The Stata Journal (2023)
23, Number 1, pp. 281–292 DOI: 10.1177/1536867X231162020

Stata tip 150: When is it appropriate to xtset a panel dataset with panelvar only?

Carlo Lazzaro
Studio di Economia Sanitaria
Milan, Italy
and
School of Pharmacology
Biology and Biotechnologies Department "Lazzaro Spallanzani"
University of Pavia
Pavia, Italy
carlo.lazzaro@tiscalinet.it

1 Introduction

The Stata command `xtset` (see [XT] **xtset**) is the requirement to access the `xt` suite of commands, which was developed to deal with datasets having both a cross-sectional (or N) and a time-series (or T) dimension (that is, panels) (Cameron and Trivedi 2005, 2022; Wooldridge 2020).

A panel dataset can be `xtset` in five ways. One of them allows the panel dataset to be `xtset` via the *panelvar* only:

Syntax 1

`xtset` *panelvar*

The code above tells Stata that the dataset is composed of panels, but the order of the observations belonging to each panel is irrelevant. The remaining four ways to `xtset` the panel dataset require a *timevar* too, with or without some additional options, to tell Stata how frequently observations are collected (for example, every two years):

Syntax 2

`xtset` *panelvar timevar* [`,` *tsoptions*]

2 Why does error r(451) occur?

From 2014, the Stata forum reports 500 queries (keyword: "repeated time values within panel"; last check July 6, 2022) concerning the error `r(451)`, the description of which can be accessed by typing the following from within Stata:


```
. search r(451)
[P]     error . . . . . . . . . . . . . . . . . . . . . . . . .  Return code 451
        invalid values for time variable
        For instance, you specified mytime as a time variable, and
        mytime contains noninteger values.
```

Often, the error `r(451)` occurs because at least one panel in the dataset has two or more observations that share the same date, dates are not detailed enough to allow these observations to coexist, or both.

In the following toy example, **`xtreg, fe`** (see [XT] **xtreg**) is fit to a short panel dataset ($N > T$) composed of six subsidiaries of the Bank of Alfa that settle their mutual transactions in foreign currencies (values in €2021) at fixed time slots during the first two weeks of November 2021 (table 1):

```
. list bank op_type op_amnt eventdate eventdate2 daily_inc, noobs
```

Table 1. Transactions in foreign currencies

bank	op_type	op_amnt	eventdate	eventdate2	daily_inc
Bank of Alfa 1	Exchange from U.K.£ to €	1000	04/11/2021	04/11/2021 14:32	34887.17
Bank of Alfa 1	Exchange from U.K.£ to €	2000	04/11/2021	04/11/2021 17:20	26688.57
Bank of Alfa 1	Exchange from U.S.$ to €	3000	05/11/2021	05/11/2021 11:20	13664.63
Bank of Alfa 1	Exchange from U.K.£ to €	4000	05/11/2021	05/11/2021 18:36	2855687
Bank of Alfa 1	Exchange from U.K.£ to €	5000	08/11/2021	08/11/2021 10:08	86893.33
Bank of Alfa 2	Exchange from U.S.$ to €	6000	04/11/2021	04/11/2021 14:32	35085.49
Bank of Alfa 2	Exchange from U.S.$ to €	7000	04/11/2021	04/11/2021 17:20	7110509
Bank of Alfa 2	Exchange from U.K.£ to €	8000	05/11/2021	05/11/2021 11:20	32336.79
Bank of Alfa 2	Exchange from U.S.$ to €	9000	05/11/2021	05/11/2021 18:36	55510.32
Bank of Alfa 2	Exchange from U.K.£ to €	10000	08/11/2021	08/11/2021 10:08	87599.10
Bank of Alfa 3	Exchange from U.S.$ to €	2000	04/11/2021	04/11/2021 14:32	20470.95
Bank of Alfa 3	Exchange from U.K.£ to €	4000	04/11/2021	04/11/2021 17:20	89275.87
Bank of Alfa 3	Exchange from U.S.$ to €	6000	05/11/2021	05/11/2021 11:20	58446.58
Bank of Alfa 3	Exchange from U.K.£ to €	8000	05/11/2021	05/11/2021 18:36	36977.91
Bank of Alfa 3	Exchange from U.K.£ to €	10000	08/11/2021	08/11/2021 10:08	85063.09
Bank of Alfa 4	Exchange from U.S.$ to €	12000	04/11/2021	04/11/2021 14:32	39138.19
Bank of Alfa 4	Exchange from U.S.$ to €	14000	04/11/2021	04/11/2021 17:20	11966.13
Bank of Alfa 4	Exchange from U.K.£ to €	16000	05/11/2021	05/11/2021 11:20	75424.34
Bank of Alfa 4	Exchange from U.K.£ to €	18000	05/11/2021	05/11/2021 18:36	69502.34
Bank of Alfa 4	Exchange from U.S.$ to €	20000	08/11/2021	08/11/2021 10:08	68661.52
Bank of Alfa 5	Exchange from U.K.£ to €	3000	04/11/2021	04/11/2021 14:32	93193.46
Bank of Alfa 5	Exchange from U.S.$ to €	4000	04/11/2021	04/11/2021 17:20	45488.82
Bank of Alfa 5	Exchange from U.S.$ to €	5000	05/11/2021	05/11/2021 11:20	6740.11
Bank of Alfa 5	Exchange from U.K.£ to €	6000	05/11/2021	05/11/2021 18:36	33798.89
Bank of Alfa 5	Exchange from U.K.£ to €	7000	08/11/2021	08/11/2021 10:08	97488.48
Bank of Alfa 6	Exchange from U.S.$ to €	12000	04/11/2021	04/11/2021 14:32	72643.84
Bank of Alfa 6	Exchange from U.K.£ to €	14000	04/11/2021	04/11/2021 17:20	4541.51
Bank of Alfa 6	Exchange from U.S.$ to €	16000	05/11/2021	05/11/2021 11:20	74596.66
Bank of Alfa 6	Exchange from U.K.£ to €	18000	05/11/2021	05/11/2021 18:36	49612.59
Bank of Alfa 6	Exchange from U.S.$ to €	20000	08/11/2021	08/11/2021 10:08	71671.62

LEGEND: `daily_inc` = daily income of the bank subsidiary; `op_type` = operation type; `op_amnt` = operation amount.

Transactions are registered via two releases of the same software:

1. an old-fashioned release that accounts only for day/month/year (`eventdate`)[1] and
2. an updated release that also registers hour/minute/second for each transaction (`eventdate2`).

As expected, with the old-fashioned release, Stata warns about repeated dates:

```
. xtset bank eventdate
repeated time values within panel
r(451);
```

The error `r(451)` occurs because `eventdate` shows calendar ties that make it impossible for Stata to sort the dates unambiguously.

Conversely, the software updated release fixes the calendar ties via a more detailed *timevar* (`eventdate2`), and consequently Stata does not issue the error message `r(451)` (table 2):[2]

```
. xtset bank eventdate2
Panel variable: bank (strongly balanced)
Time variable: eventdate2, 04nov2021 14:32:12 to 08nov2021 10:08:01, but with gaps
Delta: .001 seconds
. xtreg op_amnt i.op_type c.daily_inc i.eventdate2, fe
```

1. In fact, Stata allows dates with millisecond precision for this kind of transaction (see [FN] **Date and time functions**).
2. For all the `xtreg` toy examples on bank transactions reported in this tip, the default standard errors (SEs) have been left in because the limited number of groups would have caused the cluster–robust SEs to be potentially misleading (Cameron and Miller 2015).

Table 2. Time of transaction registered via the software updated release (SE)

	op_amnt	
Exchange operation		
Exchange from U.S.$ to €	4.661	
	(414.106)	
Daily income bank subsidiary	0.000	
	(0.008)	
Detailed *timevar*		
04nov2021 17:20:32	1505.253	*
	(559.358)	
05nov2021 11:20:54	3001.388	**
	(529.084)	
05nov2021 18:36:25	4504.244	**
	(578.601)	
08nov2021 10:08:01	5993.361	**
	(586.230)	
Intercept	5984.908	**
	(678.487)	
Number of observations	30	
Number of groups	6.00	
Largest group size	5.00	
F statistic	27.00	
R^2 for within model	0.90	
R^2 for between model	0.34	
R^2 for overall model	0.14	
R^2	0.90	
Adjusted R^2	0.84	
Panel-level standard deviation	5655.23	
Standard deviation of ϵ_{it}	912.85	
ρ	0.97	

** $p < 0.01$, * $p < 0.05$
LEGEND: `op_amnt` = operation amount.

When Stata throws the error `r(451)`, the usual fix is to `xtset` the dataset as in syntax 1. However, this fix comes at the cost of making time-series operators (such as lags and leads) unavailable because they require observations within each panel to be ordered according to *timevar*. Therefore, if time-series operators must be included in the regression equation, the dataset should be `xtset` as in syntax 2.

3 Can timevar still be used as a predictor after error r(451)?

Provided that no variable is differenced, lagged, or led, running `xtreg, fe` as in syntax 1 is perfectly appropriate. It also allows the *timevar* to be plugged in as a categorical predictor in the regression equation despite the error `r(451)` (table 3):

```
. xtset bank eventdate
repeated time values within panel
r(451);
. xtreg op_amnt i.op_type c.daily_inc i.eventdate, fe
```

Table 3. `xtreg, fe` with `i.`*timevar* among predictors despite error `r(451)` after `xtset` (SE)

	op_amnt	
Exchange operation		
Exchange from U.S.$ to €	−600.853	
	(467.742)	
Daily income bank subsidiary	−0.009	
	(0.010)	
Problematic *timevar*		
05nov2021	2922.614	**
	(470.085)	
08nov2021	5503.661	**
	(689.394)	
Intercept	7477.848	**
	(625.356)	
Number of observations	30	
Number of groups	6.00	
Largest group size	5.00	
F statistic	24.01	
R^2 for within model	0.83	
R^2 for between model	0.54	
R^2 for overall model	0.10	
R^2	0.83	
Adjusted R^2	0.75	
Panel-level standard deviation	5777.22	
Standard deviation of ϵ_{it}	1136.98	
ρ	0.96	

** $p < .01$, * $p < .05$
LEGEND: `op_amnt` = operation amount

4 Switching from xtreg, fe to areg when xtset returns error r(451): A good idea?

A tempting work-around for the error `r(451)` is switching from `xtreg, fe` to `areg` (see [R] **areg**) because the latter does not require `xtset`.

Unfortunately, this is not a good idea even in the absence of error `r(451)`, because of the consequences for cluster–robust SE calculation (Cameron and Miller 2015).

Let's expand on this issue using a well-known Stata dataset (table 4):

```
. webuse nlswork
(National Longitudinal Survey of Young Women, 14-24 years old in 1968)
. xtset idcode
Panel variable: idcode (unbalanced)
. xtreg ln_wage c.age##c.age i.year, fe vce(cluster idcode)
. areg ln_wage c.age##c.age i.year, absorb(idcode) vce(cluster idcode)
```

Table 4. `xtreg, fe` versus `areg`: A comparison (cluster–robust SE)

	`xtreg, fe`		`areg`	
Age in current year	0.073	**	0.073	**
	(0.014)		(0.015)	
Age in current year # Age in current year	−0.001	**	−0.001	**
	(0.000)		(0.000)	
Interview year				
69	0.065	**	0.065	**
	(0.016)		(0.017)	
70	0.028		0.028	
	(0.026)		(0.029)	
71	0.058		0.058	
	(0.038)		(0.042)	
72	0.051		0.051	
	(0.050)		(0.055)	
73	0.042		0.042	
	(0.062)		(0.068)	
75	0.015		0.015	
	(0.086)		(0.094)	
77	0.034		0.034	
	(0.111)		(0.121)	
78	0.054		0.054	
	(0.123)		(0.135)	
80	0.037		0.037	
	(0.147)		(0.161)	

Continued on next page

	`xtreg, fe`	`areg`
Interview year, cont.		
82	0.039	0.039
	(0.172)	(0.188)
83	0.059	0.059
	(0.184)	(0.201)
85	0.104	0.104
	(0.208)	(0.228)
87	0.124	0.124
	(0.233)	(0.255)
88	0.190	0.190
	(0.249)	(0.272)
Intercept	0.394	0.394
	(0.247)	(0.270)
Number of observations	28510	28510
F statistic	79.11	66.04
Number of groups	4710.00	
Largest group size	15.00	
R^2 for within model	0.12	
R^2 for between model	0.11	
R^2 for overall model	0.09	
R^2	0.12	0.67
Adjusted R^2	0.12	0.60
Panel-level standard deviation	0.40	
Standard deviation of ϵ_{it}	0.30	
ρ	0.64	

** $p < 0.01$, * $p < 0.05$

`xtreg, fe` and `areg` produce identical point estimates but different cluster–robust estimates of the variance matrix (Cameron and Miller 2015), because they make different assumptions about whether the number of panels increases with the sample size. While `xtreg, fe` gives back the correct cluster–robust estimates of the variance matrix, `areg` does not, because it uses the wrong degrees-of-freedom correction (Cameron and Miller 2015). This difference, which is particularly apparent when the number of observations per cluster is small, does not hold for default SEs.[3]

3. When the number of observations per cluster is small, the cluster–robust SEs estimated by `areg` should actually be multiplied by the square root of (Cameron and Miller 2015)

$$\{N - (K-1)\}/\{N - G - (K-1)\}$$

N = number of observations, K = number of regressors (intercept included), and G = number of clusters.

5 Leaving out timevar and exploiting the xt commands capabilities: The case of xtgee

The Stata command `xtgee` (see [XT] **xtgee**) fits both linear and nonlinear population-averaged panel-data models via generalized estimating equations (Hardin and Hilbe 2013). Being as flexible as generalized linear models (Deb, Norton, and Manning 2017; Hardin and Hilbe 2018), `xtgee` allows different within-panel correlation structures (via the `corr()` option), various link functions that relate the outcome to the linear index function in the right-hand side of the regression equation (via the `link()` option), and a set of theoretical probability distributions from which the regressand is generated (via the `family()` option). `xtgee`, which does not need a *timevar*, is asymptotically equivalent to `xtreg, re` and `xtreg, mle` (table 5).[4] When panel datasets are balanced, `xtgee` and `xtreg, mle` produce identical results. This equivalence does not hold when panels are unbalanced, because these two Stata commands deal with lack of panel balance differently.

```
. webuse nlswork
(National Longitudinal Survey of Young Women, 14-24 years old in 1968)
. xtset idcode
. xtgee ln_wage grade c.age##c.age, family(gaussian) link(identity)
> corr(exchangeable) nolog
. xtreg ln_wage grade c.age##c.age, re vce(cluster idcode)
. xtreg ln_wage grade c.age##c.age, mle vce(cluster idcode)
```

4. `xtgee` SEs are clustered on *panelvar* by default.

Table 5. `xtgee`, `xtreg, re`, and `xtreg, mle`: A comparison (SE)

	`xtgee`	`xtreg, re`	`xtreg, mle`
Current grade completed	0.080 **	0.080 **	0.080 **
	(0.002)	(0.002)	(0.002)
Age in current year	0.054 **	0.054 **	0.054 **
	(0.003)	(0.004)	(0.004)
Age in current year #	−0.001 **	−0.001 **	−0.001 **
Age in current year	(0.000)	(0.000)	(0.000)
Intercept	−0.369 **	−0.370 **	−0.370 **
	(0.045)	(0.061)	(0.061)
Number of observations	28508	28508	28508
Number of groups	4708.00	4708.00	4708.00
Largest group size	15.00	15.00	15.00
R^2 for within model		0.11	
R^2 for between model		0.32	
R^2 for overall model		0.24	
χ^2	5302.26	3050.74	3058.57
Panel-level standard deviation		0.31	0.30
Standard deviation of ϵ_{it}		0.30	0.30
ρ		0.50	0.49

** $p < 0.01$, * $p < 0.05$

6 Repeated cross-sectional studies and xt commands

In repeated cross-sectional (RCS) studies, a different sample of units per wave is measured on the same set of variables at a defined time point, as in a survey (Lebo and Weber 2015).[5]

Provided that the regressand is continuous, RCS studies are composed of multiple waves of data to be `append`ed (see [D] **append**) before running `regress` (see [R] **regress**).

According to the characteristics above, RCS studies fall outside the `xtset` framework.

However, their analysis can benefit from some of the `xt` commands that are frequently used to study panel datasets before running `xt`-related regressions.

5. Often, RCS studies' units are correlated within the same wave and across waves (Lebo and Weber 2015). However, because of the limited number of waves of data, that would cause SEs clustered on `i.year` to be potentially misleading (Lebo and Weber 2015). These issues are not explored in this tip.

A series of one-year RCS data was created by slightly tweaking the `nlswork.dta` file:

```
. webuse nlswork
(National Longitudinal Survey of Young Women, 14-24 years old in 1968)
. sort year
. drop if year>78
. generate cross_sectional_id=_n
. order cross_sectional_id, first
. keep cross_sectional_id ln_wage tenure race not_smsa south year wks_ue
```

The RCS dataset has been `xtset` with *panelvar* only to summarize the continuous variables (table 6):[6]

```
. xtset cross_sectional_id
. xtsum ln_wage tenure wks_ue
```

Table 6. `xtsum` applied to an RCS dataset

Variable		Mean	Std. dev.	Min	Max	Observations
`ln_wage`	overall	1.578669	0.4219723	0.0044871	4.242752	$N = 16094$
	between		0.4219723	0.0044871	4.242752	$n = 16094$
	within		0	1.578669	1.578669	$T = 1$
`tenure`	overall	1.865325	2.081362	0	18.5	$N = 15806$
	between		2.081362	0	18.5	$n = 15806$
	within		0	1.865325	1.865325	$T = 1$
`wks_ue`	overall	2.371952	6.861621	0	56	$N = 15709$
	between		6.861621	0	56	$n = 15709$
	within		0	2.371952	2.371952	$T = 1$

As expected, the overall and between standard deviations overlap (because $N = n$), whereas the within one is zero (because $T = 1$).

The `xtsum` outcome table mirrors the wide range of the continuous variables.

In addition, RCS datasets allow time-fixed effects to account for variations over time (table 7):[7,8]

```
. regress ln_wage c.tenure##c.tenure i.race i.not_smsa i.south i.year wks_ue,
> vce(robust)
```

6. Note that, unlike `xtsum` (see [XT] **xtsum**), `xtdescribe` (see [XT] **xtdescribe**) requires the dataset to be `xtset` with a *timevar* too.
7. `_robust` (see [P] **__robust**) SEs were imposed because of heteroskedasticity of the residual distribution checked via `estat hettest` (Prob $> \chi^2 = 0.0399$).
8. The joint statistical significance of `i.year` was tested via `testparm`: $F(8, 15403) = 16.56$; Prob $> F = 0.0000$, whereas the correct specification of the functional form of the regressand was confirmed via `linktest` ([R] **linktest**): `_hatsq` $P > |t| = 0.388$ (`linktest` returns prediction squared as `_hatsq`).

Table 7. Ordinary least squares (OLS) on an RCS dataset (robust SE)

	RCS_OLS	
Job tenure in years	0.114	**
	(0.004)	
Job tenure in years # Job tenure in years	−0.007	**
	(0.000)	
Race		
Black	−0.100	**
	(0.007)	
Other	−0.011	
	(0.027)	
1 if not standard metropolitan statistical area		
1	−0.189	**
	(0.007)	
1 if south		
1	−0.133	**
	(0.006)	
Interview year		
69	0.049	**
	(0.014)	
70	−0.003	
	(0.014)	
71	0.025	
	(0.013)	
72	0.027	
	(0.014)	
73	0.029	*
	(0.014)	
75	0.021	
	(0.013)	
77	0.085	**
	(0.014)	
78	0.112	**
	(0.014)	
Weeks unemployed last year	−0.003	**
	(0.001)	
Intercept	1.524	**
	(0.011)	
Number of observations	15419	
F statistic	316.91	
R^2	0.23	
Adjusted R^2	0.23	

** $p < .01$, * $p < .05$

7 Conclusion

This tip started from the evidence of frequent complaints about the error `r(451)` posted on the Stata forum and then expanded to other `xt`-related issues.

`xtset` has two dimensions to be addressed: the cross-sectional one (*panelvar*), which is mandatory because it tells Stata that the researcher is dealing with a panel dataset, and an optional one, that is, the time-series dimension (*timevar*).

Therefore, how to `xtset` the panel dataset is strictly related to the study goals.

Unlike the `xtabond` case (see [XT] **xtabond**), various panel-data commands that provide useful information without the need of a time variable, for example, `xtsum` for RCS studies, can give the researcher more information on standard deviation than `summarize`.

8 Acknowledgment

I thank Nicholas J. Cox for his constructive comments.

References

Cameron, A. C., and D. L. Miller. 2015. A practitioner's guide to cluster–robust inference. *Journal of Human Resources* 50: 317–372. https://doi.org/10.3368/jhr.50.2.317.

Cameron, A. C., and P. K. Trivedi. 2005. *Microeconometrics: Methods and Applications*. New York: Cambridge University Press.

———. 2022. *Microeconometrics Using Stata*. 2nd ed. College Station, TX: Stata Press.

Deb, P., E. C. Norton, and W. G. Manning. 2017. *Health Econometrics Using Stata*. College Station, TX: Stata Press.

Hardin, J. W., and J. M. Hilbe. 2013. *Generalized Estimating Equations*. 2nd ed. Boca Raton, FL: CRC Press.

———. 2018. *Generalized Linear Models and Extensions*. 4th ed. College Station, TX: Stata Press.

Lebo, M. J., and C. Weber. 2015. An effective approach to the repeated cross-sectional design. *American Journal of Political Science* 59: 242–258. https://doi.org/10.1111/ajps.12095.

Wooldridge, J. M. 2020. *Introductory Econometrics: A Modern Approach*. 7th ed. Boston: Cengage Learning.