0.1 `logit.net`: Network Logistic Regression for Dichotomous Proximity Matrix Dependent Variables

Use network logistic regression analysis for a dependent variable that is a binary valued proximity matrix (a.k.a. sociomatricies, adjacency matrices, or matrix representations of directed graphs).

Syntax

```
> z.out <- zelig(y ~ x1 + x2, model = "logit.net", data = mydata)
> x.out <- setx(z.out)
> s.out <- sim(z.out, x = x.out)
```

Examples

1. Basic Example

   Load the sample data (see `?friendship` for details on the structure of the network dataframe):

   ```
   > data(friendship)
   ```

   Estimate model:

   ```
   > z.out <- zelig(friends ~ advice + prestige + perpower, model = "logit.net",
   +     data = friendship)
   > summary(z.out)
   ```

   Setting values for the explanatory variables to their default values:
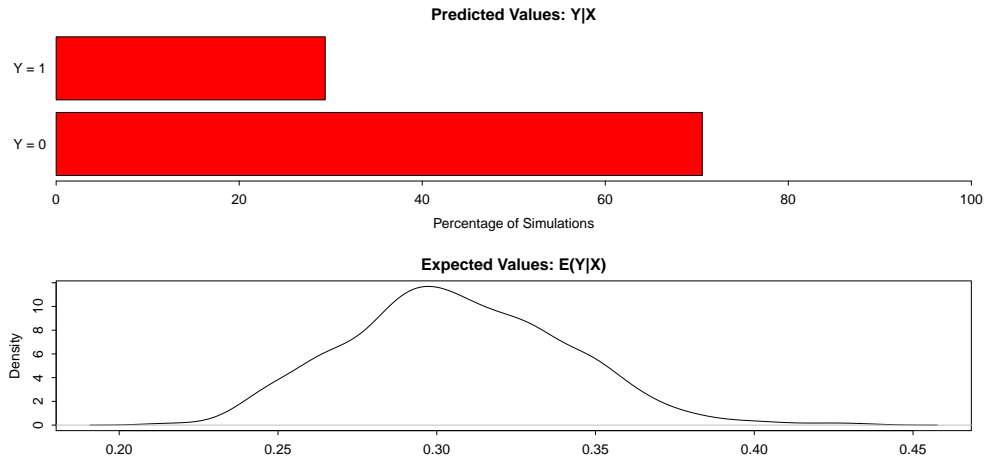
   ```
   > x.out <- setx(z.out)
   ```

   Simulating quantities of interest from the posterior distribution.
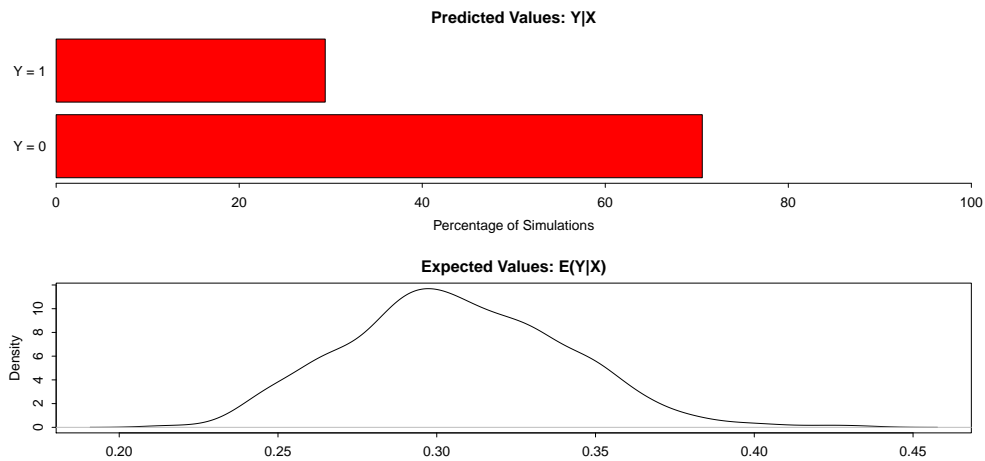
   ```
   > s.out <- sim(z.out, x = x.out)
   > summary(s.out)
   > plot(s.out)
   ```

2. Simulating First Differences

   Estimating the risk difference (and risk ratio) between low personal power (25th percentile) and high personal power (75th percentile) while all the other variables are held at their default values.

Predicted Values: Y|X

Y = 1

Y = 0

Percentage of Simulations

Expected Values: E(Y|X)

Density

```
> x.high <- setx(z.out, perpower = quantile(friendship$perpower,
+     prob = 0.75))
> x.low <- setx(z.out, perpower = quantile(friendship$perpower,
+     prob = 0.25))
> s.out2 <- sim(z.out, x = x.high, x1 = x.low)
> summary(s.out2)
> plot(s.out2)
```

Predicted Values: Y|X

Y = 1

Y = 0

Percentage of Simulations

Expected Values: E(Y|X)

Density

Model

The `logit.net` model performs a logistic regression of the proximity matrix $\mathbf{Y}$, a $m \times m$ matrix representing network ties, on a set of proximity matrices $\mathbf{X}$. This network regression model is directly analogous to standard logistic regression element-wise on the appropriately vectorized matrices. Proximity matrices are vectorized by creating Y, a $m^2 \times 1$ vector to

2

represent the proximity matrix. The vectorization which produces the Y vector from the $\mathbf{Y}$ matrix is performed by simple row-concatenation of $\mathbf{Y}$. For example, if $\mathbf{Y}$ is a 15×15 matrix, the $\mathbf{Y}_{1,1}$ element is the first element of Y, and the $\mathbf{Y}_{2,1}$ element is the second element of Y and so on. Once the input matrices are vectorized, standard logistic regression is performed.

Let Y_i be the binary dependent variable, produced by vectorizing a binary proximity matrix, for observation i which takes the value of either 0 or 1.

- The *stochastic component* is given by

$$Y_i \;\sim\; \text{Bernoulli}(y_i|\pi_i)$$
$$= \pi_i^{y_i}(1-\pi_i)^{1-y_i}$$

  where $\pi_i = \Pr(Y_i = 1)$.

- The *systematic component* is given by:

$$\pi_i = \frac{1}{1+\exp(-x_i\beta)}.$$

  where x_i is the vector of k covariates for observation i and β is the vector of coefficients.

Quantities of Interest

The quantities of interest for the network logistic regression are the same as those for the standard logistic regression.

- The expected values (`qi$ev`) for the `logit.net` model are simulations of the predicted probability of a success:

$$E(Y) = \pi_i = \frac{1}{1+\exp(-x_i\beta)},$$

  given draws of β from its sampling distribution.

- The predicted values (`qi$pr`) are draws from the Binomial distribution with mean equal to the simulated expected value π_i.

- The first difference (`qi$fd`) for the network logit model is defined as

$$FD = \Pr(Y = 1|x_1) - \Pr(Y = 1|x)$$

Output Values

The output of each Zelig command contains useful information which you may view. For example, you run `z.out <- zelig(y ~ x, model = "logit.net", 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 stored in `z.out`, you may extract:

  - `coefficients`: parameter estimates for the explanatory variables.
  - `fitted.values`: the vector of fitted values for the explanatory variables.
  - `residuals`: the working residuals in the final iteration of the IWLS fit.
  - `linear.predictors`: the vector of $x_i\beta$.
  - `aic`: Akaikeś Information Criterion (minus twice the maximized log-likelihood plus twice the number of coefficients).
  - `bic`: the Bayesian Information Criterion (minus twice the maximized log-likelihood plus the number of coefficients times $\log n$).
  - `df.residual`: the residual degrees of freedom.
  - `df.null`: the residual degrees of freedom for the null model.
  - `zelig.data`: the input data frame if `save.data = TRUE`

- From `summary(z.out)`(as well as from `zelig()`), you may extract:

  - `mod.coefficients`: the parameter estimates with their associated standard errors, p-values, and t statistics.
  - `cov.scaled`: a $k \times k$ matrix of scaled covariances.
  - `cov.unscaled`: a $k \times k$ matrix of unscaled covariances.

- From the `sim()` output stored in `s.out`, you may extract:

  - `qi$ev`: the simulated expected probabilities for the specified values of `x`.
  - `qi$pr`: the simulated predicted values for the specified values of `x`.
  - `qi$fd`: the simulated first differences in the expected probabilities simulated from `x` and `x1`.

How to Cite

To cite the *logit.net* Zelig model:

Skyler J. Cranmer. 2007. "logit.net: Social Network Logistic Regression for Dichotomous Dependent Variables" 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

The network logistic regression is part of the `netglm` package by Skyler J. Cranmer and is built using some of the functionality of the `sna` package by Carter T. Butts (Butts and Carley 2001).In addition, advanced users may wish to refer to `help(netgamma)`. Sample data are fictional.

Bibliography

Butts, C. and Carley, K. (2001), "Multivariate Methods for Interstructural Analysis," Tech. rep., CASOS working paper, Carnegie Mellon University.