Self-identification with a virtual experience and its moderating effect on self-efficacy and presence

Data analysis in R

Ni Kang, Ding Ding, M. Birna van Riemsdijk, Nexhmedin Morina, Mark A.Neerincx, and Willem-Paul Brinkman

31/12/2019

Contents

Introduction

This document presents inferential statitical analyses of participants' PRCS, ITQ, Capability comparison, Virtual performance, Self-identification, SUD, HR, Presence response in vicarious experience with different level of self-identifiaction (Low vs High self-identification) in different sessions as reported in the paper:

Self-identification with a virtual experience and its moderating effect on self-efficacy and presence

Authored by Ni Kang, Ding Ding, M. Birna van Riemsdijk, Nexhmedin Morina, Mark A.Neerincx, and Willem-Paul Brinkman.

```
library(pander) #for rendering output
library(foreign)
library(car)
library(ggplot2)
library(nlme)
library(reshape)
library(lattice)
library(MASS)
library(psych)
library(pwr)
library(ggpubr)
library(dplyr)
library(lsr)
library(outliers)
library(coin)
library(lme4)
library(ez)
library(afex)
library(Hmisc)
library(rmcorr)
library(cocor)
```

Data files

Data files for analysis

Data files is stored in all_data.csv

Functions

```
testbetween <-function(sample1,sample2)
{
  # x integer data being tested
  # g group

  # convert group into factor:
  #g <- factor(g)
  x<-c(sample1, sample2)
```

```
  g<-as.factor(c(rep(1, length(sample1)), rep(2, length(sample2))))

  l<-leveneTest(x,g)
  if (l$`Pr(>F)`[1]<.05)
  {
    cat("\nLevene's test shows sign difference between the group variance, therefore  Welch Two Sample t
    print(t.test(x~g, na.rm = TRUE,var.equal = FALSE))
  } else
  {
    cat("\nLevene's test shows no sign difference between the group variance, therefore  Two Sample t-te
    print(t.test(x~g, na.rm = TRUE,var.equal = TRUE))
  }
}

print_ezANOVA <- function(m){
  #print output ezANOVA, including effectsize eta, and Greenhouse-Geisser corrected dfs
  print(m)

  m$ANOVA$eta <- m$ANOVA$SSn/(m$ANOVA$SSn + m$ANOVA$SSd)
  for (i in 1:length(m$ANOVA$Effect))
  { cat(m$ANOVA$Effect[i]," : eta ", m$ANOVA$eta[i]," \n")
  }
  cat("see ges-value for Generalized Eta-Squared) \n\n")

  cat("Greenhouse-Geisser corrected Dfs. \n")
  for (i in 1:length(m$`Sphericity Corrections`$Effect))
  {
    for (j in 1:length(m$ANOVA$Effect)) {
      if (m$ANOVA$Effect[j] == m$`Sphericity Corrections`$Effect[i]){
        m$`Sphericity Corrections`$DFn_G[i] <- m$ANOVA$DFn[j] * m$`Sphericity Corrections`$GGe[i]
        m$`Sphericity Corrections`$DFd_G[i] <- m$ANOVA$DFd[j] * m$`Sphericity Corrections`$GGe[i]
        cat(m$`Sphericity Corrections`$Effect[i],
            " numerator ", m$`Sphericity Corrections`$DFn_G[i], " , denominator ", m$`Sphericity Correc

      }
    }
  }
}
```

Data preparation

```
setwd("/Users/ding/Documents/Ni second paper/data")
VirtualExperience<- read.csv("all_data.csv", header = TRUE, sep=",")
Length<- read.csv("Length.csv", header = TRUE, sep=";")
```

Results reported in Section 3. Participants

Analysis for demographic data: age

```
describe(VirtualExperience$age)
```

```
## VirtualExperience$age
```

```
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##       60        0       18    0.993     26.3    5.072    20.00    20.90
##      .25      .50      .75      .90      .95
##    23.00    26.00    29.00    32.00    34.05
##
## Value         19    20    21    22    23    24    25    26    27    28
## Frequency      1     5     3     2     7     4     6     8     3     4
## Proportion 0.017 0.083 0.050 0.033 0.117 0.067 0.100 0.133 0.050 0.067
##
## Value         29    30    31    32    34    35    37    42
## Frequency      5     2     3     2     2     1     1     1
## Proportion 0.083 0.033 0.050 0.033 0.033 0.017 0.017 0.017
```

Analysis for demographic data: gender

- "1" respresnts for male, "2" respresnts for female

```
count(VirtualExperience,gender)
```

```
## Warning: The `printer` argument is deprecated as of rlang 0.3.0.
## This warning is displayed once per session.
```

```
## # A tibble: 2 x 2
##   gender     n
##    <int> <int>
## 1      1    36
## 2      2    24
```

```
count(VirtualExperience[VirtualExperience$cond==1,],gender)
```

```
## # A tibble: 2 x 2
##   gender     n
##    <int> <int>
## 1      1    18
## 2      2    12
```

```
count(VirtualExperience[VirtualExperience$cond==0,],gender)
```

```
## # A tibble: 2 x 2
##   gender     n
##    <int> <int>
## 1      1    18
## 2      2    12
```

Analysis for demographic data: Nationality – skin colour

```
Nationality<-count(VirtualExperience,nationality)

#Nationality[Nationality$nationality=="brazilian"|
#                Nationality$nationality=="Brazilian"|
#                Nationality$nationality=="Indonesia"|
#                Nationality$nationality=="Indonesian"|
#                Nationality$nationality=="Mexican"|
#                Nationality$nationality=="Thai"|
#                Nationality$nationality=="singaporean",]
```

```r
#ParticipantswithDarkSkin<-Nationality[Nationality$nationality=="brazilian"|
#                 Nationality$nationality=="Brazilian"|
#                 Nationality$nationality=="Indonesia"|
#                 Nationality$nationality=="Indonesian"|
#                 Nationality$nationality=="Mexican"|
#                 Nationality$nationality=="Thai"|
#                 Nationality$nationality=="singaporean",]

ParticipantswithDarkSkin<-Nationality[Nationality$nationality=="Indian"|Nationality$nationality=="Indone

sum(ParticipantswithDarkSkin$n)
```

```
## [1] 8
```

```r
VirtualExperience_LightSkin<-subset(VirtualExperience,!(VirtualExperience$nationality=="Indian"|Virtual
```

Results reported in Section 4. Data preparation and statistical analysis

Cronbach's alpha was calculated for the questionnaires containing multiple items, such as Capability comparison, Presence response(PR), Presentation performance(PP), and Self-identification(SID). Reliability analysis shows acceptable reliability level (Cronbach's alpha >.7).

Reliability analysis on Self-identification(SID).

*Q4 is a revese scale, therefore the data of self_id_2 has to be reversed.

*Q6 has been removed because of the low correlation with other questions.

```r
VE_SID<- VirtualExperience[1:60,c(1,3,19,24,25,26,27,28,30)]
VE_SID[] <- lapply(VE_SID, function(x) as.numeric(as.character(x)))
VE_SID$self_id_2[VE_SID$self_id_2 == 6] <- 11
VE_SID$self_id_2[VE_SID$self_id_2 == 7] <- 12
VE_SID$self_id_2[VE_SID$self_id_2 == 8] <- 13
VE_SID$self_id_2[VE_SID$self_id_2 == 9] <- 14
VE_SID$self_id_2[VE_SID$self_id_2 == 10] <- 15
VE_SID$self_id_2[VE_SID$self_id_2 == 0] <- 10
VE_SID$self_id_2[VE_SID$self_id_2 == 1] <- 9
VE_SID$self_id_2[VE_SID$self_id_2 == 2] <- 8
VE_SID$self_id_2[VE_SID$self_id_2 == 3] <- 7
VE_SID$self_id_2[VE_SID$self_id_2 == 4] <- 6
VE_SID$self_id_2[VE_SID$self_id_2 == 11] <- 4
VE_SID$self_id_2[VE_SID$self_id_2 == 12] <- 3
VE_SID$self_id_2[VE_SID$self_id_2 == 13] <- 2
VE_SID$self_id_2[VE_SID$self_id_2 == 14] <- 1
VE_SID$self_id_2[VE_SID$self_id_2 == 15] <- 0
alpha_SID<-alpha(VE_SID[c(3:9)])
alpha_SID$total
```

```
##  raw_alpha std.alpha   G6(smc) average_r      S/N        ase     mean
##  0.8291767 0.8384205 0.8637615 0.4257072 5.188904 0.03386241 5.466667
##        sd  median_r
##  2.024216 0.4164972
```

Reliability test for Presence response(PR)

Reliability analysis on Presence response(PR) during passive virtual experience.

```
VE_PR<- VirtualExperience[1:60,c(1,3,35:37,43:45)]
VE_PR[] <- lapply(VE_PR, function(x) as.numeric(as.character(x)))
alpha_VE<-alpha(VE_PR[3:5])
alpha_VE$total
```

```
##  raw_alpha std.alpha   G6(smc) average_r   S/N        ase     mean      sd
##   0.850324 0.8545243 0.8350774 0.6619337 5.874 0.03501139 4.205556 2.34267
##   median_r
##  0.7322882
```

Reliability analysis on Presence response(PR) during post-measurement presentation.

```
alpha_Post<-alpha(VE_PR[6:8])
alpha_Post$total
```

```
##  raw_alpha std.alpha   G6(smc) average_r      S/N        ase     mean
##  0.8805197 0.8852194 0.8414663 0.7199475 7.712277 0.02671141 6.522222
##        sd  median_r
##  2.036731 0.7467201
```

Reliability test for Capability comparison

```
VE_Capability<- VirtualExperience[1:60,c(1,3,17,31,33)]
VE_Capability[] <- lapply(VE_Capability, function(x) as.numeric(as.character(x)))
alpha(VE_Capability[4:5])
```

```
## Warning in matrix(unlist(drop.item), ncol = 10, byrow = TRUE): data length
## [16] is not a sub-multiple or multiple of the number of columns [10]
```

```
##
## Reliability analysis
## Call: alpha(x = VE_Capability[4:5])
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean  sd median_r
##        0.8       0.8    0.67      0.67   4 0.051    5 1.7     0.67
##
##  lower alpha upper     95% confidence boundaries
## 0.7 0.8 0.9
##
##  Reliability if an item is dropped:
##                 raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r
## comp_capability      0.67      0.67    0.45      0.67  NA       NA  0.67
## comp_perform         0.45      0.67      NA        NA  NA       NA  0.45
##                 med.r
## comp_capability  0.67
## comp_perform     0.67
##
##  Item statistics
##                  n raw.r std.r r.cor r.drop mean  sd
## comp_capability 60  0.92  0.91  0.75   0.67  4.9 1.8
## comp_perform    60  0.91  0.91  0.75   0.67  5.1 1.8
```

Reliability analysis on Presentation performance(PP).

```
VE_PP<- VirtualExperience[1:60,c(1,3,40,41)]
VE_PP[] <- lapply(VE_PP, function(x) as.numeric(as.character(x)))
alpha_PP<-alpha(VE_PP[c(3:4)])
```

```
## Warning in matrix(unlist(drop.item), ncol = 10, byrow = TRUE): data length
## [16] is not a sub-multiple or multiple of the number of columns [10]
```

```
alpha_PP$total
```

```
##  raw_alpha std.alpha   G6(smc) average_r      S/N        ase  mean
##  0.7668503 0.7730492 0.6300572 0.6300572 3.406241 0.05891707 4.725
##        sd  median_r
##  2.227382 0.6300572
```

Results reported in Section 5. Results

Results reported in Table 3

- Descriptive statistics of the measures, Mean (SD), and results of independent t-tests between conditions and correlation with the self-efficacy

PRCS

- Mean and Standard deviation of high self-identification condition and low self-identification condition

```
VE_PRCS<- VirtualExperience[1:60,c(1,3,8,9)]
PRCS_High<-subset(VE_PRCS,cond==0)
PRCS_Low<-subset(VE_PRCS,cond==1)
describeBy(VE_PRCS,VE_PRCS$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##                 vars  n  mean    sd median trimmed   mad min max range
## ID1                1 30 31.27 18.99     29   30.92 24.46   2  63    61
## cond               2 30  0.00  0.00      0    0.00  0.00   0   0     0
## self_efficacy_1    3 30  6.87  2.47      7    7.08  2.97   1  10     9
## PRCS               4 30 12.03  6.53     12   11.79  8.15   2  28    26
##                  skew kurtosis   se
## ID1              0.12    -1.43 3.47
## cond              NaN      NaN 0.00
## self_efficacy_1 -0.68    -0.47 0.45
## PRCS             0.32    -0.74 1.19
## --------------------------------------------------------
## group: 1
##                 vars  n  mean    sd median trimmed   mad min max range
## ID1                1 30 31.83 18.18   33.0   32.04 20.76   1  61    60
## cond               2 30  1.00  0.00    1.0    1.00  0.00   1   1     0
## self_efficacy_1    3 30  7.07  2.50    7.5    7.33  2.22   2  10     8
## PRCS               4 30 12.47  6.39   12.0   12.12  8.15   2  25    23
##                  skew kurtosis   se
## ID1             -0.09    -1.24 3.32
## cond              NaN      NaN 0.00
## self_efficacy_1 -0.74    -0.60 0.46
```

```
## PRCS              0.34    -1.08 1.17
```

```r
testbetween(PRCS_High$PRCS,PRCS_Low$PRCS)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test cond
##
##  Two Sample t-test
##
## data:  x by g
## t = -0.25969, df = 58, p-value = 0.796
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -3.773507  2.906841
## sample estimates:
## mean in group 1 mean in group 2
##        12.03333        12.46667
```

*Correlation with self-efficacy measured at the same phase, Pearson's r

```r
cor_PRCSSE <- rcorr(as.matrix(VE_PRCS), type="pearson")
cor_PRCSSE$P
```

```
##                       ID1      cond self_efficacy_1         PRCS
## ID1                    NA 0.9064112    8.035086e-01 4.283053e-01
## cond            0.9064112        NA    7.567637e-01 7.960222e-01
## self_efficacy_1 0.8035086 0.7567637              NA 5.799742e-05
## PRCS            0.4283053 0.7960222    5.799742e-05           NA
```

```r
cor_PRCSSE$P[3,4]
```

```
## [1] 5.799742e-05
```

```r
cor_PRCSSE$r
```

```
##                         ID1       cond self_efficacy_1        PRCS
## ID1              1.00000000 0.01550298      0.03280249 -0.10417172
## cond             0.01550298 1.00000000      0.04082861  0.03407921
## self_efficacy_1  0.03280249 0.04082861      1.00000000 -0.49501966
## PRCS            -0.10417172 0.03407921     -0.49501966  1.00000000
```

```r
cor_PRCSSE$r[3,4]
```

```
## [1] -0.4950197
```

Age

- Mean and Standard deviation of high self-identification condition and low self-identification condition

```r
VE_Age<- VirtualExperience[1:60,c(1,3,5,8)]
describe(VE_Age$age)
```

```
## VE_Age$age
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##       60        0       18    0.993     26.3    5.072    20.00    20.90
##      .25      .50      .75      .90      .95
##    23.00    26.00    29.00    32.00    34.05
##
## Value         19     20     21     22     23     24     25     26     27     28
```

```
## Frequency      1     5     3     2     7     4     6     8     3     4
## Proportion 0.017 0.083 0.050 0.033 0.117 0.067 0.100 0.133 0.050 0.067
##
## Value         29    30    31    32    34    35    37    42
## Frequency      5     2     3     2     2     1     1     1
## Proportion 0.083 0.033 0.050 0.033 0.033 0.017 0.017 0.017
```

```
mean(VE_Age$age)
```

```
## [1] 26.3
```

```
sd(VE_Age$age)
```

```
## [1] 4.59292
```

```
Age_High<-subset(VE_Age,cond==0)
Age_Low<-subset(VE_Age,cond==1)
describeBy(VE_Age,VE_Age$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##                 vars  n  mean    sd median trimmed   mad min max range
## ID1                1 30 31.27 18.99     29   30.92 24.46   2  63    61
## cond               2 30  0.00  0.00      0    0.00  0.00   0   0     0
## age                3 30 26.27  4.07     26   26.12  4.45  20  34    14
## self_efficacy_1    4 30  6.87  2.47      7    7.08  2.97   1  10     9
##                  skew kurtosis   se
## ID1              0.12    -1.43 3.47
## cond              NaN      NaN 0.00
## age              0.20    -1.01 0.74
## self_efficacy_1 -0.68    -0.47 0.45
## ----------------------------------------------------------
## group: 1
##                 vars  n  mean    sd median trimmed   mad min max range
## ID1                1 30 31.83 18.18   33.0   32.04 20.76   1  61    60
## cond               2 30  1.00  0.00    1.0    1.00  0.00   1   1     0
## age                3 30 26.33  5.13   25.5   25.71  3.71  19  42    23
## self_efficacy_1    4 30  7.07  2.50    7.5    7.33  2.22   2  10     8
##                  skew kurtosis   se
## ID1             -0.09    -1.24 3.32
## cond              NaN      NaN 0.00
## age              1.13     1.25 0.94
## self_efficacy_1 -0.74    -0.60 0.46
```

```
testbetween(Age_High$age,Age_Low$age)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test cond
##
##  Two Sample t-test
##
## data:  x by g
## t = -0.05574, df = 58, p-value = 0.9557
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.460789  2.327456
```

```
## sample estimates:
## mean in group 1 mean in group 2
##        26.26667        26.33333
```

*Correlation with self-efficacy measured at the same phase, Pearson's r

```
cor_AgeSE <- rcorr(as.matrix(VE_Age), type="pearson")
cat("\n## The corrlation check\n")
```

```
##
## ## The corrlation check
```

```
cor_AgeSE$P
```

```
##                        ID1      cond       age self_efficacy_1
## ID1                     NA 0.9064112 0.5190633       0.8035086
## cond             0.9064112        NA 0.9557407       0.7567637
## age              0.5190633 0.9557407        NA       0.2197963
## self_efficacy_1 0.8035086 0.7567637 0.2197963              NA
```

```
cor_AgeSE$P[3,4]
```

```
## [1] 0.2197963
```

```
cor_AgeSE$r
```

```
##                          ID1        cond          age self_efficacy_1
## ID1               1.00000000 0.015502981 -0.084877084      0.03280249
## cond              0.01550298 1.000000000  0.007318793      0.04082861
## age              -0.08487708 0.007318793  1.000000000      0.16076309
## self_efficacy_1   0.03280249 0.040828610  0.160763088      1.00000000
```

```
cor_AgeSE$r[3,4]
```

```
## [1] 0.1607631
```

IQT

- Mean and Standard deviation of high self-identification condition and low self-identification condition

```
VE_IQT<- VirtualExperience[1:60,c(1,3,8,14)]
IQT_High<-subset(VE_IQT,cond==0)
IQT_Low<-subset(VE_IQT,cond==1)
describeBy(VE_IQT,VE_IQT$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##                 vars  n  mean    sd median trimmed   mad min max range
## ID1                1 30 31.27 18.99   29.0   30.92 24.46   2  63    61
## cond               2 30  0.00  0.00    0.0    0.00  0.00   0   0     0
## self_efficacy_1    3 30  6.87  2.47    7.0    7.08  2.97   1  10     9
## ITQ                4 30 70.10 11.63   70.5   70.17  8.90  44  96    52
##                  skew kurtosis   se
## ID1              0.12    -1.43 3.47
## cond              NaN      NaN 0.00
## self_efficacy_1 -0.68    -0.47 0.45
## ITQ             -0.11    -0.03 2.12
## --------------------------------------------------------
```

```
## group: 1
##                  vars  n  mean    sd median trimmed   mad min max range
## ID1                 1 30 31.83 18.18   33.0   32.04 20.76   1  61    60
## cond                2 30  1.00  0.00    1.0    1.00  0.00   1   1     0
## self_efficacy_1     3 30  7.07  2.50    7.5    7.33  2.22   2  10     8
## ITQ                 4 30 62.73 11.08   61.5   62.38 12.60  41  88    47
##                   skew kurtosis   se
## ID1              -0.09    -1.24 3.32
## cond               NaN      NaN 0.00
## self_efficacy_1 -0.74    -0.60 0.46
## ITQ               0.27    -0.68 2.02
```

```
testbetween(IQT_High$ITQ,IQT_Low$ITQ)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = 2.5126, df = 58, p-value = 0.01479
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   1.497826 13.235508
## sample estimates:
## mean in group 1 mean in group 2
##        70.10000        62.73333
```

*Correlation with self-efficacy measured at the same phase, Pearson's r

```
cor_IQTSE <- rcorr(as.matrix(VE_IQT), type="pearson")
cat("\n## The corrlation check\n")
```

```
##
## ## The corrlation check
```

```
cor_IQTSE$P
```

```
##                        ID1       cond self_efficacy_1        ITQ
## ID1                     NA 0.90641117       0.8035086 0.43218039
## cond             0.9064112         NA       0.7567637 0.01478766
## self_efficacy_1 0.8035086 0.75676367              NA 0.32601479
## ITQ              0.4321804 0.01478766       0.3260148         NA
```

```
cor_IQTSE$P[3,4]
```

```
## [1] 0.3260148
```

```
cor_IQTSE$r
```

```
##                         ID1        cond self_efficacy_1        ITQ
## ID1               1.00000000  0.01550298      0.03280249 -0.1033046
## cond              0.01550298  1.00000000      0.04082861 -0.3133082
## self_efficacy_1  0.03280249  0.04082861      1.00000000  0.1289805
## ITQ              -0.10330461 -0.31330817      0.12898054  1.0000000
```

```
cor_IQTSE$r[3,4]
```

```
## [1] 0.1289805
```

Capability comparison

- Mean and Standard deviation of high self-identification condition and low self-identification condition

```
VE_Capability_Formalized<- data.frame(VE_Capability$ID1,VE_Capability$cond,VE_Capability$self_efficacy_1
colnames(VE_Capability_Formalized) <- c("ID","cond","SE","Capability")
Capability_High<-subset(VE_Capability_Formalized,cond==0)
Capability_Low<-subset(VE_Capability_Formalized,cond==1)
describeBy(VE_Capability_Formalized,VE_Capability_Formalized$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##            vars  n  mean    sd median trimmed   mad min max range  skew
## ID            1 30 31.27 18.99     29   30.92 24.46 2.0  63  61.0  0.12
## cond          2 30  0.00  0.00      0    0.00  0.00 0.0   0   0.0   NaN
## SE            3 30  5.10  2.66      5    4.92  2.97 2.0  10   8.0  0.42
## Capability    4 30  5.02  1.58      5    5.10  1.48 0.5   8   7.5 -0.62
##            kurtosis   se
## ID            -1.43 3.47
## cond            NaN 0.00
## SE            -1.18 0.49
## Capability     0.40 0.29
## --------------------------------------------------------
## group: 1
##            vars  n  mean    sd median trimmed   mad min max range  skew
## ID            1 30 31.83 18.18     33   32.04 20.76   1  61    60 -0.09
## cond          2 30  1.00  0.00      1    1.00  0.00   1   1     0   NaN
## SE            3 30  6.80  2.17      7    7.00  1.48   1  10     9 -0.79
## Capability    4 30  5.02  1.75      5    4.98  1.48   2  10     8  0.45
##            kurtosis   se
## ID            -1.24 3.32
## cond            NaN 0.00
## SE             0.01 0.40
## Capability     0.43 0.32
```

```
testbetween(Capability_High$Capability,Capability_Low$Capability)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test cond
##
##  Two Sample t-test
##
## data:  x by g
## t = 0, df = 58, p-value = 1
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.8637306  0.8637306
## sample estimates:
## mean in group 1 mean in group 2
##        5.016667        5.016667
```

*Correlation with self-efficacy measured at the same phase, Pearson's r

```
cor_VE_Capability_FormalizedSE <- rcorr(as.matrix(VE_Capability_Formalized), type="pearson")
cor_VE_Capability_FormalizedSE$P
```

```
##                    ID        cond           SE   Capability
## ID                 NA 0.906411171 0.7013730560 0.3891614401
## cond       0.9064112          NA 0.0087534673 1.0000000000
## SE         0.7013731 0.008753467           NA 0.0006310292
## Capability 0.3891614 1.000000000 0.0006310292           NA
```

```
cor_VE_Capability_FormalizedSE$P[3,4]
```

```
## [1] 0.0006310292
```

```
cor_VE_Capability_FormalizedSE$r
```

```
##                     ID       cond          SE Capability
## ID          1.00000000 0.01550298 -0.0505369  0.1131958
## cond        0.01550298 1.00000000  0.3356208  0.0000000
## SE         -0.05053690 0.33562075  1.0000000  0.4287516
## Capability  0.11319584 0.00000000  0.4287516  1.0000000
```

```
cor_VE_Capability_FormalizedSE$r[3,4]
```

```
## [1] 0.4287516
```

Virtual performance

- Mean and Standard deviation of high self-identification condition and low self-identification condition

```
VE_VP<- VirtualExperience[1:60,c(1,3,17,23)]
VP_High<-subset(VE_VP,cond==0)
VP_Low<-subset(VE_VP,cond==1)
describeBy(VE_VP,VE_VP$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##                   vars  n  mean    sd median trimmed   mad min max range
## ID1                  1 30 31.27 18.99     29   30.92 24.46   2  63    61
## cond                 2 30  0.00  0.00      0    0.00  0.00   0   0     0
## self_efficacy_2      3 30  5.10  2.66      5    4.92  2.97   2  10     8
## performance_VE       4 30  6.90  2.35      7    7.08  2.97   2  10     8
##                    skew kurtosis   se
## ID1                0.12    -1.43 3.47
## cond                NaN      NaN 0.00
## self_efficacy_2    0.42    -1.18 0.49
## performance_VE    -0.41    -0.77 0.43
## --------------------------------------------------------
## group: 1
##                   vars  n  mean    sd median trimmed   mad min max range
## ID1                  1 30 31.83 18.18     33   32.04 20.76   1  61    60
## cond                 2 30  1.00  0.00      1    1.00  0.00   1   1     0
## self_efficacy_2      3 30  6.80  2.17      7    7.00  1.48   1  10     9
## performance_VE       4 30  7.67  1.81      8    7.83  1.48   4  10     6
##                    skew kurtosis   se
## ID1               -0.09    -1.24 3.32
## cond                NaN      NaN 0.00
## self_efficacy_2   -0.79     0.01 0.40
## performance_VE    -0.74    -0.58 0.33
```

```r
testbetween(VP_High$performance_VE,VP_Low$performance_VE)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = -1.4151, df = 58, p-value = 0.1624
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.8511559  0.3178226
## sample estimates:
## mean in group 1 mean in group 2
##        6.900000        7.666667
```

*Correlation with self-efficacy measured at the same phase, Pearson's r

```r
cor_VPSE <- rcorr(as.matrix(VE_VP), type="pearson")
cor_VPSE$P
```

```
##                        ID1        cond self_efficacy_2 performance_VE
## ID1                     NA 0.906411171     0.701373056     0.11404180
## cond             0.9064112          NA     0.008753467     0.16238896
## self_efficacy_2 0.7013731 0.008753467              NA     0.01110469
## performance_VE  0.1140418 0.162388958     0.011104694             NA
```

```r
cor_VPSE$P[3,4]
```

```
## [1] 0.01110469
```

```r
cor_VPSE$r
```

```
##                         ID1       cond self_efficacy_2 performance_VE
## ID1              1.00000000 0.01550298      -0.0505369     -0.2061523
## cond             0.01550298 1.00000000       0.3356208      0.1826836
## self_efficacy_2 -0.05053690 0.33562075       1.0000000      0.3256927
## performance_VE  -0.20615227 0.18268358       0.3256927      1.0000000
```

```r
cor_VPSE$r[3,4]
```

```
## [1] 0.3256927
```

Self-Identification (SID)

- Mean and Standard deviation of high self-identification condition and low self-identification condition

```r
VE_SID_Average<- data.frame(VE_SID$ID1,VE_SID$cond,rowMeans(VE_SID[3:9]))
colnames(VE_SID_Average) <- c("ID","cond","SID")
SID_High<-subset(VE_SID_Average,cond==0)
SID_Low<-subset(VE_SID_Average,cond==1)
describeBy(VE_SID_Average,VE_SID_Average$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##      vars  n  mean    sd median trimmed   mad min   max range  skew
## ID      1 30 31.27 18.99  29.00   30.92 24.46   2 63.00 61.00  0.12
```

```
## cond    2 30  0.00  0.00   0.00    0.00  0.00   0  0.00  0.00   NaN
## SID     3 30  6.62  1.52   6.79    6.70  1.38   3  9.29  6.29 -0.40
##      kurtosis   se
## ID      -1.43 3.47
## cond      NaN 0.00
## SID     -0.31 0.28
## --------------------------------------------------------
## group: 1
##      vars  n  mean    sd median trimmed   mad min max range  skew kurtosis
## ID      1 30 31.83 18.18  33.00   32.04 20.76   1  61    60 -0.09    -1.24
## cond    2 30  1.00  0.00   1.00    1.00  0.00   1   1     0   NaN      NaN
## SID     3 30  4.31  1.80   4.64    4.43  1.48   0   7     7 -0.53    -0.49
##        se
## ID   3.32
## cond 0.00
## SID  0.33
```

```
testbetween(SID_High$SID,SID_Low$SID)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = 5.3729, df = 58, p-value = 1.434e-06
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  1.452083 3.176489
## sample estimates:
## mean in group 1 mean in group 2
##        6.623810        4.309524
```

*Correlation with self-efficacy measured at the same phase, Pearson's r

```
VE_SID_SE<-data.frame(VE_SID_Average,VirtualExperience[c(17)])
cor_SIDSE <- rcorr(as.matrix(VE_SID_SE), type="pearson")
cor_SIDSE$P
```

```
##                        ID         cond          SID self_efficacy_2
## ID                     NA 9.064112e-01 1.377847e-01     0.701373056
## cond            0.9064112           NA 1.433820e-06     0.008753467
## SID             0.1377847 1.433820e-06           NA     0.669209529
## self_efficacy_2 0.7013731 8.753467e-03 6.692095e-01              NA
```

```
cor_SIDSE$P[3,4]
```

```
## [1] 0.6692095
```

```
cor_SIDSE$r
```

```
##                          ID        cond         SID self_efficacy_2
## ID               1.00000000  0.01550298 -0.19385058     -0.05053690
## cond             0.01550298  1.00000000 -0.57647393      0.33562075
## SID             -0.19385058 -0.57647393  1.00000000     -0.05629603
## self_efficacy_2 -0.05053690  0.33562075 -0.05629603      1.00000000
```

```
cor_SIDSE$r[3,4]
```

```
## [1] -0.05629603
```

SUD

- Mean and Standard deviation of high self-identification condition and low self-identification condition

```
VE_SUD<- VirtualExperience[1:60,c(1,3,15,16,39,8,17,42)]
SUD_High<-subset(VE_SUD,cond==0)
SUD_Low<-subset(VE_SUD,cond==1)
describeBy(VE_SUD,VE_SUD$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##                   vars  n  mean    sd median trimmed   mad min max range
## ID1                  1 30 31.27 18.99   29.0   30.92 24.46   2  63    61
## cond                 2 30  0.00  0.00    0.0    0.00  0.00   0   0     0
## SUD_neutral          3 30  1.03  1.19    1.0    0.83  1.48   0   4     4
## SUD_VE               4 30  2.47  1.48    2.0    2.38  1.48   0   7     7
## SUD_lec              5 30  5.03  2.33    5.5    5.12  3.71   1   8     7
## self_efficacy_1      6 30  6.87  2.47    7.0    7.08  2.97   1  10     9
## self_efficacy_2      7 30  5.10  2.66    5.0    4.92  2.97   2  10     8
## self_efficacy_3      8 30  4.43  2.66    4.0    4.25  2.97   1  10     9
##                    skew kurtosis   se
## ID1                0.12    -1.43 3.47
## cond                NaN      NaN 0.00
## SUD_neutral        1.01     0.17 0.22
## SUD_VE             0.80     1.06 0.27
## SUD_lec           -0.18    -1.41 0.42
## self_efficacy_1 -0.68    -0.47 0.45
## self_efficacy_2  0.42    -1.18 0.49
## self_efficacy_3  0.46    -0.95 0.49
## --------------------------------------------------------
## group: 1
##                   vars  n  mean    sd median trimmed   mad min max range
## ID1                  1 30 31.83 18.18   33.0   32.04 20.76   1  61    60
## cond                 2 30  1.00  0.00    1.0    1.00  0.00   1   1     0
## SUD_neutral          3 30  0.53  0.68    0.0    0.42  0.00   0   2     2
## SUD_VE               4 30  1.77  1.43    1.5    1.67  2.22   0   6     6
## SUD_lec              5 30  4.13  2.33    4.0    4.04  2.97   1   9     8
## self_efficacy_1      6 30  7.07  2.50    7.5    7.33  2.22   2  10     8
## self_efficacy_2      7 30  6.80  2.17    7.0    7.00  1.48   1  10     9
## self_efficacy_3      8 30  5.90  2.32    6.0    6.00  1.48   1  10     9
##                    skew kurtosis   se
## ID1               -0.09    -1.24 3.32
## cond                NaN      NaN 0.00
## SUD_neutral        0.83    -0.57 0.12
## SUD_VE             0.74     0.39 0.26
## SUD_lec            0.19    -1.15 0.43
## self_efficacy_1 -0.74    -0.60 0.46
## self_efficacy_2 -0.79     0.01 0.40
## self_efficacy_3 -0.25    -0.31 0.42
```

```r
Ttest_SUD_neutral<-testbetween(SUD_High$SUD_neutral,SUD_Low$SUD_neutral)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test cond
##
##  Two Sample t-test
##
## data:  x by g
## t = 1.9989, df = 58, p-value = 0.05031
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.0006977286  1.0006977286
## sample estimates:
## mean in group 1 mean in group 2
##       1.0333333       0.5333333
```

```r
p.adjust(Ttest_SUD_neutral$p.value, method = "bonferroni", n=3)
```

```
## [1] 0.150929
```

```r
Ttest_SUD_VE<-testbetween(SUD_High$SUD_VE,SUD_Low$SUD_VE)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test cond
##
##  Two Sample t-test
##
## data:  x by g
## t = 1.8629, df = 58, p-value = 0.06754
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.05214448  1.45214448
## sample estimates:
## mean in group 1 mean in group 2
##        2.466667        1.766667
```

```r
p.adjust(Ttest_SUD_VE$p.value, method = "bonferroni", n=3)
```

```
## [1] 0.2026102
```

```r
Ttest_SUD_lec<-testbetween(SUD_High$SUD_lec,SUD_Low$SUD_lec)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test cond
##
##  Two Sample t-test
##
## data:  x by g
## t = 1.4971, df = 58, p-value = 0.1398
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.3033929  2.1033929
## sample estimates:
## mean in group 1 mean in group 2
##        5.033333        4.133333
```

```r
p.adjust(Ttest_SUD_lec$p.value, method = "bonferroni", n=3)
```

```
## [1] 0.4194042
```

*Correlation with self-efficacy measured at the same phase, Pearson's r

```r
VE_SE_All<-VirtualExperience[1:60,c(1,3,8,17,42)]
colnames(VE_SE_All) <- c("ID","cond","SE1","SE2","SE3")
VE_SE_All_Formalized<-melt(VE_SE_All, id=(c("ID","cond")))
#colnames(VE_SE_All_Formalized) <- c("ID","cond","Sesiion","SE")

VE_SUD_All<-VirtualExperience[1:60,c(1,3,15,16,39)]
colnames(VE_SUD_All) <- c("ID","cond","SUD1","SUD2","SUD3")
VE_SUD_All_Formalized<-melt(VE_SUD_All, id=(c("ID","cond")))
colnames(VE_SUD_All_Formalized) <- c("ID","cond","Sesiion","SUD")
describeBy(VE_SUD_All_Formalized,VE_SUD_All_Formalized$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##          vars  n  mean    sd median trimmed   mad min max range skew
## ID          1 90 31.27 18.77     29   30.92 24.46   2  63    61 0.12
## cond        2 90  0.00  0.00      0    0.00  0.00   0   0     0  NaN
## Sesiion*    3 90  2.00  0.82      2    2.00  1.48   1   3     2 0.00
## SUD         4 90  2.84  2.39      2    2.60  1.48   0   8     8 0.76
##          kurtosis   se
## ID          -1.36 1.98
## cond          NaN 0.00
## Sesiion*    -1.53 0.09
## SUD         -0.45 0.25
## --------------------------------------------------------
## group: 1
##          vars  n  mean    sd median trimmed   mad min max range  skew
## ID          1 90 31.83 17.97     33   32.04 20.76   1  61    60 -0.09
## cond        2 90  1.00  0.00      1    1.00  0.00   1   1     0   NaN
## Sesiion*    3 90  2.00  0.82      2    2.00  1.48   1   3     2  0.00
## SUD         4 90  2.14  2.20      1    1.82  1.48   0   9     9  1.11
##          kurtosis   se
## ID          -1.16 1.89
## cond          NaN 0.00
## Sesiion*    -1.53 0.09
## SUD          0.40 0.23
```

```r
VE_SUD_All_Formalized_High<-subset(VE_SUD_All_Formalized,cond==0)
VE_SUD_All_Formalized_Low<-subset(VE_SUD_All_Formalized,cond==1)
testbetween(VE_SUD_All_Formalized_High$SUD,VE_SUD_All_Formalized_Low$SUD)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = 2.0445, df = 178, p-value = 0.04237
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
```

```
##  0.02436116 1.37563884
## sample estimates:
## mean in group 1 mean in group 2
##        2.844444        2.144444
```

```
VE_SUD_SE<-cbind(VE_SUD_All_Formalized,VE_SE_All_Formalized)
VE_SUD_SEReordered<- VE_SUD_SE[order(VE_SUD_SE$ID),]
colnames(VE_SUD_SEReordered) <- c("ID","cond","SUDSession","SUD","ID","cond","SESession","SE")

VE_SUD_SEReordered$ID<-factor(VE_SUD_SEReordered$ID)
VE_SUD_SEReordered$cond<-factor(VE_SUD_SEReordered$cond)
rmcorr(participant = VE_SUD_SEReordered$ID, measure1 = VE_SUD_SEReordered$SUD, measure2 = VE_SUD_SEReor
```

```
##
## Repeated measures correlation
##
## r
## -0.4696557
##
## degrees of freedom
## 119
##
## p-value
## 5.496891e-08
##
## 95% confidence interval
## -0.5990131 -0.316408
```

HR

- Mean and Standard deviation of high self-identification condition and low self-identification condition

```
VE_HR<- VirtualExperience[1:60,c(1,3,58:60)]
HR_High<-subset(VE_HR,cond==0)
HR_Low<-subset(VE_HR,cond==1)
describeBy(VE_HR,VE_HR$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##            vars  n  mean    sd median trimmed   mad   min    max range
## ID1           1 30 31.27 18.99  29.00   30.92 24.46  2.00  63.00 61.00
## cond          2 30  0.00  0.00   0.00    0.00  0.00  0.00   0.00  0.00
## HR_neutral    3 29 75.63 14.28  72.34   74.85 12.74 52.76 110.06 57.29
## HR_VE         4 29 84.57 14.65  83.28   84.35 17.25 57.34 113.04 55.70
## HR_lecture    5 28 87.85 12.80  83.72   86.95  9.72 67.14 116.16 49.03
##            skew kurtosis   se
## ID1        0.12    -1.43 3.47
## cond        NaN      NaN 0.00
## HR_neutral 0.60    -0.32 2.65
## HR_VE      0.20    -1.09 2.72
## HR_lecture 0.75    -0.54 2.42
## --------------------------------------------------------
## group: 1
##            vars  n  mean    sd median trimmed   mad   min    max range
## ID1           1 30 31.83 18.18  33.00   32.04 20.76  1.00  61.00 60.00
```

```
## cond          2 30  1.00  0.00  1.00   1.00  0.00  1.00   1.00  0.00
## HR_neutral    3 30 74.24 10.14 73.34  74.20 10.27 54.16  92.56 38.40
## HR_VE         4 29 87.39 10.54 87.96  87.50  8.87 65.62 108.21 42.59
## HR_lecture    5 28 86.46 10.54 85.79  86.52  7.39 64.15 108.37 44.22
##             skew kurtosis   se
## ID1        -0.09    -1.24 3.32
## cond         NaN      NaN 0.00
## HR_neutral  0.10    -0.91 1.85
## HR_VE      -0.16    -0.54 1.96
## HR_lecture -0.01     0.00 1.99
```

```r
Ttest_HR_neutral<-testbetween(HR_High$HR_neutral,HR_Low$HR_neutral)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = 0.43324, df = 57, p-value = 0.6665
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -5.046552  7.833144
## sample estimates:
## mean in group 1 mean in group 2
##        75.63431        74.24101
```

```r
p.adjust(Ttest_HR_neutral$p.value, method = "bonferroni", n=3)
```

```
## [1] 1
```

```r
Ttest_HR_VE<-testbetween(HR_High$HR_VE,HR_Low$HR_VE)
```

```
##
## Levene's test shows sign difference between the group variance, therefore  Welch Two Sample t-test co
##
##  Welch Two Sample t-test
##
## data:  x by g
## t = -0.83921, df = 50.852, p-value = 0.4053
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -9.539422  3.915404
## sample estimates:
## mean in group 1 mean in group 2
##        84.57497        87.38698
```

```r
p.adjust(Ttest_HR_VE$p.value, method = "bonferroni", n=3)
```

```
## [1] 1
```

```r
Ttest_HR_lec<-testbetween(HR_High$HR_lec,HR_Low$HR_lec)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
```

```
## data:  x by g
## t = 0.44113, df = 54, p-value = 0.6609
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -4.898417  7.662082
## sample estimates:
## mean in group 1 mean in group 2
##        87.84554        86.46371
```

```
p.adjust(Ttest_HR_lec$p.value, method = "bonferroni", n=3)
```

```
## [1] 1
```

*Correlation with self-efficacy measured at the same phase, Pearson's r

```
colnames(VE_HR) <- c("ID","cond","HR1","HR2","HR3")
VE_HR_All_Formalized<-melt(VE_HR, id=(c("ID","cond")))
colnames(VE_HR_All_Formalized) <- c("ID","cond","Sesiion","HR")
describeBy(VE_HR_All_Formalized,VE_HR_All_Formalized$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##          vars  n  mean    sd median trimmed   mad   min    max range skew
## ID          1 90 31.27 18.77  29.00   30.92 24.46  2.00  63.00  61.0 0.12
## cond        2 90  0.00  0.00   0.00    0.00  0.00  0.00   0.00   0.0  NaN
## Sesiion*    3 90  2.00  0.82   2.00    2.00  1.48  1.00   3.00   2.0 0.00
## HR          4 86 82.62 14.72  80.89   82.21 13.76 52.76 116.16  63.4 0.34
##          kurtosis   se
## ID          -1.36 1.98
## cond          NaN 0.00
## Sesiion*    -1.53 0.09
## HR          -0.55 1.59
## --------------------------------------------------------
## group: 1
##          vars  n  mean    sd median trimmed   mad   min    max range  skew
## ID          1 90 31.83 17.97  33.00   32.04 20.76  1.00  61.00 60.00 -0.09
## cond        2 90  1.00  0.00   1.00    1.00  0.00  1.00   1.00  0.00   NaN
## Sesiion*    3 90  2.00  0.82   2.00    2.00  1.48  1.00   3.00  2.00  0.00
## HR          4 87 82.56 11.94  84.29   82.47 11.23 54.16 108.37 54.21 -0.06
##          kurtosis   se
## ID          -1.16 1.89
## cond          NaN 0.00
## Sesiion*    -1.53 0.09
## HR          -0.47 1.28
```

```
VE_HR_All_Formalized_High<-subset(VE_HR_All_Formalized,cond==0)
VE_HR_All_Formalized_Low<-subset(VE_HR_All_Formalized,cond==1)
testbetween(VE_HR_All_Formalized_High$HR,VE_HR_All_Formalized_Low$HR)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test cond
##
##  Two Sample t-test
##
## data:  x by g
## t = 0.03347, df = 171, p-value = 0.9733
```

```
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -3.953394  4.089774
## sample estimates:
## mean in group 1 mean in group 2
##        82.62494        82.55675
```

```
VE_HR_SE<-cbind(VE_HR_All_Formalized,VE_SE_All_Formalized)
VE_HR_SEreordered<- VE_HR_SE[order(VE_HR_SE$ID),]
colnames(VE_HR_SEreordered) <- c("ID","cond","HRSession","HR","ID","cond","SESession","SE")

VE_HR_SEreordered$ID<-factor(VE_HR_SEreordered$ID)
VE_HR_SEreordered$cond<-factor(VE_HR_SEreordered$cond)
rmcorr(participant = VE_HR_SEreordered$ID, measure1 = VE_HR_SEreordered$HR, measure2 = VE_HR_SEreordered
```

```
##
## Repeated measures correlation
##
## r
## -0.1944432
##
## degrees of freedom
## 112
##
## p-value
## 0.03816898
##
## 95% confidence interval
## -0.3667661 -0.009220124
```

PR

- Mean and Standard deviation of high self-identification condition and low self-identification condition

```
VE_PR_Average<-data.frame(VE_PR$ID,VE_PR$cond,rowMeans(VE_PR[3:5]),rowMeans(VE_PR[6:8]))
colnames(VE_PR_Average) <- c("ID","cond","VE","Post")
PR_High<-subset(VE_PR_Average,cond==0)
PR_Low<-subset(VE_PR_Average,cond==1)
describeBy(VE_PR_Average,VE_PR_Average$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##      vars  n  mean    sd median trimmed   mad  min   max range  skew
## ID      1 30 31.27 18.99   29.0   30.92 24.46 2.00 63.00 61.00  0.12
## cond    2 30  0.00  0.00    0.0    0.00  0.00 0.00  0.00  0.00   NaN
## VE      3 30  4.58  2.10    4.5    4.65  2.22 0.00  8.67  8.67 -0.18
## Post    4 30  6.68  2.04    7.0    6.86  1.98 1.33 10.00  8.67 -0.72
##      kurtosis   se
## ID      -1.43 3.47
## cond      NaN 0.00
## VE      -0.72 0.38
## Post     0.05 0.37
## ------------------------------------------------------------
## group: 1
##      vars  n  mean    sd median trimmed   mad  min max range  skew
```

```
## ID      1 30 31.83 18.18  33.00   32.04 20.76 1.00   61 60.00 -0.09
## cond    2 30  1.00  0.00   1.00    1.00  0.00 1.00    1  0.00   NaN
## VE      3 30  3.83  2.54   4.00    3.74  1.73 0.00   10 10.00  0.10
## Post    4 30  6.37  2.05   6.67    6.47  2.47 1.33   10  8.67 -0.39
##      kurtosis   se
## ID      -1.24 3.32
## cond      NaN 0.00
## VE      -0.35 0.46
## Post    -0.49 0.37
```

```
Ttest_PR_VE<-testbetween(PR_High$VE,PR_Low$VE)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = 1.2362, df = 58, p-value = 0.2214
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.4609596  1.9498485
## sample estimates:
## mean in group 1 mean in group 2
##        4.577778        3.833333
```

```
p.adjust(Ttest_PR_VE$p.value, method = "bonferroni", n=2)
```

```
## [1] 0.4427057
```

```
Ttest_PR_Post<-testbetween(PR_High$Post,PR_Low$Post)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = 0.58831, df = 58, p-value = 0.5586
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.7474377  1.3696600
## sample estimates:
## mean in group 1 mean in group 2
##        6.677778        6.366667
```

```
p.adjust(Ttest_PR_Post$p.value, method = "bonferroni", n=2)
```

```
## [1] 1
```

```
VE_PRFormalized<-melt(VE_PR_Average, id=(c("ID","cond")))
PR_HighFormalized<-subset(VE_PRFormalized,cond==0)
PR_LowFormalized<-subset(VE_PRFormalized,cond==1)
describeBy(VE_PRFormalized,VE_PRFormalized$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
```

```
##           vars  n  mean    sd median trimmed   mad min max range  skew
## ID           1 60 31.27 18.83  29.00   30.92 24.46   2  63    61  0.12
## cond         2 60  0.00  0.00   0.00    0.00  0.00   0   0     0   NaN
## variable*    3 60  1.50  0.50   1.50    1.50  0.74   1   2     1  0.00
## value        4 60  5.63  2.31   5.67    5.73  2.47   0  10    10 -0.35
##           kurtosis   se
## ID           -1.38 2.43
## cond           NaN 0.00
## variable*    -2.03 0.07
## value        -0.61 0.30
## --------------------------------------------------------
## group: 1
##           vars  n  mean    sd median trimmed   mad min max range  skew
## ID           1 60 31.83 18.02   33.0   32.04 20.76   1  61    60 -0.09
## cond         2 60  1.00  0.00    1.0    1.00  0.00   1   1     0   NaN
## variable*    3 60  1.50  0.50    1.5    1.50  0.74   1   2     1  0.00
## value        4 60  5.10  2.62    5.0    5.20  2.47   0  10    10 -0.28
##           kurtosis   se
## ID           -1.18 2.33
## cond           NaN 0.00
## variable*    -2.03 0.07
## value        -0.53 0.34
```

*Correlation with self-efficacy measured at the same phase, Pearson's r

```
VE_PR_Formalized<-melt(VE_PR_Average, id=(c("ID","cond")))
colnames(VE_PR_Formalized) <- c("ID","cond","Session","PR")

VE_SE_VEPost<-VirtualExperience[1:60,c(1,3,17,42)]
colnames(VE_SE_VEPost) <- c("ID","cond","VE","Post")
VE_SE_VEPost_Formalized<-melt(VE_SE_VEPost, id=(c("ID","cond")))

VE_PR_SE<-cbind(VE_PR_Formalized,VE_SE_VEPost_Formalized)
VE_PR_SEReordered<- VE_PR_SE[order(VE_PR_SE$ID),]
colnames(VE_PR_SEReordered) <- c("ID","cond","PRSession","PR","ID","cond","SESession","SE")

VE_PR_SEReordered$ID<-factor(VE_PR_SEReordered$ID)
VE_PR_SEReordered$cond<-factor(VE_PR_SEReordered$cond)
rmcorr(participant = VE_PR_SEReordered$ID, measure1 = VE_PR_SEReordered$PR, measure2 = VE_PR_SEReordere
```

```
##
## Repeated measures correlation
##
## r
## -0.1808057
##
## degrees of freedom
## 59
##
## p-value
## 0.1631829
##
## 95% confidence interval
## -0.4175547 0.07893119
```

Self-efficacy

- Mean and Standard deviation of high self-identification condition and low self-identification condition

```
VE_SE<- VirtualExperience[1:60,c(1,3,8,17,42,48)]
SE_High<-subset(VE_SE,cond==0)
SE_Low<-subset(VE_SE,cond==1)
describeBy(VE_SE,VE_SE$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##                 vars  n  mean    sd median trimmed   mad min max range
## ID1                1 30 31.27 18.99   29.0   30.92 24.46   2  63    61
## cond               2 30  0.00  0.00    0.0    0.00  0.00   0   0     0
## self_efficacy_1    3 30  6.87  2.47    7.0    7.08  2.97   1  10     9
## self_efficacy_2    4 30  5.10  2.66    5.0    4.92  2.97   2  10     8
## self_efficacy_3    5 30  4.43  2.66    4.0    4.25  2.97   1  10     9
## self_efficacy_4    6 30  6.37  1.61    6.5    6.46  2.22   2   9     7
##                  skew kurtosis   se
## ID1              0.12    -1.43 3.47
## cond              NaN      NaN 0.00
## self_efficacy_1 -0.68    -0.47 0.45
## self_efficacy_2  0.42    -1.18 0.49
## self_efficacy_3  0.46    -0.95 0.49
## self_efficacy_4 -0.49    -0.07 0.29
## --------------------------------------------------------
## group: 1
##                 vars  n  mean    sd median trimmed   mad min max range
## ID1                1 30 31.83 18.18   33.0   32.04 20.76   1  61    60
## cond               2 30  1.00  0.00    1.0    1.00  0.00   1   1     0
## self_efficacy_1    3 30  7.07  2.50    7.5    7.33  2.22   2  10     8
## self_efficacy_2    4 30  6.80  2.17    7.0    7.00  1.48   1  10     9
## self_efficacy_3    5 30  5.90  2.32    6.0    6.00  1.48   1  10     9
## self_efficacy_4    6 30  6.77  1.72    7.0    6.92  1.48   2  10     8
##                  skew kurtosis   se
## ID1             -0.09    -1.24 3.32
## cond              NaN      NaN 0.00
## self_efficacy_1 -0.74    -0.60 0.46
## self_efficacy_2 -0.79     0.01 0.40
## self_efficacy_3 -0.25    -0.31 0.42
## self_efficacy_4 -0.84     0.74 0.31
```

```
SE_VEPost_High<-subset(VE_SE_VEPost,cond==0)
SE_VEPost_Low<-subset(VE_SE_VEPost,cond==1)

VE_SEFormalized<-melt(VE_SE, id=(c("ID1","cond")))
colnames(VE_SEFormalized) <- c("ID","cond","Session","SE")
SE_HighFormalized<-subset(VE_SEFormalized,cond==0)
SE_LowFormalized<-subset(VE_SEFormalized,cond==1)
describeBy(VE_SEFormalized,VE_SEFormalized$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##         vars   n  mean    sd median trimmed   mad min max range  skew
```

```
## ID          1 120 31.27 18.75   29.0   30.92 24.46   2  63    61  0.12
## cond        2 120  0.00  0.00    0.0    0.00  0.00   0   0     0   NaN
## Session*    3 120  2.50  1.12    2.5    2.50  1.48   1   4     3  0.00
## SE          4 120  5.69  2.55    6.0    5.73  2.97   1  10     9 -0.12
##          kurtosis   se
## ID          -1.35 1.71
## cond          NaN 0.00
## Session*    -1.39 0.10
## SE          -1.05 0.23
## --------------------------------------------------------
## group: 1
##          vars   n  mean    sd median trimmed   mad min max range  skew
## ID          1 120 31.83 17.94   33.0   32.04 20.76   1  61    60 -0.09
## cond        2 120  1.00  0.00    1.0    1.00  0.00   1   1     0   NaN
## Session*    3 120  2.50  1.12    2.5    2.50  1.48   1   4     3  0.00
## SE          4 120  6.63  2.22    7.0    6.80  1.48   1  10     9 -0.63
##          kurtosis   se
## ID          -1.15 1.64
## cond          NaN 0.00
## Session*    -1.39 0.10
## SE          -0.10 0.20
```

```r
Ttest_SE_S1<-testbetween(SE_High$self_efficacy_1,SE_Low$self_efficacy_1)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = -0.3112, df = 58, p-value = 0.7568
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.486447  1.086447
## sample estimates:
## mean in group 1 mean in group 2
##        6.866667        7.066667
```

```r
p.adjust(Ttest_SE_S1$p.value, method = "bonferroni", n=3)
```

```
## [1] 1
```

```r
Ttest_SE_S2<-testbetween(SE_High$self_efficacy_2,SE_Low$self_efficacy_2)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = -2.7134, df = 58, p-value = 0.008753
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.9541183 -0.4458817
## sample estimates:
## mean in group 1 mean in group 2
```

```
##             5.1             6.8
```

```
cat("\n## No correction was made for this t-test comparison on the data collected after vicarious exper
```

```
##
## ## No correction was made for this t-test comparison on the data collected after vicarious experience
```

```
cat("\n## However, even with a posthoc correction for all the phases, the result remained significant a
```

```
##
## ## However, even with a posthoc correction for all the phases, the result remained significant as sh
```

```
p.adjust(Ttest_SE_S2$p.value, method = "bonferroni", n=3)
```

```
## [1] 0.0262604
```

```
Ttest_SE_S3<-testbetween(SE_High$self_efficacy_3,SE_Low$self_efficacy_3)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test cond
##
##  Two Sample t-test
##
## data:  x by g
## t = -2.2735, df = 58, p-value = 0.02671
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.7579972 -0.1753361
## sample estimates:
## mean in group 1 mean in group 2
##        4.433333        5.900000
```

```
p.adjust(Ttest_SE_S3$p.value, method = "bonferroni", n=3)
```

```
## [1] 0.08014381
```

```
Ttest_SE_S4<-testbetween(SE_High$self_efficacy_4,SE_Low$self_efficacy_4)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test cond
##
##  Two Sample t-test
##
## data:  x by g
## t = -0.93177, df = 58, p-value = 0.3553
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.2593203  0.4593203
## sample estimates:
## mean in group 1 mean in group 2
##        6.366667        6.766667
```

```
p.adjust(Ttest_SE_S4$p.value, method = "bonferroni", n=3)
```

```
## [1] 1
```

```
testbetween(SE_HighFormalized$SE,SE_LowFormalized$SE)
```

```
##
## Levene's test shows sign difference between the group variance, therefore  Welch Two Sample t-test c
```

```
##
##  Welch Two Sample t-test
##
## data:  x by g
## t = -3.052, df = 233.36, p-value = 0.002537
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.5495569 -0.3337765
## sample estimates:
## mean in group 1 mean in group 2
##        5.691667        6.633333
```

Self-esteem

- Mean and Standard deviation of high self-identification condition and low self-identification condition

```
VE_SEsteem<- VirtualExperience[1:60,c(1,3,10,18)]
SEsteem_High<-subset(VE_SEsteem,cond==0)
SEsteem_Low<-subset(VE_SEsteem,cond==1)
describeBy(VE_SEsteem,VE_SEsteem$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##               vars  n  mean    sd median trimmed   mad min max range  skew
## ID1              1 30 31.27 18.99   29.0   30.92 24.46   2  63    61  0.12
## cond             2 30  0.00  0.00    0.0    0.00  0.00   0   0     0   NaN
## self_esteem      3 30 20.87  5.49   20.0   21.21  5.93   6  30    24 -0.56
## self_esteem_2    4 30 21.03  5.89   21.5   21.54  6.67   6  30    24 -0.69
##               kurtosis   se
## ID1              -1.43 3.47
## cond               NaN 0.00
## self_esteem       0.04 1.00
## self_esteem_2    -0.08 1.08
## --------------------------------------------------------
## group: 1
##               vars  n  mean    sd median trimmed   mad min max range  skew
## ID1              1 30 31.83 18.18   33.0   32.04 20.76   1  61    60 -0.09
## cond             2 30  1.00  0.00    1.0    1.00  0.00   1   1     0   NaN
## self_esteem      3 30 22.50  4.08   23.0   22.67  4.45  15  29    14 -0.34
## self_esteem_2    4 30 22.90  4.66   22.5   23.04  5.19  10  30    20 -0.46
##               kurtosis   se
## ID1              -1.24 3.32
## cond               NaN 0.00
## self_esteem      -1.11 0.75
## self_esteem_2    -0.08 0.85
```

```
VE_SEsteemFormalized<-melt(VE_SEsteem, id=(c("ID1","cond")))
colnames(VE_SEsteemFormalized) <- c("ID","cond","Session","SEsteem")
SEsteem_HighFormalized<-subset(VE_SEsteemFormalized,cond==0)
SEsteem_LowFormalized<-subset(VE_SEsteemFormalized,cond==1)
describeBy(VE_SEsteemFormalized,VE_SEsteemFormalized$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
```

```
##          vars  n  mean    sd median trimmed   mad min max range  skew
## ID          1 60 31.27 18.83   29.0   30.92 24.46   2  63    61  0.12
## cond        2 60  0.00  0.00    0.0    0.00  0.00   0   0     0   NaN
## Session*    3 60  1.50  0.50    1.5    1.50  0.74   1   2     1  0.00
## SEsteem     4 60 20.95  5.65   21.0   21.40  5.93   6  30    24 -0.64
##          kurtosis   se
## ID          -1.38 2.43
## cond          NaN 0.00
## Session*    -2.03 0.07
## SEsteem      0.08 0.73
## --------------------------------------------------------
## group: 1
##          vars  n  mean    sd median trimmed   mad min max range  skew
## ID          1 60 31.83 18.02   33.0   32.04 20.76   1  61    60 -0.09
## cond        2 60  1.00  0.00    1.0    1.00  0.00   1   1     0   NaN
## Session*    3 60  1.50  0.50    1.5    1.50  0.74   1   2     1  0.00
## SEsteem     4 60 22.70  4.35   22.5   22.88  5.19  10  30    20 -0.41
##          kurtosis   se
## ID          -1.18 2.33
## cond          NaN 0.00
## Session*    -2.03 0.07
## SEsteem     -0.35 0.56
```

```
Ttest_SEsteem_S1<-testbetween(SEsteem_High$self_esteem,SEsteem_Low$self_esteem)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = -1.3078, df = 58, p-value = 0.1961
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -4.1332627  0.8665961
## sample estimates:
## mean in group 1 mean in group 2
##        20.86667        22.50000
```

```
p.adjust(Ttest_SEsteem_S1$p.value, method = "bonferroni", n=2)
```

```
## [1] 0.3921851
```

```
Ttest_SEsteem_S2<-testbetween(SEsteem_High$self_esteem_2,SEsteem_Low$self_esteem_2)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = -1.3605, df = 58, p-value = 0.1789
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -4.6130168  0.8796835
## sample estimates:
```

```
## mean in group 1 mean in group 2
##        21.03333        22.90000
```

```r
p.adjust(Ttest_SEsteem_S2$p.value, method = "bonferroni", n=2)
```

```
## [1] 0.3578417
```

```r
testbetween(SEsteem_HighFormalized$SEsteem,SEsteem_LowFormalized$SEsteem)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test cond
##
##  Two Sample t-test
##
## data:  x by g
## t = -1.9018, df = 118, p-value = 0.05964
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -3.57221343  0.07221343
## sample estimates:
## mean in group 1 mean in group 2
##           20.95           22.70
```

*Correlation with self-efficacy measured at the same phase, Pearson's r

```r
VE_SE_PreVE<-VirtualExperience[1:60,c(1,3,8,17)]
colnames(VE_SE_PreVE) <- c("ID","cond","Pre","VE")
VE_SE_PreVE_Formalized<-melt(VE_SE_PreVE, id=(c("ID","cond")))

colnames(VE_SEsteem) <- c("ID","cond","Pre","VE")
VE_SEsteem_Formalized<-melt(VE_SEsteem, id=(c("ID","cond")))

VE_SEsteem_SE<-cbind(VE_SEsteem_Formalized,VE_SE_PreVE_Formalized)
VE_SEsteem_SEReordered<- VE_SEsteem_SE[order(VE_SEsteem_SE$ID),]
colnames(VE_SEsteem_SEReordered) <- c("ID","cond","SEsteemSession","SEsteem","ID","cond","SESession","SE

VE_SEsteem_SEReordered$ID<-factor(VE_SEsteem_SEReordered$ID)
VE_SEsteem_SEReordered$cond<-factor(VE_SEsteem_SEReordered$cond)
rmcorr(participant = VE_SEsteem_SEReordered$ID, measure1 = VE_SEsteem_SEReordered$SEsteem, measure2 = V
```

```
##
## Repeated measures correlation
##
## r
## 0.1814858
##
## degrees of freedom
## 59
##
## p-value
## 0.1615792
##
## 95% confidence interval
## -0.07823239 0.4181351
```

Presentation Performance

- Mean and Standard deviation of high self-identification condition and low self-identification condition

```
VE_PP_Average<-data.frame(VE_PP$ID,VE_PP$cond,rowMeans(VE_PP[3:4]),VE_SE$self_efficacy_3)
colnames(VE_PP_Average) <- c("ID","cond","PP","SEVE")

PP_High<-subset(VE_PP_Average,cond==0)
PP_Low<-subset(VE_PP_Average,cond==1)
describeBy(VE_PP_Average,VE_PP_Average$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##      vars  n  mean    sd median trimmed   mad min max range skew kurtosis
## ID      1 30 31.27 18.99   29.0   30.92 24.46   2  63    61 0.12    -1.43
## cond    2 30  0.00  0.00    0.0    0.00  0.00   0   0     0  NaN      NaN
## PP      3 30  4.38  2.41    4.5    4.27  2.22   1  10     9 0.35    -0.73
## SEVE    4 30  4.43  2.66    4.0    4.25  2.97   1  10     9 0.46    -0.95
##        se
## ID   3.47
## cond 0.00
## PP   0.44
## SEVE 0.49
## --------------------------------------------------------
## group: 1
##      vars  n  mean    sd median trimmed   mad min max range  skew kurtosis
## ID      1 30 31.83 18.18  33.00   32.04 20.76   1  61    60 -0.09    -1.24
## cond    2 30  1.00  0.00   1.00    1.00  0.00   1   1     0   NaN      NaN
## PP      3 30  5.07  2.01   5.25    5.15  2.22   0   9     9 -0.40    -0.08
## SEVE    4 30  5.90  2.32   6.00    6.00  1.48   1  10     9 -0.25    -0.31
##        se
## ID   3.32
## cond 0.00
## PP   0.37
## SEVE 0.42
```

```
testbetween(PP_High$PP,PP_Low$PP)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = -1.1924, df = 58, p-value = 0.238
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.8304421  0.4637755
## sample estimates:
## mean in group 1 mean in group 2
##        4.383333        5.066667
```

*Correlation with self-efficacy measured at the same phase, Pearson's r

```
cor_PPSE <- rcorr(as.matrix(VE_PP_Average), type="pearson")
cor_PPSE$P
```

```
##              ID      cond           PP         SEVE
## ID           NA 0.9064112 5.275638e-01 7.635033e-01
## cond 0.9064112        NA 2.379507e-01 2.671460e-02
## PP   0.5275638 0.2379507           NA 2.601874e-11
## SEVE 0.7635033 0.0267146 2.601874e-11           NA
```

```
cor_PPSE$P[3,4]
```

```
## [1] 2.601874e-11
```

```
cor_PPSE$r
```

```
##                ID       cond          PP       SEVE
## ID    1.00000000 0.01550298 -0.08316387 0.03966295
## cond  0.01550298 1.00000000  0.15468835 0.28605221
## PP   -0.08316387 0.15468835  1.00000000 0.73363964
## SEVE  0.03966295 0.28605221  0.73363964 1.00000000
```

```
cor_PPSE$r[3,4]
```

```
## [1] 0.7336396
```

Speech length

- Mean and Standard deviation of high self-identification condition and low self-identification condition

```
Length_High<-subset(Length,cond==0)
Length_Low<-subset(Length,cond==1)
describeBy(Length,Length$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##        vars  n   mean    sd median trimmed   mad min max range skew
## cond      1 30   0.00  0.00      0    0.00  0.00   0   0     0  NaN
## ID        2 30  31.27 18.99     29   30.92 24.46   2  63    61 0.12
## Time      3 30   6.23  1.57      6    6.17  1.48   3  10     7 0.35
## Length    4 30 374.00 94.12    360  370.00 88.96 180 600   420 0.35
##        kurtosis    se
## cond        NaN  0.00
## ID        -1.43  3.47
## Time      -0.16  0.29
## Length    -0.16 17.18
## --------------------------------------------------------
## group: 1
##        vars  n   mean    sd median trimmed   mad min max range  skew
## cond      1 30   1.00  0.00      1    1.00  0.00   1   1     0   NaN
## ID        2 30  31.83 18.18     33   32.04 20.76   1  61    60 -0.09
## Time      3 30   5.80  1.19      6    5.75  1.48   3   9     6  0.37
## Length    4 30 348.00 71.17    360  345.00 88.96 180 540   360  0.37
##        kurtosis    se
## cond        NaN  0.00
## ID        -1.24  3.32
## Time       0.62  0.22
## Length     0.62 12.99
```

```
testbetween(Length_High$Length,Length_Low$Length)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = 1.2068, df = 58, p-value = 0.2324
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -17.12472  69.12472
## sample estimates:
## mean in group 1 mean in group 2
##             374             348
```

*Correlation with self-efficacy measured at the same phase, Pearson's r

```
LengthSE<-cbind(Length[1:60,c(1,2,4)],VirtualExperience[1:60,c(42)])
colnames(LengthSE) <- c("cond","ID","Length","SE")
cor_LengthSE <- rcorr(as.matrix(LengthSE), type="pearson")
cor_LengthSE$P
```

```
##             cond        ID    Length        SE
## cond          NA 0.9064112 0.2323928 0.0267146
## ID     0.9064112        NA 0.7740462 0.7635033
## Length 0.2323928 0.7740462        NA 0.3313345
## SE     0.0267146 0.7635033 0.3313345        NA
```

```
cor_LengthSE$P[3,4]
```

```
## [1] 0.3313345
```

```
cor_LengthSE$r
```

```
##               cond         ID      Length          SE
## cond    1.00000000 0.01550298 -0.15651295  0.28605221
## ID      0.01550298 1.00000000  0.03784552  0.03966295
## Length -0.15651295 0.03784552  1.00000000 -0.12758134
## SE      0.28605221 0.03966295 -0.12758134  1.00000000
```

```
cor_LengthSE$r[3,4]
```

```
## [1] -0.1275813
```

Section 5.1 Manipulation check

- Comparision between the conditions on people's self-identification

```
testbetween(SID_High$SID,SID_Low$SID)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = 5.3729, df = 58, p-value = 1.434e-06
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
```

```
##  1.452083 3.176489
## sample estimates:
## mean in group 1 mean in group 2
##        6.623810        4.309524
```

- Correlation between ITQ and Self-identification

```r
VE_SID_ITQ<-data.frame(VE_SID_Average,VE_IQT$ITQ)
cor_SIDITQ_High <- rcorr(as.matrix(VE_SID_ITQ[VE_SID_ITQ$cond=="0",]), type="pearson")
cor_SIDITQ_Low <- rcorr(as.matrix(VE_SID_ITQ[VE_SID_ITQ$cond=="1",]), type="pearson")
cat("\n## The corrlation check for High self-identification condition\n")
```

```
##
## ## The corrlation check for High self-identification condition
```

```r
cor_SIDITQ_High$P
```

```
##                    ID cond       SID VE_IQT.ITQ
## ID                 NA  NaN 0.8502806  0.7705434
## cond              NaN   NA       NaN        NaN
## SID         0.8502806  NaN        NA  0.5788306
## VE_IQT.ITQ 0.7705434  NaN 0.5788306         NA
```

```r
cor_SIDITQ_High$P[3,4]
```

```
## [1] 0.5788306
```

```r
cor_SIDITQ_High$r
```

```
##                     ID cond         SID  VE_IQT.ITQ
## ID          1.00000000  NaN -0.03598053 -0.05557048
## cond               NaN    1         NaN         NaN
## SID        -0.03598053  NaN  1.00000000  0.10554682
## VE_IQT.ITQ -0.05557048  NaN  0.10554682  1.00000000
```

```r
cor_SIDITQ_High$r[3,4]
```

```
## [1] 0.1055468
```

```r
cat("\n## The corrlation check for Low self-identification condition\n")
```

```
##
## ## The corrlation check for Low self-identification condition
```

```r
cor_SIDITQ_Low$P
```

```
##                     ID cond        SID VE_IQT.ITQ
## ID                  NA  NaN 0.03008534  0.4090598
## cond               NaN   NA        NaN        NaN
## SID         0.03008534  NaN         NA  0.1329532
## VE_IQT.ITQ 0.40905985  NaN 0.13295320         NA
```

```r
cor_SIDITQ_Low$P[3,4]
```

```
## [1] 0.1329532
```

```r
cor_SIDITQ_Low$r
```

```
##                     ID cond        SID VE_IQT.ITQ
## ID           1.0000000  NaN -0.3964541 -0.1564400
## cond               NaN    1        NaN        NaN
```

```
## SID        -0.3964541  NaN  1.0000000  0.2807061
## VE_IQT.ITQ -0.1564400  NaN  0.2807061  1.0000000
```

```
cor_SIDITQ_Low$r[3,4]
```

```
## [1] 0.2807061
```

Section 5.2 Overall analyses on self-efficacy across the phases

- using Beyesian within-subject mediation package: bmlm

```
VE_SEFormalized$ID<-factor(VE_SEFormalized$ID)
VE_SEFormalized$cond<-factor(VE_SEFormalized$cond)

Sudmodel_SE<-ezANOVA(data=VE_SEFormalized,dv=SE,wid=ID,within=Session,between=cond,detailed = TRUE, typ
print_ezANOVA(Sudmodel_SE)
```

```
## $ANOVA
##         Effect DFn DFd        SSn      SSd          F            p p<.05
## 1  (Intercept)   1  58 9114.33750 717.2083 737.068367 1.153232e-34     *
## 2         cond   1  58   53.20417 717.2083   4.302574 4.250319e-02     *
## 3      Session   3 174  110.81250 506.0250  12.701201 1.514451e-07     *
## 4 cond:Session   3 174   25.41250 506.0250   2.912751 3.590846e-02     *
##         ges
## 1 0.88167111
## 2 0.04168176
## 3 0.08306499
## 4 0.02035205
##
## $`Mauchly's Test for Sphericity`
##         Effect         W         p p<.05
## 3      Session 0.8180424 0.0441659     *
## 4 cond:Session 0.8180424 0.0441659     *
##
## $`Sphericity Corrections`
##         Effect       GGe        p[GG] p[GG]<.05       HFe        p[HF]
## 3      Session 0.9019771 5.212485e-07         * 0.9503728 2.830432e-07
## 4 cond:Session 0.9019771 4.139137e-02         * 0.9503728 3.858410e-02
##   p[HF]<.05
## 3         *
## 4         *
##
## (Intercept)  : eta  0.9270503
## cond  : eta  0.06905932
## Session  : eta  0.1796462
## cond:Session  : eta  0.04781842
## see ges-value for Generalized Eta-Squared)
##
## Greenhouse-Geisser corrected Dfs.
## Session  numerator  2.705931  , denominator  156.944
## cond:Session  numerator  2.705931  , denominator  156.944
```

- compare cross different measurement moments

```
SE_SE1<-subset(VE_SEFormalized,Session=="self_efficacy_1")
SE_SE2<-subset(VE_SEFormalized,Session=="self_efficacy_2")
```

```r
SE_SE3<-subset(VE_SEFormalized,Session=="self_efficacy_3")

t.test(SE_SE1$SE,SE_SE2$SE,paired = TRUE, alternative = "two.sided")
```

```
##
##  Paired t-test
##
## data:  SE_SE1$SE and SE_SE2$SE
## t = 2.9766, df = 59, p-value = 0.004222
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.3332104 1.7001229
## sample estimates:
## mean of the differences
##                1.016667
```

```r
t.test(SE_SE1$SE,SE_SE3$SE,paired = TRUE, alternative = "two.sided")
```

```
##
##  Paired t-test
##
## data:  SE_SE1$SE and SE_SE3$SE
## t = 5.0734, df = 59, p-value = 4.168e-06
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  1.090068 2.509932
## sample estimates:
## mean of the differences
##                     1.8
```

```r
t.test(SE_SE2$SE,SE_SE3$SE,paired = TRUE, alternative = "two.sided")
```

```
##
##  Paired t-test
##
## data:  SE_SE2$SE and SE_SE3$SE
## t = 2.2901, df = 59, p-value = 0.02561
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.09888485 1.46778181
## sample estimates:
## mean of the differences
##               0.7833333
```

```r
cat("\n## The Bonferroni correction for post-hoc tests\n")
```

```
##
## ## The Bonferroni correction for post-hoc tests
```

```r
pairwise.t.test(VE_SEFormalized$SE,VE_SEFormalized$Session, p.adjust.method="bonferroni",paired=TRUE,po
```

```
##
##  Pairwise comparisons using paired t tests
##
## data:  VE_SEFormalized$SE and VE_SEFormalized$Session
##
##                 self_efficacy_1 self_efficacy_2 self_efficacy_3
```

```
## self_efficacy_2 0.025          -              -
## self_efficacy_3 2.5e-05        0.154          -
## self_efficacy_4 0.972          0.296          7.0e-06
##
## P value adjustment method: bonferroni
```

- compare the two conditions for different measurement moments

```
t.test(SE_High$self_efficacy_1,SE_High$self_efficacy_2,paired = TRUE,alternative = "two.sided")
```

```
##
##  Paired t-test
##
## data:  SE_High$self_efficacy_1 and SE_High$self_efficacy_2
## t = 3.9806, df = 29, p-value = 0.0004217
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.8589598 2.6743735
## sample estimates:
## mean of the differences
##                1.766667
```

```
pairwise.t.test(SE_HighFormalized$SE,SE_HighFormalized$Session,p.adjust.method="bonferroni",paired=TRUE
```

```
##
##  Pairwise comparisons using paired t tests
##
## data:  SE_HighFormalized$SE and SE_HighFormalized$Session
##
##                 self_efficacy_1 self_efficacy_2 self_efficacy_3
## self_efficacy_2 0.0025          -               -
## self_efficacy_3 4.7e-06         0.4887          -
## self_efficacy_4 1.0000          0.0221          3.9e-05
##
## P value adjustment method: bonferroni
```

```
t.test(SE_Low$self_efficacy_1,SE_Low$self_efficacy_2,paired = TRUE,alternative = "two.sided")
```

```
##
##  Paired t-test
##
## data:  SE_Low$self_efficacy_1 and SE_Low$self_efficacy_2
## t = 0.54555, df = 29, p-value = 0.5895
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.7330453  1.2663786
## sample estimates:
## mean of the differences
##               0.2666667
```

```
pairwise.t.test(SE_LowFormalized$SE,SE_LowFormalized$Session,p.adjust.method="bonferroni",paired=TRUE,p
```

```
##
##  Pairwise comparisons using paired t tests
##
## data:  SE_LowFormalized$SE and SE_LowFormalized$Session
##
```

```
##                  self_efficacy_1 self_efficacy_2 self_efficacy_3
## self_efficacy_2 1.00            -               -
## self_efficacy_3 0.32            0.80            -
## self_efficacy_4 1.00            1.00            0.13
##
## P value adjustment method: bonferroni
```

Section 5.3 Moderating effect of self-identification

Section 5.3.1. Performance in the vicarious experience and self-efficacy afterwards

```r
cor_VPSE <- rcorr(as.matrix(VE_VP), type="pearson")
cat("\n## The corrlation check\n")
```

```
##
## ## The corrlation check
```

```r
cor_VPSE$P
```

```
##                       ID1        cond self_efficacy_2 performance_VE
## ID1                    NA 0.906411171     0.701373056     0.11404180
## cond            0.9064112          NA     0.008753467     0.16238896
## self_efficacy_2 0.7013731 0.008753467              NA     0.01110469
## performance_VE  0.1140418 0.162388958     0.011104694             NA
```

```r
cor_VPSE$P[3,4]
```

```
## [1] 0.01110469
```

```r
cor_VPSE$r
```

```
##                         ID1       cond self_efficacy_2 performance_VE
## ID1              1.00000000 0.01550298      -0.0505369     -0.2061523
## cond             0.01550298 1.00000000       0.3356208      0.1826836
## self_efficacy_2 -0.05053690 0.33562075       1.0000000      0.3256927
## performance_VE  -0.20615227 0.18268358       0.3256927      1.0000000
```

```r
cor_VPSE$r[3,4]
```

```
## [1] 0.3256927
```

```r
cor_VPSE_High <- rcorr(as.matrix(VP_High), type="pearson")
cat("\n## The corrlation check\n")
```

```
##
## ## The corrlation check
```

```r
cor_VPSE_High$P
```

```
##                       ID1 cond self_efficacy_2 performance_VE
## ID1                    NA  NaN     0.415448265    0.314757302
## cond                  NaN   NA             NaN            NaN
## self_efficacy_2 0.4154483  NaN              NA    0.001977179
## performance_VE  0.3147573  NaN     0.001977179             NA
```

```r
cor_VPSE_High$P[3,4]
```

```
## [1] 0.001977179
```

```
cor_VPSE_High$r
```

```
##                          ID1 cond self_efficacy_2 performance_VE
## ID1                1.0000000  NaN      -0.1543423     -0.1899329
## cond                     NaN    1             NaN            NaN
## self_efficacy_2 -0.1543423  NaN       1.0000000      0.5419842
## performance_VE  -0.1899329  NaN       0.5419842      1.0000000
```

```
cor_VPSE_High$r[3,4]
```

```
## [1] 0.5419842
```

```
cor_VPSE_Low <- rcorr(as.matrix(VP_Low), type="pearson")
cat("\n## The corrlation check\n")
```

```
##
## ## The corrlation check
```

```
cor_VPSE_Low$P
```

```
##                         ID1 cond self_efficacy_2 performance_VE
## ID1                      NA  NaN       0.7447462      0.1892949
## cond                    NaN   NA             NaN            NaN
## self_efficacy_2 0.7447462  NaN              NA      0.5172046
## performance_VE  0.1892949  NaN       0.5172046             NA
```

```
cor_VPSE_Low$P[3,4]
```

```
## [1] 0.5172046
```

```
cor_VPSE_Low$r
```

```
##                          ID1 cond self_efficacy_2 performance_VE
## ID1               1.00000000  NaN      0.06201957     -0.2464145
## cond                     NaN    1             NaN            NaN
## self_efficacy_2  0.06201957  NaN      1.00000000     -0.1230230
## performance_VE  -0.24641454  NaN     -0.12302304      1.0000000
```

```
cor_VPSE_Low$r[3,4]
```

```
## [1] -0.123023
```

```
CompparedCorrleration_VPSE<-paired.r(cor_VPSE_High$r[3,4], cor_VPSE_Low$r[3,4], yz=NULL, 30, n2=NULL,tw
CompparedCorrleration_VPSE$z
```

```
## [1] 2.684433
```

```
CompparedCorrleration_VPSE$p
```

```
## [1] 0.007265285
```

Section 5.3.2. Self-efficacy

```
cor_SE_VEPost_High <- rcorr(as.matrix(VE_SE_VEPost[VE_SE_VEPost$cond=="0",]), type="pearson")
cor_SE_VEPost_Low <- rcorr(as.matrix(VE_SE_VEPost[VE_SE_VEPost$cond=="1",]), type="pearson")
cor_SE_VEPost_High$P
```

```
##              ID cond           VE         Post
## ID           NA  NaN 4.154483e-01 9.359095e-01
## cond        NaN   NA          NaN          NaN
```

```
## VE   0.4154483  NaN           NA 1.082337e-05
## Post 0.9359095  NaN 1.082337e-05           NA
```

```
cor_SE_VEPost_High$P[3,4]
```

```
## [1] 1.082337e-05
```

```
cor_SE_VEPost_High$r
```

```
##               ID cond          VE        Post
## ID    1.00000000  NaN -0.1543423 -0.01533182
## cond         NaN    1        NaN         NaN
## VE   -0.15434231  NaN  1.0000000  0.71061710
## Post -0.01533182  NaN  0.7106171  1.00000000
```

```
cor_SE_VEPost_High$r[3,4]
```

```
## [1] 0.7106171
```

```
cor_SE_VEPost_Low$P
```

```
##             ID cond        VE      Post
## ID          NA  NaN 0.7447462 0.6021201
## cond       NaN   NA       NaN       NaN
## VE   0.7447462  NaN        NA 0.9828531
## Post 0.6021201  NaN 0.9828531        NA
```

```
cor_SE_VEPost_Low$P[3,4]
```

```
## [1] 0.9828531
```

```
cor_SE_VEPost_Low$r
```

```
##              ID cond           VE         Post
## ID   1.00000000  NaN  0.062019572  0.099164276
## cond        NaN    1          NaN          NaN
## VE   0.06201957  NaN  1.000000000 -0.004098017
## Post 0.09916428  NaN -0.004098017  1.000000000
```

```
cor_SE_VEPost_Low$r[3,4]
```

```
## [1] -0.004098017
```

```
Comppared Corrleration_SE<-paired.r(cor_SE_VEPost_High$r[3,4], cor_SE_VEPost_Low$r[3,4], yz=NULL, 30, n2=
CompparedCorrleration_SE$z
```

```
## [1] 3.279355
```

```
CompparedCorrleration_SE$p
```

```
## [1] 0.001040446
```

Section 5.3.3 Presence response

```
VE_PR_Formalized$ID<-factor(VE_PR_Formalized$ID)
VE_PR_Formalized$cond<-factor(VE_PR_Formalized$cond)

ezANOVA(data=VE_PR_Formalized,dv=PR,wid=ID,within=Session,between=cond)
```

```
## $ANOVA
##         Effect DFn DFd          F            p p<.05         ges
```

```
## 2         cond   1  58  1.3157498 2.560661e-01       0.014734477
## 3      Session   1  58 49.0423632 2.897048e-09     * 0.223687985
## 4 cond:Session   1  58  0.4289716 5.150833e-01       0.002514029
```

```r
t.test(PR_High$VE,PR_High$Post,paired = TRUE)
```

```
##
##  Paired t-test
##
## data:  PR_High$VE and PR_High$Post
## t = -7.0421, df = 29, p-value = 9.571e-08
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.709901 -1.490099
## sample estimates:
## mean of the differences
##                    -2.1
```

```r
#t.test(PR ~ Session, data = VE_PR_Formalized[VE_PR_Formalized$cond==1,], paired = TRUE)
t.test(PR_Low$VE,PR_Low$Post,paired = TRUE)
```

```
##
##  Paired t-test
##
## data:  PR_Low$VE and PR_Low$Post
## t = -4.2894, df = 29, p-value = 0.0001814
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -3.741253 -1.325414
## sample estimates:
## mean of the differences
##               -2.533333
```

```r
cor_PR_VEPost_High <- rcorr(as.matrix(VE_PR_Average[VE_PR_Average$cond=="0",]), type="pearson")
cor_PR_VEPost_Low <- rcorr(as.matrix(VE_PR_Average[VE_PR_Average$cond=="1",]), type="pearson")

cor_PR_VEPost_High$P
```

```
##             ID cond           VE         Post
## ID          NA  NaN 6.563978e-01 7.889526e-01
## cond       NaN   NA          NaN          NaN
## VE   0.6563978  NaN           NA 2.475654e-05
## Post 0.7889526  NaN 2.475654e-05           NA
```

```r
cor_PR_VEPost_High$P[3,4]
```

```
## [1] 2.475654e-05
```

```r
cor_PR_VEPost_High$r
```

```
##               ID cond          VE         Post
## ID    1.00000000  NaN -0.08467773 -0.05100532
## cond         NaN    1         NaN         NaN
## VE   -0.08467773  NaN  1.00000000  0.68979881
## Post -0.05100532  NaN  0.68979881  1.00000000
```

```r
cor_PR_VEPost_High$r[3,4]
```

```
## [1] 0.6897988
```

```
cor_PR_VEPost_Low$P
```

```
##              ID cond         VE      Post
## ID           NA  NaN 0.08273719 0.7236325
## cond        NaN   NA        NaN       NaN
## VE   0.08273719  NaN         NA 0.9156381
## Post 0.72363245  NaN 0.91563810        NA
```

```
cor_PR_VEPost_Low$P[3,4]
```

```
## [1] 0.9156381
```

```
cor_PR_VEPost_Low$r
```

```
##              ID cond          VE        Post
## ID    1.0000000  NaN -0.32195386 -0.06734810
## cond        NaN    1         NaN         NaN
## VE   -0.3219539  NaN  1.00000000  0.02019627
## Post -0.0673481  NaN  0.02019627  1.00000000
```

```
cor_PR_VEPost_Low$r[3,4]
```

```
## [1] 0.02019627
```

```
ComppraredCorrleration_PR<-paired.r(cor_PR_VEPost_High$r[3,4], cor_PR_VEPost_Low$r[3,4], yz=NULL, 30, n2
ComppraredCorrleration_PR$z
```

```
## [1] 3.039962
```

```
ComppraredCorrleration_PR$p
```

```
## [1] 0.002366081
```

Section 5.4 Potential effect of skin colour on self-efficacy

```
VE_SE_LightSkin<- VirtualExperience_LightSkin[1:52,c(1,3,8,17,42,48)]
SE_High_LightSkin<-subset(VE_SE_LightSkin,cond==0)
SE_Low_LightSkin<-subset(VE_SE_LightSkin,cond==1)
describeBy(VE_SE_LightSkin,VE_SE_LightSkin$cond)
```

```
##
##  Descriptive statistics by group
## group: 0
##                 vars  n  mean    sd median trimmed   mad min max range
## ID1                1 26 32.85 18.53   31.5   32.68 25.20   2  63    61
## cond               2 26  0.00  0.00    0.0    0.00  0.00   0   0     0
## self_efficacy_1    3 26  6.81  2.43    7.0    7.00  2.22   1  10     9
## self_efficacy_2    4 26  4.85  2.60    4.5    4.68  3.71   2  10     8
## self_efficacy_3    5 26  4.27  2.43    4.0    4.14  2.22   1   9     8
## self_efficacy_4    6 26  6.35  1.57    6.5    6.45  2.22   2   9     7
##                  skew kurtosis   se
## ID1              0.07    -1.41 3.63
## cond              NaN      NaN 0.00
## self_efficacy_1 -0.71    -0.33 0.48
## self_efficacy_2  0.49    -1.12 0.51
## self_efficacy_3  0.42    -0.90 0.48
## self_efficacy_4 -0.62     0.15 0.31
```

```
## ---------------------------------------------------------
## group: 1
##                  vars  n  mean    sd median trimmed   mad min max range
## ID1                 1 26 34.69 17.63   35.5   35.27 17.05   1  61    60
## cond                2 26  1.00  0.00    1.0    1.00  0.00   1   1     0
## self_efficacy_1     3 26  7.27  2.43    8.0    7.50  1.48   2  10     8
## self_efficacy_2     4 26  6.88  2.16    7.0    7.05  1.48   1  10     9
## self_efficacy_3     5 26  5.69  2.31    6.0    5.77  1.48   1  10     9
## self_efficacy_4     6 26  6.62  1.77    7.0    6.73  1.48   2  10     8
##                   skew kurtosis   se
## ID1              -0.32    -1.03 3.46
## cond               NaN      NaN 0.00
## self_efficacy_1 -0.82    -0.38 0.48
## self_efficacy_2 -0.86     0.27 0.42
## self_efficacy_3 -0.27    -0.38 0.45
## self_efficacy_4 -0.73     0.43 0.35
```

```
testbetween(SE_High_LightSkin$self_efficacy_1,SE_High_LightSkin$self_efficacy_2)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = 2.8069, df = 50, p-value = 0.007111
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.5579261 3.3651508
## sample estimates:
## mean in group 1 mean in group 2
##        6.807692        4.846154
```

```
testbetween(SE_Low_LightSkin$self_efficacy_1,SE_Low_LightSkin$self_efficacy_2)
```

```
##
## Levene's test shows no sign difference between the group variance, therefore  Two Sample t-test condu
##
##  Two Sample t-test
##
## data:  x by g
## t = 0.60377, df = 50, p-value = 0.5487
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.8948838  1.6641145
## sample estimates:
## mean in group 1 mean in group 2
##        7.269231        6.884615
```

R version information

This analysis has been runned with following R version.

R version 3.4.2 (2017-09-28)

Platform: x86_64-apple-darwin15.6.0 (64-bit)

locale: en_US.UTF-8||en_US.UTF-8||en_US.UTF-8||C||en_US.UTF-8||en_US.UTF-8

attached base packages: *stats*, *graphics*, *grDevices*, *utils*, *datasets*, *methods* and *base*

other attached packages: *cocor(v.1.1-3)*, *rmcorr(v.0.3.0)*, *Hmisc(v.4.2-0)*, *Formula(v.1.2-3)*, *afex(v.0.23-0)*, *ez(v.4.4-0)*, *lme4(v.1.1-21)*, *Matrix(v.1.2-17)*, *coin(v.1.3-0)*, *survival(v.2.44-1.1)*, *outliers(v.0.14)*, *lsr(v.0.5)*, *dplyr(v.0.7.4)*, *ggpubr(v.0.2.2)*, *magrittr(v.1.5)*, *pwr(v.1.2-2)*, *psych(v.1.8.12)*, *MASS(v.7.3-51.4)*, *lattice(v.0.20-38)*, *reshape(v.0.8.8)*, *nlme(v.3.1-131)*, *ggplot2(v.3.2.1)*, *car(v.2.1-5)*, *foreign(v.0.8-72)* and *pander(v.0.6.3)*

loaded via a namespace (and not attached): *splines(v.3.4.2)*, *assertthat(v.0.2.1)*, *stats4(v.3.4.2)*, *latticeExtra(v.0.6-28)*, *yaml(v.2.2.0)*, *backports(v.1.1.4)*, *numDeriv(v.2016.8-1.1)*, *pillar(v.1.4.2)*, *quantreg(v.5.34)*, *glue(v.1.3.1)*, *digest(v.0.6.20)*, *checkmate(v.1.9.4)*, *RColorBrewer(v.1.1-2)*, *ggsignif(v.0.6.0)*, *minqa(v.1.2.4)*, *colorspace(v.1.4-1)*, *sandwich(v.2.5-1)*, *htmltools(v.0.3.6)*, *plyr(v.1.8.4)*, *pkgconfig(v.2.0.2)*, *SparseM(v.1.77)*, *mvtnorm(v.1.0-6)*, *scales(v.1.0.0)*, *MatrixModels(v.0.4-1)*, *htmlTable(v.1.13.1)*, *tibble(v.2.1.3)*, *mgcv(v.1.8-28)*, *TH.data(v.1.0-10)*, *withr(v.2.1.2)*, *nnet(v.7.3-12)*, *lazyeval(v.0.2.2)*, *cli(v.1.1.0)*, *pbkrtest(v.0.4-7)*, *mnormt(v.1.5-5)*, *crayon(v.1.3.4)*, *evaluate(v.0.14)*, *fansi(v.0.4.0)*, *data.table(v.1.10.4-3)*, *tools(v.3.4.2)*, *matrixStats(v.0.54.0)*, *multcomp(v.1.4-7)*, *stringr(v.1.4.0)*, *munsell(v.0.5.0)*, *cluster(v.2.0.6)*, *bindrcpp(v.0.2.2)*, *compiler(v.3.4.2)*, *rlang(v.0.4.0)*, *grid(v.3.4.2)*, *nloptr(v.1.2.1)*, *rstudioapi(v.0.10)*, *htmlwidgets(v.1.3)*, *base64enc(v.0.1-3)*, *rmarkdown(v.1.15)*, *boot(v.1.3-23)*, *gtable(v.0.3.0)*, *codetools(v.0.2-16)*, *lmerTest(v.3.1-0)*, *reshape2(v.1.4.3)*, *R6(v.2.4.0)*, *gridExtra(v.2.3)*, *zoo(v.1.8-6)*, *knitr(v.1.24)*, *zeallot(v.0.1.0)*, *utf8(v.1.1.4)*, *bindr(v.0.1.1)*, *libcoin(v.1.0-4)*, *modeltools(v.0.2-22)*, *stringi(v.1.4.3)*, *parallel(v.3.4.2)*, *Rcpp(v.1.0.2)*, *vctrs(v.0.2.0)*, *rpart(v.4.1-15)*, *acepack(v.1.4.1)* and *xfun(v.0.8)*