Week 4 Workbook

Odelia

3/17/2021

**Import data**

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(magrittr)  
library(devtools)

## Loading required package: usethis

library(easystats)

## # Attaching packages  
## ✔ insight 0.13.1.1 ✔ bayestestR 0.8.3.1   
## ✔ performance 0.7.0.1 ✔ parameters 0.12.0.1  
## ✔ see 0.6.2.1 ✔ effectsize 0.4.4.1   
## ✔ correlation 0.6.0.1 ✔ modelbased 0.5.9   
## ✔ report 0.2.0   
## Warnings or errors in CRAN checks for package(s) 'bayestestR', 'parameters', 'effectsize', 'correlation'.

library(lubridate)

##   
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':  
##   
## date, intersect, setdiff, union

library(forcats)  
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.0 ──

## ✓ ggplot2 3.3.3 ✓ readr 1.4.0  
## ✓ tibble 3.1.0 ✓ purrr 0.3.4  
## ✓ tidyr 1.1.2 ✓ stringr 1.4.0

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## x lubridate::as.difftime() masks base::as.difftime()  
## x lubridate::date() masks base::date()  
## x tidyr::extract() masks magrittr::extract()  
## x dplyr::filter() masks stats::filter()  
## x lubridate::intersect() masks base::intersect()  
## x dplyr::lag() masks stats::lag()  
## x purrr::set\_names() masks magrittr::set\_names()  
## x lubridate::setdiff() masks base::setdiff()  
## x lubridate::union() masks base::union()

nz\_0 <- readr::read\_csv2(url("https://raw.githubusercontent.com/go-bayes/psych-447/main/data/nz/nz.csv"))

## ℹ Using ',' as decimal and '.' as grouping mark. Use `read\_delim()` for more control.

##   
## ── Column specification ────────────────────────────────────────────────────────  
## cols(  
## .default = col\_double(),  
## Male = col\_character(),  
## BigDoms = col\_character(),  
## GenCohort = col\_character(),  
## Religious = col\_character(),  
## Believe.God = col\_character(),  
## Believe.Spirit = col\_character(),  
## Env.SacMade = col\_logical(),  
## FeelHopeless = col\_character(),  
## FeelDepressed = col\_character(),  
## FeelRestless = col\_character(),  
## EverythingIsEffort = col\_character(),  
## FeelWorthless = col\_character(),  
## FeelNervous = col\_character()  
## )  
## ℹ Use `spec()` for the full column specifications.

f<-c("None Of The Time",  
 "A Little Of The Time",  
 "Some Of The Time",  
 "Most Of The Time",  
 "All Of The Time")  
nz <- nz\_0 %>%  
 dplyr::mutate\_if(is.character, factor) %>%  
 select(  
 -c(  
 SWB.Kessler01,  
 SWB.Kessler02,  
 SWB.Kessler03,  
 SWB.Kessler04,  
 SWB.Kessler05,  
 SWB.Kessler06  
 )  
 ) %>%  
 dplyr::mutate(Wave = as.factor(Wave)) %>%  
 mutate(FeelHopeless = forcats::fct\_relevel(FeelHopeless, f)) %>%  
 mutate(FeelDepressed = forcats::fct\_relevel(FeelDepressed, f)) %>%  
 mutate(FeelRestless = forcats::fct\_relevel(FeelRestless, f)) %>%  
 mutate(EverythingIsEffort = forcats::fct\_relevel(EverythingIsEffort, f)) %>%  
 mutate(FeelWorthless = forcats::fct\_relevel(FeelWorthless, f)) %>%  
 mutate(FeelNervous = forcats::fct\_relevel(FeelNervous, f)) %>%  
 dplyr::mutate(Wave = as.factor(Wave)) %>%  
 dplyr::mutate(date = make\_date(year = 2009, month = 6, day = 30) + TSCORE)

**Question 1: Using the nz dataset, make all the hours variables into integers**

e <- as.integer(as.numeric(nz$Hours.Exercise))  
i <- as.integer(as.numeric(nz$Hours.Internet))  
w <- as.integer(as.numeric(nz$Hours.Work))  
n <- as.integer(as.numeric(nz$Hours.News))  
c <- as.integer(as.numeric(nz$HoursCharity))

**Question 2a: Create a new indicator that standardises the Pol.Orient variable, create a new indicator that centers the Pol.Orient variable, create a new indicator that centres the Age variable in decade-long units. Do this in a single piped workflow.**

nz1 <- nz %>%  
 select(Pol.Orient, Age)%>%  
 mutate(Pol.O = scale(Pol.Orient, scale = TRUE, center = TRUE), Pol.O2 = scale(Pol.Orient, scale = FALSE, center = TRUE), A1 = scale(Age, scale = FALSE, center = TRUE)/10)

**Question 2b: Select Hour.Exercise and filter Wave 2019**

nz2 <- nz %>%  
 select(Pol.Orient, Age, Hours.Exercise)%>%  
 mutate(Pol.O = scale(Pol.Orient, scale = TRUE, center = TRUE), Pol.O2 = scale(Pol.Orient, scale = FALSE, center = TRUE), A1 = scale(Age, scale = FALSE, center = TRUE)/10)%>%  
filter(nz$Wave!=2019)

**Question 3: What are the maximum number of responses for a single day in 2018 and the maximum number of responses for a single day in 2019?**

library(kableExtra) nz %>% select(date, Wave) %>% group\_by(Wave, date) %>% summarise(n = n())%>% kbl(caption = “No. of responses per day”) %>% kable\_classic\_2(c(“striped”, “hover”), full\_width = TRUE)%>% collapse\_rows()

Maximum number of responses for 2018: 121 on 2018-06-21

Maximum number of responses for 2019: 67 on 2019-12-03

**Question 4: How many days are there between the date with the highest number of responses and the date with the second highest number of responses? Bonus: Calculate difference between the number of responses on the highest response date and second highest response date.**

There is 1 day between the date with the highest number of responses (2018-06-21) and the date with the second highest number of responses (2018-06-22).

Difference between number of responses in the dates above: 121-104=17

**Question 5: Suppose you were born on Dec 25, 1995 at 5.02:22 am Calculate your age in months on March 20,2021, at 1:22:04pm. (Hint use the lubridate package. Look up the interval function).**

int<-lubridate::interval(ymd\_hms("1995-12-25 05:02:22"), ymd\_hms("2021-03-20 13:22:04"))  
time\_length(int,"month")

## [1] 302.8338

My age in months will be 302.8338.

**Question 6: The Religion.Church variable contains responses to the question: “How many times each month do you attend church or religious service?” Create factor with the following three levels: 1.People who attend church 0 times per month, 2.People who attend church 1-3 times per month, 3.People who attend church 4 or more times per month. Make sure to re-level the factor so that the ordinal ranking moves from lowest to highest.**

nb. People who attended 0 times/month are coded as “Low”, 1-3 times/month coded as “Moderate”, and >=4 times/month coded as “High”.

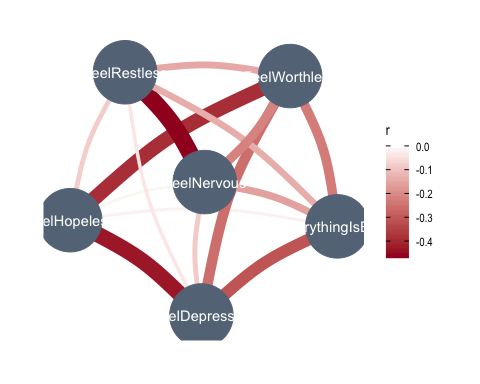
R<-nz %>%  
 dplyr::mutate(rc = as.factor(ifelse(Religion.Church <= 0,"Low",ifelse(Religion.Church <= 3, "Moderate", "High"))), rc=fct\_relevel(rc,"Low","Moderate","High")) %>%  
 group\_by(rc) %>%  
 count()

**Question 7: Using methods described in Lecture 4, create a table for average hours of sleep by month in the nz dataset. Graph the average hours of sleep by month including 95% confidence intervals. Briefly explain why some intervals are wider than others.**

I couldn’t solve this after 3 whole days on it. Please teach me :’)

**Question 8: Create a correlation graph for the items in the Kessler 6 scale. These are:-FeelHopeless, -FeelDepressed, -FeelRestless, -EverythingIsEffort, -FeelWorthless, -FeelNervous. What do you find most interesting about this plot? Explain.**

c<-nz%>%  
 select(FeelHopeless,FeelDepressed,FeelRestless,FeelWorthless,FeelNervous,EverythingIsEffort)%>%  
 mutate\_all(as.integer)%>%  
 mutate(KESSLER6sum = as.factor(nz$KESSLER6sum))  
library(correlation)  
library(ggraph)  
c1<-correlation(c, partial = FALSE, multilevel = TRUE)%>%  
 plot()  
c1



I find that the aesthetics of this plot to be the most interesting. The plot is visually a 2D plot but as I look as it, it depicts a 3D plot in my brain instead and it is very clear to see the strengths of the correlations between one another for every variable even when it is in actuality a 2D plot for as many as 5 variables. It is also able to show me negative correlations clearly with a different colour so I can easily tell positive and negative correlations apart.

**Question 9: Create a blank papaja report**

As attached.

**Question 10: Use the patchwork library to create a figure with two plots on top of each other. Use the tag\_levels function to index each of the two plots. The graphs should describe some dimension of the truncated nz dataset.**

c2<-c%>%  
 correlation(partial = FALSE, multilevel = FALSE)%>%  
 plot()  
library(patchwork)  
c1 / c2 +   
 plot\_annotation(title = "Plot of multilevel (a) and single-level (b) correlation", tag\_levels = 'a')

