0.1 `logit`: Logistic Regression for Dichotomous Dependent Variables

Logistic regression specifies a dichotomous dependent variable as a function of a set of explanatory variables. For a Bayesian implementation, see Section ??.

Syntax

```
> z.out <- zelig(Y ~ X1 + X2, model = "logit", data = mydata)
> x.out <- setx(z.out)
> s.out <- sim(z.out, x = x.out, x1 = NULL)
```

Additional Inputs

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

- `robust`: defaults to `FALSE`. If `TRUE` is selected, `zelig()` computes robust standard errors via the `sandwich` package (see Zeileis (2004)). 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 Andrews (1991).
    - `"weave"`: HAC standard errors using the weights given in Lumley and Heagerty (1999).
  - `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 Zeileis (2004) for more options.

Examples

1. Basic Example

   Attaching the sample turnout dataset:

   ```
   > data(turnout)
   ```

Estimating parameter values for the logistic regression:

```
> z.out1 <- zelig(vote ~ race + educate, model = "logit", data = turnout)
```

Setting values for the explanatory variables to their default values:

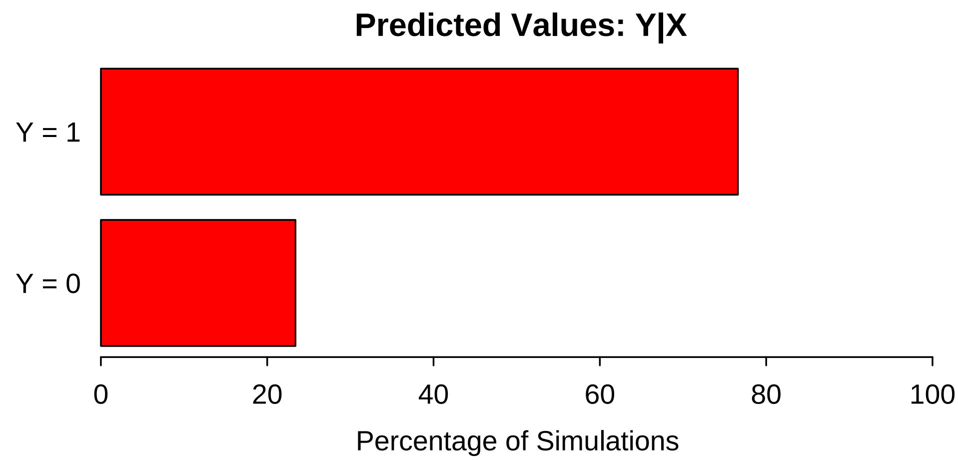
```
> x.out1 <- setx(z.out1)
```

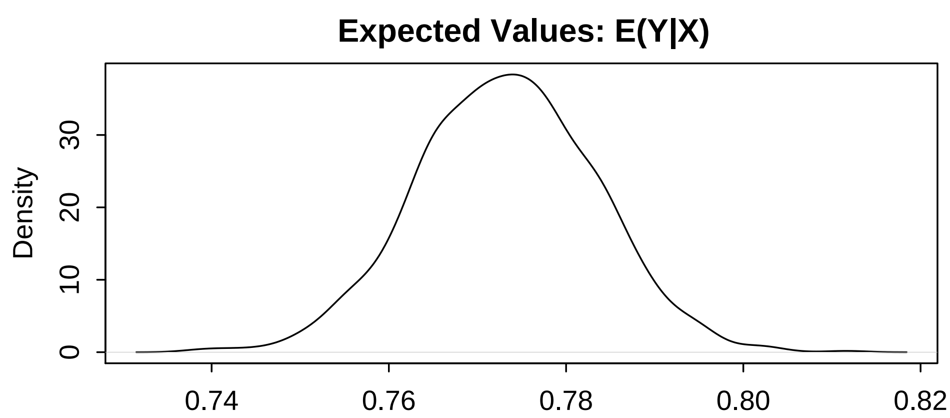
Simulating quantities of interest from the posterior distribution.

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

> summary(s.out1)

> plot(s.out1)
```





2. Simulating First Differences

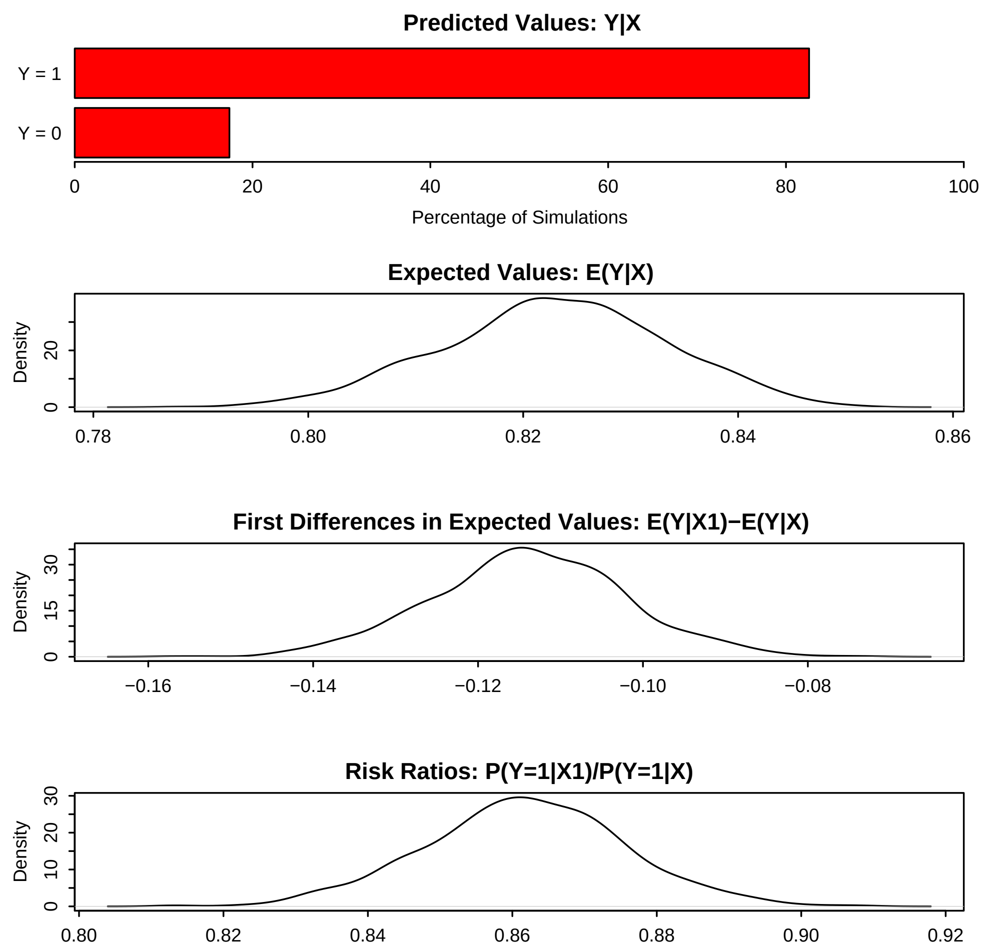
   Estimating the risk difference (and risk ratio) between low education (25th percentile) and high education (75th percentile) while all the other variables held at their default values.

```
> x.high <- setx(z.out1, educate = quantile(turnout$educate, prob = 0.75))
> x.low <- setx(z.out1, educate = quantile(turnout$educate, prob = 0.25))

> s.out2 <- sim(z.out1, x = x.high, x1 = x.low)

> summary(s.out2)

> plot(s.out2)
```

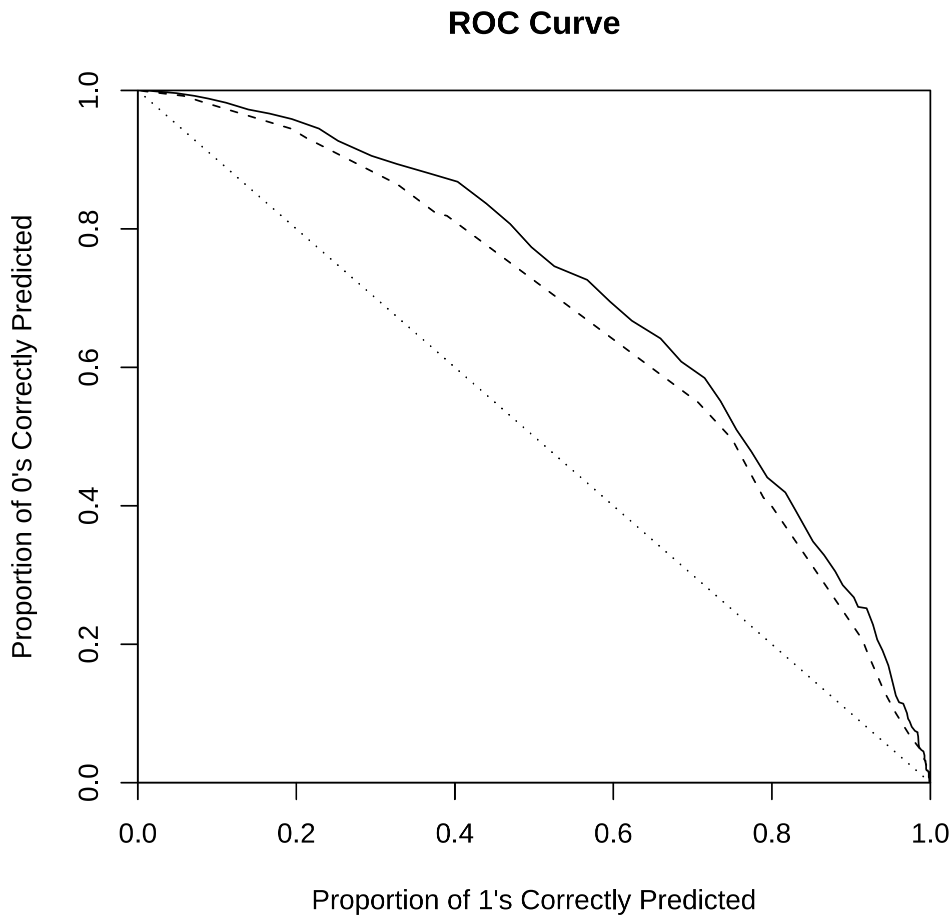


3. Presenting Results: An ROC Plot

   One can use an ROC plot to evaluate the fit of alternative model specifications. (Use `demo(roc)` to view this example, or see King and Zeng (2002).)

```
> z.out1 <- zelig(vote ~ race + educate + age, model = "logit",
+     data = turnout)
> z.out2 <- zelig(vote ~ race + educate, model = "logit", data = turnout)

> rocplot(z.out1$y, z.out2$y, fitted(z.out1), fitted(z.out2))
```



Model

Let Y_i be the binary dependent variable for observation i which takes the value of either 0 or 1.

- The *stochastic component* is given by

$$
\begin{aligned}
Y_i &\sim \text{Bernoulli}(y_i \mid \pi_i) \\
&= \pi_i^{y_i}(1-\pi_i)^{1-y_i}
\end{aligned}
$$

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 explanatory variables for observation i and β is the vector of coefficients.

Quantities of Interest

- The expected values (`qi$ev`) for the logit 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 logit model is defined as

$$\text{FD} = \Pr(Y=1 \mid x_1) - \Pr(Y=1 \mid x).$$

- The risk ratio (`qi$rr`) is defined as

$$\text{RR} = \Pr(Y=1 \mid x_1) \;/\; \Pr(Y=1 \mid x).$$

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

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

  where t_i is a binary explanatory variable defining the treatment ($t_i=1$) and control ($t_i=0$) groups. 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 = "logit", 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`: the vector of fitted values for the systemic component, π_i.
  - `linear.predictors`: the 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.
  - `data`: the name of the input data frame.
- 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$ev`: the simulated expected probabilities for the specified values of `x`.
  - `qi$pr`: the simulated predicted values for the specified values of `x`.
  - `qi$fd`: the simulated first difference in the expected probabilities for the values specified in `x` and `x1`.

- `qi$rr`: the simulated risk ratio for the expected probabilities simulated from `x` and `x1`.
- `qi$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

To cite the *logit* Zelig model:

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

> 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 gamma model is part of the base packages by William N. Venable and Brian D. Ripley (Venables and Ripley 2002). Advanced users may wish to refer to `help(glm)` and `help(family)`, as well as McCullagh and Nelder (1989). Robust standard errors are implemented via sandwich package by Achim Zeileis (Zeileis 2004).Sample data are from King et al. (2000).

Bibliography

Andrews, D. W. (1991), "Heteroskedasticity and Autocorrelation Consistent Covariance Matrix Estimation," *Econometrica*, 59, 817–858.

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.

King, G. and Zeng, L. (2002), "Improving Forecasts of State Failure," *World Politics*, 53, 623–658, http://gking.harvard.edu/files/abs/civil-abs.shtml.

Lumley, T. and Heagerty, P. (1999), "Weighted Empirical Adaptive Variance Estimators for Correlated Data Regression," *jrssb*, 61, 459–477.

McCullagh, P. and Nelder, J. A. (1989), *Generalized Linear Models*, no. 37 in Monograph on Statistics and Applied Probability, Chapman & Hall, 2nd ed.

Venables, W. N. and Ripley, B. D. (2002), *Modern Applied Statistics with S*, Springer-Verlag, 4th ed.

Zeileis, A. (2004), "Econometric Computing with HC and HAC Covariance Matrix Estimators," *Journal of Statistical Software*, 11, 1–17.