

bayes: meprobit — Bayesian multilevel probit regression
[Description](#)[Remarks and examples](#)[Quick start](#)[Stored results](#)[Menu](#)[Methods and formulas](#)[Syntax](#)[Also see](#)

Description

`bayes: meprobit` fits a Bayesian multilevel probit regression to a binary outcome; see [\[BAYES\] bayes](#) and [\[ME\] meprobit](#) for details.

Quick start

Bayesian two-level probit regression of `y` on `x1` and `x2` with random intercepts by `id`, using default normal priors for regression coefficients and default inverse-gamma prior for the variance of random intercepts

```
bayes: meprobit y x1 x2 || id:
```

Use a standard deviation of 10 instead of 100 for the default normal priors

```
bayes, normalprior(10): meprobit y x1 x2 || id:
```

Use uniform priors for the slopes and a normal prior for the intercept

```
bayes, prior({y: x1 x2}, uniform(-10,10)) ///
prior({y:_cons}, normal(0,10)): meprobit y x1 x2 || id:
```

Save simulation results to `simdata.dta`, and use a random-number seed for reproducibility

```
bayes, saving(simdata) rseed(123): meprobit y x1 x2 || id:
```

Specify 20,000 MCMC samples, set length of the burn-in period to 5,000, and request that a dot be displayed every 500 simulations

```
bayes, mcmcsample(20000) burnin(5000) dots(500): meprobit y x1 x2 || id:
```

In the above, request that the 90% HPD credible interval be displayed instead of the default 95% equal-tailed credible interval

```
bayes, clevel(90) hpd
```

Also see [Quick start](#) in [\[BAYES\] bayes](#) and [Quick start](#) in [\[ME\] meprobit](#).

Menu

Statistics > Multilevel mixed-effects models > Bayesian regression > Probit regression

Syntax

```
bayes [ , bayesopts ] : meprobit depvar fe_equation
      [ || re_equation ] [ || re_equation ... ] [ , options ]
```

where the syntax of *fe_equation* is

```
[ indepvars ] [ if ] [ in ] [ weight ] [ , fe_options ]
```

and the syntax of *re_equation* is one of the following:

for random coefficients and intercepts

```
levelvar: [ varlist ] [ , re_options ]
```

for random effects among the values of a factor variable

```
levelvar: R.varname
```

levelvar either is a variable identifying the group structure for the random effects at that level or is `_all`, representing one group comprising all observations.

<i>fe_options</i>	Description
Model	
<code>noconstant</code>	suppress constant term from the fixed-effects equation
<code>offset(<i>varname</i>)</code>	include <i>varname</i> in model with coefficient constrained to 1
<code>asis</code>	retain perfect predictor variables

<i>re_options</i>	Description
Model	
<code>covariance(<i>vartype</i>)</code>	variance–covariance structure of the random effects ; only structures <code>independent</code> , <code>identity</code> , and <code>unstructured</code> supported
<code>noconstant</code>	suppress constant term from the random-effects equation

<i>options</i>	Description
Model	
<code>binomial(<i>varname</i> #)</code>	set binomial trials if data are in binomial form
<code>collinear</code>	keep collinear variables
Reporting	
<code>notable</code>	suppress coefficient table
<code>noheader</code>	suppress output header
<code>nogroup</code>	suppress table summarizing groups
<code>display_options</code>	control spacing, line width, and base and empty cells
<code>level(#)</code>	set credible level; default is <code>level(95)</code>

indepvars may contain factor variables; see [U] 11.4.3 **Factor variables**.

depvar, *indepvars*, and *varlist* may contain time-series operators; see [U] 11.4.4 **Time-series varlists**.

fweights are allowed; see [U] 11.1.6 **weight**.

bayes: `meprobit`, `level()` is equivalent to `bayes`, `clevel()`: `meprobit`.

For a detailed description of *options*, see *Options* in [ME] `meprobit`.

<i>bayesopts</i>	Description
Priors	
* <code>normalprior(#)</code>	specify standard deviation of default normal priors for regression coefficients; default is <code>normalprior(100)</code>
* <code>igammaprior(# #)</code>	specify shape and scale of default inverse-gamma prior for variance components; default is <code>igammaprior(0.01 0.01)</code>
* <code>iwishartprior(# [...])</code>	specify degrees of freedom and, optionally, scale matrix of default inverse-Wishart prior for unstructured random-effects covariance
<code>prior(priorspec)</code>	prior for model parameters; this option may be repeated
<code>dryrun</code>	show model summary without estimation
Simulation	
<code>mcmcsize(#)</code>	MCMC sample size; default is <code>mcmcsize(10000)</code>
<code>burnin(#)</code>	burn-in period; default is <code>burnin(2500)</code>
<code>thinning(#)</code>	thinning interval; default is <code>thinning(1)</code>
<code>rseed(#)</code>	random-number seed
<code>exclude(paramref)</code>	specify model parameters to be excluded from the simulation results
<code>restubs(restub1 restub2 ...)</code>	specify stubs for random-effects parameters for all levels
Blocking	
* <code>blocksize(#)</code>	maximum block size; default is <code>blocksize(50)</code>
<code>block(paramref[, blockopts])</code>	specify a block of model parameters; this option may be repeated
<code>blocksummary</code>	display block summary
* <code>noblocking</code>	do not block parameters by default
Initialization	
<code>initial(initspec)</code>	initial values for model parameters
<code>nomleinitial</code>	suppress the use of maximum likelihood estimates as starting values
<code>initrandom</code>	specify random initial values
<code>initsummary</code>	display initial values used for simulation
* <code>noisily</code>	display output from the estimation command during initialization
Adaptation	
<code>adaptation(adaptopts)</code>	control the adaptive MCMC procedure
<code>scale(#)</code>	initial multiplier for scale factor; default is <code>scale(2.38)</code>
<code>covariance(cov)</code>	initial proposal covariance; default is the identity matrix

Reporting

<code>clevel(#)</code>	set credible interval level; default is <code>clevel(95)</code>
<code>hpd</code>	display HPD credible intervals instead of the default equal-tailed credible intervals
<code>eform[<i>(string)</i>]</code>	report exponentiated coefficients and, optionally, label as <i>string</i>
<code>remargl</code>	compute log marginal likelihood
<code>batch(#)</code>	specify length of block for batch-means calculations; default is <code>batch(0)</code>
<code>saving(filename[, replace])</code>	save simulation results to <i>filename.dta</i>
<code>nomodelsummary</code>	suppress model summary
<code>nomesummary</code>	suppress multilevel-structure summary
<code>[no]dots</code>	suppress dots or display dots every 100 iterations and iteration numbers every 1,000 iterations; default is <code>dots</code>
<code>dots#[, every(#)]</code>	display dots as simulation is performed
<code>[no]show(paramref)</code>	specify model parameters to be excluded from or included in the output
<code>showeffects[<i>(ref)</i>]</code>	specify that all or a subset of random-effects parameters be included in the output
<code>melabel</code>	display estimation table using the same row labels as <code>meprobit</code>
<code>nogroup</code>	suppress table summarizing groups
<code>notable</code>	suppress estimation table
<code>noheader</code>	suppress output header
<code>title(string)</code>	display <i>string</i> as title above the table of parameter estimates
<code>display_options</code>	control spacing, line width, and base and empty cells

Advanced

<code>search(search_options)</code>	control the search for feasible initial values
<code>corrlag(#)</code>	specify maximum autocorrelation lag; default varies
<code>corrctl(#)</code>	specify autocorrelation tolerance; default is <code>corrctl(0.01)</code>

*Starred options are specific to the `bayes` prefix; other options are common between `bayes` and `bayesmh`.

Options `prior()` and `block()` can be repeated.

`priorspec` and `paramref` are defined in [BAYES] `bayesmh`.

`paramref` may contain factor variables; see [U] 11.4.3 Factor variables.

See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

Model parameters are regression coefficients `{depvar: indepvars}`, random effects `{rename}`, and either variance components `{rename:sigma2}` or, if option `covariance(unstructured)` is specified, matrix parameter `{restub:Sigma,matrix}`; see *Likelihood model* in [BAYES] `bayes` for how `renames` and `restub` are defined. Use the `dryrun` option to see the definitions of model parameters prior to estimation.

For a detailed description of `bayesopts`, see *Options* in [BAYES] `bayes`.

Remarks and examples

[stata.com](http://www.stata.com)

For a general introduction to Bayesian analysis, see [BAYES] `intro`. For a general introduction to Bayesian estimation using an adaptive Metropolis–Hastings algorithm, see [BAYES] `bayesmh`. For remarks and examples specific to the `bayes` prefix, see [BAYES] `bayes`. For details about the estimation command, see [ME] `meprobit`.

For a simple example of the `bayes` prefix, see *Introductory example* in [BAYES] `bayes`. For multilevel examples, see *Multilevel models* in [BAYES] `bayes`.

Stored results

See *Stored results* in [BAYES] [bayesmh](#).

Methods and formulas

See *Methods and formulas* in [BAYES] [bayesmh](#).

Also see

[BAYES] [bayes](#) — Bayesian regression models using the bayes prefix

[ME] [meprobit](#) — Multilevel mixed-effects probit regression

[BAYES] [bayesian postestimation](#) — Postestimation tools for bayesmh and the bayes prefix

[BAYES] [bayesian estimation](#) — Bayesian estimation commands

[BAYES] [bayesian commands](#) — Introduction to commands for Bayesian analysis

[BAYES] [intro](#) — Introduction to Bayesian analysis

[BAYES] [Glossary](#)