Mixed Model Workshop Part 1

Lorna Le Stanc

10/22/2021

Contents

1 Goal of this workshop session

In this part one we will understand the difference between fixed and main effects.

We will try to understand the math behind linear model and what is called "contrast coding".

2 Needed libraries

```
# to manipulate and plot data
library(tidyverse)

# to do an anova
library(ez)

# to obtain main effects from linear models and linear mixed models
library(rstatix)
```

3 Dataset for the workshop

Here we will work on reaction times (continuous variable)

- Two groups (control vs test)
- Two sessions (T0 vs T1)
- 4 items per sessions (each with 5 trials) -> 20 trials per participant and session.

This is a 2x2 design, with group as a between-subjects variables and session as a within-subject variable.

The dataset was built such as :

```
# Number of subjects per groups (2 groups, control/test)
N = 30
# number of trials per subject
ntrials = 20
# reaction time at T0 for both groups (ms)
int.T0 = 950
# Retest effect
slope.retest = 100
# Training effect
slope.training = 150
# variance among subjects at T0
sd = 50
# variance among retest and training effects is sd/2
```

Columns in dataset :

```
## [1] "participant" "group"       "session"     "trial"       "rt"
## [6] "item"
```

4 What you usually do (I guess) aka ANOVA and Main effects

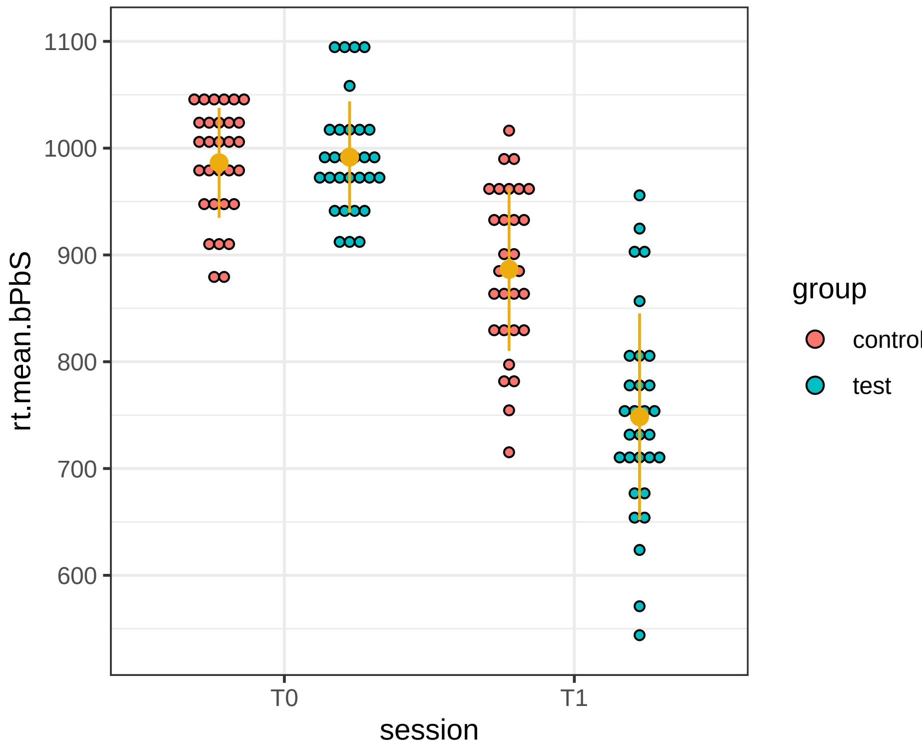
4.1 Mean by participant and session across trials and items

```
data.mean = data %>%
  group_by(participant, group,session)%>%
  summarise(rt.mean.bPbS = mean(rt))%>%
  ungroup()
```

rt.mean.bPbS for mean rt by participant by session.

```
## [1] "participant"  "group"        "session"      "rt.mean.bPbS"
```

You go from (30x2x2x20) = 2400 observations to 30x2x2 = 120 observations.



Each point represents the mean reaction time of a participant in a given session.

4.2 Runing an ANOVA in R using ez Package

```
aov = ezANOVA(data.mean,
              dv = rt.mean.bPbS,
              wid = participant,
              within = session,
              between = group,
              return_aov = TRUE)

aov$ANOVA
```

```
##          Effect DFn DFd          F            p p<.05       ges
## 2         group   1  58   13.77621 4.637192e-04     * 0.1811267
## 3       session   1  58 1246.37963 6.615923e-41     * 0.5963669
## 4 group:session   1  58  218.03776 2.624174e-21     * 0.2053836
```

dv = dependent variable ie what we measure, ie our outcome variable
wid = identification for our participants
within = for within-participant variables
between = for between session variable

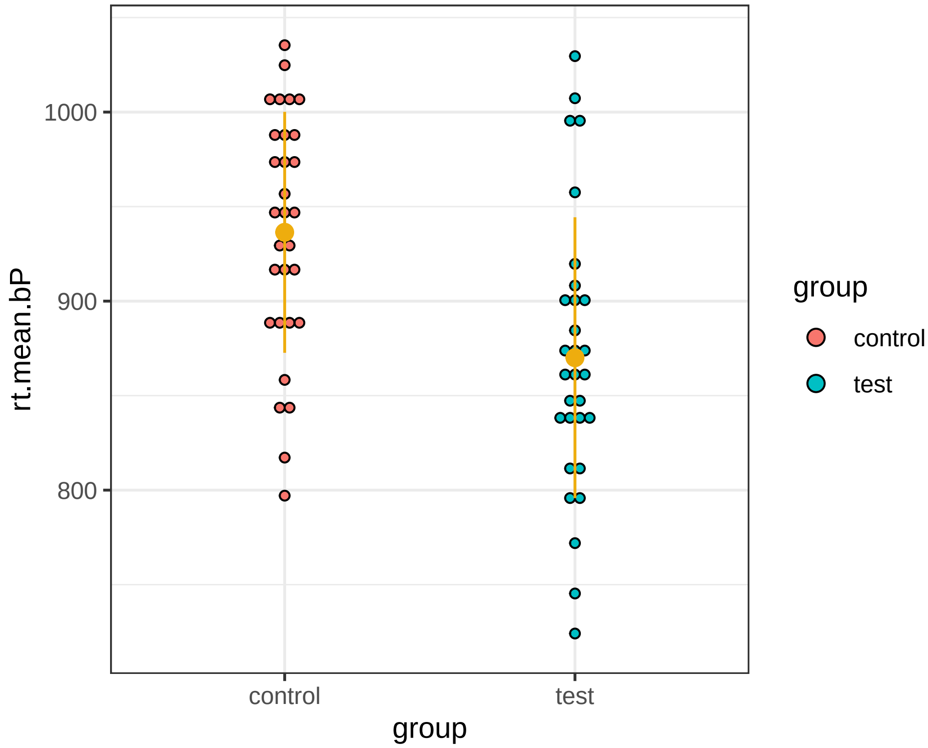
We could add covariates, chosse the type of anova etc. See `help(ezANOVA)` for more information.

4.3 Interpreting an ANOVA

In ANOVAs you are comparing whether the means are different in one variable (let's say group) across all the levels of the other variable (here session). That is called a **main effect**

4.3.1 Group effect

```
data.group = data.mean%>%
  group_by(participant, group)%>%
  summarise(rt.mean.bP = mean(rt.mean.bPbS))%>%
  ungroup()
```



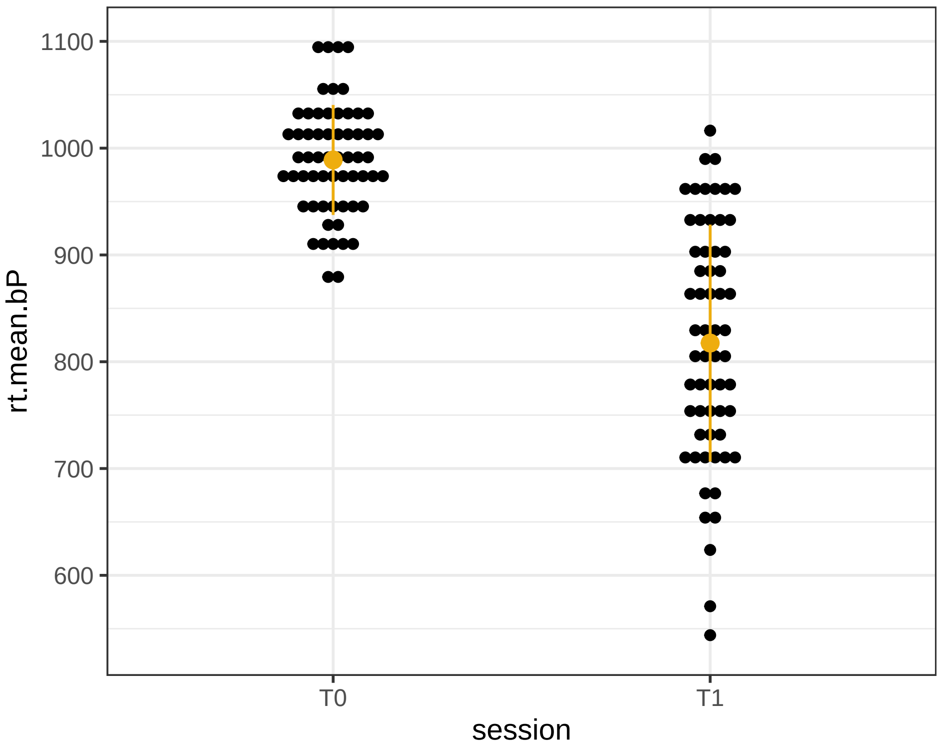
Each point represents the mean reaction time of one participant across both sessions.

> There is a main effect of group across all sessions. With group test being faster than the control group.

Notice that it can be misleading because at T0, the two groups do not seam to be different. The difference is driven by the effect at T1.

4.3.2 Session effect

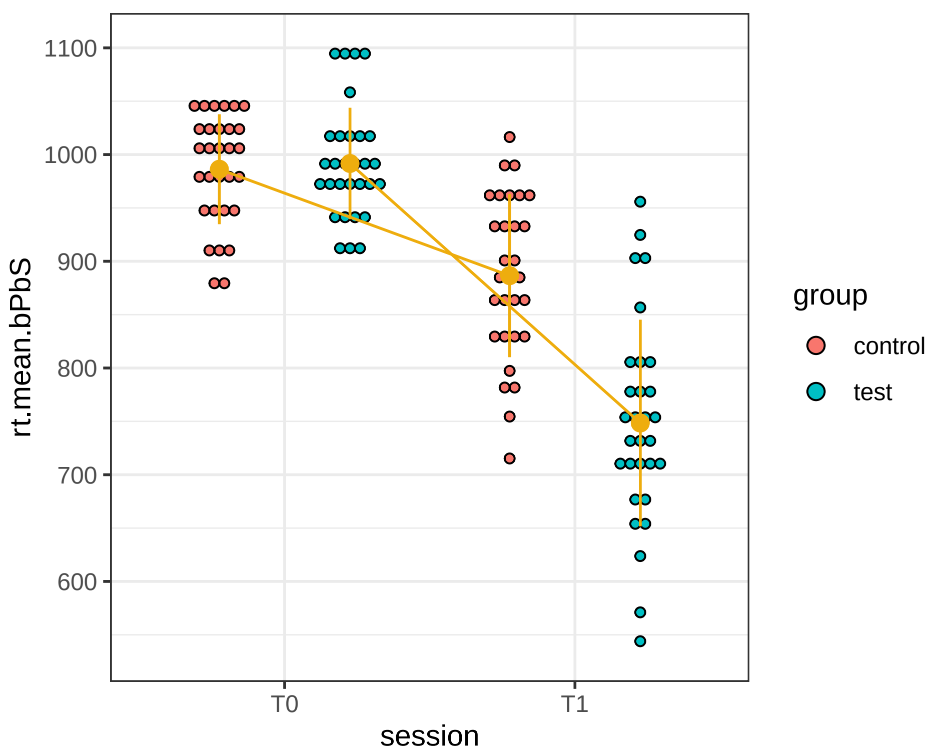
```
data.session = data.mean%>%
  group_by(session, participant)%>%
  summarise(rt.mean.bP = mean(rt.mean.bPbS))%>%
  ungroup()
```



Each point represents the mean reaction time of one participant.

There is a main effect of session across groups. With participants being faster at T1 than T0.

4.3.3 Interaction effect



There is an interaction between group and session meaning that the effect of group is different at

T0 and T1. It also means that the effect of session is different according to the group (represented on the figure).

The interaction is the statistical test to say that the slopes are different.

5 Short reminder on linear models and FIXED effect. How to interpret stuff.

So far we have talked about main effect in Anovas. Linear models (and mixed linear models) do not have main effects. At least no strait out of the model output. You can get main effects from linear models and we will see how. But we will try to understand first what are fixed effect and how to interpret the output of a linear model.

5.1 The mathematics behind linear models (because we will need it later)

Equation for a linear regression :

---

y = b + ax

means

y = Intercept + Slope * x

Where

- y is the dependent variable or outcome
- x is the independent variable or predictor

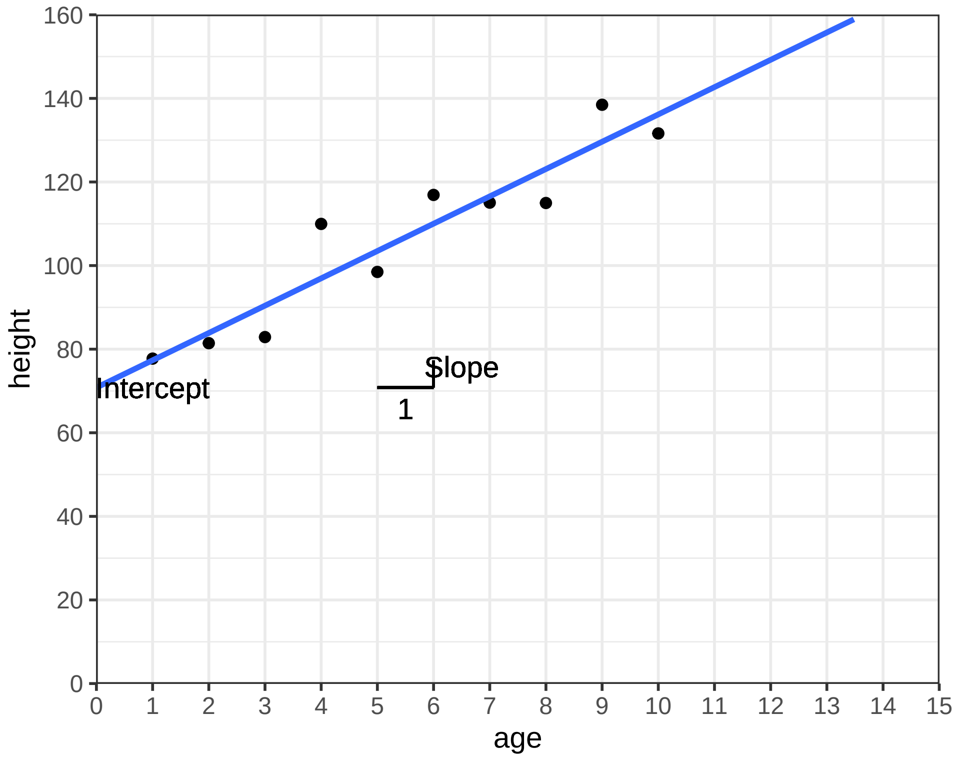
-> The predictor as an effect on the outcome

5.2 Interpretation with a continuous predictor

- Intercept is the value of y when x = 0 :
  y = Intercept + Slope*0 = Intercept
- Slope is the slope ie. how much you increase in y when you increase x by one :
  [Intercept + Slope*(.. + 1)] - [Intercept + Slope*(..)] = Slope

-> You write all of this y ~ x and the model give you an estimation of the Intercept and Slope

E.g height as a function of age



Each point represent a participant.
The blue line represent the fitted model ie the theoretical values ie the predicted values. The distance between a point and the blue line is the error of prediction for that point. It is call a residual.

```
lm.continuous = lm(height~ age, data_ageheight)
summary(lm.continuous)
```

```
##
## Call:
## lm(formula = height ~ age, data = data_ageheight)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -8.120 -4.889 -1.976  5.261 13.018
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  70.8084     5.2460   13.50 8.71e-07 ***
## age           6.5352     0.8455    7.73 5.59e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.679 on 8 degrees of freedom
## Multiple R-squared:  0.8819, Adjusted R-squared:  0.8672
## F-statistic: 59.75 on 1 and 8 DF,  p-value: 5.588e-05
```

"Call" line tells you the model you used as well as the data you fed into the model.

"Residuals" lines tell you the minimum, first quartile, median, 3rd quartile and maximum of your residuals. Knowing that your residuals are supposed to be normally distributed under linear model assumptions, you can use these line to quickly ckeck that there is not a huge problem. You can also plot your residual to check this assumption.

```
plot(residuals(lm.continuous))

qqnorm(resid(lm.continuous) )
qqline(resid(lm.continuous), col = "red")
```

"Coefficients" show you the estimated values for the Intercept and Slope as well as the standard error, t-value and p-value.
Note that the line (Intercept) reports the statistical test that the Intercept is different from 0. Similarly the age line reports the statistical test for the slope for age being different from 0. The sign of the slope (plus or minus) tells you if you increase in y when you increase in x (here you increase in height when you increase in age).

The **R-squared** relates the percentage of variance explained in the model. The fundamental difference between multiple and adjusted is that when you add predictors to your model, the multiple R-squared will always increase, as a predictor will always explain some portion of the variance. Adjusted R-squared controls against this increase, and adds penalties for the number of predictors in the model.

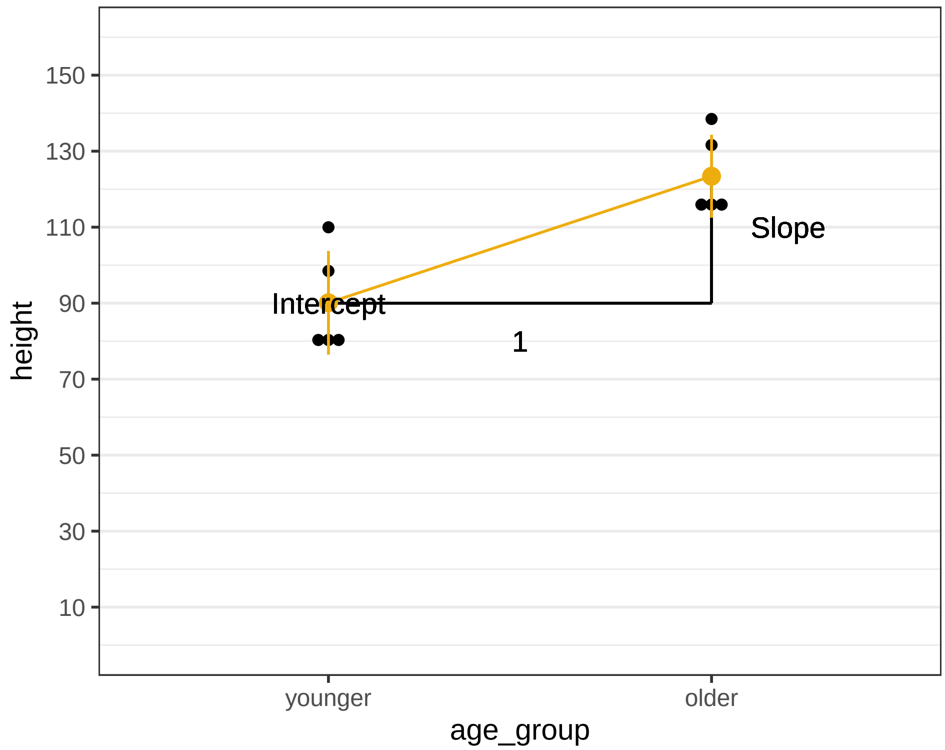
The **F-statistic** tells you if your model statically explains the data better than a model with no predictor (here yes, pvalue<.05).

5.3 Interpretation for a discreate predictor (ie categorical variable)

- WARNING : Intercept is the value of y in the group of reference.
- Slope is the difference between the two categories. How much you jump in y when you change categories in x.

Indeed mathematically, R consider one group to be equal to 0 (the reference group), and the other to 1. So again the slope is how much you jump in y when you jump one in x except that here you can only jump by one on x.

Eg. height as a function of age group

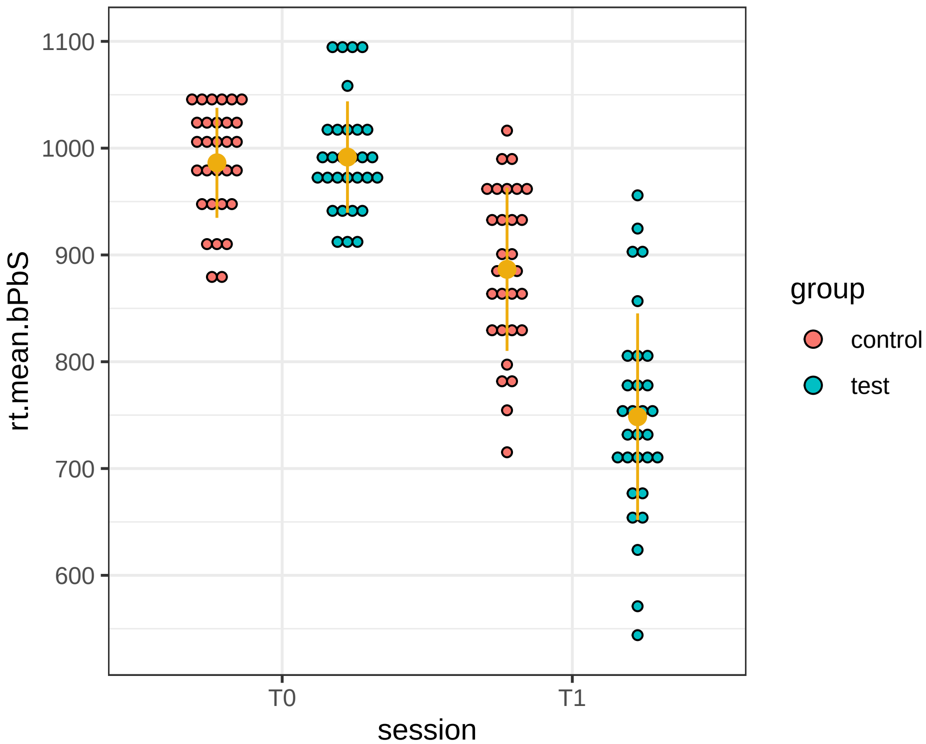



```
##
## Call:
## lm(formula = height ~ age_group, data = data_ageheight)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -12.364  -8.412  -6.854   8.338  19.867
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)         90.101      5.528  16.297 2.02e-07 ***
## age_groupolder      33.303      7.818   4.259  0.00276 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.36 on 8 degrees of freedom
## Multiple R-squared:  0.694,  Adjusted R-squared:  0.6557
## F-statistic: 18.14 on 1 and 8 DF,  p-value: 0.002763
```

Note the age_group**older** It means that the reference group was the youger group. (you can change it by releveling the factor group and changing its reference to older, by default alphabetical order). It will be important when we have more than one categorical predictor.

5.4 Interpretation for two discreate predictors

Let's get back to our previous data set and study the effect of group and condition on the reaction time.



Mathematically :

---

rt.mean.bPbS = Intercept + a x group + b x session + c x groupxsession

Notations Fixed Effects and interaction:

- + means no interaction
- : means interaction
- * means fixed effect plus interaction

Model : `rt.mean.bPbS ~ group*session` same as `rt.mean.bPbS ~ group + session + group:session`

Your model give you the estimation of the
- Intercept (mean rt value in ref level of group and session, *here controls at T0*)
- a (fixed effect of group in the base level of session, *here test vs controls at T0*),
- b (fixed effect of session in the base level of group, *here T1 vs T0 in controls*),
- c (interaction = difference in slopes)

```
lm = lm(rt.mean.bPbS ~ group*session, data = data.mean)
summary(lm)
```

```
##
## Call:
## lm(formula = rt.mean.bPbS ~ group * session, data = data.mean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -204.504  -42.796   -3.007   44.315  207.365
##
```

```
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          986.272     13.097  75.305  < 2e-16 ***
## grouptest              5.373     18.522   0.290    0.772
## sessionT1            -99.746     18.522  -5.385 3.83e-07 ***
## grouptest:sessionT1 -143.428     26.194  -5.476 2.56e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 71.73 on 116 degrees of freedom
## Multiple R-squared:  0.6618, Adjusted R-squared:  0.6531
## F-statistic: 75.68 on 3 and 116 DF,  p-value: < 2.2e-16
```

Note that there is no effect of group contrary to the anova.

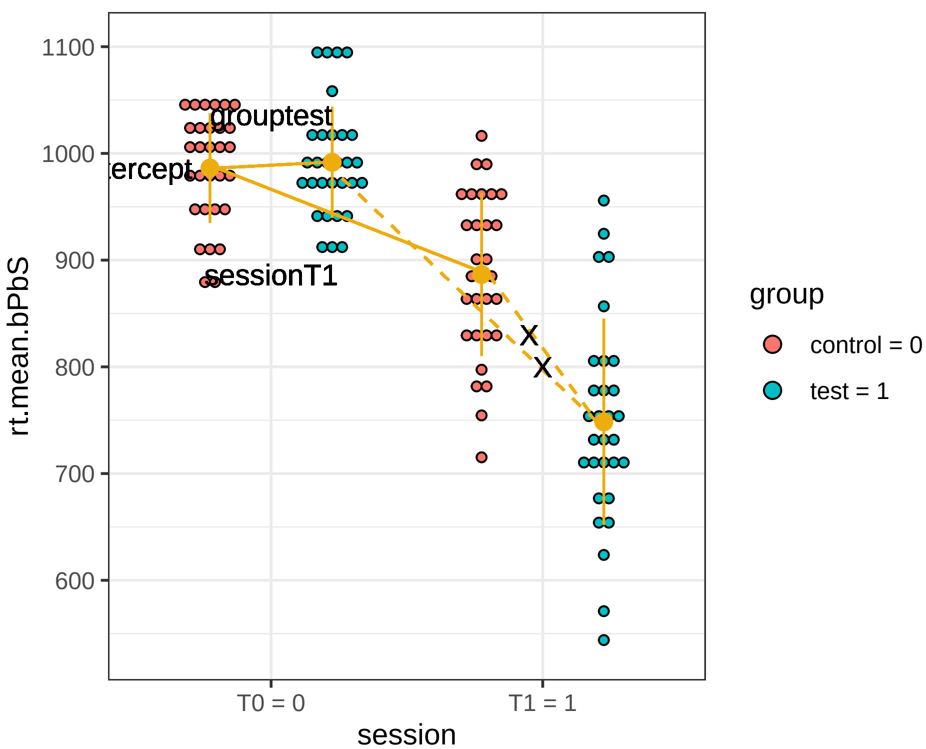
Mathematically :

---

rt.mean.bPbS = Intercept + grouptest*group + sessionT1*session + grouptest:sessionT1*group*session

Where

- session = 0 if T0, session = 1 if T1
- group = 0 if control, group = 1 if test
- Intercept is the minimum in ref level (session = 0, group = 0)



Here "sessionT1" is negative meaning that the reaction times decrease from T0 to T1 in controls (retest effect).

The interaction "grouptest:sessionT1" is negative meaning that the effect of session is stronger in the test group (retest plus training effect).

Note that if "sessionT1" was positive, a negative interaction would have a different meaning. The best way to interpret your interaction is to plot it!

6 How to get main effects (Anova style) from linear models

6.1 get main effects, style 1 = Anova function from rstatix library

```
Anova(lm)
```

```
## Anova Table (Type II tests)
##
## Response: rt.mean.bPbS
##               Sum Sq  Df F value    Pr(>F)
## group         132034   1  25.658 1.553e-06 ***
## session       881955   1 171.390 < 2.2e-16 ***
## group:session 154286   1  29.982 2.561e-07 ***
## Residuals     596925 116
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note that the effect of group is back.

6.2 get main effects, style 2 = from dummy coding (0,1) to contrast coding (-0.5,0.5)

```
data.mean.contrast = data.mean%>%
  # create a new variable containing -0.5 when session is T0 and +0.5 when session is T1
  mutate(session.contrast = case_when(
    session == "T0" ~ -0.5,
    session == "T1" ~ 0.5
  ))%>%
  mutate(group.contrast = case_when(
    # create a new variable containing -0.5 when group is control and +0.5 when group is test.
    group == "control" ~ -0.5,
    group == "test" ~ 0.5
  ))
```

```
##
## Call:
## lm(formula = rt.mean.bPbS ~ group.contrast * session.contrast,
##     data = data.mean.contrast)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -204.504  -42.796   -3.007   44.315  207.365
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      903.228      6.548 137.930  < 2e-16 ***
## group.contrast                   -66.341     13.097  -5.065 1.55e-06 ***
## session.contrast                -171.460     13.097 -13.092  < 2e-16 ***
## group.contrast:session.contrast -143.428     26.194  -5.476 2.56e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 71.73 on 116 degrees of freedom
## Multiple R-squared:  0.6618, Adjusted R-squared:  0.6531
## F-statistic: 75.68 on 3 and 116 DF,  p-value: < 2.2e-16
```

Back to the main effect results.

Contrary to the Anova you get estimates of your main effects.

The estimates are different but the interaction remains the same.

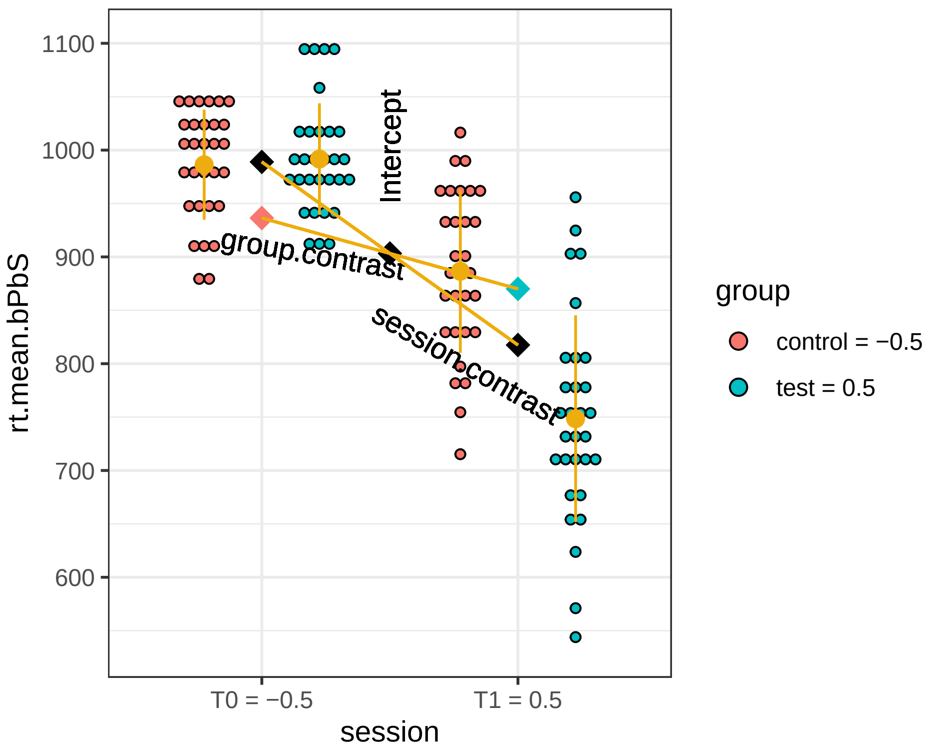
Mathematically:

---

rt.mean.bPbS = Intercept + grouptest*group + sessionT1*session + grouptest:sessionT1*group*session

Where

- session = -0.5 if T0, session = 0.5 if T1
- group = -0.5 if control, group = 0.5 if test
- The intercept is the mean over all data (session = 0, group = 0)



Note that the mean of the contrast should be 0 to set the Intercept to the mean of the overall data, and the difference to 1 to have the estimates equals to the differences between condition. "If you jump 1 in x, you jump your estimate in y"

For example if you use (-1,1), the Intercept will still be the mean of the overall data but the estimates will be equal to half the difference between conditions. "Here we have to jump 2 on x to have the real estimate. So if you jump 1, you get estimate/2"

```
data.mean.contrastdouble = data.mean%>%
  mutate(session.contrast = case_when(
    session == "T0" ~ -1,
```

```r
    session == "T1" ~ 1
  ))%>%
  mutate(group.contrast = case_when(
    group == "control" ~ -1,
    group == "test" ~ 1
  ))
```

```r
summary(lm(rt.mean.bPbS ~ group.contrast*session.contrast,
           data = data.mean.contrastdouble))
```

```
##
## Call:
## lm(formula = rt.mean.bPbS ~ group.contrast * session.contrast,
##     data = data.mean.contrastdouble)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -204.504  -42.796   -3.007   44.315  207.365
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     903.228      6.548 137.930  < 2e-16 ***
## group.contrast                  -33.170      6.548  -5.065 1.55e-06 ***
## session.contrast                -85.730      6.548 -13.092  < 2e-16 ***
## group.contrast:session.contrast -35.857      6.548  -5.476 2.56e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 71.73 on 116 degrees of freedom
## Multiple R-squared:  0.6618, Adjusted R-squared:  0.6531
## F-statistic: 75.68 on 3 and 116 DF,  p-value: < 2.2e-16
```