=210mm =297mm

Stat 145, Tue 14-Sep-2021 -- Tue 14-Sep-2021
Biostatistics
Spring 2021

Tuesday, September 14th 2021

Wk 3, Tu
Topic:: Correlation
Read:: Lock5 2.5

Warmup: "simple" tasks in R

1. make a list, calculate a mean, median, sd

2. make a relative frequency table
   tally(~ var, data=..., format="proportion")
   tally(~ var, data=..., format="percent")

3. make a two-way table of relative frequencies
   tally(~selfhandedness | sex, data=ssurv, format="proportion")
   tally(~ sex | selfhandedness, data=ssurv, format="proportion")
      each column adds up to 100%

   or
   prop.table(tally(~selfhandedness | sex))
   tally(~selfhandedness | sex) %>% prop.table()
      all cells combined add up to 100%

4. side-by-side
   boxplots of Sepal.Width broken down by Species
     gf_boxploth(Species ~ Sepal.Width, data=iris)
   histograms of same

Task for each student: To your myRstudioCheatSheet.Rmd file, add
personal helps on the use of these commands. Whenever it makes sense,
try these in univariate and bivariate settings.

- read.csv()
- head()
- help()
- dim()

```
names()
tally()
addmargins()
mean()
median()
sd()
favstats()
filter()
gf_boxplot()
    try replacing gf_boxplot() with gf_percents() and note the difference
gf_point()
gf_histogram()
    try replacing with gf_dhistogram() and note the difference
gf_density()
```

Associations – Read through, This should mesh well with things previously said concerning associations

- Requires *bivariate data*—i.e., two variables measured on the same subjects/units
- Usually come to think of one variable as explanatory and the other as response.
- Having an association means knowledge of the explanatory variable for a case makes you better informed (even just slightly) about the value of the response for that case.

  One of the main points of inferential statistics is to discern the real associations from the phantom ones.
- Pairings of variables can be
  - two categorical variables
  - one categorical variable, one quantitative

    In this case, it is usually the categorical one that serves as explanatory.
  - two quantitative variables

Q2: Write an R command.

1. If you had a (large) data frame whose variables included `ageCategory` (18 or younger, young adult 18–25, adult 25–65, senior) and `receivedShot` (Yes, No; indicates whether the person has received a Covid vaccine shot), what would a command that helped investigate an association between variables look like? Write one out.
2. If you wished to compare `waitTime` for individuals visiting the ER at one of the local hospitals (`hospital` variable has values Butterworth, Blodgett, and St. Mary's), write a command you could use to begin your investigation.

Tools when investigating associations between two quantitative variables include

- scatter plots

  Any *real*, non-horizontal-line pattern is indicative of an association

```
gf_point(weight ~ height, data=women)
```

- correlation

  A measure on how non-horizontal, linear the pattern is

````
---
title: "RStudio Cheat Sheet"
author: "T.Scofield"
date: '`r format(Sys.Date(), "%B %d, %Y")`'
output:
  pdf_document:
    fig_height: 2.2
    fig_width: 4
  html_document:
    fig_height: 2.2
    fig_width: 4
  word_document:
    fig_height: 2.2
    fig_width: 4
---

```{r, setup, include = FALSE, message=FALSE}
# load packages that are going to be used
library(mosaic)      # this loads ggformula (for plotting), etc. too
library(openintro)   # this loads data sets intro to modern stats
library(oibiostat)   # this loads data sets open intro biostats
library(pander)      # for tables

# Some customization. You can alter or delete as desired (if you know what you a
re doing).

theme_set(theme_bw())     # change theme for ggplot2/ggformula

knitr::opts_chunk$set(
  tidy = FALSE,     # display code as typed (rather than reformatted)
  size = "small")   # slightly smaller font for code
```

## Making a list of numbers, calculating statistics from it

Suppose I want things like the mean, sd, median, etc. for the
list of numbers 9, 11, 7, 13, 10.
```{r}
c(9, 11, 7, 13, 10) -> x
mean(~x)
```

## Tables
To make a frequency table
```{r}
ssurv <- read.csv("http://scofield.site/teaching/data/csv/ssurv.csv")
tally(~ selfhandedness, data=ssurv)
```

If I want a relative frequency table instead
```{r}
tally(~ selfhandedness, data=ssurv, format="proportion")
```

For bivariate data (i.e., two-way tables)
```{r}
tally(~ selfhandedness | sex, data=ssurv)
```

Adding the "format" switch
```{r}
tally(~ selfhandedness | sex, data=ssurv, format="percent")
````



````
```
gives us percentages out of the whole, done for each column.  If we want to
rse the roles of `sex` and `handedness`,
```{r}
tally(~ sex | selfhandedness, data=ssurv, format="percent")
```

Now, to make the total of all combined cells make up 100 percent,
```{r}
tally(~selfhandedness | sex, data=ssurv) %>% prop.table()
```

## Plotting bivariate data
In the `iris` data frame, there is a column (categorical) called
`Species` and another column (quantitative) called `Sepal.Width`.
If I want side-by-side boxplots for the quantitative variable
broken down by the categorical one
```{r}
gf_boxplot(~ Sepal.Width | Species, data=iris)
```
Now, try out this modification

```{r}
gf_boxplot(Species ~ Sepal.Width, data=iris)
```
````

RStudio Cheat Sheet

T.Scofield

September 14, 2021

Making a list of numbers, calculating statistics from it

Suppose I want things like the mean, sd, median, etc. for the list of numbers 9, 11, 7, 13, 10.

```
c(9, 11, 7, 13, 10) -> x
mean(~x)
```

```
## [1] 10
```

Tables

To make a frequency table

```
ssurv <- read.csv("http://scofield.site/teaching/data/csv/ssurv.csv")
tally(~ selfhandedness, data=ssurv)
```

```
## selfhandedness
##       L   R
##   1  31 248
```

If I want a relative frequency table instead

```
tally(~ selfhandedness, data=ssurv, format="proportion")
```

```
## selfhandedness
##                         L           R
## 0.003571429 0.110714286 0.885714286
```

For bivariate data (i.e., two-way tables)

```
tally(~ selfhandedness | sex, data=ssurv)
```

```
##               sex
## selfhandedness   F   M
##                  0   1
##              L  15  16
##              R 124 124
```

Adding the "format" switch

```
tally(~ selfhandedness | sex, data=ssurv, format="percent")
```

```
##               sex
## selfhandedness          F          M
##                 0.0000000  0.7092199
##              L 10.7913669 11.3475177
##              R 89.2086331 87.9432624
```

gives us percentages out of the whole, done for each column. If we want to reverse the roles of `sex` and `handedness`,

```
tally(~ sex | selfhandedness, data=ssurv, format="percent")
```

```
##    selfhandedness
## sex                 L        R
##   F   0.0000  48.3871  50.0000
##   M 100.0000  51.6129  50.0000
```

Now, to make the total of all combined cells make up 100 percent,

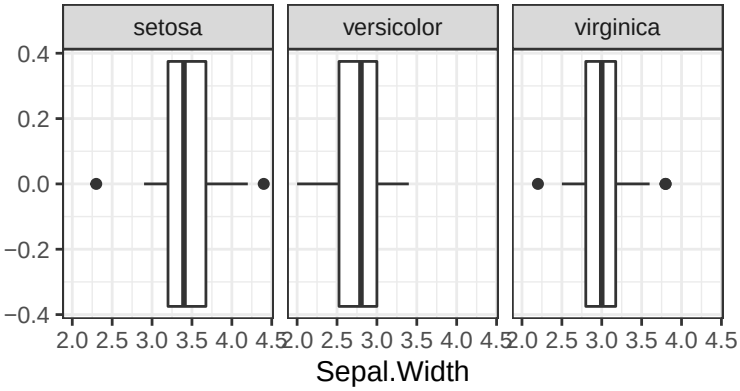
```
tally(~selfhandedness | sex, data=ssurv) %>% prop.table()
```

```
##                sex
## selfhandedness           F           M
##                0.000000000 0.003571429
##              L 0.053571429 0.057142857
##              R 0.442857143 0.442857143
```

Plotting bivariate data

In the `iris` data frame, there is a column (categorical) called `Species` and another column (quantitative) called `Sepal.Width`. If I want side-by-side boxplots for the quantitative variable broken down by the categorical one

```
gf_boxplot(~ Sepal.Width | Species, data=iris)
```



Now, try out this modification

```
gf_boxplot(Species ~ Sepal.Width, data=iris)
```