**predict — Obtain predictions, residuals, etc., after estimation**

### Syntax

**After single-equation (SE) models**

\[ \text{predict [type] newvar [if] [in] [, single_options]} \]

**After multiple-equation (ME) models**

\[ \text{predict [type] newvar [if] [in] [, multiple_options]} \]

\[ \text{predict [type] \{} \text{stub*|newvar}_1 \ldots \text{newvar}_q \} [if] [in], scores \]

### single_options

- **xb**
  - calculate linear prediction
- **stdp**
  - calculate standard error of the prediction
- **score**
  - calculate first derivative of the log likelihood with respect to \( x_j\beta \)

### Options

- **nooffset**
  - ignore any `offset()` or `exposure()` variable
- **other_options**
  - command-specific options

### multiple_options

- **equation(eqno[, eqno])**
  - specify equations
- **xb**
  - calculate linear prediction
- **stdp**
  - calculate standard error of the prediction
- **stddp**
  - calculate the difference in linear predictions

### Options

- **nooffset**
  - ignore any `offset()` or `exposure()` variable
- **other_options**
  - command-specific options

### Menu for predict

- Statistics > Postestimation > Predictions, residuals, etc.
predict — Obtain predictions, residuals, etc., after estimation

Description

predict calculates predictions, residuals, influence statistics, and the like after estimation. Exactly what predict can do is determined by the previous estimation command; command-specific options are documented with each estimation command. Regardless of command-specific options, the actions of predict share certain similarities across estimation commands:

1. predict newvar creates newvar containing “predicted values”—numbers related to the $E(y_j|x_j)$. For instance, after linear regression, predict newvar creates $x_j b$ and, after probit, creates the probability $\Phi(x_j b)$.

2. predict newvar, xb creates newvar containing $x_j b$. This may be the same result as option 1 (for example, linear regression) or different (for example, probit), but regardless, option xb is allowed.

3. predict newvar, stdp creates newvar containing the standard error of the linear prediction $x_j b$.

4. predict newvar, other_options may create newvar containing other useful quantities; see help or the reference manual entry for the particular estimation command to find out about other available options.

5. nooffset added to any of the above commands requests that the calculation ignore any offset or exposure variable specified by including the offset(varname) or exposure(varname) option when you fit the model.

predict can be used to make in-sample or out-of-sample predictions:

6. predict calculates the requested statistic for all possible observations, whether they were used in fitting the model or not. predict does this for standard options 1–3 and generally does this for estimator-specific options 4.

7. predict newvar if e(sample), ... restricts the prediction to the estimation subsample.

8. Some statistics make sense only with respect to the estimation subsample. In such cases, the calculation is automatically restricted to the estimation subsample, and the documentation for the specific option states this. Even so, you can still specify if e(sample) if you are uncertain.

9. predict can make out-of-sample predictions even using other datasets. In particular, you can

```plaintext
    . use ds1   
    . (fit a model)   
    . use two      ../another dataset     */
    . predict yhat, ... /* fill in the predictions */
```

Options

---

**xb**

This calculates the linear prediction from the fitted model. That is, all models can be thought of as estimating a set of parameters $b_1, b_2, \ldots, b_k$, and the linear prediction is $\hat{y}_j = b_1 x_{1j} + b_2 x_{2j} + \cdots + b_k x_{kj}$, often written in matrix notation as $\hat{y}_j = x_j b$. For linear regression, the values $\hat{y}_j$ are called the predicted values or, for out-of-sample predictions, the forecast. For logit and probit, for example, $\hat{y}_j$ is called the logit or probit index.

$x_{1j}, x_{2j}, \ldots, x_{kj}$ are obtained from the data currently in memory and do not necessarily correspond to the data on the independent variables used to fit the model (obtaining $b_1, b_2, \ldots, b_k$).
predict calculates the standard error of the linear prediction. Here the prediction means the same thing as the “index”, namely, $x_j b$. The statistic produced by stdp can be thought of as the standard error of the predicted expected value, or mean index, for the observation’s covariate pattern. The standard error of the prediction is also commonly referred to as the standard error of the fitted value. The calculation can be made in or out of sample.

stddp is allowed only after you have previously fit a multiple-equation model. The standard error of the difference in linear predictions $(x_{1j} b - x_{2j} b)$ between equations 1 and 2 is calculated. This option requires that equation(eqno1, eqno2) be specified.

score calculates the equation-level score, $\partial \ln L / \partial (x_j \beta)$. Here $\ln L$ refers to the log-likelihood function.

scores is the ME model equivalent of the score option, resulting in multiple equation-level score variables. An equation-level score variable is created for each equation in the model; ancillary parameters—such as $\ln \sigma$ and $\text{atanh} \rho$—make up separate equations.

equation(eqno[eqno]) — synonym outcome() — is relevant only when you have previously fit a multiple-equation model. It specifies the equation to which you are referring.

equation() is typically filled in with one eqno—it would be filled in that way with options xb and stdp, for instance. equation(#1) would mean the calculation is to be made for the first equation, equation(#2) would mean the second, and so on. You could also refer to the equations by their names. equation(income) would refer to the equation named income and equation(hours) to the equation named hours.

If you do not specify equation(), results are the same as if you specified equation(#1).

Other statistics, such as stddp, refer to between-equation concepts. In those cases, you might specify equation(#1,#2) or equation(income, hours). When two equations must be specified, equation() is required.

Options

nooffset may be combined with most statistics and specifies that the calculation should be made, ignoring any offset or exposure variable specified when the model was fit.

This option is available, even if it is not documented for predict after a specific command. If neither the offset(varname_o) option nor the exposure(varname_e) option was specified when the model was fit, specifying nooffset does nothing.

other_options refers to command-specific options that are documented with each command.

Remarks and examples

Remarks are presented under the following headings:

- Estimation-sample predictions
- Out-of-sample predictions
- Residuals
- Single-equation (SE) models
- SE model scores
- Multiple-equation (ME) models
- ME model scores

Most of the examples are presented using linear regression, but the general syntax is applicable to all estimators.
You can think of any estimation command as estimating a set of coefficients $b_1, b_2, \ldots, b_k$ corresponding to the variables $x_1, x_2, \ldots, x_k$, along with a (possibly empty) set of ancillary statistics $\gamma_1, \gamma_2, \ldots, \gamma_m$. All estimation commands store the $b_i$s and $\gamma_i$s. predict accesses that stored information and combines it with the data currently in memory to make various calculations. For instance, predict can calculate the linear prediction, $\hat{y}_j = b_1 x_{1j} + b_2 x_{2j} + \cdots + b_k x_{kj}$. The data on which predict makes the calculation can be the same data used to fit the model or a different dataset—it does not matter. predict uses the stored parameter estimates from the model, obtains the corresponding values of $x$ for each observation in the data, and then combines them to produce the desired result.

Estimation-sample predictions

Example 1

We have a 74-observation dataset on automobiles, including the mileage rating (mpg), the car’s weight (weight), and whether the car is foreign (foreign). We fit the model

```
use http://www.stata-press.com/data/r13/auto
(1978 Automobile Data)
regress mpg weight if foreign
```

```
<table>
<thead>
<tr>
<th>Source</th>
<th>SS</th>
<th>df</th>
<th>MS</th>
<th>Number of obs</th>
<th>F( 1, 20)</th>
<th>Prob &gt; F</th>
<th>R-squared</th>
<th>Adj R-squared</th>
<th>Root MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>427.990298</td>
<td>1</td>
<td>427.990298</td>
<td>22</td>
<td>17.47</td>
<td>0.0005</td>
<td>0.4663</td>
<td>0.4396</td>
<td>4.9491</td>
</tr>
<tr>
<td>Residual</td>
<td>489.873338</td>
<td>20</td>
<td>24.4936669</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>917.863636</td>
<td>21</td>
<td>43.7077922</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

mpg | Coef. | Std. Err. | t  | P>|t|  | [95% Conf. Interval] |
---|-------|-----------|----|-------|----------------------|
weight | -.010426 | .00294942  | -4.18 | 0.000 | -.0156287 - .0052232 |
_cons   | 48.9183  | 5.871851   | 8.33 | 0.000 | 36.66983  61.16676   |
```

If we were to type `predict pmpg` now, we would obtain the linear predictions for all 74 observations. To obtain the predictions just for the sample on which we fit the model, we could type

```
predict pmpg if e(sample)
(option xb assumed; fitted values)
(52 missing values generated)
```

Here `e(sample)` is true only for foreign cars because we typed `if foreign` when we fit the model and because there are no missing values among the relevant variables. If there had been missing values, `e(sample)` would also account for those.

By the way, the `if e(sample)` restriction can be used with any Stata command, so we could obtain summary statistics on the estimation sample by typing

```
summarize if e(sample)
(output omitted)
```
Out-of-sample predictions

By out-of-sample predictions, we mean predictions extending beyond the estimation sample. In the example above, typing `predict pmpg` would generate linear predictions using all 74 observations.

`predict` will work on other datasets, too. You can use a new dataset and type `predict` to obtain results for that sample.

Example 2

Using the same auto dataset, assume that we wish to fit the model

$$mpg = \beta_1 weight + \beta_2 \ln(weight) + \beta_3 foreign + \beta_4$$

We first create the \(\ln(weight)\) variable, and then type the `regress` command:

```
. use http://www.stata-press.com/data/r13/auto, clear
(1978 Automobile Data)
. generate lnweight = ln(weight)
. regress mpg weight lnweight foreign
```

If we typed `predict pmpg` now, we would obtain predictions for all 74 cars in the current data. Instead, we are going to use a new dataset.

The dataset `newautos.dta` contains the make, weight, and place of manufacture of two cars, the Pontiac Sunbird and the Volvo 260. Let’s use the dataset and create the predictions:

```
. use http://www.stata-press.com/data/r13/newautos, clear
(New Automobile Models)
. list

    make       weight    foreign
  1. Pont. Sunbird   2690 Domestic
  2. Volvo 260       3170 Foreign

. predict mpg
(option xb assumed; fitted values)
variable lnweight not found
r(111);
```

Things did not work. We typed `predict mpg`, and Stata responded with the message “variable lnweight not found”. `predict` can calculate predicted values on a different dataset only if that dataset contains the variables that went into the model. Here our dataset does not contain a variable called lnweight. lnweight is just the log of weight, so we can create it and try again:
. generate lnweight = ln(weight)
. predict mpg
(option xb assumed; fitted values)
. list

<table>
<thead>
<tr>
<th>make</th>
<th>weight</th>
<th>foreign</th>
<th>lnweight</th>
<th>mpg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pont. Sunbird</td>
<td>2690</td>
<td>Domestic</td>
<td>7.897296</td>
<td>23.25097</td>
</tr>
<tr>
<td>Volvo 260</td>
<td>3170</td>
<td>Foreign</td>
<td>8.061487</td>
<td>17.85295</td>
</tr>
</tbody>
</table>

We obtained our predicted values. The Pontiac Sunbird has a predicted mileage rating of 23.3 mpg, whereas the Volvo 260 has a predicted rating of 17.9 mpg.

Residuals

Example 3

With many estimators, predict can calculate more than predicted values. With most regression-type estimators, we can, for instance, obtain residuals. Using our regression example, we return to our original data and obtain residuals by typing

. use http://www.stata-press.com/data/r13/auto, clear
(1978 Automobile Data)
. generate lnweight = ln(weight)
. regress mpg weight lnweight foreign
(output omitted)
. predict double resid, residuals
. summarize resid

<table>
<thead>
<tr>
<th>Variable</th>
<th>Obs</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>resid</td>
<td>74</td>
<td>-1.51e-15</td>
<td>3.212091</td>
<td>-5.453078</td>
<td>13.83719</td>
</tr>
</tbody>
</table>

We could do this without refitting the model. Stata always remembers the last set of estimates, even as we use new datasets.

It was not necessary to type the double in predict double resid, residuals, but we wanted to remind you that you can specify the type of a variable in front of the variable’s name; see [U] 11.4.2 Lists of new variables. We made the new variable resid a double rather than the default float.

If you want your residuals to have a mean as close to zero as possible, remember to request the extra precision of double. If we had not specified double, the mean of resid would have been roughly $10^{-9}$ rather than $10^{-14}$. Although $10^{-14}$ sounds more precise than $10^{-9}$, the difference really does not matter.

For linear regression, predict can also calculate standardized residuals and Studentized residuals with the options rstandard and rstudent; for examples, see [R] regress postestimation.
Single-equation (SE) models

If you have not read the discussion above on using predict after linear regression, please do so. And predict’s default calculation almost always produces a statistic in the same metric as the dependent variable of the fitted model—for example, predicted counts for Poisson regression. In any case, xb can always be specified to obtain the linear prediction.

predict can calculate the standard error of the prediction, which is obtained by using the covariance matrix of the estimators.

Example 4

After most binary outcome models (for example, logistic, logit, probit, cloglog, scobit), predict calculates the probability of a positive outcome if we do not tell it otherwise. We can specify the xb option if we want the linear prediction (also known as the logit or probit index). The odd abbreviation xb is meant to suggest \( x \beta \). In logit and probit models, for example, the predicted probability is 

\[
p = F(x \beta),
\]

where \( F() \) is the logistic or normal cumulative distribution function, respectively.

```
. logistic foreign mpg weight
(output omitted)
. predict phat
(option pr assumed; Pr(foreign))
. predict idxhat, xb
. summarize foreign phat idxhat
```

<table>
<thead>
<tr>
<th>Variable</th>
<th>Obs</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>foreign</td>
<td>74</td>
<td>.2972973</td>
<td>.4601885</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>phat</td>
<td>74</td>
<td>.2972973</td>
<td>.3052979</td>
<td>.000729</td>
<td>.8980594</td>
</tr>
<tr>
<td>idxhat</td>
<td>74</td>
<td>-1.678202</td>
<td>2.321509</td>
<td>-7.223107</td>
<td>2.175845</td>
</tr>
</tbody>
</table>

Because this is a logit model, we could obtain the predicted probabilities ourselves from the predicted index.

```
. generate phat2 = exp(idxhat)/(1+exp(idxhat))
```

but using predict without options is easier.

Example 5

For all models, predict attempts to produce a predicted value in the same metric as the dependent variable of the model. We have seen that for dichotomous outcome models, the default statistic produced by predict is the probability of a success. Similarly, for Poisson regression, the default statistic produced by predict is the predicted count for the dependent variable. You can always specify the xb option to obtain the linear combination of the coefficients with an observation’s \( x \) values (the inner product of the coefficients and \( x \) values). For poisson (without an explicit exposure), this is the natural log of the count.

```
. use http://www.stata-press.com/data/r13/airline, clear
. poisson injuries XYZowned
(output omitted)
```
We note that our “hand-computed” prediction of the count (exp_idx) matches what was produced by the default operation of predict.

If our model has an exposure-time variable, we can use predict to obtain the linear prediction with or without the exposure. Let’s verify what we are getting by obtaining the linear prediction with and without exposure, transforming these predictions to count predictions and comparing them with the default count prediction from predict. We must remember to multiply by the exposure time when using predict ..., nooffset.

Looking at the identical means and standard deviations for injhat, exp_idx, and exp_idxn, we see that we can reproduce the default computations of predict for poisson estimations. We have also demonstrated the relationship between the count predictions and the linear predictions with and without exposure.

---

**SE model scores**

**Example 6**

With most maximum likelihood estimators, predict can calculate equation-level scores. The first derivative of the log likelihood with respect to $x_j \beta$ is the equation-level score.
predict — Obtain predictions, residuals, etc., after estimation

. use http://www.stata-press.com/data/r13/auto, clear
(1978 Automobile Data)
. logistic foreign mpg weight
(output omitted)
. predict double sc, score
. summarize sc

<table>
<thead>
<tr>
<th>Variable</th>
<th>Obs</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>sc</td>
<td>74</td>
<td>-1.37e-12</td>
<td>.3533133</td>
<td>-.8760856</td>
<td>.8821309</td>
</tr>
</tbody>
</table>

See [P] _robust and [SVY] variance estimation for details regarding the role equation-level scores play in linearization-based variance estimators.

FAQ: Technical note

predict after some estimation commands, such as regress and cnsreg, allows the score option as a synonym for the residuals option.

Multiple-equation (ME) models

If you have not read the above discussion on using predict after SE models, please do so. With the exception of the ability to select specific equations to predict from, the use of predict after ME models follows almost the same form that it does for SE models.

Example 7

The details of prediction statistics that are specific to particular ME models are documented with the estimation command. If you are using ME commands that do not have separate discussions on obtaining predictions, read Obtaining predicted values in [R] mlogit postestimation, even if your interest is not in multinomial logistic regression. As a general introduction to the ME models, we will demonstrate predict after sureg:

. use http://www.stata-press.com/data/r13/auto, clear
(1978 Automobile Data)
. sureg (price foreign displ) (weight foreign length)

Seemingly unrelated regression

<table>
<thead>
<tr>
<th>Equation</th>
<th>Obs</th>
<th>Parms</th>
<th>RMSE</th>
<th>&quot;R-sq&quot;</th>
<th>chi2</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>price</td>
<td>74</td>
<td>2</td>
<td>2202.447</td>
<td>0.4348</td>
<td>45.21</td>
<td>0.0000</td>
</tr>
<tr>
<td>weight</td>
<td>74</td>
<td>2</td>
<td>245.5238</td>
<td>0.8988</td>
<td>658.85</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

|               | Coef.  | Std. Err. | z     | P>|z|   | [95% Conf. Interval] |
|---------------|--------|-----------|-------|-------|----------------------|
| price foreign | 3137.894 | 697.3805 | 4.50  | 0.0000 | 1771.054 | 4504.735 |
| displacement  | 23.06938 | 3.443212 | 6.70  | 0.0000 | 16.32081 | 29.81795 |
| _cons         | 680.8438 | 859.8142 | 0.79  | 0.428  | -1004.361 | 2366.049 |
| weight foreign| -154.883 | 75.3204 | -2.06 | 0.040  | -302.5082 | -7.257674 |
| length        | 30.67594 | 1.531981 | 20.02 | 0.0000 | 27.67331 | 33.67856 |
| _cons         | -2699.498 | 302.3912 | -8.93 | 0.0000 | -3292.173 | -2106.822 |
predict — Obtain predictions, residuals, etc., after estimation

`sureg` estimated two equations, one called `price` and the other `weight`; see `*[R]* `sureg`.

```
. predict pred_p, equation(price)
(option xb assumed; fitted values)
. predict pred_w, equation(weight)
(option xb assumed; fitted values)
. summarize price pred_p weight pred_w
```

```
<table>
<thead>
<tr>
<th>Variable</th>
<th>Obs</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>price</td>
<td>74</td>
<td>6165.257</td>
<td>2949.496</td>
<td>3291</td>
<td>15906</td>
</tr>
<tr>
<td>pred_p</td>
<td>74</td>
<td>6165.257</td>
<td>1678.805</td>
<td>2664.81</td>
<td>10485.33</td>
</tr>
<tr>
<td>weight</td>
<td>74</td>
<td>3019.459</td>
<td>777.1936</td>
<td>1760</td>
<td>4840</td>
</tr>
<tr>
<td>pred_w</td>
<td>74</td>
<td>3019.459</td>
<td>726.0468</td>
<td>1501.602</td>
<td>4447.996</td>
</tr>
</tbody>
</table>
```

You may specify the equation by name, as we did above, or by number: `equation(#1)` means the same thing as `equation(price)` in this case.

## ME model scores

### Example 8

For ME models, `predict` allows you to specify a stub when generating equation-level score variables. `predict` generates new variables using this stub by appending an equation index. Depending upon the command, the index will start with 0 or 1. Here is an example where `predict` starts indexing the score variables with 0.

```
. ologit rep78 mpg weight
(output omitted)
. predict double sc*, scores
. summarize sc*
```

```
<table>
<thead>
<tr>
<th>Variable</th>
<th>Obs</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>sc0</td>
<td>69</td>
<td>-1.33e-11</td>
<td>.5337363</td>
<td>-.9854088</td>
<td>.921433</td>
</tr>
<tr>
<td>sc1</td>
<td>69</td>
<td>-7.69e-13</td>
<td>.186919</td>
<td>-.2738537</td>
<td>.9854088</td>
</tr>
<tr>
<td>sc2</td>
<td>69</td>
<td>-2.87e-11</td>
<td>.4061637</td>
<td>-.5188487</td>
<td>1.130178</td>
</tr>
<tr>
<td>sc3</td>
<td>69</td>
<td>-1.04e-10</td>
<td>.5315368</td>
<td>-1.067351</td>
<td>.8194842</td>
</tr>
<tr>
<td>sc4</td>
<td>69</td>
<td>1.47e-10</td>
<td>.360525</td>
<td>-.921433</td>
<td>.6140182</td>
</tr>
</tbody>
</table>
```

Although it involves much more typing, we could also specify the new variable names individually.

```
. predict double (sc_xb sc_1 sc_2 sc_3 sc_4), scores
. summarize sc_*
```

```
<table>
<thead>
<tr>
<th>Variable</th>
<th>Obs</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>sc_xb</td>
<td>69</td>
<td>-1.33e-11</td>
<td>.5337363</td>
<td>-.9854088</td>
<td>.921433</td>
</tr>
<tr>
<td>sc_1</td>
<td>69</td>
<td>-7.69e-13</td>
<td>.186919</td>
<td>-.2738537</td>
<td>.9854088</td>
</tr>
<tr>
<td>sc_2</td>
<td>69</td>
<td>-2.87e-11</td>
<td>.4061637</td>
<td>-.5188487</td>
<td>1.130178</td>
</tr>
<tr>
<td>sc_3</td>
<td>69</td>
<td>-1.04e-10</td>
<td>.5315368</td>
<td>-1.067351</td>
<td>.8194842</td>
</tr>
<tr>
<td>sc_4</td>
<td>69</td>
<td>1.47e-10</td>
<td>.360525</td>
<td>-.921433</td>
<td>.6140182</td>
</tr>
</tbody>
</table>
```
Methods and formulas

Denote the previously estimated coefficient vector as $\mathbf{b}$ and its estimated variance matrix as $\mathbf{V}$. predict works by recalling various aspects of the model, such as $\mathbf{b}$, and combining that information with the data currently in memory. Let’s write $\mathbf{x}_j$ for the $j$th observation currently in memory.

The predicted value (xb option) is defined as $\hat{y}_j = \mathbf{x}_j \mathbf{b} + \text{offset}_j$

The standard error of the prediction (the stdp option) is defined as $s_{p_j} = \sqrt{\mathbf{x}_j \mathbf{V} \mathbf{x}_j'}$

The standard error of the difference in linear predictions between equations 1 and 2 is defined as

$$s_{dp_j} = \left\{ (\mathbf{x}_1j, -\mathbf{x}_2j, 0, \ldots, 0) \mathbf{V} (\mathbf{x}_1j, -\mathbf{x}_2j, 0, \ldots, 0)' \right\}^{\frac{1}{2}}$$

See the individual estimation commands for information about calculating command-specific predict statistics.

Also see

[R] predictnl — Obtain nonlinear predictions, standard errors, etc., after estimation
[P] _predict — Obtain predictions, residuals, etc., after estimation programming command
[U] 20 Estimation and postestimation commands