Package 'palasso'

September 27, 2024

Version 1.0.0

Title Sparse Regression with Paired Covariates

Description Implements sparse regression with paired covariates (<doi:10.1007/s11634-019-00375-6>). The paired lasso is designed for settings where each covariate in one set forms a pair with a covariate in the other set (one-to-one correspondence). For the optional correlation shrinkage, install ashr (<https://github.com/stephens999/ashr>) and CorShrink (<https://github.com/kkdey/CorShrink>) from GitHub (see README).

Depends R (>= 3.0.0)

Imports glmnet, Matrix, survival

Suggests knitr, testthat, rmarkdown, remotes, pROC, edgeR, ashr, CorShrink

License GPL-3

Encoding UTF-8

VignetteBuilder knitr

RoxygenNote 7.3.2

URL https://github.com/rauschenberger/palasso, https://rauschenberger.github.io/palasso/

BugReports https://github.com/rauschenberger/palasso/issues

NeedsCompilation no

Author Armin Rauschenberger [aut, cre] (<https://orcid.org/0000-0001-6498-4801>)

Maintainer Armin Rauschenberger <armin.rauschenberger@uni.lu>

Repository CRAN

Date/Publication 2024-09-26 22:40:02 UTC

Contents

methods *Methods for class "palasso"*

Description

This page lists the main methods for class "palasso".

Usage

```
## S3 method for class 'palasso'
predict(object, newdata, model = "paired", s = "lambda.min", max = NULL, ...)

## S3 method for class 'palasso'
coef(object, model = "paired", s = "lambda.min", max = NULL, ...)

## S3 method for class 'palasso'
weights(object, model = "paired", max = NULL, ...)

## S3 method for class 'palasso'
fitted(object, model = "paired", s = "lambda.min", max = NULL, ...)

## S3 method for class 'palasso'
residuals(object, model = "paired", s = "lambda.min", max = NULL, ...)

## S3 method for class 'palasso'
deviance(object, model = "paired", max = NULL, ...)

## S3 method for class 'palasso'
logLik(object, model = "paired", max = NULL, ...)

## S3 method for class 'palasso'
summary(object, model = "paired", ...)
```

Arguments

object	palasso object
newdata	covariates**:** list of matrices, each with n rows (samples) and p columns (variables)
model	character "paired", or an entry of names(object)
s	penalty parameter**:** character "lambda.min" or "lambda.1se", positive numeric, or NULL (entire sequence)
max	maximum number of non-zero coefficients, positive integer, or NULL
...	further arguments for predict.cv.glmnet, coef.cv.glmnet, or deviance.glmnet

Details

By default, the function `predict` returns the linear predictor (`type="link"`). Consider predicting the response (`type="response"`).

See Also

Use palasso to fit the paired lasso.

`palasso` *Paired lasso*

Description

The function `palasso` fits the paired lasso. Use this function if you have *paired covariates* and want a *sparse model*.

Usage

```
palasso(y = y, X = X, max = 10, ...)
```

Arguments

`y`	response**:** vector of length n
`X`	covariates**:** list of matrices, each with n rows (samples) and p columns (variables)
`max`	maximum number of non-zero coefficients**:** positive numeric, or `NULL` (no sparsity constraint)
`...`	further arguments for `cv.glmnet` or `glmnet`

Details

Let `x` denote one entry of the list `X`. See glmnet for alternative specifications of `y` and `x`. Among the further arguments, `family` must equal `"gaussian"`, `"binomial"`, `"poisson"`, or `"cox"`, and `penalty.factor` must not be used.

Hidden arguments: Deactivate adaptive lasso by setting `adaptive` to `FALSE`, activate standard lasso by setting `standard` to `TRUE`, and activate shrinkage by setting `shrink` to `TRUE`.

Value

This function returns an object of class `palasso`. Available methods include `predict`, `coef`, `weights`, `fitted`, `residuals`, `deviance`, `logLik`, and `summary`.

References

Armin Rauschenberger, Iiuliana Ciocanea-Teodorescu, Marianne A. Jonker, Renee X. Menezes, and Mark A. van de Wiel (2020). "Sparse classification with paired covariates." *Advances in Data Analysis and Classification* 14:571-588. doi:10.1007/s11634019003756. (Click here to access PDF. Contact: <armin.rauschenberger@uni.lu>.)

Examples

```
set.seed(1)
n <- 50; p <- 20
y <- rbinom(n=n,size=1,prob=0.5)
X <- lapply(1:2,function(x) matrix(rnorm(n*p),nrow=n,ncol=p))
object <- palasso(y=y,X=X,family="binomial") # adaptive=TRUE,standard=FALSE
names(object)
```

Index