0.1 `mlogit`: Multinomial Logistic Regression for Dependent Variables with Unordered Categorical Values

Use the multinomial logit distribution to model unordered categorical variables. The dependent variable may be in the format of either character strings or integer values. See for a Bayesian version of this model.

Syntax

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

Input Values

If the user wishes to use the same formula across all levels, then `formula <- as.factor(Y) ~ X1 + X2` may be used. If the user wants to use different formula for each level then the following syntax should be used:

```
formulae <- list(list(id(Y, "apples")~ X1,
                      id(Y, "bananas")~ X1 + X2)
```

where Y above is supposed to be a factor variable with levels apples,bananas,oranges. By default, oranges is the last level and omitted. (You cannot specify a different base level at this time.) For J equations, there must be $J + 1$ levels.

Examples

1. The same formula for each level

   Load the sample data:

   ```
   > data(mexico)
   ```

   Estimate the empirical model:

   ```
   > z.out1 <- zelig(as.factor(vote88) ~ pristr + othcok + othsocok,
   +     model = "mlogit", data = mexico)
   ```

   Set the explanatory variables to their default values, with `pristr` (for the strength of the PRI) equal to 1 (weak) in the baseline values, and equal to 3 (strong) in the alternative values:

   ```
   > x.weak <- setx(z.out1, pristr = 1)
   > x.strong <- setx(z.out1, pristr = 3)
   ```

Generate simulated predicted probabilities `qi$ev` and differences in the predicted probabilities `qi$fd`:

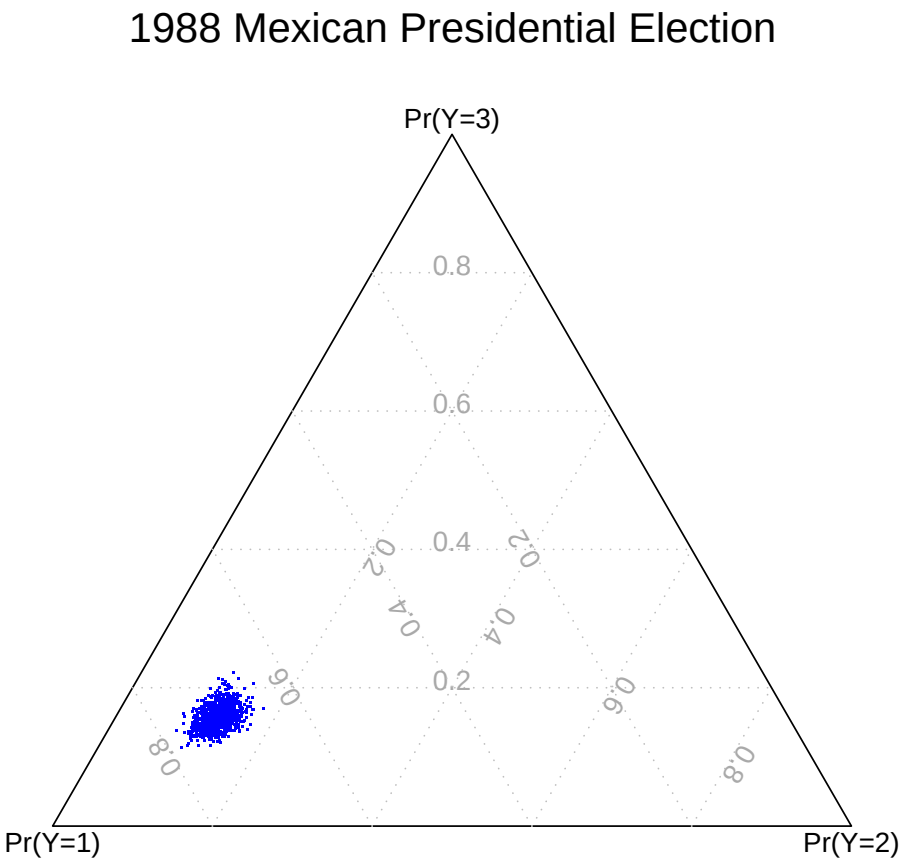
```
> s.out1 <- sim(z.out1, x = x.strong, x1 = x.weak)

> summary(s.out1)
```

Generate simulated predicted probabilities `qi$ev` for the alternative values:

```
> ev.weak <- s.out1$qi$ev + s.out1$qi$fd
```

Plot the differences in the predicted probabilities.

```
> library(vcd)

> ternaryplot(x = s.out1$qi$ev, pch = ".", col = "blue", main = "1988 Mexican Pres
```



2. Different formula for each level

   Estimate the empirical model:

```
> z.out2 <- zelig(list(id(vote88, "1") ~ pristr + othcok, id(vote88,
+     "2") ~ othsocok), model = "mlogit", data = mexico)
```

   Set the explanatory variables to their default values, with `pristr` (for the strength of the PRI) equal to 1 (weak) in the baseline values, and equal to 3 (strong) in the alternative values:

```
> x.weak <- setx(z.out2, pristr = 1)
> x.strong <- setx(z.out2, pristr = 3)
```

   Generate simulated predicted probabilities `qi$ev` and differences in the predicted probabilities `qi$fd`:

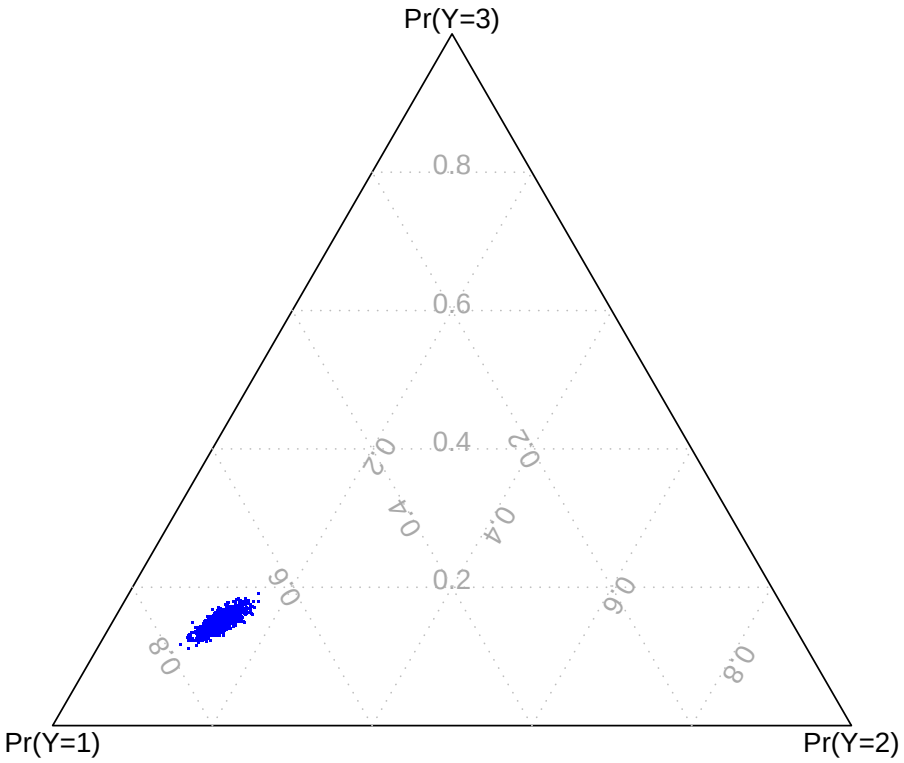
```
> s.out1 <- sim(z.out2, x = x.strong, x1 = x.weak)

> summary(s.out1)
```

   Generate simulated predicted probabilities `qi$ev` for the alternative values:

```
> ev.weak <- s.out1$qi$ev + s.out1$qi$fd
```

   Using the vcd package, plot the differences in the predicted probabilities.

```
> ternaryplot(s.out1$qi$ev, pch = ".", col = "blue", main = "1988 Mexican Presider
```



Model

Let Y_i be the unordered categorical dependent variable that takes one of the values from 1 to J, where J is the total number of categories.

- The stochastic component is given by

$$Y_i \sim \text{Multinomial}(y_i \mid \pi_{ij}),$$

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

- The systemic component is given by:

$$\pi_{ij} = \frac{\exp(x_i \beta_j)}{\sum_{k=1}^{J} \exp(x_i \beta_k)},$$

  where x_i is the vector of explanatory variables for observation i, and β_j is the vector of coefficients for category j.

Quantities of Interest

- The expected value (`qi$ev`) is the predicted probability for each category:

$$E(Y) \;=\; \pi_{ij} \;=\; \frac{\exp(x_i\beta_j)}{\sum_{k=1}^{J}\exp(x_i\beta_k)}.$$

- The predicted value (`qi$pr`) is a draw from the multinomial distribution defined by the predicted probabilities.

- The first difference in predicted probabilities (`qi$fd`), for each category is given by:

$$\text{FD}_j = \Pr(Y = j \mid x_1) - \Pr(Y = j \mid x) \quad \text{for} \quad j = 1, \ldots, J.$$

- In conditional prediction models, the average expected treatment effect (`att.ev`) for the treatment group is

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

  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 (`att.pr`) for the treatment group is

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

  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", data)`, then you may examine the available information in `z.out` by using `names(z.out)`, see the `coefficients` by using `z.out$coefficients`, and a default summary of information through `summary(z.out)`. Other elements available through the `$` operator are listed below.

- From the `zelig()` output object `z.out`, you may extract:
  - `coefficients`: the named vector of coefficients.
  - `fitted.values`: an $n \times J$ matrix of the in-sample fitted values.

- `predictors`: an $n \times (J-1)$ matrix of the linear predictors $x_i\beta_j$.
- `residuals`: an $n \times (J-1)$ matrix of the residuals.
- `df.residual`: the residual degrees of freedom.
- `df.total`: the total degrees of freedom.
- `rss`: the residual sum of squares.
- `y`: an $n \times J$ matrix of the dependent variables.
- `zelig.data`: the input data frame if `save.data = TRUE`.

- From `summary(z.out)`, you may extract:
  - `coef3`: a table of the coefficients with their associated standard errors and t-statistics.
  - `cov.unscaled`: the variance-covariance matrix.
  - `pearson.resid`: an $n \times (m-1)$ matrix of the Pearson residuals.

- From the `sim()` output object `s.out`, you may extract quantities of interest arranged as arrays. Available quantities are:
  - `qi$ev`: the simulated expected probabilities for the specified values of `x`, indexed by simulation $\times$ quantity $\times$ `x`-observation (for more than one `x`-observation).
  - `qi$pr`: the simulated predicted values drawn from the distribution defined by the expected probabilities, indexed by simulation $\times$ `x`-observation.
  - `qi$fd`: the simulated first difference in the predicted probabilities for the values specified in `x` and `x1`, indexed by simulation $\times$ quantity $\times$ `x`-observation (for more than one `x`-observation).
  - `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* Zelig model use:

> Kosuke Imai,Gary King, and Olivia Lau. 2007. "mlogit: 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`.

Kosuke Imai, Gary King, and Olivia Lau. 2007. "Toward A Common Framework for Statistical Analysis and Development," `http://gking.harvard.edu/files/abs/z-abs.shtml`.

See also

The multinomial logit function is part of the VGAM package by Thomas Yee (Yee and Hastie 2003). In addition, advanced users may wish to refer to `help(vglm)` in the VGAM library. Additional documentation is available at http://www.stat.auckland.ac.nz/~ yee.Sample data are from King et al. (2000).

Bibliography

King, G., Tomz, M., and Wittenberg, J. (2000), "Making the Most of Statistical Analyses: Improving Interpretation and Presentation," *American Journal of Political Science*, 44, 341–355, http://gking.harvard.edu/files/abs/making-abs.shtml.

Yee, T. W. and Hastie, T. J. (2003), "Reduced-rank vector generalized linear models," *Statistical Modelling*, 3, 15–41.