

# Package ‘lcc’

September 13, 2020

**Type** Package

**Title** Longitudinal Concordance Correlation

**Version** 1.1.1

**Author** Thiago de Paula Oliveira [aut, cre],  
Rafael de Andrade Moral [aut],  
Silvio Sandoval Zocchi [ctb],  
Clarice Garcia Borges Demetrio [ctb],  
John Hinde [aut]

**Maintainer** Thiago de Paula Oliveira <thiago.paula.oliveira@alumni.usp.br>

**Description** Estimates the longitudinal concordance correlation to access the longitudinal agreement profile. The estimation approach implemented is variance components approach based on polynomial mixed effects regression model, as proposed by Oliveira, Hinde and Zocchi (2018) <doi:10.1007/s13253-018-0321-1>. In addition, non-parametric confidence intervals were implemented using percentile method or normal-approximation based on Fisher Z-transformation.

**Date** 2020-09-12

**Depends** R (>= 3.2.3), nlme (>= 3.1-124), ggplot2 (>= 2.2.1)

**Imports** hnp, parallel, doSNOW, doRNG, foreach

**Suggests** roxygen2 (>= 3.0.0), covr, testthat, MASS

**License** GPL (>= 2)

**Encoding** UTF-8

**Repository** CRAN

**NeedsCompilation** yes

**LazyData** true

**RoxygenNote** 7.0.2

**Date/Publication** 2020-09-13 13:00:03 UTC

## R topics documented:

AIC.lcc . . . . . 2

anova.lcc	3
coef.lcc	5
fitted.lcc	6
getVarCov.lcc	7
hue	8
lcc	9
lccPlot	14
logLik.lcc	16
plot.lcc	17
print.anova.lcc	19
print.lcc	20
print.summary.lcc	21
ranef.lcc	22
residuals.lcc	23
simulated_hue	24
simulated_hue_block	24
summary.lcc	25
vcov.lcc	27
<b>Index</b>	<b>28</b>

AIC.lcc

*Akaike and Bayesian Information Criteria for an lcc Object.***Description**

Calculate the Akaike's 'An Information Criterion' or the BIC or SBC (Schwarz's Bayesian criterion) for an object of class lcc.

**Usage**

```
## S3 method for class 'lcc'
AIC(object, ..., k = 2)

## S3 method for class 'lcc'
BIC(object, ...)
```

**Arguments**

object	an object inheriting from class lcc, representing a fitted longitudinal concordance correlation function.
...	optional arguments passed to the AIC function.
k	numeric value, use as penalty coefficient for the number of parameters in the fitted model; the default k = 2 is the classical AIC.

## Details

A numeric value with the corresponding AIC or BIC value. See methods for [AIC](#) objects to get more details.

## Author(s)

Thiago de Paula Oliveira, <thiago.paula.oliveira@usp.br>

## See Also

[lcc](#), [summary.lcc](#), [coef.lcc](#), [vcov.lcc](#)

## Examples

```
## Not run:
attach(simulated_hue)
fm6 <- lcc(data = simulated_hue, subject = "Fruit",
           resp = "Hue", method = "Method", time = "Time",
           qf = 2, qr = 1, components = TRUE,
           time_lcc = list(n=50, from=min(Time), to=max(Time)))
AIC(fm6)
BIC(fm6)

## End(Not run)
```

---

anova.lcc

*Compare Likelihoods of Fitted Models from an lcc Object*

---

## Description

If just one `lcc` model object is declared, a data frame with the numerator degrees of freedom, denominator degrees of freedom, F-values, and P-values for the fixed terms in the model. Otherwise, when multiple `lcc` fitted objects are being compared, a data frame with the degrees of freedom, the (restricted) log-likelihood, the Akaike Information Criterion (AIC), and the Bayesian Information Criterion (BIC) of each object is returned.

## Usage

```
## S3 method for class 'lcc'
anova(object, ..., test, type, adjustSigma,
       verbose)
```

**Arguments**

object	an object inheriting from class <code>lcc</code> or <code>lme</code> , representing a fitted longitudinal concordance correlation function.
...	other optional fitted model objects inheriting from classes "lcc", or "lme".
test	an optional logical value controlling whether likelihood ratio tests should be used to compare the fitted models represented by object and the objects in ... Defaults to TRUE.
type	an optional character string specifying the type of sum of squares to be used in F-tests for the terms in the model. If <code>sequential</code> , the sequential sum of squares obtained by including the terms in the order they appear in the model is used; else, if <code>marginal</code> , the marginal sum of squares obtained by deleting a term from the model at a time is used. This argument is only used when a single fitted object is passed to the function. Partial matching of arguments is used, so only the first character needs to be provided. Defaults to <code>sequential</code> .
adjustSigma	an optional logical value. If TRUE and the estimation method used to obtain object was maximum likelihood, the residual standard error is multiplied by $\sqrt{\text{nobs}/(\text{nobs} - \text{npar})}$ , converting it to a REML-like estimate. This argument is only used when a single fitted object is passed to the function. Default is TRUE.
verbose	an optional logical value. If TRUE, the calling sequences for each fitted model object are printed with the rest of the output, being omitted if <code>verbose = FALSE</code> . Defaults to FALSE.

**Details**

This function is an adaptation from the [anova.lme](#). For more details see methods for [nlme](#).

**Author(s)**

Thiago de Paula Oliveira, <thiago.paula.oliveira@usp.br>

**See Also**

[lcc](#), [summary.lcc](#)

**Examples**

```
## Not run:
## Testing random effects
fm1.aov <- lcc(data = hue, subject = "Fruit", resp = "H_mean",
              method = "Method", time = "Time", qf = 2, qr = 1)
fm2.aov <- update(fm1.aov, qr = 2)
anova(fm1.aov, fm2.aov)

## End(Not run)

## Not run:
# Testing fixed effects
fm3.aov <- update(fm2.aov, REML = FALSE)
```

```
fm4.aov <- update(fm2.aov, REML = FALSE, qf = 3)
anova(fm3.aov, fm4.aov)

## End(Not run)

## Not run:
# Comparing the 3 lcc models
fm5.aov <- update(fm2.aov, var.class = varExp, weights.form = "time")
anova(fm1.aov, fm2.aov, fm5.aov)

## End(Not run)
```

---

`coef.lcc`*Extract Model Coefficients*

---

## Description

The fixed effects estimated and corresponding random effects estimates are obtained at subject levels less or equal to  $i$ . The resulting estimates are returned as a data frame, with rows corresponding to subject levels and columns to coefficients.

## Usage

```
## S3 method for class 'lcc'
coef(object, ...)
```

## Arguments

<code>object</code>	an object inheriting from class <code>lcc</code> , representing a fitted longitudinal concordance correlation function.
<code>...</code>	optional arguments passed to the <code>coef.lme</code> function.

## Details

See methods for [nlme](#) objects to get more details.

## Author(s)

Thiago de Paula Oliveira, <[thiago.paula.oliveira@usp.br](mailto:thiago.paula.oliveira@usp.br)>

## See Also

[lcc](#), [summary.lcc](#), [lccPlot](#), [vcov.lcc](#)

**Examples**

```
## Not run:
fm1<-lcc(data = hue, subject = "Fruit", resp = "H_mean",
          method = "Method", time = "Time", qf = 2, qr = 2)
coef(fm1)

## End(Not run)
```

---

fitted.lcc

*Extract lcc Fitted Values*


---

**Description**

Fitted values from object of class lcc returned by modeling functions.

**Usage**

```
## S3 method for class 'lcc'
fitted(object, type, digits, ...)
```

**Arguments**

object	an object inheriting from class lcc, representing a fitted longitudinal concordance correlation function.
type	an optional character string specifying the type of output to be returned. If type="lcc", prints the fitted longitudinal concordance correlation. If type="lpc", prints the fitted longitudinal Pearson correlation. If type="la", prints the fitted longitudinal accuracy. Defaults to type="lcc".
digits	a non-null value for digits specifies the minimum number of significant digits to be printed in values. The default, NULL.
...	not used.

**Value**

A data frame with columns given by methods, time, and fitted values.

**Author(s)**

Thiago de Paula Oliveira, <thiago.paula.oliveira@usp.br>

**See Also**

[lcc](#), [summary.lcc](#), [lccPlot](#)

## Examples

```
data(hue)
## Second degree polynomial model with random intercept, slope and
## quadratic term
## Not run:
fm1 <- lcc(data = hue, subject = "Fruit", resp = "H_mean",
           method = "Method", time = "Time", qf = 2, qr = 2,
           components = TRUE)
fitted(fm1)
fitted(fm1, type="lpc")
fitted(fm1, type="la")

## End(Not run)
```

---

getVarCov.lcc

*Extract Variance Components from a Fitted Model*

---

## Description

Returns the variance-covariance matrix of a fitted lcc model object.

## Usage

```
## S3 method for class 'lcc'
getVarCov(obj, type, ...)
```

## Arguments

obj	an object inheriting from class lcc, representing a fitted longitudinal concordance correlation function.
type	specifies the type of variance covariance matrix. If type = "random.effects", the default, extract the random-effects variance-covariance; if type = "conditional" extract the conditional variance-covariance of the responses; and if type = "marginal" extracts the the marginal variance-covariance of the responses.
...	optional arguments passed to the getVarCov function.

## Details

See methods for [nlme](#) objects to get more details.

## Author(s)

Thiago de Paula Oliveira, <thiago.paula.oliveira@usp.br>

## See Also

[lcc](#), [summary.lcc](#), [coef.lcc](#), [vcov.lcc](#)

## Examples

```
## Not run:
fm1<-lcc(data = hue, subject = "Fruit", resp = "H_mean",
          method = "Method", time = "Time", qf = 2, qr = 2)
getVarCov(fm1)

## End(Not run)
```

---

hue

*Hue color data*

---

## Description

An observational study conducted at the Vegetable Production Department at "Luiz de Queiroz" College of Agriculture/University of São Paulo in 2010/2011 to evaluate the peel color of 20 papaya fruits cv. Sunrise Solo over time. The color hue was measured on the equatorial region of each fruit using four points observed by the colorimeter and 1,000 points observed by the scanner. Thus, the circular mean hue was calculated for each fruit by each device at time  $t$ . The aim of the agreement study was to assess how well the colorimeter agreed with the scanner over time.

## Usage

```
data(hue)
```

## Format

A data frame with 554 observations on the mean hue variable. The format is:

H_mean	numeric; mean hue of papaya's peel
Method	a factor with levels Colorimeter, Scanner
Time	integer; time in days
Fruit	a factor with 20 levels; from 1 to 20 where each level is represented by one fruit.

## Source

Oliveira, T.P.; Hinde, J.; Zocchi S.S. Longitudinal Concordance Correlation Function Based on Variance Components: An Application in Fruit Color Analysis. *Journal of Agricultural, Biological, and Environmental Statistics*, v. 23, n. 2, 233–254, 2018.

Oliveira, T.P.; Zocchi S.S.; Jacomino, A.P. Measuring color hue in 'Sunrise Solo' papaya using a flatbed scanner. *Rev. Bras. Frutic.*, v. 39, n. 2, e-911, 2017.

## References

Oliveira, T.P.; Hinde, J.; Zocchi S.S. Longitudinal Concordance Correlation Function Based on Variance Components: An Application in Fruit Color Analysis. *Journal of Agricultural, Biological,*



and Environmental Statistics, v. 23, n. 2, 233–254, 2018.

### See Also

[lcc](#).

### Examples

```
data(hue)
summary(hue)
str(hue)
## Second degree polynomial model with random intercept, slope and
## quadratic term including an exponential variance function using
## time as covariate.
model<-lcc(data = hue, subject = "Fruit", resp = "H_mean",
           method = "Method", time = "Time", qf = 2, qr = 2,
           components = TRUE, time_lcc = list(from = min(hue$Time),
           to = max(hue$Time), n=40), var.class=varExp,
           weights.form="time")
summary(model, type="model")
summary(model, type="lcc")
## for discussion on the analysis of complete data set,
## see Oliveira et al. (2018)
```

---

lcc	<i>Longitudinal Concordance Correlation (LCC) Estimated by Fixed Effects and Variance Components using a Polynomial Mixed-Effects Regression Model</i>
-----	--

---

### Description

The `lcc` function gives fitted values and non-parametric bootstrap confidence intervals for LCC, longitudinal Pearson correlation (LPC), and longitudinal accuracy (LA) statistics. These statistics can be estimated using different structures for the variance-covariance matrix for random effects and variance functions to model heteroscedasticity among the within-group errors using or not the time as a covariate.

### Usage

```
lcc(data, resp, subject, method, time, interaction, qf,
    qr, covar, gs, pdmat, var.class, weights.form, time_lcc, ci,
    percentileMet, alpha, nboot, show.warnings, components,
    REML, lme.control, numCore)
```

### Arguments

`data` an object of class `data.frame`.  
`resp` character string. Name of the response variable in the data set.

<code>subject</code>	character string. Name of the subject variable in the data set.
<code>method</code>	character string. Name of the method variable in the data set. The first level of method is used as the gold-standard method.
<code>time</code>	character string. Name of the time variable in the data set.
<code>interaction</code>	an option to estimate the interaction effect between <code>method</code> and <code>time</code> . If <code>TRUE</code> , the default, interaction effect is estimated. If <code>FALSE</code> only the main effects of time and method are estimated.
<code>qf</code>	an integer specifying the degree time polynomial trends, normally 1, 2 or 3. (Degree 0 is not allowed). Default is <code>qf=1</code>
<code>qr</code>	an integer specifying random effects terms to account for subject-to-subject variation. Note that <code>qr=0</code> specifies a random intercept (form <code>~ 1   subject</code> ); <code>qr=1</code> specifies random intercept and slope (form <code>~ time   subject</code> ). If <code>qr=qf=q</code> , with $q \geq 1$ , random effects at subject level are added to all terms of the time polynomial regression (form <code>~ poly(time, q, raw = TRUE)   subject</code> ). Default is <code>qr=0</code> .
<code>covar</code>	character vector. Name of the covariates to be included in the model as fixed effects. Default to <code>NULL</code> , never include.
<code>gs</code>	character string. Name of method level which represents the gold-standard. Default is the first level of method.
<code>pdmat</code>	standard classes of positive-definite matrix structures defined in the <code>pdClasses</code> function. The different positive-definite matrices structures available in the <code>lcc</code> function are <code>pdSymm</code> , the default, <code>pdLogChol</code> , <code>pdDiag</code> , <code>pdIdent</code> , <code>pdCompSymm</code> , and <code>pdNatural</code> .
<code>var.class</code>	standard classes of variance functions to model the variance structure of within-group errors using covariates, see <code>varClasses</code> . Default to <code>NULL</code> , correspond to homoscedastic within-group errors. Available standard classes:  <code>varIdent</code> : allows different variances according to the levels of the stratification variable. <code>varExp</code> : exponential function of the variance covariate; see <code>varExp</code> .
<code>weights.form</code>	character string. An one-sided formula specifying a variance covariate and, optionally, a grouping factor for the variance parameters in the <code>var.class</code> . If <code>var.class=varIdent</code> , the option “method”, form <code>~1   method</code> or “time.ident”, form <code>~1   time</code> , must be used in the <code>weights.form</code> argument. If <code>var.class=varExp</code> , the option “time”, form <code>~time</code> , or “both”, form <code>~time   method</code> , must be used in the <code>weights.form</code> argument.
<code>time_lcc</code>	regular sequence for time variable merged with specific or experimental time values used for LCC, LPC, and LA predictions. Default is <code>NULL</code> . The list may contain the following components:  <code>time</code> : a vector of specific or experimental time values of given length. The experimental time values are used as default. <code>from</code> : the starting (minimum) value of time variable. <code>to</code> : the end (maximum) value of time variable. <code>n</code> : an integer specifying the desired length of the sequence. Generally, <code>n</code> between 30 and 50 is adequate.

ci	an optional non-parametric bootstrap confidence interval calculated for the LCC, LPC and LA statistics. If TRUE confidence intervals are calculated and printed in the output. Default is FALSE.
percentileMet	an optional method for calculating the non-parametric bootstrap intervals. If FALSE, the default, is the normal approximation method. If TRUE, the percentile method is used instead.
alpha	significance level. Default is 0.05.
nboot	an integer specifying the number of bootstrap samples. Default is 5,000.
show.warnings	an optional argument that shows the number of convergence errors in the bootstrap samples. If TRUE shows in which bootstrap sample the error occurred. If FALSE, the default, shows the total number of convergence errors.
components	an option to print LPC and LA statistics. If TRUE the estimates and confidence intervals for LPC and LA are printed in the output. If FALSE, the default, provides estimates and confidence interval only for the LCC statistic.
REML	if TRUE, the default, the model is fit by maximizing the restricted log-likelihood. If FALSE the log-likelihood is maximized.
lme.control	a list of control values for the estimation algorithm to replace the default values of the function <code>lmeControl</code> available in the <code>nlme</code> package. Defaults to an empty list. The returned list is used as the control argument for the <code>lme</code> function.
numCore	number of cores used in parallel during bootstrapping computation. Default is 1.

### Value

an object of class `lcc`. The output is a list with the following components:

model	summary of the polynomial mixed-effects regression model.
Summary.lcc	fitted values for the LCC or LCC, LPC and LA (if <code>components=TRUE</code> ); concordance correlation coefficient (CCC) between methods for each level of time as sampled values, and the CCC between mixed-effects model predicted values and observed values from data as goodness of fit ( <code>gof</code> )
data	the input dataset.

### Author(s)

Thiago de Paula Oliveira, <thiago.paula.oliveira@usp.br>, Rafael de Andrade Moral, John Hinde

### References

- Lin, L. A Concordance Correlation Coefficient to Evaluate Reproducibility. *Biometrics*, 45, n. 1, 255-268, 1989.
- Oliveira, T.P.; Hinde, J.; Zocchi S.S. Longitudinal Concordance Correlation Function Based on Variance Components: An Application in Fruit Color Analysis. *Journal of Agricultural, Biological, and Environmental Statistics*, v. 23, n. 2, 233–254, 2018.

**See Also**

[summary.lcc](#), [fitted.lcc](#), [print.lcc](#), [lccPlot](#), [plot.lcc](#), [coef.lcc](#), [ranef.lcc](#), [vcov.lcc](#), [getVarCov.lcc](#), [residuals.lcc](#), [AIC.lcc](#)

**Examples**

```

data(hue)
## Second degree polynomial model with random intercept, slope and
## quadratic term
fm1 <- lcc(data = hue, subject = "Fruit", resp = "H_mean",
           method = "Method", time = "Time", qf = 2, qr = 2)
print(fm1)
summary(fm1)
summary(fm1, type="model")
lccPlot(fm1) +
  ylim(0,1) +
  geom_hline(yintercept = 1, linetype = "dashed") +
  scale_x_continuous(breaks = seq(1,max(hue$Time),2))

## Estimating longitudinal Pearson correlation and longitudinal
#accuracy
fm2 <- update(fm1, components = TRUE)
summary(fm2)
lccPlot(fm2) +
  ylim(0,1) +
  geom_hline(yintercept = 1, linetype = "dashed") +
  scale_x_continuous(breaks = seq(1,max(hue$Time),2)) +
  theme_bw()

## Not run:
## A grid of points as the Time variable for prediction
fm3 <- update(fm2, time_lcc = list(from = min(hue$Time),
                                  to = max(hue$Time), n=40))
summary(fm3)
lccPlot(fm3) +
  ylim(0,1) +
  geom_hline(yintercept = 1, linetype = "dashed") +
  scale_x_continuous(breaks = seq(1,max(hue$Time),2)) +
  theme_bw()

## End(Not run)

## Including an exponential variance function using time as a
#covariate.
fm4 <- update(fm2,time_lcc = list(from = min(hue$Time),
                                  to = max(hue$Time), n=30), var.class=varExp,
              weights.form="time")
summary(fm4, type="model")
fitted(fm4)
fitted(fm4, type = "lpc")
fitted(fm4, type = "la")
lccPlot(fm4) +

```

```

  geom_hline(yintercept = 1, linetype = "dashed")
lccPlot(fm4, type = "lpc") +
  geom_hline(yintercept = 1, linetype = "dashed")
lccPlot(fm4, type = "la") +
  geom_hline(yintercept = 1, linetype = "dashed")

## Not run:
## Non-parametric confidence interval with 500 bootstrap samples
fm5 <- update(fm1, ci = TRUE, nboot = 500)
summary(fm5)
lccPlot(fm5) +
  geom_hline(yintercept = 1, linetype = "dashed")

## End(Not run)

## Considering three methods of color evaluation
## Not run:
data(simulated_hue)
attach(simulated_hue)
fm6 <- lcc(data = simulated_hue, subject = "Fruit",
  resp = "Hue", method = "Method", time = "Time",
  qf = 2, qr = 1, components = TRUE,
  time_lcc = list(n=50, from=min(Time), to=max(Time)))
summary(fm6)
lccPlot(fm6, scales = "free")
lccPlot(fm6, type="lpc", scales = "free")
lccPlot(fm6, type="la", scales = "free")
detach(simulated_hue)

## End(Not run)

## Including an additional covariate in the linear predictor
## (randomized block design)
## Not run:
data(simulated_hue_block)
attach(simulated_hue_block)
fm7 <- lcc(data = simulated_hue_block, subject = "Fruit",
  resp = "Hue", method = "Method", time = "Time",
  qf = 2, qr = 1, components = TRUE, covar = c("Block"),
  time_lcc = list(n=50, from=min(Time), to=max(Time)))
summary(fm7)
lccPlot(fm7, scales="free")
detach(simulated_hue_block)

## End(Not run)

## Testing interaction effect between time and method
fm8 <- update(fm1, interaction = FALSE)
anova(fm1, fm8)

## Not run:
## Using parallel computing with 3 cores, and a set.seed(123)
to verify model reproducibility.

```

```

set.seed(123)
fm9 <- lcc(data = hue, subject = "Fruit", resp = "H_mean",
           method = "Method", time = "Time", qf = 2, qr = 2,
           ci=TRUE, nboot = 30, numCore = 3)

# Repeating same model with same set seed.
set.seed(123)
fm10 <- lcc(data = hue, subject = "Fruit", resp = "H_mean",
            method = "Method", time = "Time", qf = 2, qr = 2,
            ci=TRUE, nboot = 30, numCore = 3)

## Verifying if both fitted values and confidence intervals
are identical
identical(fm9$Summary.lcc$fitted, fm10$Summary.lcc$fitted)

## End(Not run)

```

---

lccPlot

*Plot Fitted Curves from an lcc Object.*


---

### Description

A plot of predictions versus the time covariate is generated. Predicted values are joined by lines while sampled observations are represented by circles. If the argument `components=TRUE` is considered in the `lcc` object, single plots of each statistics are returned on different pages.

### Usage

```
lccPlot(obj, type, control, ...)
```

### Arguments

<code>obj</code>	an object inheriting from class "lcc", representing a fitted lcc model.
<code>type</code>	character string. If <code>type = "lcc"</code> , the output is the LCC plot; if <code>type = "lpc"</code> , the output is the LPC plot; and if <code>type = "la"</code> the output is the LA plot. Types "lpc" and "la" are available only if <code>components = TRUE</code> .
<code>control</code>	a list of control values or character strings returned by the function <code>plotControl</code> . Defaults to an empty list. The list may contain the following components: <ul style="list-style-type: none"> <li><code>shape</code>: draw points considering a shape parameter. Possible shape values are the numbers 0 to 25, and 32 to 127; see <a href="#">aes_linetype_size_shape</a>. Default is 1.</li> <li><code>colour</code>: specification for lines color. Default is "black".</li> <li><code>size</code>: specification for lines size. Should be specified with a numerical value (in millimetres); see <a href="#">aes_linetype_size_shape</a>. Default is 0.5.</li> <li><code>xlab</code>: title for the x axis. Default is "Time".</li> <li><code>ylab</code>: title for the y axis. Default is "LCC", "LPC", or "LA"</li> </ul>

`scale_y_continuous`: numeric vector of length two providing limits of the scale. Default is `c(0,1)`.

`all.plot`: viewport functions for the `lcc` class. If `TRUE`, the default, returns an object created by the `viewport` function with multiple plots on a single page. If `FALSE` returns a single `ggplot` object by different pages using the `marrangeGrob` function.

... arguments to be passed to `facet_wrap` function

### Author(s)

Thiago de Paula Oliveira, <thiago.paula.oliveira@usp.br>

### References

Lin, L. A Concordance Correlation Coefficient to Evaluate Reproducibility. *Biometrics*, 45, n. 1, 255-268, 1989.

Oliveira, T.P.; Hinde, J.; Zocchi S.S. Longitudinal Concordance Correlation Function Based on Variance Components: An Application in Fruit Color Analysis. *Journal of Agricultural, Biological, and Environmental Statistics*, v. 23, n. 2, 233–254, 2018.

### See Also

[lcc](#).

### Examples

```
data(hue)
## Second degree polynomial model with random intercept, slope and
## quadratic term
fm1<-lcc(data = hue, subject = "Fruit", resp = "H_mean",
         method = "Method", time = "Time", qf = 2, qr = 2,
         components=TRUE)
lccPlot(fm1, type="lcc")
lccPlot(fm1, type="lpc")
lccPlot(fm1, type="la")

## Using themes of ggplot2 package
lccPlot(fm1, type = "lpc")+
  ylim(0,1) +
  geom_hline(yintercept = 1, linetype = "dashed") +
  scale_x_continuous(breaks = seq(1,max(hue$Time),2))+
  theme_bw() +
  theme(legend.position = "none", aspect.ratio = 1,
        axis.line.x = element_line(color="black", size = 0.5),
        axis.line.y = element_line(color="black", size = 0.5),
        axis.title.x = element_text(size=14),
        axis.title.y = element_text(size=14),
        axis.text.x = element_text(size = 14, face = "plain"),
        axis.text.y = element_text(size = 14, face = "plain"))
```

```
## Using the key (+) to constructing sophisticated graphics
lccPlot(fm1, type="lcc") +
  scale_y_continuous(limits=c(-1, 1)) +
  labs(title="My title",
        y = "Longitudinal Concordance Correlation",
        x = "Time (Days)")

## Running all.plots = FALSE and saving plots as pdf
## Not run:
data(simulated_hue_block)
attach(simulated_hue_block)
fm2<-lcc(data = simulated_hue_block, subject = "Fruit",
         resp = "Hue", method = "Method", time = "Time",
         qf = 2, qr = 1, components = TRUE, covar = c("Block"),
         time_lcc = list(n=50, from=min(Time), to=max(Time)))
ggsave("myplots.pdf",
        lccPlot(fm2, type="lcc", scales = "free"))

## End(Not run)
```

---

logLik.lcc

*Extract Log-Likelihood of an lcc Object*


---

## Description

If REML=TRUE, the default, returns the restricted log-likelihood value of the linear mixed-effects model; else the log-likelihood value

## Usage

```
## S3 method for class 'lcc'
logLik(object, ..., REML)
```

## Arguments

object	an object inheriting from class lcc, representing a fitted longitudinal concordance correlation function.
...	further arguments passed to <a href="#">logLik</a> .
REML	an optional logical value. If TRUE the restricted log-likelihood is returned, else, if FALSE, the log-likelihood is returned.

## Details

See methods for [nlme](#) objects to get more details.

## Author(s)

Thiago de Paula Oliveira, <thiago.paula.oliveira@usp.br>



**See Also**

[lcc](#), [summary.lcc](#)

**Examples**

```
## Not run:
fm1<-lcc(data = hue, subject = "Fruit", resp = "H_mean",
          method = "Method", time = "Time", qf = 2, qr = 2)
logLik(fm1)

## End(Not run)
```

---

plot.lcc

*Diagnostic Plots of an lcc Object.*


---

**Description**

Diagnostic plots for conditional error and random effects from the linear mixed-effects fit are obtained. Six plots (selectable by 'which') are currently available: a plot of residuals against fitted values, a plot of residuals against time variable, a boxplot of residuals by subject, a plot of observed values against fitted values, a normal Q-Q plot with simulation envelopes based on conditional error, and a normal Q-Q plot with simulation envelopes based on the random effects. By default, all plots are provided.

**Usage**

```
## S3 method for class 'lcc'
plot(x, which = c(1L:6L),
      caption = list("Residuals vs Fitted",
                    "Residuals vs Time",
                    "Residuals by Subject",
                    "Observed values vs Fitted values",
                    "Normal Q-Q Plot (Conditional residuals)",
                    "Normal Q-Q Plot (Random effects)"),
      sub.caption = NULL, main = NULL,
      panel = if(add.smooth) panel.smooth else points,
      add.smooth = TRUE, ask = TRUE,
      id.n = 3, labels.id = names(residuals(x)),
      label.pos = c(4, 2), cex.id = 0.75, cex.caption = 1,
      cex.oma.man = 1.25, ...)
```

**Arguments**

x an object inheriting from class [lcc](#), representing a fitted longitudinal concordance correlation function.

which	if a subset of the plots is required, specify a subset of the numbers from 1 to 6.
caption	captions to appear above the plots. Vector or list of valid graphics annotations is required. All captions can be suppressed using '""' or NA.
sub.caption	common sub-title (at bottom). Default to NULL.
main	The main title (on top) above the caption.
panel	panel function. If <code>add.smooth = TRUE</code> , <code>panel.smooth</code> is used rather than <code>points</code> .
add.smooth	logical indicating if smoother should be added to most plots; see also <code>panel</code> above. Defaults to TRUE.
ask	logical; if TRUE, the default, the user is <code>_ask_ed</code> before each plot, see <code>par</code> .
id.n	number of points to be labelled is the first three plots, starting with the most extreme.
labels.id	vector of labels, from which the labels for extreme points will be chosen. Default to NULL (uses observation numbers).
label.pos	positioning of labels, for the left half and right half of the graph respectively, for plots 1-3.
cex.id	magnification of point label.
cex.caption	controls the size of caption.
cex.oma.man	controls the size of the <code>sub.caption</code> only if that is <code>_above_</code> the figures when there is more than one.
...	further graphical parameters from <code>'par'</code> .

### Details

The Q-Q plot uses the normalized residuals. The standardized residuals is pre-multiplied by the inverse square-root factor of the estimated error correlation matrix while the random effects is pre-multiplied by the inverse square root of the estimated variances obtained from matrix G. The simulate envelopes are obtained from package `hnp` (Moral et al., 2018).

Code partially adapted from `plot.lm`.

### Author(s)

Thiago de Paula Oliveira, <thiago.paula.oliveira@usp.br>

### See Also

`lccPlot`, `lcc`, `mtext`, `text`, `plotmath`

### Examples

```
## Second degree polynomial model with random intercept, slope and
## quadratic term
fm1 <- lcc(data = hue, subject = "Fruit", resp = "H_mean",
           method = "Method", time = "Time", qf = 2, qr = 2)
plot(fm1)
```

---

print.anova.lcc      *Print the Anova of an lcc Object*

---

### Description

Method print for the anova.lcc.

### Usage

```
## S3 method for class 'anova.lcc'  
print(x, verbose, ...)
```

### Arguments

x                    an object inheriting from class [anova.lcc](#), representing a fitted longitudinal concordance correlation function.

verbose             an optional logical value used to control the amount of printed output. If TRUE, the calling sequences for each fitted model object are printed with the rest of the output, being omitted if verbose = FALSE. Defaults to FALSE.

...                  further arguments passed to [print](#).

### Details

Modified from [anova.lme](#). For more details see methods for [nlme](#).

### Author(s)

Thiago de Paula Oliveira, <[thiago.paula.oliveira@usp.br](mailto:thiago.paula.oliveira@usp.br)>

### See Also

[summary.lcc](#), [lccPlot](#), [lcc](#)

### Examples

```
## Not run:  
## Second degree polynomial model with random intercept, slope and  
## quadratic term  
fm1<-lcc(data = hue, subject = "Fruit", resp = "H_mean",  
          method = "Method", time = "Time", qf = 2, qr = 2)  
print(anova(fm1))  
  
## End(Not run)
```

---

`print.lcc`*Print an lcc Object*

---

**Description**

Prints information about the longitudinal concordance correlation represented by an object of class `lcc`. The returned object has a `print` method.

**Usage**

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

**Arguments**

<code>x</code>	an object inheriting from class <code>lcc</code> , representing a fitted longitudinal concordance correlation function.
<code>digits</code>	a non-null value for <code>digits</code> specifies the minimum number of significant digits to be printed in values. The default, <code>NULL</code> .
<code>...</code>	further arguments passed to <code>{print}</code> .

**Value**

an object inheriting from class `print.lcc`.

**Author(s)**

Thiago de Paula Oliveira, <thiago.paula.oliveira@usp.br>

**See Also**

[lcc](#), [summary.lcc](#)

**Examples**

```
## Not run:  
## Second degree polynomial model with random intercept, slope and  
## quadratic term  
fm1<-lcc(data = hue, subject = "Fruit", resp = "H_mean",  
         method = "Method", time = "Time", qf = 2, qr = 2)  
print(fm1)  
  
## End(Not run)
```

---

print.summary.lcc      *Print the Summary of an lcc Object*

---

### Description

Information summarizing the fitted longitudinal concordance correlation is printed. This includes the AIC, BIC, and log-likelihood at convergence. If `type = "lcc"`, prints the fitted values while `type = "model"` prints the fixed effects estimates and their standard errors, standard deviations, correlations for the random effects, within-group correlation, and variance function parameters.

### Usage

```
## S3 method for class 'summary.lcc'  
print(x, verbose, digits, ...)
```

### Arguments

<code>x</code>	an object inheriting from class <code>summary.lcc</code> , representing a fitted longitudinal concordance correlation function.
<code>verbose</code>	an optional logical value used to control the amount of printed output when <code>type = "model"</code> . Defaults to FALSE
<code>digits</code>	a non-null value for <code>digits</code> specifies the minimum number of significant digits to be printed in values. The default, NULL.
<code>...</code>	further arguments passed to <code>print</code> .

### Author(s)

Thiago de Paula Oliveira, <thiago.paula.oliveira@usp.br>

### See Also

[summary.lcc](#), [lccPlot](#), [lcc](#)

### Examples

```
## Not run:  
## Second degree polynomial model with random intercept, slope and  
## quadratic term  
fm1<-lcc(data = hue, subject = "Fruit", resp = "H_mean",  
          method = "Method", time = "Time", qf = 2, qr = 2)  
print(summary(fm1, type="model"))  
  
## End(Not run)
```

---

ranef.lcc                      *Extract Model Random Effects*

---

## Description

Extract the estimated random effects at level *i*. A data frame with rows given by the different groups at that level and columns given by the random effects.

## Usage

```
## S3 method for class 'lcc'  
ranef(object, ...)
```

## Arguments

`object`                      an object inheriting from class `lcc`, representing a fitted longitudinal concordance correlation function.

`...`                        optional arguments passed to the `ranef.lme` function.

## Details

See methods for `nlme` objects to get more details.

## Author(s)

Thiago de Paula Oliveira, <thiago.paula.oliveira@usp.br>

## See Also

[lcc](#), [coef.lcc](#),

## Examples

```
## Not run:  
fm1<-lcc(data = hue, subject = "Fruit", resp = "H_mean",  
          method = "Method", time = "Time", qf = 2, qr = 2)  
ranef(fm1)  
  
## End(Not run)
```

---

residuals.lcc	<i>Extract Model Residuals</i>
---------------	--------------------------------

---

**Description**

Extract the residulas from the model used to estimate the longitudinal concordance correlation function.

**Usage**

```
## S3 method for class 'lcc'  
residuals(object, type, ...)
```

**Arguments**

object	an object inheriting from class <code>lcc</code> , representing a fitted longitudinal concordance correlation function.
type	an optional character string specifying the type of residulas to be used. If <code>type = "response"</code> , the default, the residuals at level <code>i</code> are obtained by subtracting the fitted values at that level from the response vector. If <code>type = "pearson"</code> , the "response" residuals is divided by the estimated within-group standard error. If <code>type = "normalized"</code> , the normalized residuals are used. Partial matching of arguments is used, so only the first character needs to be provided.
...	optional arguments passed to the <code>residuals.lme</code> function.

**Details**

See methods for `nlme` objects to get more details.

**Author(s)**

Thiago de Paula Oliveira, <thiago.paula.oliveira@usp.br>

**See Also**

[lcc](#), [summary.lcc](#), [coef.lcc](#), [vcov.lcc](#)

**Examples**

```
## Not run:  
fm1<-lcc(data = hue, subject = "Fruit", resp = "H_mean",  
         method = "Method", time = "Time", qf = 2, qr = 2)  
getVarCov(fm1)  
  
## End(Not run)
```

---

simulated\_hue      *Hue color simulated data*

---

**Description**

Simulated hue data set based on papaya's maturation over time considering three methods of measurement.

**Usage**

```
data(simulated_hue)
```

**Format**

A simulated data frame with 6,000 observations on the mean hue variable. The format is:

Hue	numeric; mean hue of papaya's peel
Method	a factor with levels labelled from Method 1 to Method 3
Time	integer; time in days from 0 to 19
Fruit	a factor with 100 levels labelled from 1 to 100 where each level is represented by one fruit.

**Details**

A total of 100 fruits were observed over 20 days by three methods to evaluate the mean hue of fruit's peel. The aim of the agreement study was to assess how well the methods 2, and 3 agreed with method 1 over time.

**See Also**

[lcc](#).

**Examples**

```
data(simulated_hue)
summary(simulated_hue)
str(simulated_hue)
```

---

simulated\_hue\_block      *Hue color simulated data in a randomized block design*

---

**Description**

Simulated hue data set based on papaya's maturation over time considering four methods of measurement in a randomized block design.



**Usage**

```
data(simulated_hue_block)
```

**Format**

A simulated data frame with 24,000 observations on the mean hue variable. The format is:

Hue	numeric; mean hue of papaya's peel
Block	factor with levels labelled from 1 to 3
Method	a factor with levels labelled from Method 1, to Method 4
Time	integer; time in days from 0 to 19
Fruit	a factor with 300 levels labelled from 1 to 300 where each level is represented by one fruit.

**Details**

A total of 100 fruits by block were observed over 20 days by four methods to evaluate the mean hue of fruit's peel. We considered three blocks in this simulation. The aim of the agreement study was to assess how well the methods 2, 3, and 4 agreed with method 1 over time.

**See Also**

[lcc](#).

**Examples**

```
data(simulated_hue_block)
summary(simulated_hue_block)
str(simulated_hue_block)
```

---

summary.lcc

*Summarize an lcc Object*

---

**Description**

Additional information about the fit of longitudinal concordance correlation, longitudinal Pearson correlation, and longitudinal accuracy represented by an object of class [lcc](#). The returned object has a [print](#) method.

**Usage**

```
## S3 method for class 'lcc'
summary(object, type, adjustSigma, verbose, ...)
```

**Arguments**

object	an object inheriting from class <code>lcc</code> , representing a fitted longitudinal concordance correlation function.
type	an optional character string specifying the type of output to be returned. If <code>type="model"</code> , prints the summary of the polynomial mixed-effects regression model. If <code>type="lcc"</code> , prints the summary of the fitted and sampled values for LCC, LPC, and LA as well as the concordance correlation coefficient between fitted values from the model and observed values as goodness of fit (gof) measurement. Defaults to <code>type="model"</code> .
adjustSigma	an optional logical value used when <code>type = model</code> . If TRUE and the estimation method used to obtain object was maximum likelihood, the residual standard error is multiplied by <code>sqrt(nobs/(nobs - npar))</code> . See <a href="#">summary.lme</a> for more information. Default is TRUE.
verbose	an optional logical value used to control the amount of output in the <code>print.summary.lme</code> method when <code>type = model</code> is used. Defaults to FALSE.
...	not used.

**Value**

an object inheriting from class `summary.lcc` including:

fitted	the fitted values extracted from the <code>lcc</code> object.
gof	the goodness of fit (gof) measurement is calculated using the concordance correlation coefficient between fitted and observed values. Value of 1 denote perfect concordance.
AIC	the Akaike Information Criterion corresponding to object.
BIC	the Bayesian Information Criterion corresponding to object.
logLik	If <code>REML=FALSE</code> , returns the log-likelihood value of the linear mixed-effects model; otherwise, the restricted log-likelihood is returned

**Author(s)**

Thiago de Paula Oliveira, <thiago.paula.oliveira@usp.br>

**See Also**

[AIC](#), [BIC](#), [print.summary.lcc](#), [lcc](#)

**Examples**

```
## Second degree polynomial model with random intercept, slope and
## quadratic term
fm1<-lcc(data = hue, subject = "Fruit", resp = "H_mean",
         method = "Method", time = "Time", qf = 2, qr = 2)
summary(fm1, type="model")
summary(fm1, type="lcc")
```

---

`vcov.lcc`*Extract Variance-Covariance Matrix of the Fixed Effects*

---

**Description**

Returns the variance-covariance matrix of a fitted lcc model object.

**Usage**

```
## S3 method for class 'lcc'  
vcov(object, ...)
```

**Arguments**

<code>object</code>	an object inheriting from class <code>lcc</code> , representing a fitted longitudinal concordance correlation function.
<code>...</code>	optional arguments passed to the <code>vcov.lme</code> function.

**Details**

See methods for `nlme` objects to get more details.

**Author(s)**

Thiago de Paula Oliveira, <thiago.paula.oliveira@usp.br>

**See Also**

[summary.lcc](#), [lccPlot](#), [lcc](#), [coef.lcc](#)

**Examples**

```
## Not run:  
fm1<-lcc(data = hue, subject = "Fruit", resp = "H_mean",  
         method = "Method", time = "Time", qf = 2, qr = 2)  
vcov(fm1)  
  
## End(Not run)
```

# Index

- \* **datasets**
  - hue, 8
  - simulated\_hue, 24
  - simulated\_hue\_block, 24
- \* **ggplot2**
  - lcc, 9
- \* **nlme**
  - lcc, 9

aes\_linetype\_size\_shape, 14  
AIC, 3, 26  
AIC.lcc, 2, 12  
anova.lcc, 3, 19  
anova.lme, 4, 19

BIC, 26  
BIC.lcc (AIC.lcc), 2

coef.lcc, 3, 5, 7, 12, 22, 23, 27

facet\_wrap, 15  
fitted.lcc, 6, 12

getVarCov.lcc, 7, 12  
ggplot, 15

hue, 8

lcc, 3–7, 9, 9, 15, 17–27  
lccPlot, 5, 6, 12, 14, 18, 19, 21, 27  
lmeControl, 11  
logLik, 16  
logLik.lcc, 16

marrangeGrob, 15

nlme, 4, 5, 7, 11, 16, 19, 22, 23, 27

par, 18  
pdClasses, 10  
plot.lcc, 12, 17  
plot.lm, 18  
plotControl, 14  
print, 19–21, 25  
print.anova.lcc, 19  
print.lcc, 12, 20  
print.summary.lcc, 21

ranef.lcc, 12, 22  
residuals.lcc, 12, 23

simulated\_hue, 24  
simulated\_hue\_block, 24  
summary.lcc, 3–7, 12, 17, 19–21, 23, 25, 27  
summary.lme, 26

varClasses, 10  
varExp, 10  
vcov.lcc, 3, 5, 7, 12, 23, 27  
viewport, 15