Package ‘icdGLM’

July 22, 2016

Type Package

Title EM by the Method of Weights for Incomplete Categorical Data in Generlized Linear Models

Version 1.0.0

Date 2016-07-21

Author Lorenz Brachtendorf <Lorenz.Brachtendorf@gmx.de>, Stephan Dlugosz <stephan.dlugosz@googlemail.com>

Maintainer Stephan Dlugosz <stephan.dlugosz@googlemail.com>

Description Provides an estimator for generalized linear models with incomplete data for discrete covariates. The estimation is based on the EM algorithm by the method of weights by Ibrahim (1990) <DOI:10.2307/2290013>.

License GPL (>= 2)

Depends R (>= 3.3.0)

Imports stats, Matrix

RoxygenNote 5.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2016-07-22 01:26:44

R topics documented:

`expand_data` *Complete incomplete data*

Description

This function fills all incomplete data with a set of possible values equally weighted. This is done in order to apply *icdglm*.

Usage

```
expand_data(data, y, missing.x, value.set, weights = rep.int(1, NROW(data)),
                    indicator = rep.int(0, NROW(data)))
```

Arguments

`data` a vector, matrix, list or data frame containing numerics. This data is checked for incompleteness and needs to contain the independent variables for a subsequent regression with n observations and k regressors. Each gap is filled with all values from `value.set`. New observations are added for each possible value.

`y` a vector of integers or numerics. This vector has to be complete and is the dependent variable for a subsequent regression.

`missing.x` a vector that contains integers and gives the position of the independent variables, for which the data will be checked for incompleteness, i.e. for a matrix the position of the corresponding columns.

`value.set` a vector of numerics containing all possible values the missing data can take. This set has to be finite.

`weights` a vector of numerics giving the initial weight of each observation. Default is 1 for each observation.

`indicator` a vector of integers that indicates which observations belong to each other. If some columns with incomplete data were already completed, this vector has to be passed here. For raw incomplete data, the function connects observations which belong to each other. Default is 0 for this vector indicating no connection.

Value

`expand_data` returns a list with the following elements:

- dataa data frame of the expanded data with all possible observations (independent variables). The dependent variable is included in the last column.
- weightsthe weights for each possible observation.
- indicatora vector which indicates which observations belong to each other. Such observations have the same integer being the indicator.

Examples

```
data(TLI.data)
          expand_data(data = TLI.data[,1:3],
          y = TLI.data[,4],
          missing.x = 1:3,
          value.set = 0:1)
```

`icdglm` *EM by the Method of Weights for Incomplete Data in GLMs*

Description

This function applies the EM algorithm by the method of weights to incomplete data in a general linearized model.

Usage

```
icdglm(formula, family = binomial(link = "logit"), data, weights = rep.int(1, NROW(data)),
              indicator = rep.int(0, NROW(data)), control = list(), model = TRUE)
```

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.

`family` a description of the error distribution and link function to be used in the model. This can be a character string naming a family function, a family function or the result of a call to a family function. (See `family` for details of family functions.)

`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)

`weights` a vector which attaches a weight to each observation. For incomplete data, this is obtained from `expand_data`.

`indicator` a vector that indicates which observations belong to each other. This is obtained from `expand_data`.

`control` a list of control characteristics used for the iteration process in *icdglm.fit*. See `glm.control` for further information how this works. Default settings are: `epsilon = 1e-10, maxit = 100, trace = FALSE`.

`model` a logical value indicating whether model frame should be included as a component of the returned value.

Value

icdglm returns an object of class inheriting from "icdglm.fit", "glm" and "lm". The function summary.icdglm can be used to obtain a summary of the results. `icdglm` returns a list with the following elements:

- xa matrix of numerics containing all independent variables
- ya vector of numerics containing the dependent variable
- new.weightsthe new weights obtained in the final iteration of *icdglm.fit*
- indicatora vector of integers indicating which observations belong to each other
- glm.fit.datatypical `glm.fit` output for the last iteration. See `glm.fit` for further information.
- coefficientsa named vector of coefficients
- qrQR Decomposition of the information matrix
- residualsthe residuals of the final iteration
- fitted.valuesthe fitted mean values, obtained by transforming the linear predictors by the inverse of the link function.
- rankthe numeric rank of the fitted linear model
- familythe family object used.
- linear.predictorsthe linear fit on link scale
- devianceup to a constant, minus twice the maximized log-likelihood. Where sensible, the constant is chosen so that a saturated model has deviance zero.
- aicsee glm
- null.devianceThe deviance for the null model, comparable with deviance. The null model will include the offset, and an intercept if there is one in the model. Note that this will be incorrect if the link function depends on the data other than through the fitted mean: specify a zero offset to force a correct calculation.
- iteran integer containing the number of iterations in *icdglm.fit* before convergence
- weightsthe working weights, that is the weights in the final iteration of the IWLS fit.
- prior.weightsthe weights initially supplied, a vector of 1s if none were.
- df.residualthe residual degrees of freedom from the initial data set
- df.nullthe residual degrees of freedom from initial data set for the null model
- modelmodel frame
- convergedTRUE if *icdglm* converged.
- callthe match call
- formulathe formula supplied
- termsthe *terms* object used
- datathe data argument
- controlthe value of the *control* argument used

References

Ibrahim, Joseph G. (1990). *Incomplete Data in Generalized Linear Models*. Journal of the American Statistical Association, Vol.85, No. 411, pp. 765 - 769.

See Also

expand_data, icdglm.fit, glm, glm.fit, glm.control, summary.glm

Examples

```
data(TLI.data)
          complete.data <- expand_data(data = TLI.data[,1:3],
                                       y = TLI.data[,4],
                                       missing.x = 1:3,
                                       value.set = 0:1)
          example <- icdglm(y ~ x1 + x2 + x3, family = binomial(link = "logit"),
                            data = complete.data$data, weights = complete.data$weights,
                            indicator = complete.data$indicator)
          summary(example)
```

icdglm.fit — *EM by the Method of Weights for Incomplete Data in GLMs (Algorithm)*

Description

This function applies the EM algorithm by the method of weights to incomplete data in a general linearized model.

Usage

```
icdglm.fit(x, y, weights = rep.int(1, NROW(x)), indicator = rep.int(0, NROW(x)),
       family = binomial(link = "logit"), control=list())
```

Arguments

`x` a vector, matrix, list or data frame containing the independent variables

`y` a vector of integers or numerics. This is the dependent variable.

`weights` a vector which attaches a weight to each observation. For incomplete data, this is obtained from `expand_data`.

`indicator` a vector that indicates which observations belong to each other. This is obtained from `expand_data`.

`family` family for glm.fit. See `glm` or `family` for more details. Default is `binomial(link = "logit")`.

`control` a list of control characteristics. See `glm.control` for further information. Default settings are: `epsilon = 1e-10, maxit = 100, trace = FALSE`.

Value

`icdglm.fit` returns a list with the following elements:

- xa matrix of numerics containing all independent variables
- ya vector of numerics containing the dependent variable
- new.weightsthe new weights obtained in the final iteration of *icdglm.fit*
- indicatora vector of integers indicating which observations belong to each other
- glm.fit.datatypical `glm.fit` output for the last iteration. See `glm.fit` for further information.
- coefficientsa named vector of coefficients
- qrQR Decomposition of the information matrix
- residualsthe residuals of the final iteration
- fitted.valuesthe fitted mean values, obtained by transforming the linear predictors by the inverse of the link function.
- rankthe numeric rank of the fitted linear model
- familythe family object used.
- linear.predictorsthe linear fit on link scale
- devianceup to a constant, minus twice the maximized log-likelihood. Where sensible, the constant is chosen so that a saturated model has deviance zero.
- aicsee glm
- null.devianceThe deviance for the null model, comparable with deviance. The null model will include the offset, and an intercept if there is one in the model. Note that this will be incorrect if the link function depends on the data other than through the fitted mean: specify a zero offset to force a correct calculation.
- iteran integer containing the number of iterations in *icdglm.fit* before convergence
- weightsthe working weights, that is the weights in the final iteration of the IWLS fit.
- prior.weightsthe weights initially supplied, a vector of 1s if none were.
- df.residualthe residual degrees of freedom from the initial data set
- df.nullthe residual degrees of freedom from initial data set for the null model
- modelmodel frame
- convergedTRUE if *icdglm* converged.
- callthe match call
- formulathe formula supplied
- termsthe *terms* object used
- datathe data argument
- controlthe value of the *control* argument used

References

Ibrahim, Joseph G. (1990). *Incomplete Data in Generalized Linear Models*. Journal of the American Statistical Association, Vol.85, No. 411, pp. 765 - 769.

See Also

expand_data, icdglm glm.fit, glm.control, summary.glm

Examples

```
data(TLI.data)
          complete.data <- expand_data(data = TLI.data[, 1:3],
                                       y = TLI.data[, 4],
                                       missing.x = 1:3,
                                       value.set = 0:1)
          example1 <- icdglm.fit(x = complete.data$data[, 1:3],
                                 y = complete.data$data[, 4],
                                 weights = complete.data$weights,
                                 indicator = complete.data$indicator,
                                 family = binomial(link = "logit"),
                                 control = list(epsilon = 1e-10,
                                                maxit = 100, trace = TRUE))
```

`Leukemia.data` *Survival Times of 33 Leukemia Patients*

Description

This data set is taken from Ibrahim, Joseph G. (1990). *Incomplete Data in Generalized Linear Models*. Journal of the American Statistical Association, Vol.85, No. 411, pp. 765 - 769. The original source is Feigl, P., and Zelen, M. (1965), *Estimation of Exponential Survival Probabilities With Concomitant Information*, Biometrics, 21, pp. 826-838.

Usage

```
data(Leukemia.data)
```

Format

A data frame with 33 observations on the following 3 variables.

`x1` a numeric vector (covariate)

`x2` a numeric vector (covariate)

`y` a numeric vector (the dependent variable)

Source

Feigl, P., and Zelen, M. (1965), *Estimation of Exponential Survival Probabilities With Concomitant Information*, Biometrics, 21, pp. 826-838. Ibrahim, Joseph G. (1990). *Incomplete Data in Generalized Linear Models*. Journal of the American Statistical Association, Vol.85, No. 411, pp. 765 - 769.

summary.icdglm *Summarizing Output of an EM Algorithm by the Method of Weights Using GLMs*

Description

This function gives a summary of the output of *icdglm*. *summary.icdglm* inherits from *summary.glm*.

Usage

```
## S3 method for class 'icdglm'
summary(object, dispersion = NULL, correlation = FALSE, symbolic.cor = FALSE, ...)
```

Arguments

`object`	an object of class "icdglm", usually, a result of a call to *icdglm*.
`dispersion`	the dispersion parameter for the family used. Either a single numerical value or NULL (the default), when it is inferred from object (see details of *summary.glm*).
`correlation`	logical, if TRUE, the correlation matrix of the estimated parameters is returned and printed.
`symbolic.cor`	logical, if TRUE, print the correlations in a symbolic form (see *symnum*) rather than as numbers.
`...`	further arguments passed to or from other methods.

Value

summary.icdglm returns an object of class "*summary.icdglm*", a list with components:

- callfunction call of *object*
- termsthe *terms* object used.
- familythe component from *object*
- deviancethe component from *object*
- aicthe component from *object*
- df.residualthe residual degrees of freedom of the initial data set
- null.deviancethe component from *object*
- df.nullthe residual degrees of freedom for the null model.
- iterthe number of iterations in *icdglm.fit*, component from *object*
- deviance.residthe deviance residuals: see *residuals.glm*
- coefficientsthe matrix of coefficients, (corrected) standard errors, t-values and p-values.
- aliasednamed logical vector showing if the original coefficients are aliased.
- dispersioneither the supplied argument or the inferred/estimated dispersion if the latter is NULL.

- dfa 3-vector of the rank of the model and the number of residual degrees of freedom, plus number of coefficients (including aliased ones).
- cov.unscaledthe unscaled (dispersion = 1) estimated covariance matrix of the estimated coefficients.
- cov.scaledditto, scaled by dispersion
- correlation(only if *correlation* is *TRUE*) The estimated correlations of the estimated coefficients.
- symbolic.cor(only if *correlation* is *TRUE*) The value of the argument symbolic.cor.

Note

The description of this function is taken from *summary.glm* apart from a few differences.

See Also

`icdglm`, `summary.glm`, `summary`, `glm`

`TLI.data` *TLI Study of 82 Patients*

Description

This data set is taken from Ibrahim, Joseph G. (1990). *Incomplete Data in Generalized Linear Models*. Journal of the American Statistical Association, Vol.85, No. 411, pp. 765 - 769. The original source is Colice, G. L., Stukel, T. A., and Dain, B. (1989), *Laryngeal Complications of Prolonged Intubation*, Chest, No. 96, pp. 877-884.

Usage

```
data(TLI.data)
```

Format

A data frame with 82 observations on the following 4 variables (3 independent variables and 1 dependent variable).

`x1` a numeric vector

`x2` a numeric vector

`x3` a numeric vector

`y` a numeric vector (the dependent variable)

Source

Colice, G. L., Stukel, T. A., and Dain, B. (1989), *Laryngeal Complications of Prolonged Intubation*, Chest, No. 96, pp. 877-884. Ibrahim, Joseph G. (1990). *Incomplete Data in Generalized Linear Models*. Journal of the American Statistical Association, Vol.85,

Index