Package 'PCLasso'

January 30, 2021

Type Package

Title A protein complex-based group lasso-Cox model for accurate prognosis and risk protein complex discovery

Version 1.0

Date 2021-01-29

Depends R (>= 3.5.0), grpreg

Imports survival

Description The PCLasso model is a prognostic model which selects important predictors at the protein complex level to achieve accurate prognosis and identify risk protein complexes. The PCLasso model has three inputs: a gene expression matrix, survival data, and protein complexes. It estimates the correlation between gene expression in protein complexes and survival data at the level of protein complexes. Similar to the traditional Lasso-Cox model, PCLasso is based on the Cox PH model and estimates the Cox regression coefficients by maximizing partial likelihood with regularization penalty. The difference is that PCLasso selects features at the level of protein complexes rather than individual genes. Considering that genes usually function by forming protein complexes, PCLasso regards genes belonging to the same protein complex as a group, and constructs a l1/l2 penalty based on the sum (i.e., l1 norm) of the l2 norms of the regression coefficients of the group members to perform the selection of features at the group level. Since a gene may belong to multiple protein complexes, that is, there is overlap between protein complexes, the classical group Lasso-Cox model for non-overlapping groups may lead to false sparse solutions. The PCLasso model deals with the overlapping problem of protein complexes by constructing a latent group Lasso-Cox model. And by reconstructing the gene expression matrix of the protein complexes, the latent group Lasso-Cox model is transformed into a non-overlapping group Lasso-Cox model in an expanded space, which can be directly solved using the classical group Lasso method. Through the final sparse solution, we can predict the patient's risk score based on a small set of protein complexes and identify risk protein complexes that are frequently selected to construct prognostic models.

License Artistic-2.0

NeedsCompilation no

Author Wei Liu [cre, aut] (<https://orcid.org/0000-0002-5496-3641>)

Maintainer Wei Liu <freelw@qq.com>

R topics documented:

`PCLasso-package` *A protein complex-based group lasso-Cox model for accurate prognosis and risk protein complex discovery*

Description

The PCLasso model is a prognostic model which selects important predictors at the protein complex level to achieve accurate prognosis and identify risk protein complexes. The PCLasso model has three inputs: a gene expression matrix, survival data, and protein complexes. It estimates the correlation between gene expression in protein complexes and survival data at the level of protein complexes. Similar to the traditional Lasso-Cox model, PCLasso is based on the Cox PH model and estimates the Cox regression coefficients by maximizing partial likelihood with regularization penalty. The difference is that PCLasso selects features at the level of protein complexes rather than individual genes. Considering that genes usually function by forming protein complexes, PCLasso regards genes belonging to the same protein complex as a group, and constructs a l1/l2 penalty based on the sum (i.e., l1 norm) of the l2 norms of the regression coefficients of the group members to perform the selection of features at the group level. Since a gene may belong to multiple protein complexes, that is, there is overlap between protein complexes, the classical group Lasso-Cox model for non-overlapping groups may lead to false sparse solutions. The PCLasso model deals with the overlapping problem of protein complexes by constructing a latent group Lasso-Cox model. And by reconstructing the gene expression matrix of the protein complexes, the latent group Lasso-Cox model is transformed into a non-overlapping group Lasso-Cox model in an expanded space, which can be directly solved using the classical group Lasso method. Through the final sparse solution, we can predict the patient's risk score based on a small set of protein complexes and identify risk protein complexes that are frequently selected to construct prognostic models.

Details

Index of help topics:

```
ExpMatrix               The expression data
PCGroup                 Protein complexes for "PCLasso"/"cv.PCLasso"
PCLasso                 Protein complex-based group lasso-Cox model
PCLasso-package         A protein complex-based group lasso-Cox model
                        for accurate prognosis and risk protein complex
                        discovery
cv.PCLasso              Cross-validation for 'PCLasso'
ext2EntrezID            Transform selection features into EntrezID
```

ext2Group	Convert selected features into ComplexID
plot.PCLasso	Plot coefficients from a PCLasso object
plot.cv.PCLasso	Plot the cross-validation curve from a 'cv.PCLasso' object
predict.PCLasso	Make predictions from a PCLasso model
predict.cv.PCLasso	Make predictions from a cross-validated PCLasso model
survData	Survival data

The PCLasso model accepts a gene expression matrix, survival data, and protein complexes for the PCLasso model, and makes predictions for new samples and identifies risk protein complexes.

PCLasso constructs a PCLasso model based on a gene expression matrix, survival data, and protein complexes.

predict.PCLasso makes predictions from a PCLasso model.

cv.PCLasso performs k-fold cross validations for the PCLasso model with grouped covariates over a grid of values for the regularization parameter lambda, and returns an optimal value for lambda.

predict.cv.PCLasso returns predictions from a fitted cv.PCLasso object, using the optimal value chosen for lambda.

plot.PCLasso produces a plot of the coefficient paths for a fitted PCLasso object.

plot.cv.PCLasso plots the cross-validation curve from a cv.PCLasso object, along with standard error bars.

References

PCLasso: a protein complex-based group lasso-Cox model for accurate prognosis and risk protein complex discovery. To be published.

Park, H., Niida, A., Miyano, S. and Imoto, S. (2015) Sparse overlapping group lasso for integrative multi-omics analysis. Journal of computational biology: a journal of computational molecular cell biology, 22, 73-84.

---

cv.PCLasso *Cross-validation for* PCLasso

Description

Perform k-fold cross validations for the PCLasso model with grouped covariates over a grid of values for the regularization parameter lambda.

Usage

```
cv.PCLasso(x, y, group, penalty = c("grLasso", "grMCP", "grSCAD", "gel",
"cMCP"), nfolds = 5, standardize = TRUE, ...)
```

Arguments

x	A n x p design matrix of gene expression measurements with n samples and p genes, as in PCLasso.

y	The time-to-event outcome, as a two-column matrix or Surv object, as in PCLasso. The first column should be time on study (follow up time); the second column should be a binary variable with 1 indicating that the event has occurred and 0 indicating (right) censoring.
group	A list of groups as in PCLasso. The feature (gene) names in group should be consistent with the feature (gene) names in x.
penalty	The penalty to be applied to the model. For group selection, one of grLasso, grMCP, or grSCAD. For bi-level selection, one of gel or cMCP. See grpsurv in the R package grpreg for details.
nfolds	The number of cross-validation folds. Default is 5.
standardize	Logical flag for x standardization, prior to fitting the model. Default is TRUE.
...	Arguments to be passed to cv.grpsurv in the R package grpreg.

Details

The function calls PCLasso nfolds times, each time leaving out 1/nfolds of the data. The cross-validation error is based on the deviance. The numbers for each outcome class are balanced across the folds; i.e., the number of outcomes in which y is equal to 1 is the same for each fold, or possibly off by 1 if the numbers do not divide evenly. cv.PCLasso uses the approach of calculating the full Cox partial likelihood using the cross-validated set of linear predictors. See cv.grpsurv in the R package grpreg for details.

Value

An object with S3 class "cv.PCLasso" containing:

cv.fit	An object of class "cv.grpsurv"
group.dt	Groups with features (genes) not included in x being filtered out.

Author(s)

Wei Liu

References

PCLasso: a protein complex-based group lasso-Cox model for accurate prognosis and risk protein complex discovery. To be published.

Park, H., Niida, A., Miyano, S. and Imoto, S. (2015) Sparse overlapping group lasso for integrative multi-omics analysis. Journal of computational biology: a journal of computational molecular cell biology, 22, 73-84.

See Also

predict.cv.PCLasso

Examples

```
library("survival")

# load data
data(ExpMatrix)
data(survData)
```

```
data(PCGroup)

x = ExpMatrix
y = Surv(time=survData[,"time"], event=survData[,"status"])

# fit model
cv.fit1 <- cv.PCLasso(x, y, group = PCGroup, penalty = "grLasso", nfolds = 10)
```

`ExpMatrix` *The expression data*

Description

An example of gene expression data.

Usage

```
data("ExpMatrix")
```

Format

The format is:

num [1:200, 1:2150] 3.402 0.791 6.592 3.13 5.794 ...

- attr(*, "dimnames")=List of 2

..$: chr [1:200] "S1" "S2" "S3" "S4" ...

..$: chr [1:2150] "8813" "2729" "1080" "5893" ...

Examples

```
data(ExpMatrix)
```

`ext2EntrezID` *Transform selection features into EntrezID*

Description

Transform selection features into EntrezID at the gene level.

Usage

```
ext2EntrezID(x)
```

Arguments

`x`	Names of selected vars by `PCLasso` or `cv.PCLasso`.

Value

Selected vars.

Author(s)

Wei Liu

`ext2Group` *Convert selected features into ComplexID*

Description

Convert selected features into ComplexID at protein complex level

Usage

```
ext2Group(x)
```

Arguments

`x`	Names of selected vars by `PCLasso` or `cv.PCLasso`.

Value

Selected protein complexes.

Author(s)

Wei Liu

`PCGroup` *Protein complexes for "PCLasso"/"cv.PCLasso"*

Description

A list of protein complexes. The genes in each protein complex are represented by EntrezID, which are consistent with the gene names in `ExpMatrix`.

Usage

```
data("PCGroup")
```

Format

The format is:

List of 2417

$ C_1 : chr [1:2] "604" "9759"

$ C_2 : chr [1:2] "604" "10014"

$ C_3 : chr [1:2] "604" "51564"

$ C_4 : chr [1:4] "2033" "1387" "8850" "8202"

$ C_11 : chr [1:2] "3257" "89781"

$ C_12 : chr [1:3] "79803" "84343" "11234"

[list output truncated]

Examples

```
data(PCGroup)
```

PCLasso *Protein complex-based group lasso-Cox model*

Description

Construct a PCLasso model based on a gene expression matrix, survival data, and protein complexes.

Usage

```
PCLasso(x, y, group, penalty = c("grLasso", "grMCP", "grSCAD", "gel", "cMCP"),
standardize = TRUE, ...)
```

Arguments

x	A n x p matrix of gene expression measurements with n samples and p genes.
y	The time-to-event outcome, as a two-column matrix or Surv object. The first column should be time on study (follow up time); the second column should be a binary variable with 1 indicating that the event has occurred and 0 indicating (right) censoring.
group	A list of groups. The feature (gene) names in group should be consistent with the feature (gene) names in x.
penalty	The penalty to be applied to the model. For group selection, one of grLasso, grMCP, or grSCAD. For bi-level selection, one of gel or cMCP. See grpsurv in the R package grpreg for details.
standardize	Logical flag for x standardization, prior to fitting the model. Default is TRUE.
...	Arguments to be passed to grpsurv in the R package grpreg.

Details

The PCLasso model is a prognostic model which selects important predictors at the protein complex level to achieve accurate prognosis and identify risk protein complexes. The PCLasso model has three inputs: a gene expression matrix, survival data, and protein complexes. It estimates the correlation between gene expression in protein complexes and survival data at the level of protein complexes. Similar to the traditional Lasso-Cox model, PCLasso is based on the Cox PH model and estimates the Cox regression coefficients by maximizing partial likelihood with regularization penalty. The difference is that PCLasso selects features at the level of protein complexes rather than individual genes. Considering that genes usually function by forming protein complexes, PCLasso regards genes belonging to the same protein complex as a group, and constructs a l1/l2 penalty based on the sum (i.e., l1 norm) of the l2 norms of the regression coefficients of the group members to perform the selection of features at the group level. Since a gene may belong to multiple protein complexes, that is, there is overlap between protein complexes, the classical group Lasso-Cox model for non-overlapping groups may lead to false sparse solutions. The PCLasso model deals with the overlapping problem of protein complexes by constructing a latent group Lasso-Cox model. And by reconstructing the gene expression matrix of the protein complexes, the latent group Lasso-Cox model is transformed into a non-overlapping group Lasso-Cox model in an expanded space, which

can be directly solved using the classical group Lasso method. Through the final sparse solution, we can predict the patient's risk score based on a small set of protein complexes and identify risk protein complexes that are frequently selected to construct prognostic models.

Value

An object with S3 class "PCLasso" containing:

`fit`	An object of class "grpsurv"
`group.dt`	Groups with features (genes) not included in `x` being filtered out.

Author(s)

Wei Liu

References

PCLasso: a protein complex-based group lasso-Cox model for accurate prognosis and risk protein complex discovery. To be published.

Park, H., Niida, A., Miyano, S. and Imoto, S. (2015) Sparse overlapping group lasso for integrative multi-omics analysis. Journal of computational biology: a journal of computational molecular cell biology, 22, 73-84.

See Also

`predict.PCLasso`, `cv.PCLasso`

Examples

```
library("survival")

# load data
data(ExpMatrix)
data(survData)
data(PCGroup)

x <- ExpMatrix
y <- Surv(time=survData[,"time"], event=survData[,"status"])

# fit the PCLasso model
fit1 <- PCLasso(x, y, group = PCGroup, penalty = "grLasso")
```

---

plot.cv.PCLasso *Plot the cross-validation curve from a* `cv.PCLasso` *object*

Description

Plot the cross-validation curve from a `cv.PCLasso` object, along with standard error bars.

Usage

```
## S3 method for class 'cv.PCLasso'
plot(x, type = c("cve", "rsq", "snr", "all"),
norm = NULL, ...)
```

Arguments

x	Fitted cv.PCLasso model.
type	What to plot on the vertical axis. "cve" plots the cross-validation error (deviance); "rsq" plots an estimate of the fraction of the deviance explained by the model (R-squared); "snr" plots an estimate of the signal-to-noise ratio; "all" produces all of the above.
norm	If TRUE, plot the norm of each group, rather than the individual coefficients.
...	Other graphical parameters to plot

Details

Error bars representing approximate +/- 1 SE (68% confidence intervals) are plotted along with the estimates at value of lambda. See plot.cv.grpreg in the R package grpreg for details.

Author(s)

Wei Liu

See Also

cv.PCLasso

Examples

```
library("survival")

# load data
data(ExpMatrix)
data(survData)
data(PCGroup)

x <- ExpMatrix
y <- Surv(time=survData[,"time"], event=survData[,"status"])

cv.fit1 <- cv.PCLasso(x, y, group = PCGroup, penalty = "grLasso",nfolds = 10)

# plot the norm of each group
plot(cv.fit1, norm = TRUE)

# plot the individual coefficients
plot(cv.fit1, norm = FALSE)

# plot the cross-validation error (deviance)
plot(cv.fit1, type = "cve")
```

plot.PCLasso *Plot coefficients from a PCLasso object*

Description

Produces a plot of the coefficient paths for a fitted PCLasso object.

Usage

```
## S3 method for class 'PCLasso'
plot(x, norm = TRUE, ...)
```

Arguments

x	Fitted PCLasso model.
norm	If TRUE, plot the norm of each group, rather than the individual coefficients.
...	Other graphical parameters to plot.

Author(s)

Wei Liu

See Also

PCLasso

Examples

```
library("survival")

# load data
data(ExpMatrix)
data(survData)
data(PCGroup)

x <- ExpMatrix
y <- Surv(time=survData[,"time"], event=survData[,"status"])

# fit the PCLasso model
fit1 <- PCLasso(x, y, group = PCGroup, penalty = "grLasso")

# plot the norm of each group
plot(fit1, norm = TRUE)

# plot the individual coefficients
plot(fit1, norm = FALSE)
```

predict.cv.PCLasso *Make predictions from a cross-validated PCLasso model*

Description

Similar to other predict methods, this function returns predictions from a fitted "cv.PCLasso" object, using the optimal value chosen for lambda.

Usage

```
## S3 method for class 'cv.PCLasso'
predict(object, x = NULL, type = c("link", "response",
"survival", "median", "norm", "coefficients", "vars", "nvars", "vars.unique",
"nvars.unique", "groups", "ngroups"), lambda, ...)
```

Arguments

object	Fitted cv.PCLasso model object.
x	Matrix of values at which predictions are to be made. The features (genes) contained in x should be consistent with those contained in x in the PCLasso function. Not used for type="coefficients" or for some of the type settings in predict.
type	Type of prediction: "link" returns the linear predictors; "response" gives the risk (i.e., exp(link)); "vars" returns the indices for the nonzero coefficients; "vars.unique" returns unique features (genes) with nonzero coefficients (If a feature belongs to multiple groups and multiple groups are selected, the feature will be repeatedly selected. Compared with "var", "var.unique" will filter out repeated features.); "groups" returns the groups with at least one nonzero coefficient; "nvars" returns the number of nonzero coefficients; "nvars.unique" returens the number of unique features (genes) with nonzero coefficients; "ngroups" returns the number of groups with at least one nonzero coefficient; "norm" returns the L2 norm of the coefficients in each group."survival" returns the estimated survival function; "median" estimates median survival times.
lambda	Values of the regularization parameter lambda at which predictions are requested. For values of lambda not in the sequence of fitted models, linear interpolation is used.
...	Arguments to be passed to predict.cv.grpsurv in the R package grpreg.

Value

The object returned depends on type.

See Also

cv.PCLasso

Examples

```
library("survival")

# load data
data(ExpMatrix)
data(survData)
data(PCGroup)

set.seed(429006)
train.Idx <- sample(nrow(ExpMatrix), floor(2/3*nrow(ExpMatrix)))
x.train <- ExpMatrix[train.Idx ,]
x.test <- ExpMatrix[-train.Idx ,]
y.train <- survData[train.Idx,]
y.test <- survData[-train.Idx,]

cv.fit1 <- cv.PCLasso(x = x.train,
                      y = Surv(time=y.train[,"time"], event=y.train[,"status"]),
                      group = PCGroup,
                      nfolds = 5)

# predict risk scores of samples in x.test
s <- predict(object = cv.fit1, x = x.test, type="link",
             lambda=cv.fit1$cv.fit$lambda.min)

# Nonzero coefficients
sel.groups <- predict(object = cv.fit1, type="groups",
                      lambda = cv.fit1$cv.fit$lambda.min)
sel.ngroups <- predict(object = cv.fit1, type="ngroups",
                       lambda = cv.fit1$cv.fit$lambda.min)
sel.vars.unique <- predict(object = cv.fit1, type="vars.unique",
                      lambda = cv.fit1$cv.fit$lambda.min)
sel.nvars.unique <- predict(object = cv.fit1, type="nvars.unique",
                      lambda = cv.fit1$cv.fit$lambda.min)
sel.vars <- predict(object = cv.fit1, type="vars",
                    lambda=cv.fit1$cv.fit$lambda.min)
sel.nvars <- predict(object = cv.fit1, type="nvars",
                      lambda=cv.fit1$cv.fit$lambda.min)
```

`predict.PCLasso` *Make predictions from a PCLasso model*

Description

Similar to other predict methods, this function returns predictions from a fitted `PCLasso` object.

Usage

```
## S3 method for class 'PCLasso'
predict(object, x = NULL, type = c("link", "response", "survival",
"median", "norm", "coefficients", "vars", "nvars", "vars.unique",
"nvars.unique", "groups", "ngroups"), lambda, ...)
```

Arguments

object	Fitted PCLasso model object.
x	Matrix of values at which predictions are to be made. The features (genes) contained in x should be consistent with those contained in x in the PCLasso function. Not used for type="coefficients" or for some of the type settings in predict.
type	Type of prediction: "link" returns the linear predictors; "response" gives the risk (i.e., exp(link)); "vars" returns the indices for the nonzero coefficients; "vars.unique" returns unique features (genes) with nonzero coefficients (If a feature belongs to multiple groups and multiple groups are selected, the feature will be repeatedly selected. Compared with "var", "var.unique" will filter out repeated features.); "groups" returns the groups with at least one nonzero coefficient; "nvars" returns the number of nonzero coefficients; "nvars.unique" returens the number of unique features (genes) with nonzero coefficients; "ngroups" returns the number of groups with at least one nonzero coefficient; "norm" returns the L2 norm of the coefficients in each group."survival" returns the estimated survival function; "median" estimates median survival times.
lambda	Values of the regularization parameter lambda at which predictions are requested. For values of lambda not in the sequence of fitted models, linear interpolation is used.
...	Arguments to be passed to predict.grpsurv in the R package grpreg.

Details

See predict.grpsurv in the R package grpreg for details.

Value

The object returned depends on type.

Author(s)

Wei Liu

See Also

PCLasso

Examples

```
library("survival")

# load data
data(ExpMatrix)
data(survData)
data(PCGroup)

set.seed(429006)
train.Idx <- sample(nrow(ExpMatrix), floor(2/3*nrow(ExpMatrix)))
x.train <- ExpMatrix[train.Idx ,]
x.test <- ExpMatrix[-train.Idx ,]
y.train <- survData[train.Idx,]
y.test <- survData[-train.Idx,]
```

```
fit1 <- PCLasso(x = x.train,
                y = Surv(time=y.train[,"time"], event=y.train[,"status"]),
                group = PCGroup)

# predict risk scores of samples in x.test
s <- predict(object = fit1, x = x.test, type="link", lambda=fit1$fit$lambda)

s <- predict(object = fit1, x = x.test, type="link", lambda=fit1$fit$lambda[10])

s <- predict(object = fit1, x = x.test, type="link", lambda=c(0.1, 0.01))

# Nonzero coefficients
sel.groups <- predict(object = fit1, type="groups",
                      lambda = fit1$fit$lambda)
sel.ngroups <- predict(object = fit1, type="ngroups",
                       lambda = fit1$fit$lambda)
sel.vars.unique <- predict(object = fit1, type="vars.unique",
                           lambda = fit1$fit$lambda)
sel.nvars.unique <- predict(object = fit1, type="nvars.unique",
                            lambda = fit1$fit$lambda)
sel.vars <- predict(object = fit1, type="vars",
                    lambda=fit1$fit$lambda)
sel.nvars <- predict(object = fit1, type="nvars",
                     lambda=fit1$fit$lambda)


# For values of lambda not in the sequence of fitted models,
# linear interpolation is used.
sel.groups <- predict(object = fit1, type="groups",
                      lambda = c(0.1, 0.01))
sel.ngroups <- predict(object = fit1, type="ngroups",
                       lambda = c(0.1, 0.01))
sel.vars.unique <- predict(object = fit1, type="vars.unique",
                           lambda = c(0.1, 0.01))
sel.nvars.unique <- predict(object = fit1, type="nvars.unique",
                            lambda = c(0.1, 0.01))
sel.vars <- predict(object = fit1, type="vars",
                    lambda=c(0.1, 0.01))
sel.nvars <- predict(object = fit1, type="nvars",
                     lambda=c(0.1, 0.01))
```

`survData` *Survival data*

Description

The survival data of patients in `ExpMatrix`.

Usage

```
data("survData")
```

Format

A data frame with 200 observations on the following 2 variables.

`status` a numeric vector

`time` a numeric vector

Examples

```
data(survData)
```

Index