Package ‘lbm’

November 2, 2021

Type Package

Title Log Binomial Regression Model in Exact Method

Version 0.9.0.2

Author Chao Zhu

Maintainer Chao Zhu `<zhuchao9966@gmail.com>`

Description
Fit the log binomial regression model (LBM) by Exact method. Limited parameter space of LBM causes trouble to find admissible estimates and fail to converge when MLE is close to or on the boundary of space. Exact method utilizes the property of boundary vectors to re-parametrize the model without losing any information, and fits the model on the standard fitting algorithm with no convergence issues.

Depends R (>= 3.5.0)

Imports stats (>= 3.5.0), utils (>= 3.5.0)

License GPL (>= 2)

Encoding UTF-8

RoxygenNote 7.1.2

NeedsCompilation no

Repository CRAN

Date/Publication 2021-11-02 14:10:02 UTC

R topics documented:

`lbm` *Log binomial regression model in Exact method*

Description

If the maximum likelihood (ML) solution lies on the boundary of the parameter space in log binomial model, a special method is needed since the standard fitting algorithm may meet numerical difficulties. Exact method can overcome the difficulties and address the ML solution when it lies on the boundary of the parameter space.`lbm` implemented the exact method to address the ML solution in the log binomial model.

Usage

```
lbm(formula, data,contrasts = NULL,subset,na.action,lfv=0.95,
          vce = "oim",rescode=NULL,control=lbm.control(),...)
```

Arguments

`formula`	an object of class "`formula`" (or one that can be coerced to that class): a symbolic description of the model to be fitted. The details of model specification are given under 'Details'.
`data`	an optional data frame, list or environment (or object coercible by `as.data.frame` to a data frame) containing the variables in the model. If not found in `data`, the variables are taken from `environment(formula)`, typically the environment from which `lbm` is called.
`contrasts`	an optional list. See the `contrasts.arg` of `model.matrix.default`.
`subset`	a specification of the rows to be used: defaults to all rows. This can be any valid indexing vector (see `[.data.frame]` for the rows of data or if that is not supplied, a data frame made up of the variables used in formula.
`na.action`	a function which indicates what should happen when the data contain `NA`s. The default is set by the `na.action` setting of `na.action`, and is `na.fail` if that is unset. The 'factory-fresh' default is `na.omit`. Another possible value is `NULL`, no action. Value `na.exclude` can be useful.
`lfv`	a testing range option which decides the range of boundary vector candidates included for testing. The default value is 0.95, which means the covariate vectors with probability greater than 0.95 will be included into boundary pairing system as boundary vector candidates.
`vce`	the type of the information matrix used to attain the variance-covariance matrix. Two options could be selected, observed information matrix (OIM) and expected information matrix (EIM). The default `vce` is `"OIM"`. This argument only works in the data with boundary vector. If there is no boundary vector included in the data, the results are from `glm`. In the `glm`, the standard error is calculated by expected information matrix.
`rescode`	is an option to code the response variable if it is a factor.
`control`	The `control` argument of `lbm` is by default passed to the arguments of `lbm.control`.

`...`	For `lbm`, arguments to be used to form the default control argument if it is not supplied directly.

Details

A typical predictor has the form `response ~ terms` where response is the (numeric) response vector and terms is a series of terms which specifies a linear predictor for response. A terms specification of the form `first + second` indicates all the terms in first together with all the terms in second with any duplicates removed. A specification of the form `first:second` indicates the set of terms obtained by taking the interactions of all terms in first with all terms in second. The specification `first*second` indicates the cross of first and second. This is the same as `first + second + first:second`. The terms in the formula will be re-ordered so that main effects come first, followed by the interactions, all second-order, all third-order and so on: to avoid this pass a terms object as the formula.

Value

`lbm` returns an object of class inheriting from `"lbm"` which inherits from the class `"lbm"`. The function `summary` (i.e., `summary.lbm`) can be used to obtain or print a summary of the estimates and the relevant confidence interval. The argument `CF.lvl` in `summary` represents the level of confidence interval claimed in the model. The default value is `CF.lvl=0.95`. Optionally, Risk ratio estimates and their related confidence interval are offered as an argument `RR` in the `summary`. The default `RR=FALSE` is not to display them.

An object of class `"lbm"` is a list containing at least the following components:

`coefficients`	a named vector of coefficients
`residuals`	the working residuals, that is the residuals in the final iteration of the IWLS fit.
`fitted.values`	the fitted mean values, obtained by transforming the linear predictors by the inverse of the log link function.
`linear.predictors`	the linear fit on log scale.
`deviance`	twice the absolute value of maximized log-likelihood.
`aic`	A version of Akaike's An Information Criterion, minus twice the maximized log-likelihood plus twice the number of parameters, computed by the `aic` component of the family. For the binomial model, the dispersion is fixed at one and the number of parameters is the number of coefficients.
`null.deviance`	The deviance for the null model, comparable with `deviance`. The null model will only include an intercept if there is one in the model.
`df.residual`	the residual degrees of freedom.
`df.null`	the residual degrees of freedom for the null model.
`response`	the response vector used in the mode.l
`vcov`	the unscaled (`dispersion = 1`) estimated covariance matrix of the estimated coefficients.
`vce`	the type of information matrix applied.
`call`	the matched call.

`na.action`	(where relevant) information returned by `stats::model.frame` on the special handling of `NA`.
`contrasts`	(where relevant) the contrasts used.
`formula`	the formula supplied.
`factor`	the order of factors used in the response variable.
`bvector`	the matrix of boundary vectors.
`bv`	logical. Determines whether the model has boundary vectors.

References

Petersen, M. R. & Deddens, J. A. (2010). Maximum likelihood estimation of the log-binomial model. *Communications in Statistics - Theory and Methods*, 39: 5, 874 - 883.

See Also

`glm`, `lm`.

Examples

```
## Two examples are from Petersen, M. R. & Deddens, J. A. (2010).

## Example 1.
x<-c(1:10)
y<-c(0,0,0,0,1,0,1,1,1,1)
data<-data.frame(x,y)
a<-lbm(formula=y~x,data=data,vce="eim")

## Example 2.
x1<-c(1:11)
x2<-x1^2
y<-c(10,6,4,3,3,2,3,3,4,6,10)
dat<-cbind(x1,x2,y)
dat1<-apply(dat, 1, function(t) {
  temp<-data.frame(x1=rep(t[1],10),x2=rep(t[2],10),y=0)
  temp$y[1:t[3]]<-1
  return(temp)
})
data<-do.call(rbind, dat1)
a<-lbm(formula=y~x1+x2,data=data)
summary(a)
```

`lbm.control` *Auxiliary control for* `lbm`

Description

Auxiliary function for `lbm` fitting. Only used internally by `lbm`.

Usage

```
lbm.control(epsilon = 1e-8, maxit = 100)
```

Arguments

`epsilon`	positive convergence tolerance `epsilon`;
`maxit`	integer giving the maximal number of iterations.

Value

A list with components named as the arguments.

See Also

`glm.control`

Index