# Title

> **meta mvregress postestimation** — Postestimation tools for meta mvregress

## Postestimation commands

The following postestimation commands are of special interest after `meta mvregress`:

| Command | Description |
|---|---|
| estat heterogeneity | compute multivariate heterogeneity statistics |
| estat recovariance | display the estimated random-effects covariance matrix |
| estat sd | display variance components as standard deviations and correlations |

The following standard postestimation commands are also available:

| Command | Description |
|---|---|
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| 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 coefficients |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | predictions and their SEs, leverage statistics, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| 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 predictions, standard errors, fitted values, residuals, and standardized residuals. After random-effects multivariate meta-regression, you can also obtain estimates of random effects and their standard errors or variances and covariances.

## Menu for predict

Statistics > Postestimation

## Syntax for predict

*Syntax for obtaining predictions other than best linear unbiased predictions (BLUPs) of random effects*

predict [ *type* ] *newvar* [ *if* ] [ *in* ] [ , *statistic* <u>fixed</u>only <u>depvar</u>(*depname* | ##) ]

*Syntax for obtaining BLUPs of random effects and the BLUPs' standard errors*

predict [ *type* ] { *stub** | *newvarlist* } [ *if* ] [ *in* ], <u>ref</u>fects [ reses(*resesspec*)

revce(*stub** | *newvarlist*) ]

| *statistic* | Description |
|---|---|
| **Main** | |
| xb | linear prediction for the fixed portion of the model only; the default |
| stdp | standard error of the fixed-portion linear prediction |
| <u>fitted</u> | fitted values, fixed-portion linear prediction plus contributions based on predicted random effects |
| <u>residuals</u> | residuals, response minus fitted values |
| <u>rstandard</u> | standardized residuals |

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
```

xb, the default, calculates the linear predictor $\mathbf{x}_j \widehat{\beta}_i$ for the specified dependent variable. For the random-effects multivariate meta-regression, this corresponds to the fixed portion of the linear predictor based on the estimated regression coefficients. That is, this is equivalent to fixing all random effects in the model to a theoretical mean value of 0.

stdp calculates the standard error of the linear predictor for the specified dependent variable.

`fitted` calculates the fitted values for the specified dependent variable. With fixed-effects multivariate meta-regression or with random-effects multivariate meta-regression when option `fixedonly` is also specified, this option is equivalent to xb. For random-effects multivariate meta-regression without `fixedonly`, it calculates $\mathbf{x}_j\widehat{\boldsymbol{\beta}}_i + \widehat{u}_{ij}$, which is equal to the fixed portion of the linear predictor plus predicted random effects for the specified dependent variable.

`residuals` calculates the residuals, which are equal to the responses minus the fitted values for the specified dependent variable. With fixed-effects multivariate meta-regression or with random-effects multivariate meta-regression when option `fixedonly` is also specified, it calculates $\widehat{\theta}_{ij} - \mathbf{x}_j\widehat{\boldsymbol{\beta}}_i$. The former is known as marginal residuals in the context of the random-effects model. For random-effects multivariate meta-regression without `fixedonly`, this option calculates $\widehat{\theta}_{ij} - (\mathbf{x}_j\widehat{\boldsymbol{\beta}}_i + \widehat{u}_{ij})$, which are known as conditional residuals.

`rstandard` calculates the standardized residuals that, for the specified dependent variable #i, are equal to the $i$th component of the residuals multiplied by the inverse square root of the within-study covariance matrices, $\boldsymbol{\Lambda}_j^{-1/2}\widehat{\boldsymbol{\epsilon}}_j$. With fixed-effects multivariate meta-regression or with random-effects multivariate meta-regression with `fixedonly`, $\widehat{\epsilon}_{ij} = \widehat{\theta}_{ij} - \mathbf{x}_j\widehat{\boldsymbol{\beta}}_i$. For random-effects multivariate meta-regression without `fixedonly`, $\widehat{\epsilon}_{ij} = \widehat{\theta}_{ij} - (\mathbf{x}_j\widehat{\boldsymbol{\beta}}_i + \widehat{u}_{ij})$.

`fixedonly` specifies that all random effects be set to zero, equivalent to using only the fixed portion of the model. This option is allowed only with options `fitted`, `residuals`, and `rstandard`.

`depvar(`*depname* | `##)` specifies the dependent (outcome) variable of interest.

> `depvar()` is filled in with one *depname* or `##` for the xb, stdp, fitted, residuals, and rstandard options. `depvar(#1)` would mean the calculation is to be made for the first outcome, `depvar(#2)` would mean the second, and so on. You could also refer to the outcomes by their variable names. `depvar(sensitivity)` would refer to the dependent variable named `sensitivity` and `depvar(specificity)` to the dependent variable named `specificity`.

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

`reffects` calculates BLUPs of the random effects. You must specify $d$ new variables, where $d$ is the number of random-effects terms in the model, which is equal to the number of *depvars*. However, it is much easier to just specify *stub*\* and let Stata name the variables *stub*1, *stub*2, ..., *stub*d for you.

`reses(`*resesspec*`)` calculates the standard errors of the random effects; see the [reffects](#) option. This option may not be combined with option `revce()`. The syntax for *resesspec* is

> *stub*\* | *newvarlist* [ , <u>compar</u>ative | <u>diag</u>nostic ]

> `comparative`, the default, computes comparative random-effects standard errors. For linear models, these correspond to posterior standard deviations of random effects and to standard errors of marginal prediction errors $\widehat{\mathbf{u}}_j - \mathbf{u}_j$. These standard errors are used for inference about the random effects.

> `diagnostic` computes diagnostic random-effects standard errors. These correspond to marginal standard errors of BLUPs, $\mathrm{SE}(\widehat{\mathbf{u}}_j)$. These standard errors are used for model diagnostics.

> You must specify $d$ new variables, where $d$ is the number of random-effects terms in the model. However, it is much easier to just specify *stub*\* and let Stata name the variables *stub*1, *stub*2, ..., *stub*d for you. The new variables will have the same storage type as the corresponding random-effects variables.

`revce(`*stub*\* | *newvarlist*`)` calculates the variances and covariances of the BLUPs of the random effects; see option [reffects](#). This option may not be combined with option `reses()`.

You must specify $q = d(d+1)/2$ new variables, where $d$ is the number of random-effects terms in the model. However, it is much easier to just specify *stub** and let Stata name the variables *stub*1, *stub*2, ..., *stub*q for you. The new variables will have the same storage type as the corresponding random-effects variables.

The reffects and reses() options generate $d$ new variables at once. The random effects (or standard errors) contained in the generated variables correspond to the order in which the dependent variables, *depvars*, are specified with meta mvregress. Option revce() generates $d(d+1)/2$ variables at once. The generated variables correspond to the same order in which you specify variables in option wcovvariables() with meta mvregress. Still, examining the variable labels of the generated variables (with the describe command, for instance) can be useful in deciphering which variables correspond to which terms in the model.

# margins

## Description for margins

margins estimates margins of response for linear predictions.

## Menu for margins

Statistics > Postestimation

## Syntax for margins

margins [ *marginlist* ] [ , *options* ]

margins [ *marginlist* ] , predict(*statistic* ...) [ *options* ]

| *statistic* | Description |
|---|---|
| xb | linear predictor for the fixed portion of the model only; the default |
| fitted | fitted values; implies fixedonly |
| stdp | not allowed with margins |
| residuals | not allowed with margins |
| standard | not allowed with margins |
| reffects | not allowed with margins |

xb and fitted default to the first outcome.

Statistics not allowed with margins are functions of stochastic quantities other than e(b).

For the full syntax, see [R] **margins**.

# Remarks and examples

Various predictions and statistics are available after fitting a multivariate meta-regression using meta mvregress. For a random-effects multivariate meta-regression, calculation centers around obtaining BLUPs of the random effects. Random effects are not estimated when the model is fit but rather must be predicted after the estimation of the model parameters. The estimates of the random effects are in turn used to obtain predicted values and residuals for each dependent variable. These are useful for checking model assumptions and may be used in general as model-building tools.

▷ Example 1: Obtaining predictions of random effects

In example 7 of [META] **meta mvregress**, we conducted a multivariate meta-analysis of the logit-transformed sensitivity and specificity for the tumor marker telomerase. We refit that model, but here we assume an independent random-effects covariance structure.

```
. use https://www.stata-press.com/data/r18/telomerase
(Telomerase for diagnosing primary bladder cancer)

. meta mvregress y*, wsevariables(s*) wcorrelation(0)
> random(reml, covariance(independent))

Performing EM optimization ...

Performing gradient-based optimization:
Iteration 0:   Log restricted-likelihood = -30.469414
Iteration 1:   Log restricted-likelihood = -27.928351
Iteration 2:   Log restricted-likelihood = -27.586358
Iteration 3:   Log restricted-likelihood = -27.456381
Iteration 4:   Log restricted-likelihood = -27.456281
Iteration 5:   Log restricted-likelihood = -27.456281
```

Multivariate random-effects meta-analysis          Number of obs     =         20
Method: REML                                       Number of studies =         10
                                                   Obs per study:
                                                                 min =          2
                                                                 avg =        2.0
                                                                 max =          2
                                                   Wald chi2(0)      =          .
Log restricted-likelihood = -27.456281             Prob > chi2       =          .

|  | Coefficient | Std. err. | z | P>\|z\| | [95% conf. interval] | |
|---|---|---|---|---|---|---|
| y1 | | | | | | |
| _cons | 1.154606 | .1855479 | 6.22 | 0.000 | .7909387 | 1.518273 |
| y2 | | | | | | |
| _cons | 1.963801 | .5413727 | 3.63 | 0.000 | .9027297 | 3.024872 |

Test of homogeneity: Q_M = chi2(18) = 90.87                    Prob > Q_M = 0.0000

| Random-effects parameters | Estimate |
|---|---|
| Independent: | |
| sd(y1) | .4310376 |
| sd(y2) | 1.544806 |

Below, we predict the random effects using `predict, reffects` and obtain their diagnostic standard errors by specifying the `reses(, diagnostic)` option. Because we have two random-effects in our model (one for each outcome), we need to specify two new variable names with `predict` and two new variable names within `reses()`. Alternatively, it is much easier to specify a stubname, say, `u*`, and `predict` will construct variables u1 and u2 for you. This way you do not have to worry about specifying the correct number of variables with `predict`. We will also specify a stubname, `u_se*`, within `reses()`. And we will use suboption `diagnostic` of the `reses()` option to request the diagnostic standard errors instead of the default comparative standard errors. The diagnostic standard errors are used for model diagnostics (Goldstein 2011; Skrondal and Rabe-Hesketh 2009).

```
. predict double u*, reffects reses(se_u*, diagnostic)
. list trialnum u* se_u*, mean(u1 u2) labvar(trialnum)
```

|  | trialnum | u1 | u2 | se_u1 | se_u2 |
|---|---|---|---|---|---|
| 1. | 1 | -.00803546 | .87413065 | .29790195 | 1.2328506 |
| 2. | 2 | .10980179 | -.56421535 | .25481757 | 1.3471946 |
| 3. | 3 | .39364529 | -1.2526626 | .3395802 | 1.4227301 |
| 4. | 4 | -.36519382 | 1.1525847 | .2988524 | 1.3641041 |
| 5. | 5 | -.20599987 | 2.0787496 | .33418853 | 1.2382177 |
| 6. | 6 | .16425798 | -.53113834 | .30890465 | 1.3953169 |
| 7. | 7 | -.64318066 | .67059071 | .32901024 | 1.0915151 |
| 8. | 8 | .16670084 | .18934479 | .28423823 | 1.3202025 |
| 9. | 9 | .12138806 | .26416706 | .23461556 | 1.3593008 |
| 10. | 10 | .26661585 | -2.8815512 | .29607045 | 1.4000379 |
| | Mean | -7.772e-17 | -2.220e-16 | | |

We listed the random-effects variables u1 and u2 with their corresponding standard error variables se_u1 and se_u2. The random effects are study-specific deviations from the overall mean effect size. For example, for study 2 and outcome y1, the predicted logit-sensitivity is 0.1098 higher than the overall logit-sensitivity $\widehat{\theta}_1 = 1.155$. We also show the mean (average) of variables u1 and u2 at the bottom. Note that the means of these study-specific deviations are close to 0, which is expected because $E(u_{ij}) = 0$.

Instead of reses(), you may specify the revce() option to obtain the full variance–covariance matrix of the predicted random effects instead of only the standard errors.

Let's use the predicted random effects and their diagnostic standard errors to compute standardized random-effects variables, ustan1 and ustan2, to check for outliers. We will use the qnorm command (see [R] **Diagnostic plots**) to obtain the normal quantile plot.

```
. generate double ustan1 = u1/se_u1
. generate double ustan2 = u2/se_u2
. label variable ustan1 "Std. predicted random effects u1"
. label variable ustan2 "Std. predicted random effects u2"
. qnorm ustan1, mlabel(trialnum) name(gr_u1) xtitle("Inverse normal")
. qnorm ustan2, mlabel(trialnum) name(gr_u2) xtitle("Inverse normal")
. graph combine gr_u1 gr_u2
```
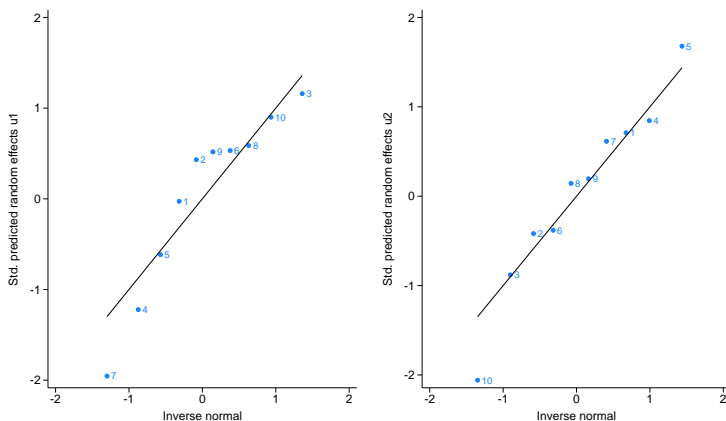


From the plots, study 7 for outcome y1 and study 10 for outcome y2 appear to be outliers. Study 7 has a sensitivity value of `invlogit(y1[7])` = `invlogit(.1866)` $\approx 54.65\%$, which is substantively lower than sensitivities of the other studies. Similarly, study 10 has a specificity value of `invlogit(y2[10])` = `invlogit(-1.145)` $\approx 24.14\%$.

◁

▷ Example 2: Checking model fit

Continuing with example 1, we specify the `fitted` option to obtain the fitted values and plot them against the observed values of y1 and y2. By default, for outcome $i$ and study $j$, the mean effect size conditional on the estimated random effects, $\mathbf{x}_j\widehat{\boldsymbol{\beta}}_i + \widehat{u}_{ij}$, is computed for multivariate meta-regression. In our case of multivariate meta-analysis, the fitted values are $\widehat{\theta}_i + \widehat{u}_{ij}$. Predicted values based on only the fixed portion of the mean effect size, $\mathbf{x}_j\widehat{\boldsymbol{\beta}}_i$ (or $\widehat{\theta}_i$ in our example), may be computed by specifying the `fixedonly` option.

```
. predict double fit1, fitted depvar(#1)
. predict double fit2, fitted depvar(#2)
. twoway (scatter fit1 y1, mlabel(trialnum))
>        (function y = x, range(y1)),
> name(graph1) legend(off) xtitle(Observed values y1) ytitle(Fitted values)
. twoway (scatter fit2 y2, mlabel(trialnum))
>        (function y = x, range(y2)),
> name(graph2) legend(off) xtitle(Observed values y2) ytitle(Fitted values)
. graph combine graph1 graph2
```
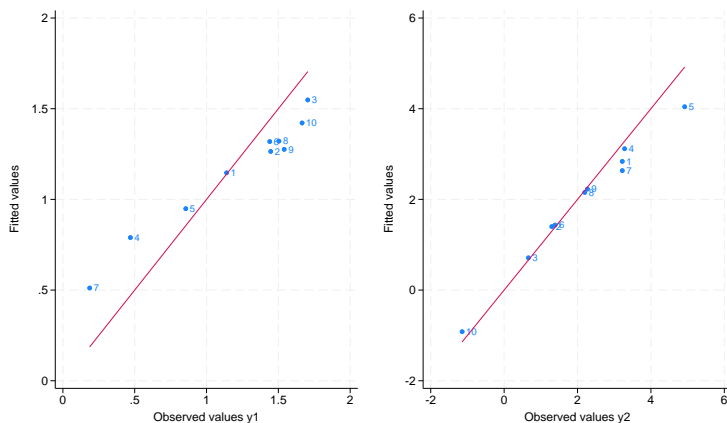


In the above code, we computed the fitted values for each dependent variable using `predict, fitted depvar()`. We then produced scatterplots of the fitted values versus the observed values of y1 and y2, respectively. We added a reference line $y = x$ to assess goodness of fit. Studies that are close to the reference line have their fitted values close to the observed values. Overall, it seems that goodness of fit is better for outcome y2.

You could also use the `rstandard` option with `predict` to compute standardized residuals. In theory, the standardized residuals are useful for checking the normality assumption. But because the number of studies is often small (for instance, only 10 in our example), standardized residuals are typically of limited use in practice with multivariate meta-regression.

◁

# Methods and formulas

Methods and formulas are presented under the following headings:

> *Random-effects multivariate meta-regression*
> *Fixed-effects multivariate meta-regression*

The following formulas are used by `predict`. The notation is based on *Methods and formulas* of [META] **meta mvregress**.

## Random-effects multivariate meta-regression

Consider a random-effects multivariate meta-regression

$$\widehat{\boldsymbol{\theta}}_j = \mathbf{X}_j \boldsymbol{\beta} + \mathbf{u}_j + \boldsymbol{\epsilon}_j, \quad \boldsymbol{\epsilon}_j \sim N_d \left(\mathbf{0}, \boldsymbol{\Lambda}_j\right)$$

where $\mathbf{u}_j = (u_{1j}, u_{2j}, \ldots, u_{dj})' \sim N_d(\mathbf{0}, \boldsymbol{\Sigma})$ $(j = 1, \ldots, K)$ define random effects, $\mathbf{x}_j = (1, x_{1j}, \ldots, x_{p-1,j})$ is a $1 \times p$ vector of moderators (covariates), $\mathbf{X}_j = \mathbf{x}_j \otimes I_d$ is a $d \times dp$ design matrix, and $\boldsymbol{\beta} = (\boldsymbol{\beta}_1', \boldsymbol{\beta}_2', \ldots, \boldsymbol{\beta}_d')'$ is a $dp \times 1$ vector of unknown regression coefficients.

The random-effects estimator for regression coefficients is

$$\widehat{\boldsymbol{\beta}}^* = \left(\sum_{j=1}^{K} \mathbf{X}_j' \mathbf{W}_j^* \mathbf{X}_j\right)^{-1} \sum_{j=1}^{K} \mathbf{X}_j' \mathbf{W}_j^* \widehat{\boldsymbol{\theta}}_j$$

where $\mathbf{W}_j^* = (\widehat{\boldsymbol{\Sigma}} + \boldsymbol{\Lambda}_j)^{-1}$.

The formulas below apply to outcome $i$, as specified in the `depvar()` option, for $i = 1, \ldots, d$ and study $j$ for $j = 1, \ldots, K$.

The fixed portion of the linear predictor (option `xb`) is $\mathbf{x}_j \widehat{\boldsymbol{\beta}}_i^*$.

The estimated standard error of the fixed portion of the linear predictor (option `stdp`) is

$$\widehat{\text{SE}}\left(\mathbf{x}_j \widehat{\boldsymbol{\beta}}_i^*\right) = \sqrt{\left\{\mathbf{X}_j \left(\sum_{l=1}^{K} \mathbf{X}_l' \mathbf{W}_l^* \mathbf{X}_l\right)^{-1} \mathbf{X}_j'\right\}_{ii}}$$

where $\mathbf{A}_{ii}$ denotes the $i$th diagonal of matrix $\mathbf{A}$.

The BLUP of a $d \times 1$ random-effects vector $\mathbf{u}_j$ (option `reffects`) is

$$\widehat{\mathbf{u}}_j = \widehat{\boldsymbol{\Sigma}} \mathbf{W}_j^* \left(\widehat{\boldsymbol{\theta}}_j - \mathbf{X}_j \widehat{\boldsymbol{\beta}}^*\right)$$

When the `reses()` option is specified with `reffects`, the estimated covariance matrix of $\widehat{\mathbf{u}}_j - \mathbf{u}_j$ is computed:

$$\widehat{\text{Var}}\left(\widehat{\mathbf{u}}_j - \mathbf{u}_j\right) = \widehat{\boldsymbol{\Sigma}} - \widehat{\boldsymbol{\Sigma}} \mathbf{W}_j^* \left\{\left(\mathbf{W}_j^*\right)^{-1} - \mathbf{X}_j \text{Var}(\widehat{\boldsymbol{\beta}}^*) \mathbf{X}_j'\right\} \mathbf{W}_j^* \widehat{\boldsymbol{\Sigma}}$$

The comparative standard errors of the random effects can be obtained by taking the square root of the diagonal elements of $\widehat{\text{Var}}(\widehat{\mathbf{u}}_j - \mathbf{u}_j)$.

If the `diagnostic` suboption is specified within `reses()`, then the estimated covariance matrix of $\widehat{\mathbf{u}}_j$ is computed:

$$\widehat{\mathrm{Var}}\left(\widehat{\mathbf{u}}_j\right) = \widehat{\boldsymbol{\Sigma}}\mathbf{W}_j^*\left\{\left(\mathbf{W}_j^*\right)^{-1} - \mathbf{X}_j\mathrm{Var}(\widehat{\boldsymbol{\beta}}^*)\mathbf{X}_j'\right\}\mathbf{W}_j^*\widehat{\boldsymbol{\Sigma}}$$

The diagnostic standard errors of the random effects can be obtained by taking the square root of the diagonal elements of $\widehat{\mathrm{Var}}\left(\widehat{\mathbf{u}}_j\right)$.

See Goldstein (2011), Skrondal and Rabe-Hesketh (2009), and Rabe-Hesketh and Skrondal (2022) for more details.

The `revce()` option generates variables corresponding to the variances and covariances in the $\widehat{\mathrm{Var}}(\widehat{\mathbf{u}}_j)$ matrix.

The fitted value (option `fitted`) is

$$\widetilde{\theta}_{ij} = \mathbf{x}_j\widehat{\boldsymbol{\beta}}_i^* + \widehat{u}_{ij}$$

The residual (option `residuals`) is

$$\widehat{\epsilon}_{ij} = \widehat{\theta}_{ij} - \widetilde{\theta}_{ij}$$

The standardized residual (option `rstandard`) is the $i$th element of the $d \times 1$ vector $\mathbf{\Lambda}_j^{-1/2}\widehat{\boldsymbol{\epsilon}}_j$,

$$\widetilde{\epsilon}_{ij} = \left(\mathbf{\Lambda}_j^{-1/2}\widehat{\boldsymbol{\epsilon}}_j\right)_i$$

where $\widehat{\boldsymbol{\epsilon}}_j = (\widehat{\epsilon}_{1j}, \widehat{\epsilon}_{2j}, \ldots, \widehat{\epsilon}_{dj})'$.

When the `fixedonly` option is specified, the formulas for the fitted values, residuals, and standardized residuals are adjusted by replacing the value of $\widehat{u}_{ij}$ with 0.

## Fixed-effects multivariate meta-regression

Consider a fixed-effects multivariate meta-regression

$$\widehat{\boldsymbol{\theta}}_j = \mathbf{X}_j\boldsymbol{\beta} + \boldsymbol{\epsilon}_j, \quad \boldsymbol{\epsilon}_j \sim N_d(\mathbf{0}, \mathbf{\Lambda}_j)$$

with a $d \times dp$ design matrix $\mathbf{X}_j = \mathbf{x}_j \otimes I_d$ and a $dp \times 1$ vector $\boldsymbol{\beta} = (\boldsymbol{\beta}_1', \boldsymbol{\beta}_2', \ldots, \boldsymbol{\beta}_d')'$ of all unknown regression coefficients.

Let $\mathbf{W}_j = \mathbf{\Lambda}_j^{-1}$. The fixed-effects estimator for regression coefficients is

$$\widehat{\boldsymbol{\beta}} = \left(\sum_{j=1}^{K}\mathbf{X}_j'\mathbf{W}_j\mathbf{X}_j\right)^{-1}\sum_{j=1}^{K}\mathbf{X}_j'\mathbf{W}_j\widehat{\boldsymbol{\theta}}_j$$

The linear predictor (option `xb`) is $\mathbf{x}_j\widehat{\boldsymbol{\beta}}$.

The estimated standard error of the linear predictor (option `stdp`) is

$$\widehat{\mathrm{SE}}\left(\mathbf{x}_j\widehat{\boldsymbol{\beta}}_i\right) = \sqrt{\mathbf{X}_j\left(\sum_{l=1}^{K}\mathbf{X}_l'\mathbf{W}_l\mathbf{X}_l\right)^{-1}\mathbf{X}_j'}$$

The fitted value (option `fitted`) is the same as the linear predictor:

$$\widetilde{\theta}_{ij} = \mathbf{x}_j\widehat{\boldsymbol{\beta}}_i$$

The residual (option `residuals`) is

$$\widehat{\epsilon}_{ij} = \widehat{\theta}_{ij} - \widetilde{\theta}_{ij}$$

The standardized residual (option `rstandard`) is the $i$th element of a $d \times 1$ vector $\boldsymbol{\Lambda}_j^{-1/2}\widehat{\boldsymbol{\epsilon}}_j$,

$$\widetilde{\epsilon}_{ij} = \left(\boldsymbol{\Lambda}_j^{-1/2}\widehat{\boldsymbol{\epsilon}}_j\right)_i$$

# References

Goldstein, H. 2011. *Multilevel Statistical Models*. 4th ed. Chichester, UK: Wiley.

Rabe-Hesketh, S., and A. Skrondal. 2022. *Multilevel and Longitudinal Modeling Using Stata*. 4th ed. College Station, TX: Stata Press.

Skrondal, A., and S. Rabe-Hesketh. 2009. Prediction in multilevel generalized linear models. *Journal of the Royal Statistical Society, Series A* 172: 659–687. https://doi.org/10.1111/j.1467-985X.2009.00587.x.

# Also see

[META] **meta mvregress** — Multivariate meta-regression

[META] **meta** — Introduction to meta

[META] **Glossary**

[META] **Intro** — Introduction to meta-analysis

[U] **20 Estimation and postestimation commands**