didregress postestimation — Postestimation tools for didregress and xtdidregress

Postestimation commands

The following postestimation commands are of special interest after `didregress` and `xtdidregress`:

Command	Description
`estat trendplots`	graphical diagnostics for parallel trends
`estat ptrends`	parallel-trends test
`estat granger`	Granger causality test
`estat grangerplot`	time-specific treatment effects

The following standard postestimation commands are also available:

Command	Description
`estat summarize`	summary statistics for the estimation sample
`estat vce`	variance–covariance matrix of the estimators (VCE)
`estimates`	cataloging estimation results
`etable`	table of estimation results
`forecast`	dynamic forecasts and simulations
`lincom`	point estimates, standard errors, testing, and inference for linear combinations of coefficients
`nlcom`	point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients
`predict`	linear predictions and residuals
`predictnl`	point estimates, standard errors, testing, and inference for generalized predictions
`test`	Wald tests of simple and composite linear hypotheses
`testnl`	Wald tests of nonlinear hypotheses

predict

Description for predict

predict creates a new variable containing predictions such as linear predictors and residuals.

Menu for predict

Statistics > Postestimation

Syntax for predict

predict [*type*] *newvar* [*if*] [*in*] [, *statistic*]

statistic	Description
Main	
xb	linear predictor; the default
residuals	residuals

Options for predict

Main

xb, the default, calculates the linear predictor. It excludes the effect of the first group or of the panel identifier. All other effects, including the time fixed effects, are included in the linear predictor.

residuals calculates the overall residuals. It is the difference of the outcome and the linear predictor, including all group, panel, and time effects. In other words, it is not just the difference of the outcome and the linear predictor.

estat

Description for estat

estat trendplots produces two diagnostic plots for assessing the parallel-trends assumption that is required for consistent estimation of the ATET using didregress or xtdidregress. The first plot consists of two lines showing the mean of the outcome over time for the treatment and the control groups. The second plot augments the DID model to include interactions of time with an indicator of treatment and plots the predicted values of this augmented model for the treatment and control groups. Both plots include a vertical line one period before treatment.

estat ptrends performs a test of whether the linear trends in the outcome variable are parallel between control and treatment groups during the pretreatment period.

estat granger performs a test of whether treatment effects can be observed in anticipation of the treatment.

estat grangerplot produces a graph of time-specific treatment effects by plotting coefficients from leads and lags of the treatment indicator variable.

Menu for estat

Statistics > Postestimation

Syntax for estat

Graphical diagnostics for parallel trends

estat trendplots [, *trend_options plot_options*]

Parallel-trends test

estat ptrends

Granger causality test

estat granger

Time-specific treatment effects

estat grangerplot [, *grangerplot_options*]

trend_options	Description
omeans	draw graph showing observed means
omeans(*plot_options*)	draw observed-means graph and affect its rendition
ltrends	draw graph showing linear trends
ltrends(*plot_options*)	draw linear-trends graph and affect its rendition
notitle	suppress overall title
noxline	suppress treatment-time reference line
nocommonlegend	display two individual legends
legendfrom(#)	specify which legend to use

plot_options	Description
Plot	
cline_options	affect rendition of the plotted trend lines; see [G-3] ***cline_options***
line1opts(*cline_options*)	affect rendition of the line for controls
line2opts(*cline_options*)	affect rendition of the line for treated
Y axis, X axis, Titles, Legend, Overall	
twoway_options	any options other than by() documented in [G-3] ***twoway_options***

grangerplot_options	Description
nleads(#)	number of leads
nlags(#)	number of lags
baseline(#)	baseline period
lagview	show lags instead of time units
verbose	display results of the underlying regression
post	post the results of the underlying regression in e()
noci	do not plot confidence intervals
noyline	suppress *y*-axis reference line
CI plot	
recastci(*plottype*)	plot confidence intervals using *plottype*
ciopts(*rcap_options*)	affect rendition of confidence intervals
level(#)	set confidence level; default is level(95)
Add plots	
addplot(*plot*)	add other plots to the graph
Y axis, X axis, Titles, Legend, Overall	
twoway_options	any options other than by() documented in [G-3] ***twoway_options***

Options for estat trendplots

omeans, omeans(*plot_options*), ltrends, and ltrends(*plot_options*) specify which graphs are to be included and how they should be individually rendered. The default is omeans ltrends, meaning that both graphs are included without any modifications.

omeans specifies that the observed-means graph be included. Specifying omeans suppresses the linear-trends model graph unless ltrends or ltrends(*plot_options*) is also specified.

omeans(*plot_options*) specifies that the observed-means graph be included and affects its rendition. Specifying omeans(*plot_options*) suppresses the linear-trends model graph unless ltrends or ltrends(*plot_options*) is also specified.

ltrends specifies that the linear-trends model graph be included. Specifying ltrends suppresses the observed-means graph unless omeans or omeans(*plot_options*) is also specified.

ltrends(*plot_options*) specifies that the linear-trends model graph be included and affects its rendition. Specifying ltrends(*plot_options*) suppresses the observed-means graph unless omeans or omeans(*plot_options*) is also specified.

notitle suppresses the overall title of the rendered graph.

`noxline` suppresses rendering of the treatment-time reference line.

`nocommonlegend` suppresses the display of one common legend and renders two individual legends.

`legendfrom(#)` specifies which legend to use; the default is `legendfrom(1)`, which refers to the legend of the first plot (observed means). `legendfrom(#)` is not allowed with the `nocommonlegend` option.

Plot

cline_options affect the rendition of the plotted trend lines, including their style, size, and color; see [G-3] ***cline_options***.

`line1opts(`*cline_options*`)` affect the rendition of the plotted trend lines for the group of controls, including their style, size, and color; see [G-3] ***cline_options***.

`line2opts(`*cline_options*`)` affect the rendition of the plotted trend lines for the group of treated, including their style, size, and color; see [G-3] ***cline_options***.

Y axis, X axis, Titles, Legend, Overall

twoway_options are any of the options documented in [G-3] ***twoway_options***, excluding `by()`. These include options for titling the graph (see [G-3] ***title_options***) and for saving the graph to disk (see [G-3] ***saving_option***).

Options for estat grangerplot

`nleads(#)` specifies the number of leads to be included in the model and plotted. By default, all available leads are included. The number of leads must be greater than 0.

`nlags(#)` specifies the number of lags to be included in the model and plotted. By default, all available lags are included. The number of lags must be greater than or equal to 0.

`baseline(#)` specifies the baseline period for which the corresponding lead or lag is omitted. By default, the first lead is omitted, which corresponds to the time period prior to intervention.

`lagview` specifies to show the values of the x axis in terms of lags. If this option is not specified, time values are shown.

`verbose` specifies to display the output of the underlying regression model.

`post` posts the results of the underlying regression as estimation results in `e()`.

`noci` removes the pointwise confidence intervals. The default is to plot the confidence intervals.

`noyline` suppresses rendering of the reference line.

CI plot

`recastci(`*plottype*`)` specifies that confidence intervals be plotted using *plottype*. *plottype* may be `rarea`, `rbar`, `rspike`, `rcap`, `rcapsym`, `rline`, `rconnected`, or `rscatter`; see [G-2] **graph twoway**. When `recastci()` is specified, the plot-rendition options appropriate to the specified *plottype* may be used in lieu of *rcap_options*. For details on those options, follow the appropriate link from [G-2] **graph twoway**.

`ciopts(`*rcap_options*`)` affects the rendition of confidence intervals; see [G-3] ***rcap_options***.

`level(#)` specifies the confidence level, as a percentage, for confidence intervals. The default is `level(95)` or as set by `set level`; see [U] **20.8 Specifying the width of confidence intervals**.

Add plots

addplot(*plot*) provides a way to add other plots to the generated graph; see [G-3] ***addplot_option***.

Y axis, X axis, Titles, Legend, Overall

twoway_options are any of the options documented in [G-3] ***twoway_options***, excluding by(). These include options for titling the graph (see [G-3] ***title_options***) and for saving the graph to disk (see [G-3] ***saving_option***).

Stored results for estat

estat ptrends stores the following results for the test of linear trends in r():

Scalars

r(N)	number of observations
r(F)	test statistic
r(df_r)	number of degrees of freedom of the residuals for the F distribution under H_0
r(p)	p-value
r(df_m)	number of degrees of freedom of the test for the F distribution under H_0

estat granger stores the following results for the test of treatment anticipation in r():

Scalars

r(N)	number of observations
r(F)	test statistic
r(df_r)	number of degrees of freedom of the residuals for the F distribution under H_0
r(p)	p-value
r(df_m)	number of degrees of freedom of the test for the F distribution under H_0

estat grangerplot, when used with option post, stores results from the underlying regression model in e() and r().

Methods and formulas

The tests performed with estat ptrends and estat granger are based on augmented difference-in-differences (DID) models. With estat ptrends, we augment the DID model with terms that capture the differences in slopes between treated and controls. With estat granger, we augment the model by interacting the dummy variable that marks treated observations with dummy variables for time periods prior to the treatment to capture any potential anticipatory treatment effects.

Let's consider the case of panel data for individuals over time, in which individuals belong to a group s. Groups could be states, occupational categories, districts, etc. Let y_{its} be the outcome of individual i, who belongs to group s, at time t, where $i = 1, \dots, N$, $t = 1, \dots, T$, and $s = 1, \dots, S$.

We can write the DID model for such setups as follows:

$$y_{ist} = \gamma_i + \gamma_t + \mathbf{x}_{ist}\boldsymbol{\beta} + D_{st}\delta + \epsilon_{ist}$$

Here γ_i are individual fixed effects, γ_t are time fixed effects, $\mathbf{x}_{ist}$ are covariates, D_{st} is a variable that is 1 if an individual belongs to a group s that is treated at time t and is 0 otherwise, and ϵ_{ist} is an error term. The coefficient δ represents the average treatment effect on the treated (ATET).

To simplify the exposition below, we rewrite the model as follows:

$$
\begin{aligned}
y_{ist} &= \gamma_i + \gamma_t + \mathbf{x}_{ist}\boldsymbol{\beta} + D_{st}\delta + \epsilon_{ist} \\
y_{ist} &= \text{DID}_{ist} + \epsilon_{ist}
\end{aligned} \tag{1}
$$

The linear-trends model that is used for the parallel-trends test with `estat ptrends` augments the above model with two more terms. Let $d_{t,0} = 1(d_t = 0)$ be a variable indicating pretreatment time periods, and let $d_{t,1} = 1(d_t = 1)$ be a variable indicating posttreatment time periods. Also, let w_i be a variable that is 1 if the individual belongs to a treated group and is 0 otherwise. The augmentation terms then consist of two 3-way interactions between $d_{t,0}$, w_i, and t, and $d_{t,1}$, w_i, and t:

$$
y_{ist} = \text{DID}_{ist} + w_i d_{t,0} t \zeta_1 + w_i d_{t,1} t \zeta_2 + \varepsilon_{ist} \tag{2}
$$

Under this specification, the coefficient ζ_1 captures the differences in slopes between treatment group and control group in pretreatment periods, while ζ_2 captures the differences in slopes in posttreatment periods. If ζ_1 is 0, the linear trends in the outcome are parallel during pretreatment periods. The same is true for ζ_2 with respect to the posttreatment period; however, posttreatment differences in trends are not relevant for assessing the parallel-trends assumption. `estat ptrends` uses a Wald test of ζ_1 against 0 to assess whether the linear trends are parallel prior to treatment. Thus, the null hypothesis of this test is that the linear trends are parallel.

`estat granger` performs a Granger-type causality test to assess whether treatment effects are observed prior to the treatment. To illustrate this, suppose the treatment took place at time $t = j$. We could express D_{st} as $D_{st} = 1(t \geq j)w_i$. The Granger-type test augments the model with counterfactual treatment-time indicators. For example, if the treatment occurred at time $j - 1$, then we could construct a new treatment as $1(t_{it} \geq j - 1)w_i$, and if we have sufficient time points, we could construct another counterfactual treatment as $1(t_{it} \geq j - 2)w_i$, and so on. These terms are referred to as leads in the DID literature. The model used by `estat granger` uses the model in (1) and augments it with all leads leaving out one for identification purposes. Let J index the time at which the treatment occurs.

$$
y_{ist} = \text{DID}_{ist} + \sum_{j=2}^{J-1} 1(t_{it} \geq j) w_i \lambda_j + \nu_{ist} \tag{3}
$$

The test result is then obtained by performing a joint Wald test on the coefficients λ_j. Thus, the null hypothesis for this test is that the coefficients in λ_j are jointly 0, which is to say there are no anticipatory effects.

`estat grangerplot` fits a generalization of the DID model in (1) and plots the estimated coefficients from this model, including their 95% confidence intervals. The model is similar to (3), but this model parameterizes the leads differently and includes lags in addition to leads. Let I_s be the time of treatment, $m < 0$ be the number of time periods prior to I_s, $q \geq 0$ be the number of periods after I_s, and b be the baseline period. The model is

$$
y_{ist} = \gamma_i + \gamma_t + \mathbf{x}_{ist}\boldsymbol{\beta} + \sum_{k=m, k \neq b}^{q} B_{st}^{k} w_i \lambda_k + \epsilon_{ist}
$$

where

$$B_{st}^k = \begin{cases} 1(t_{it} \leq I_s + k), \text{ if } & k = m \\ 1(t_{it} = I_s + k), \text{ if } & m < k < q \\ 1(t_{it} \geq I_s + k), \text{ if } & k = q \end{cases}$$

This yields a model with $|m|$ leads and q lags. By default, `estat grangerplot` uses all available leads and lags. If, without loss of generality, we set the base to the period prior to treatment, $b = -1$ (the default), then with $t = 1, \ldots, T$ and $I_s = J$, a maximum of $n_{\text{leads}} = J - 2$ leads and $n_{\text{lags}} = T - J + 1$ lags is available. Notice that, if all available leads and lags are used, B_{st}^k reduces to $B_{st}^k = 1(t_{it} = I_s + k)$ because there are no periods before or after $I_s + k$. With fewer than available leads and lags, that is, $|m| < n_{\text{leads}}$ or $q < n_{\text{lags}}$, notice that the corresponding indicator variables capture the periods beyond the endpoints that correspond to m and q.

At a minimum, the model has to include a single lead. In that case, we have that $m = -1$ and $q = 0$. After omitting the base, we have that $k = 0$ and B_{st}^k reduces to $B_{st}^k = 1(t_{it} \geq I_s)$. Notice that $B_{st}^k w_i$ now yields our original treatment indicator D_{st}. In other words, the model with a single lead and no lags yields the special case of the DID model in (1). Notice also that the model in (3) is a special case, too. It is equivalent to the model fit by `estat grangerplot` with all available leads and no lags. However, (3) uses a different parameterization because the indicator variables are constructed differently. `estat grangerplot` plots the coefficients λ_k against the corresponding time periods.

The `estat trendplots` command produces two plots. The first plot is simply plotting the observed means for each treatment group at each point in time. The second plot is based on the model in (2), which is the model used for the parallel-trends test, but this model centers the continuous time variable around its minimum value:

$$y_{ist} = \text{DID}_{ist} + w_i d_{t,0}\{t - \min(t)\}\zeta_1 + w_i d_{t,1}\{t - \min(t)\}\zeta_2 + \mu_{ist}$$

Centering around the minimum time value provides a common reference point at the first observed time point such that deviations from parallelism are easily detectable. The graph then shows the predicted values from this model, evaluated at all observed time points for each of the treatment groups and at the means of the covariates.

While the formulas above are shown for the case of panel data, these methods work the same way for data that consist of repeated cross-sections.

Also see

[TE] **didregress** — Difference-in-differences estimation

[TE] **DID intro** — Introduction to difference-in-differences estimation

[U] 20 Estimation and postestimation commands