Using panelstat to compute statistics for panel data

Marta Silva (Banco de Portugal)

4th Stata Users Group Meeting

15/09/2017

Panel Data

- Several individual units (workers, firms, regions, ...) observed over time.
- Increasing trend in google searches using the expression **'stata+"panel data"'**.

Source: Google Trends

Panel Data

- Understanding the structure of the data is crucial
- It is important to know about:
  - patterns
  - gaps
  - flows
  - statistics along panelvar and timevar dimension
  - potential miscoding and strange absolute/relative changes
  - ...
- So far, doing all this requires some programming...

Panelstat

- User-written command by Paulo Guimarães (Banco de Portugal, FEP)
- This command analyzes a panel data set and produces a full characterization of the panel structure
- It is implemented for a typical panel and requires both a panel variable and a time variable
- The options that were added reflect particular needs felt by the restricted group of users at BPlim - the Microdata Research Laboratory of Banco de Portugal - who use it on a regular basis

Syntax

panelstat *panelvar timevar* [if] [in], [CONT FORCE1 FAST GAPS RUNS PATTERN DEMO TABOVERT(varlist) WIV(varlist, keep)] WTV(varlist, keep) ABS(varlist, keep) REL(varlist, keep) QUANTR(varlist, keep rel) FLOWS(varlist) TRANS(varlist) CHECKID(var) MISCODE(stud) DEMOBY(var, keep)]

A simple example using nlswork.dta

panelstat idcode year

```
*****************************************************
Analyzing http://www.stata-press.com/data/r14/nlswork.dta
*****************************************************

*****************************************************
There are 28534 time x individuals observations
There are 4711 unique individuals
Time values range from 68 to 88
Maximum time range is 21
The average number of periods per individual is 6.056888134154107
The level of completeness is 28.84%(100% is a fully balanced panel)
Average number of gaps per individual is 2.7450647
Average gap size is 1.8427931
Largest gap is 19
*****************************************************

*****************************************************
Distribution of number of observations per individual
*****************************************************
```

Observ per individual	Freq.	Percent	Cum.
1	547	11.61	11.61
2	498	10.57	22.18
3	484	10.27	32.46
4	411	8.72	41.18
5	421	8.94	50.12
6	398	8.45	58.57
7	345	7.32	65.89
8	323	6.86	72.74
9	302	6.41	79.16
10	270	5.73	84.89
11	202	4.29	89.17
12	158	3.35	92.53
13	147	3.12	95.65
14	119	2.53	98.17
15	86	1.83	100.00
Total	4,711	100.00	

A simple example using nlswork.dta (cont)

```
*****************************************************************
Number of individuals per time unit
*****************************************************************
```

interview year	Freq.	Percent	Cum.
68	1,375	4.82	4.82
69	1,232	4.32	9.14
70	1,686	5.91	15.05
71	1,851	6.49	21.53
72	1,693	5.93	27.47
73	1,981	6.94	34.41
75	2,141	7.50	41.91
77	2,171	7.61	49.52
78	1,964	6.88	56.40
80	1,847	6.47	62.88
82	2,085	7.31	70.18
83	1,987	6.96	77.15
85	2,085	7.31	84.45
87	2,164	7.58	92.04
88	2,272	7.96	100.00
Total	28,534	100.00	

General Options

CONT	ignores a time gap common to all individuals
FORCE1	keeps only one observation per panelvar x timevar pair
FORCE2	drops all duplicate observations
FAST	accelerates the computations by using ftools (mata)

Options - Basic Descriptives

The following options perform some basic descriptives to get to know the panel structure:

GAPS	characterizes the (temporal) gap structure
RUNS	provides information on a sequence of consecutive values for the same individual
PATTERN	describes the most common patterns in the data
DEMO	characterizes the flows over consecutive time periods
ALL	GAPS + RUNS + PATTERN + DEMO

Gaps (GAPS): Example using nlswork.dta

panelstat idcode year, gaps keepmaxgap(max_gap) keepngaps(ngaps) cont fast nosum

```
*****************************************************
Size of time gap vs number of gaps per individual
*****************************************************
```

Size of time gaps	Number of gaps per individual 1	2	3	4	5	Total
1	821	805	386	69	12	2,093
2	224	270	143	34	2	673
3	133	126	73	8	0	340
4	102	89	32	4	0	227
5	91	62	12	1	1	167
6	70	41	5	0	0	116
7	44	20	3	0	0	67
8	32	17	0	0	0	49
9	23	5	0	0	0	28
10	9	5	0	0	0	14
11	10	2	0	0	0	12
12	8	0	0	0	0	8
13	2	0	0	0	0	2
Total	1,569	1,442	654	116	15	3,796

Complete runs (RUNS): Example using nlswork.dta (cont)

panelstat idcode year, runs fast nosum cont

```
**************************************************
Distribution of complete runs by size
**************************************************
```

Length of run	Freq.	Percent	Cum.
1	3,001	35.28	35.28
2	1,635	19.22	54.50
3	1,113	13.08	67.58
4	674	7.92	75.50
5	523	6.15	81.65
6	402	4.73	86.38
7	256	3.01	89.39
8	227	2.67	92.05
9	188	2.21	94.26
10	131	1.54	95.80
11	85	1.00	96.80
12	80	0.94	97.74
13	78	0.92	98.66
14	28	0.33	98.99
15	86	1.01	100.00
Total	8,507	100.00	

Patterns (PATTERN): Example using nlswork.dta (cont)

panelstat idcode year, pattern fast nosum cont

```
**************************************************
Top 10 patterns in the data
**************************************************

     +-----------------------------+
     |          Pattern   Frequency |
     |-----------------------------|
  1. | 100000000000000         136 |
  2. | 000000000000001         114 |
  3. | 000000000000111          89 |
  4. | 000000000000011          87 |
  5. | 111111111111111          86 |
     |-----------------------------|
  6. | 000000000011111          61 |
  7. | 110000000000000          56 |
  8. | 000000111111111          54 |
  9. | 000000000001111          54 |
 10. | 000000011111111          49 |
     +-----------------------------+

Note: 1 if observation is in the dataset; 0 otherwise
```

Flows (DEMO): Example using nlswork.dta (cont)

panelstat idcode year, demo fast nosum cont

```
*****************************************************
Time changes - incumbents, entrants and exits
*****************************************************
```

	period	total	inc1	entry	first	reent	inc2	exit	last	reexit
1.	68	1375	0	1375	1375	0	851	524	136	388
2.	69	1232	851	381	381	0	1001	231	79	152
3.	70	1686	1001	685	476	209	1315	371	93	278
4.	71	1851	1315	536	381	155	1224	627	156	471
5.	72	1693	1224	469	331	138	1411	282	97	185
6.	73	1981	1411	570	257	313	1534	447	132	315
7.	75	2141	1534	607	304	303	1617	524	189	335
8.	77	2171	1617	554	275	279	1625	546	163	383
9.	78	1964	1625	339	134	205	1399	565	199	366
10.	80	1847	1399	448	142	306	1502	345	143	202
11.	82	2085	1502	583	159	424	1647	438	155	283
12.	83	1987	1647	340	126	214	1484	503	239	264
13.	85	2085	1484	601	147	454	1600	485	311	174
14.	87	2164	1600	564	109	455	1817	347	347	0
15.	88	2272	1817	455	114	341	0	2272	2272	0

```
period - time period
total - total number of individuals at period t
inc1 - number of individuals at t that are also present at t-1
entry - number of individuals at t that are not present at t-1
first - number of individuals at t who show up for the first time at t
reent - number of individuals at t that are reentering at period t
inc2 - number of individuals at t that are also present at t+1
exit - number of individuals at t that are not present at t+1
last - number of individuals at t that are not present at any future period
reexit - number of individuals at t that are not present at t+1 but appear in later periods
```

Options - Advanced Descriptives

We can characterize each variable in terms of missing values, range, variation along individuals and/or time

WIV	provides statistics for varlist along the panelvar dimension
WTV	provides statistics for varlist along the timevar dimension

WIV: Example using nlswork.dta

panelstat idcode year, wiv(ind_code, keep) nosum cont fast

```
*****************************************************
Analyzing variable ind_code within idcode
*****************************************************

There are 98.80% nonmissing observations (28193 out of 28534)

For the variable ind_code we have:
        values range from 1 to 12
        1986 complete invariant idcode-observations (42.16%)
        2404 complete variant idcode-observations (51.03%)
        16 completely missing idcode-observations ( 0.34%)
        85 invariant idcode-observations with missing values ( 1.80%)
        220 variant idcode-observations with missing values ( 4.67%)
```

_wiv_ind_code	Freq.	Percent	Cum.
1 complete time-invariant	8,505	29.81	29.81
2 complete time-variant	17,376	60.90	90.70
3 complete missing	17	0.06	90.76
4 time-invariant with miss	670	2.35	93.11
5 time-variant with miss	1,966	6.89	100.00
Total	28,534	100.00	

Options - Advanced Descriptives

Panelstat allows identifying and signalling abnormal absolute and relative changes:

ABS	reports on absolute changes over time for each variable in varlist
REL	reports on relative changes over time for each variable in varlist
QUANTR	computes year to year changes for quantiles of varlist

Relative Changes (REL): Example using nlswork.dta

panelstat idcode year, rel(ln_wage, keep) setrelv(150) cont nosum fast

```
**************************************************
Relative changes over time for ln_wage (relv set to 150)
**************************************************

      _rel_ln_wage |      Freq.     Percent        Cum.
-------------------+-----------------------------------
 1 positive change |     12,077       42.32       42.32
 2 negative change |      7,527       26.38       68.70
       3 no change |        363        1.27       69.98
4 abnormal pos chg |         36        0.13       70.10
5 abnormal neg chg |         24        0.08       70.19
         6 missing |      8,507       29.81      100.00
-------------------+-----------------------------------
             Total |     28,534      100.00
Note: Relative change is calculated relative to the average of x_{t} and x_{t-1}
```

Changes for Quantiles (QUANTR): Example using nlswork.dta

panelstat idcode year, quantr(ln_wage, keep rel) setqtll(10) setqtul(90) cont nosum fast

```
**************************************************
changes (t-1 to t) in the quantiles of ln_wage
**************************************************
```

Time (cont)	Distribution of quantile changes									Total
	1to1	1to2	1to3	2to1	2to2	2to3	3to1	3to2	3to3	
68	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100.00
69	1.87	3.08	0.00	2.60	51.30	2.84	0.16	1.95	5.28	100.00
70	2.85	4.39	0.00	2.19	52.55	2.85	0.06	1.78	5.10	100.00
71	2.97	4.65	0.16	3.35	56.89	3.03	0.16	2.49	5.73	100.00
72	2.84	4.96	0.18	3.43	58.06	3.43	0.06	2.48	5.02	100.00
73	3.89	4.95	0.15	3.99	63.55	2.17	0.15	1.62	6.56	100.00
75	2.29	6.77	0.14	5.65	59.08	3.83	0.28	3.13	4.62	100.00
77	3.18	4.65	0.05	5.39	61.86	2.86	0.32	3.73	5.30	100.00
78	3.41	4.53	0.31	6.16	67.06	2.90	0.05	3.16	5.60	100.00
80	3.95	4.71	0.16	9.31	61.94	3.63	0.22	3.36	5.04	100.00
82	5.08	8.54	0.19	4.80	61.06	3.88	0.10	3.36	5.37	100.00
83	3.93	4.98	0.10	4.98	67.44	2.42	0.20	3.17	6.44	100.00
85	3.65	4.94	0.19	5.42	66.57	3.17	0.05	3.31	5.66	100.00
87	4.34	4.53	0.05	4.99	68.53	3.47	0.09	2.87	6.10	100.00
88	5.28	4.71	0.35	4.31	67.52	3.61	0.13	3.48	5.59	100.00
Total	3.45	4.86	0.14	4.62	59.23	3.01	0.14	2.77	5.28	100.00

Changes for Quantiles (QUANTR): Example using nlswork.dta (cont)

Time (cont)	Distribution of quantile changes .to1	.to2	.to3	Total
68	10.47	79.85	9.67	100.00
69	6.09	23.05	1.79	100.00
70	5.04	21.23	1.96	100.00
71	4.54	15.02	1.03	100.00
72	3.72	14.59	1.24	100.00
73	2.02	9.84	1.11	100.00
75	1.96	11.02	1.21	100.00
77	1.15	9.72	1.80	100.00
78	0.56	5.24	1.02	100.00
80	1.19	5.36	1.14	100.00
82	0.86	6.24	0.53	100.00
83	0.91	4.93	0.50	100.00
85	1.01	5.37	0.67	100.00
87	0.60	4.07	0.37	100.00
88	1.06	3.52	0.44	100.00
Total	2.40	12.68	1.43	100.00

_quantr_ln_wage
Notes:
 quantile 1 defined as values below 10
 quantile 2 defined as values above 10 and below 90
 quantile 3 defined as values above 90

Options - Advanced Descriptives

FLOWS	decomposes the changes on the sum of the time observations for each variable
DEMOBY	checks movements of individuals across units of var
MISCODE	identifies changes between pairs of variables (category miscoding?)
CHECKID	checks whether variable can be used as alternative panelvar

FLOWS: Example using nlswork.dta

panelstat idcode year, flows(ln_wage) nosum cont fast

```
*****************************************************
Time flows for variable ln_wage
*****************************************************
```

	period	ln_wage	chg	c_inc	c_exp	c_cont	c_entry	c_exit	c_inc1	c_inc2
1.	68	1981.871	.	0	0	0	1981.871	.	0	0
2.	69	1886.232	-95.638738	87.20004	125.6389	-38.43882	531.6566	-714.49537	0	0
3.	70	2540.883	654.651	25.78702	86.75755	-60.97053	962.4709	-333.60689	0	0
4.	71	2863.236	322.35277	100.422	176.4238	-76.00177	735.0456	-513.11488	0	0
5.	72	2653.409	-209.82632	64.61437	151.9803	-87.36596	657.6903	-932.13104	0	0
6.	73	3126.905	473.49605	68.03374	142.1779	-74.14416	816.1984	-410.73606	0	0
7.	75	3382.492	255.58656	58.49344	213.4937	-155.0003	887.2773	-690.18419	0	0
8.	77	3601.283	218.79105	100.6323	215.9816	-115.3493	900.9496	-782.79088	0	0
9.	78	3370.793	-230.49016	68.34719	172.3399	-103.9927	572.7049	-871.54221	0	0
10.	80	3194.891	-175.90186	42.05785	162.1536	-120.0958	727.3137	-945.2734	0	0
11.	82	3599.611	404.71956	57.79589	178.4448	-120.6489	942.9634	-596.0397	0	0
12.	83	3513.564	-86.046415	47.68224	168.1555	-120.4733	573.738	-707.46661	0	0
13.	85	3819.391	305.82683	103.9355	202.5548	-98.61931	1046.263	-844.37219	0	0
14.	87	3986.022	166.63086	86.60191	193.1064	-106.5045	929.6242	-849.59529	0	0
15.	88	4271.217	285.19514	125.6112	254.0469	-128.4357	782.9087	-623.32471	0	0

```
Notes:
ln_wage - total sum of ln_wage at period t
chg - sum of ln_wage at t minus t-1
c_inc - changes from individuals present at t and at t-1 of which:
    c_exp - positive changes (expansions) from individuals present at t and at t-1
    c_cont - negative changes (contractions) from individuals present at t and at t-1
c_entry - change resulting from entry (present at t but not at t-1)
c_exit - change resulting from exits (present at t-1 but not at t)
c_inc1 - change from individuals present at t and t-1 but with missing data at t-1
c_inc2 - change from individuals present at t and t-1 but with missing data at t
ln_wage[t]=ln_wage[t-1]+chg, chg=c_inc+c_entry+c_exit+c_inc1+c_inc2, c_inc=c_exp+c_cont
```

DEMOBY: Example using nlswork.dta

panelstat idcode year, demoby(msp, keep) nosum cont fast

```
**************************************************
Decomposition of changes across msp over time
**************************************************
```

	period	total	first	last	sing	stay	mover	return
1.	68	1375	1375	136	136	0	0	0
2.	69	1232	381	79	23	716	135	0
3.	70	1686	476	94	24	995	197	18
4.	71	1851	381	156	42	1229	208	33
5.	72	1693	331	97	25	1156	170	36
6.	73	1981	257	133	26	1495	185	44
7.	75	2141	304	189	24	1504	256	77
8.	77	2166	275	163	25	1603	187	101
9.	78	1953	132	197	12	1632	113	76
10.	80	1847	144	143	11	1461	136	106
11.	82	2085	159	155	14	1706	106	114
12.	83	1987	126	239	20	1741	65	55
13.	85	2085	147	311	31	1715	122	101
14.	87	2164	109	347	22	1846	101	108
15.	88	2272	114	2272	114	2004	59	95

```
period - time period
total - total number of individuals at period t
first - number of individuals at t that show up for the first time
last - number of individuals at t that show up for the last time
singleton - number of individuals at t that show only at one period (singletons)
stayer - number of individuals at t that were present at the same msp unit since their last observation
mover - number of individuals at t that were present at a new msp unit
return - number of individuals at t that returned to a msp unit
```

Options - Advanced Descriptives

For categorical variables, two additional options are available:

TABOVERT	creates a tabulation of variables in varlist over time
TRANS	calculates the share of individuals that have the same movement across categories of varlist from t-1 to t

Tab over time (TABOVERT): Example using nlswork.dta

panelstat idcode year, tabovert(occ_code) nosum cont fast

Tabulation of occ_code over time

	occ_code	n1	n2	n3	n4	n5	n6	n7	n8	n9	n10	n11	n12	n13	n14	n15
1.	1	78	72	108	121	112	155	194	249	227	225	276	280	301	291	319
2.	2	12	12	16	30	36	42	72	86	87	104	137	148	206	237	269
3.	3	580	556	716	798	739	827	846	824	730	678	752	717	711	767	733
4.	4	60	60	87	92	94	105	99	99	93	81	101	84	83	81	104
5.	5	20	10	16	18	19	26	27	32	39	33	31	33	35	45	54
6.	6	273	240	319	304	266	329	374	346	289	291	299	259	249	223	248
7.	7	57	33	55	71	55	62	28	33	21	18	36	28	29	17	28
8.	8	217	187	286	328	289	332	351	335	301	261	285	273	281	282	292
9.	9	.	.	1	1	.	.	1	1	.	.	.	1	.	.	1
10.	10	9	7	11	11	10	16	10	10	21	5	8	9	5	3	9
11.	11	8	10	11	7	10	12	20	15	11	14	17	12	13	18	16
12.	12	.	.	.	.	.	.	1	.	.	1	.	1	.	2	2
13.	13	47	42	58	69	53	69	110	139	125	124	135	135	168	183	188
14.	.	14	3	2	1	10	6	8	2	20	12	8	7	4	15	9

How to export the results?

- It is possible to export the results to excel for the following options:
  - ALL (GAPS RUNS PATTERN DEMO)
  - WIV
  - WTV
  - FLOWS
  - TABOVERT
- Syntax:

panelstat *panelvar timevar* [if] [in], OPTION **excel(example.xlsx)**

Performance and further work

Program: Stata-MP 14.2 (Single-user 8-core)
OS: 64-bit Windows
Processor: Intel(R) Core(TM) i5-6300U CPU @ 2.40GHz
Installed Memory (RAM): 12.0 GB

Option	-	GAPS	RUNS	DEMO	PATTERN	WIV
Time (sec)	0.19	0.33	0.22	0.28	0.42	0.33

Option	WTV	REL	QUANTR	FLOWS	DEMOBY	TABOVERT
Time (sec)	0.31	0.23	0.47	0.30	0.44	0.27

With bigger data sets...

- It takes longer to run with big data sets but it is still feasible.

Number ID	Time periods	Observations	Time (sec)	
			No options	ALL (GAPS, RUNS, DEMO, PATTERN)
9,799	50	465,336	2.84	5.42
113,326	50	5,383,323	25.04	65.47
1,074,208	50	51,026,187	275.19	662.40

Dependencies

- group2hdfe by Paulo Guimarães
- excelcol by Sergiy Radyakin
- sreshape by Kenneth L. Simons
- ftools package by Sergio Correia

Where to get *panelstat*?

net install panelstat,
from("https://github.com/pguimaraes99/panelstat/raw/master/")

Thank you for your attention!