Package: mmabig (via r-universe)

October 9, 2026

Type Package

Title Multiple Mediation Analysis for Big Data Sets

Version 3.2-0

Date 2023-08-29

Author Qingzhao Yu and Bin Li

Maintainer Qingzhao Yu <qyu@lsuhsc.edu>

Depends R (>= 2.14.1), glmnet, mma, splines, survival, car

Imports gplots, methods

Suggests knitr, rmarkdown

VignetteBuilder knitr

Encoding UTF-8

Description Used for general multiple mediation analysis with big data sets.

License GPL (>= 2)

URL https://www.r-project.org,
https://publichealth.lsuhsc.edu/Faculty_pages/qyu/index.html

RoxygenNote 7.1.1

NeedsCompilation no

Config/pak/sysreqs cmake make libicu-dev

Repository https://qingzhaoyu.r-universe.dev

Date/Publication 2023-08-29 17:31:22 UTC

RemoteUrl https://github.com/cran/mmabig

RemoteRef HEAD

RemoteSha 1d30fb4bbf37ef14008d3be916a66c049309d0a9

Contents

mmabig-package 2
data.org.big 3
joint.effect 6
med.big 7
mma.big 9
plot.mma.big 19
print.med.big 20
print.mma.big 21
summary.med_iden.big 22
summary.mma.big 23

Index **25**

`mmabig-package` *Mediation Analysis Package for Big Data Sets*

Description

This package is used to identify mediators and for general mediation analysis with big data sets. Mediation effect refers to the effect conveyed by intervening variables to an observed relationship between an exposure and a response variable (outcome). In this package, the exposure is called the predictor, the intervening variables are called mediators. The mediation effects include the total effect, direct effect, and indirect effect, which are defined and the statistical inferences described in Li and Yu (2018).

Details

"`data.org.big`" is used to identify potential mediators. "`med.big`" , and "`mma.big`" is a combined function that automatically identify potential mediators, based on which to make statistical inference on the mediation effects.

Author(s)

Qingzhao Yu <qyu@lsuhsc.edu> and Bin Li <bli@lsu.edu>

Maintainer: Qingzhao Yu <qyu@lsuhsc.edu>

References

Yu, Q., Fan, Y., and Wu, X. (2014) <doi:10.4172/2155-6180.1000189>. "General Multiple Mediation Analysis With an Application to Explore Racial Disparity in Breast Cancer Survival," Journal of Biometrics & Biostatistics,5(2): 189.

Yu, Q., Scribner, R.A., Leonardi, C., Zhang, L., Park, C., Chen, L., and Simonsen, N.R. (2017) <doi:10.1016/j.sste.2017.02.001>. "Exploring racial disparity in obesity: a mediation analysis considering geo-coded environmental factors," Spatial and Spatio-temporal Epidemiology, 21, 13-23.

Yu, Q., and Li, B. (2017) <doi:10.5334/hors.160>. "mma: An r package for multiple mediation analysis," Journal of Open Research Software, 5(1), 11.

Li, B. and Yu, Q. (2018). "Mediation Analysis with Large Data Sets".

Yu, Q. and Li, B., 2022. Statistical Methods for Mediation, Confounding and Moderation Analysis Using R and SAS. Chapman and Hall/CRC. ISBN 9780367365479.

`data.org.big` *Data Organization and Identify Potential Mediators*

Description

Do a preliminary data analysis to identify potential mediators and covariates. Each variable listed in jointm is forced in the final estimation model as a mediator. Also organize the data into a format that can be directly used for the mediation analysis functions.

Usage

```
data.org.big(x,y,pred,mediator=NULL,contmed=NULL,binmed=NULL,binref=NULL,catmed=NULL,
                       catref=NULL,jointm=NULL,
                       family1=as.list(rep(NA,ncol(data.frame(y)))),
                predref=NULL,alpha=1,alpha1=0.01,alpha2=0.01,testtype=2, w=NULL,
                       lambda=exp(seq(log(0.001), log(5), length.out=15)))
```

Arguments

`x`	a data frame contains the predictor, all potential mediators and covariates
`y`	the vector of outcome variable.The outcome can be binary, continuous, multi-categorical or of "Surv" class (see survival package for help). For a factor, the last level in alphabetical order is the target class.
`pred`	the column or matrix of predictor(s): the predictor is the exposure variable, it can be a binary or multi-categorical factor or one/a few contiuous variable(s).
`mediator`	the list of mediators (column numbers in x or by variable names). The mediators to be checked can be identified by "contmed", "binmed" and "catmed", or by this argument, "mediator", where binary and categorical mediators in x are identified as factors or characters, the reference group is the first level of the factor or factorized character. if a mediator has only two unique values, the mediator is identified as binary. If the reference groups need to be changed, the binary or categorical mediators can be listed in binmed or catmed, and the corresponding reference group in binref or catref.
`contmed`	a vector of variable names or column numbers that locate the potential continuous mediators in x.
`binmed`	a vector of column numbers that locate the potential binary mediators in x.
`binref`	the defined reference groups of the binary potential mediators in binmed. The first levels of the mediators if is null.
`catmed`	a vector of variable names or column numbers that locate the potential categorical mediators in x. The first levels of the mediators if is null.

catref	the defined reference groups of the categorical potential mediators in catmed.
jointm	a vector of variables that need to be forced in as mediators.
family1	define the conditional distribution of y given x, and the linkage function that links the mean of y with the system component in generalized linear model. The default value of family1 is binomial(link = "logit") for binary y, and gaussian(link="identity") for continuous y.
predref	if the predictor is binary, identify the reference group of the binary predictor. The default is the first level of the predictor. The value of the predictor is 0 for the reference grop.
alpha	the elasticnet mixing parameter such that 0<=alpha<=1, with alpha=1 is the lasso penalty, and alpha=0 be the ridge penalty. By default, alpha=1.
alpha1	the significance level at which to test if the potential mediators (identified by contmed, binmed, and catmed) can be used as a covariate or mediator in estimating y when all variables in x are included in the model. The default value is alpha1=0.01.
alpha2	the significant level at which to test if a potential mediator is related with the predictor. The default value is alpha2=0.01.
testtype	if the testtype is 1, covariates/mediators are identified using full model; if the testtype is 2 (by default), covariates/mediators are tested one by one in models with the predictor only.
w	the weight for data analysis, by default is rep(1,length(y)).
lambda	the lambda sequence when fit cv.glmnet. By default, it is exp(seq(log(0.001), log(5), length.out=15)).

Value

data.org.big returns a "med_iden" class (see mma package), a list with the organized data and identifiers of the potential mediators in the organized data set.

x	the organized data frame that include all potential mediators and covariates that should be used to estimate the outcome.
dirx	the vector/matrix of predictor(s)/exposure variable(s).
contm	the column numbers of x that locate the potential continuous mediators.
binm	when the predictor is continuous, binm gives the column numbers of x that locate the potential binary mediators.
catm	when the predictor is binary, catm gives the column numbers of x that locate the potential binary or categorical mediators; when the predictor is continuous, catm gives a list where the first item is the number of potential categorical mediators, and the following items give the column numbers of each binarized categorical mediator in x.
jointm	a list where the first item is the number of groups of joint mediators, and each of the following items identifies the column numbers of the mediators in the newly organized x for each group of joint mediators.
y	the vector/matrix of outcome(s).

y_type	the variable type of outcome(s): 1 is continuous, 2 is binary, 3 is reserved for multi-categorical (no 3 would show in y_type, since all categorical responses are binarized), and 4 is survival.
fullmodel	a list with each item the full linear model fitted with all potential mediators and covariates for each response.
rela	p-values of tests on the realtionship between the predictor(s) and each potiential mediator.
P1	If testtype=2, P1 gives the p-value of the corresponding variables in predicting the outcome(s) when only the variable and predictor are covariates in the model.

Note

All other variables in x but not identified by mediator, contmed, binmed, or catmed are forced in the final model as covariates. Compared with data.org, joint mediators are considered in this function. Every variable in the jointm should be listed in contmed, binmed, or catmed, and these variables are forced to be included as mediators for further mediation analysis. A variables can be included in more than one groups of joint mediators in jointm.

Author(s)

Qingzhao Yu <qyu@lsuhsc.edu> and Bin Li <bli@lsu.edu>

References

Baron, R.M., and Kenny, D.A. (1986) <doi:10.1037/0022-3514.51.6.1173>. The moderator-mediator variable distinction in social psychological research: conceptual, strategic, and statistical considerations. J. Pers Soc Psychol, 51(6), 1173-1182.

Yu, Q., Fan, Y., and Wu, X. (2014) <doi:10.4172/2155-6180.1000189>. "General Multiple Mediation Analysis With an Application to Explore Racial Disparity in Breast Cancer Survival," Journal of Biometrics & Biostatistics,5(2): 189.

Yu, Q., Scribner, R.A., Leonardi, C., Zhang, L., Park, C., Chen, L., and Simonsen, N.R. (2017) <doi:10.1016/j.sste.2017.02.001>. "Exploring racial disparity in obesity: a mediation analysis considering geo-coded environmental factors," Spatial and Spatio-temporal Epidemiology, 21, 13-23.

Yu, Q., and Li, B. (2017) <doi:10.5334/hors.160>. "mma: An r package for multiple mediation analysis," Journal of Open Research Software, 5(1), 11.

Li, B. and Yu, Q. (2018). "Mediation Analysis with Large Data Sets".

Examples

```
#See help(mma.big)
```

`joint.effect` *Summary of the joint effect of a list of mediators*

Description

Compute the estimations, standard deviations and confidence intervals of the joint effect of the list of variables identified by vari.

Usage

```
joint.effect(object,vari,alpha=0.05)
```

Arguments

`object`	a mma.big object created initially call to mma.big.
`vari`	a list of mediators whose joint effect is to be estimated. vari can alsobe the name of a multicategorical mediator.
`alpha`	the significance level for the confidence interval.

Details

joint.effect gives a list of the estimations and summary statistics for the joint mediation effect of a list of mediators based on the bootstrap results.

Value

The function joint.effect return a list. The first item list all the included variables. The second item, effect, is the list for the joint mediation effects. The third item, relative.effect, is the list of relative effects. Under them, est is the estimation of the corresponding (relative) joint mediation effect based on the whole data; mean is the average estimated (relative) effects from the bootstrap samples; and sd is the standard deviation of the estimate from the bootstrap sample. upbd and lwbd are the upper and lower bounds of the confidence interval of the estimation using parametric method from the bootstrap sample, upbd_q and lwbd_q are the corresponding quantiles of the estimation from the bootstrap sample.

Author(s)

Qingzhao Yu <qyu@lsuhsc.edu> and Bin Li <bli@lsu.edu>

References

Yu, Q., Fan, Y., and Wu, X. (2014) <doi:10.4172/2155-6180.1000189>. "General Multiple Mediation Analysis With an Application to Explore Racial Disparity in Breast Cancer Survival," Journal of Biometrics & Biostatistics,5(2): 189.

Yu, Q., Scribner, R.A., Leonardi, C., Zhang, L., Park, C., Chen, L., and Simonsen, N.R. (2017) <doi:10.1016/j.sste.2017.02.001>. "Exploring racial disparity in obesity: a mediation analysis considering geo-coded environmental factors," Spatial and Spatio-temporal Epidemiology, 21, 13-23.

Yu, Q., and Li, B. (2017) <doi:10.5334/hors.160>. "mma: An r package for multiple mediation analysis," Journal of Open Research Software, 5(1), 11.

Li, B. and Yu, Q. (2018). "Mediation Analysis with Large Data Sets".

See Also

"mma.big"

Examples

```
#See help(mma.big)
```

`med.big` *Mediation Analysis with Binary or Continuous Predictor*

Description

To estimate the mediation effects when the predictor is binary or continuous.

Usage

```
med.big(data, x=data$x, y=data$y, dirx=data$dirx, binm=data$binm,
                  contm = data$contm, catm = data$catm, jointm = data$jointm,
                  allm = c(contm, binm, unique(unlist(catm)[-1])),margin=1,
                  df=1,family1=data$family1,refy=rep(NA,ncol(y)),
                  binpred=data$binpred,type=NULL,w=NULL,alpha=1,
                  lambda=exp(seq(log(0.001), log(5), length.out=15)))
```

Arguments

`data`	the list of result from data.org that organize the covariates, mediators, predictor and outcome. If data is FALSE, then need to set x1, y1, dirx, contm, catm, and jointm.
`x`	a data frame contains all mediators and covariates. Need to set up only when data is FALSE.
`y`	the vector of outcome variable. Need to set up only when data is FALSE.
`dirx`	the vector or matrix of predictor(s). The reference group is set to be 0. Need to set up only when data is FALSE.
`binm`	the variable names or the column number of x that locates the binary mediators. Need to set up only when data is FALSE.
`contm`	the variable names or the column numbers of x that locate the potential continuous mediators. Need to set up only when data is FALSE.
`catm`	categorical mediators should be binarized and be presented as a list, where the first item is the number of categorical variables and the following items are the names or the column numbers of each binarized categorical variable in x. data.org organizes the categorical mediators in this format after they pass the mediator tests. Need to set up only when data is FALSE.

jointm	a list where the first item is the number of groups of joint mediators to be considered, and each of the following items identifies the names or the column numbers of the mediators in x for each group of joint mediators. Need to set up only when data is FALSE.
allm	the column numbers of all mediators. Need to set up only when data is FALSE. The default value of allm is c(contm,catm).
margin	the change in predictor when calculating the mediation effects, see Yu et al. (2014).
df	Natural cubic spline will be used to fit the relationship between the predictor and each mediator when the predictors are continuous. The df is the degree of freedom in the ns() function, the default is 1 for linear relationship.
family1	a list with the ith item define the conditional distribution of y[,i] given x, and the linkage function that links the mean of y with the system component if generalized linear model is used as the final full model. The default value of family1 is gaussian(link="identity") for contiuous y[,i], and binomial(link = "logit") for binary y[,i].
refy	if y is binary, the reference group of y.
binpred	if TRUE, the predict variable is binary.
type	the type of prediction when y is class Surv. By default, type is "risk".
w	the weight for each case in x.
alpha	the elasticnet mixing parameter such that 0<=alpha<=1, with alpha=1 is the lasso penalty, and alpha=0 be the ridge penalty. By default, alpha=1.
lambda	the lambda sequence when fit cv.glmnet. By default, it is exp(seq(log(0.001), log(5), length.out=15)).

Details

The mediators are not tested in this function. data.org.big should be used first for the tests and data organizing, and then the resulting list from data.org.big can be used directly to define the arguments in this function. med considers all variables in x as mediators or covariates in the final model and all variables identified by contm, binm, catm, or jointm as mediators.

Value

The result is an med.big object with:

denm	a matrix gives the estimated direct effects. Each row is for a different response variable, and each column is for difference predictor.
model	a list, where full.model item is the dth item is the fitted final full model where y[,d] is the outcome and all predictor, covariates, and transformed mediators are the explanatory variables. Other items including family, surv, and biny.
coef.m	a matrix of fitted coefficients for transformed mediators where each row is for a different response variable.
coef.model	a matrix of fitted coefficients from the full model where each row is for a different response variable.

deltaM — The transformation vector for mediators.

dm — If the predictors are binary, dm is a matrix showing the changes in the mean of mediators when the predictor changes from the reference group to the corresponding group. Each row is for a differenct predictor. If the predictors are continuous, dm is a list. Each item is a matrix for a different predictor. The matrix shows the changes in the mean of mediators when the corresponding predictor changes by one margin.

Author(s)

Qingzhao Yu <qyu@lsuhsc.edu> and Bin Li <bli@lsu.edu>

References

Yu, Q., Fan, Y., and Wu, X. (2014) <doi:10.4172/2155-6180.1000189>. "General Multiple Mediation Analysis With an Application to Explore Racial Disparity in Breast Cancer Survival," Journal of Biometrics & Biostatistics,5(2): 189.

Yu, Q., Scribner, R.A., Leonardi, C., Zhang, L., Park, C., Chen, L., and Simonsen, N.R. (2017) <doi:10.1016/j.sste.2017.02.001>. "Exploring racial disparity in obesity: a mediation analysis considering geo-coded environmental factors," Spatial and Spatio-temporal Epidemiology, 21, 13-23.

Yu, Q., and Li, B. (2017) <doi:10.5334/hors.160>. "mma: An r package for multiple mediation analysis," Journal of Open Research Software, 5(1), 11.

Li, B. and Yu, Q. (2018). "Mediation Analysis with Large Data Sets".

See Also

"mma.big" to make inferences on the estimated mediation effects using bootstrap method.

Examples

```
#See help(mma.big)
```

---

mma.big — *Multiple Mediation Analysis for mediaiton analysis with big data sets*

Description

Test for mediators and do statistical inferences on the identified mediation effects.

Usage

```
mma.big(data=NULL,x=data$x, y=data$y,pred=data$dirx, mediator=NULL, binm=data$binm,
     contm=data$contm,catm=data$catm,jointm=data$jointm,margin=1,df=1,binref=NULL,
        catref=NULL,predref=NULL,alpha=1,alpha1=0.01,alpha2=0.01,
        family1=data$family1,n2=50,w=rep(1,nrow(x)),
        refy=NULL,pred.new=NULL,binpred=data$binpred,type=NULL,w.new=NULL,
        lambda=exp(seq(log(0.001), log(5), length.out=15)))
```

Arguments

data	the organized dataset from data.org.big. The default is NULL
x	a data frame contains the predictor, all potential mediators and covariates.
y	the vector of outcome variable.
pred	the vector/matrix of the predictor(s).
mediator	the list of mediators (column numbers in x or by variable names). The mediators to be checked can be identified by "contmed", "binmed" and "catmed", or by this argument, "mediator", where binary and categorical mediators in x are identified by factors, the reference group is the first level of the factor.
binm	a vector of column numbers that locate the potential binary mediators in x.
contm	a vector of column numbers that locate the potential continuous mediators in x.
catm	a vector of column numbers that locate the potential categorical mediators in x.
jointm	a list that identifies the mediators that need to be forced as mediators.
margin	if binpred is FALSE, define the change in predictor when calculating the mediation effects, see Yu et al. (2014).
df	Natural cubic spline will be used to fit the relationship between the countinuous predictor and each mediator. The df is the degree of freedom in the ns() function, the default is 1.
binref	the defined reference groups of the binary potential mediators in binm.
catref	the defined reference groups of the categorical potential mediators in catm.
predref	if predictor is binary, identify the reference group of the binary predictor. By default, the reference group will be the first level of the predictor.
alpha	the elasticnet mixing parameter such that 0<=alpha<=1, with alpha=1 is the lasso penalty, and alpha=0 be the ridge penalty. By default, alpha=1.
alpha1	the significance level at which to test if the potential mediators (identified by contmed, binmed, and catmed) can be used as a covariate or mediator in estimating y when all variables in x are included in the model. The default value is alpha=0.1
alpha2	the significant level at which to test if a potential mediator is related with the predictor. The default value is alpha2=0.1.
family1	define the conditional distribution of y given x, and the linkage function that links the mean of y with the system component if generalized linear model is used as the final full model. The default value of family1 is binomial(link = "logit") for binary y, gaussian(link="identity") for continuous y.
n2	the number of times of bootstrap resampling. The default value is n2=50.
w	the weight for each observation.
refy	if y is binary, the reference group of y.By default, the reference group will be the first level of as.factor(y).
pred.new	A new set of predictor(s), of the same format as x (after data.org), on which to calculate the mediation effects. For continuous predictor only.
binpred	indicator of whether the predictor is binary or not.

type	the type of prediction when y is class Surv. Is "risk" if not specified.
w.new	the weights for new.x.
lambda	the lambda sequence when fit cv.glmnet. By default, it is exp(seq(log(0.001), log(5), length.out=15)).

Details

mma first tests if the potential mediators defined by binm, contm, and catm should be treated as mediators or covariates (if none, the variable will be deleted from further analysis). All variables identified by jointm are treated as mediators. All other variables in x that are not tested are treated as covariates. Then mma does the mediation effects estimation and inference on the selected variables.

Value

Returns an mma.big object.

dm	The matrix of changes in mediator when the predictor change by a margin. Each item of the list is for a different predictor.
coef	The coefficients of mediators fitted by elastic net. Each item of the list is for a different response variable.
deltaM	a matrix of deltaM: each row is for a bootstrap sample, and each column is for a different mediator identified by the column name.
bootresults	estimated mediation effects from each bootstrap sample. Each row is for a different mediation effects, identified by the rownames, the columnes is by predictors then by the bootstrap sample order. Each item of the list is for a different response variable.
results	the med.big results using all data.
data	a list that contains all the used data: x=x, y=y, dirx=dirx, binm=binm, contm=contm, catm=catm, jointm=jointm, binpred=F.

Author(s)

Qingzhao Yu <qyu@lsuhsc.edu> and Bin Li <bli@lsu.edu>

References

Baron, R.M., and Kenny, D.A. (1986) <doi:10.1037/0022-3514.51.6.1173>. "The moderator-mediator variable distinction in social psychological research: conceptual, strategic, and statistical considerations," J. Pers Soc Psychol, 51(6), 1173-1182.

Yu, Q., Fan, Y., and Wu, X. (2014) <doi:10.4172/2155-6180.1000189>. "General Multiple Mediation Analysis With an Application to Explore Racial Disparity in Breast Cancer Survival," Journal of Biometrics & Biostatistics,5(2): 189.

Yu, Q., Scribner, R.A., Leonardi, C., Zhang, L., Park, C., Chen, L., and Simonsen, N.R. (2017) <doi:10.1016/j.sste.2017.02.001>. "Exploring racial disparity in obesity: a mediation analysis considering geo-coded environmental factors," Spatial and Spatio-temporal Epidemiology, 21, 13-23.

Yu, Q., and Li, B. (2017) <doi:10.5334/hors.160>. "mma: An r package for multiple mediation analysis," Journal of Open Research Software, 5(1), 11.

Li, B. and Yu, Q. (2018). "Mediation Analysis with Large Data Sets".

See Also

"data.org.big" is for mediator tests, and "med.big" for mediation analysis and inferences.

Examples

```
#A. binary predictor
set.seed(1)
n=100
pred=rbinom(n,1,0.5)
m1=matrix(rnorm(n*10),n,10)
m2<-matrix(rnorm(n*10,mean=pred,sd=1),n,10)
m3.1=m2[,6:10]
m3=m3.1
m2=m2[,1:5]
m3[m3.1<=0.1]=0
m3[0.1<m3.1 & m3.1<=1]=1
m3[m3.1>1]=2
m3<-apply(m3,2,as.factor)
m<-data.frame(m1,m2,m3)
colnames(m)<-c(paste("m0",1:9,sep=""),paste("m",10:20,sep=""))

lu<--0.5363+0.701*pred+0.801*m[,1]+0.518*m[,2]+1.402*m[,11]+0.773*m[,12]+
    ifelse(m[,16]=="2",2.15,0)+ifelse(m[,16]=="1",0.201,0)

#a. continuous y
y<-rnorm(n,lu,1)

data.e1<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),
                      pred=data.frame(pred),testtype=1)
summary(data.e1,only=TRUE)
data.e1.2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred))
summary(data.e1.2,only=TRUE)
med.e1<-med.big(data.e1)
med.e1
med.e1.2<-med.big(data.e1.2)
med.e1.2
mma.e1<-mma.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                alpha=1,alpha1=0.05,alpha2=0.05,n2=3)  #use only the test results.
mma.e1.2<-mma.big(data=data.e1.2,alpha1=0.05,alpha2=0.05,n2=3) #use only the test results.
summary(mma.e1)
summary(mma.e1.2,RE=TRUE)
plot(mma.e1.2,vari="m16")
plot(mma.e1.2,vari="m11")
joint.effect(mma.e1.2,vari=c("m11","m16")) #find the joint effect of m11 and m16
joint.effect(mma.e1.2,vari="m16") #find the joint effect of m16


#b. binary y
y<-rbinom(n,1,exp(lu)/(1+exp(lu)))
data.e2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),
                      pred=data.frame(pred),testtype=1)
summary(data.e2,only=TRUE)
```

```
data.e2.2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                        alpha1=0.05,alpha2=0.05)
summary(data.e2.2,only=TRUE)
med.e2<-med.big(data.e2)
med.e2
med.e2.2<-med.big(data.e2.2)
med.e2.2
mma.e2<-mma.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                alpha=1,alpha1=0.05,alpha2=0.05,n2=3)  #use only the test results.
mma.e2.2<-mma.big(data=data.e2.2,alpha1=0.05,alpha2=0.05,n2=3) #use only the test results.
summary(mma.e2)
summary(mma.e2.2,RE=TRUE)
plot(mma.e2.2,vari="m16")
plot(mma.e2.2,vari="m11")

#c.time-to-event y
lambda=1/500
survt=-log(runif(n))/lambda/exp(lu)
st=round(runif(n,1,500),0)
time=ifelse(st+survt>600,600,st+survt)-st
cen=ifelse(st+survt>600,0,1)
y=Surv(time,cen)
data.e3<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),testtype=1)
summary(data.e3,only=TRUE)
data.e3.2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                        alpha1=0.05,alpha2=0.05)
summary(data.e3.2,only=TRUE)
med.e3<-med.big(data.e3)
med.e3
med.e3.2<-med.big(data.e3.2)
med.e3.2
mma.e3<-mma.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),alpha=1,alpha1=0.05,
                alpha2=0.05,n2=3)  #use only the test results.
mma.e3.2<-mma.big(data=data.e3.2,alpha1=0.05,alpha2=0.05,n2=3) #use only the test results.
summary(mma.e3)
summary(mma.e3.2,RE=TRUE)
plot(mma.e3.2,vari="m16")
plot(mma.e3.2,vari="m11")

#B. continuous predictor
set.seed(1)
n=100
pred=runif(n,-1,1)
m1=matrix(rnorm(n*10),n,10)
m2<-matrix(rnorm(n*5,mean=0.701*pred,sd=1),n,5)
m3.1=matrix(rnorm(n*5,mean=0.518*pred,sd=1),n,5)
m3=m3.1
m3[m3.1<=0]=0
m3[0<m3.1 & m3.1<=1.28]=1
m3[m3.1>1.28]=2
m3<-apply(m3,2,as.factor)
m<-data.frame(m1,m2,m3)
colnames(m)<-c(paste("m0",1:9,sep=""),paste("m",10:20,sep=""))
```

```
lu<--0.6852+0.701*pred+0.801*m[,1]+0.518*m[,2]+1.402*m[,11]+0.773*m[,12]+
    ifelse(m[,16]=="2",2.15,0)+ifelse(m[,16]=="1",0.201,0)

#a. continuous y
y<-rnorm(n,lu,1)

data.c.e1<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                        testtype=1,alpha1=0.05,alpha2=0.05)
summary(data.c.e1,only=TRUE)
data.c.e1.2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                          alpha1=0.05,alpha2=0.05)
summary(data.c.e1.2,only=TRUE)
med.c.e1<-med.big(data.c.e1)
med.c.e1
med.c.e1.2<-med.big(data.c.e1.2)
med.c.e1.2
mma.c.e1<-mma.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),alpha1=0.05,
                  alpha2=0.05,n2=3)
mma.c.e1.2<-mma.big(data=data.c.e1.2,alpha1=0.05,alpha2=0.05,n2=3) #use only the test results.
summary(mma.c.e1)
summary(mma.c.e1.2,RE=TRUE)
plot(mma.c.e1,vari="m11")
plot(mma.c.e1,vari="m16")

#b. binary y
y<-rbinom(n,1,exp(lu)/(1+exp(lu)))
data.c.e2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                        testtype=1,alpha1=0.05,alpha2=0.05)
summary(data.c.e2,only=TRUE)
data.c.e2.2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                          alpha1=0.05,alpha2=0.05)
summary(data.c.e2.2,only=TRUE)
med.c.e2<-med.big(data.c.e2)
med.c.e2
med.c.e2.2<-med.big(data.c.e2.2)
med.c.e2.2
mma.c.e2<-mma.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),alpha1=0.05,
                  alpha2=0.05,n2=3)
mma.c.e2.2<-mma.big(data=data.c.e2.2,alpha1=0.05,alpha2=0.05,n2=3) #use only the test results.
summary(mma.c.e2)
summary(mma.c.e2.2,RE=TRUE)
plot(mma.c.e2,vari="m11")
plot(mma.c.e2,vari="m16")

#c.time-to-event y
lambda=1/500
survt=-log(runif(n))/lambda/exp(lu)
st=round(runif(n,1,500),0)
time=ifelse(st+survt>600,600,st+survt)-st
cen=ifelse(st+survt>600,0,1)
y=Surv(time,cen)
data.c.e3<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),testtype=1)
```

```
summary(data.c.e3,only=TRUE)
data.c.e3.2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                          alpha1=0.05,alpha2=0.05)
summary(data.c.e3.2,only=TRUE)
med.c.e3<-med.big(data.c.e3)
med.c.e3
med.c.e3.2<-med.big(data.c.e3.2)
med.c.e3.2
mma.c.e3<-mma.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),alpha1=0.05,
                  alpha2=0.05,n2=3)
mma.c.e3.2<-mma.big(data=data.c.e3.2,alpha1=0.05,alpha2=0.05,n2=3) #use only the test results.
summary(mma.c.e3)
summary(mma.c.e3.2,RE=TRUE,quant=FALSE)
plot(mma.c.e3,vari="m11")
plot(mma.c.e3,vari="m16")


#C. multiple predictors
#A) categorical predictor
set.seed(1)
n=100
pred=rmultinom(100,1,c(0.5, 0.3, 0.2))
pred=pred[1,]*0+pred[2,]*1+pred[3,]*2
m1=matrix(rnorm(n*10),n,10)
m2<-matrix(rnorm(n*10,mean=pred,sd=1),n,10)
m3.1=m2[,6:10]
m2=m2[,1:5]
m3=m3.1
m3[m3.1<=0.1]=0
m3[0.1<m3.1 & m3.1<=1]=1
m3[m3.1>1]=2
m3<-apply(m3,2,as.factor)
m<-data.frame(m1,m2,m3)
colnames(m)<-c(paste("m0",1:9,sep=""),paste("m",10:20,sep=""))
pred<-as.factor(pred)

lu<--0.5363+ifelse(pred=="1",0.3,0)+ifelse(pred=="2",0.7,0)+0.801*m[,1]+0.518*m[,2]+
    1.402*m[,11]+0.773*m[,12]+ifelse(m[,16]=="2",2.15,0)+ifelse(m[,16]=="1",0.201,0)

#a. continuous y
y<-rnorm(n,lu,1)

data.m.e1<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),testtype=1)
summary(data.m.e1,only=TRUE)
data.m.e1.2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred))
summary(data.m.e1.2,only=TRUE)
med.m.e1<-med.big(data.m.e1)
med.m.e1
med.m.e1.2<-med.big(data.m.e1.2)
med.m.e1.2
mma.m.e1<-mma.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),alpha=1,
                  alpha1=0.05,alpha2=0.05,n2=3)  #use only the test results.
mma.m.e1.2<-mma.big(data=data.m.e1.2,alpha1=0.05,alpha2=0.05,n2=3) #use only the test results.
```

```
summary(mma.m.e1)
summary(mma.m.e1.2,RE=TRUE)
summary(mma.m.e1.2,use.ball=TRUE)
plot(mma.m.e1,vari="m11")
plot(mma.m.e1,vari="m16")

#b. binary y
y<-rbinom(n,1,exp(lu)/(1+exp(lu)))
data.m.e2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),testtype=1)
summary(data.m.e2,only=TRUE)
data.m.e2.2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                          alpha1=0.05,alpha2=0.05)
summary(data.m.e2.2,only=TRUE)
med.m.e2<-med.big(data.m.e2)
med.m.e2
#med.m.e2.2<-med.big(data.m.e2.2)
#med.m.e2.2
#mma.m.e2<-mma.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),alpha=1,
#                  alpha1=0.05,alpha2=0.05,n2=3)  #use only the test results.
#mma.m.e2.2<-mma.big(data=data.m.e2.2,alpha1=0.05,alpha2=0.05,n2=3) #use only the test results.
#summary(mma.m.e2)
#summary(mma.m.e2.2,RE=TRUE)
#plot(mma.m.e2,vari="m11")
#plot(mma.m.e2,vari="m16")

#c.time-to-event y
lambda=1/500
survt=-log(runif(n))/lambda/exp(lu)
st=round(runif(n,1,500),0)
time=ifelse(st+survt>600,600,st+survt)-st
cen=ifelse(st+survt>600,0,1)
y=Surv(time,cen)
data.m.e3<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),testtype=1)
summary(data.m.e3,only=TRUE)
data.m.e3.2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                          alpha1=0.05,alpha2=0.05)
summary(data.m.e3.2,only=TRUE)
med.m.e3<-med.big(data.m.e3)
med.m.e3
med.m.e3.2<-med.big(data.m.e3.2)
med.m.e3.2
mma.m.e3<-mma.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),alpha=1,
                  alpha1=0.05,alpha2=0.05,n2=3)  #use only the test results.
mma.m.e3.2<-mma.big(data=data.m.e3.2,alpha1=0.05,alpha2=0.05,n2=3) #use only the test results.
summary(mma.m.e3)
summary(mma.m.e3.2,RE=TRUE)
plot(mma.m.e3,vari="m11")
plot(mma.m.e3,vari="m16")

#B) continuous predictor
set.seed(1)
n=100
pred=cbind(runif(n,-1,1),rnorm(n))
```

```
m1=matrix(rnorm(n*10),n,10)
m2<-matrix(rnorm(n*5,mean=0.3*pred[,1]+0.4*pred[,2],sd=1),n,5)
m3.1=matrix(rnorm(n*5,mean=0.7*pred[,1]+0.8*pred[,2],sd=1),n,5)
m3=m3.1
m3[m3.1<=0]=0
m3[0<m3.1 & m3.1<=1.28]=1
m3[m3.1>1.28]=2
m3<-apply(m3,2,as.factor)
m<-data.frame(m1,m2,m3)
colnames(m)<-c(paste("m0",1:9,sep=""),paste("m",10:20,sep=""))
colnames(pred)=c("x1","x2")

lu<--0.6852+0.3*pred[,1]+0.7*pred[,2]+0.801*m[,1]+0.518*m[,2]+1.402*m[,11]+0.773*m[,12]+
     ifelse(m[,16]=="2",2.15,0)+ifelse(m[,16]=="1",0.201,0)

#a. continuous y
y<-rnorm(n,lu,1)

data.m.c.e1<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                          testtype=1,alpha1=0.05,alpha2=0.05)
summary(data.m.c.e1,only=TRUE)
data.m.c.e1.2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                            alpha1=0.05,alpha2=0.05)
summary(data.m.c.e1.2,only=TRUE)
med.m.c.e1<-med.big(data.m.c.e1)
med.m.c.e1
med.m.c.e1.2<-med.big(data.m.c.e1.2)
med.m.c.e1.2
mma.m.c.e1<-mma.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                    alpha1=0.05,alpha2=0.05,n2=3)
mma.m.c.e1.2<-mma.big(data=data.c.e1.2,alpha1=0.05,alpha2=0.05,n2=3)  #use only the test results.
summary(mma.m.c.e1)
summary(mma.m.c.e1.2,RE=TRUE)
plot(mma.m.c.e1,vari="m11")
plot(mma.m.c.e1,vari="m16")

#b. binary y
y<-rbinom(n,1,exp(lu)/(1+exp(lu)))
data.m.c.e2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                          testtype=1,alpha1=0.05,alpha2=0.05)
summary(data.m.c.e2,only=TRUE)
data.m.c.e2.2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                            alpha1=0.05,alpha2=0.05)
summary(data.m.c.e2.2,only=TRUE)
med.m.c.e2<-med.big(data.m.c.e2)
med.m.c.e2
med.m.c.e2.2<-med.big(data.m.c.e2.2)
med.m.c.e2.2
pred.new=cbind(runif(n,-1,1),rnorm(n))
colnames(pred.new)=c("x1","x2")
print(med.m.c.e2.2,pred.new=pred.new)
mma.m.c.e2<-mma.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),alpha1=0.05,
                    alpha2=0.05,n2=3)
```

```
mma.m.c.e2.2<-mma.big(data=data.m.c.e2.2,alpha1=0.05,alpha2=0.05,n2=3) #use only the test results.
summary(mma.m.c.e2)
summary(mma.m.c.e2.2,RE=TRUE)
plot(mma.m.c.e2,vari="m11")
plot(mma.m.c.e2,vari="m16")


#c.time-to-event y
lambda=1/500
survt=-log(runif(n))/lambda/exp(lu)
st=round(runif(n,1,500),0)
time=ifelse(st+survt>600,600,st+survt)-st
cen=ifelse(st+survt>600,0,1)
y=Surv(time,cen)
data.m.c.e3<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),testtype=1)
summary(data.m.c.e3,only=TRUE)
data.m.c.e3.2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                            alpha1=0.05,alpha2=0.05)
summary(data.m.c.e3.2,only=TRUE)
med.m.c.e3<-med.big(data.m.c.e3)
med.m.c.e3
med.m.c.e3.2<-med.big(data.m.c.e3.2)
med.m.c.e3.2
mma.m.c.e3<-mma.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),alpha1=0.05,
                    alpha2=0.05,n2=3)
mma.m.c.e3.2<-mma.big(data=data.m.c.e3.2,alpha1=0.05,alpha2=0.05,n2=3) #use only the test results.
summary(mma.m.c.e3)
summary(mma.m.c.e3.2,RE=TRUE)
plot(mma.m.c.e3,vari="m11")
plot(mma.m.c.e3,vari="m16")


#d. multivariate y
y<-cbind(rnorm(n,lu,1),rbinom(n,1,exp(lu)/(1+exp(lu))))
colnames(y)=c("y1","y2")
data.m.m.c.e2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                            testtype=1,alpha1=0.05,alpha2=0.05)
summary(data.m.m.c.e2,only=TRUE)
data.m.m.c.e2.2<-data.org.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),
                              alpha1=0.05,alpha2=0.05)
summary(data.m.m.c.e2.2,only=TRUE)
med.m.m.c.e2<-med.big(data.m.m.c.e2)
med.m.m.c.e2
med.m.m.c.e2.2<-med.big(data.m.m.c.e2.2)
med.m.m.c.e2.2
pred.new=cbind(runif(n,-1,1),rnorm(n))
colnames(pred.new)=c("x1","x2")
print(med.m.m.c.e2.2,pred.new=pred.new)
mma.m.m.c.e2<-mma.big(x=m,y=data.frame(y),mediator=1:ncol(m),pred=data.frame(pred),alpha1=0.05,
                      alpha2=0.05,n2=3)
mma.m.m.c.e2.2<-mma.big(data=data.m.m.c.e2.2,alpha1=0.05,alpha2=0.05,n2=3)
summary(mma.m.m.c.e2)
summary(mma.m.m.c.e2.2,RE=TRUE)
```

```
plot(mma.m.m.c.e2,vari="m11")
plot(mma.m.m.c.e2,vari="m16")
mma.m.m.c.e2
mma.m.m.c.e2.2
```

`plot.mma.big` *Relative effects plot of the fitted mma.big object*

Description

Plot the marginal effect of the selected variable(mediator) on the outcome, and the marginal effect of the predictor on the selected variable.

Usage

```
## S3 method for class 'mma.big'
plot(x,vari,...)
```

Arguments

`x`	a mma.big object created initially call to mma.big.
`vari`	an indices or the name of the mediator to plot.
`...`	other arguments passed to the plot function.

Details

plot.mma.big plots on the upper panel, the boxplot of fitted coefficients of the selected mediator on each outcome from bootstrap samples. The lower panel is for each predictor, the marginal change in the mediator when the predictor changes. If the predictors are binary, show the boxplot of the mean changes in mediator when the corresponding predictor changes from 0 to 1. If the predictors are continuous, show the smoothed changes of the mean mediator with the predictor, with 95% confidence bounds.

Author(s)

Qingzhao Yu <qyu@lsuhsc.edu> and Bin Li <bli@lsu.edu>

References

Yu, Q., Fan, Y., and Wu, X. (2014) <doi:10.4172/2155-6180.1000189>. "General Multiple Mediation Analysis With an Application to Explore Racial Disparity in Breast Cancer Survival," Journal of Biometrics & Biostatistics,5(2): 189.

Yu, Q., Scribner, R.A., Leonardi, C., Zhang, L., Park, C., Chen, L., and Simonsen, N.R. (2017) <doi:10.1016/j.sste.2017.02.001>. "Exploring racial disparity in obesity: a mediation analysis considering geo-coded environmental factors," Spatial and Spatio-temporal Epidemiology, 21, 13-23.

Yu, Q., and Li, B. (2017) <doi:10.5334/hors.160>. "mma: An r package for multiple mediation analysis," Journal of Open Research Software, 5(1), 11.

Li, B. and Yu, Q. (2018). "Mediation Analysis with Large Data Sets".

See Also

"mma.big"

Examples

```
#See help(mma.big)
```

`print.med.big` *Print an med.big object*

Description

Print the estimation of mediation effects from an med.big object: from the function med.big.

Usage

```
## S3 method for class 'med.big'
print(x,...,pred.new=NULL,w.new=NULL,print=T)
```

Arguments

`x`	a med object created initially call to med.
`...`	other arguments passed to the print function.
`pred.new`	the new set of predictors.
`w.new`	the new weight with the new set of predictors.
`print`	print the results if TRUE.

Author(s)

Qingzhao Yu <qyu@lsuhsc.edu> and Bin Li <bli@lsu.edu>

References

Yu, Q., Fan, Y., and Wu, X. (2014) <doi:10.4172/2155-6180.1000189>. "General Multiple Mediation Analysis With an Application to Explore Racial Disparity in Breast Cancer Survival," Journal of Biometrics & Biostatistics,5(2): 189.

Yu, Q., Scribner, R.A., Leonardi, C., Zhang, L., Park, C., Chen, L., and Simonsen, N.R. (2017) <doi:10.1016/j.sste.2017.02.001>. "Exploring racial disparity in obesity: a mediation analysis considering geo-coded environmental factors," Spatial and Spatio-temporal Epidemiology, 21, 13-23.

Yu, Q., and Li, B. (2017) <doi:10.5334/hors.160>. "mma: An r package for multiple mediation analysis," Journal of Open Research Software, 5(1), 11.

Li, B. and Yu, Q. (2018). "Mediation Analysis with Large Data Sets".

See Also

"med.big"

Examples

```
#See help(mma.big)
```

`print.mma.big` *Print a mma.big object*

Description

Print the estimation of mediation effects from an mma.big object.

Usage

```
## S3 method for class 'mma.big'
print(x,...)
```

Arguments

`x`	a mma object created initially call to mma, boot.med.binx, or boot.med.contx.
`...`	other arguments passed to the print function.

Author(s)

Qingzhao Yu <qyu@lsuhsc.edu> and Bin Li <bli@lsu.edu>

References

Yu, Q., Fan, Y., and Wu, X. (2014) <doi:10.4172/2155-6180.1000189>. "General Multiple Mediation Analysis With an Application to Explore Racial Disparity in Breast Cancer Survival," Journal of Biometrics & Biostatistics,5(2): 189.

Yu, Q., Scribner, R.A., Leonardi, C., Zhang, L., Park, C., Chen, L., and Simonsen, N.R. (2017) <doi:10.1016/j.sste.2017.02.001>. "Exploring racial disparity in obesity: a mediation analysis considering geo-coded environmental factors," Spatial and Spatio-temporal Epidemiology, 21, 13-23.

Yu, Q., and Li, B. (2017) <doi:10.5334/hors.160>. "mma: An r package for multiple mediation analysis," Journal of Open Research Software, 5(1), 11.

Li, B. and Yu, Q. (2018). "Mediation Analysis with Large Data Sets".

See Also

"mma.big"

Examples

```
#See help(mma.big)
```

`summary.med_iden.big` *Summary method for class "med_iden.big".*

Description

Compute the estimations, standard deviations and confidence intervals of the mediation effects.

Usage

```
## S3 method for class 'med_iden.big'
summary(object,...,only=F)
## S3 method for class 'summary.med_iden.big'
print(x,...)
```

Arguments

`object`	a med_iden object created initially call to data.org.big.
`x`	a summary.med_iden.big object created initially call to summary.med_iden.big
`...`	other arguments passed to the print function.
`only`	if only=T, show test results for selected covariates and mediators only.

Details

summary.med_iden gives a list of identified mediators, covariates and the test results.

Value

The function summary.med_iden return a list of covariates and mediators as identified by a series tests.

`mediator`	variable names of the identified mediators, either siginicant in both full model and in relate to the predictor, or being a member of the pre-identified joint mediators.
`covariate`	variable names of covariates: being significant in the full model but not significantly relate with the predictor.
`tests`	statistical test results.
`results`	the original object.

Author(s)

Qingzhao Yu <qyu@lsuhsc.edu> and Bin Li <bli@lsu.edu>

References

Yu, Q., Fan, Y., and Wu, X. (2014) <doi:10.4172/2155-6180.1000189>. "General Multiple Mediation Analysis With an Application to Explore Racial Disparity in Breast Cancer Survival," Journal of Biometrics & Biostatistics,5(2): 189.

See Also

"mma.big"

Examples

```
#see help(mma.big)
```

---

summary.mma.big *Summary of an mma.big project*

Description

Compute the estimations, standard deviations and confidence intervals of the mediation effects.

Usage

```
## S3 method for class 'mma.big'
summary(object,...,alpha=0.05,plot=TRUE,RE=FALSE,quant=T,ball.use=T)
## S3 method for class 'summary.mma.big'
print(x,...,digit=3)
```

Arguments

object	a mma.big object created initially call to mma.big.
x	a summary.mma.big object created initially call to summary.mma.big.
...	other arguments passed to the print function.
alpha	the alpha level for confidence interval.
plot	default is TRUE, if ture, draw a barplot of the mediation effects with confidence intervals.
RE	default is FALSE, if ture, show the inferences on relative mediation effects.
quant	default is TRUE, if ture and ball.use is F, draw the confidence intervals of relative effects using quantile.
ball.use	default is TRUE, if ture, draw the confidence intervals of relative effects using the confidence ball. If both quant and ball.use are false, draw the confidence intervals based on the standard deviaitons from bootstrap estimates.
digit	the number of decimal digits to keep.

Details

summary.mma.big gives a list of the estimations and summary statistics based on the bootstrap results. If plot=T, draw a barplot of the estimated direct effect of the predictor and indirect effects of the mediators. Relative effects is defined as the (in)direct effect divided by the total effect. The plot is arranged in order from the largest to the smallest mediation effects.

Value

The function summary.mma.big return a list. The first item, result, is the list for mediation effects, and the second item, re, is the list for relative effects. Under them, est is the estimation of the corresponding (relative) mediation effects based on the whole data; mean is the average estimated (relative) effects from the bootstrap samples; and sd is the standard deviation of the estimates from the bootstrap sample. upbd and lwbd are the upper and lower bound of the confidence interval of the estimation using parametric method from the bootstrap sample, upbd_q and lwbd_q are the corresponding quantiles of the estimation from the bootstrap sample; upbd_b and lwbd_b are the corresponding ball confidence interval of the estimation from the bootstrap sample. If plot=T, draw a barplot of the mediation effects with confidence intervals.

Author(s)

Qingzhao Yu <qyu@lsuhsc.edu> and Bin Li <bli@lsu.edu>

References

Yu, Q., Fan, Y., and Wu, X. (2014) <doi:10.4172/2155-6180.1000189>. "General Multiple Mediation Analysis With an Application to Explore Racial Disparity in Breast Cancer Survival," Journal of Biometrics & Biostatistics,5(2): 189.

Yu, Q., Scribner, R.A., Leonardi, C., Zhang, L., Park, C., Chen, L., and Simonsen, N.R. (2017) <doi:10.1016/j.sste.2017.02.001>. "Exploring racial disparity in obesity: a mediation analysis considering geo-coded environmental factors," Spatial and Spatio-temporal Epidemiology, 21, 13-23.

Yu, Q., and Li, B. (2017) <doi:10.5334/hors.160>. "mma: An r package for multiple mediation analysis," Journal of Open Research Software, 5(1), 11.

Li, B. and Yu, Q. (2018). "Mediation Analysis with Large Data Sets".

See Also

"mma.big"

Examples

```
#See help(mma.big)
```

Index

∗ **Mediation Analysis**
med.big, 7
mma.big, 9
∗ **Mediator Tests**
data.org.big, 3
mma.big, 9
∗ **Package**
mmabig-package, 2

data.org.big, *2*, 3, *12*

joint.effect, 6

med.big, *2*, 7, *12*, *20*
mma.big, *2*, *7*, 9, *9*, *20, 21*, *23, 24*
mmabig-package, 2

plot.mma.big, 19
print.med.big, 20
print.mma.big, 21
print.summary.med_iden.big
*(*summary.med_iden.big*)*, 22
print.summary.mma.big
*(*summary.mma.big*)*, 23

summary.med_iden.big, 22
summary.mma.big, 23