BMI Percentile by SES and Individual Differences in Adolescents

Contents

Data for this study come from the subset of responses collected on the SAPA-project.org website between February 17, 2017 and July 22, 2019. The initial date is the day that the semi-random presentation of items to participants was changed to increase presentation of SPI-135 items, which are the basis for personality measurement in this study. This period also represents a new period of data collection on SAPA containing data that are not available in the public domain at the time of analysis. The end date of data collection was the first day following preregistration of analysis that the authors were able to analyze data.

All data cleaning and analyses were completed using R version 4.1.0 (2021-05-18) (Camp Pontanezen). Table **??** lists the packages (and versions) used in these analyses.

Table 1: Packages (and their versions) used for theses analyses.

Package	Version	Authors and contributors
kableExtra	1.3.4	Hao Zhu [aut, cre] (<https://orcid.org/0000-0002-3386-6076>), Thomas Travison [ctb], Timothy Tsai [ctb], Will Beasley [ctb], Yihui Xie [ctb], GuangChuang Yu [ctb], Stéphane Laurent [ctb], Rob Shepherd [ctb], Yoni Sidi [ctb], Brian Salzer [ctb], George Gui [ctb], Yeliang Fan [ctb], Duncan Murdoch [ctb], Bill Evans [ctb]
glmnet	4.1-2	Jerome Friedman [aut], Trevor Hastie [aut, cre], Rob Tibshirani [aut], Balasubramanian Narasimhan [aut], Kenneth Tay [aut], Noah Simon [aut], Junyang Qian [ctb]
Matrix	1.3-4	Douglas Bates [aut], Martin Maechler [aut, cre] (<https://orcid.org/0000-0002-8685-9910>), Timothy A. Davis [ctb] (SuiteSparse and 'cs' C libraries, notably CHOLMOD, AMD; collaborators listed in dir(pattern = '^[A-Z]+[.]txt$', full.names=TRUE, system.file('doc', 'SuiteSparse', package='Matrix'))), Jens Oehlschlägel [ctb] (initial nearPD()), Jason Riedy [ctb] (condest() and onenormest() for octave, Copyright: Regents of the University of California), R Core Team [ctb] (base R matrix implementation)
papaja	0.1.0.9997	Frederik Aust [aut, cre] (<https://orcid.org/0000-0003-4900-788X>), Marius Barth [aut] (<https://orcid.org/0000-0002-3421-6665>), Birk Diedenhofen [ctb], Christoph Stahl [ctb], Joseph V. Casillas [ctb], Rudolf Siegel [ctb]
ggpubr	0.4.0	Alboukadel Kassambara [aut, cre]
rsample	0.1.0	Julia Silge [aut, cre] (<https://orcid.org/0000-0002-3671-836X>), Fanny Chow [aut], Max Kuhn [aut], Hadley Wickham [aut], RStudio [cph]
sjPlot	2.8.9	Daniel Lüdecke [aut, cre] (<https://orcid.org/0000-0002-8895-3206>), Alexander Bartel [ctb] (<https://orcid.org/0000-0002-1280-6138>), Carsten Schwemmer [ctb], Chuck Powell [ctb] (<https://orcid.org/0000-0002-3606-2188>), Amir Djalovski [ctb], Johannes Titz [ctb] (<https://orcid.org/0000-0002-1102-5719>)

Table 1: Packages (and their versions) used for theses analyses.
(continued)

Package	Version	Authors and contributors
broom	0.7.9.9000	David Robinson [aut], Alex Hayes [aut] (<https://orcid.org/0000-0002-4985-5160>), Simon Couch [aut, cre] (<https://orcid.org/0000-0001-5676-5107>), Indrajeet Patil [ctb] (<https://orcid.org/0000-0003-1995-6531>), Derek Chiu [ctb], Matthieu Gomez [ctb], Boris Demeshev [ctb], Dieter Menne [ctb], Benjamin Nutter [ctb], Luke Johnston [ctb], Ben Bolker [ctb], Francois Briatte [ctb], Jeffrey Arnold [ctb], Jonah Gabry [ctb], Luciano Selzer [ctb], Gavin Simpson [ctb], Jens Preussner [ctb], Jay Hesselberth [ctb], Hadley Wickham [ctb], Matthew Lincoln [ctb], Alessandro Gasparini [ctb], Lukasz Komsta [ctb], Frederick Novometsky [ctb], Wilson Freitas [ctb], Michelle Evans [ctb], Jason Cory Brunson [ctb], Simon Jackson [ctb], Ben Whalley [ctb], Karissa Whiting [ctb], Yves Rosseel [ctb], Michael Kuehn [ctb], Jorge Cimentada [ctb], Erle Holgersen [ctb], Karl Dunkle Werner [ctb] (<https://orcid.org/0000-0003-0523-7309>), Ethan Christensen [ctb], Steven Pav [ctb], Paul PJ [ctb], Ben Schneider [ctb], Patrick Kennedy [ctb], Lily Medina [ctb], Brian Fannin [ctb], Jason Muhlenkamp [ctb], Matt Lehman [ctb], Bill Denney [ctb] (<https://orcid.org/0000-0002-5759-428X>), Nic Crane [ctb], Andrew Bates [ctb], Vincent Arel-Bundock [ctb] (<https://orcid.org/0000-0003-2042-7063>), Hideaki Hayashi [ctb], Luis Tobalina [ctb], Annie Wang [ctb], Wei Yang Tham [ctb], Clara Wang [ctb], Abby Smith [ctb] (<https://orcid.org/0000-0002-3207-0375>), Jasper Cooper [ctb] (<https://orcid.org/0000-0002-8639-3188>), E Auden Krauska [ctb] (<https://orcid.org/0000-0002-1466-5850>), Alex Wang [ctb], Malcolm Barrett [ctb] (<https://orcid.org/0000-0003-0299-5825>), Charles Gray [ctb] (<https://orcid.org/0000-0002-9978-011X>), Jared Wilber [ctb], Vilmantas Gegzna [ctb] (<https://orcid.org/0000-0002-9500-5167>), Eduard Szoecs [ctb], Frederik Aust [ctb] (<https://orcid.org/0000-0003-4900-788X>), Angus Moore [ctb], Nick Williams [ctb], Marius Barth [ctb] (<https://orcid.org/0000-0002-3421-6665>), Bruna Wundervald [ctb] (<https://orcid.org/0000-0001-8163-220X>), Joyce Cahoon [ctb] (<https://orcid.org/0000-0001-7217-4702>), Grant McDermott [ctb] (<https://orcid.org/0000-0001-7883-8573>), Kevin Zarca [ctb], Shiro Kuriwaki [ctb] (<https://orcid.org/0000-0002-5687-2647>), Lukas Wallrich [ctb] (<https://orcid.org/0000-0003-2121-5177>), James Martherus [ctb] (<https://orcid.org/0000-0002-8285-3300>), Chuliang Xiao [ctb] (<https://orcid.org/0000-0002-8466-9398>), Joseph Larmarange [ctb], Max Kuhn [ctb], Michal Bojanowski [ctb], Hakon Malmedal [ctb], Clara Wang [ctb], Sergio Oller [ctb], Luke Sonnet [ctb], Jim Hester [ctb], Cory Brunson [ctb], Ben Schneider [ctb], Bernie Gray [ctb] (<https://orcid.org/0000-0001-9190-6032>), Mara Averick [ctb], Aaron Jacobs [ctb], Andreas Bender [ctb], Sven Templer [ctb], Paul-Christian Buerkner [ctb], Matthew Kay [ctb], Erwan Le Pennec [ctb], Johan Junkka [ctb], Hao Zhu [ctb], Benjamin Soltoff [ctb], Zoe Wilkinson Saldana [ctb], Tyler Littlefield [ctb], Charles T. Gray [ctb], Shabbh E. Banks [ctb], Serina Robinson [ctb], Roger Bivand [ctb], Riinu Ots [ctb], Nicholas Williams [ctb], Nina Jakobsen [ctb], Michael Weylandt [ctb], Lisa Lendway [ctb], Karl Hailperin [ctb], Josue Rodriguez [ctb], Jenny Bryan [ctb], Chris Jarvis [ctb], Greg Macfarlane [ctb], Brian Mannakee [ctb], Drew Tyre [ctb], Shreyas Singh [ctb], Laurens Geffert [ctb], Hong Ooi [ctb], Henrik Bengtsson [ctb], Eduard Szocs [ctb], David Hugh-Jones [ctb], Matthieu Stigler [ctb], Hugo Tavares [ctb] (<https://orcid.org/0000-0001-9373-2726>), R. Willem Vervoort [ctb], Brenton M. Wiernik [ctb], Josh Yamamoto [ctb], Jasme Lee [ctb], Taren Sanders [ctb] (<https://orcid.org/0000-0002-4504-6008>)

Table 1: Packages (and their versions) used for theses analyses. *(continued)*

Package	Version	Authors and contributors
corrplot	0.90	Taiyun Wei [cre, aut], Viliam Simko [aut], Michael Levy [ctb], Yihui Xie [ctb], Yan Jin [ctb], Jeff Zemla [ctb], Moritz Freidank [ctb], Jun Cai [ctb], Tomas Protivinsky [ctb]
caret	6.0-88	Max Kuhn [aut, cre], Jed Wing [ctb], Steve Weston [ctb], Andre Williams [ctb], Chris Keefer [ctb], Allan Engelhardt [ctb], Tony Cooper [ctb], Zachary Mayer [ctb], Brenton Kenkel [ctb], R Core Team [ctb], Michael Benesty [ctb], Reynald Lescarbeau [ctb], Andrew Ziem [ctb], Luca Scrucca [ctb], Yuan Tang [ctb], Can Candan [ctb], Tyler Hunt [ctb]
lattice	0.20-44	Deepayan Sarkar [aut, cre] (<https://orcid.org/0000-0003-4107-1553>), Felix Andrews [ctb], Kevin Wright [ctb] (documentation), Neil Klepeis [ctb], Johan Larsson [ctb] (colorkey title), Paul Murrell [ctb]
measurements	1.4.0	Matthew A. Birk
PAutilities	1.0.1	Paul R. Hibbing [aut, cre], Centers for Disease Control and Prevention [ctb]
devtools	2.4.2	Hadley Wickham [aut], Jim Hester [aut, cre], Winston Chang [aut], RStudio [cph, fnd]
usethis	2.0.1	Hadley Wickham [aut] (<https://orcid.org/0000-0003-4757-117X>), Jennifer Bryan [aut, cre] (<https://orcid.org/0000-0002-6983-2759>), RStudio [cph, fnd]
psych	2.1.6	William Revelle [aut, cre] (<https://orcid.org/0000-0003-4880-9610>)
janitor	2.1.0	Sam Firke [aut, cre], Bill Denney [ctb], Chris Haid [ctb], Ryan Knight [ctb], Malte Grosser [ctb], Jonathan Zadra [ctb]
ggthemr	1.1.0	Ciaran Tobin [aut], Amy Tzu-Yu Chen [cre] (<https://orcid.org/0000-0002-4085-572X>), Gergely Daróczi [ctb] (<https://orcid.org/0000-0003-3149-8537>), Mikata [fnd]
forcats	0.5.1	Hadley Wickham [aut, cre], RStudio [cph, fnd]
stringr	1.4.0	Hadley Wickham [aut, cre, cph], RStudio [cph, fnd]
dplyr	1.0.7	Hadley Wickham [aut, cre] (<https://orcid.org/0000-0003-4757-117X>), Romain François [aut] (<https://orcid.org/0000-0002-2444-4226>), Lionel Henry [aut], Kirill Müller [aut] (<https://orcid.org/0000-0002-1416-3412>), RStudio [cph, fnd]
purrr	0.3.4	Lionel Henry [aut, cre], Hadley Wickham [aut], RStudio [cph, fnd]
readr	1.4.0	Hadley Wickham [aut], Jim Hester [aut, cre], Romain Francois [ctb], R Core Team [ctb] (Date time code adapted from R), RStudio [cph, fnd], Jukka Jylänki [ctb, cph] (grisu3 implementation), Mikkel Jørgensen [ctb, cph] (grisu3 implementation)
tidyr	1.1.3	Hadley Wickham [aut, cre], RStudio [cph]
tibble	3.1.4	Kirill Müller [aut, cre], Hadley Wickham [aut], Romain Francois [ctb], Jennifer Bryan [ctb], RStudio [cph]
ggplot2	3.3.5	Hadley Wickham [aut] (<https://orcid.org/0000-0003-4757-117X>), Winston Chang [aut] (<https://orcid.org/0000-0002-1576-2126>), Lionel Henry [aut], Thomas Lin Pedersen [aut, cre] (<https://orcid.org/0000-0002-5147-4711>), Kohske Takahashi [aut], Claus Wilke [aut] (<https://orcid.org/0000-0002-7470-9261>), Kara Woo [aut] (<https://orcid.org/0000-0002-5125-4188>), Hiroaki Yutani [aut] (<https://orcid.org/0000-0002-3385-7233>), Dewey Dunnington [aut] (<https://orcid.org/0000-0002-9415-4582>), RStudio [cph, fnd]
tidyverse	1.3.1	Hadley Wickham [aut, cre], RStudio [cph, fnd]
here	1.0.1	Kirill Müller [aut, cre] (<https://orcid.org/0000-0002-1416-3412>), Jennifer Bryan [ctb] (<https://orcid.org/0000-0002-6983-2759>)

Table 1: Packages (and their versions) used for theses analyses.
(continued)

Package	Version	Authors and contributors
knitr	1.34	Yihui Xie [aut, cre] (<https://orcid.org/0000-0003-0645-5666>), Abhraneel Sarma [ctb], Adam Vogt [ctb], Alastair Andrew [ctb], Alex Zvoleff [ctb], Andre Simon [ctb] (the CSS files under inst/themes/ were derived from the Highlight package http://www.andre-simon.de), Aron Atkins [ctb], Aaron Wolen [ctb], Ashley Manton [ctb], Atsushi Yasumoto [ctb] (<https://orcid.org/0000-0002-8335-495X>), Ben Baumer [ctb], Brian Diggs [ctb], Brian Zhang [ctb], Bulat Yapparov [ctb], Cassio Pereira [ctb], Christophe Dervieux [ctb], David Hall [ctb], David Hugh-Jones [ctb], David Robinson [ctb], Doug Hemken [ctb], Duncan Murdoch [ctb], Elio Campitelli [ctb], Ellis Hughes [ctb], Emily Riederer [ctb], Fabian Hirschmann [ctb], Fitch Simeon [ctb], Forest Fang [ctb], Frank E Harrell Jr [ctb] (the Sweavel package at inst/misc/Sweavel.sty), Garrick Aden-Buie [ctb], Gregoire Detrez [ctb], Hadley Wickham [ctb], Hao Zhu [ctb], Heewon Jeon [ctb], Henrik Bengtsson [ctb], Hiroaki Yutani [ctb], Ian Lyttle [ctb], Hodges Daniel [ctb], Jake Burkhead [ctb], James Manton [ctb], Jared Lander [ctb], Jason Punyon [ctb], Javier Luraschi [ctb], Jeff Arnold [ctb], Jenny Bryan [ctb], Jeremy Ashkenas [ctb, cph] (the CSS file at inst/misc/docco-classic.css), Jeremy Stephens [ctb], Jim Hester [ctb], Joe Cheng [ctb], Johannes Ranke [ctb], John Honaker [ctb], John Muschelli [ctb], Jonathan Keane [ctb], JJ Allaire [ctb], Johan Toloe [ctb], Jonathan Sidi [ctb], Joseph Larmarange [ctb], Julien Barnier [ctb], Kaiyin Zhong [ctb], Kamil Slowikowski [ctb], Karl Forner [ctb], Kevin K. Smith [ctb], Kirill Mueller [ctb], Kohske Takahashi [ctb], Lorenz Walthert [ctb], Lucas Gallindo [ctb], Marius Hofert [ctb], Martin Modrák [ctb], Michael Chirico [ctb], Michael Friendly [ctb], Michal Bojanowski [ctb], Michel Kuhlmann [ctb], Miller Patrick [ctb], Nacho Caballero [ctb], Nick Salkowski [ctb], Niels Richard Hansen [ctb], Noam Ross [ctb], Obada Mahdi [ctb], Pavel N. Krivitsky [ctb] (<https://orcid.org/0000-0002-9101-3362>), Qiang Li [ctb], Ramnath Vaidyanathan [ctb], Richard Cotton [ctb], Robert Krzyzanowski [ctb], Romain Francois [ctb], Ruaridh Williamson [ctb], Scott Kostyshak [ctb], Sebastian Meyer [ctb], Sietse Brouwer [ctb], Simon de Bernard [ctb], Sylvain Rousseau [ctb], Taiyun Wei [ctb], Thibaut Assus [ctb], Thibaut Lamadon [ctb], Thomas Leeper [ctb], Tim Mastny [ctb], Tom Torsney-Weir [ctb], Trevor Davis [ctb], Viktoras Veitas [ctb], Weicheng Zhu [ctb], Wush Wu [ctb], Zachary Foster [ctb]

1 Clean data

```r
set.seed(052319)

#read in data
load(here("../../SAPA data/original data/SAPAdata07feb2017thru22jul2019forSara2.rdata"))
sapa = SAPAdata07feb2017thru22jul2019x

source(here("scripts/personality_scales.R"))
keys = read.csv("data/superKey.csv", header = TRUE, row.names = 1)
```

Participants were included in the analysis if they were under the age of 18, from the United States, and had reported their biological sex at birth, height, and weight.

```r
# remove participants who are 18 years or older and from the US
sapa = sapa %>%
  filter(age < 18) %>%
  filter(country == "USA") %>%
  filter(!is.na(sex)) %>%
  filter(!is.na(height)) %>%
  filter(!is.na(weight)) %>%
  filter(!is.na(p1edu) | !is.na(p2edu) |
           !is.na(p1occIncomeEst) | !is.na(p2occIncomeEst) |
           !is.na(p1occPrestige) | !is.na(p2occPrestige))
```

Parental education was transformed into a numeric variable which indexes the (expected) number of years to complete the degree. All parental SES variables – education, estimated income and estimated prestige, were standardized to the sample and averaged to create a single index of parental SES.

```r
# make sure occupational variables are numeric
sapa = sapa %>%
  mutate_at(vars(matches("^(p)\\d(occ)")), as.numeric)

#or years
sapa = sapa %>%
  mutate(p1edu = case_when(
    p2edu == "less12yrs" ~ "6",
    p2edu == "HSgrad" ~ "12",
    p2edu == "SomeCollege" ~ "14",
    p2edu == "CurrentInUniv" ~ "14",
    p2edu == "AssociateDegree" ~ "14",
    p2edu == "CollegeDegree" ~ "16",
    p2edu == "InGradOrProSchool" ~ "18",
    p2edu == "GradOrProDegree" ~ "20"))

sapa = sapa %>%
  mutate(p2edu = case_when(
    p2edu == "less12yrs" ~ "6",
    p2edu == "HSgrad" ~ "12",
    p2edu == "SomeCollege" ~ "14",
    p2edu == "CurrentInUniv" ~ "14",
    p2edu == "AssociateDegree" ~ "14",
```

```r
    p2edu == "CollegeDegree" ~ "16",
    p2edu == "InGradOrProSchool" ~ "18",
    p2edu == "GradOrProDegree" ~ "20"))

sapa$p1edu = as.numeric(sapa$p1edu)
sapa$p2edu = as.numeric(sapa$p2edu)

#estimate SES composite

sapa = sapa %>%
  mutate(z.p1edu = scale(p1edu),
         z.p2edu = scale(p2edu),
         z.p1occIncomeEst = scale(p1occIncomeEst),
         z.p2occIncomeEst = scale(p2occIncomeEst),
         z.p1occPrestige = scale(p1occPrestige),
         z.p2occPrestige = scale(p2occPrestige))

sapa$ses = rowMeans(sapa[,grepl("^z\\.", names(sapa))], na.rm=T)

sapa = sapa %>%
  dplyr::select(-starts_with("z"))
```

Big Five traits were scored using a sum-score method, averaged across non-missing responses.

```r
# select just the rows that correspond to variables in the current SAPA dataset
vars = names(sapa)
keys = keys[rownames(keys) %in% vars, ]

# select just the Big 5 scales that are scored using the SPI_135 form
bfkeys = keys %>%
  select(contains("SPI_135")) %>%
  select(1:5)

bfkeys = keys2list(as.matrix(bfkeys), sign = T)


# score the items (this contains item and scale statistics too!)
b5scored = scoreItems(keys = bfkeys, items = sapa)

# add scores to SAPA
b5scores = as.data.frame(b5scored$scores[,1:5])
names(b5scores) = gsub("135_27_5_", "", names(b5scores))
sapa = cbind(sapa, b5scores)
```

The narrower traits, the SPI-27, were scored using IRT scoring. Calibration parameters were taken from a different dataset and are available on request.

```r
load(here("../../SAPA data/created/IRTinfoSPI27.rdata"))

# IRT score
dataSet <- subset(sapa, select = c(orderForItems))

SPIirtScores <- matrix(nrow=dim(dataSet)[1], ncol=27)
```

```r
scaleNames = gsub("SPI27_", "", names(IRToutputSPI27))
spi_keys = keys %>%
  select(matches("SPI_135")) %>%
  select(-c(1:5)) %>%
  mutate(item = rownames(.)) %>%
  gather("scale", "key", -item) %>%
  filter(key != 0)

for (i in 1:length(IRToutputSPI27)) {
  data <- subset(dataSet, select = c(rownames(IRToutputSPI27[[i]]$irt$difficulty[[1]])))
  calibrations <- IRToutputSPI27[[i]]
  #check calibration direction
  loadings = calibrations$fa$loadings[,1]
  loadings = ifelse(loadings < 0, -1, 1)
  loadings = data.frame(item = names(loadings), loadings = loadings)
  keys_direction = spi_keys %>%
    filter(grepl(scaleNames[i], scale)) %>%
    full_join(loadings)
  same = sum(keys_direction$key == keys_direction$loadings)
  if(same == 0) data[,1:ncol(data)] = apply(data[,1:ncol(data)], 2, function(x) max(x, na.rm=T) + 1 - x)
  if (same > 0 & same < 5) print("Error in loadings")
  scored <- scoreIrt(calibrations, data, keys = NULL, cut = 0)
  trait_scores = scored$theta1
  trait_scores = (trait_scores - mean(trait_scores, na.rm = T))/sd(trait_scores, na.rm=T)
  Tscores = trait_scores*10 + 50
  SPIirtScores[,i] <- Tscores
}

SPIirtScores <- as.data.frame(SPIirtScores)
colnames(SPIirtScores) <- paste0("SPI_", scaleNames)

#add to sapa dataset
sapa = cbind(sapa, SPIirtScores)
```

Cognition was also scored using IRT scoring, with calibrations from a separate dataset.

```r
load(here("../../SAPA data/created/IRTinfoSPI27.rdata"))

# IRT score
dataSet <- subset(sapa, select = c(orderForItems))

SPIirtScores <- matrix(nrow=dim(dataSet)[1], ncol=27)

scaleNames = gsub("SPI27_", "", names(IRToutputSPI27))
spi_keys = keys %>%
  select(matches("SPI_135")) %>%
  select(-c(1:5)) %>%
  mutate(item = rownames(.)) %>%
  gather("scale", "key", -item) %>%
  filter(key != 0)

for (i in 1:length(IRToutputSPI27)) {
  data <- subset(dataSet, select = c(rownames(IRToutputSPI27[[i]]$irt$difficulty[[1]])))
```

```
  calibrations <- IRToutputSPI27[[i]]
  #check calibration direction
  loadings = calibrations$fa$loadings[,1]
  loadings = ifelse(loadings < 0, -1, 1)
  loadings = data.frame(item = names(loadings), loadings = loadings)
  keys_direction = spi_keys %>%
    filter(grepl(scaleNames[i], scale)) %>%
    full_join(loadings)
  same = sum(keys_direction$key == keys_direction$loadings)
  if(same == 0) data[,1:ncol(data)] = apply(data[,1:ncol(data)], 2, function(x) max(x, na.rm=T) + 1 - x)
  if (same > 0 & same < 5) print("Error in loadings")
  scored <- scoreIrt(calibrations, data, keys = NULL, cut = 0)
  trait_scores = scored$theta1
  trait_scores = (trait_scores - mean(trait_scores, na.rm = T))/sd(trait_scores, na.rm=T)
  Tscores = trait_scores*10 + 50
  SPIirtScores[,i] <- Tscores
}

SPIirtScores <- as.data.frame(SPIirtScores)
colnames(SPIirtScores) <- paste0("SPI_", scaleNames)

#add to sapa dataset
sapa = cbind(sapa, SPIirtScores)
```

BMI percentile represents a participant's percentile score on BMI relative to others of their assigned sex at birth. These were estimated from the `PAutilities` package (Hibbing, 2020), developed by WHO Multicentre Growth Reference Study (MGRS). Information about the development of these reference standards can be found at https://www.cdc.gov/obesity/childhood/defining.html. These standards in turn were developed using the 2000 CDC growth charts (Kuczmarski, 2002), based on data from 5 national health examination surveys that occurred from 1963 to 1994 and supplemental data from surveys that occurred from 1960 to 1995.

BMI category is assigned based on BMI percentile: participants in the bottom 10% are labeled Underweight, between the top 10% and 5% are Overweight, and top 5% are Obese. All others are labeled Normal.

```
sapa = sapa %>%
  filter(sex != "other") %>%
  mutate(sex = as.factor(as.character(sex))) %>%
  mutate(sex2 = ifelse(sex == "male", "M", "F"),
         weight = conv_unit(weight, from = "lbs", to = "kg"),
         height = conv_unit(height, from = "inch", to = "cm"))

for(i in 1:nrow(sapa)){
  sapa$BMI_p[i] = get_BMI_percentile(weight_kg = sapa$weight[i],
                                     height = sapa$height[i],
                                     age_yrs = sapa$age[i],
                                     sex = sapa$sex2[i],
                                     output = "percentile")
  sapa$BMI_c[i] = as.character(
    get_BMI_percentile(weight_kg = sapa$weight[i],
                       height = sapa$height[i],
                       age_yrs = sapa$age[i],
                       sex = sapa$sex2[i],
```

```
                        output = "class"))
}
```

All analyses were performed separately by gender.

```
sapa = sapa %>%
  mutate(cog = ICAR60) %>%
  select(sex, age, height, weight, BMI, BMI_p, BMI_c, p1edu,
         p1occPrestige, p1occIncomeEst, p2edu,
         p2occPrestige, p2occIncomeEst, ses, cog, contains("SPI"))

sapa_male = sapa %>%
  filter(sex == "male") %>%
  dplyr::select(-sex)

sapa_female = sapa %>%
  filter(sex == "female") %>%
  dplyr::select(-sex)

save(b5scored, file = here("data/alpha.Rdata"))
```

The datasets were split into training (75%) and test (25%) sets; all regression models are estimated using the training sets. The test set was reserved to estimate model accuarcy, comparing models with different sets of individual differences.

```
# set seed
set.seed(090919)

# parition into training and test sets. objects identify just training rows
train_male = createDataPartition(sapa_male$BMI_c, p = .75, list = FALSE)
train_female = createDataPartition(sapa_female$BMI_c, p = .75, list = FALSE)

# ---- save data -----
save(sapa,
     sapa_male, sapa_female,
     train_male, train_female, file = here("data/cleaned.Rdata"))

save(b5scored,
     file = here("data/reliability.Rdata"))
```

2 Descriptive Statistics

2.1 Univariate descriptives

Descriptive statistics are estimated using the `psych` package (Revelle, 2021).

```
descriptives = describeBy(sapa, group = "sex")
```

```
save(descriptives, file = "data/descriptives.Rdata")
```

```
#pull descriptive statistics into a list
descriptives.df = data.frame(gender = names(descriptives))
descriptives.df$data = descriptives

#add variable names and unnest
descriptives.df = descriptives.df %>%
  mutate(data = map(data, function(x) mutate(x, vars = rownames(x)))) %>%
  unnest(cols = c(data))
```

2.1.1 Descriptives Table by Gender

```
descriptives.df %>%
  select(gender, vars, n, mean, sd, min, max) %>%
  gather(stat, value, -gender, -vars) %>%
  unite(stat, stat, gender) %>%
  spread(stat, value) %>%
  select(vars, n_female, mean_female, sd_female, min_female, max_female,
         n_male, mean_male, sd_male, min_male, max_male) %>%
  kable(.,
        col.names = c("Variable", rep(c("N", "Mean", "SD", "Min", "Max"), 2)),
        digits = 2,
        caption = "Univariate descriptive statistics of study variables, broken down by gender",
        longtable = T,
        booktabs = T) %>%
  kable_styling() %>%
  landscape() %>%
  add_header_above(c(" " = 1, "Female" = 5, "Male" = 5))
```

Table 2: Univariate descriptive statistics of study variables, broken down by gender

	Female					Male				
Variable	N	Mean	SD	Min	Max	N	Mean	SD	Min	Max
BMI	6502	23.07	5.00	15.01	52.78	2934	22.84	4.90	15.01	53.76
BMI_c*	6530	1.51	0.84	1.00	4.00	2952	1.57	0.91	1.00	4.00
BMI_p	6530	62.70	27.61	0.00	99.80	2952	60.00	30.53	0.00	99.90
cog	6507	48.06	9.19	18.46	76.53	2948	49.65	9.60	19.86	76.53
p1edu	5876	14.62	3.66	6.00	20.00	2660	14.71	3.58	6.00	20.00
p1occIncomeEst	5594	61625.23	21784.89	21980.00	112490.00	2570	61491.45	22195.84	21980.00	112490.00
p1occPrestige	5723	60.76	14.64	24.22	79.09	2620	60.20	15.22	24.22	79.09
p2edu	5876	14.62	3.66	6.00	20.00	2660	14.71	3.58	6.00	20.00
p2occIncomeEst	4729	59058.07	22926.91	21980.00	112490.00	2111	57247.11	22364.35	21980.00	112490.00
p2occPrestige	4818	57.87	15.76	24.22	79.09	2147	57.07	15.59	24.22	79.09
ses	6452	-0.03	0.78	-2.38	1.55	2917	-0.05	0.79	-2.38	1.55
sex*	6530	1.00	0.00	1.00	1.00	2952	2.00	0.00	2.00	2.00
SPI_Adaptability	6488	49.44	10.12	28.77	70.29	2921	51.24	9.60	28.77	70.29
SPI_Agree	6530	4.26	0.67	1.00	6.00	2952	4.11	0.70	1.00	6.00
SPI_Anxiety	6530	51.88	8.90	19.27	62.10	2952	45.79	10.97	17.73	62.10
SPI_ArtAppreciation	6488	51.35	8.94	15.08	72.72	2921	46.96	11.47	15.08	72.72
SPI_AttentionSeeking	6530	49.68	10.08	26.43	66.95	2952	50.72	9.77	26.43	65.70
SPI_Authoritarianism	6489	50.55	9.69	15.22	67.09	2925	48.87	10.55	13.46	67.09
SPI_Charisma	6487	49.74	10.00	22.58	71.58	2926	50.59	9.97	21.90	71.58
SPI_Compassion	6530	51.31	9.51	15.93	63.13	2952	47.10	10.40	15.93	63.13
SPI_Conformity	6489	50.59	9.93	28.22	71.14	2926	48.76	10.03	28.22	70.64
SPI_Consc	6530	3.87	0.67	1.00	6.00	2952	3.77	0.63	1.43	6.00
SPI_Conservatism	6480	49.54	10.13	28.73	70.97	2919	51.07	9.62	28.73	70.97
SPI_Creativity	6530	49.72	10.03	18.38	64.67	2952	50.56	9.92	18.38	64.67
SPI_EasyGoingness	6485	49.56	9.99	13.49	69.53	2920	50.97	9.98	13.49	69.53
SPI_EmotionalExpressiveness	6489	50.36	10.04	32.70	71.81	2923	49.18	9.84	32.70	71.81
SPI_EmotionalStability	6529	48.20	10.02	28.98	68.44	2952	54.01	8.69	28.98	67.27
SPI_Extra	6530	3.61	0.83	1.00	6.00	2952	3.62	0.81	1.00	6.00
SPI_Honesty	6530	50.65	9.56	3.87	77.00	2952	48.62	10.74	3.87	77.00
SPI_Humor	6489	50.65	9.62	7.52	64.74	2923	48.57	10.62	7.52	64.74

SPI_Impulsivity	6488	49.85	10.09	31.84	72.72	2921	50.31	9.78	31.84	72.72
SPI_Industry	6530	50.29	10.01	27.91	73.23	2952	49.39	9.93	28.29	73.92
SPI_Intellect	6530	49.36	10.20	14.10	65.73	2952	51.43	9.39	14.10	65.73
SPI_Introspection	6529	50.04	9.89	15.16	62.58	2951	49.92	10.21	15.16	62.58
SPI_Irritability	6530	50.97	9.87	27.87	72.50	2952	47.83	9.94	27.87	72.50
SPI_Neuro	6530	4.31	0.77	1.29	6.00	2952	3.79	0.81	1.00	6.00
SPI_Open	6530	4.48	0.55	2.07	6.00	2952	4.55	0.53	2.21	6.00
SPI_Order	6529	50.18	10.06	25.20	74.44	2951	49.64	9.87	25.20	74.44
SPI_Perfectionism	6530	50.73	9.83	19.29	69.55	2952	48.37	10.18	18.72	69.55
SPI_SelfControl	6485	49.31	9.86	25.93	74.34	2919	51.54	10.12	25.93	74.34
SPI_SensationSeeking	6488	49.55	10.03	29.19	71.62	2921	50.98	9.86	30.74	71.42
SPI_Sociability	6530	49.82	10.02	15.89	66.44	2952	50.46	9.87	15.89	66.44
SPI_Trust	6530	49.83	10.00	27.80	73.20	2952	50.44	9.94	27.80	73.20
SPI_WellBeing	6488	49.03	9.80	26.47	74.40	2925	52.24	10.04	26.47	74.40

2.1.2 Distribution of BMI

We present the distribution of BMI percentile in Figure 1.

```
dens_f = density(sapa_female$BMI_p)
df_f = data.frame(BMI = dens_f$x,
                  y = dens_f$y,
                  gender = "Adolescent Females")

dens_m = density(sapa_male$BMI_p)
df_m = data.frame(BMI = dens_m$x,
                  y = dens_m$y,
                  gender = "Adolescent Males")
df_f = df_f %>%
  full_join(df_m) %>%
  mutate(quantile = case_when(
    BMI < 5 ~ "Underweight",
    BMI < 85 ~ "Normal",
    BMI < 95 ~ "Overweight",
    TRUE ~ "Obese"),
    quantile = factor(quantile,
                      levels = c("Underweight", "Normal", "Overweight", "Obese")))

df_f %>%
  ggplot(aes(x = BMI, y = y)) +
  geom_line() +
  geom_ribbon(aes(ymin=0, ymax=y, fill=quantile)) +
  scale_fill_brewer() +
  scale_x_continuous(limits = c(0,100)) +
  labs(x = "BMI Percentile", y = "Density", fill = "CDC weight categories") +
  facet_wrap(~gender) +
  theme_pubr()
```

```
ggsave(here("figures/BMI distributions.jpeg"), width = 7.5, height = 4.5)
```

2.2 Bivariate

```
R_male = sapa_male %>%
  dplyr::select(-BMI_c) %>%
  cor(use = "pairwise")

R_female = sapa_female %>%
  dplyr::select(-BMI_c) %>%
  cor(use = "pairwise")

#predictors
pred = names(sapa_male) %>% str_subset("BMI", negate = TRUE)


r_bmi_male = corr.test(x = sapa_male$BMI, y = sapa_male[,pred])
r_bmi_female = corr.test(x = sapa_female$BMI, y = sapa_female[,pred])
```

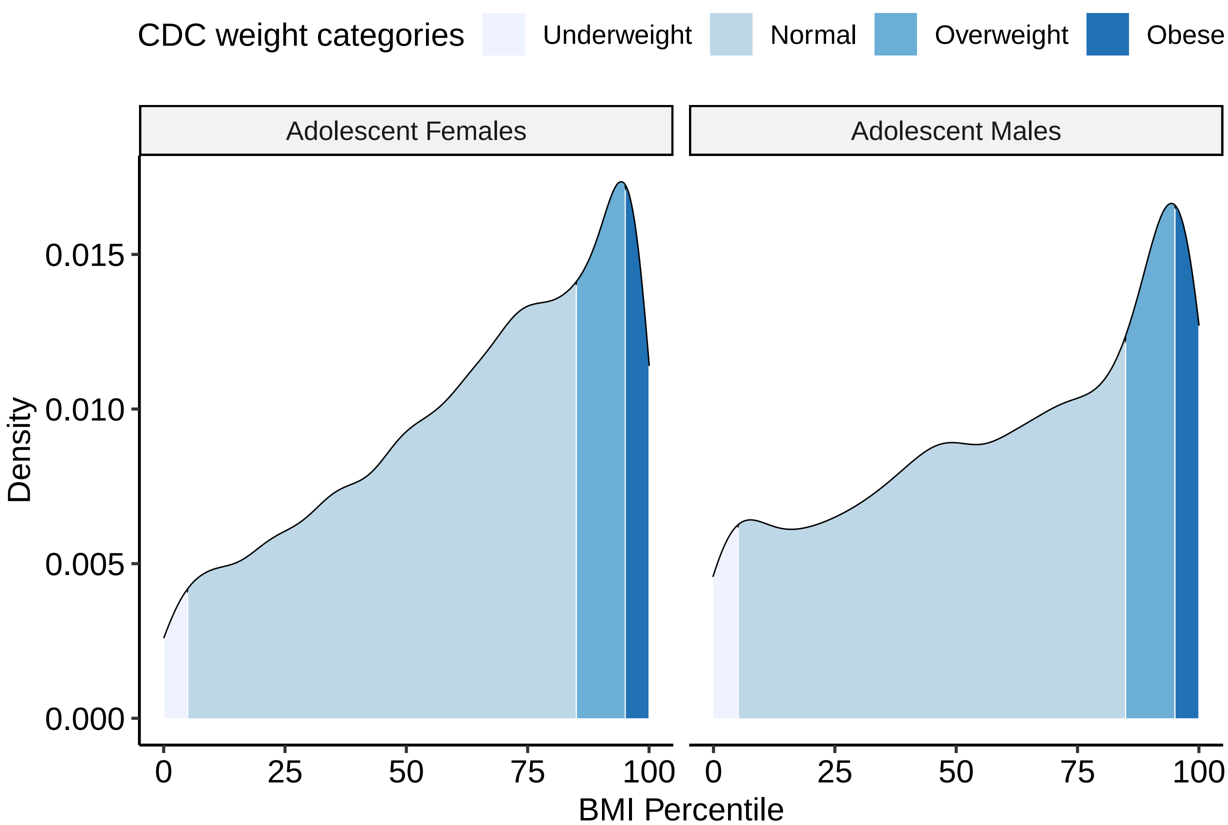

Figure 1: BMI percentile distributions by gender.

```r
r_bmi_male = modify(r_bmi_male, as.vector)
r_bmi_female = modify(r_bmi_female, as.vector)

cor.data = data.frame(gender = c("male", "female"))
cor.data$fullr = list(r_bmi_male, r_bmi_female)


cor.data = cor.data %>%
  mutate(r = map(fullr, "r")) %>%
  mutate(r = map(r, unlist)) %>%
  mutate(rp = map(fullr, "p")) %>%
  mutate(rp = map(rp, unlist)) %>%
  dplyr::select(-fullr) %>%
  unnest(cols = c(r, rp)) %>%
  mutate(pred = rep(pred,2)) %>%
  gather("key", "value", -gender, -pred) %>%
  unite(gender, gender, key) %>%
  spread(gender, value)

save(R_male, R_female, cor.data, file = "data/cor_output.Rdata")
```

2.2.1 Females

```r
corrplot(R_female, method = "square",
         title = NULL,
         tl.col = "black",
         mar=c(0,0,1,0))
```

2.2.2 Males

```r
corrplot(R_male, method = "square",
         title = NULL,
         tl.col = "black",
         mar=c(0,0,1,0))
```

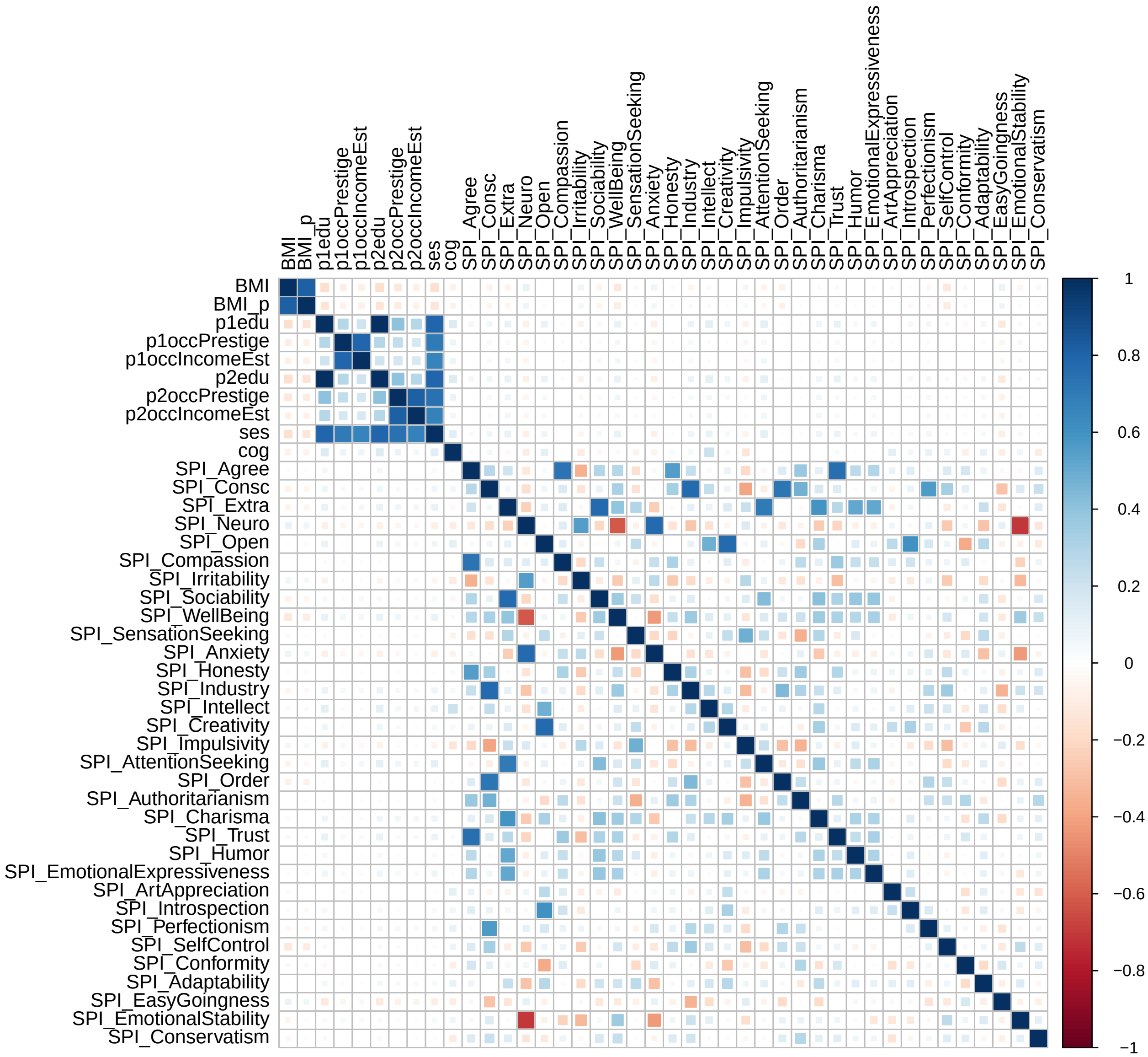

Figure 2: Zero-order correlations among study variables (Female Participants)

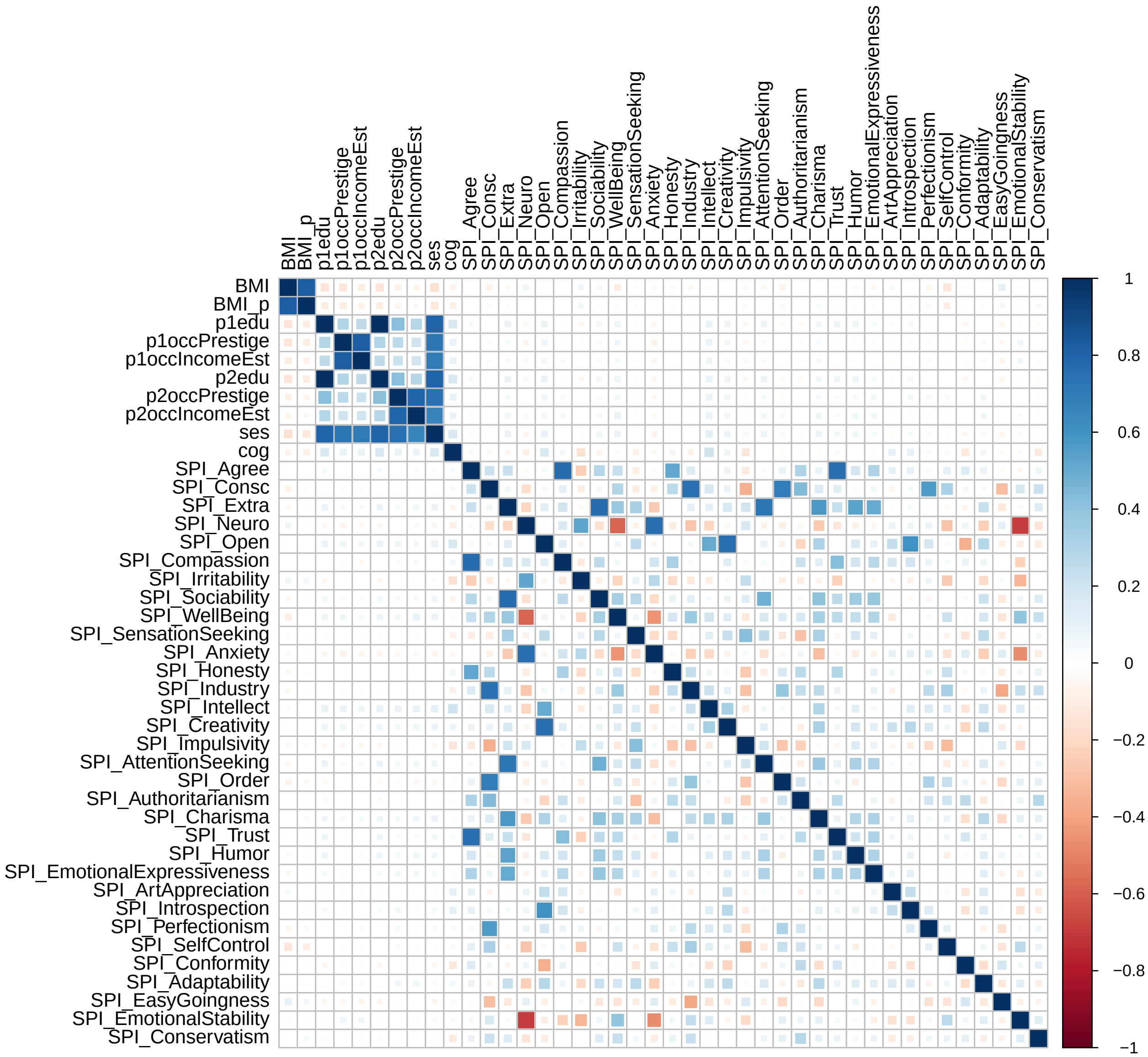

Figure 3: Zero-order correlations among study variables (Male Participants)

3 BMI percentile (Regression Models)

Regression models were built that regressed BMI percentile onto parental socio-economic status and adolescent individual differences. Two basic models were constructed: one that hypothesized parental SES:

$$BMIp_i = b_0 + b_1(SES_i) + b_2(ID_i) + e_i$$

and an individual difference were two independent predictors of BMI, and a second that hypothesized these variables interacted with each other:

$$BMIp_i = b_0 + b_1(SES_i) + b_2(ID_i) + b_3(SES_i \times ID_i) + e_i$$

We iterated through all individual differences – the broad Big Five personality traits, the narrow SPI-27 traits, and cognitive functioning – and tested each one independently in the model as an individual difference.

Models were estimated separately for men and women.

```
#end goal of wrangling is a data frame of data frames
# nested data frames correspond to a single personality trait
# score refers to a participant's score on that trait
# we also standardize each of our variables within gender
sapa_male_trait = sapa_male[train_male, ] %>%
  dplyr::select(-starts_with("p1"), -starts_with("p2")) %>%
  #identify which rows in test and training
  mutate(set = ifelse(row_number() %in% train_male[,1], "train", "test")) %>%
  # gather all personality variables
  gather("trait_name", "trait_score", -ses, -BMI_c, -BMI, -BMI_p, -set) %>%
  # group by trait and also by whether in test/train
  group_by(trait_name, set) %>%
  mutate(trait_score = scale(trait_score)) %>% #standardize
  mutate(ses = scale(ses)) %>% #standardize
  ungroup() %>% group_by(trait_name) %>%  #group only by trait
  nest() #nest data frames


sapa_female_trait = sapa_female[train_female, ] %>%
  dplyr::select(-starts_with("p1"), -starts_with("p2")) %>%
  #identify which rows in test and training
  mutate(set = ifelse(row_number() %in% train_male[,1], "train", "test")) %>%
  # gather all personality variables
  gather("trait_name", "trait_score", -ses, -BMI_c, -BMI, -BMI_p, -set) %>%
  # group by trait and also by whether in test/train
  group_by(trait_name, set) %>%
  mutate(trait_score = scale(trait_score)) %>% #standardize
  mutate(ses = scale(ses)) %>% #standardize
  ungroup() %>% group_by(trait_name) %>%  #group only by trait
  nest() #nest data frames
```

```
# number of bootstrap samples
boot.n = 10000

# ---- regression iteration (males) -----
```

```r
# apply to each dataset the linear model estimating coefficients
male_reg = sapa_male_trait %>%
  mutate(cov = map(data, ~lm(BMI_p ~ trait_score + ses, data = .))) %>%
  mutate(int = map(data, ~lm(BMI_p ~ trait_score*ses, data = .)))

# duplicate object for later
male_plot = male_reg

# select summary statistics and clean using `tidy`
male_reg = male_reg %>%
  dplyr::select(-data) %>%
  gather("model", "output", cov, int) %>%
  mutate(output = map(output, broom::tidy, conf.int = FALSE)) %>%
  unnest(cols = c(output))

set.seed(031720)

# bootstrap coefficient estimates
male_boot = sapa_male_trait %>%
  mutate(samples = map(data, bootstraps, times = boot.n)) %>%
  dplyr::select(-data) %>%
  unnest(samples) %>%
  mutate(boot_cov = map(splits,
                        ~broom::tidy(lm(BMI_p ~ trait_score + ses,
                                        analysis(.))))) %>%
  mutate(boot_int = map(splits,
                        ~broom::tidy(lm(BMI_p ~ trait_score*ses,
                                        analysis(.)))))
# extract summary statistics
male_boot = male_boot %>%
  dplyr::select(-splits, -id) %>%
  gather("model", "summary", -trait_name) %>%
  unnest(cols = c(summary))

# identify 95% CI from bootstraps
male_boot = male_boot %>%
  group_by(trait_name, model, term) %>%
  summarise(conf.low = quantile(estimate, probs = .025),
            conf.high = quantile(estimate, probs = .975)) %>%
  ungroup() %>%
  mutate(model = gsub("boot_", "", model))

# add to summary data frame
male_reg = male_reg %>%
  full_join(male_boot)

# save for later
save(male_reg, male_plot, file = "data/regression_output_male.Rdata")

# ---- regression iteration (females) -----


# we run the models for men and women separately because R kept
```

```r
# crashing when trying to run this whole script.

# apply to each dataset the linear model estimating coefficients
female_reg = sapa_female_trait %>%
  mutate(cov = map(data, ~lm(BMI_p ~ trait_score + ses, data = .))) %>%
  mutate(int = map(data, ~lm(BMI_p ~ trait_score*ses, data = .)))

# duplicate object for later
female_plot = female_reg

# select summary statistics and clean using `tidy`
female_reg = female_reg %>%
  dplyr::select(-data) %>%
  gather("model", "output", cov, int) %>%
  mutate(output = map(output, broom::tidy, conf.int = FALSE)) %>%
  unnest(cols = c(output))


# ----  bootstrap confidence intervals (females) -----

set.seed(031720)

# bootstrap coefficient estimates
female_boot = sapa_female_trait %>%
  mutate(samples = map(data, bootstraps, times = boot.n)) %>%
  dplyr::select(-data) %>%
  unnest(samples) %>%
  mutate(boot_cov = map(splits,
                        ~broom::tidy(lm(BMI_p ~ trait_score + ses,
                                        analysis(.))))) %>%
  mutate(boot_int = map(splits,
                        ~broom::tidy(lm(BMI_p ~ trait_score*ses,
                                        analysis(.)))))
# extract summary statistics
female_boot = female_boot %>%
  dplyr::select(-splits, -id) %>%
  gather("model", "summary", -trait_name) %>%
  unnest()

# identify 95% CI from bootstraps
female_boot = female_boot %>%
  group_by(trait_name, model, term) %>%
  summarise(conf.low = quantile(estimate, probs = .025),
            conf.high = quantile(estimate, probs = .975)) %>%
  ungroup() %>%
  mutate(model = gsub("boot_", "", model))

# add to summary data frame
female_reg = female_reg %>%
  full_join(female_boot)

# save for later
save(female_reg, female_plot, file = "data/regression_output_female.Rdata")
```

3.1 SES controlling for personality

In this section we use the output from the models to estimate and visulaize the relationship of SES to BMI.

To estimate the effect of socioeconomic status on BMI percentile, we graph the estimates of the SES slope coefficient across all regression models controlling for individual differences. This presents not only the average estimate across all models (solid line), but the range of estimates – a wide range suggests that the effect of SES on BMI is sensitive to the inclusion of different individual difference measures, while a narrow range suggests that the effect of SES on BMI is persistent through personality and cognition.

3.1.1 Females

```
avg_female = female_reg %>%
  filter(term == "ses") %>%
  filter(model == "cov") %>%
  arrange(estimate)
```

Higher parental socioeconomic status was associated with lower BMI percentile. The largest estimated effect size suggested every one standard deviation increase in parental SES resulted in a 3.57 percent lower BMI percentile score ($t = -9.09$, $p < .001$, CI = [-4.34, -2.81]). The smallest estimated effect size suggested every one standard deviation increase in parental SES resulted in a 3.31 percent lower BMI percentile score ($t = -8.41$, $p < .001$, CI = [-4.08, -2.54]). Figure 4 depicts all confidence intervals – all are statistically significant.

```
female_plot_1 = female_reg %>%
   ungroup() %>%
  filter(term == "ses") %>%
  filter(model == "cov") %>%
  mutate(psig = ifelse(p.value < .05, "yes", "no")) %>%
  arrange(estimate) %>%
  mutate(spec = row_number()) %>%
  ggplot(aes(x = spec, y = conf.low)) +
  geom_segment(aes(xend = spec, yend = conf.high, color = psig)) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  geom_hline(aes(yintercept = mean), data = avg_female) +
  scale_color_manual(values = c("red", "grey")) +
  scale_y_continuous(limits = c(-5.5, 0.25), breaks = c(-5:0))+
  labs(x = "Specification",
       y = "SES coefficient, controlling for personality",
       title = "Adolescent Females") +
  guides(color = "none") +
  theme_pubr()

female_plot_1
```

3.1.2 Males

```
avg_male = male_reg %>%
  filter(term == "ses") %>%
  filter(model == "cov") %>%
  arrange(estimate)
```

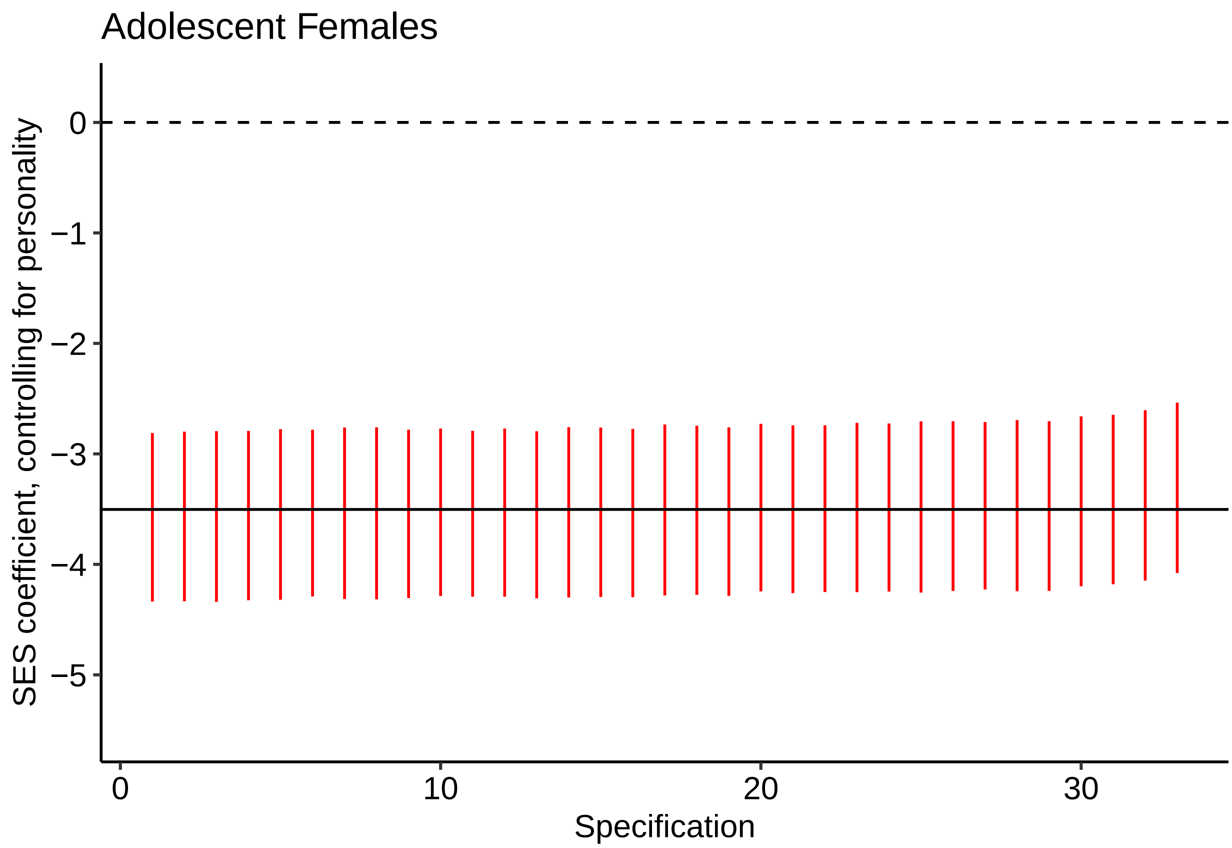

Figure 4: Estimates of SES coefficient from linear models predicing female adolescent BMI. Estimates control for individual differences. Bars represent 95% confidence intervals and are colored red if statistically significant.

Table 3: Effect of SES controlling for various traits (adolescent females)

Controlling for	SES estimate	SE	t	p-value	CI-low	CI-high
cog	-3.42	0.40	-8.64	$< .001$	-4.18	-2.65
SPI_Agree	-3.54	0.39	-9.00	$< .001$	-4.32	-2.76
SPI_Consc	-3.47	0.39	-8.83	$< .001$	-4.23	-2.71
SPI_Extra	-3.46	0.39	-8.76	$< .001$	-4.24	-2.70
SPI_Neuro	-3.43	0.39	-8.71	$< .001$	-4.20	-2.66
SPI_Open	-3.53	0.39	-8.96	$< .001$	-4.30	-2.76
SPI_Compassion	-3.55	0.39	-9.04	$< .001$	-4.32	-2.79
SPI_Irritability	-3.49	0.39	-8.88	$< .001$	-4.25	-2.72
SPI_Sociability	-3.49	0.39	-8.87	$< .001$	-4.25	-2.73
SPI_WellBeing	-3.31	0.39	-8.41	$< .001$	-4.08	-2.54
SPI_SensationSeeking	-3.54	0.39	-8.98	$< .001$	-4.29	-2.79
SPI_Anxiety	-3.47	0.39	-8.81	$< .001$	-4.24	-2.69
SPI_Honesty	-3.57	0.39	-9.09	$< .001$	-4.34	-2.81
SPI_Industry	-3.50	0.39	-8.89	$< .001$	-4.26	-2.74
SPI_Intellect	-3.52	0.39	-8.91	$< .001$	-4.28	-2.73
SPI_Creativity	-3.54	0.39	-9.01	$< .001$	-4.31	-2.76
SPI_Impulsivity	-3.48	0.39	-8.84	$< .001$	-4.26	-2.71
SPI_AttentionSeeking	-3.48	0.40	-8.79	$< .001$	-4.24	-2.70
SPI_Order	-3.53	0.39	-9.01	$< .001$	-4.31	-2.79
SPI_Authoritarianism	-3.54	0.39	-8.99	$< .001$	-4.31	-2.78
SPI_Charisma	-3.57	0.39	-9.04	$< .001$	-4.33	-2.80
SPI_Trust	-3.53	0.39	-8.94	$< .001$	-4.30	-2.76
SPI_Humor	-3.56	0.39	-9.05	$< .001$	-4.34	-2.79
SPI_EmotionalExpressiveness	-3.49	0.39	-8.85	$< .001$	-4.25	-2.74
SPI_ArtAppreciation	-3.53	0.39	-8.96	$< .001$	-4.29	-2.77
SPI_Introspection	-3.52	0.39	-8.97	$< .001$	-4.30	-2.77
SPI_Perfectionism	-3.52	0.39	-8.93	$< .001$	-4.28	-2.76
SPI_SelfControl	-3.50	0.39	-8.94	$< .001$	-4.24	-2.73
SPI_Conformity	-3.52	0.39	-8.93	$< .001$	-4.28	-2.75
SPI_Adaptability	-3.54	0.39	-8.98	$< .001$	-4.29	-2.77
SPI_EasyGoingness	-3.38	0.40	-8.55	$< .001$	-4.15	-2.60
SPI_EmotionalStability	-3.55	0.39	-9.02	$< .001$	-4.29	-2.78
SPI_Conservatism	-3.55	0.39	-9.01	$< .001$	-4.32	-2.78

Higher parental socioeconomic status was associated with lower BMI percentile. The largest estimated effect size suggested every one standard deviation increase in parental SES resulted in a 3.75 percent lower BMI percentile score ($t = -5.73$, $p < .001$, CI = [-4.99, -2.47]). The smallest estimated effect size suggested every one standard deviation increase in parental SES resulted in a 3.41 percent lower BMI percentile score ($t = -5.21$, $p < .001$, CI = [-4.70, -2.12]). Figure 5 depicts all confidence intervals – all are statistically significant.

```
male_plot_1 = male_reg %>%
  filter(term == "ses") %>%
  filter(model == "cov") %>%
  mutate(psig = ifelse(p.value < .05, "yes", "no")) %>%
  arrange(estimate) %>%
  mutate(spec = row_number()) %>%
  ggplot(aes(x = spec, y = conf.low)) +
  geom_segment(aes(xend = spec, yend = conf.high, color = psig)) +
  #geom_point(aes(y = estimate)) +
  geom_hline(aes(yintercept = 0), linetype = "dashed", color = "black") +
  geom_hline(aes(yintercept = mean), data = avg_male, color = "black") +
  #geom_label(aes(x = 25, y = 1.25, label = round(mean,2)), data = avg_male )+
  scale_color_manual(values = c("red", "grey")) +
  scale_y_continuous(limits = c(-5.5, .25), breaks = c(-5:0))+
  labs(x = "Specification", title = "Adolescent Males", y = NULL) +
  guides(color = "none") +
  theme_pubr()

male_plot_1
```

3.2 Individual differences controlling for SES

Next, we examine the coefficients for individual differences from the estimated linear models. These coefficients represent the partial association between each trait and BMI percentile, controlling for SES. To facilitate interpretation, we plot all results in Figure 6.

Two patterns stand out quickly: first, more traits are significantly associated with BMI among adolescent females compared to adolescent males. This may partially be driven by power (the sample of adolescent females is more than twice the sample of adolescent males), but we also note that effect sizes tend to be larger among females.

Second, traits associated with Neuroticism (Neuroticism itself, but also Irritability, Anxiety, and Well-being) tend to have the largest associations with BMI.

Of note, Self-control and Cognitive Functioning are largely, negatively associated with BMI across biological sex.

```
colors = RColorBrewer::brewer.pal(n = 3, "Dark2")
source(here("scripts/personality_scales.R"))
names(SPI_5_names) = str_remove(names(SPI_5_names), "135_27_5_")
names(SPI_27_names) = str_remove(names(SPI_27_names), "135_27_5_")

female_reg = female_reg %>%
  mutate(Gender = "Female")  %>%
  ungroup()

male_reg = male_reg %>%
```

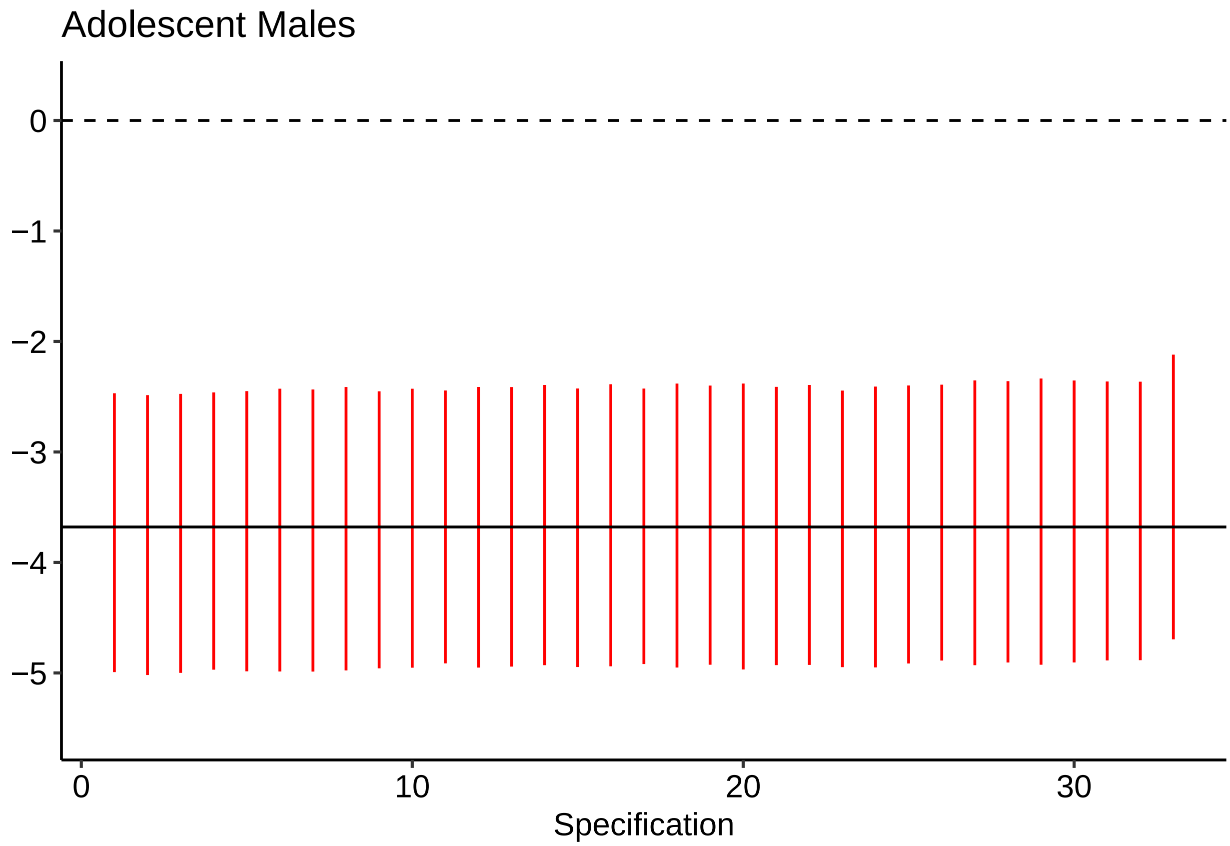

Figure 5: Estimates of SES coefficient from linear models predicing male adolescent BMI. Estimates control for individual differences. Bars represent 95% confidence intervals and are colored red if statistically significant.

Table 4: Effect of SES controlling for various traits (adolescent males)

Controlling for	SES estimate	SE	t	p-value	CI-low	CI-high
cog	-3.41	0.65	-5.21	< .001	-4.70	-2.12
SPI_Agree	-3.69	0.65	-5.69	< .001	-4.92	-2.43
SPI_Consc	-3.69	0.65	-5.70	< .001	-4.94	-2.41
SPI_Extra	-3.74	0.65	-5.75	< .001	-5.02	-2.49
SPI_Neuro	-3.70	0.65	-5.71	< .001	-4.95	-2.43
SPI_Open	-3.69	0.65	-5.67	< .001	-4.93	-2.39
SPI_Compassion	-3.69	0.65	-5.69	< .001	-4.95	-2.43
SPI_Irritability	-3.68	0.65	-5.67	< .001	-4.95	-2.41
SPI_Sociability	-3.72	0.65	-5.72	< .001	-4.98	-2.41
SPI_WellBeing	-3.63	0.65	-5.56	< .001	-4.89	-2.36
SPI_SensationSeeking	-3.68	0.65	-5.65	< .001	-4.93	-2.40
SPI_Anxiety	-3.72	0.65	-5.72	< .001	-4.99	-2.43
SPI_Honesty	-3.69	0.65	-5.70	< .001	-4.91	-2.44
SPI_Industry	-3.69	0.65	-5.70	< .001	-4.95	-2.41
SPI_Intellect	-3.72	0.65	-5.71	< .001	-4.97	-2.46
SPI_Creativity	-3.71	0.65	-5.71	< .001	-4.96	-2.45
SPI_Impulsivity	-3.68	0.65	-5.64	< .001	-4.97	-2.38
SPI_AttentionSeeking	-3.68	0.65	-5.67	< .001	-4.95	-2.38
SPI_Order	-3.72	0.65	-5.74	< .001	-5.00	-2.47
SPI_Authoritarianism	-3.65	0.65	-5.60	< .001	-4.91	-2.36
SPI_Charisma	-3.75	0.65	-5.73	< .001	-4.99	-2.47
SPI_Trust	-3.68	0.65	-5.67	< .001	-4.95	-2.44
SPI_Humor	-3.72	0.65	-5.69	< .001	-4.99	-2.43
SPI_EmotionalExpressiveness	-3.65	0.65	-5.59	< .001	-4.93	-2.35
SPI_ArtAppreciation	-3.69	0.65	-5.66	< .001	-4.94	-2.39
SPI_Introspection	-3.68	0.65	-5.67	< .001	-4.93	-2.39
SPI_Perfectionism	-3.67	0.65	-5.66	< .001	-4.89	-2.39
SPI_SelfControl	-3.64	0.65	-5.59	< .001	-4.90	-2.35
SPI_Conformity	-3.64	0.65	-5.58	< .001	-4.93	-2.34
SPI_Adaptability	-3.67	0.65	-5.63	< .001	-4.91	-2.40
SPI_EasyGoingness	-3.62	0.65	-5.54	< .001	-4.88	-2.36
SPI_EmotionalStability	-3.72	0.65	-5.74	< .001	-4.99	-2.45
SPI_Conservatism	-3.68	0.65	-5.64	< .001	-4.93	-2.41

```
  mutate(Gender = "Male")  %>%
  ungroup()

female_plot = female_reg %>%
  filter(term == "trait_score") %>%
  filter(model == "cov") %>%
  mutate(psig = ifelse(p.value < .05, "yes", "no"),
         est_r = papaja::printnum(estimate),
         yloc = ifelse(estimate > 0, conf.high + .5, conf.low - .5),
         trait_name = factor(trait_name,
                             levels = c("cog", names(SPI_27_names), names(SPI_5_names)),
                             labels = c("Cognitive Functioning", SPI_27_names, SPI_5_names))) %>%
  ggplot(aes(x = reorder(trait_name, estimate), y = estimate)) +
  geom_bar(stat = "identity", aes(fill = psig)) +
  geom_hline(aes(yintercept = 0), color = "grey") +
  geom_text(aes(label = est_r, y = yloc)) +
  scale_fill_manual(values = c("grey", "orange")) +
  scale_y_continuous(limits = c(-4.5, 4.5)) +
  geom_errorbar(aes(ymin = conf.low, ymax = conf.high), width = .3)+
  guides(fill = "none") +
  labs(y = "Regression Coefficient", x = NULL) +
  coord_flip() +
  theme_pubr(base_size = 10)

male_plot = male_reg %>%
  filter(term == "trait_score") %>%
  filter(model == "cov") %>%
  mutate(psig = ifelse(p.value < .05, "yes", "no"),
         est_r = papaja::printnum(estimate),
         yloc = ifelse(estimate > 0, conf.high + .5, conf.low - .5),
         trait_name = factor(trait_name,
                             levels = c("cog", names(SPI_27_names), names(SPI_5_names)),
                             labels = c("Cognitive Functioning", SPI_27_names, SPI_5_names))) %>%
  ggplot(aes(x = reorder(trait_name, estimate), y = estimate)) +
  geom_bar(stat = "identity", aes(fill = psig)) +
  geom_hline(aes(yintercept = 0), color = "grey") +
  geom_text(aes(label = est_r, y = yloc)) +
  scale_fill_manual(values = c("grey", "orange")) +
  scale_y_continuous(limits = c(-4.5, 4.5)) +
  geom_errorbar(aes(ymin = conf.low, ymax = conf.high), width = .3)+
  guides(fill = "none") +
  labs(y = "Regression Coefficient", x = NULL) +
  coord_flip() +
  theme_pubr(base_size = 10)

ggarrange(female_plot, male_plot, ncol = 2, labels = c("Female", "Male"))
```

3.2.1 Table

```
load(here("data/regression_output_male.Rdata"))
load(here("data/regression_output_female.Rdata"))
```

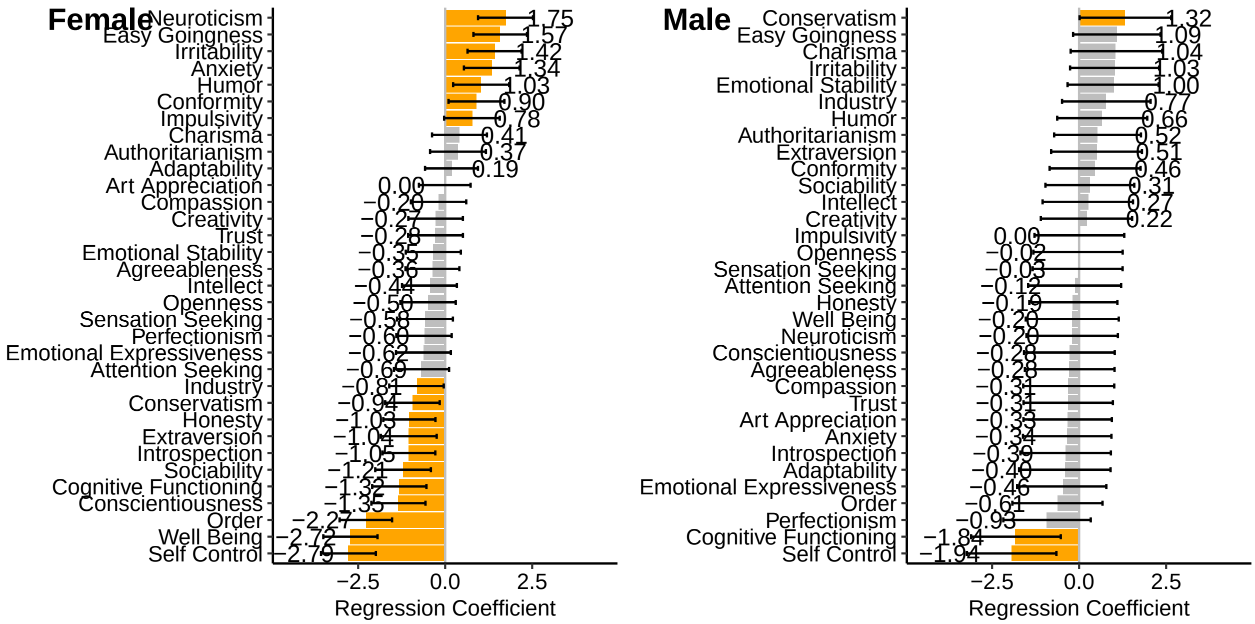

Figure 6: Horizontal bars represent the coefficient estimate, with 95% confidence intervals included. Bars are colored if the estimate is statistically significant ($p < .05$). Estimate values are printed next to each bar.

```
names(SPI_5_names) = str_remove(names(SPI_5_names), "135_27_5_")
names(SPI_27_names) = str_remove(names(SPI_27_names), "135_27_5_")

female_reg = female_reg %>%
  mutate(gender = "female") %>%
  ungroup()

male_reg = male_reg %>%
  mutate(gender = "male") %>%
  ungroup()

all_reg_tab = female_reg %>%
  full_join(male_reg) %>%
  filter(grepl("trait", term)) %>%
  mutate(b1_est = printnum(estimate),
         b1_est = ifelse(conf.low > 0 | conf.high < 0, paste0(b1_est, "*"), b1_est),
         conf.low = printnum(conf.low),
         conf.high = printnum(conf.high),
         b2_conf = paste0("[", conf.low, ", ", conf.high, "]"),
         pval = paste("p =", printp(p.value)),
         str_replace(pval, "= <", "<")) %>%
  dplyr::select(trait_name, model, term, b1_est, b2_conf, pval, gender) %>%
  gather("key", "value", b1_est, b2_conf, pval) %>%
  unite(col = "newkey", gender, model, term) %>%
  spread(newkey, value) %>%
  mutate(trait_name = factor(trait_name,
                             levels = c("cog", names(SPI_27_names), names(SPI_5_names)),
```

```
                              labels = c("Cognitive Ability", SPI_27_names, SPI_5_names))) %>%
  arrange(trait_name) %>%
  group_by(trait_name) %>%
  mutate(
    trait_name = as.character(trait_name),
    trait_name = ifelse(row_number() == 1,
                        trait_name,
                        NA_character_)) %>%
  ungroup() %>%
  dplyr::select(-key)
```

```
## Joining, by = c("trait_name", "model", "term", "estimate", "std.error", "statistic", "p.value", "con
```

```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```

```
all_reg_tab %>%
  kable(.,
        booktabs = T, escape = F,
        longtable = T,
        col.names = rep(c("Trait", rep(c("b", "b", "b x SES"), 2)))) %>%
  kable_styling() %>%
  add_header_above(c(" ", "Additive\nModel" = 1, "Interaction\nModel" = 2, "Additive\nModel" = 1, "Inte
  add_header_above(c(" ", "Female" = 3, "Male" = 3)) %>%
  group_rows("SPI: 27 Factors", 4, 78) %>%
  group_rows("SPI: 5 Factors", 79, 99)
```

	Female			Male		
	Additive Model	Interaction Model		Additive Model	Interaction Model	
Trait	b	b	b x SES	b	b	b x SES
Cognitive Ability	-1.32* [-2.10, -0.54] p = .001	-1.33* [-2.11, -0.54] p = .001	0.09 [-0.68, 0.86] p = .812	-1.84* [-3.10, -0.53] p = .005	-1.83* [-3.10, -0.52] p = .005	-0.08 [-1.37, 1.20] p = .904
SPI: 27 Factors						
Compassion	-0.20 [-0.98, 0.61] p = .619	-0.19 [-0.98, 0.61] p = .628	-0.38 [-1.14, 0.37] p = .316	-0.31 [-1.61, 1.01] p = .629	-0.34 [-1.64, 0.98] p = .597	0.44 [-0.81, 1.65] p = .511
Irritability	1.42* [0.64, 2.20] p = < .001	1.43* [0.65, 2.20] p = < .001	0.24 [-0.49, 0.97] p = .527	1.03 [-0.26, 2.30] p = .113	1.03 [-0.26, 2.31] p = .112	0.29 [-0.98, 1.60] p = .656
Sociability	-1.21* [-2.01, -0.41] p = .002	-1.21* [-2.01, -0.41] p = .002	0.33 [-0.47, 1.11] p = .401	0.31 [-0.97, 1.58] p = .637	0.39 [-0.88, 1.69] p = .544	1.22 [-0.04, 2.46] p = .066
Well Being	-2.72* [-3.50, -1.94] p = < .001	-2.70* [-3.48, -1.93] p = < .001	0.81* [0.04, 1.58] p = .039	-0.20 [-1.53, 1.14] p = .764	-0.18 [-1.52, 1.15] p = .785	0.57 [-0.69, 1.78] p = .365
Sensation Seeking	-0.58 [-1.38, 0.22] p = .142	-0.56 [-1.37, 0.24] p = .154	0.63 [-0.13, 1.41] p = .106	-0.03 [-1.33, 1.25] p = .963	-0.03 [-1.35, 1.25] p = .962	-0.40 [-1.68, 0.90] p = .553

Anxiety	1.34*	1.38*	-0.50	-0.34	-0.35	0.65
	[0.54, 2.14]	[0.59, 2.18]	[-1.29, 0.29]	[-1.61, 0.93]	[-1.62, 0.92]	[-0.60, 1.92]
	p = .001	p = < .001	p = .214	p = .597	p = .584	p = .326
Honesty	-1.03*	-1.04*	0.49	-0.19	-0.24	0.81
	[-1.78, -0.28]	[-1.79, -0.29]	[-0.24, 1.25]	[-1.43, 1.10]	[-1.47, 1.07]	[-0.40, 2.00]
	p = .009	p = .009	p = .183	p = .766	p = .717	p = .219
Industry	-0.81*	-0.81*	-0.21	0.77	0.75	0.35
	[-1.61, -0.04]	[-1.61, -0.04]	[-0.97, 0.54]	[-0.49, 2.05]	[-0.50, 2.03]	[-0.96, 1.61]
	p = .039	p = .039	p = .597	p = .236	p = .244	p = .598
Intellect	-0.44	-0.45	-0.22	0.27	0.22	-0.55
	[-1.24, 0.34]	[-1.26, 0.33]	[-0.95, 0.51]	[-1.05, 1.55]	[-1.10, 1.49]	[-1.87, 0.80]
	p = .270	p = .258	p = .573	p = .684	p = .742	p = .394
Creativity	-0.27	-0.27	0.02	0.22	0.22	0.11
	[-1.06, 0.51]	[-1.06, 0.51]	[-0.76, 0.77]	[-1.10, 1.52]	[-1.10, 1.52]	[-1.28, 1.53]
	p = .494	p = .495	p = .961	p = .739	p = .733	p = .868
Impulsivity	0.78	0.77	0.39	0.00	0.01	-0.65
	[-0.03, 1.57]	[-0.04, 1.56]	[-0.42, 1.20]	[-1.28, 1.30]	[-1.26, 1.32]	[-1.98, 0.65]
	p = .048	p = .050	p = .329	p = .995	p = .989	p = .337
Attention Seeking	-0.69	-0.65	0.50	-0.12	0.01	1.26
	[-1.47, 0.11]	[-1.44, 0.15]	[-0.25, 1.25]	[-1.46, 1.21]	[-1.32, 1.35]	[-0.04, 2.55]
	p = .081	p = .099	p = .207	p = .857	p = .993	p = .050
Order	-2.27*	-2.26*	-0.80*	-0.61	-0.60	-0.50
	[-3.03, -1.52]	[-3.02, -1.51]	[-1.54, -0.06]	[-1.92, 0.67]	[-1.90, 0.69]	[-1.81, 0.78]
	p = < .001	p = < .001	p = .040	p = .345	p = .357	p = .442
Authoritarianism	0.37	0.37	0.17	0.52	0.44	1.51*
	[-0.43, 1.17]	[-0.44, 1.17]	[-0.61, 0.96]	[-0.72, 1.78]	[-0.81, 1.68]	[0.25, 2.76]
	p = .348	p = .349	p = .650	p = .421	p = .503	p = .025
Charisma	0.41	0.41	0.19	1.04	1.04	0.49
	[-0.38, 1.20]	[-0.38, 1.20]	[-0.56, 0.94]	[-0.24, 2.39]	[-0.24, 2.38]	[-0.81, 1.75]
	p = .300	p = .299	p = .631	p = .111	p = .112	p = .464
Trust	-0.28	-0.28	0.02	-0.31	-0.40	0.96
	[-1.06, 0.51]	[-1.06, 0.50]	[-0.77, 0.80]	[-1.60, 0.97]	[-1.68, 0.90]	[-0.29, 2.21]
	p = .471	p = .471	p = .958	p = .627	p = .541	p = .141
Humor	1.03*	1.03*	-0.30	0.66	0.66	0.66
	[0.23, 1.84]	[0.22, 1.84]	[-1.04, 0.44]	[-0.63, 1.96]	[-0.63, 1.96]	[-0.70, 2.02]
	p = .009	p = .010	p = .423	p = .313	p = .313	p = .331
Emotional Expressiveness	-0.62	-0.63	0.33	-0.46	-0.53	1.36*
	[-1.41, 0.16]	[-1.42, 0.16]	[-0.46, 1.09]	[-1.78, 0.78]	[-1.84, 0.73]	[0.06, 2.66]
	p = .114	p = .110	p = .395	p = .477	p = .420	p = .043
Art Appreciation	0.00	0.00	-0.19	-0.33	-0.33	-0.05
	[-0.75, 0.73]	[-0.75, 0.74]	[-0.95, 0.55]	[-1.60, 0.94]	[-1.60, 0.94]	[-1.36, 1.19]
	p = .996	p = > .999	p = .636	p = .614	p = .613	p = .937
Introspection	-1.05*	-1.05*	0.37	-0.39	-0.37	0.47
	[-1.80, -0.28]	[-1.81, -0.29]	[-0.37, 1.08]	[-1.69, 0.91]	[-1.66, 0.92]	[-0.74, 1.69]
	p = .008	p = .007	p = .343	p = .550	p = .570	p = .467
Perfectionism	-0.60	-0.61	-0.58	-0.93	-0.93	0.60
	[-1.40, 0.19]	[-1.41, 0.17]	[-1.33, 0.20]	[-2.18, 0.33]	[-2.18, 0.33]	[-0.66, 1.83]
	p = .130	p = .124	p = .137	p = .154	p = .153	p = .353
Self Control	-2.79*	-2.79*	-0.07	-1.94*	-1.98*	1.00
	[-3.57, -1.99]	[-3.57, -1.99]	[-0.81, 0.67]	[-3.22, -0.65]	[-3.26, -0.70]	[-0.31, 2.34]
	p = < .001	p = < .001	p = .851	p = .003	p = .002	p = .134
Conformity	0.90*	0.89*	-0.24	0.46	0.45	-0.19
	[0.10, 1.70]	[0.09, 1.70]	[-1.01, 0.55]	[-0.85, 1.77]	[-0.86, 1.76]	[-1.48, 1.03]
	p = .023	p = .024	p = .532	p = .482	p = .487	p = .762

Adaptability	0.19	0.19	0.23	-0.40	-0.44	0.96
	[-0.58, 0.94]	[-0.58, 0.94]	[-0.50, 0.97]	[-1.72, 0.90]	[-1.76, 0.87]	[-0.36, 2.29]
	p = .634	p = .632	p = .565	p = .536	p = .505	p = .150
Easy Goingness	1.57*	1.59*	-0.33	1.09	1.19	-1.41*
	[0.81, 2.35]	[0.82, 2.37]	[-1.11, 0.41]	[-0.17, 2.34]	[-0.08, 2.45]	[-2.67, -0.18]
	p = < .001	p = < .001	p = .400	p = .096	p = .069	p = .033
SPI: 5 Factors						
Emotional Stability	-0.35	-0.35	0.23	1.00	1.00	-0.49
	[-1.13, 0.45]	[-1.14, 0.45]	[-0.55, 1.01]	[-0.33, 2.29]	[-0.33, 2.29]	[-1.73, 0.73]
	p = .377	p = .378	p = .554	p = .123	p = .124	p = .447
Conservatism	-0.94*	-0.97*	0.86*	1.32*	1.25	1.44*
	[-1.72, -0.16]	[-1.77, -0.19]	[0.05, 1.65]	[0.01, 2.65]	[-0.05, 2.58]	[0.10, 2.83]
	p = .017	p = .014	p = .030	p = .043	p = .056	p = .027
Agreeableness	-0.36	-0.36	-0.28	-0.28	-0.37	0.76
	[-1.14, 0.41]	[-1.13, 0.41]	[-1.06, 0.52]	[-1.56, 1.02]	[-1.65, 0.94]	[-0.50, 2.01]
	p = .356	p = .366	p = .455	p = .664	p = .567	p = .254
Conscientiousness	-1.35*	-1.33*	-0.76	-0.28	-0.28	0.49
	[-2.12, -0.57]	[-2.10, -0.55]	[-1.54, 0.05]	[-1.58, 1.02]	[-1.58, 1.02]	[-0.73, 1.67]
	p = .001	p = .001	p = .055	p = .667	p = .662	p = .426
Extraversion	-1.04*	-1.06*	0.56	0.51	0.56	1.45*
	[-1.85, -0.24]	[-1.87, -0.27]	[-0.20, 1.31]	[-0.80, 1.80]	[-0.76, 1.86]	[0.13, 2.72]
	p = .008	p = .007	p = .160	p = .432	p = .391	p = .025
Neuroticism	1.75*	1.77*	-0.48	-0.20	-0.20	0.17
	[0.94, 2.52]	[0.97, 2.55]	[-1.24, 0.29]	[-1.51, 1.11]	[-1.51, 1.12]	[-1.04, 1.43]
	p = < .001	p = < .001	p = .224	p = .753	p = .757	p = .783
Openness	-0.50	-0.50	0.04	-0.02	-0.04	-0.16
	[-1.29, 0.30]	[-1.28, 0.30]	[-0.76, 0.83]	[-1.31, 1.25]	[-1.33, 1.24]	[-1.40, 1.09]
	p = .206	p = .207	p = .921	p = .971	p = .951	p = .807

3.3 Interaction of SES with personality

To estimate the joint effect of socioeconomic status and individual differences on BMI percentile, we graph the estimates of the interaction terms of SES by individual differences by BMI percentile. Like before, we present the average effect (solid black line) and the 95% confidence intervals for each model. Figure 7 shows the results for adolescent females, while Figure 9 shows the results for adolescent males.

3.3.1 Females

```
avg_female = ungroup(female_reg) %>%
  filter(grepl(":", term)) %>%
  summarise(mean = mean(estimate))

female_plot_2 = female_reg %>%
  filter(grepl(":", term)) %>%
  mutate(psig = ifelse(p.value < .05, "yes", "no")) %>%
  arrange(estimate) %>%
  mutate(spec = row_number()) %>%
  ggplot(aes(x = spec, y = conf.low)) +
  geom_segment(aes(xend = spec, yend = conf.high, color = psig)) +
  geom_hline(aes(yintercept = 0), linetype = "dashed", color = "black") +
  geom_hline(aes(yintercept = mean), data = avg_female, color = "black") +
```

```
  #geom_label(aes(x = 25, y = 1.25, label = round(mean,2)), data = avg_female )+
  scale_color_manual(values = c("grey", "red")) +
  #scale_y_continuous(limits = c(0.30, 2.20))+
  labs(x = "Specification",
       y = "SES x perosnality term in model", title = "Adolescent Females") +
  guides(color = "none") +
  theme_pubr()

female_plot_2
```

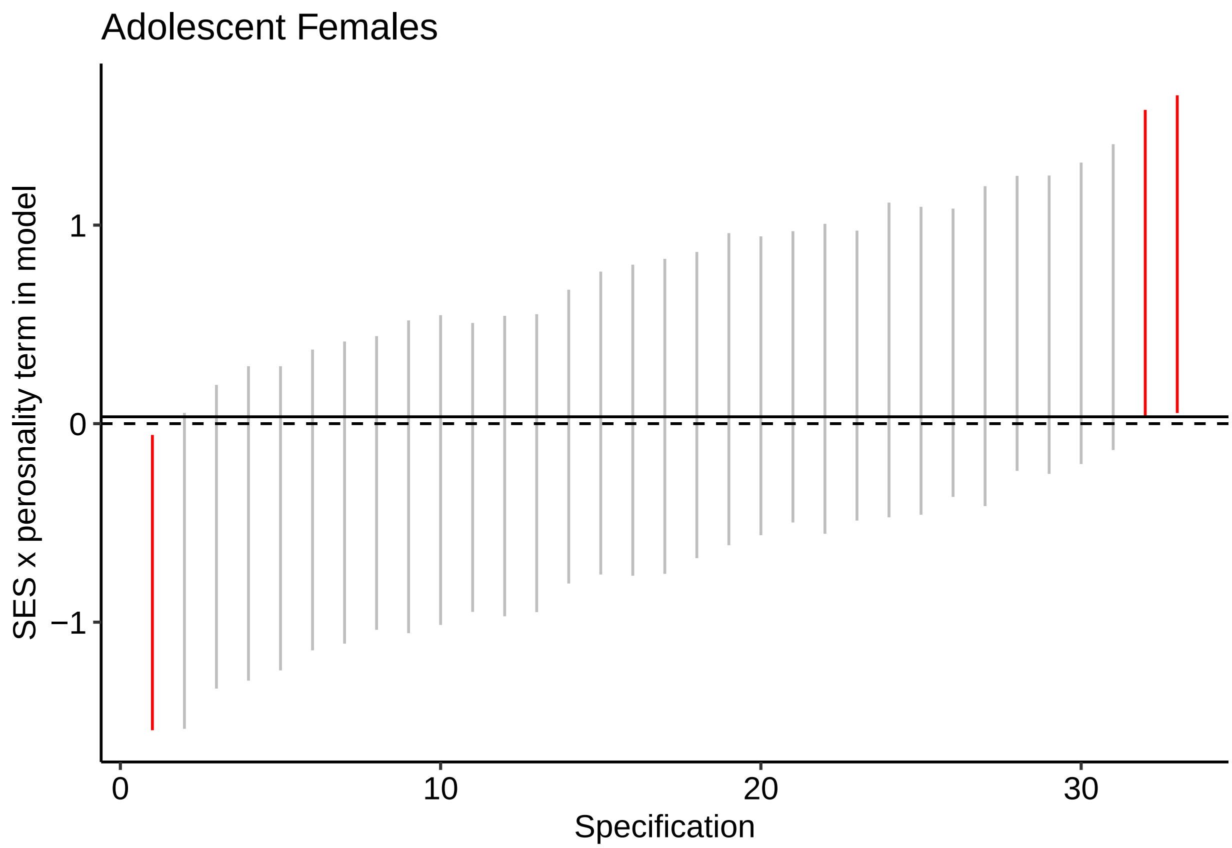

Figure 7: Coefficient estimates for interaction of SES and personality traits, predicting BMI percentile (adolescent females)

```
female_reg %>%
  filter(grepl(":", term)) %>%
  filter(p.value < .05) %>%
  select(-model, -term, -gender) %>%
  mutate(p.value = printp(p.value),
         across(where(is.numeric), printnum)) %>%
  kable(booktabs = T,
        caption = "Tests of the interaction of SES with traits when predicting BMI percentile (adolesce
  kable_styling()
```

We present the significant interactions for adolescent females in Table 6. All p-values were larger than .01. As SES increased, so did the relationship between Well-Being and Conservatism and BMI percentile; as SES

Table 6: Tests of the interaction of SES with traits when predicting BMI percentile (adolescent females).

trait_name	estimate	std.error	statistic	p.value	conf.low	conf.high
SPI_WellBeing	0.81	0.39	2.06	.039	0.04	1.58
SPI_Order	-0.80	0.39	-2.06	.040	-1.54	-0.06
SPI_Conservatism	0.86	0.40	2.16	.030	0.05	1.65

increased, the relationship between Order and BMI percentile decreased. These relationships are depicted in Figure 8.

```
plot_wellbeing = plot_model(lm(BMI_p ~ ses*SPI_WellBeing, data = sapa_female),
                            type = "pred", terms =c("SPI_WellBeing", "ses[meansd]"))
plot_conservat = plot_model(lm(BMI_p ~ ses*SPI_Conservatism, data = sapa_female),
                            type = "pred", terms =c("SPI_Conservatism", "ses[meansd]"))
plot_order = plot_model(lm(BMI_p ~ ses*SPI_Order, data = sapa_female),
                            type = "pred", terms =c("SPI_Order", "ses[meansd]"))

ggarrange(plot_wellbeing, plot_conservat, plot_order)
```

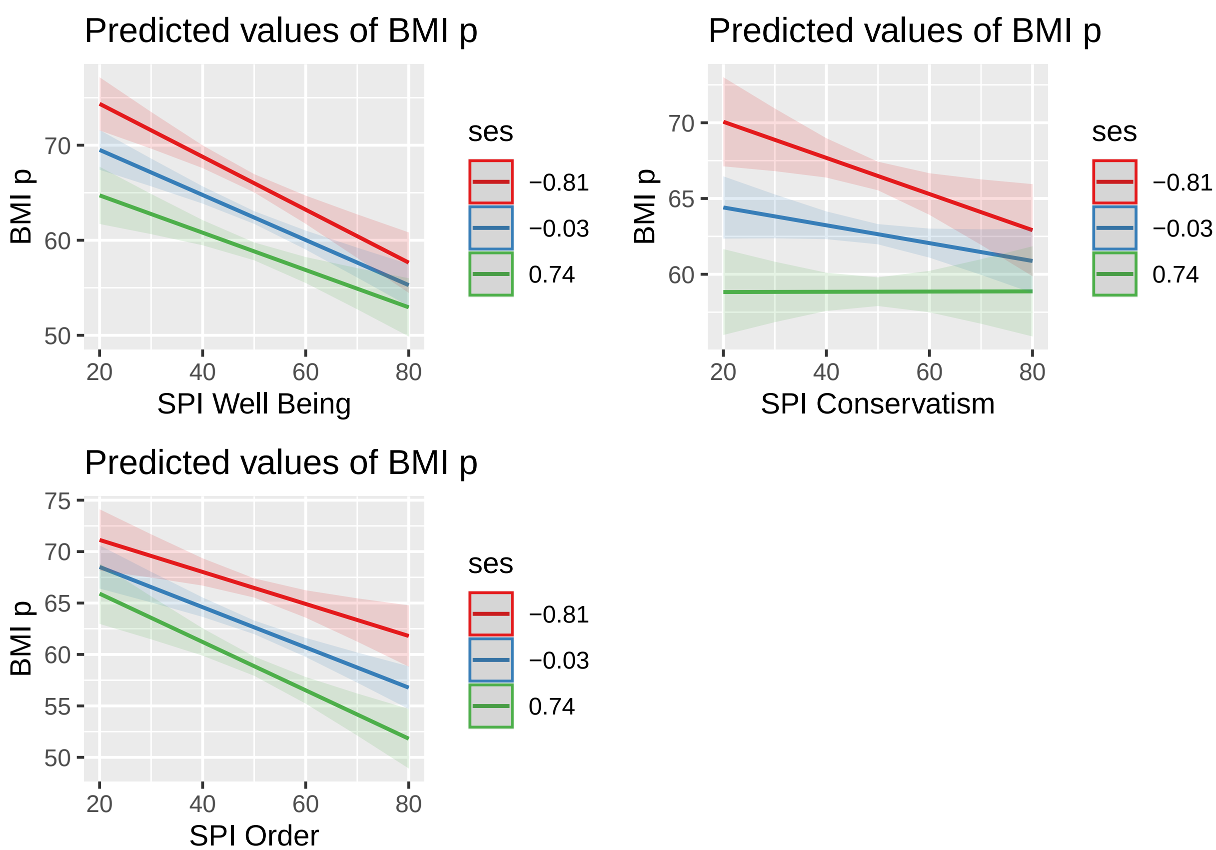

Figure 8: Significant interactions between SES and personality in adolescent females sample

Table 7: Tests of the interaction of SES with traits when predicting BMI percentile (males).

trait_name	estimate	std.error	statistic	p.value	conf.low	conf.high
SPI_Extra	1.45	0.65	2.24	.025	0.13	2.72
SPI_AttentionSeeking	1.26	0.64	1.96	.050	-0.04	2.55
SPI_Authoritarianism	1.51	0.67	2.25	.025	0.25	2.76
SPI_EmotionalExpressiveness	1.36	0.67	2.03	.043	0.06	2.66
SPI_EasyGoingness	-1.41	0.66	-2.13	.033	-2.67	-0.18
SPI_Conservatism	1.44	0.65	2.21	.027	0.10	2.83

3.3.2 Males

```
male_reg %>%
  filter(grepl(":", term)) %>%
  filter(p.value < .05) %>%
  select(-model, -term, -gender) %>%
  mutate(p.value = printp(p.value),
         across(where(is.numeric), printnum)) %>%
  kable(booktabs = T,
        caption = "Tests of the interaction of SES with traits when predicting BMI percentile (males).")
  kable_styling()
```

We present the significant interactions for adolescent males in Table 7. Again, all p-values were larger than .01. As SES increased, so did the relationship between BMI percentile and the traits of Extraversion, Attention Seeking, Authoritarianism, Emotional Expressiveness, and Conservatism; as SES increased, the relationship between Easy-Goingness and BMI percentile decreased. These relationships are depicted in Figure 10.

```
avg_male = ungroup(male_reg) %>%
  filter(grepl(":", term)) %>%
  summarise(mean = mean(estimate))

male_plot_2 = male_reg %>%
  filter(grepl(":", term)) %>%
  mutate(psig = ifelse(p.value < .05, "yes", "no")) %>%
  arrange(estimate) %>%
  mutate(spec = row_number()) %>%
  ggplot(aes(x = spec, y = conf.low)) +
  geom_segment(aes(xend = spec, yend = conf.high, color = psig)) +
  geom_hline(aes(yintercept = 0), linetype = "dashed", color = "black") +
  geom_hline(aes(yintercept = mean), data = avg_male, color = "black") +
  #geom_label(aes(x = 25, y = 1.25, label = round(mean,2)), data = avg_male )+
  scale_color_manual(values = c("grey", "red")) +
  #scale_y_continuous(limits = c(0.30, 2.20))+
  labs(x = "Specification",
       y = "SES x perosnality term in model", title = "Adolescent Males") +
  guides(color = "none") +
  theme_pubr()

male_plot_2
```

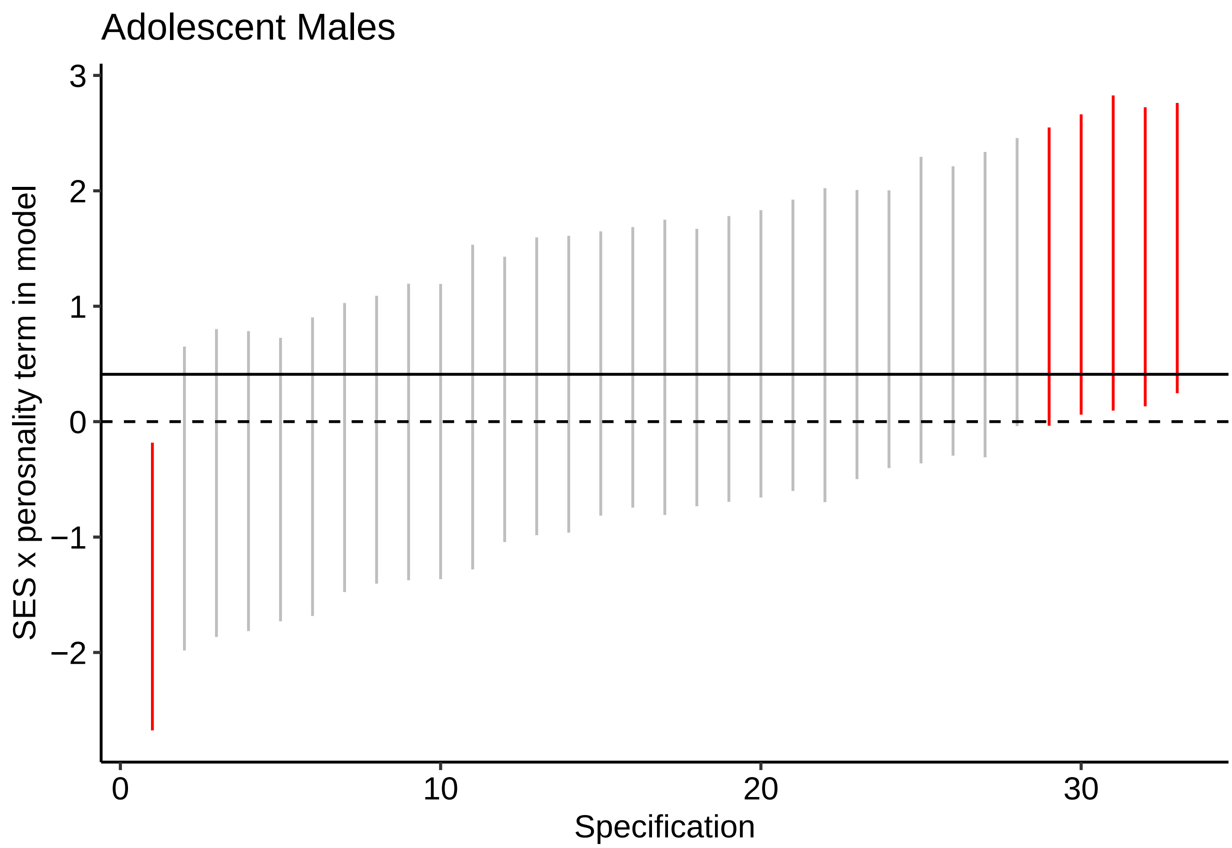

Figure 9: Coefficient estimates for interaction of SES and personality traits, predicting BMI percentile (adolescent males)

```
plot_extra = plot_model(lm(BMI_p ~ ses*SPI_Extra, data = sapa_male),
                            type = "pred",
                        terms =c("SPI_Extra", "ses[meansd]")) +
  theme_pubr(base_size = 8)
plot_attention = plot_model(lm(BMI_p ~ ses*SPI_AttentionSeeking, data = sapa_male),
                            type = "pred",
                            terms =c("SPI_AttentionSeeking", "ses[meansd]")) +
  theme_pubr(base_size = 8)
plot_authoritarian = plot_model(lm(BMI_p ~ ses*SPI_Authoritarianism, data = sapa_male),
                            type = "pred",
                            terms =c("SPI_Authoritarianism", "ses[meansd]")) +
  theme_pubr(base_size = 8)
plot_emoexp = plot_model(lm(BMI_p ~ ses*SPI_EmotionalExpressiveness, data = sapa_male),
                            type = "pred",
                         terms =c("SPI_EmotionalExpressiveness", "ses[meansd]")) +
  theme_pubr(base_size = 8)
plot_conservat = plot_model(lm(BMI_p ~ ses*SPI_Conservatism, data = sapa_male),
                            type = "pred",
                            terms =c("SPI_Conservatism", "ses[meansd]")) +
  theme_pubr(base_size = 8)
plot_easy = plot_model(lm(BMI_p ~ ses*SPI_EasyGoingness, data = sapa_male),
                            type = "pred",
                       terms =c("SPI_EasyGoingness", "ses[meansd]")) +
  theme_pubr(base_size = 8)


ggarrange(plot_extra, plot_attention, plot_authoritarian, plot_emoexp, plot_conservat, plot_easy)
```

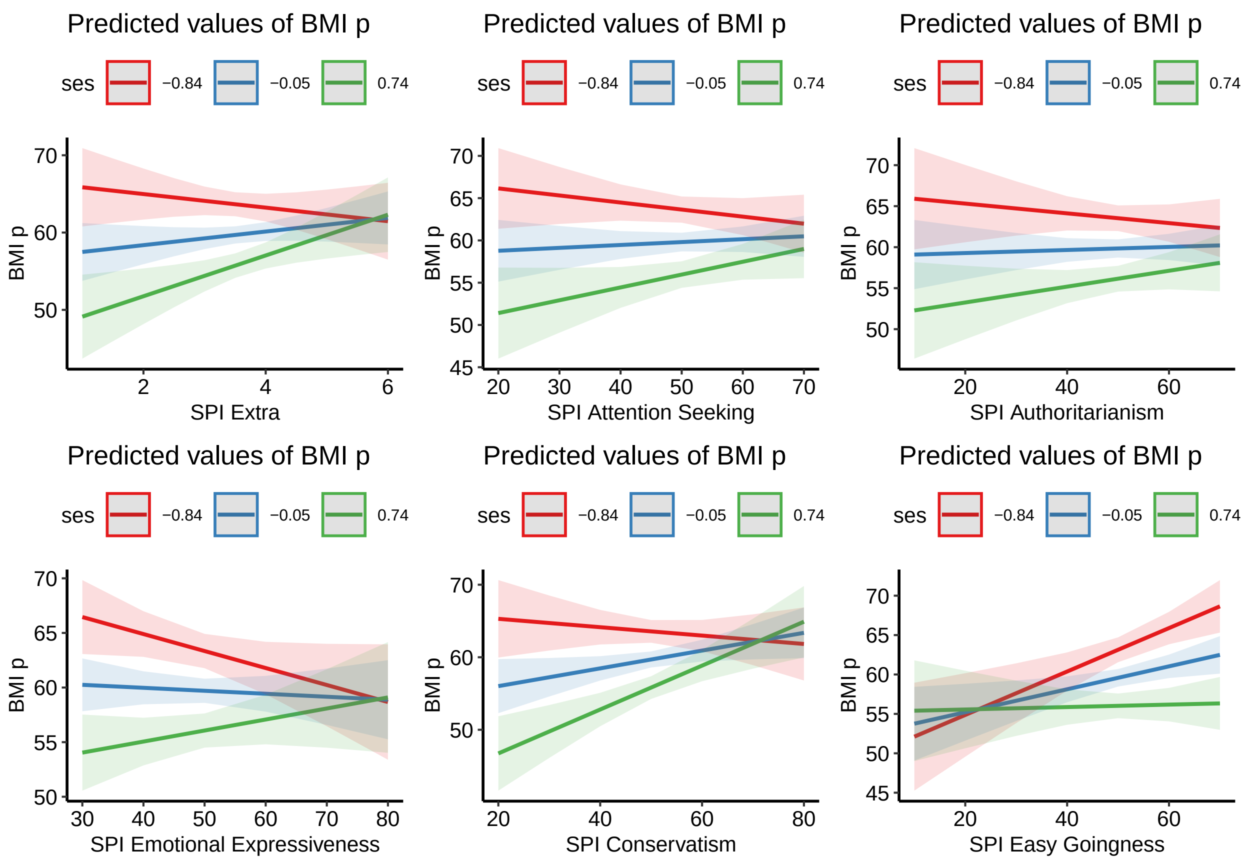

Figure 10: Significant interactions between SES and personality in adolescent males sample

4 BMI weight category (Logistic Regression)

Multinomial logistic regression models were built that regressed BMI category onto parental socioeconomic status and adolescent individual differences. Two basic models were constructed: one that hypothesized parental SES:

$$BMI_i = b_0 + b_1(SES_i) + b_2(ID_i) + e_i$$

and an individual difference were two independent predictors of BMI, and a second that hypothesized these variables interacted with each other:

$$BMI_i = b_0 + b_1(SES_i) + b_2(ID_i) + b_3(SES_i \times ID_i) + e_i$$

We iterated through all individual differences – the broad Big Five personality traits, the narrow SPI-27 traits, and cognitive functioning – and tested each one independently in the model as an individual difference.

Models were estimated separately for men and women.

```
set.seed(090919)

ctrl <- trainControl(method = "repeatedcv", # cross-validation
                     number = 10,  # 10 fold cross validation
                     repeats = 10, #repeated 10 times
                     verboseIter = FALSE,
                     search = "random",
                     sampling = "smote")
sapa_male_trait = sapa_male %>%
  dplyr::select(-starts_with("p1"), -starts_with("p2"), -starts_with("edu")) %>%
  mutate(BMI_c = factor(BMI_c, levels = c("Normal Weight", "Underweight", "Overweight", "Obese"))) %>%
  mutate(set = ifelse(row_number() %in% train_male[,1], "train", "test")) %>%
  gather("trait_name", "trait_score", -ses, -BMI_c, -BMI, -BMI_p, -set) %>%
  group_by(trait_name, set) %>%
  mutate(trait_score = scale(trait_score)) %>%
  ungroup() %>%
  group_by(trait_name) %>%
  nest()

sapa_female_trait = sapa_female %>%
  dplyr::select(-starts_with("p1"), -starts_with("p2"), -starts_with("edu")) %>%
  mutate(BMI_c = factor(BMI_c, levels = c("Normal Weight", "Underweight", "Overweight", "Obese"))) %>%
  mutate(set = ifelse(row_number() %in% train_male[,1], "train", "test")) %>%
  gather("trait_name", "trait_score", -ses, -BMI_c, -BMI, -BMI_p, -set) %>%
  group_by(trait_name, set) %>%
  mutate(trait_score = scale(trait_score)) %>%
  ungroup() %>%
  group_by(trait_name) %>%
  nest()

# ---- ordered logistic regression iteration (males)         ----

male_ses_only = train(BMI_c ~ ses, data = sapa_male,
                 subset = train_male,
                 method = "multinom",
```

```
                 maxit= 1000,
                 na.action = "na.exclude",
                 trControl = ctrl)

accuracy = predict(male_ses_only, type="raw", newdata=sapa_male[-train_male, ])
postResample(sapa_male[-train_male, "BMI_c"], accuracy)

male_log = sapa_male_trait %>%
  # train models on training subset; use mulintomial logistic regression; use specific formula
  mutate(
    cov = map(data, ~train(BMI_c ~ trait_score + ses, data = .,
                           subset = train_male,
                           method = "multinom",
                           na.action = "na.exclude",
                           trControl = ctrl)),
    int = map(data, ~train(BMI_c ~ trait_score*ses, data = .,
                           subset = train_male,
                           method = "multinom",
                           na.action = "na.exclude",
                           trControl = ctrl))) %>%
  gather("model", "output", cov, int) %>%
  # create test data from all rows not used in training
  mutate(test_data = map(data, .f = function(x) x[-train_male, ]),
         #extract reference (true) BMI categories from test data
         test_reference = map(test_data, "BMI_c"),
         # predict categories from model output; na.pass puts NAs in any row with missing data
         predicted = map2(output, test_data, predict, na.action = "na.pass"),
         # calculate accuracy, sensitivity, specificity, etc
         confusion = map2(predicted, test_reference, confusionMatrix),
         # extract final model coefficients
         final_mod = map(output, "finalModel"),
         # tidy output for printing
         coef = map(final_mod, broom::tidy, conf.int = TRUE))

# ---- ordered logistic regression iteration (females)           ----

female_ses_only = train(BMI_c ~ ses, data = sapa_female,
                      subset = train_female,
                      method = "multinom",
                      na.action = "na.exclude",
                      trControl = ctrl)


female_log = sapa_female_trait %>%
  # train models on training subset; use mulintomial logistic regression; use specific formula
  mutate(
    cov = map(data, ~train(BMI_c ~ trait_score + ses, data = .,
                         subset = train_female,
                         method = "multinom",
                         na.action = "na.exclude",
                         trControl = ctrl)),
    int = map(data, ~train(BMI_c ~ trait_score*ses, data = .,
                         subset = train_female,
```

```
                              method = "multinom",
                              na.action = "na.exclude",
                              trControl = ctrl))) %>%
  gather("model", "output", cov, int) %>%
  # create test data from all rows not used in training
  mutate(test_data = map(data, .f = function(x) x[-train_female, ]),
         #extract reference (true) BMI categories from test data
         test_reference = map(test_data, "BMI_c"),
         # predict categories from model output; na.pass puts NAs in any row with missing data
         predicted = map2(output, test_data, predict, na.action = "na.pass"),
         # calculate accuracy, sensitivity, specificity, etc
         confusion = map2(predicted, test_reference, confusionMatrix),
         # extract final model coefficients
         final_mod = map(output, "finalModel"),
         # tidy output for printing
         coef = map(final_mod, broom::tidy, conf.int = TRUE))


# ---- save output


save(train_male, male_ses_only, male_log, train_female, female_ses_only, female_log, file = "data/logis
```

4.1 Controlling for personality

To estimate the effect of socioeconomic status on BMI category, we graph the estimates of the SES slope coefficient across all logistic regression models controlling for individual differences. This presents not only the average estimate across all models (solid line), but the range of estimates – a wide range suggests that the effect of SES on BMI is sensitive to the inclusion of different individual difference measures, while a narrow range suggests that the effect of SES on BMI is persistent through personality and cognition.

4.1.1 Females

Parental socioeconomic status positively predicted greater likelihood of all non-normal categories (Underweight, Overweight, and Obese) compared to Normal among females. Adolescent females living in higher SES households were, on average, 103% less likely to be Underweight, 133% less likely, and 160% less likely to be Obese compared to low SES counterparts. Figure @ref(fig:notebook 34) depicts the coefficients of SES on BMI category, controlling for personality.

4.1.2 Males

Parental socioeconomic status positively predicted greater likelihood of all non-normal categories (Underweight, Overweight, and Obese) compared to Normal among males. Adolescent males living in higher SES households were, on average, 100% less likely to be Underweight, 126% less likely, and 144% less likely to be Obese compared to low SES counterparts. These results are depicted in Figure 12.

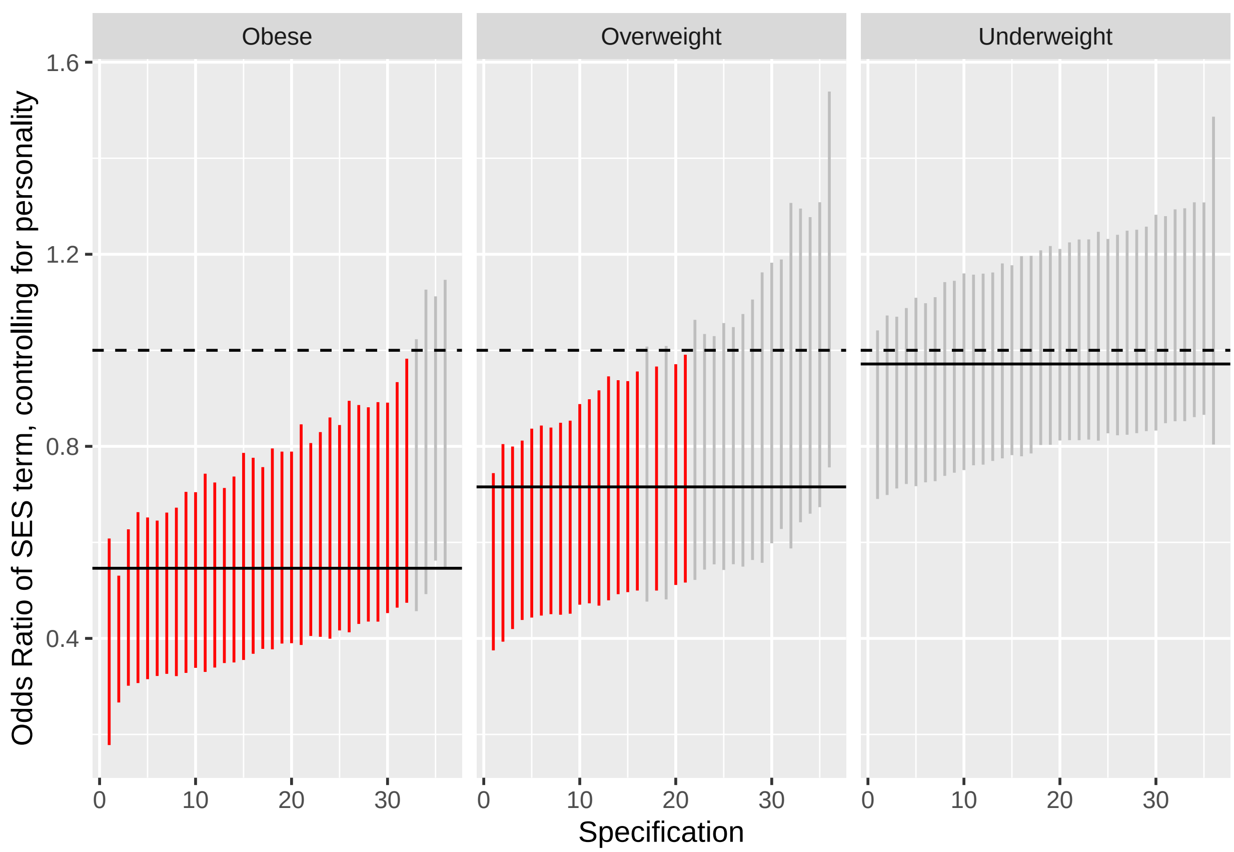

Figure 11: Bars represent the 95% confidence interval of the SES odds ratio for each model (one model for each personality trait and cognitive functioning, 33 in total) using the sample of adolescent females. Bars are red if confidence intervals exclude OR = 1 (i.e., are statitstically significant). The horizontal dashed line is the null hyptothesis, and the horizontal solid line is the average estimate across models.

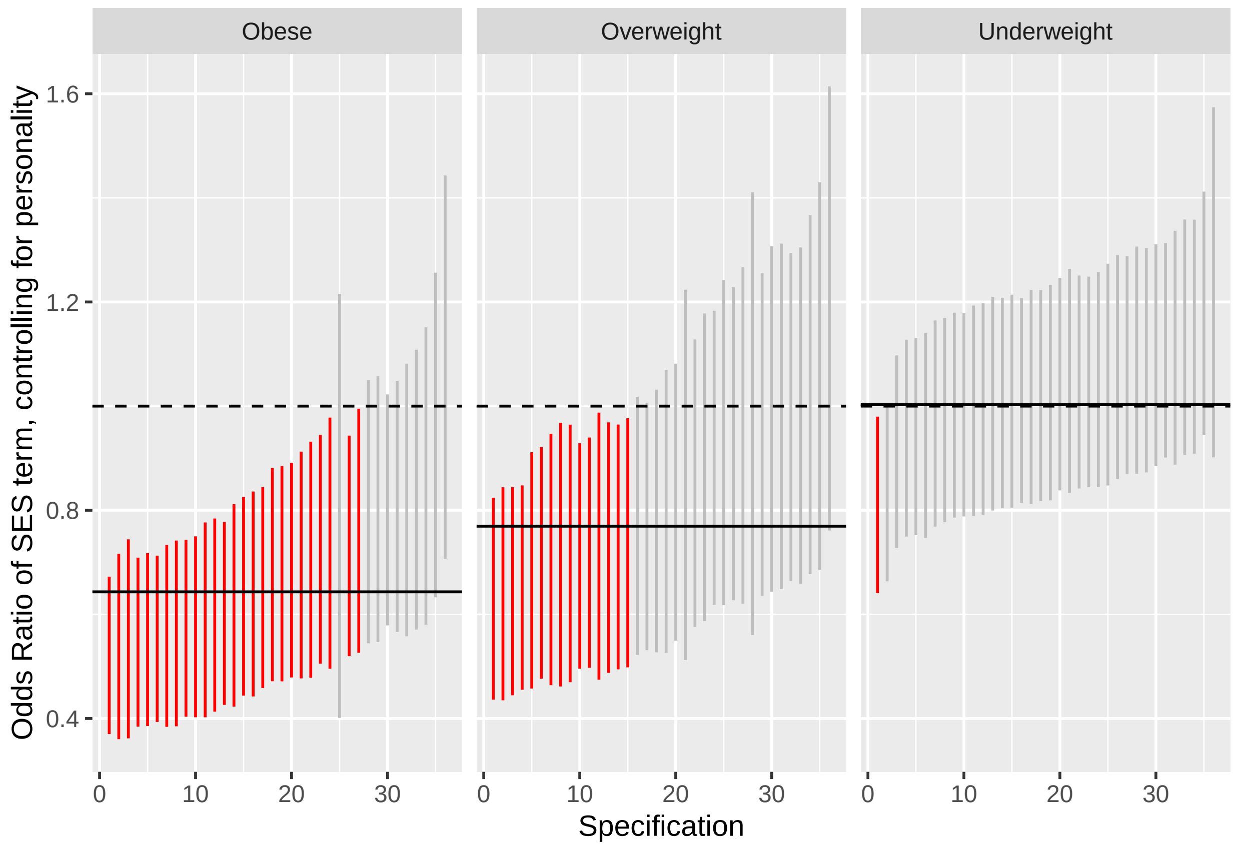

Figure 12: Bars represent the 95% confidence interval of the SES odds ratio for each model (one model for each personality trait and cognitive functioning, 33 in total) using the sample of adolescent males. Bars are red if confidence intervals exclude OR = 1 (i.e., are statitstically significant). The horizontal dashed line is the null hyptothesis, and the horizontal solid line is the average estimate across models.

4.2 Personality (controlling for SES)

4.2.1 Females

A primary finding is that more traits are associated with the likelihood of adolescent females being in the underweight category than are associated with the likelihood of being overweight or obese. This is seen most clearly in the table of coefficient estimates (Table 8). In addition, we show the probability of inclusion in each weight category across levels of personality (Figure 13).

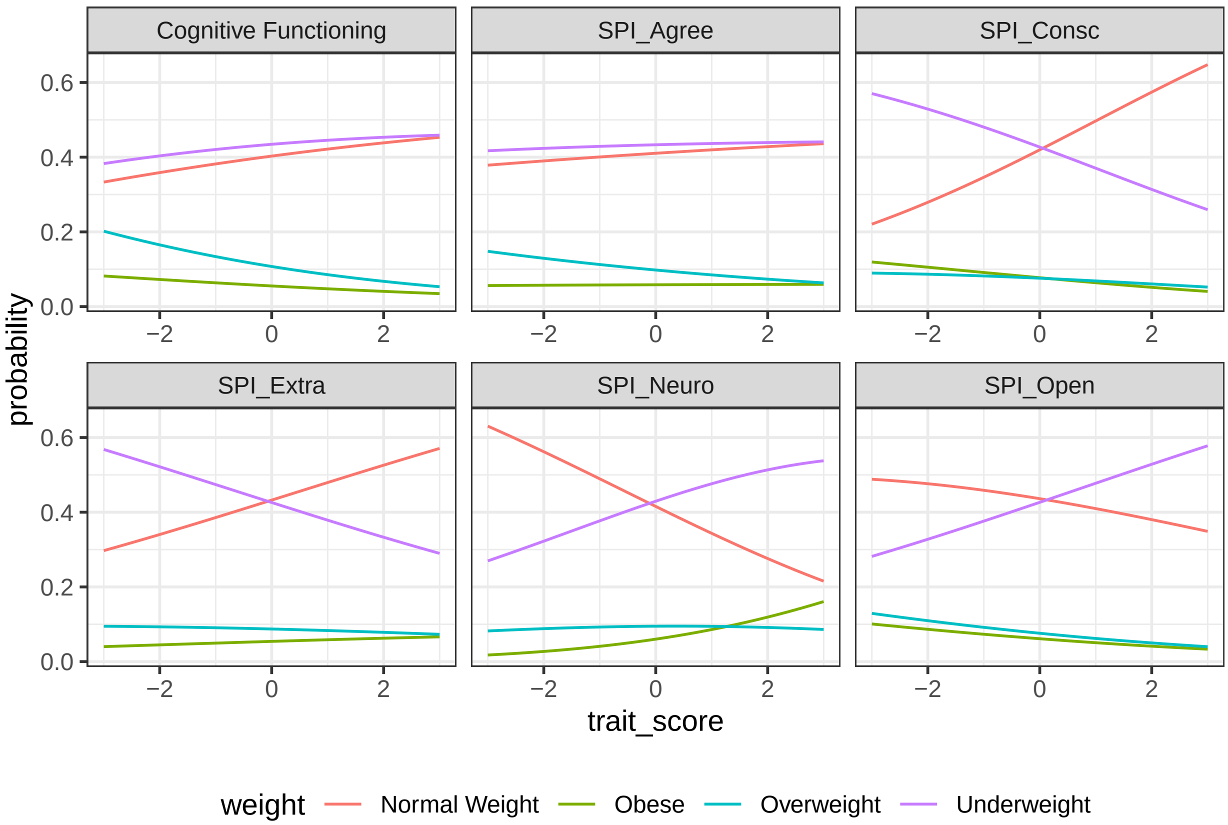

Figure 13: Relationship of traits (cognitive functioning and Big Five) to probability of being in each weight category. Models estimated using the sample of adolescent females.

4.2.2 Males

Among adolescent males, a notable finding was the non-linear association of traits with weight status. More specifically, traits like low Sociability and high Easy-Goingness were associated with both increased risk of obesity and being underweight (Table 9). We show the probability of inclusion in each weight category across levels of personality (Figure **??**).

Table 8: Odds ratios of BMI categories from individual differences estimated from sample of adolescent females. Estimates are adjusted for SES. Significant ($p < .05$) odds ratios are in bold and red.

Trait	Obese	Overweight	Underweight
Cognitive Functioning	0.82 [0.61, 1.12]	**0.76* [0.60, 0.97]**	0.98 [0.84, 1.14]
SPI: 27 Factors			
Compassion	0.94 [0.72, 1.23]	0.99 [0.78, 1.25]	**1.17* [1.00, 1.37]**
Irritability	1.07 [0.81, 1.43]	1.12 [0.89, 1.43]	1.13 [0.97, 1.32]
Sociability	0.95 [0.70, 1.28]	**0.77* [0.61, 0.97]**	**0.67* [0.58, 0.78]**
Well Being	**0.67* [0.50, 0.91]**	**0.64* [0.49, 0.84]**	**0.66* [0.56, 0.77]**
Sensation Seeking	0.95 [0.72, 1.24]	1.18 [0.94, 1.49]	1.00 [0.87, 1.16]
Anxiety	1.07 [0.79, 1.44]	1.04 [0.81, 1.34]	1.13 [0.97, 1.32]
Honesty	1.18 [0.81, 1.73]	0.91 [0.70, 1.20]	1.02 [0.86, 1.20]
Industry	**0.67* [0.50, 0.90]**	0.91 [0.72, 1.15]	**0.77* [0.67, 0.89]**
Intellect	0.97 [0.75, 1.26]	1.12 [0.87, 1.43]	0.94 [0.81, 1.08]
Creativity	0.86 [0.66, 1.11]	0.90 [0.71, 1.13]	1.12 [0.96, 1.30]
Impulsivity	1.00 [0.76, 1.31]	1.16 [0.91, 1.49]	**1.20* [1.04, 1.38]**
Attention Seeking	1.09 [0.82, 1.44]	0.91 [0.70, 1.17]	**0.81* [0.70, 0.94]**
Order	**0.73* [0.55, 0.97]**	0.88 [0.68, 1.13]	0.87 [0.75, 1.02]
Authoritarianism	1.15 [0.83, 1.60]	1.11 [0.87, 1.41]	0.89 [0.77, 1.03]
Charisma	1.16 [0.89, 1.52]	1.01 [0.77, 1.33]	**0.83* [0.72, 0.96]**
Trust	0.79 [0.59, 1.06]	0.83 [0.64, 1.08]	**0.74* [0.63, 0.85]**
Humor	0.94 [0.70, 1.26]	1.17 [0.89, 1.55]	0.93 [0.80, 1.07]
Emotional Expressiveness	1.03 [0.79, 1.35]	0.79 [0.62, 1.00]	**0.86* [0.75, 0.99]**
Art Appreciation	0.95 [0.71, 1.26]	1.03 [0.82, 1.31]	**1.52* [1.26, 1.83]**
Introspection	0.80 [0.62, 1.03]	0.83 [0.66, 1.06]	**1.22* [1.03, 1.44]**
Perfectionism	1.01 [0.76, 1.35]	1.05 [0.82, 1.34]	**0.84* [0.73, 0.97]**
Self Control	0.76 [0.57, 1.02]	0.79 [0.61, 1.03]	0.94 [0.81, 1.10]
Conformity	0.80 [0.61, 1.05]	1.14 [0.89, 1.46]	**0.82* [0.71, 0.96]**
Adaptability	0.97 [0.73, 1.29]	0.91 [0.70, 1.17]	**0.77* [0.66, 0.90]**
Easy Goingness	1.14 [0.84, 1.55]	1.23 [0.94, 1.61]	1.12 [0.95, 1.31]
Emotional Stability	0.84 [0.66, 1.08]	0.80 [0.63, 1.02]	**0.75* [0.65, 0.87]**
Conservatism	0.85 [0.63, 1.14]	0.91 [0.71, 1.18]	0.87 [0.75, 1.02]
SPI: 5 Factors			
Agreeableness	0.99 [0.74, 1.31]	0.85 [0.68, 1.06]	0.99 [0.85, 1.14]
Conscientiousness	**0.70* [0.54, 0.91]**	0.76 [0.58, 1.00]	**0.73* [0.63, 0.85]**
Extraversion	0.98 [0.73, 1.30]	0.86 [0.68, 1.09]	**0.80* [0.70, 0.92]**
Neuroticism	**1.73* [1.27, 2.34]**	1.21 [0.94, 1.54]	**1.34* [1.15, 1.56]**
Openness	0.88 [0.66, 1.17]	0.87 [0.67, 1.13]	**1.19* [1.03, 1.38]**

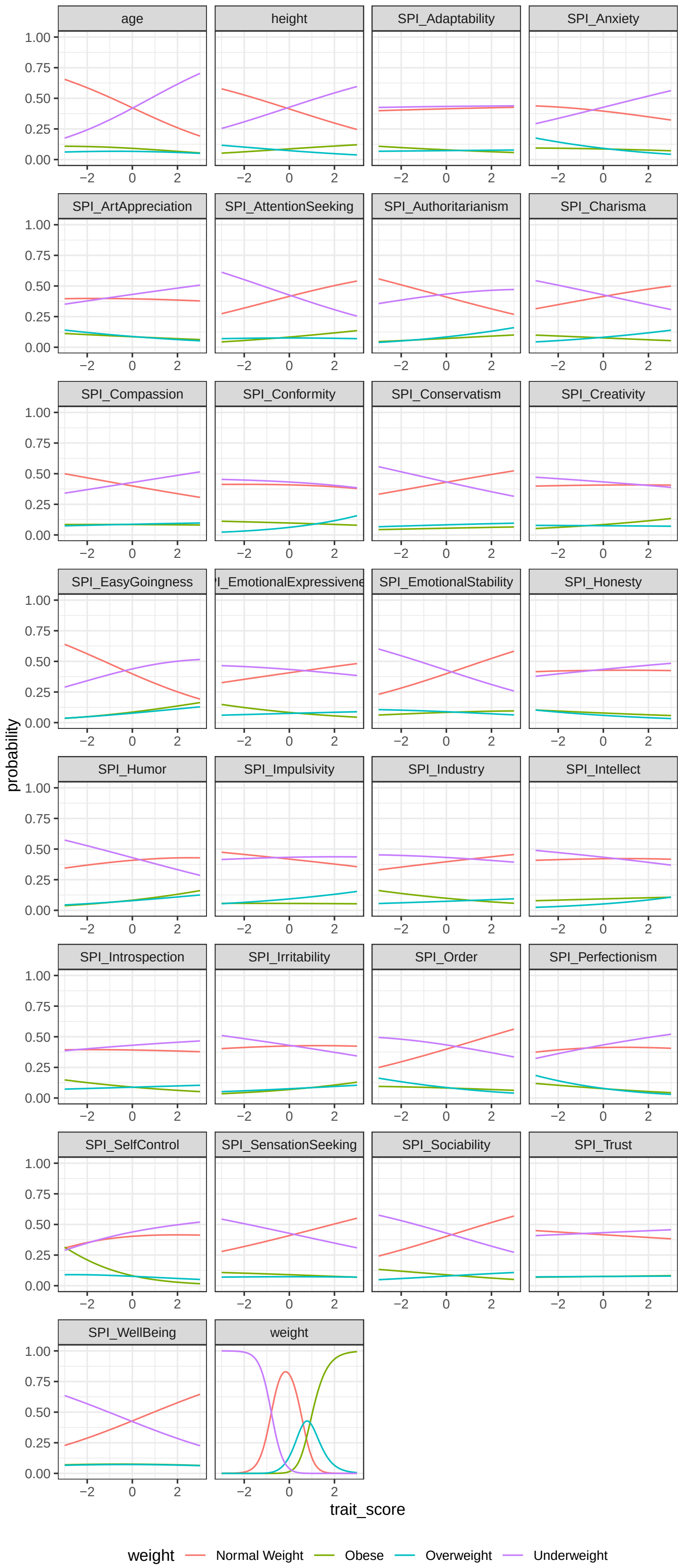
age
height
SPI_Adaptability
SPI_Anxiety
SPI_ArtAppreciation
SPI_AttentionSeeking
SPI_Authoritarianism
SPI_Charisma
SPI_Compassion
SPI_Conformity
SPI_Conservatism
SPI_Creativity
SPI_EasyGoingness
I_EmotionalExpressivene
SPI_EmotionalStability
SPI_Honesty
SPI_Humor
SPI_Impulsivity
SPI_Industry
SPI_Intellect
SPI_Introspection
SPI_Irritability
SPI_Order
SPI_Perfectionism
SPI_SelfControl
SPI_SensationSeeking
SPI_Sociability
SPI_Trust
SPI_WellBeing
weight
probability
1.00
0.75
0.50
0.25
0.00
−2
0
2
trait_score
weight
Normal Weight
Obese
Overweight
Underweight

Table 9: Odds ratios of BMI categories from individual differences estimated from sample of adolescent males Estimates are adjusted for SES. Significant ($p < .05$) odds ratios are in bold and red.

Trait	Obese	Overweight	Underweight
Cognitive Functioning	0.93 [0.72, 1.20]	0.84 [0.63, 1.11]	**0.85* [0.73, 0.99]**
SPI: 27 Factors			
Compassion	1.08 [0.82, 1.41]	1.13 [0.86, 1.49]	1.16 [0.99, 1.36]
Irritability	1.23 [0.92, 1.65]	1.12 [0.84, 1.48]	0.93 [0.79, 1.09]
Sociability	**0.74* [0.58, 0.94]**	0.99 [0.74, 1.32]	**0.77* [0.66, 0.89]**
Well Being	0.83 [0.64, 1.08]	0.83 [0.64, 1.09]	**0.71* [0.61, 0.82]**
Sensation Seeking	0.83 [0.64, 1.07]	0.89 [0.67, 1.18]	**0.81* [0.70, 0.95]**
Anxiety	1.01 [0.78, 1.30]	0.83 [0.65, 1.06]	**1.17* [1.00, 1.37]**
Honesty	0.90 [0.70, 1.17]	0.83 [0.62, 1.10]	1.04 [0.88, 1.22]
Industry	0.80 [0.62, 1.03]	1.03 [0.78, 1.37]	0.93 [0.79, 1.08]
Intellect	1.05 [0.80, 1.38]	1.28 [0.87, 1.87]	0.95 [0.81, 1.12]
Creativity	1.17 [0.90, 1.52]	0.98 [0.75, 1.29]	0.97 [0.83, 1.12]
Impulsivity	1.04 [0.77, 1.41]	1.25 [0.98, 1.60]	1.06 [0.91, 1.23]
Attention Seeking	1.08 [0.83, 1.41]	0.89 [0.69, 1.16]	**0.77* [0.67, 0.89]**
Order	0.82 [0.63, 1.06]	**0.69* [0.53, 0.90]**	**0.82* [0.70, 0.95]**
Authoritarianism	1.28 [0.95, 1.75]	**1.43* [1.06, 1.92]**	**1.18* [1.01, 1.39]**
Charisma	0.84 [0.65, 1.07]	1.12 [0.87, 1.46]	**0.84* [0.73, 0.97]**
Trust	1.06 [0.80, 1.40]	1.04 [0.78, 1.38]	1.05 [0.89, 1.22]
Humor	1.23 [0.91, 1.65]	1.15 [0.86, 1.54]	0.86 [0.74, 1.00]
Emotional Expressiveness	**0.77* [0.59, 0.99]**	1.00 [0.77, 1.30]	0.91 [0.78, 1.05]
Art Appreciation	0.91 [0.71, 1.18]	0.86 [0.67, 1.09]	1.07 [0.91, 1.26]
Introspection	0.85 [0.67, 1.07]	1.07 [0.82, 1.39]	1.04 [0.89, 1.21]
Perfectionism	0.83 [0.64, 1.09]	**0.73* [0.56, 0.94]**	1.07 [0.92, 1.24]
Self Control	**0.58* [0.45, 0.75]**	0.87 [0.66, 1.14]	1.05 [0.89, 1.23]
Conformity	0.96 [0.75, 1.22]	**1.40* [1.04, 1.88]**	0.99 [0.85, 1.15]
Adaptability	0.89 [0.68, 1.16]	1.01 [0.77, 1.33]	0.99 [0.86, 1.15]
Easy Goingness	**1.58* [1.20, 2.09]**	**1.51* [1.12, 2.04]**	**1.34* [1.15, 1.57]**
Emotional Stability	0.92 [0.71, 1.20]	0.79 [0.61, 1.00]	**0.74* [0.64, 0.87]**
Conservatism	0.99 [0.73, 1.35]	0.98 [0.75, 1.28]	**0.84* [0.73, 0.98]**
SPI: 5 Factors			
Agreeableness	1.17 [0.89, 1.54]	0.88 [0.67, 1.16]	1.13 [0.97, 1.32]
Conscientiousness	**0.75* [0.57, 0.99]**	**0.71* [0.52, 0.97]**	0.93 [0.80, 1.09]
Extraversion	0.80 [0.62, 1.04]	1.03 [0.78, 1.37]	**0.76* [0.65, 0.89]**
Neuroticism	1.18 [0.91, 1.53]	0.86 [0.67, 1.10]	1.11 [0.95, 1.29]
Openness	1.04 [0.80, 1.37]	0.81 [0.62, 1.07]	1.00 [0.86, 1.17]

4.3 Interaction of SES with personality

To estimate the joint effect of socioeconomic status and individual differences on BMI category, we graph the estimates of the interaction terms of SES by individual differences by BMI category. Like before, we present the average effect (solid black line) and the 95% confidence intervals for each model.

```
load(here("data/logistic_output.Rdata"))
```

4.3.1 Females

We test the interaction of SES and personality among adolescent females. Few interactions were statistically significant (see Figure 14)

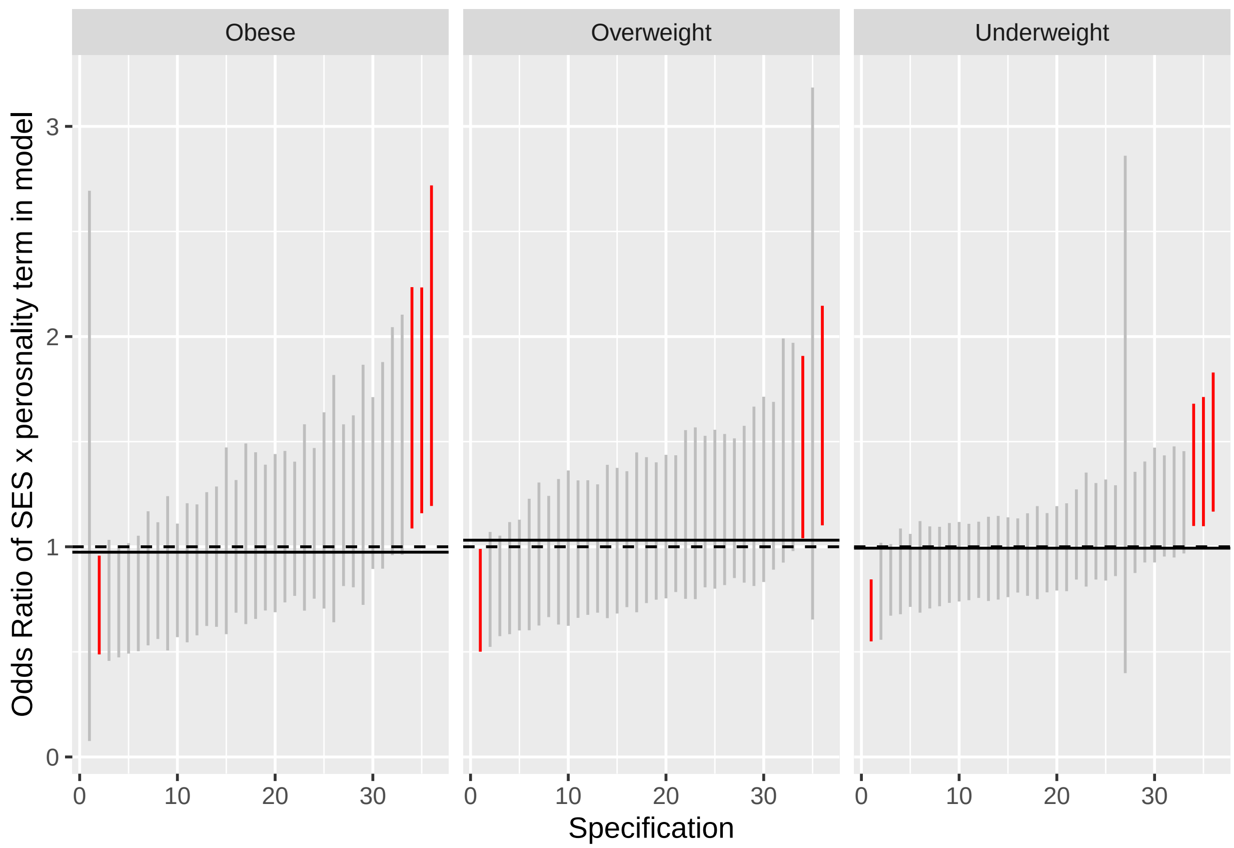

Figure 14: Coefficient estimates for interaction of SES and personality traits, predicting BMI category (adolescent females).

4.3.2 Males

We test the interaction of SES and personality among adolescent females. Few interactions were statistically significant (see Figure 15)

Parental socioeconomic status positively predicted greater likelihood of all non-normal categories (Underweight, Overweight, and Obese) compared to Normal among males. Adolescent males living in higher SES households were, on average, -98% more likely to be Underweight, -104% more likely, and -100% more likely to be Obese compared to low SES counterparts.

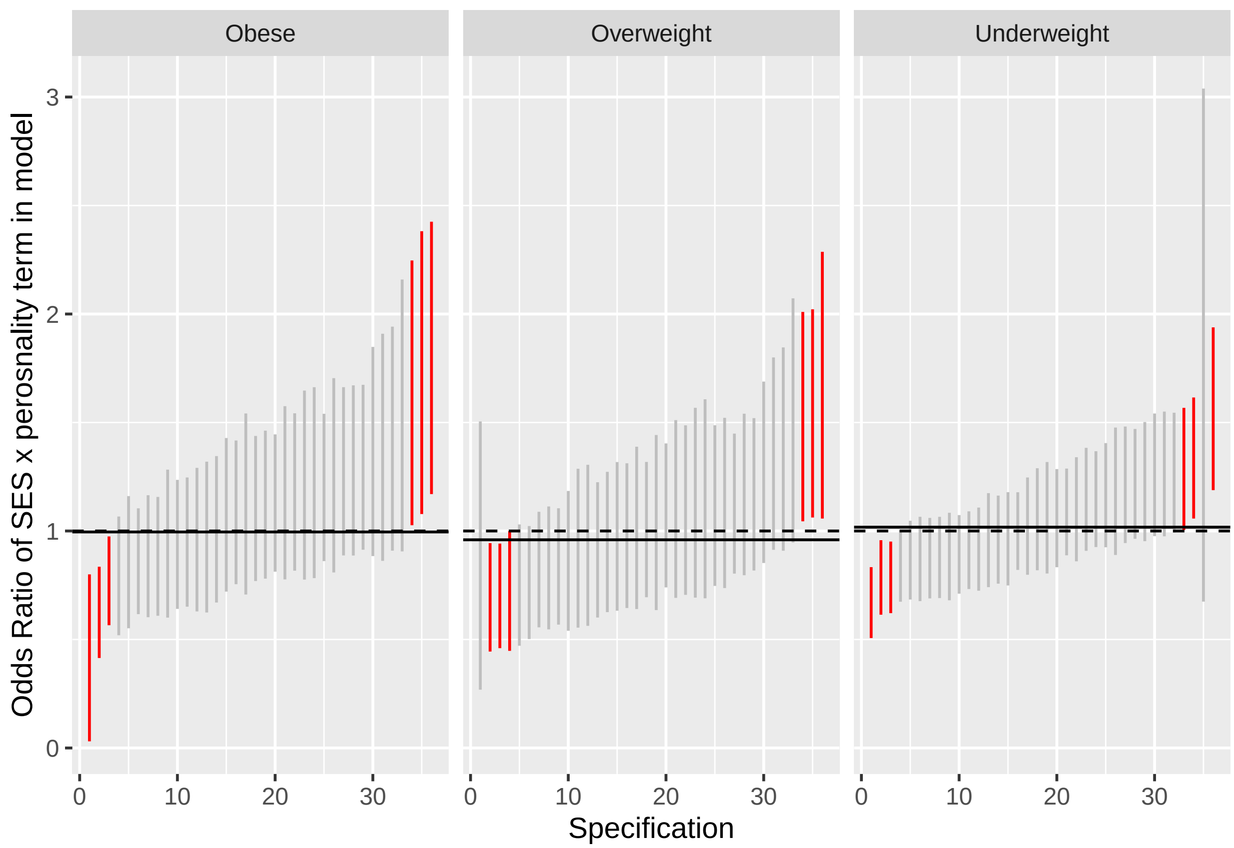

Figure 15: Coefficient estimates for interaction of SES and personality traits, predicting BMI category (adolescent males).

5 Model accuracy

We use the splits generated in the Cleaning data section of the notebook to create our training and test data. (Note that to this point, only the training data have been used in the regression models).

```
female_train = sapa_female[train_female, ] %>% select(BMI_p, ses, cog, contains("SPI")) %>% filter(compl
female_test = sapa_female[-train_female, ] %>% select(BMI_p, ses, cog, contains("SPI")) %>% filter(compl
female_bmi = female_train$BMI_p

male_train = sapa_male[train_male, ] %>% select(BMI_p, ses, cog, contains("SPI")) %>% filter(complete.ca
male_test = sapa_male[-train_male, ] %>% select(BMI_p, ses, cog, contains("SPI")) %>% filter(complete.ca
male_bmi = male_train$BMI_p
```

We build a function to fit the lasso models.

```
fit_model = function(data, outcome){
  cv_value = model.matrix(BMI_p ~ .,
                 data = data) %>%
    cv.glmnet(x = .,
              y = outcome,
              alpha = 1)
  model =  model.matrix(BMI_p ~ ., data = data) %>%
    glmnet(y = outcome,
           alpha = 1,
           lambda = cv_value$lambda.min)
  return(model)
}
```

We also a build a model to get the predictions in the test set from the model best fit in the training data.

```
pred_model = function(model, test.data){

  if(length(model$coefficients) == 2){
    x = test.data
    }else{
      x = test.data[, c("BMI_p", rownames(model$beta)[-1])]
      x <- model.matrix(BMI_p~., x)
      }
  predictions = model %>% predict(x) %>% as.vector()
  # Model performance metrics
  fit = data.frame(
    RMSE = RMSE(predictions, test.data$BMI_p),
    Rsquare = R2(predictions, test.data$BMI_p)
  )
  return(fit)
}
```

We fit these models separately for adolescent males and adolescent females

```
set.seed(060821)

mod1_f = lm(BMI_p ~ ses, data = female_train)
```

```r
mod2_f = female_train %>%
  select(BMI_p, ses, cog) %>%
  fit_model(data = ., outcome = female_bmi)
mod3_f = female_train %>%
  select(BMI_p, ses, contains("SPI")) %>%
  select(1:7) %>%
  fit_model(data = ., outcome = female_bmi)
mod4_f = female_train %>%
  select(BMI_p, ses, contains("SPI")) %>%
  select(1:2,8:34) %>%
  fit_model(data = ., outcome = female_bmi)
mod5_f = female_train %>%
  select(BMI_p, ses, cog, contains("SPI")) %>%
  select(1:8) %>%
  fit_model(data = ., outcome = female_bmi)
mod6_f = female_train %>%
  select(BMI_p, ses, cog, contains("SPI")) %>%
  select(1:3,9:35) %>%
  fit_model(data = ., outcome = female_bmi)

female_fits = data.frame(
  vars = c(
    "SES only",
    "SES + Cog",
    "SES + Big Five",
    "SES + Narrow 27",
    "SES + Cog + Big Five",
    "SES + Cog + Narrow 27"))
female_fits$model = list(mod1_f, mod2_f, mod3_f, mod4_f, mod5_f, mod6_f)
female_fits = mutate(female_fits, fits = map(model, pred_model, test.data = female_test))

female_fits = female_fits %>%
  select(-model) %>%
  unnest(cols = c(fits)) %>%
  mutate(gender = "Adolescent Females")
```

```r
set.seed(060821)

mod1_m = lm(BMI_p ~ ses, data = male_train)
mod2_m = male_train %>%
  select(BMI_p, ses, cog) %>%
  fit_model(data = ., outcome = male_bmi)
mod3_m = male_train %>%
  select(BMI_p, ses, contains("SPI")) %>%
  select(1:7) %>%
  fit_model(data = ., outcome = male_bmi)
mod4_m = male_train %>%
  select(BMI_p, ses, contains("SPI")) %>%
  select(1:2,8:34) %>%
  fit_model(data = ., outcome = male_bmi)
mod5_m = male_train %>%
  select(BMI_p, ses, cog, contains("SPI")) %>%
  select(1:8) %>%
```

```
  fit_model(data = ., outcome = male_bmi)
mod6_m = male_train %>%
  select(BMI_p, ses, cog, contains("SPI")) %>%
  select(1:3,9:35) %>%
  fit_model(data = ., outcome = male_bmi)

male_fits = data.frame(
  vars = c(
    "SES only",
    "SES + Cog",
    "SES + Big Five",
    "SES + Narrow 27",
    "SES + Cog + Big Five",
    "SES + Cog + Narrow 27"))
male_fits$model = list(mod1_m, mod2_m, mod3_m, mod4_m, mod5_m, mod6_m)
male_fits = mutate(male_fits, fits = map(model, pred_model, test.data = male_test))

male_fits = male_fits %>%
  select(-model) %>%
  unnest(cols = c(fits)) %>%
  mutate(gender = "Adolescent Males")
```

The results are shown in Table 10. Across gender, the models with the highest R^2 values (i.e., the best out-of-sample prediction) included SES, cognitive functioning, and the narrow 27 traits. For the sample of adolescent males, this model accounted for more than twice the variance as the model using Big Five traits instead of narrow 27; for adolescent females, this model accounted for about 30% more variance.

```
female_fits %>%
  full_join(male_fits) %>%
  mutate(gender = str_remove(gender, "Adolescent ")) %>%
  gather(stat, value, starts_with("R")) %>%
  unite(stat, gender, stat) %>%
  spread(stat, value) %>%
  mutate(vars = factor(vars, levels = c("SES only",
                                        "SES + Cog",
                                        "SES + Big Five",
                                        "SES + Narrow 27",
                                        "SES + Cog + Big Five",
                                        "SES + Cog + Narrow 27"))) %>%
  arrange(vars) %>%
  kable(col.names = c("Model", rep(c("RMSE", "R-squared"), 2)),
        booktabs = T,
        caption = "Out of sample model fits from lasso regression models. Models including cognitive fu
        digits = c(0,2,3,2,3)) %>%
  kable_styling() %>%
  add_header_above(c(" ", "Adolescent Males" = 2, "Adolescent Females" = 2))
```

6 Sensitivity analysis: Missing data

Once we filter for adolescents living in the United States, approximately half our sample did not report either their height or weight. Given the sensitivity of body image, especially for the adolescent females in

Table 10: Out of sample model fits from lasso regression models. Models including cognitive functioning and the Narrow 27 traits yielded the best out-of-sample prediction for both genders

	Adolescent Males		Adolescent Females	
Model	RMSE	R-squared	RMSE	R-squared
SES only	27.02	0.031	30.09	0.020
SES + Cog	26.95	0.036	30.02	0.024
SES + Big Five	27.02	0.030	30.11	0.020
SES + Narrow 27	26.90	0.038	29.76	0.052
SES + Cog + Big Five	26.96	0.034	30.05	0.024
SES + Cog + Narrow 27	26.86	0.041	29.64	0.055

our sample, we suspect these values are missing not at random (MNAR) and may impact the estimates here. To test for these effects, we imputed missing height and weight values using a principal components analysis approach, using only the other variables in the SAPA dataset that were not included in the analyses above. We repeated the regression models with 10-fold cross validation, repeated 10 times, and report here the differences in significance across models and the differences in effect sizes.

Imputation using a principle components analysis suggested that many of the missing BMI percentiles were between 75 and 100 for both adolescent females and males (see Figure 16). This would be consistent with a common sense rationale for missing data; in other words, we might expect those adolescents with larger BMI to be the individuals least likely to share their height and weight. However, the proposed distributions from the imputation were heavily skewed. We would not expect to see a distribution like this unless there were some association between BMI and completing the online personality assessment, which is unlikely.

6.1 Females

We compared the significance of coefficients across models using the complete dataset and the imputed dataset, for adolescent females (Figure **??**). Approximately 6 traits were significantly associated with BMI using the imputed dataset but not complete; meanwhile, 4 traits were significant using complete data but not imputed. The majority of trait associations had the same significance across datasets. In all cases, SES was always a significant predictor. In one case, there was a significant interaction effect when using imputed data, but this seems likely be to sampling error.

We also calculated the difference in estimated effect size across the two samples. Figure 18 presents the difference in effect sizes, which is the effect estimated in the imputed data minus the effect estimated in the complete dataset. In other words, positive values indicate the estimated effect size is larger when using imputed data, while negative values indicated the estimated effect size is smaller when using imputed data.

Estimated of the effect of SES (controlling for personality) were lower in the imputed dataset for all models. Half of the trait estimates were larger in the imputed dataset, as were half of the interaction estimates.

6.2 Males

We compared the significance of coefficients across models using the complete dataset and the imputed dataset, for adolescent males (Figure 19). Approximately 11 traits were significantly associated with BMI using the imputed dataset but not complete; meanwhile, 4 traits were significant using complete data but not imputed. This discrepancy may reflect a lack of statistical power in the adolescent males sample or bias in the complete cases.

For both the complete and imputed datasets, SES was always a significant predictor. None of the interaction terms found in the complete data were present in the imputed data (although we note these significant

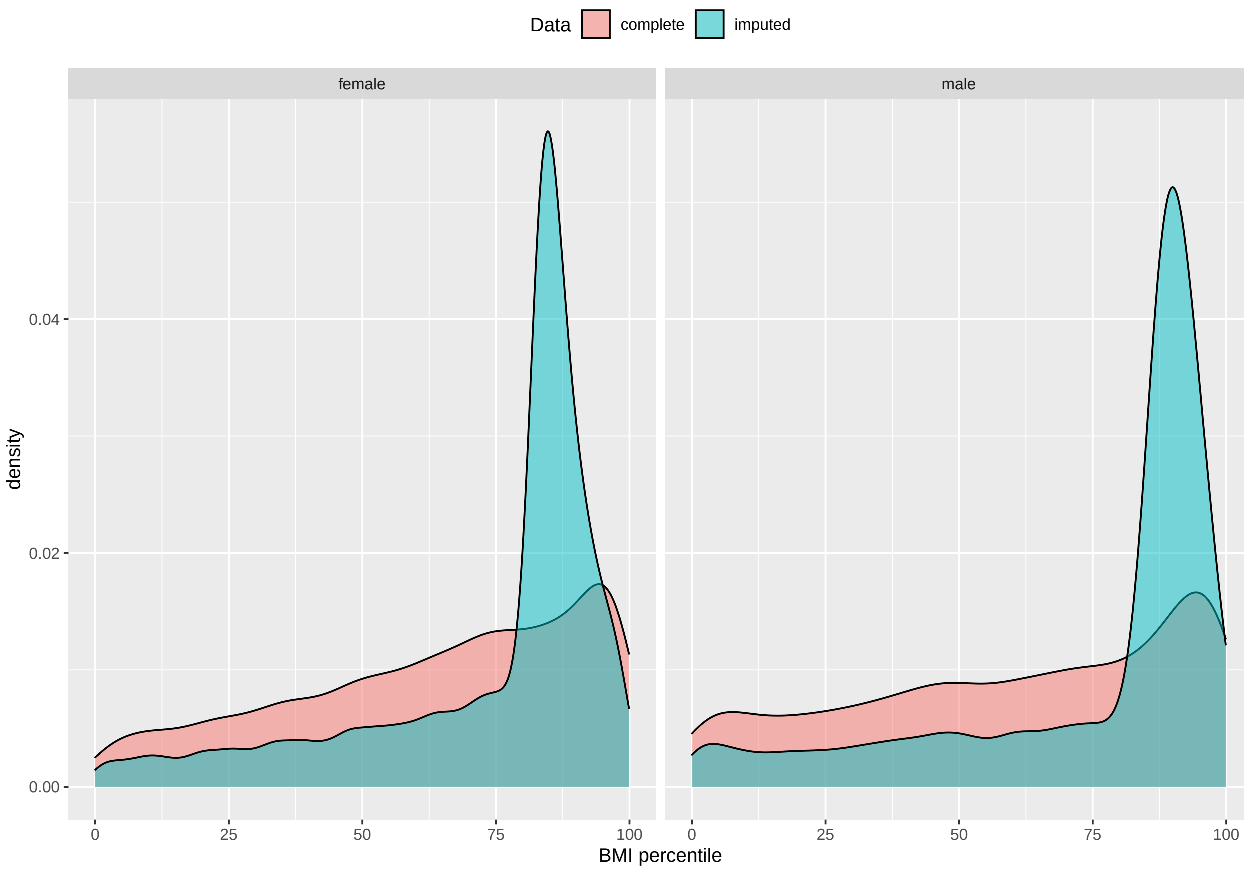

Figure 16: A comparison of BMI percentile distribution in complete and imputed datasets.

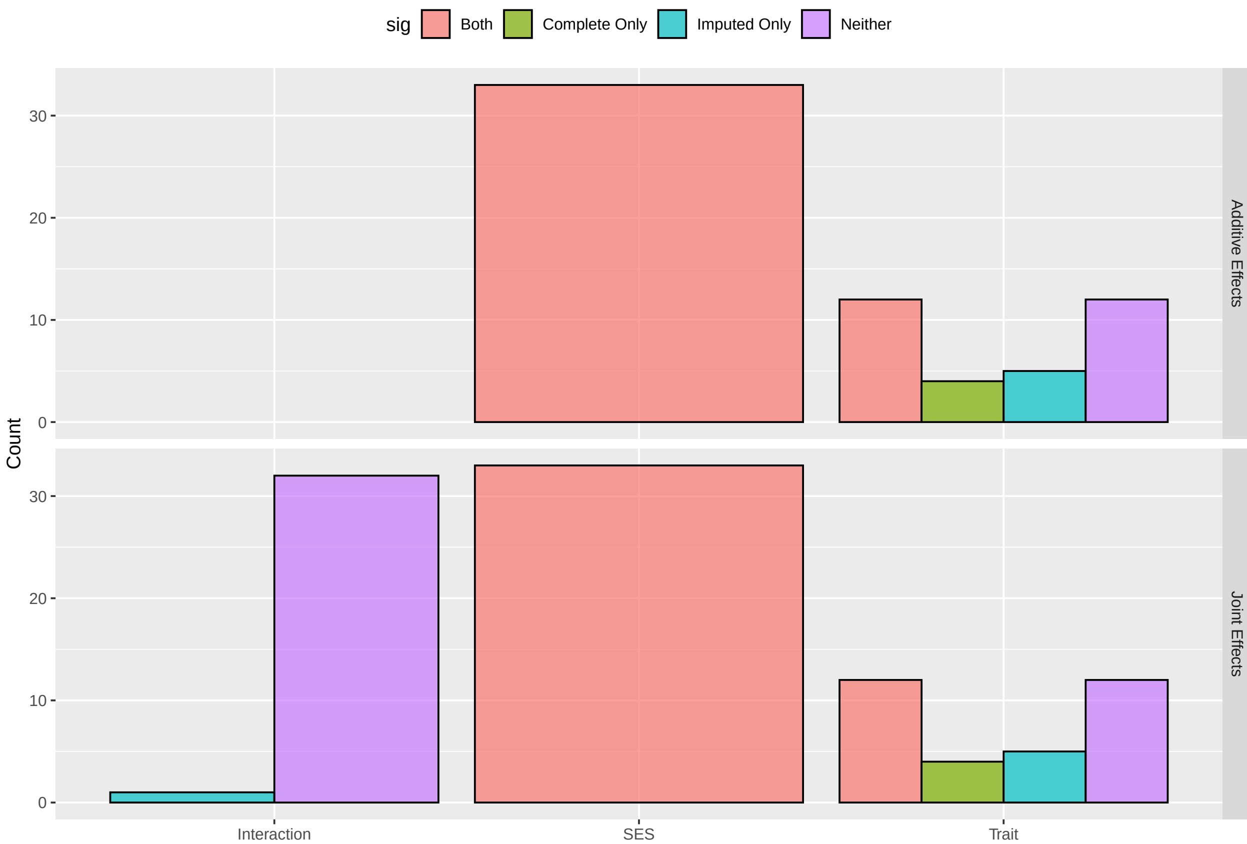

Figure 17: Statistical significance across models fit on adolescent females sample.

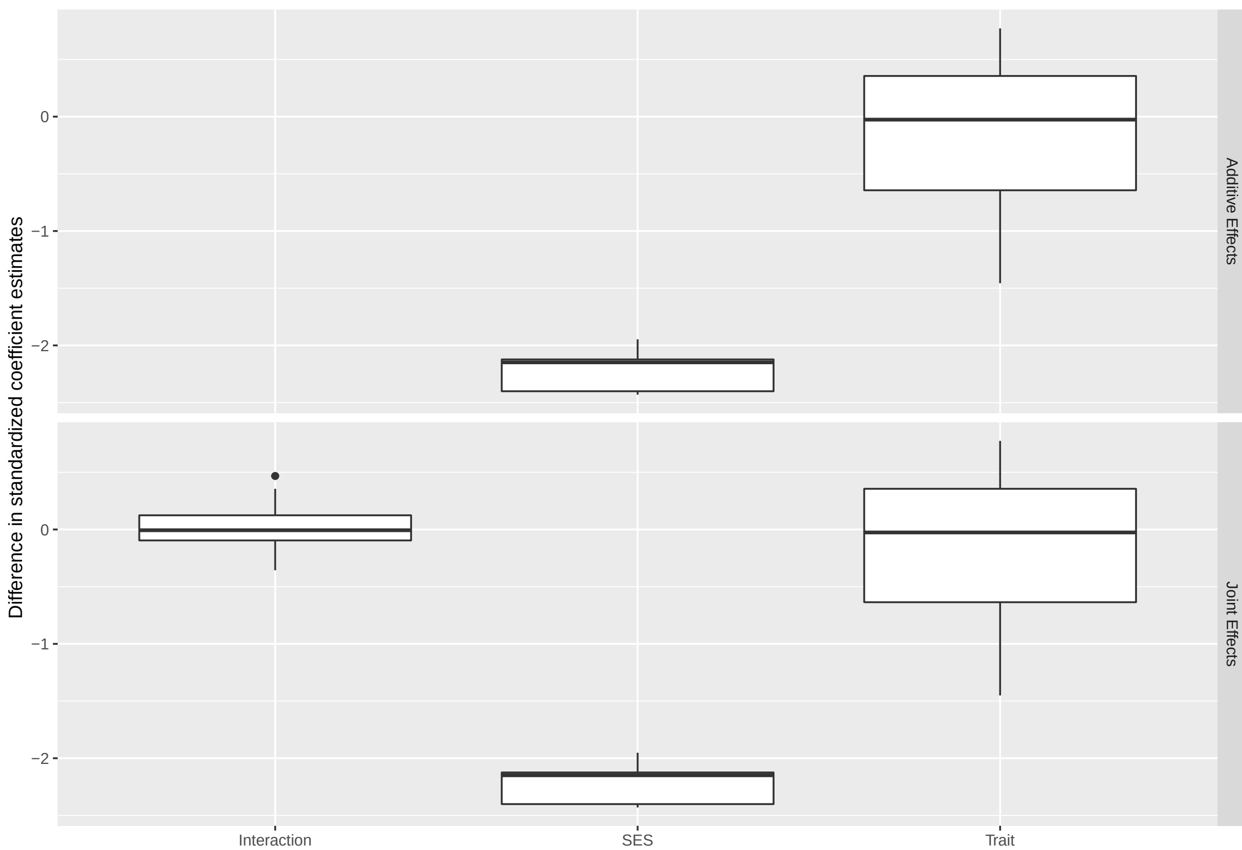

Figure 18: Difference in effect sizes from the imputed data to the complete data (females).

interactions were assumed to be sampling error and not interpreted in the manuscript); the imputed data yielded no joint effects of personality and SEDS.

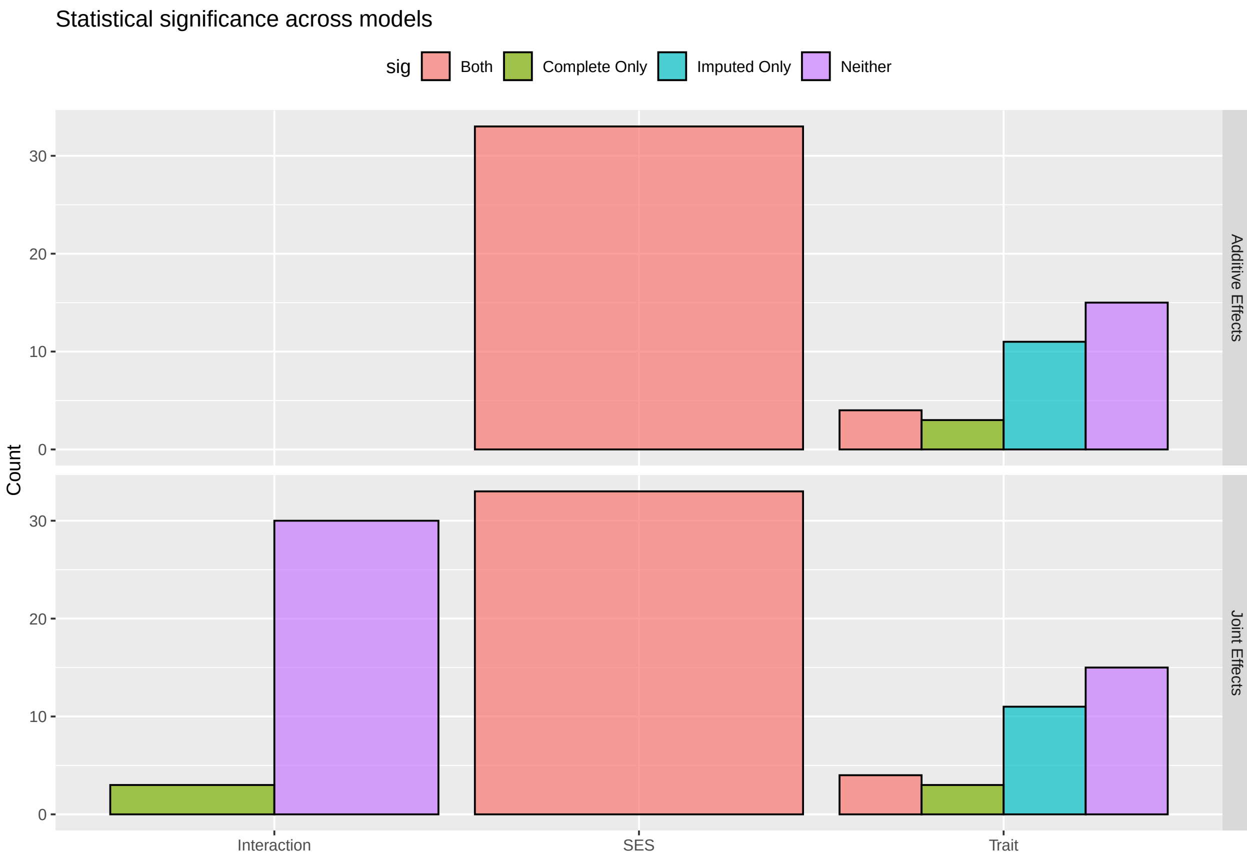

Figure 19: Statistical significance across models fit on adolescent males sample.

Again, we calculated the difference in estimated effect size across the two samples. Figure 20 presents the difference in effect sizes, which is the effect estimated in the imputed data minus the effect estimated in the complete dataset.

Estimated of the effect of SES (controlling for personality) were lower in the imputed dataset for all models. However, a majority of the estimates of trait effects were larger in the imputed dataset, as were a majority of the interaction estimates.

Hibbing, P. R. (2020). *PAutilities: Streamline physical activity research*. https://github.com/paulhibbing/PAutilities

Kuczmarski, R. J. (2002). *2000 CDC Growth Charts for the United States: Methods and development*. Department of Health; Human Services, Centers for Disease Control; . . . .

Revelle, W. (2021). *Psych: Procedures for psychological, psychometric, and personality research*. https://personality-project.org/r/psych/%20https://personality-project.org/r/psych-manual.pdf

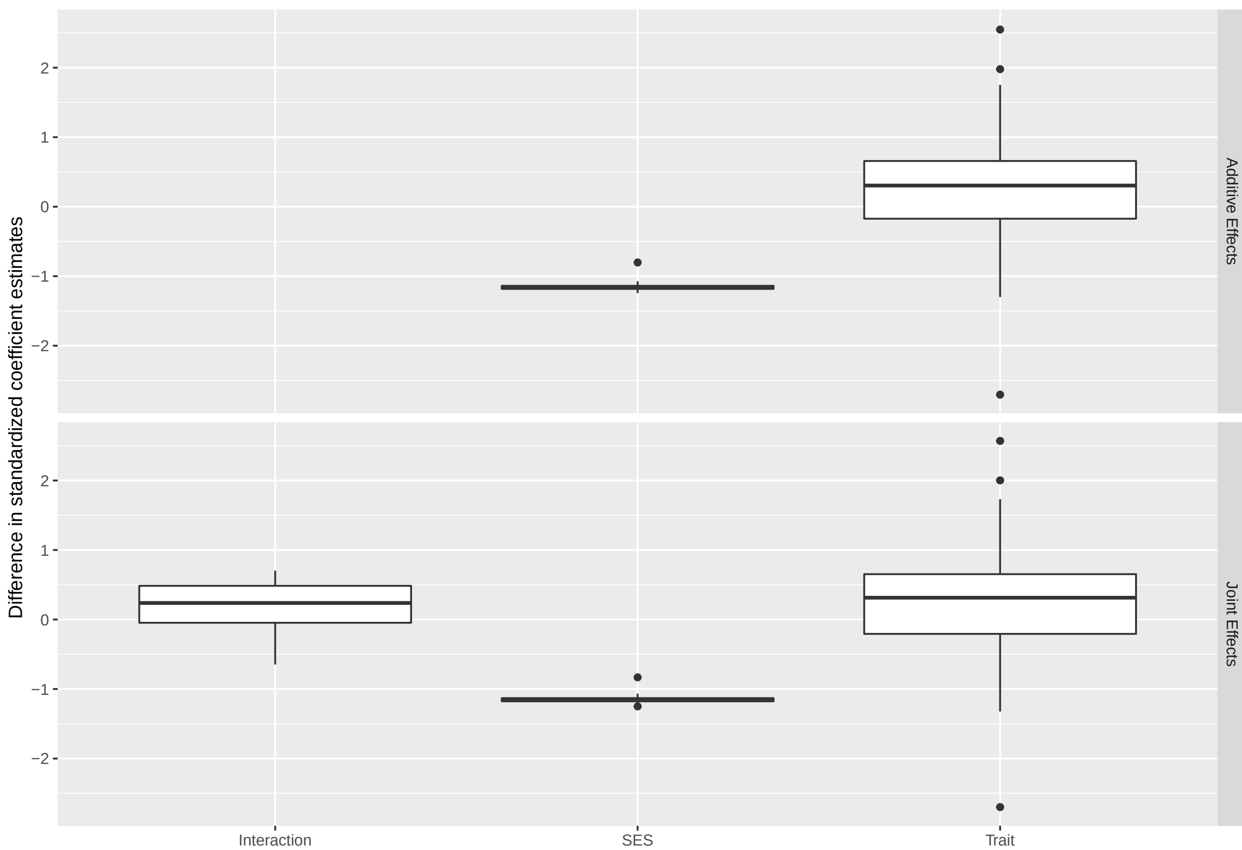

Figure 20: Difference in effect sizes from the imputed data to the complete data (males).