Lab #1
Introduction to R

August 12, 2020

Today's Big Ideas for the Lab

- Get comfortable with the basics of `R`
- Start exploring, summarizing, and plotting data in `R`

The Golden Rule of Working Directories

- It's always a good idea to tell `R` what directory/folder you are working in

  - To do this, we set the **working directory**: `setwd("some_higher_folder/folder_where_my_data_is/")`
  - Or use the `Files` pane

- Always save your `R` scripts in your working directory!

  - Relevant data files should also "live" in your working directory

- NOTE WELL: You only need to *install* a package **once**. You need to *load* the package every time you open `R` and want to use it.

- There are several ways to get your data into `R` for analysis or manipulation.

1) Use the Files window in RStudio to find the directory where your data file(s) are located, click on the file, and then click on `Import Dataset...`

2) In RStudio, go to `Files --> Import Dataset` and then choose the appropriate option to fit your file format.

3) Write code in your script.

Today's Example

Let's import a synthetic data set:

```r
dat <- read.csv("data_example_Lab_8.12.20.csv")
```

Basics of `R`

Mathematical Operations

- We can use `R` as a glorified calculator

- **Exercise:**

See what the following lines of code will do:

```r
2 + 2     # Addition
2 - 2     # Subtraction

2 * 2     # Multiplication
2 / 2     # Division

2 ^ 2     # Powers
sqrt(2)   # Square root
exp(2)    # Exponentiation (base e)
log(2)    # Natural logarithm
exp(log(2))
```

Objects and Assignment

- Any result of a function you use is called an *object*

```r
head(dat)   # See the first few rows of our data
```

```
##   X StudyCode MonthsOld Pre PartialCode Post Counting PrePostImprove
## 1 1         3     58.13   2           0    2       13              0
## 2 2         2     55.43   2           1    5       15              1
## 3 3         4     38.27   3           0    3        7              0
## 4 4         2     62.20   3           0    3       11              0
## 5 5         4     41.00   2           1    3        3              1
## 6 6         2     53.92   4           1    5       12              1
```

```r
tail(dat)   # See the last few rows of our data
```

```
##     X StudyCode MonthsOld Pre PartialCode Post Counting PrePostImprove
## 25 25         1     44.45   2           0    2        7              0
## 26 26         2     36.76   2           0    2        2              0
## 27 27         4     36.62   2           0    4       11              1
## 28 28         4     41.00   2           0    1        2              0
## 29 29         2     47.08   3           1    5       11              1
## 30 30         5     38.56   3           0    3        3              0
```

- Often, we want to store our objects for use later on
  - Objects must be named carefully:
    * `R` is case sensitive: `DAT` is not the same as `dat` or `Dat`
    * No spaces in object names: `da t` is invalid

    ∗ Object names cannot start with a number: `2day` is invalid, `day2` is fine

   – We can store our results in an object with the assignment operator `<-`

```r
dat <- dat[, -1]   # Remove the first column of our data and replace the object
head(dat)
```

```
##   StudyCode MonthsOld Pre PartialCode Post Counting PrePostImprove
## 1         3     58.13   2           0    2       13              0
## 2         2     55.43   2           1    5       15              1
## 3         4     38.27   3           0    3        7              0
## 4         2     62.20   3           0    3       11              0
## 5         4     41.00   2           1    3        3              1
## 6         2     53.92   4           1    5       12              1
```

Useful Commands for Objects

- See what objects are currently available: `ls()`

- Delete objects: `rm()`

Let's Explore Our Data Set

- Interactive: `View()`

```r
View(dat)
```

- Top and bottom rows: `head()` and `tail()`

```r
head(dat)
```

```
##   StudyCode MonthsOld Pre PartialCode Post Counting PrePostImprove
## 1         3     58.13   2           0    2       13              0
## 2         2     55.43   2           1    5       15              1
## 3         4     38.27   3           0    3        7              0
## 4         2     62.20   3           0    3       11              0
## 5         4     41.00   2           1    3        3              1
## 6         2     53.92   4           1    5       12              1
```

```r
tail(dat)
```

```
##    StudyCode MonthsOld Pre PartialCode Post Counting PrePostImprove
## 25         1     44.45   2           0    2        7              0
## 26         2     36.76   2           0    2        2              0
## 27         4     36.62   2           0    4       11              1
## 28         4     41.00   2           0    1        2              0
## 29         2     47.08   3           1    5       11              1
## 30         5     38.56   3           0    3        3              0
```

- Data structure: `str()`

```r
str(dat)
```

```
## 'data.frame':    30 obs. of  7 variables:
##  $ StudyCode    : int  3 2 4 2 4 2 2 3 3 4 ...
##  $ MonthsOld    : num  58.1 55.4 38.3 62.2 41 ...
##  $ Pre          : int  2 2 3 3 2 4 3 2 2 2 ...
##  $ PartialCode  : int  0 1 0 0 1 1 1 1 0 0 ...
##  $ Post         : int  2 5 3 3 3 5 3 5 0 3 ...
##  $ Counting     : int  13 15 7 11 3 12 15 7 4 3 ...
##  $ PrePostImprove: int  0 1 0 0 1 1 0 1 0 1 ...
```

- Number of participants / variables: `dim()`

```r
dim(dat)
```

```
## [1] 30  7
```

Data Summarization

- Summary of data: descriptives

```r
summary(dat)
```

```
##    StudyCode       MonthsOld          Pre         PartialCode        Post      
##  Min.   :1.000   Min.   :36.62   Min.   :2.0   Min.   :0.0   Min.   :0.000  
##  1st Qu.:2.000   1st Qu.:38.90   1st Qu.:2.0   1st Qu.:0.0   1st Qu.:2.000  
##  Median :3.000   Median :44.27   Median :2.0   Median :0.0   Median :3.000  
##  Mean   :2.867   Mean   :46.43   Mean   :2.4   Mean   :0.3   Mean   :2.733  
##  3rd Qu.:4.000   3rd Qu.:53.34   3rd Qu.:3.0   3rd Qu.:1.0   3rd Qu.:3.000  
##  Max.   :5.000   Max.   :65.65   Max.   :4.0   Max.   :1.0   Max.   :5.000  
##     Counting     PrePostImprove  
##  Min.   : 2.0   Min.   :0.0000  
##  1st Qu.: 3.0   1st Qu.:0.0000  
##  Median : 7.0   Median :0.0000  
##  Mean   : 8.5   Mean   :0.3667  
##  3rd Qu.:13.0   3rd Qu.:1.0000  
##  Max.   :20.0   Max.   :1.0000  
```

```r
library(psych)
describe(dat)  # From the `psych` package
```

```
vars n mean   sd median trimmed mad   min   max range skew
StudyCode   1 30 2.87 1.20   3.00   2.83 1.48 1.00 5.00 4.00 0.25
MonthsOld   2 30 46.43 8.60 44.27   45.69 8.85 36.62 65.65 29.03 0.66
Pre   3 30 2.40 0.62   2.00   2.29 0.00 2.00 4.00 2.00 1.20
PartialCode   4 30 0.30 0.47   0.00   0.25 0.00 0.00 1.00 1.00 0.83
Post   5 30 2.73 1.36   3.00   2.75 1.48 0.00 5.00 5.00 0.07
Counting   6 30 8.50 5.26   7.00   8.29 5.93 2.00 20.00 18.00 0.24
PrePostImprove   7 30 0.37 0.49   0.00   0.33 0.00 0.00 1.00 1.00 0.53
kurtosis   se
```

```
## StudyCode        -1.10 0.22
## MonthsOld        -0.93 1.57
## Pre               0.28 0.11
## PartialCode      -1.35 0.09
## Post             -0.42 0.25
## Counting         -1.27 0.96
## PrePostImprove   -1.78 0.09
```

Plotting

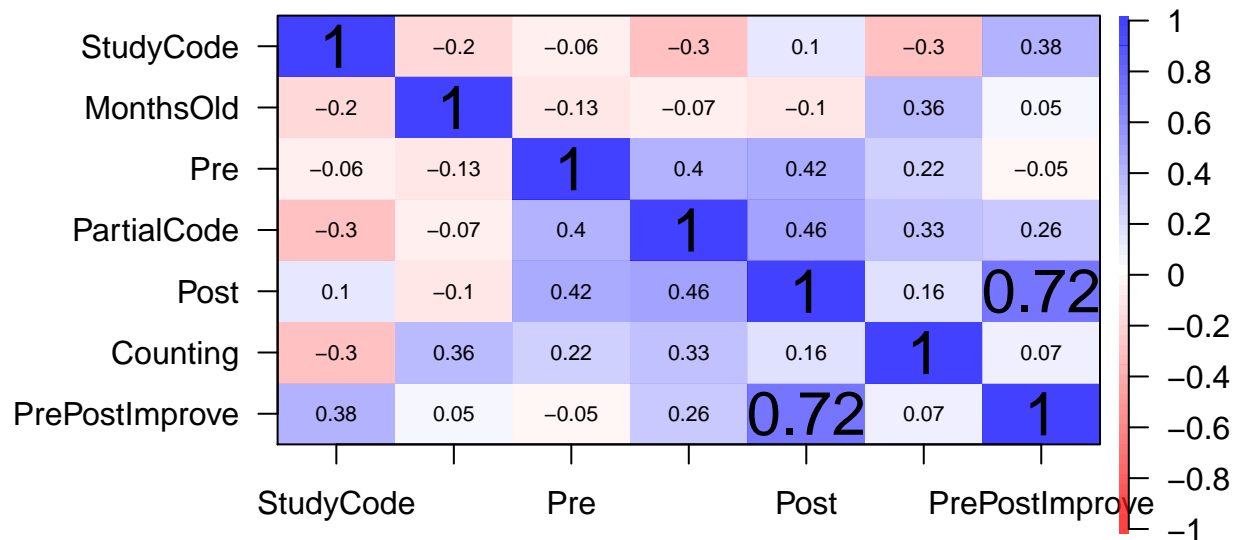- Pairwise distributions of all or some variables

```
with(dat, plot(Pre, Post, xlim = c(0, 5)))
```
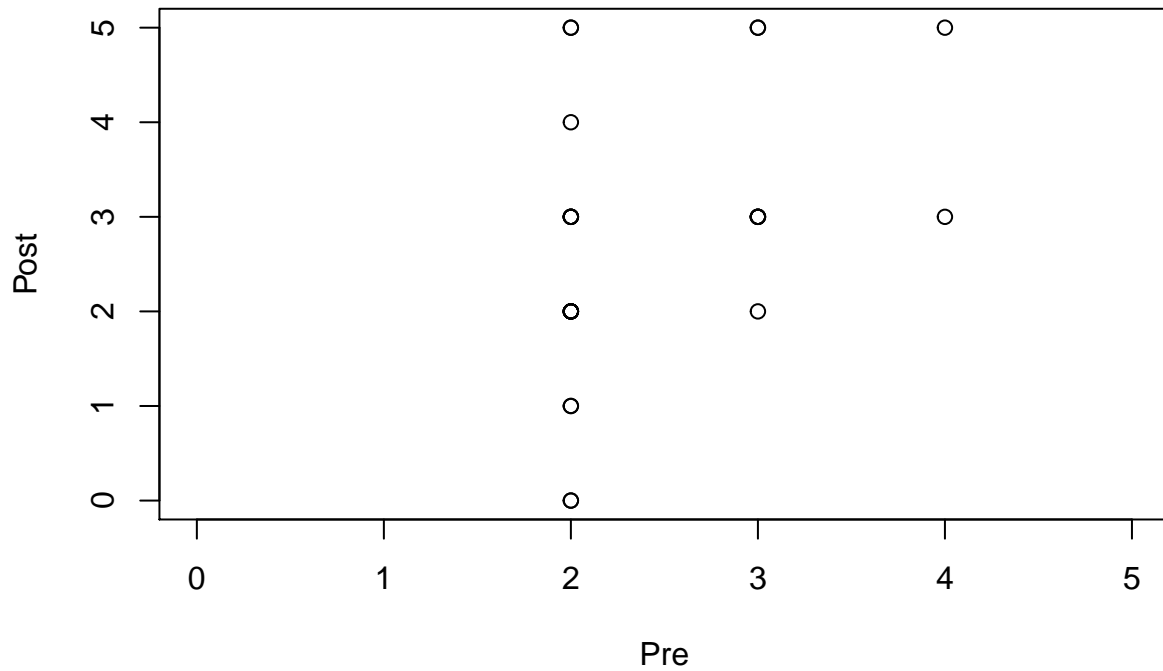

- Correlation plot

```
cor.plot(dat)  # From the `psych` package
```

Correlation plot


```r
round(cor(dat, method = "pearson"), 3)
```

```
##            StudyCode MonthsOld    Pre PartialCode   Post Counting
## StudyCode      1.000    -0.204 -0.065      -0.297  0.104   -0.302
## MonthsOld     -0.204     1.000 -0.134      -0.074 -0.103    0.362
## Pre           -0.065    -0.134  1.000       0.405  0.415    0.222
## PartialCode   -0.297    -0.074  0.405       1.000  0.456    0.331
## Post           0.104    -0.103  0.415       0.456  1.000    0.164
## Counting      -0.302     0.362  0.222       0.331  0.164    1.000
## PrePostImprove 0.380     0.055 -0.045       0.257  0.719    0.074
##            PrePostImprove
## StudyCode           0.380
## MonthsOld           0.055
## Pre                -0.045
## PartialCode         0.257
## Post                0.719
## Counting            0.074
## PrePostImprove      1.000
```

Selecting Observations and Variables

- To extract a single column/variable from our data frame

```
dat$Post          # The `$` operator extracts by name
```

```
## [1] 2 5 3 3 3 5 3 5 0 3 2 3 3 2 1 3 5 2 2 3 2 0 2 3 2 2 4 1 5 3
```

```
dat[, "Post"]  # Or I can use brackets
```

```
## [1] 2 5 3 3 3 5 3 5 0 3 2 3 3 2 1 3 5 2 2 3 2 0 2 3 2 2 4 1 5 3
```
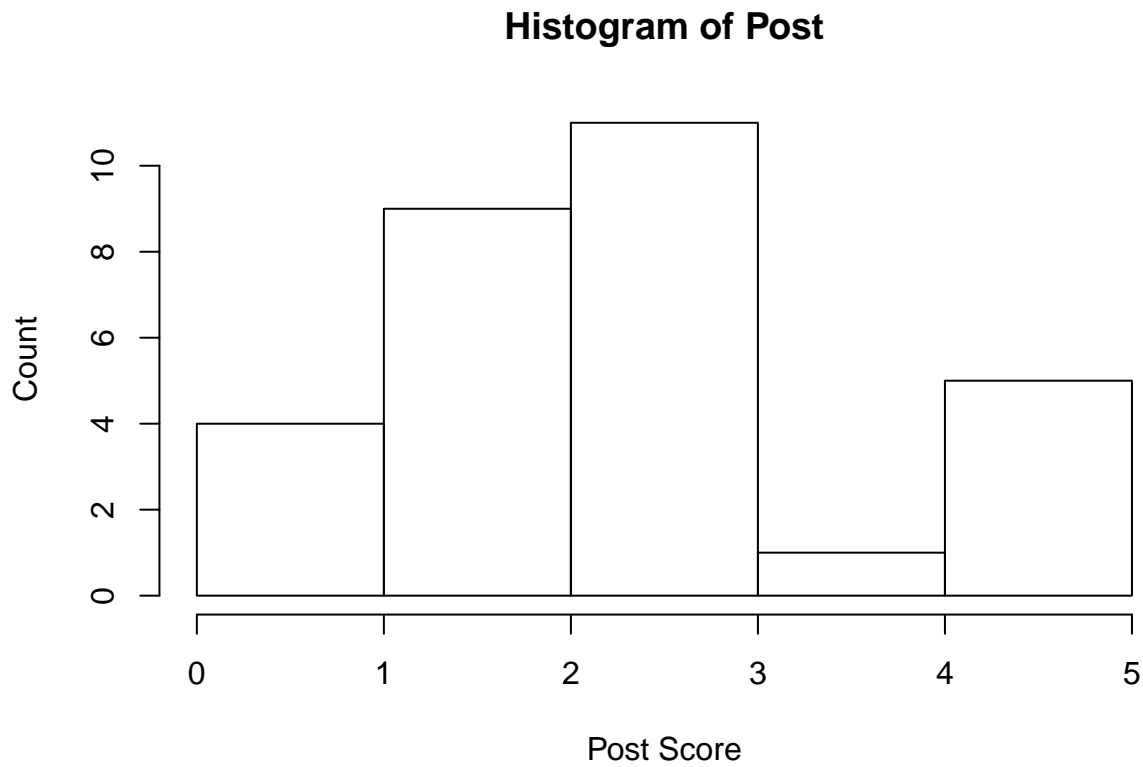
- To extract several rows

```
dat[2:5, ]                  # Observations 2, 3, 4, and 5
dat[1, ]                    # First observation / row
dat[nrow(dat), ]            # Last observation
dat[c(1, nrow(dat)), ]      # First and last observations
dat[c(1, nrow(dat)), "Post"]  # First and last observations score on Post
```
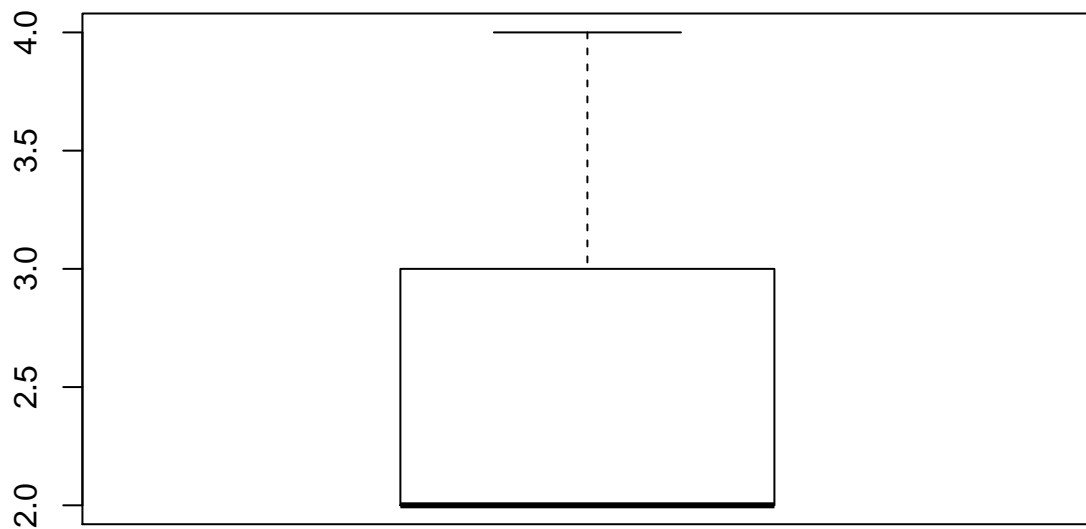
Plots

- Histograms

```
hist(dat[, "Post"], main = "Histogram of Post", xlab = "Post Score", ylab = "Count")
```
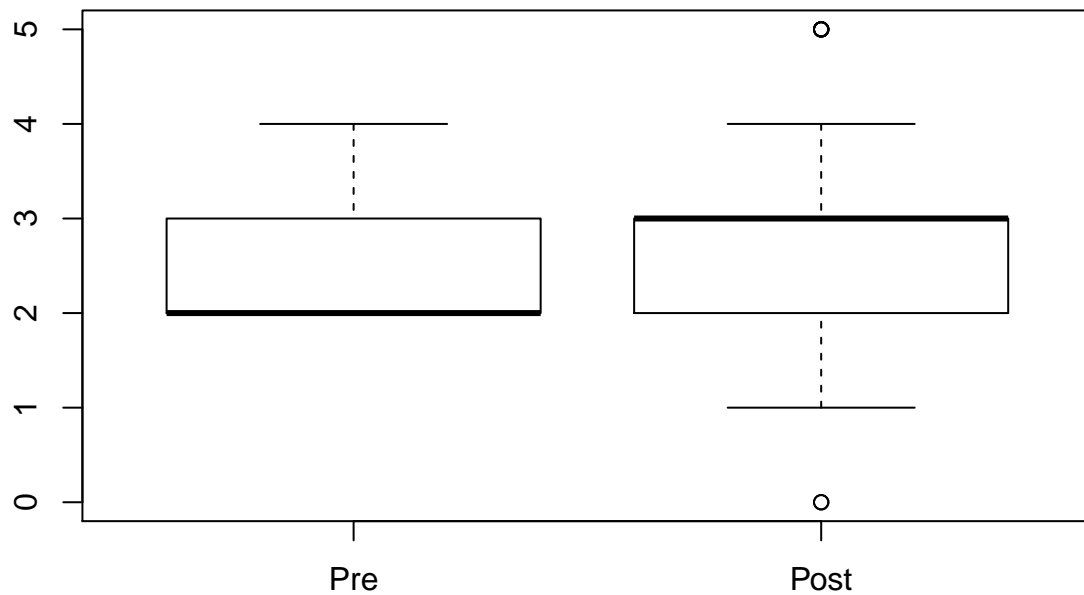
- Boxplots

```r
boxplot(dat$Pre)
```



```r
boxplot(dat[, c("Pre", "Post")])  # Note that I can select multiple columns at once
```

Other Useful Summary Statistics

```r
mean(dat$Post)          # Average Post score
apply(dat, 2, mean)     # Mean of each variable, does this make sense?

sd(dat$Post)            # Standard deviation of Post
apply(dat, 2, sd)       # SD of each variable

median(dat$Post)        # Median of Post
apply(dat, 2, median)   # Median of each variable

min(dat$Post)           # Minimum of Post
apply(dat, 2, min)      # Minimum of each variable

max(dat$Post)           # Maximum of Post
apply(dat, 2, max)      # Maximum of each variable
```

Homework for Sunday- Due 5PM

Load the `bfi` data from the `psych` package:

```r
library(psych)
data(bfi)

?bfi  # Description of the `bfi` data set
```

1. Show the first few rows of `bfi`

2. Show the last few rows of `bfi`

3. Compute the usual summary statistics for gender and age (min, max, mean, SD, median)

4. Attempt to run summary statistcs for education. Look at the data. Why might this not be working?

5. Make a new object called `bfi_agree` containing only the Agreeableness items (A1, A2, A3, A4, A5)

6. Examine (plot) the correlations of the Agreeableness items

7. Write two of your favorite random facts

8. Submit as an R-Markdown File

Summary of Key Functions

- Install a package/packages: `install.packages()`
- Load a package: `library()`
- Show the current working directory: `getwd()`
- Set the working directory: `setwd()`
- Read in a data set: `read.table()`, `read.csv()`, `foreign::read.spss()`, ...
- View the data frame: `View()`
- See first/last observations in your data frame: `head()`, `tail()`
- Describe the data: `summary()`, `psych::describe()`
- Histogram: `hist()`
- Scatterplot: `plot()`
- Boxplot: `boxplot()`