Lab Demo

Don Zhang

2022-08-30

Project Folder Directory

The Project Template is organized into the following folders with specific functions:

1. Data - All raw data should be placed in this folder. In this folder, you will find a markdown file for you record informatino on each dataset.
2. ExternalAnalysis - Statistical analysis performed on specific software such as MPlus can be placed in here.
3. Outputs - All statistical outputs (e.g., figures, tables) go in here.
4. References - All reference material (e.g., codebook, scale info) go here.

Project Files

- 01_load.R - Script for loading essential libraries and functions
- 02_clean.R - Script for loading data, cleaning data, wrangling data
- readme.md - Markdown file for project description, notes, updates
- notebook.rmd - RMarkdown file for reporting, analyses, results

Data Management Principles

1. One shall never make destructive changes to data files
2. All data manipulation must be done in R
3. No duplicates or multiple iteration of data file, with the exception of specific data formats for external analyses

01_load.R

- Load the essential libraries for your analyses.

```
## Libraries ------------------------
            library(tidyverse)
            library(psych)
            library(lavaan)
            library(readr)
            library(ez)
            library(jmv)
            library(apaTables)
            set.seed(6958683) # set seed
```

02_load.R

1. Navigate to the "Data" directory and click on the df.raw.t1.csv file, followed by "Import Dataset. . ."
2. This interface allows you to change the name of the imported dataset, and make some adjustments to variables
3. Once finished, copy the data import script and paste it into the 02_clean.R file under the Data Import section.
4. Repeat for df.raw.t2.csv

```
df.raw.t1 <-
              read_csv("Data/df.raw.t1.csv")

df.raw.t2 <-
               read_csv("Data/df.raw.t2.csv")
```

Looking at Data

1. In the Enviornment Tab, you'll find the two dataframe objects
2. To see a list of variables, you can use the following command

```
names(df.raw.t1)
```

This is a very useful command and one you'll be using frequently in your analyses

Merging two datasets

- You will frequently be given data split into multiple files (e.g., longitudinal dataset)
- To merge two dataframes (e.g., time 1 and time 2) with same subjects, you will need to have a unique identifier in each data frame
- In these two datasets, we will link participants based on their ProlificID, which are the following

```
df.raw.t1$PROLIFID_T1
df.raw.t2$PROLIFIC_PID
```

Merging two datasets

- The following script will perform the merge

```
df.raw.combine <- merge(df.raw.t1, df.raw.t2,
                        by.x = "PROLIFID_T1",
                        by.y = "PROLIFIC_PID")
```

- What if I want to keep all the individuals from T1 even without a match?

```
df.raw.combine <- merge(df.raw.t1, df.raw.t2,
                        by.x = "PROLIFID_T1",
                        by.y = "PROLIFIC_PID",
                        all.x = T)
```

Renaming Variables

- You can rename one or multiple variable using the `rename()` command in the *dplyr* package

```
df.raw.combine <- rename(df.raw.combine,
        DV_Height = DV8,
        DV_Weight = DV9)
```

Recoding Variables

- Coding for attention checks
- Turning continuous variables to categories (but why would you?)
- Coding text data to numerical data (e.g., Strongly Disgree $=>$ 1)
- Reverse coding items (more on that later)

Creating new variables

- You can use the `mutate()` function in *dplyr* to create and recode variables

```
df.raw.combine <- mutate(df.raw.combine,
       DV_BMI = DV_Weight*703/(DV_Height^2))
```

- Can be used to code for attention
- The DOSPERT_T1_20 item is an attention check:

```
table(df.raw.combine$DOSPERT_T1_20)
```

- To recode:

```
df.raw.combine <- mutate(df.raw.combine,
                         ATTENTION_1 = case_when(
                         DOSPERT_T1_20 == 7 ~ 1,
                         DOSPERT_T1_20 < 7 ~ 0))
```

Filter data

- Now we want to clean the data by removing participants who failed the attention check question
- Filter by rows can be done using the `filter()` function in the *dplyr* package.
- At this stage, you can create a new "cleaned" dataframe

```
df.clean.combine <- filter(df.raw.combine,
                           ATTENTION_1 == 1)
```

- What would you do if you have three attention check questions?

Select variables

- You can select subsets of variables using the `select()` function from the *dplyr* package
- Select one variable

```
select(df.clean.combine, PROLIFID_T1)
```

- Select a range of variables

```
select(df.clean.combine, PROLIFID_T1:DV1)
```

- Deselect a variable

```
select(df.clean.combine, -PROLIFID_T1)
```

- Mix and match

```
select(df.clean.combine,
       PROLIFID_T1,
       DV4:DV6)
```

Piping

- So far, we performed each action (select, filter, rename, mutate) individually, and creating new iteration of the data as we go.
- Piping allows you perform these action altogether with the %>% command
- Putting it altogether

```
df.clean.combine <- df.raw.combine %>%
              select(-PROLIFID_T1) %>%
              rename(DV_Height = DV8,
                     DV_Weight = DV9) %>%
              mutate(DV_BMI = DV_Weight*703/(DV_Height^2),
                           ATTENTION_1 = case_when(
                           DOSPERT_T1_20 == 7 ~ 1,
                           DOSPERT_T1_20 < 7 ~ 0)) %>%
              filter(ATTENTION_1 == 1)
```

Scale Scoring

- You will often need to calculate a scale score based on some items
- There are may methods to accomplish this. I prefer using the scoreItem functiom from the *psych* package
- Lets calculate the GRiPS score based on the 8 items

```
as.vector(scoreItems(
              keys = c(1,1,1,1,1,1,1,1),
              items = select(df.clean.combine,
              GRIPS_T1_1:GRIPS_T1_8))$score)
```

Is everything reproducible?

- Lets check to see if all your script is reproducible
- Delete your entire work environment by clicking the broom.
- Run your script, does it reproduce all your cleaned data?
- Can someone else clone your project and execute/follow your analyses?