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.

1 Clean data

```
set.seed(052319)

# load packages
packages = c("tidyverse", "janitor", "psych", "devtools",
             "PAutilities", "measurements", "here", "caret")
lapply(packages, library, character.only = TRUE)
rm(packages)

#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)

# super key -- this contains the master key list for all of SAPA. every item ever administered and every
# each row is a single item
# each column is a scale
# the value of a cell is 0 if that item is not part of that scale, 1 if that item positively loads on th
```

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.

```
# 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: 1 (less than 12 years of education), 2 (high school graduate or GED), 3 (some college), 4 (currently in college/university), 5 (Associate's degree), 6 (university degree), 7 (currently in graduate or professional school), and 8 (graduate or professional 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.

```
# 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",
    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.

```
# 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.

```
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)
```

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

```
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, 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, 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.

Kuczmarski RJ, Ogden CL, Guo SS, et al. 2000 CDC growth charts for the United States: methods and development. National Center for Health Statistics. *Vital Health Stat 11. 2002;*(246):1-190

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 labelled Normal.

All analyses were performed separately by gender.

```r
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.

```r
# 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)
```

2 Descriptive Statistics

2.1 Univariate descriptives

Descriptive statistics are estimated using the `psych` package.

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

```
library(kableExtra)
#pull descriptives 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,
        longtable = T,
        booktabs = T) %>%
  kable_styling() %>%
  landscape() %>%
  add_header_above(c(" " = 1, "Female" = 5, "Male" = 5))
```

	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

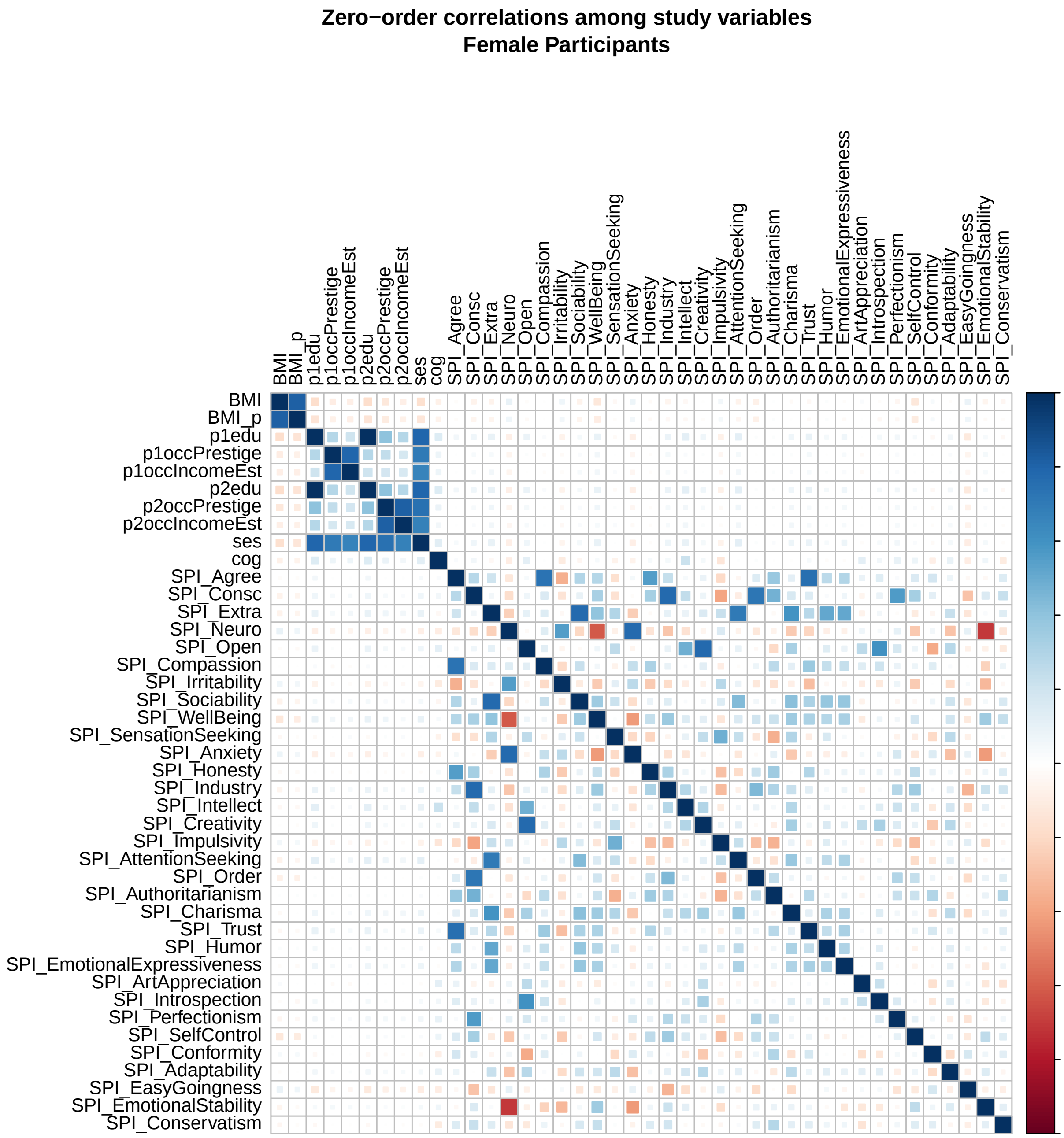
SPI_Introspection	6529	50.04	9.89	15.16	62.58	2951	49.92	10.21	15.16	62.58
SPI_Irritability	6530	50.97	9.87	27.87	72.50	2952	47.83	9.94	27.87	72.50
SPI_Neuro	6530	4.31	0.77	1.29	6.00	2952	3.79	0.81	1.00	6.00
SPI_Open	6530	4.48	0.55	2.07	6.00	2952	4.55	0.53	2.21	6.00
SPI_Order	6529	50.18	10.06	25.20	74.44	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

2.2 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])

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")
```

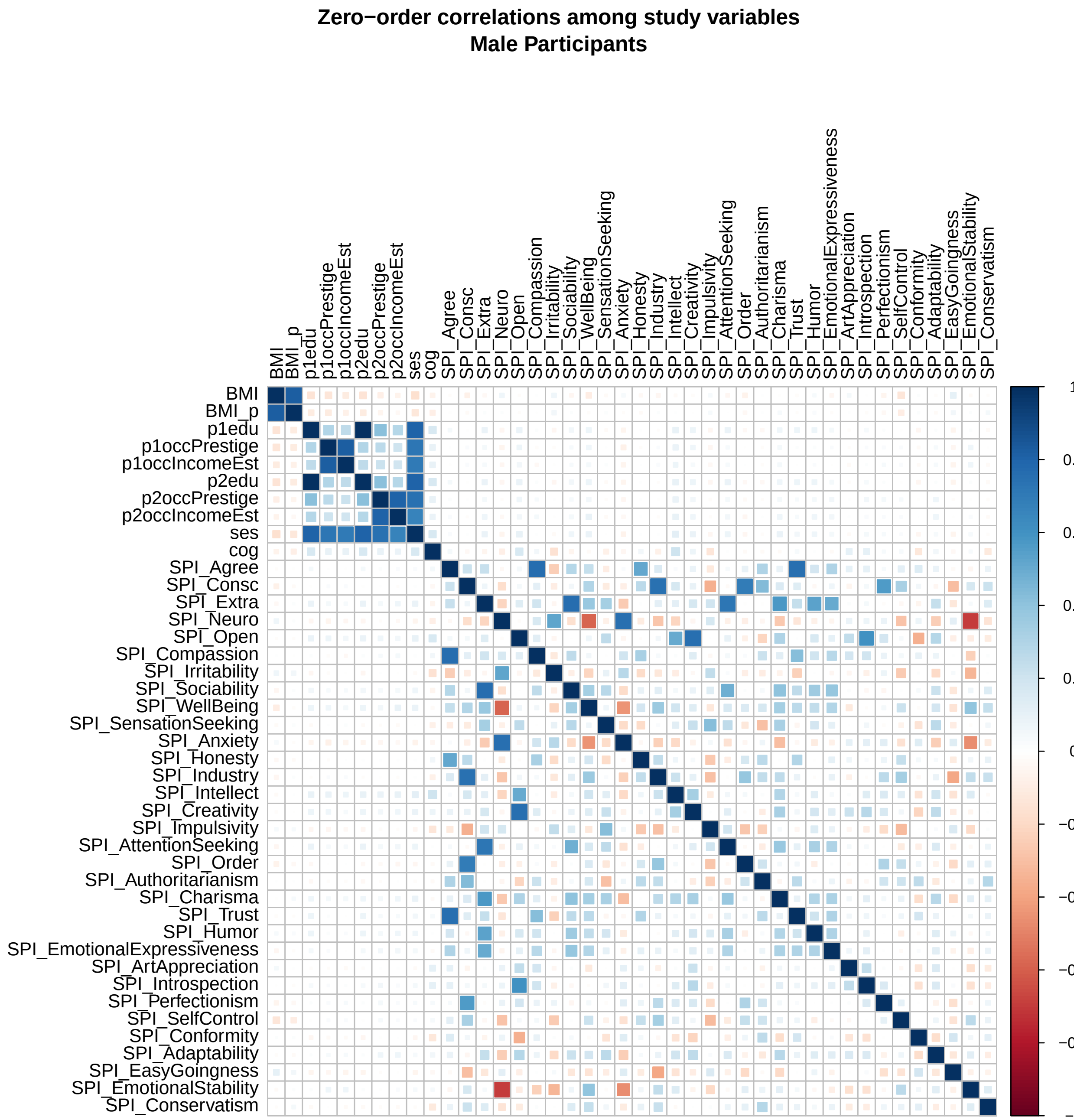
2.2.1 Figure

```
library(corrplot)
corrplot(R_female, method = "square",
         title = "\nZero-order correlations among study variables\nFemale Participants",
         tl.col = "black",
         mar=c(0,0,1,0))
```



2.2.1.1 Female

```
corrplot(R_male, method = "square",
         title = "\nZero-order correlations among study variables\nMale Participants",
         tl.col = "black",
         mar=c(0,0,1,0))
```


Zero−order correlations among study variables
Male Participants
BMI
BMI_p
p1edu
p1occPrestige
p1occIncomeEst
p2edu
p2occPrestige
p2occIncomeEst
ses
cog
SPI_Agree
SPI_Consc
SPI_Extra
SPI_Neuro
SPI_Open
SPI_Compassion
SPI_Irritability
SPI_Sociability
SPI_WellBeing
SPI_SensationSeeking
SPI_Anxiety
SPI_Honesty
SPI_Industry
SPI_Intellect
SPI_Creativity
SPI_Impulsivity
SPI_AttentionSeeking
SPI_Order
SPI_Authoritarianism
SPI_Charisma
SPI_Trust
SPI_Humor
SPI_EmotionalExpressiveness
SPI_ArtAppreciation
SPI_Introspection
SPI_Perfectionism
SPI_SelfControl
SPI_Conformity
SPI_Adaptability
SPI_EasyGoingness
SPI_EmotionalStability
SPI_Conservatism
1

2.2.1.2 Male

3 Regression Models (SES)

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 ability – and tested each one independently in the model as an individual difference.

Models were estimated separately for men and women.

```r
#end goal of wrangling is a data frame of data frames
# nested dataframes 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
```

3.1 Controlling for personality

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.

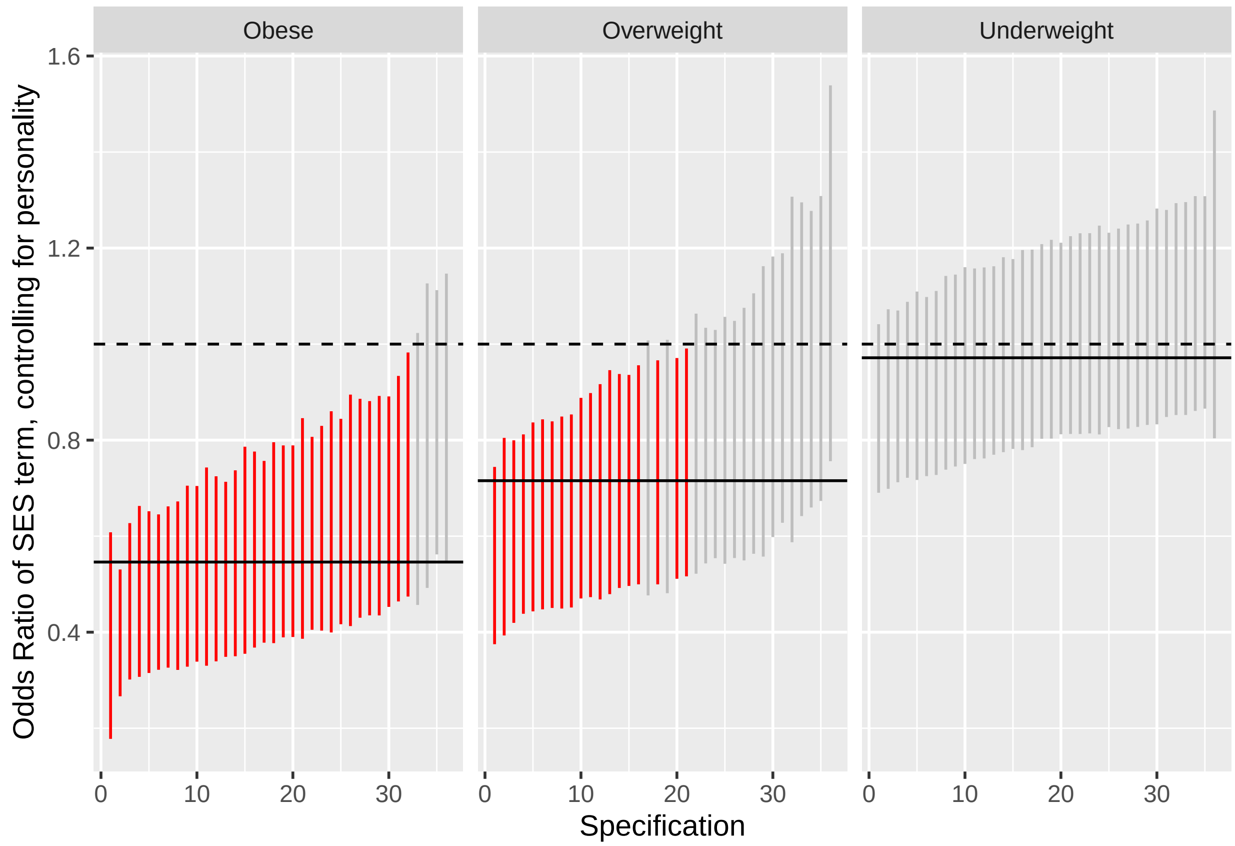
3.1.1 Female

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

```
female_plot_1 = female_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, color = "grey")) +
  geom_hline(aes(yintercept = 0), linetype = "dashed") +
  geom_hline(aes(yintercept = mean), data = avg_female) +
  #geom_label(aes(x = 25, y = 1.25, label = round(mean,2)), 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 Girls") +
  guides(color = F) +
  theme_pubr()
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```

```
female_plot_1
```



3.1.2 Male

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

```
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 Boys", y = NULL) +
  guides(color = F) +
  theme_pubr()
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```

```
male_plot_1
```

3.2 Interaction 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.

3.2.1 Female

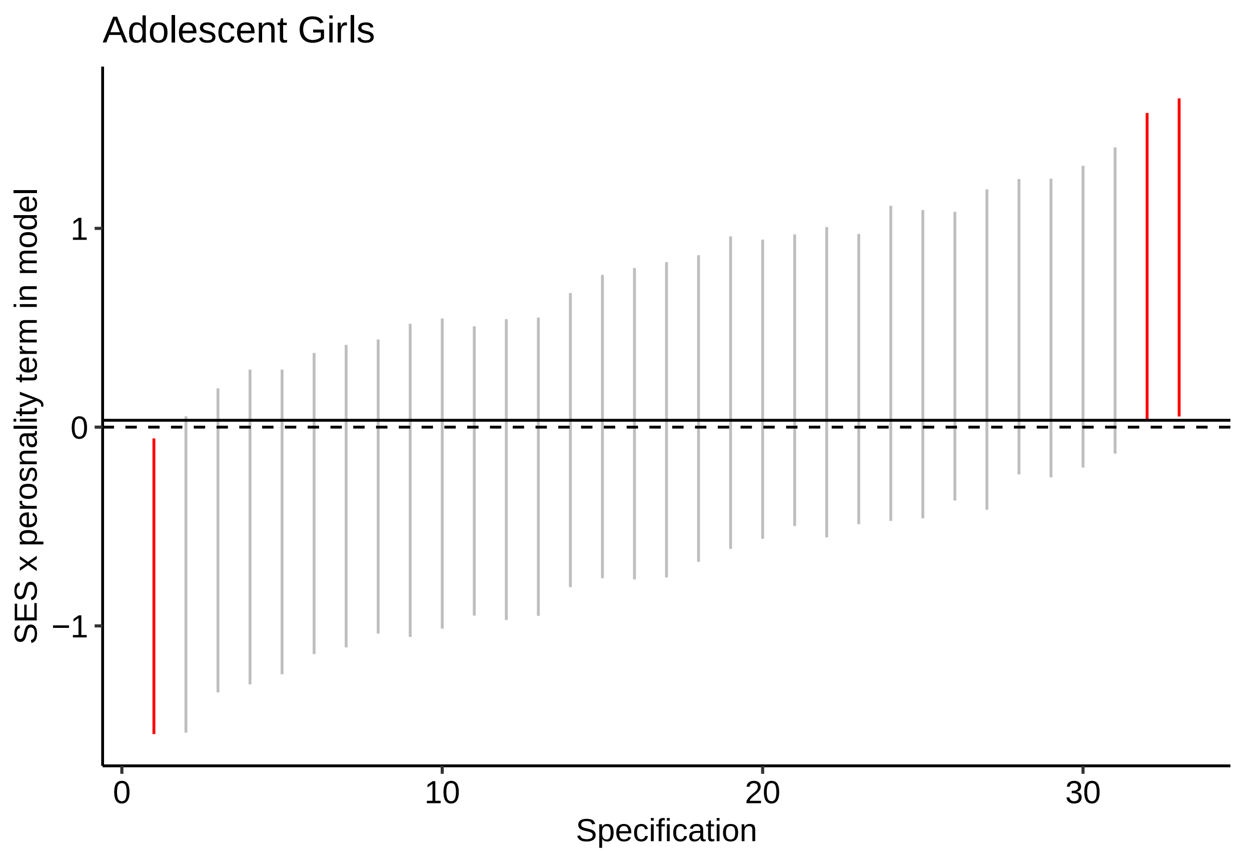
```
avg_female = female_reg %>%
  filter(grepl(":", term)) %>%
  summarize(mean = mean(estimate))
```

Parental socioeconomic status positively predicted greater likelihood of all non-normal categories (Underweight, Overweight, and Obese) compared to Normal among girls. Adolescent girls living in higher SES households were, on average, NA% less likely to be Underweight, NA% less likely, and 97% less likely to be Obese compared to low SES counterparts.

```r
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 Girls") +
  guides(color = F) +
  theme_pubr()
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```

```r
female_plot_2
```



3.2.2 Male

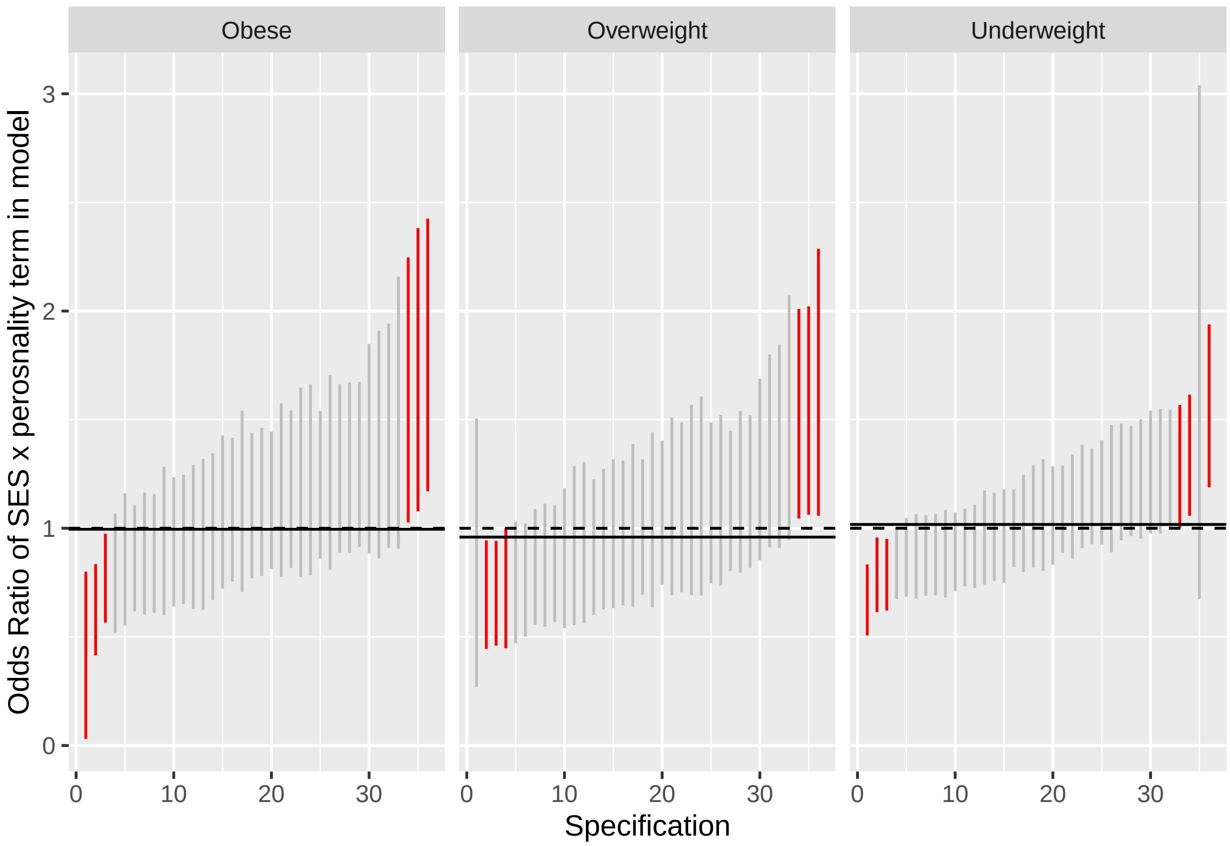
```
avg_male = male_reg %>%
  filter(grepl(":", term)) %>%
  summarize(mean = mean(estimate))
```

Parental socioeconomic status positively predicted greater likelihood of all non-normal categories (Underweight, Overweight, and Obese) compared to Normal among boys. Adolescent boys living in higher SES households were, on average, NA% more likely to be Underweight, NA% more likely, and -59% more likely to be Obese compared to low SES counterparts.

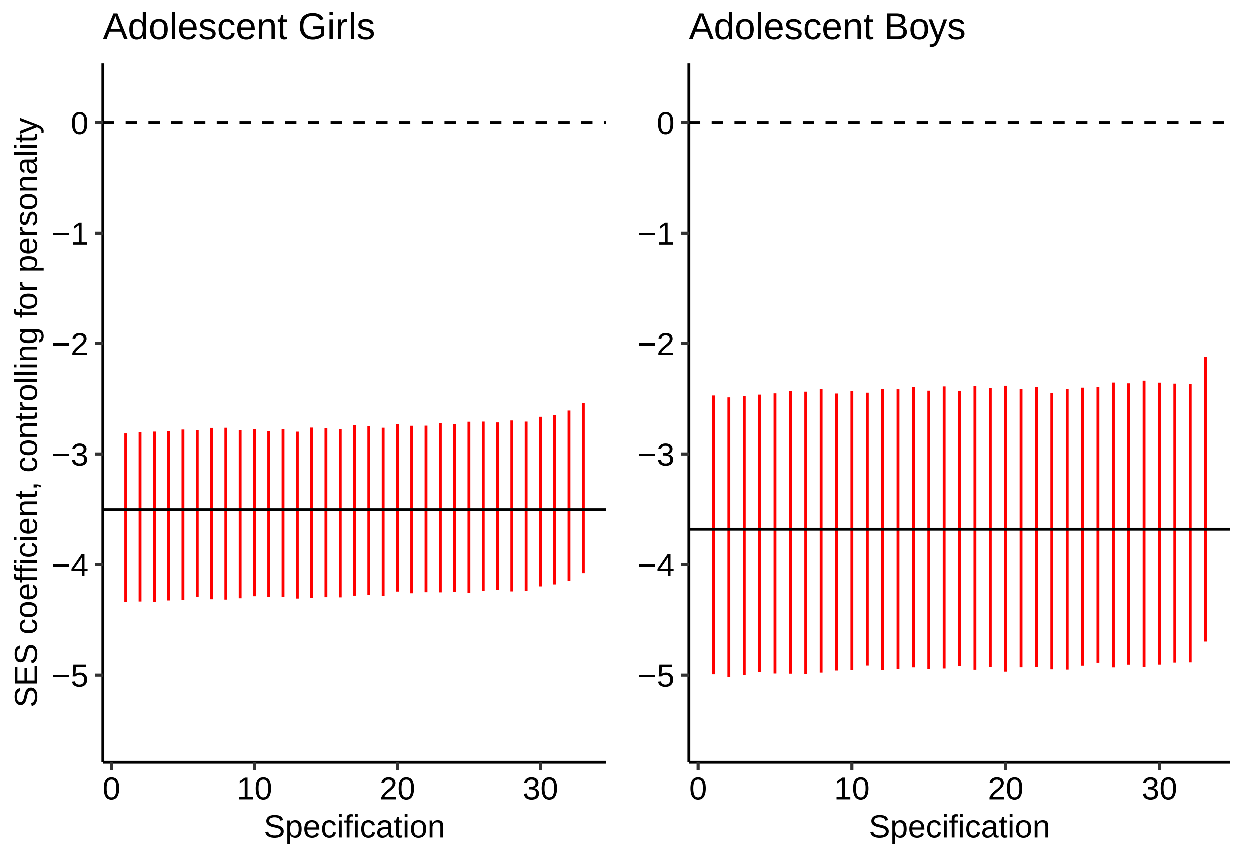
```
male_plot_2 = male_reg %>%
  filter(grepl(":", term)) %>%
  mutate(psig = ifelse(p.value < .05, "yes", "no")) %>%
  arrange(estimate) %>%
  mutate(spec = row_number()) %>%
  ggplot(aes(x = spec, y = conf.low)) +
  geom_segment(aes(xend = spec, yend = conf.high, color = psig)) +
  geom_hline(aes(yintercept = 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(-5.5, .25))+
  labs(x = "Specification",
       y = "Odds Ratio of SES x perosnality term in model") +
  guides(color = F) +
  theme_pubr()
```

```
## Warning: `guides(<scale> = FALSE)` is deprecated. Please use `guides(<scale> =
## "none")` instead.
```

```
male_plot_2
```




```
# both plots together ------------------------------------------------------

ggarrange(female_plot_1, male_plot_1)
```




```
ggsave(here("figures/SES_specification.jpeg"), width = 6, height = 4)
```

3.3 Code (Female)

```
# we run the models for men and women separately because R kept
# crashing when trying to run this whole script.

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 = .)))

female_plot = female_reg %>%
  mutate(cov = map(data, ~lm(BMI_p ~ trait_score + ses, data = .))) %>%
  mutate(int = map(data, ~lm(BMI_p ~ trait_score*ses, data = .)))


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))
```

3.4 Code (Male)

```
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 = .)))

male_plot = male_reg %>%
  mutate(cov = map(data, ~lm(BMI_p ~ trait_score + ses, data = .))) %>%
  mutate(int = map(data, ~lm(BMI_p ~ trait_score*ses, data = .)))


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))
```

4 Regression (Individual differences)

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


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, "]")) %>%
  dplyr::select(trait_name, model, term, b1_est, b2_conf, gender) %>%
  gather("key", "value", b1_est, b2_conf) %>%
  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) %>%
  mutate(
    trait_name = as.character(trait_name),
    trait_name = ifelse(!(row_number() %% 2),
                        NA_character_,
                        trait_name)) %>%
  dplyr::select(-key)
```

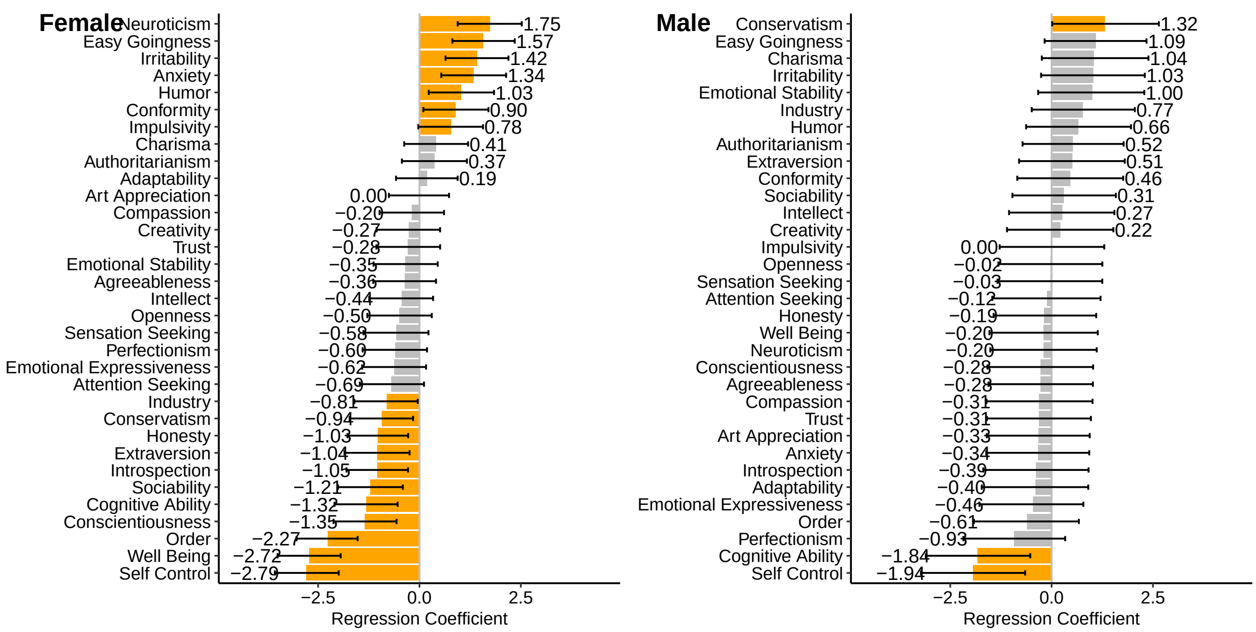
```
## Joining, by = c("trait_name", "model", "term", "estimate", "std.error", "statistic", "p.value", "conf

## 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", 3, 56) %>%
group_rows("SPI: 5 Factors", 57, 66)
```

	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*	-1.33*	0.09	-1.84*	-1.83*	-0.08
	[-2.10, -0.54]	[-2.11, -0.54]	[-0.68, 0.86]	[-3.10, -0.53]	[-3.10, -0.52]	[-1.37, 1.20]
SPI: 27 Factors						
Compassion	-0.20	-0.19	-0.38	-0.31	-0.34	0.44
	[-0.98, 0.61]	[-0.98, 0.61]	[-1.14, 0.37]	[-1.61, 1.01]	[-1.64, 0.98]	[-0.81, 1.65]
Irritability	1.42*	1.43*	0.24	1.03	1.03	0.29
	[0.64, 2.20]	[0.65, 2.20]	[-0.49, 0.97]	[-0.26, 2.30]	[-0.26, 2.31]	[-0.98, 1.60]
Sociability	-1.21*	-1.21*	0.33	0.31	0.39	1.22
	[-2.01, -0.41]	[-2.01, -0.41]	[-0.47, 1.11]	[-0.97, 1.58]	[-0.88, 1.69]	[-0.04, 2.46]
Well Being	-2.72*	-2.70*	0.81*	-0.20	-0.18	0.57
	[-3.50, -1.94]	[-3.48, -1.93]	[0.04, 1.58]	[-1.53, 1.14]	[-1.52, 1.15]	[-0.69, 1.78]
Sensation Seeking	-0.58	-0.56	0.63	-0.03	-0.03	-0.40
	[-1.38, 0.22]	[-1.37, 0.24]	[-0.13, 1.41]	[-1.33, 1.25]	[-1.35, 1.25]	[-1.68, 0.90]
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]
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]
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]
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]
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]
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]
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]
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]
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]
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]
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]
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]
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]
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]
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]
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]
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]
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]
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]
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]
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]
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]
SPI: 5 Factors						
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]
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]
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]
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]
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]



5 Logistic Regression (SES)

Multinomial logistic regression models were built that regressed BMI category onto parental socio-economic 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 ability – and tested each one independently in the model as an individual difference.

Models were estimated separately for men and women.

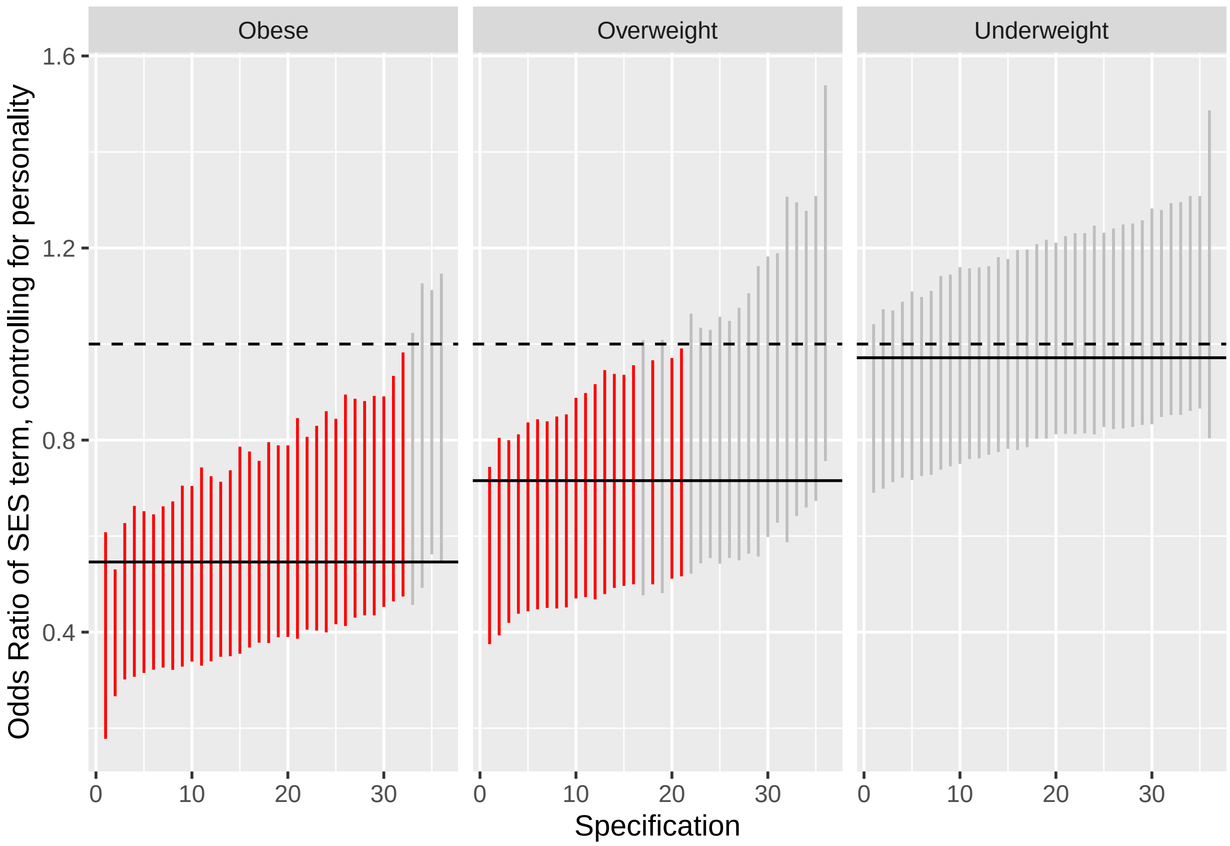
```
# end goal of wrangling is a data frame of data frames
# nested dataframes correspond to a single personality trait
# score refers to a participant's score on that trait
# we also standardize each of our variables within gender


sapa_male_trait = sapa_male %>%
  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()
```

5.1 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.

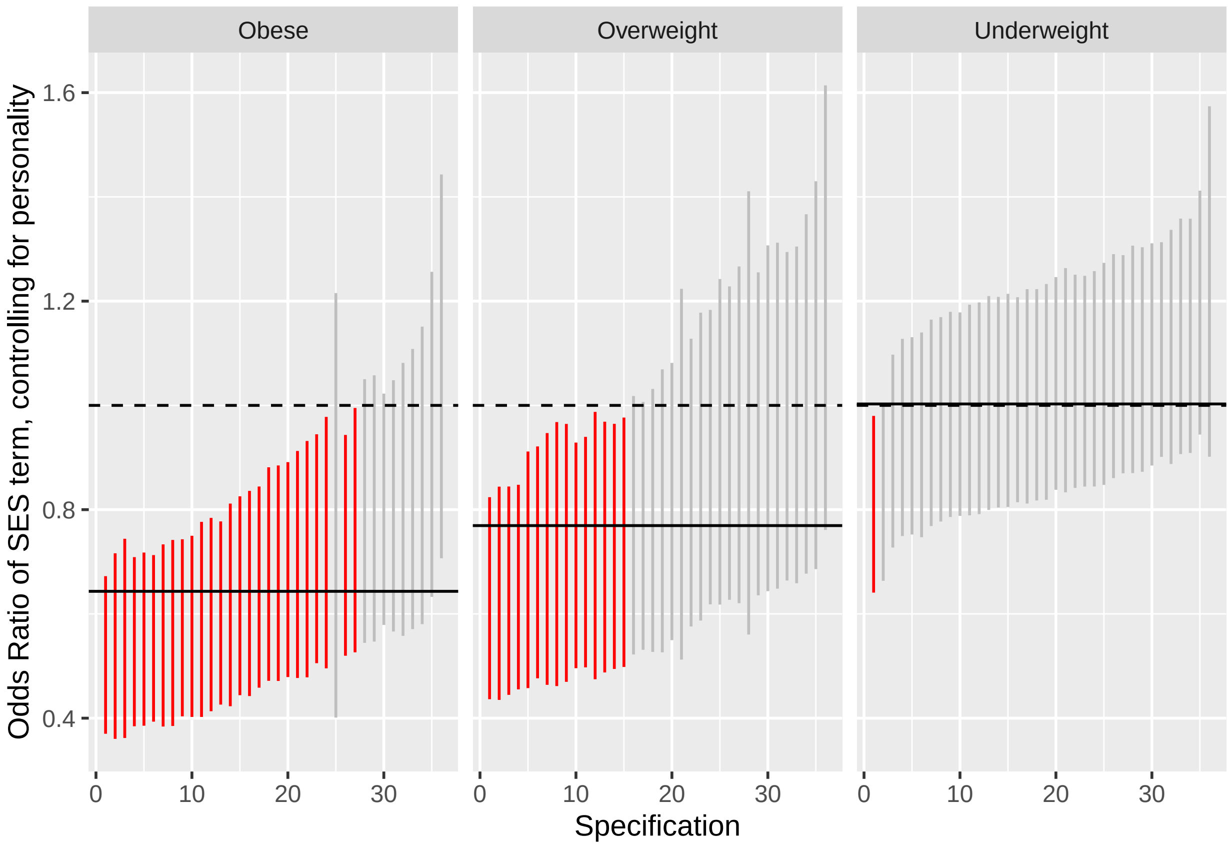
5.1.1 Female

Parental socioeconomic status positively predicted greater likelihood of all non-normal categories (Underweight, Overweight, and Obese) compared to Normal among girls. Adolescent girls 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.



5.1.2 Male

Parental socioeconomic status positively predicted greater likelihood of all non-normal categories (Underweight, Overweight, and Obese) compared to Normal among boys. Adolescent boys 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.

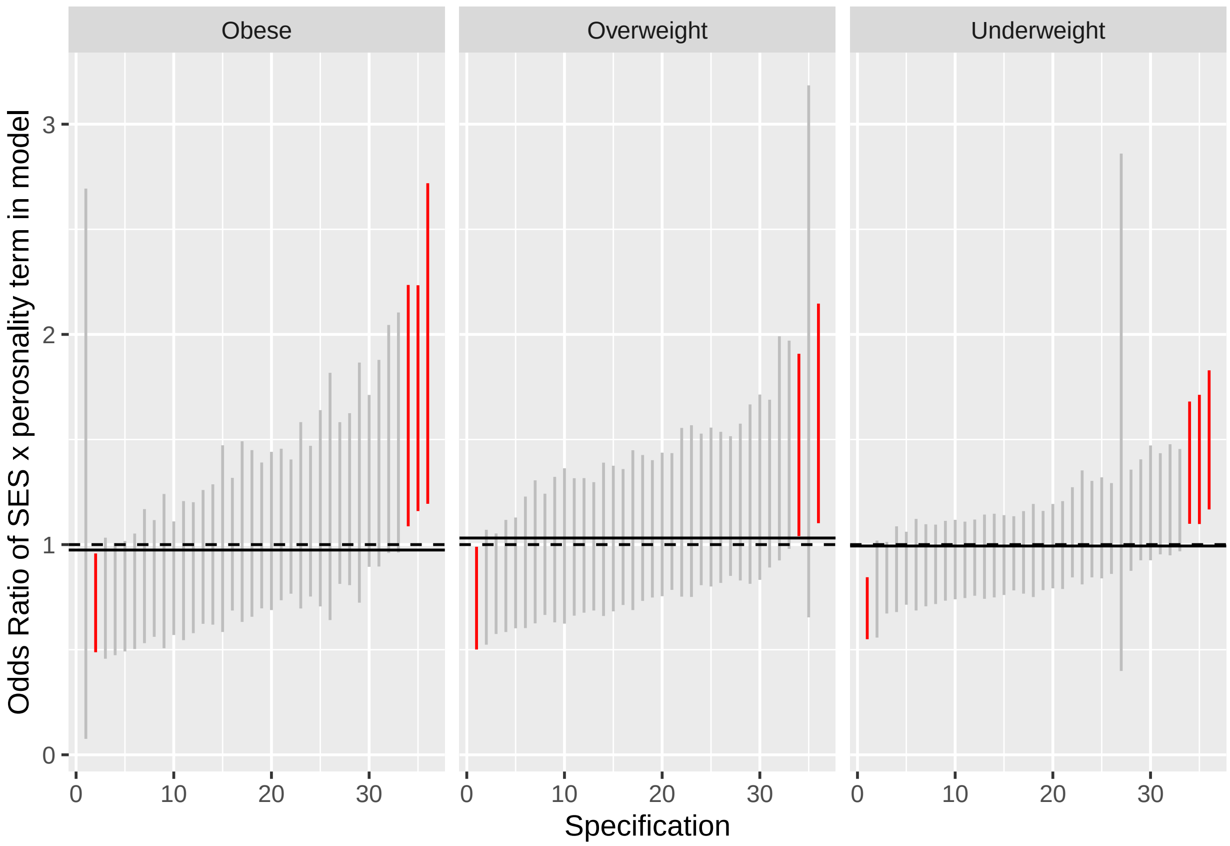


5.2 Interaction with personality

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.

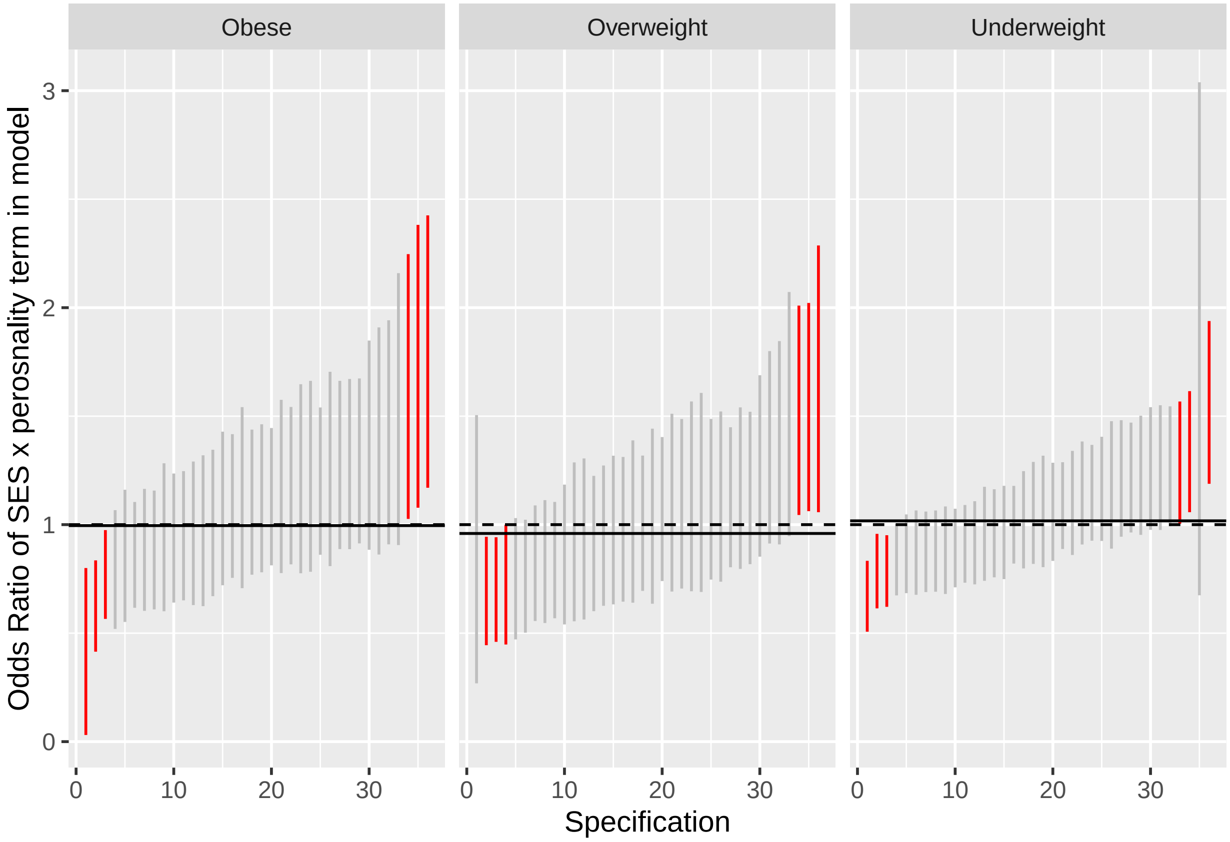
5.2.1 Female

Parental socioeconomic status positively predicted greater likelihood of all non-normal categories (Underweight, Overweight, and Obese) compared to Normal among girls. Adolescent girls living in higher SES households were, on average, 101% less likely to be Underweight, 97% less likely, and 103% less likely to be Obese compared to low SES counterparts.



5.2.2 Male

Parental socioeconomic status positively predicted greater likelihood of all non-normal categories (Underweight, Overweight, and Obese) compared to Normal among boys. Adolescent boys 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.



5.3 Code (Female)

```
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, ]),
```

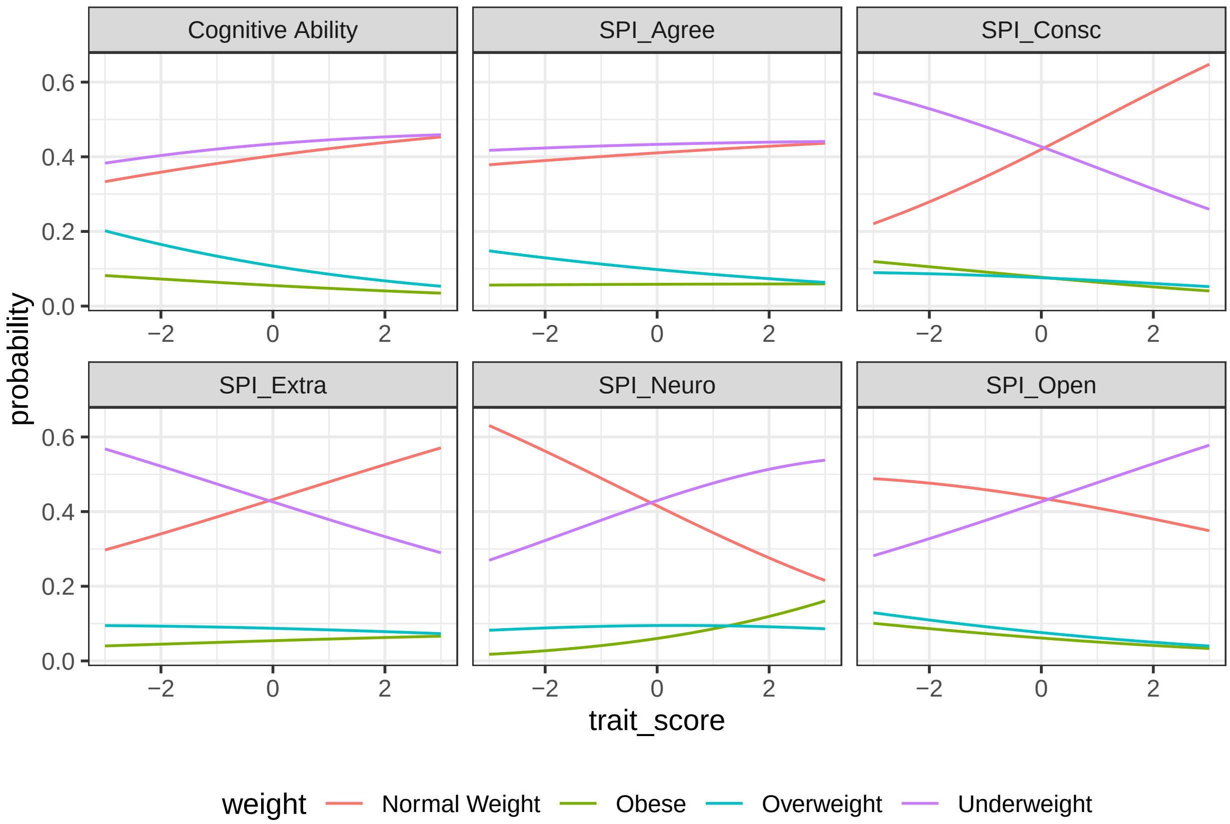
```
        #extract reference (true) BMI categories from test data
        test_reference = map(test_data, "BMI_c"),
        # predict categories from model output; na.pass puts NAs in any row with missing data
        predicted = map2(output, test_data, predict, na.action = "na.pass"),
        # calculate 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))
```

5.4 Code (Male)

```
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))
```

6 Logistic Regression (Individual differences)

6.1 Female



6.2 Male

Trait	Obese	Overweight	Underweight
Cognitive Ability	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]**

Trait	Obese	Overweight	Underweight
Cognitive Ability	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]

7 Model accuracy

```
packages = c("glmnet", "caret", "ggpubr", "knitr", "kableExtra")
lapply(packages, library, character.only = TRUE)
rm(packages)

load("data/cleaned.Rdata")
```

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 boys and adolescent girls

```
set.seed(060821)

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

```
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 Boys")
```

We extract the relevant information for a table.

```
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,
        digits = c(0,2,3,2,3)) %>%
  kable_styling() %>%
  add_header_above(c(" ", "Adolescent Boys" = 2, "Adolescent Girls" = 2))
```

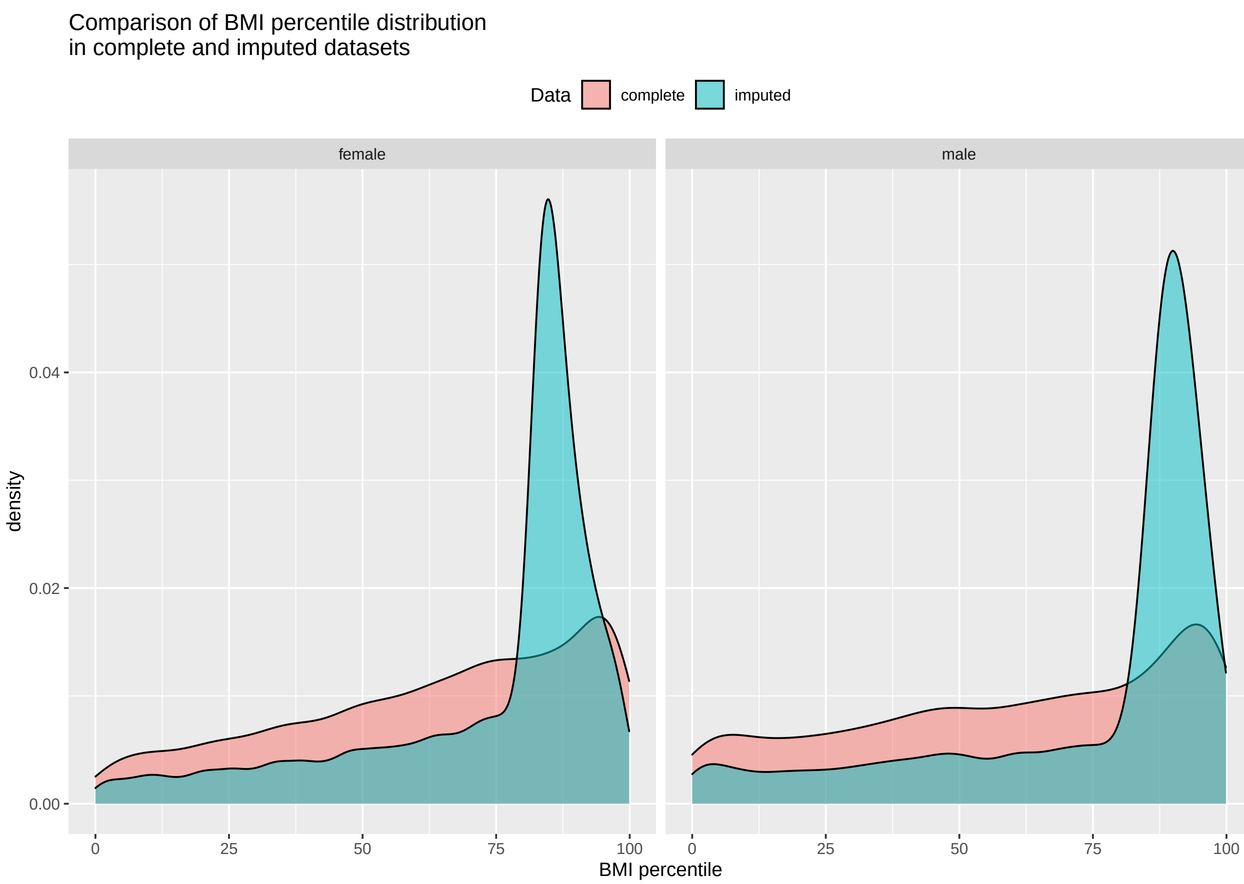
8 Sensitivity analysis: Missing data

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 girls in our

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

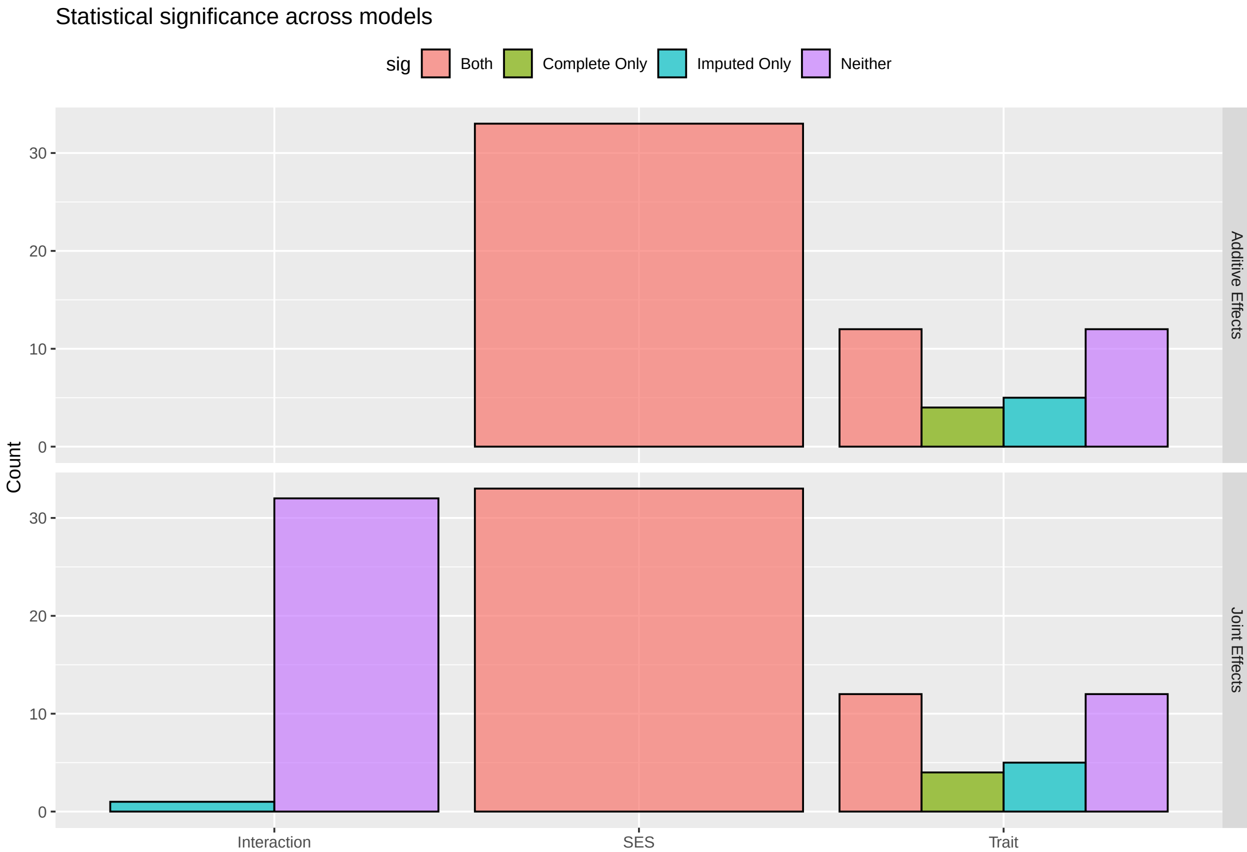
sample, we suspect these values are missing not at random (MNAR) and may impact the estimates here. To test for these effects, we imputed missing height and weight values using a principal components analysis approach, using only the other variables in the SAPA dataset that were not included in the analyses 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 girls and boys (see Figure **??**). 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.

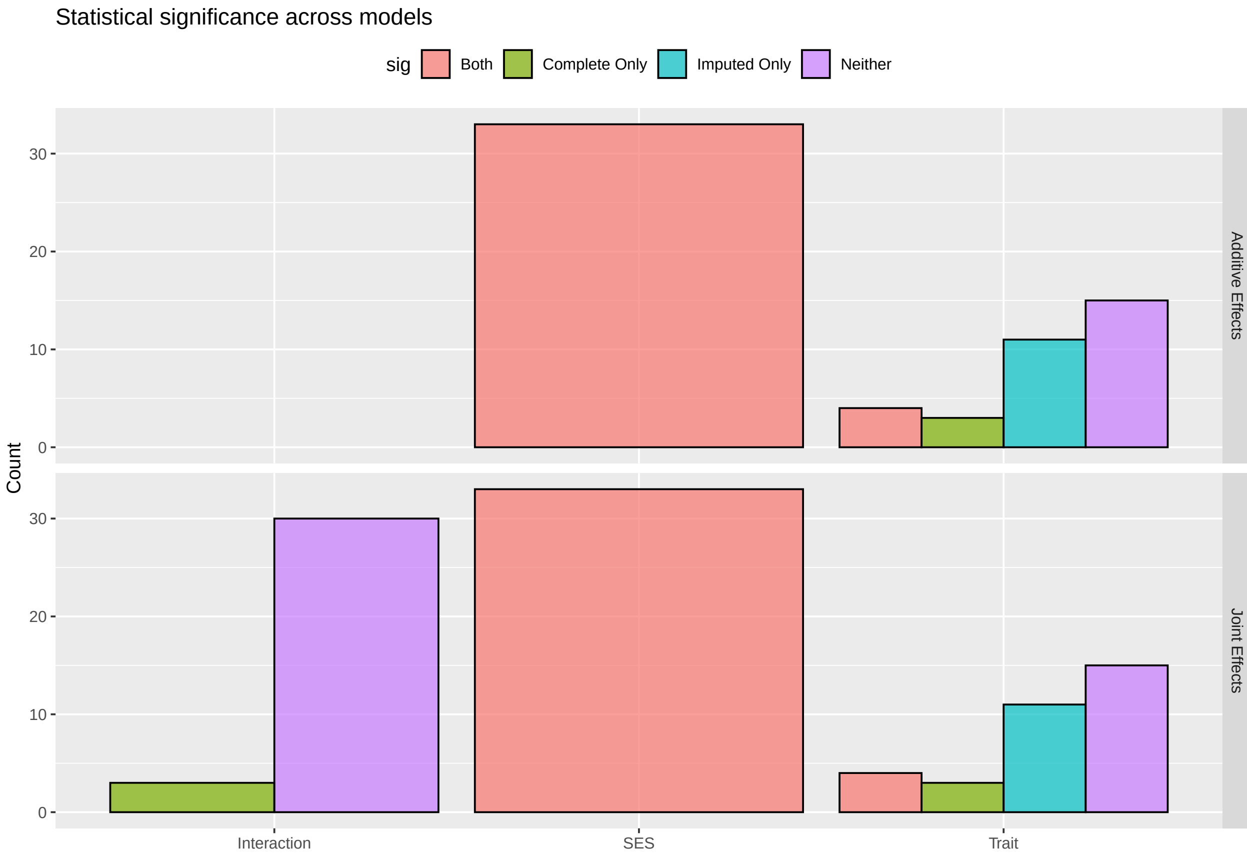


8.1 Female

We compared the significance of coefficients across models using the complete dataset and the imputed dataset, for adolescent girls (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.



8.2 Male



Statistical significance across models
sig
Both
Complete Only
Imputed Only
Neither
Count
30
20
10
0
Additive Effects
Joint Effects
Interaction
SES
Trait