Package 'multiModTest'

August 1, 2026

Type Package

Title Information Assessment for Individual Modalities in Multimodal Regression Models

Version 1.1

Description Provides methods for quantifying the information gain contributed by individual modalities in multimodal regression models. Information gain is measured using Expected Relative Entropy (ERE) or pseudo-R^2 metrics, with corresponding confidence intervals. Currently supports linear regression, logistic regression, and the Cox proportional hazards model. A robust Median-of-Means based estimator is also provided for heavy-tailed responses under the Gaussian and Negative-Binomial families, with basic bootstrap confidence intervals.

License GPL-3

Encoding UTF-8

RoxygenNote 7.3.2

Imports MASS, SIS, glmnet, ncvreg, MBESS, survival, dplyr

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Depends R (>= 3.6.0)

NeedsCompilation no

Author Wanting Jin [aut, cre],
Quefeng Li [aut]

Maintainer Wanting Jin <jinwanting5@gmail.com>

LazyData true

Repository CRAN

Date/Publication 2026-08-01 02:50:02 UTC

Contents

`data_cox_model` *Example Dataset*

Description

A toy dataset to demonstrate running this package on multimodal Cox proportional hazards models. Survival times follow a Cox model with independent right censoring (about 27% of the observations are censored).

Usage

```
data_cox_model
```

Format

A data object that contains

`y` A 200 $\times$ 2 matrix of survival outcomes, with columns `time` (the observed follow-up time) and `status` (the event indicator: 1 = event, 0 = censored). It may be passed directly to `mglm.test`, or converted with `Surv`.

`X` A 200 $\times$ 800 matrix containing all training data.

`mod.idx` A list of modality indices: columns 1–400 form modality 1 and columns 401–800 form modality 2.

`data_linear_model` *Example Dataset*

Description

A toy dataset to demonstrate running this package on multimodal linear models.

Usage

```
data_linear_model
```

Format

A data object that contains

`y` A vector of 200 observations of continuous outcomes.

`X` A 200 $\times$ 600 matrix containing all training data.

`mod.idx` A list of modality indices.

data_logistic_model *Example Dataset*

Description

A toy dataset to demonstrate running this package on multimodal logistic models.

Usage

```
data_logistic_model
```

Format

A data object that contains

`y` A vector 200 observations of outcomes. (0 or 1)

`X` A 200 $\times$ 600 matrix containing all training data.

`mod.idx` A list of modality indices.

data_nb_model *Example Dataset*

Description

A toy dataset to demonstrate robust (Median-of-Means) estimation on multimodal Negative-Binomial models with heavy-tailed responses. The counts are strongly overdispersed (variance-to-mean ratio near 19, with roughly half of the observations equal to zero).

Usage

```
data_nb_model
```

Format

A data object that contains

`y` A vector of 300 observations of non-negative counts.

`X` A 300 $\times$ 600 matrix containing all training data.

`mod.idx` A list of modality indices: columns 1–200, 201–400, and 401–600 form modalities 1, 2, and 3.

`theta` The true Negative-Binomial dispersion used to generate the counts, $\theta = 0.5$. It may be supplied to `mglm.test` through the `theta` argument; otherwise it is estimated from the data.

mglm.test *Modality Assessment in Multimodal Generalized Linear Models*

Description

Provides statistical inference for modality-specific information gain in multimodal GLMs. Estimates ERE and pseudo-R^2 with confidence intervals using Sure Independence Screening for variable selection and penalized likelihood for inference.

Usage

```
mglm.test(
  X,
  y,
  mod.idx,
  family = c("gaussian", "binomial", "cox", "negative.binomial"),
  robust = c("none", "mom"),
  iter = TRUE,
  penalty = c("SCAD", "MCP", "lasso"),
  tune = c("bic", "ebic", "aic"),
  lambda = NULL,
  nlambda = 100,
  conf.level = 0.95,
  CI.type = c("two.sided", "one.sided"),
  CI = NULL,
  nboot = 200,
  nblock = 5,
  theta = NULL,
  trace = FALSE
)
```

Arguments

X	The finite numeric $n \times p$ data matrix consisting of features from all modalities.
y	The response. For family = "gaussian" or "binomial" this is the length-n vector (or one-column matrix/data frame) of finite responses; binomial responses must contain both 0 and 1. Negative-binomial responses must be non-negative integer counts. For family = "cox" this is a Surv object, or equivalently an $n \times 2$ matrix / data frame whose first column is the observed time and whose second column is the event indicator.
mod.idx	A non-empty list of disjoint column-index vectors for all modalities in the concatenated data matrix X. Every column of X must be assigned to exactly one modality.
family	A description of the error distribution and link function to be used in the model. With robust = "none" the Gaussian ("gaussian") and Binomial ("binomial") families with canonical links and the Cox proportional hazards model ("cox")

are available. With robust = "mom" the Gaussian ("gaussian") and Negative-Binomial ("negative.binomial") families are available.

robust — Estimator for the modality information gain. "none" (default) uses the standard maximum-likelihood / partial-likelihood estimator with closed-form non-central chi-square confidence intervals. "mom" uses the Median-of-Means gradient-descent (MoM-GD) estimator, which is robust to heavy-tailed responses, with basic bootstrap confidence intervals.

iter — Specifies whether to perform iterative SIS. For the Cox family this toggles between a single BIC-driven screening pass (iter=FALSE) and the full iterative (ISIS) procedure (iter=TRUE). The default is iter=TRUE.

penalty — Specifies the type of penalty to be used in the variable selection and inference procedure. Options include 'MCP', 'SCAD', and 'lasso'. The default is penalty='SCAD'.

tune — Specifies the method for selecting the optimal tuning parameters in (I)SIS and penalized likelihood procedure. Options include 'bic', 'ebic' and 'aic'. The default is tune='bic'.

lambda — A user-specified finite, positive, strictly decreasing sequence of at least two lambda values for penalized likelihood procedure. By default, a sequence of values of length nlambda is automatically computed and equally spaced on the log scale.

nlambda — The number of lambda values, an integer of at least 2. The default is 100.

conf.level — Confidence level, a finite number strictly between 0 and 1. The default is conf.level=0.95.

CI.type — A string specifying the type of the confidence interval. Options include 'two.sided' and 'one.sided'. The default is CI.type='two.sided'.

CI — Logical; whether to compute confidence intervals. When robust = "none" the default is TRUE (closed-form non-central chi-square intervals). When robust = "mom" the default is FALSE; set CI = TRUE to run the basic bootstrap. If CI = FALSE, only point estimates are returned (CI bounds are NA).

nboot — Number of bootstrap resamples (at least 2) used for the MoM basic bootstrap confidence interval (used only when robust = "mom" and CI = TRUE). The default is 200.

nblock — Number of Median-of-Means blocks used by the MoM-GD estimator (used only when robust = "mom"). Must be an integer between 1 and nrow(X). The rows of X are partitioned into nblock contiguous, near-equal blocks. The default is 5.

theta — Negative-Binomial dispersion parameter (used only when family = "negative.binomial"), a finite positive number treated as fixed in the MoM fit. If NULL (default), it is estimated once from the screened design.

trace — Specifies whether to print out logs of iterations in SIS procedure. The default is trace=FALSE.

Value

An object with S3 class "mglm.test" containing:

sel.idx	List of indices of selected features by (I)SIS in each modality.
num.nonzeros	Number of selected features by (I)SIS in each modality.
ERE	Point estimation of ERE for each modality.
ERE.CI.L	Lower bound of the confidence interval of ERE for each modality
ERE.CI.U	Upper bound of the confidence interval of ERE for each modality
R2	Point estimate of pseudo-R^2 for each modality.
R2.CI.L	Lower bound of the confidence interval of pseudo-R^2 for each modality
R2.CI.U	Upper bound of the confidence interval of pseudo-R^2 for each modality
conf.level	Level of confidence intervals.

Examples

```
## Example 1: Linear model
data(data_linear_model)
X <- data_linear_model$X
y <- data_linear_model$y
mod.idx <- data_linear_model$mod.idx
test <- mglm.test(X = X, y = y, mod.idx = mod.idx, family = "gaussian",
               iter = TRUE, penalty = "SCAD", tune = "bic",
               conf.level = 0.95, CI.type = "one.sided")
summary(test)


## Example 2: Logistic regression
data(data_logistic_model)
X <- data_logistic_model$X
y <- data_logistic_model$y
mod.idx <- data_logistic_model$mod.idx
test <- mglm.test(X = X, y = y, mod.idx = mod.idx, family = "binomial",
               iter = TRUE, penalty = "SCAD", tune = "bic",
               conf.level = 0.95, CI.type = "two.sided")
sum.test <- summary(test)


## Example 3: Cox proportional hazards model

data(data_cox_model)
X <- data_cox_model$X
y <- data_cox_model$y          # two-column matrix: (time, status)
mod.idx <- data_cox_model$mod.idx
test <- mglm.test(X = X, y = y, mod.idx = mod.idx, family = "cox",
               iter = TRUE, penalty = "SCAD", tune = "ebic",
               conf.level = 0.95, CI.type = "two.sided")
summary(test)



## Example 4: Robust (MoM) estimation under heavy-tailed responses
data(data_nb_model)
```

```
X <- data_nb_model$X
y <- data_nb_model$y
mod.idx <- data_nb_model$mod.idx
## Point estimates only (CI = FALSE is the default when robust = "mom"):
test <- mglm.test(X = X, y = y, mod.idx = mod.idx,
                family = "negative.binomial", robust = "mom",
                iter = FALSE, penalty = "SCAD", tune = "bic",
                nblock = 5, theta = data_nb_model$theta)
summary(test)
```

`summary.mglm.test` *Summary method for objects of class "mglm.test"*

Description

Summary method for objects of class "mglm.test"

Usage

```
## S3 method for class 'mglm.test'
summary(object, ...)

## S3 method for class 'summary.mglm.test'
print(x, ...)
```

Arguments

`object`	An `mglm.test` object.
`...`	Additional arguments that could be passed to `summary.mglm.test`.
`x`	A `summary.mglm.test` object.

Value

An object with S3 class `summary.mglm.test`. The class has its own print method and contains the following list of elements.

`sum.ERE`	The summary table of point estimate and confidence interval of ERE for each modality.
`sum.R2`	The summary table of point estimate and confidence interval of pseudo-R^2 for each modality.
`conf.level`	Level of confidence intervals.
`sel.mod`	Index of the most informative modality.

Index