Demographic and Behavioral Data Analysis

Setup

```r
# load packages
packages <- c("here","dplyr","ggplot2","ppcor","tidyverse","multcomp",
          "effectsize", "xlsx","effectsize")
lapply(packages, library, character.only = TRUE)
attachNamespace("nlme")

# load function
source(here("code/posttest_interaction.R"))
```

Load data

```r
variables_all <- read.table(here("data/variables_TD_N50.txt"),sep = "\t", header = T)
posttest_long <- read.table(here("data/posttest_N50.txt"), header = T)
```

Demographic information

```r
# gender
table(variables_all$gender[!duplicated(variables_all$Subject)])
```

```
##
##  1  2
## 30 20
```

```r
# age, mean FD
variables_all[!duplicated(variables_all$Subject),] %>%
    summarise(mean(age),sd(age),max(age),min(age), mean(mean_FD),
          sd(mean_FD),max(mean_FD),min(mean_FD))
```

```
##   mean(age)  sd(age) max(age) min(age) mean(mean_FD) sd(mean_FD) max(mean_FD)
## 1   10.3822 1.332384    12.97     8.18       0.22786  0.07765465        0.417
##   min(mean_FD)
## 1        0.096
```

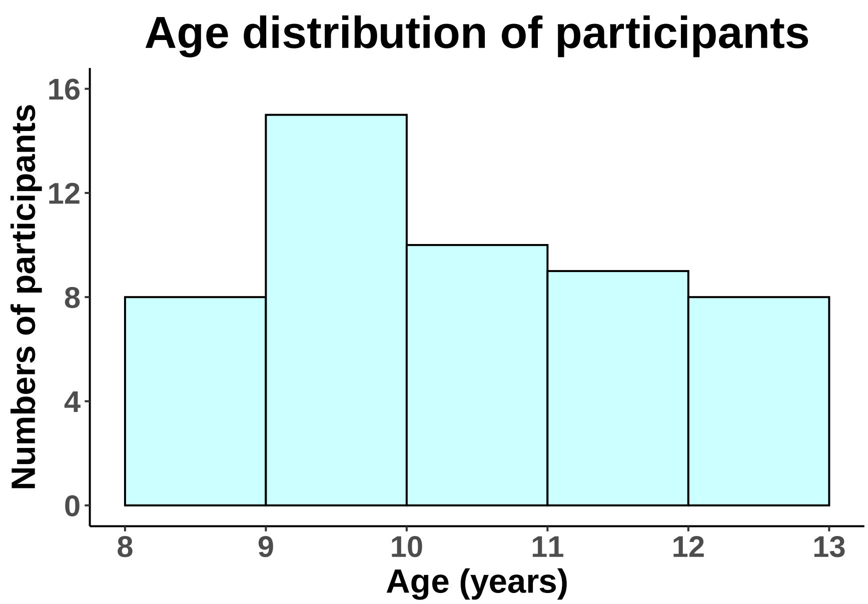
```r
# correlation between age and mean FD

rr <- cor.test(variables_all$age[!duplicated(variables_all$Subject)],
     variables_all$mean_FD[!duplicated(variables_all$Subject)])

rr
```

```
##
##  Pearson's product-moment correlation
##
## data:  variables_all$age[!duplicated(variables_all$Subject)] and variables_all$mean_FD[!duplicated(va
## t = -1.7305, df = 48, p-value = 0.08996
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.48777851  0.03861993
## sample estimates:
##        cor
## -0.2423328
```

```r
r_to_d(rr$estimate)
```

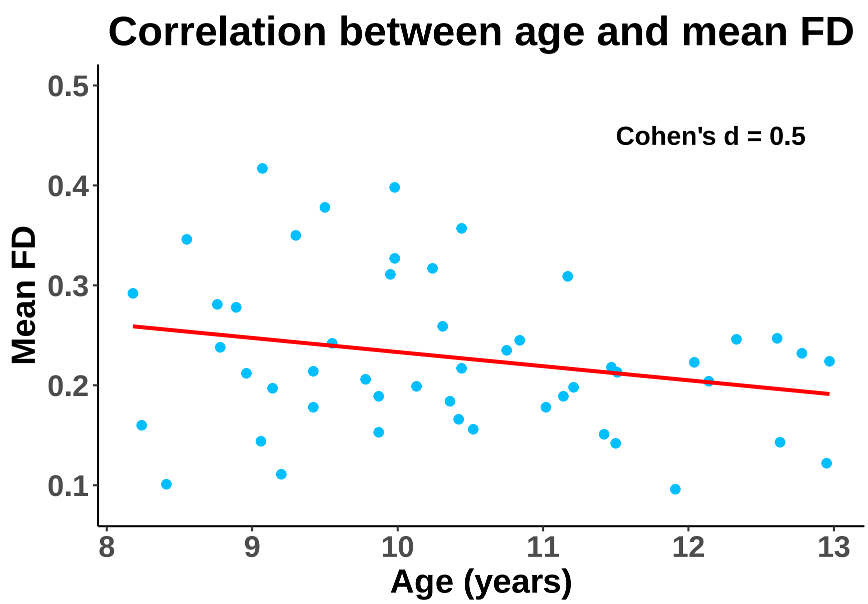
```
##        cor
## -0.4995558
```

```r
## plots
# age distribution
ggplot(variables_all[!duplicated(variables_all$Subject),], aes(x=age)) +
    geom_histogram(binwidth = 1,boundary = 0, closed = "left",
               colour="black",fill="#CCFFFF") +
    theme(panel.background = element_blank(),
          axis.line = element_line(colour = "black")) +
    scale_y_continuous(breaks=seq(0, 16, 4),limits = c(0,16)) +
    scale_x_continuous(breaks=seq(8, 13, 1),limits = c(8,13)) +
    labs(x = "Age (years)", y = "Numbers of participants") +
    theme(axis.text = element_text(size = 16, face = "bold"),
          axis.title = element_text(size = 18, face = "bold") ) +
    ggtitle("Age distribution of participants") +
    theme(plot.title = element_text(hjust = 0.5),
          text = element_text(size = 20, face = "bold"))
```




```r
# plots for age and mean FD
ggplot(variables_all[!duplicated(variables_all$Subject),],
       aes(x=age, y = mean_FD)) +
    geom_point(color = "deepskyblue1", size = 2) +
    geom_smooth(method=lm,se=F,color = "red",fullrange = T) +
    labs(x = "Age (years)", y = "Mean FD", title = "Correlation between age and mean FD") +
    scale_y_continuous(breaks=seq(0.0,0.5,0.1),limits = c(0.08,0.5)) +
    theme(panel.background = element_blank(),
          axis.line = element_line(colour = "black")) +
    theme(axis.text = element_text(size = 16, face = "bold"),
          axis.title = element_text(size = 18, face = "bold") ) +
     annotate("text", x=11.5,y=.45, label = "Cohen's d = 0.5",
              hjust=0,size = 5, fontface = "bold") +
    theme(plot.title = element_text(hjust = 0.5),
          text = element_text(size = 18, face ="bold"))
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Ethnicity, race, and income

```
demog_cmnt <- read.xlsx(here("data/demog_cmnt.xlsx"),sheetIndex = 1)
demog_cat <- read.csv(here("data/demog_CAT.csv"))

demog_cmnt_new <- demog_cmnt[demog_cmnt$Participant %in%
        variables_all$Subject[!duplicated(variables_all$Subject)],
        c("Participant","X3..Ethnicity","X4..Race","X15..Household.Income")]

colnames(demog_cmnt_new) <- c("subjectid", "ethnicity", "race", "income")

demog_cat_new <- demog_cat[demog_cat$RED_ID %in%
        variables_all$Subject[!duplicated(variables_all$Subject)],
        c("RED_ID","ethnicity","race","income")]
# ethnicity
demog_cat_new$Ethnicity[demog_cat_new$ethnicity == "Not Hispanic or Latino"] <- "Not Hispanic/Latino"
demog_cat_new$Ethnicity[demog_cat_new$ethnicity == "Hispanic or Latino"] <- "Hispanic/Latino"
demog_cat_new$Ethnicity[demog_cat_new$ethnicity == "Does not wish to disclose"] <- "Unknown"

colnames(demog_cat_new)[1] <- c("subjectid")
head(demog_cat_new)
```

```
##      subjectid              ethnicity
## 1  RED_CAT_112 Not Hispanic or Latino
## 2  RED_CAT_123 Not Hispanic or Latino
## 5  RED_CAT_124     Hispanic or Latino
## 6  RED_CAT_118 Not Hispanic or Latino
## 11 RED_CAT_133 Not Hispanic or Latino
## 12 RED_CAT_150 Not Hispanic or Latino
##                                                                  race
## 1  American Indian or Alaskan Native,Black or African American
## 2                                  Black or African American,Asian
```

```
## 5                                            White or Caucasian
## 6                                            White or Caucasian
## 11                                           White or Caucasian
## 12                                           White or Caucasian
##                         income          Ethnicity
## 1  more than $75,000 per year Not Hispanic/Latino
## 2  more than $75,000 per year Not Hispanic/Latino
## 5  more than $75,000 per year     Hispanic/Latino
## 6  more than $75,000 per year Not Hispanic/Latino
## 11 more than $75,000 per year Not Hispanic/Latino
## 12 more than $75,000 per year Not Hispanic/Latino
```

```
demog_cmnt_new$Ethnicity[demog_cmnt_new$ethnicity == "N"] <- "Not Hispanic/Latino"
demog_cmnt_new$Ethnicity[demog_cmnt_new$ethnicity == "H"] <- "Hispanic/Latino"

# race
demog_cat_new$Race[demog_cat_new$race == "Asian,White or Caucasian" |
           demog_cat_new$race == "American Indian or Alaskan Native,Black or African American"|
           demog_cat_new$race == "Black or African American,White or Caucasian" |
           demog_cat_new$race == "Black or African American,Asian"] <- "more than one race"
demog_cat_new$Race[demog_cat_new$race == "White or Caucasian"] <- "White/Caucasian"
demog_cat_new$Race[demog_cat_new$race == "Black or African American"] <- "Black/African American"

demog_cmnt_new$Race <- demog_cmnt_new$race
demog_cmnt_new$Race[demog_cmnt_new$race == "W"] <- "White/Caucasian"
demog_cmnt_new$Race[demog_cmnt_new$race == "B"] <- "Black/African American"
demog_cmnt_new$Race[demog_cmnt_new$race == "A, W" |
            demog_cmnt_new$race == "B, W"  ] <- "more than one race"

# income
demog_cmnt_new$Income <- demog_cmnt_new$income
demog_cmnt_new$Income[demog_cmnt_new$income == "7"] <- ">75k"
demog_cmnt_new$Income[demog_cmnt_new$income == "6"] <- "65k-75k"
demog_cmnt_new$Income[demog_cmnt_new$income == "4"] <- "45k-55k"
demog_cmnt_new$Income[is.na(demog_cmnt_new$income)] <- "Unknown"

demog_cat_new$Income[demog_cat_new$income == "more than $75,000 per year"] <- ">75k"
demog_cat_new$Income[demog_cat_new$income == "$15, 000-$25, 000 per year"] <- "15k-25k"
demog_cat_new$Income[demog_cat_new$income == "$35, 000-$45, 000 per year"] <- "35k-45k"
demog_cat_new$Income[demog_cat_new$income == ""] <- "Unknown"

demog_all <- rbind.data.frame(demog_cmnt_new[,c("subjectid","Ethnicity", "Race", "Income")],
                  demog_cat_new[,c("subjectid","Ethnicity", "Race","Income")])

demog_all$Ethnicity <- factor(demog_all$Ethnicity, levels = c("Not Hispanic/Latino",
                                  "Hispanic/Latino", "Unknown"))
demog_all$Race <- factor(demog_all$Race, levels = c("White/Caucasian","Black/African American",
                            "more than one race"))
demog_all$Income <- factor(demog_all$Income, levels = c(">75k","65k-75k","45k-55k",
                            "35k-45k","15k-25k","Unknown"))

a <- table(demog_all$Ethnicity)
b <- table(demog_all$Race)
inc <- table(demog_all$Income)
```

```r
print(paste0((a["Hispanic/Latino"]/50)*100,"% as Hispanic/Latino"))
```

```
## [1] "10% as Hispanic/Latino"
```

```r
print(paste0((b["White/Caucasian"]/50)*100,"% as White/Caucasian"))
```

```
## [1] "60% as White/Caucasian"
```

```r
print(paste0((b["Black/African American"]/50)*100,"% as Black/African American"))
```

```
## [1] "22% as Black/African American"
```

```r
print(paste0((b["more than one race"]/50)*100,"% as more than one race"))
```

```
## [1] "18% as more than one race"
```

```r
print(paste0((inc[">75k"]/50)*100,"% reporting over $75,000 in total family income"))
```

```
## [1] "86% reporting over $75,000 in total family income"
```

```r
print(paste0(((inc["65k-75k"] + inc["45k-55k"] + inc["35k-45k"])/50)*100,
         "% reporting family income between $35,000-$75,000"))
```

```
## [1] "6% reporting family income between $35,000-$75,000"
```

```r
print(paste0((inc["15k-25k"]/50)*100, "% reporting family income less than $35,000"))
```

```
## [1] "4% reporting family income less than $35,000"
```

```r
print(paste0((inc["Unknown"]/50)*100, "% did not report on income"))
```

```
## [1] "4% did not report on income"
```

In-scanner performance

```r
# mean RT and accuracy
variables_all %>%
    summarise(mean(RT),sd(RT),mean(ACC_per), sd(ACC_per))
```

```
##   mean(RT)    sd(RT) mean(ACC_per) sd(ACC_per)
## 1 2.043435 0.3197043      90.16661     7.49084
```

```r
# regression analysis on RT
anova(lme(RT ~ social*age + mental +  gender + mean_FD + IQ,
      random = ~1|Subject,
      data = variables_all))
```

```
##             numDF denDF  F-value p-value
## (Intercept)     1   147 3449.402  <.0001
## social          1   147   27.080  <.0001
## age             1    45   27.059  <.0001
## mental          1   147    0.339  0.5614
## gender          1    45    1.430  0.2380
## mean_FD         1    45    0.167  0.6852
## IQ              1    45    1.718  0.1966
## social:age      1   147    0.163  0.6872
```

```r
as.matrix(by(variables_all[,c("age","RT")], variables_all$MentalState,
         function(x) {cor.test(x$age, x$RT)}))
```

```
## : CM
## :
##
##  Pearson's product-moment correlation
##
## data:  x$age and x$RT
## t = -4.409, df = 48, p-value = 5.831e-05
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.7092431 -0.3039618
## sample estimates:
##        cor
## -0.5368853
##
## ------------------------------------------------------------
## : CNM
## :
##
##  Pearson's product-moment correlation
##
## data:  x$age and x$RT
## t = -5.1098, df = 48, p-value = 5.54e-06
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.7482830 -0.3776036
## sample estimates:
##        cor
## -0.5935646
##
## ------------------------------------------------------------
## : PM
## :
##
##  Pearson's product-moment correlation
##
```

```
## data:  x$age and x$RT
## t = -5.2166, df = 48, p-value = 3.841e-06
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.7536694 -0.3881461
## sample estimates:
##        cor
## -0.6015072
##
## ------------------------------------------------------------
## : PNM
## :
##
##  Pearson's product-moment correlation
##
## data:  x$age and x$RT
## t = -4.6228, df = 48, p-value = 2.875e-05
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.7218574 -0.3272382
## sample estimates:
##        cor
## -0.5550304
```

```
# post-hoc analysis on RT: mean RT of collapsed conditions between Peer vs. Character condition
beha <- as.data.frame(matrix(0,100,0))
beha$subj <- rep(variables_all$Subject[!duplicated(variables_all$Subject)],2)
beha$social <- rep(c("P","C"),each = 50)
beha$RT <- c((variables_all[variables_all$MentalState == "PM","RT"] +
    variables_all[variables_all$MentalState == "PNM","RT"])/2,
    (variables_all[variables_all$MentalState == "CM","RT"] +
        variables_all[variables_all$MentalState == "CNM","RT"])/2)

tt0 <- t.test(beha[beha$social == "P","RT"],beha[beha$social == "C","RT"],
          paired = T)

print(paste0("Peer vs. Character differences = ", round(tt0$estimate[1],2)))
```

```
## [1] "Peer vs. Character differences = -0.08"
```

```
print(paste0("t=", round(tt0$statistic,1), " p=",round(tt0$p.value,5)))
```

```
## [1] "t=-4.7 p=2e-05"
```

```
# regression analysis on accuracy
anova(lme(ACC_per ~ social*age + mental + gender + mean_FD + IQ,
      random = ~1|Subject,
      data = variables_all))
```

```
##             numDF denDF   F-value p-value
## (Intercept)     1   147 23396.459  <.0001
## social          1   147     2.042  0.1551
```

```
## age              1    45     1.770  0.1900
## mental           1   147     3.277  0.0723
## gender           1    45     0.042  0.8386
## mean_FD          1    45     2.487  0.1218
## IQ               1    45    24.331  <.0001
## social:age       1   147     5.818  0.0171
```

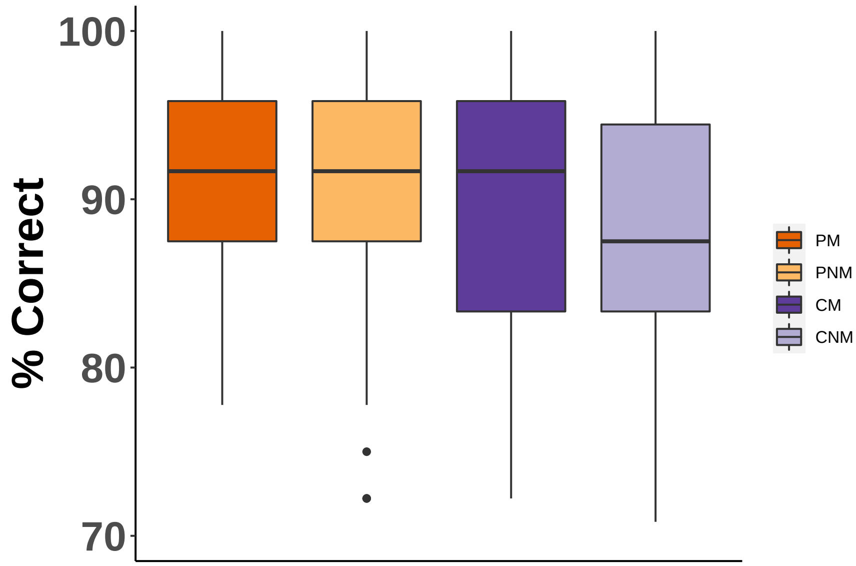
```r
anova(lme(ACC_per ~ age + mental + gender + mean_FD + IQ,
      random = ~1|Subject,
      data = variables_all[variables_all$social == "C",]))
```

```
##             numDF denDF   F-value p-value
## (Intercept)     1    49 12940.761  <.0001
## age             1    45     5.460  0.0240
## mental          1    49     2.069  0.1566
## gender          1    45     0.509  0.4794
## mean_FD         1    45     0.392  0.5346
## IQ              1    45    13.133  0.0007
```

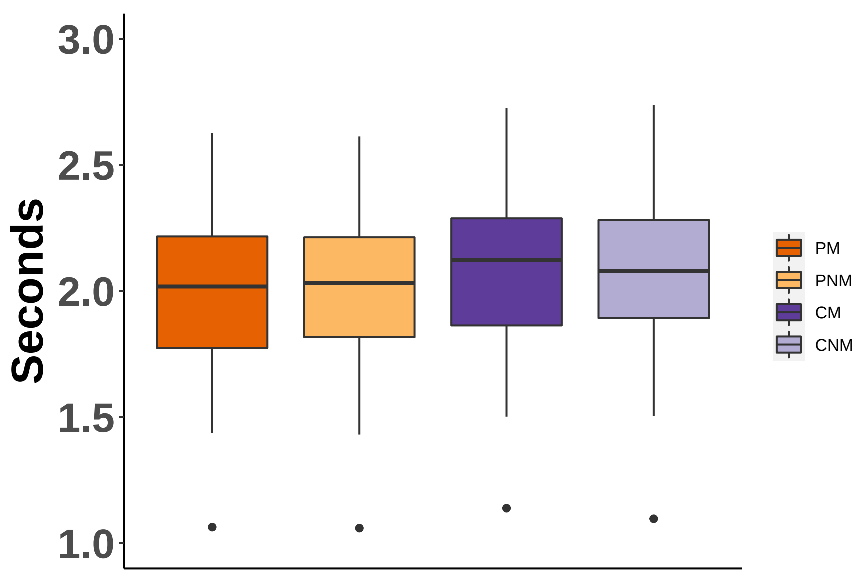
```r
anova(lme(ACC_per ~ age + mental + gender + mean_FD + IQ,
      random = ~1|Subject,
      data = variables_all[variables_all$social == "P",]))
```

```
##             numDF denDF   F-value p-value
## (Intercept)     1    49 18531.579  <.0001
## age             1    45     0.165  0.6869
## mental          1    49     1.220  0.2747
## gender          1    45     1.450  0.2349
## mean_FD         1    45     4.198  0.0463
## IQ              1    45    19.735  0.0001
```

```r
# boxplots
# function
boxplot_inscan <- function(data,x,y,n,beh,title) {
    p <- ggplot(data, aes_string(x, y, fill=x)) +
        geom_boxplot() +
        scale_fill_manual(values = c("#e66101","#fdb863","#5e3c99","#b2abd2")) +
        labs(y=beh, x="") +
        coord_cartesian(ylim = n)+
        theme(axis.text.x = element_blank(),
              axis.ticks.x = element_blank(),
              axis.text.y =element_text(size=22,face="bold"),
              axis.title.y =element_text(size=24,face="bold")) +
        theme(legend.title=element_blank()) +
        theme(panel.background = element_blank(),
              axis.line = element_line(colour = "black"))   #remove background
    print(p)
}

variables_all$MentalState <- factor(variables_all$MentalState,levels = c("PM","PNM","CM","CNM"))

boxplot_inscan(variables_all,"MentalState","ACC_per",c(70,100),c("% Correct"),c("Accuracy"))
```




```
boxplot_inscan(variables_all,"MentalState","RT",c(1,3),c("Seconds"),c("Reaction Time"))
```



Posttest questionnaire

```
# subjective reports assessed by the post-scan questionnaire
mean_P <- sapply(4:9,function(x) summary(posttest_long[posttest_long$social == "Peer",x]))
sd_P <- sapply(4:9,function(x) sd(posttest_long[posttest_long$social == "Peer",x]))
mean_C <- sapply(4:9,function(x) summary(posttest_long[posttest_long$social == "Character",x]))
sd_C <- sapply(4:9,function(x) sd(posttest_long[posttest_long$social == "Character",x]))

tt  <-  sapply(4:9,function(x) wilcox.test(posttest_long[posttest_long$social == "Peer",x],
```

```r
                      posttest_long[posttest_long$social == "Character",x],
                       paired = T))

reports <- as.data.frame(matrix(0, 6, 8))
colnames(reports) <- c("Measure","P.median","P.mean±sd","P.range","C.median",
               "C.mean±sd","C.range","PvsC")

for (i in 1:length(4:9)) {
    reports[i,1] <- colnames(posttest_long)[3+i]
    reports[i,2:8] <- c(mean_P["Median",i], paste0(mean_P["Mean",i],"±",round(sd_P[i],2)),
               paste0(mean_P["Min.",i],"-",mean_P["Max.",i]),mean_C["Median",i],
               paste0(mean_C["Mean",i],"±",round(sd_C[i],2)),
               paste0(mean_C["Min.",i],"-",mean_C["Max.",i]),
               round(tt[,i]$p.value,4))
}

posttest <- c("like","likeguess","agreed","wantsee","attention","hardguess")

l <- match(posttest, reports[,1])

knitr::kable(reports[l,])
```

	Measure	P.median	P.mean±sd	P.range	C.median	C.mean±sd	C.range	PvsC
1	like	4.5	4.3±0.81	2–5	3	2.76±1.04	1–5	0
4	likeguess	4	3.84±0.96	1–5	3	3.26±1.12	1–5	9e-04
6	agreed	4	4.26±0.83	2–5	4	3.96±0.88	2–5	0.0312
5	wantsee	4	4.08±0.92	2–5	3	3.38±1.23	1–5	0
2	attention	4	3.9±0.91	1–5	3	3.4±0.97	1–5	0.0019
3	hardguess	2	2.38±1.23	1–5	2	2.2±1.29	1–5	0.3831

```r
## variability of subjective reports of enjoyment
# percent of subjects who rated 4 or 5 on Liked Chatting
k <- length(which(posttest_long$like[posttest_long$social == "Peer" ]>3))
(k/50)*100
```

```
## [1] 82
```

```r
# percent of  subjects who rated 4 or 5 when their answer matched the answer from the peer
k <- length(which(posttest_long$agreed[posttest_long$social == "Peer" ]>3))
(k/50)*100
```

```
## [1] 80
```

```r
# subjective reports when answering questions about the character
m <-  table(posttest_long$like[posttest_long$social == "Character" ])

(m/50)*100
```

```
##
##  1  2  3  4  5
## 12 24 48  8  8
```

```r
n <- as.numeric((m/50)*100)

print(paste0(n[1]+n[2],"% rated 1 or 2"))
```

```
## [1] "36% rated 1 or 2"
```

```r
print(paste0(n[3],"% rated 3"))
```

```
## [1] "48% rated 3"
```

```r
print(paste0(n[4]+n[5],"% rated 4 or 5"))
```

```
## [1] "16% rated 4 or 5"
```

```r
## regression analysis
posttest_long_new <- cbind.data.frame(posttest_long[rep(rownames(posttest_long),
                            each = 2),],
                variables_all[,c("gender","IQ","RT","ACC_per")])

# Liked Chatting
ano1 <- anova(lme(like ~ social*Age + gender + IQ, random = ~1|Subj,
          data = posttest_long_new))
# Liked Guessing
ano2 <- anova(lme(likeguess ~ social*Age + gender + IQ, random = ~1|Subj,
          data = posttest_long_new))
# Felt When Matched
ano3 <- anova(lme(agreed~ social*Age + gender + IQ,random = ~1|Subj,
      data=posttest_long_new))
# Wanted to See
ano4 <- anova(lme(wantsee ~ social*Age + gender + IQ,random = ~1|Subj,
          data=posttest_long_new))
# Paid Attention
ano5 <- anova(lme(attention ~ social*Age + gender + IQ,random = ~1|Subj,
      data = posttest_long_new))
# Perceived Difficulty
ano6 <- anova(lme(hardguess ~ social*Age + gender + IQ,random = ~1|Subj,
          data = posttest_long_new))

# multiple comparisons correction
c(ano1$`p-value`[6],ano2$`p-value`[6],ano3$`p-value`[6],ano4$`p-value`[6],
  ano5$`p-value`[6],ano6$`p-value`[6])
```

```
## [1] 2.212346e-02 1.956633e-01 4.782630e-01 1.195158e-02 8.648441e-05
## [6] 6.460372e-01
```

```r
pjusted <- p.adjust(c(ano1$`p-value`[6],ano2$`p-value`[6],ano3$`p-value`[6],ano4$`p-value`[6],
       ano5$`p-value`[6],ano6$`p-value`[6]),method = "fdr")
reports <- c("Liked Chatting","Liked Guessing","Felt When Matched",
         "Wanted to See","Paid Attention","Perceived Difficulty")

which(pjusted < 0.05)
```

```
## [1] 1 4 5
```

```r
reports[which(pjusted < 0.05)]
```

```
## [1] "Liked Chatting" "Wanted to See"  "Paid Attention"
```

```r
rbind.data.frame(ano1[6,],ano4[6,],ano5[6,])
```

```
##              numDF denDF   F-value p-value
## social:Age       1   148  5.348077  0.0221
## social:Age1      1   148  6.477069  0.0120
## social:Age2      1   148 16.299305  0.0001
```

```r
## post-hoc Spearman correlation
test1 <- by(posttest_long_new[,c("Age","like")], posttest_long_new$social,
   function(x) { cor.test(x$Age, x$like, method = "spearman")})
test2 <- by(posttest_long_new[,c("Age","likeguess")], posttest_long_new$social,
   function(x) { cor.test(x$Age, x$likeguess, method = "spearman")})
test3 <- by(posttest_long_new[,c("Age","agreed")], posttest_long_new$social,
   function(x) { cor.test(x$Age, x$agreed, method = "spearman")})
test4 <- by(posttest_long_new[,c("Age","wantsee")], posttest_long_new$social,
   function(x) { cor.test(x$Age, x$wantsee, method = "spearman")})
test5 <- by(posttest_long_new[,c("Age","attention")], posttest_long_new$social,
   function(x) { cor.test(x$Age, x$attention, method = "spearman")})
test6 <- by(posttest_long_new[,c("Age","hardguess")], posttest_long_new$social,
   function(x) { cor.test(x$Age, x$hardguess, method = "spearman")})

posthoc_test <- data.frame(matrix(0,6,6))

posthoc_test <- rbind(cbind(round(ano1$`F-value`[6],3),round(ano1$`p-value`[6],3),
                round(test1$Character$estimate,3),
                round(test1$Character$p.value,3),
                round(test1$Peer$estimate,3),round(test1$Peer$p.value,3)),
                cbind(round(ano2$`F-value`[6],3),round(ano2$`p-value`[6],3),
                      round(test2$Character$estimate,3),
                      round(test2$Character$p.value,3),
                      round(test2$Peer$estimate,3),round(test2$Peer$p.value,3)),
                cbind(round(ano3$`F-value`[6],3),round(ano3$`p-value`[6],3),
                      round(test3$Character$estimate,3),
                      round(test3$Character$p.value,3),
                      round(test3$Peer$estimate,3),
                      round(test3$Peer$p.value,3)),
                cbind(round(ano4$`F-value`[6],3),round(ano4$`p-value`[6],3),
                      round(test4$Character$estimate,3),
                      round(test4$Character$p.value,3),
                      round(test4$Peer$estimate,3),
                      round(test4$Peer$p.value,3)),
                cbind(round(ano5$`F-value`[6],3),round(ano5$`p-value`[6],3),
                      round(test5$Character$estimate,3),
                      round(test5$Character$p.value,3),
                      round(test5$Peer$estimate,3),
                      round(test5$Peer$p.value,3)),
```

```r
                 cbind(round(ano6$`F-value`[6],3),round(ano6$`p-value`[6],3),
                       round(test6$Character$estimate,3),
                       round(test6$Character$p.value,3),
                       round(test6$Peer$estimate,3),
                       round(test6$Peer$p.value,3)))

colnames(posthoc_test) <- c("interaction-F","interaction-p","Character-rho",
                 "Character-p","Peer-rho","Peer-p")
rownames(posthoc_test) <- c("Liked Chatting","Liked Guessing","Felt When Matched",
         "Wanted to See","Paid Attention","Perceived Difficulty")

posthoc_test
```

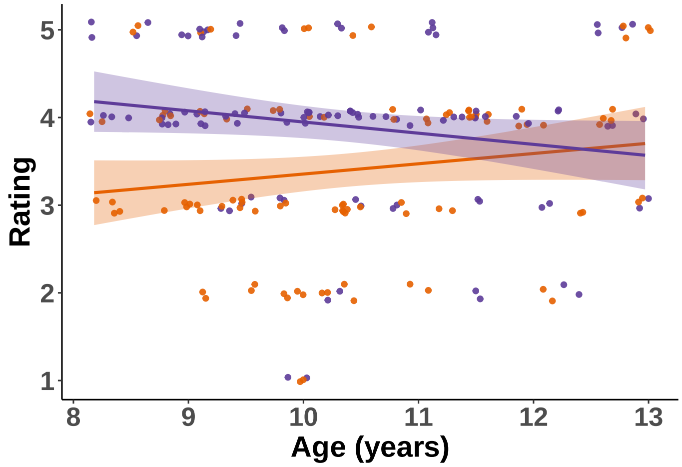
```
##                      interaction-F interaction-p Character-rho Character-p
## Liked Chatting               5.348         0.022         0.218       0.029
## Liked Guessing               1.690         0.196        -0.003       0.975
## Felt When Matched            0.505         0.478        -0.139       0.166
## Wanted to See                6.477         0.012        -0.222       0.026
## Paid Attention              16.299         0.000         0.154       0.127
## Perceived Difficulty         0.212         0.646        -0.428       0.000
##                      Peer-rho Peer-p
## Liked Chatting         -0.154  0.126
## Liked Guessing         -0.074  0.466
## Felt When Matched      -0.102  0.314
## Wanted to See          -0.080  0.429
## Paid Attention         -0.223  0.026
## Perceived Difficulty   -0.433  0.000
```

```r
posthoc_test[which(pjusted < 0.05),]
```

```
##                interaction-F interaction-p Character-rho Character-p Peer-rho
## Liked Chatting         5.348         0.022         0.218       0.029   -0.154
## Wanted to See          6.477         0.012        -0.222       0.026   -0.080
## Paid Attention        16.299         0.000         0.154       0.127   -0.223
##                Peer-p
## Liked Chatting  0.126
## Wanted to See   0.429
## Paid Attention  0.026
```

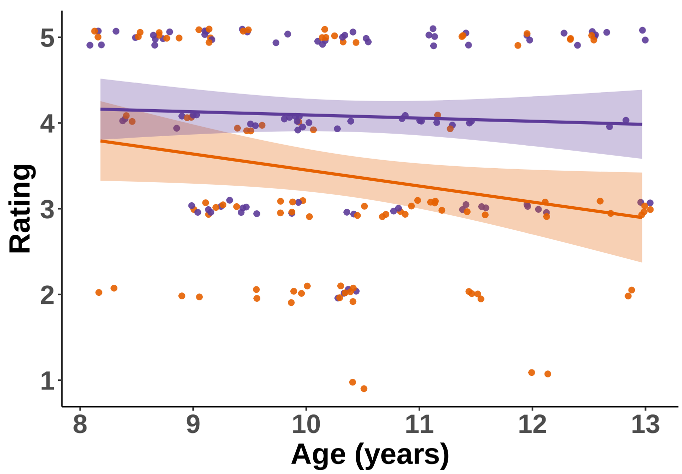
```r
posttest_interaction(posttest_long_new,"attention","Paid Attention","attention_age")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



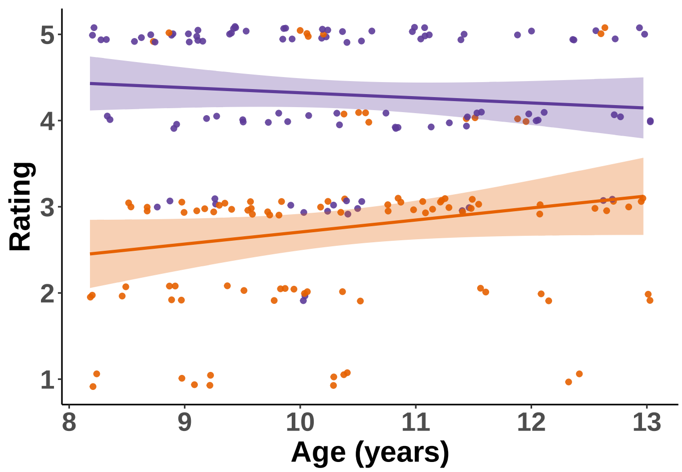

```r
posttest_interaction(posttest_long_new,"wantsee","Want to See","wantsee_age")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



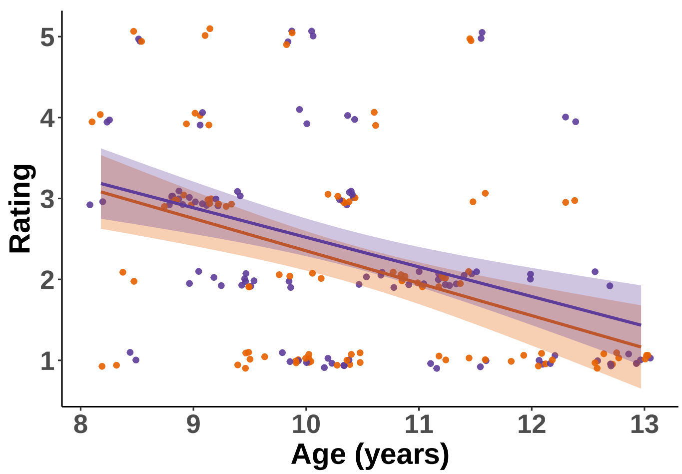

```r
posttest_interaction(posttest_long_new,"like","Like Chatting","likechatting_age")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



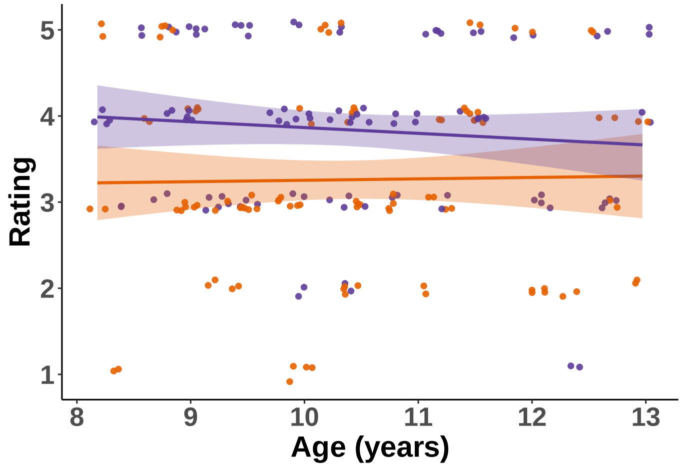

```
posttest_interaction(posttest_long_new,"hardguess","Like Chatting","hardguess_age")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



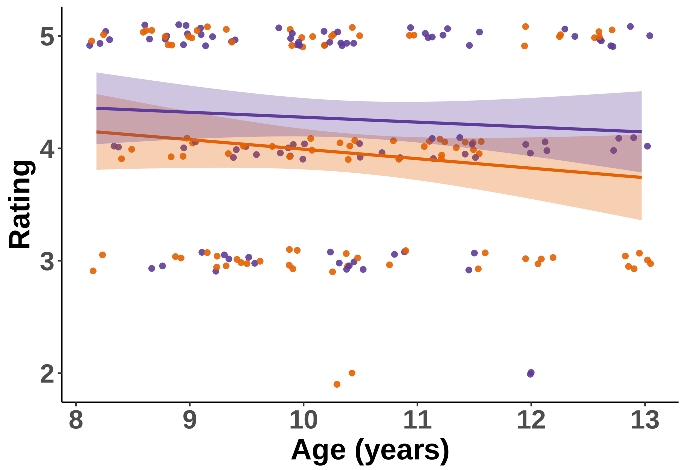

```
posttest_interaction(posttest_long_new,"likeguess","Like Chatting","likeguess_age")
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
posttest_interaction(posttest_long_new,"agreed","Like Chatting","agreed_age")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

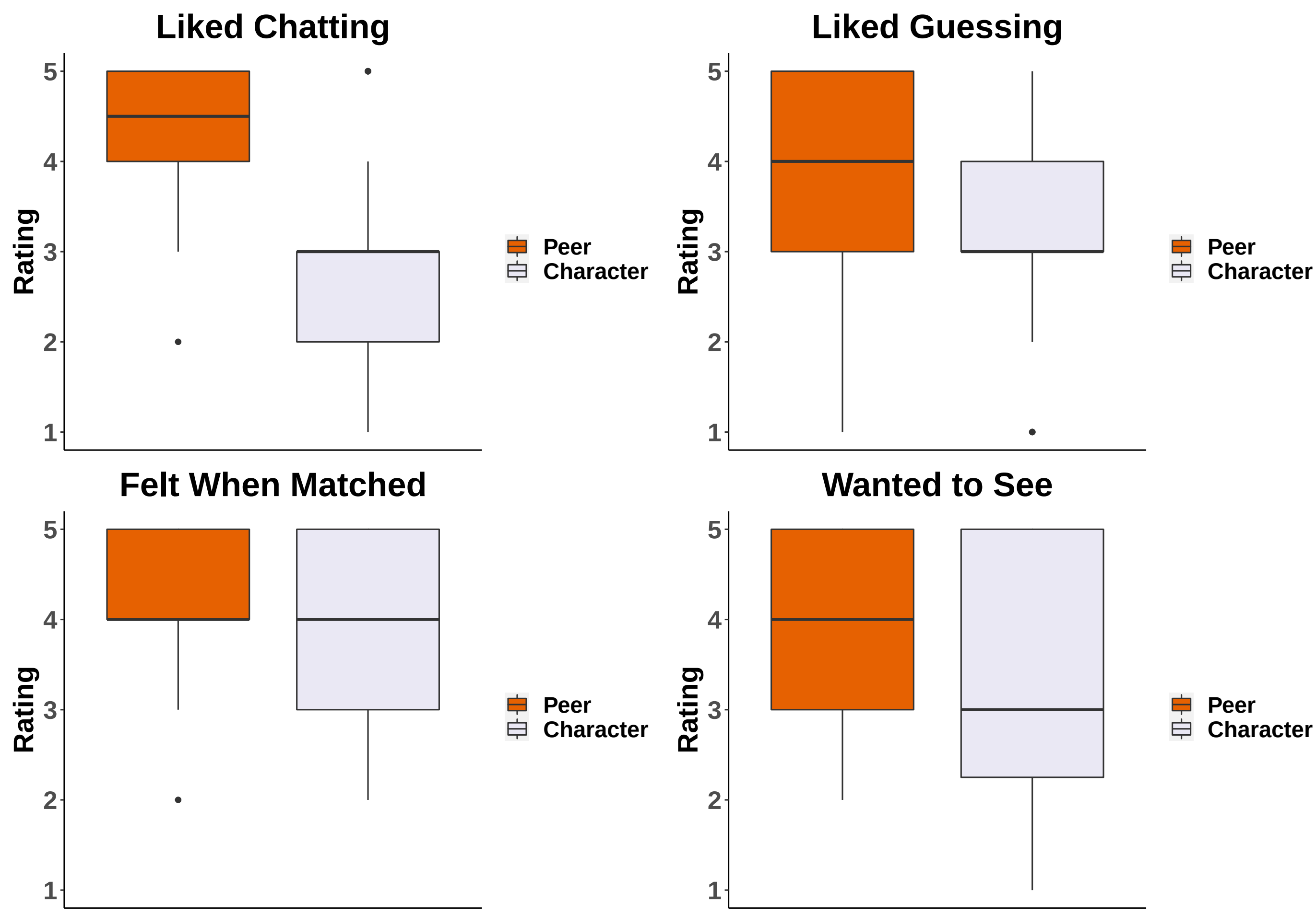


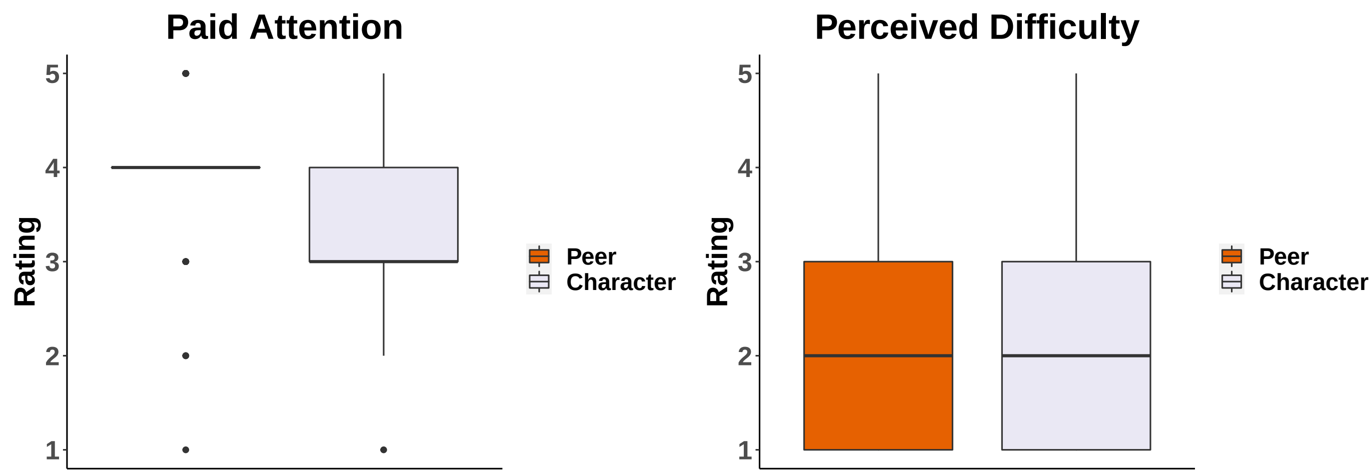

```
# boxplots
measures <- c("like","likeguess","agreed","wantsee","attention","hardguess")
titles <- c("Liked Chatting","Liked Guessing","Felt When Matched","Wanted to See",
        "Paid Attention","Perceived Difficulty")

posttest_long$social <- factor(posttest_long$social,levels = c("Peer", "Character"))

summary(posttest_long$attention)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    3.00    4.00    3.65    4.00    5.00
```

```
for (i in 1:6) {
    p <- ggplot(posttest_long, aes_string(x="social",y=measures[i],fill = "social")) +
        geom_boxplot() +
        labs(y="Rating") +
```

```r
        ggtitle(titles[i]) +
        theme(plot.title = element_text(hjust = 0.5),
              text = element_text(size = 20, face = "bold")) +
        theme(legend.title=element_blank()) +
        theme(axis.title.x = element_blank(),
              axis.ticks.x = element_blank(),
              axis.text.x = element_blank(),
              axis.title.y = element_text(face = "bold",size=20),
              axis.text.y = element_text(face = "bold",size=18)) +
        coord_cartesian(ylim = c(1,5)) +
        scale_fill_manual(values = c("#e66101", "#eae8f4"))+
        theme(panel.background = element_blank(),
              axis.line = element_line(colour = "black"))
    print(p)
}
```





Correlations between mean RT and scial motivation

```
# mean RT
RT_mean <- aggregate(variables_all$RT,by = list(variables_all$social,variables_all$Subj),
             FUN=mean)

Cor_RT_Social <- as.data.frame(matrix(0,5,3))

posttest <- c("like","likeguess","agreed","wantsee","attention")

for (i in 1:length(posttest)) {
    mytest_P <- cor.test(RT_mean$x[RT_mean$Group.1 == "P"], posttest_long[posttest_long$social == "Peer
     method = "spearman",alternative = "less")

    Cor_RT_Social[i,1] <- posttest[i]
    Cor_RT_Social[i,2:3] <- c(round(mytest_P$estimate,3),round(mytest_P$p.value,3))
}

colnames(Cor_RT_Social) <- c("","rho","p")
Cor_RT_Social[,1] <- titles <- c("Liked Chatting","Liked Guessing","Felt When Matched",
                  "Wanted to See","Paid Attention")
knitr::kable(Cor_RT_Social)
```

	rho	p
Liked Chatting	-0.123	0.197
Liked Guessing	-0.065	0.327
Felt When Matched	-0.252	0.038
Wanted to See	-0.238	0.048
Paid Attention	-0.030	0.418

Functional Connectivity and Brain-Behavior Analysis

Setup

```
# load packages
packages <- c("here","dplyr","ggplot2","ppcor","tidyverse","nlme","multcomp","xlsx")
lapply(packages, library, character.only = TRUE)

# load function
source(here("code/flm_FC.R"))
source(here("code/FC_scatterplot.R"))
```

Load data

```
mean_networks_FC <- read.table(here("data/mean_networks_FC.txt"), header = T)

mean_control_FC <- read.table(here("data/mean_control_FC.txt"), header = T)

variables_all <- read.table(here("data/variables_TD_N50.txt"), sep = "\t", header = T)
```

Within vs. between network

```
tmp0 <- as.data.frame(matrix(0,50,5))

tmp0$reward <- (mean_networks_FC$reward_FC[mean_networks_FC$conditions=="PM"] +
            mean_networks_FC$reward_FC[mean_networks_FC$conditions=="PNM"] +
            mean_networks_FC$reward_FC[mean_networks_FC$conditions=="CM"] +
            mean_networks_FC$reward_FC[mean_networks_FC$conditions=="CNM"])/4

tmp0$mentalizing <- (mean_networks_FC$mental_FC[mean_networks_FC$conditions=="PM"] +
        mean_networks_FC$mental_FC[mean_networks_FC$conditions=="PNM"] +
            mean_networks_FC$mental_FC[mean_networks_FC$conditions=="CM"] +
            mean_networks_FC$mental_FC[mean_networks_FC$conditions=="CNM"])/4

tmp0$between <- (mean_networks_FC$between_FC[mean_networks_FC$conditions=="PM"] +
         mean_networks_FC$between_FC[mean_networks_FC$conditions=="PNM"] +
         mean_networks_FC$between_FC[mean_networks_FC$conditions=="CM"] +
         mean_networks_FC$between_FC[mean_networks_FC$conditions=="CNM"])/4

# within vs. between: one-tailed paired t-tests
t1 <- t.test(tmp0$mentalizing,tmp0$between,alternative = "greater",paired = T)
```

```r
t2 <- t.test(tmp0$reward,tmp0$between,alternative = "greater",paired = T)

print(paste0("within-mentalizing versus between-network: ","t = ",
         round(t1$statistic,2),", p = ", round(t1$p.value,4)))
```

```
## [1] "within-mentalizing versus between-network: t = 9.96, p = 0"
```

```r
print(paste0("within-reward versus between-network: ","t = ",
         round(t2$statistic,2),", p = ", round(t2$p.value,4)))
```

```
## [1] "within-reward versus between-network: t = 2.13, p = 0.0191"
```

```r
round(p.adjust(c(t1$p.value,t2$p.value),method = "fdr"),4)
```

```
## [1] 0.0000 0.0191
```

Regression analysis on within- and between-network connectivity

```r
# the main effect of social interaction and interaction effects
flm1 <- flm_FC(mean_networks_FC, "mental_FC","Age")
flm2 <- flm_FC(mean_networks_FC, "reward_FC","Age")
flm3 <- flm_FC(mean_networks_FC, "between_FC","Age")
flm4 <- flm_FC(mean_control_FC, "motor_FC","Age")
flm5 <- flm_FC(mean_control_FC, "mirror_FC","Age")
flm6 <- flm_FC(mean_control_FC, "salience_FC","Age")

# summary of variance inflation factor
flm1[[1]]
```

```
##     socialP         age    mentalNM      gender     mean_FD          IQ
##   62.957537    1.348847    1.000000    1.055093    1.104513    1.055296
## socialP:age
##   63.169124
```

```r
vlf_all <- as.data.frame(matrix(0,  6, 7))
vlf_all <- rbind(flm1[[1]],flm2[[1]],flm3[[1]],flm4[[1]],flm5[[1]],flm6[[1]])
rownames(vlf_all) <- c("mentalization", "reward","between","motor","mirror","salience")
knitr::kable(vlf_all)
```

	socialP	age	mentalNM	gender	mean_FD	IQ	socialP:age
mentalization	62.95754	1.348847	1	1.055093	1.104513	1.055296	63.16912
reward	62.95754	1.358689	1	1.055093	1.104513	1.055296	63.17897
between	62.95754	1.392987	1	1.055093	1.104513	1.055296	63.21326
motor	62.95754	1.471361	1	1.055093	1.104513	1.055296	63.29164
mirror	62.95754	1.376867	1	1.055093	1.104513	1.055296	63.19714
salience	62.95754	1.365470	1	1.055093	1.104513	1.055296	63.18575

```r
# summary of the main regression results
flm_all <- as.data.frame(matrix(0,6,9))
flm_all<-rbind(cbind(as.data.frame(flm1[[3]])[2,2:4],as.data.frame(flm1[[3]])[3,2:4],
             as.data.frame(flm1[[3]])[8,2:4]),
             cbind(as.data.frame(flm2[[3]])[2,2:4],as.data.frame(flm2[[3]])[3,2:4],
                   as.data.frame(flm2[[3]])[8,2:4]),
             cbind(as.data.frame(flm3[[3]])[2,2:4],as.data.frame(flm3[[3]])[3,2:4],
                   as.data.frame(flm3[[3]])[8,2:4]),
             cbind(as.data.frame(flm4[[3]])[2,2:4],as.data.frame(flm4[[3]])[3,2:4],
                   as.data.frame(flm4[[3]])[8,2:4]),
             cbind(as.data.frame(flm5[[3]])[2,2:4],as.data.frame(flm5[[3]])[3,2:4],
                   as.data.frame(flm5[[3]])[8,2:4]),
             cbind(as.data.frame(flm6[[3]])[2,2:4],as.data.frame(flm6[[3]])[3,2:4],
                   as.data.frame(flm6[[3]])[8,2:4]))
colnames(flm_all) <- c("social_df","social_F-value","social_p-value",
               "age_df","age_F-value","age_p-value",
               "social*age_df","social*age_F-value","social*age_p-value")
rownames(flm_all) <- c("mentalization", "reward","between","motor","mirror","salience")

knitr::kable(flm_all)
```

	social_df	social_F-value	social_p-value	age_df	age_F-value	age_p-value	social*age_df	social*age_F-value	social*age_p-value
mentalization	147	0.2004686	0.6550020	45	0.0172539	0.8960807	147	9.401684	0.0025807
reward	147	0.0203753	0.8866893	45	0.3088200	0.5811578	147	7.130540	0.0084315
between	147	0.4598243	0.4987721	45	0.5023064	0.4821441	147	3.480606	0.0640849
motor	147	1.8618277	0.1744992	45	0.0942088	0.7603091	147	1.053926	0.3062910
mirror	147	0.6907475	0.4072582	45	1.0666883	0.3072136	147	2.860422	0.0929027
salience	147	0.4394368	0.5084325	45	3.7533138	0.0589949	147	5.869676	0.0166202

```r
# summary of the post-hoc regression results
flm_post <- as.data.frame(matrix(0,6,6))

flm_post<-rbind(cbind(as.data.frame(flm1[[4]])[2,2:4],as.data.frame(flm1[[5]])[2,2:4]),
        cbind(as.data.frame(flm2[[4]])[2,2:4],as.data.frame(flm2[[5]])[2,2:4]),
        cbind(as.data.frame(flm3[[4]])[2,2:4],as.data.frame(flm3[[5]])[2,2:4]),
        cbind(as.data.frame(flm4[[4]])[2,2:4],as.data.frame(flm4[[5]])[2,2:4]),
        cbind(as.data.frame(flm5[[4]])[2,2:4],as.data.frame(flm5[[5]])[2,2:4]),
        cbind(as.data.frame(flm6[[4]])[2,2:4],as.data.frame(flm6[[5]])[2,2:4]))

colnames(flm_post) <- c("P_df","P_F-value","P_p-value","C_df","C_F-value","C_p-value")
rownames(flm_post) <- c("mentalization", "reward","between","motor","mirror","salience")

knitr::kable(flm_post)
```

	P_df	P_F-value	P_p-value	C_df	C_F-value	C_p-value
mentalization	45	1.4060972	0.2419291	45	1.6836125	0.2010576
reward	45	0.3563243	0.5535464	45	3.2473667	0.0782393
between	45	0.0433606	0.8359874	45	2.1716313	0.1475384
motor	45	0.0506730	0.8229154	45	0.7498698	0.3911110

	P_df	P_F-value	P_p-value	C_df	C_F-value	C_p-value
mirror	45	2.5282886	0.1188233	45	0.0388680	0.8445982
salience	45	5.6805600	0.0214349	45	0.7549965	0.3895098

```r
# correlation with age
mean_FC <- "mental_FC"
cor.test(mean_networks_FC[mean_networks_FC$social == "P", mean_FC],
     mean_networks_FC[mean_networks_FC$social == "P", "age"])
```

```
##
##  Pearson's product-moment correlation
##
## data:  mean_networks_FC[mean_networks_FC$social == "P", mean_FC] and mean_networks_FC[mean_networks_F
## t = 1.5069, df = 98, p-value = 0.1351
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.0473314  0.3369444
## sample estimates:
##       cor
## 0.1504857
```

```r
cor.test(mean_networks_FC[mean_networks_FC$social == "C", mean_FC],
     mean_networks_FC[mean_networks_FC$social == "C", "age"])
```

```
##
##  Pearson's product-moment correlation
##
## data:  mean_networks_FC[mean_networks_FC$social == "C", mean_FC] and mean_networks_FC[mean_networks_F
## t = -1.6531, df = 98, p-value = 0.1015
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.34981164  0.03276754
## sample estimates:
##        cor
## -0.1647107
```

```r
mean_FC <- "reward_FC"
cor.test(mean_networks_FC[mean_networks_FC$social == "P", mean_FC],
     mean_networks_FC[mean_networks_FC$social == "P", "age"])
```

```
##
##  Pearson's product-moment correlation
##
## data:  mean_networks_FC[mean_networks_FC$social == "P", mean_FC] and mean_networks_FC[mean_networks_F
## t = 0.73604, df = 98, p-value = 0.4635
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.1240786  0.2666808
## sample estimates:
##        cor
## 0.07414653
```

```r
cor.test(mean_networks_FC[mean_networks_FC$social == "C", mean_FC],
     mean_networks_FC[mean_networks_FC$social == "C", "age"])
```

```
##
##  Pearson's product-moment correlation
##
## data:  mean_networks_FC[mean_networks_FC$social == "C", mean_FC] and mean_networks_FC[mean_networks_F
## t = -2.114, df = 98, p-value = 0.03705
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.38929113 -0.01295517
## sample estimates:
##        cor
## -0.2088419
```

```r
mean_FC <- "between_FC"
cor.test(mean_networks_FC[mean_networks_FC$social == "P", mean_FC],
     mean_networks_FC[mean_networks_FC$social == "P", "age"])
```

```
##
##  Pearson's product-moment correlation
##
## data:  mean_networks_FC[mean_networks_FC$social == "P", mean_FC] and mean_networks_FC[mean_networks_F
## t = 0.24697, df = 98, p-value = 0.8054
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.1723224  0.2202789
## sample estimates:
##        cor
## 0.02493985
```

```r
cor.test(mean_networks_FC[mean_networks_FC$social == "C", mean_FC],
     mean_networks_FC[mean_networks_FC$social == "C", "age"])
```

```
##
##  Pearson's product-moment correlation
##
## data:  mean_networks_FC[mean_networks_FC$social == "C", mean_FC] and mean_networks_FC[mean_networks_F
## t = -1.7433, df = 98, p-value = 0.08441
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.35766798  0.02379467
## sample estimates:
##       cor
## -0.173434
```

```r
mean_FC <-  "salience_FC"
cor.test(mean_control_FC[mean_control_FC$social == "P", mean_FC],
     mean_control_FC[mean_control_FC$social == "P", "age"])
```

```
##
```

```
##  Pearson's product-moment correlation
##
## data:  mean_control_FC[mean_control_FC$social == "P", mean_FC] and mean_control_FC[mean_control_FC$so
## t = 3.1021, df = 98, p-value = 0.00251
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1090068 0.4679560
## sample estimates:
##       cor
## 0.2990225
```

```
cor.test(mean_control_FC[mean_control_FC$social == "C", mean_FC],
     mean_control_FC[mean_control_FC$social == "C", "age"])
```

```
##
##  Pearson's product-moment correlation
##
## data:  mean_control_FC[mean_control_FC$social == "C", mean_FC] and mean_control_FC[mean_control_FC$so
## t = 0.97311, df = 98, p-value = 0.3329
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.1005225  0.2886979
## sample estimates:
##        cor
## 0.09782716
```

```
mean_FC <-  "mirror_FC"
cor.test(mean_control_FC[mean_control_FC$social == "P", mean_FC],
     mean_control_FC[mean_control_FC$social == "P", "age"])
```

```
##
##  Pearson's product-moment correlation
##
## data:  mean_control_FC[mean_control_FC$social == "P", mean_FC] and mean_control_FC[mean_control_FC$so
## t = 1.9304, df = 98, p-value = 0.05645
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.005220282  0.373761293
## sample estimates:
##       cor
## 0.1913941
```

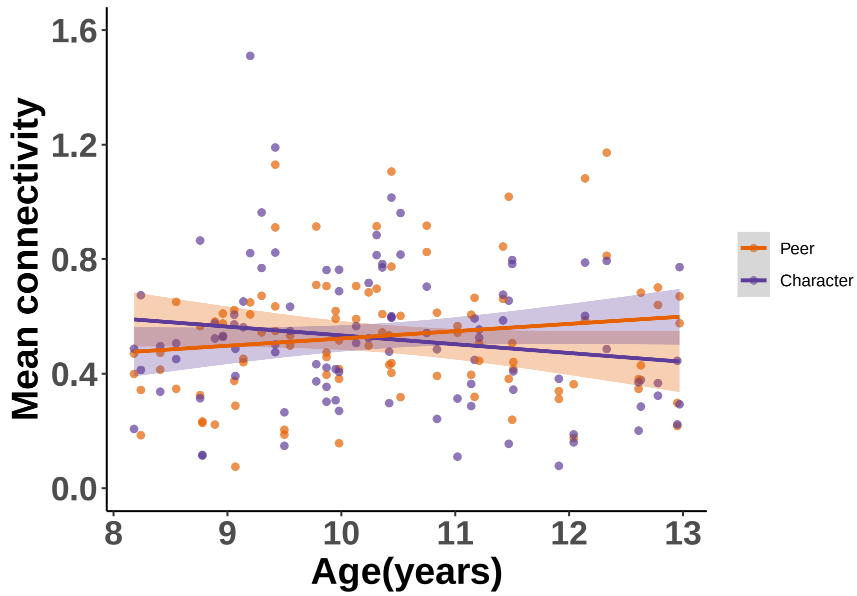
```
cor.test(mean_control_FC[mean_control_FC$social == "C", mean_FC],
     mean_control_FC[mean_control_FC$social == "C", "age"])
```

```
##
##  Pearson's product-moment correlation
##
## data:  mean_control_FC[mean_control_FC$social == "C", mean_FC] and mean_control_FC[mean_control_FC$so
## t = 0.22014, df = 98, p-value = 0.8262
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
```

```
## -0.1749499 0.2176996
## sample estimates:
##        cor
## 0.02223217
```

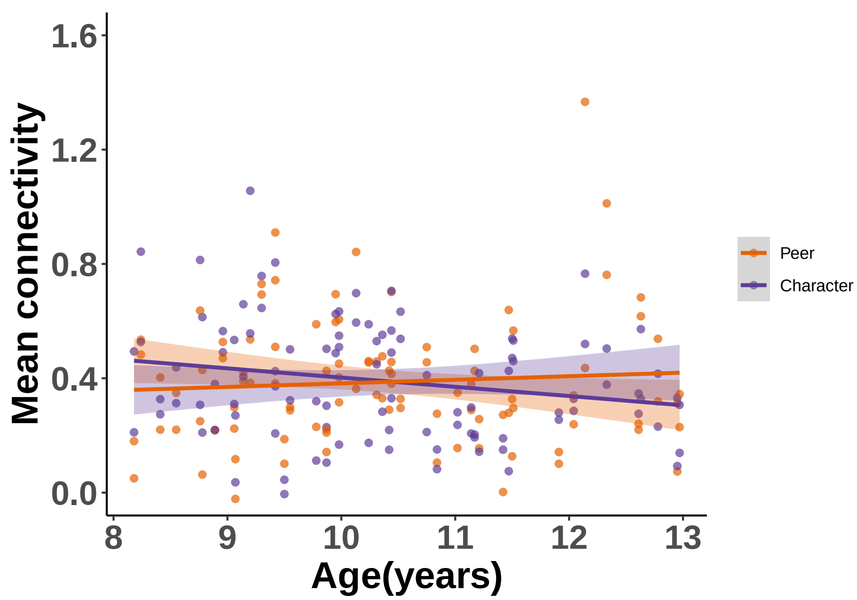
```r
# scatterplots for interaction effects of social interaction and age
FC_scatterplot(mean_networks_FC,"age","mental_FC","Age(years)","Age")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



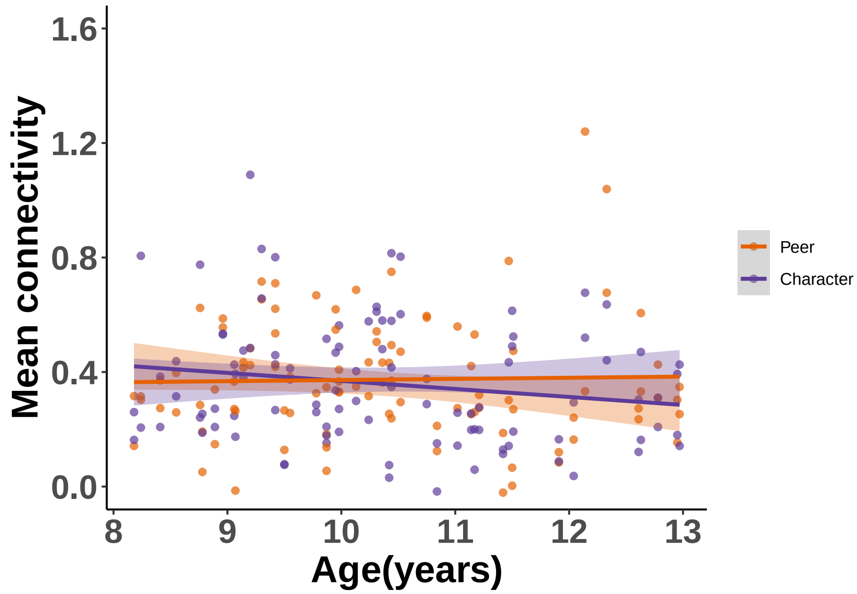

```r
FC_scatterplot(mean_networks_FC,"age","reward_FC","Age(years)","Age")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



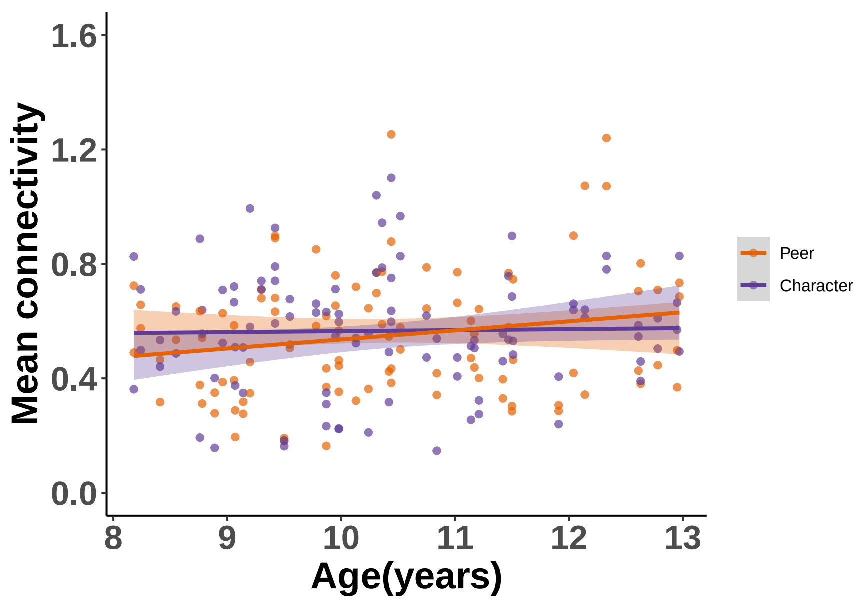
```r
FC_scatterplot(mean_networks_FC,"age","between_FC","Age(years)","Age")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



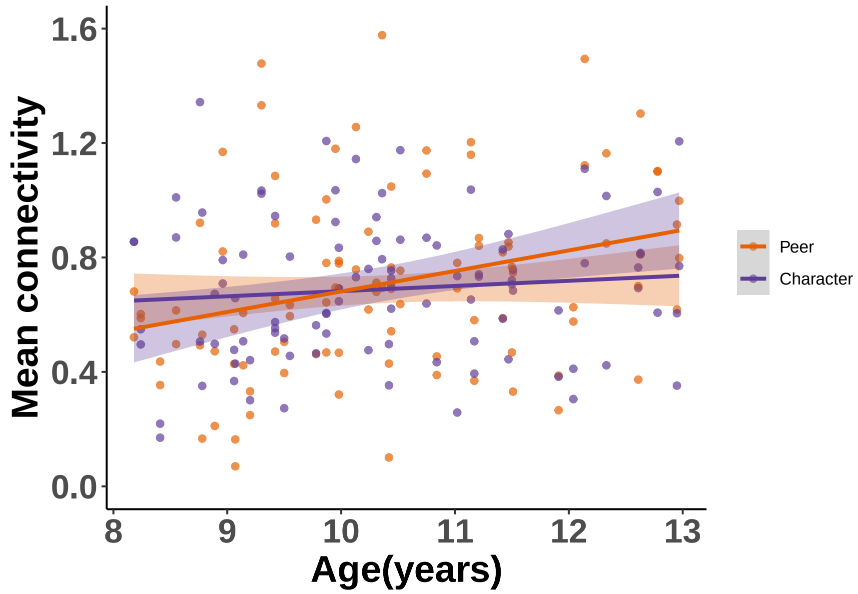

```r
FC_scatterplot(mean_control_FC,"age","mirror_FC","Age(years)","Age")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

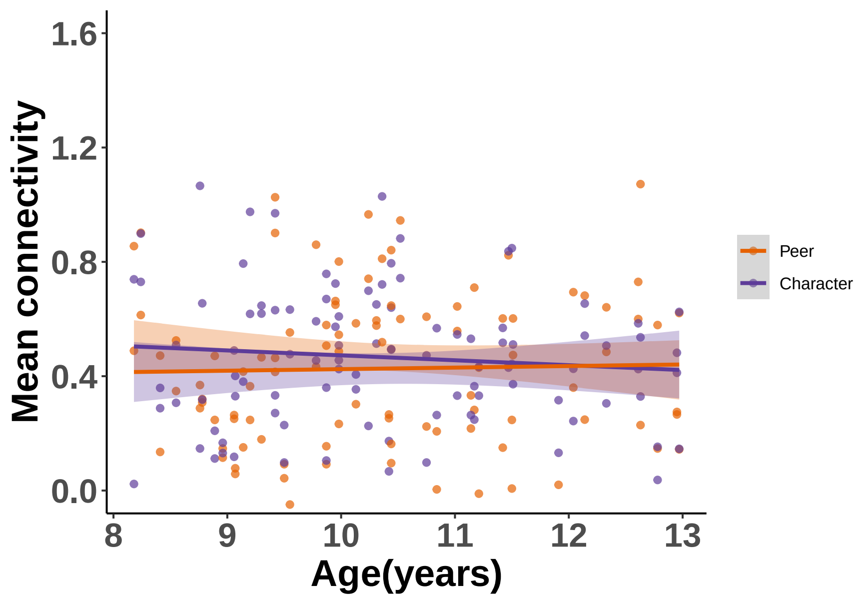



```r
FC_scatterplot(mean_control_FC,"age","salience_FC","Age(years)","Age")
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
FC_scatterplot(mean_control_FC,"age","motor_FC","Age(years)","Age")

## `geom_smooth()` using formula 'y ~ x'
```



FC differences between subjects at upper vs. lower quartile of age

```
# connectivity averaged across collapsed conditions
tmp <- as.data.frame(matrix(0,100,0))
tmp$subj <- rep(mean_networks_FC$Subject[!duplicated(mean_networks_FC$Subject)],2)
tmp$age <- rep(mean_networks_FC$age[!duplicated(mean_networks_FC$Subject)],2)
tmp$social <- rep(c("P","C"),each = 50)
```

```
tmp$mental_FC <- c((mean_networks_FC$mental_FC[mean_networks_FC$conditions == "PM"] +
            mean_networks_FC$mental_FC[mean_networks_FC$conditions == "PNM"])/2,
           (mean_networks_FC$mental_FC[mean_networks_FC$conditions == "CM"] +
            mean_networks_FC$mental_FC[mean_networks_FC$conditions == "CNM"])/2)

tmp$reward_FC <- c((mean_networks_FC$reward_FC[mean_networks_FC$conditions == "PM"] +
            mean_networks_FC$reward_FC[mean_networks_FC$conditions=="PNM"])/2,
           (mean_networks_FC$reward_FC[mean_networks_FC$conditions == "CM"] +
            mean_networks_FC$reward_FC[mean_networks_FC$conditions == "CNM"])/2)

tmp$between_FC <- c((mean_networks_FC$between_FC[mean_networks_FC$conditions == "PM"] +
           mean_networks_FC$between_FC[mean_networks_FC$conditions == "PNM"])/2,
           (mean_networks_FC$between_FC[mean_networks_FC$conditions == "CM"] +
           mean_networks_FC$between_FC[mean_networks_FC$conditions == "CNM"])/2)

# FC differences: upper vs. lower quantile of age
age <- summary(variables_all$age[!duplicated(variables_all$Subject)])
print(paste0("The upper quartile of age is ", round(age["3rd Qu."],2)))
```

```
## [1] "The upper quartile of age is 11.37"
```

```
print(paste0("The lower quartile of age is ", age["1st Qu."]))
```

```
## [1] "The lower quartile of age is 9.33"
```

```
networks <- c("mental_FC","reward_FC","between_FC")

contrast_FC_age <- as.data.frame(matrix(0,3,3))
colnames(contrast_FC_age)  <- c("network","older children","younger children")

k <-0
for (net in networks) {
    k <- k+1
    tt1 <- t.test(tmp[tmp$age > as.numeric(age["3rd Qu."]) & tmp$social == "P",net],
              tmp[tmp$age > as.numeric(age["3rd Qu."]) & tmp$social == "C",net],
              paired = T)

    tt2 <- t.test(tmp[tmp$age < as.numeric(age["1st Qu."]) & tmp$social == "P",net],
              tmp[tmp$age < as.numeric(age["1st Qu."]) & tmp$social == "C",net],
              paired = T)

    contrast_FC_age[k, 1] <- net
    contrast_FC_age[k, 2:3] <- c(paste0("t=", round(tt1$statistic,3),", p=",
                        round(tt1$p.value,3)),
                     paste0("t=", round(tt2$statistic,3),", p=",
                           round(tt2$p.value,3)))

}

    knitr::kable(contrast_FC_age)
```

network	older children	younger children
mental_FC	t=1.928, p=0.078	t=-2.331, p=0.038
reward_FC	t=0.928, p=0.372	t=-4.552, p=0.001
between_FC	t=0.677, p=0.511	t=-2.299, p=0.04

ROI analysis

```r
# load data
mental_node_FC <- read.table(here("data/mental_nodes_FC.txt"), header = T)
reward_node_FC <- read.table(here("data/reward_nodes_FC.txt"), header = T)
between_node_FC <- read.table(here("data/between_nodes_FC.txt"), header = T)
salience_node_FC <- read.table(here("data/salience_nodes_FC.txt"), header = T)
mirror_node_FC <- read.table(here("data/mirror_nodes_FC.txt"), header = T)
motor_node_FC <- read.table(here("data/motor_nodes_FC.txt"), header = T)

# mental network
ROIs <- c("dmPFC","vmPFC","PCC","RTPJ","LTPJ","RATL","LATL")
colnames(mental_node_FC)
```

```
##  [1] "Subject"    "conditions" "social"     "mental"     "age"
##  [6] "gender"     "mean_FD"    "IQ"         "RT"         "Accuracy"
## [11] "dmPFC"      "vmPFC"      "PCC"        "RTPJ"       "LTPJ"
## [16] "RATL"       "LATL"
```

```r
flm_sum1 <- as.data.frame(matrix(0, 7, 2))
rownames(flm_sum1) <- ROIs
colnames(flm_sum1) <- c("F-value", "p-value")

k <- 0
for (ROI in ROIs) {
    k <- k + 1
    flm_ROI <- flm_FC(mental_node_FC, ROI,"Age")

    flm_sum1[k, ] <- c(flm_ROI[[3]]$`F-value`[4], flm_ROI[[3]]$`p-value`[4])
}

# reward network
ROIs <- c("LOFC","RVFC","ACC","LVS","RVS","LAmygdala","RAmygdala" )
colnames(reward_node_FC)
```

```
##  [1] "Subject"    "conditions" "social"     "mental"     "age"
##  [6] "gender"     "mean_FD"    "IQ"         "RT"         "Accuracy"
## [11] "LOFC"       "RVFC"       "ACC"        "LVS"        "RVS"
## [16] "LAmygdala"  "RAmygdala"
```

```r
flm_sum2 <- as.data.frame(matrix(0, 7, 2))
rownames(flm_sum2) <- ROIs
colnames(flm_sum2) <- c("F-value", "p-value")
```

```r
k <- 0
for (ROI in ROIs) {
    k <- k + 1
    flm_ROI <- flm_FC(reward_node_FC, ROI,"Age")

    flm_sum2[k, ] <- c(flm_ROI[[3]]$`F-value`[4], flm_ROI[[3]]$`p-value`[4])

}

# between noodes:
ROIs <- c("dmPFC","vmPFC","PCC","RTPJ","LTPJ","RATL","LATL","LOFC","RVFC","ACC",
      "LVS","RVS","LAmygdala","RAmygdala")

flm_sum3 <- as.data.frame(matrix(0, 14, 2))
rownames(flm_sum3) <- ROIs
colnames(flm_sum3) <- c("F-value", "p-value")

k <- 0
for (ROI in ROIs) {
    k <- k + 1
    flm_ROI <- flm_FC(between_node_FC, ROI,"Age")

    flm_sum3[k, ] <- c(flm_ROI[[3]]$`F-value`[8], flm_ROI[[3]]$`p-value`[8])

}

# significant interaction effects
knitr::kable(rbind.data.frame(flm_sum1[flm_sum1$`p-value` < 0.05,],
                  flm_sum2[flm_sum2$`p-value` < 0.05,],
                  flm_sum3[flm_sum3$`p-value` < 0.05,]))
```

	F-value	p-value
dmPFC	4.907437	0.0282794
ACC	7.999469	0.0053325
dmPFC1	5.798029	0.0172835
LTPJ	5.173727	0.0243779
LATL	5.499957	0.0203539
ACC1	5.817598	0.0170996
LVS	4.788573	0.0302281

```r
p.adjust(c(flm_sum1$`p-value`,flm_sum2$`p-value`,flm_sum3$`p-value`))
```

```
##  [1] 0.6504256 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000
##  [8] 1.0000000 1.0000000 0.1493110 1.0000000 1.0000000 1.0000000 1.0000000
## [15] 0.4616898 1.0000000 1.0000000 1.0000000 0.5850687 1.0000000 0.5088469
## [22] 1.0000000 1.0000000 0.4616898 0.6650178 1.0000000 1.0000000 1.0000000
```

```r
## regions within specificity and control networks
# motor nodes
colnames(motor_node_FC)[1] <- "Subject"
```

```r
ROIs <- c("X3","X4","X5")
flm_sum <- as.data.frame(matrix(0, 3, 2))
rownames(flm_sum) <- ROIs
colnames(flm_sum) <- c("F-value", "p-value")

k <- 0
for (ROI in ROIs) {
    k <- k + 1
    flm_ROI <- flm_FC(motor_node_FC, ROI,"Age")
    flm_sum[k, ] <- c(flm_ROI[[3]]$`F-value`[8], flm_ROI[[3]]$`p-value`[8])

}

flm_sum[flm_sum$`p-value` < 0.05,]
```

```
## [1] F-value p-value
## <0 rows> (or 0-length row.names)
```

```r
# mirror nodes
colnames(mirror_node_FC)[1] <- "Subject"

ROIs <- c("X3","X4","X5","X6","X7","X8","X9","X10","X11","X12")
flm_sum <- as.data.frame(matrix(0, 10, 2))
rownames(flm_sum) <- ROIs
colnames(flm_sum) <- c("F-value", "p-value")

k <- 0
for (ROI in ROIs) {
    k <- k + 1
    flm_ROI <- flm_FC(mirror_node_FC, ROI,"Age")

    flm_sum[k, ] <- c(flm_ROI[[3]]$`F-value`[8], flm_ROI[[3]]$`p-value`[8])

}

flm_sum[flm_sum$`p-value` < 0.05,]
```

```
## [1] F-value p-value
## <0 rows> (or 0-length row.names)
```

```r
# salience nodes
colnames(salience_node_FC)[1] <- "Subject"

ROIs <- c("RdACC", "LaInsula", "RaInsula")
flm_sum <- as.data.frame(matrix(0, 3, 2))
rownames(flm_sum) <- ROIs
colnames(flm_sum) <- c("F-value", "p-value")

k <- 0
for (ROI in ROIs) {
    k <- k + 1
    flm_ROI <- flm_FC(salience_node_FC, ROI,"Age")
```

```r
    flm_sum[k, ] <- c(flm_ROI[[3]]$`F-value`[8], flm_ROI[[3]]$`p-value`[8])

}

flm_sum[flm_sum$`p-value` < 0.05,]
```

```
## [1] F-value p-value
## <0 rows> (or 0-length row.names)
```

Correlations between salience/mirror neuron and mentalizing/reward networks

```r
tmp0$mirror <- (mean_control_FC$mirror_FC[mean_control_FC$conditions == "PM"] +
            mean_control_FC$mirror_FC[mean_control_FC$conditions == "PNM"] +
            mean_control_FC$mirror_FC[mean_control_FC$conditions == "CM"] +
            mean_control_FC$mirror_FC[mean_control_FC$conditions == "CNM"])/4

tmp0$salience <- (mean_control_FC$salience_FC[mean_control_FC$conditions == "PM"] +
         mean_control_FC$salience_FC[mean_control_FC$conditions == "PNM"] +
         mean_control_FC$salience_FC[mean_control_FC$conditions == "CM"] +
         mean_control_FC$salience_FC[mean_control_FC$conditions == "CNM"])/4

# correlations between salience network and mentalizing/reward networks
cor.test(tmp0$mentalizing,tmp0$salience)
```

```
##
##  Pearson's product-moment correlation
##
## data:  tmp0$mentalizing and tmp0$salience
## t = 3.724, df = 48, p-value = 0.0005155
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.2247139 0.6642578
## sample estimates:
##       cor
## 0.4734481
```

```r
cor.test(tmp0$reward,tmp0$salience)
```

```
##
##  Pearson's product-moment correlation
##
## data:  tmp0$reward and tmp0$salience
## t = 4.3194, df = 48, p-value = 7.815e-05
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.2940001 0.7037641
## sample estimates:
##       cor
## 0.5290532
```

```r
# correlation between mirror neuron network and mentalizing network
cor.test(tmp0$mentalizing,tmp0$mirror)
```

```
##
##  Pearson's product-moment correlation
##
## data:  tmp0$mentalizing and tmp0$mirror
## t = 6.0831, df = 48, p-value = 1.872e-07
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4672542 0.7925798
## sample estimates:
##       cor
## 0.6597902
```

```r
cor.test(mean_networks_FC$mental_FC[mean_networks_FC$social == "P"],
     mean_control_FC$mirror_FC[mean_control_FC$social == "P"])
```

```
##
##  Pearson's product-moment correlation
##
## data:  mean_networks_FC$mental_FC[mean_networks_FC$social == "P"] and mean_control_FC$mirror_FC[mean_
## t = 9.1785, df = 98, p-value = 7.343e-15
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5579963 0.7730756
## sample estimates:
##       cor
## 0.6798972
```

```r
cor.test(mean_networks_FC$mental_FC[mean_networks_FC$social == "C"],
     mean_control_FC$mirror_FC[mean_control_FC$social == "C"])
```

```
##
##  Pearson's product-moment correlation
##
## data:  mean_networks_FC$mental_FC[mean_networks_FC$social == "C"] and mean_control_FC$mirror_FC[mean_
## t = 8.3404, df = 98, p-value = 4.739e-13
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5127933 0.7462865
## sample estimates:
##       cor
## 0.6443148
```

Brain-behavior correlations (RT, subjective reports)

```r
# regression analysis on RT
flm1 <- flm_FC(mean_networks_FC, "mental_FC","RT")
flm2 <- flm_FC(mean_networks_FC, "reward_FC","RT")
```

```r
flm3 <- flm_FC(mean_networks_FC, "between_FC","RT")

flm4 <- flm_FC(mean_control_FC, "motor_FC","RT")
flm5 <- flm_FC(mean_control_FC, "mirror_FC","RT")
flm6 <- flm_FC(mean_control_FC, "salience_FC","RT")

# summary of the main regression results
flm_all <- as.data.frame(matrix(0,6,9))
flm_all <- rbind(cbind(as.data.frame(flm1[[3]])[2,2:4],as.data.frame(flm1[[3]])[3,2:4],
             as.data.frame(flm1[[3]])[8,2:4]),
             cbind(as.data.frame(flm2[[3]])[2,2:4],as.data.frame(flm2[[3]])[3,2:4],
                   as.data.frame(flm2[[3]])[8,2:4]),
             cbind(as.data.frame(flm3[[3]])[2,2:4],as.data.frame(flm3[[3]])[3,2:4],
                   as.data.frame(flm3[[3]])[8,2:4]),
             cbind(as.data.frame(flm4[[3]])[2,2:4],as.data.frame(flm4[[3]])[3,2:4],
                   as.data.frame(flm4[[3]])[8,2:4]),
             cbind(as.data.frame(flm5[[3]])[2,2:4],as.data.frame(flm5[[3]])[3,2:4],
                   as.data.frame(flm5[[3]])[8,2:4]),
             cbind(as.data.frame(flm6[[3]])[2,2:4],as.data.frame(flm6[[3]])[3,2:4],
                   as.data.frame(flm6[[3]])[8,2:4]))

colnames(flm_all) <- c("social_df","social_F-value","social_p-value",
               "RT_df","RT_F-value","RT_p-value",
               "social*RT_df","social*RT_F-value","social*RT_p-value")
rownames(flm_all) <- c("mentalization", "reward","between","motor","mirror","salience")

knitr::kable(flm_all)
```

	social_df	social_F-value	social_p-value	RT_df	RT_F-value	RT_p-value	social*RT_df	social*RT_F-value	social*RT_p-value
mentalization	146	0.2057666	0.6507795	146	0.5194799	0.4722161	146	13.7188173	0.0003005
reward	146	0.0203308	0.8868136	146	0.0000590	0.9938815	146	7.5508297	0.0067550
between	146	0.4724018	0.4929746	146	0.0801034	0.7775582	146	8.3190439	0.0045175
motor	146	1.8800552	0.1724318	146	0.0137008	0.9069810	146	3.0118387	0.0847684
mirror	146	0.7182700	0.3980992	146	2.5447947	0.1128207	146	7.4875435	0.0069843
salience	146	0.4228982	0.5165164	146	0.7242081	0.3961606	146	0.6449687	0.4232228

```r
# summary of the post-hoc regression results
flm_post <- as.data.frame(matrix(0,6,6))

flm_post<-rbind(cbind(as.data.frame(flm1[[4]])[2,2:4],as.data.frame(flm1[[5]])[2,2:4]),
        cbind(as.data.frame(flm2[[4]])[2,2:4],as.data.frame(flm2[[5]])[2,2:4]),
        cbind(as.data.frame(flm3[[4]])[2,2:4],as.data.frame(flm3[[5]])[2,2:4]),
        cbind(as.data.frame(flm4[[4]])[2,2:4],as.data.frame(flm4[[5]])[2,2:4]),
        cbind(as.data.frame(flm5[[4]])[2,2:4],as.data.frame(flm5[[5]])[2,2:4]),
        cbind(as.data.frame(flm6[[4]])[2,2:4],as.data.frame(flm6[[5]])[2,2:4]))

colnames(flm_post) <- c("P_df","P_F-value","P_p-value","C_df","C_F-value","C_p-value")
rownames(flm_post) <- c("mentalization", "reward","between","motor","mirror","salience")

knitr::kable(flm_post)
```

	P_df	P_F-value	P_p-value	C_df	C_F-value	C_p-value
mentalization	48	4.4741657	0.0396275	48	2.1271212	0.1512252
reward	48	1.6147360	0.2099517	48	2.6460498	0.1103556
between	48	0.8605380	0.3582295	48	4.3070943	0.0433358
motor	48	0.1918808	0.6633209	48	0.8001456	0.3755134
mirror	48	4.3632759	0.0420483	48	0.0464303	0.8303086
salience	48	0.9769435	0.3279110	48	0.1808576	0.6725378

```
# correlations between RT and FC averaged across conditions
tmp0$RT <- (mean_networks_FC$RT[mean_networks_FC$conditions=="PM"] +
            mean_networks_FC$RT[mean_networks_FC$conditions=="PNM"] +
            mean_networks_FC$RT[mean_networks_FC$conditions=="CM"] +
            mean_networks_FC$RT[mean_networks_FC$conditions=="CNM"])/4

# correlation between RT within-mentalizing, reward, and between-networks
cor.test(tmp0$mentalizing, tmp0$RT)
```

```
##
##  Pearson's product-moment correlation
##
## data:  tmp0$mentalizing and tmp0$RT
## t = -0.16419, df = 48, p-value = 0.8703
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.3000616  0.2563456
## sample estimates:
##         cor
## -0.02369269
```

```
cor.test(tmp0$reward, tmp0$RT)
```

```
##
##  Pearson's product-moment correlation
##
## data:  tmp0$reward and tmp0$RT
## t = 0.2718, df = 48, p-value = 0.7869
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.2417845  0.3141216
## sample estimates:
##        cor
## 0.03920144
```

```
cor.test(tmp0$between, tmp0$RT)
```

```
##
##  Pearson's product-moment correlation
##
## data:  tmp0$between and tmp0$RT
## t = 0.86768, df = 48, p-value = 0.3899
## alternative hypothesis: true correlation is not equal to 0
```

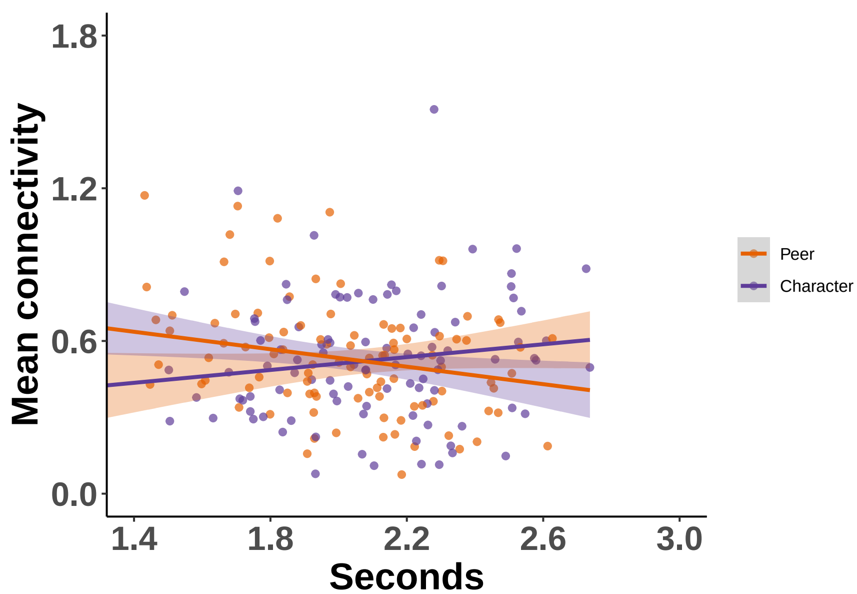
```
## 95 percent confidence interval:
##  -0.1596003  0.3891549
## sample estimates:
##      cor
## 0.124268
```

```
# correlation between RT and mirror neuron networks
tmp0$mirror <- (mean_control_FC$mirror_FC[mean_control_FC$conditions == "PM"] +
            mean_control_FC$mirror_FC[mean_control_FC$conditions == "PNM"] +
            mean_control_FC$mirror_FC[mean_control_FC$conditions == "CM"] +
            mean_control_FC$mirror_FC[mean_control_FC$conditions == "CNM"])/4

cor.test(tmp0$mirror, tmp0$RT)
```

```
##
##  Pearson's product-moment correlation
##
## data:  tmp0$mirror and tmp0$RT
## t = -0.84355, df = 48, p-value = 0.4031
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.3862183  0.1629665
## sample estimates:
##        cor
## -0.1208638
```

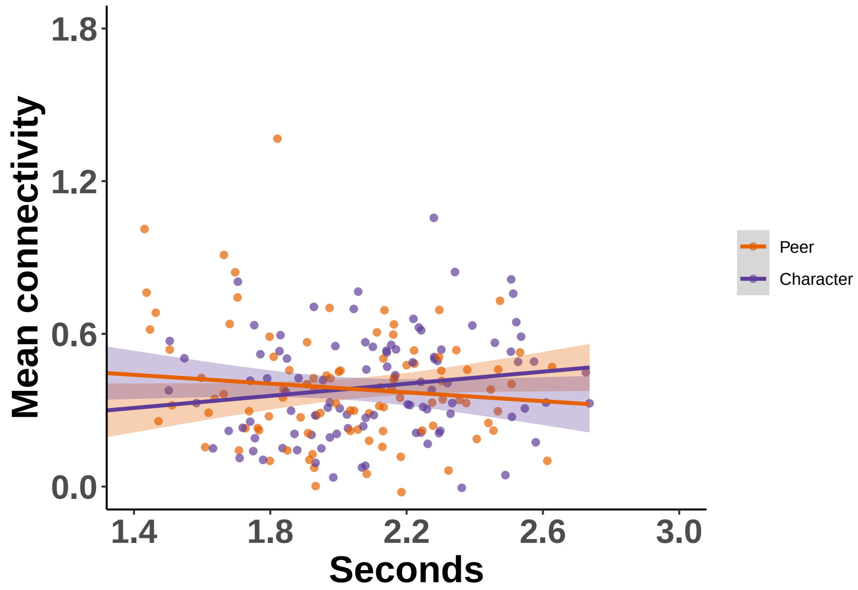
```
# scatter plot
FC_scatterplot(mean_networks_FC,"RT","mental_FC","Seconds","RT")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



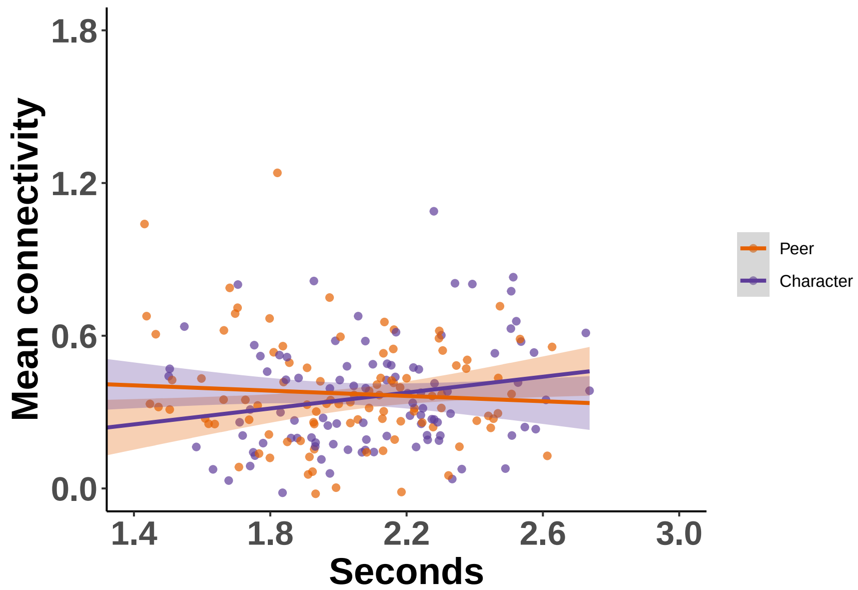
```
FC_scatterplot(mean_networks_FC,"RT","reward_FC","Seconds","RT")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



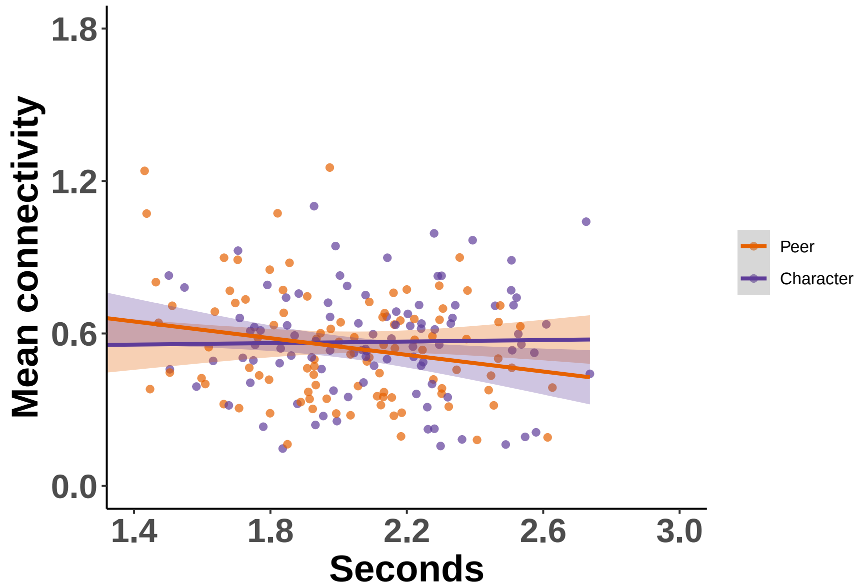

```
FC_scatterplot(mean_networks_FC,"RT","between_FC","Seconds","RT")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



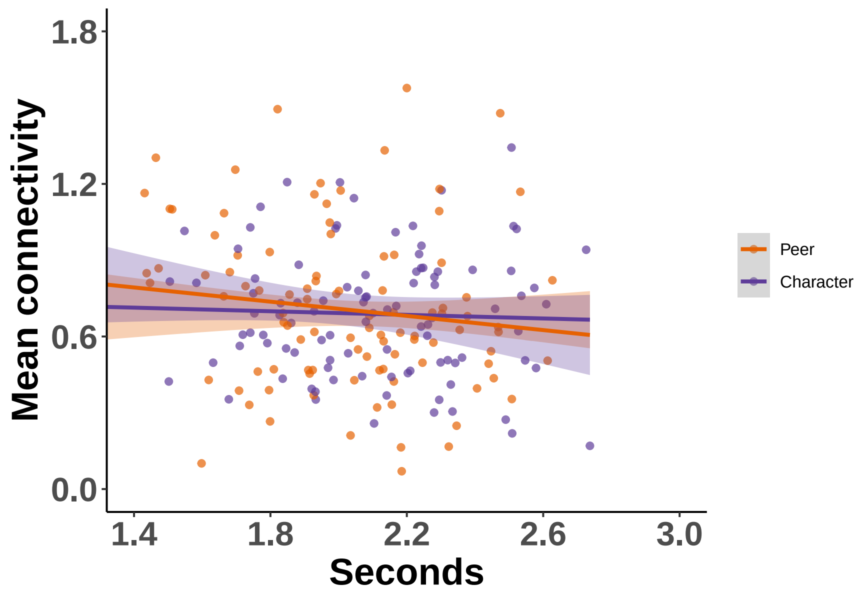

```
FC_scatterplot(mean_control_FC,"RT","mirror_FC","Seconds","RT")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



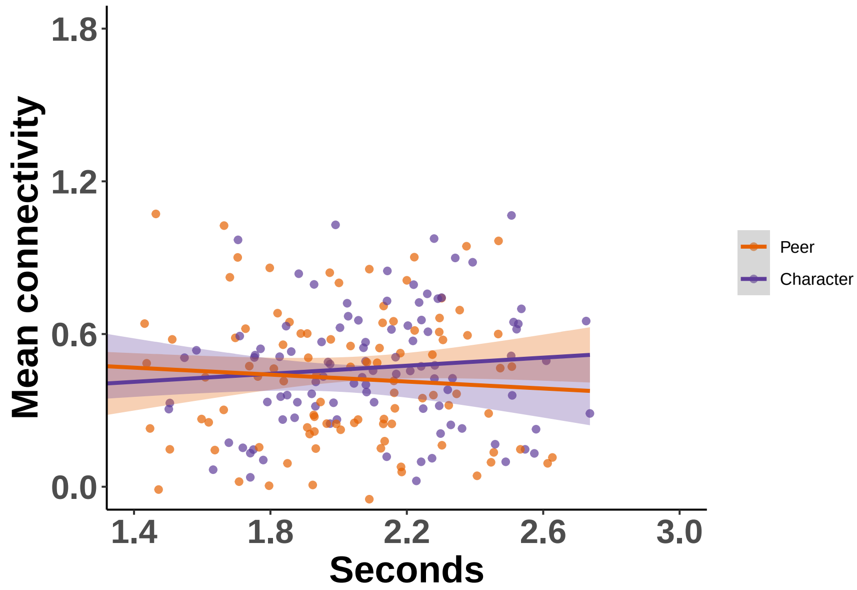

```
FC_scatterplot(mean_control_FC,"RT","salience_FC","Seconds","RT")
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
FC_scatterplot(mean_control_FC,"RT","motor_FC","Seconds","RT")
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
# regression analysis on subjective reports
posttest_long <- read.table(here("data/posttest_N50.txt"), header = T)
posttest_long$Subject <- gsub("_","",posttest_long$Subj)

mean_networks_FC$social[mean_networks_FC$social == "P"] <- "Peer"
mean_networks_FC$social[mean_networks_FC$social == "C"] <- "Character"

mean_networks_FC_new <- merge(mean_networks_FC,posttest_long, by = c("Subject","social"))

colnames(mean_networks_FC_new)
```

```
##  [1] "Subject"             "social"              "conditions"
##  [4] "mental"              "age"                 "gender"
##  [7] "mean_FD"             "IQ"                  "Accuracy"
## [10] "RT"                  "motion_byCondition" "mental_FC"
## [13] "reward_FC"           "between_FC"          "Subj"
## [16] "Age"                 "like"                "attention"
## [19] "hardguess"           "likeguess"           "wantsee"
## [22] "agreed"
```

```
sum_FC_report <- as.data.frame(matrix(0,18,11))
k <- 0
for (mean_FC in c("mental_FC","reward_FC","between_FC")) {

    for (report in c("like","attention","likeguess","hardguess","wantsee",
             "agreed")) {
        k <- k+1
        fl <- as.formula(paste0(mean_FC, "~ social*", report," + mental + gender +
                 mean_FD+IQ"))
        flm0 <- anova(lme(fl, random = ~1|Subject, data = mean_networks_FC_new))

    sum_FC_report[k,1:2] <- c(mean_FC,report)
    sum_FC_report[k,3:5] <- flm0[2,2:4]
```

```
    sum_FC_report[k,6:8] <- flm0[3,2:4]
    sum_FC_report[k,9:11] <- flm0[7,2:4]
    }
}


p.adjust(sum_FC_report$V5, method = "fdr")
```

```
##  [1] 0.8896709 0.8896709 0.8896709 0.8896709 0.8896709 0.8896709 0.8896709
##  [8] 0.8896709 0.8896709 0.8896709 0.8896709 0.8896709 0.8896709 0.8896709
## [15] 0.8896709 0.8896709 0.8896709 0.8896709
```

```
p.adjust(sum_FC_report$V8, method = "fdr")
```

```
##  [1] 0.1386560 0.1386560 0.7724179 0.7416456 0.9584689 0.6996508 0.7416456
##  [8] 0.2956724 0.9243718 0.6996508 0.8944720 0.9584689 0.2944737 0.2944737
## [15] 0.6996508 0.6996508 0.6996508 0.6996508
```

```
p.adjust(sum_FC_report$V11, method = "fdr")
```

```
##  [1] 0.9671393 0.9671393 0.9671393 0.9671393 0.9671393 0.9671393 0.4657774
##  [8] 0.4657774 0.4657774 0.4930539 0.4657774 0.4657774 0.4657774 0.4657774
## [15] 0.4657774 0.4657774 0.4657774 0.4657774
```