lcc: an R package to estimate the concordance correlation, Pearson correlation, and accuracy over time

Thiago P Oliveira [Corresp., Equal first author, 1], **Rafael A Moral** [Equal first author, 2], **Silvio S Zocchi** [3], **Clarice GB Demetrio** [3], **John Hinde** [1]

[1] School of Mathematics, Statistics and Applied Mathematics, National University of Ireland, Galway, Galway, Co. Galway, Ireland

[2] Department of Mathematics and Statistics, National University of Ireland, Maynooth, Maynooth, Co. Kildare, Ireland

[3] Exact Science, Universidade de São Paulo, Piracicaba, São Paulo, Brazil

Corresponding Author: Thiago P Oliveira
Email address: thiago.paula.oliveira@usp.br

Background and Objective: Observational studies and experiments in medicine, pharmacology, and agronomy are often concerned with assessing whether different methods/raters produce similar values over the time when measuring a quantitative variable. This paper aims to describe the statistical package lcc, for are, that can be used to estimate the extent of agreement between two (or more) methods over the time, and illustrate the developed methodology using three real examples.

Methods: The longitudinal concordance correlation, longitudinal Pearson correlation, and longitudinal accuracy functions can be estimated based on fixed effects and variance components of the mixed-effects regression model. Inference is made through bootstrap confidence intervals and diagnostic can be done via plots, and statistical tests.

Results: The main features of the package are estimation and inference about the extent of agreement using numerical and graphical summaries. Moreover, our approach accommodates both balanced and unbalanced experimental designs or observational studies, and allows for different within-group error structures, while allowing for the inclusion of covariates in the linear predictor to control systematic variations in the response. All examples show that our methodology is flexible and can be applied to many different data types.

Conclusions: The lcc package, available on the CRAN repository, proved to be a useful tool to describe the agreement between two or more methods over time, allowing the detection of changes in the extent of agreement. The inclusion of different structures for the variance-covariance matrices of random effects and residuals makes the package flexible for working with different types of databases.

lcc: an R package to estimate the concordance correlation, Pearson correlation, and accuracy over time

Thiago P. Oliveira[1,2], Rafael A. Moral[3], Silvio S. Zocchi[4], Clarice G. B. Demétrio[4], and John Hinde[1]

[1]**School of Mathematics, Statistics and Applied Mathematics, NUI Galway, Galway Ireland**
[2]**The Insight Centre for Data Analytics, NUI Galway, Galway Ireland**
[3]**Maynooth University, Maynooth, Co. Kildare, Ireland**
[4]**Luiz de Queiroz College of Agriculture - USP, Piracicaba/SP, Brazil**

Corresponding author:
Thiago Oliveira[1]

Email address: thiago.paula.oliveira@usp.br

ABSTRACT

Background and Objective: Observational studies and experiments in medicine, pharmacology, and agronomy are often concerned with assessing whether different methods/raters produce similar values over the time when measuring a quantitative variable. This paper aims to describe the statistical package `lcc`, for are, that can be used to estimate the extent of agreement between two (or more) methods over the time, and illustrate the developed methodology using three real examples.

Methods: The longitudinal concordance correlation, longitudinal Pearson correlation, and longitudinal accuracy functions can be estimated based on fixed effects and variance components of the mixed-effects regression model. Inference is made through bootstrap confidence intervals and diagnostic can be done via plots, and statistical tests.

Results: The main features of the package are estimation and inference about the extent of agreement using numerical and graphical summaries. Moreover, our approach accommodates both balanced and unbalanced experimental designs or observational studies, and allows for different within-group error structures, while allowing for the inclusion of covariates in the linear predictor to control systematic variations in the response. All examples show that our methodology is flexible and can be applied to many different data types.

Conclusions: The `lcc` package, available on the CRAN repository, proved to be a useful tool to describe the agreement between two or more methods over time, allowing the detection of changes in the extent of agreement. The inclusion of different structures for the variance-covariance matrices of random effects and residuals makes the package flexible for working with different types of databases.

INTRODUCTION

Agreement indices are generally used when the same experimental unit is measured by at least two methods or observers (King et al., 2007). Measurements of agreement between raters or methods can be used in any field to explore their interchangeability considering a certain degree of agreement between the measurements they provide (Barnhart and Williamson, 2001; Chen and Barnhart, 2013). In biomedical sciences it is often necessary to study the reproducibility of continuous measurements made using specific diagnostic tools or methods, and that measurements can be taken over the time on the subjects of interest, such as in the studies of Pandit et al. (2019); Shinar et al. (2019) and Loecher et al. (2019).

The concordance correlation coefficient (CCC) introduced by Lin (1989) is a statistic commonly used to measure the agreement between methods when the response is continuous. Let Y_1 and Y_2 be two

random variables with a joint normal distribution

$$\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} \sim N_2\left(\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix}, \mathbf{\Sigma} = \begin{bmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{bmatrix}\right).$$

Here the expected value of the squared difference between Y_1 and Y_2 can be used as an agreement value. However, it ranges from 0 (perfect agreement) to infinity, which makes its interpretation difficult. Lin (1989) proposed standardizing this agreement index so that its values lie between -1 and $+1$:

$$\rho_{CCC} = 1 - \frac{\mathrm{E}\left[(Y_1 - Y_2)^2\right]}{\sigma_1^2 + \sigma_2^2 + (\mu_1 - \mu_2)^2} = \frac{2\sigma_{12}}{\sigma_1^2 + \sigma_2^2 + (\mu_1 - \mu_2)^2} = \rho C_b,$$

where $\mu_1 = \mathrm{E}(Y_1)$, $\mu_2 = \mathrm{E}(Y_2)$, $\sigma_1^2 = \mathrm{Var}(Y_1)$, $\sigma_2^2 = \mathrm{Var}(Y_2)$, and $\sigma_{12} = \mathrm{Cov}(Y_1, Y_2)$. This coefficient takes the value -1 when there is perfect disagreement, zero when there is no agreement, and $+1$ when there is perfect agreement. Moreover, ρ, the Pearson correlation coefficient ($|\rho| \leq 1$), measures how far each observation deviated from the best-fit line (a precision measure), and C_b, the accuracy ($0 < C_b \leq 1$), measures how far the best-fit line deviates from the 45° line through the origin, defined as $C_b = 2\left(v + v^{-1} + u^2\right)^{-1}$, where $v = \sigma_1^2/\sigma_2^2$ is a scale shift and $u = (\mu_1 - \mu_2)/\sqrt{\sigma_1 \sigma_2}$ is a location shift relative to the scale (Lin, 1989). Note that $C_b = 1$ indicates no deviation from the 45° line.

When pairs of samples (Y_{i1k}, Y_{i2k}),for $i = 1, 2, \ldots, N$ subjects and $k = 1, 2, \ldots K$ repeated measures, corresponding to observations on the same subject or experimental unit over time, the use of generalized multivariate analysis of variance to compute a weighted version of the CCC for repeated measurements is recommended (Chinchilli et al., 1996). Moreover, this coefficient has also been expanded to assess the agreement between more than two methods (King and Chinchilli, 2001).

When it is necessary to add extra variability sources due to within-subject measurements and/or other covariates in the model, the CCC can be estimated through the variance components (VC) of a mixed-effects model (Carrasco et al., 2009). The advantages of the mixed-effects models are that they give a general approach to analyse repeated measures and unbalanced data; they allow for the inclusion of different variance-covariance structures for both random effects and sampling errors. The restricted maximum likelihood (REML) approach can be used to obtain unbiased estimates of the VC.

Nevertheless, sometimes the researcher is not interested in reducing the CCC for repeated measurements to a single value, as proposed by Carrasco et al. (2009) and Carrasco et al. (2013), but in describing the extent of agreement between methods over time. To do this, we can consider a linear or non-linear function of the time and/or covariates in the model to describe the response variable, as proposed by Rathnayake and Choudhary (2017) and Oliveira et al. (2018). Here, we present the implementation of this methodology as an `R` (R core Team, 2019) package `lcc` (Oliveira et al., 2019), which provides functions for estimating the longitudinal concordance correlation (LCC) between methods based on variance components and fixed effects using polynomial mixed-effects models. It also computes estimates for the longitudinal Pearson correlation (LPC), which measures the precision, and the longitudinal bias correction factor (LA), which provides an accuracy measure.

The `lcc()` function gives fitted values and non-parametric bootstrap confidence intervals for the LCC, LPC, and LA statistics. Moreover, they can be estimated using different structures for the variance-covariance matrices of the random effects and different variance functions to model heteroskedasticity of within-group errors, with the option of using time as a variance covariate.

The remainder of the paper is organized as follows: Section introduces the theoretical definition of the LCC. Section introduces the `lcc()` function input and output, describing in detail the various options as well the `summary()` and other generic methods. Section briefly discusses model specification, which is illustrated more extensively in Section using three real data examples. The first and third shows an application in biomedical science, while the second from food science was the motivation for the development of the methodology and software and nicely shows the utility of the approach. Finally, Section presents some final remarks about the `lcc` package.

MODELS AND COMPUTATIONAL METHODS

Suppose a researcher is interested in investigating the extent of agreement between two or more methods, indexed as $j = 1, 2, \ldots, J$. Let N be the number of subjects in the experiment or observational study,

indexed as $i = 1,2,\ldots,N$, and suppose that each subject is observed n_i times (visits) with associated nuisance factors and/or covariates, these could include, for example, the effect of block or group. Let y_{ijk} be a realization of a random variable Y_{ijk} measured on the i-th subject by the j-th method at time t_k, $k = 1,2,\ldots,n_i$, with additional subject level (nuisance) covariates $\boldsymbol{x}_i$. Here t_k assumes values of the time covariate $t \in \mathscr{T}$, where $\mathscr{T}$ denotes the set of measurement times. Hence, the linear mixed-effects model including a polynomial function of time per method, random effects of subject, as well random effects for as subject/time interactions, is given by

$$Y_{ijk} = \boldsymbol{\gamma}^T \boldsymbol{x}_i + \sum_{h=0}^{p} \beta_{hj} t_{ik}^h + \sum_{h=0}^{q} b_{hi} t_{ik}^h + \varepsilon_{ijk}, \tag{1}$$

$$\text{with } \boldsymbol{b}_i \sim \text{MVN}(\boldsymbol{0}, \boldsymbol{G}) \text{ and } \boldsymbol{\varepsilon}_i \sim \text{MVN}(\boldsymbol{0}, \boldsymbol{R}_i),$$

where $h = 1,2,\ldots,q,q+1,\ldots,p$ is an index identifying the degree of the polynomial, with $q \leq p$; Y_{ijk} is the response measured on the i-th subject by the j-th method at time t_{ik}; t_{ik} represents the time (seconds, minutes, days, etc) at which the i-th individual was observed; $\boldsymbol{\gamma}$ is a vector of fixed effect parameters for the subject level covariates; $\boldsymbol{\beta}_j = \left[\beta_{0j}, \beta_{1j}, ..., \beta_{pj}\right]^T$ is a $(p+1)$-dimensional vector of fixed effects for the j-th method; $\boldsymbol{b}_i = [b_{0i}, b_{1i}, ..., b_{qi}]^T$ is a $(q+1)$-dimensional vector of random effects with mean vector $\boldsymbol{0}$ and covariance matrix $\boldsymbol{G}$; $\boldsymbol{\varepsilon}_i$ is a $(J \times n_i)$-dimensional error vector assumed to be independent for different i and independent of the random effects, with independent entries over j and k, with mean vector $\boldsymbol{0}$ and diagonal variance matrix $\boldsymbol{R}_i$.

Under model (1), the longitudinal concordance correlation (LCC) function between methods j and j', $j \neq j'$, is given by

$$\rho_{jj'}(t_k) = \frac{\boldsymbol{t}_k \boldsymbol{G} \boldsymbol{t}_k^T}{\boldsymbol{t}_k \boldsymbol{G} \boldsymbol{t}_k^T + \frac{1}{2}\left\{\sigma_\varepsilon^2 \left[g(t_k, \boldsymbol{\delta}_j) + g\left(t_k, \boldsymbol{\delta}_{j'}\right)\right] + S_{jj'}^2(t_k)\right\}} = \rho_{jj'}^{(p)}(t_k) C_{jj'}(t_k) \tag{2}$$

where $S_{jj'}(t_k) = \boldsymbol{t}_k\left(\boldsymbol{\beta}_j - \boldsymbol{\beta}_{j'}\right)$ is the systematic difference between methods j and j'; $\boldsymbol{t}_k^T = \left(t_k^0, t_k^1, \ldots, t_k^q\right)^T$; $g(\cdot)$ is a variance function assumed continuous in $\boldsymbol{\delta}$; δ_j is a vector of variance parameters for observations measured by j-th method or observer. We have that $\rho_{jj'}^{(p)}(t_k)$ is the longitudinal Pearson correlation (LPC) that measures how far each observation deviated from the best-fit line at a fixed time $t_k = t$, given by

$$\rho_{jj'}^{(p)}(t_k) = \frac{\boldsymbol{t}_k \boldsymbol{G} \boldsymbol{t}_k^T}{\sqrt{\left[\boldsymbol{t}_k \boldsymbol{G} \boldsymbol{t}_k^T + \sigma_\varepsilon^2 g(t_k, \boldsymbol{\delta}_j)\right]\left[\boldsymbol{t}_k \boldsymbol{G} \boldsymbol{t}_k^T + \sigma_\varepsilon^2 g\left(t_k, \boldsymbol{\delta}_{j'}\right)\right]}}.$$

$C_{jj'}(t_k)$, the longitudinal accuracy (LA), measures how far the best-fit line deviates from the 45° line at a fixed time $t_k = t$, given by

$$C_{jj'}(t_k) = \frac{2}{v_{jj'}(t_k) + \left[v_{jj'}(t_k)\right]^{-1} + u_{jj'}^2(t_k)},$$

where

$$v_{jj'}(t_k) = \sqrt{\frac{\text{Var}\left(Y_{ijkl}\right)}{\text{Var}\left(Y_{ij'kl}\right)}} = \sqrt{\frac{\boldsymbol{t}_k \boldsymbol{G} \boldsymbol{t}_k^T + \sigma_\varepsilon^2 g(t_k, \boldsymbol{\delta}_j)}{\boldsymbol{t}_k \boldsymbol{G} \boldsymbol{t}_k^T + \sigma_\varepsilon^2 g\left(t_k, \boldsymbol{\delta}_{j'}\right)}}$$

denotes the scale shift at time $t_k = t$, and

$$\begin{aligned} u_{jj'}(t_k) &= \frac{\text{E}\left(Y_{ijkl}\right) - \text{E}\left(Y_{ij'kl}\right)}{\left[\text{Var}\left(Y_{ijkl}\right)\text{Var}\left(Y_{ij'kl}\right)\right]^{\frac{1}{4}}} \\ &= \frac{\boldsymbol{t}_k\left(\boldsymbol{\beta}_j - \boldsymbol{\beta}_{j'}\right)}{\left\{\left[\boldsymbol{t}_k \boldsymbol{G} \boldsymbol{t}_k^T + \sigma_\varepsilon^2 g(t_k, \boldsymbol{\delta}_j)\right]\left[\boldsymbol{t}_k \boldsymbol{G} \boldsymbol{t}_k^T + \sigma_\varepsilon^2 g\left(t_k, \boldsymbol{\delta}_{j'}\right)\right]\right\}^{\frac{1}{4}}} \end{aligned}$$

denotes the location shift at time t_k relative to the scale (Lin, 1989; Oliveira et al., 2018). Consequently, when $\text{Var}\left(Y_{ijkl}\right) = \text{Var}\left(Y_{ij'kl}\right)$ and $\text{E}\left(Y_{ijkl}\right) = \text{E}\left(Y_{ij'kl}\right)$ then $C_{jj'}(t_k) = 1$ and there is no deviation from the 45° line.

Estimation and Inference

Point estimation and statistical inference for the LCC $\left(\rho_{jj'}(t_k)\right)$ has been proposed by Oliveira et al. (2018). It is estimated by replacing $\boldsymbol{\beta}$ and the variance components by their respective REML estimates:

$$\widehat{\rho}_{jj'}(t_k) = \frac{\boldsymbol{t}_k \widehat{\boldsymbol{G}} \boldsymbol{t}_k^T}{\boldsymbol{t}_k \widehat{\boldsymbol{G}} \boldsymbol{t}_k^T + \frac{1}{2}\left\{\widehat{\sigma}_\varepsilon^2 \left[\widehat{g}\left(t_k, \widehat{\boldsymbol{\delta}}_j\right) + \widehat{g}\left(t_k, \widehat{\boldsymbol{\delta}}_{j'}\right)\right] + \widehat{S}^2_{jj'}(t_k)\right\}}.$$

Since the variance components are estimated using the REML approach, their estimates are asymptotically normally distributed and the bias is smaller when compared to the maximum likelihood (ML) approach. Moreover, Oliveira et al. (2018) showed a satisfactory performance of the LCC even in settings with severe imbalance and only a small number of subjects ($N = 20$).

A confidence interval (CI) for $\rho_{jj'}(t_k)$ can be constructed using a nonparametric bootstrap based on M (e.g. 5000) bootstrap samples with either the percentile method (recommended for $N \leq 30$) or, otherwise, a normal approximation confidence interval, as described by Oliveira et al. (2018).

When we use a normal approximation for the CI, the Fisher Z-transformation given by

$$\rho^*_{j,j'}(t_k) = \frac{1}{2}\ln\left[\frac{1+\rho_{j,j'}(t_k)}{1-\rho_{j,j'}(t_k)}\right]$$

should be used with the normal approximation made to the empirical distribution of $\rho^*_{j,j'}(t_k)$ (Lin, 1989). Consequently, the confidence limits can be estimated using the bootstrap estimator of $\rho^*_{j,j'}(t_k)$ for a fixed time $t_k = t$ given by

$$\widehat{\rho}^*_{j,j'}(t_k = t) = \frac{1}{2M}\sum_{m=1}^{M} \ln\left[\frac{1+\widehat{\rho}^{(m)}_{j,j'}(t)}{1-\widehat{\rho}^{(m)}_{j,j'}(t)}\right], \quad m = 1,2,\ldots,M,$$

where $\left\{\widehat{\rho}^{(m)}_{j,j'}\right\}$ are the estimates from the M bootstrap samples. The standard deviation of the bootstrap distribution of $\widehat{\rho}^*_{j,j'}(t_k)$ for a fixed time $t_k = t$ given by

$$\widehat{SE}^*_{j,j'}(t_k = t) = \sqrt{\frac{1}{M-1}\sum_{m=1}^{M}\left[\frac{1}{2}\ln\left(\frac{1+\widehat{\rho}^{(m)}_{j,j'}(t)}{1-\widehat{\rho}^{(m)}_{j,j'}(t)}\right) - \widehat{\rho}^*_{j,j'}(t)\right]^2}.$$

Thus, an approximate bootstrap confidence interval of level $(1-\alpha)$ for $\rho_{j,j'}$ is $[LB, UB]$, where

$$LB = \frac{\exp\left\{2\left[\widehat{\rho}^*_{j,j'}(t_k = t) - z_{(1-\frac{\alpha}{2})}\widehat{SE}^*_{j,j'}(t_k = t)\right]\right\} - 1}{\exp\left\{2\left[\widehat{\rho}^*_{j,j'}(t_k = t) - z_{(1-\frac{\alpha}{2})}\widehat{SE}^*_{j,j'}(t_k = t)\right]\right\} + 1}$$

and

$$UB = \frac{\exp\left\{2\left[\widehat{\rho}^*_{j,j'}(t_k = t) - z_{\frac{\alpha}{2}}\widehat{SE}^*_{j,j'}(t_k = t)\right]\right\} - 1}{\exp\left\{2\left[\widehat{\rho}^*_{j,j'}(t_k = t) - z_{\frac{\alpha}{2}}\widehat{SE}^*_{j,j'}(t_k = t)\right]\right\} + 1},$$

where $z_{\frac{\alpha}{2}}$ and $z_{(1-\frac{\alpha}{2})}$ denote the $\frac{\alpha}{2}$ and $(1-\frac{\alpha}{2})$ percentiles of the standard normal distribution.

On the other hand, the CI based on the percentile method uses the percentiles of the bootstrap distribution of $\widehat{\rho}_{j,j'}(t_k = t)$ directly and is given by

$$\left(\widehat{\rho}_{j,j'_{(\alpha/2)}}(t_k = t), \widehat{\rho}_{(j,j')_{(1-\alpha/2)}}(t_k = t)\right) \approx \left(\widehat{\rho}^{(m)}_{j,j'_{(\alpha/2)}}(t_k = t), \widehat{\rho}^{(m)}_{(j,j')_{(1-\alpha/2)}}(t_k = t)\right),$$

where $\widehat{\rho}^{(m)}_{(j,j')_{(\alpha/2)}}(t_k = t)$ and $\widehat{\rho}^{(m)}_{(j,j')_{(1-\alpha/2)}}(t_k = t)$ are the $(100\times\frac{\alpha}{2})-$th and $(100\times 1-\frac{\alpha}{2})-$th empirical percentiles of the $\widehat{\rho}^{(m)}_{j,j'}(t_k = t)$ values, $m = 1,2,\ldots,M$. If the bootstrap distribution of $\rho^*_{j,j'}(t_k = t)$ is approximately normal, then both proposed methods will give very similar confidence intervals as N increases.

Inference for $C_{jj'}(t_k)$ can be performed in a similar way as to that presented for the LCC. Since $C_{(j,j')_{(1-\alpha/2)}}(t_k = t)$ belongs to the interval $[0,1]$, we suggest the use the arc-sine transformation

$$C^*_{(j,j')_{(1-\alpha/2)}}(t_k = t) = \sin^{-1}\sqrt{C_{jj'}(t_k)}$$

instead of the Fisher Z-transformation, nor logistic transformation (used by Oliveira et al. (2018)) to approximate the distribution of $C_{(j,j')_{(1-\alpha/2)}}(t_k = t)$ by a normal distribution. Thus, the confidence limits can be estimated using the bootstrap estimator of $C^*_{j,j'}(t_k)$ for a fixed time $t_k = t$ given by

$$\widehat{C}^*_{j,j'}(t_k = t) = \frac{1}{M}\sum_{m=1}^{M}\sin^{-1}\sqrt{\widehat{C}^{(m)}_{j,j'}(t)}, \quad m = 1,2,\ldots,M,$$

and standard deviation of the bootstrap distribution of $\widehat{C}^*_{j,j'}(t_k)$ for a fixed time $t_k = t$ is given by

$$\widehat{SE}^*_{C_{j,j'}}(t_k = t) = \sqrt{\frac{1}{M-1}\sum_{m=1}^{M}\left[\sin^{-1}\sqrt{\widehat{C}^{(m)}_{j,j'}(t)} - \widehat{C}^*_{j,j'}(t)\right]^2}.$$

Therefore, an approximate bootstrap confidence interval of level $(1-\alpha)$ for $\widehat{C}_{j,j'}$ is $[LB_C, UB_C]$, where

$$LB_C = \text{sign}\left[\widehat{C}^*_{j,j'}(t_k = t) - z_{(1-\frac{\alpha}{2})}\widehat{SE}^*_{C_{j,j'}}(t_k = t)\right]\left\{\sin\left[\widehat{C}^*_{j,j'}(t_k = t) - z_{(1-\frac{\alpha}{2})}\widehat{SE}^*_{C_{j,j'}}(t_k = t)\right]\right\}^2$$

and

$$UB_C = \text{sign}\left[\widehat{C}^*_{j,j'}(t_k = t) - z_{\frac{\alpha}{2}}\widehat{SE}^*_{C_{j,j'}}(t_k = t)\right]\left\{\sin\left[\widehat{C}^*_{j,j'}(t_k = t) - z_{\frac{\alpha}{2}}\widehat{SE}^*_{C_{j,j'}}(t_k = t)\right]\right\}^2,$$

where $z_{\frac{\alpha}{2}}$ and $z_{(1-\frac{\alpha}{2})}$ denote the $(\frac{\alpha}{2})$ and $(1-\frac{\alpha}{2})$ quantiles of the standard normal distribution. Bootstrap percentile intervals are calculated in the obvious way from the bootstrap values $\widehat{C}^{(m)}_{j,j'}(t)$, $m = 1,2\ldots,M$.

OVERVIEW OF THE PACKAGE `lcc` AND `R` SYNTAX

This section provides some details on the implementation of the function `lcc` and explains its technical arguments, whose default settings were carefully chosen. The package is freely available for download from the CRAN website https://CRAN.R-project.org/package=lcc, and installation can be performed using

```
R> install.packages("lcc")
R> library(lcc)
```

The `lcc` package has 21 arguments that are briefly summarised in Table 1.

Table 1. Input arguments for LCC package

Argument	Type	Description	Default	Required
`data`	`data.frame`	Specifies the input dataset		Yes
`resp`	Character string	Name of the response variable		Yes
`subject`	Character string	Name of the subject variable		Yes
`method`	Character string	Name of the method variable		Yes
`time`	Character string	Name of the time variable		Yes

Continued on next page

Table 1 – continued from previous page

Argument	Type	Description	Default	Required
`interaction`	Logical	an option to estimate the interaction effects between method and time. If `TRUE` the interaction effects are estimated. If `FALSE` only the main effects of time and method are estimated	`TRUE`	No
`qf`	Numeric	An integer specifying the degree of the polynomial time trends, usually 1, 2 or 3 (0 is not allowed).	1	No
`qr`	Numeric	An integer specifying terms having random effects to account for subject-to-subject variation, such that `qr`$\leq$`qf`, and `qr=0` means there is just a random intercept.	0	No
`covar`	Character vector	Names of the covariates (factors and/or variables) to include in the model as fixed effects, e.g. block, group, etc.	`NULL`	No
`gs`	Character string	Name of method level which represents the gold-standard.	first level	No
`pdmat`	Function	Standard classes of positive-definite matrix structures available in the `nlme` package.	`pdSymm`	No
`var.class`	Function	Standard classes of variance function structures used to model the variance structure of within-group errors using covariates.	`NULL`	No
`weights.form`	Formula	An one-sided formula specifying a variance covariate and, optionally, a grouping factor for the variance parameters in the `var.class`. If `var.class = varIdent`, the form `"method"`, (or $\sim$ `1 | method`), or `"time.ident"` ($\sim$ `1 | time`), must be used. If `var.class` = `varExp`, the form `"time"` ($\sim$ `time`), or `"both"` ($\sim$ `time | method`), must be used.	`NULL`	No[1]
`time_lcc`	List	Regular sequence for time variable merged with specific or experimental time values used for LCC, LPC, and LA predictions.	`NULL`	No
`ci`	Logical	An optional non-parametric bootstrap confidence interval for the LCC, LPC and LA statistics. If `TRUE` confidence intervals are calculated and printed in the output.	`FALSE`	No

Continued on next page

[1]Required when `var.class` is specified.

Table 1 – continued from previous page

Argument	Type	Description	Default	Required
`percentileMet`	Logical	an optional method for calculating the non-parametric bootstrap intervals. If `FALSE` the normal approximation method is used. If `TRUE` the percentile method is used.	`FALSE`	No[2]
`alpha`	Numeric	Confidence level for the CI.	0.05	No[2]
`nboot`	Numeric	An integer specifying the number of bootstrap samples.	5000	No[2]
`show.warnings`	Logical	an optional argument that shows the number of convergence errors in the bootstrap samples. If `TRUE` shows in which bootstrap samples the errors occurred. If `FALSE` shows the total number of convergence errors.	`FALSE`	No
`components`	Logical	An option to estimate the LPC and LA statistics. If `TRUE` the estimates and confidence intervals for LPC and LA are printed in the output. If `FALSE` provides estimates and confidence intervals only for the LCC statistic.	`FALSE`	No
`REML`	Logical	The estimation method. If `TRUE` the model is fit by maximizing the restricted log-likelihood. If `FALSE` full maximum likelihood is used.	`TRUE`	No
`lme.control`	List	A list of control values passed to the estimation algorithm to replace the default values of the function `lmeControl` available in the `nlme` package.	empty list	No

We present a more detailed description of some arguments below:

1. `data`: must be a data frame containing the following variables: response, subject identification, method, and time;

2. `method`: name of the method variable in the dataset. The `lcc` package recognizes the first level of the variable associated with this argument as the gold-standard method, and then compares it with all other levels;

3. `qr`: when we specify `qr = 0` a random intercept is included in the polynomial model while `qr = 1` specifies random intercepts and slopes. If `qr = qf = q`, with $q \geq 1$, all polynomial terms are specified to have random effects at the individual level.

4. `time_lcc`: a named list with values for arguments `time`, `from`, `to`, and `n` used in the `time_lcc()` function to generate a regular sequence merged with specific or experimental time values of the time variable used for LCC, LPC and LA predictions. Argument `time` is a vector of specific or experimental time values of a given length, where the experimental time values are used as default; `from` and `to` are used to define, respectively, the starting and end values of the time variable, and `n` is used to define the desired length of the sequence. We recommend a grid $\boldsymbol{t} = (t_1, t_2, \ldots, t_{n^*})^T$ of n^* points in $\mathscr{T}$ to construct the agreement curve and confidence intervals. In practice, n^* between 30 and 50 is generally adequate. Example:

[2]It can only be specified when `ci = TRUE`

```
R> Time <- seq(0,20,1)
R> str(tk <- time_lcc(time=Time, from=min(Time), to=max(Time),
+                      n=30))
num [1:49] 0 0.69 1 1.38 2 ...
```

5. `pdmat`: the `lcc` package provides six standard classes of positive-definite matrix structures that can be included in the model to estimate the LCC, LPC and LA statistics. Available standard classes are `pdSymm`, `pdLogChol`, `pdDiag`, `pdIdent`, `pdCompSymm`, and `pdNatural`. More information about these classes are available in Pinheiro and Bates (2000).

6. `var.class`: a class of variance functions that are used to model the variance structure of within-group errors using covariates (Pinheiro and Bates, 2000). We generalize this class as

$$\text{Var}\left(\varepsilon_{ijk}\right) = \sigma_{\varepsilon}^2 g\left(t_k, \boldsymbol{\delta}\right), \tag{3}$$

where $g(\cdot)$ is the variance function assumed continuous in $\boldsymbol{\delta}$; t_k is the time covariate and $\boldsymbol{\delta}$ is a vector of variance parameters. The `lcc` package provides two different standard variance functions classes that are included in the `nlme` library (Pinheiro et al., 2017).

The first one is the `varIdent` class that represent a variance model with different variances for each level of a stratification variable s, $s = 1, 2, \ldots, S$, given by

$$\text{Var}\left(\varepsilon_{ijk}\right) = \sigma_{\varepsilon}^2 \delta_{s_{ijk}}^2.$$

As we have $S+1$ parameters to represent S variances, we need to add the restriction $\delta_1 = 1$, and consequently $\delta'_{s^*} = \delta_{s^*}/\delta_1$, $s^* = 2, 3, \ldots, S$ and $\delta'_{s^*} > 0$. Here each level of method/observer or time represents a stratum of a homogeneous subgroup.

The second variance function is an exponential function of the variance covariate, the `varExp` class, represented as

$$\text{Var}\left(\varepsilon_{ijk}\right) = \sigma_{\varepsilon}^2 \exp\left(2\delta_{s_{ijk}} t_k\right)$$

where $\delta_{s_{ijk}}$ is unrestricted, so the variance model (4) allows $\text{Var}\left(\varepsilon_{ijk}\right)$ to increase or decrease over time.

7. `weights.form`: a `varFunc` class object, representing a constructor to the `form` argument in the `nlme` library. The `weights.form` argument is based on a one-sided formula specifying a variance covariate and, optionally, a grouping factor for the variance parameters. Moreover, this argument must be specified only when `var.class` is specified as well.

The first class `varIdent` represents a variance model with different variances for each level of the grouping factor and has two options of `weights.form` in the `lcc` package:

   (a) `"method"`: specifies a variance model with different variances for each level of factor method/observer and is given by

$$\text{Var}\left(\varepsilon_{ijk}\right) = \sigma_{\varepsilon}^2 \delta_{\text{method}_j}^2, \quad j = 1, 2, \ldots, J,$$

   where $g\left(\text{method}_j, \delta_j\right) = \delta_{\text{method}_j}^2$ is the variance function, and δ_{method_j} is the variance parameter for observations measured by the jth method. The `form` argument in the `varFunc` is `form = ~ 1|method`;

   (b) `"time.ident"`: specifies a variance model with different variances for each level of stratification in the time variable and is given by

$$\text{Var}\left(\varepsilon_{ijk}\right) = \sigma_{\varepsilon}^2 \delta_{t_k}^2, \quad k = 1, 2, \ldots, K,$$

   where $g\left(t_k, \delta_k\right) = \delta_{t_k}^2$ is the variance function, and δ_{t_k} is the variance parameter for observations measured at time $t_k = t$, with $t \in [t_0, t_K]$ and $t_0 \geq 0$. The `form` argument in the `varFunc` class is `form = ~ 1|time`.

The class `varExp` represents a variance model whose variance function $g(.)$ is an exponential function of the variance covariate. This class has also two options of `weights.form` in the `lcc` package:

(a) `"time"`: specifies a variance model given by

$$\mathrm{Var}\left(\varepsilon_{ijk}\right) = \sigma_{\varepsilon}^{2} \exp\left(2\delta t_k\right),$$

where the variance function $g\left(t_k, \delta\right) = \exp\left(2\delta t_k\right)$ is an exponential function of the time $t_k = t$; and δ is the variance parameter. The `form` argument in the `varFunc` class is `form = ~ time`;

(b) `"both"`: specify a variance model for each level of the factor method given by

$$\mathrm{Var}\left(\varepsilon_{ijk}\right) = \sigma_{\varepsilon}^{2} \exp\left(2\delta_{\mathrm{method}_j} t_k\right), \quad j = 1, 2, \ldots, J,$$

where the variance function $g\left(t_k, \mathrm{method}_j, \delta\right) = \exp\left(2\delta_{\mathrm{method}_j} t_k\right)$ is an exponential function of the time $t_k = t$ for each level of method; and $\delta_{\mathrm{method}_j}$ is the variance parameter for the jth level of method. The form argument in the `varFunc` class is `form = ~ time|method`;

The `lcc` package uses the REML method as default because it is less biased, less sensitive to outliers, and deals more effectively with high correlations when compared to standard ML estimation (Harville, 1977; Giesbrecht and Burns, 1985). However, we offer the user the possibility to change the estimation method to ML because this approach should be used when comparing models with nested fixed effects but with the same random effects structure. Furthermore, the package depends on the `nlme` (Pinheiro et al., 2017), and `ggplot2` (Wickham, 2009) packages, and imports some functions from packages `gdata` (Warnes et al., 2017), `gridExtra` (Auguie and Antonov, 2017), and `hnp` (Moral et al., 2017).

Generic functions and outputs

A typical call of the `lcc` function is similar to a call to `lme` as the LCC estimation is based on a mixed-effects regression model. Several variations in the specifications of linear mixed-effects models to estimate the LCC are possible, and we can query the fitted `lcc` object through different generic functions. Table 2 gives details of a set of S3 generic extractor functions for objects of class `lcc`.

The output of the `summary()` function includes the values of Akaike Information Criterion (AIC) (Akaike, 1974), the Bayesian Information Criterion (BIC) (Schwarz, 1978), log-likelihood value, and a goodness of fit measurement `gof`, which is calculated using the concordance correlation coefficient (Lin, 1989) between fitted values extracted from the mixed-effects model and observed values. This measure can be used, with care, to describe the overall agreement between observed and fitted values, where a value equal to -1 represents a perfect disagreement between them, zero represents no agreement, and $+1$ perfect agreement. Clearly, a high model performance is related with a high positive value of `gof` (generally between 0.8-1).

The fitted curves of LCC, LPC, or LA values versus the time covariate, as well as their bootstrap confidence intervals, can be visualised through the `lccPlot()` function, which is specified as `lccPlot(obj, type, control)`, where `obj` is an object of class `lcc`; `type` specifies required output that could be `type="lcc"` for the LCC, the default, `type="lpc"` for the LPC, or `type="la"` for the LA statistics; and `control` is a list of control values or character strings returned by the `plotControl()` function used to modify the plot structure. This function uses the `ggplot2` package internally to build the final plot, where predicted values are joined by lines, sampled observations are represented by circles, and confidence intervals by a ribbon (grey as default) defined by its lower and upper bounds.

SPECIFYING MODELS IN THE `LCC()` FUNCTION

In the `lcc` package, to describe the LCC we need to specify the subject, response, method and time variables, a polynomial mixed-effect model, and the data. These arguments are specified through an

Table 2. Generic functions for use with objects of class `lcc`

Function	Description
`print()`	a simple printed display
`summary()`	returns an object of class `summary.lcc` containing the relevant summary statistics (which has a `print()` method). If `type = "lcc"` it provides information about $\rho_{jj'}(t_k)$, and if `components = TRUE` in the `lcc()` function, also provides information about $\rho^{(p)}_{jj'}(t_k)$, and $C_{jj'}(t_k)$. If `type = "model"` it provides additional information about the linear mixed-effects fit. The default is `type = "model"`.
`anova()`	Summarise and compare likelihoods of fitted models from `lcc` objects
`coef()`	The fixed effects estimated and corresponding random effects estimates are obtained at subject levels less or equal to N. The resulting estimates are returned as a data frame, with rows corresponding to subject levels and columns as coefficients.
`fitted()`	Fitted values for $\widehat{\rho}_{jj'}(t_k)$, $\widehat{\rho}^{(p)}_{jj'}(t_k)$, or $\widehat{C}_{jj'}(t_k)$. The output depends on the argument `type`, where `type = "lcc"` (the default), `type = "lpc"`, or `type = "la"` gives output for $\widehat{\rho}_{jj'}(t_k)$, $\widehat{\rho}^{(p)}_{jj'}(t_k)$, or $\widehat{C}_{jj'}(t_k)$, respectively.
`getVarCov()`	Returns the variance components estimates.
`residuals()`	Extract residuals (response, Pearson, and normalized), defaulting to Pearson. residuals
`ranef()`	Extract the estimated random effects.
`vcov()`	Returns the variance-covariance matrix of the fixed effects.
`AIC()`	Compute the Akaike criterion
`BIC()`	Compute the Bayesian criterion
`logLik()`	Extract the log-likelihood
`plot()`	A series of six built-in diagnostic plots to evaluate the assumptions underlying the linear mixed-effects regression model. Comprises: a plot of conditional residuals against fitted values; plot of conditional residuals over time; box-plot of residuals given subject; observed against fitted values; normal Q-Q plot with simulation envelopes for the conditional errors; and normal Q-Q plot with simulation envelopes for the random effects are provided.

easy-to-use syntax. Consider a first degree polynomial model with random intercepts for a continuous dependent variable y observed on N subjects ($i = 1,2,\ldots,N$) using J methods at times t_k ($k = 1,2,\ldots,n_i$). Such model can be written as

$$Y_{ijk} = \beta_{0j} + b_{0i} + \beta_{1j}t_k + \varepsilon_{ijk}, \text{ with}$$
$$b_{0i} \sim N(0,\sigma^2_{b_0}) \quad \text{and} \quad \varepsilon_{ijk} \sim N\left(0,\sigma^2_{\varepsilon}\right)$$

Thus, the LCC based on fixed effects and variance components at time t_k is given by

$$\rho_{jj'}(t_k) = \frac{\sigma^2_{b_0}}{\sigma^2_{b_0} + \sigma^2_{\varepsilon} + \frac{1}{2}\left[\beta_{01} - \beta_{02} + (\beta_{11} - \beta_{12})t_k\right]^2}$$

and the syntax to specify this model in the `lcc()` function is

```
R> library(lcc)
R> data(simulated_hue_block)
R> m1 <- lcc(data = simulated_hue_block, subject = "Fruit",
+             resp = "Hue", method = "Method",time = "Time",
+             qf = 1, qr = 0)
```

where `qf = 1` represents the polynomial degree for the fixed effects, and `qr = 0` specifies a random intercepts model. Here, the names of the columns in the dataframe `data` are supplied as strings to the arguments of the `lcc()` function.

Suppose now that the experimental design in the previous example was a randomized complete block design. Then, the fixed effect of blocks can be included in that model by specifying the `covar` argument, i.e.

```
R> m2 <- update(m1, covar ="Block")
```

If we suppose different variances for each level of the method factor, the corresponding model would include a variance function such as $g(\delta_j) = \sigma_\varepsilon^2 \delta_j^2$, and the syntax would then be

```
R> m3 <- update(m2, var.class = varIdent, weights.form = "method",
+                lme.control = list(opt="optim"))
```

To visualize the summary and graphical output of model `m3` we call `summary(m3)` and `lccPlot(m3)`, respectively.

Many other possible models can be built to estimate the LCC through the function `lcc()` options, see Section . Model selection can be performed using likelihood-ratio tests for nested models; or using the AIC or BIC criteria, e.g.

```
R> AIC(m2, m3); BIC(m2, m3); anova(m2, m3)
```

EXAMPLES

We will now use three example datasets, drawn from Lloyd et al. (1998), Martin et al. (2002) and Oliveira et al. (2018), to illustrate the implemented functions in the following sections of this paper. The first dataset is an observational study of a cohort of 82 adolescent females to assess the percentage body fat and the aim is to determine the agreement profile between measurements made over time using a skinfold caliper and dual-energy X-ray absorptiometry. The second is a canonical example from agriculture and was the motivation for the original development of these methods; here the goal is to investigate if a colorimeter can compete with a digital scanner in measuring the peel hue of papayas over time. The final example is again related to medicine and the goal here is to verify the agreement between cortisol concentration measured on patients every hour and every two hours.

Percentage body fat dataset

These data came from a longitudinal observational study conducted as part of the Penn State Young Women's Health Study (Lloyd et al., 1998). Percentage body fat was measured using skinfold calipers and dual-energy X-ray absorptiometry (DEXA) on a cohort of 82 adolescent white females attending public schools in Pennsylvania. The initial visit occurred at age 12 (baseline) and subsequent visits occurred every six months, in which one skinfold caliper and one DEXA measurement were taken to assess the percentage of body fat. As the skinfold measurement is the most frequently used method for laboratory and field studies, the objective was to determine the agreement profile between the skinfold caliper and DEXA measurements. Figure 1 shows that the agreement between skinfold and DEXA apparently decreases over the visits. King et al. (2007) explained that this phenomenon may occur because the

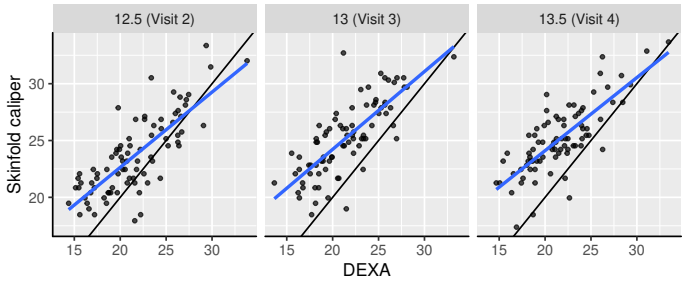


Figure 1. Scatter plot of body fat data, where the panels represent visits, the blue line is the best fit line, and the black line is the line of equality.

skinfold method is only capable of detecting subcutaneous fat, while DEXA detects subcutaneous, breast, lower body and visceral fat. Moreover, female adolescents may have a considerable fat increase in breast, lower body and/or visceral fat over this age range (King et al., 2007). Consequently, this reinforces the interest in estimating the agreement profile between these methods for the body fat measurements over ages ranging from 12.5 to 13.5 years old, rather than summarizing it in a single coefficient as proposed by King et al. (2007). Hence, we created a new variable called `TIME` given by $12 \times (\text{age} - 12)$, which represents the time in months after the first visit (baseline).

Now let y_{ijk} be the measurement taken on the i-th individual, by the j-th method at the k-th visit. We then fit a random intercepts and slopes linear regression model, given by

$$\begin{aligned} y_{ijk} &= \beta_{0j} + b_{0i} + \left(\beta_{1j} + b_{1i}\right) t_k + \varepsilon_{ijk} \\ \boldsymbol{b} = [b_{0i}, b_{1i}]^T &\sim N_2(\boldsymbol{0}, \boldsymbol{G}) \quad \text{and} \quad \varepsilon_{ijk} \sim N\left(0, \sigma_\varepsilon^2\right), \end{aligned} \tag{4}$$

where $\text{vech}(\boldsymbol{G}) = \left[\sigma_{b_0}^2, \sigma_{b_{01}}, \sigma_{b_1}^2\right]^T$ (vech$(\cdot)$ is the half-vectorization of a symmetric matrix $\boldsymbol{G}$ formed from only the lower triangular part). Using model (4), we estimate the LCC, LPC and LA statistics as well as their 95% bootstrap confidence intervals based on 10,000 pseudo-samples using the `lcc()` function:

```
R> data(bfat, package = "cccrm")
R> library(dplyr)
R> bfat <- bfat %>%
+          mutate(VISITNO = replace(VISITNO, VISITNO == 2, 12.5)) %>%
+          mutate(VISITNO = replace(VISITNO, VISITNO == 3, 13)) %>%
+          mutate(VISITNO = replace(VISITNO, VISITNO == 4, 13.5)) %>%
+          mutate(SUBJECT = factor(SUBJECT)) %>%
+          mutate(MET = factor(MET,  labels = c("1 hour",  "2 hours")))
R> bfat$TIME <- 12 * (bfat$VISITNO - 12)
R> m.bfat.1 <- lcc(data = bfat, subject = "SUBJECT", resp = "BF",
+              method = "MET", time = "TIME", qf = 1, qr = 1,
+              components = TRUE, ci = TRUE, nboot = 10000)
Convergence error in 951 out of 10000 bootstrap samples.
```

The output of model `m.bfat.1` indicates that in 951 (9.51%) of the pseudo-samples, the likelihood maximization algorithm failed to converge, where most of these failures were a consequence of specific bootstrap sample patterns. An alternative procedure to decrease the percentage of convergence failures is by increasing the iteration limit and/or changing the optimization method from `nlminb` to `optim`. In the `lcc()` function, the user can include a list of optimisation control additional arguments in the `lme.control()` function:

```
R> m.bfat.2 <- update(m.bfat.1, lme.control = list(opt = "optim"))
Convergence error in 72 out of 10000 bootstrap samples.
```

The output of `m.bfat.2` shows a lower number of failures (0.72%) compared with the previous approach. We proceed to examine the bootstrap confidence intervals computed for the LCC, LPC and LA:

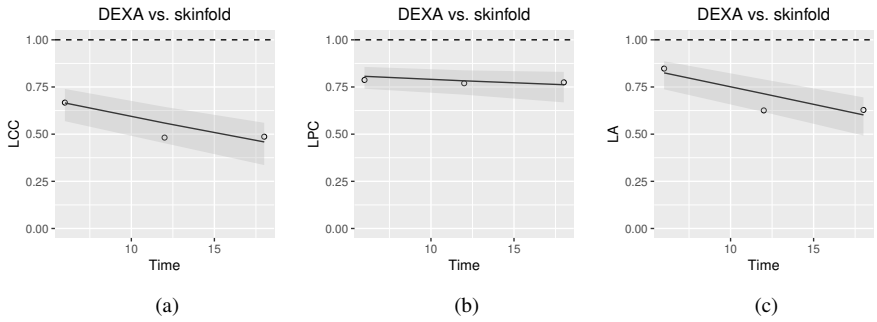
```
R> summary(m.bfat.2,  type = "lcc")
Longitudinal concordance correlation model fit by REML
AIC      BIC      logLik
2182.068  2215.59  -1083.034

gof: 0.9201

Lower and upper bound of % bootstrap confidence interval
Number of bootstrap samples:

DEXA vs. skinfold
$LCC
   Time     LCC        Lower       Upper
```

```
1    6   0.6653516   0.5687779   0.7395459
2   12   0.5589258   0.4516374   0.6442955
3   18   0.4588008   0.3353932   0.5599172

$LPC
   Time      LPC       Lower       Upper
1    6   0.8065578   0.7415331   0.8558988
2   12   0.7826493   0.7092871   0.8378992
3   18   0.7620551   0.6676806   0.8300397

$LA
   Time       LA       Lower       Upper
1    6   0.8249273   0.7431156   0.8898124
2   12   0.7141458   0.6201347   0.7923521
3   18   0.6020573   0.4934167   0.6961643
```

We may then plot the LCC, LPC, and LA with their respective confidence intervals by executing

```
R> lccPlot(m.bfat.2)
R> lccPlot(m.bfat.2, type = "lpc")
R> lccPlot(m.bfat.2, type = "la")
```

The estimates of LCC, LPC and LA, their confidence intervals, and figures indicate that the agreement and accuracy profiles between the skinfold caliper and DEXA measurements decrease over time, while the precision profile, represented by LPC, remains constant (Figure 2). Therefore, a first conclusion is that the agreement profile decreases over time because the accuracy is decreasing.



(a) (b) (c)

Figure 2. Estimate and 95% bootstrap confidence interval for the (a) longitudinal concordance correlation (LCC); (b) longitudinal Pearson correlation (LPC); and (c) longitudinal accuracy (LA) between percentage body fat measured on adolescent girls by skinfold caliper and DEXA. Points represent (a) the sample CCC, (b) sample Pearson correlation, and (c) sample accuracy.

Moreover, there is a moderate to weak agreement profile, where the greatest LCC estimate was 0.6654 at age 12.5 (95% CI: $[0.5688, 0.7395]$) and the smallest LCC estimate was 0.4588 at age 13.5 (95% CI: $[0.3354, 0.5599]$). This result reinforces the discussion presented by King et al. (2007), who provided physiological explanations for this phenomenon due to fact that the skinfold method is not capable to detect breast, lower body and visceral fat, which increases over this age range. Clearly, as the skinfold method detects less fat than the other, the accuracy between them tends to decrease since the expected value difference is greater (Figure 2(c)). The concordance correlation coefficient between fitted values of the mixed-effects model and observed values is presented as goodness of fit (`gof`) and was approximately 0.92. This result shows that the model can reproduce the observed values quite well.

The papaya peel hue dataset

In commercial fruit classification, one of the most important variables is the peel hue because it is used to determine fruit ripeness (Mendoza and Aguilera, 2004; Oliveira et al., 2017). This is very important to

plan harvesting procedures. In an experiment described in Oliveira et al. (2018), the hue component was measured for a sample of 20 papaya fruits using a flat-bed scanner (HP Scanjet G2410) and a colorimeter (Minolta CR-300) (Konica Minolta, 2003). The hue of each fruit was measured daily using both devices for a period of 15 days, where four equidistant points on the equatorial region were observed using a colorimeter, and 1,000 points over the same region were observed using a scanner. The circular mean hue was calculated for the ith fruit, $i = 1,2,\ldots,N$, measured by the jth method, $j = 1,2$ at time t_{ik}, $k = 1,2,\ldots,n_i$. As the multivariate von Mises distribution of the hue is highly concentrated around its overall mean, we assume that its distribution can be treated as a normal distribution with mean $\mu_{\bar{h}}$ and covariance matrix $\boldsymbol{R} = \boldsymbol{I}\sigma_{\varepsilon}^2$.

The aim of the agreement study here was to determine whether the scanner can reproduce the mean hue measurements taken by the colorimeter on the same fruit over time. The colorimeter is faster and easier to use than a flatbed scanner. Additionally, each image obtained with the scanner needs to processed by an image manipulation program to select the object and extract its pixel-by-pixel information. Our major interest here is in the longitudinal accuracy profile, because high values over time would suggests that the fruit's topography does not influence the measurements taken by the scanner.

We start by making a plot of individual profiles grouped by measurement device, as well as a scatterplot of the hue data (Figure 3). We fit a second-degree polynomial model over time for each fruit considering all observations taken by both devices, and obtain the 95% confidence intervals for the coefficients (Figure 3(c)). Apparently, there is a moderate agreement between the scanner and the colorimeter, which increases as the mean hue decreases. However, this could be due to the smaller number of fruits at the end of the experiment (fruits that presented disease had to be dropped out of the study).

```
R> data(hue)
R> hue_wide <- dcast(hue, Fruit + Time ~ Method, value.var="H_mean")
R> p1 <- ggplot(hue_wide, aes(x = Colorimeter, y = Scanner)) +
+        geom_point(alpha = 0.7, colour = "black", fill = "gray", size = 1) +
+        geom_abline(intercept = 0, slope = 1) +
+        geom_smooth(se = FALSE, method = "lm")
R> p1 +labs(y = "Scanner", x = "Colorimeter")+
+  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))
R> p2 <- ggplot(hue, aes(y = H_mean, x = Time, group = Fruit)) +
+        facet_wrap(~ Method) + geom_line(aes(color = Fruit))
R> p2 +labs(y = "Mean Hue", x = "Time (Day)")+
+  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))
R> m.hue.1 <- lmList(H_mean ~ poly(Time, 2, raw = TRUE)| Fruit, hue)
R> plot(intervals(m.hue.1))
```

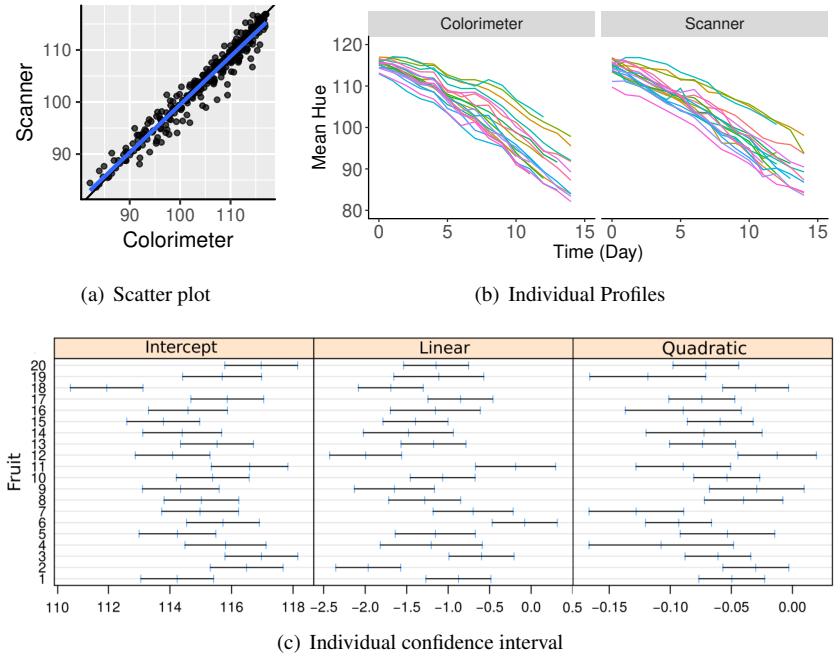
Let y_{ijk} be the peel hue measured on fruit i, using method j at time point k. We start by fitting a second degree polynomial mixed-effects model with random intercepts, linear and quadratic coefficients, written as

$$\begin{aligned} y_{ijk} &= \beta_{0j} + b_{0i} + \left(\beta_{1j} + b_{1i}\right) t_k + \left(\beta_{2j} + b_{2i}\right) t_k^2 + \varepsilon_{ijk}, \\ \boldsymbol{b} &= [b_{0i}, b_{1i}, b_{2i}]^T \sim N_3\left(\boldsymbol{0}, \boldsymbol{G}\right) \quad \text{and} \quad \varepsilon_{ijk} \sim N\left(0, \sigma_{\varepsilon}^2\right), \end{aligned} \tag{5}$$

where $\text{vech}(\boldsymbol{G}) = \left[\sigma_{b_0}^2, \sigma_{b_{01}}, \sigma_{b_{02}}, \sigma_{b_1}^2, \sigma_{b_{12}}, \sigma_{b_2}^2\right]^T$. Under the model (5), the LCC is given by

$$\rho_{jj'}\left(t_k\right) = \frac{\boldsymbol{t}_k \boldsymbol{G} \boldsymbol{t}_k^T}{\boldsymbol{t}_k \boldsymbol{G} \boldsymbol{t}_k^T + \sigma_{\varepsilon}^2 + \frac{1}{2} S_{jj'}^2\left(t_k\right)}.$$

We can fit this model to estimate the LCC, LPC and LA statistics as well as to compute their 95% bootstrap confidence intervals based on 10,000 pseudo-samples using the `lcc()` function directly:



(a) Scatter plot

(b) Individual Profiles

(c) Individual confidence interval

Figure 3. (a) Scatterplot of hue data considering all repeated measurements with a blue line representing the best fit line and the black one the line of equality, (b) Individual profiles of the peel hue of 20 papaya fruits measured by a colorimeter and a scanner, and (c) individual 95% confidence intervals for second degree polynomial coefficients fitted to the data on each fruit considering all methods together

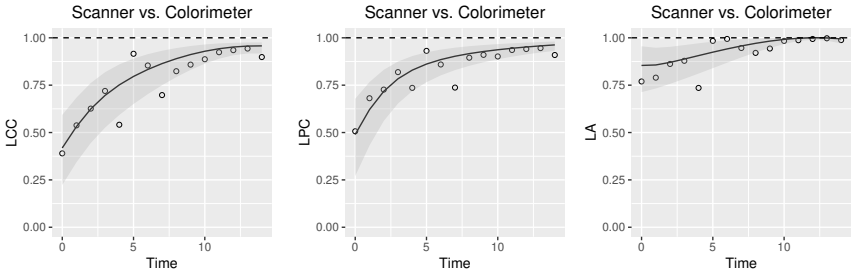
```
R> m.hue.2 <- lcc(data = hue, subject = "Fruit", resp = "H_mean",
+                 method = "Method", time = "Time", qf = 2, qr = 2,
+                 ci = TRUE, nboot = 10000, components = TRUE)
Convergence error in 3074 out of 10000 bootstrap samples.
```

The model used to estimate $\rho_{jj'}(t_k)$ as well as its sampled and fitted values can be extracted by using `summary(m.hue.2, type = "model")` and `summary(m.hue.2, type = "lcc")`, respectively. Moreover, a graphical representation of fitted values and confidence intervals for LCC, LPC and LA can be obtained by executing

```
R> lccPlot(m.hue.2)
R> lccPlot(m.hue.2, type = "lpc")
R> lccPlot(m.hue.2, type = "la")
```

Apparently, the estimated LCC increases over time (Figure 4(a)). However, note that it is necessary to check whether the model assumptions were fulfilled because the estimates for the LCC and its bootstrap confidence intervals may be biased under a misspecified model. We therefore checked (i) the normality assumption for the errors, by producing a normal plot of the within-group standardized residuals (Figure S1(a)), which indicates that this assumption for the within-group errors is almost plausible, and is not far from a normal distribution; ii) the homoscedasticity over time was evaluated via a plot of the standardized residuals versus time (Figure S1(b)), which indicates an apparent residual correlation for observations taken by the colorimeter and greater between-subject variance for observations taken by the scanner (Figure S2); iii) the normality assumption for the random effects (Figure S1(c)), which are verified by producing a normal plot for b_{0i}, b_{1i} and b_{2i}. Additionally, the goodness of fit (`gof`) was 0.992, indicating a high concordance among the model fitted values and observed values. Thus, we update the model `m.hue.2` to include different variances for each level of the factor "method", where the variance function is given by:

$$\mathrm{Var}\left(\varepsilon_{ijk}\right) = \sigma_{\varepsilon}^2 \delta_j^2, \text{ with } j = 1,2.$$



(a) (b) (c)

Figure 4. Estimate and 95% bootstrap confidence interval for the (a) longitudinal concordance correlation (LCC); (b) longitudinal Pearson correlation; and (c) longitudinal accuracy between observations measured by the scanner and the colorimeter with points that represent the (a) sample CCC, (b) sample Pearson correlation coefficient, and (c) sample accuracy, using model (5)

To ensure identifiability we assume that $\delta_1 = 1$. We also created a regular sequence from the time variable that can be used to make predictions

```
R> lcc_time <- with(hue, list(time = Time, from =min(Time),
+                              to=max(Time), n=50))
```

This model can be specified in the `lcc()` as

```
R> m.hue.3 <- update(m.hue.2, var.class = varIdent, weights.form = "method",
+                    time_lcc = lcc_time,
+                    lme.control = lmeControl(opt = "optim"))
Convergence error in 981 out of 10000 bootstrap samples.
```

As models `m.hue.2` and `m.hue.3` are nested, we can use the likelihood ratio to test the hypothesis $H_0 : \delta_2^2 = 1$ versus $H_a : \delta_2^2 \neq 1$:

```
R> anova(m.hue.2, m.hue.3)
Model df    AIC        BIC      logLik    Test    L.Ratio   p-value
 1    13   1938.125  1994.107  -956.0625
 2    14   1934.920  1995.207  -953.4598  1 vs 2  5.205331  0.0225
```

The result shows that we reject H_0 in favour of H_a at a significance level of $\alpha = 0.05$, that is, the inclusion of the function $g(\delta_j) = \delta_j^2$ was significantly important in explaining the extra variability between observations taken at different times.

Moreover, the `gof` between fitted and observed values for `m.hue.3` model has, practically, the same value as presented for the `m.hue.2` model.

```
R> summary(m.hue.3,  type = "lcc")$gof
[1] 0.9915905
```

Although the parameter δ_2^2 was important to explain the variability by method, we can see in Figure S3 that the model assumptions were still not completely fulfilled because there is a possible correlation among residuals for the colorimeter methodology. However, this model is more plausible than the first one. The sample semivariogram estimate is presented in Figure S3(b) and it appears to vary non-randomly around 0.9. Further studies involving the inclusion of correlation structures for the within-group residuals to compute the longitudinal concordance correlation function are still in development.

The agreement profile changes over time, being smaller at the beginning of the experiment and increasing to values close to 1 (Figure 5). If we consider values above 0.80 for the lower bound of the CI as an indication for interchangeability between the use of the two methods, the colorimeter could be used from the 12th day onwards.

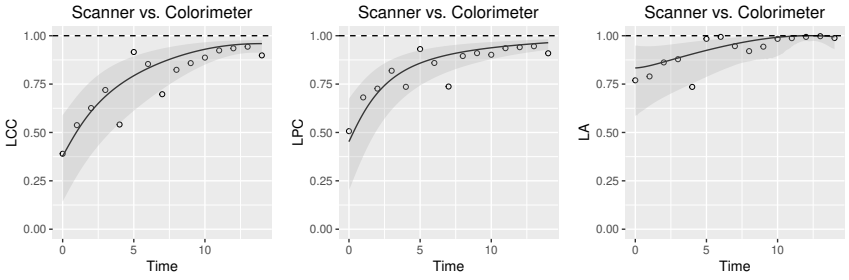


Figure 5. Estimate and 95% bootstrap confidence interval for the (a) longitudinal concordance correlation (LCC); (b) longitudinal Pearson correlation; and (c) longitudinal accuracy between observations measured by the scanner and the colorimeter with points that represent the (a) sample CCC, (b) sample Pearson correlation coefficient, and (c) sample accuracy, using the model that estimates different variances for each method

The blood draw dataset

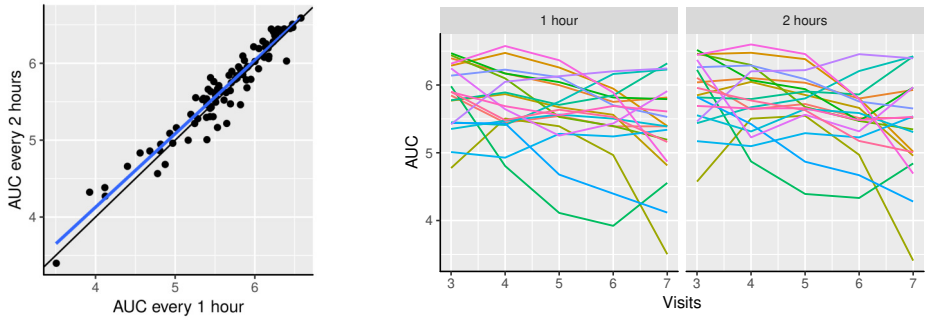
The blood draw dataset was used as an example in the `cccrm` package developed by Carrasco et al. (2013). This dataset comes from a study conducted by the Asthma Clinical Research Network (ACRN) (Martin et al., 2002). In this double-blinded clinical trial, 144 subjects were randomized to one of six inhaled corticosteroid combinations, and the primary aim of the study was to estimate dose-response curves with respect to adrenal suppression. After two weeks, the subjects were admitted for overnight testing once a week, for the next five weeks (visits). Blood samples were collected hourly between 8pm and 8am. Then, the plasma cortisol area under the curve (AUC) was calculated using the trapezoidal rule. A secondary objective here was to assess the agreement of the results from blood sampling performed hourly or every two hours, when calculating the plasma cortisol AUC. As an example, we used all individual profiles whose expected value can be described using a second or lower degree polynomial mixed-effects model:

```
R> data(bdaw, package = "cccrm")
R> bdaw$SUBJ <- as.factor(bdaw$SUBJ)
R> bdaw$MET <- as.factor(bdaw$MET)
R> levels(bdaw$MET) <- c("1 hour",  "2 hours")
R> length(unique(bdaw$SUBJ))
R> library(nlme)
R> fit_list <- lmList(AUC ~ poly(VNUM, 4) | SUBJ, data = bdaw)
R> int <- intervals(fit_list)
R> zero_included <- function(x) {
+   flag <- min(x) < 0 & max(x) > 0
+   return(flag)
+  }
R> selected_subj<- names(
+   which(apply(int[,,4], 1, zero_included) &
+   apply(int[,,5], 1, zero_included)))
R> bdaw_subset <- subset(bdaw, SUBJ %in% selected_subj)
```

The scatterplot of the AUC taken every two hours as a function of the AUC taken each hour and plots of the 19 selected individual profiles are presented in Figure 6.

```
R> bdaw1 <- bdaw_subset %>%
+  filter(MET == "1 hour")
R> bdaw2 <- bdaw_subset %>%
+  filter(MET != "1 hour")
R> p1 <- ggplot(bdaw_subset, aes(y = AUC, x = VNUM,  group = SUBJ,
+                color = SUBJ)) +
+        geom_line(size = 0.6) + facet_wrap(~MET)
```

```
R> p1 +labs(y = "AUC", x = "Visits") +
+  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))
R> p2 <- ggplot(bdaw1, aes(y = bdaw2$AUC, x = bdaw1$AUC)) +
+         geom_point() +
+         geom_smooth(method = "lm", se = FALSE) +
+         geom_abline(intercept = 0, slope = 1)
R> p2 +labs(y = "AUC every 2 hours", x = "AUC every 1 hour")+
+  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))
```



(a) Scatter plot (b) Subject profiles

Figure 6. (a) Scatterplot of the blood draw data considering all repeated measurements (best fit line in blue and equality line in black), and (b) individual profiles of the plasma cortisol AUC calculated from measurements taken every hour and every two hours.

There seems to be a moderate to strong agreement between the plasma cortisol AUC measurements from blood draw samples taken hourly and every two hours (Figure 6(a)). Furthermore, we can also see high variability between subjects and that the AUC decreases over time for some subjects (Figure 6(b)). We begin by fitting a first degree polynomial model with a subject random intercept and slope model.

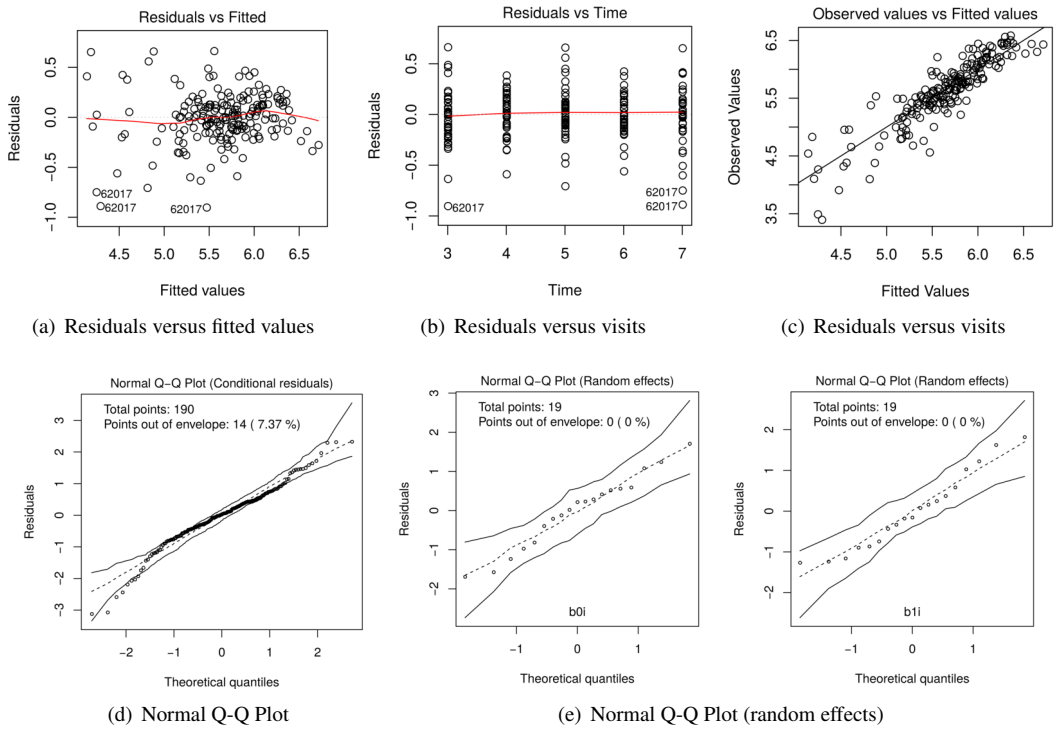
```
R> m.bw.1 <- lcc(data = bdaw_subset, subject = "SUBJ",
+                resp = "AUC", method = "MET", time = "VNUM",
+                qf = 1, qr = 1)
R> summary(m.bw.1, type = "lcc")$gof
[1] 0.8850628
```

This model gives only a moderate fit to the data and this is confirmed by the estimated CCC between fitted and sampled values of 0.885 (Figure 7(c)). Two possible reasons are (i) we need a higher degree polynomial mixed model to correctly describe some subject profiles, and/or (ii) a possible heteroscedasticity across time, potentially caused by three somewhat different subject profiles, that should be included in the model (7(b)). In addition, the normality assumptions for the within group error and random effects were easily checked by producing the normal plot with simulation envelope (7(e)) and seem to be broadly plausible.

```
R> plot(m.bw.1, which = c(1, 2, 4, 5, 6))
```

We mow fit a second degree polynomial model with random subject effects for all coefficients and compute the 95% bootstrap confidence intervals based on 10.000 bootstrap samples for LCC, LPC and LA components.

```
R> m.bw.2 <- update(m.bw.1, qf = 2, qr = 2, components = TRUE,
+                   time_lcc = list(from = 3, to = 7, n = 50),
```



(a) Residuals versus fitted values (b) Residuals versus visits (c) Residuals versus visits

(d) Normal Q-Q Plot (e) Normal Q-Q Plot (random effects)

Figure 7. (a) plot of standardized residuals versus fitted values, (b) standardized residuals versus visits; (c) observed values versus fitted values; (d) Normal Q-Q plot with 95% simulation envelop for the conditional residuals; and (e) normal Q-Q plot with 95% simulation envelop for random effects

```
+                  ci = TRUE, nboot = 10000, show.warnings = TRUE,
+                  lme.control = lmeControl(msMaxIter = 200,
+                  msMaxEval = 600, maxIter = 200))
Convergence error in 0 out of 10000 bootstrap samples.
```

The summary of the mixed effects model used to estimate LCC, LPC and LA is presented below:

```
R> summary(m.bw.2)
Linear mixed-effects model fit by REML
Data: Data
AIC       BIC       logLik
33.93831  75.73247  -3.969153

Random effects:
Formula: ~fmla.rand - 1 | subject
Structure: General positive-definite
                                                StdDev      Corr
fmla.rand(Intercept)                         3.1753653 fm.(I) fd=qr=T
fmla.randpoly(time, degree = qr, raw = TRUE)1 1.3857944 -0.986
fmla.randpoly(time, degree = qr, raw = TRUE)2 0.1404521  0.961 -0.991
Residual                                     0.1269293

Fixed effects: resp ~ fixed - 1
                          Value  Std.Error  DF  t-value  p-value
fixed(Intercept)          6.0147  0.75167  166   8.0018  0.0000
fixedmethod2 hours        0.0471  0.26203  166   0.1796  0.8576
fixedPoly1               -0.0277  0.32744  166  -0.0847  0.9326
```

```
fixedPoly2                -0.0101  0.03315  166  -0.3046  0.7611
fixedmethod2 hours:Poly1   0.0107  0.11083  166   0.0967  0.9231
fixedmethod2 hours:Poly2 -0.0017  0.01101  166  -0.1500  0.8809
Correlation:
                           fxd(I) fxdm2h fxdPl1 fxdPl2 f2h:P1
fixedmethod2 hours        -0.174
fixedPoly1                -0.986  0.167
fixedPoly2                 0.961 -0.160 -0.991
fixedmethod2 hours:Poly1   0.172 -0.989 -0.169  0.165
fixedmethod2 hours:Poly2 -0.168  0.966  0.168 -0.166 -0.993

Standardized Within-Group Residuals:
       Min          Q1         Med          Q3         Max
-2.97645030 -0.48398412  0.03947773  0.59922913  1.87267383

Number of Observations: 190
Number of Groups: 19
```

Now we can test the hypotheses

$$H_0 : \sigma^2_{b_0} > 0, \sigma^2_{b_1} > 0, \sigma_{b_{12}} > 0, \sigma^2_{b_2} = \sigma_{b_{02}} = \sigma_{b_{12}} = 0 \quad \text{vs.} \quad H_a : D \text{ is positive definite}$$

which is equivalent to testing whether the additional variance components of the model `m.bw.2` in relation to `m.bw.1` are equal to zero:

```
R> m.bw.3 <- update(m.bw.1, qf = 2)
R> anova(m.bw.3, m.bw.2)
        Model  df   AIC      BIC   logLik  Test  L.Ratio  p-value
m.bw.3    1    10 207.642 239.792 -93.821
m.bw.2    2    13  33.938  75.732  -3.969 1 vs 2  179.70  <.0001
```

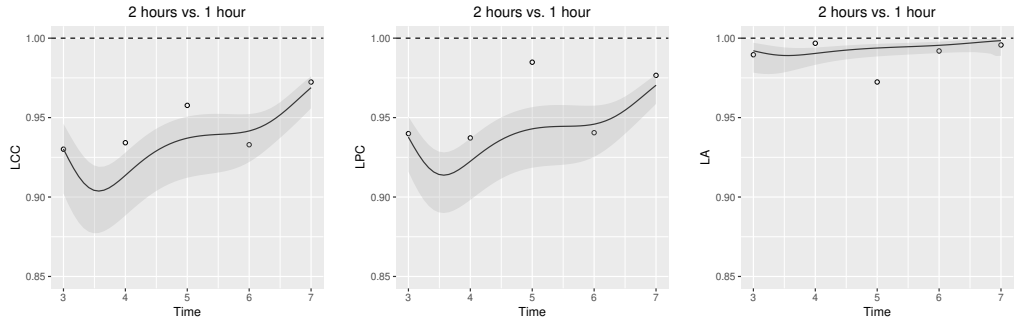
and these results clearly show that those additional variance components are important. Furthermore, the CCC between fitted and observed values also indicates that model `m.bw.2` fits better than model `m.bw.1`, and `m.bw.3`.

```
R> summary(m.bw.1, type="lcc")$gof
[1] 0.8850628
R> summary(m.bw.2, type="lcc")$gof
[1] 0.9830078
R> summary(m.bw.3, type="lcc")$gof
[1] 0.8856218
```

Figure 5 shows the fitted LCC, LPC, and LA for concentration of plasma cortisol AUC between measurements taken every hour and taken every 2 hours and their respective 95% confidence intervals.

```
R> lccPlot(m.bw.2, control = list(scale_y_continuous = c(0.85, 1)))
R> lccPlot(m.bw.2, type = "lpc",
+          control = list(scale_y_continuous = c(0.85, 1)))
R> lccPlot(m.bw.2, type = "la",
+          control = list(scale_y_continuous = c(0.85, 1)))
```

These results show that even though the trend across time is essentially linear at the population level, there is a non-linear trend at the individual level to be more investigated. We can observe that the fitted values and confidence intervals for the LA component were very close to 1 over time, indicating a very high accuracy between methods (Figure 8(c)). Consequently, the LCC values depend almost exclusively on the LPC, which indicates a possible problem related to the precision between methods over time, suggesting the use of blood sampled every hour, rather than every two hours, is desirable for this group of patients. It is worthy to note that, as the diagnostic seems broadly plausible for the second degree mixed effects



(a) (b) (c)

Figure 8. Estimate and 95% bootstrap confidence interval for (a) longitudinal concordance correlation (LCC); (b) longitudinal Pearson correlation; and (c) longitudinal accuracy for the plasma cortisol AUC between measurements taken every hour and taken every 2 hours. In addition, points that represent the sample CCC, sample Pearson correlation coefficient, and sample accuracy, respectively

polynomial model (`m.bw.2`), under this model the LCC, LPC, and LA are fourth degree polynomials functions of the time variable.

Additionally, as the `lcc()` function includes the interaction between time and method as default through the argument `interaction = TRUE`, we can test if the interaction effect is necessary using, for example, the following code:

```
R> m.bw.4 <- lcc(data = bdaw_subset, subject = "SUBJ",
+                resp = "AUC",method = "MET", time = "VNUM",
+                qf = 2, qr = 2, REML = FALSE,  interaction = FALSE)
R> m.bw.5 <- update(m.bw.4,  interaction = TRUE)
R> anova(m.bw.4,  fit.bw5)
       Model  df   AIC    BIC    logLik   Test   L.Ratio  p-value
m.bw.4   1    11 -2.5416 33.176  12.271
m.bw.5   2    13  1.2332 43.445  12.383  1 vs 2  0.22520  0.8935
```

As the p-value was 0.8935, we can conclude that there is no interaction effect and, consequently, the fitted curves for each level of method over time can be considered parallel. Thus, all of these examples show that our methodology is very flexible and can be applied to many different data types, but the user should be careful about avoiding overfitting. We have also created a Shiny app (https://prof-thiagooliveira.shinyapps.io/lccApp/) using simulated data in order to stimulate people to learn more about the LCC and verify how each parameter's value can affect the estimation of the LCC, LPC, and LA.

COMPARISON BETWEEN THE `lcc` AND `cccrm` PACKAGES

In this section we review the `cccrm` `R` package proposed by Carrasco et al. (2013) to estimate the CCC for repeated or non-repeated measures data and discuss how this package differs from our package `lcc`. We can use the `ccclon()` and `ccclonw()` functions to estimate the CCC for repeated measures (CCC_{rm}) and these were the first ones of this type available for `R`. Both functions estimate the CCC based on variance components of a mixed-effects model for longitudinal data, however, the difference between them is that `ccclonw()` uses a non-negative definite matrix of weights between different repeated measurements (Carrasco et al., 2009).

These functions have been introduced to produce a value that summarizes the interchangeability between methods in relation to all of their measurements, rather than by modeling the agreement as a function over time (Carrasco et al., 2009). On the other hand, the `lcc()` function in package `lcc` was developed to capture changes in the extent of the agreement profile between methods. Furthermore, the `lcc` package also provides estimates and confidence intervals for LPC and LA that are important statistics to make inferences on both the precision and accuracy of the measurements, respectively, and how they

affect the LCC at different time points, allowing the evaluation the interchangeability between methods over time.

We now estimate the CCC_{rm}, using package `cccrm`, between the scanner and the colorimeter measurements of the papaya peel hue dataset:

```
R> library(cccrm)
R> data(hue, package = "lcc")
R> ccclon(hue,"H_mean","Fruit","Time","Method")
CCC estimated by variance components:
     CCC          LL CI 95%      UL CI 95%       SE CCC
0.83767698    0.72520268    0.90660774    0.04486742
```

The estimate of the CCC_{rm} shows a moderate/poor agreement between the methods, suggesting that the digital image analysis of the equatorial peel region should be not used to compute the mean hue. However, suppose that the researcher had stipulated the following condition: "we would only take measurements on the equatorial region using a colorimeter if the lower band of confidence interval is greater than or equal to 0.90". Thus, based on the lower band of the CCC_{rm} (0.725), the researcher should not use the colorimeter to compute the mean hue. On the other hand, the lower band of the LCC (Figure 5) indicates that the papaya's equatorial region can be adequately sampled through four equidistant points using a colorimeter from the ninth day. Clearly, this conclusion is only valid under the same experimental conditions.

DISCUSSION

The package `lcc` provides a convenient and versatile tool for estimation and inference about the LCC, LPC, and LA. The estimation of these three statistics provides a complete evaluation of the agreement between methods over time (Oliveira et al., 2018). These statistics are also very appealing for graphical illustration.

The package supports balanced or unbalanced (dropouts) experimental designs or observational studies, multiple methods, inclusion of covariates in the linear predictor to control systematic variation in the response, and the inclusion of different variance-covariance structures for random-effects and residuals. Residual diagnostic and goodness of fit can be evaluated easily via the generic function `plot()`, which provides up to six built-in diagnostic plots. Furthermore, the `anova()`, `AIC()`, and/or `BIC()` functions can be used to aid in model selection.

Statistical inference for the estimators of $\rho_{jj'}(t_k)$, $\rho^{(p)}_{jj'}(t_k)$, and $C_{jj'}(t_k)$ can be obtained using bootstrap confidence intervals based on approximations of their empirical distributions by the normal distribution, or from percentiles of their bootstrap sampling distribution. These methods are, however, computationally intensive.

To the best of our knowledge, there is no package available to estimate the extent of longitudinal agreement between methods. The `lcc` package can be viewed as an extension of the `R` and `SAS` `cccrm` package developed by Carrasco et al. (2013). This package handles the time as a factor in the model, and computes the concordance correlation coefficient, which can be viewed as a measure that summarises the interchangeability between methods in relation to all their measurements.

The importance in estimating the LPC, as a measure of precision, and the LA, as a measure of accuracy, was demonstrated in Section (Figure 5). In particular, both of these statistics can be used jointly to determine if a moderate or small agreement between methods at time $t_k = t$ is related to a precision or an accuracy problem, as suggested by Lin (1989); Barnhart and Williamson (2001); Lin (1992); Ma et al. (2010). In the papaya hue example, the moderate LCC is highly influenced by a moderate LPC, suggesting that if we increase the number of points observed with the colorimeter on the equatorial region up to day 10, the colorimeter will probably be able to reproduce the measurements taken by the scanner. Future studies involve the determination of the sample size over time based on the least acceptable LCC, assuming we can accept up to a certain amount of loss in the LPC and in the LA, as discussed by Lin (1992).

Finally, all examples discussed in Section show that our methodology is flexible, and can be applied to many different data types. One limitation of the `lcc` package is that, for the time being, the `covar` argument only allows for including fixed-effect covariates in the linear predictor. We plan to update our

package in the near future to handle with the inclusion of fixed-effects and random-effects covariates, as well as interaction effects.

CONCLUSION

The `lcc` package implements methods to estimate the LCC, LPC and LA functions as well as their bootstrap confidence intervals. In this package, we included different structures for the variance-covariance matrices of random-effects and residuals, allowing estimation of the extent of longitudinal agreement between methods under different assumptions. Functions `plot()`, for diagnostics, `summary()` and `lccPlot()`, for numerical and graphical summaries, respectively, and `anova()`, `AIC()`, `BIC()`, for model selection, make the package flexible and easy to use. Furthermore, the mixed-effects model based approach to compute the LCC allows us to work with both balanced and unbalanced experimental designs and observational studies.

ACKNOWLEDGEMENTS

The authors are grateful to CNPq (National Council of Technological and Scientific Development), CAPES (Coordination for the Improvement of Higher Education Personnel), University of São Paulo, and National University of Ireland Galway, that supported this research project. We extend our thanks for the Science Foundation Ireland (SFI) under grant number SFI/12/RC/2289, co-funded by the European Regional Development Fund.

REFERENCES

Akaike, H. (1974). A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control*, 19(6):716–723.

Auguie, B. and Antonov, A. (2017). gridExtra: Miscellaneous Functions for 'Grid' Graphics.

Barnhart, H. X. and Williamson, J. M. (2001). Modeling concordance correlation via GEE to evaluate reproducibility. *Biometrics*, 57(3):931–940.

Carrasco, J. L., King, T. S., and Chinchilli, V. M. (2009). The concordance correlation coefficient for repeated measures estimated by variance components. *Journal of biopharmaceutical statistics*, 19(1):90–105.

Carrasco, J. L., Phillips, B. R., Puig-Martinez, J., King, T. S., and Chinchilli, V. M. (2013). Estimation of the concordance correlation coefficient for repeated measures using SAS and R. *Computer Methods and Programs in Biomedicine*, 109:293–304.

Chen, C. C. and Barnhart, H. X. (2013). Assessing agreement with intraclass correlation coefficient and concordance correlation coefficient for data with repeated measures. *Computational Statistics and Data Analysis*, 60(1):132–145.

Chinchilli, V. M., Martel, J. K., Kumanyika, S., and Lloyd, T. (1996). A Weighted Concordance Correlation Coefficient for Repeated Measurement Designs. *Biometrics*, 52(1):341–353.

Giesbrecht, F. G. and Burns, J. C. (1985). Two-Stage Analysis Based on a Mixed Model: Large-Sample Asymptotic Theory and Small- Sample Simulation Results. *International Biometric Society*, 41(2):477–486.

Harville, D. A. (1977). Maximum Likelihood Approaches to Variance Component Estimation and to Related Problems. *Journal of the American Statistical Association*, 72(358):320–338.

King, T. S. and Chinchilli, V. M. (2001). A generalized concordance correlation coefficient for continuous and categorical data. *Statistics in Medicine*, 20(14):2131–2147.

King, T. S., Chinchilli, V. M., Wang, K.-L., and Carrasco, J. L. (2007). A class of repeated measures concordance correlation coefficients. *Journal of biopharmaceutical statistics*, 17:653–672.

Konica Minolta (2003). *Precise colour communication, colour control from perception to instrumentation.* Konica Minolta Sensing.

Lin, L. I. (1989). A Concordance Correlation Coefficient to Evaluate Reproducibility. *Biometrics*, 45(1):255–268.

Lin, L. I.-K. (1992). Assay Validation Using the Concordance Correlation Coefficient. *Biometrics*, 48(2):599.

753 Lloyd, T., Chinchilli, V. M., Eggli, D. F., Rollings, N., and Kulin, H. E. (1998). Body composition
754 development of adolescent white females. *Archives of Pediatric Adolescence Medicine*, 152:998—-
755 1002.
756 Loecher, M., Magrath, P., Aliotta, E., and Ennis, D. B. (2019). Time-optimized 4D phase contrast MRI
757 with real-time convex optimization of gradient waveforms and fast excitation methods. *Magnetic*
758 *Resonance in Medicine*, 82(1):213–224.
759 Ma, Y., Tang, W., Yu, Q., and Tu, X. M. (2010). Modeling concordance correlation coefficient for
760 longitudinal study data. *Psychometrika*, 75(1):99–119.
761 Martin, R. J., Szefler, S. J., Chinchilli, V. M., Kraft, M., Dolovich, M., Boushey, H. A., Cherniack, R. M.,
762 Craig, T. J., Drazen, J. M., Fagan, J. K., Fahy, J. V., Fish, J. E., Ford, J. G., Israel, E., Kunselman, S. J.,
763 Lazarus, S. C., Lemanske, R. F., Peters, S. P., and Sorkness, C. A. (2002). Systemic effect comparisons
764 of six inhaled corticosteroid preparations. *American Journal of Respiratory and Critical Care Medicine*,
765 165(10):1377–1383.
766 Mendoza, F. and Aguilera, J. M. (2004). Application of image analysis for classification of ripening
767 bananas. *Journal of food science*, 69(9):471–477.
768 Moral, R. A., Hinde, J., and Demétrio, C. G. (2017). Half-normal plots and overdispersed models in R:
769 The hnp package. *Journal of Statistical Software*, 81(10).
770 Oliveira, T. d. P., Hinde, J., and Zocchi, S. S. (2018). Longitudinal concordance correlation function based
771 on variance components: an application in fruit color analysis. *Journal of Agricultural, Biological, and*
772 *Enviromental Statistics*, 23(2):233–254.
773 Oliveira, T. d. P., Zocchi, S. S., and Jacomino, A. P. (2017). Measuring color hue in "Sunrise Solo" papaya
774 using flatbed scanner. *Revista Brasileira de Fruticultura*, 39(2):e—-911.
775 Oliveira, T. P., Moral, R. A., Hinde, J., Zocchi, S. S., and Demetrio, C. G. B. (2019). Longitudinal
776 Concordance Correlation.
777 Pandit, V., Chair, Z. B., and Schuller, B. (2019). On Many-to-Many Mapping Between Concordance
778 Correlation Coefficient and Mean Square Error. *arXiv*, pages 1–24.
779 Pinheiro, J., Bates, D., Debroy, S., Sarkar, D., and core team, R. (2017). nlme: linear and nonlinear mixed
780 effects models.
781 Pinheiro, J. C. and Bates, D. M. (2000). *Mixed-Effects Models in S and S-PLUS*. Springer, New York.
782 R core Team (2019). The R environment.
783 Rathnayake, L. N. and Choudhary, P. K. (2017). Semiparametric modeling and analysis of longitudinal
784 method comparison data. *Statistics in Medicine*, 36(13):2003–2015.
785 Schwarz, G. (1978). Estimating the dimension of a model. In *Annals of Statistics*, pages 461–464.
786 Shinar, S., Berger, H., De Souza, L. R., and Ray, J. G. (2019). Difference in Visceral Adipose Tissue in
787 Pregnancy and Postpartum and Related Changes in Maternal Insulin Resistance. *Journal of Ultrasound*
788 *in Medicine*, 38(3):667–673.
789 Warnes, G. R., Bolker, B., Gorjanc, G., Grothendieck, G., Korosec, A., Lumley, T., MacQueen, D.,
790 Magnusson, A., and Rogers, J. (2017). gdata: Various R Programming Tools for Data Manipulation.
791 Wickham, H. (2009). *ggplot2: Elegant Graphics for Data Analysis*. New York.