Package 'GEInter'

January 20, 2025

Type Package

Title Robust Gene-Environment Interaction Analysis

Version 0.3.2

Maintainer Xing Qin <qin.xing@163.sufe.edu.cn>

Description Description: For the risk, progression, and response to treatment of many complex diseases, it has been increasingly recognized that gene-environment interactions play important roles beyond the main genetic and environmental effects. In practical interaction analyses, outliers in response variables and covariates are not uncommon. In addition, missingness in environmental factors is routinely encountered in epidemiological studies. The developed package consists of five robust approaches to address the outliers problems, among which two approaches can also accommodate missingness in environmental factors. Both continuous and right censored responses are considered. The proposed approaches are based on penalization and sparse boosting techniques for identifying important interactions, which are realized using efficient algorithms. Beyond the gene-environment analysis, the developed package can also be adopted to conduct analysis on interactions between other types of low-dimensional and high-dimensional data. (Mengyun Wu et al (2017), <doi:10.1080/00949655.2018.1523411>; Mengyun Wu et al (2017), <doi:10.1

License GPL-2

Encoding UTF-8

LazyData true

Imports MASS, splines, pcaPP, Hmisc, survival, quantreg, reshape2, ggplot2, stats, graphics

RoxygenNote 7.1.2

Depends R (>= 3.5.0)

NeedsCompilation no

Author Mengyun Wu [aut], Xing Qin [aut, cre], Shuangge Ma [aut]

Repository CRAN

Date/Publication 2022-05-19 06:50:19 UTC

Contents

`AR` *The covariance matrix with an autoregressive (AR) structure among variables*

Description

The covariance matrix with an AR structure among variables, where the marginal variances are 1 and the `j`th and `k`th variables have correlation coefficient `rho^abs(j-k)`.

Usage

```
AR(rho, p)
```

Arguments

rho	The correlation coefficient indicating the AR relationship between the variables.
p	The dimension of variables.

Value

A covariance matrix.

Augmented.data *Accommodating missingness in environmental measurements in gene-environment interaction analysis*

Description

We consider the scenario with missingness in environmental (E) measurements. Our approach consists of two steps. We first develop a nonparametric kernel-based data augmentation approach to accommodate missingness. Then, we adopt a penalization approach BLMCP for regularized estimation and selection of important interactions and main genetic (G) effects, where the "main effects-interactions" hierarchical structure is respected. As E variables are usually preselected and have a low dimension, selection is not conducted on E variables. With a well-designed weighting scheme, a nice "byproduct" is that the proposed approach enjoys a certain robustness property.

Usage

```
Augmented.data(G, E, Y, h, family = c("continuous", "survival"), E_type)
```

Arguments

G	Input matrix of p genetic measurements consisting of n rows. Each row is an observation vector.
E	Input matrix of q environmental risk factors. Each row is an observation vector.
Y	Response variable. A quantitative vector for family="continuous". For family="survival", Y should be a two-column matrix with the first column being the log(survival time) and the second column being the censoring indicator. The indicator is a binary variable, with "1" indicating dead, and "0" indicating right censored.
h	The bandwidths of the kernel functions with the first and second elements corresponding to the discrete and continuous E factors.
family	Response type of Y (see above).
E_type	A vector indicating the type of each E factor, with "ED" representing discrete E factor, and "EC" representing continuous E factor.

Value

E_w	The augmented data corresponding to E.
G_w	The augmented data corresponding to G.
y_w	The augmented data corresponding to response y.
weight	The weights of the augmented observation data for accommodating missingness and also right censoring if family="survival".

References

Mengyun Wu, Yangguang Zang, Sanguo Zhang, Jian Huang, and Shuangge Ma. *Accommodating missingness in environmental measurements in gene-environment interaction analysis. Genetic Epidemiology, 41(6):523-554, 2017.*

Jin Liu, Jian Huang, Yawei Zhang, Qing Lan, Nathaniel Rothman, Tongzhang Zheng, and Shuangge Ma. *Identification of gene-environment interactions in cancer studies using penalization. Genomics, 102(4):189-194, 2013.*

Examples

```
set.seed(100)
sigmaG=AR(0.3,50)
G=MASS::mvrnorm(100,rep(0,50),sigmaG)
E=matrix(rnorm(100*5),100,5)
E[,2]=E[,2]>0
E[,3]=E[,3]>0
alpha=runif(5,2,3)
beta=matrix(0,5+1,50)
beta[1,1:7]=runif(7,2,3)
beta[2:4,1]=runif(3,2,3)
beta[2:3,2]=runif(2,2,3)
beta[5,3]=runif(1,2,3)

# continuous with Normal error N(0,4)
y1=simulated_data(G=G,E=E,alpha=alpha,beta=beta,error=rnorm(100,0,4),family="continuous")

# survival with Normal error N(0,1)
y2=simulated_data(G,E,alpha,beta,rnorm(100,0,1),family="survival",0.7,0.9)

# generate E measurements with missingness
miss_label1=c(2,6,8,15)
miss_label2=c(4,6,8,16)
E1=E2=E;E1[miss_label1,1]=NA;E2[miss_label2,1]=NA

# continuous
data_new1<-Augmented.data(G,E1,y1,h=c(0.5,1), family="continuous",
E_type=c("EC","ED","ED","EC","EC"))
fit1<-BLMCP(data_new1$G_w, data_new1$E_w, data_new1$y_w, data_new1$weight,
lambda1=0.025,lambda2=0.06,gamma1=3,gamma2=3,max_iter=200)
coef1=coef(fit1)
y1_hat=predict(fit1,E[c(1,2),],G[c(1,2),])
plot(fit1)
```

```
## survival
data_new2<-Augmented.data(G,E2,y2, h=c(0.5,1), family="survival",
E_type=c("EC","ED","ED","EC","EC"))
fit2<-BLMCP(data_new2$G_w, data_new2$E_w, data_new2$y_w, data_new2$weight,
lambda1=0.04,lambda2=0.05,gamma1=3,gamma2=3,max_iter=200)
coef2=coef(fit2)
y2_hat=predict(fit2,E[c(1,2),],G[c(1,2),])
plot(fit2)
```

`bic.BLMCP` *BIC for BLMCP*

Description

Selects a point along the regularization path of a fitted `BLMCP` object according to the BIC.

Usage

```
bic.BLMCP(
  G,
  E,
  Y,
  weight = NULL,
  lambda1_set = NULL,
  lambda2_set = NULL,
  nlambda1 = 20,
  nlambda2 = 20,
  gamma1 = 6,
  gamma2 = 6,
  max_iter = 200
)
```

Arguments

`G`	Input matrix of `p` genetic (G) measurements consisting of `n` rows. Each row is an observation vector.
`E`	Input matrix of `q` environmental (E) risk factors. Each row is an observation vector.
`Y`	Response variable. A quantitative vector for continuous response. For survival response, `Y` should be a two-column matrix with the first column being the log(survival time) and the second column being the censoring indicator. The indicator is a binary variable, with "1" indicating dead, and "0" indicating right censored.
`weight`	Observation weights.

lambda1_set	A user supplied lambda sequence for group minimax concave penalty (MCP), where each main G effect and its corresponding interactions are regarded as a group.
lambda2_set	A user supplied lambda sequence for MCP accommodating interaction selection.
nlambda1	The number of lambda1 values.
nlambda2	The number of lambda2 values.
gamma1	The regularization parameter of the group MCP penalty.
gamma2	The regularization parameter of the MCP penalty.
max_iter	Maximum number of iterations.

Value

An object with S3 class "bic.BLMCP" is returned, which is a list with the ingredients of the BIC fit.

call	The call that produced this object.
alpha	The matrix of the coefficients for main E effects, each column corresponds to one combination of (lambda1,lambda2).
beta	The coefficients for main G effects and G-E interactions, each column corresponds to one combination of (lambda1,lambda2). For each column, the first element is the first G effect and the second to (q+1) elements are the interactions for the first G factor, and so on.
df	The number of nonzeros for each value of (lambda1,lambda2).
BIC	Bayesian Information Criterion for each value of (lambda1,lambda2).
alpha_estimate	Final alpha estimate using Bayesian Information Criterion.
beta_estimate	Final beta estimate using Bayesian Information Criterion.
lambda_combine	The matrix of (lambda1, lambda2), with the first column being the values of lambda1, the second being the values of lambda2.

References

Mengyun Wu, Yangguang Zang, Sanguo Zhang, Jian Huang, and Shuangge Ma. *Accommodating missingness in environmental measurements in gene-environment interaction analysis. Genetic Epidemiology, 41(6):523-554, 2017.*
Jin Liu, Jian Huang, Yawei Zhang, Qing Lan, Nathaniel Rothman, Tongzhang Zheng, and Shuangge Ma. *Identification of gene-environment interactions in cancer studies using penalization. Genomics, 102(4):189-194, 2013.*

See Also

predict, coef and plot methods, and the BLMCP function.

Examples

```
set.seed(100)
sigmaG=AR(0.3,50)
G=MASS::mvrnorm(150,rep(0,50),sigmaG)
E=matrix(rnorm(150*5),150,5)
E[,2]=E[,2]>0;E[,3]=E[,3]>0
alpha=runif(5,2,3)
beta=matrix(0,5+1,50);beta[1,1:8]=runif(8,2,3)
beta[2:4,1]=runif(3,2,3)
beta[2:3,2]=runif(2,2,3)
beta[5,3]=runif(1,2,3)

# continuous with Normal error
y1=simulated_data(G=G,E=E,alpha=alpha,beta=beta,error=rnorm(150),family="continuous")

# survival with Normal error
y2=simulated_data(G,E,alpha,beta,rnorm(150,0,1),family="survival",0.8,1)

# continuous
fit1<-bic.BLMCP(G,E,y1,weight=NULL,lambda1_set=NULL,lambda2_set=NULL,
nlambda1=10,nlambda2=10,gamma1=6,gamma2=6,max_iter=200)
coef1=coef(fit1)
y1_hat=predict(fit1,E,G)
plot(fit1)


## survival
fit2<-bic.BLMCP(G,E,y2,weight=NULL,lambda1_set=NULL,lambda2_set=NULL,
nlambda1=20,nlambda2=20,gamma1=6,gamma2=6,max_iter=200)
coef2=coef(fit2)
y2_hat=predict(fit2,E,G)
plot(fit2)
```

`bic.PTReg` *BIC for PTReg*

Description

Selects a point along the regularization path of a fitted `PTReg` object according to the BIC.

Usage

```
bic.PTReg(
  G,
  E,
  Y,
  lambda1_set,
  lambda2_set,
```

```
  gamma1,
  gamma2,
  max_init,
  h = NULL,
  tau = 0.4,
  mu = 2.5,
  family = c("continuous", "survival")
)
```

Arguments

G	Input matrix of p genetic (G) measurements consisting of n rows. Each row is an observation vector.
E	Input matrix of q environmental (E) risk factors. Each row is an observation vector.
Y	Response variable. A quantitative vector for family="continuous". For family="survival", Y should be a two-column matrix with the first column being the log(survival time) and the second column being the censoring indicator. The indicator is a binary variable, with "1" indicating dead, and "0" indicating right censored.
lambda1_set	A user supplied lambda sequence for minimax concave penalty (MCP) accommodating main G effect selection.
lambda2_set	A user supplied lambda sequence for MCP accommodating interaction selection.
gamma1	The regularization parameter of the MCP penalty corresponding to G effects.
gamma2	The regularization parameter of the MCP penalty corresponding to G-E interactions.
max_init	The number of initializations.
h	The number of the trimmed samples if the parameter mu is not given.
tau	The threshold value used in stability selection.
mu	The parameter for screening outliers with extreme absolute residuals if the number of the trimmed samples h is not given.
family	Response type of Y (see above).

Value

An object with S3 class "bic.PTReg" is returned, which is a list with the ingredients of the BIC fit.

call	The call that produced this object.
alpha	The matrix of the coefficients for main E effects, each column corresponds to one combination of (lambda1,lambda2).
beta	The coefficients for main G effects and G-E interactions, each column corresponds to one combination of (lambda1,lambda2). For each column, the first element is the first G effect and the second to (q+1) elements are the interactions for the first G factor, and so on.

intercept	Matrix of the intercept estimate, each column corresponds to one combination of (lambda1,lambda2).
df	The number of nonzeros for each value of (lambda1,lambda2).
BIC	Bayesian Information Criterion for each value of (lambda1,lambda2).
family	The same as input family.
intercept_estimate	Final intercept estimate using Bayesian Information Criterion.
alpha_estimate	Final alpha estimate using Bayesian Information Criterion.
beta_estimate	Final beta estimate using Bayesian Information Criterion.
lambda_combine	Matrix of (lambda1, lambda2), with the first column being the values of lambda1, the second being the values of lambda2.

References

Yaqing Xu, Mengyun Wu, Shuangge Ma, and Syed Ejaz Ahmed. *Robust gene-environment interaction analysis using penalized trimmed regression. Journal of Statistical Computation and Simulation, 88(18):3502-3528, 2018.*

Examples

```
sigmaG<-AR(rho=0.3,p=30)
sigmaE<-AR(rho=0.3,p=3)
set.seed(300)
G=MASS::mvrnorm(150,rep(0,30),sigmaG)
EC=MASS::mvrnorm(150,rep(0,2),sigmaE[1:2,1:2])
ED = matrix(rbinom((150),1,0.6),150,1)
E=cbind(EC,ED)
alpha=runif(3,0.8,1.5)
beta=matrix(0,4,30)
beta[1,1:4]=runif(4,1,1.5)
beta[2,c(1,2)]=runif(2,1,1.5)
lambda1_set=lambda2_set=c(0.2,0.25,0.3,0.35,0.4,0.5)


#continuous response with outliers/contaminations in response variable
y1=simulated_data(G,E,alpha,beta,error=c(rnorm(140),rcauchy(10,0,5)),family="continuous")
fit1<-bic.PTReg(G,E,y1,lambda1_set,lambda2_set,gamma1=6,gamma2=6,
max_init=50,tau=0.6,mu=2.5,family="continuous")
coefficients1=coefficients(fit1)
y_predict=predict(fit1,E,G)
plot(fit1)

# survival with Normal error
y2=simulated_data(G,E,alpha,beta,rnorm(150,0,1),family="survival",0.7,0.9)
fit2<-bic.PTReg(G,E,y2,lambda1_set,lambda2_set,gamma1=6,gamma2=6,
max_init=50,tau=0.6,mu=2.5,family="survival")
coefficients2=coefficients(fit2)
y_predict=predict(fit2,E,G)
plot(fit2)
```

BLMCP *Accommodating missingness in environmental measurements in gene-environment interaction analysis: penalized estimation and selection*

Description

The joint gene-environment (G-E) interaction analysis approach developed in Liu et al, 2013. To accommodate "main effects, interactions" hierarchy, two types of penalty, group minimax concave penalty (MCP) and MCP are adopted. Specifically, for each G factor, its main effect and corresponding G-E interactions are regarded as a group, where the group MCP is imposed to identify whether this G factor has any effect at all. In addition, the MCP is imposed on the interaction terms to further identify important interactions.

Usage

```
BLMCP(
  G,
  E,
  Y,
  weight = NULL,
  lambda1,
  lambda2,
  gamma1 = 6,
  gamma2 = 6,
  max_iter = 200
)
```

Arguments

G	Input matrix of p G measurements consisting of n rows. Each row is an observation vector.
E	Input matrix of q environmental risk factors. Each row is an observation vector.
Y	Response variable. A quantitative vector for continuous response. For survival response, Y should be a two-column matrix with the first column being the log(survival time) and the second column being the censoring indicator. The indicator is a binary variable, with "1" indicating dead, and "0" indicating right censored.
weight	Observation weights.
lambda1	A user supplied lambda for group MCP, where each main G effect and its corresponding interactions are regarded as a group.
lambda2	A user supplied lambda for MCP accommodating interaction selection.
gamma1	The regularization parameter of the group MCP penalty.
gamma2	The regularization parameter of the MCP penalty.
max_iter	Maximum number of iterations.

Value

An object with S3 class "BLMCP" is returned, which is a list with the following components.

call	The call that produced this object.
alpha	The matrix of the coefficients for main E effects.
beta	The matrix of the regression coefficients for all main G effects (the first row) and interactions.
df	The number of nonzeros.
BIC	Bayesian Information Criterion.
aa	The indicator representing whether the algorithm reaches convergence.

References

Mengyun Wu, Yangguang Zang, Sanguo Zhang, Jian Huang, and Shuangge Ma. *Accommodating missingness in environmental measurements in gene-environment interaction analysis. Genetic Epidemiology, 41(6):523-554, 2017.*

Jin Liu, Jian Huang, Yawei Zhang, Qing Lan, Nathaniel Rothman, Tongzhang Zheng, and Shuangge Ma. *Identification of gene-environment interactions in cancer studies using penalization. Genomics, 102(4):189-194, 2013.*

See Also

predict, and coef, and plot, and bic.BLMCP and Augmentated.data methods.

Examples

```
set.seed(100)
sigmaG=AR(0.3,100)
G=MASS::mvrnorm(250,rep(0,100),sigmaG)
E=matrix(rnorm(250*5),250,5)
E[,2]=E[,2]>0;E[,3]=E[,3]>0
alpha=runif(5,2,3)
beta=matrix(0,5+1,100);beta[1,1:8]=runif(8,2,3)
beta[2:4,1]=runif(3,2,3);beta[2:3,2]=runif(2,2,3);beta[5,3]=runif(1,2,3)

# continuous with Normal error
y1=simulated_data(G,E,alpha,beta,error=rnorm(250),family="continuous")
fit1<-BLMCP(G,E,y1,weight=NULL,lambda1=0.05,lambda2=0.06,gamma1=3,gamma2=3,max_iter=200)
coef1=coef(fit1)
y1_hat=predict(fit1,E,G)
plot(fit1)

# survival with Normal error
y2=simulated_data(G,E,alpha,beta,rnorm(250,0,1),family="survival",0.7,0.9)
fit2<-BLMCP(G,E,y2,weight=NULL,lambda1=0.05,lambda2=0.06,gamma1=3,gamma2=3,max_iter=200)
coef2=coef(fit2)
y2_hat=predict(fit2,E,G)
plot(fit2)
```

`coef.bic.BLMCP` *Extract coefficients from a "bic.BLMCP" object*

Description

This function extracts the coefficients of main effects and interactions from a BIC BLMCP model, using the stored "`bic.BLMCP`" object.

Usage

```
## S3 method for class 'bic.BLMCP'
coef(object, ...)
```

Arguments

`object`	Fitted "`bic.BLMCP`" model object.
`...`	Not used. Other arguments to get coefficients.

Value

The object returned depends on the ... argument which is passed on to the `coef` method for `bic.BLMCP` objects.

`alpha`	The matrix of the coefficients for main environmental effects.
`beta`	The matrix of the regression coefficients for all main genetic effects (the first row) and interactions.

References

Mengyun Wu, Yangguang Zang, Sanguo Zhang, Jian Huang, and Shuangge Ma. *Accommodating missingness in environmental measurements in gene-environment interaction analysis. Genetic Epidemiology, 41(6):523-554, 2017.*
Jin Liu, Jian Huang, Yawei Zhang, Qing Lan, Nathaniel Rothman, Tongzhang Zheng, and Shuangge Ma. *Identification of gene-environment interactions in cancer studies using penalization. Genomics, 102(4):189-194, 2013.*

See Also

`bic.BLMCP`, and `predict`, and `plot` methods, and the `BLMCP` function.

`coef.bic.PTReg` *Extract coefficients from a "bic.PTReg" object*

Description

This function extracts the coefficients of main effects and interactions from a BIC PTReg model, using the stored `"bic.PTReg"` object.

Usage

```
## S3 method for class 'bic.PTReg'
coef(object, ...)
```

Arguments

`object`	Fitted "bic.PTReg" model object.
`...`	Not used. Other arguments to get coefficients.

Value

The object returned depends on the ... argument which is passed on to the `coef` method for `bic.PTReg` objects.

`intercept`	The intercept estimate.
`alpha`	The matrix of the coefficients for main environmental effects.
`beta`	The matrix of the regression coefficients for all main genetic effects (the first row) and interactions.

References

Yaqing Xu, Mengyun Wu, Shuangge Ma, and Syed Ejaz Ahmed. *Robust gene-environment interaction analysis using penalized trimmed regression. Journal of Statistical Computation and Simulation, 88(18):3502-3528, 2018.*

See Also

`bic.PTReg`, and `predict`, and `plot` methods, and `PTReg`.

`coef.BLMCP` *Extract coefficients from a "BLMCP" object*

Description

This function extracts the coefficients of main effects and interactions from a BLMCP model, using the stored "BLMCP" object.

Usage

```
## S3 method for class 'BLMCP'
coef(object, ...)
```

Arguments

`object`	Fitted "BLMCP" model object.
`...`	Not used. Other arguments to get coefficients.

Value

The object returned depends on the ... argument which is passed on to the `coef` method for `BLMCP` objects.

`alpha`	The matrix of the coefficients for main environmental effects.
`beta`	The matrix of the regression coefficients for all main genetic effects (the first row) and interactions.

References

Mengyun Wu, Yangguang Zang, Sanguo Zhang, Jian Huang, and Shuangge Ma. *Accommodating missingness in environmental measurements in gene-environment interaction analysis. Genetic Epidemiology, 41(6):523-554, 2017.*
Jin Liu, Jian Huang, Yawei Zhang, Qing Lan, Nathaniel Rothman, Tongzhang Zheng, and Shuangge Ma. *Identification of gene-environment interactions in cancer studies using penalization. Genomics, 102(4):189-194, 2013.*

See Also

`BLMCP`, and `predict`, `plot` methods, and `bic.BLMCP`.

`coef.PTReg` *Extract coefficients from a "PTReg" object*

Description

This function extracts main effect and interaction coefficients from a PTReg model, using the stored `"PTReg"` object.

Usage

```
## S3 method for class 'PTReg'
coef(object, ...)
```

Arguments

`object`	Fitted `"PTReg"` model object.
`...`	Not used. Other arguments to get coefficients.

Value

The object returned depends on the ... argument which is passed on to the `coef` method for `PTReg` objects.

`intercept`	The intercept estimate.
`alpha`	The matrix of the coefficients for main environmental effects.
`beta`	The matrix of the regression coefficients for all main genetic effects (the first row) and interactions.

References

Yaqing Xu, Mengyun Wu, Shuangge Ma, and Syed Ejaz Ahmed. *Robust gene-environment interaction analysis using penalized trimmed regression. Journal of Statistical Computation and Simulation, 88(18):3502-3528, 2018.*

See Also

`PTReg`, and `predict` methods, and `bic.PTReg`.

`coef.RobSBoosting` *Extract coefficients from a "RobSBoosting" object*

Description

This function extracts coefficients from a RobSBoosting model, using the stored `"RobSBoosting"` object.

Usage

```
## S3 method for class 'RobSBoosting'
coef(object, ...)
```

Arguments

`object`	Fitted `"RobSBoosting"` model object.
`...`	Not used. Other arguments to get coefficients.

Value

`intercept`	The intercept estimate.
`unique_variable`	A matrix with two columns that represents the variables that are selected for the model after removing the duplicates, since the `loop_time` iterations of the method may produce variables that are repeatedly selected into the model. Here, the first and second columns correspond to the indexes of environmental (E) factors and genetic (G) factors. For example, (1, 0) represents that this variable is the first E factor, and (1,2) represents that the variable is the interaction between the first E factor and second G factor.
`unique_coef`	Coefficients corresponding to `unique_variable`. Here, the coefficients are simple regression coefficients for the linear effect (discrete E factor, G factor, and their interaction), and B spline coefficients for the nonlinear effect (continuous E factor, and corresponding G-E interaction).
`unique_knots`	A list of knots corresponding to `unique_variable`. Here, when the type of `unique_variable` is discrete E factor, G factor, or their interaction, knot will be NULL, and knots will be B spline otherwise.
`unique_Boundary.knots`	A list of boundary knots corresponding to `unique_variable`.
`unique_vtype`	A vector representing the variable type of `unique_variable`. Here, "EC" stands for continuous E effect, "ED" for discrete E effect, "G" for G effect, "EC-G" for the interaction between "EC" and "G", and "ED-G" for the interaction between "ED" and "G".
`estimation_results`	A list of estimation results for each variable. Here, the first `q` elemnets are for the E effects, the `(q+1)` element is for the first G effect and the `(q+2)` to `(2q+1)` elements are for the interactions corresponding to the first G factor, and so on.

References

Mengyun Wu and Shuangge Ma. *Robust semiparametric gene-environment interaction analysis using sparse boosting. Statistics in Medicine, 38(23):4625-4641, 2019.*

See Also

`RobSBoosting`, and `predict`, and `plot` methods.

`HNSCC` *A data frame containing the TCGA head and neck squamous cell carcinoma (HNSCC) data.*

Description

A data frame containing the 7 environmental (E) effects (the first 7 columns), 2000 genetic (G) effects (column 8 to column 2007), logarithm of survival time (column 2008), and censoring indicator (column 2009). All of them can be downloaded from TCGA Provisional using the R package `cgdsr`. See details.

Usage

```
data(HNSCC)
```

Format

A data frame with 484 rows and 2009 variables.

Details

There are seven E effects, namely alcohol consumption frequency (ACF), smoking pack years (SPY), age, gender, PN, PT, and ICD O3 site. For G effects, 2,000 gene expressions are considered. Among 484 subjects, 343 subjects have missingness in ACF and/or SPY. For G effects, we analyze mRNA gene expressions. A total of 18,409 gene expression measurements are available, then prescreening is conducted using marginal Cox models, finally, the top 2,000 genes with the smallest p-values are selected for downstream analysis.

Examples

```
data(HNSCC)
E=as.matrix(HNSCC[,1:7])
G=as.matrix(HNSCC[,8:2007])
Y=as.matrix(HNSCC[,2008:2009])

fit<-Miss.boosting(G,E,Y,im_time=10,loop_time=1000,v=0.25,num.knots=5,degree=3,tau=0.3,
family="survival",E_type=c(rep("EC",3),rep("ED",4)))
plot(fit)
```

`Miss.boosting` *Robust gene-environment interaction analysis approach via sparse boosting, where the missingness in environmental measurements is effectively accommodated using multiple imputation approach*

Description

This gene-environment analysis approach includes three steps to accommodate both missingness in environmental (E) measurements and long-tailed or contaminated outcomes. At the first step, the multiple imputation approach based on sparse boosting method is developed to accommodate missingness in E measurements, where we use `NA` to represent those E measurments which are missing. Here a semiparametric model is assumed to accommodate nonlinear effects, where we model continuous E factors in a nonlinear way, and discrete E factors in a linear way. For estimating the nonlinear functions, the B spline expansion is adopted. At the second step, for each imputed data, we develop `RobSBoosting` approach for identifying important main E and genetic (G) effects, and G-E interactions, where the Huber loss function and Qn estimator are adopted to accommodate long-tailed distribution/data contamination (see `RobSBoosting`). At the third step, the identification results from Step 2 are combined based on stability selection technique.

Usage

```
Miss.boosting(
  G,
  E,
  Y,
  im_time = 10,
  loop_time = 500,
  num.knots = c(2),
  Boundary.knots,
  degree = c(2),
  v = 0.1,
  tau,
  family = c("continuous", "survival"),
  knots = NULL,
  E_type
)
```

Arguments

`G`	Input matrix of `p` genetic measurements consisting of `n` rows. Each row is an observation vector.
`E`	Input matrix of `q` environmental risk factors. Each row is an observation vector.
`Y`	Response variable. A quantitative vector for `family="continuous"`. For `family="survival"`, `Y` should be a two-column matrix with the first column being the log(survival time) and the second column being the censoring indicator. The indicator is a binary variable, with "1" indicating dead, and "0" indicating right censored.

im_time	Number of imputation for accommodating missingness in E variables.
loop_time	Number of iterations of the sparse boosting.
num.knots	Numbers of knots for the B spline basis.
Boundary.knots	The boundary of knots for the B spline basis.
degree	Degree for the B spline basis.
v	The step size used in the sparse boosting process. Default is 0.1.
tau	Threshold used in the stability selection at the third step.
family	Response type of Y (see above).
knots	List of knots for the B spline basis. Default is NULL and knots can be generated with the given num.knots, degree and Boundary.knots.
E_type	A vector indicating the type of each E factor, with "ED" representing discrete E factor, and "EC" representing continuous E factor.

Value

An object with S3 class "Miss.boosting" is returned, which is a list with the following components

call	The call that produced this object.
alpha0	A vector with each element indicating whether the corresponding E factor is selected.
beta0	A vector with each element indicating whether the corresponding G factor or G-E interaction is selected. The first element is the first G effect and the second to (q+1) elements are the interactions for the first G factor, and so on.
intercept	The intercept estimate.
unique_variable	A matrix with two columns that represents the variables that are selected for the model after removing the duplicates, since the loop_time iterations of the method may produce variables that are repeatedly selected into the model. Here, the first and second columns correspond to the indexes of E factors and G factors. For example, (1, 0) represents that this variable is the first E factor, and (1,2) represents that the variable is the interaction between the first E factor and second G factor.
unique_coef	Coefficients corresponding to unique_variable. Here, the coefficients are simple regression coefficients for the linear effect (discrete E factor, G factor, and their interaction), and B spline coefficients for the nonlinear effect (continuous E factor, and corresponding G-E interaction).
unique_knots	A list of knots corresponding to unique_variable. Here, when the type of unique_variable is discrete E factor, G factor or their interaction, knot will be NULL, and knots will be B spline otherwise.
unique_Boundary.knots	A list of boundary knots corresponding to unique_variable.

unique_vtype	A vector representing the variable type of unique_variable. Here, "EC" stands for continuous E effect, "ED" for discrete E effect, "G" for genetic factor variable, "EC-G" for the interaction between "EC" and "G", and "ED-G" for the interaction between "ED" and "G".
degree	Degree for the B spline basis.
NorM	The values of B spline basis.
E_type	The type of E effects.

References

Mengyun Wu and Shuangge Ma. *Robust semiparametric gene-environment interaction analysis using sparse boosting. Statistics in Medicine, 38(23):4625-4641, 2019.*

Examples

```
data(Rob_data)
G=Rob_data[,1:20];E=Rob_data[,21:24]
Y=Rob_data[,25];Y_s=Rob_data[,26:27]
knots=list();Boundary.knots=matrix(0,(20+4),2)
for (i in 1:4){
  knots[[i]]=c(0,1)
  Boundary.knots[i,]=c(0,1)
}
E2=E1=E

##continuous
E1[7,1]=NA
fit1<-Miss.boosting(G,E1,Y,im_time=1,loop_time=100,num.knots=c(2),Boundary.knots,
degree=c(2),v=0.1,tau=0.3,family="continuous",knots=knots,E_type=c("EC","EC","ED","ED"))
y1_hat=predict(fit1,matrix(E1[1,],nrow=1),matrix(G[1,],nrow=1))
plot(fit1)


##survival
E2[4,1]=NA
fit2<-Miss.boosting(G,E2,Y_s,im_time=2,loop_time=200,num.knots=c(2),Boundary.knots,
degree=c(2),v=0.1,tau=0.3,family="survival",knots,E_type=c("EC","EC","ED","ED"))
y2_hat=predict(fit2,matrix(E1[1,],nrow=1),matrix(G[1,],nrow=1))
plot(fit2)
```

---

plot.bic.BLMCP *Plot coefficients from a "bic.BLMCP" object*

Description

Draw a heatmap for estimated coefficients in a fitted "bic.BLMCP" object.

Usage

```
## S3 method for class 'bic.BLMCP'
plot(x, ...)
```

Arguments

x	Fitted "bic.BLMCP" model.
...	Other graphical parameters to plot.

Value

A heatmap for estimated coefficients.

References

Mengyun Wu, Yangguang Zang, Sanguo Zhang, Jian Huang, and Shuangge Ma. *Accommodating missingness in environmental measurements in gene-environment interaction analysis. Genetic Epidemiology, 41(6):523-554, 2017.*
Jin Liu, Jian Huang, Yawei Zhang, Qing Lan, Nathaniel Rothman, Tongzhang Zheng, and Shuangge Ma. *Identification of gene-environment interactions in cancer studies using penalization. Genomics, 102(4):189-194, 2013.*

See Also

predict, coef and BLMCP methods.

---

plot.bic.PTReg *Plot coefficients from a "bic.PTReg" object*

Description

Draw a heatmap for estimated coefficients in a fitted "bic.PTReg" object.

Usage

```
## S3 method for class 'bic.PTReg'
plot(x, ...)
```

Arguments

x	Fitted "bic.PTReg" model.
...	Other graphical parameters to plot.

Value

A heatmap for estimated coefficients.

References

Yaqing Xu, Mengyun Wu, Shuangge Ma, and Syed Ejaz Ahmed. *Robust gene-environment interaction analysis using penalized trimmed regression. Journal of Statistical Computation and Simulation, 88(18):3502-3528, 2018.*

See Also

`bic.PTReg`, and `predict`, and `coef` methods.

`plot.BLMCP` *Plot coefficients from a "BLMCP" object*

Description

Draw a heatmap for estimated coefficients in a fitted `"BLMCP"` object.

Usage

```
## S3 method for class 'BLMCP'
plot(x, ...)
```

Arguments

`x`	Fitted `"BLMCP"` model.
`...`	Other graphical parameters to plot.

Value

A heatmap for estimated coefficients.

References

Mengyun Wu, Yangguang Zang, Sanguo Zhang, Jian Huang, and Shuangge Ma. *Accommodating missingness in environmental measurements in gene-environment interaction analysis. Genetic Epidemiology, 41(6):523-554, 2017.*

Jin Liu, Jian Huang, Yawei Zhang, Qing Lan, Nathaniel Rothman, Tongzhang Zheng, and Shuangge Ma. *Identification of gene-environment interactions in cancer studies using penalization. Genomics, 102(4):189-194, 2013.*

See Also

`BLMCP`, and `predict` and `coef` methods.

plot.Miss.boosting *Plot coefficients from a "Miss.boosting" object*

Description

Draw plots for estimated parameters in a fitted "Miss.boosting" object, including a heatmap for discrete environmental (E) effects, and selected genetic (G) effects and G-E interactions, and plots for each of selected continuous E (EC) effect and interactions between EC and G.

Usage

```
## S3 method for class 'Miss.boosting'
plot(x, ...)
```

Arguments

x	Fitted "Miss.boosting" model.
...	Other graphical parameters to plot.

Value

A heatmap for estimated coefficients.

References

Mengyun Wu and Shuangge Ma. *Robust semiparametric gene-environment interaction analysis using sparse boosting. Statistics in Medicine, 38(23):4625-4641, 2019.*

See Also

Miss.boosting, and predict methods.

plot.PTReg *Plot coefficients from a "PTReg" object*

Description

Draw a heatmap for estimated coefficients in a fitted "PTReg" object.

Usage

```
## S3 method for class 'PTReg'
plot(x, ...)
```

Arguments

x	Fitted "PTReg" model.
...	Other graphical parameters to plot.

Value

A heatmap for estimated coefficients.

References

Yaqing Xu, Mengyun Wu, Shuangge Ma, and Syed Ejaz Ahmed. *Robust gene-environment interaction analysis using penalized trimmed regression. Journal of Statistical Computation and Simulation, 88(18):3502-3528, 2018.*

See Also

PTReg, and predict, and coef methods.

---

plot.RobSBoosting *Plot coefficients from a "RobSBoosting" object*

Description

Draw plots for estimated parameters in a fitted "RobSBoosting" object, including a heatmap for discrete environmental (E) effects, and selected genetic (G) effects and G-E interactions, and plots for each of selected continuous E (EC) effect and interactions between EC and G.

Usage

```
## S3 method for class 'RobSBoosting'
plot(x, ...)
```

Arguments

x	Fitted "RobSBoosting" model.
...	Other graphical parameters to plot.

Value

Plots for estimated coefficients.

References

Mengyun Wu and Shuangge Ma. *Robust semiparametric gene-environment interaction analysis using sparse boosting. Statistics in Medicine, 38(23):4625-4641, 2019.*

See Also

RobSBoosting, predict and coef methods.

`predict.bic.BLMCP` *Make predictions from a "bic.BLMCP" object.*

Description

This function makes predictions from a BIC BLMCP model, using the stored `"bic.BLMCP"` object. This function makes it easier to use the results of BIC to make a prediction.

Usage

```
## S3 method for class 'bic.BLMCP'
predict(object, newE, newG, ...)
```

Arguments

`object`	Fitted `"bic.BLMCP"` object.
`newE`	Matrix of new values for `E` at which predictions are to be made.
`newG`	Matrix of new values for `G` at which predictions are to be made.
`...`	Not used. Other arguments to predict.

Value

The object returned depends on the ... argument which is passed on to the `predict` method for `BLMCP` objects.

References

Mengyun Wu, Yangguang Zang, Sanguo Zhang, Jian Huang, and Shuangge Ma. *Accommodating missingness in environmental measurements in gene-environment interaction analysis. Genetic Epidemiology, 41(6):523-554, 2017.*

Jin Liu, Jian Huang, Yawei Zhang, Qing Lan, Nathaniel Rothman, Tongzhang Zheng, and Shuangge Ma. *Identification of gene-environment interactions in cancer studies using penalization. Genomics, 102(4):189-194, 2013.*
http://europepmc.org/backend/ptpmcrender.fcgi?accid=PMC3869641&blobtype=pdf

See Also

`coef`, and `plot` and `bic.BLMCP` methods, and `BLMCP`.

predict.bic.PTReg *Make predictions from a "bic.PTReg" object*

Description

This function makes predictions from a BIC PTReg model, using the stored "bic.PTReg" object. This function makes it easier to use the results of BIC to make a prediction.

Usage

```
## S3 method for class 'bic.PTReg'
predict(object, newE, newG, ...)
```

Arguments

object	Fitted "bic.PTReg" object.
newE	Matrix of new values for E at which predictions are to be made.
newG	Matrix of new values for G at which predictions are to be made.
...	Not used. Other arguments to predict.

Value

The object returned depends on the ... argument which is passed on to the predict method for PTReg objects.

References

Yaqing Xu, Mengyun Wu, Shuangge Ma, and Syed Ejaz Ahmed. *Robust gene-environment interaction analysis using penalized trimmed regression. Journal of Statistical Computation and Simulation, 88(18):3502-3528, 2018.*

See Also

bic.PTReg, and coef, and plot methods, and PTReg.

predict.BLMCP *Make predictions from a "BLMCP" object*

Description

This function makes predictions from a BLMCP model, using the stored "BLMCP" object.

Usage

```
## S3 method for class 'BLMCP'
predict(object, newE, newG, ...)
```

Arguments

object	Fitted "BLMCP" object.
newE	Matrix of new values for E at which predictions are to be made.
newG	Matrix of new values for G at which predictions are to be made.
...	Not used. Other arguments to predict.

Value

The object returned depends on the ... argument which is passed on to the predict method for BLMCP objects.

References

Mengyun Wu, Yangguang Zang, Sanguo Zhang, Jian Huang, and Shuangge Ma. *Accommodating missingness in environmental measurements in gene-environment interaction analysis. Genetic Epidemiology, 41(6):523-554, 2017.*
Jin Liu, Jian Huang, Yawei Zhang, Qing Lan, Nathaniel Rothman, Tongzhang Zheng, and Shuangge Ma. *Identification of gene-environment interactions in cancer studies using penalization. Genomics, 102(4):189-194, 2013.*

See Also

BLMCP, coef, and plot methods, and bic.BLMCP method.

---

predict.Miss.boosting *Make predictions from a "Miss.boosting" object*

Description

This function makes predictions from a Miss.boosting model, using the stored "Miss.boosting" object.

Usage

```
## S3 method for class 'Miss.boosting'
predict(object, newE, newG, ...)
```

Arguments

object	Fitted "Miss.boosting" object.
newE	Matrix of new values for E at which predictions are to be made.
newG	Matrix of new values for G at which predictions are to be made.
...	Not used. Other arguments to predict.

Value

The object returned depends on the ... argument which is passed on to the `predict` method for `Miss.boosting` objects.

References

Mengyun Wu and Shuangge Ma. *Robust semiparametric gene-environment interaction analysis using sparse boosting. Statistics in Medicine, 38(23):4625-4641, 2019.*

See Also

`Miss.boosting`, and `plot` methods.

`predict.PTReg` *Make predictions from a "PTReg" object*

Description

This function makes predictions from a PTReg model, using the stored `"PTReg"` object.

Usage

```
## S3 method for class 'PTReg'
predict(object, newE, newG, ...)
```

Arguments

`object`	Fitted `"PTReg"` object.
`newE`	Matrix of new values for `E` at which predictions are to be made.
`newG`	Matrix of new values for `G` at which predictions are to be made.
`...`	Not used. Other arguments to predict.

Value

The object returned depends on the ... argument which is passed on to the `predict` method for `PTReg` objects.

References

Yaqing Xu, Mengyun Wu, Shuangge Ma, and Syed Ejaz Ahmed. *Robust gene-environment interaction analysis using penalized trimmed regression. Journal of Statistical Computation and Simulation, 88(18):3502-3528, 2018.*

See Also

`PTReg`, `coef` and `plot` methods, and `bic.PTReg`.

predict.RobSBoosting *Make predictions from a "RobSBoosting" object*

Description

This function makes predictions from a RobSBoosting model, using the stored "RobSBoosting" object.

Usage

```
## S3 method for class 'RobSBoosting'
predict(object, newE, newG, ...)
```

Arguments

object	Fitted "RobSBoosting" object.
newE	Matrix of new values for E at which predictions are to be made.
newG	Matrix of new values for G at which predictions are to be made.
...	Not used. Other arguments to predict.

Value

The object returned depends on the ... argument which is passed on to the predict method for RobSBoosting objects.

References

Mengyun Wu and Shuangge Ma. *Robust semiparametric gene-environment interaction analysis using sparse boosting. Statistics in Medicine, 38(23):4625-4641, 2019.*

See Also

RobSBoosting, coef, and plot methods.

PTReg *Robust gene-environment interaction analysis using penalized trimmed regression*

Description

Gene-environment interaction analysis using penalized trimmed regression, which is robust to outliers in both predictor and response spaces. The objective function is based on trimming technique, where the samples with extreme absolute residuals are trimmed. A decomposition framework is adopted for accommodating "main effects-interactions" hierarchy, and minimax concave penalty (MCP) is adopted for regularized estimation and interaction (and main genetic effect) selection.

Usage

```
PTReg(
  G,
  E,
  Y,
  lambda1,
  lambda2,
  gamma1 = 6,
  gamma2 = 6,
  max_init,
  h = NULL,
  tau = 0.4,
  mu = 2.5,
  family = c("continuous", "survival")
)
```

Arguments

G	Input matrix of p genetic (G) measurements consisting of n rows. Each row is an observation vector.
E	Input matrix of q environmental (E) risk factors. Each row is an observation vector.
Y	Response variable. A quantitative vector for family="continuous". For family="survival", Y should be a two-column matrix with the first column being the log(survival time) and the second column being the censoring indicator. The indicator is a binary variable, with "1" indicating dead, and "0" indicating right censored.
lambda1	A user supplied lambda for MCP accommodating main G effect selection.
lambda2	A user supplied lambda for MCP accommodating G-E interaction selecton.
gamma1	The regularization parameter of the MCP penalty corresponding to G effects.
gamma2	The regularization parameter of the MCP penalty corresponding to G-E interactions.
max_init	The number of initializations.
h	The number of the trimmed samples if the parameter mu is not given.
tau	The threshold value used in stability selection.
mu	The parameter for screening outliers with extreme absolute residuals if the number of the trimmed samples h is not given.
family	Response type of Y (see above).

Value

An object with S3 class "PTReg" is returned, which is a list with the following components.

call	The call that produced this object.
intercept	The intercept estimate.
alpha	The matrix of the coefficients for main E effects.

beta	The matrix of the regression coefficients for all main G effects (the first row) and interactions.
df	The number of nonzeros.
BIC	Bayesian Information Criterion.
select_sample	Selected samples where samples with extreme absolute residuals are trimmed.
family	The same as input family.

References

Yaqing Xu, Mengyun Wu, Shuangge Ma, and Syed Ejaz Ahmed. *Robust gene-environment interaction analysis using penalized trimmed regression. Journal of Statistical Computation and Simulation, 88(18):3502-3528, 2018.*

See Also

coef, predict, and plot methods, and bic.PTReg method.

Examples

```
sigmaG<-AR(rho=0.3,p=30)
sigmaE<-AR(rho=0.3,p=3)
set.seed(300)
G=MASS::mvrnorm(150,rep(0,30),sigmaG)
EC=MASS::mvrnorm(150,rep(0,2),sigmaE[1:2,1:2])
ED = matrix(rbinom((150),1,0.6),150,1)
E=cbind(EC,ED)
alpha=runif(3,0.8,1.5)
beta=matrix(0,4,30)
beta[1,1:4]=runif(4,1,1.5)
beta[2,c(1,2)]=runif(2,1,1.5)


#continuous response
y1=simulated_data(G=G,E=E,alpha=alpha,beta=beta,error=c(rnorm(130),
rcauchy(20,0,5)),family="continuous")
fit1<-PTReg(G=G,E=E,y1,lambda1=0.3,lambda2=0.3,gamma1=6,gamma2=6,
max_init=50,h=NULL,tau=0.6,mu=2.5,family="continuous")
coef1=coef(fit1)
y_hat1=predict(fit1,E,G)
plot(fit1)

# survival response
y2=simulated_data(G,E,alpha,beta,rnorm(150,0,1),
family="survival",0.7,0.9)
fit2<-PTReg(G=G,E=E,y2,lambda1=0.3,lambda2=0.3,gamma1=6,gamma2=6,
max_init=50,h=NULL,tau=0.6,mu=2.5,family="survival")
coef2=coef(fit2)
y_hat2=predict(fit2,E,G)
plot(fit2)
```

QPCorr.matrix *Robust identification of gene-environment interactions using a quantile partial correlation approach*

Description

A robust gene-environment interaction identification approach using the quantile partial correlation technique. This approach is a marginal analysis approach built on the quantile regression technique, which can accommodate long-tailed or contaminated outcomes. For response with right censoring, Kaplan-Meier (KM) estimator-based weights are adopted to easily accommodate censoring. In addition, it adopts partial correlation to identify important interactions while properly controlling for the main genetic (G) and environmental (E) effects.

Usage

```
QPCorr.matrix(G, E, Y, tau, w = NULL, family = c("continuous", "survival"))
```

Arguments

G	Input matrix of p G measurements consisting of n rows. Each row is an observation vector.
E	Input matrix of q E risk factors. Each row is an observation vector.
Y	Response variable. A quantitative vector for family="continuous". For family="survival", Y should be a two-column matrix with the first column being the log(survival time) and the second column being the censoring indicator. The indicator is a binary variable, with "1" indicating dead, and "0" indicating right censored.
tau	Quantile.
w	Weight for accommodating censoring if family="survival". Default is NULL and a Kaplan-Meier estimator-based weight is used.
family	Response type of Y (see above).

Value

Matrix of (censored) quantile partial correlations for interactions.

References

Yaqing Xu, Mengyun Wu, Qingzhao Zhang, and Shuangge Ma. *Robust identification of gene-environment interactions for prognosis using a quantile partial correlation approach. Genomics, 111(5):1115-1123, 2019.*

See Also

QPCorr.pval method.

Examples

```
alpha=matrix(0,5,1)
alpha[1:2]=1
beta=matrix(0,6,100)
beta[1,1:5]=1
beta[2:3,1:5]=2
beta[4:6,6:7]=2
sigmaG<-AR(rho=0.3,100)
sigmaE<-AR(rho=0.3,5)
G<-MASS::mvrnorm(200,rep(0,100),sigmaG)
E<-MASS::mvrnorm(200,rep(0,5),sigmaE)
e1<-rnorm(200*.05,50,1);e2<-rnorm(200*.05,-50,1);e3<-rnorm(200*.9)
e<-c(e1,e2,e3)

# continuous
y1=simulated_data(G=G,E=E,alpha=alpha,beta=beta,error=e,family="continuous")
cpqcorr_stat1<-QPCorr.matrix(G,E,y1,tau=0.5,w=NULL,family="continuous")

# survival
y2=simulated_data(G,E,alpha,beta,rnorm(200,0,1),family="survival",0.7,0.9)
cpqcorr_stat<-QPCorr.matrix(G,E,y2,tau=0.5,w=NULL,family="survival")
```

QPCorr.pval *P-values of the "QPCorr.matrix" obtained using a permutation approach*

Description

P-values of the "QPCorr.matrix " obtained using a permutation approach, the interactions with smaller p-values are regarded as more important.

Usage

```
QPCorr.pval(
  G,
  E,
  Y,
  tau,
  w = NULL,
  permutation_t = 1000,
  family = c("continuous", "survival")
)
```

Arguments

G	Input matrix of p genetic (G) measurements consisting of n rows. Each row is an observation vector.

E	Input matrix of q environmental (E) risk factors, each row is an observation vector.
Y	Response variable. A quantitative vector for family="continuous". For family="survival", Y should be a two-column matrix with the first column being the log(survival time) and the second column being the censoring indicator. The indicator is a binary variable, with "1" indicating dead, and "0" indicating right censored.
tau	Quantile.
w	Weight for accommodating censoring if family="survival". Default is NULL and a Kaplan-Meier estimator-based weight is used.
permutation_t	Number of permutation.
family	Response type of Y (see above).

Value

Matrix of p-value, with the element in the ith row and the j column represents the p-value of the (censored) quantile partial correlation corresponding to the ith E and the jth G.

References

Yaqing Xu, Mengyun Wu, Qingzhao Zhang, and Shuangge Ma. *Robust identification of gene-environment interactions for prognosis using a quantile partial correlation approach. Genomics, 111(5):1115-1123, 2019.*

See Also

QPCorr.matrix method.

Examples

```
n=50
alpha=matrix(0,5,1)
alpha[1:2]=1
beta=matrix(0,6,20)
beta[1,1:4]=1
beta[2:3,1:4]=2
sigmaG<-AR(rho=0.3,20)
sigmaE<-AR(rho=0.3,5)
G<-MASS::mvrnorm(n,rep(0,20),sigmaG)
E<-MASS::mvrnorm(n,rep(0,5),sigmaE)
e1<-rnorm(n*.05,50,1);e2<-rnorm(n*.05,-50,1);e3<-rnorm((n-length(e1)-length(e2)))
e<-c(e1,e2,e3)


# continuous
y1=simulated_data(G=G,E=E,alpha=alpha,beta=beta,error=e,family="continuous")
cpqcorr_pvalue1<-QPCorr.pval(G,E,y1,tau=0.5,permutation_t=500,family="continuous")

# survival
y2=simulated_data(G,E,alpha,beta,rnorm(n,0,1),family="survival",0.7,0.9)
cpqcorr_pvalue2<-QPCorr.pval(G,E,y2,tau=0.5,permutation_t=500,family="survival")
```

RobSBoosting *Robust semiparametric gene-environment interaction analysis using sparse boosting*

Description

Robust semiparametric gene-environment interaction analysis using sparse boosting. Here a semiparametric model is assumed to accommodate nonlinear effects, where we model continuous environmental (E) factors in a nonlinear way, and discrete E factors and all genetic (G) factors in a linear way. For estimating the nonlinear functions, the B spline expansion is adopted. The Huber loss function and Qn estimator are adopted to accommodate long-tailed distribution/data contamination. For model estimation and selection of relevant variables, we adopt an effective sparse boosting approach, where the strong hierarchy is respected.

Usage

```
RobSBoosting(
  G,
  E,
  Y,
  loop_time,
  num.knots = NULL,
  Boundary.knots = NULL,
  degree = 1,
  v = 0.1,
  family = c("continuous", "survival"),
  knots = NULL,
  E_type
)
```

Arguments

G	Input matrix of p genetic measurements consisting of n rows. Each row is an observation vector.
E	Input matrix of q environmental risk factors, each row is an observation vector.
Y	Response variable. A quantitative vector for family="continuous". For family="survival", Y should be a two-column matrix with the first column being the log(survival time) and the second column being the censoring indicator. The indicator is a binary variable, with "1" indicating dead, and "0" indicating right censored.
loop_time	Number of iterations of the sparse boosting.
num.knots	Numbers of knots for the B spline basis.
Boundary.knots	The boundary of knots for the B spline basis.
degree	Degree for the B spline basis.

v	The step size used in the sparse boosting process. Default is 0.1.
family	Response type of Y (see above).
knots	List of knots for the B spline basis. Default is NULL and knots can be generated with the given num.knots, degree and Boundary.knots.
E_type	A vector indicating the type of each E factor, with "ED" representing discrete E factor, and "EC" representing continuous E factor.

Value

An object with S3 class "RobSBoosting" is returned, which is a list with the following components.

call	The call that produced this object.
max_t	The stopping iteration time of the sparse boosting.
spline_result	A list of length max_t that includes the estimation results of each iteration.
BIC	A vector of length max_t that includes Bayesian Information Criterion based on the Huber's prediction error.
variable	A vector of length max_t that includes the index of selected variable in each iteration.
id	The iteration time with the smallest BIC.
variable_pair	A matrix with two columns that include the set of variables that can potentially enter the regression model at the stopping iteration time. Here, the first and second columns correspond to the indexes of E factors and G factors. For example, (1, 0) represents that this variable is the first E factor, and (1,2) represents that the variable is the interaction between the first E factor and second G factor.
v_type	A vector whose length is the number of rows of variable_pair, with each element representing the variable type of the corresponding row of variable_pair. Here, "EC" stands for continuous E effect, "ED" for discrete E effect, and "G" for G effect, "EC-G" for the interaction between "EC" and "G", "ED-G" for the interaction between "ED" and "G".
family	The same as input family.
degree	Degree for the B spline basis.
v	The step size used in the sparse boosting process.
NorM	The values of B spline basis.
estimation_results	A list of estimation results for each variable. Here, the first q elemnets are for the E effects, the (q+1) element is for the first G effect and the (q+2) to (2q+1) elements are for the interactions corresponding to the first G factor, and so on.

References

Mengyun Wu and Shuangge Ma. *Robust semiparametric gene-environment interaction analysis using sparse boosting. Statistics in Medicine, 38(23):4625-4641, 2019.*

See Also

bs method for B spline expansion, coef, predict, and plot methods, and Miss.boosting method.

Examples

```
data(Rob_data)
G=Rob_data[,1:20];E=Rob_data[,21:24]
Y=Rob_data[,25];Y_s=Rob_data[,26:27]
knots = list();Boundary.knots = matrix(0, 24, 2)
for(i in 1:4) {
  knots[[i]] = c(0, 1)
  Boundary.knots[i, ] = c(0, 1)
  }

#continuous
fit1= RobSBoosting(G,E,Y,loop_time = 80,num.knots = 2,Boundary.knots=Boundary.knots,
degree = 2,family = "continuous",knots = knots,E_type=c("EC","EC","ED","ED"))
coef1 = coef(fit1)
predict1=predict(fit1,newE=E[1:2,],newG=G[1:2,])
plot(fit1)


#survival
fit2= RobSBoosting(G,E,Y_s,loop_time = 200, num.knots = 2, Boundary.knots=Boundary.knots,
family = "survival", knots = knots,E_type=c("EC","EC","ED","ED"))
coef2 = coef(fit2)
predict2=predict(fit2,newE=E[1:2,],newG=G[1:2,])
plot(fit2)
```

Rob_data *A matrix containing the simulated data for* RobSBoosting *and* Miss.boosting *methods*

Description

A matrix containing the simulated genetic (G) effects (the first 20 columns), environmental (E) effects (column 21 to column 24), continuous response (column 25), logarithm of survival time (column 26), and censoring indicator (column 27).

Usage

```
data(Rob_data)
```

Format

A matrix with 100 rows and 27 variables.

Examples

```
data(Rob_data)
```

`simulated_data` *Simulated data for generating response*

Description

Generate simulated response.

Usage

```
simulated_data(
  G,
  E,
  alpha,
  beta,
  error,
  family = c("continuous", "survival"),
  a1 = NULL,
  a2 = NULL
)
```

Arguments

`G`	Input matrix of `p` genetic (G) measurements consisting of `n` rows. Each row is an observation vector.
`E`	Input matrix of `q` environmental (E) risk factors. Each row is an observation vector.
`alpha`	Matrix of the true coefficients for main E effects.
`beta`	Matrix of the true regression coefficients for all main G effects (the first row) and interactions.
`error`	Error terms.
`family`	Type of the response variable. If `family="continuous"`, a quantitative vector is generated. If `family="survival"`, a two-column matrix with the first column being the log(survival time) and the second column being the censoring indicator is generated.The indicator is a binary variable, with "1" indicating dead, and "0" indicating right censored.
`a1`	If `family="survival"`, we generate the censoring time from a uniform distribution where `a1` is the left endpoint.
`a2`	If `family="survival"`, we generate the censoring time from a uniform distribution where `a2` is the right endpoint.

Value

Response variable. A quantitative vector for `family="continuous"`. For `family="survival"`, it would be a two-column matrix with the first column being the log(survival time) and the second column being the censoring indicator. The indicator is a binary variable, with "1" indicating dead, and "0" indicating right censored.

Index