Package: care (via r-universe)

October 23, 2024

Version 1.1.11

Date 2021-11-21

Title High-Dimensional Regression and CAR Score Variable Selection

Author Verena Zuber and Korbinian Strimmer.

Maintainer Korbinian Strimmer <strimmerlab@gmail.com>

Depends R (>= 3.0.2), corpcor (>= 1.6.10)

Imports stats

Suggests crossval

Enhances sda

Description Implements the regression approach of Zuber and Strimmer (2011) ``High-dimensional regression and variable selection using CAR scores" SAGMB 10: 34, <DOI:10.2202/1544-6115.1730>. CAR scores measure the correlation between the response and the Mahalanobis-decorrelated predictors. The squared CAR score is a natural measure of variable importance and provides a canonical ordering of variables. This package provides functions for estimating CAR scores, for variable selection using CAR scores, and for estimating corresponding regression coefficients. Both shrinkage as well as empirical estimators are available.

License GPL (>= 3)

URL https://strimmerlab.github.io/software/care/

NeedsCompilation no

Repository CRAN

Date/Publication 2021-11-21 16:40:06 UTC

Contents

`care-package` *The care Package*

Description

The "care" package implements the CAR regression approach described in Zuber and Strimmer (2011). CAR scores measure the correlations between the response and the Mahalanobis-decorrelated predictors. The squared CAR score is a natural measure of variable importance and provides a canonical ordering of variables - see Zuber and Strimmer (2011) for details.

This package provides functions for estimating CAR scores, for variable selection using CAR scores, and for estimating corresponding regression coefficients. Both shrinkage as well as empirical estimators are available.

The name of the package refers to **CAR e**stimation and **CAR re**gression.

Author(s)

Verena Zuber and Korbinian Strimmer (https://strimmerlab.github.io/)

References

Zuber, V., and K. Strimmer. 2011. High-dimensional regression and variable selection using CAR scores. Statist. Appl. Genet. Mol. Biol. 10: 34. <DOI:10.2202/1544-6115.1730>

Website: https://strimmerlab.github.io/software/care/

See Also

`carscore`, `slm`, `efron2004`, `lu2004`.

`carscore` *Estimate CAR Scores and Marginal Correlations*

Description

`carscore` estimates the vector of CAR scores, either using the standard empirical estimator of the correlation matrix, or a shrinkage estimator.

Usage

```
carscore(Xtrain, Ytrain, lambda, diagonal=FALSE, verbose=TRUE)
```

Arguments

`Xtrain`	Matrix of predictors (columns correspond to variables).
`Ytrain`	Univariate response variable.
`lambda`	The correlation shrinkage intensity (range 0-1). If not specified (the default) it is estimated using an analytic formula from Sch\"afer and Strimmer (2005). For `lambda=0` the empirical correlations are used.
`diagonal`	For `diagonal=FALSE` (the default) CAR scores are computed; otherwise with `diagonal=TRUE` marginal correlations.
`verbose`	If `verbose=TRUE` then the shrinkage intensity used in estimating the shrinkage correlation matrix is reported.

Details

The CAR scores are the correlations between the response and the Mahalanobis-decorrelated predictors. CAR score is an abbreviation for Correlation-Adjusted (marginal) coRelation, where the first correlation matrix refers dependencies among predictors.

In Zuber and Strimmer (2011) it is argued that squared CAR scores are a natural measure for variable importance and it is shown that variable selection based on CAR scores is highly efficient compared to competing approaches such as elastic net lasso, or boosting.

If the response is binary (or descrete) the corresponding quantity are CAT scores (see `catscore`).

Value

`carscore` returns a vector containing the CAR scores (or marginal correlations for `diagonal=TRUE`).

Author(s)

Verena Zuber and Korbinian Strimmer (https://strimmerlab.github.io).

References

Zuber, V., and K. Strimmer. 2011. High-dimensional regression and variable selection using CAR scores. Statist. Appl. Genet. Mol. Biol. 10: 34. <DOI:10.2202/1544-6115.1730>

See Also

`catscore`.

Examples

```
# load care library
library("care")

######

# empirical CAR scores for diabetes data
data(efron2004)
xnames = colnames(efron2004$x)
```

```
n = dim(efron2004$x)[1]

car = carscore(efron2004$x, efron2004$y, lambda=0)
car

# compare orderings

# variables ordered by squared CAR scores
xnames[order(car^2, decreasing=TRUE)]
# "bmi" "s5"  "bp"  "s3"  "s4"  "s6"  "sex" "age" "s2"  "s1"

# compare with ordering by t-scores / partial correlations
pcor = pcor.shrink(cbind(efron2004$y,efron2004$x), lambda=0, verbose=FALSE)[-1,1]
xnames[order(pcor^2, decreasing=TRUE)]
# "bmi" "bp"  "s5"  "sex" "s1"  "s2"  "s4"  "s6"  "s3"  "age"

# compare with ordering by marginal correlations
mcor = cor(efron2004$y,efron2004$x)
#mcor = carscore(efron2004$x, efron2004$y, diagonal=TRUE, lambda=0)
xnames[order(mcor^2, decreasing=TRUE)]
# "bmi" "s5"  "bp"  "s4"  "s3"  "s6"  "s1"  "age" "s2"  "sex"

# decomposition of R^2
sum(car^2)
slm(efron2004$x, efron2004$y, lambda=0, lambda.var=0)$R2

# pvalues for empirical CAR scores
pval = 1-pbeta(car^2, shape1=1/2, shape2=(n-2)/2)
pval <= 0.05

######

# shrinkage CAR scores for Lu et al. (2004) data
data(lu2004)
dim(lu2004$x)    # 30 403

# compute shrinkage car scores
car = carscore(lu2004$x, lu2004$y)

# most important genes
order(car^2, decreasing=TRUE)[1:10]

# compare with empirical marginal correlations
mcor = cor(lu2004$y, lu2004$x)
order(mcor^2, decreasing=TRUE)[1:10]

# decomposition of R^2
sum(car^2)
slm(lu2004$x, lu2004$y)$R2
```

efron2004 *Diabetes Data from Efron et al. (2004)*

Description

Diabetes data (10 variables, 442 measurements) as used in the study of Efron et al. (2004). The data is standardized such that the means of all variables are zero, and all variances are equal to one.

Usage

```
data(efron2004)
```

Format

`efron2004$x` is a 422 x 10 matrix containing the measurements of the explanatory variables (age, sex, body mass, etc.). The rows contain the samples and the columns the variables.

`efron2004$y` contains the response.

Source

The orginal data are available in the lars R package, see https://cran.r-project.org/package=lars. Note that this uses a slightly different standardization.

References

Efron, B., et al. 2004. Least angle regression (with discussion). Ann. Statist. 32:407–499. <DOI:10.1214/009053604000000067>

Examples

```
# load care library
library("care")

# load Efron et al. (2004) diabetes data set
data(efron2004)
dim(efron2004$x) # 442 10
colnames(efron2004$x)
length(efron2004$y) # 442
```

lu2004 *Brain Aging Study of Lu et al. (2004)*

Description

Gene expression data (403 genes for 30 samples) from the microarray study of Lu et al. (2004).

Usage

```
data(lu2004)
```

Format

`lu2004$x` is a 30 x 403 matrix containing the log expression levels. The rows contain the samples and the columns the genes.

`lu2004$y` is the age of for each sample.

Details

This data set contains measurements of the gene expression of 403 genes from 30 human brain samples. In addition, the age of each patient is provided.

Source

The original data are available from the GEO public functional genomics database at URL https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE1572 and are described in Lu et al. (2004). The selected 403 genes result from prescreening and preprocessing as described in Zuber and Strimmer (2011).

References

Lu, T., et al. 2004. Gene regulation and DNA damage in the ageing human brain. Nature 429:883–891. <DOI:10.1038/nature02661>

Zuber, V., and K. Strimmer. 2011. High-dimensional regression and variable selection using CAR scores. Statist. Appl. Genet. Mol. Biol. 10: 34. <DOI:10.2202/1544-6115.1730>

Examples

```
# load care library
library("care")

# load Lu et al. (2004) data set
data(lu2004)
dim(lu2004$x) # 30 403
hist(lu2004$x)
length(lu2004$y) # 30
lu2004$y # age
```

slm *Shrinkage Estimation of Regression Coefficients*

Description

slm fits a linear model and computes (standardized) regression coefficients by plugin of shrinkage estimates of correlations and variances. Using the argument predlist several models can be fitted on the same data set.

make.predlist constructs a predlist argument for use with slm.

Usage

```
slm(Xtrain, Ytrain, predlist, lambda, lambda.var, diagonal=FALSE, verbose=TRUE)
## S3 method for class 'slm'
predict(object, Xtest, verbose=TRUE, ...)
make.predlist(ordering, numpred, name="SIZE")
```

Arguments

Xtrain	Matrix of predictors (columns correspond to variables).
Ytrain	Univariate continous response variable.
predlist	A list specifying the predictors to be included when fitting the linear regression. Each entry in the list is a vector containing the indices of variables used per model. If left unspecified single full-sized model using all variables in Xtrain is assumed. For a given ordering of covariables a suitable predlist can be generated using the helper function make.predlist - see examples below.
lambda	The correlation shrinkage intensity (range 0-1). If not specified (the default) it is estimated using an analytic formula from Sch\"afer and Strimmer (2005). For lambda=0 the empirical correlations are used.
lambda.var	The variance shrinkage intensity (range 0-1). If not specified (the default) it is estimated using an analytic formula from Opgen-Rhein and Strimmer (2007). For lambda.var=0 the empirical variances are used.
diagonal	If diagonal=FALSE (the default) then the correlation among predictor veriables assumed to be non-zero and is estimated from data. If diagonal=TRUE then it is assumed that the correlation among predictors vanishes and is set to zero.
verbose	If verbose=TRUE then the estimated shrinkage intensities are reported.
object	An slm fit object obtained from the function slm.
Xtest	A matrix containing the test data set. Note that the rows correspond to observations and the columns to variables.
...	Additional arguments for generic predict.
ordering	The ordering of the predictors (most important predictors are first).
numpred	The number of included predictors (may be a scalar or a vector). The predictors are included in the order specified by ordering.
name	The name assigned to each model is name plus "." and the number of included predictors.

Details

The regression coefficients are obtained by estimating the joint joint covariance matrix of the response and the predictors, and subsequently computing the the regression coefficients by inversion of this matrix - see Opgen-Rhein and Strimmer (2007). As estimators for the covariance matrix either the standard empirical estimator or a Stein-type shrinkage estimator is employed. The use of the empirical covariance leads to the OLS estimates of the regression coefficients, whereas otherwise shrinkage estimates are obtained.

Value

`slm` returns a list with the following components:

`regularization`: The shrinkage intensities used for estimating correlations and variances.

`std.coefficients`: The standardized regression coefficients, i.e. the regression coefficients computed from centered and standardized input data. Thus, by construction the intercept is zero. Furthermore, for `diagonal=TRUE` the standardized regression coefficient for each predictor is identical to the respective marginal correlation.

`coefficients`: Regression coefficients.

`numpred`: The number of predictors used in each investigated model.

`R2`: For `diagonal=TRUE` this is the multiple correlation coefficient between the response and the predictor, or the proportion of explained variance, with range from 0 to 1. For `diagonal=TRUE` this equals the sum of squared marginal correlations. Note that this sum may be larger than 1!

`sd.resid`: The residual unexplained error.

`predict.slm` returns the means predicted for each sample and model as well as the corresponding predictive standard deviations (attached as attribute "sd").

Author(s)

Korbinian Strimmer (https://strimmerlab.github.io).

References

Opgen-Rhein, R., and K. Strimmer. 2007. From correlation to causation networks: a simple approximate learning algorithm and its application to high-dimensional plant gene expression data. BMC Syst. Biol. 1: 37. <DOI:10.1186/1752-0509-1-37>

Sch\"afer, J., and K. Strimmer. 2005. A shrinkage approach to large-scale covariance estimation and implications for functional genomics. Statist. Appl. Genet. Mol. Biol. 4: 32. <DOI:10.2202/1544-6115.1175>

See Also

carscore

Examples

```
# load care library
library("care")

## example with large number of samples and small dimension
## (using empirical estimates of regression coefficients)

# diabetes data
data(efron2004)
x = efron2004$x
y = efron2004$y
n = dim(x)[1]
d = dim(x)[2]
xnames = colnames(x)

# empirical regression coefficients
fit = slm(x, y, lambda=0, lambda.var=0)
fit
# note that in this example the regression coefficients
# and the standardized regression coefficients are identical
# as the input data have been standardized to mean zero and variance one

# compute corresponding t scores / partial correlations
df = n-d-1
pcor = pcor.shrink(cbind(y,x), lambda=0)[-1,1]
t = pcor * sqrt(df/(1-pcor^2))
t.pval = 2 - 2 * pt(abs(t), df)
b = fit$coefficients[1,-1]
cbind(b, pcor, t, t.pval)

# compare results with those from lm function
lm.out = lm(y ~ x)
summary(lm.out)

# prediction of fitted values at the position of the training data
lm.out$fitted.values
mu.hat = predict(fit, x) # precticted means
mu.hat
attr(mu.hat, "sd") # predictive error
sd(y-mu.hat)


# ordering of the variables using squared empirical CAR score
car = carscore(x, y, lambda=0)
ocar = order(car^2, decreasing=TRUE)
xnames[ocar]

# CAR regression models with 5, 7, 9 included predictors
car.predlist = make.predlist(ocar, numpred = c(5,7,9), name="CAR")
car.predlist
slm(x, y, car.predlist, lambda=0, lambda.var=0)
```

```
# plot regression coefficients for all possible CAR models

p=ncol(x)
car.predlist = make.predlist(ocar, numpred = 1:p, name="CAR")
cm = slm(x, y, car.predlist, lambda=0, lambda.var=0)
bmat = cm$coefficients[,-1]
bmat

par(mfrow=c(2,1))

plot(1:p, bmat[,1], type="l",
  ylab="estimated regression coefficients",
  xlab="number of included predictors",
  main="CAR Regression Models for Diabetes Data",
  xlim=c(1,p+1), ylim=c(min(bmat), max(bmat)))

for (i in 2:p) lines(1:p, bmat[,i], col=i, lty=i)
for (i in 1:p) points(1:p, bmat[,i], col=i)
for (i in 1:p) text(p+0.5, bmat[p,i], xnames[i])

plot(1:p, cm$R2, type="l",
  ylab="estimated R2",
  xlab="number of included predictors",
  main="Proportion of Explained Variance",
  ylim=c(0,0.6))
R2max = max(cm$R2)
lines(c(1,p), c(R2max, R2max), col=2)

par(mfrow=c(1,1))


## example with small number of samples and large dimension
## (using shrinkage estimates of regression coefficients)

data(lu2004)
dim(lu2004$x)    # 30 403

fit = slm(lu2004$x, lu2004$y)
fit
```

Index