# 1 `poisson`: Poisson Regression for Event Count Dependent Variables

Use the Poisson regression model if the observations of your dependent variable represents the number of independent events that occur during a fixed period of time (see the negative binomial model, Section ??, for over-dispersed event counts.).

### 1.0.1 Syntax

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

### 1.0.2 Additional Inputs

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

- `robust`: defaults to `FALSE`. If `TRUE` is selected, `zelig()` computes robust standard errors via the `sandwich` package (see [7]). 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 [1].
    - `"weave"`: HAC standard errors using the weights given in [3].
  - `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 [7] for more options.

### 1.0.3 Example

Load sample data:

```
> data(sanction)
```

Estimate Poisson model:

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

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

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

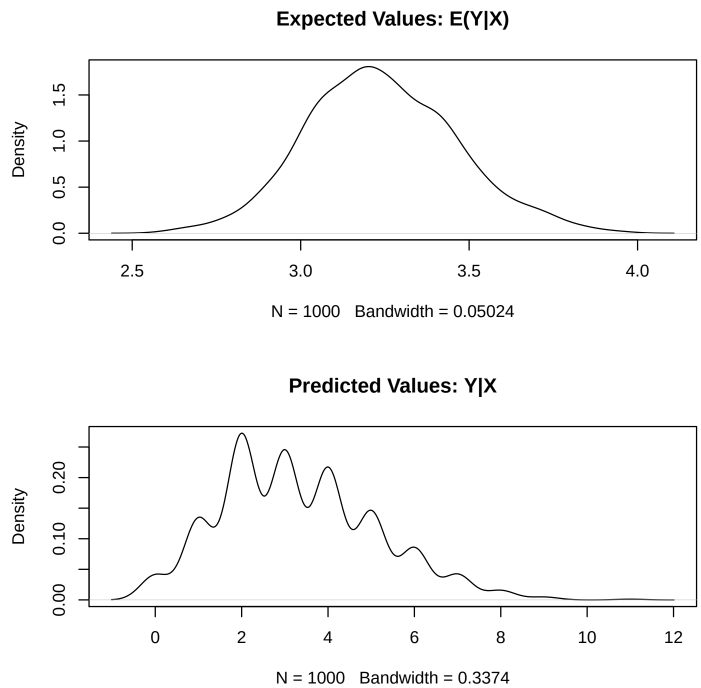
```
> x.out <- setx(z.out)
```

Simulate fitted values:

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

> summary(s.out)

> plot(s.out)
```




### 1.0.4 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.

- The Poisson distribution has *stochastic component*

$$Y_i \sim \text{Poisson}(\lambda_i),$$

where $\lambda_i$ is the mean and variance parameter.

- The *systematic component* is

$$\lambda_i = \exp(x_i \beta),$$

where $x_i$ is the vector of explanatory variables, and $\beta$ is the vector of coefficients.

#### 1.0.5 Quantities of Interest

- The expected value (`qi$ev`) is the mean of simulations from the stochastic component,

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

given draws of $\beta$ from its sampling distribution.

- The predicted value (`qi$pr`) is a random draw from the poisson distribution defined by mean $\lambda_i$.

- The first difference in the expected values (`qi$fd`) is given by:

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

- 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 $t_i = 0$.

### 1.0.6 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 = "poisson", 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.
  - `residuals`: the working residuals in the final iteration of the IWLS fit.
  - `fitted.values`: a vector of the fitted values for the systemic component $\lambda$.
  - `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 $\times$ `x`-observation (for more than one `x`-observation). Available quantities are:
  - `qi$ev`: the simulated expected values given the specified values of `x`.
  - `qi$pr`: the simulated predicted values drawn from the distributions defined by $\lambda_i$.
  - `qi$fd`: the simulated first differences in the expected values given the specified values of `x` and `x1`.
  - `qi$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 the Poisson Regression Model

Kosuke Imai, Olivia Lau, and Gary King. *poisson: Poisson Regression for Event Count Dependent Variables*, 2011

### How to Cite the Zelig Software Package

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 poisson model is part of the stats package by **(author?)** [6]. Advanced users may wish to refer to `help(glm)` and `help(family)`, as well as [5]. Robust standard errors are implemented via the sandwich package by **(author?)** [7]. Sample data are from [4].

## References


[1] Donald W.K. Andrews. Heteroskedasticity and autocorrelation consistent covariance matrix estimation. *Econometrica*, 59(3):817–858, May 1991.

[2] Kosuke Imai, Olivia Lau, and Gary King. *poisson: Poisson Regression for Event Count Dependent Variables*, 2011.

[3] Thomas Lumley and Patrick Heagerty. Weighted empirical adaptive variance estimators for correlated data regression. *jrssb*, 61(2):459–477, 1999.

[4] Lisa Martin. *Coercive Cooperation: Explaining Multilateral Economic Sanctions*. Princeton University Press, 1992. Please inquire with Lisa Martin before publishing results from these data, as this dataset includes errors that have since been corrected.

[5] Peter McCullagh and James A. Nelder. *Generalized Linear Models*. Number 37 in Monograph on Statistics and Applied Probability. Chapman & Hall, 2nd edition, 1989.

[6] William N. Venables and Brian D. Ripley. *Modern Applied Statistics with S*. Springer-Verlag, 4th edition, 2002.

[7] Achim Zeileis. Econometric computing with hc and hac covariance matrix estimators. *Journal of Statistical Software*, 11(10):1–17, 2004.