Partial correlation in linear models

Brice Ozenne

September 18, 2022

1 Summary

This document starts by presenting how to extract from a (univariate) linear regression model partial correlation coefficients. It also precise what type of "partial" (i.e. adjusted on which covariate) we get. When having multiple measurements of pairs of variables, various technics to estimate (partial) correlations are being compared.

2 Example

For illustration we will use the following packages:

```
library(LMMstar)
library(ggplot2)
library(lme4)
library(lmerTest)
library(Matrix)
library(data.table)
```

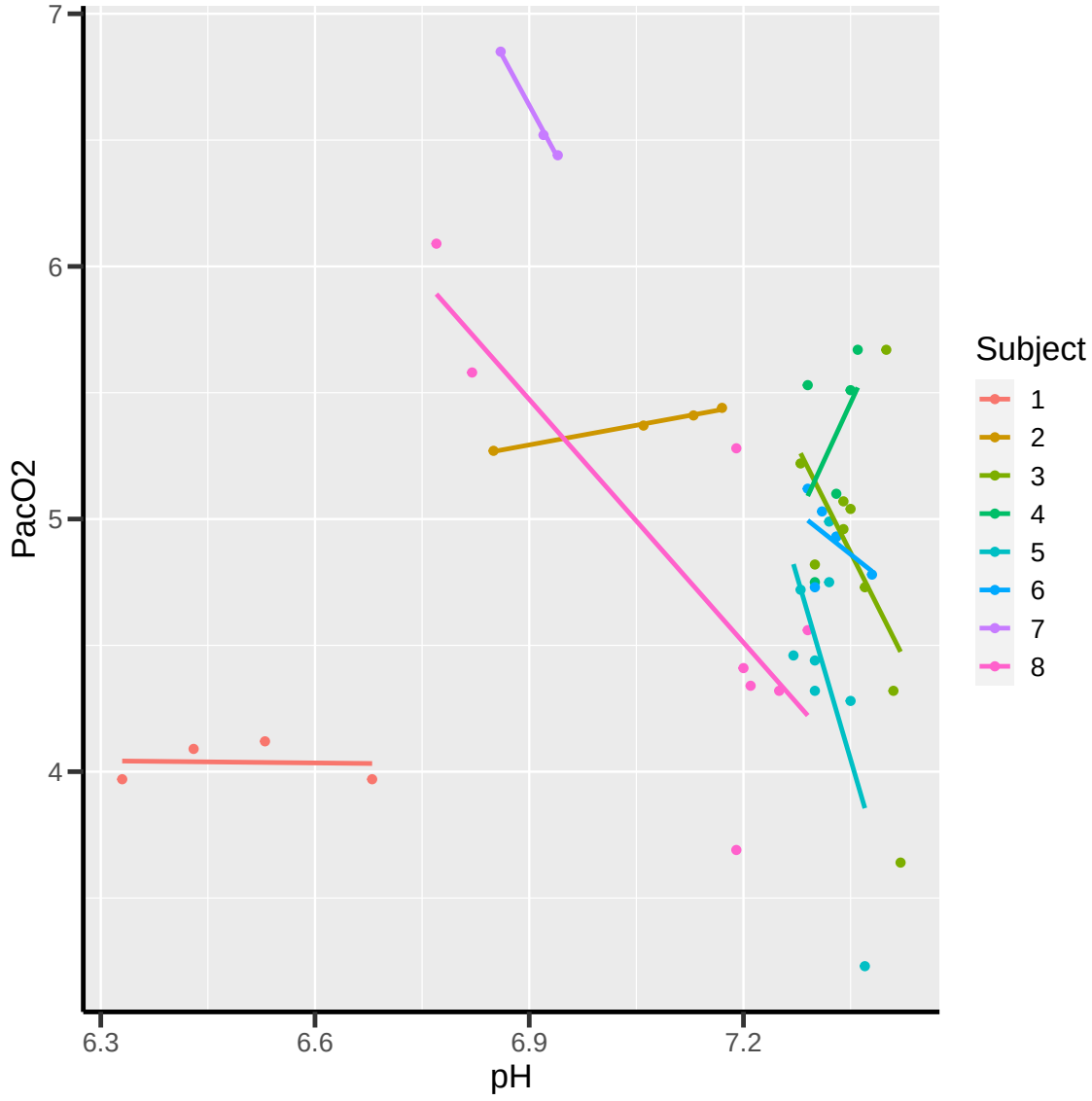
and dataset (Bland and Altman, 1995):

```
data("bland1995", package = "rmcorr")
bland1995$Subject <- as.factor(bland1995$Subject)
bland1995$time <- unlist(tapply(bland1995$Subject,bland1995$Subject,
   function(x){1:length(x)}))
head(bland1995)
```

```
  Subject   pH PacO2 time
1       1 6.68  3.97    1
2       1 6.53  4.12    2
3       1 6.43  4.09    3
4       1 6.33  3.97    4
5       2 6.85  5.27    1
6       2 7.06  5.37    2
```

The aim is to relate intramural pH and PaCO2 using eight subjects:

```
gg <- ggplot(bland1995, aes(x = pH, y = PacO2,
       group = Subject, color = Subject))
gg <- gg + geom_point() + geom_smooth(method = "lm", se = FALSE)
gg
```



3 Partial partial in multiple linear regression

Consider the linear model:

```
e.lmm <- lmm(pH ~ Subject + PacO2, data = bland1995)
eTable.lmm <- model.tables(e.lmm)
eTable.lmm
```

	estimate	se	df	lower	upper	p.value
(Intercept)	6.9298543	0.12946898	38.0076	6.6677598	7.19194884	0.000000e+00
Subject2	0.7046113	0.07735488	38.0076	0.5480155	0.86120702	4.269674e-11
Subject3	0.9500127	0.06109545	38.0076	0.8263322	1.07369313	0.000000e+00
Subject4	0.9715577	0.07350906	38.0076	0.8227474	1.12036807	8.881784e-16
Subject5	0.8603817	0.05839543	38.0076	0.7421671	0.97859630	0.000000e+00
Subject6	0.9264284	0.06599450	38.0076	0.7928304	1.06002642	0.000000e+00
Subject7	0.6921056	0.10490935	38.0076	0.4797291	0.90448203	8.662210e-08
Subject8	0.7033361	0.06157141	38.0076	0.5786921	0.82798005	7.438494e-14
PacO2	-0.1083230	0.02989281	38.0076	-0.1688375	-0.04780862	8.469583e-04

We claim the partial correlation (adjusting `pH` and `PacO2` for `Subject`) can be deduced from the Wald statistic and degrees of freedom:

$$\rho = \frac{\frac{\beta}{\sigma_\beta}}{\sqrt{\frac{\beta^2}{\sigma_\beta^2} + df}} = \frac{\beta}{\sqrt{\beta^2 + df * \sigma_\beta^2}} \tag{1}$$

```
Wald <- eTable.lmm["PacO2","estimate"]/eTable.lmm["PacO2","se"]
Wald/sqrt(Wald^2+eTable.lmm["PacO2","df"])
```

```
[1] -0.5067321
```

The proof can be split in three steps:

1. the F-statistic testing the effect of each factor equals the Wald-statistic squared (divided by 1, the number of parameters)

```
Wald^2
```

```
[1] 13.13132
```

```
anova(e.lmm)
```

```
             Multivariate Wald test

              F-statistic       df  p.value
mean: Subject      48.247 (7,38.0)  < 2e-16 ***
    : PacO2        13.131 (1,38.0) 0.000847 ***
```

2. this F-statistic equals $\frac{MSSR}{MSSE}$ where $MSSR = SSR/1$ and $MSSE = SSE/(n-p)$ with SSE and SSR being the explained and residual sum of squares. We can check that this extends to multiple regression using the usual anova table:

```
anova(lm(pH ~ Subject + PacO2, data = bland1995))
```

```
Analysis of Variance Table

Response: pH
          Df  Sum Sq Mean Sq F value    Pr(>F)
Subject    7 2.86484 0.40926  46.600 < 2.2e-16 ***
PacO2      1 0.11532 0.11532  13.131 0.0008471 ***
Residuals 38 0.33373 0.00878
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

which is to be compared to[1]

```
sigma2 <- as.double(sigma(e.lmm))
beta <- eTable.lmm["PacO2","estimate"]
sigma_beta <- eTable.lmm["PacO2","se"]
c(MSSE = sigma2, MSSR = sigma2 * beta^2 /sigma_beta^2)
```

```
       MSSE        MSSR
0.008782435 0.115324959
```

This result can be easily proved when considering a model with a single regressor:

$$Y = X\beta + \varepsilon,\ \varepsilon \sim \mathcal{N}(0, \sigma^2)$$

where we would have centered the outcome Y. Here we denote by X the design matrix, n the number of observations and $p = 1$ the number of coefficients, $H =$

[1] ⚠ Since **R** output type 1 anova only the last and second to last line are relevant. The first line (`Subject`) is for a model without `PacO2` so it should be expected that the F-value does not match with the one of `Subject` in a model with `PacO2`.

$X(XX^\intercal)^{-1}X^\intercal$ the hat matrix and $\widehat{\beta} = (XX^\intercal)^{-1}X^\intercal Y$ the OLS estimator of the regression coefficients.

$$\begin{aligned}
\mathbb{V}ar(Y) = YY^\intercal =& YHY^\intercal + Y(1-H)Y^\intercal \\
SST =& SSR + SSE \\
=& \hat{\beta}(XX^\intercal)\hat{\beta}^\intercal + Y(1-H)Y^\intercal \\
=& \sigma^2(\hat{\beta}\Sigma_{\hat{\beta}}^{-1}\hat{\beta}^\intercal + n - p) \\
\frac{MSSR}{MSSE} = \frac{\hat{\beta}^2}{\Sigma_{\hat{\beta}}} =& Wald^2
\end{aligned}$$

3. the R^2 is defined as the proportion of variance explained, so using the previous results we get:

$$\begin{aligned}
R^2 =& \frac{SSR}{SSR + SSE} \\
=& \frac{1}{1 + SSE/SSR} \\
=& \frac{1}{1 + (n-p)/(\beta^2/\sigma_\beta^2)} \\
= \frac{Wald^2}{Wald^2 + n - p}&
\end{aligned}$$

This formula matches exactly the partial correlation coefficient when **both** outcome are adjusted for `Subject`:

```
e.partialCor <- partialCor(list(pH ~ Subject, PaCO2 ~ Subject),
      data = bland1995)
print(e.partialCor, digit = 5)
```

```
              estimate      se     df    lower   upper   p.value
rho(pH,PaCO2) -0.50677 0.12514 25.674 -0.71027 -0.2251 0.0017753
```

Similar values can be obtained using dedicated packages, e.g.:

```
library(rmcorr)
rmcorr(Subject, PaCO2, pH, bland1995)$r
```

```
[1] -0.5067697
```

4 Partial correlation with repeated measurements

4.1 Marginal and conditional correlation

There are several references on the subject (Bland and Altman, 1995; Lipsitz et al., 2001; Bakdash and Marusich, 2017; Shan et al., 2020). We will focus on the mixed model approach. The idea is to jointly model the variance and covariance of all measurements under appropriate constrains. For instance denoting one measurement X and the other measurement Y, both indexed by time t, our target parameter may be $\rho = \mathbb{C}or(X(t), Y(t))$ (marginal) assumed independent of t while X and Y may or may not be stationnary. Another target parameter could be the correlation between a de-noised version of X and Y, where we have for instance removed individual-specific variations (conditional).

To be more specific let's consider the following statistical model:

$$
\begin{aligned}
X_i(t) &= \mu_{X,i}(t) + u_i + \varepsilon_{X,i}(t) \\
Y_i(t) &= \mu_{Y,i}(t) + v_i + \varepsilon_{Y,i}(t) \\
\text{where } \begin{bmatrix} u \\ v \\ \varepsilon_X(t) \\ \varepsilon_Y(t) \end{bmatrix} &= \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \tau_u & \tau_{uv} & 0 & 0 \\ \tau_{uv} & \tau_v & 0 & 0 \\ 0 & 0 & \sigma_X & \sigma_{XY} \\ 0 & 0 & \sigma_{XY} & \sigma_X \end{bmatrix} \right)
\end{aligned}
$$

It implies the following residual covariance matrix:

$$
\begin{aligned}
\Omega = \mathbb{V}ar \begin{bmatrix} X(1) \\ X(2) \\ X(3) \\ Y(1) \\ Y(2) \\ Y(3) \end{bmatrix} &= \begin{bmatrix} \tau_u + \sigma_X & \tau_u & \tau_u & \tau_{uv} + \sigma_{XY} & \tau_{uv} & \tau_{uv} \\ \tau_u & \tau_u + \sigma_X & \tau_u & \tau_{uv} & \tau_{uv} + \sigma_{XY} & \tau_{uv} \\ \tau_u & \tau_u & \tau_u + \sigma_X & \tau_{uv} & \tau_{uv} & \tau_{uv} + \sigma_{XY} \\ \tau_{uv} + \sigma_{XY} & \tau_{uv} & \tau_{uv} & \tau_v + \sigma_Y & \tau_v & \tau_v \\ \tau_{uv} & \tau_{uv} + \sigma_{XY} & \tau_{uv} & \tau_v & \tau_v + \sigma_Y & \tau_v \\ \tau_{uv} & \tau_{uv} & \tau_{uv} + \sigma_{XY} & \tau_v & \tau_v & \tau_v + \sigma_Y \end{bmatrix} \\
&= \begin{bmatrix} \sigma_1 & \sigma_2 & \sigma_2 & \sigma_3 & \sigma_4 & \sigma_4 \\ \sigma_2 & \sigma_1 & \sigma_2 & \sigma_4 & \sigma_3 & \sigma_4 \\ \sigma_2 & \sigma_2 & \sigma_1 & \sigma_4 & \sigma_4 & \sigma_3 \\ \sigma_3 & \sigma_4 & \sigma_4 & \sigma_5 & \sigma_6 & \sigma_6 \\ \sigma_4 & \sigma_3 & \sigma_4 & \sigma_6 & \sigma_5 & \sigma_6 \\ \sigma_4 & \sigma_4 & \sigma_3 & \sigma_6 & \sigma_6 & \sigma_5 \end{bmatrix}
\end{aligned}
$$

and the following residual correlation matrix:

$$R = \mathbb{C}or \begin{bmatrix} X(1) \\ X(2) \\ X(3) \\ Y(1) \\ Y(2) \\ Y(3) \end{bmatrix} = \begin{bmatrix} 1 & \rho_1 & \rho_1 & \rho_2 & \rho_3 & \rho_3 \\ \rho_1 & 1 & \rho_1 & \rho_3 & \rho_2 & \rho_3 \\ \rho_1 & \rho_1 & 1 & \rho_3 & \rho_3 & \rho_2 \\ \rho_2 & \rho_3 & \rho_3 & 1 & \rho_4 & \rho_4 \\ \rho_3 & \rho_2 & \rho_3 & \rho_4 & 1 & \rho_4 \\ \rho_3 & \rho_3 & \rho_2 & \rho_4 & \rho_4 & 1 \end{bmatrix}$$

The marginal correlation is:

$$\begin{aligned} \rho_M &= \frac{\mathbb{C}ov\left[u_i + \varepsilon_{X,i}(t), v_i + \varepsilon_{Y,i}(t)\right]}{\sqrt{\mathbb{V}ar\left[u_i + \varepsilon_{X,i}(t)\right]\mathbb{V}ar\left[v_i + \varepsilon_{Y,i}(t)\right]}} \\ &= \frac{\tau_{uv} + \sigma_{XY}}{\sqrt{(\tau_u + \sigma_X)(\tau_v + \sigma_Y)}} = \frac{\sigma_3}{\sqrt{\sigma_1\sigma_5}} = \rho_2 \end{aligned}$$

while the conditional correlation is:

$$\begin{aligned} \rho_C &= \frac{\mathbb{C}ov\left[\varepsilon_{X,i}(t), \varepsilon_{Y,i}(t)\right]}{\sqrt{\mathbb{V}ar\left[\varepsilon_{X,i}(t)\right]\mathbb{V}ar\left[\varepsilon_{Y,i}(t)\right]}} \\ &= \frac{\sigma_{XY}}{\sqrt{\sigma_X\sigma_Y}} = \frac{\sigma_3 - \sigma_4}{\sqrt{(\sigma_1 - \sigma_2)(\sigma_5 - \sigma_6)}} = \frac{\rho_2 - \rho_3}{\sqrt{(1-\rho_1)(1-\rho_2)}} \end{aligned}$$

4.2 Approximated conditional correlation

We now show that formula 1 generalizes to mixed models. Consider the following mixed model relating $\mathbf{Y} = (Y_1, \ldots, Y_T)$ and $\mathbf{X} = (X_1, \ldots, X_T)$:

$$\mathbf{Y} = \mathbf{X}\beta + \boldsymbol{\varepsilon}$$

where $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \Omega)$. Introducing the cholesky decomposition $\Omega = \omega\omega^\intercal$, we can equivalently study:

$$\omega^{-1}\mathbf{Y} = \omega^{-1}\mathbf{X} + \boldsymbol{\zeta}$$

where $\boldsymbol{\zeta}$ follow a standard normal distribution. We are back the univariate case up to a factor ω^{-1}.

1. F-statistics are still equal the Wald statistic squared (divided by the number of parameters).

2. F-statistics still equal $\frac{MSSR}{MSSE}$. Indeed:

$$
\begin{aligned}
SSE &= \left(\omega^{-1}\mathbf{Y}\right)^{\intercal}\left(I-\omega^{-1}\mathbf{X}\left(\left(\omega^{-1}\mathbf{X}\right)^{\intercal}\left(\omega^{-1}\mathbf{X}\right)\right)^{-1}\left(\omega^{-1}\mathbf{X}\right)^{\intercal}\right)\left(\omega^{-1}\mathbf{Y}\right)\\
&= \mathbf{Y}^{\intercal}\Omega^{-1}\mathbf{Y}-\mathbf{Y}^{\intercal}\Omega^{-1}\mathbf{X}\left(\mathbf{X}^{\intercal}\Omega^{-1}\mathbf{X}\right)^{-1}\mathbf{X}^{\intercal}\Omega^{-1}\mathbf{Y}\\
&= \mathbf{Y}^{\intercal}(I-H^{\intercal})\Omega^{-1}(I-H^{\intercal})\mathbf{Y}
\end{aligned}
$$

where $H=\mathbf{X}\left(\mathbf{X}^{\intercal}\Omega^{-1}\mathbf{X}\right)^{-1}\mathbf{X}^{\intercal}\Omega^{-1}$. Indeed:

$$(I-H^{\intercal})\Omega^{-1}(I-H^{\intercal})=\Omega^{-1}-H^{\intercal}\Omega^{-1}-\Omega^{-1}H+H^{\intercal}\Omega^{-1}H=\Omega^{-1}-H^{\intercal}\Omega^{-1}$$

and $MSSE=\frac{SSE}{n-p}=\sigma^2$ with p being the rank of X. Using that $HH=H$:

$$
\begin{aligned}
SSR &= \left(\omega^{-1}\mathbf{Y}\right)^{\intercal}\left(\omega^{-1}\mathbf{X}\left(\left(\omega^{-1}\mathbf{X}\right)^{\intercal}\left(\omega^{-1}\mathbf{X}\right)\right)^{-1}\left(\omega^{-1}\mathbf{X}\right)^{\intercal}\right)\left(\omega^{-1}\mathbf{Y}\right)\\
&= \mathbf{Y}^{\intercal}\Omega^{-1}\mathbf{X}\left(\mathbf{X}^{\intercal}\Omega^{-1}\mathbf{X}\right)^{-1}\mathbf{X}^{\intercal}\Omega^{-1}\mathbf{Y}\\
&= \mathbf{Y}^{\intercal}H^{\intercal}\Omega^{-1}\mathbf{Y}=\mathbf{Y}^{\intercal}H^{\intercal}H^{\intercal}\Omega^{-1}\mathbf{Y}\\
&= \mathbf{Y}^{\intercal}H^{\intercal}\Omega^{-1}H\mathbf{Y}\\
&= \widehat{\beta}^{\intercal}X^{\intercal}\Omega^{-1}X\widehat{\beta}=\widehat{\beta}^{\intercal}\Sigma_{\widehat{\beta}}^{-1}\widehat{\beta}
\end{aligned}
$$

where $\widehat{\beta}=\left(\mathbf{X}^{\intercal}\Omega^{-1}\mathbf{X}\right)^{-1}\mathbf{X}^{\intercal}\Omega^{-1}\mathbf{Y}$ is the GLS estimator of β. So for a single covariate:

$$F=\frac{MSSR}{MSSE}=\frac{\widehat{\beta}\Sigma^{-1}\widehat{\beta}}{\sigma^2}$$

3. Defining R^2 as the proportion of variance explained, we get back

$$R^2=\frac{\beta^2}{\beta^2+df\,\sigma_\beta^2}$$

where $df=n-p$. A corresponding correlation coefficient can computed as:

$$\rho=\frac{\beta}{\sqrt{\beta^2+df\,\sigma_\beta^2}}$$

4.3 Back to the example

In the example, we see a very small marginal correlation and a large conditional one:

```
e.pcor <- partialCor(c(pH,PacO2)~1, repetition = ~time|Subject, data =
    bland1995, heterogeneous = 0.5)
e.pcor
```

```
                  estimate    se   df  lower    upper p.value
rho(1.pH,1.PacO2) -1.63e-05 0.313 1.23 -0.989 0.988993  1.0000
r(1.pH,1.PacO2)   -5.09e-01 0.125 2.59 -0.808 0.000496  0.0501
```

This matches the estimate (but not the uncertainty) of another software:

```
c(r = rmcorr(Subject, pH, PacO2, bland1995)$r,
  p = rmcorr(Subject, pH, PacO2, bland1995)$p)
```

```
           r            p
-0.5067697422  0.0008471081
```

We can also extract the underlying correlation coefficients:

```
round(coef(attr(e.pcor,"lmm"), effects = "correlation"),5)
```

```
rho(1.pH,1.PacO2)    rho(1.pH,2.PacO2) rho(1.PacO2,2.PacO2)       rho(1.pH,2.pH)
         -0.00002              0.10168              0.66317              0.88129
```

that reveal a very strong within pH correlation (almost 0.9) and a rather strong within PacO2 correlation (about 0.65). The instantaneous correlation is nearly 0 but the lag correlation is about 0.1 leading to the observed conditional correlation.

An alternative approach is to fit a mixed model on only one outcome, regressing out the other:

```
e.CS <- lmm(pH ~ PacO2, repetition = ~time|Subject, data = bland1995,
     structure = "CS")
```

Then estimate the partial correlation formula:

```
e.CSaov <- anova(e.CS, effects = "PacO2=0")
confint(e.CSaov, columns = c("estimate","se","df","partial.r"))
```

```
      estimate     se   df partial.r
PacO2   -0.103 0.0295 39.6    -0.486
```

Here approximate degrees of freedom are used, i.e. 39.6 instead of:

```
NROW(bland1995)-2
```

```
[1] 45
```

which would lead to a correlation of:

```
e.CSaov$univariate$statistic/sqrt(e.CSaov$univariate$statistic^2+45)
```

```
[1] -0.4627676
```

Finally we could also compute the Person's correlation (ignoring repeated measurements):

```
cor(dtW$pH,dtW$PacO2)
```

```
[1] -0.06521774
```

and use a bootstrap at the individual level for assessing the uncertainty:

```
library(boot)
library(data.table)
dtW <- as.data.table(bland1995)
dtL <- dcast(dtW, value.var = c("pH","PacO2"), formula = Subject ~ time)
calcCor <- function(data, statistic){
  data2 <- data[statistic]
  data3 <- melt(data2, id.vars = c("Subject"),
  measure=patterns("pH","PacO2"),
  variable.name = "time", value.name = c("pH","PacO2"))
  cor(data3$pH, data3$PacO2)
}
e.boot <- boot(dtW, calcCor, R = 1000)
e.boot
```

```
ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot(data = dtW, statistic = calcCor, R = 1000)


Bootstrap Statistics :
       original      bias    std. error
t1* -0.06521774 -0.01635557   0.1989813
```

In summary we have obtained the following estimates:

- for the marginal correlation

```
out.naive <- c(estimate = e.boot$t0, se = sd(e.boot$t), df = NA,
        lower = boot.ci(e.boot, type = "perc")$percent[4],
        upper = boot.ci(e.boot, type = "perc")$percent[5],
        p.value = NA)
e.pcor2 <- partialCor(c(pH,PacO2)~1, repetition = ~time|Subject, df =
    FALSE,
        data = bland1995, heterogeneous = 0.5)
```

- for the conditional correlation

```
e.rmcorr <- rmcorr(Subject, PacO2, pH, bland1995)
out.rmcorr <- c(estimate = e.rmcorr$r, se = NA, df = e.rmcorr$df,
  lower = e.rmcorr[[4]][1], upper = e.rmcorr[[4]][2], p.value = e.rmcorr
    [[3]])
out.magic <- estimate(e.CS, f = function(p){
  e.vcov <- vcov(e.CS, df = TRUE, p = p)
  p["PacO2"]/sqrt(p["PacO2"]^2+e.vcov["PacO2","PacO2"]*attr(e.vcov,"df")["
    PacO2"])
})
```

So overall:

```
out <- rbind(
  data.frame(type = "marginal", rbind(naive = out.naive, lmmM = e.pcor2
    [1,])),
  data.frame(type = "conditional", rbind(rmcorr = out.rmcorr, lmmC = e.
    pcor2[2,], magic = out.magic))
)
out <- cbind(name = rownames(out), out)
rownames(out) <- NULL
out
```

name	type	estimate	se	df	lower	upper	p.value
1 naive	marginal	-6.521774e-02	0.19898132	NA	-0.5183738	0.2809898	NA
2 lmmM	marginal	-1.627833e-05	0.31296494	Inf	-0.5465274	0.5465046	9.999585e-01
3 rmcorr	conditional	-5.067697e-01	NA	38.00000	-0.7112297	-0.2232550	8.471081e-04
4 lmmC	conditional	-5.085547e-01	0.12542915	Inf	-0.7043437	-0.2408608	4.862469e-04
5 magic	conditional	-4.864796e-01	0.08698358	30.83874	-0.6639214	-0.3090378	3.992869e-06

```
gg.forest <- ggplot(out, aes(x = name, y = estimate, color = type))
gg.forest <- gg.forest + geom_hline(yintercept=0, linetype = 2)
gg.forest <- gg.forest + geom_point(size = 2) + geom_errorbar(aes(ymin =
    lower, ymax = upper))
gg.forest <- gg.forest + coord_flip()
gg.forest
```

4.4 Simulation study (compound symmetry model)

We'll compare ρ and r in the case of 3 timepoints, $r = 0.8$, and 250 individuals:

```
n.time <- 3
n.id <- 250
Sigma <- matrix(c(1,0.8,0.8,1),2,2)
Sigma
```

```
     [,1] [,2]
[1,]  1.0  0.8
[2,]  0.8  1.0
```

```
set.seed(11)
df.W <- data.frame(id = unlist(lapply(1:n.id, rep, n.time)),
     time = rep(1:n.time,n.id),
     rmvnorm(n.time*n.id, mean = c(3,3), sigma = Sigma)
     )
head(df.W)
```

```
  id time       X1       X2
1  1    1 2.483259 2.759470
2  1    2 1.034157 1.102983
3  1    3 3.636308 2.691506
4  2    1 4.463341 4.150878
5  2    2 2.510048 2.081439
6  2    3 2.103239 2.317938
```

We use random effects to obtain a constant correlation within X and within Y:

```
sd.id <- 1.5
df.W$X1 <- df.W$X1 + rnorm(n.id, sd = sd.id/4)[df.W$id]
df.W$X2 <- df.W$X2 + rnorm(n.id, sd = sd.id)[df.W$id]
df.W$id <- as.factor(df.W$id)
df.L <- reshape2::melt(df.W, id.vars = c("id","time"))
df.L$time2 <- as.factor(as.numeric(as.factor(paste(df.L$variable,df.L$time
    ,sep="."))))
```

This will lead to the following correlation structure:

```
Sigma.GS <- as.matrix(bdiag(Sigma,Sigma,Sigma))[c(1,3,5,2,4,6),c
    (1,3,5,2,4,6)]
Sigma.GS[1:3,1:3] <- Sigma.GS[1:3,1:3] + (sd.id/4)^2
Sigma.GS[4:6,4:6] <- Sigma.GS[4:6,4:6] + sd.id^2
cov2cor(Sigma.GS)
```

```
          [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
[1,] 1.0000000 0.1232877 0.1232877 0.4155056 0.0000000 0.0000000
[2,] 0.1232877 1.0000000 0.1232877 0.0000000 0.4155056 0.0000000
[3,] 0.1232877 0.1232877 1.0000000 0.0000000 0.0000000 0.4155056
[4,] 0.4155056 0.0000000 0.0000000 1.0000000 0.6923077 0.6923077
[5,] 0.0000000 0.4155056 0.0000000 0.6923077 1.0000000 0.6923077
[6,] 0.0000000 0.0000000 0.4155056 0.6923077 0.6923077 1.0000000
```

We can now estimate two types of correlation: marginal and conditional

```
e.LMMstar <- partialCor(c(X1,X2) ~ 1, repetition = ~ time|id, data = df.W
    , heterogeneous = 0.5)
e.LMMstar
```

```
               Partial correlation

               estimate     se   df lower upper  p.value
rho(1.X1,1.X2)    0.427 0.0346 34.7 0.356 0.493 6.76e-13
r(1.X1,1.X2)      0.798 0.0251 58.9 0.764 0.829 0.00e+00
        ------------------------------------------------------
        rho: marginal correlation
        r  : correlation conditional on the individual
        estimates, standard errors, confidence intervals have been back-transformed (tanh).
```

The conditional coefficient is identical to what other packages output:

```
rmcorr:::rmcorr(id, X1, X2, df.W)$r
```

```
[1] 0.7983617
```

Here the modeled correlation matrix is:

```
Omega <- sigma(attr(e.LMMstar,"lmm"))
Rho <- cov2cor(Omega)
Rho
```

```
           1.X1        2.X1        3.X1        1.X2        2.X2        3.X2
1.X1  1.00000000  0.06545230  0.06545230  0.42652595 -0.00432106 -0.00432106
2.X1  0.06545230  1.00000000  0.06545230 -0.00432106  0.42652595 -0.00432106
3.X1  0.06545230  0.06545230  1.00000000 -0.00432106 -0.00432106  0.42652595
1.X2  0.42652595 -0.00432106 -0.00432106  1.00000000  0.68836567  0.68836567
2.X2 -0.00432106  0.42652595 -0.00432106  0.68836567  1.00000000  0.68836567
3.X2 -0.00432106 -0.00432106  0.42652595  0.68836567  0.68836567  1.00000000
```

From which the conditional correlation can be deduced:

```
(Rho[1,4]-Rho[1,5])/sqrt((1-Rho[1,2])*(1-Rho[4,5]))
```

```
[1] 0.7983617
```

or equivalently:

```
(Omega[1,4]-Omega[1,5])/sqrt((Omega[1,1]-Omega[1,2])*(Omega[4,4]-Omega
   [4,5]))
```

```
[1] 0.7983617
```

Replicating this a thousand times:

```
n.id <- 100
n.sim <- 1000
n.cpus <- 25 ## run on the server
warper <- function(n){
  df.W <- data.frame(id = unlist(lapply(1:n, rep, n.time)),
       time = rep(1:n.time,n),
       rmvnorm(n.time*n, mean = c(3,3), sigma = Sigma)
       )
  df.W$X1 <- df.W$X1 + rnorm(n, sd = sd.id/4)[df.W$id]
  df.W$X2 <- df.W$X2 + rnorm(n, sd = sd.id)[df.W$id]
  df.W$id <- as.factor(df.W$id)

  res1 <- setNames(c(rmcorr(id, X1, X2, df.W)$r, rmcorr(id, X1, X2, df.W)$
   CI), c("estimate","lower","upper"))
  res2 <- partialCor(c(X1,X2) ~ 1, repetition = ~ time|id, data = df.W,
   heterogeneous = 0.5)
  return(rbind(cbind(as.data.frame(as.list(res1)), se = NA, method = "
   rmcorr"),
```

```
        cbind(res2[2,c("estimate","lower","upper","se")],method="lmm")))
}

ls.res <- pbapply::pblapply(1:n.sim,function(iSim){
  cbind(sim = iSim, warper(n.id))
}, cl = n.cpus)
dt.res <- as.data.table(do.call(rbind, ls.res))
```

lead to the same estimate for the two implementations:

```
range(dt.res[method=="rmcorr",estimate]-dt.res[method=="lmm",estimate], na
    .rm=TRUE)
```

```
[1] -8.572216e-10  2.108167e-09
```

and lead to a reasonnable coverage:

```
dt.res[,.(missing = mean(is.na(estimate)), coverage = mean((0.8>=lower)*
    (0.8<=upper), na.rm=TRUE)), by = "method"]
```

```
   method missing coverage
1: rmcorr   0.000 0.941000
2:    lmm   0.026 0.949692
```

4.5 Simulation study (crossed random effect model)

We will modify the previous simulation setting by introducing more structure on the correlation. More precisely, observations will be correlated within individual (biological variation) and within timepoint (batch effect). This violates the compound symmetry structure and therefore we expect `rmcorr` to give biased estimates. We will use `lmer` instead of `lmm` as a reference since `lmer` is very convenient to use and fast when dealing with crossed random effects. Note that, however, it is not straightforward to have a measure of uncertainty.

```
n.time <- 4
n.id <- 100
warper <- function(n){
  df.W <- data.frame(id = unlist(lapply(1:n.id, rep, n.time)),
       time = rep(1:n.time,n.id),
       rmvnorm(n.time*n.id, mean = c(3,3), sigma = Sigma)
       )
  df.W$X1 <- df.W$X1 + rnorm(n.id, sd = sd.id/4)[df.W$id]
  df.W$X2 <- df.W$X2 + rnorm(n.id, sd = sd.id)[df.W$id]
  df.W$X1 <- df.W$X1 + rnorm(n.time, sd = sd.id/3)[df.W$time]
  df.W$X2 <- df.W$X2 + rnorm(n.time, sd = sd.id/2)[df.W$time]
```

```
  df.W$id <- as.factor(df.W$id)
  df.W$time <- as.factor(df.W$time)

  e.lm <- lm(X1~X2+id+time, data = df.W)
  e.Slm <- summary(e.lm)$coef

  e.lmer <- lmer(X2 ~ X1 + (1|time) + (1|id), data = df.W)
  e.Slmer <- summary(e.lmer)$coefficient

  res0 <- c(estimate = e.Slm["X2","t value"]/sqrt(e.Slm["X2","t value"]^2+
    df.residual(e.lm)), lower = NA, upper = NA)
  res1 <- setNames(c(rmcorr(id, X1, X2, df.W)$r, rmcorr(id, X1, X2, df.W)$
    CI), c("estimate","lower","upper"))
  res2 <- c(estimate = e.Slmer["X1","t value"]/sqrt(e.Slmer["X1","t value"
    ]^2+e.Slmer["X1","df"]), lower = NA, upper = NA)

  return(rbind(cbind(as.data.frame(as.list(res0)), method = "lm"),
        cbind(as.data.frame(as.list(res1)), method = "rmcorr"),
        cbind(as.data.frame(as.list(res2)), method= "lmer")))
}

ls.res <- pbapply::pblapply(1:101,function(iSim){
  cbind(sim = iSim, warper(100))
})
dt.res <- as.data.table(do.call(rbind, ls.res))
```

We can clearly see that the `rmcorr` estimator is biased and very variable while the `lmer`-based estimator (i.e. using Equation 1) gives reasonnable results:

```
rbind(lm = quantile(dt.res[method=="lm",estimate]),
      rmcorr = quantile(dt.res[method=="rmcorr",estimate]),
      lmer = quantile(dt.res[method=="lmer",estimate]))
```

```
              0%       25%       50%       75%      100%
lm     0.74113022 0.7802503 0.7989544 0.8145657 0.8544672
rmcorr 0.05260171 0.4773873 0.6189341 0.7273679 0.8425171
lmer   0.73416614 0.7739002 0.7937885 0.8103669 0.8529145
```

Note that the linear regression approach can be fixed in that example by adjusting on time. However with more complex covariance pattern it may not always be possible to find an appropriate `lm` approach.

5 Reference

Bakdash, J. Z. and Marusich, L. R. (2017). Repeated measures correlation. *Frontiers in psychology*, 8:456.

Bland, J. M. and Altman, D. G. (1995). Calculating correlation coefficients with repeated observations: Part 2—correlation between subjects. *Bmj*, 310(6980):633.

Lipsitz, S. R., Leong, T., Ibrahim, J., and Lipshultz, S. (2001). A partial correlation coefficient and coefficient of determination for multivariate normal repeated measures data. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 50(1):87–95.

Shan, G., Zhang, H., and Jiang, T. (2020). Correlation coefficients for a study with repeated measures. *Computational and mathematical methods in medicine*, 2020.