mediate postestimation — Postestimation tools for mediate

Postestimation commands

The following postestimation commands are of special interest after `mediate`:

Command	Description
`estat proportion`	proportion mediated
`estat cde`	controlled direct effects
`estat or`	effects on the odds-ratio scale
`estat rr`	effects on the risk-ratio scale
`estat irr`	effects on the incidence-rate–ratio scale
`estat effectsplot`	effects plot

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
`lincom`	point estimates, standard errors, testing, and inference for linear combinations of parameters
`nlcom`	point estimates, standard errors, testing, and inference for nonlinear combinations of parameters
`predict`	treatment effects, conditional means, etc.
`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 (or variables) containing predictions such as treatment effects, conditional means, linear predictions, and expected values.

Menu for predict

Statistics > Postestimation

Syntax for predict

```
predict [type] {stub* | newvar | newvarlist} [if] [in]
        [, effect_statistic tlevel(treat_level)]

predict [type] {stub* | newvar | newvarlist} [if] [in]
        [, po_statistic polevels(t,t')]

predict [type] newvar [if] [in] [, fitted_statistic]
```

effect_statistic	Description
Main	
`nie`	natural indirect effect; the default
`nde`	natural direct effect
`te`	total effect
`pnie`	pure natural indirect effect
`tnde`	total natural direct effect
`ite`	indirect treatment effect; synonym for `nie`
`dte`	direct treatment effect; synonym for `nde`
`tte`	total treatment effect; synonym for `te`
`itec`	indirect treatment effect with respect to controls; synonym for `pnie`
`dtet`	direct treatment effect with respect to the treated; synonym for `tnde`

po_statistic	Description
Main	
`cmean`	conditional mean at treatment levels

fitted_statistic	Description
Main	
`xb`	linear prediction for outcome model
`medxb`	linear prediction for mediator model
`mu`	expected values for outcome model
`medmu`	expected values for mediator model

If you do not specify `tlevel()` and only specify one new variable, then *effect_statistic*s assume `tlevel()` specifies the first noncontrol treatment level. You specify one or $t-1$ new variables with *effect_statistic*, where t is the number of treatment levels.

If you do not specify `polevels()` and only specify one new variable, then `polevels(`*c*`,`*c*`)` is assumed, where *c* is the control group. You specify one or *d* new variables with `cmean`, where *d* is the number of potential outcomes.

You specify one new variable with *fitted_statistic*.

Options for predict

Main

`nie`, the default, calculates the natural indirect effect for each noncontrol treatment level or for the treatment level specified in `tlevel()`. If you specify the `tlevel()` option, you must specify only one new variable; otherwise, you must specify a new variable for each treatment level (except the control level).

`nde` calculates the natural direct effect for each noncontrol treatment level or for the treatment level specified in `tlevel()`. If you specify the `tlevel()` option, you must specify only one new variable; otherwise, you must specify a new variable for each treatment level (except the control level).

`te` calculates the total effect for each noncontrol treatment level or for the treatment level specified in `tlevel()`. If you specify the `tlevel()` option, you must specify only one new variable; otherwise, you must specify a new variable for each treatment level (except the control level).

`pnie` calculates the pure natural indirect effect for each noncontrol treatment level or for the treatment level specified in `tlevel()`. If you specify the `tlevel()` option, you must specify only one new variable; otherwise, you must specify a new variable for each treatment level (except the control level).

`tnde` calculates the total natural direct effect for each noncontrol treatment level or for the treatment level specified in `tlevel()`. If you specify the `tlevel()` option, you must specify only one new variable; otherwise, you must specify a new variable for each treatment level (except the control level).

`ite` calculates the indirect treatment effect for each noncontrol treatment level or for the treatment level specified in `tlevel()`. If you specify the `tlevel()` option, you must specify only one new variable; otherwise, you must specify a new variable for each treatment level (except the control level).

`dte` calculates the direct treatment effect for each noncontrol treatment level or for the treatment level specified in `tlevel()`. If you specify the `tlevel()` option, you must specify only one new variable; otherwise, you must specify a new variable for each treatment level (except the control level).

`tte` calculates the total treatment effect for each noncontrol treatment level or for the treatment level specified in `tlevel()`. If you specify the `tlevel()` option, you must specify only one new variable; otherwise, you must specify a new variable for each treatment level (except the control level).

`itec` calculates the indirect treatment effect with respect to controls for each noncontrol treatment level or for the treatment level specified in `tlevel()`. If you specify the `tlevel()` option, you must specify only one new variable; otherwise, you must specify a new variable for each treatment level (except the control level).

`dtet` calculates the direct treatment effect with respect to the treated for each noncontrol treatment level or for the treatment level specified in `tlevel()`. If you specify the `tlevel()` option, you must specify only one new variable; otherwise, you must specify a new variable for each treatment level (except the control level).

`tlevel(`*treat_level*`)` specifies the treatment level for prediction.

`cmean` calculates the conditional mean for each potential outcome $Y(t, M(t'))$ or the potential outcome specified in `polevels()`. If you specify the `polevels()` option, you must specify only one new variable; otherwise, you must specify a new variable for each potential outcome.

`polevels(`t`,`t'`)` specifies the values of the treatment for which potential outcomes are to be calculated. The first value, t, refers to the value that the treatment is set to in the outcome equation; the second value, t', refers to the value of the treatment in the mediator equation.

`xb` calculates the linear prediction for the outcome model.

`medxb` calculates the linear prediction for the mediator model.

`mu` calculates the expected values of the dependent variable of the outcome model.

`medmu` calculates the expected values of the dependent variable of the mediator model.

estat

Description for estat

`estat proportion` calculates the indirect effect as a proportion of the total effect.

`estat cde` calculates controlled direct effects.

`estat or` calculates effects on the odds-ratio scale after `mediate` with the `logit` or `probit` outcome model.

`estat rr` calculates effects on the risk-ratio scale after `mediate` with the `logit` or `probit` outcome model.

`estat irr` calculates effects on the incidence-rate–ratio scale after `mediate` with the `poisson` or `expmean` outcome model.

`estat effectsplot` plots the estimated effects. Typically, this is useful if there are more than two treatment groups in the case of a multivalued treatment or if a continuous treatment is evaluated at more than two points. By default, `estat effectsplot` plots the effects estimated in the previous `mediate` command.

Menu for estat

Statistics > Postestimation

Syntax for estat

Proportion mediated

`estat proportion` [, *prop_options*]

Controlled direct effects

`estat cde, mvalue(`*numlist*`)` [*cde_options*]

Effects on the odds-ratio scale

`estat or` [, *scale_options*]

Effects on the risk-ratio scale

`estat rr` [, *scale_options*]

Effects on the incidence-rate–ratio scale

`estat irr` [, *scale_options*]

Effects plot

`estat effectsplot` [, *effectsplot_options*]

prop_options	Description
level(#)	set confidence level; default is level(95)
percent	display percentage instead of proportion
force	force calculations to proceed in case of conflicting signs
nolegend	suppress table legend
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

cde_options	Description
* mvalue(*numlist*)	value of the mediator variable
rr	controlled direct effect on risk-ratio scale
or	controlled direct effect on odds-ratio scale
irr	controlled direct effect on incidence-rate–ratio scale
level(#)	set confidence level; default is level(95)
contrast	differences of controlled direct effects
nolegend	suppress table legend
atmeans	controlled direct effect at the means of covariates
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

* mvalue(*numlist*) is required.

scale_options	Description
level(#)	set confidence level; default is level(95)
nolegend	suppress table legend
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

estat or, estat rr, and estat irr require estimation of potential-outcome means with mediate.

If no potential-outcome means were estimated, estat or, estat rr, and estat irr will refit the model in the background; the reestimation does not affect the results, but computation takes longer.

effectsplot_options	Description
When `mediate` *had Pearl's labeling of effects*	
`nie`	plot natural indirect effects
`nde`	plot natural direct effects
`te`	plot total effects
`pnie`	plot pure natural indirect effects
`tnde`	plot total natural direct effects
When `mediate` *had ATE labeling of effects*	
`aite`	plot average indirect treatment effects
`adte`	plot average direct treatment effects
`ate`	plot average treatment effects
`aitec`	plot average indirect treatment effects with respect to controls
`adtet`	plot average direct treatment effects with respect to the treated
Main	
`noci`	do not plot confidence intervals
Plot	
plot_options	affect rendition of all effect plots
`plot`#`opts(`*plot_options*`)`	affect rendition of #th effect plot
`recast(`*plottype*`)`	plot effects using *plottype*
CI plot	
`ciopts(`*rcap_options*`)`	affect rendition of confidence intervals
`ci`#`opts(`*rcap_options*`)`	affect rendition of #th confidence interval plot
`recastci(`*plottype*`)`	plot confidence intervals using *plottype*
`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***

plot_options	Description
marker_options	change look of markers (color, size, etc.)
marker_label_options	add marker labels; change look or position
cline_options	change look of the line

Options for estat proportion

`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**.

`percent` specifies to calculate percentages. By default, `estat proportion` calculates proportions.

`force` forces calculations to proceed in case of conflicting signs. By default, `estat proportion` issues an error message if opposite signs among indirect, direct, and total effects are detected. In that case, the result is typically not interpretable in a meaningful way.

`nolegend` suppresses the display of the table legend.

display_options: `noci`, `nopvalues`, `nofvlabel`, `fvwrap(#)`, `fvwrapon(style)`, `cformat(%fmt)`, `pformat(%fmt)`, `sformat(%fmt)`, and `nolstretch`; see [R] **Estimation options**.

Options for estat cde

`mvalue(numlist)` specifies the value of the mediator variable at which to evaluate the controlled direct effect. If the causal mediation model contained a continuous treatment variable, only a single value may be specified. `mvalue()` is required.

`rr` specifies to calculate controlled direct effect on the risk-ratio scale after `mediate` with the `logit` or `probit` outcome model.

`or` specifies to calculate controlled direct effect on the odds-ratio scale after `mediate` with the `logit` or `probit` outcome model.

`irr` specifies to calculate controlled direct effect on the incidence-rate–ratio scale after `mediate` with the `poisson` or `expmean` outcome model.

`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**.

`contrast` specifies to calculate differences of controlled direct effects between evaluations at different points of the mediator, where the base effect is the one defined by the first value in `mvalue()`; this option requires at least two evaluation points to be specified in `mvalue()`.

`nolegend` suppresses the display of the table legend.

`atmeans` specifies to evaluate the controlled direct effect at the means of covariates. By default, the counterfactual predictions are averaged over the covariates.

display_options: `noci`, `nopvalues`, `nofvlabel`, `fvwrap(#)`, `fvwrapon(style)`, `cformat(%fmt)`, `pformat(%fmt)`, `sformat(%fmt)`, and `nolstretch`; see [R] **Estimation options**.

Options for estat or, estat rr, and estat irr

`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**.

`nolegend` suppresses the display of the table legend.

display_options: `noci`, `nopvalues`, `nofvlabel`, `fvwrap(#)`, `fvwrapon(style)`, `cformat(%fmt)`, `pformat(%fmt)`, `sformat(%fmt)`, and `nolstretch`; see [R] **Estimation options**.

Options for estat effectsplot

`nie`, `nde`, `te`, `pnie`, `tnde`, `aite`, `adte`, `ate`, `aitec`, and `adtet` specify to plot the respective treatment effects. For these effects to be plotted, they must be part of the model estimates. By default, `estat effectsplot` plots the effects estimated in the previous `mediate` command.

Main

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

Plot

plot_options affects the rendition of all effect plots. The *plot_options* can affect the size and color of markers, whether and how the markers are labeled, and whether and how the points are connected; see [G-3] ***marker_options***, [G-3] ***marker_label_options***, and [G-3] ***cline_options***.

These settings may be overridden for specific plots by using the `plot`#`opts()` option.

`plot`#`opts(`*plot_options*`)` affects the rendition of the #th effect plot. The *plot_options* can affect the size and color of markers, whether and how the markers are labeled, and whether and how the points are connected; see [G-3] ***marker_options***, [G-3] ***marker_label_options***, and [G-3] ***cline_options***.

`recast(`*plottype*`)` specifies that effects be plotted using *plottype*. *plottype* may be `scatter`, `line`, `connected`, `bar`, `area`, `spike`, `dropline`, or `dot`; see [G-2] **graph twoway**. When `recast()` is specified, the plot-rendition options appropriate to the specified *plottype* may be used in lieu of *plot_options*. For details on those options, follow the appropriate link from [G-2] **graph twoway**.

CI plot

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

These settings may be overridden for specific confidence interval plots with the `ci`#`opts()` option.

`ci`#`opts(`*rcap_options*`)` affects the rendition of the #th confidence interval; see [G-3] ***rcap_options***.

`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**.

`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***).

Remarks and examples

Below we provide examples for predict. To see an example of estat proportion, see *A simple causal mediation model* in [CAUSAL] **mediate**. To see an example of estat cde, see *Estimating controlled direct effects* in [CAUSAL] **mediate**. To see an example of estat effectsplot, see *Causal mediation model with continuous treatment* in [CAUSAL] **mediate**. An example of estat rr and estat or is shown in *Estimating treatment effects on different scales* in [CAUSAL] **mediate**.

Example 1: Predicting individual-level direct, indirect, and total effects

We can use predict to make a variety of predictions from the fitted mediation model, such as individual-level direct, indirect, and total effects; potential outcomes; and linear predictions and expected values of the outcome and mediator. Suppose we have the following mediation model with binary outcome and binary mediator:

```
. use https://www.stata-press.com/data/r19/wellbeing
(Fictional well-being data)
. mediate (bwellbeing age gender i.hstatus basewell, logit)
>         (bbonotonin, logit)
>         (exercise)
Iteration 0:  EE criterion = 8.253e-18
Iteration 1:  EE criterion = 6.067e-33
Causal mediation analysis                          Number of obs = 2,000
Outcome model:     Logit
Mediator model:    Logit
Mediator variable: bbonotonin
Treatment type:    Binary
```

bwellbeing	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
NIE						
exercise (Exercise vs Control)	.1052971	.0170666	6.17	0.000	.0718472	.1387471
NDE						
exercise (Exercise vs Control)	.1524917	.0208284	7.32	0.000	.1116689	.1933146
TE						
exercise (Exercise vs Control)	.2577889	.0143	18.03	0.000	.2297613	.2858164

```
Note: Outcome equation includes treatment-mediator interaction.
```

Using predict without options yields estimated individual-level natural indirect effects:

```
. predict nie
(option nie assumed; natural indirect effect)
```

We could go ahead and predict individual-level direct and total effects by using options `nde` and `te`, respectively:

```
. predict nde, nde
. predict te, te
```

Here is an excerpt from the data showing the predicted effects for five individuals:

```
. list nie nde te in 1/5
```

	nie	nde	te
1.	.0504899	.2496191	.3001091
2.	.1693522	.1037404	.2730926
3.	.2145208	.3612216	.5757424
4.	.0265223	.1576028	.1841251
5.	.2005004	.3735286	.574029

We can see that the indirect and direct effects sum to the total effect for each individual. The differences in effects between individuals are due to their differences in covariates. Had we fit the model without covariates, the predicted effects would be constant over the sample.

If we look at the sample means of the newly generated variables `nie`, `nde`, and `te`, we can see that their averages match the estimates from `mediate` for NIE, NDE, and TE, respectively:

```
. summarize nie nde te
```

Variable	Obs	Mean	Std. dev.	Min	Max
nie	2,000	.1052971	.0883299	.000014	.2529227
nde	2,000	.1524917	.1354418	.0001044	.3877526
te	2,000	.2577889	.2037248	.0001184	.5757825

◁

▷ Example 2: Predicting potential outcomes

In addition to individual-level effects, we can also predict individual-level potential outcomes by using the `cmean` option. By default, `predict` with `cmean` will compute the potential outcomes for the control level of the treatment variable. For example, if the treatment variable is binary and takes on the values 0 and 1, where 0 is the control level, we will predict potential outcomes $Y_i[0, M_i(0)]$:

```
. predict po_y0m0, cmean
```

We can also target other potential outcomes by using the `polevels()` option. For instance, to compute potential outcomes $Y_i[1, M_i(0)]$, we specify option `polevels(1,0)`:

```
. predict po_y1m0, cmean polevels(1,0)
```

If we wish to predict all potential outcomes at once, we can use the *stub*∗ notation:

```
. predict po_*, cmean
```

In this case, there are four potential outcomes available, so Stata creates four new variables. Using `describe`, we can also see that the new variables are labeled according to the estimated potential outcome:

```
. describe po_?
Variable      Storage   Display    Value
    name         type    format    label      Variable label
------------------------------------------------------------------------------
po_1            float   %9.0g                 Conditional mean, Y[0,M(0)]
po_2            float   %9.0g                 Conditional mean, Y[1,M(0)]
po_3            float   %9.0g                 Conditional mean, Y[0,M(1)]
po_4            float   %9.0g                 Conditional mean, Y[1,M(1)]
```

◁

Stored results

`estat proportion` stores the following results in `r()`:

Scalars
- `r(N)` number of observations

Macros
- `r(title)` title in estimation output

Matrices
- `r(b)` vector of estimated proportions or percentages
- `r(V)` variance–covariance matrix of the estimates
- `r(table)` matrix containing the estimates with their standard errors, test statistics, p-values, and confidence intervals

`estat cde` stores the following results in `r()`:

Scalars
- `r(N)` number of observations

Macros
- `r(title)` title in estimation output

Matrices
- `r(b)` vector of estimated controlled direct effects or their contrasts
- `r(V)` variance–covariance matrix of the estimates
- `r(table)` matrix containing the estimates with their standard errors, test statistics, p-values, and confidence intervals

`estat or`, `estat rr`, and `estat irr` store the following results in `r()`:

Scalars
- `r(N)` number of observations
- `r(level)` confidence level

Matrices
- `r(b)` vector of transformed treatment effects (log scale)
- `r(V)` variance–covariance matrix of the estimates
- `r(table)` matrix containing the estimates with their standard errors, test statistics, p-values, and confidence intervals

Also see

[CAUSAL] **mediate** — Causal mediation analysis

[U] 20 Estimation and postestimation commands