Untitled

Emma Livingston

10/23/2019

```
library(psysds364data)
library(tidyverse)
```

```
## -- Attaching packages ------------------------------------------------------ tidyverse 1.2.1 --

## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## -- Conflicts --------------------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(sjlabelled)
```

```
##
## Attaching package: 'sjlabelled'

## The following object is masked from 'package:forcats':
##
##     as_factor

## The following object is masked from 'package:dplyr':
##
##     as_label
```

```
library(psych)
```

```
##
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
library(skimr)
```

```
##
## Attaching package: 'skimr'

## The following object is masked from 'package:stats':
##
##     filter
```

```r
data("Goldberg_gav")
```

```r
var_labels <- data.frame(column_name = colnames(Goldberg_gav),
                         variable_label = sjlabelled::get_label(Goldberg_gav))

Goldberg_gav_sm <- Goldberg_gav %>%
  select(ID,
         Partner,
         group,
         ChildAge,
         chgend,
         gendexp,
         gender,
         sexor,
         w11RelDur,
         contains("chrace"),
         religion,
         howrelig,
         contains("otherch"),
         occup,
         contains("otherwork"),
         educ,
         faminc,
         paraccme_5,
         contains("polit"),
         contains("gayaff"),
         cityst,
         contains("genderparent"),
         sexor,
         contains("myrace"),
         contains("h12"),
         contains("w12"),
         h12social,
         h12selfdir,
         h12conform,
         contains("soccomp"),
         SocComp,
         Prosocial,
         Emotional
         )
```

```r
#exclude individuals who do not have data for parental values or social competence

#find the subset of dyads who have information for both people for all of the variables
Goldberg_gav_completes <- Goldberg_gav_sm %>%
  filter(!is.na(SocComp) & !is.na(Prosocial) & !is.na(Emotional) & !is.na(h12social) & !is.na(h12selfdi
  group_by(ID) %>%
  count() %>%
  filter(n == 2)

#join this subset with the full set of individuals
Goldberg_gav_sm <- Goldberg_gav_sm %>%
  inner_join(Goldberg_gav_completes, by = "ID")
```

```r
Goldberg_gav_sm %>%
  group_by(gender) %>%
  count()
```

```
## # A tibble: 4 x 2
## # Groups:   gender [4]
##                 gender     n
##              <dbl+lbl> <int>
## 1  1 [Female]             93
## 2  2 [Male]               83
## 3  3 [Something else:]     1
## 4 NA                       1
```

```r
Goldberg_gav_sm <- Goldberg_gav_sm %>%
  mutate(gender = ifelse((gender == "NA" & (ID == 1 | ID == 3)), 1, gender))

Goldberg_gav_sm$gender[is.na(Goldberg_gav_sm$gender)] <- 1

Goldberg_gav_sm %>%
  group_by(gender) %>%
  count()
```

```
## # A tibble: 3 x 2
## # Groups:   gender [3]
##   gender     n
##    <dbl> <int>
## 1      1    94
## 2      2    83
## 3      3     1
```

```r
#impute ID 231 partner 2's relationship duration
Goldberg_gav_sm <- Goldberg_gav_sm %>%
  mutate(w11RelDur = ifelse(ID == 231 & Partner == 2, 7.0, w11RelDur))
```

```r
#write_csv(Goldberg_gav_sm, "C:/Users/livin/Documents/Smith Senior Year/Intergroup Relationships/Resear
```

```r
lesbians <- Goldberg_gav_sm %>%
  filter(group == 1)

gays <- Goldberg_gav_sm %>%
  filter(group == 2)

straight_men <- Goldberg_gav_sm %>%
  filter(group == 3 & gender == 2)

straight_women <- Goldberg_gav_sm %>%
  filter(group == 3 & gender == 1)
```

```r
mean(lesbians$SocComp)
```

```
## [1] 2.208838
```

```r
mean(gays$SocComp)
```

```
## [1] 2.382734
```

```r
mean(straight_men$SocComp)
```

```
## [1] 2.190693
```

```r
mean(straight_women$SocComp)
```

```
## [1] 2.297619
```

```r
mean(lesbians$h12conform)
```

```
## [1] 23.7
```

```r
mean(gays$h12conform)
```

```
## [1] 22.14583
```

```r
mean(straight_men$h12conform)
```

```
## [1] 21.8
```

```r
mean(straight_women$h12conform)
```

```
## [1] 23.62857
```

```r
# dyads <- Goldberg_gav_sm %>%
#   mutate(Partner = case_when(Partner == 1 ~ "A",
#                              Partner == 2 ~ "B")) %>%
#   gather(variable, value, group:n) %>%
#   unite(var_partner, variable, Partner) %>%
#   spread(var_partner, value)
#
# small_dyad <- small %>%
#   mutate(gender = case_when(gender == -1 ~ "W",
#                             gender == 1 ~ "H")) %>%
#   select(-person) %>%
#   gather(variable, value, self1:self4)%>%
#   unite(var_gender, variable, gender) %>%
#   spread(var_gender, value)
```

Report basic demographics for the sample. That is, percentages of gender, percentages of racial groups, participant age, major, income, age of children, adoption type, etc. This depends, of course, on what you have in your data.

```r
#count how many dyads in each group type
Goldberg_gav_groups <- Goldberg_gav_sm %>%
  filter(Partner == 2) %>%
  group_by(group) %>%
  count()

#count how many people of each race
Goldberg_gav_sm$myrace_1[is.na(Goldberg_gav_sm$myrace_1)] <- 0
Goldberg_gav_sm$myrace_2[is.na(Goldberg_gav_sm$myrace_2)] <- 0
Goldberg_gav_sm$myrace_3[is.na(Goldberg_gav_sm$myrace_3)] <- 0
Goldberg_gav_sm$myrace_4[is.na(Goldberg_gav_sm$myrace_4)] <- 0
Goldberg_gav_sm$myrace_5[is.na(Goldberg_gav_sm$myrace_5)] <- 0
Goldberg_gav_sm$myrace_6[is.na(Goldberg_gav_sm$myrace_6)] <- 0
Goldberg_gav_sm$myrace_7[is.na(Goldberg_gav_sm$myrace_7)] <- 0

race <- Goldberg_gav_sm %>%
  select(contains("myrace"))

colSums(race)
```

```
## myrace_1 myrace_2 myrace_3 myrace_4 myrace_5 myrace_6 myrace_7
##        5        1        9        3        1      165        4
```

```r
race <- data.frame(column_name = colnames(race),
                   variable_label = get_label(race),
                   num_participants = colSums(race))

race <- race %>%
  mutate(perc_participants = (num_participants / 178) * 100)

#find level of education of participants
education <- Goldberg_gav_sm %>%
  group_by(educ) %>%
  count()

ed_labels <- data.frame(get_labels(Goldberg_gav_sm$educ))
data.table::setDT(ed_labels, keep.rownames = TRUE)[]
```

```
##    rn      get_labels.Goldberg_gav_sm.educ.
## 1:  1       Less than a High School Diploma
## 2:  2            High School Diploma or GED
## 3:  3 Some college or an Associate's Degree
## 4:  4           College (bachelor's) degree
## 5:  5                       Master's degree
## 6:  6                             PhD/JD/MD
```

```r
ed_labels$rn <- as.numeric(ed_labels$rn)

education <- education %>%
  left_join(ed_labels, by = c("educ" = "rn"))
```

```
## Warning: Column `educ`/`rn` has different attributes on LHS and RHS of join
```

```
names(education) <- c("educ", "num_participants", "level")
education <- education %>%
  mutate(perc_participants = (num_participants / 178) * 100)

#average relationship length
mean(lesbians$w11RelDur)
```

```
## [1] 7.916667
```

```
mean(gays$w11RelDur)
```

```
## [1] 7.895833
```

```
mean(straight_women$w11RelDur)
```

```
## [1] 8.428571
```

```
mean(straight_men$w11RelDur)
```

```
## [1] 8.7
```

```
#sexual orientation
sexual_orientation <- Goldberg_gav_sm %>%
  select(ID, Partner, sexor, group) %>%
  group_by(sexor) %>%
  count()

sexor_labels <- data.frame(variable_label =
                           get_labels(Goldberg_gav_sm$sexor))
data.table::setDT(sexor_labels, keep.rownames = TRUE)[]
```

```
##    rn                     variable_label
## 1:  1 Exclusively gay/lesbian/homosexual
## 2:  2      Mostly gay/lesbian/homosexual
## 3:  3                           Bisexual
## 4:  4                Mostly heterosexual
## 5:  5           Exclusively heterosexual
## 6:  6                             Queer
## 7:  7                          Pansexual
## 8:  8                   Something else:
```

```
sexor_labels$rn <- as.numeric(sexor_labels$rn)
sexual_orientation <- sexual_orientation %>%
  left_join(sexor_labels, by = c("sexor" = "rn"))
```

```
## Warning: Column `sexor`/`rn` has different attributes on LHS and RHS of
## join
```

```r
#parent's age
parent_age <- Goldberg_gav_sm %>%
  select(ID, Partner, group, paraccme_5) %>%
  group_by(paraccme_5) %>%
  count()
```

```r
counts_labels <- function(data, x) {
  #if factor convert to numeric
  dims <- dim(data)
  counts <- data %>%
    select_(x) %>%
    group_by_(x) %>%
    count_()
  labels <- tibble(variable_label =
                     get_labels(data[[x]]))
  data.table::setDT(labels, keep.rownames = TRUE)[]
  labels$rn <- as.numeric(labels[,rn])
  names(counts)[1] <- "x"
  x_bins <- counts %>%
    right_join(labels, by = c("x" = "rn"))
  names(x_bins)[1] <- c(x, "n", "variable_label")
  x_bins[["n"]][is.na(x_bins[["n"]])] <- 0
  x_bins <- x_bins %>%
    mutate(percent = (n / dims[[1]]) * 100)
  x_bins <- x_bins %>%
    select(1, variable_label, everything())
  return(x_bins)
}
```

```r
education_counts <- counts_labels(Goldberg_gav_sm, "educ")
```

```
## Warning: select_() is deprecated.
## Please use select() instead
##
## The 'programming' vignette or the tidyeval book can help you
## to program with select() : https://tidyeval.tidyverse.org
## This warning is displayed once per session.

## Warning: group_by_() is deprecated.
## Please use group_by() instead
##
## The 'programming' vignette or the tidyeval book can help you
## to program with group_by() : https://tidyeval.tidyverse.org
## This warning is displayed once per session.

## Warning: count_() is deprecated.
## Please use count() instead
##
## The 'programming' vignette or the tidyeval book can help you
## to program with count() : https://tidyeval.tidyverse.org
## This warning is displayed once per session.

## Warning: Column `x`/`rn` has different attributes on LHS and RHS of join
```

```
## Warning in names(x_bins)[1] <- c(x, "n", "variable_label"): number of items
## to replace is not a multiple of replacement length
```

```
sexor_counts <- counts_labels(Goldberg_gav_sm, "sexor")
```

```
## Warning: Column `x`/`rn` has different attributes on LHS and RHS of join

## Warning: number of items to replace is not a multiple of replacement length
```

```
#income_counts <- counts_labels(Goldberg_gav_sm, "faminc")

race_child <- counts_labels(Goldberg_gav_sm, "w12Race")
```

```
## Warning: Column `x`/`rn` has different attributes on LHS and RHS of join

## Warning: number of items to replace is not a multiple of replacement length
```

```
income <- skim(Goldberg_gav_sm$faminc)
#127 missings
#51 completes
# mean = 175327.5
#Q1 100000.0
#Q2 167000.0
#Q3 205000.0
#Q4 565000.0

childage_at_adoption <- skim(Goldberg_gav_sm$h12Age)
#THIS VARIABLE IS IN MONTHS
#mean =     10.03
# sd = 23.45
# Q3 = 7, max = 144
```

```
corr.test(gays %>% select(SocComp, Prosocial, Emotional, h12social, h12selfdir, h12conform))
```

```
## Call:corr.test(x = gays %>% select(SocComp, Prosocial, Emotional,
##     h12social, h12selfdir, h12conform))
## Correlation matrix
##            SocComp Prosocial Emotional h12social h12selfdir h12conform
## SocComp       1.00      0.93      0.95      0.10      -0.06       0.31
## Prosocial     0.93      1.00      0.76      0.04      -0.07       0.36
## Emotional     0.95      0.76      1.00      0.15      -0.05       0.22
## h12social     0.10      0.04      0.15      1.00      -0.10       0.04
## h12selfdir   -0.06     -0.07     -0.05     -0.10       1.00      -0.22
## h12conform    0.31      0.36      0.22      0.04      -0.22       1.00
## Sample Size
## [1] 48
## Probability values (Entries above the diagonal are adjusted for multiple tests.)
##            SocComp Prosocial Emotional h12social h12selfdir h12conform
## SocComp       0.00      0.00      0.00      1.00       1.00       0.37
## Prosocial     0.00      0.00      0.00      1.00       1.00       0.15
## Emotional     0.00      0.00      0.00      1.00       1.00       1.00
```

```
## h12social     0.49      0.80      0.31      0.00       1.00       1.00
## h12selfdir    0.66      0.61      0.75      0.49       0.00       1.00
## h12conform    0.03      0.01      0.13      0.79       0.13       0.00
##
##  To see confidence intervals of the correlations, print with the short=FALSE option
```

```
corr.test(lesbians %>% select(SocComp, Prosocial, Emotional, h12social, h12selfdir, h12conform))
```

```
## Call:corr.test(x = lesbians %>% select(SocComp, Prosocial, Emotional,
##     h12social, h12selfdir, h12conform))
## Correlation matrix
##            SocComp Prosocial Emotional h12social h12selfdir h12conform
## SocComp       1.00      0.90      0.89      0.09      -0.01       0.19
## Prosocial     0.90      1.00      0.60      0.11      -0.01       0.23
## Emotional     0.89      0.60      1.00      0.04      -0.01       0.11
## h12social     0.09      0.11      0.04      1.00      -0.33       0.07
## h12selfdir   -0.01     -0.01     -0.01     -0.33       1.00      -0.17
## h12conform    0.19      0.23      0.11      0.07      -0.17       1.00
## Sample Size
## [1] 60
## Probability values (Entries above the diagonal are adjusted for multiple tests.)
##            SocComp Prosocial Emotional h12social h12selfdir h12conform
## SocComp       0.00      0.00      0.00      1.00       1.00       1.00
## Prosocial     0.00      0.00      0.00      1.00       1.00       0.84
## Emotional     0.00      0.00      0.00      1.00       1.00       1.00
## h12social     0.52      0.42      0.73      0.00       0.12       1.00
## h12selfdir    0.93      0.92      0.96      0.01       0.00       1.00
## h12conform    0.14      0.08      0.40      0.61       0.19       0.00
##
##  To see confidence intervals of the correlations, print with the short=FALSE option
```

```
corr.test(Goldberg_gav_sm %>% select(SocComp, Prosocial, Emotional, h12social, h12selfdir, h12conform))
```

```
## Call:corr.test(x = Goldberg_gav_sm %>% select(SocComp, Prosocial,
##     Emotional, h12social, h12selfdir, h12conform))
## Correlation matrix
##            SocComp Prosocial Emotional h12social h12selfdir h12conform
## SocComp       1.00      0.92      0.92      0.03       0.05       0.07
## Prosocial     0.92      1.00      0.69      0.00       0.06       0.08
## Emotional     0.92      0.69      1.00      0.06       0.03       0.05
## h12social     0.03      0.00      0.06      1.00      -0.21       0.08
## h12selfdir    0.05      0.06      0.03     -0.21       1.00      -0.05
## h12conform    0.07      0.08      0.05      0.08      -0.05       1.00
## Sample Size
## [1] 178
## Probability values (Entries above the diagonal are adjusted for multiple tests.)
##            SocComp Prosocial Emotional h12social h12selfdir h12conform
## SocComp       0.00      0.00      0.00      1.00       1.00          1
## Prosocial     0.00      0.00      0.00      1.00       1.00          1
## Emotional     0.00      0.00      0.00      1.00       1.00          1
## h12social     0.69      0.96      0.42      0.00       0.06          1
## h12selfdir    0.55      0.45      0.74      0.01       0.00          1
## h12conform    0.35      0.29      0.52      0.30       0.50          0
```

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

```
corr.test(straight_women %>% select(SocComp, Prosocial, Emotional, h12social, h12selfdir, h12conform))
```

```
## Call:corr.test(x = straight_women %>% select(SocComp, Prosocial, Emotional,
##     h12social, h12selfdir, h12conform))
## Correlation matrix
##            SocComp Prosocial Emotional h12social h12selfdir h12conform
## SocComp       1.00      0.92      0.92      0.04       0.03      -0.23
## Prosocial     0.92      1.00      0.70     -0.04       0.11      -0.18
## Emotional     0.92      0.70      1.00      0.12      -0.06      -0.24
## h12social     0.04     -0.04      0.12      1.00      -0.22       0.16
## h12selfdir    0.03      0.11     -0.06     -0.22       1.00      -0.03
## h12conform   -0.23     -0.18     -0.24      0.16      -0.03       1.00
## Sample Size
## [1] 35
## Probability values (Entries above the diagonal are adjusted for multiple tests.)
##            SocComp Prosocial Emotional h12social h12selfdir h12conform
## SocComp       0.00      0.00      0.00      1.00       1.00          1
## Prosocial     0.00      0.00      0.00      1.00       1.00          1
## Emotional     0.00      0.00      0.00      1.00       1.00          1
## h12social     0.82      0.80      0.51      0.00       1.00          1
## h12selfdir    0.88      0.51      0.73      0.20       0.00          1
## h12conform    0.18      0.31      0.16      0.37       0.87          0
##
##  To see confidence intervals of the correlations, print with the short=FALSE option
```

```
corr.test(straight_men %>% select(SocComp, Prosocial, Emotional, h12social, h12selfdir, h12conform))
```

```
## Call:corr.test(x = straight_men %>% select(SocComp, Prosocial, Emotional,
##     h12social, h12selfdir, h12conform))
## Correlation matrix
##            SocComp Prosocial Emotional h12social h12selfdir h12conform
## SocComp       1.00      0.95      0.93     -0.23       0.28      -0.02
## Prosocial     0.95      1.00      0.76     -0.27       0.26      -0.08
## Emotional     0.93      0.76      1.00     -0.15       0.26       0.05
## h12social    -0.23     -0.27     -0.15      1.00      -0.20       0.18
## h12selfdir    0.28      0.26      0.26     -0.20       1.00       0.21
## h12conform   -0.02     -0.08      0.05      0.18       0.21       1.00
## Sample Size
## [1] 35
## Probability values (Entries above the diagonal are adjusted for multiple tests.)
##            SocComp Prosocial Emotional h12social h12selfdir h12conform
## SocComp       0.00      0.00      0.00      1.00       1.00          1
## Prosocial     0.00      0.00      0.00      1.00       1.00          1
## Emotional     0.00      0.00      0.00      1.00       1.00          1
## h12social     0.19      0.12      0.40      0.00       1.00          1
## h12selfdir    0.10      0.13      0.13      0.24       0.00          1
## h12conform    0.90      0.63      0.76      0.29       0.24          0
##
##  To see confidence intervals of the correlations, print with the short=FALSE option
```

```
#Above the diagonal are lesbians, below the diagonal are gay males
corr_matrix1 <- tibble(
  c("Social Competence", "Prosocial Competence", "Emotional Competence", "Social Values", "Self-Directi
  c("1.00", "0.93*", "0.95*", "0.10", "-0.06", "0.31*"),
  c("0.9*", "1.00", "0.76*", "0.04", "-0.07", "0.36*"),
  c("0.89*", "0.60*", "1.00", "0.15", "-0.05", "0.22"),
  c("0.09", "0.11", "0.04", "1.00", "-0.10", "0.04"),
  c("-0.01", "-0.01", "-0.01", "-0.33*", "1.00", "-0.22"),
  c("0.19", "0.23", "0.11", "0.07", "-0.17", "1.00")
)

names(corr_matrix1) <- c("Social Competence", "Prosocial Competence", "Emotional Competence", "Social V
```

```
## Warning: The `names` must have length 7, not 6.
## This warning is displayed once per session.
```

```
knitr::kable(corr_matrix1)
```

Social Competence	Prosocial Competence	Emotional Competence	Social Values	Self-Direction Values	Conform
Social Competence	1.00	0.9*	0.89*	0.09	-0.01
Prosocial Competence	0.93*	1.00	0.60*	0.11	-0.01
Emotional Competence	0.95*	0.76*	1.00	0.04	-0.01
Social Values	0.10	0.04	0.15	1.00	-0.33*
Self-Direction Values	-0.06	-0.07	-0.05	-0.10	1.00
Conformity Values	0.31*	0.36*	0.22	0.04	-0.22

```
t.test(filter(gays, Partner == 1)[["w11RelDur"]], (filter(lesbians, Partner == 1)[["w11RelDur"]]))
```

```
##
##  Welch Two Sample t-test
##
## data:  filter(gays, Partner == 1)[["w11RelDur"]] and (filter(lesbians, Partner == 1)[["w11RelDur"]])
## t = -0.11271, df = 46.306, p-value = 0.9107
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.380080  2.127625
## sample estimates:
## mean of x mean of y
##  7.760870  7.887097
```

```
t.test(straight_men$w11RelDur,filter(gays, Partner == 1)[["w11RelDur"]])
```

```
##
##  Welch Two Sample t-test
##
## data:  straight_men$w11RelDur and filter(gays, Partner == 1)[["w11RelDur"]]
## t = 0.84451, df = 47.126, p-value = 0.4027
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
```

```
##  -1.297848  3.176109
## sample estimates:
## mean of x mean of y
##   8.70000   7.76087
```

```
t.test(straight_women$w11RelDur,filter(lesbians, Partner == 1)[["w11RelDur"]])
```

```
##
##  Welch Two Sample t-test
##
## data:  straight_women$w11RelDur and filter(lesbians, Partner == 1)[["w11RelDur"]]
## t = 0.52937, df = 63.941, p-value = 0.5984
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.501988  2.584937
## sample estimates:
## mean of x mean of y
##  8.428571  7.887097
```

Out of the 178 participants, 94 are female and 83 are male. 41.5730337078652% identify as Exclusively gay/lesbian/homosexual, 10.6741573033708% identify as Mostly gay/lesbian/homosexual, 5.61797752808989% identify as Bisexual, 1.68539325842697% identify as Mostly heterosexual, 37.0786516853933% identify as Exclusively heterosexual, 1.68539325842697% identify as Queer, and 1.12359550561798% identify as Pansexual.