bayes: metobit — Bayesian multilevel tobit regression

Description

bayes: metobit fits a Bayesian multilevel tobit regression to a censored continuous outcome; see [BAYES] bayes and [ME] metobit for details.

Quick start

Bayesian two-level tobit regression of \( y \) on \( x_1 \) and \( x_2 \) with random intercepts by \( id \), using a lower censoring limit of 17, and using default normal priors for regression coefficients and default inverse-gamma priors for the error variance and for the variance of random intercepts

\[
\text{bayes: metobit } y \ x_1 \ x_2 \ || \ id:, \ ll(17)
\]

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

\[
\text{bayes, normalprior}(10): \text{metobit } y \ x_1 \ x_2 \ || \ id:, \ ll(17)
\]

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

\[
\text{prior}({y: x_1 \ x_2}, \text{uniform}(-10,10)) /// \text{prior}({y: \text{cons}}, \text{normal}(0,10)): \text{metobit } y \ x_1 \ x_2 \ || \ id:, \ ll(17)
\]

Save simulation results to \( \text{simdata.dta} \), and use a random-number seed for reproducibility

\[
\text{bayes, saving(simdata) rseed(123):} /// \text{metobit } y \ x_1 \ x_2 \ || \ id:, \ ll(17)
\]

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

\[
\text{bayes, mcmcsize}(20000) \ \text{burnin}(5000) \ \text{dots}(500): /// \text{metobit } y \ x_1 \ x_2 \ || \ id:, \ ll(17)
\]

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

\[
\text{bayes, clevel}(90) \ \text{hpd}
\]

Also see Quick start in [BAYES] bayes and Quick start in [ME] metobit.

Menu

Statistics > Multilevel mixed-effects models > Bayesian regression > Tobit regression
Syntax

```
bayes [, bayesopts] : metobit 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.

### fe_options

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>noconstant</strong></td>
</tr>
<tr>
<td>suppress constant term from the fixed-effects equation</td>
</tr>
<tr>
<td><strong>offset(varname)</strong></td>
</tr>
<tr>
<td>include <code>varname</code> in model with coefficient constrained to 1</td>
</tr>
</tbody>
</table>

### re_options

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>covariance(vartype)</strong></td>
</tr>
<tr>
<td>variance–covariance structure of the random effects; only structures independent, identity, and unstructured supported</td>
</tr>
<tr>
<td><strong>noconstant</strong></td>
</tr>
<tr>
<td>suppress constant term from the random-effects equation</td>
</tr>
</tbody>
</table>

### options

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>**ll(varname</td>
</tr>
<tr>
<td>left-censoring variable or limit</td>
</tr>
<tr>
<td>**ul(varname</td>
</tr>
<tr>
<td>right-censoring variable or limit</td>
</tr>
<tr>
<td><strong>notable</strong></td>
</tr>
<tr>
<td>suppress coefficient table</td>
</tr>
<tr>
<td><strong>noheader</strong></td>
</tr>
<tr>
<td>suppress output header</td>
</tr>
<tr>
<td><strong>nogroup</strong></td>
</tr>
<tr>
<td>suppress table summarizing groups</td>
</tr>
<tr>
<td><strong>display_options</strong></td>
</tr>
<tr>
<td>control spacing, line width, and base and empty cells</td>
</tr>
<tr>
<td><strong>level(#)</strong></td>
</tr>
<tr>
<td>set credible level; default is <code>level(95)</code></td>
</tr>
</tbody>
</table>
bayes: metobit — Bayesian multilevel tobit regression

`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.
`fweight`s are allowed; see [U] 11.1.6 weight.
`bayes: metobit, level()` is equivalent to `bayes, clevel(): metobit`.
For a detailed description of options, see Options in [ME] metobit.

### Bayesian options

**Priors**

- `normalpr(#)` specify standard deviation of default normal priors for regression coefficients; default is `normalpr(100)`
- `igammapr(# #)` specify shape and scale of default inverse-gamma prior for variance components; default is `igammapr(0.01 0.01)`
- `iwishartpr(# [..])` specify degrees of freedom and, optionally, scale matrix of default inverse-Wishart prior for unstructured random-effects covariance prior for model parameters; this option may be repeated
- `prior(priorspec)`
- `dryrun` show model summary without estimation

**Simulation**

- `nchains(#)` number of chains; default is to simulate one chain
- `mcmcsize(#)` MCMC sample size; default is `mcmcsize(10000)`
- `burnin(#)` burn-in period; default is `burnin(2500)`
- `thinning(#)` thinning interval; default is `thinning(1)`
- `rseed(#)` random-number seed
- `exclude(paramref)` specify model parameters to be excluded from the simulation results
- `restubs(restub1 restub2 .....)` specify stubs for random-effects parameters for all levels

**Blocking**

- `blocksize(#)` maximum block size; default is `blocksize(50)`
- `block(paramref[, blockopts])` specify a block of model parameters; this option may be repeated
- `blocksummary` display block summary
- `noblock` do not block parameters by default

**Initialization**

- `initial(initspec)` specify initial values for model parameters with a single chain
- `init#(initspec)` specify initial values for #th chain; requires `nchains()`
- `initall(initspec)` specify initial values for all chains; requires `nchains()`
- `nomleinitial` suppress the use of maximum likelihood estimates as starting values
- `initrandom` specify random initial values
- `initsummary` display initial values used for simulation
- `noisily` display output from the estimation command during initialization

**Adaptation**

- `adaptation(adaptopts)` control the adaptive MCMC procedure
- `scale(#)` initial multiplier for scale factor; default is `scale(2.38)`
- `covariance(cov)` initial proposal covariance; default is the identity matrix
bayes: metobit — Bayesian multilevel tobit regression

Reporting

- `clevel(#)`: set credible interval level; default is `clevel(95)`
- `hpd`: display HPD credible intervals instead of the default equal-tailed credible intervals
- `eform[(string)]`: report exponentiated coefficients and, optionally, label as `string`
- `remargl`: compute log marginal-likelihood
- `batch(#)`: specify length of block for batch-means calculations; default is `batch(0)`

Options for saving simulation results:

- `saving(filename[, replace])`: save simulation results to `filename.dta`
- `nomodelsummary`: suppress model summary
- `nomsesummary`: suppress multilevel-structure summary
- `chainsdetail`: display detailed simulation summary for each chain
- `no dots`: suppress dots or display dots every 100 iterations and iteration numbers every 1,000 iterations; default is `dots`
- `dots(#[, every(#)])`: display dots as simulation is performed
- `show(paramref)`: specify model parameters to be excluded from or included in the output

- `melabel`: display estimation table using the same row labels as `metobit`
- `nogroup`: suppress table summarizing groups
- `table` (arguments): display parameter estimates
- `display_options`: control spacing, line width, and base and empty cells

Advanced options:

- `search(search_options)`: control the search for feasible initial values
- `corrlag(#)`: specify maximum autocorrelation lag; default varies
- `corrtol(#)`: specify autocorrelation tolerance; default is `corrtol(0.01)`

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

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

Options `prior()` and `block()` may 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}`, error variance `{e.depvar:sigma2}`, 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 `rename`s and `restub`s 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

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] metobit.
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] bayes.

**Methods and formulas**

See *Methods and formulas* in [BAYES] bayesmh.

**Also see**

- [BAYES] bayes — Bayesian regression models using the bayes prefix
- [ME] metobit — Multilevel mixed-effects tobit 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