

Automating Episode Splitting: Introducing the `splitting` Command for Stata

Davide Bussi¹

¹Department of Sociology and Social Research, Milano-Bicocca University

`d.bussi@campus.unimib.it`

Presented at the 2025 Italian Stata Conference

September 25, 2025

Introduction

Survival analysis is well supported in Stata through numerous built-in commands

This technique is particularly useful when researchers aim to estimate the effect of time-varying variables. In particular, `stsplit` helps break the time axis into episodes to include time-varying covariates in the analysis

Introduction

When the time axis is split by a polytomous variable, using `stsplit` becomes more demanding, as it requires additional code and varies depending on whether the variable is categorical or ordinal, which increases the risk of errors

To overcome this, I propose the Stata program `splitting`, which standardizes the procedure and allows users to split the time axis on either dichotomous or polytomous variables with a single command

How stsplot works with dichotomous variable (1)

Suppose we want to study how entry into the labour market affects the likelihood of marriage

	id	married	married_cm	sex	birthy	birth_ms	area	educ4	feduc4
1	1	Yes	1053	Men	1958	705	Islands	High School	Middle School
2	2	Yes	1053	Women	1961	735	Islands	Middle School	Primary School
3	3	No	1435	Women	1939	474	Central	Primary School	Primary School
4	4	Yes	1204	Men	1969	835	Northeast	High School	Middle School
5	5	No	1439	Men	2001	1215	Northeast	Middle School	High School
6	6	No	1439	Men	1952	625	Northwest	University	Middle School
7	7	Yes	936	Women	1954	650	Northwest	High School	Middle School
8	8	Yes	1218	Women	1967	812	Northeast	University	High School
9	9	Yes	963	Women	1954	656	Northeast	High School	Primary School
10	10	Yes	963	Men	1955	671	Northeast	Middle School	Primary School
11	11	Yes	1050	Men	1958	706	Northeast	Middle School	Primary School
12	12	Yes	1038	Women	1955	671	Northeast	High School	Middle School
13	13	No	1443	Women	1987	1049	Northeast	High School	Middle School
14	14	No	1444	Men	1989	1075	Northeast	Middle School	Middle School
15	15	No	1439	Men	1960	723	Islands	Middle School	Primary School

How stsplot works with dichotomous variable (2)

First step

The data must be organized so that each individual is associated with the time point at which the episode change occurs

	id	firstjob_y	firstjob_m	firstjob_cm
1	1	1979	9. September	957
2	4	1988	8. August	1064
3	9	1971	7. July	859
4	12	1973	10. October	886
5	16	2006	5. May	1277
6	17	1970	7. July	847
7	19	1988	10. October	1066
8	20	2016	9. September	1401
9	25	1982	3. March	987
10	30	1985	3. March	1023
11	31	1998	1. January	1177
12	32	1982	5. May	989
13	33	2002	7. July	1231
14	34	2008	6. June	1302
15	37	2006	9. September	1281

How stsplits works with dichotomous variable (3)

Second step

The obtained dataset should be merged with the original one, which will be used for estimation

	id	married	married_cm	firstjob_cm	sex	birthy	area	educ4	feduc4
1	1	Yes	1053	957	Men	1958	Islands	High School	Middle School
2	2	Yes	1053	100000	Women	1961	Islands	Middle School	Primary School
3	3	No	1435	100000	Women	1939	Central	Primary School	Primary School
4	4	Yes	1204	1064	Men	1969	Northeast	High School	Middle School
5	5	No	1439	100000	Men	2001	Northeast	Middle School	High School
6	6	No	1439	100000	Men	1952	Northwest	University	Middle School
7	7	Yes	936	100000	Women	1954	Northwest	High School	Middle School
8	8	Yes	1218	100000	Women	1967	Northeast	University	High School
9	9	Yes	963	859	Women	1954	Northeast	High School	Primary School
10	10	Yes	963	100000	Men	1955	Northeast	Middle School	Primary School
11	11	Yes	1050	100000	Men	1958	Northeast	Middle School	Primary School
12	12	Yes	1038	886	Women	1955	Northeast	High School	Middle School
13	13	No	1443	100000	Women	1987	Northeast	High School	Middle School
14	14	No	1444	100000	Men	1989	Northeast	Middle School	Middle School
15	15	No	1439	100000	Men	1960	Islands	Middle School	Primary School

How stsplit works with dichotomous variable (4)

Third step

The `stsplit` command is applied to the final dataset, producing a time axis divided into two segments: before and after the event (if it occurred)

```
stset married_cm, origin(birth_cm) failure(married)  
id(id) exit(time .)  
stsplit job, after(firstjob_cm) at(0)  
recode job (-1 = 0) (0 = 1)
```

How stsplot works with dichotomous variable (5)

	id	married	_d	birth_cm	married_cm	_t	_t0	job	firstjob_cm	sex	birthy	area
1	1	No	0	705	957	252	0	No	957	Men	1958	Islands
2	1	Yes	1	705	1053	348	252	Yes	957	Men	1958	Islands
3	2	Yes	1	735	1053	318	0	No	100000	Women	1961	Islands
4	3	No	0	474	1435	961	0	No	100000	Women	1939	Central
5	4	No	0	835	1064	229	0	No	1064	Men	1969	Northeast
6	4	Yes	1	835	1204	369	229	Yes	1064	Men	1969	Northeast
7	5	No	0	1215	1439	224	0	No	100000	Men	2001	Northeast
8	6	No	0	625	1439	814	0	No	100000	Men	1952	Northwest
9	7	Yes	1	650	936	286	0	No	100000	Women	1954	Northwest
10	8	Yes	1	812	1218	406	0	No	100000	Women	1967	Northeast
11	9	No	0	656	859	203	0	No	859	Women	1954	Northeast
12	9	Yes	1	656	963	307	203	Yes	859	Women	1954	Northeast
13	10	Yes	1	671	963	292	0	No	100000	Men	1955	Northeast
14	11	Yes	1	706	1050	344	0	No	100000	Men	1958	Northeast
15	12	No	0	671	886	215	0	No	886	Women	1955	Northeast
16	12	Yes	1	671	1038	367	215	Yes	886	Women	1955	Northeast

How `stsplot` works with polytomous variable (1)

However, when the time axis must be split according to a polytomous variable, steps one and three become more complex, require more code, and differ slightly depending on whether the polytomous variable is categorical or ordinal

This increases the likelihood of coding errors

How stsplot works with polytomous variable (2)

Suppose that we want to estimate the effect of residential mobility on the likelihood of entering the labor market

	id	area_cm1	area1	area_cm2	area2	area_cm3	area3	area_cm4	area4	area_cm5	area5	area_cm6	area6
1	1	705	5	918	5	1053	5	.	.	.	.	.	.
2	2	735	5	1053	5	.	.	.	.	.	.	.	.
3	3	474	3	685	3	.	.	.	.	.	.	.	.
4	4	835	2	.	.	.	.	.	.	.	.	.	.
5	5	1215	2	.	.	.	.	.	.	.	.	.	.
6	6	625	1	936	1	.	.	.	.	.	.	.	.
7	7	650	1	936	1	.	.	.	.	.	.	.	.
8	8	812	2	1217	2	.	.	.	.	.	.	.	.
9	9	656	2	963	2	1267	2	.	.	.	.	.	.
10	10	671	2	963	2	1267	2	.	.	.	.	.	.
11	11	706	2	922	2	1033	2	1113	2	.	.	.	.
12	12	671	2	1026	2	.	.	.	.	.	.	.	.
13	13	1049	2	.	.	.	.	.	.	.	.	.	.
14	14	1075	2	.	.	.	.	.	.	.	.	.	.
15	15	723	5	789	5	867	5	933	6	945	6	957	6

How stsplits works with polytomous variable (3)

```
foreach var of varlist area_ms* {  
    replace `var' = 10000 if `var'==.  
}  
  
local var "area_ms1 area_ms2 area_ms3 area_ms4 area_ms5  
area_ms6 area_ms7 area_ms8 area_ms9 area_ms10  
area_ms11 area_ms12 area_ms13 area_ms14 area_ms15  
area_ms16 area_ms17"  
  
local level "res1 res2 res3 res4 res5 res6 res7 res8 res9  
res10 res11 res12 res13 res14 res15 res16 res17"  
  
local n : word count `var'  
forvalues i = 1/`n' {  
    local a : word `i' of `var'  
    local b : word `i' of `level'  
    stsplits `b', after(`a') at(0)  
}
```

How stsplits works with polytomous variable (4)

```
foreach var of varlist res1-res17 {  
    recode `var' (-1= 0) (0 = 1)  
}  
gen area_tv = res1 + res2 + res3 + res4 + res5 + res6 +  
    res7 + res8 + res9 + res10 + res11 + res12 + res13 +  
    res14 + res15 + res16 + res17  
gen area_tvold = area_tv  
replace area_tv = area1 if area_tvold==1  
replace area_tv = area2 if area_tvold==2  
...  
replace area_tv = area16 if area_tvold==16  
replace area_tv = area17 if area_tvold==17  
recode area_nascita (0 = 6)  
replace area_tv = area_nascita if area_tvold==0
```

How stsplits works with polytomous variable (5)

	id	employed	_d	birth_cm	firstjob	_t	_t0	area_tv	area_cm1	area_cm2	area_cm3	area_cm4
1	1	No	0	705	918	213	120	Islands	705	918	1053	10000
2	1	Yes	1	705	957	252	213	Islands	705	918	1053	10000
3	2	No	0	735	1053	318	120	Islands	735	1053	10000	10000
4	2	No	0	735	1395	660	318	Islands	735	1053	10000	10000
5	3	No	0	474	685	211	120	Central	474	685	10000	10000
6	3	No	0	474	1134	660	211	Central	474	685	10000	10000
7	4	Yes	1	835	1064	229	120	Northeast	835	10000	10000	10000
8	5	No	0	1215	1439	224	120	Northeast	1215	10000	10000	10000
9	6	No	0	625	936	311	120	Northwest	625	936	10000	10000
10	6	Yes	1	625	962	337	311	Northwest	625	936	10000	10000
11	7	No	0	650	936	286	120	Northwest	650	936	10000	10000
12	7	No	0	650	1310	660	286	Northwest	650	936	10000	10000
13	8	Yes	1	812	1118	306	120	Northeast	812	1217	10000	10000
14	9	Yes	1	656	859	203	120	Northeast	656	963	1267	10000
15	10	Yes	1	671	846	175	120	Northeast	671	963	1267	10000

The `splitting` command (1)

`splitting` aims to standardize the event-splitting procedure, allowing users to divide the time axis by either a dichotomous or a polytomous variable with a single command

Users need to prepare a dataset in long format, including four variables:

1. respondent's ID;
2. the time at which the episode occurs;
3. the state of the episode;
4. the episode number (arranged in ascending order)

The program is built upon the `stsplit` command, so the data must be set using `stset` before running it.

The splitting command (3)

	id	area	area_cm	spell
1	1	Islands	705	1
2	1	Islands	918	2
3	1	Islands	1053	3
4	2	Islands	735	1
5	2	Islands	1053	2
6	3	Central	474	1
7	3	Central	685	2
8	4	Northeast	835	1
9	5	Northeast	1215	1
10	6	Northwest	625	1
11	6	Northwest	936	2
12	7	Northwest	650	1
13	7	Northwest	936	2
14	8	Northeast	812	1
15	8	Northeast	1217	2
16	9	Northeast	656	1
17	9	Northeast	963	2
18	9	Northeast	1267	3
19	10	Northeast	671	1
20	10	Northeast	963	2
21	10	Northeast	1267	3

The splitting command (2)

```
splitting varsplit state spell newvar, VARtype()  
From() ID()
```

- ▶ `varsplit` specifies the variable containing the time point at which the episode should be split
- ▶ `state` refers to the variable indicating the state of each episode for each individual
- ▶ `spell` is the variable representing the episode number associated with `state` and `varsplit`
- ▶ `newvar` is the name of the new variable to be created
- ▶ `VARtype()` is an option that requires users to specify the type of variable used to split the time axis: dummy, categorical, or ordinal
- ▶ `From()` specifies the directory where the prepared dataset in long format is stored
- ▶ `ID()` indicates the variable containing the respondent ID

The splitting command (4)

```
splitting area_cm area spell area_tv,  
vartype(categorical) from(`dataset_area`) id(id)
```

	id	employed	_d	birth_cm	firstjob	_t	_t0	area_tv
1	1	No	0	705	918	213	120	Islands
2	1	Yes	1	705	957	252	213	Islands
3	2	No	0	735	1053	318	120	Islands
4	2	No	0	735	1395	660	318	Islands
5	3	No	0	474	685	211	120	Central
6	3	No	0	474	1134	660	211	Central
7	4	Yes	1	835	1064	229	120	Northeast
8	5	No	0	1215	1439	224	120	Northeast
9	6	No	0	625	936	311	120	Northwest
10	6	Yes	1	625	962	337	311	Northwest
11	7	No	0	650	936	286	120	Northwest
12	7	No	0	650	1310	660	286	Northwest
13	8	Yes	1	812	1118	306	120	Northeast
14	9	Yes	1	656	859	203	120	Northeast
15	10	Yes	1	671	846	175	120	Northeast

**Thank you for your
attention!**