Package 'CorrMixed'

May 7, 2026

Type Package

Title Estimate Correlations Between Repeatedly Measured Endpoints (E.g., Reliability) Based on Linear Mixed-Effects Models

Version 1.1

Author Wim Van der Elst, Geert Molenberghs, Dieter Hilgers, & Nicole Heussen

Maintainer Wim Van der Elst <Wim.vanderelst@gmail.com>

Description In clinical practice and research settings in medicine and the behavioral sciences, it is often of interest to quantify the correlation of a continuous endpoint that was repeatedly measured (e.g., test-retest correlations, ICC, etc.). This package allows for estimating these correlations based on mixed-effects models. Part of this software has been developed using funding provided from the European Union's 7th Framework Programme for research, technological development and demonstration under Grant Agreement no 602552.

Depends nlme

Imports psych

License GPL (>= 2)

Repository CRAN

NeedsCompilation no

Date/Publication 2022-04-18 15:24:38 UTC

Contents

Example.Data — *An example dataset*

Description

`Example.Data` is a hypothetical dataset constructed to demonstrate some of the functions in the package. Data are provided for a hypothetical experiment in which a stimulus is provided under different experimental conditions. The outcome is a normally distributed variable. The entire experiment is repeated multiple times (cycle) in each patient.

Usage

```
data(Example.Data)
```

Format

A `data.frame` with 360 observations on 5 variables.

- `Id` The Subject identifier.
- `Cycle` The same experiment is repeated multiple times in a patient. Cycle indicates the order of these repeated experiments.
- `Condition` The experimental condition under which the outcome was measured.
- `Time` The time point at which the outcome was measured.
- `Outcome` A continuous outcome.

Explore.WS.Corr — *Explore within-subject correlations (reliabilities)*

Description

This function allows for exploring the within-subject (test-retest) correlation (R) structure in the data, taking relevant covariates into account. Estimated correlations as a function of time lag (= absolute difference between measurement moments t_1 and t_2) are provided as well as their confidence intervals (based on a non-parametric bootstrap).

Usage

```
Explore.WS.Corr(OLS.Model=" ", Dataset, Id, Time,
Alpha=0.05, Smoother.Span=.2, Number.Bootstrap=100,
Seed=1)
```

Arguments

`OLS.Model`	`OLS.Model` is a formula passed to `lm` (to obtain the OLS residuals, i.e., to take covariates into account in the computation of R). `OLS.Model` should thus be a formula that specifies the outcome of interest followed by a ~ sign and the covariates to be taken into account, e.g. `OLS.Model="Outcome~1+as.factor(Time) + as.factor(Treatment)"`.
`Dataset`	A `data.frame` that should consist of multiple lines per subject ('long' format).
`Id`	The subject indicator.
`Time`	The time indicator. Should be coded as 1, 2, etc.
`Alpha`	The α-level to be used in the non-parametric bootstrap-based Confidence Interval for R. Default `Alpha=0.05`
`Smoother.Span`	A smoothing (loess) technique is used to estimate R as a function of time lag. The smoother span gives the proportion of points in the plot which influence the smooth at each value. Larger values give more smoothness. For details, see https://stat.ethz.ch/R-manual/R-patched/library/stats/html/lowess.html. Defauls `Smoother.Span=.2`.
`Number.Bootstrap`	The number of non-parametric bootstrap samples to be used to estimate the Confidence Interval for R. Default `Number.Bootstrap=100`
`Seed`	The seed to be used in the bootstrap. Default `Seed=1`.

Value

`Est.Corr`	The estimated correlations R as a function of time lag. A smoothing (loess) technique is used to estimate R as a function of time lag (based on the output in `All.Corrs)`.
`All.Corrs`	A `matrix` that contains the estimated correlations R for all individual time lags.
`Bootstrapped.Corrs`	A `matrix` that contains the estimated correlations R as a function of time lag in the bootstrapped samples.
`Alpha`	The α level used in the estimation of the confidence interval.
`CI.Upper`	The upper bounds of the confidence intervals.
`CI.Lower`	The lower bounds of the confidence intervals.

Author(s)

Wim Van der Elst, Geert Molenberghs, Ralf-Dieter Hilgers, & Nicole Heussen

References

Van der Elst, W., Molenberghs, G., Hilgers, R., & Heussen, N. (2015). Estimating the reliability of repeatedly measured endpoints based on linear mixed-effects models. A tutorial. *Submitted.*

See Also

`plot.Explore.WS.Corr`

Examples

```
# Open data
data(Example.Data)

# Explore correlation structure
Expl_Corr <- Explore.WS.Corr(OLS.Model="Outcome~as.factor(Time)+
  as.factor(Cycle) + as.factor(Condition)", Dataset=Example.Data,
  Id="Id", Time="Time", Alpha=.05, Number.Bootstrap=50, Seed=123)

# explore results
summary(Expl_Corr)

# plot with correlations for all time lags, and
# add smoothed (loess) correlation function
plot(Expl_Corr, Indiv.Corrs=TRUE)
# plot bootstrapped smoothed (loess) correlation function
plot(Expl_Corr)
```

`Fract.Poly` *Fit fractional polynomials*

Description

Fits regression models with m terms of the form X^p, where the exponents p are selected from a small predefined set S of both integer and non-integer values.

Usage

```
Fract.Poly(Covariate, Outcome, S=c(-2,-1,-0.5,0,0.5,1,2,3), Max.M=5, Dataset)
```

Arguments

`Covariate`	The covariate to be considered in the models.
`Outcome`	The outcome to be considered in the models.
`S`	The set S from which each power p^m is selected. Default `S={-2, -1, -0.5, 0, 0.5, 1, 2, 3}`.
`Max.M`	The maximum order M to be considered for the fractional polynomial. This value can be 5 at most. When $M = 5$, then fractional polynomials of order 1 to 5 are considered. Default `Max.M=5`.
`Dataset`	A `data.frame` that should consist of multiple lines per subject ('long' format).

Value

`Results.M1`	The results (powers and AIC values) of the fractional polynomials of order 1.
`Results.M2`	The results (powers and AIC values) of the fractional polynomials of order 2.
`Results.M3`	The results (powers and AIC values) of the fractional polynomials of order 3.
`Results.M4`	The results (powers and AIC values) of the fractional polynomials of order 4.
`Results.M5`	The results (powers and AIC values) of the fractional polynomials of order 5.

Author(s)

Wim Van der Elst, Geert Molenberghs, Ralf-Dieter Hilgers, & Nicole Heussen

References

Van der Elst, W., Molenberghs, G., Hilgers, R., & Heussen, N. (2015). Estimating the reliability of repeatedly measured endpoints based on linear mixed-effects models. A tutorial. *Submitted.*

Examples

```
# Open data
data(Example.Data)

# Fit fractional polynomials, mox. order = 3
FP <- Fract.Poly(Covariate = Time, Outcome = Outcome,
Dataset = Example.Data, Max.M=3)

# Explore results
summary(FP)
# best fitting model (based on AIC) for m=3,
# powers:  p_{1}=3,  p_{2}=3, and  p_{3}=2

# Fit model and compare with observed means

    # plot of mean
Spaghetti.Plot(Dataset = Example.Data, Outcome = Outcome,
Time = Time, Id=Id, Add.Profiles = FALSE, Lwd.Me=1,
ylab="Mean Outcome")

    # Coding of predictors (note that when p_{1}=p_{2},
    # beta_{1}*X ** {p_{1}} + beta_{2}*X ** {p_{1}} * log(X)
    #  and when p=0, X ** {0}= log(X)    )
term1 <- Example.Data$Time**3
term2 <- (Example.Data$Time**3) * log(Example.Data$Time)
term3 <- Example.Data$Time**2

    # fit model
Model <- lm(Outcome~term1+term2+term3, data=Example.Data)
Model$coef # regression weights (beta's)

    # make prediction for time 1 to 47
term1 <- (1:47)**3
term2 <- ((1:47)**3) * log(1:47)
term3 <- (1:47)**2
    # compute predicted values
pred <- Model$coef[1] + (Model$coef[2] * term1) +
  (Model$coef[3] * term2) + (Model$coef[4] * term3)
   # Add predicted values to plot
lines(x = 1:47, y=pred,  lty=2)
legend("topright", c("Observed", "Predicted"), lty=c(1, 2))
```

Heatmap — *Plot a heatmap of the correlation structure*

Description

This function plots a heatmap of the correlation structure (reliability) in the data. It is a wrapper function for the `cor.plot` function of the `psych` package.

Usage

```
Heatmap(Dataset, Id, Outcome, Time, ...)
```

Arguments

`Dataset`	A `data.frame` that should consist of multiple lines per subject ('long' format).
`Id`	The subject indicator.
`Outcome`	The outcome indicator.
`Time`	The time indicator.
`...`	Other arguments to be passed to `cor.plot`.

Author(s)

Wim Van der Elst, Geert Molenberghs, Ralf-Dieter Hilgers, & Nicole Heussen

References

Van der Elst, W., Molenberghs, G., Hilgers, R., & Heussen, N. (2015). Estimating the reliability of repeatedly measured endpoints based on linear mixed-effects models. A tutorial. *Submitted.*

See Also

`plot.Explore.WS.Corr`

Examples

```
# Open data
data(Example.Data)

# Make heatmap
Heatmap(Dataset=Example.Data, Id = "Id",
Outcome="Outcome", Time = "Time")

# Make heatmap in black and white
Heatmap(Dataset=Example.Data, Id = "Id",
Outcome="Outcome", Time = "Time", colors=FALSE)
```

Model.Fit *Compare the fit of linear mixed-effects models*

Description

This function compares the fit of Model 1 (random intercept) and 2 (random intercept and Gausssian serial correlation), and of Model 2 (random intercept and Gausssian serial correlation) and 3 (random intercept, slope and Gausssian serial correlation)

Usage

```
Model.Fit(Model.1, Model.2)
```

Arguments

Model.1	An object of class WS.Corr.Mixed, the first fitted model.
Model.2	Another object of class WS.Corr.Mixed, the second fitted model.

Author(s)

Wim Van der Elst, Geert Molenberghs, Ralf-Dieter Hilgers, & Nicole Heussen

References

Van der Elst, W., Molenberghs, G., Hilgers, R., & Heussen, N. (2015). Estimating the reliability of repeatedly measured endpoints based on linear mixed-effects models. A tutorial. *Submitted.*

See Also

WS.Corr.Mixed

Examples

```
data(Example.Data)

# Code predictors for time
Example.Data$Time2 <- Example.Data$Time**2
Example.Data$Time3 <- Example.Data$Time**3
Example.Data$Time3_log <- (Example.Data$Time**3) * (log(Example.Data$Time))

# model 1
Model1 <- WS.Corr.Mixed(
Fixed.Part=Outcome ~ Time2 + Time3 + Time3_log + as.factor(Cycle)
+ as.factor(Condition), Random.Part = ~ 1|Id,
Dataset=Example.Data, Model=1, Id="Id",
Number.Bootstrap = 0, Seed = 12345)

# model 2
Model2 <- WS.Corr.Mixed(
```

```
Fixed.Part=Outcome ~ Time2 + Time3 + Time3_log + as.factor(Cycle)
+ as.factor(Condition), Random.Part = ~ 1|Id,
Correlation=corGaus(form= ~ Time, nugget = TRUE),
Dataset=Example.Data, Model=2, Id="Id",
Number.Bootstrap = 0, Seed = 12345)

# model 3
Model3 <- WS.Corr.Mixed(
  Fixed.Part=Outcome ~ Time2 + Time3 + Time3_log + as.factor(Cycle)
  + as.factor(Condition), Random.Part = ~ 1 + Time|Id,
  Correlation=corGaus(form= ~ Time, nugget = TRUE),
  Dataset=Example.Data, Model=3, Id="Id",
  Number.Bootstrap = 0, Seed = 12345)

# compare models 1 and 2
Model.Fit(Model.1=Model1, Model.2=Model2)

# compare models 2 and 3
Model.Fit(Model.1=Model2, Model.2=Model3)
```

---

plot Explore.WS.Corr — *Plot of exploratory within-subject correlations (reliabilities)*

Description

Provides an exploratory plot that allows for examining the within-subject correlations R (reliabilities) as a function if time lag.

Usage

```
## S3 method for class 'Explore.WS.Corr'
plot(x, Est.Corrs=TRUE, Indiv.Corrs=FALSE,
Add.CI=FALSE, Add.CI.Smoothed=TRUE, Smoother.Span=0.2,
Add.Boot.Corrs=FALSE, Add.CI.Polygon=FALSE,
ylim=c(-1, 1), xlab="Time Lag", ylab="Reliability", ...)
```

Arguments

`x`	A fitted object of class `Explore.WS.Corr`.
`Est.Corrs`	Logical. Should the smoothed (loess) correlation function as a function of time lag be added? Default `TRUE`.
`Indiv.Corrs`	Logical. Should the estimated correlations for all individual time lags be added? Default `FALSE`.
`Add.CI`	Logical. Should a bootstrapped $100(1-\alpha)\%$ Confidence Interval be added around the smoothed correlation function? Default `FALSE`.
`Add.CI.Smoothed`	Logical. Should a smoothed bootstrapped $100(1-\alpha)\%$ Confidence Interval be added around the smoothed correlation function? Default `FALSE`.

`Smoother.Span` The smoother span to be used. The smoother span gives the proportion of points in the plot which influence the smooth at each value. Larger values give more smoothness. For details, see https://stat.ethz.ch/R-manual/R-patched/library/stats/html/lowess.html. Defauls `Smoother.Span=.2`.

`Add.Boot.Corrs` Logical. Should the inidividual bootstrapped smoothed (loess) correlation functions be added? Default `FALSE`.

`Add.CI.Polygon` Logical. Similar to `Add.CI` but adds a grey polygon to mark the a bootstrapped $100(1-\alpha)\%$ Confidence Interval (instead of dashed lines). Default `FALSE`.

`ylim` The minimum and maximum values of the Y-axis. Default `ylim=c(-1,1)`.

`xlab` The label of the X-axis. Default `xlab="Time Lag"`.

`ylab` The label of the Y-axis. Default `ylab="Reliability"`.

`...` Other arguments to be passed to the plot function.

Author(s)

Wim Van der Elst, Geert Molenberghs, Ralf-Dieter Hilgers, & Nicole Heussen

References

Van der Elst, W., Molenberghs, G., Hilgers, R., & Heussen, N. (2015). Estimating the reliability of repeatedly measured endpoints based on linear mixed-effects models. A tutorial. *Submitted.*

See Also

`Explore.WS.Corr`, `Heatmap`

Examples

```
# Open data
data(Example.Data)

# Explore correlation structure
Expl_Corr <- Explore.WS.Corr(OLS.Model="Outcome~as.factor(Time)+
  as.factor(Cycle) + as.factor(Condition)", Dataset=Example.Data,
  Id="Id", Time="Time", Alpha=.05, Number.Bootstrap=50, Seed=123)

# explore results
summary(Expl_Corr)

# plot with correlations for all time lags, and
# add smoothed (loess) correlation function
plot(Expl_Corr, Indiv.Corrs=TRUE, Add.CI=FALSE, Add.Boot.Corrs=FALSE)
# plot bootstrapped smoothed (loess) correlation function
plot(Expl_Corr, Add.Boot.Corrs=TRUE)
```

`plot.WS.Corr.Mixed` *Plot the within-subject correlations (reliabilities) obtained by using the mixed-effects modeling approch*

Description

Plots the within-subject correlations (reliabilities) and $100(1-\alpha)\%$ Confidence Intervals based on the fitted mixed-effect models.

Usage

```
## S3 method for class 'WS.Corr.Mixed'
plot(x, xlab, ylab, ylim, main, All.Individual=FALSE, ...)
```

Arguments

`x`	A fitted object of class `WS.Corr.Mixed`
`xlab`	The label of the X-axis.
`ylab`	The label of the Y-axis.
`ylim`	The min, max values of the Y-axis.
`main`	The main title of the plot.
`All.Individual`	`Logical`. Should correlation functions be provided that show the correlations between all indidual measurement moments $R(t_i, t_k)$? Argument is only used if Model 2 was fitted. Default `All.Individual=FALSE`.
`...`	Other arguments to be passed to the plot function.

Author(s)

Wim Van der Elst, Geert Molenberghs, Ralf-Dieter Hilgers, & Nicole Heussen

References

Van der Elst, W., Molenberghs, G., Hilgers, R., & Heussen, N. (2015). Estimating the reliability of repeatedly measured endpoints based on linear mixed-effects models. A tutorial. *Submitted.*

See Also

`WS.Corr.Mixed`, `plot WS.Corr.Mixed`

Examples

```
# open data
data(Example.Data)

# Make covariates used in mixed model
Example.Data$Time2 <- Example.Data$Time**2
```

```
Example.Data$Time3 <- Example.Data$Time**3
Example.Data$Time3_log <- (Example.Data$Time**3) * (log(Example.Data$Time))

# model 1: random intercept model
Model1 <- WS.Corr.Mixed(
Fixed.Part=Outcome ~ Time2 + Time3 + Time3_log + as.factor(Cycle)
+ as.factor(Condition), Random.Part = ~ 1|Id,
Dataset=Example.Data, Model=1, Id="Id", Number.Bootstrap = 50,
Seed = 12345)

  # plot the results
plot(Model1)

## Not run: time-consuming code parts
# model 2: random intercept + Gaussian serial corr
Model2 <- WS.Corr.Mixed(
Fixed.Part=Outcome ~ Time2 + Time3 + Time3_log + as.factor(Cycle)
+ as.factor(Condition), Random.Part = ~ 1|Id,
Correlation=corGaus(form= ~ Time, nugget = TRUE),
Dataset=Example.Data, Model=2, Id="Id", Seed = 12345)

  # plot the results
    # estimated corrs as a function of time lag (default plot)
plot(Model2)
    # estimated corrs for all pairs of time points
plot(Model2, All.Individual = T)

# model 3
Model3 <- WS.Corr.Mixed(
  Fixed.Part=Outcome ~ Time2 + Time3 + Time3_log + as.factor(Cycle)
  + as.factor(Condition), Random.Part = ~ 1 + Time|Id,
  Correlation=corGaus(form= ~ Time, nugget = TRUE),
  Dataset=Example.Data, Model=3, Id="Id", Seed = 12345)

  # plot the results
    # estimated corrs for all pairs of time points
plot(Model3)
    # estimated corrs as a function of time lag

## End(Not run)
```

`Spaghetti.Plot` *Make a Spaghetti plot*

Description

Makes a spaghetti plot, i.e., a plot that depicts the outcome as a function of time for each individual subject.

Usage

```
Spaghetti.Plot(Dataset, Outcome, Time, Id, Add.Profiles=TRUE, Add.Mean=TRUE,
Add.Median=FALSE, Col=8, Lwd.Me=3, xlim, ylim, ...)
```

Arguments

`Dataset`	A `data.frame` that should consist of multiple lines per subject ('long' format).
`Outcome`	The name of the outcome variable.
`Time`	The name of the time indicator.
`Id`	The subject indicator.
`Add.Profiles`	Logical. Should the individual profiles be added? Default `Add.Profiles=TRUE`.
`Add.Mean`	Logical. Should a line that depicts the mean as a function of time be added? Default `Add.Mean=TRUE`.
`Add.Median`	Logical. Should a line that depicts the medean as a function of time be added? Default `Add.Mean=FALSE`.
`Col`	The color of the individual profiles. Default `Col=8` (grey).
`Lwd.Me`	The line width of the lines with mean and/or median. Default `Lwd.Me=3`.
`xlim`	The (min, max) values for the x-axis.
`ylim`	The (min, max) values for the y-axis.
`...`	Other arguments to be passed to the `plot()` function.

Author(s)

Wim Van der Elst, Geert Molenberghs, Ralf-Dieter Hilgers, & Nicole Heussen

References

Van der Elst, W., Molenberghs, G., Hilgers, R., & Heussen, N. (2015). Estimating the reliability of repeatedly measured endpoints based on linear mixed-effects models. A tutorial. *Submitted.*

Examples

```
# Open data
data(Example.Data)

# Plot individual profiles + mean
Spaghetti.Plot(Dataset = Example.Data, Outcome = Outcome, Id=Id, Time = Time)

# Plot individual profiles + median
Spaghetti.Plot(Dataset = Example.Data, Outcome = Outcome, Id=Id, Time = Time,
Add.Mean = FALSE, Add.Median = TRUE)
```

WS.Corr.Mixed	*Estimate within-subject correlations (reliabilities) based on a mixed-effects model.*

Description

This function allows for the estimation of the within-subject correlations using a general and flexible modeling approach that allows at the same time to capture hierarchies in the data, the presence of covariates, and the derivation of correlation estimates. Non-parametric bootstrap-based confidence intervals can be requested.

Usage

```
WS.Corr.Mixed(Dataset, Fixed.Part=" ", Random.Part=" ",
Correlation=" ", Id, Time=Time, Model=1,
Number.Bootstrap=100, Alpha=.05, Seed=1)
```

Arguments

`Dataset` A `data.frame` that should consist of multiple lines per subject ('long' format).

`Fixed.Part` The outcome and fixed-effect part of the mixed-effects model to be fitted. The model should be specified in agreement with the `lme` function requirements of the `nlme` package. See **examples** below.

`Random.Part` The random-effect part of the mixed-effects model to be fitted (specified in line with the `lme` function requirements). See **examples** below.

`Correlation` An optional object describing the within-group correlation structure (specified in line with the `lme` function requirements). See **examples** below.

`Id` The subject indicator.

`Time` The time indicator. Default `Time=Time`.

`Model` The type of model that should be fitted. `Model=1`: random intercept model, `Model=2`: random intercept and Gaussian serial correlation, `Model=3`: random intercept, slope, and Gaussian serial correlation, and `Model=4`: random intercept + slope. Default `Model=1`.

`Number.Bootstrap` The number of bootstrap samples to be used to estimate the Confidence Intervals around R. Default `Number.Bootstrap=100`. As an alternative to obtain confidence intervals, the Delta method can be used (see WS.Corr.Mixed.SAS).

`Alpha` The α-level to be used in the bootstrap-based Confidence Interval for R. Default $Alpha = 0.05$

`Seed` The seed to be used in the bootstrap. Default $Seed = 1$.

Details

Warning 1

To avoid problems with the `lme` function, do not specify powers directly in the function call. For example, rather than specifying `Fixed.Part=ZSV ~ Time + Time**2` in the function call, first add `Time**2` to the dataset (`Dataset$TimeSq <- Dataset$Time ** 2`) and then use the new variable name in the call: `Fixed.Part=ZSV ~ Time + TimeSq`

Warning 2 To avoid problems with the `lme` function, specify the Random.Part and Correlation arguments like e.g., `Random.Part = ~ 1| Subject` and `Correlation=corGaus(form= ~ Time, nugget = TRUE)`

not like e.g., `Random.Part = ~ 1| Subject` and `Correlation=corGaus(form= ~ Time| Subject, nugget = TRUE)`

(i.e., do not use `Time| Subject`)

Value

`Model`	The type of model that was fitted (model 1, 2, or 3.)
`D`	The D matrix of the fitted model.
`Tau2`	The τ^2 component of the fitted model. This component is only obtained when serial correlation is requested (Model 2 or 3), $\varepsilon_2 \sim N(0, \tau^2 H_i))$.
`Rho`	The ρ component of the fitted model which determines the matrix H_i, $\rho(\|t_{ij} - t_{ik}\|)$. This component is only obtained when serial correlation is considered (Model 2 or 3).
`Sigma2`	The residual variance.
`AIC`	The AIC value of the fitted model.
`LogLik`	The log likelihood value of the fitted model.
`R`	The estimated reliabilities.
`CI.Upper`	The upper bounds of the bootstrapped confidence intervals.
`CI.Lower`	The lower bounds of the bootstrapped confidence intervals.
`Alpha`	The α level used in the estimation of the confidence interval.
`Coef.Fixed`	The estimated fixed-effect parameters.
`Std.Error.Fixed`	The standard errors of the fixed-effect parameters.
`Time`	The time values in the dataset.
`Fitted.Model`	A fitted model of class `lme`.

Author(s)

Wim Van der Elst, Geert Molenberghs, Ralf-Dieter Hilgers, & Nicole Heussen

References

Van der Elst, W., Molenberghs, G., Hilgers, R., & Heussen, N. (2015). Estimating the reliability of repeatedly measured endpoints based on linear mixed-effects models. A tutorial. *Submitted.*

See Also

Explore.WS.Corr, WS.Corr.Mixed.SAS

Examples

```
# open data
data(Example.Data)

# Make covariates used in mixed model
Example.Data$Time2 <- Example.Data$Time**2
Example.Data$Time3 <- Example.Data$Time**3
Example.Data$Time3_log <- (Example.Data$Time**3) * (log(Example.Data$Time))

# model 1: random intercept model
Model1 <- WS.Corr.Mixed(
Fixed.Part=Outcome ~ Time2 + Time3 + Time3_log + as.factor(Cycle)
+ as.factor(Condition), Random.Part = ~ 1|Id,
Dataset=Example.Data, Model=1, Id="Id", Number.Bootstrap = 50,
Seed = 12345)

  # summary of the results
summary(Model1)
  # plot the results
plot(Model1)

## Not run: time-consuming code parts
# model 2: random intercept + Gaussian serial corr
Model2 <- WS.Corr.Mixed(
Fixed.Part=Outcome ~ Time2 + Time3 + Time3_log + as.factor(Cycle)
+ as.factor(Condition), Random.Part = ~ 1|Id,
Correlation=corGaus(form= ~ Time, nugget = TRUE),
Dataset=Example.Data, Model=2, Id="Id", Seed = 12345)

  # summary of the results
summary(Model2)

  # plot the results
    # estimated corrs as a function of time lag (default plot)
plot(Model2)
    # estimated corrs for all pairs of time points
plot(Model2, All.Individual = T)

# model 3
Model3 <- WS.Corr.Mixed(
  Fixed.Part=Outcome ~ Time2 + Time3 + Time3_log + as.factor(Cycle)
  + as.factor(Condition), Random.Part = ~ 1 + Time|Id,
  Correlation=corGaus(form= ~ Time, nugget = TRUE),
  Dataset=Example.Data, Model=3, Id="Id", Seed = 12345)

  # summary of the results
summary(Model3)
```

```
  # plot the results
    # estimated corrs for all pairs of time points
plot(Model3)
    # estimated corrs as a function of time lag

## End(Not run)
```

WS.Corr.Mixed.SAS *Estimate within-subject (test-retest) correlations based on a mixed-effects model using the SAS proc MIXED output.*

Description

This function allows for the estimation of the within-subject correlations using a general and flexible modeling approach that allows at the same time to capture hierarchies in the data, the presence of covariates, and the derivation of correlation estimates. The output of proc MIXED (SAS) is used as the input for this function. Confidence intervals for the correlations based on the Delta method are provided.

Usage

```
WS.Corr.Mixed.SAS(Model, D, Sigma2, Asycov, Rho, Tau2, Alpha=0.05, Time)
```

Arguments

`Model`	The type of model that should be fitted. `Model=1`: random intercept model, `Model=2`: random intercept and serial correlation, and `Model=3`: random intercept, slope, and serial correlation. Default `Model=1`.
`D`	The D matrix of the fitted model.
`Sigma2`	The residual variance.
`Asycov`	The asymptotic correlation matrix of covariance parameter estimates.
`Rho`	The ρ component of the fitted model which determines the matrix H_i, $\rho(\vert t_{ij} - t_{ik}\vert)$. This component is only needed when serial correlation is involved, i.e., when Model 2 or 3 used.
`Tau2`	The τ^2 component of the fitted model. This component is only needed when serial correlation is involved (i.e., when Model 2 or 3 used), $\varepsilon_2 \sim N(0, \tau^2 H_i))$.
`Alpha`	The α-level to be used in the computation of the Confidence Intervals around the within-subject correlation. The Confidence Intervals are based on the Delta method. Default `Alpha=0.05`.
`Time`	The time points available in the dataset on which the analysis was conducted.

Value

Model	The type of model that was fitted.
R	The estimated within-subject correlations.
Alpha	The α-level used to computed the Confidence Intervals around R.
CI.Upper	The upper bounds of the confidence intervals (Delta-method based).
CI.Lower	The lower bounds of the confidence intervals (Delta-method based).
Time	The time values in the dataset.

Author(s)

Wim Van der Elst, Geert Molenberghs, Ralf-Dieter Hilgers, & Nicole Heussen

References

Van der Elst, W., Molenberghs, G., Hilgers, R., & Heussen, N. (2015). Estimating the reliability of repeatedly measured endpoints based on linear mixed-effects models. A tutorial. *Submitted.*

See Also

WS.Corr.Mixed

Examples

```
# Open data
data(Example.Data)

# Estimate R and Delta method-based CI
# based on SAS output of fitted Model 2

# First specify asycov matrix
Asy_mat <- matrix(c(129170, -10248, -12.0814, -74.8605,
                    -10248, 25894, 21.0976, -50.1059,
                    -12.0814, 21.0976, 0.07791, 1.2120,
                    -74.8605, -50.1059, 1.212, 370.65), nrow = 4)
Model2_SAS <-  WS.Corr.Mixed.SAS(Model="Model 2",
D=500.98, Tau2=892.97, Rho=3.6302, Sigma2=190.09,
Asycov = Asy_mat, Time=c(1:45))
summary(Model2_SAS)
plot(Model2_SAS)
```

Index