Package 'cfma'

May 24, 2018

Type Package

Title Causal Functional Mediation Analysis

Version 1.0

Date 2018-05-16

Author Yi Zhao <zhaoyi1026@gmail.com>, Xi Luo <xi.rossi.luo@gmail.com>, Martin Lindquist <mal2053@gmail.com>, Brian Caffo <bcaffo@gmail.com>

Maintainer Yi Zhao <zhaoyi1026@gmail.com>

Description Performs causal functional mediation analysis (CFMA) for functional treatment, functional mediator, and functional outcome. This package includes two functional mediation model types: (1) a concurrent mediation model and (2) a historical influence mediation model. See Zhao et al. (2018), Functional Mediation Analysis with an Application to Functional Magnetic Resonance Imaging Data, <arXiv:1805.06923> for details.

License GPL (>= 2)

Depends R (>= 2.10)

NeedsCompilation no

Repository CRAN

Date/Publication 2018-05-24 12:45:18 UTC

R topics documented:

cfma-package *Causal Functional Mediation Analysis*

Description

cfma package performs causal functional mediation analysis (CFMA) for functional treatment, functional mediator, and functional outcome. This package includes two functional mediation model type: (1) a concurrent mediation model and (2) a historical influence mediation model.

Details

Package:	cfma
Type:	Package
Version:	1.0
Date:	2018-05-16
License:	GPL (>=2)

Author(s)

Yi Zhao <zhaoyi1026@gmail.com> and Xi Luo <xi.rossi.luo@gmail.com> and Martin Lindquist <mal2053@gmail.com> and Brian Caffo <bcaffo@gmail.com>

Maintainer: Yi Zhao <zhaoyi1026@gmail.com>

env.concurrent *Simulated data from the concurrent mediation model*

Description

"env.concurrent" is an R environment containing the data generated from a concurrent mediation model.

Usage

```
data("env.concurrent")
```

Format

An R environment

`Z` a $n \times T$ data matrix, treatment trajectory of n subjects for T time points.

`M` a $n \times T$ data matrix, mediator trajectory of n subjects for T time points.

`Y` a $n \times T$ data matrix, outcome trajectory of n subjects for T time points.

`alpha` a length T vector model coefficient.

`beta` a length T vector model coefficient.

`gamma` a length T vector model coefficient.

Details

The data was generated from the concurrent mediation model

$$M(t) = Z(t)\alpha(t) + \epsilon_1(t),$$

$$R(t) = Z(t)\gamma(t) + M(t)\beta(t) + \epsilon_2(t).$$

$Z(t)$ is the convolution of hemodynamic response function (HRF) and event onsets.

Examples

```
data(env.concurrent)
Z<-get("Z",env.concurrent)
M<-get("M",env.concurrent)
Y<-get("Y",env.concurrent)
```

---

env.historical *Simulated data from the historical influence mediation model*

Description

"env.historical" is an R environment containing the data generated from a historical influence mediation model.

Usage

```
data("env.historical")
```

Format

An R environment

`Z` a $n \times T$ data matrix, treatment trajectory of n subjects for T time points.

`M` a $n \times T$ data matrix, mediator trajectory of n subjects for T time points.

`Y` a $n \times T$ data matrix, outcome trajectory of n subjects for T time points.

`alpha` a $T \times T$ matrix model coefficient.

`beta` a $T \times T$ matrix model coefficient.

`gamma` a $T \times T$ matrix model coefficient.

Details

The data was generated from the historical influence mediation model

$$M(t) = \int_{\Omega_t^1} Z(s)\alpha(s,t)ds + \epsilon_1(t),$$

$$Y(t) = \int_{\Omega_t^2} Z(s)\gamma(s,t)ds + \int_{\Omega_t^3} M(s)\beta(s,t)ds + \epsilon_2(t),$$

where $\alpha(s,t)$, $\beta(s,t)$, $\gamma(s,t)$ are coefficient curves; $\Omega_t^j = [(t-\delta_j) \vee 0, t]$ for $j = 1,2,3$. $Z(t)$ is the convolution of hemodynamic response function (HRF) and event onsets.

Examples

```
data(env.historical)
Z<-get("Z",env.historical)
M<-get("M",env.historical)
Y<-get("Y",env.historical)
```

`FMA.concurrent` *Functional mediation analysis under concurrent regression model*

Description

This function performs functional mediation regression under the concurrent model with given tuning parameter.

Usage

```
FMA.concurrent(Z, M, Y, intercept = TRUE, basis = NULL, Ld2.basis = NULL,
    basis.type = c("fourier"), nbasis = 3, timeinv = c(0, 1), timegrids = NULL,
    lambda.m = 0.01, lambda.y = 0.01)
```

Arguments

`Z`	a data matrix. `Z` is the treatment trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
`M`	a data matrix. `M` is the mediator trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
`Y`	a data matrix. `Y` is the outcome trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
`intercept`	a logic variable. Default is `TRUE`, an intercept term is included in the regression model.

basis	a data matrix. Basis function used in the functional data analysis. The number of columns is the number of basis function considered. If basis = NULL, Fourier basis functions will be generated.
Ld2.basis	a data matrix. The second derivative of the basis function. The number of columns is the number of basis function considered. If Ld2.basis = NULL, the second derivative of Fourier basis functions will be generated.
basis.type	a character of basis function type. Default is Fourier basis (basis.type = "fourier").
nbasis	an integer, the number of basis function included. If basis is provided, this argument will be ignored.
timeinv	a numeric vector of length two, the time interval considered in the analysis. Default is (0,1).
timegrids	a numeric vector of time grids of measurement. If timegrids = NULL, it is assumed the between measurement time interval is constant.
lambda.m	a numeric value of the tuning parameter in the mediator model.
lambda.y	a numeric value of the tuning parameter in the outcome model.

Details

The concurrent mediation model is

$$M(t) = Z(t)\alpha(t) + \epsilon_1(t),$$

$$Y(t) = Z(t)\gamma(t) + M(t)\beta(t) + \epsilon_2(t),$$

where $\alpha(t)$, $\beta(t)$, $\gamma(t)$ are coefficient curves. The model coefficient curves are estimated by minimizing the penalized L_2-loss.

Value

basis	the basis functions used in the analysis.
M	a list of output for the mediator model coefficientthe estimated coefficient with respect to the basis function curve: the estimated coefficient curve fitted: the fitted value of M lambda: λ value
Y	a list of output for the outcome model coefficient: the estimated coefficient with respect to the basis function curve: the estimated coefficient curve fitted: the fitted value of Y lambda: λ value
IE	a list of output for the indirect effect comparing $Z_1(t) = 1$ versus $Z_0(t) = 0$ coefficients: the coefficient with respect to the basis function curve: the estimated causal curve
DE	a list of output for the direct effect comparing $Z_1(t) = 1$ versus $Z_0(t) = 0$ coefficients: the coefficient with respect to the basis function curve: the estimated causal curve

Author(s)

Yi Zhao, Johns Hopkins University, <zhaoyi1026@gmail.com>;

Xi Luo, Brown University <xi.rossi.luo@gmail.com>;

Martin Lindquist, Johns Hopkins University, <mal2053@gmail.com>;

Brian Caffo, Johns Hopkins University, <bcaffo@gmail.com>

References

Zhao et al. (2017). *Functional Mediation Analysis with an Application to Functional Magnetic Resonance Imaging Data.* arXiv preprint arXiv:1805.06923.

Examples

```
################################################
# Concurrent functional mediation model
data(env.concurrent)
Z<-get("Z",env.concurrent)
M<-get("M",env.concurrent)
Y<-get("Y",env.concurrent)

# consider Fourier basis
fit<-FMA.concurrent(Z,M,Y,intercept=FALSE,timeinv=c(0,300))

# estimate of alpha
plot(fit$M$curve[1,],type="l",lwd=5)
lines(get("alpha",env.concurrent),lty=2,lwd=2,col=2)

# estimate of gamma
plot(fit$Y$curve[1,],type="l",lwd=5)
lines(get("gamma",env.concurrent),lty=2,lwd=2,col=2)

# estimate of beta
plot(fit$Y$curve[2,],type="l",lwd=5)
lines(get("beta",env.concurrent),lty=2,lwd=2,col=2)

# estimate of causal curves
plot(fit$IE$curve,type="l",lwd=5)
plot(fit$DE$curve,type="l",lwd=5)
################################################
```

`FMA.concurrent.boot` *Functional mediation analysis under concurrent regression model with point-wise bootstrap confidence interval*

Description

This function performs functional mediation regression under the concurrent model with given tuning parameter. Point-wise confidence bands are obtained from bootstrap.

Usage

```
FMA.concurrent.boot(Z, M, Y, intercept = TRUE, basis = NULL, Ld2.basis = NULL,
    basis.type = c("fourier"), nbasis = 3, timeinv = c(0, 1), timegrids = NULL,
    lambda.m = 0.01, lambda.y = 0.01, sims = 1000, boot = TRUE,
    boot.ci.type = c("bca", "perc"), conf.level = 0.95, verbose = TRUE)
```

Arguments

Z	a data matrix. Z is the treatment trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
M	a data matrix. M is the mediator trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
Y	a data matrix. Y is the outcome trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
intercept	a logic variable. Default is TRUE, an intercept term is included in the regression model.
basis	a data matrix. Basis function used in the functional data analysis. The number of columns is the number of basis function considered. If basis = NULL, Fourier basis functions will be generated.
Ld2.basis	a data matrix. The second derivative of the basis function. The number of columns is the number of basis function considered. If Ld2.basis = NULL, the second derivative of Fourier basis functions will be generated.
basis.type	a character of basis function type. Default is Fourier basis (basis.type = "fourier").
nbasis	an integer, the number of basis function included. If basis is provided, this argument will be ignored.
timeinv	a numeric vector of length two, the time interval considered in the analysis. Default is (0,1).
timegrids	a numeric vector of time grids of measurement. If timegrids = NULL, it is assumed the between measurement time interval is constant.
lambda.m	a numeric value of the tuning parameter in the mediator model.
lambda.y	a numeric value of the tuning parameter in the outcome model.
sims	an integer indicating the number of simulations for inference.
boot	a logical value, indicating whether or not bootstrap should be used. Default is TRUE.
boot.ci.type	a character of confidence interval method. boot.ci.type = "bca" bias corrected confidence interval; boot.ci.type = "perc" percentile confidence interval.
conf.level	a number of significance level. Default is 0.95.
verbose	a logical value, indicating whether print out bootstrap replications.

Details

The concurrent mediation model is

$$M(t) = Z(t)\alpha(t) + \epsilon_1(t),$$

$$Y(t) = Z(t)\gamma(t) + M(t)\beta(t) + \epsilon_2(t),$$

where $\alpha(t)$, $\beta(t)$, $\gamma(t)$ are coefficient curves. The model coefficient curves are estimated by minimizing the penalized L_2-loss.

Value

`alpha`	a list of output for α estimate
	`coefficients`: the result of the coefficient estimates corresponding to the basis function
	`curve`: the point-wise estimate of the coefficient curve
`gamma`	: a list of output for γ estimate
	`coefficients`: the result of the coefficient estimates corresponding to the basis function
	`curve`: the point-wise estimate of the coefficient curve
`beta`	a list of output for β estimate
	`coefficients`: the result of the coefficient estimates corresponding to the basis function
	`curve`: the point-wise estimate of the coefficient curve
`IE`	a list of output for indirect effect estimate
	`coefficients`: the result of the coefficient estimates corresponding to the basis function
	`curve`: the point-wise estimate of the coefficient curve
`DE`	a list of output for direct effect estimate
	`coefficients`: the result of the coefficient estimates corresponding to the basis function
	`curve`: the point-wise estimate of the coefficient curve

Author(s)

Yi Zhao, Johns Hopkins University, <zhaoyi1026@gmail.com>;

Xi Luo, Brown University <xi.rossi.luo@gmail.com>;

Martin Lindquist, Johns Hopkins University, <mal2053@gmail.com>;

Brian Caffo, Johns Hopkins University, <bcaffo@gmail.com>

References

Zhao et al. (2017). *Functional Mediation Analysis with an Application to Functional Magnetic Resonance Imaging Data.* arXiv preprint arXiv:1805.06923.

Examples

```
################################################
# Concurrent functional mediation model
data(env.concurrent)
Z<-get("Z",env.concurrent)
M<-get("M",env.concurrent)
Y<-get("Y",env.concurrent)


# consider Fourier basis
fit.boot<-FMA.concurrent.boot(Z,M,Y,intercept=FALSE,timeinv=c(0,300))

################################################
```

`FMA.concurrent.CV` *Functional mediation analysis under concurrent regression model*

Description

This function performs functional mediation regression under the concurrent model. Tuning parameter is chosen based on cross-validation.

Usage

```
FMA.concurrent.CV(Z, M, Y, intercept = TRUE, basis = NULL, Ld2.basis = NULL,
    basis.type = c("fourier"), nbasis = 3, timeinv = c(0, 1), timegrids = NULL,
    lambda = NULL, nfolds = 5)
```

Arguments

`Z`	a data matrix. `Z` is the treatment trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
`M`	a data matrix. `M` is the mediator trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
`Y`	a data matrix. `Y` is the outcome trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
`intercept`	a logic variable. Default is `TRUE`, an intercept term is included in the regression model.
`basis`	a data matrix. Basis function used in the functional data analysis. The number of columns is the number of basis function considered. If `basis = NULL`, Fourier basis functions will be generated.

`Ld2.basis`	a data matrix. The second derivative of the basis function. The number of columns is the number of basis function considered. If `Ld2.basis = NULL`, the second derivative of Fourier basis functions will be generated.
`basis.type`	a character of basis function type. Default is Fourier basis (`basis.type = "fourier"`).
`nbasis`	an integer, the number of basis function included. If `basis` is provided, this argument will be ignored.
`timeinv`	a numeric vector of length two, the time interval considered in the analysis. Default is (0,1).
`timegrids`	a numeric vector of time grids of measurement. If `timegrids = NULL`, it is assumed the between measurement time interval is constant.
`lambda`	a numeric vector of tuning parameter values.
`nfolds`	a number gives the number of folds in cross-validation.

Details

The concurrent mediation model is

$$M(t) = Z(t)\alpha(t) + \epsilon_1(t),$$

$$Y(t) = Z(t)\gamma(t) + M(t)\beta(t) + \epsilon_2(t),$$

where $\alpha(t)$, $\beta(t)$, $\gamma(t)$ are coefficient curves. The model coefficient curves are estimated by minimizing the penalized L_2-loss. Tuning parameter λ controls the smoothness of the estimated curves, and is chosen by cross-validation.

Value

`basis`	the basis functions used in the analysis.
`M`	a list of output for the mediator model `coefficient`: the estimated coefficient with respect to the basis function `curve`: the estimated coefficient curve `fitted`: the fitted value of `M` `lambda`: the chosen λ value
`Y`	a list of output for the outcome model `coefficient`: the estimated coefficient with respect to the basis function `curve`: the estimated coefficient curve `fitted`: the fitted value of `Y` `lambda`: the chosen λ value
`IE`	a list of output for the indirect effect comparing $Z_1(t) = 1$ versus $Z_0(t) = 0$ `coefficients`: the coefficient with respect to the basis function `curve`: the estimated causal curve
`DE`	a list of output for the direct effect comparing $Z_1(t) = 1$ versus $Z_0(t) = 0$ `coefficients`: the coefficient with respect to the basis function `curve`: the estimated causal curve

Author(s)

Yi Zhao, Johns Hopkins University, <zhaoyi1026@gmail.com>;

Xi Luo, Brown University <xi.rossi.luo@gmail.com>;

Martin Lindquist, Johns Hopkins University, <mal2053@gmail.com>;

Brian Caffo, Johns Hopkins University, <bcaffo@gmail.com>

References

Zhao et al. (2017). *Functional Mediation Analysis with an Application to Functional Magnetic Resonance Imaging Data.* arXiv preprint arXiv:1805.06923.

Examples

```
################################################
# Concurrent functional mediation model
data(env.concurrent)
Z<-get("Z",env.concurrent)
M<-get("M",env.concurrent)
Y<-get("Y",env.concurrent)


# consider Fourier basis
fit<-FMA.concurrent.CV(Z,M,Y,intercept=FALSE,timeinv=c(0,300))

# estimate of alpha
plot(fit$M$curve[1,],type="l",lwd=5)
lines(get("alpha",env.concurrent),lty=2,lwd=2,col=2)

# estimate of gamma
plot(fit$Y$curve[1,],type="l",lwd=5)
lines(get("gamma",env.concurrent),lty=2,lwd=2,col=2)

# estimate of beta
plot(fit$Y$curve[2,],type="l",lwd=5)
lines(get("beta",env.concurrent),lty=2,lwd=2,col=2)

# estimate of causal curves
plot(fit$IE$curve,type="l",lwd=5)
plot(fit$DE$curve,type="l",lwd=5)

################################################
```

FMA.historical *Functional mediation analysis under historical influence model*

Description

This function performs functional mediation regression under the historical influence model with given tuning parameter.

Usage

```
FMA.historical(Z, M, Y, delta.grid1 = 1, delta.grid2 = 1, delta.grid3 = 1,
   intercept = TRUE, basis1 = NULL, Ld2.basis1 = NULL, basis2 = NULL, Ld2.basis2 = NULL,
    basis.type = c("fourier"), nbasis1 = 3, nbasis2 = 3,
    timeinv = c(0, 1), timegrids = NULL,
    lambda1.m = 0.01, lambda2.m = 0.01, lambda1.y = 0.01, lambda2.y = 0.01)
```

Arguments

`Z`	a data matrix. `Z` is the treatment trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
`M`	a data matrix. `M` is the mediator trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
`Y`	a data matrix. `Y` is the outcome trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
`delta.grid1`	a number indicates the width of treatment-mediator time interval in the mediator model.
`delta.grid2`	a number indicates the width of treatment-outcome time interval in the outcome model.
`delta.grid3`	a number indicates the width of mediator-outcome time interval in the outcome model.
`intercept`	a logic variable. Default is `TRUE`, an intercept term is included in the regression model.
`basis1`	a data matrix. Basis function on the s domain used in the functional data analysis. The number of columns is the number of basis function considered. If `basis = NULL`, Fourier basis functions will be generated.
`Ld2.basis1`	a data matrix. The second derivative of the basis function on the s domain. The number of columns is the number of basis function considered. If `Ld2.basis = NULL`, the second derivative of Fourier basis functions will be generated.
`basis2`	a data matrix. Basis function on the t domain used in the functional data analysis. The number of columns is the number of basis function considered. If `basis = NULL`, Fourier basis functions will be generated.
`Ld2.basis2`	a data matrix. The second derivative of the basis function on the t domain. The number of columns is the number of basis function considered. If `Ld2.basis = NULL`, the second derivative of Fourier basis functions will be generated.
`basis.type`	a character of basis function type. Default is Fourier basis (`basis.type = "fourier"`).

nbasis1	an integer, the number of basis function on the s domain included. If basis1 is provided, this argument will be ignored.
nbasis2	an integer, the number of basis function on the t domain included. If basis2 is provided, this argument will be ignored.
timeinv	a numeric vector of length two, the time interval considered in the analysis. Default is (0,1).
timegrids	a numeric vector of time grids of measurement. If timegrids = NULL, it is assumed the between measurement time interval is constant.
lambda1.m	a numeric vector of tuning parameter values on the s domain in the mediator model.
lambda2.m	a numeric vector of tuning parameter values on the t domain in the mediator model.
lambda1.y	a numeric vector of tuning parameter values on the s domain in the outcome model.
lambda2.y	a numeric vector of tuning parameter values on the t domain in the outcome model.

Details

The historical influence mediation model is

$$M(t) = \int_{\Omega_t^1} Z(s)\alpha(s,t)ds + \epsilon_1(t),$$

$$Y(t) = \int_{\Omega_t^2} Z(s)\gamma(s,t)ds + \int_{\Omega_t^3} M(s)\beta(s,t)ds + \epsilon_2(t),$$

where $\alpha(s,t)$, $\beta(s,t)$, $\gamma(s,t)$ are coefficient curves; $\Omega_t^j = [(t-\delta_j) \vee 0, t]$ for $j = 1,2,3$. The model coefficient curves are estimated by minimizing the penalized L_2-loss.

Value

basis1	the basis functions on the s domain used in the analysis.
basis2	the basis functions on the t domain used in the analysis.
M	a list of output for the mediator model coefficient: the estimated coefficient with respect to the basis function curve: the estimated coefficient curve fitted: the fitted value of M lambda1: the λ value on the s domain lambda2: the λ value on the t domain
Y	a list of output for the outcome model coefficient: the estimated coefficient with respect to the basis function curve: the estimated coefficient curve fitted: the fitted value of Y lambda1: the λ value on the s domain lambda2: the λ value on the t domain

IE	a list of output for the indirect effect comparing $Z_1(t) = 1$ versus $Z_0(t) = 0$ curve: the estimated causal curve
DE	a list of output for the direct effect comparing $Z_1(t) = 1$ versus $Z_0(t) = 0$ curve: the estimated causal curve

Author(s)

Yi Zhao, Johns Hopkins University, <zhaoyi1026@gmail.com>;

Xi Luo, Brown University <xi.rossi.luo@gmail.com>;

Martin Lindquist, Johns Hopkins University, <mal2053@gmail.com>;

Brian Caffo, Johns Hopkins University, <bcaffo@gmail.com>

References

Zhao et al. (2017). *Functional Mediation Analysis with an Application to Functional Magnetic Resonance Imaging Data.* arXiv preprint arXiv:1805.06923.

Examples

```
################################################
# Historical influence functional mediation model
data(env.historical)
Z<-get("Z",env.historical)
M<-get("M",env.historical)
Y<-get("Y",env.historical)

# consider Fourier basis
fit<-FMA.historical(Z,M,Y,delta.grid1=3,delta.grid2=3,delta.grid3=3,
    intercept=FALSE,timeinv=c(0,300))

# estimate of causal curves
plot(fit$IE$curve,type="l",lwd=5)
plot(fit$DE$curve,type="l",lwd=5)
################################################
```

FMA.historical.boot *Functional mediation analysis under historical influence regression model with point-wise bootstrap confidence interval*

Description

This function performs functional mediation regression under the historical influence model with given tuning parameter. Point-wise confidence bands are obtained from bootstrap.

Usage

```
FMA.historical.boot(Z, M, Y, delta.grid1 = 1, delta.grid2 = 1, delta.grid3 = 1,
   intercept = TRUE, basis1 = NULL, Ld2.basis1 = NULL, basis2 = NULL, Ld2.basis2 = NULL,
    basis.type = c("fourier"), nbasis1 = 3, nbasis2 = 3,
    timeinv = c(0, 1), timegrids = NULL,
    lambda1.m = 0.01, lambda2.m = 0.01, lambda1.y = 0.01, lambda2.y = 0.01,
    sims = 1000, boot = TRUE, boot.ci.type = c("bca", "perc"),
    conf.level = 0.95, verbose = TRUE)
```

Arguments

Z	a data matrix. Z is the treatment trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
M	a data matrix. M is the mediator trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
Y	a data matrix. Y is the outcome trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
delta.grid1	a number indicates the width of treatment-mediator time interval in the mediator model.
delta.grid2	a number indicates the width of treatment-outcome time interval in the outcome model.
delta.grid3	a number indicates the width of mediator-outcome time interval in the outcome model.
intercept	a logic variable. Default is TRUE, an intercept term is included in the regression model.
basis1	a data matrix. Basis function on the s domain used in the functional data analysis. The number of columns is the number of basis function considered. If basis = NULL, Fourier basis functions will be generated.
Ld2.basis1	a data matrix. The second derivative of the basis function on the s domain. The number of columns is the number of basis function considered. If Ld2.basis = NULL, the second derivative of Fourier basis functions will be generated.
basis2	a data matrix. Basis function on the t domain used in the functional data analysis. The number of columns is the number of basis function considered. If basis = NULL, Fourier basis functions will be generated.
Ld2.basis2	a data matrix. The second derivative of the basis function on the t domain. The number of columns is the number of basis function considered. If Ld2.basis = NULL, the second derivative of Fourier basis functions will be generated.
basis.type	a character of basis function type. Default is Fourier basis (basis.type = "fourier").
nbasis1	an integer, the number of basis function on the s domain included. If basis1 is provided, this argument will be ignored.
nbasis2	an integer, the number of basis function on the t domain included. If basis2 is provided, this argument will be ignored.

`timeinv`	a numeric vector of length two, the time interval considered in the analysis. Default is (0,1).
`timegrids`	a numeric vector of time grids of measurement. If `timegrids = NULL`, it is assumed the between measurement time interval is constant.
`lambda1.m`	a numeric vector of tuning parameter values on the s domain in the mediator model.
`lambda2.m`	a numeric vector of tuning parameter values on the t domain in the mediator model.
`lambda1.y`	a numeric vector of tuning parameter values on the s domain in the outcome model.
`lambda2.y`	a numeric vector of tuning parameter values on the t domain in the outcome model.
`sims`	an integer indicating the number of simulations for inference.
`boot`	a logical value, indicating whether or not bootstrap should be used. Default is `TRUE`.
`boot.ci.type`	a character of confidence interval method. `boot.ci.type = "bca"` bias corrected confidence interval; `boot.ci.type = "perc"` percentile confidence interval.
`conf.level`	a number of significance level. Default is 0.95.
`verbose`	a logical value, indicating whether print out bootstrap replications.

Details

The historical influence mediation model is

$$M(t) = \int_{\Omega_t^1} Z(s)\alpha(s,t)ds + \epsilon_1(t),$$

$$Y(t) = \int_{\Omega_t^2} Z(s)\gamma(s,t)ds + \int_{\Omega_t^3} M(s)\beta(s,t)ds + \epsilon_2(t),$$

where $\alpha(s,t)$, $\beta(s,t)$, $\gamma(s,t)$ are coefficient curves; $\Omega_t^j = [(t-\delta_j) \vee 0, t]$ for $j = 1,2,3$. The model coefficient curves are estimated by minimizing the penalized L_2-loss.

Value

`alpha`	a list of output for α estimate
	`coefficients`: the result of the coefficient estimates corresponding to the basis function
	`curve`: the point-wise estimate of the coefficient curve
`gamma`	a list of output for γ estimate
	`coefficients`: the result of the coefficient estimates corresponding to the basis function
	`curve`: the point-wise estimate of the coefficient curve

beta	a list of output for β estimate coefficients: the result of the coefficient estimates corresponding to the basis function curve: the point-wise estimate of the coefficient curve
IE	a list of output for indirect effect estimate curve: the point-wise estimate of the coefficient curve
DE	a list of output for direct effect estimate curve: the point-wise estimate of the coefficient curve

Author(s)

Yi Zhao, Johns Hopkins University, <zhaoyi1026@gmail.com>;

Xi Luo, Brown University <xi.rossi.luo@gmail.com>;

Martin Lindquist, Johns Hopkins University, <mal2053@gmail.com>;

Brian Caffo, Johns Hopkins University, <bcaffo@gmail.com>

References

Zhao et al. (2017). *Functional Mediation Analysis with an Application to Functional Magnetic Resonance Imaging Data.* arXiv preprint arXiv:1805.06923.

Examples

```
################################################
# Historical influence functional mediation model
data(env.historical)
Z<-get("Z",env.historical)
M<-get("M",env.historical)
Y<-get("Y",env.historical)


# consider Fourier basis
fit.boot<-FMA.historical.boot(Z,M,Y,delta.grid1=3,delta.grid2=3,delta.grid3=3,
    intercept=FALSE,timeinv=c(0,300))

################################################
```

FMA.historical.CV *Functional mediation analysis under historical influence model*

Description

This function performs functional mediation regression under the historical influence model. Tuning parameter is chosen based on cross-validation.

Usage

```
FMA.historical.CV(Z, M, Y, delta.grid1 = 1, delta.grid2 = 1, delta.grid3 = 1,
   intercept = TRUE, basis1 = NULL, Ld2.basis1 = NULL, basis2 = NULL, Ld2.basis2 = NULL,
    basis.type = c("fourier"), nbasis1 = 3, nbasis2 = 3,
   timeinv = c(0, 1), timegrids = NULL, lambda1 = NULL, lambda2 = NULL, nfolds = 5)
```

Arguments

Z	a data matrix. Z is the treatment trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
M	a data matrix. M is the mediator trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
Y	a data matrix. Y is the outcome trajectory in the mediation analysis. The number of rows is the number of subjects, and the number of columns is the number of measured time points.
delta.grid1	a number indicates the width of treatment-mediator time interval in the mediator model.
delta.grid2	a number indicates the width of treatment-outcome time interval in the outcome model.
delta.grid3	a number indicates the width of mediator-outcome time interval in the outcome model.
intercept	a logic variable. Default is TRUE, an intercept term is included in the regression model.
basis1	a data matrix. Basis function on the s domain used in the functional data analysis. The number of columns is the number of basis function considered. If basis = NULL, Fourier basis functions will be generated.
Ld2.basis1	a data matrix. The second derivative of the basis function on the s domain. The number of columns is the number of basis function considered. If Ld2.basis = NULL, the second derivative of Fourier basis functions will be generated.
basis2	a data matrix. Basis function on the t domain used in the functional data analysis. The number of columns is the number of basis function considered. If basis = NULL, Fourier basis functions will be generated.
Ld2.basis2	a data matrix. The second derivative of the basis function on the t domain. The number of columns is the number of basis function considered. If Ld2.basis = NULL, the second derivative of Fourier basis functions will be generated.
basis.type	a character of basis function type. Default is Fourier basis (basis.type = "fourier").
nbasis1	an integer, the number of basis function on the s domain included. If basis1 is provided, this argument will be ignored.
nbasis2	an integer, the number of basis function on the t domain included. If basis2 is provided, this argument will be ignored.
timeinv	a numeric vector of length two, the time interval considered in the analysis. Default is (0,1).

timegrids	a numeric vector of time grids of measurement. If timegrids = NULL, it is assumed the between measurement time interval is constant.
lambda1	a numeric vector of tuning parameter values on the s domain.
lambda2	a numeric vector of tuning parameter values on the t domain.
nfolds	a number gives the number of folds in cross-validation.

Details

The historical influence mediation model is

$$M(t) = \int_{\Omega_t^1} Z(s)\alpha(s,t)ds + \epsilon_1(t),$$

$$Y(t) = \int_{\Omega_t^2} Z(s)\gamma(s,t)ds + \int_{\Omega_t^3} M(s)\beta(s,t)ds + \epsilon_2(t),$$

where $\alpha(s,t)$, $\beta(s,t)$, $\gamma(s,t)$ are coefficient curves; $\Omega_t^j = [(t-\delta_j) \vee 0, t]$ for $j = 1,2,3$. The model coefficient curves are estimated by minimizing the penalized L_2-loss. Tuning parameter λ controls the smoothness of the estimated curves, and is chosen by cross-validation.

Value

basis1	the basis functions on the s domain used in the analysis.
basis2	the basis functions on the t domain used in the analysis.
M	a list of output for the mediator model coefficient: the estimated coefficient with respect to the basis function curve: the estimated coefficient curve fitted: the fitted value of M lambda1: the chosen λ value on the s domain lambda2: the chosen λ value on the t domain
Y	a list of output for the outcome model coefficient: the estimated coefficient with respect to the basis function curve: the estimated coefficient curve fitted: the fitted value of Y lambda1: the chosen λ value on the s domain lambda2: the chosen λ value on the t domain
IE	a list of output for the indirect effect comparing $Z_1(t) = 1$ versus $Z_0(t) = 0$ curve: the estimated causal curve
DE	a list of output for the direct effect comparing $Z_1(t) = 1$ versus $Z_0(t) = 0$ curve: the estimated causal curve

Author(s)

Yi Zhao, Johns Hopkins University, <zhaoyi1026@gmail.com>;

Xi Luo, Brown University <xi.rossi.luo@gmail.com>;

Martin Lindquist, Johns Hopkins University, <mal2053@gmail.com>;

Brian Caffo, Johns Hopkins University, <bcaffo@gmail.com>

References

Zhao et al. (2017). *Functional Mediation Analysis with an Application to Functional Magnetic Resonance Imaging Data.* arXiv preprint arXiv:1805.06923.

Examples

```
##################################################
# Historical influence functional mediation model
data(env.historical)
Z<-get("Z",env.historical)
M<-get("M",env.historical)
Y<-get("Y",env.historical)


# consider Fourier basis
fit<-FMA.historical.CV(Z,M,Y,delta.grid1=3,delta.grid2=3,delta.grid3=3,
    intercept=FALSE,timeinv=c(0,300))

##################################################
```

Index