0.1 `mlogit.bayes`: Bayesian Multinomial Logistic Regression

Use Bayesian multinomial logistic regression to model unordered categorical variables. The dependent variable may be in the format of either character strings or integer values. The model is estimated via a random walk Metropolis algorithm or a slice sampler. See Section **??** for the maximum-likelihood estimation of this model.

Syntax

```
> z.out <- zelig(Y ~ X1 + X2, model = "mlogit.bayes", data = mydata)
> x.out <- setx(z.out)
> s.out <- sim(z.out, x = x.out)
```

Additional Inputs

`zelig()` accepts the following arguments for `mlogit.bayes`:

- `baseline`: either a character string or numeric value (equal to one of the observed values in the dependent variable) specifying a baseline category. The default value is `NA` which sets the baseline to the first alphabetical or numerical unique value of the dependent variable.

The model accepts the following additional arguments to monitor the Markov chains:

- `burnin`: number of the initial MCMC iterations to be discarded (defaults to 1,000).

- `mcmc`: number of the MCMC iterations after burnin (defaults to 10,000).

- `thin`: thinning interval for the Markov chain. Only every `thin`-th draw from the Markov chain is kept. The value of `mcmc` must be divisible by this value. The default value is 1.

- `mcmc.method`: either `"MH"` or `"slice"`, specifying whether to use Metropolis Algorithm or slice sampler. The default value is `"MH"`.

- `tune`: tuning parameter for the Metropolis-Hasting step, either a scalar or a numeric vector (for k coefficients, enter a k vector). The tuning parameter should be set such that the acceptance rate is satisfactory (between 0.2 and 0.5). The default value is 1.1.

- `verbose`: defaults to `FALSE`. If `TRUE`, the progress of the sampler (every 10%) is printed to the screen.

- `seed`: seed for the random number generator. The default is `NA` which corresponds to a random seed of 12345.

- `beta.start`: starting values for the Markov chain, either a scalar or a vector (for k coefficients, enter a k vector). The default is `NA` where the maximum likelihood estimates are used as the starting values.

Use the following arguments to specify the priors for the model:

- `b0`: prior mean for the coefficients, either a scalar or vector. If a scalar, that value will be the prior mean for all the coefficients. The default is 0.

- `B0`: prior precision parameter for the coefficients, either a square matrix with the dimensions equal to the number of coefficients or a scalar. If a scalar, that value times an identity matrix will be the prior precision parameter. The default is 0 which leads to an improper prior.

Zelig users may wish to refer to `help(MCMCmnl)` for more information.

Convergence

Users should verify that the Markov Chain converges to its stationary distribution. After running the `zelig()` function but before performing `setx()`, users may conduct the following convergence diagnostics tests:

- `geweke.diag(z.out$coefficients)`: The Geweke diagnostic tests the null hypothesis that the Markov chain is in the stationary distribution and produces z-statistics for each estimated parameter.

- `heidel.diag(z.out$coefficients)`: The Heidelberger-Welch diagnostic first tests the null hypothesis that the Markov Chain is in the stationary distribution and produces p-values for each estimated parameter. Calling `heidel.diag()` also produces output that indicates whether the mean of a marginal posterior distribution can be estimated with sufficient precision, assuming that the Markov Chain is in the stationary distribution.

- `raftery.diag(z.out$coefficients)`: The Raftery diagnostic indicates how long the Markov Chain should run before considering draws from the marginal posterior distributions sufficiently representative of the stationary distribution.

If there is evidence of non-convergence, adjust the values for `burnin` and `mcmc` and rerun `zelig()`.

Advanced users may wish to refer to `help(geweke.diag)`, `help(heidel.diag)`, and `help(raftery.diag)` for more information about these diagnostics.

Examples

1. Basic Example
   Attaching the sample dataset:

   ```
   > data(mexico)
   ```

   Estimating multinomial logistics regression using `mlogit.bayes`:

   ```
   > z.out <- zelig(vote88 ~ pristr + othcok + othsocok, model = "mlogit.bayes",
   +      data = mexico)
   ```

   Checking for convergence before summarizing the estimates:

   ```
   > heidel.diag(z.out$coefficients)
   ```

   ```
   > raftery.diag(z.out$coefficients)
   ```

   ```
   > summary(z.out)
   ```

   Setting values for the explanatory variables to their sample averages:

   ```
   > x.out <- setx(z.out)
   ```

   Simulating quantities of interest from the posterior distribution given `x.out`.

   ```
   > s.out1 <- sim(z.out, x = x.out)
   ```

   ```
   > summary(s.out1)
   ```

2. Simulating First Differences
   Estimating the first difference (and risk ratio) in the probabilities of voting different candidates when `pristr` (the strength of the PRI) is set to be weak (equal to 1) versus strong (equal to 3) while all the other variables held at their default values.

   ```
   > x.weak <- setx(z.out, pristr = 1)
   > x.strong <- setx(z.out, pristr = 3)
   ```

   ```
   > s.out2 <- sim(z.out, x = x.strong, x1 = x.weak)
   ```

   ```
   > summary(s.out2)
   ```

Model

Let Y_i be the (unordered) categorical dependent variable for observation i which takes an integer values $j = 1, \ldots, J$.

- The *stochastic component* is given by:

$$Y_i \quad \sim \quad \text{Multinomial}(Y_i \mid \pi_{ij}).$$

  where $\pi_{ij} = \Pr(Y_i = j)$ for $j = 1, \ldots, J$.

- The *systematic component* is given by

$$\pi_{ij} = \frac{\exp(x_i \beta_j)}{\sum_{k=1}^{J} \exp(x_i \beta_k)}, \quad \text{for } j = 1, \ldots, J - 1,$$

  where x_i is the vector of k explanatory variables for observation i and β_j is the vector of coefficient for category j. Category J is assumed to be the baseline category.

- The *prior* for β is given by

$$\beta_j \sim \text{Normal}_k \left(b_0, B_0^{-1} \right) \text{ for } j = 1, \ldots, J - 1,$$

  where b_0 is the vector of means for the k explanatory variables and B_0 is the $k \times k$ precision matrix (the inverse of a variance-covariance matrix).

Quantities of Interest

- The expected values (`qi$ev`) for the multinomial logistics regression model are the predicted probability of belonging to each category:

$$\Pr(Y_i = j) = \pi_{ij} = \frac{\exp(x_i \beta_j)}{\sum_{k=1}^{J} \exp(x_J \beta_k)}, \quad \text{for } j = 1, \ldots, J - 1,$$

  and

$$\Pr(Y_i = J) = 1 - \sum_{j=1}^{J-1} \Pr(Y_i = j)$$

  given the posterior draws of β_j for all categories from the MCMC iterations.

- The predicted values (`qi$pr`) are the draws of Y_i from a multinomial distribution whose parameters are the expected values(`qi$ev`) computed based on the posterior draws of β from the MCMC iterations.

- The first difference ($\texttt{qi\$fd}$) in category j for the multinomial logistic model is defined as

$$\text{FD}_j = \Pr(Y_i = j \mid X_1) - \Pr(Y_i = j \mid X).$$

- The risk ratio ($\texttt{qi\$rr}$) in category j is defined as

$$\text{RR}_j = \Pr(Y_i = j \mid X_1) \ / \ \Pr(Y_i = j \mid X).$$

- In conditional prediction models, the average expected treatment effect ($\texttt{qi\$att.ev}$) for the treatment group in category j is

$$\frac{1}{n_j} \sum_{i:t_i=1}^{n_j} [Y_i(t_i = 1) - E[Y_i(t_i = 0)]],$$

where t_i is a binary explanatory variable defining the treatment ($t_i = 1$) and control ($t_i = 0$) groups, and n_j is the number of treated observations in category j.

- In conditional prediction models, the average predicted treatment effect ($\texttt{qi\$att.pr}$) for the treatment group in category j is

$$\frac{1}{n_j} \sum_{i:t_i=1}^{n_j} [Y_i(t_i = 1) - \widehat{Y_i(t_i = 0)}],$$

where t_i is a binary explanatory variable defining the treatment ($t_i = 1$) and control ($t_i = 0$) groups, and n_j is the number of treated observations in category j.

Output Values

The output of each Zelig command contains useful information which you may view. For example, if you run:

```
z.out <- zelig(y ~ x, model = "mlogit.bayes", data)
```

then you may examine the available information in $\texttt{z.out}$ by using $\texttt{names(z.out)}$, see the draws from the posterior distribution of the $\texttt{coefficients}$ by using $\texttt{z.out\$coefficients}$, and view a default summary of information through $\texttt{summary(z.out)}$. Other elements available through the $\texttt{\$}$ operator are listed below.

- From the $\texttt{zelig()}$ output object $\texttt{z.out}$, you may extract:

    - $\texttt{coefficients}$: draws from the posterior distributions of the estimated coefficients β for each category except the baseline category.
    - $\texttt{zelig.data}$: the input data frame if $\texttt{save.data = TRUE}$.

5

– `seed`: the random seed used in the model.

- From the `sim()` output object `s.out`:

  – `qi$ev`: the simulated expected values(probabilities) of each of the J categories given the specified values of `x`.

  – `qi$pr`: the simulated predicted values drawn from the multinomial distribution defined by the expected values(`qi$ev`) given the specified values of `x`.

  – `qi$fd`: the simulated first difference in the expected values of each of the J categories for the values specified in `x` and `x1`.

  – `qi$rr`: the simulated risk ratio for the expected values of each of the J categories simulated from `x` and `x1`.

  – `qi$att.ev`: the simulated average expected treatment effect for the treated from conditional prediction models.

  – `qi$att.pr`: the simulated average predicted treatment effect for the treated from conditional prediction models.

How to Cite

To cite the *mlogit.bayes* Zelig model:

> Ben Goodrich and Ying Lu. 2007. "mlogit.bayes: Bayesian Multinomial Logistic Regression for Dependent Variables with Unordered Categorical Values" in Kosuke Imai, Gary King, and Olivia Lau, "Zelig: Everyone's Statistical Software,"`http://gking.harvard.edu/zelig`

To cite Zelig as a whole, please reference these two sources:

> Kosuke Imai, Gary King, and Olivia Lau. 2007. "Zelig: Everyone's Statistical Software," `http://GKing.harvard.edu/zelig`.

> Imai, Kosuke, Gary King, and Olivia Lau. (2008). "Toward A Common Framework for Statistical Analysis and Development." Journal of Computational and Graphical Statistics, Vol. 17, No. 4 (December), pp. 892-913.

See also

Bayesian logistic regression is part of the MCMCpack library by Andrew D. Martin and Kevin M. Quinn (Martin and Quinn 2005). The convergence diagnostics are part of the CODA library by Martyn Plummer, Nicky Best, Kate Cowles, and Karen Vines (Plummer et al. 2005).

Bibliography

Martin, A. D. and Quinn, K. M. (2005), *MCMCpack: Markov chain Monte Carlo (MCMC) Package.*

Plummer, M., Best, N., Cowles, K., and Vines, K. (2005), *coda: Output analysis and diagnostics for MCMC.*