Package 'cTMed'

December 15, 2024
Title Continuous Time Mediation
Version 1.0.2.9000
Description Calculates standard errors and confidence intervals for effects in continuous-time mediation models. This package extends the work of Deboeck and Preacher (2015) <doi:10.1080 10705511.2014.973960=""> and Ryan and Hamaker (2021) <doi:10.1007 s11336-021-09767-0=""> by providing methods to generate standard errors and confidence intervals for the total, direct, and indirect effects in these models.</doi:10.1007></doi:10.1080>
<pre>URL https://github.com/jeksterslab/cTMed,</pre>
https://jeksterslab.github.io/cTMed/
BugReports https://github.com/jeksterslab/cTMed/issues
License GPL (>= 3)
Encoding UTF-8
Roxygen list(markdown = TRUE)
Depends R (>= $4.2.0$)
LinkingTo Rcpp, RcppArmadillo
Imports Rcpp, numDeriv, parallel, ctsem, simStateSpace
Suggests knitr, rmarkdown, testthat, expm
RoxygenNote 7.3.2
NeedsCompilation yes
Author Ivan Jacob Agaloos Pesigan [aut, cre, cph] (https://orcid.org/0000-0003-4818-8420)
Maintainer Ivan Jacob Agaloos Pesigan <r.jeksterslab@gmail.com></r.jeksterslab@gmail.com>
Contents
confint.ctmeddelta

2 Contents

DeltaIndirectCentral	14
DeltaMed	17
DeltaMedStd	20
DeltaTotalCentral	25
Direct	28
DirectStd	30
ExpCov	33
ExpMean	34
Indirect	36
IndirectCentral	39
IndirectStd	41
MCBeta	43
MCBetaStd	46
MCIndirectCentral	50
MCMed	
MCMedStd	57
MCPhi	62
MCTotalCentral	64
Med	
MedStd	
plot.ctmeddelta	72
plot.ctmedmc	74
plot.ctmedmed	
plot.ctmedtraj	
PosteriorBeta	
PosteriorIndirectCentral	
PosteriorMed	
PosteriorPhi	
PosteriorTotalCentral	
print.ctmeddelta	
print.ctmedeffect	
print.ctmedmc	
print.ctmedmcphi	
print.ctmedmed	
print.ctmedtraj	
summary.ctmeddelta	
summary.ctmedmc	
summary.ctmedmed	
summary.ctmedposteriorphi	
summary.ctmedtraj	
Total	
TotalCentral	
TotalStd	
Trajectory	
<u> </u>	110
	112

Index

confint.ctmeddelta 3

confint.ctmeddelta

Delta Method Confidence Intervals

Description

Delta Method Confidence Intervals

Usage

```
## S3 method for class 'ctmeddelta'
confint(object, parm = NULL, level = 0.95, ...)
```

Arguments

object Object of class ctmeddelta.

parm a specification of which parameters are to be given confidence intervals, either

a vector of numbers or a vector of names. If missing, all parameters are consid-

ered.

level the confidence level required.

... additional arguments.

Value

Returns a data frame of confidence intervals.

Author(s)

Ivan Jacob Agaloos Pesigan

```
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
vcov_phi_vec <- matrix(</pre>
  data = c(
    0.00843, 0.00040, -0.00151,
    -0.00600, -0.00033, 0.00110,
    0.00324, 0.00020, -0.00061,
    0.00040, 0.00374, 0.00016,
    -0.00022, -0.00273, -0.00016,
    0.00009, 0.00150, 0.00012,
```

4 confint.ctmedmc

```
-0.00151, 0.00016, 0.00389,
   0.00103, -0.00007, -0.00283,
   -0.00050, 0.00000, 0.00156,
   -0.00600, -0.00022, 0.00103,
   0.00644, 0.00031, -0.00119,
   -0.00374, -0.00021, 0.00070,
   -0.00033, -0.00273, -0.00007,
   0.00031, 0.00287, 0.00013,
   -0.00014, -0.00170, -0.00012,
   0.00110, -0.00016, -0.00283,
   -0.00119, 0.00013, 0.00297,
   0.00063, -0.00004, -0.00177,
   0.00324, 0.00009, -0.00050,
   -0.00374, -0.00014, 0.00063,
   0.00495, 0.00024, -0.00093,
   0.00020, 0.00150, 0.00000,
   -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
   -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
   -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
# Specific time interval ------
delta <- DeltaMed(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1,
 from = "x",
 to = "y",
 med = "m"
confint(delta, level = 0.95)
# Range of time intervals ------
delta <- DeltaMed(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1:5,
 from = "x",
 to = "y",
 med = "m"
confint(delta, level = 0.95)
```

confint.ctmedmc 5

Description

Monte Carlo Method Confidence Intervals

Usage

```
## S3 method for class 'ctmedmc'
confint(object, parm = NULL, level = 0.95, ...)
```

Arguments

object Object of class ctmedmc.

parm a specification of which parameters are to be given confidence intervals, either

a vector of numbers or a vector of names. If missing, all parameters are consid-

ered.

level the confidence level required.

... additional arguments.

Value

Returns a data frame of confidence intervals.

Author(s)

Ivan Jacob Agaloos Pesigan

```
set.seed(42)
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
vcov_phi_vec <- matrix(</pre>
  data = c(
    0.00843, 0.00040, -0.00151,
    -0.00600, -0.00033, 0.00110,
    0.00324, 0.00020, -0.00061,
    0.00040, 0.00374, 0.00016,
    -0.00022, -0.00273, -0.00016,
    0.00009, 0.00150, 0.00012,
    -0.00151, 0.00016, 0.00389,
    0.00103, -0.00007, -0.00283,
    -0.00050, 0.00000, 0.00156,
    -0.00600, -0.00022, 0.00103,
    0.00644, 0.00031, -0.00119,
```

```
-0.00374, -0.00021, 0.00070,
   -0.00033, -0.00273, -0.00007,
   0.00031, 0.00287, 0.00013,
   -0.00014, -0.00170, -0.00012,
   0.00110, -0.00016, -0.00283,
   -0.00119, 0.00013, 0.00297,
   0.00063, -0.00004, -0.00177,
   0.00324, 0.00009, -0.00050,
   -0.00374, -0.00014, 0.00063,
   0.00495, 0.00024, -0.00093,
   0.00020, 0.00150, 0.00000,
   -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
   -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
   -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
# Specific time interval ------
mc <- MCMed(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1,
 from = "x",
 to = "y",
 med = "m",
 R = 100L # use a large value for R in actual research
)
confint(mc, level = 0.95)
# Range of time intervals ------
mc <- MCMed(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1:5,
 from = "x",
 to = "y",
 med = "m"
 R = 100L # use a large value for R in actual research
confint(mc, level = 0.95)
```

DeltaBeta

Delta Method Sampling Variance-Covariance Matrix for the Elements of the Matrix of Lagged Coefficients Over a Specific Time Interval or a Range of Time Intervals

Description

This function computes the delta method sampling variance-covariance matrix for the elements of the matrix of lagged coefficients β over a specific time interval Δt or a range of time intervals using the first-order stochastic differential equation model's drift matrix Φ .

Usage

DeltaBeta(phi, vcov_phi_vec, delta_t, ncores = NULL, tol = 0.01)

Arguments

Numeric matrix. The drift matrix (Φ) , phi should have row and column names pertaining to the variables in the system.

vcov_phi_vec Numeric matrix. The sampling variance-covariance matrix of vec (Φ) .

delta_t Vector of positive numbers. Time interval (Δt) .

Positive integer. Number of cores to use. If ncores = NULL, use a single core. Consider using multiple cores when the length of delta_t is long.

tol Numeric. Smallest possible time interval to allow.

Details

See Total().

Delta Method:

Let θ be $\operatorname{vec}(\Phi)$, that is, the elements of the Φ matrix in vector form sorted column-wise. Let $\hat{\theta}$ be $\operatorname{vec}(\hat{\Phi})$. By the multivariate central limit theory, the function \mathbf{g} using $\hat{\theta}$ as input can be expressed as:

$$\sqrt{n}\left(\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right) - \mathbf{g}\left(\boldsymbol{\theta}\right)\right) \xrightarrow{\mathrm{D}} \mathcal{N}\left(0, \mathbf{J}\boldsymbol{\Gamma}\mathbf{J}'\right)$$

where **J** is the matrix of first-order derivatives of the function **g** with respect to the elements of θ and Γ is the asymptotic variance-covariance matrix of $\hat{\theta}$.

From the former, we can derive the distribution of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ as follows:

$$\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right) \approx \mathcal{N}\left(\mathbf{g}\left(\boldsymbol{\theta}\right), n^{-1}\mathbf{J}\boldsymbol{\Gamma}\mathbf{J}'\right)$$

The uncertainty associated with the estimator $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ is, therefore, given by $n^{-1}\mathbf{J}\Gamma\mathbf{J}'$. When Γ is unknown, by substitution, we can use the estimated sampling variance-covariance matrix of $\hat{\boldsymbol{\theta}}$, that is, $\hat{\mathbb{V}}\left(\hat{\boldsymbol{\theta}}\right)$ for $n^{-1}\Gamma$. Therefore, the sampling variance-covariance matrix of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ is given by

$$\mathbf{g}\left(\hat{oldsymbol{ heta}}
ight)pprox\mathcal{N}\left(\mathbf{g}\left(oldsymbol{ heta}
ight),\mathbf{J}\hat{\mathbb{V}}\left(\hat{oldsymbol{ heta}}
ight)\mathbf{J}'
ight).$$

Value

Returns an object of class ctmeddelta which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("DeltaBeta").

output A list the length of which is equal to the length of delta_t.

Each element in the output list has the following elements:

delta_t Time interval.

jacobian Jacobian matrix.

est Estimated elements of the matrix of lagged coefficients.

vcov Sampling variance-covariance matrix of estimated elements of the matrix of lagged coefficients.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

```
Other Continuous Time Mediation Functions: DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), IndirectStd(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()
```

```
phi <- matrix(
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
),
  nrow = 3</pre>
```

```
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
vcov_phi_vec <- matrix(</pre>
 data = c(
   0.00843, 0.00040, -0.00151,
   -0.00600, -0.00033, 0.00110,
   0.00324, 0.00020, -0.00061,
   0.00040, 0.00374, 0.00016,
   -0.00022, -0.00273, -0.00016,
   0.00009, 0.00150, 0.00012,
   -0.00151, 0.00016, 0.00389,
   0.00103, -0.00007, -0.00283,
   -0.00050, 0.00000, 0.00156,
   -0.00600, -0.00022, 0.00103,
   0.00644, 0.00031, -0.00119,
   -0.00374, -0.00021, 0.00070,
   -0.00033, -0.00273, -0.00007,
   0.00031, 0.00287, 0.00013,
   -0.00014, -0.00170, -0.00012,
   0.00110, -0.00016, -0.00283,
   -0.00119, 0.00013, 0.00297,
   0.00063, -0.00004, -0.00177,
   0.00324, 0.00009, -0.00050,
   -0.00374, -0.00014, 0.00063,
   0.00495, 0.00024, -0.00093,
   0.00020, 0.00150, 0.00000,
   -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
   -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
   -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
# Specific time interval ------
DeltaBeta(
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1
# Range of time intervals -----
delta <- DeltaBeta(
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1:5
)
plot(delta)
# Methods ------
# DeltaBeta has a number of methods including
# print, summary, confint, and plot
```

```
print(delta)
summary(delta)
confint(delta, level = 0.95)
plot(delta)
```

DeltaBetaStd

Delta Method Sampling Variance-Covariance Matrix for the Elements of the Standardized Matrix of Lagged Coefficients Over a Specific Time Interval or a Range of Time Intervals

Description

This function computes the delta method sampling variance-covariance matrix for the elements of the standardized matrix of lagged coefficients β over a specific time interval Δt or a range of time intervals using the first-order stochastic differential equation model's drift matrix Φ and process noise covariance matrix Σ .

Usage

```
DeltaBetaStd(phi, sigma, vcov_theta, delta_t, ncores = NULL, tol = 0.01)
```

Arguments

phi	Numeric matrix. The drift matrix (Φ). phi should have row and column names pertaining to the variables in the system.
sigma	Numeric matrix. The process noise covariance matrix (Σ) .
vcov_theta	Numeric matrix. The sampling variance-covariance matrix of $\operatorname{vec}\left(\Phi\right)$ and $\operatorname{vech}\left(\Sigma\right)$
delta_t	Numeric. Time interval (Δt).
ncores	Positive integer. Number of cores to use. If ncores = NULL, use a single core. Consider using multiple cores when number of replications R is a large value.
tol	Numeric. Smallest possible time interval to allow.

Details

See TotalStd().

Delta Method:

Let θ be a vector that combines $\operatorname{vec}(\Phi)$, that is, the elements of the Φ matrix in vector form sorted column-wise and $\operatorname{vech}(\Sigma)$, that is, the unique elements of the Σ matrix in vector form sorted column-wise. Let $\hat{\theta}$ be a vector that combines $\operatorname{vec}(\hat{\Phi})$ and $\operatorname{vech}(\hat{\Sigma})$. By the multivariate central limit theory, the function \mathbf{g} using $\hat{\theta}$ as input can be expressed as:

$$\sqrt{n}\left(\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right) - \mathbf{g}\left(\boldsymbol{\theta}\right)\right) \xrightarrow{\mathrm{D}} \mathcal{N}\left(0, \mathbf{J}\boldsymbol{\Gamma}\mathbf{J}'\right)$$

where **J** is the matrix of first-order derivatives of the function **g** with respect to the elements of θ and Γ is the asymptotic variance-covariance matrix of $\hat{\theta}$.

From the former, we can derive the distribution of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ as follows:

$$\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right) \approx \mathcal{N}\left(\mathbf{g}\left(\boldsymbol{\theta}\right), n^{-1}\mathbf{J}\boldsymbol{\Gamma}\mathbf{J}'\right)$$

The uncertainty associated with the estimator $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ is, therefore, given by $n^{-1}\mathbf{J}\Gamma\mathbf{J}'$. When Γ is unknown, by substitution, we can use the estimated sampling variance-covariance matrix of $\hat{\boldsymbol{\theta}}$, that is, $\hat{\mathbb{V}}\left(\hat{\boldsymbol{\theta}}\right)$ for $n^{-1}\Gamma$. Therefore, the sampling variance-covariance matrix of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ is given by

$$\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right) \approx \mathcal{N}\left(\mathbf{g}\left(\boldsymbol{\theta}\right), \mathbf{J}\hat{\mathbb{V}}\left(\hat{\boldsymbol{\theta}}\right) \mathbf{J}'\right).$$

Value

Returns an object of class ctmeddelta which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("DeltaBetaStd").

output A list the length of which is equal to the length of delta_t.

Each element in the output list has the following elements:

delta_t Time interval.

jacobian Jacobian matrix.

est Estimated elements of the matrix of lagged coefficients.

vcov Sampling variance-covariance matrix of estimated elements of the matrix of lagged coefficients.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), IndirectStd(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ),
 nrow = 3
)
colnames(phi) \leftarrow rownames(phi) \leftarrow c("x", "m", "y")
sigma <- matrix(</pre>
 data = c(
   0.24455556, 0.02201587, -0.05004762,
    0.02201587, 0.07067800, 0.01539456,
    -0.05004762, 0.01539456, 0.07553061
 ),
 nrow = 3
vcov_theta <- matrix(</pre>
 data = c(
    0.00843, 0.00040, -0.00151, -0.00600, -0.00033,
    0.00110, 0.00324, 0.00020, -0.00061, -0.00115,
    0.00011, 0.00015, 0.00001, -0.00002, -0.00001,
    0.00040, 0.00374, 0.00016, -0.00022, -0.00273,
    -0.00016, 0.00009, 0.00150, 0.00012, -0.00010,
    -0.00026, 0.00002, 0.00012, 0.00004, -0.00001,
    -0.00151, 0.00016, 0.00389, 0.00103, -0.00007,
    -0.00283, -0.00050, 0.00000, 0.00156, 0.00021,
    -0.00005, -0.00031, 0.00001, 0.00007, 0.00006,
    -0.00600, -0.00022, 0.00103, 0.00644, 0.00031,
    -0.00119, -0.00374, -0.00021, 0.00070, 0.00064,
    -0.00015, -0.00005, 0.00000, 0.00003, -0.00001,
    -0.00033, -0.00273, -0.00007, 0.00031, 0.00287,
    0.00013, -0.00014, -0.00170, -0.00012, 0.00006,
     0.00014, \ -0.00001, \ -0.00015, \ 0.00000, \ 0.00001, \\
    0.00110, -0.00016, -0.00283, -0.00119, 0.00013,
    0.00297, 0.00063, -0.00004, -0.00177, -0.00013,
    0.00005, 0.00017, -0.00002, -0.00008, 0.00001,
    0.00324, 0.00009, -0.00050, -0.00374, -0.00014,
    0.00063, 0.00495, 0.00024, -0.00093, -0.00020,
    0.00006, -0.00010, 0.00000, -0.00001, 0.00004,
    0.00020, 0.00150, 0.00000, -0.00021, -0.00170,
    -0.00004, 0.00024, 0.00214, 0.00012, -0.00002,
```

```
-0.00004, 0.00000, 0.00006, -0.00005, -0.00001,
   -0.00061, 0.00012, 0.00156, 0.00070, -0.00012,
   -0.00177, -0.00093, 0.00012, 0.00223, 0.00004,
   -0.00002, -0.00003, 0.00001, 0.00003, -0.00013,
   -0.00115, -0.00010, 0.00021, 0.00064, 0.00006,
   -0.00013, -0.00020, -0.00002, 0.00004, 0.00057,
   0.00001, -0.00009, 0.00000, 0.00000, 0.00001,
   0.00011, -0.00026, -0.00005, -0.00015, 0.00014,
   0.00005, 0.00006, -0.00004, -0.00002, 0.00001,
   0.00012, 0.00001, 0.00000, -0.00002, 0.00000,
   0.00015, 0.00002, -0.00031, -0.00005, -0.00001,
   0.00017, -0.00010, 0.00000, -0.00003, -0.00009,
   0.00001, 0.00014, 0.00000, 0.00000, -0.00005,
   0.00001, 0.00012, 0.00001, 0.00000, -0.00015,
   -0.00002, 0.00000, 0.00006, 0.00001, 0.00000,
   0.00000, 0.00000, 0.00010, 0.00001, 0.00000,
   -0.00002, 0.00004, 0.00007, 0.00003, 0.00000,
   -0.00008, -0.00001, -0.00005, 0.00003, 0.00000,
   -0.00002, 0.00000, 0.00001, 0.00005, 0.00001,
   -0.00001, -0.00001, 0.00006, -0.00001, 0.00001,
   0.00001, 0.00004, -0.00001, -0.00013, 0.00001,
   0.00000, -0.00005, 0.00000, 0.00001, 0.00012
 ),
 nrow = 15
)
# Specific time interval -------
DeltaBetaStd(
 phi = phi,
 sigma = sigma,
 vcov_theta = vcov_theta,
 delta_t = 1
# Range of time intervals ------
delta <- DeltaBetaStd(</pre>
 phi = phi,
 sigma = sigma,
 vcov_theta = vcov_theta,
 delta_t = 1:5
plot(delta)
# Methods ------
# DeltaBetaStd has a number of methods including
# print, summary, confint, and plot
print(delta)
summary(delta)
confint(delta, level = 0.95)
plot(delta)
```

14 DeltaIndirectCentral

DeltaIndirectCentral Delta Method Sampling Variance-Covariance Matrix for the Indirect Effect Centrality Over a Specific Time Interval or a Range of Time Intervals

Description

This function computes the delta method sampling variance-covariance matrix for the indirect effect centrality over a specific time interval Δt or a range of time intervals using the first-order stochastic differential equation model's drift matrix Φ .

Usage

DeltaIndirectCentral(phi, vcov_phi_vec, delta_t, ncores = NULL, tol = 0.01)

Arguments

phi	Numeric matrix. The drift matrix (Φ). phi should have row and column names pertaining to the variables in the system.
vcov_phi_vec	Numeric matrix. The sampling variance-covariance matrix of $\operatorname{vec}\left(\Phi\right)$.
delta_t	Vector of positive numbers. Time interval (Δt) .
ncores	Positive integer. Number of cores to use. If ncores = NULL, use a single core. Consider using multiple cores when the length of delta_t is long.
tol	Numeric. Smallest possible time interval to allow.

Details

See IndirectCentral() more details.

Delta Method:

Let θ be $\operatorname{vec}(\Phi)$, that is, the elements of the Φ matrix in vector form sorted column-wise. Let $\hat{\theta}$ be $\operatorname{vec}(\hat{\Phi})$. By the multivariate central limit theory, the function g using $\hat{\theta}$ as input can be expressed as:

$$\sqrt{n}\left(\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right) - \mathbf{g}\left(\boldsymbol{\theta}\right)\right) \xrightarrow{\mathrm{D}} \mathcal{N}\left(0, \mathbf{J}\boldsymbol{\Gamma}\mathbf{J}'\right)$$

where **J** is the matrix of first-order derivatives of the function **g** with respect to the elements of θ and Γ is the asymptotic variance-covariance matrix of $\hat{\theta}$.

From the former, we can derive the distribution of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ as follows:

$$\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right) \approx \mathcal{N}\left(\mathbf{g}\left(\boldsymbol{\theta}\right), n^{-1}\mathbf{J}\boldsymbol{\Gamma}\mathbf{J}'\right)$$

The uncertainty associated with the estimator $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ is, therefore, given by $n^{-1}\mathbf{J}\Gamma\mathbf{J}'$. When Γ is unknown, by substitution, we can use the estimated sampling variance-covariance matrix of $\hat{\boldsymbol{\theta}}$,

DeltaIndirectCentral 15

that is, $\hat{\mathbb{V}}\left(\hat{\boldsymbol{\theta}}\right)$ for $n^{-1}\Gamma$. Therefore, the sampling variance-covariance matrix of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ is given by

$$\mathbf{g}\left(\hat{oldsymbol{ heta}}
ight)pprox\mathcal{N}\left(\mathbf{g}\left(oldsymbol{ heta}
ight),\mathbf{J}\hat{\mathbb{V}}\left(\hat{oldsymbol{ heta}}
ight)\mathbf{J}'
ight).$$

Value

Returns an object of class ctmeddelta which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("DeltaIndirectCentral").

output A list the length of which is equal to the length of delta_t.

Each element in the output list has the following elements:

delta_t Time interval.

jacobian Jacobian matrix.

est Estimated indirect effect centrality.

vcov Sampling variance-covariance matrix of estimated indirect effect centrality.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), IndirectStd(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

16 DeltaIndirectCentral

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
vcov_phi_vec <- matrix(</pre>
 data = c(
   0.002704274, -0.001475275, 0.000949122,
   -0.001619422, 0.000885122, -0.000569404,
   0.00085493, -0.000465824, 0.000297815,
   -0.001475275, 0.004428442, -0.002642303,
   0.000980573, -0.00271817, 0.001618805,
   -0.000586921, 0.001478421, -0.000871547,
   0.000949122, -0.002642303, 0.006402668,
   -0.000697798, 0.001813471, -0.004043138,
   0.000463086, -0.001120949, 0.002271711,
   -0.001619422, 0.000980573, -0.000697798,
   0.002079286, -0.001152501, 0.000753,
   -0.001528701, 0.000820587, -0.000517524,
   0.000885122, -0.00271817, 0.001813471,
   -0.001152501, 0.00342605, -0.002075005,
   0.000899165, -0.002532849, 0.001475579,
   -0.000569404, 0.001618805, -0.004043138,
   0.000753, -0.002075005, 0.004984032,
   -0.000622255, 0.001634917, -0.003705661,
    \hbox{\tt 0.00085493, -0.000586921, 0.000463086,} \\
   -0.001528701, 0.000899165, -0.000622255,
   0.002060076, -0.001096684, 0.000686386,
   -0.000465824, 0.001478421, -0.001120949,
   0.000820587, -0.002532849, 0.001634917,
   -0.001096684, 0.003328692, -0.001926088,
   0.000297815, -0.000871547, 0.002271711,
   -0.000517524, 0.001475579, -0.003705661,
   0.000686386, -0.001926088, 0.004726235
 ),
 nrow = 9
)
# Specific time interval ------
DeltaIndirectCentral(
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1
# Range of time intervals -------
delta <- DeltaIndirectCentral(</pre>
```

DeltaMed 17

DeltaMed

Delta Method Sampling Variance-Covariance Matrix for the Total, Direct, and Indirect Effects of X on Y Through M Over a Specific Time Interval or a Range of Time Intervals

Description

This function computes the delta method sampling variance-covariance matrix for the total, direct, and indirect effects of the independent variable X on the dependent variable Y through mediator variables \mathbf{m} over a specific time interval Δt or a range of time intervals using the first-order stochastic differential equation model's drift matrix $\mathbf{\Phi}$.

Usage

```
DeltaMed(phi, vcov_phi_vec, delta_t, from, to, med, ncores = NULL, tol = 0.01)
```

Arguments

phi	Numeric matrix. The drift matrix (Φ). phi should have row and column names pertaining to the variables in the system.
vcov_phi_vec	Numeric matrix. The sampling variance-covariance matrix of $\operatorname{vec}\left(\mathbf{\Phi}\right)$.
delta_t	Vector of positive numbers. Time interval (Δt) .
from	Character string. Name of the independent variable X in phi.
to	Character string. Name of the dependent variable Y in phi.
med	Character vector. Name/s of the mediator variable/s in phi.
ncores	Positive integer. Number of cores to use. If ncores = NULL, use a single core. Consider using multiple cores when the length of delta_t is long.
tol	Numeric. Smallest possible time interval to allow.

DeltaMed

Details

See Total(), Direct(), and Indirect() for more details.

Delta Method:

Let θ be $\operatorname{vec}(\Phi)$, that is, the elements of the Φ matrix in vector form sorted column-wise. Let $\hat{\theta}$ be $\operatorname{vec}(\hat{\Phi})$. By the multivariate central limit theory, the function \mathbf{g} using $\hat{\theta}$ as input can be expressed as:

$$\sqrt{n}\left(\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right) - \mathbf{g}\left(\boldsymbol{\theta}\right)\right) \xrightarrow{\mathrm{D}} \mathcal{N}\left(0, \mathbf{J}\boldsymbol{\Gamma}\mathbf{J}'\right)$$

where **J** is the matrix of first-order derivatives of the function **g** with respect to the elements of θ and Γ is the asymptotic variance-covariance matrix of $\hat{\theta}$.

From the former, we can derive the distribution of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ as follows:

$$\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right) \approx \mathcal{N}\left(\mathbf{g}\left(\boldsymbol{\theta}\right), n^{-1}\mathbf{J}\mathbf{\Gamma}\mathbf{J}'\right)$$

The uncertainty associated with the estimator $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ is, therefore, given by $n^{-1}\mathbf{J}\Gamma\mathbf{J}'$. When Γ is unknown, by substitution, we can use the estimated sampling variance-covariance matrix of $\hat{\boldsymbol{\theta}}$, that is, $\hat{\mathbb{V}}\left(\hat{\boldsymbol{\theta}}\right)$ for $n^{-1}\Gamma$. Therefore, the sampling variance-covariance matrix of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ is given by

$$\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right) \approx \mathcal{N}\left(\mathbf{g}\left(\boldsymbol{\theta}\right), \mathbf{J}\hat{\mathbb{V}}\left(\hat{\boldsymbol{\theta}}\right) \mathbf{J}'\right).$$

Value

Returns an object of class ctmeddelta which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("DeltaMed").

output A list the length of which is equal to the length of delta_t.

Each element in the output list has the following elements:

delta_t Time interval.

jacobian Jacobian matrix.

est Estimated total, direct, and indirect effects.

vcov Sampling variance-covariance matrix of the estimated total, direct, and indirect effects.

Author(s)

Ivan Jacob Agaloos Pesigan

DeltaMed 19

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), IndirectStd(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

```
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ).
  nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
vcov_phi_vec <- matrix(</pre>
  data = c(
    0.00843, 0.00040, -0.00151,
    -0.00600, -0.00033, 0.00110,
    0.00324, 0.00020, -0.00061,
    0.00040, 0.00374, 0.00016,
    -0.00022, -0.00273, -0.00016,
    0.00009, 0.00150, 0.00012,
    -0.00151, 0.00016, 0.00389,
    0.00103, -0.00007, -0.00283,
    -0.00050, 0.00000, 0.00156,
    -0.00600, -0.00022, 0.00103,
    0.00644, 0.00031, -0.00119,
    -0.00374, -0.00021, 0.00070,
    -0.00033, -0.00273, -0.00007,
    0.00031, 0.00287, 0.00013,
    -0.00014, -0.00170, -0.00012,
    0.00110, -0.00016, -0.00283,
    -0.00119, 0.00013, 0.00297,
    0.00063, -0.00004, -0.00177,
    0.00324, 0.00009, -0.00050,
```

20 DeltaMedStd

```
-0.00374, -0.00014, 0.00063,
   0.00495, 0.00024, -0.00093,
   0.00020, 0.00150, 0.00000,
   -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
   -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
   -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
DeltaMed(
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1,
 from = "x",
 to = "y",
 med = "m"
)
# Range of time intervals ------
delta <- DeltaMed(
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1:5,
 from = "x",
 to = "y",
 med = "m"
)
plot(delta)
# Methods -----
# DeltaMed has a number of methods including
# print, summary, confint, and plot
print(delta)
summary(delta)
confint(delta, level = 0.95)
plot(delta)
```

DeltaMedStd

Delta Method Sampling Variance-Covariance Matrix for the Standardized Total, Direct, and Indirect Effects of X on Y Through M Over a Specific Time Interval or a Range of Time Intervals

Description

This function computes the delta method sampling variance-covariance matrix for the standardized total, direct, and indirect effects of the independent variable X on the dependent variable Y through

DeltaMedStd 21

mediator variables \mathbf{m} over a specific time interval Δt or a range of time intervals using the first-order stochastic differential equation model's drift matrix $\mathbf{\Phi}$ and process noise covariance matrix $\mathbf{\Sigma}$.

Usage

```
DeltaMedStd(
   phi,
   sigma,
   vcov_theta,
   delta_t,
   from,
   to,
   med,
   ncores = NULL,
   tol = 0.01
)
```

Arguments

phi	Numeric matrix. The drift matrix (Φ) . phi should have row and column names pertaining to the variables in the system.
sigma	Numeric matrix. The process noise covariance matrix (Σ) .
vcov_theta	Numeric matrix. The sampling variance-covariance matrix of $\operatorname{vec}\left(\Phi\right)$ and $\operatorname{vech}\left(\Sigma\right)$
delta_t	Numeric. Time interval (Δt).
from	Character string. Name of the independent variable X in phi.
to	Character string. Name of the dependent variable Y in phi.
med	Character vector. Name/s of the mediator variable/s in phi.
ncores	Positive integer. Number of cores to use. If ncores = NULL, use a single core. Consider using multiple cores when number of replications R is a large value.
tol	Numeric. Smallest possible time interval to allow.

Details

See TotalStd(), DirectStd(), and IndirectStd() for more details.

Delta Method:

Let θ be a vector that combines $\operatorname{vec}(\Phi)$, that is, the elements of the Φ matrix in vector form sorted column-wise and $\operatorname{vech}(\Sigma)$, that is, the unique elements of the Σ matrix in vector form sorted column-wise. Let $\hat{\theta}$ be a vector that combines $\operatorname{vec}(\hat{\Phi})$ and $\operatorname{vech}(\hat{\Sigma})$. By the multivariate central limit theory, the function \mathbf{g} using $\hat{\theta}$ as input can be expressed as:

$$\sqrt{n}\left(\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right) - \mathbf{g}\left(\boldsymbol{\theta}\right)\right) \xrightarrow{\mathrm{D}} \mathcal{N}\left(0, \mathbf{J}\boldsymbol{\Gamma}\mathbf{J}'\right)$$

where **J** is the matrix of first-order derivatives of the function **g** with respect to the elements of θ and Γ is the asymptotic variance-covariance matrix of $\hat{\theta}$.

From the former, we can derive the distribution of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ as follows:

$$\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right) \approx \mathcal{N}\left(\mathbf{g}\left(\boldsymbol{\theta}\right), n^{-1}\mathbf{J}\boldsymbol{\Gamma}\mathbf{J}'\right)$$

The uncertainty associated with the estimator $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ is, therefore, given by $n^{-1}\mathbf{J}\Gamma\mathbf{J}'$. When Γ is unknown, by substitution, we can use the estimated sampling variance-covariance matrix of $\hat{\boldsymbol{\theta}}$, that is, $\hat{\mathbb{V}}\left(\hat{\boldsymbol{\theta}}\right)$ for $n^{-1}\Gamma$. Therefore, the sampling variance-covariance matrix of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ is given by

$$\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right) \approx \mathcal{N}\left(\mathbf{g}\left(\boldsymbol{\theta}\right), \mathbf{J}\hat{\mathbb{V}}\left(\hat{\boldsymbol{\theta}}\right) \mathbf{J}'\right).$$

Value

Returns an object of class ctmeddelta which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("DeltaMedStd").

output A list the length of which is equal to the length of delta_t.

Each element in the output list has the following elements:

delta t Time interval.

jacobian Jacobian matrix.

est Estimated total, direct, and indirect effects.

vcov Sampling variance-covariance matrix of the estimated total, direct, and indirect effects.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

DeltaMedStd 23

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), IndirectStd(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ),
 nrow = 3
)
colnames(phi) \leftarrow rownames(phi) \leftarrow c("x", "m", "y")
sigma <- matrix(</pre>
 data = c(
   0.24455556, 0.02201587, -0.05004762,
   0.02201587, 0.07067800, 0.01539456,
    -0.05004762, 0.01539456, 0.07553061
 ),
 nrow = 3
vcov_theta <- matrix(</pre>
 data = c(
    0.00843, 0.00040, -0.00151, -0.00600, -0.00033,
    0.00110, 0.00324, 0.00020, -0.00061, -0.00115,
    0.00011, 0.00015, 0.00001, -0.00002, -0.00001,
    0.00040, 0.00374, 0.00016, -0.00022, -0.00273,
    -0.00016, 0.00009, 0.00150, 0.00012, -0.00010,
    -0.00026, 0.00002, 0.00012, 0.00004, -0.00001,
    -0.00151, 0.00016, 0.00389, 0.00103, -0.00007,
    -0.00283, -0.00050, 0.00000, 0.00156, 0.00021,
    -0.00005, -0.00031, 0.00001, 0.00007, 0.00006,
    -0.00600, -0.00022, 0.00103, 0.00644, 0.00031,
    -0.00119, -0.00374, -0.00021, 0.00070, 0.00064,
    -0.00015, -0.00005, 0.00000, 0.00003, -0.00001,
    -0.00033, -0.00273, -0.00007, 0.00031, 0.00287,
    0.00013, -0.00014, -0.00170, -0.00012, 0.00006,
     0.00014, \ -0.00001, \ -0.00015, \ 0.00000, \ 0.00001, \\
    0.00110, -0.00016, -0.00283, -0.00119, 0.00013,
    0.00297, 0.00063, -0.00004, -0.00177, -0.00013,
    0.00005, 0.00017, -0.00002, -0.00008, 0.00001,
    0.00324, 0.00009, -0.00050, -0.00374, -0.00014,
    0.00063, 0.00495, 0.00024, -0.00093, -0.00020,
    0.00006, -0.00010, 0.00000, -0.00001, 0.00004,
    0.00020, 0.00150, 0.00000, -0.00021, -0.00170,
    -0.00004, 0.00024, 0.00214, 0.00012, -0.00002,
```

24 DeltaMedStd

```
-0.00004, 0.00000, 0.00006, -0.00005, -0.00001,
   -0.00061, 0.00012, 0.00156, 0.00070, -0.00012,
   -0.00177, -0.00093, 0.00012, 0.00223, 0.00004,
   -0.00002, -0.00003, 0.00001, 0.00003, -0.00013,
   -0.00115, -0.00010, 0.00021, 0.00064, 0.00006,
   -0.00013, -0.00020, -0.00002, 0.00004, 0.00057,
   0.00001, -0.00009, 0.00000, 0.00000, 0.00001,
   0.00011, -0.00026, -0.00005, -0.00015, 0.00014,
   0.00005, 0.00006, -0.00004, -0.00002, 0.00001,
   0.00012, 0.00001, 0.00000, -0.00002, 0.00000,
   0.00015, 0.00002, -0.00031, -0.00005, -0.00001,
   0.00017, -0.00010, 0.00000, -0.00003, -0.00009,
   0.00001, 0.00014, 0.00000, 0.00000, -0.00005,
   0.00001, 0.00012, 0.00001, 0.00000, -0.00015,
   -0.00002, 0.00000, 0.00006, 0.00001, 0.00000,
   0.00000, 0.00000, 0.00010, 0.00001, 0.00000,
   -0.00002, 0.00004, 0.00007, 0.00003, 0.00000,
   -0.00008, -0.00001, -0.00005, 0.00003, 0.00000,
   -0.00002, 0.00000, 0.00001, 0.00005, 0.00001,
   -0.00001, -0.00001, 0.00006, -0.00001, 0.00001,
   0.00001, 0.00004, -0.00001, -0.00013, 0.00001,
   0.00000, -0.00005, 0.00000, 0.00001, 0.00012
 ),
 nrow = 15
)
# Specific time interval -------
DeltaMedStd(
 phi = phi,
 sigma = sigma,
 vcov_theta = vcov_theta,
 delta_t = 1,
 from = "x",
 to = "y",
 med = "m"
)
# Range of time intervals ------
delta <- DeltaMedStd(</pre>
 phi = phi,
 sigma = sigma,
 vcov_theta = vcov_theta,
 delta_t = 1:5,
 from = "x",
 to = "y",
 med = "m"
)
plot(delta)
# Methods ------
# DeltaMedStd has a number of methods including
# print, summary, confint, and plot
print(delta)
```

DeltaTotalCentral 25

```
summary(delta)
confint(delta, level = 0.95)
plot(delta)
```

DeltaTotalCentral

Delta Method Sampling Variance-Covariance Matrix for the Total Effect Centrality Over a Specific Time Interval or a Range of Time Intervals

Description

This function computes the delta method sampling variance-covariance matrix for the total effect centrality over a specific time interval Δt or a range of time intervals using the first-order stochastic differential equation model's drift matrix Φ .

Usage

DeltaTotalCentral(phi, vcov_phi_vec, delta_t, ncores = NULL, tol = 0.01)

Arguments

phi	Numeric matrix. The drift matrix (Φ) , phi should have row and column names pertaining to the variables in the system.
vcov_phi_vec	Numeric matrix. The sampling variance-covariance matrix of $\operatorname{vec}\left(\Phi\right)$.
delta_t	Vector of positive numbers. Time interval (Δt) .
ncores	Positive integer. Number of cores to use. If ncores = NULL, use a single core. Consider using multiple cores when the length of delta_t is long.
tol	Numeric. Smallest possible time interval to allow.

Details

See TotalCentral() more details.

Delta Method:

Let θ be $\operatorname{vec}(\Phi)$, that is, the elements of the Φ matrix in vector form sorted column-wise. Let $\hat{\theta}$ be $\operatorname{vec}(\hat{\Phi})$. By the multivariate central limit theory, the function \mathbf{g} using $\hat{\theta}$ as input can be expressed as:

$$\sqrt{n}\left(\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right) - \mathbf{g}\left(\boldsymbol{\theta}\right)\right) \xrightarrow{\mathrm{D}} \mathcal{N}\left(0, \mathbf{J}\boldsymbol{\Gamma}\mathbf{J}'\right)$$

where **J** is the matrix of first-order derivatives of the function **g** with respect to the elements of θ and Γ is the asymptotic variance-covariance matrix of $\hat{\theta}$.

From the former, we can derive the distribution of $g(\hat{\theta})$ as follows:

$$\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right) \approx \mathcal{N}\left(\mathbf{g}\left(\boldsymbol{\theta}\right), n^{-1}\mathbf{J}\boldsymbol{\Gamma}\mathbf{J}'\right)$$

26 DeltaTotalCentral

The uncertainty associated with the estimator $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ is, therefore, given by $n^{-1}\mathbf{J}\Gamma\mathbf{J}'$. When Γ is unknown, by substitution, we can use the estimated sampling variance-covariance matrix of $\hat{\boldsymbol{\theta}}$, that is, $\hat{\mathbb{V}}\left(\hat{\boldsymbol{\theta}}\right)$ for $n^{-1}\Gamma$. Therefore, the sampling variance-covariance matrix of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ is given by

$$\mathbf{g}\left(\hat{\boldsymbol{ heta}}\right) pprox \mathcal{N}\left(\mathbf{g}\left(oldsymbol{ heta}
ight), \mathbf{J}\hat{\mathbb{V}}\left(\hat{oldsymbol{ heta}}
ight) \mathbf{J}'
ight).$$

Value

Returns an object of class ctmeddelta which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("DeltaTotalCentral").

output A list the length of which is equal to the length of delta_t.

Each element in the output list has the following elements:

delta_t Time interval.

jacobian Jacobian matrix.

est Estimated total effect centrality.

vcov Sampling variance-covariance matrix of estimated total effect centrality.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), IndirectStd(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

DeltaTotalCentral 27

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ),
 nrow = 3
)
colnames(phi) \leftarrow rownames(phi) \leftarrow c("x", "m", "y")
vcov_phi_vec <- matrix(</pre>
 data = c(
   0.00843, 0.00040, -0.00151,
   -0.00600, -0.00033, 0.00110,
   0.00324, 0.00020, -0.00061,
   0.00040, 0.00374, 0.00016,
    -0.00022, -0.00273, -0.00016,
   0.00009, 0.00150, 0.00012,
   -0.00151, 0.00016, 0.00389,
   0.00103, -0.00007, -0.00283,
   -0.00050, 0.00000, 0.00156,
   -0.00600, -0.00022, 0.00103,
   0.00644, 0.00031, -0.00119,
   -0.00374, -0.00021, 0.00070,
   -0.00033, -0.00273, -0.00007,
   0.00031, 0.00287, 0.00013,
   -0.00014, -0.00170, -0.00012,
   0.00110, -0.00016, -0.00283,
    -0.00119, 0.00013, 0.00297,
   0.00063, -0.00004, -0.00177,
   0.00324, 0.00009, -0.00050,
    -0.00374, -0.00014, 0.00063,
   0.00495, 0.00024, -0.00093,
   0.00020, 0.00150, 0.00000,
   -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
   -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
   -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
# Specific time interval ------
DeltaTotalCentral(
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1
)
# Range of time intervals ------
delta <- DeltaTotalCentral(</pre>
```

28 Direct

Direct

Direct Effect of X on Y Over a Specific Time Interval

Description

This function computes the direct effect of the independent variable X on the dependent variable Y through mediator variables \mathbf{m} over a specific time interval Δt using the first-order stochastic differential equation model's drift matrix $\mathbf{\Phi}$.

Usage

```
Direct(phi, delta_t, from, to, med)
```

Arguments

phi	Numeric matrix. The drift matrix (Φ). phi should have row and column names pertaining to the variables in the system.
delta_t	Numeric. Time interval (Δt).
from	Character string. Name of the independent variable X in \mathtt{phi} .
to	Character string. Name of the dependent variable Y in phi.
med	Character vector. Name/s of the mediator variable/s in phi.

Details

The direct effect of the independent variable X on the dependent variable Y relative to some mediator variables \mathbf{m} is given by

$$\operatorname{Direct}_{\Delta t_{i,j}} = \exp\left(\Delta t \mathbf{D} \mathbf{\Phi} \mathbf{D}\right)_{i,j}$$

where Φ denotes the drift matrix, \mathbf{D} a diagonal matrix where the diagonal elements corresponding to mediator variables \mathbf{m} are set to zero and the rest to one, i the row index of Y in Φ , j the column index of X in Φ , and Δt the time interval.

Direct 29

Linear Stochastic Differential Equation Model:

The measurement model is given by

$$\mathbf{y}_{i,t} = \mathbf{\nu} + \mathbf{\Lambda} \boldsymbol{\eta}_{i,t} + \boldsymbol{\varepsilon}_{i,t}, \quad ext{with} \quad \boldsymbol{\varepsilon}_{i,t} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Theta}\right)$$

where $\mathbf{y}_{i,t}$, $\eta_{i,t}$, and $\varepsilon_{i,t}$ are random variables and ν , Λ , and Θ are model parameters. $\mathbf{y}_{i,t}$ represents a vector of observed random variables, $\eta_{i,t}$ a vector of latent random variables, and $\varepsilon_{i,t}$ a vector of random measurement errors, at time t and individual t. ν denotes a vector of intercepts, Λ a matrix of factor loadings, and Θ the covariance matrix of ε .

An alternative representation of the measurement error is given by

$$oldsymbol{arepsilon}_{i,t} = oldsymbol{\Theta}^{rac{1}{2}} \mathbf{z}_{i,t}, \quad ext{with} \quad \mathbf{z}_{i,t} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{I}
ight)$$

where $\mathbf{z}_{i,t}$ is a vector of independent standard normal random variables and $\left(\Theta^{\frac{1}{2}}\right)\left(\Theta^{\frac{1}{2}}\right)' = \Theta$. The dynamic structure is given by

$$\mathrm{d} \boldsymbol{\eta}_{i,t} = \left(\boldsymbol{\iota} + \boldsymbol{\Phi} \boldsymbol{\eta}_{i,t} \right) \mathrm{d} t + \boldsymbol{\Sigma}^{\frac{1}{2}} \mathrm{d} \mathbf{W}_{i,t}$$

where ι is a term which is unobserved and constant over time, Φ is the drift matrix which represents the rate of change of the solution in the absence of any random fluctuations, Σ is the matrix of volatility or randomness in the process, and $\mathrm{d}W$ is a Wiener process or Brownian motion, which represents random fluctuations.

Value

Returns an object of class ctmedeffect which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("Direct").

output The direct effect.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

30 DirectStd

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), IndirectStd(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

Examples

```
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
delta_t <- 1
Direct(
  phi = phi,
  delta_t = delta_t,
  from = "x",
  to = "y",
  med = "m"
phi <- matrix(</pre>
  data = c(
    -6, 5.5, 0, 0,
    1.25, -2.5, 5.9, -7.3,
    0, 0, -6, 2.5,
    5, 0, 0, -6
  ),
  nrow = 4
)
colnames(phi) <- rownames(phi) <- paste0("y", 1:4)</pre>
Direct(
  phi = phi,
  delta_t = delta_t,
  from = "y2",
  to = "y4",
  med = c("y1", "y3")
)
```

DirectStd

DirectStd 31

Description

This function computes the standardized direct effect of the independent variable X on the dependent variable Y through mediator variables \mathbf{m} over a specific time interval Δt using the first-order stochastic differential equation model's drift matrix $\mathbf{\Phi}$ and process noise covariance matrix $\mathbf{\Sigma}$.

Usage

```
DirectStd(phi, sigma, delta_t, from, to, med)
```

Arguments

phi	Numeric matrix. The drift matrix (Φ). phi should have row and column names pertaining to the variables in the system.
sigma	Numeric matrix. The process noise covariance matrix (Σ) .
delta_t	Numeric. Time interval (Δt).
from	Character string. Name of the independent variable X in \mathtt{phi} .
to	Character string. Name of the dependent variable Y in phi.
med	Character vector. Name/s of the mediator variable/s in phi.

Details

The standardized direct effect of the independent variable X on the dependent variable Y relative to some mediator variables \mathbf{m} is given by

$$\mathrm{Direct}_{\Delta t_{i,j}}^{*} = \mathbf{S} \left(\exp \left(\Delta t \mathbf{D} \mathbf{\Phi} \mathbf{D} \right)_{i,j} \right) \mathbf{S}^{-1}$$

where Φ denotes the drift matrix, \mathbf{D} a diagonal matrix where the diagonal elements corresponding to mediator variables \mathbf{m} are set to zero and the rest to one, i the row index of Y in Φ , j the column index of X in Φ , \mathbf{S} a diagonal matrix with model-implied standard deviations on the diagonals, and Δt the time interval.

Value

Returns an object of class ctmedeffect which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("DirectStd").

output The direct effect.

Author(s)

Ivan Jacob Agaloos Pesigan

32 DirectStd

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), IndirectStd(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

```
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
sigma <- matrix(</pre>
  data = c(
    0.24455556, 0.02201587, -0.05004762,
    0.02201587, 0.07067800, 0.01539456,
    -0.05004762, 0.01539456, 0.07553061
  ),
  nrow = 3
)
delta_t <- 1
DirectStd(
  phi = phi,
  sigma = sigma,
  delta_t = delta_t,
  from = "x",
  to = "y",
  med = "m"
```

ExpCov

33

ExpCov

Model-Implied State Covariance Matrix

Description

The function returns the model-implied state covariance matrix for a particular time interval Δt given by

$$\operatorname{vec}\left(\operatorname{Cov}\left(\boldsymbol{\eta}\right)\right) = \left(\mathbf{J} - \boldsymbol{\beta}_{\Delta t} \otimes \boldsymbol{\beta}_{\Delta t}\right)^{-1} \operatorname{vec}\left(\boldsymbol{\Psi}_{\Delta t}\right)$$

where

$$\begin{split} \boldsymbol{\beta}_{\Delta t} &= \exp \left(\Delta t \boldsymbol{\Phi} \right), \\ \boldsymbol{\Psi}_{\Delta t} &= \boldsymbol{\Phi}^{\#} \left(\exp \left(\Delta t \boldsymbol{\Phi} \right) - \mathbf{J} \right) \operatorname{vec} \left(\boldsymbol{\Sigma} \right), \quad \text{and} \\ \boldsymbol{\Phi}^{\#} &= \left(\boldsymbol{\Phi} \otimes \mathbf{I} \right) + \left(\mathbf{I} \otimes \boldsymbol{\Phi} \right). \end{split}$$

Note that I and J are identity matrices.

Usage

ExpCov(phi, sigma, delta_t)

Arguments

Numeric matrix. The drift matrix (Φ). phi should have row and column names phi pertaining to the variables in the system.

sigma Numeric matrix. The process noise covariance matrix (Σ) .

delta_t Numeric. Time interval (Δt).

Details

Linear Stochastic Differential Equation Model:

The measurement model is given by

$$\mathbf{y}_{i,t} = \mathbf{\nu} + \mathbf{\Lambda} \boldsymbol{\eta}_{i,t} + \boldsymbol{\varepsilon}_{i,t}, \quad ext{with} \quad \boldsymbol{\varepsilon}_{i,t} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Theta}\right)$$

where $y_{i,t}$, $\eta_{i,t}$, and $\varepsilon_{i,t}$ are random variables and ν , Λ , and Θ are model parameters. $y_{i,t}$ represents a vector of observed random variables, $\eta_{i,t}$ a vector of latent random variables, and $\varepsilon_{i,t}$ a vector of random measurement errors, at time t and individual i. ν denotes a vector of intercepts, Λ a matrix of factor loadings, and Θ the covariance matrix of ε .

An alternative representation of the measurement error is given by

$$oldsymbol{arepsilon}_{i,t} = oldsymbol{\Theta}^{rac{1}{2}} \mathbf{z}_{i,t}, \quad ext{with} \quad \mathbf{z}_{i,t} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{I}\right)$$

where $\mathbf{z}_{i,t}$ is a vector of independent standard normal random variables and $\left(\mathbf{\Theta}^{\frac{1}{2}}\right)\left(\mathbf{\Theta}^{\frac{1}{2}}\right)' = \mathbf{\Theta}$. The dynamic structure is given by

$$\mathrm{d}\boldsymbol{\eta}_{i,t} = \left(\boldsymbol{\iota} + \boldsymbol{\Phi}\boldsymbol{\eta}_{i,t}\right) \mathrm{d}t + \boldsymbol{\Sigma}^{\frac{1}{2}} \mathrm{d}\mathbf{W}_{i,t}$$

where ι is a term which is unobserved and constant over time, Φ is the drift matrix which represents the rate of change of the solution in the absence of any random fluctuations, Σ is the matrix of volatility or randomness in the process, and dW is a Wiener process or Brownian motion, which represents random fluctuations.

34 ExpMean

Value

Returns a numeric matrix.

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

```
Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpMean(), Indirect(), IndirectCentral(), IndirectStd(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()
```

Examples

```
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
sigma <- matrix(</pre>
  data = c(
    0.24, 0.02, -0.05,
    0.02, 0.07, 0.02,
    -0.05, 0.02, 0.08
  ),
  nrow = 3
)
delta_t <- 1
ExpCov(
  phi = phi,
  sigma = sigma,
  delta_t = delta_t
)
```

ExpMean

ExpMean 35

Description

The function returns the model-implied state mean vector for a particular time interval Δt given by

Mean
$$(\boldsymbol{\eta}) = (\mathbf{I} - \boldsymbol{\beta}_{\Delta t})^{-1} \boldsymbol{\alpha}_{\Delta t}$$

where

$$oldsymbol{eta}_{\Delta t} = \exp\left(\Delta t oldsymbol{\Phi}\right),$$
 $oldsymbol{lpha}_{\Delta t} = oldsymbol{\Phi}^{-1} \left(oldsymbol{eta}_{\Delta t} - \mathbf{I}\right) oldsymbol{\iota}.$

Note that I is an identity matrix.

Usage

ExpMean(phi, iota, delta_t)

Arguments

phi Numeric matrix. The drift matrix (Φ) , phi should have row and column names

pertaining to the variables in the system.

iota Numeric vector. An unobserved term that is constant over time (ι).

delta_t Numeric. Time interval (Δt).

Details

Linear Stochastic Differential Equation Model:

The measurement model is given by

$$\mathbf{y}_{i,t} = \boldsymbol{\nu} + \boldsymbol{\Lambda} \boldsymbol{\eta}_{i,t} + \boldsymbol{\varepsilon}_{i,t}, \quad ext{with} \quad \boldsymbol{\varepsilon}_{i,t} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Theta}\right)$$

where $\mathbf{y}_{i,t}$, $\eta_{i,t}$, and $\varepsilon_{i,t}$ are random variables and ν , Λ , and Θ are model parameters. $\mathbf{y}_{i,t}$ represents a vector of observed random variables, $\eta_{i,t}$ a vector of latent random variables, and $\varepsilon_{i,t}$ a vector of random measurement errors, at time t and individual i. ν denotes a vector of intercepts, Λ a matrix of factor loadings, and Θ the covariance matrix of ε .

An alternative representation of the measurement error is given by

$$\boldsymbol{\varepsilon}_{i,t} = \boldsymbol{\Theta}^{\frac{1}{2}} \mathbf{z}_{i,t}, \quad \text{with} \quad \mathbf{z}_{i,t} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{I}\right)$$

where $\mathbf{z}_{i,t}$ is a vector of independent standard normal random variables and $\left(\Theta^{\frac{1}{2}}\right)\left(\Theta^{\frac{1}{2}}\right)' = \Theta$. The dynamic structure is given by

$$\mathrm{d}oldsymbol{\eta}_{i,t} = \left(oldsymbol{\iota} + oldsymbol{\Phi}oldsymbol{\eta}_{i,t}
ight)\mathrm{d}t + oldsymbol{\Sigma}^{rac{1}{2}}\mathrm{d}\mathbf{W}_{i,t}$$

where ι is a term which is unobserved and constant over time, Φ is the drift matrix which represents the rate of change of the solution in the absence of any random fluctuations, Σ is the matrix of volatility or randomness in the process, and $\mathrm{d} W$ is a Wiener process or Brownian motion, which represents random fluctuations.

Value

Returns a numeric matrix.

36 Indirect

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), Indirect(), IndirectCentral(), IndirectStd(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

Examples

```
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
)
colnames(phi) \leftarrow rownames(phi) \leftarrow c("x", "m", "y")
iota <-c(.5, .3, .4)
delta_t <- 1
ExpMean(
  phi = phi,
  iota = iota,
  delta_t = delta_t
)
```

Indirect

Indirect Effect of X on Y Through M Over a Specific Time Interval

Description

This function computes the indirect effect of the independent variable X on the dependent variable Y through mediator variables \mathbf{m} over a specific time interval Δt using the first-order stochastic differential equation model's drift matrix $\mathbf{\Phi}$.

Usage

```
Indirect(phi, delta_t, from, to, med)
```

Indirect 37

Arguments

phi	Numeric matrix. The drift matrix (Φ) . phi should have row and column names
	pertaining to the variables in the system.
delta_t	Numeric. Time interval (Δt).
from	Character string. Name of the independent variable X in phi.
to	Character string. Name of the dependent variable Y in phi.
med	Character vector. Name/s of the mediator variable/s in phi.

Details

The indirect effect of the independent variable X on the dependent variable Y relative to some mediator variables \mathbf{m} over a specific time interval Δt is given by

$$Indirect_{\Delta t_{i,j}} = \exp(\Delta t \mathbf{\Phi})_{i,j} - \exp(\Delta t \mathbf{D_m} \mathbf{\Phi} \mathbf{D_m})_{i,j}$$

where Φ denotes the drift matrix, $\mathbf{D_m}$ a matrix where the off diagonal elements are zeros and the diagonal elements are zero for the index/indices of mediator variables \mathbf{m} and one otherwise, i the row index of Y in Φ , j the column index of X in Φ , and Δt the time interval.

Linear Stochastic Differential Equation Model:

The measurement model is given by

$$\mathbf{y}_{i,t} = \mathbf{\nu} + \mathbf{\Lambda} oldsymbol{\eta}_{i.t} + oldsymbol{arepsilon}_{i,t}, \quad ext{with} \quad oldsymbol{arepsilon}_{i,t} \sim \mathcal{N}\left(\mathbf{0}, oldsymbol{\Theta}
ight)$$

where $\mathbf{y}_{i,t}$, $\eta_{i,t}$, and $\varepsilon_{i,t}$ are random variables and ν , Λ , and Θ are model parameters. $\mathbf{y}_{i,t}$ represents a vector of observed random variables, $\eta_{i,t}$ a vector of latent random variables, and $\varepsilon_{i,t}$ a vector of random measurement errors, at time t and individual t. ν denotes a vector of intercepts, Λ a matrix of factor loadings, and Θ the covariance matrix of ε .

An alternative representation of the measurement error is given by

$$oldsymbol{arepsilon}_{i,t} = oldsymbol{\Theta}^{rac{1}{2}} \mathbf{z}_{i,t}, \quad ext{with} \quad \mathbf{z}_{i,t} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{I}
ight)$$

where $\mathbf{z}_{i,t}$ is a vector of independent standard normal random variables and $\left(\mathbf{\Theta}^{\frac{1}{2}}\right)\left(\mathbf{\Theta}^{\frac{1}{2}}\right)' = \mathbf{\Theta}$. The dynamic structure is given by

$$\mathrm{d}\boldsymbol{\eta}_{i,t} = \left(\boldsymbol{\iota} + \boldsymbol{\Phi}\boldsymbol{\eta}_{i,t}\right)\mathrm{d}t + \boldsymbol{\Sigma}^{\frac{1}{2}}\mathrm{d}\mathbf{W}_{i,t}$$

where ι is a term which is unobserved and constant over time, Φ is the drift matrix which represents the rate of change of the solution in the absence of any random fluctuations, Σ is the matrix of volatility or randomness in the process, and $\mathrm{d}W$ is a Wiener process or Brownian motion, which represents random fluctuations.

Value

Returns an object of class ctmedeffect which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("Indirect").

output The indirect effect.

38 Indirect

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), IndirectCentral(), IndirectStd(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

```
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
delta_t <- 1
Indirect(
  phi = phi,
  delta_t = delta_t,
  from = "x",
  to = "y",
  med = "m"
)
phi <- matrix(</pre>
  data = c(
    -6, 5.5, 0, 0,
    1.25, -2.5, 5.9, -7.3,
    0, 0, -6, 2.5,
    5, 0, 0, -6
  ),
  nrow = 4
)
```

```
colnames(phi) <- rownames(phi) <- paste0("y", 1:4)
Indirect(
  phi = phi,
  delta_t = delta_t,
  from = "y2",
  to = "y4",
  med = c("y1", "y3")
)</pre>
```

IndirectCentral

Indirect Effect Centrality

Description

Indirect Effect Centrality

Usage

```
IndirectCentral(phi, delta_t, tol = 0.01)
```

Arguments

phi	Numeric matrix. The drift matrix (Φ) , phi should have row and column names pertaining to the variables in the system.
delta_t	Vector of positive numbers. Time interval (Δt) .
tol	Numeric. Smallest possible time interval to allow.

Details

Indirect effect centrality is the sum of all possible indirect effects between different pairs of variables in which a specific variable serves as the only mediator.

Value

Returns an object of class ctmedmed which is a list with the following elements:

```
call Function call.args Function arguments.fun Function used ("IndirectCentral").output A matrix of indirect effect centrality.
```

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectStd(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ),
 nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
# Specific time interval ------
IndirectCentral(
 phi = phi,
 delta_t = 1
)
# Range of time intervals ------
indirect_central <- IndirectCentral(</pre>
 phi = phi,
 delta_t = 1:30
plot(indirect_central)
# Methods -----
# IndirectCentral has a number of methods including
# print, summary, and plot
indirect_central <- IndirectCentral(</pre>
 phi = phi,
 delta_t = 1:5
)
```

IndirectStd 41

```
print(indirect_central)
summary(indirect_central)
plot(indirect_central)
```

IndirectStd	Standardized Indirect Effect of X on Y Through M Over a Specific Time
	Interval

Description

This function computes the standardized indirect effect of the independent variable X on the dependent variable Y through mediator variables \mathbf{m} over a specific time interval Δt using the first-order stochastic differential equation model's drift matrix $\mathbf{\Phi}$ and process noise covariance matrix $\mathbf{\Sigma}$.

Usage

```
IndirectStd(phi, sigma, delta_t, from, to, med)
```

Arguments

phi	Numeric matrix. The drift matrix (Φ) , phi should have row and column names pertaining to the variables in the system.
sigma	Numeric matrix. The process noise covariance matrix (Σ) .
delta_t	Numeric. Time interval (Δt).
from	Character string. Name of the independent variable X in phi.
to	Character string. Name of the dependent variable Y in phi.
med	Character vector. Name/s of the mediator variable/s in phi.

Details

The standardized indirect effect of the independent variable X on the dependent variable Y relative to some mediator variables \mathbf{m} over a specific time interval Δt is given by

$$Indirect^*_{\Delta t_{i,j}} = Total^*_{\Delta t} - Direct^*_{\Delta t}$$

where $\operatorname{Total}_{\Delta t}^*$ and $\operatorname{Direct}_{\Delta t}^*$ are standardized total and direct effects for time interval Δt .

Value

Returns an object of class ctmedeffect which is a list with the following elements:

```
call Function call.args Function arguments.fun Function used ("IndirectStd").output The indirect effect.
```

42 IndirectStd

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

```
Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()
```

```
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
sigma <- matrix(</pre>
  data = c(
    0.24455556, 0.02201587, -0.05004762,
    0.02201587, 0.07067800, 0.01539456,
    -0.05004762, 0.01539456, 0.07553061
  ),
  nrow = 3
)
delta_t <- 1
IndirectStd(
  phi = phi,
  sigma = sigma,
  delta_t = delta_t,
  from = "x",
  to = "y",
  med = "m"
)
```

MCBeta 43

MCBeta	Monte Carlo Sampling Distribution for the Elements of the Matrix of
	Lagged Coefficients Over a Specific Time Interval or a Range of Time
	Intervals

Description

This function generates a Monte Carlo method sampling distribution for the elements of the matrix of lagged coefficients β over a specific time interval Δt or a range of time intervals using the first-order stochastic differential equation model drift matrix Φ .

Usage

```
MCBeta(
   phi,
   vcov_phi_vec,
   delta_t,
   R,
   test_phi = TRUE,
   ncores = NULL,
   seed = NULL,
   tol = 0.01
)
```

Arguments

ph1	Numeric matrix. The drift matrix (Φ), phi should have row and column names pertaining to the variables in the system.
vcov_phi_vec	Numeric matrix. The sampling variance-covariance matrix of $\operatorname{vec}\left(\mathbf{\Phi}\right)$.
delta_t	Numeric. Time interval (Δt).
R	Positive integer. Number of replications.
test_phi	Logical. If test_phi = TRUE, the function tests the stability of the generated drift matrix Φ . If the test returns FALSE, the function generates a new drift matrix Φ and runs the test recursively until the test returns TRUE.
ncores	Positive integer. Number of cores to use. If ncores = NULL, use a single core. Consider using multiple cores when number of replications R is a large value.
seed	Random seed.
tol	Numeric. Smallest possible time interval to allow.

Details

44

See Total().

Monte Carlo Method:

Let θ be $\operatorname{vec}(\Phi)$, that is, the elements of the Φ matrix in vector form sorted column-wise. Let $\hat{\theta}$ be $\operatorname{vec}(\hat{\Phi})$. Based on the asymptotic properties of maximum likelihood estimators, we can assume that estimators are normally distributed around the population parameters.

MCBeta

$$\hat{oldsymbol{ heta}} \sim \mathcal{N}\left(oldsymbol{ heta}, \mathbb{V}\left(\hat{oldsymbol{ heta}}
ight)
ight)$$

Using this distributional assumption, a sampling distribution of $\hat{\theta}$ which we refer to as $\hat{\theta}^*$ can be generated by replacing the population parameters with sample estimates, that is,

$$\hat{oldsymbol{ heta}}^* \sim \mathcal{N}\left(\hat{oldsymbol{ heta}}, \hat{\mathbb{V}}\left(\hat{oldsymbol{ heta}}
ight)
ight).$$

Let $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ be a parameter that is a function of the estimated parameters. A sampling distribution of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$, which we refer to as $\mathbf{g}\left(\hat{\boldsymbol{\theta}}^*\right)$, can be generated by using the simulated estimates to calculate \mathbf{g} . The standard deviations of the simulated estimates are the standard errors. Percentiles corresponding to $100\left(1-\alpha\right)\%$ are the confidence intervals.

Value

Returns an object of class ctmedmc which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("MCBeta").

output A list the length of which is equal to the length of delta_t.

Each element in the output list has the following elements:

est A vector of total, direct, and indirect effects.

thetahatstar A matrix of Monte Carlo total, direct, and indirect effects.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

MCBeta 45

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

```
set.seed(42)
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
vcov_phi_vec <- matrix(</pre>
  data = c(
    0.00843, 0.00040, -0.00151,
    -0.00600, -0.00033, 0.00110,
    0.00324, 0.00020, -0.00061,
    0.00040, 0.00374, 0.00016,
    -0.00022, -0.00273, -0.00016,
    0.00009, 0.00150, 0.00012,
    -0.00151, 0.00016, 0.00389,
    0.00103, -0.00007, -0.00283,
    -0.00050, 0.00000, 0.00156,
    -0.00600, -0.00022, 0.00103,
    0.00644, 0.00031, -0.00119,
    -0.00374, -0.00021, 0.00070,
    -0.00033, -0.00273, -0.00007,
    0.00031, 0.00287, 0.00013,
    -0.00014, -0.00170, -0.00012,
    0.00110, -0.00016, -0.00283,
    -0.00119, 0.00013, 0.00297,
    0.00063, -0.00004, -0.00177,
    0.00324, 0.00009, -0.00050,
    -0.00374, -0.00014, 0.00063,
    0.00495, 0.00024, -0.00093,
    0.00020, 0.00150, 0.00000,
    -0.00021, -0.00170, -0.00004,
    0.00024, 0.00214, 0.00012,
    -0.00061, 0.00012, 0.00156,
    0.00070, -0.00012, -0.00177,
    -0.00093, 0.00012, 0.00223
  ),
  nrow = 9
)
```

```
# Specific time interval ------
MCBeta(
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1,
 R = 100L # use a large value for R in actual research
# Range of time intervals ------
mc <- MCBeta(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1:5,
 R = 100L # use a large value for R in actual research
)
plot(mc)
# Methods ------
# MCBeta has a number of methods including
# print, summary, confint, and plot
print(mc)
summary(mc)
confint(mc, level = 0.95)
plot(mc)
```

MCBetaStd

Monte Carlo Sampling Distribution for the Elements of the Standardized Matrix of Lagged Coefficients Over a Specific Time Interval or a Range of Time Intervals

Description

This function generates a Monte Carlo method sampling distribution for the elements of the standardized matrix of lagged coefficients β over a specific time interval Δt or a range of time intervals using the first-order stochastic differential equation model drift matrix Φ and process noise covariance matrix Σ .

Usage

```
MCBetaStd(
   phi,
   sigma,
   vcov_theta,
   delta_t,
   R,
   test_phi = TRUE,
   ncores = NULL,
```

```
seed = NULL,
tol = 0.01
)
```

Arguments

phi Numeric matrix. The drift matrix (Φ) , phi should have row and column names pertaining to the variables in the system. sigma Numeric matrix. The process noise covariance matrix (Σ) . vcov_theta Numeric matrix. The sampling variance-covariance matrix of vec (Φ) and vech (Σ) delta_t Numeric. Time interval (Δt). R Positive integer. Number of replications. test_phi Logical. If test_phi = TRUE, the function tests the stability of the generated drift matrix Φ . If the test returns FALSE, the function generates a new drift matrix Φ and runs the test recursively until the test returns TRUE. Positive integer. Number of cores to use. If ncores = NULL, use a single core. ncores Consider using multiple cores when number of replications R is a large value. Random seed. seed

Numeric. Smallest possible time interval to allow.

Details

tol

See TotalStd().

Monte Carlo Method:

Let θ be a vector that combines $\operatorname{vec}(\Phi)$, that is, the elements of the Φ matrix in vector form sorted column-wise and $\operatorname{vech}(\Sigma)$, that is, the unique elements of the Σ matrix in vector form sorted column-wise. Let $\hat{\theta}$ be a vector that combines $\operatorname{vec}(\hat{\Phi})$ and $\operatorname{vech}(\hat{\Sigma})$. Based on the asymptotic properties of maximum likelihood estimators, we can assume that estimators are normally distributed around the population parameters.

$$\hat{oldsymbol{ heta}} \sim \mathcal{N}\left(oldsymbol{ heta}, \mathbb{V}\left(\hat{oldsymbol{ heta}}
ight)
ight)$$

Using this distributional assumption, a sampling distribution of $\hat{\theta}$ which we refer to as $\hat{\theta}^*$ can be generated by replacing the population parameters with sample estimates, that is,

$$\hat{oldsymbol{ heta}}^* \sim \mathcal{N}\left(\hat{oldsymbol{ heta}}, \hat{\mathbb{V}}\left(\hat{oldsymbol{ heta}}
ight)
ight).$$

Let $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ be a parameter that is a function of the estimated parameters. A sampling distribution of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$, which we refer to as $\mathbf{g}\left(\hat{\boldsymbol{\theta}}^*\right)$, can be generated by using the simulated estimates to calculate \mathbf{g} . The standard deviations of the simulated estimates are the standard errors. Percentiles corresponding to $100\left(1-\alpha\right)\%$ are the confidence intervals.

Value

Returns an object of class ctmedmc which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("MCBetaStd").

output A list the length of which is equal to the length of delta_t.

Each element in the output list has the following elements:

est A vector of total, direct, and indirect effects.

thetahatstar A matrix of Monte Carlo total, direct, and indirect effects.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

```
Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), IndirectStd(), MCBeta(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()
```

```
phi <- matrix(
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")
sigma <- matrix(
  data = c(</pre>
```

```
0.24455556, 0.02201587, -0.05004762,
   0.02201587, 0.07067800, 0.01539456,
   -0.05004762, 0.01539456, 0.07553061
 ),
 nrow = 3
)
vcov_theta <- matrix(
 data = c(
   0.00843, 0.00040, -0.00151, -0.00600, -0.00033,
   0.00110, 0.00324, 0.00020, -0.00061, -0.00115,
   0.00011, 0.00015, 0.00001, -0.00002, -0.00001,
   0.00040, 0.00374, 0.00016, -0.00022, -0.00273,
   -0.00016, 0.00009, 0.00150, 0.00012, -0.00010,
   -0.00026, 0.00002, 0.00012, 0.00004, -0.00001,
   -0.00151, 0.00016, 0.00389, 0.00103, -0.00007,
   -0.00283, -0.00050, 0.00000, 0.00156, 0.00021,
   -0.00005, -0.00031, 0.00001, 0.00007, 0.00006,
   -0.00600, -0.00022, 0.00103, 0.00644, 0.00031,
   -0.00119, -0.00374, -0.00021, 0.00070, 0.00064,
   -0.00015, -0.00005, 0.00000, 0.00003, -0.00001,
   -0.00033, -0.00273, -0.00007, 0.00031, 0.00287,
   0.00013, -0.00014, -0.00170, -0.00012, 0.00006,
   0.00014, -0.00001, -0.00015, 0.00000, 0.00001,
   0.00110, -0.00016, -0.00283, -0.00119, 0.00013,
   0.00297, 0.00063, -0.00004, -0.00177, -0.00013,
   0.00005, 0.00017, -0.00002, -0.00008, 0.00001,
   0.00324, 0.00009, -0.00050, -0.00374, -0.00014,
   0.00063, 0.00495, 0.00024, -0.00093, -0.00020,
   0.00006, -0.00010, 0.00000, -0.00001, 0.00004,
   0.00020, 0.00150, 0.00000, -0.00021, -0.00170,
   -0.00004, 0.00024, 0.00214, 0.00012, -0.00002,
   -0.00004, 0.00000, 0.00006, -0.00005, -0.00001,
   -0.00061, 0.00012, 0.00156, 0.00070, -0.00012,
   -0.00177, -0.00093, 0.00012, 0.00223, 0.00004,
   -0.00002, -0.00003, 0.00001, 0.00003, -0.00013,
   -0.00115, -0.00010, 0.00021, 0.00064, 0.00006,
   -0.00013, -0.00020, -0.00002, 0.00004, 0.00057,
   0.00001, -0.00009, 0.00000, 0.00000, 0.00001,
   0.00011, -0.00026, -0.00005, -0.00015, 0.00014,
   0.00005, 0.00006, -0.00004, -0.00002, 0.00001,
   0.00012, 0.00001, 0.00000, -0.00002, 0.00000,
   0.00015, 0.00002, -0.00031, -0.00005, -0.00001,
   0.00017, -0.00010, 0.00000, -0.00003, -0.00009,
   0.00001, 0.00014, 0.00000, 0.00000, -0.00005,
   0.00001, 0.00012, 0.00001, 0.00000, -0.00015,
   -0.00002, 0.00000, 0.00006, 0.00001, 0.00000,
   0.00000, 0.00000, 0.00010, 0.00001, 0.00000,
   -0.00002, 0.00004, 0.00007, 0.00003, 0.00000,
   -0.00008, -0.00001, -0.00005, 0.00003, 0.00000,
   -0.00002, 0.00000, 0.00001, 0.00005, 0.00001,
   -0.00001, -0.00001, 0.00006, -0.00001, 0.00001,
   0.00001, 0.00004, -0.00001, -0.00013, 0.00001,
   0.00000, -0.00005, 0.00000, 0.00001, 0.00012
```

```
),
 nrow = 15
)
# Specific time interval ------
MCBetaStd(
 phi = phi,
 sigma = sigma,
 vcov_theta = vcov_theta,
 delta_t = 1,
 R = 100L # use a large value for R in actual research
# Range of time intervals -----
mc <- MCBetaStd(</pre>
 phi = phi,
 sigma = sigma,
 vcov_theta = vcov_theta,
 delta_t = 1:5,
 R = 100L # use a large value for R in actual research
)
plot(mc)
# Methods -----
# MCBetaStd has a number of methods including
# print, summary, confint, and plot
print(mc)
summary(mc)
confint(mc, level = 0.95)
plot(mc)
```

MCIndirectCentral

Monte Carlo Sampling Distribution of Indirect Effect Centrality Over a Specific Time Interval or a Range of Time Intervals

Description

This function generates a Monte Carlo method sampling distribution of the indirect effect centrality at a particular time interval Δt using the first-order stochastic differential equation model drift matrix Φ .

Usage

```
MCIndirectCentral(
  phi,
  vcov_phi_vec,
  delta_t,
  R,
```

```
test_phi = TRUE,
  ncores = NULL,
 seed = NULL.
  tol = 0.01
)
```

Arguments

phi Numeric matrix. The drift matrix (Φ). phi should have row and column names pertaining to the variables in the system. Numeric matrix. The sampling variance-covariance matrix of $\text{vec}(\Phi)$. vcov_phi_vec Numeric. Time interval (Δt). delta_t Positive integer. Number of replications. test_phi Logical. If test_phi = TRUE, the function tests the stability of the generated drift matrix Φ . If the test returns FALSE, the function generates a new drift matrix Φ and runs the test recursively until the test returns TRUE. Positive integer. Number of cores to use. If ncores = NULL, use a single core. ncores Consider using multiple cores when number of replications R is a large value. Random seed. seed

tol Numeric. Smallest possible time interval to allow.

Details

See IndirectCentral() for more details.

Monte Carlo Method:

Let θ be vec (Φ) , that is, the elements of the Φ matrix in vector form sorted column-wise. Let $\hat{\theta}$ be vec $(\hat{\Phi})$. Based on the asymptotic properties of maximum likelihood estimators, we can assume that estimators are normally distributed around the population parameters.

$$\hat{oldsymbol{ heta}} \sim \mathcal{N}\left(oldsymbol{ heta}, \mathbb{V}\left(\hat{oldsymbol{ heta}}
ight)
ight)$$

Using this distributional assumption, a sampling distribution of $\hat{\theta}$ which we refer to as $\hat{\theta}^*$ can be generated by replacing the population parameters with sample estimates, that is,

$$\hat{oldsymbol{ heta}}^* \sim \mathcal{N}\left(\hat{oldsymbol{ heta}}, \hat{\mathbb{V}}\left(\hat{oldsymbol{ heta}}
ight)
ight).$$

Let $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ be a parameter that is a function of the estimated parameters. A sampling distribution of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$, which we refer to as $\mathbf{g}\left(\hat{\boldsymbol{\theta}}^*\right)$, can be generated by using the simulated estimates to calculate g. The standard deviations of the simulated estimates are the standard errors. Percentiles corresponding to $100 (1 - \alpha) \%$ are the confidence intervals.

Value

Returns an object of class ctmedmc which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("MCIndirectCentral").

output A list the length of which is equal to the length of delta_t.

Each element in the output list has the following elements:

est A vector of indirect effect centrality.

thetahatstar A matrix of Monte Carlo indirect effect centrality.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), IndirectStd(), MCBeta(), MCBetaStd(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

```
set.seed(42)
phi <- matrix(
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")
vcov_phi_vec <- matrix(</pre>
```

```
data = c(
   0.00843, 0.00040, -0.00151,
   -0.00600, -0.00033, 0.00110,
   0.00324, 0.00020, -0.00061,
   0.00040, 0.00374, 0.00016,
   -0.00022, -0.00273, -0.00016,
   0.00009, 0.00150, 0.00012,
   -0.00151, 0.00016, 0.00389,
   0.00103, -0.00007, -0.00283,
   -0.00050, 0.00000, 0.00156,
   -0.00600, -0.00022, 0.00103,
   0.00644, 0.00031, -0.00119,
   -0.00374, -0.00021, 0.00070,
   -0.00033, -0.00273, -0.00007,
   0.00031, 0.00287, 0.00013,
   -0.00014, -0.00170, -0.00012,
   0.00110, -0.00016, -0.00283,
   -0.00119, 0.00013, 0.00297,
   0.00063, -0.00004, -0.00177,
   0.00324, 0.00009, -0.00050,
   -0.00374, -0.00014, 0.00063,
   0.00495, 0.00024, -0.00093,
   0.00020, 0.00150, 0.00000,
   -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
   -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
   -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
# Specific time interval -------
MCIndirectCentral(
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1,
 R = 100L # use a large value for R in actual research
)
# Range of time intervals ------
mc <- MCIndirectCentral(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1:5,
 R = 100L # use a large value for R in actual research
)
plot(mc)
# Methods ------
# MCIndirectCentral has a number of methods including
# print, summary, confint, and plot
print(mc)
```

54 MCMed

```
summary(mc)
confint(mc, level = 0.95)
plot(mc)
```

MCMed

Monte Carlo Sampling Distribution of Total, Direct, and Indirect Effects of X on Y Through M Over a Specific Time Interval or a Range of Time Intervals

Description

This function generates a Monte Carlo method sampling distribution of the total, direct and indirect effects of the independent variable X on the dependent variable Y through mediator variables \mathbf{m} over a specific time interval Δt or a range of time intervals using the first-order stochastic differential equation model drift matrix $\mathbf{\Phi}$.

Usage

```
MCMed(
   phi,
   vcov_phi_vec,
   delta_t,
   from,
   to,
   med,
   R,
   test_phi = TRUE,
   ncores = NULL,
   seed = NULL,
   tol = 0.01
)
```

Arguments

pertaining to the variables in the system.

vcov_phi_vec Numeric matrix. The sampling variance-covariance matrix of $vec(\Phi)$.

delta_t Numeric. Time interval (Δt).

from Character string. Name of the independent variable X in phi. to Character string. Name of the dependent variable Y in phi. med Character vector. Name/s of the mediator variable/s in phi.

R Positive integer. Number of replications.

test_phi Logical. If test_phi = TRUE, the function tests the stability of the generated

drift matrix Φ . If the test returns FALSE, the function generates a new drift

matrix Φ and runs the test recursively until the test returns TRUE.

MCMed 55

ncores Positive integer. Number of cores to use. If ncores = NULL, use a single core.

Consider using multiple cores when number of replications R is a large value.

seed Random seed.

Numeric. Smallest possible time interval to allow.

Details

See Total(), Direct(), and Indirect() for more details.

Monte Carlo Method:

Let θ be $\operatorname{vec}(\Phi)$, that is, the elements of the Φ matrix in vector form sorted column-wise. Let $\hat{\theta}$ be $\operatorname{vec}(\hat{\Phi})$. Based on the asymptotic properties of maximum likelihood estimators, we can assume that estimators are normally distributed around the population parameters.

$$\hat{oldsymbol{ heta}} \sim \mathcal{N}\left(oldsymbol{ heta}, \mathbb{V}\left(\hat{oldsymbol{ heta}}
ight)
ight)$$

Using this distributional assumption, a sampling distribution of $\hat{\theta}$ which we refer to as $\hat{\theta}^*$ can be generated by replacing the population parameters with sample estimates, that is,

$$\hat{oldsymbol{ heta}}^* \sim \mathcal{N}\left(\hat{oldsymbol{ heta}}, \hat{\mathbb{V}}\left(\hat{oldsymbol{ heta}}
ight)
ight).$$

Let $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ be a parameter that is a function of the estimated parameters. A sampling distribution of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$, which we refer to as $\mathbf{g}\left(\hat{\boldsymbol{\theta}}^*\right)$, can be generated by using the simulated estimates to calculate \mathbf{g} . The standard deviations of the simulated estimates are the standard errors. Percentiles corresponding to $100\left(1-\alpha\right)\%$ are the confidence intervals.

Value

Returns an object of class ctmedmc which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("MCMed").

output A list with length of length(delta_t).

Each element in the output list has the following elements:

est A vector of total, direct, and indirect effects.

thetahatstar A matrix of Monte Carlo total, direct, and indirect effects.

Author(s)

Ivan Jacob Agaloos Pesigan

56 MCMed

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

```
set.seed(42)
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
vcov_phi_vec <- matrix(</pre>
  data = c(
    0.00843, 0.00040, -0.00151,
    -0.00600, -0.00033, 0.00110,
    0.00324, 0.00020, -0.00061,
    0.00040, 0.00374, 0.00016,
    -0.00022, -0.00273, -0.00016,
    0.00009, 0.00150, 0.00012,
    -0.00151, 0.00016, 0.00389,
    0.00103, -0.00007, -0.00283,
    -0.00050, 0.00000, 0.00156,
    -0.00600, -0.00022, 0.00103,
    0.00644, 0.00031, -0.00119,
    -0.00374, -0.00021, 0.00070,
    -0.00033, -0.00273, -0.00007,
    0.00031, 0.00287, 0.00013,
    -0.00014, -0.00170, -0.00012,
    0.00110, -0.00016, -0.00283,
    -0.00119, 0.00013, 0.00297,
    0.00063, -0.00004, -0.00177,
```

```
0.00324, 0.00009, -0.00050,
   -0.00374, -0.00014, 0.00063,
   0.00495, 0.00024, -0.00093,
   0.00020, 0.00150, 0.00000,
   -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
   -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
   -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
# Specific time interval -----
MCMed(
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1,
 from = "x",
 to = "y",
 med = "m",
 R = 100L # use a large value for R in actual research
)
# Range of time intervals ------
mc <- MCMed(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1:5,
 from = "x",
 to = "y",
 med = "m",
 R = 100L # use a large value for R in actual research
)
plot(mc)
# Methods -----
# MCMed has a number of methods including
# print, summary, confint, and plot
print(mc)
summary(mc)
confint(mc, level = 0.95)
```

MCMedStd

Monte Carlo Sampling Distribution of Standardized Total, Direct, and Indirect Effects of X on Y Through M Over a Specific Time Interval or a Range of Time Intervals

Description

This function generates a Monte Carlo method sampling distribution of the standardized total, direct and indirect effects of the independent variable X on the dependent variable Y through mediator variables \mathbf{m} over a specific time interval Δt or a range of time intervals using the first-order stochastic differential equation model drift matrix $\mathbf{\Phi}$ and process noise covariance matrix $\mathbf{\Sigma}$.

Usage

```
MCMedStd(
    phi,
    sigma,
    vcov_theta,
    delta_t,
    from,
    to,
    med,
    R,
    test_phi = TRUE,
    ncores = NULL,
    seed = NULL,
    tol = 0.01
)
```

Arguments

phi	Numeric matrix. The drift matrix (Φ) . phi should have row and column names pertaining to the variables in the system.
sigma	Numeric matrix. The process noise covariance matrix (Σ) .
vcov_theta	Numeric matrix. The sampling variance-covariance matrix of $\operatorname{vec}\left(\mathbf{\Phi}\right)$ and $\operatorname{vech}\left(\mathbf{\Sigma}\right)$
delta_t	Numeric. Time interval (Δt).
from	Character string. Name of the independent variable X in phi.
to	Character string. Name of the dependent variable Y in phi.
med	Character vector. Name/s of the mediator variable/s in phi.
R	Positive integer. Number of replications.
test_phi	Logical. If test_phi = TRUE, the function tests the stability of the generated drift matrix Φ . If the test returns FALSE, the function generates a new drift matrix Φ and runs the test recursively until the test returns TRUE.
ncores	Positive integer. Number of cores to use. If ncores = NULL, use a single core. Consider using multiple cores when number of replications R is a large value.
seed	Random seed.
tol	Numeric. Smallest possible time interval to allow.

Details

See TotalStd(), DirectStd(), and IndirectStd() for more details.

Monte Carlo Method:

Let θ be a vector that combines $\operatorname{vec}(\Phi)$, that is, the elements of the Φ matrix in vector form sorted column-wise and $\operatorname{vech}(\Sigma)$, that is, the unique elements of the Σ matrix in vector form sorted column-wise. Let $\hat{\theta}$ be a vector that combines $\operatorname{vec}(\hat{\Phi})$ and $\operatorname{vech}(\hat{\Sigma})$. Based on the asymptotic properties of maximum likelihood estimators, we can assume that estimators are normally distributed around the population parameters.

$$\hat{oldsymbol{ heta}} \sim \mathcal{N}\left(oldsymbol{ heta}, \mathbb{V}\left(\hat{oldsymbol{ heta}}
ight)
ight)$$

Using this distributional assumption, a sampling distribution of $\hat{\theta}$ which we refer to as $\hat{\theta}^*$ can be generated by replacing the population parameters with sample estimates, that is,

$$\hat{oldsymbol{ heta}}^* \sim \mathcal{N}\left(\hat{oldsymbol{ heta}}, \hat{\mathbb{V}}\left(\hat{oldsymbol{ heta}}
ight)
ight).$$

Let $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ be a parameter that is a function of the estimated parameters. A sampling distribution of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$, which we refer to as $\mathbf{g}\left(\hat{\boldsymbol{\theta}}^*\right)$, can be generated by using the simulated estimates to calculate \mathbf{g} . The standard deviations of the simulated estimates are the standard errors. Percentiles corresponding to $100\left(1-\alpha\right)\%$ are the confidence intervals.

Value

Returns an object of class ctmedmc which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("MCMedStd").

output A list with length of length(delta_t).

Each element in the output list has the following elements:

est A vector of total, direct, and indirect effects.

thetahatstar A matrix of Monte Carlo total, direct, and indirect effects.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ),
 nrow = 3
)
colnames(phi) \leftarrow rownames(phi) \leftarrow c("x", "m", "y")
sigma <- matrix(</pre>
 data = c(
   0.24455556, 0.02201587, -0.05004762,
   0.02201587, 0.07067800, 0.01539456,
    -0.05004762, 0.01539456, 0.07553061
 ),
 nrow = 3
vcov_theta <- matrix(</pre>
 data = c(
    0.00843, 0.00040, -0.00151, -0.00600, -0.00033,
    0.00110, 0.00324, 0.00020, -0.00061, -0.00115,
    0.00011, 0.00015, 0.00001, -0.00002, -0.00001,
    0.00040, 0.00374, 0.00016, -0.00022, -0.00273,
    -0.00016, 0.00009, 0.00150, 0.00012, -0.00010,
    -0.00026, 0.00002, 0.00012, 0.00004, -0.00001,
    -0.00151, 0.00016, 0.00389, 0.00103, -0.00007,
    -0.00283, -0.00050, 0.00000, 0.00156, 0.00021,
    -0.00005, -0.00031, 0.00001, 0.00007, 0.00006,
    -0.00600, -0.00022, 0.00103, 0.00644, 0.00031,
    -0.00119, -0.00374, -0.00021, 0.00070, 0.00064,
    -0.00015, -0.00005, 0.00000, 0.00003, -0.00001,
    -0.00033, -0.00273, -0.00007, 0.00031, 0.00287,
    0.00013, -0.00014, -0.00170, -0.00012, 0.00006,
     0.00014, \ -0.00001, \ -0.00015, \ 0.00000, \ 0.00001, \\
    0.00110, -0.00016, -0.00283, -0.00119, 0.00013,
    0.00297, 0.00063, -0.00004, -0.00177, -0.00013,
    0.00005, 0.00017, -0.00002, -0.00008, 0.00001,
    0.00324, 0.00009, -0.00050, -0.00374, -0.00014,
    0.00063, 0.00495, 0.00024, -0.00093, -0.00020,
    0.00006, -0.00010, 0.00000, -0.00001, 0.00004,
    0.00020, 0.00150, 0.00000, -0.00021, -0.00170,
    -0.00004, 0.00024, 0.00214, 0.00012, -0.00002,
```

```
-0.00004, 0.00000, 0.00006, -0.00005, -0.00001,
   -0.00061, 0.00012, 0.00156, 0.00070, -0.00012,
   -0.00177, -0.00093, 0.00012, 0.00223, 0.00004,
   -0.00002, -0.00003, 0.00001, 0.00003, -0.00013,
   -0.00115, -0.00010, 0.00021, 0.00064, 0.00006,
   -0.00013, -0.00020, -0.00002, 0.00004, 0.00057,
   0.00001, -0.00009, 0.00000, 0.00000, 0.00001,
   0.00011, -0.00026, -0.00005, -0.00015, 0.00014,
   0.00005, 0.00006, -0.00004, -0.00002, 0.00001,
   0.00012, 0.00001, 0.00000, -0.00002, 0.00000,
   0.00015, 0.00002, -0.00031, -0.00005, -0.00001,
   0.00017, -0.00010, 0.00000, -0.00003, -0.00009,
   0.00001, 0.00014, 0.00000, 0.00000, -0.00005,
   0.00001, 0.00012, 0.00001, 0.00000, -0.00015,
   -0.00002, 0.00000, 0.00006, 0.00001, 0.00000,
   0.00000, 0.00000, 0.00010, 0.00001, 0.00000,
   -0.00002, 0.00004, 0.00007, 0.00003, 0.00000,
   -0.00008, -0.00001, -0.00005, 0.00003, 0.00000,
   -0.00002, 0.00000, 0.00001, 0.00005, 0.00001,
   -0.00001, -0.00001, 0.00006, -0.00001, 0.00001,
   0.00001, 0.00004, -0.00001, -0.00013, 0.00001,
   0.00000, -0.00005, 0.00000, 0.00001, 0.00012
 ),
 nrow = 15
)
# Specific time interval -------
MCMedStd(
 phi = phi,
 sigma = sigma,
 vcov_theta = vcov_theta,
 delta_t = 1,
 from = "x",
 to = "y",
 med = "m",
 R = 100L # use a large value for R in actual research
# Range of time intervals ------
mc <- MCMedStd(</pre>
 phi = phi,
 sigma = sigma,
 vcov_theta = vcov_theta,
 delta_t = 1:5,
 from = "x",
 to = "y",
 med = "m",
 R = 100L # use a large value for R in actual research
)
plot(mc)
# Methods -----
# MCMedStd has a number of methods including
```

62 MCPhi

```
# print, summary, confint, and plot
print(mc)
summary(mc)
confint(mc, level = 0.95)
```

MCPhi

Generate Random Drift Matrices Using the Monte Carlo Method

Description

This function generates random drift matrices Φ using the Monte Carlo method.

Usage

```
MCPhi(phi, vcov_phi_vec, R, test_phi = TRUE, ncores = NULL, seed = NULL)
```

Arguments

phi Numeric matrix. The drift matrix (Φ), phi should have row and column names

pertaining to the variables in the system.

vcov_phi_vec Numeric matrix. The sampling variance-covariance matrix of vec (Φ) .

R Positive integer. Number of replications.

test_phi Logical. If test_phi = TRUE, the function tests the stability of the generated

drift matrix Φ . If the test returns FALSE, the function generates a new drift

matrix Φ and runs the test recursively until the test returns TRUE.

ncores Positive integer. Number of cores to use. If ncores = NULL, use a single core.

Consider using multiple cores when number of replications R is a large value.

seed Random seed.

Details

Monte Carlo Method:

Let θ be $\operatorname{vec}(\Phi)$, that is, the elements of the Φ matrix in vector form sorted column-wise. Let $\hat{\theta}$ be $\operatorname{vec}(\hat{\Phi})$. Based on the asymptotic properties of maximum likelihood estimators, we can assume that estimators are normally distributed around the population parameters.

$$\hat{oldsymbol{ heta}} \sim \mathcal{N}\left(oldsymbol{ heta}, \mathbb{V}\left(\hat{oldsymbol{ heta}}
ight)
ight)$$

Using this distributional assumption, a sampling distribution of $\hat{\theta}$ which we refer to as $\hat{\theta}^*$ can be generated by replacing the population parameters with sample estimates, that is,

$$\hat{oldsymbol{ heta}}^* \sim \mathcal{N}\left(\hat{oldsymbol{ heta}}, \hat{\mathbb{V}}\left(\hat{oldsymbol{ heta}}
ight)
ight).$$

MCPhi 63

Value

Returns an object of class ctmedmc which is a list with the following elements:

```
call Function call.args Function arguments.fun Function used ("MCPhi").output A list simulated drift matrices.
```

Author(s)

Ivan Jacob Agaloos Pesigan

See Also

```
Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()
```

```
set.seed(42)
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
MCPhi(
  phi = phi,
  vcov_phi_vec = 0.1 * diag(9),
  R = 100L # use a large value for R in actual research
phi <- matrix(</pre>
  data = c(
    -6, 5.5, 0, 0,
    1.25, -2.5, 5.9, -7.3,
    0, 0, -6, 2.5,
    5, 0, 0, -6
  ),
  nrow = 4
)
colnames(phi) <- rownames(phi) <- paste0("y", 1:4)</pre>
MCPhi(
  phi = phi,
```

64 MCTotalCentral

```
vcov_phi_vec = 0.1 * diag(16),
R = 100L, # use a large value for R in actual research
test_phi = FALSE
)
```

MCTotalCentral

Monte Carlo Sampling Distribution of Total Effect Centrality Over a Specific Time Interval or a Range of Time Intervals

Description

This function generates a Monte Carlo method sampling distribution of the total effect centrality at a particular time interval Δt using the first-order stochastic differential equation model drift matrix Φ .

Usage

```
MCTotalCentral(
   phi,
   vcov_phi_vec,
   delta_t,
   R,
   test_phi = TRUE,
   ncores = NULL,
   seed = NULL,
   tol = 0.01
)
```

Arguments

tol

phi	Numeric matrix. The drift matrix (Φ) . phi should have row and column names pertaining to the variables in the system.
vcov_phi_vec	Numeric matrix. The sampling variance-covariance matrix of $\operatorname{vec}\left(\mathbf{\Phi}\right)$.
delta_t	Numeric. Time interval (Δt).
R	Positive integer. Number of replications.
test_phi	Logical. If test_phi = TRUE, the function tests the stability of the generated drift matrix Φ . If the test returns FALSE, the function generates a new drift matrix Φ and runs the test recursively until the test returns TRUE.
ncores	Positive integer. Number of cores to use. If ncores = NULL, use a single core. Consider using multiple cores when number of replications R is a large value.
seed	Random seed.

Numeric. Smallest possible time interval to allow.

MCTotalCentral 65

Details

See TotalCentral() for more details.

Monte Carlo Method:

Let θ be $\operatorname{vec}(\Phi)$, that is, the elements of the Φ matrix in vector form sorted column-wise. Let $\hat{\theta}$ be $\operatorname{vec}(\hat{\Phi})$. Based on the asymptotic properties of maximum likelihood estimators, we can assume that estimators are normally distributed around the population parameters.

$$\hat{oldsymbol{ heta}} \sim \mathcal{N}\left(oldsymbol{ heta}, \mathbb{V}\left(\hat{oldsymbol{ heta}}
ight)
ight)$$

Using this distributional assumption, a sampling distribution of $\hat{\theta}$ which we refer to as $\hat{\theta}^*$ can be generated by replacing the population parameters with sample estimates, that is,

$$\hat{oldsymbol{ heta}}^* \sim \mathcal{N}\left(\hat{oldsymbol{ heta}}, \hat{\mathbb{V}}\left(\hat{oldsymbol{ heta}}
ight)
ight).$$

Let $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$ be a parameter that is a function of the estimated parameters. A sampling distribution of $\mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$, which we refer to as $\mathbf{g}\left(\hat{\boldsymbol{\theta}}^*\right)$, can be generated by using the simulated estimates to calculate \mathbf{g} . The standard deviations of the simulated estimates are the standard errors. Percentiles corresponding to $100\left(1-\alpha\right)\%$ are the confidence intervals.

Value

Returns an object of class ctmedmc which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("MCTotalCentral").

output A list the length of which is equal to the length of delta_t.

Each element in the output list has the following elements:

est A vector of total effect centrality.

thetahatstar A matrix of Monte Carlo total effect centrality.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

66 MCTotalCentral

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

```
set.seed(42)
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
vcov_phi_vec <- matrix(</pre>
  data = c(
    0.00843, 0.00040, -0.00151,
    -0.00600, -0.00033, 0.00110,
    0.00324, 0.00020, -0.00061,
    0.00040, 0.00374, 0.00016,
    -0.00022, -0.00273, -0.00016,
    0.00009, 0.00150, 0.00012,
    -0.00151, 0.00016, 0.00389,
    0.00103, -0.00007, -0.00283,
    -0.00050, 0.00000, 0.00156,
    -0.00600, -0.00022, 0.00103,
    0.00644, 0.00031, -0.00119,
    -0.00374, -0.00021, 0.00070,
    -0.00033, -0.00273, -0.00007,
    0.00031, 0.00287, 0.00013,
    -0.00014, -0.00170, -0.00012,
    0.00110, -0.00016, -0.00283,
    -0.00119, 0.00013, 0.00297,
    0.00063, -0.00004, -0.00177,
    0.00324, 0.00009, -0.00050,
    -0.00374, -0.00014, 0.00063,
    0.00495, 0.00024, -0.00093,
    0.00020, 0.00150, 0.00000,
    -0.00021, -0.00170, -0.00004,
    0.00024, 0.00214, 0.00012,
    -0.00061, 0.00012, 0.00156,
    0.00070, -0.00012, -0.00177,
    -0.00093, 0.00012, 0.00223
  ),
  nrow = 9
)
```

Med 67

```
# Specific time interval ------
MCTotalCentral(
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1,
 R = 100L # use a large value for R in actual research
# Range of time intervals -----
mc <- MCTotalCentral(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1:5,
 R = 100L # use a large value for R in actual research
plot(mc)
# Methods ------
# MCTotalCentral has a number of methods including
# print, summary, confint, and plot
print(mc)
summary(mc)
confint(mc, level = 0.95)
plot(mc)
```

Med

Total, Direct, and Indirect Effects of X on Y Through M Over a Specific Time Interval or a Range of Time Intervals

Description

This function computes the total, direct, and indirect effects of the independent variable X on the dependent variable Y through mediator variables \mathbf{m} over a specific time interval Δt or a range of time intervals using the first-order stochastic differential equation model's drift matrix $\mathbf{\Phi}$.

Usage

```
Med(phi, delta_t, from, to, med, tol = 0.01)
```

Arguments

phi	Numeric matrix. The drift matrix (Φ) , phi should have row and column names pertaining to the variables in the system.
delta_t	Vector of positive numbers. Time interval (Δt) .
from	Character string. Name of the independent variable X in phi.
to	Character string. Name of the dependent variable Y in phi.
med	Character vector. Name/s of the mediator variable/s in phi.
tol	Numeric. Smallest possible time interval to allow.

68 Med

Details

See Total(), Direct(), and Indirect() for more details.

Linear Stochastic Differential Equation Model:

The measurement model is given by

$$\mathbf{y}_{i,t} = \mathbf{\nu} + \mathbf{\Lambda} oldsymbol{\eta}_{i,t} + oldsymbol{arepsilon}_{i,t}, \quad ext{with} \quad oldsymbol{arepsilon}_{i,t} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Theta}
ight)$$

where $\mathbf{y}_{i,t}$, $\eta_{i,t}$, and $\varepsilon_{i,t}$ are random variables and ν , Λ , and Θ are model parameters. $\mathbf{y}_{i,t}$ represents a vector of observed random variables, $\eta_{i,t}$ a vector of latent random variables, and $\varepsilon_{i,t}$ a vector of random measurement errors, at time t and individual i. ν denotes a vector of intercepts, Λ a matrix of factor loadings, and Θ the covariance matrix of ε .

An alternative representation of the measurement error is given by

$$oldsymbol{arepsilon}_{i.t} = oldsymbol{\Theta}^{rac{1}{2}} \mathbf{z}_{i.t}, \quad ext{with} \quad \mathbf{z}_{i.t} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{I}\right)$$

where $\mathbf{z}_{i,t}$ is a vector of independent standard normal random variables and $\left(\boldsymbol{\Theta}^{\frac{1}{2}}\right)' = \boldsymbol{\Theta}$. The dynamic structure is given by

$$\mathrm{d}\boldsymbol{\eta}_{i,t} = \left(\boldsymbol{\iota} + \boldsymbol{\Phi}\boldsymbol{\eta}_{i,t}\right) \mathrm{d}t + \boldsymbol{\Sigma}^{\frac{1}{2}} \mathrm{d}\mathbf{W}_{i,t}$$

where ι is a term which is unobserved and constant over time, Φ is the drift matrix which represents the rate of change of the solution in the absence of any random fluctuations, Σ is the matrix of volatility or randomness in the process, and $\mathrm{d}W$ is a Wiener process or Brownian motion, which represents random fluctuations.

Value

Returns an object of class ctmedmed which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("Med").

output A matrix of total, direct, and indirect effects.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

Med 69

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ),
 nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
# Specific time interval ------
Med(
 phi = phi,
 delta_t = 1,
 from = "x",
 to = "y",
 med = "m"
)
# Range of time intervals ------
med <- Med(</pre>
 phi = phi,
 delta_t = 1:30,
 from = "x",
 to = "y",
 med = "m"
)
plot(med)
# Methods ------
# Med has a number of methods including
# print, summary, and plot
med <- Med(
 phi = phi,
 delta_t = 1:5,
 from = "x",
 to = "y",
 med = "m"
print(med)
summary(med)
plot(med)
```

70 MedStd

MedStd	Standardized Total, Direct, and Indirect Effects of X on Y Through M
	Over a Specific Time Interval or a Range of Time Intervals

Description

This function computes the standardized total, direct, and indirect effects of the independent variable X on the dependent variable Y through mediator variables \mathbf{m} over a specific time interval Δt or a range of time intervals using the first-order stochastic differential equation model's drift matrix $\mathbf{\Phi}$ and process noise covariance matrix $\mathbf{\Sigma}$.

Usage

```
MedStd(phi, sigma, delta_t, from, to, med, tol = 0.01)
```

Arguments

phi	Numeric matrix. The drift matrix (Φ) , phi should have row and column names pertaining to the variables in the system.
sigma	Numeric matrix. The process noise covariance matrix (Σ) .
delta_t	Numeric. Time interval (Δt) .
from	Character string. Name of the independent variable X in phi.
to	Character string. Name of the dependent variable Y in phi.
med	Character vector. Name/s of the mediator variable/s in phi.
tol	Numeric. Smallest possible time interval to allow.

Details

```
See TotalStd(), DirectStd(), and IndirectStd() for more details.
```

Value

Returns an object of class ctmedmed which is a list with the following elements:

```
call Function call.args Function arguments.fun Function used ("MedStd").output A matrix of total, direct, and indirect effects.
```

Author(s)

Ivan Jacob Agaloos Pesigan

MedStd 71

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ),
 nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
sigma <- matrix(</pre>
 data = c(
   0.24455556, 0.02201587, -0.05004762,
   0.02201587, 0.07067800, 0.01539456,
   -0.05004762, 0.01539456, 0.07553061
 ),
 nrow = 3
)
# Specific time interval -------
MedStd(
 phi = phi,
 sigma = sigma,
 delta_t = 1,
 from = "x",
 to = "y",
 med = "m"
# Range of time intervals -------
med <- MedStd(</pre>
```

72 plot.ctmeddelta

```
phi = phi,
 sigma = sigma,
 delta_t = 1:30,
 from = "x",
 to = "y",
 med = "m"
)
plot(med)
# Methods ------
# MedStd has a number of methods including
# print, summary, and plot
med <- MedStd(</pre>
 phi = phi,
 sigma = sigma,
 delta_t = 1:5,
 from = "x",
 to = "y",
 med = "m"
print(med)
summary(med)
plot(med)
```

plot.ctmeddelta

Plot Method for an Object of Class ctmeddelta

Description

Plot Method for an Object of Class ctmeddelta

Usage

```
## S3 method for class 'ctmeddelta'
plot(x, alpha = 0.05, col = NULL, ...)
```

Arguments

x	Object of class ctmeddelta.
alpha	Numeric. Significance level
col	Character vector. Optional argument. Character vector of colors.
	Additional arguments.

Value

Displays plots of point estimates and confidence intervals.

plot.ctmeddelta 73

Author(s)

Ivan Jacob Agaloos Pesigan

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ),
 nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
vcov_phi_vec <- matrix(</pre>
 data = c(
   0.00843, 0.00040, -0.00151,
    -0.00600, -0.00033, 0.00110,
   0.00324, 0.00020, -0.00061,
   0.00040, 0.00374, 0.00016,
    -0.00022, -0.00273, -0.00016,
   0.00009, 0.00150, 0.00012,
   -0.00151, 0.00016, 0.00389,
   0.00103, -0.00007, -0.00283,
   -0.00050, 0.00000, 0.00156,
   -0.00600, -0.00022, 0.00103,
   0.00644, 0.00031, -0.00119,
   -0.00374, -0.00021, 0.00070,
   -0.00033, -0.00273, -0.00007,
   0.00031, 0.00287, 0.00013,
    -0.00014, -0.00170, -0.00012,
   0.00110, -0.00016, -0.00283,
    -0.00119, 0.00013, 0.00297,
   0.00063, -0.00004, -0.00177,
   0.00324, 0.00009, -0.00050,
   -0.00374, -0.00014, 0.00063,
   0.00495, 0.00024, -0.00093,
   0.00020, 0.00150, 0.00000,
   -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
   -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
    -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
# Range of time intervals -------
delta <- DeltaMed(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1:5,
```

74 plot.ctmedmc

```
from = "x",
  to = "y",
  med = "m"
)
plot(delta)
```

plot.ctmedmc

Plot Method for an Object of Class ctmedmc

Description

Plot Method for an Object of Class ctmedmc

Usage

```
## S3 method for class 'ctmedmc'
plot(x, alpha = 0.05, col = NULL, ...)
```

Arguments

X	Object of class ctmedmc.
alpha	Numeric. Significance level
col	Character vector. Optional argument. Character vector of colors.
	Additional arguments.

Value

Displays plots of point estimates and confidence intervals.

Author(s)

Ivan Jacob Agaloos Pesigan

```
set.seed(42)
phi <- matrix(
   data = c(
     -0.357, 0.771, -0.450,
     0.0, -0.511, 0.729,
     0, 0, -0.693
   ),
   nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")
vcov_phi_vec <- matrix(
   data = c(</pre>
```

plot.ctmedmed 75

```
0.00843, 0.00040, -0.00151,
   -0.00600, -0.00033, 0.00110,
   0.00324, 0.00020, -0.00061,
   0.00040, 0.00374, 0.00016,
   -0.00022, -0.00273, -0.00016,
   0.00009, 0.00150, 0.00012,
   -0.00151, 0.00016, 0.00389,
   0.00103, -0.00007, -0.00283,
   -0.00050, 0.00000, 0.00156,
   -0.00600, -0.00022, 0.00103,
   0.00644, 0.00031, -0.00119,
   -0.00374, -0.00021, 0.00070,
   -0.00033, -0.00273, -0.00007,
   0.00031, 0.00287, 0.00013,
   -0.00014, -0.00170, -0.00012,
   0.00110, -0.00016, -0.00283,
   -0.00119, 0.00013, 0.00297,
   0.00063, -0.00004, -0.00177,
   0.00324, 0.00009, -0.00050,
   -0.00374, -0.00014, 0.00063,
   0.00495, 0.00024, -0.00093,
   0.00020, 0.00150, 0.00000,
   -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
   -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
    -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
# Range of time intervals ------
mc <- MCMed(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1:5,
 from = "x",
 to = "y",
 med = "m"
 R = 100L # use a large value for R in actual research
plot(mc)
```

plot.ctmedmed

Plot Method for an Object of Class ctmedmed

Description

Plot Method for an Object of Class ctmedmed

76 plot.ctmedmed

Usage

```
## S3 method for class 'ctmedmed'
plot(x, col = NULL, legend_pos = "topright", ...)
```

Arguments

x Object of class ctmedmed.

col Character vector. Optional argument. Character vector of colors.

legend_pos Character vector. Optional argument. Legend position.

... Additional arguments.

Value

Displays plots of point estimates and confidence intervals.

Author(s)

Ivan Jacob Agaloos Pesigan

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ),
 nrow = 3
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
# Range of time intervals ------
med <- Med(
 phi = phi,
 delta_t = 1:5,
 from = "x",
 to = "y",
 med = "m"
)
plot(med)
```

plot.ctmedtraj 77

plot.ctmedtraj

Plot Method for an Object of Class ctmedtraj

Description

Plot Method for an Object of Class ctmedtraj

Usage

```
## S3 method for class 'ctmedtraj'
plot(x, legend_pos = "topright", total = TRUE, ...)
```

Arguments

x Object of class ctmedtraj.
 legend_pos Character vector. Optional argument. Legend position.
 total Logical. If total = TRUE, include the total effect trajectory. If total = FALSE, exclude the total effect trajectory.
 ... Additional arguments.

Value

Displays trajectory plots of the effects.

Author(s)

Ivan Jacob Agaloos Pesigan

```
phi <- matrix(
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
),
  nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")

traj <- Trajectory(
  mu0 = c(3, 3, -3),
    time = 150,
    phi = phi,
    med = "m"
)

plot(traj)</pre>
```

78 PosteriorBeta

PosteriorBeta	Posterior Sampling Distribution for the Elements of the Matrix of Lagged Coefficients Over a Specific Time Interval or a Range of Time Intervals

Description

This function generates a posterior sampling distribution for the elements of the matrix of lagged coefficients β over a specific time interval Δt or a range of time intervals using the first-order stochastic differential equation model drift matrix Φ .

Usage

```
PosteriorBeta(phi, delta_t, ncores = NULL, tol = 0.01)
```

Arguments

phi	Numeric matrix. The drift matrix (Φ) , phi should have row and column names pertaining to the variables in the system.
delta_t	Numeric. Time interval (Δt).
ncores	Positive integer. Number of cores to use. If ncores = NULL, use a single core. Consider using multiple cores when number of replications R is a large value.
tol	Numeric. Smallest possible time interval to allow.

Details

```
See Total().
```

Value

Returns an object of class ctmedmc which is a list with the following elements:

```
call Function call.
```

args Function arguments.

fun Function used ("PosteriorBeta").

output A list the length of which is equal to the length of delta_t.

Each element in the output list has the following elements:

est A vector of total, direct, and indirect effects.

thetahatstar A matrix of Monte Carlo total, direct, and indirect effects.

Author(s)

Ivan Jacob Agaloos Pesigan

PosteriorBeta 79

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), IndirectStd(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

```
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ).
  nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
vcov_phi_vec <- matrix(</pre>
  data = c(
    0.00843, 0.00040, -0.00151,
    -0.00600, -0.00033, 0.00110,
    0.00324, 0.00020, -0.00061,
    0.00040, 0.00374, 0.00016,
    -0.00022, -0.00273, -0.00016,
    0.00009, 0.00150, 0.00012,
    -0.00151, 0.00016, 0.00389,
    0.00103, -0.00007, -0.00283,
    -0.00050, 0.00000, 0.00156,
    -0.00600, -0.00022, 0.00103,
    0.00644, 0.00031, -0.00119,
    -0.00374, -0.00021, 0.00070,
    -0.00033, -0.00273, -0.00007,
    0.00031, 0.00287, 0.00013,
    -0.00014, -0.00170, -0.00012,
    0.00110, -0.00016, -0.00283,
    -0.00119, 0.00013, 0.00297,
    0.00063, -0.00004, -0.00177,
    0.00324, 0.00009, -0.00050,
```

80 PosteriorIndirectCentral

```
-0.00374, -0.00014, 0.00063,
   0.00495, 0.00024, -0.00093,
   0.00020, 0.00150, 0.00000,
   -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
   -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
   -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
phi <- MCPhi(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 R = 1000L
)$output
# Specific time interval ------
PosteriorBeta(
 phi = phi,
 delta_t = 1
)
# Range of time intervals ------
posterior <- PosteriorBeta(</pre>
 phi = phi,
 delta_t = 1:5
plot(posterior)
# Methods -----
# PosteriorBeta has a number of methods including
# print, summary, confint, and plot
print(posterior)
summary(posterior)
confint(posterior, level = 0.95)
plot(posterior)
```

PosteriorIndirectCentral

Posterior Distribution of the Indirect Effect Centrality Over a Specific Time Interval or a Range of Time Intervals

Description

This function generates a posterior distribution of the indirect effect centrality over a specific time interval Δt or a range of time intervals using the posterior distribution of the first-order stochastic differential equation model drift matrix Φ .

PosteriorIndirectCentral 81

Usage

PosteriorIndirectCentral(phi, delta_t, ncores = NULL, tol = 0.01)

Arguments

phi List of numeric matrices. Each element of the list is a sample from the posterior

distribution of the drift matrix (Φ). Each matrix should have row and column

names pertaining to the variables in the system.

delta_t Numeric. Time interval (Δt).

ncores Positive integer. Number of cores to use. If ncores = NULL, use a single core.

Consider using multiple cores when number of replications R is a large value.

tol Numeric. Smallest possible time interval to allow.

Details

See TotalCentral() for more details.

Value

Returns an object of class ctmedmc which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("PosteriorIndirectCentral").

output A list the length of which is equal to the length of delta_t.

Each element in the output list has the following elements:

est Mean of the posterior distribution of the total, direct, and indirect effects.

thetahatstar Posterior distribution of the total, direct, and indirect effects.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

82 PosteriorIndirectCentral

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ).
 nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
vcov_phi_vec <- matrix(</pre>
 data = c(
    0.00843, 0.00040, -0.00151,
    -0.00600, -0.00033, 0.00110,
   0.00324, 0.00020, -0.00061,
   0.00040, 0.00374, 0.00016,
    -0.00022, -0.00273, -0.00016,
   0.00009, 0.00150, 0.00012,
    -0.00151, 0.00016, 0.00389,
    0.00103, -0.00007, -0.00283,
    -0.00050, 0.00000, 0.00156,
    -0.00600, -0.00022, 0.00103,
    0.00644, 0.00031, -0.00119,
    -0.00374, -0.00021, 0.00070,
    -0.00033, -0.00273, -0.00007,
    0.00031, 0.00287, 0.00013,
    -0.00014, -0.00170, -0.00012,
    0.00110, -0.00016, -0.00283,
    -0.00119, 0.00013, 0.00297,
    0.00063, -0.00004, -0.00177,
    0.00324, 0.00009, -0.00050,
    -0.00374, -0.00014, 0.00063,
    0.00495, 0.00024, -0.00093,
    0.00020, 0.00150, 0.00000,
    -0.00021, -0.00170, -0.00004,
    0.00024, 0.00214, 0.00012,
    -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
    -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
phi <- MCPhi(</pre>
```

PosteriorMed 83

```
phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 R = 1000L
)$output
# Specific time interval -------
PosteriorIndirectCentral(
 phi = phi,
 delta_t = 1
)
# Range of time intervals ------
posterior <- PosteriorIndirectCentral(</pre>
 phi = phi,
 delta_t = 1:5
# Methods ------
# PosteriorIndirectCentral has a number of methods including
# print, summary, confint, and plot
print(posterior)
summary(posterior)
confint(posterior, level = 0.95)
plot(posterior)
```

PosteriorMed

Posterior Distribution of Total, Direct, and Indirect Effects of X on Y Through M Over a Specific Time Interval or a Range of Time Intervals

Description

This function generates a posterior distribution of the total, direct and indirect effects of the independent variable X on the dependent variable Y through mediator variables \mathbf{m} over a specific time interval Δt or a range of time intervals using the posterior distribution of the first-order stochastic differential equation model drift matrix Φ .

Usage

```
PosteriorMed(phi, delta_t, from, to, med, ncores = NULL, tol = 0.01)
```

Arguments

phi	List of numeric matrices. Each element of the list is a sample from the posterior distribution of the drift matrix (Φ). Each matrix should have row and column names pertaining to the variables in the system.	
delta_t	Numeric. Time interval (Δt).	
from	Character string. Name of the independent variable X in phi.	

84 PosteriorMed

to	Character string. Name of the dependent variable Y in phi.	
med	Character vector. Name/s of the mediator variable/s in phi.	
ncores	Positive integer. Number of cores to use. If ncores = NULL, use a single core. Consider using multiple cores when number of replications R is a large value.	
tol	Numeric. Smallest possible time interval to allow.	

Details

See Total(), Direct(), and Indirect() for more details.

Value

Returns an object of class ctmedmc which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("PosteriorMed").

output A list the length of which is equal to the length of delta_t.

Each element in the output list has the following elements:

est Mean of the posterior distribution of the total, direct, and indirect effects.

thetahatstar Posterior distribution of the total, direct, and indirect effects.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

```
Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()
```

PosteriorMed 85

```
phi <- matrix(</pre>
  data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
  ),
  nrow = 3
)
colnames(phi) \leftarrow rownames(phi) \leftarrow c("x", "m", "y")
vcov_phi_vec <- matrix(</pre>
  data = c(
    0.00843, 0.00040, -0.00151,
    -0.00600, -0.00033, 0.00110,
    0.00324, 0.00020, -0.00061,
    0.00040, 0.00374, 0.00016,
    -0.00022, -0.00273, -0.00016,
    0.00009, 0.00150, 0.00012,
    -0.00151, 0.00016, 0.00389,
    0.00103, -0.00007, -0.00283,
    -0.00050, 0.00000, 0.00156,
    -0.00600, -0.00022, 0.00103,
   0.00644, 0.00031, -0.00119,
   -0.00374, -0.00021, 0.00070,
    -0.00033, -0.00273, -0.00007,
   0.00031, 0.00287, 0.00013,
    -0.00014, -0.00170, -0.00012,
   0.00110, -0.00016, -0.00283,
    -0.00119, 0.00013, 0.00297,
    0.00063, -0.00004, -0.00177,
    0.00324, 0.00009, -0.00050,
    -0.00374, -0.00014, 0.00063,
    0.00495, 0.00024, -0.00093,
    0.00020, 0.00150, 0.00000,
    -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
    -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
   -0.00093, 0.00012, 0.00223
  ),
  nrow = 9
)
phi <- MCPhi(</pre>
  phi = phi,
  vcov_phi_vec = vcov_phi_vec,
  R = 1000L
)$output
# Specific time interval ------
PosteriorMed(
  phi = phi,
```

86 PosteriorPhi

```
delta_t = 1,
 from = "x",
 to = "y",
 med = "m"
)
# Range of time intervals ------
posterior <- PosteriorMed(</pre>
 phi = phi,
 delta_t = 1:5,
 from = "x",
 to = "y",
 med = "m"
# PosteriorMed has a number of methods including
# print, summary, confint, and plot
print(posterior)
summary(posterior)
confint(posterior, level = 0.95)
plot(posterior)
```

PosteriorPhi

Extract the Posterior Samples of the Drift Matrix

Description

The function extracts the posterior samples of the drift matrix from a fitted model from the ctsem::ctStanFit() function.

Usage

PosteriorPhi(object)

Arguments

object

Object of class ctStanFit. Output of the ctsem::ctStanFit() function.

Value

Returns an object of class ctmedposteriorphi which is a list drift matrices sampled from the posterior distribution.

Author(s)

Ivan Jacob Agaloos Pesigan

PosteriorTotalCentral 87

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd(), Trajectory()

PosteriorTotalCentral Posterior Distribution of the Total Effect Centrality Over a Specific Time Interval or a Range of Time Intervals

Description

This function generates a posterior distribution of the total effect centrality over a specific time interval Δt or a range of time intervals using the posterior distribution of the first-order stochastic differential equation model drift matrix Φ .

Usage

```
PosteriorTotalCentral(phi, delta_t, ncores = NULL, tol = 0.01)
```

Arguments

phi	List of numeric matrices. Each element of the list is a sample from the posterior distribution of the drift matrix (Φ). Each matrix should have row and column names pertaining to the variables in the system.
delta_t	Numeric. Time interval (Δt).
ncores	Positive integer. Number of cores to use. If ncores = NULL, use a single core. Consider using multiple cores when number of replications R is a large value.
tol	Numeric. Smallest possible time interval to allow.

Details

See TotalCentral() for more details.

Value

Returns an object of class ctmedmc which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("PosteriorTotalCentral").

output A list the length of which is equal to the length of delta_t.

Each element in the output list has the following elements:

est Mean of the posterior distribution of the total, direct, and indirect effects.

thetahatstar Posterior distribution of the total, direct, and indirect effects.

88 PosteriorTotalCentral

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

```
Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), IndirectStd(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), Total(), TotalCentral(), TotalStd(), Trajectory()
```

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ),
 nrow = 3
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
vcov_phi_vec <- matrix(</pre>
 data = c(
    0.00843, 0.00040, -0.00151,
    -0.00600, -0.00033, 0.00110,
   0.00324, 0.00020, -0.00061,
    0.00040, 0.00374, 0.00016,
    -0.00022, -0.00273, -0.00016,
    0.00009, 0.00150, 0.00012,
    -0.00151, 0.00016, 0.00389,
    0.00103, -0.00007, -0.00283,
    -0.00050, 0.00000, 0.00156,
    -0.00600, -0.00022, 0.00103,
    0.00644, 0.00031, -0.00119,
    -0.00374, -0.00021, 0.00070,
    -0.00033, -0.00273, -0.00007,
    0.00031, 0.00287, 0.00013,
    -0.00014, -0.00170, -0.00012,
    0.00110, -0.00016, -0.00283,
```

print.ctmeddelta 89

```
-0.00119, 0.00013, 0.00297,
   0.00063, -0.00004, -0.00177,
   0.00324, 0.00009, -0.00050,
   -0.00374, -0.00014, 0.00063,
   0.00495, 0.00024, -0.00093,
   0.00020, 0.00150, 0.00000,
   -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
   -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
   -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
phi <- MCPhi(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 R = 1000L
)$output
# Specific time interval -----
PosteriorTotalCentral(
 phi = phi,
 delta_t = 1
)
# Range of time intervals ------
posterior <- PosteriorTotalCentral(</pre>
 phi = phi,
 delta_t = 1:5
)
# Methods ------
# PosteriorTotalCentral has a number of methods including
# print, summary, confint, and plot
print(posterior)
summary(posterior)
confint(posterior, level = 0.95)
plot(posterior)
```

print.ctmeddelta

Print Method for Object of Class ctmeddelta

Description

Print Method for Object of Class ctmeddelta

90 print.ctmeddelta

Usage

```
## S3 method for class 'ctmeddelta'
print(x, alpha = 0.05, digits = 4, ...)
```

Arguments

```
x an object of class ctmeddelta. 
alpha Numeric vector. Significance level \alpha. 
digits Integer indicating the number of decimal places to display. 
... further arguments.
```

Value

Prints a list of matrices of time intervals, estimates, standard errors, test statistics, p-values, and confidence intervals.

Author(s)

Ivan Jacob Agaloos Pesigan

```
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
vcov_phi_vec <- matrix(</pre>
  data = c(
    0.00843, 0.00040, -0.00151,
    -0.00600, -0.00033, 0.00110,
    0.00324, 0.00020, -0.00061,
    0.00040, 0.00374, 0.00016,
    -0.00022, -0.00273, -0.00016,
    0.00009, 0.00150, 0.00012,
    -0.00151, 0.00016, 0.00389,
    0.00103, -0.00007, -0.00283,
    -0.00050, 0.00000, 0.00156,
    -0.00600, -0.00022, 0.00103,
    0.00644, 0.00031, -0.00119,
    -0.00374, -0.00021, 0.00070,
    -0.00033, -0.00273, -0.00007,
    0.00031, 0.00287, 0.00013,
    -0.00014, -0.00170, -0.00012,
    0.00110, -0.00016, -0.00283,
    -0.00119, 0.00013, 0.00297,
```

print.ctmedeffect 91

```
0.00063, -0.00004, -0.00177,
   0.00324, 0.00009, -0.00050,
   -0.00374, -0.00014, 0.00063,
   0.00495, 0.00024, -0.00093,
   0.00020, 0.00150, 0.00000,
   -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
   -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
   -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
# Specific time interval ------
delta <- DeltaMed(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1,
 from = x^*,
 to = "y",
 med = "m"
)
print(delta)
# Range of time intervals ------
delta <- DeltaMed(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1:5,
 from = "x",
 to = "y",
 med = "m"
print(delta)
```

print.ctmedeffect

Print Method for Object of Class ctmedeffect

Description

Print Method for Object of Class ctmedeffect

Usage

```
## S3 method for class 'ctmedeffect'
print(x, digits = 4, ...)
```

92 print.ctmedeffect

Arguments

x an object of class ctmedeffect.digits Integer indicating the number of decimal places to display.... further arguments.

Value

Prints the effects.

Author(s)

Ivan Jacob Agaloos Pesigan

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ),
 nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
delta_t <- 1
# Time Interval of One ------
## Total Effect ---------------
total_dt <- Total(</pre>
 phi = phi,
 delta_t = delta_t
print(total_dt)
## Direct Effect -------
direct_dt <- Direct(</pre>
 phi = phi,
 delta_t = delta_t,
 from = "x",
 to = "y",
 med = "m"
)
print(direct_dt)
## Indirect Effect ------
indirect_dt <- Indirect(</pre>
 phi = phi,
 delta_t = delta_t,
 from = "x",
 to = "y",
```

print.ctmedmc 93

```
med = "m"
)
print(indirect_dt)
```

print.ctmedmc

Print Method for Object of Class ctmedmc

Description

Print Method for Object of Class ctmedmc

Usage

```
## S3 method for class 'ctmedmc'
print(x, alpha = 0.05, digits = 4, ...)
```

Arguments

```
x an object of class ctmedmc. 
alpha Numeric vector. Significance level \alpha. 
digits Integer indicating the number of decimal places to display. 
... further arguments.
```

Value

Prints a list of matrices of time intervals, estimates, standard errors, number of Monte Carlo replications, and confidence intervals.

Author(s)

Ivan Jacob Agaloos Pesigan

```
set.seed(42)
phi <- matrix(
   data = c(
     -0.357, 0.771, -0.450,
     0.0, -0.511, 0.729,
     0, 0, -0.693
   ),
   nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")
vcov_phi_vec <- matrix(
   data = c(
     0.00843, 0.00040, -0.00151,</pre>
```

94 print.ctmedmc

```
-0.00600, -0.00033, 0.00110,
   0.00324, 0.00020, -0.00061,
   0.00040, 0.00374, 0.00016,
   -0.00022, -0.00273, -0.00016,
   0.00009, 0.00150, 0.00012,
   -0.00151, 0.00016, 0.00389,
   0.00103, -0.00007, -0.00283,
   -0.00050, 0.00000, 0.00156,
   -0.00600, -0.00022, 0.00103,
   0.00644, 0.00031, -0.00119,
   -0.00374, -0.00021, 0.00070,
   -0.00033, -0.00273, -0.00007,
   0.00031, 0.00287, 0.00013,
   -0.00014, -0.00170, -0.00012,
   0.00110, -0.00016, -0.00283,
   -0.00119, 0.00013, 0.00297,
   0.00063, -0.00004, -0.00177,
   0.00324, 0.00009, -0.00050,
   -0.00374, -0.00014, 0.00063,
   0.00495, 0.00024, -0.00093,
   0.00020, 0.00150, 0.00000,
   -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
   -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
   -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
# Specific time interval -------
mc <- MCMed(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1,
 from = "x",
 to = "y",
 med = "m"
 R = 100L # use a large value for R in actual research
print(mc)
# Range of time intervals ------
mc <- MCMed(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1:5,
 from = x^{*},
 to = "y",
 med = "m",
 R = 100L # use a large value for R in actual research
print(mc)
```

print.ctmedmcphi 95

print.ctmedmcphi

Print Method for Object of Class ctmedmcphi

Description

Print Method for Object of Class ctmedmcphi

Usage

```
## S3 method for class 'ctmedmcphi'
print(x, digits = 4, ...)
```

Arguments

```
x an object of class ctmedmcphi.digits Integer indicating the number of decimal places to display.... further arguments.
```

Value

Prints a list of drift matrices.

Author(s)

Ivan Jacob Agaloos Pesigan

```
set.seed(42)
phi <- matrix(
    data = c(
        -0.357, 0.771, -0.450,
        0.0, -0.511, 0.729,
        0, 0, -0.693
    ),
    nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")
mc <- MCPhi(
    phi = phi,
    vcov_phi_vec = 0.1 * diag(9),
    R = 100L # use a large value for R in actual research
)
print(mc)</pre>
```

96 print.ctmedmed

print.ctmedmed

Print Method for Object of Class ctmedmed

Description

Print Method for Object of Class ctmedmed

Usage

```
## S3 method for class 'ctmedmed'
print(x, digits = 4, ...)
```

Arguments

```
x an object of class ctmedmed.digits Integer indicating the number of decimal places to display.... further arguments.
```

Value

Prints a matrix of effects.

Author(s)

Ivan Jacob Agaloos Pesigan

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ),
 nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
# Specific time interval ------
med <- Med(</pre>
 phi = phi,
 delta_t = 1,
 from = "x",
 to = "y",
 med = "m"
print(med)
# Range of time intervals ------
```

print.ctmedtraj 97

```
med <- Med(
    phi = phi,
    delta_t = 1:5,
    from = "x",
    to = "y",
    med = "m"
)
print(med)</pre>
```

print.ctmedtraj

Print Method for Object of Class ctmedtraj

Description

Print Method for Object of Class ctmedtraj

Usage

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

Arguments

x an object of class ctmedtraj.... further arguments.

Value

Prints a data frame of simulated data.

Author(s)

Ivan Jacob Agaloos Pesigan

```
phi <- matrix(
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")
traj <- Trajectory(
  mu0 = c(3, 3, -3),</pre>
```

98 summary.ctmeddelta

```
time = 150,
phi = phi,
med = "m"
)
print(traj)
```

summary.ctmeddelta

Summary Method for an Object of Class ctmeddelta

Description

Summary Method for an Object of Class ctmeddelta

Usage

```
## S3 method for class 'ctmeddelta'
summary(object, alpha = 0.05, ...)
```

Arguments

object Object of class ctmeddelta. alpha Numeric vector. Significance level α additional arguments.

Value

Returns a data frame of effects, time intervals, estimates, standard errors, test statistics, p-values, and confidence intervals.

Author(s)

Ivan Jacob Agaloos Pesigan

```
phi <- matrix(
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")
vcov_phi_vec <- matrix(
  data = c(
    0.00843, 0.00040, -0.00151,</pre>
```

summary.ctmeddelta 99

```
-0.00600, -0.00033, 0.00110,
   0.00324, 0.00020, -0.00061,
   0.00040, 0.00374, 0.00016,
   -0.00022, -0.00273, -0.00016,
   0.00009, 0.00150, 0.00012,
   -0.00151, 0.00016, 0.00389,
   0.00103, -0.00007, -0.00283,
   -0.00050, 0.00000, 0.00156,
   -0.00600, -0.00022, 0.00103,
   0.00644, 0.00031, -0.00119,
   -0.00374, -0.00021, 0.00070,
   -0.00033, -0.00273, -0.00007,
   0.00031, 0.00287, 0.00013,
   -0.00014, -0.00170, -0.00012,
   0.00110, -0.00016, -0.00283,
   -0.00119, 0.00013, 0.00297,
   0.00063, -0.00004, -0.00177,
   0.00324, 0.00009, -0.00050,
   -0.00374, -0.00014, 0.00063,
   0.00495, 0.00024, -0.00093,
   0.00020, 0.00150, 0.00000,
   -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
   -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
   -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
# Specific time interval ------
delta <- DeltaMed(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1,
 from = "x",
 to = "y",
 med = "m"
summary(delta)
# Range of time intervals -----
delta <- DeltaMed(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1:5,
 from = "x",
 to = "y",
 med = "m"
)
summary(delta)
```

100 summary.ctmedmc

summary.ctmedmc

Summary Method for an Object of Class ctmedmc

Description

Summary Method for an Object of Class ctmedmc

Usage

```
## S3 method for class 'ctmedmc'
summary(object, alpha = 0.05, ...)
```

Arguments

```
object Object of class ctmedmc.  
alpha Numeric vector. Significance level \alpha.  
additional arguments.
```

Value

Returns a data frame of effects, time intervals, estimates, standard errors, number of Monte Carlo replications, and confidence intervals.

Author(s)

Ivan Jacob Agaloos Pesigan

```
set.seed(42)
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
vcov_phi_vec <- matrix(</pre>
  data = c(
    0.00843, 0.00040, -0.00151,
    -0.00600, -0.00033, 0.00110,
    0.00324, 0.00020, -0.00061,
    0.00040, 0.00374, 0.00016,
    -0.00022, -0.00273, -0.00016,
    0.00009, 0.00150, 0.00012,
    -0.00151, 0.00016, 0.00389,
    0.00103, -0.00007, -0.00283,
```

summary.ctmedmed 101

```
-0.00050, 0.00000, 0.00156,
   -0.00600, -0.00022, 0.00103,
   0.00644, 0.00031, -0.00119,
   -0.00374, -0.00021, 0.00070,
   -0.00033, -0.00273, -0.00007,
   0.00031, 0.00287, 0.00013,
   -0.00014, -0.00170, -0.00012,
   0.00110, -0.00016, -0.00283,
   -0.00119, 0.00013, 0.00297,
   0.00063, -0.00004, -0.00177,
   0.00324, 0.00009, -0.00050,
   -0.00374, -0.00014, 0.00063,
   0.00495, 0.00024, -0.00093,
   0.00020, 0.00150, 0.00000,
   -0.00021, -0.00170, -0.00004,
   0.00024, 0.00214, 0.00012,
   -0.00061, 0.00012, 0.00156,
   0.00070, -0.00012, -0.00177,
   -0.00093, 0.00012, 0.00223
 ),
 nrow = 9
)
# Specific time interval ------
mc <- MCMed(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1,
 from = "x",
 to = "y",
 med = "m",
 R = 100L # use a large value for R in actual research
)
summary(mc)
# Range of time intervals ------
mc <- MCMed(</pre>
 phi = phi,
 vcov_phi_vec = vcov_phi_vec,
 delta_t = 1:5,
 from = "x",
 to = "y",
 med = "m"
 R = 100L # use a large value for R in actual research
)
summary(mc)
```

102 summary.ctmedmed

Description

Summary Method for an Object of Class ctmedmed

Usage

```
## S3 method for class 'ctmedmed'
summary(object, digits = 4, ...)
```

Arguments

```
object an object of class ctmedmed.

digits Integer indicating the number of decimal places to display.

further arguments.
```

Value

Returns a matrix of effects.

Author(s)

Ivan Jacob Agaloos Pesigan

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ),
 nrow = 3
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
# Specific time interval ------
med <- Med(
 phi = phi,
 delta_t = 1,
 from = "x",
 to = "y",
 med = "m"
)
summary(med)
# Range of time intervals -----
med <- Med(
 phi = phi,
 delta_t = 1:5,
 from = "x",
 to = "y",
 med = "m"
```

```
)
summary(med)
```

```
summary.ctmedposteriorphi
```

Summary Method for Object of Class ctmedposteriorphi

Description

Summary Method for Object of Class ctmedposteriorphi

Usage

```
## S3 method for class 'ctmedposteriorphi'
summary(object, ...)
```

Arguments

```
object an object of class ctmedposteriorphi.
... further arguments.
```

Value

Returns a list of the posterior means (in matrix form) and covariance matrix.

Author(s)

Ivan Jacob Agaloos Pesigan

summary.ctmedtraj

Summary Method for an Object of Class ctmedtraj

Description

Summary Method for an Object of Class ctmedtraj

Usage

```
## S3 method for class 'ctmedtraj'
summary(object, ...)
```

Arguments

```
object an object of class ctmedtraj.
... further arguments.
```

104 Total

Value

Returns a data frame of simulated data.

Author(s)

Ivan Jacob Agaloos Pesigan

Examples

```
phi <- matrix(
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
),
  nrow = 3
)
colnames(phi) <- rownames(phi) <- c("x", "m", "y")

traj <- Trajectory(
  mu0 = c(3, 3, -3),
  time = 150,
  phi = phi,
  med = "m"
)
summary(traj)</pre>
```

Total

Total Effect Matrix Over a Specific Time Interval

Description

This function computes the total effects matrix over a specific time interval Δt using the first-order stochastic differential equation model's drift matrix Φ .

Usage

```
Total(phi, delta_t)
```

Arguments

Phi Numeric matrix. The drift matrix (Φ) , phi should have row and column names pertaining to the variables in the system.

delta_t Numeric. Time interval (Δt).

Total 105

Details

The total effect matrix over a specific time interval Δt is given by

$$Total_{\Delta t} = \exp(\Delta t \mathbf{\Phi})$$

where Φ denotes the drift matrix, and Δt the time interval.

Linear Stochastic Differential Equation Model:

The measurement model is given by

$$\mathbf{y}_{i,t} = \mathbf{\nu} + \mathbf{\Lambda} \boldsymbol{\eta}_{i,t} + \boldsymbol{arepsilon}_{i,t}, \quad ext{with} \quad \boldsymbol{arepsilon}_{i,t} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Theta}
ight)$$

where $\mathbf{y}_{i,t}$, $\boldsymbol{\eta}_{i,t}$, and $\boldsymbol{\varepsilon}_{i,t}$ are random variables and $\boldsymbol{\nu}$, $\boldsymbol{\Lambda}$, and $\boldsymbol{\Theta}$ are model parameters. $\mathbf{y}_{i,t}$ represents a vector of observed random variables, $\boldsymbol{\eta}_{i,t}$ a vector of latent random variables, and $\boldsymbol{\varepsilon}_{i,t}$ a vector of random measurement errors, at time t and individual i. $\boldsymbol{\nu}$ denotes a vector of intercepts, $\boldsymbol{\Lambda}$ a matrix of factor loadings, and $\boldsymbol{\Theta}$ the covariance matrix of $\boldsymbol{\varepsilon}$.

An alternative representation of the measurement error is given by

$$\boldsymbol{\varepsilon}_{i,t} = \boldsymbol{\Theta}^{\frac{1}{2}} \mathbf{z}_{i,t}, \quad \text{with} \quad \mathbf{z}_{i,t} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{I}\right)$$

where $\mathbf{z}_{i,t}$ is a vector of independent standard normal random variables and $\left(\Theta^{\frac{1}{2}}\right)\left(\Theta^{\frac{1}{2}}\right)' = \Theta$. The dynamic structure is given by

$$\mathrm{d}\boldsymbol{\eta}_{i,t} = \left(\boldsymbol{\iota} + \boldsymbol{\Phi}\boldsymbol{\eta}_{i,t}\right) \mathrm{d}t + \boldsymbol{\Sigma}^{\frac{1}{2}} \mathrm{d}\mathbf{W}_{i,t}$$

where ι is a term which is unobserved and constant over time, Φ is the drift matrix which represents the rate of change of the solution in the absence of any random fluctuations, Σ is the matrix of volatility or randomness in the process, and $\mathrm{d}W$ is a Wiener process or Brownian motion, which represents random fluctuations.

Value

Returns an object of class ctmedeffect which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("Total").

output The matrix of total effects.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

106 TotalCentral

See Also

Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), IndirectStd(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), TotalCentral(), TotalStd(), Trajectory()

Examples

```
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
)
colnames(phi) \leftarrow rownames(phi) \leftarrow c("x", "m", "y")
delta_t <- 1
Total(
  phi = phi,
  delta_t = delta_t
phi <- matrix(</pre>
  data = c(
    -6, 5.5, 0, 0,
    1.25, -2.5, 5.9, -7.3,
    0, 0, -6, 2.5,
    5, 0, 0, -6
  ),
  nrow = 4
colnames(phi) <- rownames(phi) <- paste0("y", 1:4)</pre>
Total(
  phi = phi,
  delta_t = delta_t
```

TotalCentral

Total Effect Centrality

Description

Total Effect Centrality

Usage

```
TotalCentral(phi, delta_t, tol = 0.01)
```

TotalCentral 107

Arguments

phi	Numeric matrix. The drift matrix (Φ) , phi should have row and column names pertaining to the variables in the system.
delta_t	Vector of positive numbers. Time interval (Δt) .
tol	Numeric. Smallest possible time interval to allow.

Details

The total effect centrality of a variable is the sum of the total effects of a variable on all other variables at a particular time interval.

Value

Returns an object of class ctmedmed which is a list with the following elements:

```
call Function call.
```

args Function arguments.

fun Function used ("TotalCentral").

output A matrix of total effect centrality.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

```
Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalStd(), Trajectory()
```

108 TotalStd

Examples

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ),
 nrow = 3
colnames(phi) <- rownames(phi) <- c("x", "m", "y")</pre>
# Specific time interval ------
TotalCentral(
 phi = phi,
 delta_t = 1
)
# Range of time intervals ------
total_central <- TotalCentral(</pre>
 phi = phi,
 delta_t = 1:30
plot(total_central)
# Methods ------
# TotalCentral has a number of methods including
# print, summary, and plot
total_central <- TotalCentral(</pre>
 phi = phi,
 delta_t = 1:5
print(total_central)
summary(total_central)
plot(total_central)
```

TotalStd

Standardized Total Effect Matrix Over a Specific Time Interval

Description

This function computes the standardized total effects matrix over a specific time interval Δt using the first-order stochastic differential equation model's drift matrix Φ and process noise covariance matrix Σ .

Usage

```
TotalStd(phi, sigma, delta_t)
```

TotalStd 109

Arguments

phi	Numeric matrix. The drift matrix (Φ). phi should have row and column names
	pertaining to the variables in the system.
sigma	Numeric matrix. The process noise covariance matrix (Σ) .
delta_t	Numeric. Time interval (Δt).

Details

The standardized total effect matrix over a specific time interval Δt is given by

$$\operatorname{Total}_{\Delta t}^{*} = \mathbf{S} \left(\exp \left(\Delta t \mathbf{\Phi} \right) \right) \mathbf{S}^{-1}$$

where Φ denotes the drift matrix, S a diagonal matrix with model-implied standard deviations on the diagonals and Δt the time interval.

Value

Returns an object of class ctmedeffect which is a list with the following elements:

call Function call.

args Function arguments.

fun Function used ("TotalStd").

output The matrix of total effects.

Author(s)

Ivan Jacob Agaloos Pesigan

References

Bollen, K. A. (1987). Total, direct, and indirect effects in structural equation models. Sociological Methodology, 17, 37. doi:10.2307/271028

Deboeck, P. R., & Preacher, K. J. (2015). No need to be discrete: A method for continuous time mediation analysis. Structural Equation Modeling: A Multidisciplinary Journal, 23 (1), 61–75. doi:10.1080/10705511.2014.973960

Ryan, O., & Hamaker, E. L. (2021). Time to intervene: A continuous-time approach to network analysis and centrality. Psychometrika, 87 (1), 214–252. doi:10.1007/s11336021097670

See Also

```
Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), Trajectory()
```

Trajectory Trajectory

Examples

```
phi <- matrix(</pre>
  data = c(
    -0.357, 0.771, -0.450,
    0.0, -0.511, 0.729,
    0, 0, -0.693
  ),
  nrow = 3
colnames(phi) \leftarrow rownames(phi) \leftarrow c("x", "m", "y")
sigma <- matrix(</pre>
  data = c(
    0.24455556, 0.02201587, -0.05004762,
    0.02201587, 0.07067800, 0.01539456,
    -0.05004762, 0.01539456, 0.07553061
  ),
  nrow = 3
)
delta_t <- 1
TotalStd(
  phi = phi,
  sigma = sigma,
  delta_t = delta_t
```

Trajectory

Simulate Trajectories of Variables

Description

This function simulates trajectories of variables without measurement error or process noise. Total corresponds to the total effect and Direct corresponds to the portion of the total effect where the indirect effect is removed.

Usage

```
Trajectory(mu0, time, phi, med)
```

Arguments

mu0	Numeric vector. Initial values of the variables.
time	Positive integer. Number of time points.
phi	Numeric matrix. The drift matrix (Φ) . phi should have row and column names pertaining to the variables in the system.
med	Character vector. Name/s of the mediator variable/s in phi.

Trajectory 111

Value

```
Returns an object of class ctmedtraj which is a list with the following elements:
```

call Function call.args Function arguments.fun Function used ("Trajectory").

output A data frame of simulated data.

See Also

```
Other Continuous Time Mediation Functions: DeltaBeta(), DeltaBetaStd(), DeltaIndirectCentral(), DeltaMed(), DeltaMedStd(), DeltaTotalCentral(), Direct(), DirectStd(), ExpCov(), ExpMean(), Indirect(), IndirectCentral(), IndirectStd(), MCBeta(), MCBetaStd(), MCIndirectCentral(), MCMed(), MCMedStd(), MCPhi(), MCTotalCentral(), Med(), MedStd(), PosteriorBeta(), PosteriorIndirectCentral(), PosteriorMed(), PosteriorPhi(), PosteriorTotalCentral(), Total(), TotalCentral(), TotalStd()
```

```
phi <- matrix(</pre>
 data = c(
   -0.357, 0.771, -0.450,
   0.0, -0.511, 0.729,
   0, 0, -0.693
 ).
 nrow = 3
)
colnames(phi) \leftarrow rownames(phi) \leftarrow c("x", "m", "y")
traj <- Trajectory(</pre>
 mu0 = c(3, 3, -3),
 time = 150,
 phi = phi,
 med = "m"
)
plot(traj)
# Methods ------
# Trajectory has a number of methods including
# print, summary, and plot
traj <- Trajectory(</pre>
 mu0 = c(3, 3, -3),
 time = 25,
 phi = phi,
 med = "m"
)
print(traj)
summary(traj)
plot(traj)
```

Index

* Continuous Time Mediation Functions	DeltaBeta, 6
DeltaBeta, 6	DeltaBetaStd, 10
DeltaBetaStd, 10	DeltaIndirectCentral, 14
DeltaIndirectCentral, 14	DeltaMed, 17
DeltaMed, 17	DeltaMedStd, 20
DeltaMedStd, 20	DeltaTotalCentral, 25
DeltaTotalCentral, 25	Direct, 28
Direct, 28	DirectStd, 30
DirectStd, 30	ExpCov, 33
ExpCov, 33	ExpMean, 34
ExpMean, 34	Indirect, 36
Indirect, 36	IndirectCentral, 39
IndirectCentral, 39	IndirectStd, 41
IndirectStd, 41	MCBeta, 43
MCBeta, 43	MCBetaStd, 46
MCBetaStd, 46	MCIndirectCentral, 50
MCIndirectCentral, 50	MCMed, 54
MCMed, 54	MCMedStd, 57
MCMedStd, 57	MCPhi, 62
MCPhi, 62	MCTotalCentral, 64
MCTotalCentral, 64	Med, 67
Med, 67	MedStd, 70
MedStd, 70	PosteriorBeta, 78
PosteriorBeta, 78	PosteriorIndirectCentral, 80
PosteriorIndirectCentral, 80	PosteriorMed, 83
PosteriorMed, 83	PosteriorPhi, 86
PosteriorPhi, 86	PosteriorTotalCentral, 87
PosteriorTotalCentral, 87	Total, 104
Total, 104	TotalCentral, 106
TotalCentral, 106	TotalStd, 108
TotalStd, 108	Trajectory, 110
Trajectory, 110	* delta
* beta	DeltaBeta, 6
DeltaBeta, 6	DeltaBetaStd, 10
DeltaBetaStd, 10	DeltaIndirectCentral, 14
MCBeta, 43	DeltaMed, 17
MCBetaStd, 46	DeltaMedStd, 20
PosteriorBeta, 78	DeltaTotalCentral, 25
* cTMed	* effects

INDEX 113

Direct, 28	TotalCentral, 106
DirectStd, 30	* path
Indirect, 36	DeltaMed, 17
<pre>IndirectCentral, 39</pre>	DeltaMedStd, 20
IndirectStd, 41	MCMed, 54
Med, 67	MCMedStd, 57
MedStd, 70	Med, 67
Total, 104	MedStd, 70
TotalCentral, 106	PosteriorMed, 83
TotalStd, 108	Trajectory, 110
Trajectory, 110	* posterior
* expectations	PosteriorBeta, 78
ExpCov, 33	PosteriorIndirectCentral, 80
ExpMean, 34	PosteriorMed, 83
* mc	PosteriorPhi,86
MCBeta, 43	PosteriorTotalCentral, 87
MCBetaStd, 46	
MCIndirectCentral, 50	confint.ctmeddelta,3
MCMed, 54	confint.ctmedmc, 4
MCMedStd, 57	ctsem::ctStanFit(), 86
MCPhi, 62	- 1
MCTotalCentral, 64	DeltaBeta, 6, 12, 15, 19, 23, 26, 30, 32, 34,
* methods	36, 38, 40, 42, 45, 48, 52, 56, 60, 63,
confint.ctmeddelta,3	66, 69, 71, 79, 82, 84, 87, 88, 106,
confint.ctmedmc, 4	107, 109, 111
plot.ctmeddelta,72	DeltaBetaStd, 8, 10, 15, 19, 23, 26, 30, 32,
plot.ctmedmc, 74	34, 36, 38, 40, 42, 45, 48, 52, 56, 60,
plot.ctmedmed, 75	63, 66, 69, 71, 79, 82, 84, 87, 88,
plot.ctmedtraj,77	106, 107, 109, 111
print.ctmeddelta, 89	DeltaIndirectCentral, 8, 12, 14, 19, 23, 26,
print.ctmedeffect, 91	30, 32, 34, 36, 38, 40, 42, 45, 48, 52,
print.ctmedmc, 93	56, 60, 63, 66, 69, 71, 79, 82, 84, 87,
print.ctmedmcphi, 95	88, 106, 107, 109, 111
print.ctmedmed, 96	DeltaMed, 8, 12, 15, 17, 23, 26, 30, 32, 34, 36,
print.ctmedtraj,97	38, 40, 42, 45, 48, 52, 56, 60, 63, 66,
summary.ctmeddelta, 98	69, 71, 79, 82, 84, 87, 88, 106, 107, 109, 111
summary.ctmedmc, 100	
summary.ctmedmed, 101	DeltaMedStd, 8, 12, 15, 19, 20, 26, 30, 32, 34, 36, 38, 40, 42, 45, 48, 52, 56, 60, 63,
summary.ctmedposteriorphi, 103	66, 69, 71, 79, 82, 84, 87, 88, 106,
summary.ctmedtraj, 103	107, 109, 111
* network	DeltaTotalCentral, 8, 12, 15, 19, 23, 25, 30,
DeltaIndirectCentral, 14	32, 34, 36, 38, 40, 42, 45, 48, 52, 56,
DeltaTotalCentral, 25	60, 63, 66, 69, 71, 79, 82, 84, 87, 88,
IndirectCentral, 39	106, 107, 109, 111
MCIndirectCentral, 50	Direct, 8, 12, 15, 19, 23, 26, 28, 32, 34, 36,
MCTotalCentral, 64	38, 40, 42, 45, 48, 52, 56, 60, 63, 66,
PosteriorIndirectCentral, 80	69, 71, 79, 82, 84, 87, 88, 106, 107,
PosteriorTotalCentral, 87	109, 111
I OU CCI IOI IO CAICCIICI AI. U/	107, 111

114 INDEX

```
Direct(), 18, 55, 68, 84
                                                         MCMedStd, 8, 12, 15, 19, 23, 26, 30, 32, 34, 36,
DirectStd, 8, 12, 15, 19, 23, 26, 30, 30, 34,
                                                                   38, 40, 42, 45, 48, 52, 56, 57, 63, 66,
                                                                   69, 71, 79, 82, 84, 87, 88, 106, 107,
          36, 38, 40, 42, 45, 48, 52, 56, 60, 63,
                                                                   109, 111
          66, 69, 71, 79, 82, 84, 87, 88, 106,
          107, 109, 111
                                                         MCPhi, 8, 12, 15, 19, 23, 26, 30, 32, 34, 36, 38,
                                                                   40, 42, 45, 48, 52, 56, 60, 62, 66, 69,
DirectStd(), 21, 59, 70
                                                                   71, 79, 82, 84, 87, 88, 106, 107, 109,
ExpCov, 8, 12, 15, 19, 23, 26, 30, 32, 33, 36,
                                                         MCTotalCentral, 8, 12, 15, 19, 23, 26, 30, 32,
          38, 40, 42, 45, 48, 52, 56, 60, 63, 66,
                                                                   34, 36, 38, 40, 42, 45, 48, 52, 56, 60,
          69, 71, 79, 82, 84, 87, 88, 106, 107,
                                                                   63, 64, 69, 71, 79, 82, 84, 87, 88,
          109.111
                                                                   106, 107, 109, 111
ExpMean, 8, 12, 15, 19, 23, 26, 30, 32, 34, 34,
                                                         Med, 8, 12, 15, 19, 23, 26, 30, 32, 34, 36, 38,
          38, 40, 42, 45, 48, 52, 56, 60, 63, 66,
                                                                   40, 42, 45, 48, 52, 56, 60, 63, 66, 67,
          69, 71, 79, 82, 84, 87, 88, 106, 107,
                                                                   71, 79, 82, 84, 87, 88, 106, 107, 109,
          109, 111
                                                         MedStd, 8, 12, 15, 19, 23, 26, 30, 32, 34, 36,
Indirect, 8, 12, 15, 19, 23, 26, 30, 32, 34, 36,
                                                                   38, 40, 42, 45, 48, 52, 56, 60, 63, 66,
          36, 40, 42, 45, 48, 52, 56, 60, 63, 66,
                                                                   69, 70, 79, 82, 84, 87, 88, 106, 107,
          69, 71, 79, 82, 84, 87, 88, 106, 107,
                                                                   109, 111
          109.111
Indirect(), 18, 55, 68, 84
                                                         plot.ctmeddelta, 72
IndirectCentral, 8, 12, 15, 19, 23, 26, 30,
                                                         plot.ctmedmc, 74
          32, 34, 36, 38, 39, 42, 45, 48, 52, 56,
                                                         plot.ctmedmed, 75
          60, 63, 66, 69, 71, 79, 82, 84, 87, 88,
                                                         plot.ctmedtraj, 77
          106, 107, 109, 111
                                                         PosteriorBeta, 8, 12, 15, 19, 23, 26, 30, 32,
IndirectCentral(), 14, 51
                                                                   34, 36, 38, 40, 42, 45, 48, 52, 56, 60,
IndirectStd, 8, 12, 15, 19, 23, 26, 30, 32, 34,
                                                                   63, 66, 69, 71, 78, 82, 84, 87, 88,
          36, 38, 40, 41, 45, 48, 52, 56, 60, 63,
                                                                   106, 107, 109, 111
          66, 69, 71, 79, 82, 84, 87, 88, 106,
                                                         PosteriorIndirectCentral, 8, 12, 15, 19,
          107, 109, 111
                                                                   23, 26, 30, 32, 34, 36, 38, 40, 42, 45,
IndirectStd(), 21, 59, 70
                                                                   48, 52, 56, 60, 63, 66, 69, 71, 79, 80,
                                                                   84, 87, 88, 106, 107, 109, 111
MCBeta, 8, 12, 15, 19, 23, 26, 30, 32, 34, 36,
                                                         PosteriorMed, 8, 12, 15, 19, 23, 26, 30, 32,
          38, 40, 42, 43, 48, 52, 56, 60, 63, 66,
                                                                   34, 36, 38, 40, 42, 45, 48, 52, 56, 60,
          69, 71, 79, 82, 84, 87, 88, 106, 107,
                                                                   63, 66, 69, 71, 79, 82, 83, 87, 88,
          109, 111
                                                                   106, 107, 109, 111
MCBetaStd, 8, 12, 15, 19, 23, 26, 30, 32, 34,
                                                         PosteriorPhi, 8, 12, 15, 19, 23, 26, 30, 32,
          36, 38, 40, 42, 45, 46, 52, 56, 60, 63,
                                                                   34, 36, 38, 40, 42, 45, 48, 52, 56, 60,
          66, 69, 71, 79, 82, 84, 87, 88, 106,
                                                                   63, 66, 69, 71, 79, 82, 84, 86, 88,
          107, 109, 111
                                                                   106, 107, 109, 111
MCIndirectCentral, 8, 12, 15, 19, 23, 26, 30,
                                                         PosteriorTotalCentral, 8, 12, 15, 19, 23,
          32, 34, 36, 38, 40, 42, 45, 48, 50, 56,
                                                                   26, 30, 32, 34, 36, 38, 40, 42, 45, 48,
          60, 63, 66, 69, 71, 79, 82, 84, 87, 88,
                                                                   52, 56, 60, 63, 66, 69, 71, 79, 82, 84,
          106, 107, 109, 111
                                                                   87, 87, 106, 107, 109, 111
MCMed, 8, 12, 15, 19, 23, 26, 30, 32, 34, 36, 38,
                                                         print.ctmeddelta, 89
          40, 42, 45, 48, 52, 54, 60, 63, 66, 69,
                                                         print.ctmedeffect, 91
          71, 79, 82, 84, 87, 88, 106, 107, 109,
                                                         print.ctmedmc, 93
          111
                                                         print.ctmedmcphi, 95
```

INDEX 115

```
print.ctmedmed, 96
print.ctmedtraj, 97
summary.ctmeddelta,98
summary.ctmedmc, 100
summary.ctmedmed, 101
summary.ctmedposteriorphi, 103
summary.ctmedtraj, 103
Total, 8, 12, 15, 19, 23, 26, 30, 32, 34, 36, 38,
         40, 42, 45, 48, 52, 56, 60, 63, 66, 69,
         71, 79, 82, 84, 87, 88, 104, 107, 109,
         111
Total(), 7, 18, 44, 55, 68, 78, 84
TotalCentral, 8, 12, 15, 19, 23, 26, 30, 32,
         34, 36, 38, 40, 42, 45, 48, 52, 56, 60,
         63, 66, 69, 71, 79, 82, 84, 87, 88,
         106, 106, 109, 111
TotalCentral(), 25, 65, 81, 87
TotalStd, 8, 12, 15, 19, 23, 26, 30, 32, 34, 36,
         38, 40, 42, 45, 48, 52, 56, 60, 63, 66,
         69, 71, 79, 82, 84, 87, 88, 106, 107,
         108, 111
TotalStd(), 10, 21, 47, 59, 70
Trajectory, 8, 12, 15, 19, 23, 26, 30, 32, 34,
         36, 38, 40, 42, 45, 48, 52, 56, 60, 63,
         66, 69, 71, 79, 82, 84, 87, 88, 106,
         107, 109, 110
```