Package 'MultiRR'

October 12, 2022

Type Package

Title Bias, Precision, and Power for Multi-Level Random Regressions

Version 1.1

Date 2015-10-21

Author Yimen G. Araya-Ajoy

Maintainer Yimen G. Araya-Ajoy <yimencr@gmail.com>

Depends

Description Calculates bias, precision, and power for multi-level random regressions. Random regressions are types of hierarchical models in which data are structured in groups and (regression) coefficients can vary by groups. Tools to estimate model performance are designed mostly for scenarios where (regression) coefficients vary at just one level. 'MultiRR' provides simulation and analytical tools (based on 'lme4') to study model performance for random regressions that vary at more than one level (multi-level random regressions), allowing researchers to determine optimal sampling designs.

License GPL-2

Imports MASS, lme4

NeedsCompilation no

Repository CRAN

Date/Publication 2015-10-21 11:36:43

R topics documented:

MultiRR-package *Simulation Package for Multi-level random regressions*

Description

Calculates bias, precision, and power for multi-level random regressions. Random regressions are types of hierarchical models in which data are structured in groups and (regression) coefficients can vary by groups. Tools to estimate model performance are designed mostly for scenarios where (regression) coefficients vary at just one level. 'MultiRR' provides simulation and analytical tools (based on 'lme4') to study model performance for random regressions that vary at more than one level (multi-level random regressions), allowing researchers to determine optimal sampling designs.

Details

Package:	MultiRR
Type:	Package
Version:	1.0
Date:	2015-05-11
License:	GPL -2

Use the function `Sim.MultiRR` to simulate n data sets, then use the function `Anal.MultiRR` to perform a multi-level random regression to n simulated data sets. You can view the results using the function `Summary` or `Plot.Sim`, estimate bias using the function `Bias`, imprecision using the function `Imprecision`, and power using the function `Power`.

Author(s)

Yimen Araya: <yimencr@gmail.com>

References

Araya-Ajoy Y.G., Mathot, K. J., Dingemanse N. J. (2015) An approach to estimate short-term, long-term, and reaction norm repeatability. Methods in Ecology and Evolution.

Examples

```
#Example 1: Balanced sampling design.
#Define sample sizes.
n.ind <-c(40, 50) ##Numbers of individuals to simulate.
SeriesPerInd <- c(4, 5) ##Number of series per individual to simulate.
ObsPerLevel <- 2 ##Number of observations per level in the environmental gradient.

#Number of simulated data sets, use at least 10.
n.sim=3

#Define the environmetal gradient.
```

```
EnvGradient <- c(-0.5, 0.5)

#Define the population level parameters.
PopInt <- 0 ##Population level intercept.
PopSlope <- -0.5 ##Population level slope.

#Define individual level parameters
VCVInd <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define series level parameters
VCVSeries <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define the residual variance.
ResVar <- 0.4

#Simulate the data sets.
sim.data <- Sim.MultiRR(n.ind=n.ind, SeriesPerInd=SeriesPerInd,
ObsPerLevel=ObsPerLevel, EnvGradient=EnvGradient, PopInt=PopInt, PopSlope=PopSlope,
VCVInd=VCVInd, VCVSeries=VCVSeries, ResVar=ResVar, n.sim=3)

#Analyze the simulated data sets. This may take a while.
ressim <- Anal.MultiRR(sim.data)

#Summarize the results of the multi-level random regressions.
Summary(ressim)

#Estimate bias.
Bias(ressim)

#Estiamte imprecision.
Imprecision(ressim)

#Estimate power.
Power(ressim)

#Example 2: Unbalanced sampling design.
#Define sample sizes.
n.ind <-40 ##Numbers of individuals to simulate.
SeriesPerInd <- 4 ##Number of series per individual to simulate.
ObsPerLevel <- 2 ##Number of observations per level in the environmental gradient.

#Define the proportion of individuals that were sampled in all the series.
#All individuals were assayed at least once, 0.9 of individuals twice...

prop.ind<-c(1, 0.9, 0.8, 0.7)

#Define the total number of observations
n.obs=300

#Number of simulated data sets, use at least 10.
n.sim=3

#Define the environmetal gradient.
```

```
EnvGradient <- c(-0.5, 0.5)

#Define the population level parameters.
PopInt <- 0 ##Population level intercept.
PopSlope <- -0.5 ##Population level slope.

#Define the individual level parameters.
VCVInd <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define the series level parameters.
VCVSeries <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define the residual variance.
ResVar <- 0.4

#Simulate the data.
sim.data <- Sim.MultiRR(n.ind=n.ind, SeriesPerInd=SeriesPerInd, ObsPerLevel=ObsPerLevel,
EnvGradient=EnvGradient, PopInt=PopInt, PopSlope=PopSlope, VCVInd= VCVInd, VCVSeries=VCVSeries,
ResVar=ResVar, n.sim=n.sim, unbalanced=TRUE, prop.ind=c(1, 0.9, 0.8, 0.7),
complete.observations=FALSE, n.obs=n.obs)

#Analyze simulated data sets. This may take a while.
ressim <- Anal.MultiRR(sim.data)

#Summarize the results of the multi-level random regressions.
Summary(ressim)

#Estimate bias.
Bias(ressim)

#Estiamte imprecision.
Imprecision(ressim)

#Estimate power.
Power(ressim)
```

`Anal.MultiRR` *Fits a multilevel random regression to n simulated data frames.*

Description

Performs multilevel random regressions to objects created with the function Sim.MultiRR.

Usage

```
Anal.MultiRR(x)
```

Arguments

`x`	Object created with the function sim.MultiRR.

Value

A list of results from the multi-level random regression for n simulated data sets.

Author(s)

Yimen Araya

References

Araya-Ajoy Y.G., Mathot, K. J., Dingemanse N. J. (2015) An approach to estimate short-term, long-term, and reaction norm repeatability. Methods in Ecology and Evolution.

See Also

Sim.MultiRR

Examples

```
#Example 1: Balanced sampling design.
#Define sample sizes.
n.ind <-c(40, 50) ##Numbers of individuals to simulate.
SeriesPerInd <- c(4, 5) ##Number of series per individual to simulate.
ObsPerLevel <- 2 ##Number of observations per level in the environmental gradient.

#Number of simulated data sets, use at least 10.
n.sim=3

#Define the environmetal gradient.
EnvGradient <- c(-0.5, 0.5)

#Define the population level parameters.
PopInt <- 0 ##Population level intercept.
PopSlope <- -0.5 ##Population level slope.

#Define individual level parameters
VCVInd <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define series level parameters
VCVSeries <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define the residual variance.
ResVar <- 0.4

#Simulate the data sets.
sim.data <- Sim.MultiRR(n.ind=n.ind, SeriesPerInd=SeriesPerInd,
ObsPerLevel=ObsPerLevel, EnvGradient=EnvGradient, PopInt=PopInt, PopSlope=PopSlope,
VCVInd=VCVInd, VCVSeries=VCVSeries, ResVar=ResVar, n.sim=3)

#Analyze the simulated data sets. This may take a while.
ressim <- Anal.MultiRR(sim.data)

#Summarize the results of the multi-level random regressions.
```

```
Summary(ressim)

#Estimate bias.
Bias(ressim)

#Estiamte imprecision.
Imprecision(ressim)

#Estimate power.
Power(ressim)

#Example 2: Unbalanced sampling design.
#Define sample sizes.
n.ind <-40 ##Numbers of individuals to simulate.
SeriesPerInd <- 4 ##Number of series per individual to simulate.
ObsPerLevel <- 2 ##Number of observations per level in the environmental gradient.

#Define the proportion of individuals that were sampled in all the series.
#All individuals were assayed at least once, 0.9 of individuals twice...

prop.ind<-c(1, 0.9, 0.8, 0.7)

#Define the total number of observations
n.obs=300

#Number of simulated data sets, use at least 10.
n.sim=3

#Define the environmetal gradient.
EnvGradient <- c(-0.5, 0.5)

#Define the population level parameters.
PopInt <- 0 ##Population level intercept.
PopSlope <- -0.5 ##Population level slope.

#Define the individual level parameters.
VCVInd <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define the series level parameters.
VCVSeries <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define the residual variance.
ResVar <- 0.4

#Simulate the data.
sim.data <- Sim.MultiRR(n.ind=n.ind, SeriesPerInd=SeriesPerInd, ObsPerLevel=ObsPerLevel,
EnvGradient=EnvGradient, PopInt=PopInt, PopSlope=PopSlope, VCVInd= VCVInd, VCVSeries=VCVSeries,
ResVar=ResVar, n.sim=n.sim, unbalanced=TRUE, prop.ind=c(1, 0.9, 0.8, 0.7),
complete.observations=FALSE, n.obs=n.obs)

#Analyze simulated data sets. This may take a while.
ressim <- Anal.MultiRR(sim.data)
```

```
#Summarize the results of the multi-level random regressions.
Summary(ressim)

#Estimate bias.
Bias(ressim)

#Estiamte imprecision.
Imprecision(ressim)

#Estimate power.
Power(ressim)
```

`Bias` *Estimates bias for n number of multi-level random regression models performed to n simulated data sets.*

Description

Calculates bias for an object created with anal.MultiRR.

Usage

```
Bias(x)
```

Arguments

`x`	Object created with Anal.MultiRR.

Value

A list of data frames with the bias and relative bias for all the estimated variance components and repeatabilities.

Author(s)

Yimen Araya

References

Araya-Ajoy Y.G., Mathot, K. J., Dingemanse N. J. (2015) An approach to estimate short-term, long-term, and reaction norm repeatability. Methods in Ecology and Evolution.

See Also

`Sim.MultiRR`, `Anal.MultiRR`

Examples

```
#Example 1: Balanced sampling design.
#Define sample sizes.
n.ind <-c(40, 50) ##Numbers of individuals to simulate.
SeriesPerInd <- c(4, 5) ##Number of series per individual to simulate.
ObsPerLevel <- 2 ##Number of observations per level in the environmental gradient.

#Number of simulated data sets, use at least 10.
n.sim=3

#Define the environmetal gradient.
EnvGradient <- c(-0.5, 0.5)

#Define the population level parameters.
PopInt <- 0 ##Population level intercept.
PopSlope <- -0.5 ##Population level slope.

#Define individual level parameters
VCVInd <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define series level parameters
VCVSeries <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define the residual variance.
ResVar <- 0.4

#Simulate the data sets.
sim.data <- Sim.MultiRR(n.ind=n.ind, SeriesPerInd=SeriesPerInd,
ObsPerLevel=ObsPerLevel, EnvGradient=EnvGradient, PopInt=PopInt, PopSlope=PopSlope,
VCVInd=VCVInd, VCVSeries=VCVSeries, ResVar=ResVar, n.sim=3)

#Analyze the simulated data sets. This may take a while.
ressim <- Anal.MultiRR(sim.data)

#Summarize the results of the multi-level random regressions.
Summary(ressim)

#Estimate bias.
Bias(ressim)

#Estiamte imprecision.
Imprecision(ressim)

#Estimate power.
Power(ressim)
```

`Imprecision` *Calculates imprecision for n multi-level random regressions perfromed to n simulated dats sets.*

Description

Calculates imprecision for an object created with anal.MultiRR.

Usage

```
Imprecision(x)
```

Arguments

x	Object created with Anal.MultiRR.

Value

A list of data frames with the imprecision for all the estimated variance components and repeatabilities.

Author(s)

Yimen Araya

References

Araya-Ajoy Y.G., Mathot, K. J., Dingemanse N. J. (2015) An approach to estimate short-term, long-term, and reaction norm repeatability. Methods in Ecology and Evolution.

See Also

Sim.MultiRR, Anal.MultiRR

Examples

```
#Example 1: Balanced sampling design.
#Define sample sizes.
n.ind <-c(40, 50) ##Numbers of individuals to simulate.
SeriesPerInd <- c(4, 5) ##Number of series per individual to simulate.
ObsPerLevel <- 2 ##Number of observations per level in the environmental gradient.

#Number of simulated data sets, use at least 10.
n.sim=3

#Define the environmetal gradient.
EnvGradient <- c(-0.5, 0.5)

#Define the population level parameters.
PopInt <- 0 ##Population level intercept.
PopSlope <- -0.5 ##Population level slope.

#Define individual level parameters
VCVInd <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define series level parameters
```

```
VCVSeries <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define the residual variance.
ResVar <- 0.4

#Simulate the data sets.
sim.data <- Sim.MultiRR(n.ind=n.ind, SeriesPerInd=SeriesPerInd,
ObsPerLevel=ObsPerLevel, EnvGradient=EnvGradient, PopInt=PopInt, PopSlope=PopSlope,
VCVInd=VCVInd, VCVSeries=VCVSeries, ResVar=ResVar, n.sim=3)

#Analyze the simulated data sets. This may take a while.
ressim <- Anal.MultiRR(sim.data)

#Summarize the results of the multi-level random regressions.
Summary(ressim)

#Estimate bias.
Bias(ressim)

#Estiamte imprecision.
Imprecision(ressim)

#Estimate power.
Power(ressim)
```

`Plot.Sim` *Density plots for each variance component.*

Description

Plots distributions of the estimated parameters from the simulations.

Usage

```
Plot.Sim(x)
```

Arguments

`x`	Object created with Anal.MultiRR.

Note

Better use with only one combination of parameters (i.e., number of individuals and series per individual.

Author(s)

Yimen Araya

References

Araya-Ajoy Y.G., Mathot, K. J., Dingemanse N. J. (2015) An approach to estimate short-term, long-term, and reaction norm repeatability. Methods in Ecology and Evolution.

See Also

Sim.MultiRR, Anal.MultiRR, Summary

Examples

```
#Example: Unbalanced sampling design.
#Define sample sizes.
n.ind <-40 ##Numbers of individuals to simulate.
SeriesPerInd <- 4 ##Number of series per individual to simulate.
ObsPerLevel <- 2 ##Number of observations per level in the environmental gradient.

#Define the proportion of individuals that were sampled in all the series.
#All individuals were assayed at least once, 0.9 of individuals twice...

prop.ind<-c(1, 0.9, 0.8, 0.7)

#Define the total number of observations
n.obs=300

#Number of simulated data sets, use at least 10.
n.sim=3

#Define the environmetal gradient.
EnvGradient <- c(-0.5, 0.5)

#Define the population level parameters.
PopInt <- 0 ##Population level intercept.
PopSlope <- -0.5 ##Population level slope.

#Define the individual level parameters.
VCVInd <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define the series level parameters.
VCVSeries <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define the residual variance.
ResVar <- 0.4

#Simulate the data.
sim.data <- Sim.MultiRR(n.ind=n.ind, SeriesPerInd=SeriesPerInd, ObsPerLevel=ObsPerLevel,
EnvGradient=EnvGradient, PopInt=PopInt, PopSlope=PopSlope, VCVInd= VCVInd, VCVSeries=VCVSeries,
ResVar=ResVar, n.sim=n.sim, unbalanced=TRUE, prop.ind=c(1, 0.9, 0.8, 0.7),
complete.observations=FALSE, n.obs=n.obs)

#Analyze simulated data sets. This may take a while.
ressim <- Anal.MultiRR(sim.data)
```

```
#Summarize the results of the multi-level random regressions.
Summary(ressim)

#Estimate bias.
Bias(ressim)

#Estiamte imprecision.
Imprecision(ressim)

#Estimate power.
Power(ressim)

#Plot the simulations
Plot.Sim(ressim)
```

`Power` *Estimates power to detect significant among-individual variation in intercepts and slopes.*

Description

Power analysis for object created with anal.MultiRR.

Usage

```
Power(x)
```

Arguments

`x`	Object created with Anal.MultiRR.

Value

A list of data frames with the power to detect among-individual variation in intercepts and slopes in a multi-level random regression model.

Author(s)

Yimen Araya

References

Araya-Ajoy Y.G., Mathot, K. J., Dingemanse N. J. (2015) An approach to estimate short-term, long-term, and reaction norm repeatability. Methods in Ecology and Evolution.

See Also

`Sim.MultiRR`, `Anal.MultiRR`

Examples

```
#Example 1: Balanced sampling design.
#Define sample sizes.
n.ind <-c(40, 50) ##Numbers of individuals to simulate.
SeriesPerInd <- c(4, 5) ##Number of series per individual to simulate.
ObsPerLevel <- 2 ##Number of observations per level in the environmental gradient.

#Number of simulated data sets, use at least 10.
n.sim=3

#Define the environmetal gradient.
EnvGradient <- c(-0.5, 0.5)

#Define the population level parameters.
PopInt <- 0 ##Population level intercept.
PopSlope <- -0.5 ##Population level slope.

#Define individual level parameters
VCVInd <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define series level parameters
VCVSeries <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define the residual variance.
ResVar <- 0.4

#Simulate the data sets.
sim.data <- Sim.MultiRR(n.ind=n.ind, SeriesPerInd=SeriesPerInd,
ObsPerLevel=ObsPerLevel, EnvGradient=EnvGradient, PopInt=PopInt, PopSlope=PopSlope,
VCVInd=VCVInd, VCVSeries=VCVSeries, ResVar=ResVar, n.sim=3)

#Analyze the simulated data sets. This may take a while.
ressim <- Anal.MultiRR(sim.data)

#Summarize the results of the multi-level random regressions.
Summary(ressim)

#Estimate bias.
Bias(ressim)

#Estiamte imprecision.
Imprecision(ressim)

#Estimate power.
Power(ressim)
```

`Sim.MultiRR` *Simulate data setes to be analyzed by a multi-level random regression.*

Description

Simulate n data sets to be analyzed with a multi-level random regression.

Usage

```
Sim.MultiRR(n.ind, SeriesPerInd, ObsPerLevel, EnvGradient, PopInt,
PopSlope, VCVInd, VCVSeries, ResVar, n.sim, unbalanced = FALSE,
prop.ind, complete.observations = TRUE, n.obs)
```

Arguments

`n.ind`	A vector consisting of the total individuals sampled.
`SeriesPerInd`	A vector consisting of the number of series sampled for each individual.
`ObsPerLevel`	The number of observations per series in each level of the environment.
`EnvGradient`	A vector consisting of the levels in the environmental gradient.
`PopInt`	Population level intercept.
`PopSlope`	Population level slope.
`VCVInd`	A positive definite variance covariance matrix of dimensions 2 X 2, defining the among-individual variance in intercepts and slopes in the diagonals and their covariance in the off diagonals.
`VCVSeries`	A positive definite variance covariance matrix of dimensions 2 X 2, defining the among-series variance in intercepts and slopes in the diagonals and their covariance in the off diagonals.
`ResVar`	Residual variance
`n.sim`	Number of data sets to simulate.
`unbalanced`	Optional argument determining whether not all the individuals were assayed the same number of series. The default is "FALSE".
`prop.ind`	When unbalanced = "TRUE", A vector that has the same length as the number of series per individual, with the proportion of individuals measured n times. All individuals should have been measured once (1,.,.,.).
`complete.observations`	Optional argument determining whether all the levels were assayed the same number of times. The default is "TRUE".
`n.obs`	The total number of observations, if complete.observartions = "FALSE".

Value

A list of data sets to be analyzed by Anal.MultiRR.

Author(s)

Yimen Araya

References

Araya-Ajoy Y.G., Mathot, K. J., Dingemanse N. J. (2015) An approach to estimate short-term, long-term, and reaction norm repeatability. Methods in Ecology and Evolution.

See Also

Anal.MultiRR

Examples

```
#Example 1: Balanced sampling design.
#Define sample sizes.
n.ind <-c(40, 50) ##Numbers of individuals to simulate.
SeriesPerInd <- c(4, 5) ##Number of series per individual to simulate.
ObsPerLevel <- 2 ##Number of observations per level in the environmental gradient.

#Number of simulated data sets, use at least 10.
n.sim=3

#Define the environmetal gradient.
EnvGradient <- c(-0.5, 0.5)

#Define the population level parameters.
PopInt <- 0 ##Population level intercept.
PopSlope <- -0.5 ##Population level slope.

#Define individual level parameters
VCVInd <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define series level parameters
VCVSeries <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define the residual variance.
ResVar <- 0.4

#Simulate the data sets.
sim.data <- Sim.MultiRR(n.ind=n.ind, SeriesPerInd=SeriesPerInd,
ObsPerLevel=ObsPerLevel, EnvGradient=EnvGradient, PopInt=PopInt, PopSlope=PopSlope,
VCVInd=VCVInd, VCVSeries=VCVSeries, ResVar=ResVar, n.sim=3)

#Analyze the simulated data sets. This may take a while.
ressim <- Anal.MultiRR(sim.data)

#Summarize the results of the multi-level random regressions.
Summary(ressim)

#Estimate bias.
Bias(ressim)

#Estiamte imprecision.
Imprecision(ressim)
```

```
#Estimate power.
Power(ressim)

#Example 2: Unbalanced sampling desing.
#Define sample sizes.
n.ind <-40 ##Numbers of individuals to simulate.
SeriesPerInd <- 4 ##Number of series per individual to simulate.
ObsPerLevel <- 2 ##Number of observations per level in the environmental gradient.

#Define the proportion of individuals that were sampled in all the series.
#All individuals were assayed at least once, 0.9 of individuals twice...

prop.ind<-c(1, 0.9, 0.8, 0.7)

#Define the total number of observations
n.obs=300

#Number of simulated data sets, use at least 10.
n.sim=3

#Define the environmetal gradient.
EnvGradient <- c(-0.5, 0.5)

#Define the population level parameters.
PopInt <- 0 ##Population level intercept.
PopSlope <- -0.5 ##Population level slope.

#Define the individual level parameters.
VCVInd <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define the series level parameters.
VCVSeries <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define the residual variance.
ResVar <- 0.4

#Simulate the data.
sim.data <- Sim.MultiRR(n.ind=n.ind, SeriesPerInd=SeriesPerInd, ObsPerLevel=ObsPerLevel,
EnvGradient=EnvGradient, PopInt=PopInt, PopSlope=PopSlope, VCVInd= VCVInd, VCVSeries=VCVSeries,
ResVar=ResVar, n.sim=n.sim, unbalanced=TRUE, prop.ind=c(1, 0.9, 0.8, 0.7),
complete.observations=FALSE, n.obs=n.obs)

#Analyze simulated data sets. This may take a while.
ressim <- Anal.MultiRR(sim.data)

#Summarize the results of the multi-level random regressions.
Summary(ressim)

#Estimate bias.
Bias(ressim)

#Estiamte imprecision.
Imprecision(ressim)
```

```
#Estimate power.
Power(ressim)
```

`Summary` *Summary of the results of the multi-level random regressions performed to n simulated data sets.*

Description

Summary of object created with anal.MultiRR.

Usage

```
Summary(x)
```

Arguments

`x`	Object created with anal.MultiRR.

Value

A list of data frames with a summary of the model estimates for all the simulations.

Author(s)

Yimen Araya

References

Araya-Ajoy Y.G., Mathot, K. J., Dingemanse N. J. (2015) An approach to estimate short-term, long-term, and reaction norm repeatability. Methods in Ecology and Evolution.

See Also

`Sim.MultiRR`, `Anal.MultiRR`

Examples

```
#Example 1: Balanced sampling design.
#Define sample sizes.
n.ind <-c(40, 50) ##Numbers of individuals to simulate.
SeriesPerInd <- c(4, 5) ##Number of series per individual to simulate.
ObsPerLevel <- 2 ##Number of observations per level in the environmental gradient.

#Number of simulated data sets, use at least 10.
n.sim=3

#Define the environmetal gradient.
```

```
EnvGradient <- c(-0.5, 0.5)

#Define the population level parameters.
PopInt <- 0 ##Population level intercept.
PopSlope <- -0.5 ##Population level slope.

#Define individual level parameters
VCVInd <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define series level parameters
VCVSeries <-matrix(c(0.3, 0.15, 0.15, 0.3),2,2) ##Creates a variance-covariance matrix.

#Define the residual variance.
ResVar <- 0.4

#Simulate the data sets.
sim.data <- Sim.MultiRR(n.ind=n.ind, SeriesPerInd=SeriesPerInd,
ObsPerLevel=ObsPerLevel, EnvGradient=EnvGradient, PopInt=PopInt, PopSlope=PopSlope,
VCVInd=VCVInd, VCVSeries=VCVSeries, ResVar=ResVar, n.sim=3)

#Analyze the simulated data sets. This may take a while.
ressim <- Anal.MultiRR(sim.data)

#Summarize the results of the multi-level random regressions.
Summary(ressim)

#Estimate bias.
Bias(ressim)

#Estiamte imprecision.
Imprecision(ressim)

#Estimate power.
Power(ressim)
```

Index