

# Single-subject grey matter networks in AD

2024-05-10

## Set up

```
packages <- c("here", "readxl", "dplyr", "ggplot2", "psych", "lmerTest", "WriteXLS",
              "lme4", "emmeans")

lapply(packages, library, character.only = TRUE)

## [[1]]
## [1] "here"      "stats"      "graphics"   "grDevices" "utils"      "datasets"
## [7] "methods"   "base"
##
## [[2]]
## [1] "readxl"    "here"       "stats"      "graphics"   "grDevices" "utils"
## [7] "datasets"  "methods"    "base"
##
## [[3]]
## [1] "dplyr"     "readxl"    "here"       "stats"      "graphics"   "grDevices"
## [7] "utils"     "datasets"  "methods"    "base"
##
## [[4]]
## [1] "ggplot2"   "dplyr"     "readxl"    "here"       "stats"      "graphics"
## [7] "grDevices" "utils"     "datasets"  "methods"    "base"
##
## [[5]]
## [1] "psych"     "ggplot2"   "dplyr"     "readxl"    "here"       "stats"
## [7] "graphics"  "grDevices" "utils"     "datasets"  "methods"    "base"
##
## [[6]]
## [1] "lmerTest"  "lme4"      "Matrix"    "psych"     "ggplot2"   "dplyr"
## [7] "readxl"    "here"      "stats"     "graphics"  "grDevices" "utils"
## [13] "datasets"  "methods"   "base"
##
## [[7]]
## [1] "WriteXLS"  "lmerTest"  "lme4"      "Matrix"    "psych"     "ggplot2"
## [7] "dplyr"     "readxl"    "here"      "stats"     "graphics"  "grDevices"
## [13] "utils"     "datasets"  "methods"   "base"
##
## [[8]]
## [1] "WriteXLS"  "lmerTest"  "lme4"      "Matrix"    "psych"     "ggplot2"
## [7] "dplyr"     "readxl"    "here"      "stats"     "graphics"  "grDevices"
## [13] "utils"     "datasets"  "methods"   "base"
##
```

```
## [[9]]
## [1] "emmeans" "WriteXLS" "lmerTest" "lme4" "Matrix" "psych"
## [7] "ggplot2" "dplyr" "readxl" "here" "stats" "graphics"
## [13] "grDevices" "utils" "datasets" "methods" "base"
```

## Load data

```
sw_all_info <- read_xlsx("data/sw_all_info.xlsx", sheet = 1)

sw_all_info <- as.data.frame(sw_all_info)

dim(sw_all_info)
```

```
## [1] 488 31
```

```
colnames(sw_all_info)
```

```
## [1] "Subject.ID" "Age"
## [3] "Gender" "Research.Group"
## [5] "Education" "Global.CDR"
## [7] "MMSE.Total.Score" "MoCA.Total"
## [9] "MoCA.MIS" "Description"
## [11] "Imaging.Protocol" "Image.ID"
## [13] "nvol" "number.of.nodes"
## [15] "mean.degree" "mean.custering_coef"
## [17] "fp.rate" "mean.pathlength"
## [19] "rand.pathlength" "gamma"
## [21] "lambda" "sigma"
## [23] "rand.mean.custering_coef" "mean.intensity.roi"
## [25] "std.intensity.roi" "std.degree"
## [27] "sd.custering_coef" "rand.sd.custering_coef"
## [29] "connectivity.density" "rand.sd.pathlength"
## [31] "TIV"
```

## Demographic and clinical information

```
# convert to numeric format
sw_all_info[,c("Age", "Education", "Global.CDR", "MMSE.Total.Score", "MoCA.Total")] <-
  sapply(sw_all_info[,c("Age", "Education", "Global.CDR", "MMSE.Total.Score",
    "MoCA.Total")], as.numeric)

# check data
#View(sw_all_info[sw_all_info$MoCA.Total < 9, c("Subject.ID", "Research.Group",
#      "Global.CDR", "MMSE.Total.Score",
#      "MoCA.Total", "MoCA.MIS")])

#View(sw_all_info[sw_all_info$Research.Group == "AD", c("Subject.ID", "Research.Group",
```

```

#                               "Global.CDR", "MMSE.Total.Score",
#                               "MoCA.Total", "MoCA.MIS"]])

#sw_all_info$MoCA.Total[sw_all_info$MoCA.Total == 0] <- NA

# summary table
demo_info <- as.data.frame(matrix(0,6,7))

colnames(demo_info) <- c("CN_mean", "CN_range", "MCI_mean", "MCI_range", "AD_mean",
                        "AD_range", "p value")

rownames(demo_info) <- c("Gender", "Age", "Education", "CDR", "MMSE", "MoCA")

# gender: chi-squared test
gender_gr <- table(sw_all_info$Research.Group, sw_all_info$Gender)

chisq.test(gender_gr)$p.value

```

```
## [1] 0.1900976
```

```

demo_info["Gender", 1] <- paste0(gender_gr["CN",2] ,"/",gender_gr["CN",1])
demo_info["Gender", 3] <- paste0(gender_gr["MCI",2] ,"/",gender_gr["MCI",1])
demo_info["Gender", 5] <- paste0(gender_gr["AD",2] ,"/",gender_gr["AD",1])
demo_info["Gender", 7] <- round(chisq.test(gender_gr)$p.value,3)

# Age, Education, CDR, MMSE, MoCA: mean, sd, range
sw_all_info$Research.Group <- factor(sw_all_info$Research.Group,
                                     levels = c("CN", "MCI", "AD"))

mean_all <- describeBy(sw_all_info[, c("Research.Group", "Age", "Education",
                                       "Global.CDR", "MMSE.Total.Score",
                                       "MoCA.Total")],
                       group = "Research.Group",
                       digits = 3)

demo_info[c("Age", "Education", "CDR", "MMSE", "MoCA"), 1:6] <-
  cbind(paste0(round(mean_all[[1]]$mean[2:6], 2), "±",
               round(mean_all[[1]]$sd[2:6], 2)),
        paste0(round(mean_all[[1]]$min[2:6], 2), "-",
               round(mean_all[[1]]$max[2:6], 2)),
        paste0(round(mean_all[[2]]$mean[2:6], 2), "±",
               round(mean_all[[2]]$sd[2:6], 2)),
        paste0(round(mean_all[[2]]$min[2:6], 2), "-",
               round(mean_all[[2]]$max[2:6], 2)),
        paste0(round(mean_all[[3]]$mean[2:6], 2), "±",
               round(mean_all[[3]]$sd[2:6], 2)),
        paste0(round(mean_all[[3]]$min[2:6], 2), "-",
               round(mean_all[[3]]$max[2:6], 2)))

# one way ANOVA
ano_age <- anova(lm(Age ~ Research.Group, sw_all_info))

```

```

ano_education <- anova(lm(Education ~ Research.Group, sw_all_info))
ano_MMSE <- anova(lm(MMSE.Total.Score ~ Research.Group, sw_all_info))
ano_MoCA <- anova(lm(MoCA.Total ~ Research.Group, sw_all_info))

demo_info["Age", 7] <- round(ano_age$`Pr(>F)`[1],3)
demo_info["Education", 7] <- round(ano_education$`Pr(>F)`[1],3)
demo_info["MMSE", 7] <- round(ano_MMSE$`Pr(>F)`[1],4)
demo_info["MoCA", 7] <- round(ano_MoCA$`Pr(>F)`[1],4)

# CDR
kruskal.test(Global.CDR ~ Research.Group, data = sw_all_info)

##
## Kruskal-Wallis rank sum test
##
## data: Global.CDR by Research.Group
## Kruskal-Wallis chi-squared = 425.75, df = 2, p-value < 2.2e-16

demo_info["CDR", 7] <- "< 0.001"

#knitr::kable(demo_info)

writexl::write_xlsx(demo_info, "results/demographic_info.xlsx")

# subjects with education
length(which(!is.na(sw_all_info$Education[sw_all_info$Research.Group == "CN"])))

## [1] 185

length(which(!is.na(sw_all_info$Education[sw_all_info$Research.Group == "MCI"])))

## [1] 150

length(which(!is.na(sw_all_info$Education[sw_all_info$Research.Group == "AD"])))

## [1] 153

# subjects with MMSE
length(which(!is.na(sw_all_info$MMSE.Total.Score[sw_all_info$Research.Group == "CN"])))

## [1] 185

length(which(!is.na(sw_all_info$MMSE.Total.Score[sw_all_info$Research.Group == "MCI"])))

## [1] 150

length(which(!is.na(sw_all_info$MMSE.Total.Score[sw_all_info$Research.Group == "AD"])))

## [1] 153

```

```

# subjects with MoCA
length(which(!is.na(sw_all_info$MoCA.Total[sw_all_info$Research.Group == "CN"])))

## [1] 174

length(which(!is.na(sw_all_info$MoCA.Total[sw_all_info$Research.Group == "MCI"])))

## [1] 100

length(which(!is.na(sw_all_info$MoCA.Total[sw_all_info$Research.Group == "AD"])))

## [1] 92

```

## Summary of global network metrics

```

sum_metrics <- as.data.frame(matrix(0,8,6))

mean_all <- describeBy(sw_all_info[, c(4,14:15, 29, 16,18,20:22)], group = "Research.Group",
                        digits = 3)

sum_metrics[1:8, 1:6] <-
  cbind(paste0(round(mean_all[[1]]$mean[2:9],3),"±",
                round(mean_all[[1]]$sd[2:9],3)),
        paste0(round(mean_all[[1]]$min[2:9],3),"-",
                round(mean_all[[1]]$max[2:9],3)),
        paste0(round(mean_all[[2]]$mean[2:9],3),"±",
                round(mean_all[[2]]$sd[2:9],3)),
        paste0(round(mean_all[[2]]$min[2:9],3),"-",
                round(mean_all[[2]]$max[2:9],3)),
        paste0(round(mean_all[[3]]$mean[2:9],3),"±",
                round(mean_all[[3]]$sd[2:9],3)),
        paste0(round(mean_all[[3]]$min[2:9],3),"-",
                round(mean_all[[3]]$max[2:9],3)))

row.names(sum_metrics) <- colnames(sw_all_info[, c(14:15, 29, 16,18,20:22)])

colnames(sum_metrics) <- c("CN.mean", "CN.range", "MCI.mean", "MCI.range",
                          "AD.mean", "AD.range")

#knitr::kable(sum_metrics)

WriteXLS::WriteXLS(sum_metrics,"results/sum_metrics.xlsx", row.names = T, col.names = T)

```

## Correlations between education and global network metrics (revision on 2024-3-21)

```

gr_education <- as.data.frame(matrix(0, 24,4))

i <- 0
for (sw in colnames(sw_all_info)[c(14:15, 29, 16,18,20:22)])
{
  for (gr in c("CN","AD","MCI")) {
    i <- i+1

    my.t <- cor.test(sw_all_info[sw_all_info$Research.Group == gr,"Education"],
      sw_all_info[sw_all_info$Research.Group == gr, sw])
    gr_education[i,1:2] <- c(sw, gr)

    gr_education[i, 3:4] <- c(round(my.t$estimate, 3), round(my.t$p.value, 3))
  }
}

colnames(gr_education) <- c("network_measure","group","t-value","p-value")
knitr::kable(gr_education)

```

network_measure	group	t-value	p-value
number.of.nodes	CN	0.133	0.072
number.of.nodes	AD	0.135	0.097
number.of.nodes	MCI	0.247	0.002
mean.degree	CN	0.084	0.256
mean.degree	AD	0.149	0.066
mean.degree	MCI	0.252	0.002
connectivity.density	CN	-0.048	0.515
connectivity.density	AD	0.084	0.301
connectivity.density	MCI	0.038	0.645
mean.custering.coef	CN	-0.039	0.602
mean.custering.coef	AD	0.141	0.082
mean.custering.coef	MCI	0.089	0.281
mean.pathlength	CN	0.023	0.758
mean.pathlength	AD	0.010	0.898
mean.pathlength	MCI	0.008	0.923
gamma	CN	0.000	0.998
gamma	AD	0.146	0.071
gamma	MCI	0.123	0.134
lambda	CN	-0.018	0.806
lambda	AD	0.108	0.183
lambda	MCI	0.056	0.496
sigma	CN	0.005	0.947
sigma	AD	0.143	0.078
sigma	MCI	0.130	0.113

```
#View(gr_education)
```

## Group differences in global network metrics

```
# network size/density across all subjects  
mean(sw_all_info$number.of.nodes)
```

```
## [1] 6589.988
```

```
sd(sw_all_info$number.of.nodes)
```

```
## [1] 607.7006
```

```
mean(sw_all_info$connectivity.density)
```

```
## [1] 17.32638
```

```
sd(sw_all_info$connectivity.density)
```

```
## [1] 1.133136
```

```
# network metrics in each group  
sw_all_info$Research.Group <- factor(sw_all_info$Research.Group,  
                                     levels = c("CN","MCI","AD"))  
  
gr_diff <- as.data.frame(matrix(0, 8, 12))  
  
i <- 0  
  
for (sw in colnames(sw_all_info)[c(14:15, 29, 16,18,20:22)]) {  
  i <- i+1  
  
  fl <- formula(paste0(sw, " ~ Research.Group + Age + Gender + TIV + Education"))  
  
  diff <- anova(lm(fl, sw_all_info))  
  
  gr_diff[i,1] <- sw  
  
  gr_diff[i, 2:6] <- round(diff$`Pr(>F)`[1:5],3)  
  
  # post-hoc analysis  
  tt <- pairwise.t.test(sw_all_info[,sw], sw_all_info[, "Research.Group"],  
                        p.adjust="fdr", pool.sd = T)  
  
  gr_diff[i,7] <- "CN vs. MCI"  
  gr_diff[i,8] <- round(tt$p.value[1,1],3)
```

```

gr_diff[i,9] <- "CN vs. AD"
gr_diff[i,10] <- round(tt$p.value[2,1],3)

gr_diff[i,11] <- "MCI vs. AD"
gr_diff[i,12] <- round(tt$p.value[2,2],3)

}

colnames(gr_diff) <- c("parameter", "Group", "Age", "Gender", "TIV", "Education",
                      "contrast1", "p-value1", "contrast2", "p-value2",
                      "contrast3", "p-value3")
#View(gr_diff)
knitr::kable(gr_diff)

```

parameter	Group	Age	Gender	TIV	Education	contrast1	p-value1	contrast2	p-value2	contrast3	p-value3
number.of.nodes	0.010	0.060	0.000	0.000	0.885	CN vs. MCI	0.605	CN vs. AD	0.412	MCI vs. AD	0.377
mean.degree	0.001	0.000	0.000	0.000	0.326	CN vs. MCI	0.917	CN vs. AD	0.026	MCI vs. AD	0.026
connectivity.density	0.009	0.001	0.026	0.010	0.203	CN vs. MCI	0.451	CN vs. AD	0.010	MCI vs. AD	0.058
mean.custering_coef	0.000	0.000	0.001	0.107	0.019	CN vs. MCI	0.081	CN vs. AD	0.000	MCI vs. AD	0.000
mean.pathlength	0.002	0.017	0.071	0.225	0.589	CN vs. MCI	0.397	CN vs. AD	0.002	MCI vs. AD	0.021
gamma	0.000	0.000	0.021	0.001	0.016	CN vs. MCI	0.014	CN vs. AD	0.000	MCI vs. AD	0.000
lambda	0.000	0.000	0.000	0.354	0.015	CN vs. MCI	0.030	CN vs. AD	0.000	MCI vs. AD	0.000
sigma	0.000	0.000	0.203	0.000	0.029	CN vs. MCI	0.020	CN vs. AD	0.000	MCI vs. AD	0.000

```
which(p.adjust(gr_diff$Group) < 0.05)
```

```
## [1] 1 2 3 4 5 6 7 8
```

```
#writexl::write_xlsx(gr_diff, "results/group_diff_all.xlsx", col_names = T)
```

## Plots for group differences in global network metrics

```

sw_all_info$Research.Group <- factor(sw_all_info$Research.Group, levels = c("CN","MCI","AD"))

for (sw in c("mean.custering_coef", "mean.degree", "mean.pathlength", "connectivity.density",
             "number.of.nodes", "gamma", "sigma", "lambda")) {
  p <- ggplot(sw_all_info, aes_string(x = "Research.Group", y = sw, fill = "Research.Group")) +
    geom_violin(position = position_dodge(width=0.9), alpha = 0.6)+

```



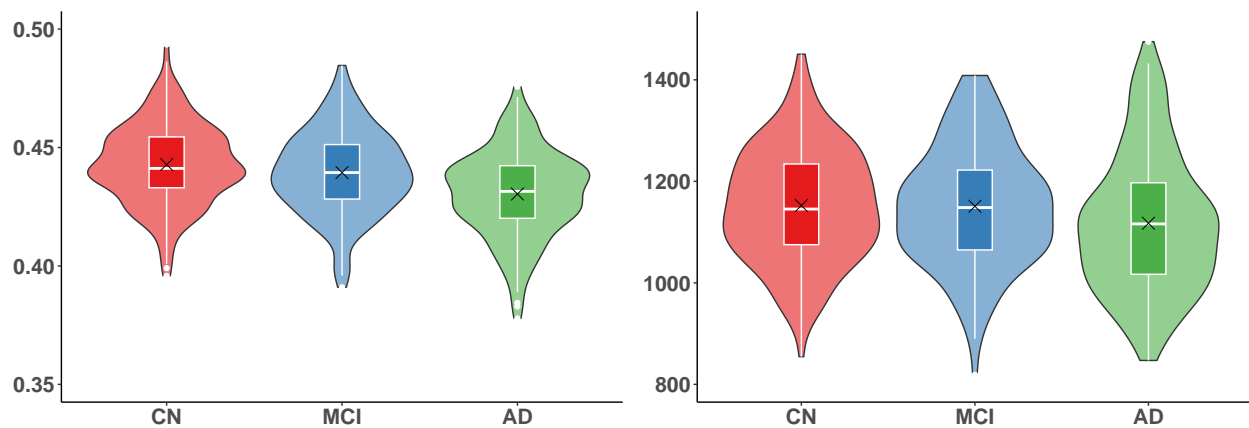
```

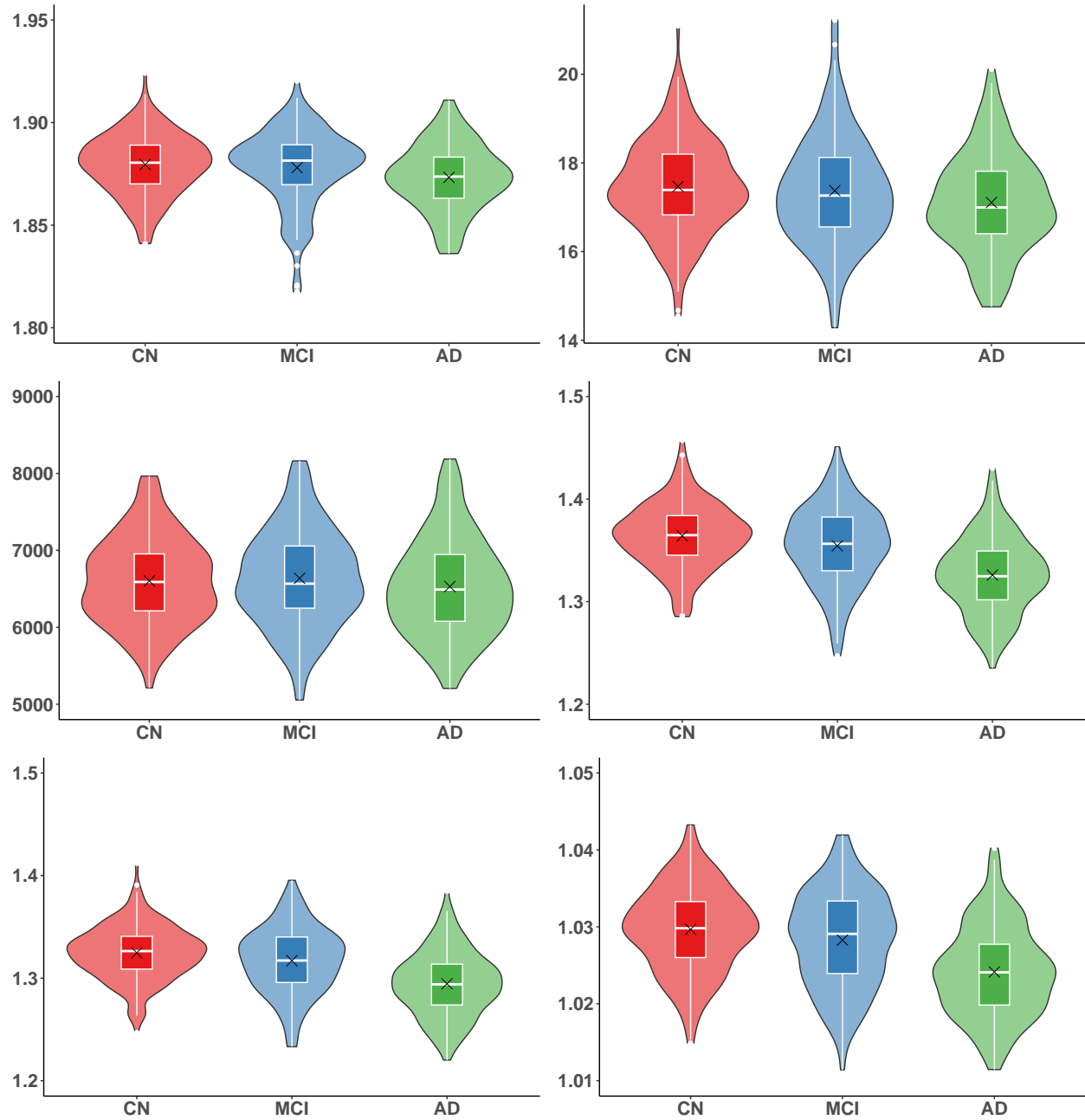
geom_boxplot(color = "white",width = 0.2) + # position = position_dodge(width=0.9)
guides(fill = "none") +
scale_fill_brewer(palette = "Set1")+
stat_summary(fun=mean, size = 4, position = position_dodge(width=0.9),
             geom = "point",shape=4, col = "black") + # shape=1
theme(axis.text = element_text(size = 16, face = "bold"),
      axis.title = element_blank()+
theme(panel.border = element_blank(),
      panel.background = element_blank(),
      panel.grid = element_blank(),
      axis.line = element_line(colour = "black"))
if (sw == "mean.custering_coef") {
  p <- p + coord_cartesian(ylim = c(0.35, 0.5))
} else if (sw == "mean.degree") {
  p <- p + coord_cartesian(ylim = c(800, 1500))
} else if (sw == "mean.pathlength") {
  p <- p + coord_cartesian(ylim = c(1.8, 1.95))
} else if (sw == "connection.density") {
  p <- p + coord_cartesian(ylim = c(14, 22))
} else if (sw == "number.of.nodes") {
  p <- p + coord_cartesian(ylim = c(5000, 9000))
} else if (sw == "gamma") {
  p <- p + coord_cartesian(ylim = c(1.2, 1.5))
} else if (sw == "lambda") {
  p <- p + coord_cartesian(ylim = c(1.01, 1.05))
} else if (sw == "sigma") {
  p <- p + coord_cartesian(ylim = c(1.2, 1.5))
}

print(p)

ggsave(paste0("results/","violin_plot.",sw,".png"), width = 3, height = 3,
       units = c("in"), dpi = 200)
}

```





## Correlations between network metrics and cognitive ability

```
gr_correlation <- as.data.frame(matrix(0, 7, 27))
i <- 0
for (sw in colnames(sw_all_info)[c(15, 29, 16,18,20:22)]) {
  i <- i+1

  f1 <- formula(paste0(sw, " ~ MoCA.Total + Age + Gender + TIV+Education" ))
```

```

corr <- anova(lm(fl, sw_all_info))

gr_correlation[i,1] <- sw
gr_correlation[i, 2:6] <- round(corr$`Pr(>F)`[1:5],4)

# CN
lmCN <- lm(fl, sw_all_info[sw_all_info$Research.Group == "CN",])
ano_CN <- anova(lmCN)

mycor <- cor.test(sw_all_info[sw_all_info$Research.Group == "CN",sw],
  sw_all_info[sw_all_info$Research.Group == "CN","MoCA.Total"])

gr_correlation[i, 7] <- round(mycor$estimate,4)
gr_correlation[i, 8] <- coef(lmCN)[2]

gr_correlation[i, 9:13] <- round(ano_CN$`Pr(>F)`[1:5],4)

# MCI
lmMCI <- lm(fl, sw_all_info[sw_all_info$Research.Group == "MCI",])
ano_MCI <- anova(lmMCI)

mycor <- cor.test(sw_all_info[sw_all_info$Research.Group == "MCI",sw],
  sw_all_info[sw_all_info$Research.Group == "MCI","MoCA.Total"])

gr_correlation[i, 14] <- round(mycor$estimate,4)
gr_correlation[i, 15] <- coef(lmMCI)[2]
gr_correlation[i, 16:20] <- round(ano_MCI$`Pr(>F)`[1:5],4)

# AD
lmAD <- lm(fl, sw_all_info[sw_all_info$Research.Group == "AD",])
ano_AD <- anova(lmAD)

mycor <- cor.test(sw_all_info[sw_all_info$Research.Group == "AD",sw],
  sw_all_info[sw_all_info$Research.Group == "AD","MoCA.Total"])

gr_correlation[i, 21] <- round(mycor$estimate,4)
gr_correlation[i, 22] <- coef(lmAD)[2]

gr_correlation[i, 23:27] <- round(ano_AD$`Pr(>F)`[1:5],4)
}

colnames(gr_correlation) <- c("parameter", "MoCA","Age","Gender","TIV","Education",
  "r-CN","beta-CN","MoCA_CN","Age_CN","Gender_CN",
  "TIV_CN","Education_CN","r-MCI","beta-MCI","MoCA_MCI",
  "Age_MCI","Gender_MCI","TIV_MCI", "Education_MCI",
  "r-AD","beta-AD","MoCA_AD","Age_AD","Gender_AD",
  "TIV_AD","Education_AD")

#View(gr_correlation)
#knitr::kable(gr_correlation)

writexl::write_xlsx(gr_correlation, "results/MoCA.correlation_new.xlsx", col_names = T)

```

## Scatter plots for correlations between network metrics and cognitive ability

```
library(RColorBrewer)
library(ppcor)

## Loading required package: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##      select

brewer.pal(3, "Set1")

## [1] "#E41A1C" "#377EB8" "#4DAF4A"

#table(ggplot_build(p)$data[[1]]$colour)

# CN
mm <- gr_correlation$parameter[gr_correlation$MoCA_CN < 0.05]
mm

## [1] "mean.degree"

gr <- "CN"
tmp <- sw_all_info[sw_all_info$Research.Group == gr & !is.na(sw_all_info$MoCA.Total),]

for(sig_sw in mm) {

  plm <- lm(formula(paste0(sig_sw, " ~ MoCA.Total + Age + Gender + TIV + Education")),
            data=tmp)

  anova(plm)

  tmp$Research.Group <- droplevels(tmp$Research.Group)
  p <- ggplot(data=tmp, aes_string(y=sig_sw, x="MoCA.Total")) +
    geom_point(size = 5, col = '#E41A1C', alpha = 0.8) + #col = 'blue',
    # geom_smooth(method = "lm", col = 'black', linewidth = 2, se = FALSE) +
    geom_smooth(method = "lm", mapping=aes(y=predict(plm,tmp)),
                col = 'black',linewidth = 2, se = FALSE) +
    guides(color = "none") +
    labs(x = "", y = "") +
    theme(legend.title = element_text(colour="black", size=14, face="bold"),
          legend.text = element_text(colour="black", size=14, face="bold")) +
    theme(plot.title = element_text(hjust = 0.5))+
    theme(axis.title = element_blank(),
```

```

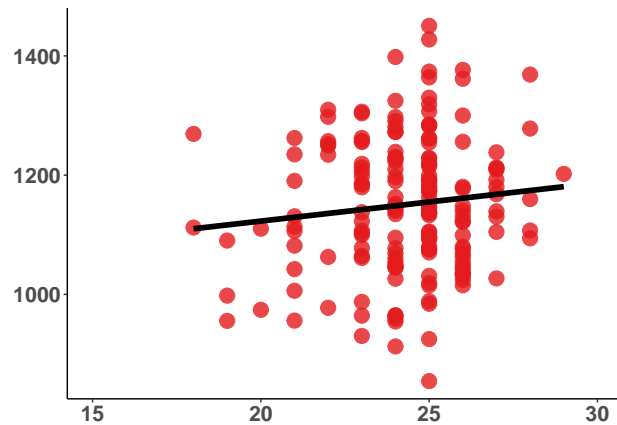
    axis.text = element_text(size = 16, face = "bold")) +
  theme(panel.background = element_blank(),
        panel.border = element_blank(),
        panel.grid = element_blank(),
        axis.line = element_line(colour = "black")) +
  coord_cartesian(xlim=c(15,30))

  print(p)
  ggsave(paste0("results/",gr,".",sig_sw,".png"), width = 6,
        height = 4, units = c("in"), dpi = 200)
}

```

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

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



```

# MCI
mm <- gr_correlation$parameter[gr_correlation$MoCA_MCI < 0.05]

gr <- "MCI"
tmp <- sw_all_info[sw_all_info$Research.Group == gr,]

cor.test(tmp[, "gamma"], tmp$MoCA.Total)

```

```

##
## Pearson's product-moment correlation
##
## data: tmp[, "gamma"] and tmp$MoCA.Total
## t = 1.9518, df = 98, p-value = 0.05381
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.003094065 0.375589060
## sample estimates:
## cor
## 0.1934416

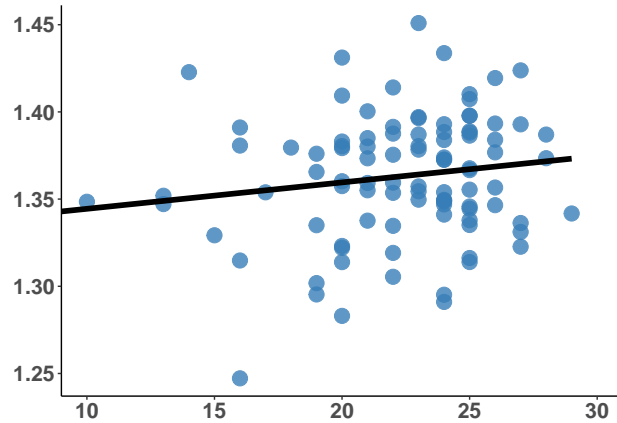
```

```
cor.test(tmp[, "sigma"], tmp$MoCA.Total)
```

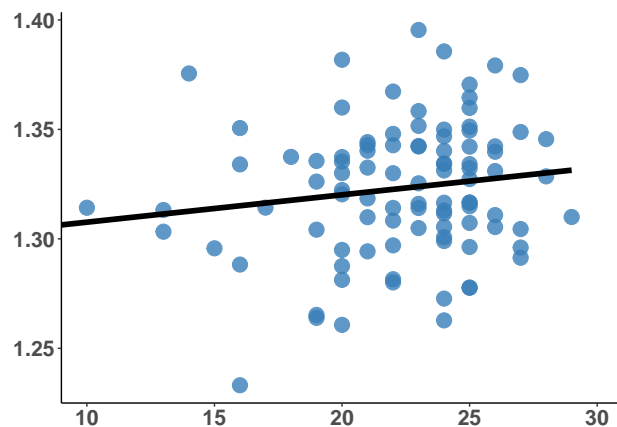
```
##  
## Pearson's product-moment correlation  
##  
## data: tmp[, "sigma"] and tmp$MoCA.Total  
## t = 1.8979, df = 98, p-value = 0.06066  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## -0.008445788 0.370982910  
## sample estimates:  
## cor  
## 0.1882847
```

```
for(sig_sw in mm) {  
  
  plm <- lm(formula(paste0(sig_sw, " ~ MoCA.Total + Age + Gender + TIV")),  
            data=tmp)  
  
  anova(plm)  
  
  tmp$Research.Group <- droplevels(tmp$Research.Group)  
  p <- ggplot(data=tmp, aes_string(y=sig_sw, x="MoCA.Total")) +  
    geom_point(size = 5, col = '#377EB8', alpha = 0.8) + #col = 'blue',  
    # geom_smooth(method = "lm", col = 'black', linewidth = 2, se = FALSE) +  
    geom_smooth(method = "lm", mapping=aes(y=predict(plm,tmp)),  
              col = 'black', linewidth = 2, se = FALSE) +  
    guides(color = "none") +  
    labs(x = "", y = "") +  
    theme(legend.title = element_text(colour="black", size=14, face="bold"),  
          legend.text = element_text(colour="black", size=14, face="bold")) +  
    theme(plot.title = element_text(hjust = 0.5))+  
    theme(axis.title = element_blank(),  
          axis.text = element_text(size = 16, face = "bold")) +  
    theme(panel.background = element_blank(),  
          panel.border = element_blank(),  
          panel.grid = element_blank(),  
          axis.line = element_line(colour = "black")) +  
    coord_cartesian(xlim=c(10,30))  
  
  print(p)  
  
  ggsave(paste0("results/",gr,".",sig_sw,".png"), width = 6,  
         height = 4, units = c("in"), dpi = 200)  
  
}
```

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



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



```
# AD
km <- gr_correlation$parameter[gr_correlation$MoCA_AD < 0.05]
gr <- "AD"
tmp <- sw_all_info[sw_all_info$Research.Group == gr,]
tmp$Research.Group <- droplevels(tmp$Research.Group)

for (sig_sw in km) {

  plm <- lm(formula(paste0(sig_sw, " ~ MoCA.Total + Age + Gender + TIV")),
            data=tmp)

  anova(plm)

p <- ggplot(data=tmp, aes_string(y=sig_sw, x="MoCA.Total")) +
  geom_point(size = 5, col = '#4DAF4A', alpha = 0.8) + #col = 'blue',
  geom_smooth(method = "lm", mapping=aes(y=predict(plm,tmp)),
            col = 'black',linewidth = 2, se = FALSE) +
  guides(color = "none") +
  labs(x = "", y = "") +
  theme(legend.title = element_text(colour="black", size=14, face="bold"),
        legend.text = element_text(colour="black", size=14, face="bold")) +
```

```

theme(plot.title = element_text(hjust = 0.5))+
theme(axis.title = element_blank(),
      axis.text = element_text(size = 16, face = "bold")) +
theme(panel.background = element_blank(),
      panel.border = element_blank(),
      panel.grid = element_blank(),
      axis.line = element_line(colour = "black")) +
coord_cartesian(xlim=c(5,30)) +
scale_x_continuous(breaks = seq(5,30,5))

print(p)

ggsave(paste0("results/",gr,".",sig_sw,".png"), width = 6,
      height = 4, units = c("in"), dpi = 200)

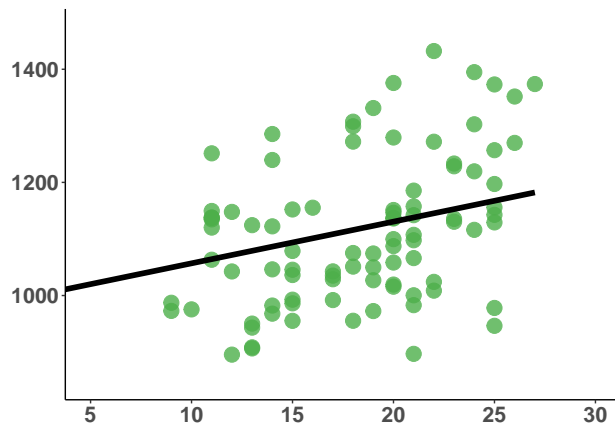
}

```

```

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

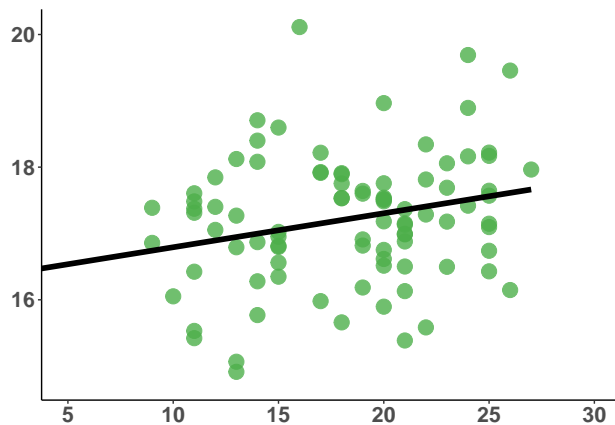
```



```

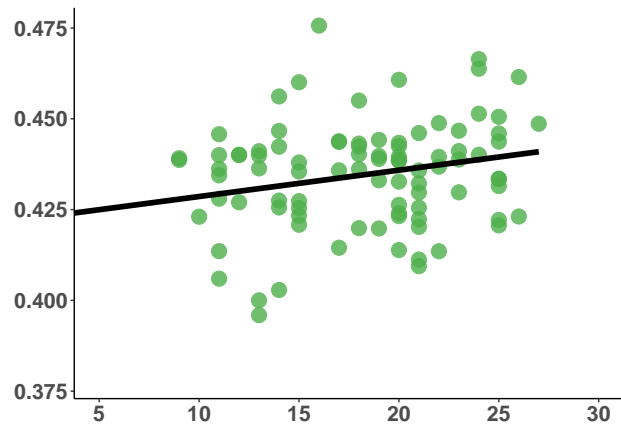
## 'geom_smooth()' using formula = 'y ~ x'
## 'geom_smooth()' using formula = 'y ~ x'

```





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



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

