories: Manual

This manual explains and advises on use of code to analyse dynamic features of work-life histories contained in the LAMAFLO GitHub.

What dynamic analysis does the code enable?

- Creates a transition variable (capturing current and previous state) and date of transition (current spell start), at each spell start date. ("Transition" consists of remaining in the same state if there is no change in status.)
- Spell durations of each spell. (Spell duration is measured at the end of each spell, so the relevant date is the spell end date.)
- Durations to date in each status type. (Durations are cumulatively measured up the end of each spell, so the relevant date is the spell end date.)
- Counts over work-life history to date of spells in user-chosen statuses. The counts include the current spell, so relate to spells that have all been completed at each spell end date. Between the spell start and spell end date, the count is correct but includes the current incomplete spell.
- Counts over work-life history to date of transitions, for each transition type. Th is cumulative measure of transitions is valid at the current spell start date.
- Conversion of data organised by pidp-spell to data organised by pidp-month. This monthly data is available in either 'wide' or 'long' form. The code as written creates labour force status at monthly frequency, for the user-specified states and over a user-specified time interval (using optional age restrictions). The code could be adapted to collect other variables in monthly form.
- Creation of labour market flows based on user-specified states. Defaults create quarterly flows.

Defining states

The user is asked to specify the states they wish to analyse and the definitions of those states in terms of the underlying values of the original `raw' data's status variable.

The 'raw' data used for the dynamic analysis must contain a variable describing the status (state, activity) of the individual. In LW-based data, this variable is named `Status`. For default settings in the updated "`Launch programs.do`", `Status` takes the values shown in the second column of Table 1 (values taken from the `jbstat` variable in BHPS and UKHLS data). Column 3 shows a possible classification for dynamic analysis. Dynamic analysis is generally best with a small well-defined status, and column 3 uses a conventional classification of categories into three states (employment, unemployment, nonparticipation).

Table 1: Tabulation of Status variable from Merged Dataset (using default settings) with an example of a possible definition of states for dynamic analysis

Economic activity	Code	Classification for 3-state dynamic analysis	Freq.	Percent
Self-Employed	1	E	30,758	5.88
Paid Employment	2	E	240,992	46.05
Unemployed	3	U	40,789	7.79
Retired	4	N	34,343	6.56
Maternity Leave	5	E	8,942	1.71
Family Carer or Homemaker	6	N	34,529	6.60
Full-Time Student	7	N	104,858	20.04
Sick or Disabled	8	N	12,358	2.36
Govt Training Scheme	9	E	3,370	0.64
Unpaid, Family Business	10	E	216	0.04
Apprenticeship	11	E	378	0.07
Doing Something Else	97	N	9,318	1.78
Paid Work	100	E	774	0.15
Something Else	101	N	24	0.00
National Service/War Service	103	E	1,686	0.32
Total			523,335	100.00

Data organisation by pidp-Spell

The code works with `raw' data initially organised by pidp-Spell. Any final dataset resulting from analysis based on code by Liam Wright (2020) would be suitable. That code produces a final dataset "Merged Dataset.dta", which is the default name for the data file to be analysed via the overarching dynamic analysis Stata .do file "`Launch Program - Dynamics.do`".

User choices in "`Launch Program_Dynamics.do`"

- Users need to specify the names of the variables in their `raw' dataset that measure:
  - Status
  - Spell start date
  - Spell end date
- Users need to define the states they wish to analyse dynamically:
  - Definitions in terms of values of the `raw' data status variable (see Table 1 for an example).
  - Names for the states to be analysed (e.g. "E", "U", "N").
- Users can choose to restrict analysis to a particular age range by setting minimum and maximum ages.
- Users wanting to create flows can specify the date at which they want flows data to start.

Time taken for the analysis

The dynamic analyses are fast, apart from the (optional) creation of flows.

Flows creation takes a substantial amount of time. The conversion to pidp-month data is slow as information has to be allocated to each month for each individual. For example, taking individuals in the "Merged Dataset.dta" (default settings) over the 386 months from January 1990 to March 2022, there are over 31 million observations. The length of time to create flows will depend on set-up, but it is normal for it to take 1.5 to 2 hours.