# Multilevel mixed-effects models

Whether the groupings in your data arise in a nested fashion (students nested in schools and schools nested in districts) or in a nonnested fashion (regions crossed with occupations), you can fit a multilevel model to account for the lack of independence within these groups.

Fit models for continuous, binary, count, ordinal, and survival outcomes. Fit linear, generalized linear, and nonlinear models.

Estimate variances of random intercepts and random coefficients. Compute intraclass correlations. Predict random effects. Estimate relationships that are population averaged over the random effects. Perform Bayesian analysis. And much more.
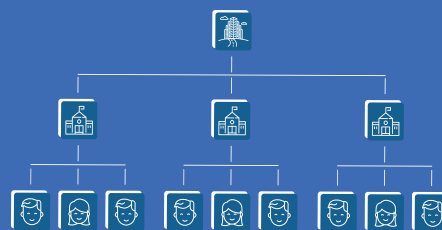
- **Models for**
  - Continuous outcomes
  - Binary outcomes
  - Count outcomes
  - Categorical outcomes
  - Ordinal outcomes
  - Survival-time outcomes

- **Types of models**
  - Linear, generalized linear, and nonlinear
  - Two-, three-, and higher-level models
  - Nested (hierarchical) models
  - Crossed models
  - Mixed models
  - Balanced and unbalanced designs

- **Estimated effects**
  - Random intercepts
  - Random coefficients (slopes)
  - Variances of random effects (variance components)
  - Fixed effects (regression coefficients)

- **Covariance structures**
  - Identity
  - Independent
  - Exchangeable
  - Unstructured

- **Error (residual) structures**
  - Independent
  - Exchangeable
  - Autoregressive
  - Moving average
  - Exponential
  - Banded
  - Toeplitz
  - Unstructured

- **Estimation methods**
  - Maximum likelihood (ML)
  - Restricted maximum likelihood (REML)
  - Mean-variance or mode-curvature adaptive Gauss–Hermite quadrature
  - Nonadaptive Gauss–Hermite quadrature
  - Laplacian approximation

- **Inferences**
  - Expected means, probabilities, counts
  - Marginal effects
  - Conditional or population averaged
  - Tests, contrasts, and pairwise comparisons

- **Adjustments for small-sample inference**
  - Kenward–Roger
  - Satterthwaite
  - ANOVA
  - Repeated-measures ANOVA
  - Residual

- **Account for survey design**
  - Weights at each level
  - Clustering
  - Stratification

- **Estimates of random effects**
  - BLUPs for linear models
  - Empirical Bayes posterior means
  - Empirical Bayes posterior modes
  - With standard errors

- **Bayesian analysis**
  - Incorporate prior knowledge
  - Estimate distributions of random effects
  - Compute probability for any hypothesis of interest
  - Linear and generalized linear models
  - Nonlinear, joint, and SEM-like models

Linear models

When you are modeling continuous outcomes—weight, log of income, test scores, and the like—the **mixed** command can fit a variety of multilevel mixed-effects linear models.

Fit your model

You can fit a two-level model (observations nested in **id**) with random intercepts for **id** and with fixed coefficients for continuous variables **x1** and **x2** and categorical variable **a**.

```
⊙ Viewer - view mlmm1.smcl                              —  □  ×
view mlmm1.smcl  ✕
+                                    Dialog ▾  Also see ▾  Jump to ▾

. mixed y x1 x2 i.a || id:, allbase

Performing EM optimization ...

Performing gradient-based optimization:
Iteration 0:  Log likelihood =  822.62635
Iteration 1:  Log likelihood =  822.62635

Computing standard errors ...

Mixed-effects ML regression            Number of obs    =     816
Group variable: id                     Number of groups =      48
                                       Obs per group:
                                                    min =      17
                                                    avg =    17.0
                                                    max =      17
                                       Wald chi2(5)     = 4648.06
Log likelihood =  822.62635            Prob > chi2      =  0.0000

         y │ Coefficient  Std. err.     z    P>|z|    [95% conf. interval]

        x1 │   .4774876   .0351938   13.57   0.000     .4085089   .5464662
        x2 │   .4979791   .0197476   25.22   0.000     .4592745   .5366837

         a │
         1 │          0  (base)
         2 │    .135029   .0213549    6.32   0.000     .0931741   .1768839
         3 │   .2148923   .0276171    7.78   0.000     .1607638   .2690209
         4 │   .3061946   .0373997    8.19   0.000     .2328925   .3794967

      _cons │   2.298825   .2314329    9.93   0.000     1.845225   2.752425

  Random-effects parameters │   Estimate   Std. err.    [95% conf. interval]

id: Identity                 │
               var(_cons)    │   .0201472   .0042906      .013272   .0305837

             var(Residual)   │   .0061479   .0003142     .0055619   .0067957

LR test vs. linear model: chibar2(01) = 859.26      Prob >= chibar2 = 0.0000
                                                          CAP  NUM  INS
```

Two-level model with random slopes for **x1**

```
. mixed y x1 x2 i.a || id: x1
```

Unstructured covariance for random slopes and random intercepts

```
. mixed y x1 x2 i.a || id: x1,
        covariance(unstructured)
```

Autoregressive residual variance structure

```
. mixed y x1 x2 i.a || id,
        residuals(ar 1, t(time))
```

REML estimation

```
. mixed y x1 x2 i.a || id:, reml
```

Kenward–Roger adjustment for small-sample inference

```
. mixed y x1 x2 i.a || id:, reml dfmethod(kroger)
```

Three-level model, **id** nested in **group**

```
. mixed y x1 x2 i.a || group: || id:
```

Three-level model with random slopes

```
. mixed y x1 x2 i.a || group: x1 || id: x2
```

Four-level model

```
. mixed y x1 x2 i.a || group: || subgroup: || id:
```
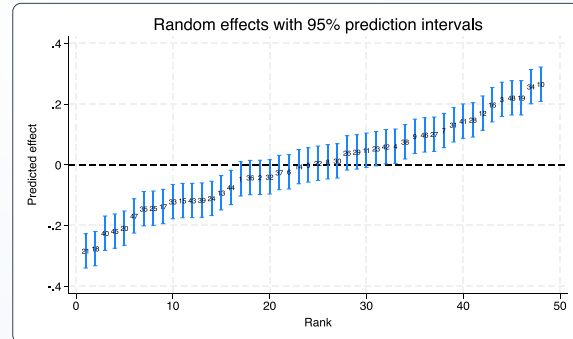
Crossed effects for **grp1** and **grp2**

```
. mixed y x1 x2 i.a || _all: R.grp1 || grp2:
```

Explore the results

Predict random effects (BLUPs) and their standard errors

```
. predict re, reffects reses(re_se)
```


Random effects with 95% prediction intervals

Estimate intraclass correlations

```
. estat icc
```

Perform Wald tests of coefficients

```
. test x1=1
```

Compare models

```
. lrtest model1 model2
```

Estimate and plot marginal means

```
. margins a
. marginsplot
```
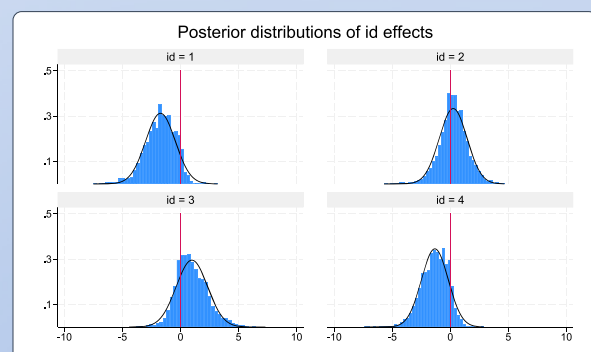

Expected marginal means with 95% CIs
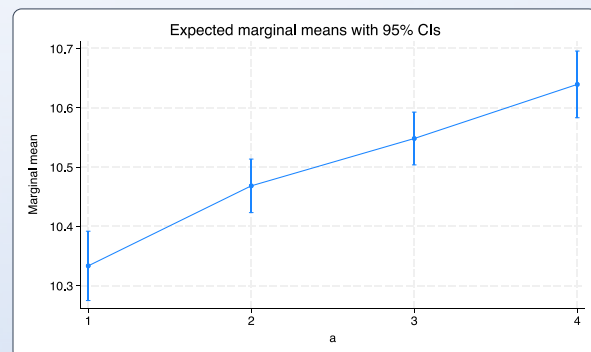
Bayesian estimation

Bayesian two-level model with random intercepts

```
. bayes: mixed y x1 x2 i.a || id:
```

Bayesian multilevel linear models

```
. bayes, priors ...: mixed ...
. bayesgraph histogram ...
```


Posterior distributions of id effects

Generalized linear models

When your outcome is not continuous—pass or fail a test, number of doctor visits, survival time, etc.—the **me** commands can fit logit, probit, Poisson, ordinal logit, Weibull, and many other multilevel mixed-effects models using syntax similar to that for **mixed** with linear models.

Explore the results

You can use **melogit**, **meprobit**, and **mecloglog** to fit multilevel logit, probit, and complementary log-log models.

Two-level logit model with random intercepts

```
. melogit y x1 x2 i.a || id:
```

Three-level model

```
. melogit y x1 x2 i.a || group: || id:
```

Random coefficients

```
. melogit y x1 x2 i.a || group: x1 || id: x2
```

You can easily modify this syntax, just as with **mixed**, to incorporate different covariance structures and fit four- and higher-level nested models and crossed-effects models. Replace **melogit** with **meprobit** or **mecloglog** to fit a probit or complementary log-log model.

Ordinal outcomes

Use **meologit** and **meoprobit** to fit multilevel ordinal logit and ordinal probit models.

Two-level ordinal logit model with random intercepts

```
. meologit y x1 x2 i.a || id:
```

Again, the same syntax modifications allow for random coefficients, different covariance structures, more levels, and crossed effects.

Count outcomes

Use **mepoisson** and **menbreg** to fit multilevel Poisson and negative binomial models. And the syntax is the same.

```
. mepoisson y x1 x2 i.a || ...
. menbreg y x1 x2 i.a || ...
```

Censored outcomes

When your outcome is continuous but censored, fit a multilevel interval-censored or tobit model.

```
. meintreg ylower yupper x1 x2 i.a || ...
. metobit y x1 x2 i.a || ..., ll(0)
```

Survival-time outcomes

Use **mestreg** to fit multilevel exponential, Weibull, lognormal, loglogistic, or gamma models.

Specify the time and failure variables

```
. stset time, failure(fail)
```

Two-level Weibull model with random intercepts

```
. mestreg x1 x2 i.a || id:, distribution(weibull)
```

Additional multilevel parametric survival models

```
. mestreg x1 x2 i.a || ..., distribution(...)
```

Account for complex survey design

Specify the survey design—weights at each level, clustering, stratification, and more.

```
. svyset cluster, weight(wvar2) strata(stvar) ||
        _n, weight(wvar1)
```

Fit two-level logit model with random intercepts

```
. svy: melogit y x1 x2 i.a || id:
```

Fit any **me** model

```
. svy: mecmd ...
```

Explore the results

Predict random effects (empirical Bayes means or modes) and their standard errors

```
. predict re, reffects reses(re_se)
```

Predict probabilities, counts, incidence rates, survival times, and more

```
. predict mypred
```

Estimate intraclass correlations

```
. estat icc
```

Perform Wald tests of coefficients

```
. test x1=1
```

Compare models via a likelihood-ratio test

```
. lrtest model1 model2
```

Estimate and plot marginal predictions—expected probabilities, counts, survival times, and more

```
. margins a, at(x1 = (-10(1)10))
. marginsplot ...
```


Marginal predictions with 95% CIs
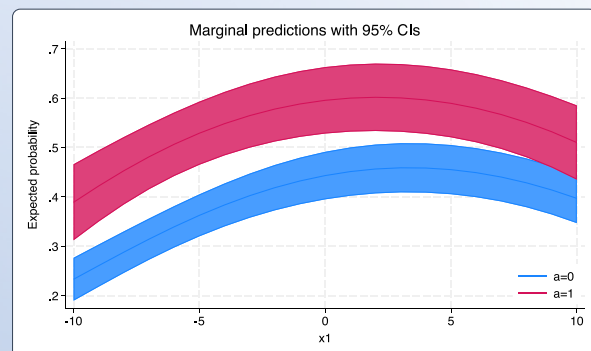
Bayesian estimation

Bayesian two-level logit model with random intercepts

```
. bayes: melogit y x1 x2 i.a || id:
```

Bayesian multilevel models

```
. bayes, priors ...: mecmd ...
```

Nonlinear models

When your model is not linear in the parameters, the **menl** command can fit nonlinear multilevel mixed-effects models.

Random-intercept model

Say you are modeling growth in the weight of plant leaves with random intercepts U for plots:
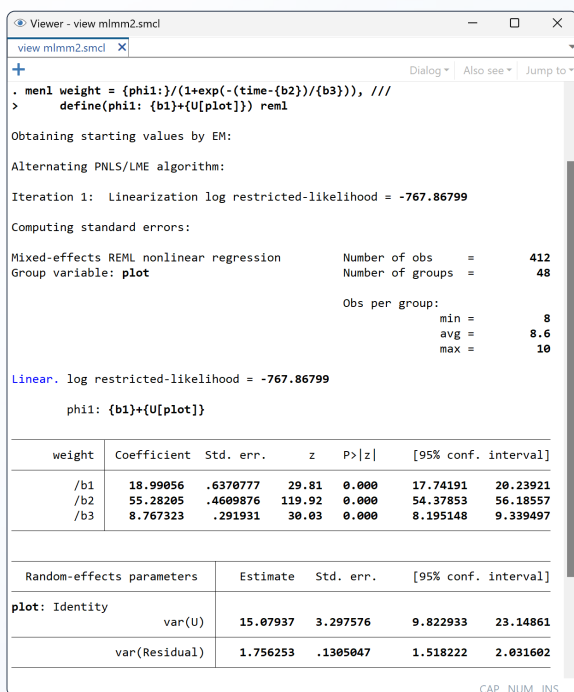
$$\text{weight} = \phi_1/[1 + \exp\{-(\text{time}-\beta_2)/\beta_3\}] + \epsilon$$
$$\phi_1 = \phi_{1j} = \beta_1 + U_j$$

You can type your command almost the same way you wrote your model:

```
. menl weight = {phi1:}/(1+exp(-(time-{b2})/{b3})),
    define(phi1: {b1}+{U[plot]})
```

You can also fit the model using restricted maximum likelihood instead of the default maximum likelihood:

```
. menl weight = {phi1:}/(1+exp(-(time-{b2})/{b3})),
    define(phi1: {b1}+{U[plot]}) reml
```

```
Viewer - view mlmm2.smcl                                    —    □    ✕

view mlmm2.smcl   ✕

+                                    Dialog ▾  Also see ▾  Jump to ▾

. menl weight = {phi1:}/(1+exp(-(time-{b2})/{b3})), ///
>    define(phi1: {b1}+{U[plot]}) reml

Obtaining starting values by EM:

Alternating PNLS/LME algorithm:

Iteration 1:  Linearization log restricted-likelihood = -767.86799

Computing standard errors:

Mixed-effects REML nonlinear regression     Number of obs    =      412
Group variable: plot                        Number of groups =       48

                                            Obs per group:
                                                        min =        8
                                                        avg =      8.6
                                                        max =       10

Linear. log restricted-likelihood = -767.86799

    phi1: {b1}+{U[plot]}

      weight │ Coefficient  Std. err.      z    P>|z|    [95% conf. interval]
        /b1  │   18.99056   .6370777    29.81   0.000    17.74191    20.23921
        /b2  │   55.28205   .4609876   119.92   0.000    54.37853    56.18557
        /b3  │    8.767323   .291931    30.03   0.000     8.195148    9.339497

  Random-effects parameters │  Estimate   Std. err.    [95% conf. interval]

plot: Identity
                     var(U) │  15.07937   3.297576     9.822933    23.14861

               var(Residual)│   1.756253   .1305047    1.518222    2.031602

                                                       CAP  NUM  INS
```

Random-slope model

Extending the model to include random slopes for variety is just as easy. To fit

$$\text{weight} = \phi_1/[1 + \exp\{-(\text{time}-\beta_2)/\beta_3\}] + \epsilon$$
$$\phi_1 = \phi_{1j} = \beta_{10} + U_{0j} + \beta_{11} \times \text{variety} + U_{1j} \times \text{variety}$$

we type

```
. menl weight = {phi1:}/(1+exp(-(time-{b2})/{b3})),
    define(phi1: {b10}+{U0[plot]})+
    {b11}*variety+{1.variety#U1[plot]})
```

Or we can use the efficient linear-combination notation:

```
. menl weight = {phi1:}/(1+exp(-(time-{b2})/{b3})),
    define(phi1:
    U0[plot] variety 1.variety#U1[plot])
```

Multilevel models

We can also extend the model to three or more levels.

If plots are nested within zones, we can fit a model with both zone-level and plot-level random intercepts by typing

```
. menl weight = {phi1:}/(1+exp(-(time-{phi2:}))/{b3})),
    define(phi1: {b1}+{U1[zone]}+{UU1[plot<zone]})
    define(phi2: {b2}+{U2[zone]}+{UU2[plot<zone]})
```

In the above, we allowed **phi1** and **phi2** to be zone and plot specific.

Residual covariance structures

Sometimes, you need to model error heteroskedasticity and its within-group dependence, so **menl** allows flexible variance–covariance residual structures.

Graphs by Subject ID
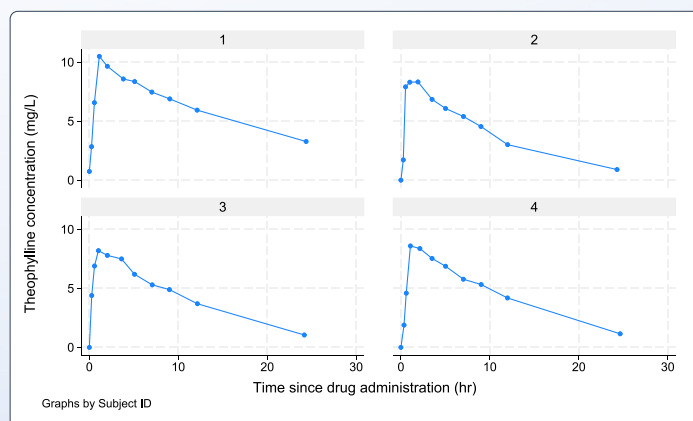
For instance, we model these drug concentration-time profiles and use a power-plus-constant function of the predicted mean to model the within-subject error variance.

$$\text{conc} = \text{dose} \times k_e \times k_a / \{c_l \times (k_a - k_e)\} \times \{\exp(-k_e \times \text{time}) - \exp(-k_a \times \text{time})\} + \epsilon$$
$$c_l = c_{lj} = \exp(\beta_0 + U_{0j})$$
$$k_a = k_{aj} = \exp(\beta_1 + U_{1j})$$
$$k_e = \exp(\beta_2)$$
$$\text{Var}(\epsilon) = \sigma^2(a + \widehat{\text{conc}}^b)^2$$

To fit this model, we type

```
. menl conc = (dose*{ke:}*{ka:}/({cl:}*({ka:}-{ke:})))*
    (exp(-{ke:}*time)-exp(-{ka:}*time)),
    define(cl: exp({b0}+{U0[subject]}))
    define(ka: exp({b1}+{U1[subject]}))
    define(ke: exp({b2}))
    resvariance(power _yhat)
```

Predictions

After fitting the model, we can predict parameters of interest by typing, for instance,

```
. predict double (cl = {cl:})
```

We can also predict random effects at a specific level:

```
. predict u*, reffects relevel(subject)
```