

Package ‘iccbeta’

January 28, 2019

Type Package

Title Multilevel Model Intraclass Correlation for Slope Heterogeneity

Version 1.2.0

Description A function and vignettes for computing an intraclass correlation described in Aguinis & Culpepper (2015) <doi:10.1177/1094428114563618>. This package quantifies the share of variance in a dependent variable that is attributed to group heterogeneity in slopes.

Depends R (>= 3.4.0)

Imports Rcpp, lme4, stats, methods

LinkingTo Rcpp (>= 1.0.0), RcppArmadillo (>= 0.9.200)

Suggests RLRsim, testthat, covr

License GPL (>= 2)

URL <https://github.com/tmsalab/iccbeta>

BugReports <https://github.com/tmsalab/iccbeta/issues>

RoxygenNote 6.1.1

Encoding UTF-8

NeedsCompilation yes

Author Steven Andrew Culpepper [aut, cph, cre]
(<<https://orcid.org/0000-0003-4226-6176>>),
Herman Aguinis [aut, cph] (<<https://orcid.org/0000-0002-3485-9484>>)

Maintainer Steven Andrew Culpepper <sculpepp@illinois.edu>

Repository CRAN

Date/Publication 2019-01-28 21:50:02 UTC

R topics documented:

iccbeta-package	2
Hofmann	3
icc_beta	5
simICCdata	8

Index**10**

iccbeta-package	<i>iccbeta: Multilevel Model Intraclass Correlation for Slope Heterogeneity</i>
-----------------	---

Description

A function and vignettes for computing an intraclass correlation described in Aguinis & Culpepper (2015) <doi:10.1177/1094428114563618>. This package quantifies the share of variance in a dependent variable that is attributed to group heterogeneity in slopes.

Author(s)

Maintainer: Steven Andrew Culpepper <sculpepp@illinois.edu> (0000-0003-4226-6176) [copyright holder]

Authors:

- Herman Aguinis <haguinis@gwu.edu> (0000-0002-3485-9484) [copyright holder]

References

Aguinis, H., & Culpepper, S.A. (2015). An expanded decision making procedure for examining cross-level interaction effects with multilevel modeling. *Organizational Research Methods*. Available at: <http://www.hermanaguinis.com/pubs.html>

See Also

Useful links:

- <https://github.com/tmsalab/iccbeta>
- Report bugs at <https://github.com/tmsalab/iccbeta/issues>

Examples

```
## Not run:

if(requireNamespace("lme4") && requireNamespace("RLRsim")){
# Simulated Data Example
data(simICCData)
library('lme4')

# computing icca
vy <- var(simICCData$Y)
lmm0 <- lmer(Y ~ (1|l2id), data = simICCData, REML = FALSE)
VarCorr(lmm0)$l2id[1,1]/vy

# Create simICCData2
grp_means = aggregate(simICCData[c('X1','X2')], simICCData['l2id'],mean)
colnames(grp_means)[2:3] = c('m_X1','m_X2')
```

```

simICCdata2 = merge(simICCdata,grp_means,by='l2id')

# Estimating random slopes model
lmm1 <- lmer(Y ~ I(X1-m_X1) + I(X2-m_X2) + (I(X1-m_X1) + I(X2-m_X2) | l2id),
            data = simICCdata2, REML = FALSE)
X <- model.matrix(lmm1)
p <- ncol(X)
T1 <- VarCorr(lmm1)$l2id[1:p, 1:p]

# computing iccb
# Notice '+1' because icc_beta assumes l2ids are from 1 to 30.
icc_beta(X, simICCdata2$l2id + 1, T1, vy)$rho_beta

# Hofmann 2000 Example
data(Hofmann)
library('lme4')

# Random-Intercepts Model
lmmHofmann0 <- lmer(helping ~ (1|id), data = Hofmann)
vy_Hofmann <- var(Hofmann[, 'helping'])
# computing icca
VarCorr(lmmHofmann0)$id[1,1]/vy_Hofmann

# Estimating Group-Mean Centered Random Slopes Model, no level 2 variables
lmmHofmann1 <- lmer(helping ~ mood_grp_cent + (mood_grp_cent | id),
                  data = Hofmann, REML = FALSE)
X_Hofmann <- model.matrix(lmmHofmann1)
P <- ncol(X_Hofmann)
T1_Hofmann <- VarCorr(lmmHofmann1)$id[1:P, 1:P]
# computing iccb
icc_beta(X_Hofmann, Hofmann[, 'id'], T1_Hofmann, vy_Hofmann)$rho_beta

# Performing LR test
library('RLRsim')
lmmHofmann1a <- lmer(helping ~ mood_grp_cent + (1 | id),
                   data = Hofmann, REML = FALSE)
obs.LRT <- 2*(logLik(lmmHofmann1) - logLik(lmmHofmann1a))[1]
X <- getME(lmmHofmann1, "X")
Z <- t(as.matrix(getME(lmmHofmann1, "Zt")))
sim.LRT <- LRTSim(X, Z, 0, diag(ncol(Z)))
(pval <- mean(sim.LRT > obs.LRT))
} else {
  stop("Please install packages `RLRsim` and `lme4` to run the above example.")
}

## End(Not run)

```

Description

A multilevel dataset from Hofmann, Griffin, and Gavin (2000).

Usage

Hofmann

Format

A data frame with 1,000 observations and 7 variables.

`id` a numeric vector of group ids.

`helping` a numeric vector of the helping outcome variable construct.

`mood` a level 1 mood predictor.

`mood_grp_mn` a level 2 variable of the group mean of mood.

`cohesion` a level 2 covariate measuring cohesion.

`mood_grp_cent` group-mean centered mood predictor.

`mood_grd_cent` grand-mean centered mood predictor.

Source

Hofmann, D.A., Griffin, M.A., & Gavin, M.B. (2000). The application of hierarchical linear modeling to management research. In K.J. Klein, & S.W.J. Kozlowski (Eds.), *Multilevel theory, research, and methods in organizations: Foundations, extensions, and new directions* (pp. 467-511). Hoboken, NJ: Jossey-Bass.

References

Aguinis, H., & Culpepper, S.A. (2015). An expanded decision making procedure for examining cross-level interaction effects with multilevel modeling. *Organizational Research Methods*. Available at: <http://hermanaguinis.com/pubs.html>

See Also

[lmer](#), [model.matrix](#), [VarCorr](#), [LRTsim](#), [simICCdata](#)

Examples

```
## Not run:

if(requireNamespace("lme4") && requireNamespace("RLRsim")){
  data(Hofmann)
  library("lme4")

  # Random-Intercepts Model
  lmmHofmann0 = lmer(helping ~ (1|id), data = Hofmann)
  vy_Hofmann = var(Hofmann[, 'helping'])

  # Computing icca
```

```

VarCorr(lmmHofmann0)$id[1,1]/vy_Hofmann

# Estimating Group-Mean Centered Random Slopes Model, no level 2 variables
lmmHofmann1 <- lmer(helping ~ mood_grp_cent + (mood_grp_cent |id),
                    data = Hofmann, REML = FALSE)
X_Hofmann = model.matrix(lmmHofmann1)
P = ncol(X_Hofmann)
T1_Hofmann = VarCorr(lmmHofmann1)$id[1:P,1:P]

# Computing iccb
icc_beta(X_Hofmann, Hofmann[, 'id'], T1_Hofmann, vy_Hofmann)$rho_beta

# Performing LR test
# Need to install 'RLRsim' package
library("RLRsim")
lmmHofmann1a <- lmer(helping ~ mood_grp_cent + (1 | id),
                    data = Hofmann, REML = FALSE)
obs.LRT <- 2*(logLik(lmmHofmann1) - logLik(lmmHofmann1a))[1]
X <- getME(lmmHofmann1, "X")
Z <- t(as.matrix(getME(lmmHofmann1, "Zt")))
sim.LRT <- LRsim(X, Z, 0, diag(ncol(Z)))
(pval <- mean(sim.LRT > obs.LRT))
} else {
  stop("Please install packages `RLRsim` and `lme4` to run the above example.")
}

## End(Not run)

```

 icc_beta

Intraclass correlation used to assess variability of lower-order relationships across higher-order processes/units.

Description

A function and vignettes for computing the intraclass correlation described in Aguinis & Culpepper (2015). `iccbeta` quantifies the share of variance in an outcome variable that is attributed to heterogeneity in slopes due to higher-order processes/units.

Usage

```

icc_beta(x, ...)

## S3 method for class 'lmerMod'
icc_beta(x, ...)

## Default S3 method:
icc_beta(x, l2id, T, vy, ...)

```

Arguments

x	A lmer model object or a design matrix with no missing values.
...	Additional parameters...
l2id	A vector that identifies group membership. The vector must be coded as a sequence of integers from 1 to J, the number of groups.
T	A matrix of the estimated variance-covariance matrix of a lmer model fit.
vy	The variance of the outcome variable.

Value

A list with:

- J
- means
- XcpXc
- Nj
- rho_beta

Author(s)

Steven Andrew Culpepper

References

Aguinis, H., & Culpepper, S.A. (2015). An expanded decision making procedure for examining cross-level interaction effects with multilevel modeling. *Organizational Research Methods*. Available at: <http://hermanaguinis.com/pubs.html>

See Also

[lme4::lmer\(\)](#), [model.matrix\(\)](#), [lme4::VarCorr\(\)](#), [RLRsim::LRTsim\(\)](#), [iccbeta::Hofmann](#), and [iccbeta::simICCdata](#)

Examples

```
## Not run:

if(requireNamespace("lme4") && requireNamespace("RLRsim")){

## Example 1: Simulated Data Example from Aguinis & Culpepper (2015) ----
data(simICCdata)
library("lme4")

# Computing icca
vy <- var(simICCdata$Y)
lmm0 <- lmer(Y ~ (1 | l2id), data = simICCdata, REML = FALSE)
VarCorr(lmm0)$l2id[1, 1]/vy
```

```

# Create simICCdata2
grp_means = aggregate(simICCdata[c('X1', 'X2')], simICCdata['l2id'], mean)
colnames(grp_means)[2:3] = c('m_X1', 'm_X2')
simICCdata2 = merge(simICCdata, grp_means, by='l2id')

# Estimating random slopes model
lmm1 <- lmer(Y ~ I(X1 - m_X1) + I(X2 - m_X2) +
             (I(X1 - m_X1) + I(X2 - m_X2) | l2id),
             data = simICCdata2, REML = FALSE)

## iccbeta calculation on `lmer` object
icc_beta(lmm1)

## Manual specification of iccbeta

# Extract components from model.
X <- model.matrix(lmm1)
p <- ncol(X)
T1 <- VarCorr(lmm1)$l2id[1:p,1:p]

# Note: vy was computed under "icca"

# Computing iccb
# Notice '+1' because icc_beta assumes l2ids are from 1 to 30.
icc_beta(X, simICCdata2$l2id + 1, T1, vy)$rho_beta

## Example 2: Hofmann et al. (2000) ----

data(Hofmann)
library("lme4")

# Random-Intercepts Model
lmmHofmann0 = lmer(helping ~ (1|id), data = Hofmann)
vy_Hofmann = var(Hofmann[, 'helping'])

# Computing icca
VarCorr(lmmHofmann0)$id[1,1]/vy_Hofmann

# Estimating Group-Mean Centered Random Slopes Model, no level 2 variables
lmmHofmann1 <- lmer(helping ~ mood_grp_cent + (mood_grp_cent | id),
                   data = Hofmann, REML = FALSE)

## Automatic calculation of iccbeta using the lmer model
amod = icc_beta(lmmHofmann1)

## Manual calculation of iccbeta

X_Hofmann <- model.matrix(lmmHofmann1)
P <- ncol(X_Hofmann)
T1_Hofmann <- VarCorr(lmmHofmann1)$id[1:P,1:P]

# Computing iccb
bmod = icc_beta(X_Hofmann, Hofmann[, 'id'], T1_Hofmann, vy_Hofmann)$rho_beta

```

```

# Performing LR test
library("RLRsim")
lmmHofmann1a <- lmer(helping ~ mood_grp_cent + (1 |id),
                    data = Hofmann, REML = FALSE)
obs.LRT <- 2*(logLik(lmmHofmann1) - logLik(lmmHofmann1a))[1]
X <- getME(lmmHofmann1,"X")
Z <- t(as.matrix(getME(lmmHofmann1,"Zt")))
sim.LRT <- LRtsim(X, Z, 0, diag(ncol(Z)))
(pval <- mean(sim.LRT > obs.LRT))
} else {
  stop("Please install packages `RLRsim` and `lme4` to run the above example.")
}

## End(Not run)

```

simICCData

Simulated data example from Aguinis and Culpepper (2015).

Description

A simulated data example from Aguinis and Culpepper (2015) to demonstrate the `icc_beta` function for computing the proportion of variance in the outcome variable that is attributed to heterogeneity in slopes due to higher-order processes/units.

Usage

```
simICCData
```

Format

A data frame with 900 observations (i.e., 30 observations nested within 30 groups) on the following 6 variables.

- l1id A within group ID variable.
- l2id A group ID variable.
- one A column of 1's for the intercept.
- X1 A simulated level 1 predictor.
- X2 A simulated level 1 predictor.
- Y A simulated outcome variable.

Details

See Aguinis and Culpepper (2015) for the model used to simulate the dataset.

Source

Aguinis, H., & Culpepper, S.A. (2015). An expanded decision making procedure for examining cross-level interaction effects with multilevel modeling. *Organizational Research Methods*. Available at: <http://www.hermanaguinis.com/pubs.html>

See Also

[lmer](#), [model.matrix](#), [VarCorr](#), [LRTSim](#), [Hofmann](#)

Examples

```
## Not run:
data(simICCdata)
if(requireNamespace("lme4")){
  library("lme4")

  # computing icca
  vy <- var(simICCdata$Y)
  lmm0 <- lmer(Y ~ (1|l2id), data = simICCdata, REML = FALSE)
  VarCorr(lmm0)$l2id[1,1]/vy

  # Create simICCdata2
  grp_means = aggregate(simICCdata[c('X1','X2')], simICCdata['l2id'],mean)
  colnames(grp_means)[2:3] = c('m_X1','m_X2')
  simICCdata2 = merge(simICCdata, grp_means, by='l2id')

  # Estimating random slopes model
  lmm1 <- lmer(Y ~ I(X1-m_X1) + I(X2-m_X2) + (I(X1-m_X1) + I(X2-m_X2) | l2id),
              data = simICCdata2, REML = FALSE)
  X <- model.matrix(lmm1)
  p <- ncol(X)
  T1 <- VarCorr(lmm1) $l2id[1:p,1:p]
  # computing iccb
  # Notice '+1' because icc_beta assumes l2ids are from 1 to 30.
  icc_beta(X, simICCdata2$l2id+1, T1, vy)$rho_beta
} else {
  stop("Please install `lme4` to run the above example.")
}

## End(Not run)
```

Index

*Topic **datasets**

Hofmann, 3
simICCdata, 8

Hofmann, 3, 9

icc_beta, 5, 8
iccbeta (iccbeta-package), 2
iccbeta-package, 2
iccbeta::Hofmann, 6
iccbeta::simICCdata, 6

lme4::lmer(), 6
lme4::VarCorr(), 6
lmer, 4, 9
LRTsim, 4, 9

model.matrix, 4, 9
model.matrix(), 6

RLRsim::LRTsim(), 6

simICCdata, 4, 8

VarCorr, 4, 9