Diet items in the Hawaii Personality and Health Cohort: Analysis Code and Results

```
rm(list=ls())
knitr::opts_chunk$set(echo = TRUE)
options(width=120)
```

1 Data preparation

1.1 Load data and packages

We load necessary packages into our workspace.

```
library(knitr)
library(haven)
library(foreign)
library(lsr)
library(psych)
library(stargazer)
library(sjPlot)
```

We then load the data files, which are saved as SPSS data files. Both the haven (Wickham Miller, 2019) and foreign (R Core Team, 2018) packages are used to load these data files.

```
demographics <- read_sav("../../data/ORI Data/demograph2016.sav")
age <- read_sav("../../data/ORI Data/cog_function_c1c2.sav")
Q5_personality <- read.spss("../../data/ORI Data/Original Q5_NEO.sav",
                            use.value.labels = F, to.data.frame = T)
Q1_diet <- read.spss("../../data/ORI Data/Q1.sav",
                     use.value.labels = F, to.data.frame = T)
Q7_health <- read_sav("../../data/ORI Data/Q7_labels2.sav")
child_clinic <- read_sav("../../data/ORI Data/childc1q1.sav")
```

1.2 Prepare variables of interest

We create new variables in these data sets. In Q7, for the variable of self-rated health we simply change the name of the variable. For Type 2 Diabetes, we recode the binary variable from 1 (Yes) and 2 (No) to 1 and 0, respectively. We also extract age at onset of Type 2 diabetes.

```
Q7_health$SelfRatedHealth <- Q7_health$q7srhlth
Q7_health$Type2Diabetes <- ifelse(Q7_health$q7diab3 == 1, 1, 0)
```

In the childhood personality and clinic data, we rename the clinic assessment of BMI.

```
child_clinic$BMI <- child_clinic$c1bmi
```

In the Q1 diet questionnaire, we set all the variable names to lower-case, to make the code easier to create and to read.

```
colnames(Q1_diet) <- tolower(colnames(Q1_diet))
```

1.3 Identify diet items

We then create a vector of the names of the diet items of interest. These items are named q1sea23, q1sea24, ..., q1sea46.

```
diet.items <- paste("q1sea",23:46, sep="")
```

We identify the location of these items in the Q1 data frame. We then extract the item text of those variables from the variable labels attribute of the data frame, which has been imported from the SPSS file.

```
loc.diet <- which(colnames(Q1_diet) %in% diet.items)
diet.labels <- attr(Q1_diet, "variable.labels")[loc.diet]
```

1.4 Merge data frames

We create our data set of interest by merging the data frames together by the ID numbers of participants. From the diet questionnaire, we only merge the diet items we have identified. From the child and clinic data, we only merge the BMI variable. We then remove the original data frames from our workspace.

```
ori.diet <- merge(demographics, Q5_personality, by.x="pchid", by.y="PCHID", all.x=T, all.y=T)
ori.diet <- merge(ori.diet, Q7_health, by="pchid", all.x=T, all.y=T)
ori.diet <- merge(ori.diet, Q1_diet[,c("pchid", diet.items)], by="pchid", all.x=T, all.y=T)
ori.diet <- merge(ori.diet, subset(child_clinic, select=c(pchid, BMI)),
                  by="pchid", all.x=T, all.y=T)
ori.diet <- merge(ori.diet, subset(age, select=c(pchid, age)), all.x=T)

colnames(ori.diet)[colnames(ori.diet) == "age"] = "age_clinic"

rm(list=c("demographics", "Q5_personality", "Q7_health", "child_clinic", "Q1_diet", "age"))
```

1.5 Create personality scores

Personality items were scored using the make.keys and scoreItems functions in the psych package (Revelle, 2019).

```
personality.list <- list(
  Extra = c("PQ2","PQ32","-PQ62","-PQ92",
            "PQ7","PQ37","-PQ67","-PQ97",
            "PQ12","PQ42","PQ72","-PQ102",
            "PQ17","PQ47","PQ77","-PQ107",
            "PQ22","PQ52","PQ82","PQ112",
            "PQ27","PQ57","PQ87","PQ117"),
  Agree = c("PQ4","PQ34","PQ64","-PQ94",
            "-PQ9","-PQ39","-PQ69","-PQ99",
            "PQ14","PQ44","-PQ74","-PQ104",
            "-PQ19","-PQ49","-PQ79","-PQ109",
```

```
             "-PQ24","-PQ54","-PQ84","-PQ114",
             "PQ29","PQ59","-PQ89","-PQ119"),
  Con = c("PQ5","PQ35","PQ65","PQ95",
           "PQ10","-PQ40","-PQ70","-PQ100",
           "PQ15","PQ45","-PQ75","-PQ105",
           "PQ20","PQ50","-PQ80","-PQ110",
           "PQ25","PQ55","-PQ85","-PQ115",
           "-PQ30","-PQ60","-PQ90","-PQ120"),
  Neur = c("PQ1","PQ31","PQ61","PQ91",
            "PQ6","PQ36","PQ66","-PQ96",
            "PQ11","PQ41","PQ71","-PQ101",
            "PQ16","PQ46","PQ76","-PQ106",
            "PQ21","-PQ51","-PQ81","-PQ111",
            "PQ26","PQ56","PQ86","-PQ116"),
  Open = c("PQ3","PQ33","PQ63","PQ93",
            "PQ8","PQ38","-PQ68","-PQ98",
            "PQ13","PQ43","-PQ73","-PQ103",
            "PQ18","-PQ48","-PQ78","-PQ108",
            "PQ23","-PQ53","-PQ83","-PQ113",
            "PQ28","PQ58","-PQ88","-PQ118"))

personality.keys <- make.keys(
  nvars=120,
  keys.list = personality.list,
  item.labels = colnames(ori.diet)[grepl("^PQ", colnames(ori.diet))])

personality.scores <- scoreItems(
  keys = personality.keys,
  items=ori.diet[,grepl("^PQ", colnames(ori.diet))],
  impute = "none")

personality.scores

## Call: scoreItems(keys = personality.keys, items = ori.diet[, grepl("^PQ",
##     colnames(ori.diet))], impute = "none")
##
## (Standardized) Alpha:
##       Extra Agree  Con Neur Open
## alpha  0.88  0.81 0.87 0.88 0.77
##
## Standard errors of unstandardized Alpha:
##        Extra  Agree   Con   Neur   Open
## ASE   0.0047 0.0066 0.005 0.0047 0.0079
##
## Standardized Alpha of observed scales:
##      Extra Agree  Con Neur Open
## [1,]  0.88  0.81 0.87 0.88 0.77
##
## Average item correlation:
##           Extra Agree  Con Neur Open
## average.r  0.24  0.15 0.22 0.23 0.12
##
```

```
## Median item correlation:
## Extra Agree   Con  Neur  Open
##  0.23  0.16  0.22  0.22  0.11
##
##  Guttman 6* reliability:
##         Extra Agree  Con Neur Open
## Lambda.6  0.93  0.88 0.92 0.92 0.85
##
## Signal/Noise based upon av.r :
##             Extra Agree Con Neur Open
## Signal/Noise   7.4   4.4 6.8  7.3  3.3
##
## Scale intercorrelations corrected for attenuation
##  raw correlations below the diagonal, alpha on the diagonal
##  corrected correlations above the diagonal:
##
## Note that these are the correlations of the complete scales based on the correlation matrix,
##  not the observed scales based on the raw items.
##       Extra Agree    Con  Neur  Open
## Extra  0.88  0.25  0.427 -0.53  0.44
## Agree  0.22  0.81  0.463 -0.41  0.31
## Con    0.37  0.39  0.872 -0.59  0.12
## Neur  -0.46 -0.35 -0.514  0.88 -0.18
## Open   0.36  0.24  0.099 -0.15  0.77
##
##  In order to see the item by scale loadings and frequency counts of the data
##  print with the short option = FALSE
```

These scores are then added to the data frame.

```
ori.diet <- cbind(ori.diet, personality.scores$scores)
```

1.6 Socio-economic status composite

We first test the correlation between maximum lifetime income and highest grade completed in education to determine whether these may be averaged together to create a single indicator of socio-economic status.

```
cor(ori.diet[,c("q7maxinc", "highgrd")], use="pairwise")

##           q7maxinc   highgrd
## q7maxinc 1.0000000 0.3636618
## highgrd  0.3636618 1.0000000
```

The correlation is 1, 0.36, 0.36, 1, which is sufficient evidence to create a composite, which we do below.

```
ori.diet$SES <- as.numeric(pca(ori.diet[,c("q7maxinc", "highgrd")])$scores)
```

1.7 Age calculation

We the participant's age at at the diet assesment by subtracting their birth year from the year of assessment. Birth years are recorded as the last two digits in the year (all participants were born between 1900 and 1999), so year of assessment was measured as years from 1900.

```
ori.diet$age_diet <- 99-ori.diet$birthyear
```

We remove participants who do not have scores for at least one of the big five personality scores and at least one of the covariates. We will repeat this process at a later step, although this ensures that the structure of the diet items remains relatively consistent throughout the analyses.

```
B5_scores <- c("Extra","Agree","Con","Neur","Open")
cov <- c("gender","age_diet")
ori.diet <- ori.diet[!is.na(rowSums(ori.diet[,B5_scores])), ]
ori.diet <- ori.diet[!is.na(rowSums(ori.diet[,cov])), ]
```

1.8 Create data frame of diet items, labels and text

We create a data frame of diet item labels and item text to reference after the factor analysis stage. This will allow for easier interpretation of the factors.

```
diet.items <- ori.diet[,diet.items]

df.dietItems <- data.frame(item = colnames(diet.items), stringsAsFactors = F)
df.dietItems$question <- diet.labels

colnames(diet.items) <- df.dietItems$question
```

2 Partial correlation function

I ammend the corr.test function available in the psych package by inserting code to calculate the residuls of the original variables after controlling for specified covariates. This function will be used to calculate the partial correlations of the diet factors with variables of interest.

```r
partial.corr.test <- function (x, y = NULL, m,
                               use = "pairwise", method = "pearson",
                               adjust = "holm", alpha = 0.05, ci = TRUE)
{
  cl <- match.call()

  #NEW CODE BEGINS HERE
  covariates <- colnames(m)
  covariate.fun <- paste(covariates, collapse = " + ")

  new.x <- matrix(data = as.numeric(NA),
                  nrow = nrow(x), ncol=ncol(x))

  for(i in 1:ncol(x)){
    outcomes <- colnames(x)[i]
    prediction.eq <- as.formula(paste(outcomes, covariate.fun, sep = " ~ "))
    model <- lm(prediction.eq, data = cbind(x, m))
    missing <- attr(model$model, "na.action")
    if(is.null(missing)) new.x[,i] <- model$residuals
      else new.x[-missing,i] <- model$residuals
  }
  new.x <- as.data.frame(new.x)
  colnames(new.x) <- colnames(x)

  new.y <- matrix(data = as.numeric(NA),
                  nrow = nrow(y), ncol=ncol(y))

  for(i in 1:ncol(y)){
    outcomes <- colnames(y)[i]
    prediction.eq <- as.formula(paste(outcomes, covariate.fun, sep = " ~ "))
    model <- lm(prediction.eq, data = cbind(y, m))
    missing <- attr(model$model, "na.action")
    if(is.null(missing)) new.y[,i] <- model$residuals else new.y[-missing,i] <- model$residuals
  }
  new.y <- as.data.frame(new.y)
  colnames(new.y) <- colnames(y)

  x <- new.x
  y <- new.y

  # NEW CODE ENDS HERE

  if (is.null(y)) {
    r <- cor(x, use = use, method = method)
    sym <- TRUE
    n <- t(!is.na(x)) %*% (!is.na(x))
```

```r
  }
  else {
    r <- cor(x, y, use = use, method = method)
    sym = FALSE
    n <- t(!is.na(x)) %*% (!is.na(y))
  }
  if ((use == "complete") | (min(n) == max(n)))
    n <- min(n)
  t <- (r * sqrt(n - 2))/sqrt(1 - r^2)
  p <- 2 * (1 - pt(abs(t), (n - 2)))
  se <- sqrt((1 - r * r)/(n - 2))
  nvar <- ncol(r)
  p[p > 1] <- 1
  if (adjust != "none") {
    if (is.null(y)) {
      lp <- upper.tri(p)
      pa <- p[lp]
      pa <- p.adjust(pa, adjust)
      p[upper.tri(p, diag = FALSE)] <- pa
    }
    else {
      p[] <- p.adjust(p, adjust)
    }
  }
  z <- fisherz(r[lower.tri(r)])
  if (ci) {
    if (min(n) < 4) {
      warning("Number of subjects must be greater than 3 to find confidence intervals.")
    }
    alpha <- 1 - alpha/2
    dif <- qnorm(alpha)
    if (sym) {
      if (is.matrix(n)) {
        se <- 1/sqrt(n[lower.tri(n)] - 3)
      }
      else {
        se <- 1/sqrt(n - 3)
      }
      lower <- fisherz2r(z - dif * se)
      upper <- fisherz2r(z + dif * se)
      ci <- data.frame(lower = lower, r = r[lower.tri(r)],
                       upper = upper, p = p[lower.tri(p)])
      cnR <- abbreviate(colnames(r), minlength = 5)
      k <- 1
      for (i in 1:(nvar - 1)) {
        for (j in (i + 1):nvar) {
          rownames(ci)[k] <- paste(cnR[i], cnR[j], sep = "-")
          k <- k + 1
        }
      }
    }
```

```
    else {
      z <- fisherz(r)
      se <- 1/sqrt(n - 3)
      lower <- as.vector(fisherz2r(z - dif * se))
      upper <- as.vector(fisherz2r(z + dif * se))
      ci <- data.frame(lower = lower, r = as.vector(r),
                       upper = upper, p = as.vector(p))
      cnR <- abbreviate(rownames(r), minlength = 5)
      cnC <- abbreviate(colnames(r), minlength = 5)
      k <- 1
      for (i in 1:ncol(y)) {
        for (j in 1:ncol(x)) {
          rownames(ci)[k] <- paste(cnR[j], cnC[i], sep = "-")
          k <- k + 1
        }
      }
    }
  }
  else {
    ci <- NULL
  }
  result <- list(r = r, n = n, t = t, p = p, se = se, adjust = adjust,
                 sym = sym, ci = ci, Call = cl)
  class(result) <- c("psych", "corr.test")
  return(result)
}
```

3 Diet items factor analysis

3.1 How many factors

To determine the number of factors to extract from the diet items, we use the nfactors function from the psych package (Revelle, 2019). The function returns both a plot of fit statistics at many numbers of factors as well as a table of fit statistics. We examine these plots and tables to generate potential solutions.

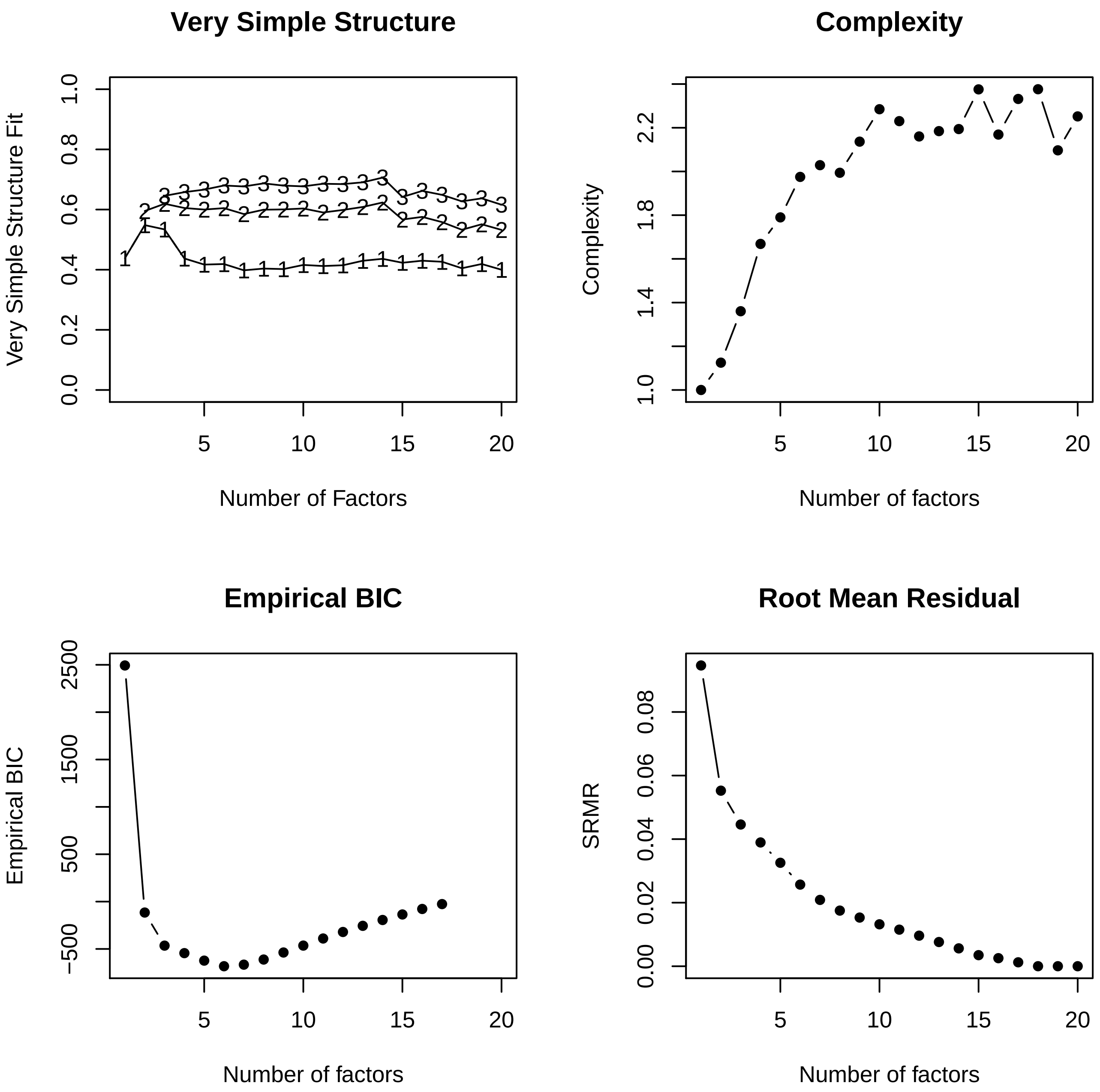
We examine the plot for large jumps from one number of factors to the next. Jumps to and from variables can indicate solutions that are superior to the number of factors preeceeding them, although whether the jump should begin or end with the solution at hand depends upon the fit statistic. In general, the fit statistics plotted should as low as possible, so whichever value is at the low point of the jump is the better solution.

We also examine the tables for points where solutions jump and/or meet agreed-upon cutoffs. For example, we might look for where the RMSEA drops below .10.

Based upon our examinations, we examine the **one-factor** solution (because RMSEA is below .10), the **two-factor** solution (based on jumps down to two at (because the empirical BIC and root mean residual jump down to two) and **three-factor** (because complexity jumps from three to four).

We will extract each of these solutions using factor analysis with an oblimin rotation. We chose factor analysis for two reasons. First, we hope to estimate a structure that is more generalizable, and factor analysis accounts for less variability than prinicipal components analysis. Second, the lower variability in the factors provides a more conservative test of relationships of diet factors with other outcomes. We chose an oblimin rotation as we don't believe eating certain categories or groups of food should be independent of eating other groups of food.

```
nfactors(diet.items)
```


Very Simple Structure
Very Simple Structure Fit
0.0
0.2
0.4
0.6
0.8
1.0
5
10
15
20
Number of Factors
Complexity
Complexity
1.0
1.4
1.8
2.2
5
10
15
20
Number of factors
Empirical BIC
Empirical BIC
−500
500
1500
2500
5
10
15
20
Number of factors
Root Mean Residual
SRMR
0.00
0.02
0.04
0.06
0.08
5
10
15
20
Number of factors

```
##
## Number of factors
## Call: vss(x = x, n = n, rotate = rotate, diagonal = diagonal, fm = fm,
##     n.obs = n.obs, plot = FALSE, title = title, use = use, cor = cor)
## VSS complexity 1 achieves a maximimum of 0.55  with  2  factors
## VSS complexity 2 achieves a maximimum of 0.62  with  14  factors
## The Velicer MAP achieves a minimum of 0.01  with  2  factors
## Empirical BIC achieves a minimum of  -682.39  with  6  factors
## Sample Size adjusted BIC achieves a minimum of  -229.82  with  8  factors
##
## Statistics by number of factors
##    vss1 vss2   map dof   chisq     prob sqresid  fit  RMSEA  BIC SABIC complex  eChisq    SRMR eCRMS eBIC
## 1  0.44 0.00 0.014 252 2.1e+03 3.8e-281    22.8 0.44 0.0927  363  1164     1.0 4.2e+03 9.5e-02 0.099 2493
## 2  0.55 0.60 0.010 229 1.1e+03 2.9e-111    16.4 0.60 0.0669 -459   268     1.1 1.4e+03 5.5e-02 0.061 -116
## 3  0.53 0.62 0.011 207 8.4e+02  1.6e-77    14.4 0.65 0.0607 -553   104     1.4 9.3e+02 4.5e-02 0.051 -465
## 4  0.44 0.61 0.014 186 6.9e+02  1.7e-58    13.2 0.67 0.0569 -568    23     1.7 7.1e+02 3.9e-02 0.047 -544
## 5  0.42 0.60 0.016 166 5.1e+02  5.5e-37    12.2 0.70 0.0502 -607   -80     1.8 5.0e+02 3.3e-02 0.042 -623
## 6  0.42 0.60 0.018 147 3.4e+02  1.8e-17    11.1 0.73 0.0400 -650  -183     2.0 3.1e+02 2.6e-02 0.035 -682
## 7  0.40 0.58 0.021 129 2.5e+02  1.5e-09    10.3 0.74 0.0336 -622  -212     2.0 2.0e+02 2.1e-02 0.031 -666
## 8  0.40 0.60 0.026 112 1.7e+02  3.6e-04     9.7 0.76 0.0253 -585  -230     2.0 1.4e+02 1.8e-02 0.027 -612
## 9  0.40 0.60 0.031  96 1.3e+02  6.7e-03     9.2 0.77 0.0222 -514  -209     2.1 1.1e+02 1.5e-02 0.026 -538
## 10 0.42 0.60 0.036  81 1.1e+02  1.0e-02     8.7 0.79 0.0225 -433  -175     2.3 8.2e+01 1.3e-02 0.024 -465
## 11 0.41 0.59 0.043  67 8.7e+01  5.0e-02     8.2 0.80 0.0196 -365  -152     2.2 6.2e+01 1.2e-02 0.023 -390
## 12 0.41 0.60 0.052  54 5.7e+01  3.6e-01     7.8 0.81 0.0098 -307  -135     2.2 4.3e+01 9.6e-03 0.022 -321
## 13 0.43 0.61 0.062  42 3.9e+01  6.2e-01     7.2 0.82 0.0000 -245  -111     2.2 2.7e+01 7.6e-03 0.020 -256
## 14 0.44 0.62 0.075  31 2.3e+01  8.3e-01     6.5 0.84 0.0000 -186   -87     2.2 1.5e+01 5.6e-03 0.017 -194
## 15 0.42 0.57 0.088  21 1.1e+01  9.7e-01     6.1 0.85 0.0000 -131   -64     2.4 5.7e+00 3.5e-03 0.013 -136
## 16 0.43 0.58 0.105  12 6.3e+00  9.0e-01     5.8 0.86 0.0000  -75   -36     2.2 3.0e+00 2.5e-03 0.012  -78
## 17 0.43 0.56 0.127   4 1.3e+00  8.5e-01     5.5 0.86 0.0000  -26   -13     2.3 7.2e-01 1.2e-03 0.010  -26
## 18 0.40 0.53 0.153  -3 7.4e-05       NA     5.9 0.86     NA   NA    NA     2.4 3.5e-05 8.6e-06    NA   NA
## 19 0.42 0.55 0.187  -9 1.5e-05       NA     5.8 0.86     NA   NA    NA     2.1 6.9e-06 3.8e-06    NA   NA
## 20 0.40 0.53 0.241 -14 9.1e-06       NA     6.0 0.85     NA   NA    NA     2.3 4.5e-06 3.1e-06    NA   NA
```

3.2 One-factor solution

```
one.factor <- fa(diet.items)
print(fa.sort(one.factor), cut = .2)
```

```
## Factor Analysis using method =  minres
## Call: fa(r = diet.items)
## Standardized loadings (pattern matrix) based upon correlation matrix
##                                      MR1      h2   u2 com
## HAMBURGERS/CHEESEBURGERS            0.66 0.43256 0.57   1
## FRENCH FRIES                        0.63 0.39441 0.61   1
## SAUSAGE/BACON/HOT DOGS              0.61 0.36673 0.63   1
## SPAM/HAM/LUNCHEON MEATS             0.51 0.26124 0.74   1
## DOUGHNUTS/PASTRY..                  0.48 0.22674 0.77   1
## STEAK/ROAST                         0.48 0.22574 0.77   1
## POTATO/CORN CHIPS                   0.47 0.22232 0.78   1
## ICE CREAM                           0.46 0.21578 0.78   1
## MAYONNAISE/REGULAR SALAD DRESSINGS  0.36 0.12656 0.87   1
## SKIM/1% FAT MILK                   -0.34 0.11309 0.89   1
## MACARONI,POTATO..SALAD W/MAYO       0.34 0.11274 0.89   1
## PIZZA                               0.33 0.11199 0.89   1
## CHOW MEIN                           0.33 0.11120 0.89   1
## SPAGHETTI W/MEAT OR CREAM SAUCE     0.32 0.10021 0.90   1
## BUTTER/MARGARINE/PEANUT BUTTER      0.31 0.09719 0.90   1
## RICE                                0.29 0.08670 0.91   1
## HIGH FIBER CEREALS                 -0.27 0.07173 0.93   1
## FRUIT                              -0.25 0.06059 0.94   1
## VEGETABLES                         -0.25 0.06018 0.94   1
## DARK WHOLE GRAIN BREADS            -0.24 0.05601 0.94   1
## 2% FAT/WHOLE MILK                   0.21 0.04519 0.95   1
## POI                                      0.01742 0.98   1
## TOFU                                     0.00645 0.99   1
## CHICKEN/FISH                             0.00002 1.00   1
##
##                 MR1
## SS loadings    3.52
## Proportion Var 0.15
##
## Mean item complexity =  1
```

```
## Test of the hypothesis that 1 factor is sufficient.
##
## The degrees of freedom for the null model are  276  and the objective function was  4.78 with Chi Square of  4010.45
## The degrees of freedom for the model are 252  and the objective function was  2.46
##
## The root mean square of the residuals (RMSR) is  0.09
## The df corrected root mean square of the residuals is  0.1
##
## The harmonic number of observations is  703 with the empirical chi square  3474.54  with prob <  0
## The total number of observations was  848  with Likelihood Chi Square =  2062.52  with prob <  3.8e-281
##
## Tucker Lewis Index of factoring reliability =  0.469
## RMSEA index =  0.093  and the 90 % confidence intervals are  0.088 0.096
## BIC =  363.32
## Fit based upon off diagonal values = 0.7
## Measures of factor score adequacy
##                                                    MR1
## Correlation of (regression) scores with factors   0.91
## Multiple R square of scores with factors          0.83
## Minimum correlation of possible factor scores     0.65
```

3.3 Two-factor solution

```
two.factor <- fa(diet.items, nfactors = 2, rotate="oblimin")
print(fa.sort(two.factor), cut = .2)
```

```
## Factor Analysis using method =  minres
## Call: fa(r = diet.items, nfactors = 2, rotate = "oblimin")
## Standardized loadings (pattern matrix) based upon correlation matrix
##                                       MR1   MR2    h2   u2 com
## HAMBURGERS/CHEESEBURGERS            0.62       0.425 0.57 1.1
## FRENCH FRIES                        0.62       0.397 0.60 1.0
## SAUSAGE/BACON/HOT DOGS              0.57       0.362 0.64 1.1
## DOUGHNUTS/PASTRY..                  0.53       0.274 0.73 1.1
## POTATO/CORN CHIPS                   0.51       0.250 0.75 1.0
## SPAM/HAM/LUNCHEON MEATS             0.50       0.262 0.74 1.0
## STEAK/ROAST                         0.49       0.236 0.76 1.0
## ICE CREAM                           0.48       0.227 0.77 1.0
## MAYONNAISE/REGULAR SALAD DRESSINGS  0.44       0.195 0.81 1.3
## BUTTER/MARGARINE/PEANUT BUTTER      0.42  0.24 0.198 0.80 1.6
## PIZZA                               0.35       0.120 0.88 1.0
## MACARONI,POTATO..SALAD W/MAYO       0.35       0.120 0.88 1.0
## SPAGHETTI W/MEAT OR CREAM SAUCE     0.35       0.116 0.88 1.1
## CHOW MEIN                           0.30       0.111 0.89 1.3
## RICE                                0.28       0.086 0.91 1.0
## 2% FAT/WHOLE MILK                   0.24       0.058 0.94 1.1
## POI                                            0.030 0.97 1.4
## FRUIT                                     0.77 0.586 0.41 1.0
## VEGETABLES                                0.71 0.503 0.50 1.0
## HIGH FIBER CEREALS                        0.58 0.354 0.65 1.0
## DARK WHOLE GRAIN BREADS                   0.51 0.273 0.73 1.0
## SKIM/1% FAT MILK                          0.39 0.219 0.78 1.5
## CHICKEN/FISH                              0.26 0.068 0.93 1.3
## TOFU                                      0.25 0.062 0.94 1.0
##
##                         MR1  MR2
## SS loadings           3.40 2.14
## Proportion Var        0.14 0.09
## Cumulative Var        0.14 0.23
## Proportion Explained  0.61 0.39
```

```
## Cumulative Proportion 0.61 1.00
##
##  With factor correlations of
##       MR1   MR2
## MR1  1.00 -0.19
## MR2 -0.19  1.00
##
## Mean item complexity =  1.1
## Test of the hypothesis that 2 factors are sufficient.
##
## The degrees of freedom for the null model are  276  and the objective function was  4.78 with Chi Square of  4010.45
## The degrees of freedom for the model are 229  and the objective function was  1.3
##
## The root mean square of the residuals (RMSR) is  0.06
## The df corrected root mean square of the residuals is  0.06
##
## The harmonic number of observations is  703 with the empirical chi square  1184.65  with prob <  1.4e-128
## The total number of observations was  848  with Likelihood Chi Square =  1085.05  with prob <  2.9e-111
##
## Tucker Lewis Index of factoring reliability =  0.723
## RMSEA index =  0.067  and the 90 % confidence intervals are  0.062 0.07
## BIC =  -459.07
## Fit based upon off diagonal values = 0.9
## Measures of factor score adequacy
##                                                    MR1  MR2
## Correlation of (regression) scores with factors   0.91 0.89
## Multiple R square of scores with factors          0.83 0.80
## Minimum correlation of possible factor scores     0.65 0.59
```

3.4 Three-factor solution

```
three.factor <- fa(diet.items, nfactors = 3, , rotate="oblimin")
print(fa.sort(three.factor), cut = .2)
```

```
## Factor Analysis using method =  minres
## Call: fa(r = diet.items, nfactors = 3, rotate = "oblimin")
## Standardized loadings (pattern matrix) based upon correlation matrix
##                                      MR1   MR2   MR3    h2   u2 com
## HAMBURGERS/CHEESEBURGERS            0.62             0.424 0.58 1.1
## FRENCH FRIES                        0.61             0.394 0.61 1.0
## SAUSAGE/BACON/HOT DOGS              0.58             0.377 0.62 1.1
## DOUGHNUTS/PASTRY..                  0.53             0.272 0.73 1.1
## POTATO/CORN CHIPS                   0.50             0.251 0.75 1.1
## SPAM/HAM/LUNCHEON MEATS             0.50             0.260 0.74 1.0
## STEAK/ROAST                         0.49             0.243 0.76 1.0
## ICE CREAM                           0.48             0.227 0.77 1.0
## BUTTER/MARGARINE/PEANUT BUTTER      0.47  0.30 -0.33 0.373 0.63 2.5
## MAYONNAISE/REGULAR SALAD DRESSINGS  0.44             0.197 0.80 1.3
## PIZZA                               0.34             0.121 0.88 1.1
## MACARONI,POTATO..SALAD W/MAYO       0.34             0.120 0.88 1.1
## SPAGHETTI W/MEAT OR CREAM SAUCE     0.33             0.131 0.87 1.4
## 2% FAT/WHOLE MILK                   0.25             0.072 0.93 1.4
## FRUIT                                     0.75       0.571 0.43 1.0
## VEGETABLES                                0.69       0.499 0.50 1.0
## HIGH FIBER CEREALS                        0.60       0.371 0.63 1.1
## DARK WHOLE GRAIN BREADS                   0.51       0.275 0.72 1.0
## SKIM/1% FAT MILK                          0.41       0.229 0.77 1.5
## CHICKEN/FISH                              0.23       0.098 0.90 2.2
## TOFU                                      0.21  0.63 0.455 0.55 1.2
## RICE                                0.26        0.39 0.234 0.77 1.9
## CHOW MEIN                           0.28        0.30 0.197 0.80 2.5
## POI                                             0.21 0.066 0.93 2.0
##
##                        MR1  MR2  MR3
## SS loadings           3.39 2.16 0.91
## Proportion Var        0.14 0.09 0.04
## Cumulative Var        0.14 0.23 0.27
## Proportion Explained  0.52 0.33 0.14
```

```
## Cumulative Proportion 0.52 0.86 1.00
##
##  With factor correlations of
##       MR1   MR2   MR3
## MR1  1.00 -0.16 -0.02
## MR2 -0.16  1.00  0.06
## MR3 -0.02  0.06  1.00
##
## Mean item complexity =  1.4
## Test of the hypothesis that 3 factors are sufficient.
##
## The degrees of freedom for the null model are  276  and the objective function was  4.78 with Chi Square of  4010.45
## The degrees of freedom for the model are 207  and the objective function was  1.01
##
## The root mean square of the residuals (RMSR) is  0.04
## The df corrected root mean square of the residuals is  0.05
##
## The harmonic number of observations is  703 with the empirical chi square  771.25  with prob <  5.7e-66
## The total number of observations was  848  with Likelihood Chi Square =  842.44  with prob <  1.6e-77
##
## Tucker Lewis Index of factoring reliability =  0.773
## RMSEA index =  0.061  and the 90 % confidence intervals are  0.056 0.064
## BIC =  -553.33
## Fit based upon off diagonal values = 0.93
## Measures of factor score adequacy
##                                                    MR1  MR2  MR3
## Correlation of (regression) scores with factors   0.91 0.89 0.77
## Multiple R square of scores with factors          0.83 0.80 0.59
## Minimum correlation of possible factor scores     0.66 0.59 0.17
```

We choose to use the two-factor solution as we believe its parsimony is not compromised by failure to adequately repesent the items. To this point, all items have loadings of at least absolute value .20. We name these factors “Unhealthy foods” and “Healthy foods” respectively, and add the estimated factor scores to the data frame.

```
colnames(two.factor$scores) <- c("Unhealthy","Healthy")
ori.diet <- cbind(ori.diet, two.factor$scores)
```

4 Subset data and calculate demographics

We create vectors that hold the names of the variables of interest by category. Specifically, we create a vector for the diet factors estimated, for the Big Five personality traits, for the health variables of interest, for proposed moderators of the relationship between diet and health, and finally for the covariates.

```
B5_scores <- c("Extra","Agree","Con","Neur","Open")
diet <- c("Healthy","Unhealthy")
cov <- c("gender","age_diet")
health <- c("SelfRatedHealth","BMI","Type2Diabetes")
moderators <- c("SES")
```

We remove participants who do not have scores for at least one of the big five personality scores, at least one of the diet estimates and at least one of the covariates.

```
ori.diet <- ori.diet[!is.na(rowSums(ori.diet[,B5_scores])), ]
ori.diet <- ori.diet[!is.na(rowSums(ori.diet[,diet])), ]
ori.diet <- ori.diet[!is.na(rowSums(ori.diet[,cov])), ]
```

Because sample sizes varied between questionnaires, we estimate sample sizes for the various analyses we present.

```
has.personality <- which(!is.na(rowMeans(ori.diet[, B5_scores], na.rm=T)))
has.health <- which(!is.na(rowMeans(ori.diet[, health], na.rm=T)))
has.SES <- which(!is.na(ori.diet[, "SES"]))
has.food <- which(!is.na(ori.diet[, "Healthy"]))

# Personality and health
length(which(has.personality %in% has.health))

## [1] 592

# SES and health
length(which(has.SES %in% has.health))

## [1] 414

#has all three
has.person_and_health <- has.personality[which(has.personality %in% has.health)]
length(which(has.person_and_health %in% has.SES))

## [1] 414
```

We then calculate descriptive statistics for each of our variables using the describe function in the psych package (Revelle, 2019) and format the resulting table using the stargazer package (Hlavac, 2018).

```
stargazer(describe(ori.diet[,c(B5_scores, diet, cov, "age_clinic",
                               health,
                               "q7maxinc", "highgrd", moderators)]),
          summary=F, digits=2,
          title="Descriptives of ORI data", out = "descriptives.tex", label="describe")
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
Extra	1	665	3.41	0.53	3.42	3.42	0.56	1.88	4.71	2.83	-0.28	-0.19	0.02
Agree	2	665	3.83	0.40	3.83	3.84	0.37	2.21	4.92	2.71	-0.30	0.43	0.02
Con	3	665	4.01	0.46	4	4.02	0.43	2.46	5	2.54	-0.19	-0.14	0.02
Neur	4	665	2.58	0.56	2.50	2.56	0.49	1.12	4.38	3.25	0.42	0.18	0.02
Open	5	665	3.14	0.42	3.12	3.13	0.37	1.50	4.75	3.25	0.24	0.76	0.02
Healthy	6	665	-0.02	0.89	-0.04	-0.04	0.98	-2.20	2.79	4.99	0.21	-0.50	0.03
Unhealthy	7	665	0.01	0.90	-0.10	-0.04	0.80	-2.28	4.91	7.19	0.65	1.44	0.03
gender	8	665	0.48	0.50	0	0.48	0	0	1	1	0.06	-2.00	0.02
age_diet	9	665	44.09	1.97	44	44.07	2.97	40	48	8	-0.07	-0.83	0.08
age_clinic	10	425	51.27	2.68	51.08	51.14	2.47	45.50	59.67	14.17	0.41	-0.07	0.13
SelfRatedHealth	11	487	3.29	0.94	3	3.28	1.48	1	5	4	-0.15	-0.29	0.04
BMI	12	493	28.58	6.39	27.41	27.94	5.58	16.22	56.41	40.20	1.19	2.19	0.29
Type2Diabetes	13	486	0.19	0.39	0	0.11	0	0	1	1	1.62	0.61	0.02
q7maxinc	14	415	3.69	1.04	4	3.76	1.48	1	5	4	-0.15	-0.83	0.05
highgrd	15	663	7.00	1.73	7	7.15	1.48	1	9	8	-0.83	-0.37	0.07
SES	16	414	0.20	0.91	0.29	0.25	0.86	-2.41	1.51	3.91	-0.38	-0.61	0.04

Table 1: Descriptives of ORI data

Next we examine the correlations between study variables and the covariates, and we also calculate means and standard deviations by gender.

```
corr.test(x = ori.diet[,c(B5_scores, diet, health, cov, "age_clinic",
                               "q7maxinc", "highgrd", moderators)],
          y = ori.diet[,c(cov, "SES")], adjust = "none")

## Call:corr.test(x = ori.diet[, c(B5_scores, diet, health, cov, "age_clinic",
##     "q7maxinc", "highgrd", moderators)], y = ori.diet[, c(cov,
##     "SES")], adjust = "none")
## Correlation matrix
##                 gender age_diet   SES
## Extra            -0.10    -0.07 -0.01
## Agree            -0.26     0.01  0.00
## Con              -0.06    -0.01  0.19
## Neur             -0.06    -0.02 -0.17
## Open             -0.18     0.04  0.16
## Healthy          -0.13     0.11  0.23
## Unhealthy         0.16    -0.09 -0.13
## SelfRatedHealth  -0.12     0.06  0.24
## BMI               0.20    -0.03 -0.20
## Type2Diabetes     0.02    -0.01 -0.13
## gender            1.00     0.02  0.15
## age_diet          0.02     1.00  0.23
```

```
## age_clinic        0.07     0.64  0.11
## q7maxinc          0.26     0.17  0.86
## highgrd          -0.04     0.17  0.79
## SES               0.15     0.23  1.00
## Sample Size
##                 gender age_diet SES
## Extra              665      665 414
## Agree              665      665 414
## Con                665      665 414
## Neur               665      665 414
## Open               665      665 414
## Healthy            665      665 414
## Unhealthy          665      665 414
## SelfRatedHealth    487      487 412
## BMI                493      493 328
## Type2Diabetes      486      486 412
## gender             665      665 414
## age_diet           665      665 414
## age_clinic         425      425 286
## q7maxinc           415      415 414
## highgrd            663      663 414
## SES                414      414 414
##                 gender age_diet  SES
## Extra             0.01     0.05 0.88
## Agree             0.00     0.79 0.92
## Con               0.13     0.70 0.00
## Neur              0.11     0.56 0.00
## Open              0.00     0.32 0.00
## Healthy           0.00     0.01 0.00
## Unhealthy         0.00     0.02 0.01
## SelfRatedHealth   0.01     0.16 0.00
## BMI               0.00     0.57 0.00
## Type2Diabetes     0.65     0.78 0.01
## gender            0.00     0.65 0.00
## age_diet          0.65     0.00 0.00
## age_clinic        0.16     0.00 0.05
## q7maxinc          0.00     0.00 0.00
## highgrd           0.28     0.00 0.00
## SES               0.00     0.00 0.00
##
##  To see confidence intervals of the correlations, print with the short=FALSE option
```

```
describeBy(ori.diet[,c(B5_scores, diet, health,
                               "q7maxinc", "highgrd", moderators,
                       cov, "age_clinic")],
            group = ori.diet$gender)
```

```
##
##  Descriptive statistics by group
## group: 0
##                 vars   n  mean   sd median trimmed  mad   min   max range  skew kurtosis   se
## Extra              1 343  3.46 0.52   3.50    3.48 0.49  1.88  4.71  2.83 -0.37     0.01 0.03
```

```
## Agree              2 343  3.93 0.38   3.96    3.94 0.37  2.50  4.92  2.42 -0.35     0.42 0.02
## Con                3 343  4.03 0.46   4.04    4.04 0.43  2.75  5.00  2.25 -0.19    -0.16 0.02
## Neur               4 343  2.62 0.59   2.54    2.59 0.56  1.12  4.38  3.25  0.45    -0.05 0.03
## Open               5 343  3.21 0.42   3.17    3.20 0.37  1.96  4.75  2.79  0.40     0.74 0.02
## Healthy            6 343  0.10 0.88   0.11    0.09 0.92 -2.20  2.52  4.73  0.11    -0.53 0.05
## Unhealthy          7 343 -0.13 0.82  -0.21   -0.17 0.77 -2.28  3.57  5.85  0.58     1.30 0.04
## SelfRatedHealth    8 260  3.40 0.89   3.00    3.41 1.48  1.00  5.00  4.00 -0.30    -0.03 0.06
## BMI                9 256 27.34 6.70  25.60   26.56 5.49 16.22 56.41 40.20  1.30     2.24 0.42
## Type2Diabetes     10 259  0.18 0.38   0.00    0.10 0.00  0.00  1.00  1.00  1.68     0.82 0.02
## q7maxinc          11 209  3.43 1.00   3.00    3.45 1.48  1.00  5.00  4.00  0.03    -0.41 0.07
## highgrd           12 343  7.07 1.67   7.00    7.23 1.48  2.00  9.00  7.00 -0.89    -0.27 0.09
## SES               13 209  0.06 0.86   0.03    0.08 0.86 -2.41  1.51  3.91 -0.26    -0.45 0.06
## gender            14 343  0.00 0.00   0.00    0.00 0.00  0.00  0.00  0.00   NaN      NaN 0.00
## age_diet          15 343 44.06 2.04  44.00   44.00 2.97 40.00 48.00  8.00  0.03    -0.88 0.11
## age_clinic        16 221 51.09 2.78  51.00   50.99 2.72 45.50 58.67 13.17  0.27    -0.39 0.19
## --------------------------------------------------------------------------------------------
## group: 1
##                 vars   n  mean   sd median trimmed  mad   min   max range  skew kurtosis   se
## Extra              1 322  3.35 0.53   3.36    3.36 0.54  1.88  4.71  2.83 -0.19    -0.35 0.03
## Agree              2 322  3.73 0.39   3.74    3.73 0.38  2.21  4.79  2.58 -0.29     0.62 0.02
## Con                3 322  3.98 0.46   3.98    3.99 0.46  2.46  4.96  2.50 -0.20    -0.13 0.03
## Neur               4 322  2.55 0.52   2.49    2.53 0.45  1.17  4.29  3.12  0.30     0.31 0.03
## Open               5 322  3.07 0.40   3.06    3.06 0.40  1.50  4.42  2.92  0.00     0.50 0.02
## Healthy            6 322 -0.14 0.88  -0.22   -0.17 0.95 -2.14  2.79  4.93  0.32    -0.40 0.05
## Unhealthy          7 322  0.16 0.95   0.10    0.12 0.90 -2.10  4.91  7.01  0.59     1.32 0.05
## SelfRatedHealth    8 227  3.17 0.98   3.00    3.14 1.48  1.00  5.00  4.00  0.04    -0.45 0.07
## BMI                9 237 29.91 5.76  28.90   29.24 4.65 20.47 53.91 33.44  1.41     2.84 0.37
## Type2Diabetes     10 227  0.19 0.40   0.00    0.12 0.00  0.00  1.00  1.00  1.54     0.37 0.03
## q7maxinc          11 206  3.96 1.00   4.00    4.03 1.48  1.00  5.00  4.00 -0.36    -1.09 0.07
## highgrd           12 320  6.93 1.79   7.00    7.07 1.48  1.00  9.00  8.00 -0.76    -0.50 0.10
## SES               13 205  0.34 0.93   0.55    0.42 0.95 -1.83  1.51  3.33 -0.57    -0.60 0.07
## gender            14 322  1.00 0.00   1.00    1.00 0.00  1.00  1.00  0.00   NaN      NaN 0.00
## age_diet          15 322 44.12 1.88  44.00   44.16 2.97 40.00 48.00  8.00 -0.20    -0.78 0.10
## age_clinic        16 204 51.46 2.56  51.12   51.29 2.16 46.83 59.67 12.83  0.64     0.24 0.18
```

We also calculate standardized difference between variables on gender using the cohensD function in lsr Navarro (2015).

```
d.scores <- data.frame()
for(i in c(B5_scores, diet, health, "age_diet", "q7maxinc", "highgrd", moderators)){
  formula <- as.formula(paste(i, "gender", sep=" ~ "))
  d.scores[i, "d"] <- cohensD(formula, data = ori.diet)
  d.scores[i, "p"] <- t.test(formula, data = ori.diet)$p.value
  d.scores[i, "t"] <- t.test(formula, data = ori.diet)$statistic
  d.scores[i, "df"] <- t.test(formula, data = ori.diet)$parameter
}
round(d.scores, 2)
```

```
##                     d    p     t     df
## Extra           0.19 0.01  2.50 658.59
## Agree           0.53 0.00  6.84 659.23
## Con             0.12 0.13  1.50 661.26
```

```
## Neur             0.12 0.11  1.61 659.02
## Open             0.36 0.00  4.71 662.88
## Healthy          0.27 0.00  3.49 660.55
## Unhealthy        0.33 0.00 -4.23 633.14
## SelfRatedHealth 0.24 0.01  2.63 459.33
## BMI              0.41 0.00 -4.58 488.34
## Type2Diabetes    0.04 0.65 -0.46 471.00
## age_diet         0.03 0.65 -0.45 662.77
## q7maxinc         0.53 0.00 -5.39 412.93
## highgrd          0.08 0.28  1.07 649.09
## SES              0.31 0.00 -3.17 408.17
```

```
for(i in c(B5_scores, B5_scores, diet, cov, "age_clinic",
           health, moderators, "q7maxinc", "highgrd")){
  ori.diet[,i] <- as.numeric(scale(ori.diet[,i]))
}
```

5 Correlations with Big Five and outcomes

We then calculate zero-rder correlations between personality traits/health outcomes and the diet factors. We also calculate partial correlations between these variables controlling for our covariates.

```
R.zero <- corr.test(x = ori.diet[,c(B5_scores, health)],
                    y = ori.diet[,c(diet, health)],
                    adjust = "none")
R.part <- partial.corr.test(x = ori.diet[,c(B5_scores, health)],
                            y = ori.diet[,c(diet, health)],
                            m = ori.diet[,cov],
                            adjust = "none")
```

These results are printed, unformatted, here for those interested in seeing exact p-values and sample sizes for each correlation.

```
R.zero

## Call:corr.test(x = ori.diet[, c(B5_scores, health)], y = ori.diet[,
##     c(diet, health)], adjust = "none")
## Correlation matrix
##                 Healthy Unhealthy SelfRatedHealth   BMI Type2Diabetes
## Extra              0.07     -0.01            0.19  0.01          0.00
## Agree              0.12     -0.13            0.14 -0.07         -0.01
## Con                0.17     -0.14            0.23 -0.05         -0.04
## Neur              -0.16      0.11           -0.33  0.13          0.06
## Open               0.20     -0.16            0.18 -0.13         -0.04
## SelfRatedHealth    0.24     -0.16            1.00 -0.31         -0.31
## BMI               -0.20      0.09           -0.31  1.00          0.38
## Type2Diabetes     -0.10      0.06           -0.31  0.38          1.00
## Sample Size
##                 Healthy Unhealthy SelfRatedHealth BMI Type2Diabetes
```

```
## Extra               665       665             487 493           486
## Agree               665       665             487 493           486
## Con                 665       665             487 493           486
## Neur                665       665             487 493           486
## Open                665       665             487 493           486
## SelfRatedHealth     487       487             487 389           484
## BMI                 493       493             389 493           388
## Type2Diabetes       486       486             484 388           486
##                 Healthy Unhealthy SelfRatedHealth  BMI Type2Diabetes
## Extra              0.09      0.79               0 0.81          0.96
## Agree              0.00      0.00               0 0.15          0.76
## Con                0.00      0.00               0 0.27          0.38
## Neur               0.00      0.00               0 0.00          0.22
## Open               0.00      0.00               0 0.00          0.36
## SelfRatedHealth    0.00      0.00               0 0.00          0.00
## BMI                0.00      0.06               0 0.00          0.00
## Type2Diabetes      0.03      0.21               0 0.00          0.00
##
##  To see confidence intervals of the correlations, print with the short=FALSE option
```

```
R.part
```

```
## Call:partial.corr.test(x = ori.diet[, c(B5_scores, health)], y = ori.diet[,
##     c(diet, health)], m = ori.diet[, cov], adjust = "none")
## Correlation matrix
##                 Healthy Unhealthy SelfRatedHealth   BMI Type2Diabetes
## Extra              0.06      0.00            0.18  0.04          0.00
## Agree              0.09     -0.09            0.12 -0.01         -0.01
## Con                0.16     -0.13            0.23 -0.04         -0.04
## Neur              -0.16      0.12           -0.33  0.15          0.06
## Open               0.18     -0.14            0.16 -0.10         -0.04
## SelfRatedHealth    0.22     -0.14            1.00 -0.29         -0.31
## BMI               -0.18      0.05           -0.29  1.00          0.38
## Type2Diabetes     -0.09      0.05           -0.31  0.38          1.00
## Sample Size
##                 Healthy Unhealthy SelfRatedHealth BMI Type2Diabetes
## Extra               665       665             487 493           486
## Agree               665       665             487 493           486
## Con                 665       665             487 493           486
## Neur                665       665             487 493           486
## Open                665       665             487 493           486
## SelfRatedHealth     487       487             487 389           484
## BMI                 493       493             389 493           388
## Type2Diabetes       486       486             484 388           486
##                 Healthy Unhealthy SelfRatedHealth  BMI Type2Diabetes
## Extra              0.11      0.97            0.00 0.41          0.93
## Agree              0.02      0.02            0.01 0.75          0.85
## Con                0.00      0.00            0.00 0.34          0.40
## Neur               0.00      0.00            0.00 0.00          0.22
## Open               0.00      0.00            0.00 0.02          0.41
## SelfRatedHealth    0.00      0.00            0.00 0.00          0.00
## BMI                0.00      0.24            0.00 0.00          0.00
```

```
## Type2Diabetes      0.04      0.23            0.00 0.00          0.00
##
##  To see confidence intervals of the correlations, print with the short=FALSE option
```

We also examine the partial correlations of BMI with personality and diet controlling for age at the clinic assessment, as this is not perfectly correlated with age at the diet assessment. Past analyses of the Hawaii clinic sample suggest this age variable has stronger correlations with clinic-assessed outcomes. In the manuscript, we present these correlations.

```
R.part.BMI <- partial.corr.test(x = ori.diet[,c(B5_scores, health, diet)],
                            y = ori.diet[,c("BMI","SelfRatedHealth")],
                            m = ori.diet[,c("gender","age_clinic")],
                            adjust = "none")

R.part.BMI

## Call:partial.corr.test(x = ori.diet[, c(B5_scores, health, diet)],
##     y = ori.diet[, c("BMI", "SelfRatedHealth")], m = ori.diet[,
##         c("gender", "age_clinic")], adjust = "none")
## Correlation matrix
##                   BMI SelfRatedHealth
## Extra            0.06            0.12
## Agree            0.00            0.15
## Con             -0.04            0.23
## Neur             0.13           -0.34
## Open            -0.10            0.13
## SelfRatedHealth -0.31            1.00
## BMI              1.00           -0.31
## Type2Diabetes    0.37           -0.32
## Healthy         -0.18            0.30
## Unhealthy        0.02           -0.15
## Sample Size
##                 BMI SelfRatedHealth
## Extra           424             337
## Agree           424             337
## Con             424             337
## Neur            424             337
## Open            424             337
## SelfRatedHealth 337             337
## BMI             424             337
## Type2Diabetes   336             335
## Healthy         424             337
## Unhealthy       424             337
##                  BMI SelfRatedHealth
## Extra           0.21            0.02
## Agree           0.95            0.00
## Con             0.44            0.00
## Neur            0.01            0.00
## Open            0.04            0.02
## SelfRatedHealth 0.00            0.00
## BMI             0.00            0.00
## Type2Diabetes   0.00            0.00
```

```
## Healthy         0.00            0.00
## Unhealthy       0.61            0.00
##
##  To see confidence intervals of the correlations, print with the short=FALSE option
```

We extract the results of these analyses and format them, such that correlations significant at $\alpha < .001$, are given significance stars.

```
R.zero.r <- as.data.frame(R.zero$r)
R.zero.p <- R.zero$p

for(i in 1:ncol(R.zero.r)){
  R.zero.r[,i] <- as.character(round(R.zero.r[,i], 2))
  R.zero.r[,i] <- ifelse(R.zero.p[,i] < .001,
                         paste(R.zero.r[,i],"*",sep=""),
                         R.zero.r[,i])
}

R.part.r <- as.data.frame(R.part$r)
R.part.p <- R.part$p

for(i in 1:ncol(R.part.r)){
  R.part.r[,i] <- as.character(round(R.part.r[,i], 2))
  R.part.r[,i] <- ifelse(R.part.p[,i] < .001,
                         paste(R.part.r[,i],"*",sep=""),
                         R.part.r[,i])
  }

R.all <- as.data.frame(cbind(R.zero.r[,"Healthy"],
                             R.part.r[,"Healthy"],
                             R.zero.r[,"Unhealthy"],
                             R.part.r[,"Unhealthy"],
                             R.zero.r[,health],
                             R.part.r[,health]),
                       row.names = row.names(R.zero.r))
```

We print this formatted table using the stargazer function (Hlavac, 2018)

```
colnames(R.all) <- c(c("Healthy r", "Healthy r_p", "Unhealthy r", "Unhealthy r_p"),
                     paste(health,"r"), paste(health,"r_p"))
colnames(R.all) <-gsub("SelfRatedHealth", "SRH", colnames(R.all))
stargazer(R.all, summary=F,
          title = "Bivariate correlations between personality, diet dimensions and health outcomes",
          out = "bivariate.tex", label="bivariate")
```

	Healthy r	Healthy r_p	Unhealthy r	Unhealthy r_p	SRH r	BMI r	Type2Diabetes r	SRH r_p	BMI r_p	Type2Diabetes r_p
Extra	0.07	0.06	-0.01	0	0.19∗	0.01	0	0.18∗	0.04	0
Agree	0.12	0.09	-0.13	-0.09	0.14	-0.07	-0.01	0.12	-0.01	-0.01
Con	0.17∗	0.16∗	-0.14∗	-0.13∗	0.23∗	-0.05	-0.04	0.23∗	-0.04	-0.04
Neur	-0.16∗	-0.16∗	0.11	0.12	-0.33∗	0.13	0.06	-0.33∗	0.15	0.06
Open	0.2∗	0.18∗	-0.16∗	-0.14∗	0.18∗	-0.13	-0.04	0.16∗	-0.1	-0.04
SelfRatedHealth	0.24∗	0.22∗	-0.16∗	-0.14	1∗	-0.31∗	-0.31∗	1∗	-0.29∗	-0.31∗
BMI	-0.2∗	-0.18∗	0.09	0.05	-0.31∗	1∗	0.38∗	-0.29∗	1∗	0.38∗
Type2Diabetes	-0.1	-0.09	0.06	0.05	-0.31∗	0.38∗	1∗	-0.31∗	0.38∗	1∗

Table 2: Bivariate correlations between personality, diet dimensions and health outcomes

6 Moderation by SES

For our final set of analyses, we estimate the degree to which relationships between personality/health outcomes and the diet factors change depending upon socio-economic status.

```r
interaction.coef <- data.frame()
n <- 0

low.ses.data <- ori.diet

for(i in c(B5_scores, health, "SES")){
  low.ses.data[,i] <- as.numeric(scale(low.ses.data[,i]))
}

low.ses.data$SES <- low.ses.data$SES + 1

hgh.ses.data <- ori.diet
for(i in c(B5_scores, health, "SES")){
  hgh.ses.data[,i] <- as.numeric(scale(hgh.ses.data[,i]))
}
hgh.ses.data$SES <- hgh.ses.data$SES - 1

for(j in diet)
  {for(i in c(B5_scores, health)){
    n <- n + 1
    if(i != "BMI") cov.used <- cov
    if(i == "BMI") cov.used <- gsub("age_diet","age_clinic",cov)
    equation = paste(c(i,cov.used), collapse = " + ")
    equation <- gsub(i, paste(i, "SES", sep="*"), equation)
    equation <- paste(c(j, equation), collapse = " ~ ")
    equation <- as.formula(equation)
    mod <- lm(equation, data=ori.diet)
    C <- coef(summary(mod))
    interaction <- C[which(grepl(":", rownames(C))), ]
    interaction.coef[n,"Outcome"] <- j
    interaction.coef[n,"Predictor"] <- i
    interaction.coef[n,"Est"] <- interaction["Estimate"]
    interaction.coef[n,"StdError"] <- interaction["Std. Error"]
    interaction.coef[n,"tvalue"] <- interaction["t value"]
    interaction.coef[n,"pvalue"] <- interaction["Pr(>|t|)"]

    low.mod <- lm(equation, data=low.ses.data)
    C.low <- coef(summary(low.mod))
    interaction.coef[n,"low SES r"] <- ifelse(C.low[i, "Pr(>|t|)"] < .01,
                                              paste(round(C.low[i, "Estimate"],2),"*",sep=""),
                                              as.character(round(C.low[i, "Estimate"],2)))

    hgh.mod <- lm(equation, data=hgh.ses.data)
    C.hgh <- coef(summary(hgh.mod))
    interaction.coef[n,"hgh SES r"] <- ifelse(C.hgh[i, "Pr(>|t|)"] < .01,
                                              paste(round(C.hgh[i, "Estimate"],2),"*",sep=""),
                                              as.character(round(C.hgh[i, "Estimate"],2)))
```

```
  }
}



interaction.coef$Est <- ifelse(interaction.coef$pvalue < .01,
                              paste(round(interaction.coef$Est,2), "*", sep=""),
                              as.character(round(interaction.coef$Est, 2)))
```

```
interaction.coef$Predictor <- gsub("Extra","Extraversion", interaction.coef$Predictor)
interaction.coef$Predictor <- gsub("Agree","Agreeableness", interaction.coef$Predictor)
interaction.coef$Predictor <- gsub("Con","Conscientiousness", interaction.coef$Predictor)
interaction.coef$Predictor <- gsub("Neur","Neuroticism", interaction.coef$Predictor)
interaction.coef$Predictor <- gsub("Open","Openness", interaction.coef$Predictor)
interaction.coef$Predictor <- gsub("Type2","Type II ", interaction.coef$Predictor)
interaction.coef$Predictor <- gsub("SelfRatedHealth","Self-Rated Health", interaction.coef$Predictor)

stargazer(interaction.coef, summary=F, digits = 2, rownames = F,
          title="Interaction of SES with personality and health variables on diet factors", out = "intera
```

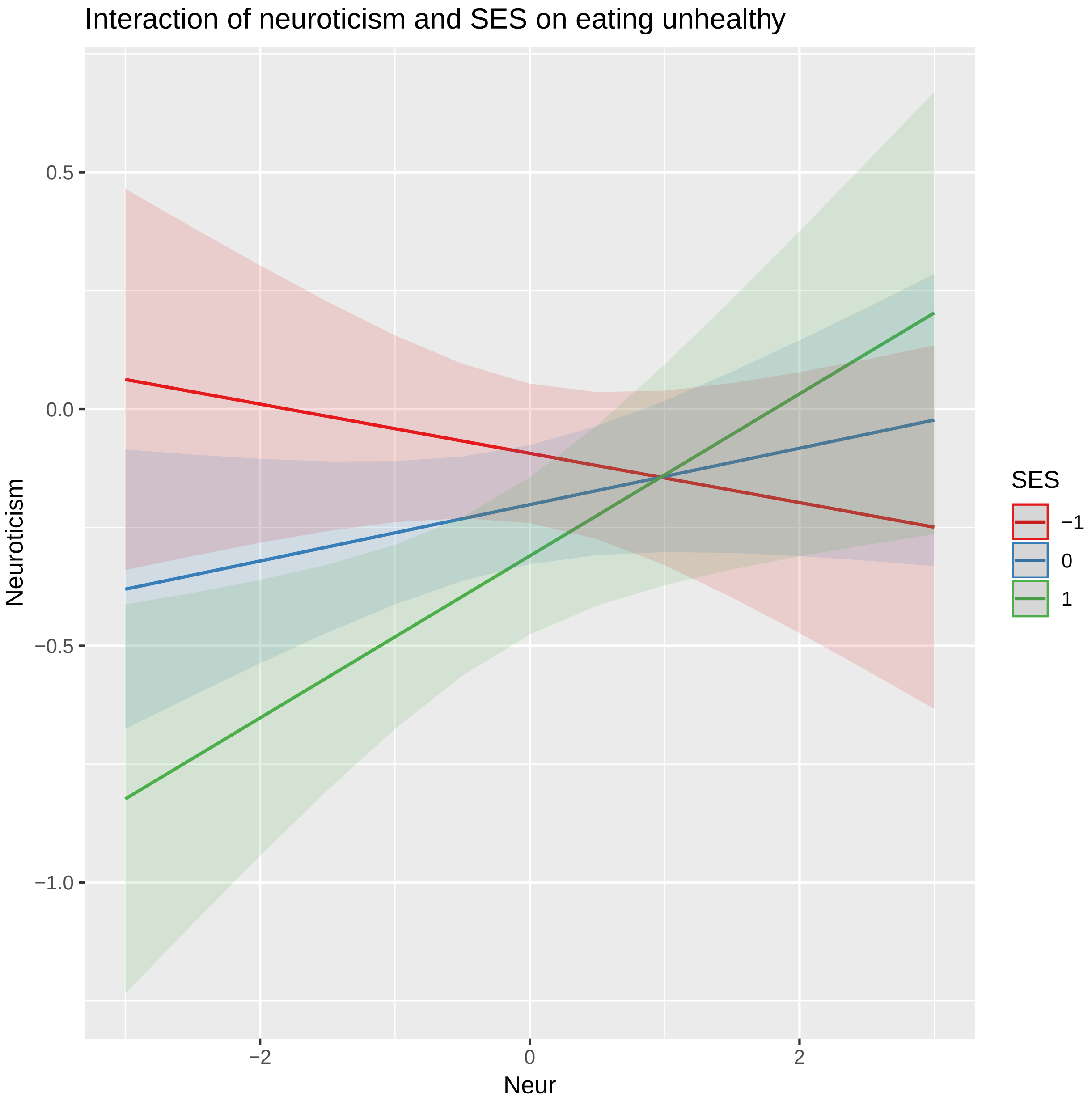
Outcome	Predictor	Est	StdError	tvalue	pvalue	low SES r	hgh SES r
Healthy	Extraversion	0.01	0.04	0.25	0.80	0.01	0.04
Healthy	Agreeableness	-0.02	0.04	-0.55	0.58	0.08	0.03
Healthy	Conscientiousness	-0.03	0.05	-0.63	0.53	0.11	0.04
Healthy	Neuroticism	0.03	0.05	0.57	0.57	-0.12	-0.06
Healthy	Openness	0.03	0.05	0.54	0.59	0.1	0.15
Healthy	Self-Rated Health	-0.01	0.04	-0.28	0.78	0.19*	0.16*
Healthy	BMI	0	0.06	-0.02	0.98	-0.16	-0.16
Healthy	Type II Diabetes	0.01	0.05	0.23	0.82	-0.08	-0.06
Unhealthy	Extraversion	-0.08	0.04	-1.88	0.06	0.12	-0.05
Unhealthy	Agreeableness	-0.06	0.04	-1.40	0.16	0.02	-0.1
Unhealthy	Conscientiousness	-0.07	0.05	-1.35	0.18	0	-0.14
Unhealthy	Neuroticism	0.11	0.05	2.41	0.02	-0.05	0.17
Unhealthy	Openness	-0.02	0.05	-0.45	0.65	-0.05	-0.09
Unhealthy	Self-Rated Health	0.01	0.04	0.17	0.86	-0.13	-0.11
Unhealthy	BMI	0	0.06	0.04	0.97	0.03	0.03
Unhealthy	Type II Diabetes	-0.01	0.05	-0.19	0.85	0.05	0.03

Table 3: Interaction of SES with personality and health variables on diet factors

6.1 Plot significant interaction(s)

```
ori.diet$gender <- as.factor(ori.diet$gender)
unhealthy.n <- lm(Unhealthy ~ Neur*SES + age_diet + gender,
                  data=ori.diet)
```

```
plot_model(unhealthy.n,
        type = "pred",
        swap.pred = T,
        terms = c("Neur", "SES"),
        mdrt.values = "meansd",
        title="Interaction of neuroticism and SES on eating unhealthy",
        legend.labels = c("-1SD", "Mean", "+1SD"),
        axis.title = "Neuroticism")
```



7 Sensitivity analyses

We recalculate the interactions with education and maximum income, instead of the socio-economic status composite, as the covariate. We hide the code for the compiled version of this document, to preserve space, but the code is available in the Rnw file.

Outcome	Predictor	Est	StdError	tvalue	pvalue	low q7maxinc r	hgh q7maxinc r
Healthy	Extraversion	0.04	0.04	1.02	0.31	-0.02	0.07
Healthy	Agreeableness	-0.04	0.05	-0.90	0.37	0.12	0.03
Healthy	Conscientiousness	-0.04	0.05	-0.86	0.39	0.15*	0.06
Healthy	Neuroticism	-0.01	0.05	-0.25	0.80	-0.11	-0.13
Healthy	Openness	0.06	0.05	1.36	0.17	0.09	0.21***
Healthy	Self-Rated Health	0.02	0.04	0.51	0.61	0.18**	0.23***
Healthy	BMI	0.01	0.05	0.21	0.83	-0.2**	-0.18*
Healthy	Type II Diabetes	0.02	0.05	0.35	0.73	-0.11	-0.07
Unhealthy	Extraversion	-0.1*	0.04	-2.31	0.02	0.13*	-0.06
Unhealthy	Agreeableness	-0.04	0.05	-0.88	0.38	0	-0.09
Unhealthy	Conscientiousness	-0.06	0.05	-1.14	0.25	-0.03	-0.14*
Unhealthy	Neuroticism	0.09*	0.05	1.98	0.05	-0.01	0.17*
Unhealthy	Openness	-0.02	0.05	-0.40	0.69	-0.07	-0.11
Unhealthy	Self-Rated Health	0.02	0.04	0.50	0.61	-0.16*	-0.12
Unhealthy	BMI	-0.01	0.06	-0.22	0.82	0.11	0.08
Unhealthy	Type II Diabetes	0	0.05	-0.09	0.93	0.06	0.05

Table 4: Interaction of q7maxinc with personality and health variables on diet factors

Outcome	Predictor	Est	StdError	tvalue	pvalue	low highgrd r	hgh highgrd r
Healthy	Extraversion	0.02	0.04	0.60	0.55	0.06	0.1
Healthy	Agreeableness	0.02	0.04	0.45	0.65	0.04	0.07
Healthy	Conscientiousness	-0.01	0.04	-0.20	0.84	0.13*	0.11*
Healthy	Neuroticism	0	0.04	-0.12	0.91	-0.1	-0.11*
Healthy	Openness	0.04	0.04	0.97	0.33	0.07	0.15**
Healthy	Self-Rated Health	-0.02	0.04	-0.55	0.58	0.19**	0.14*
Healthy	BMI	-0.03	0.04	-0.63	0.53	-0.1	-0.15*
Healthy	Type II Diabetes	-0.04	0.04	-0.87	0.39	-0.02	-0.09
Unhealthy	Extraversion	-0.02	0.04	-0.55	0.58	0.01	-0.03
Unhealthy	Agreeableness	-0.06	0.04	-1.54	0.12	-0.01	-0.13*
Unhealthy	Conscientiousness	0.02	0.04	0.42	0.68	-0.12*	-0.09
Unhealthy	Neuroticism	0.02	0.04	0.40	0.69	0.06	0.09
Unhealthy	Openness	0	0.04	-0.12	0.91	-0.09	-0.1*
Unhealthy	Self-Rated Health	-0.04	0.05	-0.92	0.36	-0.05	-0.13*
Unhealthy	BMI	0.07	0.04	1.57	0.12	-0.05	0.08
Unhealthy	Type II Diabetes	0.01	0.04	0.23	0.82	0.02	0.04

Table 5: Interaction of highgrd with personality and health variables on diet factors

8 Additional analysis: Moderation by Gender

As requested during the peer review process, we estimate the degree to which relationships between personality/health outcomes and the diet factors change depending upon gender. We do not include a three-way interaction between gender, SES, and diet, as we believe we are not sufficiently powered.

```
interaction.coef <- data.frame()
n <- 0

for(j in diet)
  {for(i in c(B5_scores, health)){
    n <- n + 1
    if(i != "BMI") cov.used <- cov
    if(i == "BMI") cov.used <- gsub("age_diet","age_clinic",cov)
    equation = paste(c(i,cov.used), collapse = " + ")
    equation <- gsub(i, paste(i, "gender", sep="*"), equation)
    equation <- paste(c(j, equation), collapse = " ~ ")
    equation <- as.formula(equation)
    mod <- lm(equation, data=ori.diet)
    C <- coef(summary(mod))
    interaction <- C[which(grepl(":", rownames(C))), ]
    interaction.coef[n,"Outcome"] <- j
    interaction.coef[n,"Predictor"] <- i
    interaction.coef[n,"Est"] <- interaction["Estimate"]
    interaction.coef[n,"StdError"] <- interaction["Std. Error"]
    interaction.coef[n,"tvalue"] <- interaction["t value"]
    interaction.coef[n,"pvalue"] <- interaction["Pr(>|t|)"]
  }
}



interaction.coef$Est <- ifelse(interaction.coef$pvalue < .01,
                               paste(round(interaction.coef$Est,2), "*", sep=""),
                               as.character(round(interaction.coef$Est, 2)))
```

```
interaction.coef$Predictor <- gsub("Extra","Extraversion", interaction.coef$Predictor)
interaction.coef$Predictor <- gsub("Agree","Agreeableness", interaction.coef$Predictor)
interaction.coef$Predictor <- gsub("Con","Conscientiousness", interaction.coef$Predictor)
interaction.coef$Predictor <- gsub("Neur","Neuroticism", interaction.coef$Predictor)
interaction.coef$Predictor <- gsub("Open","Openness", interaction.coef$Predictor)
interaction.coef$Predictor <- gsub("Type2","Type II ", interaction.coef$Predictor)
interaction.coef$Predictor <- gsub("SelfRatedHealth","Self-Rated Health", interaction.coef$Predictor)

stargazer(interaction.coef, summary=F, digits = 2, rownames = F,
          title="Interaction of gender with personality and health variables on diet factors",
          out = "interactions_gender.tex")
```

Outcome	Predictor	Est	StdError	tvalue	pvalue
Healthy	Extraversion	-0.02	0.08	-0.31	0.76
Healthy	Agreeableness	-0.04	0.08	-0.56	0.57
Healthy	Conscientiousness	-0.07	0.08	-0.94	0.35
Healthy	Neuroticism	0.04	0.08	0.50	0.62
Healthy	Openness	-0.01	0.08	-0.16	0.87
Healthy	Self-Rated Health	0.16	0.09	1.87	0.06
Healthy	BMI	-0.01	0.10	-0.13	0.90
Healthy	Type II Diabetes	0.05	0.09	0.62	0.54
Unhealthy	Extraversion	0	0.08	0.05	0.96
Unhealthy	Agreeableness	-0.09	0.08	-1.09	0.28
Unhealthy	Conscientiousness	-0.03	0.08	-0.42	0.67
Unhealthy	Neuroticism	0.12	0.08	1.60	0.11
Unhealthy	Openness	0	0.08	0.06	0.95
Unhealthy	Self-Rated Health	-0.09	0.09	-1.04	0.30
Unhealthy	BMI	-0.04	0.10	-0.38	0.70
Unhealthy	Type II Diabetes	-0.03	0.09	-0.31	0.75

Table 6: Interaction of gender with personality and health variables on diet factors

9 Food consumption item frequency

For transparency purposes, we report the frequency of responses to each response category on each of the food consumption items.

question	Not at all	1-3 times in the last month	Once a week	2-3 times a week	4-6 times a week	Once a day	2+ times a day	Total responses
chow mein	151	482	63	8	3	0	0	707
spaghetti w/meat or cream sauce	148	483	58	13	3	0	0	705
macaroni,potato..salad w/mayo	116	441	102	42	6	0	1	708
hamburgers/cheeseburgers	133	356	157	54	3	3	1	707
pizza	118	497	77	8	3	2	1	706
steak/roast	116	406	134	45	3	0	1	705
chicken/fish	7	87	166	293	119	18	15	705
tofu	188	312	118	64	17	7	1	707
spam/ham/luncheon meats	161	340	119	61	21	3	1	706
sausage/bacon/hot dogs	104	380	151	63	8	1	0	707
french fries	98	390	153	52	8	1	1	703
vegetables	6	75	86	162	144	122	108	703
fruit	22	119	112	185	104	108	55	705
high fiber cereals	257	147	78	103	53	61	7	706
dark whole grain breads	219	179	68	122	62	45	12	707
skim/1% fat milk	364	98	39	51	40	85	28	705
2% fat/whole milk	344	136	70	61	39	32	16	698
butter/margarine/peanut butter	53	206	138	176	75	42	15	705
doughnuts/pastry..	77	319	154	115	28	12	1	706
potato/corn chips	90	313	168	103	22	8	2	706
ice cream	181	343	100	62	14	7	0	707
poi	386	247	38	17	11	1	8	708
rice	14	48	47	111	153	194	139	706
mayonnaise/regular salad dressings	48	190	134	187	89	53	7	708

Table 7: Food consumption items and frequency of response endorsement

Bibliography

R-stargazerHlavac, M. 2018. stargazer: Well-Formatted Regression and Summary Statistics Tables stargazer: Well-formatted regression and summary statistics tables []. https://CRAN.R-project.org/package=stargazer R package version 5.2.2

R-lsrNavarro, D. 2015. lsr: Companion to "Learning Statistics with R" lsr: Companion to "learning statistics with r" []. https://CRAN.R-project.org/package=lsr R package version 0.5

R-foreignR Core Team. 2018. foreign: Read Data Stored by 'Minitab', 'S', 'SAS', 'SPSS', 'Stata', 'Systat', 'Weka', 'dBase', ... foreign: Read data stored by 'minitab', 's', 'sas', 'spss', 'stata', 'systat', 'weka', 'dbase', ... []. https://CRAN.R-project.org/package=foreign R package version 0.8-71

R-psychRevelle, W. 2019. psych: Procedures for Psychological, Psychometric, and Personality Research psych: Procedures for psychological, psychometric, and personality research []. https://CRAN.R-project.org/package=psych R package version 1.8.12

R-havenWickham, H. Miller, E. 2019. haven: Import and Export 'SPSS', 'Stata' and 'SAS' Files haven: Import and export 'spss', 'stata' and 'sas' files []. https://CRAN.R-project.org/package=haven R package version 2.1.0