Using RStudio for the Plots of Sections 2.1-2.2

Exercise: Look at the information in Table 2.1, p. 47. Identify

(a) the cases

(b) variable(s) and variable type(s)

The data in this table is summarized. It is called a **frequency table**, and like any frequency table it provides the **distribution** of a variable—i.e., what are the *values* (sometimes called *levels*) of the variable, and how often they occur. Try to imagine the look of the spreadsheet containing the raw data—i.e., how it would appear when first entered, one case at a time.

While a number of datasets considered in the Lock5 textbook are made available in the **Lock5withR** package, the data of Table 2.1 is not among them. To generate it we learn a few commands. Type the following commands in RStudio, and observe the results.

```
rep(5, 2)
rep(2, 5)
rep("hi", 3)
c("hi", "hi", "hi")
c(rep("hi",3), rep("there",5))
```

Exercise: Discuss with a classmate

(a) what the `rep()` command does.

(b) how you might use `rep()` and `c()` to generate the raw data that is summarized in Table 2.1. Carry out your command, and give the name `oneTrueLove` to the resulting data.

If you have done part (b) of the exercise correctly, then by typing the following command you should see output much like mine displayed here.

```
head(oneTrueLove)

[1] "Agree" "Agree" "Agree" "Agree" "Agree" "Agree"
```

Displays of univariate categorical data

Our `oneTrueLove` data is consists of a single categorical variable "measured" on the cases (*univariate*, since there is no second variable that was measured). We give here several ways we might display the *distribution* of this variable using RStudio commands.

`Frequency table`. There are many commands that generate frequency tables, something it is good to be aware of if you happen to do a web search on frequency tables in R. The one we use primarily in this course is `tally()`. I give examples below, which you should type out yourself in order to observe the results.

```
tally(~oneTrueLove)
```

There are some additional options one may employ, such as

```
tally(~oneTrueLove, margins=TRUE)
tally(~oneTrueLove, format="proportion")
```

Bar chart. Once again, there are various commands, both native to R, and provided in add-on packages, which can produce bar charts. For many of our plots we will use commands provided by the **ggformula** package. If you have not already loaded that package, do so now, either checking the appropriate box off the packages tab, or by typing the command:

```
require(ggformula)
```

Then type

```
gf_bar(~oneTrueLove)
```

Again, there are various additional *switches*—additional options you can place in your commands to tweak the appearance of the figure:

```
gf_bar(~oneTrueLove, color="red", fill="navy")
gf_props(~oneTrueLove, fill="navy", color="red")     # a relative frequency bar chart
```

Often a dataset has multiple variables measured on the cases, resulting in what the R people call a **data frame**. The `NHANES` dataset, provided in the **NHANES** package, contains measurements on 10,000 human subjects collected in the National Health and Nutrition Survey.

Exercise: Load the **NHANES** package. Use the command

```
data(package = "NHANES")
```

to see what datasets are provided in this package. Then use commands such as `names()` and `str()` to see the sorts of variables which are stored in the `NHANES` data frame.

One of the many categorical variables in the `NHANES` data frame is `MaritalStatus`. To extract the distribution of that particular variable from this dataset, we can use commands like

```
tally(~MaritalStatus, data=NHANES)
gf_bar(~MaritalStatus, data=NHANES)
```

Displays of bivariate categorical data

Suppose we want to investigate the possibility of an association between two categorical variables. There are several natural methods.

Two-way table. Here we break down the distribution of one variable for each value of the other one. The two variables, `MaritalStatus` and `PhysActive`, both come from the `NHANES` data frame.

```
tally(~MaritalStatus | PhysActive, data=NHANES)
tally(~MaritalStatus | PhysActive, data=NHANES, margin=TRUE)
tally(~MaritalStatus + PhysActive, data=NHANES, margin=TRUE)    # slightly different output
```

Exercise: Compare the result from the last-typed command with that of

```
tally(  PhysActive | MaritalStatus, data=NHANES)
```

Discuss the differences with those around you.

Stacked and side-by-side bar charts. Compare the results of the following commands. The first generates a bar charts for the variable `PhysActive`, one for each level of `MaritalStatus`. The second uses the `file` option with the tilde character to produced a stacked bar chart conveying the same information. The third offers a different version of a side-by-side bar chart option.

```
gf_bar(~PhysActive|MaritalStatus, data=NHANES)
gf_bar(~PhysActive, fill=~MaritalStatus, data=NHANES)
gf_bar(~PhysActive, fill=~MaritalStatus, data=NHANES, position='dodge')
gf_bar(~MaritalStatus | PhysActive, data=NHANES)
gf_bar(~MaritalStatus, fill=~PhysActive, data=NHANES, position='dodge')
```

Which do you like better and/or find more easily read?

Exercise: The final two plot commands above reverse the rolls of the variables from the three previous plots. Discuss with a friend which of the two is more effective and why.

Displays of univariate quantitative data

If you have the **Lock5withR** package loaded, then you have access to the **MammalLongevity** data frame containing the data from Table 2.14. View it.

Histograms. A histogram is a visual display of the distribution of a quantitative variable. Try out the `gf_histogram()` command:

```
gf_histogram(~Longevity, data=MammalLongevity, color="red", bins=8)
```

Exercise: Repeat the last command several times, each time using a different number for the `bins` setting. Discuss with a friend just what this setting is for. Can you reproduce the figure at the bottom of p. 62?

Histograms are so often-used, it is not surprising that there are a number of commands available both in standard R, and in add-on packages, which produce them. A competing one (to `gf_histogram()`) found in the **lattice** package is `histogram()`. It allows one to indicate the locations of breaks between bins. Try out this version, loading **lattice** beforehand if necessary.

```
histogram(~Longevity, data=MammalLongevity, breaks=seq(0,40,5))
```

It should be a (near?) match for the figure on p. 62.

`Density plots`. The number of bins used for a histogram can affect the visual appearance/shape. To ascertain distribution shape, it may be helpful to, in a sense, blur ones eyes to the "blocky" structure of histogram bars. A density plot is helpful for this purpose. Type this command to see the result.

```
gf_density(~Longevity, data=MammalLongevity, fill="red")
```

One quantitative variable (response) and one categorical variable (explanatory)

The **`Lock5withR`** package supplies with a data frame called **`StudentSurvey`**, where the respondents this time are not students from Calvin College, but probably took a course from one of our textbook's authors. Look at this data frame, and note that it has both quantitative and categorical variables. One might wish to investigate whether there is a difference in `MathSAT` scores (response variable) across `gender` (explanatory variable). One usually begins to investigate this question graphically. The different possible plots using `gf_histogram()` are similar to the various bivariate options with `gf_bar()`:

```
gf_histogram(~GPA, fill=~Gender, data=StudentSurvey, color="red")   # stacked
gf_histogram(~GPA | Gender, data=StudentSurvey, color="red")   # side-by-side
```

`Exercises:`

1. Type out the command

   ```
   oneTrueLoveFull <- data.frame( opinion = c( rep("Agree",735),
   rep("Disagree",1812), rep("Don't know",78)), sex =
   c(rep("Male",372), rep("Female",363), rep("Male",807),
   rep("Female",1005),rep("Male",34),rep("Female",44)))
   ```

   to create a data frame with raw data like that summarized in Table 2.3. Then try to duplicate Tables 2.3 and 2.4 (that is, add margins). Finally, draw (using software) a barchart of the distribution.

2. Read in the file "ssurv.csv" from the url

   http://scofield.site/teaching/data/csv/ssurv.csv

   and graph two histograms on the variable `cds`, one that does not treat differently men and women, and a second producing a breakdown by sex.