Package 'HierFabs'

April 9, 2021

Type Package

Title A General Framework for Identifying Hierachical Interaction of Genomics Data interaction of genomics data

Version 0.1.0

Author Xiao Zhang [aut,cre],
Xu Liu [aut],
Yiming Liu [aut],
Xingjie Shi [aut]

Maintainer Xiao Zhang <zhangxiao_0422@163.com>

Description Provides an efficient procedure for fitting the entire solution path adaptively for identifying hierachical interaction of genomics data in linear model or cox model under the strong or weak heredity constraint. Lasso penalty is used to generate a sparse solution. Both gene-gene type interactions and gene-environment interactions are supported.

License GPL (>= 2)

Imports Matrix, methods

Repository github

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

NeedsCompilation yes

R topics documented:

| HierFabs | *A hierarchical Forward and Backward Stagewise (HierFabs) algorithm for identifying hierachical interaction of genomics data.* |

Description

A hierarchical Forward and Backward Stagewise (HierFabs) algorithm for identifying hierachical interaction of genomics data.

Usage

```
HierFabs(
  G,
  y,
  E,
  weight = NULL,
  model = c("gaussian", "cox"),
  back = TRUE,
  stoping = TRUE,
  eps = 0.01,
  xi = 10^-6,
  iter = 3000,
  lambda.min = NULL,
  hier = c("strong", "weak"),
  max_s = NULL,
  diagonal = FALSE,
  status = NULL,
  gamma = NULL
)
```

Arguments

G	Gene matrix, each row is an observation vector.
y	Response variable.
E	An optional environment matrix. If Z is given, the interactions between environment and gene are of interest. Otherwise, the gene-gene interactions are of interest.
weight	An optional weights. Default is 1 for each observation.
model	A character string representing one of the built-in models. 'gaussian' for linear model and 'cox' for cox model.
back	An indicator of whether to take backward steps.
stoping	An indicator of whether to stop iteration when lambda is less than lambda.min.
eps	Step size. Default is 0.01.
xi	A tolerate to improve program stability. Default is 10^{-6}.
iter	Maximum number of outer-loop iterations allowed. Default is 3000.
lambda.min	Smallest value for lambda, as a fraction of lambda.max.
hier	Whether to enforce strong or weak heredity. Default is 'strong'.

max_s	Limit the maximum number of variables in the model. When exceed this limit, program will early stopped.
diagonal	An indicator of whether to include "pure" quadratic terms. Work when gene-gene interactions are of interest.
status	A censoring indicator.
gamma	A tuning parameter in EBIC.

Value

A list.

- theta - The coefficients of covariates, each column is a solution.
- beta - The optimal solution.
- lambda - Lambda sequence.
- direction - Update indicator. 1 means take a forward step, -1 means take a backward step.
- iter - Number of iterations.
- EBIC - EBIC for each solution.
- loss - loss for each solution.
- df - Number of nonzero coefficients.
- opt - Position of the optimal tuning based on EBIC.
- intercept - The intercept term, which appearance is due to standardization.

See Also

predict.HierFabs, print.HierFabs

Examples

```
set.seed(0)
n = 500
p = 100
x = matrix(rnorm(n*p),n,p)
eta = x[,1:4] %*% rep(1,4) + 3*x[,1]*x[,2] + 3*x[,1]*x[,4]
y =  eta + rnorm(n)
xtest = matrix(rnorm(n*p),n,p)
eta.test = xtest[,1:4] %*% rep(1,4) + 3*xtest[,1]*xtest[,2] + 3*xtest[,1]*xtest[,4]
ytest =  eta.test + rnorm(n)
fit.gg.strong = HierFabs(x, y)
y.pred.gg.s = predict(fit.gg.strong, xtest, ytest)
y.pred.gg.s$mse
print(fit.gg.strong)

## Weak hierarchy
fit.gg.weak = HierFabs(x, y, hier="weak")
y.pred.gg.w = predict(fit.gg.weak, xtest, ytest)
y.pred.gg.w$mse
print(fit.gg.weak)

## Cox model with Gene-Environment interactions
pz = 10
z = matrix(rnorm(n*pz),n,pz)
eta.ge = x[,1:4] %*% rep(1,4) + z[,1] + z[,2] + 3*x[,1]*z[,1] + 3*x[,2]*z[,2]
```

```
err = log(rweibull(n, shape = 1, scale = 1))
y0 = exp(-eta.ge + err)
cens = quantile(y0, 0.9)
y.ge = pmin(y0, cens)
status = 1 * (y0<=cens)
ztest = matrix(rnorm(n*pz),n,pz)
eta.ge.test = rowSums(xtest[,1:4]) + ztest[,1] + ztest[,2]
eta.ge.test = eta.ge.test + 3*xtest[,1]*ztest[,1] + 3*xtest[,2]*ztest[,2]
err.test = log(rweibull(n, shape = 1, scale = 1))
y0.test = exp(-eta.ge.test + err.test)
cens = quantile(y0.test, 0.9)
y.ge.test = pmin(y0.test, cens)
status.test = 1 * (y0.test<=cens)
fit.ge.strong = HierFabs(x, y.ge, z, model="cox", status=status)
y.pred.ge.s = predict(fit.ge.strong, xtest, y.ge.test, ztest, status.test)
y.pred.ge.s$c.index
print(fit.ge.strong)

## Weak hierarchy
fit.ge.weak = HierFabs(x, y.ge, z, model="cox", status=status, hier="weak")
y.pred.ge.w = predict(fit.ge.weak, xtest, y.ge.test, ztest, status.test)
y.pred.ge.w$c.index
print(fit.ge.weak)
```

---

predict.HierFabs   *Model prediction based on a fitted HierFabs object.*

Description

Similar to the usual predict methods, this function returns predictions from a fitted `'HierFabs'` object.

Usage

```
## S3 method for class 'HierFabs'
predict(object, new.G, new.y, new.E, new.s, allpath = FALSE, ...)
```

Arguments

object	Fitted `'HierFabs'` model object.
new.G	Matrix of new values for genes.
new.y	Vector of new values for response.
new.E	An optional new values for environment variables.
new.s	An optional new values for status variables. Used in cox model prediction only.
allpath	allpath = T will output all the predictions on the solution path. allpath = FALSE will only output the optimal solution selected in the 'HierFabs' object.
...	Not used. Other arguments to predict.

Value

A list.

- pred - The linear predict.
- mse - Mean square error for linear model.
- c.index - C index for cox model.

See Also

HierFabs

---

print.HierFabs	*Result summary of a fitted HierFabs x.*

Description

Similar to the usual print methods, this function summarize results from a fitted `'HierFabs'` object.

Usage

```
print.HierFabs(x, digits = max(getOption("digits") - 2, 3), ...)
```

Arguments

x	Fitted `'HierFabs'` model object..
digits	The number of significant digits for the coefficient estimates.
...	Not used. Other arguments to print.

Value

No value is returned.

See Also

HierFabs

Index