Importing, working with, and exploring data
Week 2, Lecture 03

Richard E.W. Berl

Spring 2019

Contents

Loading data

From a package

For a full list of data sets included in the base `datasets` R package, use: `data()` or `library(help="datasets")`.

You can load a built-in data set like this:

```
data("HairEyeColor")
```

There is documentation available for each one.

```
?HairEyeColor
```

Now let's examine the data.

```
HairEyeColor
```

```
## , , Sex = Male
##
##        Eye
## Hair    Brown Blue Hazel Green
##   Black    32   11    10     3
##   Brown    53   50    25    15
##   Red      10   10     7     7
##   Blond     3   30     5     8
##
## , , Sex = Female
##
##        Eye
## Hair    Brown Blue Hazel Green
##   Black    36    9     5     2
##   Brown    66   34    29    14
##   Red      16    7     7     7
##   Blond     4   64     5     8
```

Whoa, this isn't a data frame. What is it?

```
class(HairEyeColor)
```

```
## [1] "table"
```

It's a three-way contingency table. This makes it easier to look at and is suitable for some analyses. But we can coerce it to a data frame, which will make it easier for us to work with:

```
hairEyeColor = as.data.frame(HairEyeColor)
hairEyeColor
```

```
##     Hair   Eye    Sex Freq
## 1  Black Brown   Male   32
## 2  Brown Brown   Male   53
## 3    Red Brown   Male   10
## 4  Blond Brown   Male    3
## 5  Black  Blue   Male   11
## 6  Brown  Blue   Male   50
## 7    Red  Blue   Male   10
## 8  Blond  Blue   Male   30
## 9  Black Hazel   Male   10
## 10 Brown Hazel   Male   25
## 11   Red Hazel   Male    7
## 12 Blond Hazel   Male    5
## 13 Black Green   Male    3
## 14 Brown Green   Male   15
## 15   Red Green   Male    7
## 16 Blond Green   Male    8
## 17 Black Brown Female   36
## 18 Brown Brown Female   66
## 19   Red Brown Female   16
## 20 Blond Brown Female    4
## 21 Black  Blue Female    9
## 22 Brown  Blue Female   34
## 23   Red  Blue Female    7
## 24 Blond  Blue Female   64
## 25 Black Hazel Female    5
## 26 Brown Hazel Female   29
## 27   Red Hazel Female    7
## 28 Blond Hazel Female    5
## 29 Black Green Female    2
## 30 Brown Green Female   14
## 31   Red Green Female    7
## 32 Blond Green Female    8
```

Much better. And since we renamed the data frame (to the "mixedCase" style), we can remove the old object:

```
rm(HairEyeColor)
```

From a CSV file

Visit: https://osf.io/s7d9d/

Read the description of the data set. The source of the data is:

- Weiss, A., et al. (2017). Personality in the chimpanzees of Gombe National Park. Scientific Data, 4, 170146. doi: 10.1038/sdata.2017.146

You can also click "Codebook for Gombe data personality variables.pdf" in the "Files" box of the OSF page to learn about how the data are coded.

In the “Files” box, click “gombe_128.csv” and then the “Download” button on the top right of the following page to download it. Place it in your class `/data` directory (or wherever you are placing your raw data for the course).

You can open the file (in Excel or a text editor) to see how it is formatted.

```r
gombe = read.csv(file="./data/gombe_128.csv", header=TRUE)
```

```r
head(gombe)
```

```
##   chimpcode sex  kasekela      dom      sol     impl     symp     stbl
## 1      E131   0 0.1428571 2.428571 3.857143 3.000000 5.571429 4.285714
## 2       P70   1 1.0000000 4.666667 3.333333 4.333333 4.666667 4.000000
## 3       G74   1 0.0000000 3.333333 3.166667 3.500000 5.500000 5.166667
## 4      A364   0 0.0000000 1.666667 1.333333 2.000000 2.666667 4.666667
## 5       B89   0 1.0000000 3.000000 4.666667 3.000000 4.333333 2.666667
## 6       G19   1 1.0000000 4.000000 2.666667 2.666667 3.333333 4.000000
##       invt     depd      soc    thotl     help     exct     inqs     decs
## 1 4.142857 4.285714 4.571429 1.857143 5.000000 3.714286 3.285714 4.571429
## 2 2.666667 4.666667 4.333333 2.333333 6.333333 4.000000 3.666667 6.666667
## 3 4.166667 5.666667 5.666667 2.833333 5.500000 3.666667 3.666667 4.833333
## 4 3.333333 2.666667 5.333333 2.000000 3.666667 3.333333 4.000000 4.333333
## 5 3.000000 5.000000 6.000000 3.000000 4.666667 3.000000 3.333333 4.000000
## 6 2.333333 5.000000 6.333333 3.000000 5.666667 2.666667 3.333333 4.644833
##       indv    reckl     sens     unem      cur     vuln     actv     pred
## 1 3.142857 2.000000 4.571429 2.714286 3.142857 3.000000 5.000000 3.428571
## 2 4.000000 4.333333 6.000000 2.666667 3.333333 4.666667 4.333333 5.333333
## 3 4.000000 3.000000 4.666667 3.500000 3.000000 3.666667 4.500000 3.166667
## 4 3.666667 2.333333 4.666667 2.666667 3.000000 3.000000 4.333333 4.000000
## 5 3.000000 3.000000 3.333333 2.666667 3.666667 4.000000 2.666667 3.666667
## 6 4.000000 4.000000 2.000000 3.333333 4.666667 5.000000 4.333333 4.000000
##       conv     cool    innov dominance extraversion conscientiousness
## 1 4.285714 5.285714 4.000000  3.571429     4.642857          4.809524
## 2 5.333333 3.666667 4.333333  4.888889     4.333333          4.222222
## 3 3.333333 4.833333 4.666667  3.500000     4.750000          4.222222
## 4 3.000000 4.333333 4.666667  3.777778     5.166667          5.222222
## 5 3.333333 5.333333 5.333333  3.333333     4.250000          4.555556
## 6 3.666667 4.333333 4.000000  3.881611     5.000000          4.444444
##   agreeableness neuroticism openness
## 1      5.047619    3.714286 3.642857
## 2      5.666667    4.000000 3.500000
## 3      5.222222    3.250000 3.875000
## 4      3.666667    3.333333 3.750000
## 5      4.111111    4.166667 3.833333
## 6      3.666667    3.333333 3.583333
```

From a tab-delimited file

Visit: http://www.randomservices.org/random/data/HorseKicks.html

Read the description of the data set.

At the bottom of the page, click the highlighted text “Horse-kick data” to download it. Place it in your class `/data` directory (or wherever you are placing your raw data for the course).

You can open the file to see how it is formatted.

```
horseKicks = read.table(file="./data/HorseKicks.txt", header=TRUE, sep="\t")
```

```
horseKicks
```

```
##    Year GC C1 C2 C3 C4 C5 C6 C7 C8 C9 C10 C11 C14 C15
## 1  1875  0  0  0  0  0  0  0  1  1  0   0   0   1   0
## 2  1876  2  0  0  0  1  0  0  0  0  0   0   0   1   1
## 3  1877  2  0  0  0  0  0  1  1  0  0   1   0   2   0
## 4  1878  1  2  2  1  1  0  0  0  0  0   1   0   1   0
## 5  1879  0  0  0  1  1  2  2  0  1  0   0   2   1   0
## 6  1880  0  3  2  1  1  1  0  0  0  2   1   4   3   0
## 7  1881  1  0  0  2  1  0  0  1  0  1   0   0   0   0
## 8  1882  1  2  0  0  0  0  1  0  1  1   2   1   4   1
## 9  1883  0  0  1  2  0  1  2  1  0  1   0   3   0   0
## 10 1884  3  0  1  0  0  0  0  1  0  0   2   0   1   1
## 11 1885  0  0  0  0  0  0  1  0  0  2   0   1   0   1
## 12 1886  2  1  0  0  1  1  1  0  0  1   0   1   3   0
## 13 1887  1  1  2  1  0  0  3  2  1  1   0   1   2   0
## 14 1888  0  1  1  0  0  1  1  0  0  0   0   1   1   0
## 15 1889  0  0  1  1  0  1  1  0  0  1   2   2   0   2
## 16 1890  1  2  0  2  0  1  1  2  0  2   1   1   2   2
## 17 1891  0  0  0  1  1  1  0  1  1  0   3   3   1   0
## 18 1892  1  3  2  0  1  1  3  0  1  1   0   1   1   0
## 19 1893  0  1  0  0  0  1  0  2  0  0   1   3   0   0
## 20 1894  1  0  0  0  0  0  0  0  1  0   1   1   0   0
```

From an Excel spreadsheet

Visit: https://royalsocietypublishing.org/doi/suppl/10.1098/rsos.150645

You can click "View Full Text" on the left side to read the article (or just the abstract) to learn about the data set.

Under the "Supplemental Material" heading, click the highlighted text "rsos150645supp1.xlsx" to download it. Place it in your class `/data` directory.

You can open the file to see how it is formatted.

```
install.packages("tidyverse")
# OR
install.packages("readxl")
```

```
library(readxl)
```

Documentation is available at: https://readxl.tidyverse.org/

```
folktales = read_xlsx(path="./data/rsos150645supp1.xlsx",
                      sheet=1, range="A2:JP52")
folktales
```

```
## # A tibble: 50 x 276
##    X__1  `300` `300A` `301` `301D` `302` `302B` `302C*` `303` `303A` `304`
##    <chr> <dbl>  <dbl> <dbl>  <dbl> <dbl>  <dbl>   <dbl> <dbl>  <dbl> <dbl>
##  1 Ital~     1      0     1      0     1      0       0     1      0     0
##  2 Ladin     1      0     1      0     1      0       0     1      0     1
##  3 Sard~     1      0     1      0     1      0       0     1      0     0
##  4 Wall~     1      0     1      0     0      0       0     1      0     0
##  5 Fren~     1      0     1      0     1      0       0     1      1     1
##  6 Span~     1      0     1      0     1      0       0     1      0     1
##  7 Port~     1      0     1      0     1      0       0     1      0     1
##  8 Cata~     1      0     1      0     1      0       0     1      0     0
##  9 Roma~     1      1     1      0     1      0       1     1      1     1
## 10 Welsh     0      0     0      0     0      0       0     0      0     0
## # ... with 40 more rows, and 265 more variables: `305` <dbl>, `306` <dbl>,
## #   `307` <dbl>, `310` <dbl>, `311` <dbl>, `311B*` <dbl>, `312` <dbl>,
## #   `312A` <dbl>, `312 C` <dbl>, `312D` <dbl>, `313` <dbl>, `313E*` <dbl>,
## #   `314` <dbl>, `314A` <dbl>, `314A*` <dbl>, `315` <dbl>, `315 A` <dbl>,
## #   `316` <dbl>, `317` <dbl>, `318` <dbl>, `321` <dbl>, `322*` <dbl>,
## #   `325` <dbl>, `325*` <dbl>, `325**` <dbl>, `326` <dbl>, `326A*` <dbl>,
## #   `326B*` <dbl>, `327` <dbl>, `327 A` <dbl>, `327B` <dbl>, `327C` <dbl>,
## #   `327D` <dbl>, `327F` <dbl>, `327G` <dbl>, `328` <dbl>, `328A` <dbl>,
## #   `328*` <dbl>, `328A*` <dbl>, `329` <dbl>, `330` <dbl>, `331` <dbl>,
## #   `332` <dbl>, `332C*` <dbl>, `333` <dbl>, `334` <dbl>, `335` <dbl>,
## #   `360` <dbl>, `361` <dbl>, `361*` <dbl>, `362*` <dbl>, `363` <dbl>,
## #   `365` <dbl>, `366` <dbl>, `368C*` <dbl>, `369` <dbl>, `400` <dbl>,
## #   `401A*` <dbl>, `402` <dbl>, `402*` <dbl>, `402A*` <dbl>, `403` <dbl>,
## #   `403C` <dbl>, `404` <dbl>, `405` <dbl>, `406` <dbl>, `407` <dbl>,
## #   `408` <dbl>, `409` <dbl>, `409A` <dbl>, `409A*` <dbl>, `409B*` <dbl>,
## #   `410` <dbl>, `410*` <dbl>, `411` <dbl>, `412` <dbl>, `413` <dbl>,
## #   `425` <dbl>, `425A` <dbl>, `425B` <dbl>, `425C` <dbl>, `425D` <dbl>,
## #   `425E` <dbl>, `425M` <dbl>, `425*` <dbl>, `426` <dbl>, `430` <dbl>,
## #   `431` <dbl>, `432` <dbl>, `433B` <dbl>, `434` <dbl>, `434*` <dbl>,
## #   `440` <dbl>, `441` <dbl>, `442` <dbl>, `444*` <dbl>, `449` <dbl>,
## #   `450` <dbl>, `451` <dbl>, `452B*` <dbl>, ...
```

```
folktales = as.data.frame(folktales)
folktales[1:5,1:15]
```

```
##        X__1 300 300A 301 301D 302 302B 302C* 303 303A 304 305 306 307 310
## 1   Italian   1    0   1    0   1    0      0   1    0   0   0   1   1   1
## 2     Ladin   1    0   1    0   1    0      0   1    0   1   0   1   1   0
## 3 Sardinian   1    0   1    0   1    0      0   1    0   0   0   0   1   1
## 4   Walloon   1    0   1    0   0    0      0   1    0   0   0   0   1   0
## 5    French   1    0   1    0   1    0      0   1    1   1   0   1   1   1
```

```
colnames(folktales)[1] = "society"
folktales[1:5,1:15]
```

```
##     society 300 300A 301 301D 302 302B 302C* 303 303A 304 305 306 307 310
## 1   Italian   1    0   1    0   1    0      0   1    0   0   0   1   1   1
## 2     Ladin   1    0   1    0   1    0      0   1    0   1   0   1   1   0
## 3 Sardinian   1    0   1    0   1    0      0   1    0   0   0   0   1   1
## 4   Walloon   1    0   1    0   0    0      0   1    0   0   0   0   1   0
## 5    French   1    0   1    0   1    0      0   1    1   1   0   1   1   1
```

```
folktales$society
```

```
##  [1] "Italian"       "Ladin"         "Sardinian"     "Walloon"
##  [5] "French"        "Spanish"       "Portuguese"    "Catalan"
##  [9] "Romanian"      "Welsh"         "Irish"         "Scottish"
## [13] "Luxembourgish" "German"        "Austrian"      "Flemish"
## [17] "Dutch"         "Frisian"       "English"       "Swedish"
## [21] "Norwegian"     "Danish"        "Faroese"       "Icelandic"
## [25] "Czech"         "Slovak"        "Lusatian"      "Polish"
## [29] "Byelorussian"  "Ukrainian"     "Russian"       "Bulgarian"
## [33] "Macedonian"    "Serbian"       "Croation"      "Slovenenian"
## [37] "Latvian"       "Lithuanian"    "Pakistani"     "Indian"
## [41] "Nepali"        "Gypsy"         "Tadzhik"       "Iranian"
## [45] "Kurdish"       "Afghan"        "Ossetian"      "Albanian"
## [49] "Greek"         "Armenian"
```

(pdf / Rmd)