ADHD Study 3 Analysis

Elisabeth R Silver

2023-08-28

library(pacman)  
p\_load(tidyverse, stats, psych, scales, cowplot, rstatix, statstring,  
 apaTables, effects, emmeans, careless, openxlsx, flextable)  
filter <- dplyr::filter  
select <- dplyr::select  
source("calculate\_adhd\_scores.R")

df <- read.csv("data\_nadhd.csv")  
df <- df[3:nrow(df),]  
df.2 <- read.csv("data\_adhd.csv")  
df.2 <- df.2[3:nrow(df.2),]  
df <- rbind(df, df.2)

df <- df %>%   
 filter(PROLIFIC\_PID != "") %>%   
 filter(PROLIFIC\_PID != "TEST") %>%   
 filter(FL\_35\_DO != "")

Create a dataset for randomly selecting people for the bonus payments:

award <- df %>%   
 filter(PROLIFIC\_PID != "") %>%   
 filter(PROLIFIC\_PID != "TEST") %>%   
 filter(FL\_35\_DO != "") %>%   
 filter(survey\_about!="Pedophilia")  
set.seed(1234)  
award <- award[sample(nrow(award), 10), ]  
award <- award %>%   
 arrange(STUDY\_ID)  
cat(str\_c(award$PROLIFIC\_PID,collapse="\n"))

## 63cbd5095083beef0c8b1a7a  
## 5f5561166d9ce792e07134e2  
## 599314b360937e0001899338  
## 6042794fe3c49714bee3e9cf  
## 580670eceee5dc00017584c6  
## 5a1efc2b517dfb00013dd370  
## 5be3237a3193e90001798302  
## 60e1d5f866e681d7e33fd01c  
## 61bb396b40db417c1a138e43  
## 5dd03c5486c8861300974102

#convert text to numeric data  
df <- df %>%   
 mutate(across(c(starts\_with("ocb"), starts\_with("cwb")),  
 ~as.numeric(str\_extract(.x, "\\d"))))  
df <- df %>%   
 mutate(across(matches("C\\d"),  
 ~as.numeric(str\_extract(.x, "\\d"))))

We recruited 551 participants from Prolific. We excluded 8 participants who did not indicate their ADHD status, 1 participant who gave insincere responses to two open-ended questions on the survey, and 2 participants who did not respond to all ADHD symptom screener questions.

vars <- read.csv("variable\_recode.csv")  
all\_facets <- c(vars$facet[!is.na(vars$facet)],   
 str\_c("W", vars$facet[!is.na(vars$facet)]))  
all\_facets <- all\_facets[!duplicated(all\_facets)]  
#note which variables should be reverse-scored (indicated by "rev\_score"==1)  
facet\_items <- vars %>%   
 filter(!is.na(facet)) %>%   
 mutate(  
 new\_varname = if\_else(rev\_score==1, str\_c("R\_", og\_varname),  
 og\_varname),  
 facet = if\_else(str\_starts(og\_varname, "WC"), str\_c("W", facet), facet),  
 facet\_name = if\_else(str\_starts(og\_varname, "WC"),   
 str\_c("work ", facet\_name),   
 facet\_name)  
 )  
  
  
rev\_score\_items <- facet\_items %>%  
 filter(rev\_score==1)  
  
df <- df %>%   
 mutate(across(any\_of(rev\_score\_items$og\_varname),  
 ~6-.x,  
 .names = "R\_{.col}"))

Exclude participants who provided the same response value for all 24 conscientiousness items:

We excluded 26 participants who provided the same response value for all 24 of the conscientiousness items (i.e., the same response to all 24 conscientiousness items.

The sample was mostly comprised of women (55.1%) and white participants (83.9% white, 7.2% Black, 7.6% Asian, 6.0% Latino/a, 2.1% Native Am., Native AK, Native HI, Pac. Isl., and 0.4% no response. Note that percentages sum to more than 100 because some participants selected more than one racial identity. The average (SD) age was 38.54 years (12.01).

Get scores and reliabilities for work conscientiousness and its facets:

get\_scale\_stats <- function(df, columns, compname){  
 calpha = psych::alpha(df %>%   
 select(all\_of(columns)),check.keys = T)   
 cat(paste0("\nCronbach's $\\alpha$ for \*\*",compname, "\*\*: ",  
 number(calpha$total$raw\_alpha[1],accuracy=.01)))  
   
}  
  
for(f in all\_facets){  
 select\_vars <- facet\_items %>%   
 filter(facet==f) #just select relevant columns  
 select\_name <- select\_vars$facet\_name[1]  
 get\_scale\_stats(df, select\_vars$new\_varname, select\_name)  
 new\_col = str\_replace\_all(select\_name, "([-]|\\s)", ".")  
 print(new\_col)  
 df[new\_col] <- rowMeans(df[,select\_vars$new\_varname], na.rm=T)  
 cat(" \n")  
}

Cronbach’s for **self-efficacy**: 0.86[1] “self.efficacy”

Cronbach’s for **orderliness**: 0.85[1] “orderliness”

Cronbach’s for **dutifulness**: 0.72[1] “dutifulness”

Cronbach’s for **achievement-striving**: 0.79[1] “achievement.striving”

Cronbach’s for **self-discipline**: 0.77[1] “self.discipline”

Cronbach’s for **cautiousness**: 0.94[1] “cautiousness”

Cronbach’s for **work self-efficacy**: 0.85[1] “work.self.efficacy”

Cronbach’s for **work orderliness**: 0.77[1] “work.orderliness”

Cronbach’s for **work dutifulness**: 0.81[1] “work.dutifulness”

Cronbach’s for **work achievement-striving**: 0.75[1] “work.achievement.striving”

Cronbach’s for **work self-discipline**: 0.77[1] “work.self.discipline”

Cronbach’s for **work cautiousness**: 0.90[1] “work.cautiousness”

get\_scale\_stats(df, consc.gen$new\_varname, "conscientiousness")

Cronbach’s for **conscientiousness**: 0.92

cat(" \n")

df["consc.gen"] <- rowMeans(df[,consc.gen$new\_varname],na.rm=T)  
  
get\_scale\_stats(df, consc.work$new\_varname, "work conscientiousness")

Cronbach’s for **work conscientiousness**: 0.92

cat(" \n")

df["consc.work"] <- rowMeans(df[,consc.work$new\_varname],na.rm=T)

get\_scale\_stats(df, colnames(df %>% select(starts\_with("ocb"))), "OCB")

Cronbach’s for **OCB**: 0.90

df["total.ocb"] <- rowMeans(df[,colnames(df %>% select(starts\_with("ocb")))],  
 na.rm=T)  
get\_scale\_stats(df, colnames(df %>% select(starts\_with("cwb"))), "CWB")

Cronbach’s for **CWB**: 0.86

df["total.cwb"] <- rowMeans(df[,colnames(df %>% select(starts\_with("cwb")))],  
 na.rm=T)

Move the work conscientiousness columns to a general conscientiousness set of columns–the form condition is tracked already in the spreadsheet.

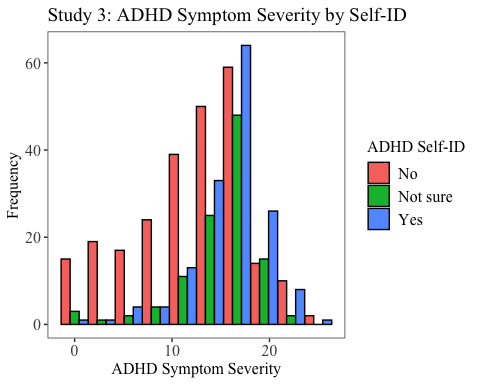
work.parts <- df %>%   
 filter(FL\_35\_DO=="work\_att") %>%   
 select(-matches("\\bC\\d")) %>%   
 select(-matches("\\bR[\_]C\\d")) %>%   
 select(-c(self.efficacy, self.discipline,   
 orderliness, dutifulness,   
 achievement.striving, cautiousness, consc.gen)) %>%   
 rename\_with(~str\_remove\_all(.x, "W"), starts\_with("WC")) %>%   
 rename\_with(~str\_remove\_all(.x, "W"), starts\_with("R\_WC")) %>%   
 rename\_with(~str\_remove\_all(.x, "work."), starts\_with("work.")) %>%   
 rename(consc.gen=consc.work)  
  
gen.parts <- df %>%   
 filter(FL\_35\_DO=="gen\_attn") %>%   
 select(-matches("\\bWC\\d")) %>%   
 select(-matches("\\bR\_WC\\d")) %>%   
 select(-starts\_with("work.")) %>%   
 select(-consc.work)  
  
df.all <- rbind(work.parts, gen.parts)

psych::alpha(df.all[,asrs\_cols])

##   
## Reliability analysis   
## Call: psych::alpha(x = df.all[, asrs\_cols])  
##   
## raw\_alpha std.alpha G6(smc) average\_r S/N ase mean sd median\_r  
## 0.66 0.66 0.63 0.25 1.9 0.023 2.7 0.73 0.24  
##   
## 95% confidence boundaries   
## lower alpha upper  
## Feldt 0.61 0.66 0.70  
## Duhachek 0.61 0.66 0.71  
##   
## Reliability if an item is dropped:  
## raw\_alpha std.alpha G6(smc) average\_r S/N alpha se var.r med.r  
## attn\_1 0.58 0.58 0.53 0.22 1.4 0.029 0.0028 0.20  
## attn\_2 0.62 0.62 0.58 0.25 1.6 0.027 0.0044 0.24  
## attn\_3 0.64 0.64 0.59 0.26 1.7 0.025 0.0032 0.25  
## attn\_4 0.62 0.62 0.58 0.25 1.6 0.026 0.0047 0.24  
## attn\_5 0.64 0.64 0.60 0.27 1.8 0.025 0.0038 0.26  
## attn\_6 0.60 0.60 0.56 0.23 1.5 0.028 0.0039 0.22  
##   
## Item statistics   
## n raw.r std.r r.cor r.drop mean sd  
## attn\_1 515 0.68 0.69 0.61 0.49 2.9 1.1  
## attn\_2 515 0.60 0.60 0.47 0.38 2.0 1.2  
## attn\_3 515 0.58 0.57 0.43 0.34 3.0 1.2  
## attn\_4 515 0.61 0.60 0.47 0.38 2.8 1.2  
## attn\_5 515 0.54 0.55 0.39 0.32 3.1 1.2  
## attn\_6 515 0.65 0.64 0.53 0.43 2.4 1.3  
##   
## Non missing response frequency for each item  
## 1 2 3 4 5 miss  
## attn\_1 0.17 0.10 0.37 0.32 0.04 0  
## attn\_2 0.53 0.04 0.31 0.10 0.01 0  
## attn\_3 0.17 0.15 0.29 0.29 0.09 0  
## attn\_4 0.23 0.11 0.31 0.31 0.04 0  
## attn\_5 0.12 0.15 0.30 0.33 0.10 0  
## attn\_6 0.39 0.06 0.32 0.20 0.03 0

Look at ASRS scores by ADHD self-identification:

p<-ggplot(df.all, aes(asrs\_sum,fill=adhd\_yn))+  
 geom\_histogram(position="dodge",bins=10,colour="black")+  
 theme\_bw()+  
 xlab("ADHD Symptom Severity")+  
 ylab("Frequency")+  
 ggtitle("Study 3: ADHD Symptom Severity by Self-ID")+  
 labs(fill = "ADHD Self-ID")+  
 theme(panel.grid.major=element\_blank(),  
 panel.grid.minor=element\_blank())+  
 theme(text=element\_text(family="Times New Roman",size=12),  
 axis.text.x = element\_text(size=12),  
 axis.text.y = element\_text(size=12),  
 legend.text = element\_text(size=12))  
print(p)



save\_plot("adhd\_sx\_screener.png", p)

msd <- df.all %>%   
 group\_by(adhd\_yn) %>%   
 summarise(  
 msd = paste0(  
 number(mean(asrs\_sum,na.rm=T), accuracy = .01),  
 " (",number(sd(asrs\_sum,na.rm=T), accuracy = .01),  
 ")"  
 ),  
 m = number(mean(asrs\_sum,na.rm=T), accuracy = .01),  
 sd = number(sd(asrs\_sum,na.rm=T), accuracy = .01),  
 n = n()  
 )  
  
df.all$adhd\_yn <- factor(df.all$adhd\_yn)  
res <- df.all %>% anova\_test(asrs\_sum~adhd\_yn, type = 3, effect.size ="pes")  
tmp.res <- get\_anova\_table(res)  
tmp.res["result"] <- format\_anova\_string(tmp.res)  
cat(paste0("There was a significant effect of ADHD self-identification on ADHD symptom severity scores, ",  
 tmp.res$result[1], ". "))

There was a significant effect of ADHD self-identification on ADHD symptom severity scores, *F*(2, 512) = 30.78, *p* < .001, partial = 0.11.

tukeyres <- df.all %>% tukey\_hsd(asrs\_sum~adhd\_yn)  
df.all["adhd"] <- if\_else(df.all$adhd\_yn == "No",  
 "Non-ADHD", "ADHD")  
cat(paste0("\"No\" respondents ( \_M\_ = ", msd$m[1], ", \_SD\_ = ", msd$sd[1],  
 "N = ", msd$n[1],") reported fewer symptoms than \"Yes\" respondents ( \_M\_ = ",  
 msd$m[3],", \_SD\_ = ", msd$sd[3],  
 " N = ", msd$n[3],"), ",  
 format\_pval\_apa(tukeyres$p.adj[2]),   
 ", and \"Not sure\" respondents ( \_M\_ = ",  
 msd$m[2], ", \_SD\_ = ", msd$sd[2],  
 " N = ", msd$n[2], "), ",  
 format\_pval\_apa(tukeyres$p.adj[1]), "."))

“No” respondents ( *M* = 12.12, *SD* = 5.89N = 249) reported fewer symptoms than “Yes” respondents ( *M* = 15.83, *SD* = 3.82 N = 155), *p* < .001, and “Not sure” respondents ( *M* = 15.14, *SD* = 4.22 N = 111), *p* < .001.

cat(paste0("\"Yes\" respondents did not significantly differ ",  
 "from \"Not sure\" respondents, ",  
 format\_pval\_apa(tukeyres$p.adj[3])))

“Yes” respondents did not significantly differ from “Not sure” respondents, *p* = .50 ## Descriptives

Should the correlation table include separate correlations for people who completed the work vs. general conscientiousness forms (i.e., on upper and lower triangle)?

table(df.all$adhd, df.all$FL\_35\_DO)

##   
## gen\_attn work\_att  
## ADHD 133 133  
## Non-ADHD 124 125

df.all$age <- as.numeric(df.all$age)  
facet\_items["facet.vnames"] <- str\_replace\_all(facet\_items$facet\_name, "([-]|\\s)", ".")  
col.order <- c("asrs\_sum","age", "consc.gen",   
 facet\_items$facet.vnames[str\_starts(facet\_items$facet.vnames, "work")==F],  
 "total.ocb", "total.cwb","FL\_35\_DO")  
cor\_vars <- df.all %>%   
 select(all\_of(col.order))

cor.gensamp <- data.frame(apa.cor.table(cor\_vars %>%   
 filter(FL\_35\_DO=="gen\_attn"))$table.body)  
cor.gensamp <- cor.gensamp %>%   
 filter(M != " ")  
write.csv(cor.gensamp, "general\_FOR\_correlations.csv")  
cor.worksamp <- data.frame(apa.cor.table(cor\_vars %>%   
 filter(FL\_35\_DO=="work\_att"))$table.body)  
cor.worksamp <- cor.worksamp %>%   
 filter(M != " ")  
write.csv(cor.worksamp, "work\_FOR\_correlations.csv")

## Analyses

#function runs analyses for each outcome variable  
do.pw <- function(longdat, comp.name){  
 return.list = list()  
 rename.tmp = list(outcome.var = comp.name)  
 tmp = longdat %>%   
 rename(!!!rename.tmp)  
 tmp\_tabs = data.frame(get\_anova\_table(anova\_test(tmp,   
 dv = "outcome.var",  
 between = c("adhd", "FL\_35\_DO"),  
 type = 3,  
 effect.size = "pes")))  
 tmp\_tabs["outcome"] = comp.name  
 return.list[["anova"]] = tmp\_tabs  
 #pairwise between comparison  
 pw.btwn = tmp %>%   
 t\_test(outcome.var~adhd,   
 p.adjust.method = "none",  
 var.equal=T)   
   
 #effect size between  
 pw.btwn = pw.btwn %>%   
 left\_join(  
 tmp %>%   
 cohens\_d(outcome.var~adhd,  
 var.equal=T, ci=T) %>%   
 select(.y., effsize,magnitude,conf.low,conf.high),  
 by = ".y.")   
 pw.tmp = pw.btwn %>%   
 mutate(.y. = comp.name) %>%   
 adjust\_pvalue(method="fdr")  
 return.list[["pw"]] = pw.tmp  
 return(return.list)  
}

### Run analyses in loop

#consc -----  
analysis.cols <- facet\_items %>%   
 filter(!duplicated(facet.vnames)) %>%   
 filter(!str\_starts(facet.vnames, "work"))  
analysis.cols <- analysis.cols$facet.vnames  
consc.res <- do.pw(df.all, "consc.gen")  
anova\_tabs <- consc.res$anova  
pw\_result <- consc.res$pw  
  
for(i in 1:length(analysis.cols)){  
 tmp.res <- do.pw(df.all, analysis.cols[i])  
 anova\_tabs <- rbind(anova\_tabs, tmp.res$anova)  
 pw\_result <- rbind(pw\_result, tmp.res$pw)  
}

Format results and save as .xlsx

anova\_tabs["stat"] <- number(anova\_tabs$`F`, accuracy = .01)  
anova\_tabs["stars"] <- sapply(anova\_tabs$p, format\_sig\_stars)  
anova\_tabs["pfmt"] <- sapply(anova\_tabs$p, format\_pval\_apa)  
anova\_tabs$stat1 <- str\_c(anova\_tabs$stat, anova\_tabs$stars)  
anova\_tabs$stat2 <- str\_remove\_all(str\_c(anova\_tabs$stat,", ", anova\_tabs$pfmt, ", ",  
 number(anova\_tabs$pes, accuracy = .001)),  
 "[\_]")  
anova\_tabs$stat2 <- str\_replace\_all(anova\_tabs$stat2, ",", "\n")  
  
anova\_wide <- anova\_tabs %>%  
 pivot\_wider(id\_cols = Effect,  
 names\_from=outcome,  
 values\_from = stat2)  
write.xlsx(anova\_wide, paste0("rstatix\_results.xlsx"))

anova\_wide %>%   
 flextable()

| Effect | consc.gen | self.efficacy | orderliness | dutifulness | achievement.striving | self.discipline | cautiousness |
| --- | --- | --- | --- | --- | --- | --- | --- |
| adhd | 17.20  p < .001  0.033 | 7.63  p = .006  0.015 | 15.34  p < .001  0.029 | 2.85  p = .09  0.006 | 0.13  p = .72  0.000 | 28.40  p < .001  0.053 | 14.78  p < .001  0.028 |
| FL\_35\_DO | 20.40  p < .001  0.038 | 13.70  p < .001  0.026 | 24.96  p < .001  0.047 | 10.23  p = .001  0.020 | 0.96  p = .33  0.002 | 16.08  p < .001  0.030 | 9.84  p = .002  0.019 |
| adhd:FL\_35\_DO | 2.27  p = .13  0.004 | 1.82  p = .18  0.004 | 2.90  p = .09  0.006 | 0.01  p = .92  0.000 | 0.39  p = .53  0.001 | 5.80  p = .02  0.011 | 0.30  p = .58  0.001 |

Look at interaction effects for self-discipline and orderliness:

mod <- lm(self.discipline~adhd\*FL\_35\_DO, data = df.all)  
em.obj <- emmeans(mod,specs=c("adhd","FL\_35\_DO"))  
contrast(em.obj, method="mean\_chg",interaction=T)

## adhd\_mean\_chg FL\_35\_DO\_mean\_chg estimate SE df t.ratio p.value  
## ADHD|(Non-ADHD) gen\_attn|work\_att -0.328 0.136 511 -2.408 0.0164

mod <- lm(orderliness~adhd\*FL\_35\_DO, data = df.all)  
em.obj <- emmeans(mod,specs=c("adhd","FL\_35\_DO"))  
contrast(em.obj, method="mean\_chg",interaction=T)

## adhd\_mean\_chg FL\_35\_DO\_mean\_chg estimate SE df t.ratio p.value  
## ADHD|(Non-ADHD) gen\_attn|work\_att -0.276 0.162 511 -1.702 0.0894

mod <- lm(consc.gen~asrs\_sum+FL\_35\_DO, data = df.all)  
summary(mod)

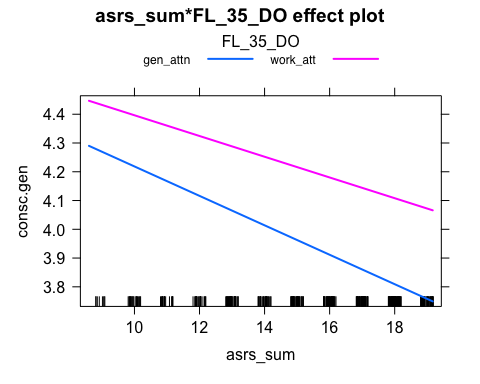
##   
## Call:  
## lm(formula = consc.gen ~ asrs\_sum + FL\_35\_DO, data = df.all)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.17266 -0.32879 0.07175 0.39111 1.07361   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.62474 0.06970 66.353 < 2e-16 \*\*\*  
## asrs\_sum -0.04353 0.00444 -9.804 < 2e-16 \*\*\*  
## FL\_35\_DOwork\_att 0.23667 0.04688 5.049 6.19e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.5318 on 512 degrees of freedom  
## Multiple R-squared: 0.19, Adjusted R-squared: 0.1869   
## F-statistic: 60.07 on 2 and 512 DF, p-value: < 2.2e-16

mod <- lm(consc.gen~asrs\_sum\*FL\_35\_DO, data = df.all)  
summary(mod)

##   
## Call:  
## lm(formula = consc.gen ~ asrs\_sum \* FL\_35\_DO, data = df.all)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.13296 -0.32046 0.08852 0.38019 1.09803   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 4.730245 0.093051 50.835 < 2e-16 \*\*\*  
## asrs\_sum -0.051173 0.006299 -8.124 3.41e-15 \*\*\*  
## FL\_35\_DOwork\_att 0.026530 0.131668 0.201 0.8404   
## asrs\_sum:FL\_35\_DOwork\_att 0.015135 0.008864 1.707 0.0883 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.5308 on 511 degrees of freedom  
## Multiple R-squared: 0.1946, Adjusted R-squared: 0.1899   
## F-statistic: 41.17 on 3 and 511 DF, p-value: < 2.2e-16

p\_load(effects)  
low.adhd <- mean(df.all$asrs\_sum) - sd(df.all$asrs\_sum)  
hi.adhd <- mean(df.all$asrs\_sum) + sd(df.all$asrs\_sum)  
em.obj <- effect(c("asrs\_sum\*FL\_35\_DO"),  
 mod = mod,   
 xlevels = list(asrs\_sum=c(low.adhd,  
 mean(df.all$asrs\_sum),  
 hi.adhd)))  
plot(em.obj,multiline = T)

## Warning in effect.llines(x[sub][good], y[sub][good], lwd = lwd, type = "l", :  
## spline interpolation may be unstable with only 3 points  
  
## Warning in effect.llines(x[sub][good], y[sub][good], lwd = lwd, type = "l", :  
## spline interpolation may be unstable with only 3 points



## Predictive validity

mod <- lm(total.cwb~consc.gen+adhd+FL\_35\_DO, data = df.all)  
summary(mod)

##   
## Call:  
## lm(formula = total.cwb ~ consc.gen + adhd + FL\_35\_DO, data = df.all)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.8773 -0.2903 -0.0762 0.1739 2.6791   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.32705 0.15279 21.775 <2e-16 \*\*\*  
## consc.gen -0.44044 0.03780 -11.653 <2e-16 \*\*\*  
## adhdNon-ADHD -0.01594 0.04371 -0.365 0.7156   
## FL\_35\_DOwork\_att 0.11279 0.04384 2.573 0.0104 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.4876 on 511 degrees of freedom  
## Multiple R-squared: 0.2176, Adjusted R-squared: 0.213   
## F-statistic: 47.37 on 3 and 511 DF, p-value: < 2.2e-16

mod <- lm(total.cwb~consc.gen\*adhd\*FL\_35\_DO, data = df.all)  
summary(mod)

##   
## Call:  
## lm(formula = total.cwb ~ consc.gen \* adhd \* FL\_35\_DO, data = df.all)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.83468 -0.29191 -0.07998 0.17227 2.66513   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 3.12018 0.27576 11.315 < 2e-16  
## consc.gen -0.39053 0.07010 -5.571 4.12e-08  
## adhdNon-ADHD 0.19936 0.41870 0.476 0.634  
## FL\_35\_DOwork\_att 0.44335 0.42920 1.033 0.302  
## consc.gen:adhdNon-ADHD -0.04861 0.10252 -0.474 0.636  
## consc.gen:FL\_35\_DOwork\_att -0.07636 0.10476 -0.729 0.466  
## adhdNon-ADHD:FL\_35\_DOwork\_att -0.14243 0.64283 -0.222 0.825  
## consc.gen:adhdNon-ADHD:FL\_35\_DOwork\_att 0.02641 0.15302 0.173 0.863  
##   
## (Intercept) \*\*\*  
## consc.gen \*\*\*  
## adhdNon-ADHD   
## FL\_35\_DOwork\_att   
## consc.gen:adhdNon-ADHD   
## consc.gen:FL\_35\_DOwork\_att   
## adhdNon-ADHD:FL\_35\_DOwork\_att   
## consc.gen:adhdNon-ADHD:FL\_35\_DOwork\_att   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.4889 on 507 degrees of freedom  
## Multiple R-squared: 0.2197, Adjusted R-squared: 0.2089   
## F-statistic: 20.39 on 7 and 507 DF, p-value: < 2.2e-16

mod <- lm(total.cwb~consc.gen\*asrs\_sum\*FL\_35\_DO, data = df.all)  
summary(mod)

##   
## Call:  
## lm(formula = total.cwb ~ consc.gen \* asrs\_sum \* FL\_35\_DO, data = df.all)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.85749 -0.27830 -0.06435 0.17654 2.63814   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.228833 0.739583 3.014 0.00271 \*\*  
## consc.gen -0.220692 0.166679 -1.324 0.18608   
## asrs\_sum 0.053953 0.045422 1.188 0.23547   
## FL\_35\_DOwork\_att -0.544937 1.120217 -0.486 0.62685   
## consc.gen:asrs\_sum -0.009948 0.010466 -0.951 0.34230   
## consc.gen:FL\_35\_DOwork\_att 0.096771 0.250208 0.387 0.69910   
## asrs\_sum:FL\_35\_DOwork\_att 0.049659 0.068807 0.722 0.47080   
## consc.gen:asrs\_sum:FL\_35\_DOwork\_att -0.007950 0.015672 -0.507 0.61217   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.4782 on 507 degrees of freedom  
## Multiple R-squared: 0.2533, Adjusted R-squared: 0.243   
## F-statistic: 24.57 on 7 and 507 DF, p-value: < 2.2e-16

mod <- lm(total.ocb~consc.gen+adhd+FL\_35\_DO, data = df.all)  
summary(mod)

##   
## Call:  
## lm(formula = total.ocb ~ consc.gen + adhd + FL\_35\_DO, data = df.all)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.00446 -0.52331 -0.00915 0.49620 2.13850   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.84137 0.24903 7.394 5.87e-13 \*\*\*  
## consc.gen 0.26394 0.06160 4.284 2.19e-05 \*\*\*  
## adhdNon-ADHD -0.11262 0.07124 -1.581 0.1146   
## FL\_35\_DOwork\_att -0.12670 0.07145 -1.773 0.0768 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7947 on 511 degrees of freedom  
## Multiple R-squared: 0.03754, Adjusted R-squared: 0.03189   
## F-statistic: 6.643 on 3 and 511 DF, p-value: 0.0002083

mod <- lm(total.ocb~consc.gen\*adhd+FL\_35\_DO, data = df.all)  
summary(mod)

##   
## Call:  
## lm(formula = total.ocb ~ consc.gen \* adhd + FL\_35\_DO, data = df.all)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.00450 -0.52330 -0.00917 0.49622 2.13850   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.8416124 0.3356445 5.487 6.46e-08 \*\*\*  
## consc.gen 0.2638797 0.0834024 3.164 0.00165 \*\*   
## adhdNon-ADHD -0.1131525 0.5094639 -0.222 0.82432   
## FL\_35\_DOwork\_att -0.1266993 0.0716336 -1.769 0.07754 .   
## consc.gen:adhdNon-ADHD 0.0001289 0.1215452 0.001 0.99915   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7955 on 510 degrees of freedom  
## Multiple R-squared: 0.03754, Adjusted R-squared: 0.02999   
## F-statistic: 4.973 on 4 and 510 DF, p-value: 0.0006122

mod <- lm(total.ocb~consc.gen\*asrs\_sum\*FL\_35\_DO, data = df.all)  
summary(mod)

##   
## Call:  
## lm(formula = total.ocb ~ consc.gen \* asrs\_sum \* FL\_35\_DO, data = df.all)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.1050 -0.5168 0.0184 0.5048 2.2893   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.61414 1.21646 -0.505 0.61388   
## consc.gen 0.76207 0.27415 2.780 0.00564 \*\*  
## asrs\_sum 0.14135 0.07471 1.892 0.05906 .   
## FL\_35\_DOwork\_att -1.45398 1.84252 -0.789 0.43041   
## consc.gen:asrs\_sum -0.02872 0.01721 -1.668 0.09589 .   
## consc.gen:FL\_35\_DOwork\_att 0.27055 0.41154 0.657 0.51122   
## asrs\_sum:FL\_35\_DOwork\_att 0.09228 0.11317 0.815 0.41523   
## consc.gen:asrs\_sum:FL\_35\_DOwork\_att -0.01931 0.02578 -0.749 0.45421   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7866 on 507 degrees of freedom  
## Multiple R-squared: 0.06459, Adjusted R-squared: 0.05167   
## F-statistic: 5.001 on 7 and 507 DF, p-value: 1.71e-05

Means by condition and ADHD status:

mean.sd.vars <- facet\_items$facet.vnames[!duplicated(facet\_items$facet.vnames)]  
mean.sd.vars

## [1] "self.efficacy" "orderliness"   
## [3] "dutifulness" "achievement.striving"   
## [5] "self.discipline" "cautiousness"   
## [7] "work.self.efficacy" "work.orderliness"   
## [9] "work.dutifulness" "work.achievement.striving"  
## [11] "work.self.discipline" "work.cautiousness"

ms.sds.adhd <- df.all %>%   
 group\_by(adhd, FL\_35\_DO) %>%   
 summarise(across(c(consc.gen,  
 self.efficacy, orderliness, dutifulness,  
 achievement.striving, self.discipline, cautiousness),  
 ~str\_c(number(mean(.x, na.rm=T), accuracy=.01),  
 " (",  
 number(sd(.x, na.rm=T), accuracy = .01),  
 ")"))) %>%   
 pivot\_longer(cols=c(consc.gen,  
 self.efficacy, orderliness, dutifulness,  
 achievement.striving, self.discipline, cautiousness)) %>%   
 pivot\_wider(names\_from=c(adhd, FL\_35\_DO),  
 values\_from=value) %>%   
 select(name, starts\_with("N", ignore.case=F), starts\_with("A",ignore.case=F)) #order columns to align with study 2 summary table

## `summarise()` has grouped output by 'adhd'. You can override using the  
## `.groups` argument.

write.csv(ms.sds.adhd, "means\_sds\_conscientiousness.csv")