

Package: LogisticCopula (via r-universe)

September 27, 2024

Type Package

Title A Copula Based Extension of Logistic Regression

Version 0.1.0

Maintainer Simon Boge Brant <simbrant91@gmail.com>

Description An implementation of a method of extending a logistic regression model beyond linear effects of the co-variates. The extension is constructed by first equating the logistic regression model to a naive Bayes model where all the margins are specified to follow natural exponential distributions conditional on Y, that is, a model for Y given X that is specified through the distribution of X given Y, where the columns of X are assumed to be mutually independent conditional on Y. Subsequently, the model is expanded by adding vine-copulas to relax the assumption of mutual independence, where pair-copulas are added in a stage-wise, forward selection manner. Some heuristics are employed during the process of selecting edges, as well as the families of pair-copula models. After each component is added, the parameters are updated by a (smaller) number of gradient steps to maximise the likelihood. When the algorithm has stopped adding edges, based the criterion that a new edge should improve the likelihood more than k times the number new parameters, the parameters are updated with a larger number of gradient steps, or until convergence.

License MIT + file LICENSE

Imports VineCopula (>= 2.5.0), rvinecopulib (>= 0.6.3.1.1), igraph (>= 2.0.3), numDeriv (>= 8-1.1), stringr (>= 1.5.1)

Depends R (>= 3.5.0), brglm2 (>= 0.9)

Encoding UTF-8

LazyData true

RoxygenNote 7.3.1

Author Simon Boge Brant [aut, cre], Ingrid Hobæk Haff [aut]

NeedsCompilation no

Repository CRAN
Date/Publication 2024-06-28 09:20:06 UTC

Contents

fit_copula_interactions	2
fit_model	4
Ionosphere	5
predict.logistic_copula	6
Index	7

fit_copula_interactions
<i>fit_copula_interactions</i>

Description

This is the main function of the package, which starting from an initial logistic regression model with only main effects of each covariate, selects and fits interaction terms in the form of two R-vine models with identical graphical structure, one for each class.

Usage

```
fit_copula_interactions(  
  y,  
  x,  
  xtype,  
  family_set = c("gaussian", "clayton", "gumbel"),  
  oos_validation = FALSE,  
  tau = 2,  
  which_include = NULL,  
  reg.method = "glm",  
  maxit_final = 1000,  
  maxit_intermediate = 50,  
  verbose = FALSE,  
  adjust_intercept = TRUE,  
  max_t = Inf,  
  test_x = NULL,  
  test_y = NULL,  
  set_nonsig_zero = FALSE,  
  reltol = sqrt(.Machine$double.eps)  
)
```

Arguments

<code>y</code>	A vector of n observations of the (univariate) binary outcome variable y
<code>x</code>	A ($n \times p$) matrix of n observations of p covariates
<code>xtype</code>	A vector of p characters that have to take the value "c_a", "c_p", "d_b" or "d_b", to indicate whether each margin of the is continuous with full support, continuous with support on the positive real line, discrete (binary) or a counting variable.
<code>family_set</code>	A vector of strings that specifies the set of pair-copula families that the fitting algorithm chooses from. For an overview of which values that can be specified, see the documentation for bicop .
<code>oos_validation</code>	Whether to use an external sample for validation instead of an in-sample likelihood based criteria. Would require that both <code>test_x</code> and <code>test_y</code> are provided if set to TRUE.
<code>tau</code>	Parameter used when selecting the structure, where the criteria is (<code>new_likelihood - previous_likelihood - tau</code>), so that an additional edge in the copulas is only accepted if it leads to an increase in the likelihood that exceeds <code>tau</code> . Setting <code>tau</code> to NULL, has the same effect as <code>-Inf</code> .
<code>which_include</code>	The column indices of the covariates that could be included in the copula effects.
<code>reg.method</code>	The method by which the initial regression coefficients are fitted.
<code>maxit_final</code>	The maximum number of gradient optimisation iterations to use when the full structure has been selected to refit all the parameters. Defaults to 1000.
<code>maxit_intermediate</code>	The maximum number of gradient optimisation iterations to use when adding a newly selected component to refit the parameters. Defaults to 10.
<code>verbose</code>	Whether information about the progress should be printed to the console.
<code>adjust_intercept</code>	Whether to intermediately refit the intercept during the model/structure selection procedure. Defaults to true.
<code>max_t</code>	The maximum number of trees in the copula models. Defaults to <code>Inf</code> , i.e., no maximum.
<code>test_x</code>	Part of the optional validation set, see <code>@oos_validation</code> .
<code>test_y</code>	Part of the optional validation set, see <code>@oos_validation</code> .
<code>set_nonsig_zero</code>	If true, non-significant regression coefficients (in the initial glm model) will be set to zero
<code>reltol</code>	Relative convergence tolerance, see the documentation for optim .

Value

A `logistic_copula` object, which contains the regression coefficients of the model, the parameters of the chosen conditional covariate distribution that corresponds to the regression coefficients, and the pair of vine-models that extend the logistic regression model.

Examples

```
data("Ionosphere")

dset <- Ionosphere[, -(1:2)]

set.seed(20)
rowss <- sample(nrow(dset), round(nrow(dset) * 0.75))
colss <- sample(ncol(dset) - 1, 5)
x <- as.matrix(dset[rowss, colss])
xte <- as.matrix(dset[-rowss, colss])
y <- dset[rowss, ncol(dset)] == "bad"
yte <- dset[-rowss, ncol(dset)] == "bad"

xtype <- apply(x, 2, function(x) if(length(unique(x)) > 2) "c_a" else "d")

# Model with selection penalty tau=log(n)
md <- LogisticCopula::fit_copula_interactions(
  y, as.matrix(x), xtype, tau = log(nrow(x))
)
# Model with selection penalty tau=Inf, returns just the logistic
# regression model
mdglm <- LogisticCopula::fit_copula_interactions(
  y, as.matrix(x), xtype, tau = Inf
)

plot(predict(mdglm, xte), predict(md, xte), col = 3 + yte)
```

fit_model

fit_model

Description

This function updates the parameters of a LogisticCopula model by maximum likelihood.

Usage

```
fit_model(
  y,
  x,
  m_obj,
  maxit = 5,
  num_grad = FALSE,
  verbose = FALSE,
  hessian = FALSE,
  reltol = sqrt(.Machine$double.eps)
)
```

Arguments

<code>y</code>	A vector of n observations of the (univariate) binary outcome variable y
<code>x</code>	A $(n \times p)$ matrix of n observations of p covariates
<code>m_obj</code>	The model object as returned from <code>fit_copula_interactions</code>
<code>maxit</code>	The maximum number of gradient steps
<code>num_grad</code>	Whether to compute gradients numerically.
<code>verbose</code>	Whether information about the progress should be printed to the console.
<code>hessian</code>	Whether to numerically compute the hessian matrix, see the documentation for optim .
<code>reltol</code>	Relative convergence tolerance, see the documentation for optim .

Value

A `logistic_copula` object, which contains the regression coefficients of the model, the parameters of the chosen conditional covariate distribution that corresponds to the regression coefficients, and the pair of vine-models that extend the logistic regression model.

Ionosphere	<i>Example data set</i>
------------	-------------------------

Description

This radar data was collected by a system in Goose Bay, Labrador. This system consists of a phased array of 16 high-frequency antennas with a total transmitted power on the order of 6.4 kilowatts. See Sigillito, V. G., Wing, S. P., Hutton, L. V., & Baker, K. B. (1989) for more details. The targets were free electrons in the ionosphere. "Good" radar returns are those showing evidence of some type of structure in the ionosphere. "Bad" returns are those that do not; their signals pass through the ionosphere.

Usage

```
data(Ionosphere)
```

Format

List containing the following elements:

x 351 by 34 matrix of numeric values.

Class Character vector of length 351 containing 126 entries labeled "bad" and 225 labeled "good".

```
predict.logistic_copula  
  predict.logistic_copula
```

Description

Computes predicted probability of $Y=1$ for a logistic regression model with a vine extension.

Usage

```
## S3 method for class 'logistic_copula'  
predict(object, new_x, ...)
```

Arguments

<code>object</code>	The model object as returned by <code>fit_copula_interactions</code>
<code>new_x</code>	A matrix of covariate values to compute predictions for.
<code>...</code>	Not used.

Value

A numeric vector of estimates of the conditional probability of $Y=1 \mid x$, computed for each row of `new_x`.

Index

* **data**

Ionosphere, [5](#)

bicop, [3](#)

fit_copula_interactions, [2](#)

fit_model, [4](#)

Ionosphere, [5](#)

optim, [3](#), [5](#)

predict.logistic_copula, [6](#)