0.1 gamma.net: Network Gamma Regression for Continuous, Positive Proximity Matrix Dependent Variables

Use the network gamma regression model if you have a positive-valued dependent variable that is a binary valued proximity matrix (a.k.a. sociomatricies, adjacency matrices, or matrix representations of directed graphs). The gamma distribution assumes that all waiting times are complete by the end of the study (censoring is not allowed).

Syntax

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

Additional Inputs

In addition to the standard inputs, `zelig()` takes the following additional options for network gamma regression:

- `LF`: specifies the link function to be used for the network gamma regression. Default is `LF="inverse"`, but `LF` can also be set to `"identity"` or `"log"` by the user.

Examples

1. Basic Example

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

   ```
   > data(friendship)
   ```

   Estimate model:

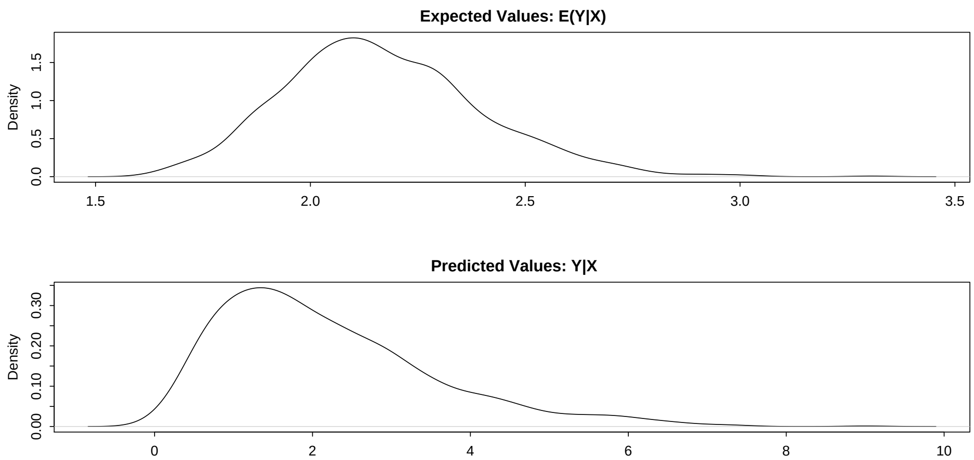
   ```
   > z.out <- zelig(per ~ perpower, LF = "inverse", model = "gamma.net",
   +     data = friendship)
   > summary(z.out)
   ```

   Setting values for the explanatory variables to their default values:

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

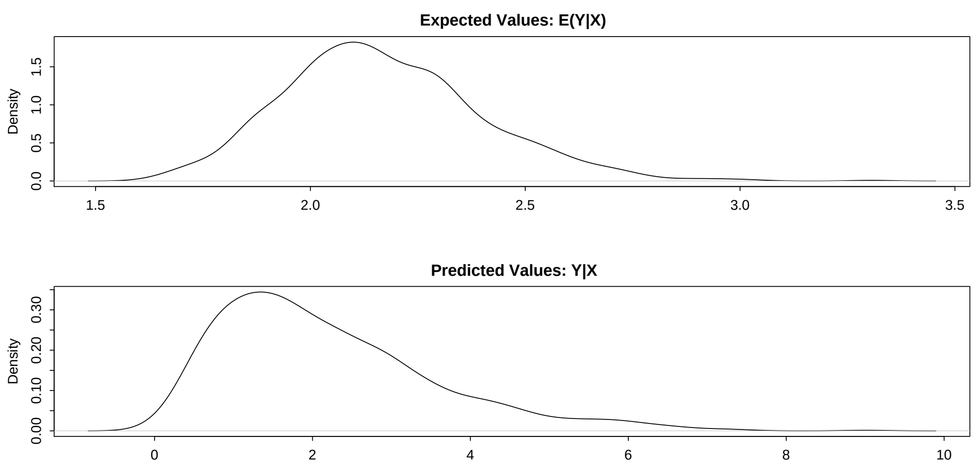
   Simulating quantities of interest from the posterior distribution.

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



2. Simulating First Differences

```
> x.low <- setx(z.out, numst2 = 0)
> x.high <- setx(z.out, numst2 = 1)
> s.out2 <- sim(z.out, x = x.low, x1 = x.high)
> summary(s.out2)
> plot(s.out2)
```



Model

The `gamma.net` model performs a gamma 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 gamma regression element-wise on the appropriately vectorized matrices. Proximity matrices are vectorized by creating Y, a $m^2 \times 1$ vector to 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 gamma regression is performed.

Let Y_i be the dependent variable, produced by vectorizing a binary proximity matrix, for observation i.

- The Gamma distribution with scale parameter α has a *stochastic component* given by

$$\begin{aligned} Y &\sim \text{Gamma}(y_i|\lambda_i, \alpha) \\ f(y) &= \tfrac{1}{\alpha^{\lambda_i}\Gamma\lambda_i} y_i^{\lambda_i - 1} \exp - \left[\tfrac{y_i}{\alpha}\right] \end{aligned}$$

  for $\alpha, \lambda_i, y_i > 0$.

- The *systematic component* is given by:

$$\lambda_i = \frac{1}{x_i \beta}.$$

Quantities of Interest

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

- The expected values (`qi$ev`) are simulations of the mean of the stochastic component given draws of α and β from their posteriors:

$$E(Y) = \alpha_i \lambda.$$

- The predicted values (`qi$pr`) are draws from the gamma distribution for each set of parameters (α, λ_i).

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

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

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 = "gamma.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 drawn from a distribution defined by (α_i, λ).
  - `qi$fd`: the simulated first differences in the expected probabilities simulated from `x` and `x1`.

How to Cite

To cite the *gamma.net* Zelig model:

> Skyler J. Cranmer. 2007. "gamma.net: Network Gamma Regression for Continuous, Positive Proximity Matrix 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`.

> Kosuke Imai, Gary King, and Olivia Lau. 2008. "Toward A Common Framework for Statistical Analysis and Development," *Journal of Computational and Graphical Statistics*, forthcoming, `http://gking.harvard.edu/files/abs/z-abs.shtml`.

See also

The network gamma 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(gamma.net)`. Sample data are fictional.

Bibliography

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