

0.1 negbin: Negative Binomial Regression for Event Count Dependent Variables

Use the negative binomial regression if you have a count of events for each observation of your dependent variable. The negative binomial model is frequently used to estimate over-dispersed event count models.

Syntax

```
> z.out <- zelig(Y ~ X1 + X2, model = "negbin", 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 negative binomial regression:

- **robust**: defaults to **FALSE**. If **TRUE** is selected, `zelig()` computes robust standard errors via the **sandwich** package (see `?`). The default type of robust standard error is heteroskedastic and autocorrelation consistent (HAC), and assumes that observations are ordered by time index.

In addition, **robust** may be a list with the following options:

- **method**: Choose from
 - * **"vcovHAC"**: (default if **robust** = **TRUE**) HAC standard errors.
 - * **"kernHAC"**: HAC standard errors using the weights given in `?`).
 - * **"weave"**: HAC standard errors using the weights given in `?`).
- **order.by**: defaults to **NULL** (the observations are chronologically ordered as in the original data). Optionally, you may specify a vector of weights (either as **order.by** = **z**, where **z** exists outside the data frame; or as **order.by** = **~z**, where **z** is a variable in the data frame). The observations are chronologically ordered by the size of **z**.
- **...**: additional options passed to the functions specified in **method**. See the **sandwich** library and `?` for more options.

Example

Load sample data:

```
> data(sanction)
```

Estimate the model:

```
> z.out <- zelig(num ~ target + coop, model = "negbin", data = sanction)
```

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

Set values for the explanatory variables to their default mean values:

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

Simulate fitted values:

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

```
> summary(s.out)
```

Model

Let Y_i be the number of independent events that occur during a fixed time period. This variable can take any non-negative integer value.

- The negative binomial distribution is derived by letting the mean of the Poisson distribution vary according to a fixed parameter ζ given by the Gamma distribution. The *stochastic component* is given by

$$\begin{aligned} Y_i \mid \zeta_i &\sim \text{Poisson}(\zeta_i \mu_i), \\ \zeta_i &\sim \frac{1}{\theta} \text{Gamma}(\theta). \end{aligned}$$

The marginal distribution of Y_i is then the negative binomial with mean μ_i and variance $\mu_i + \mu_i^2/\theta$:

$$\begin{aligned} Y_i &\sim \text{NegBin}(\mu_i, \theta), \\ &= \frac{\Gamma(\theta + y_i)}{y! \Gamma(\theta)} \frac{\mu_i^{y_i} \theta^\theta}{(\mu_i + \theta)^{\theta + y_i}}, \end{aligned}$$

where θ is the systematic parameter of the Gamma distribution modeling ζ_i .

- The *systematic component* is given by

$$\mu_i = \exp(x_i \beta)$$

where x_i is the vector of k explanatory variables and β is the vector of coefficients.

Quantities of Interest

- The expected values (`qi$ev`) are simulations of the mean of the stochastic component. Thus,

$$E(Y) = \mu_i = \exp(x_i\beta),$$

given simulations of β .

- The predicted value (`qi$pr`) drawn from the distribution defined by the set of parameters (μ_i, θ) .
- The first difference (`qi$fd`) is

$$\text{FD} = E(Y|x_1) - E(Y | x)$$

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

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

where t_i is a binary explanatory variable defining the treatment ($t_i = 1$) and control ($t_i = 0$) groups. Variation in the simulations are due to uncertainty in simulating $E[Y_i(t_i = 0)]$, the counterfactual expected value of Y_i for observations in the treatment group, under the assumption that everything stays the same except that the treatment indicator is switched to $t_i = 0$.

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

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

where t_i is a binary explanatory variable defining the treatment ($t_i = 1$) and control ($t_i = 0$) groups. Variation in the simulations are due to uncertainty in simulating $\widehat{Y_i(t_i = 0)}$, the counterfactual predicted value of Y_i for observations in the treatment group, under the assumption that everything stays the same except that the treatment indicator is switched to $t_i = 0$.

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 = "negbin", 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`: parameter estimates for the explanatory variables.
 - `theta`: the maximum likelihood estimate for the stochastic parameter θ .
 - `SE.theta`: the standard error for `theta`.
 - `residuals`: the working residuals in the final iteration of the IWLS fit.
 - `fitted.values`: a vector of the fitted values for the systemic component λ .
 - `linear.predictors`: a vector of $x_i\beta$.
 - `aic`: Akaike's Information Criterion (minus twice the maximized log-likelihood plus twice the number of coefficients).
 - `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)`, you may extract:
 - `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 object `s.out`, you may extract quantities of interest arranged as matrices indexed by simulation \times \mathbf{x} -observation (for more than one \mathbf{x} -observation). Available quantities are:
 - `qi$ev`: the simulated expected values given the specified values of \mathbf{x} .
 - `qi$pr`: the simulated predicted values drawn from the distribution defined by (μ_i, θ) .
 - `qi$fd`: the simulated first differences in the simulated expected values given the specified values of \mathbf{x} and $\mathbf{x}1$.
 - `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.

Contributors

The negative binomial model is part of the MASS library by William N. Venables and Brian D. Ripley. Please cite the model as:

.

In addition, advanced users may wish to refer to `help(glm.nb)` in the MASS library and `?`).

Robust standard errors are implemented via the sandwich package by Achim Zeileis. Please cite as

Sample data are from

.

Kosuke Imai, Gary King, and Olivia Lau added Zelig functionality.