

Postestimation commands
References

[predict](#)
Also see

Remarks and examples

Methods and formulas

Postestimation commands

The following postestimation commands are available after `boxcox`:

Command	Description
<code>estat ic</code>	Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC, respectively)
<code>estat summarize</code>	summary statistics for the estimation sample
<code>estat vce</code>	variance–covariance matrix of the estimators (VCE)
<code>estimates</code>	cataloging estimation results
<code>etable</code>	table of estimation results
<code>hausman</code>	Hausman's specification test
* <code>lincom</code>	point estimates, standard errors, testing, and inference for linear combinations of parameters
* <code>nlcom</code>	point estimates, standard errors, testing, and inference for nonlinear combinations of parameters
<code>predict</code>	predictions and residuals
* <code>test</code>	Wald tests of simple and composite linear hypotheses
* <code>testnl</code>	Wald tests of nonlinear hypotheses

*Inference is valid only for hypotheses concerning λ and θ .

predict

Description for predict

`predict` creates a new variable containing predictions such as predicted values and residuals.

Menu for predict

Statistics > Postestimation

Syntax for predict

```
predict [type] newvar [if] [in] [ , statistic options ]
```

<i>statistic</i>	Description
------------------	-------------

Main

<code>yhat</code>	predicted value of y ; the default
<code>residuals</code>	residuals

<i>options</i>	Description
----------------	-------------

Options

<code>smearing</code>	compute statistic using smearing method; the default
<code>btransform</code>	compute statistic using back-transform method

These statistics are available both in and out of sample; type `predict ... if e(sample) ...` if wanted only for the estimation sample.

Options for predict

Main

`yhat`, the default, calculates the predicted value of the dependent variable.

`residuals` calculates the residuals, that is, the observed value minus the predicted value.

Options

`smearing` calculates the statistics `yhat` and `residuals` using the smearing method proposed by [Duan \(1983\)](#) (see [Methods and formulas](#) for a description of this method). `smearing` is the default.

`btransform` calculates the statistics `yhat` and `residuals` using the back-transform method (see [Methods and formulas](#) for a description of this method).

Remarks and examples

Below, we present two examples that illustrate how to use the `smearing` and `btransform` options.

▷ Example 1: Predictions with the `smearing` option

In this example, we calculate the predicted values of diastolic blood pressure, `bpdiast`, that arise from the `theta` model calculated in [example 1](#) of [R] `boxcox`.

```
. use https://www.stata-press.com/data/r19/nhanes2
. boxcox bpdiast bmi tcresult, notrans(age sex) model(theta) lrtest
  (output omitted)
. predict yhat
  (statistic yhat and option smearing are assumed)
```

In the expression above, `yhat` is the name we gave to the estimates of the conditional expectation. Given that we did not specify any statistic or option, the corresponding defaults `yhat` and `smearing` were assumed.

As the summary table below illustrates, the mean of the dependent variable is close to the mean of the predicted value `yhat`. This indicates that the `theta` model does a good job approximating the true value of diastolic blood pressure, `bpdiast`.

```
. summarize bpdiast yhat
```

Variable	Obs	Mean	Std. dev.	Min	Max
bpdiast	10,351	81.715	12.92722	35	150
yhat	10,351	81.71406	5.983486	66.93709	110.5283

Similarly, we could have asked that residuals be calculated. Here we again use the default `smearing` option:

```
. predict resid, residuals
  (option smearing assumed to compute residuals)
```



▷ Example 2: Predictions with the `btransform` option

In this example, we illustrate the tradeoffs involved by using the `btransform` option as opposed to the default `smearing` option. Continuing with [example 1](#), we compute the predicted values using the back-transform method.

```
. predict yhatb, btransform
  (statistic yhat assumed)
```

We now compute the predicted values using the `smearing` option and summarize both computations.

```
. predict yhats
  (statistic yhat and option smearing are assumed)
. summarize bpdiast yhats yhatb
```

Variable	Obs	Mean	Std. dev.	Min	Max
bpdiast	10,351	81.715	12.92722	35	150
yhats	10,351	81.71406	5.983486	66.93709	110.5283
yhatb	10,351	81.08018	5.95549	66.37479	109.7671

As can be seen from the mean and the standard deviation of the summary table, the predicted values using the back-transform method give biased estimates but are less variable than those coming from the smearing method. However, the efficiency loss is small compared with the bias reduction.

◀

□ Technical note

boxcox estimates variances only for the λ and θ parameters (see the [technical note](#) in [R] `boxcox`), so the extent to which postestimation commands can be used following `boxcox` is limited. Formulas used in `lincom`, `nlcom`, `test`, and `testnl` are dependent on the estimated variances. Therefore, the use of these commands is limited and generally applicable only to inferences on the λ and θ coefficients.

□

Methods and formulas

The computation of the expected value of the dependent variable conditional on the regressors for the Box–Cox model does not follow the logic of the standard linear regression model because the random disturbance does not vanish from the conditional expectation and must be accounted for. To show this, we will revisit the `lhsonly` model described by

$$y_j^{(\lambda)} = \beta_0 + \beta_1 x_{1j} + \beta_2 x_{2j} + \dots + \beta_{(k-1)} x_{(k-1)j} + \epsilon_j$$

where

$$y^{(\lambda)} = \frac{y^\lambda - 1}{\lambda}$$

and

$$y^{(\lambda)} = \begin{cases} y - 1 & \text{if } \lambda = 1 \\ \ln(y) & \text{if } \lambda = 0 \\ 1 - 1/y & \text{if } \lambda = -1 \end{cases}$$

For the presentation below, let $\mathbf{y}^{(\lambda)}$ be an $N \times 1$ vector of elementwise transformed data, \mathbf{X} be an $N \times k$ matrix of regressors, $\boldsymbol{\beta}$ be a $k \times 1$ vector of parameters, and $\boldsymbol{\iota}$ be an $n \times 1$ vector of ones.

If we were interested in $E(\mathbf{y}^{(\lambda)}|\mathbf{X})$, then the conventional logic would follow, and we would obtain predictions as $y^{(\lambda)} = \mathbf{X}\hat{\boldsymbol{\beta}}$, where $\hat{\boldsymbol{\beta}}$ is the estimate of $\boldsymbol{\beta}$. However, to estimate the conditional expectation of \mathbf{y} , we need to isolate it on the left-hand side of the model. In the case of the `lhsonly` model, this yields

$$\mathbf{y} = \{\lambda(\mathbf{X}\hat{\boldsymbol{\beta}} + \boldsymbol{\epsilon}) + \boldsymbol{\iota}\}^{1/\lambda}$$

The conditional expectation is then defined by

$$E(\mathbf{y}|\mathbf{X}) = \int \{\lambda(\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}) + \boldsymbol{\iota}\}^{1/\lambda} dF(\boldsymbol{\epsilon}|\mathbf{X})$$

In the expression above, $dF(\boldsymbol{\epsilon}|\mathbf{X})$ corresponds to the cdf of $\boldsymbol{\epsilon}$ conditional on the regressors. It is also clear that the random disturbance does not vanish.

To address this issue, the default methodology used by `predict` computes this integral using the smearing method proposed by [Duan \(1983\)](#) to implement a two-step estimator, as was suggested by [Abrevaya \(2002\)](#).

In the first step, we get an estimate for ϵ defined as

$$\hat{\epsilon} = \mathbf{y}^{(\hat{\lambda})} - \mathbf{X}\hat{\beta}$$

In the second step, for each j we compute our predicted values as the sum:

$$\hat{y}_j = \frac{1}{N} \sum_{i=1}^N \{ \hat{\lambda}(\mathbf{x}_j \hat{\beta} + \hat{\epsilon}_i) + 1 \}^{1/\hat{\lambda}}$$

In the expression above, \mathbf{x}_j is the j th row of the matrix \mathbf{X} (in other words, the values of the covariates for individual j), and $\hat{\epsilon}_i$ is the residual for individual i . The result of this summation gives us the conditional expectation of the dependent variable for individual j . Given that this operation is performed for each individual j , the methodology is computationally intensive.

The back-transform method can be understood as a naïve estimate that disregards the random disturbance. The predictions using this approach are given by

$$\hat{y}_j = (\hat{\lambda} \mathbf{x}_j \hat{\beta} + 1)^{1/\hat{\lambda}}$$

References

- Abrevaya, J. 2002. Computing marginal effects in the Box–Cox model. *Econometric Reviews* 21: 383–393. <https://doi.org/10.1081/ETC-120015789>.
- Duan, N. 1983. Smearing estimate: A nonparametric retransformation method. *Journal of the American Statistical Association* 78: 605–610. <https://doi.org/10.1080/01621459.1983.10478017>.

Also see

- [R] **boxcox** — Box–Cox regression models
- [U] **20 Estimation and postestimation commands**

Stata, Stata Press, and Mata are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow and NetCourseNow are trademarks of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2025 StataCorp LLC, College Station, TX, USA. All rights reserved.

For suggested citations, see the FAQ on [citing Stata documentation](#).

