Data Wrangling: The Portal Data Set

Luna L Sanchez Reyes

2023-03-02

1. Intro to the Portal data set

Homework: create an intro describing the location of the experiment and the different experimental treatments. Paper here https://esajournals.onlinelibrary.wiley.com/doi/full/10.1890/15-2115.1

2. Load the data

There are three different data sets, so to load them we need to create three data frames

```
surveys <- read.csv(file = "../data-raw/surveys.csv")
head(surveys)
```

```
##   record_id month day year plot_id species_id sex hindfoot_length weight
## 1         1     7  16 1977       2         NL   M              32     NA
## 2         2     7  16 1977       3         NL   M              33     NA
## 3         3     7  16 1977       2         DM   F              37     NA
## 4         4     7  16 1977       7         DM   M              36     NA
## 5         5     7  16 1977       3         DM   M              35     NA
## 6         6     7  16 1977       1         PF   M              14     NA
```

```
species <- read.csv(file = "../data-raw/species.csv")
head(species)
```

```
##   species_id            genus         species   taxa
## 1         AB       Amphispiza       bilineata   Bird
## 2         AH Ammospermophilus         harrisi Rodent
## 3         AS       Ammodramus      savannarum   Bird
## 4         BA          Baiomys         taylori Rodent
## 5         CB  Campylorhynchus brunneicapillus   Bird
## 6         CM      Calamospiza     melanocorys   Bird
```

```
plots <- read.csv(file = "../data-raw/plots.csv")
head(plots)
```

```
##   plot_id                 plot_type
## 1       1         Spectab exclosure
## 2       2                   Control
## 3       3  Long-term Krat Exclosure
## 4       4                   Control
## 5       5         Rodent Exclosure
## 6       6 Short-term Krat Exclosure
```

---

3. The dplyr package for data wrangling

Subset columns from a data frame with the function select()

```
head(surveys)
```

```
##   record_id month day year plot_id species_id sex hindfoot_length weight
## 1         1     7  16 1977       2         NL   M              32     NA
## 2         2     7  16 1977       3         NL   M              33     NA
## 3         3     7  16 1977       2         DM   F              37     NA
## 4         4     7  16 1977       7         DM   M              36     NA
## 5         5     7  16 1977       3         DM   M              35     NA
## 6         6     7  16 1977       1         PF   M              14     NA
```

```
surveys_subset <- select(surveys, month, day, year)
```

Create new variables from existing variables or transform existing variables with mutate()

The hindfoot_length variable is measured in mm. I want a new variable that stores hindfoot length in cm.

```
mutate(head(surveys), hindfoot_length_cm = hindfoot_length/10)
```

```
##   record_id month day year plot_id species_id sex hindfoot_length weight
## 1         1     7  16 1977       2         NL   M              32     NA
## 2         2     7  16 1977       3         NL   M              33     NA
## 3         3     7  16 1977       2         DM   F              37     NA
## 4         4     7  16 1977       7         DM   M              36     NA
## 5         5     7  16 1977       3         DM   M              35     NA
## 6         6     7  16 1977       1         PF   M              14     NA
##   hindfoot_length_cm
## 1                3.2
## 2                3.3
## 3                3.7
## 4                3.6
## 5                3.5
## 6                1.4
```

```
head(surveys)
```

```
##   record_id month day year plot_id species_id sex hindfoot_length weight
## 1         1     7  16 1977       2         NL   M              32     NA
## 2         2     7  16 1977       3         NL   M              33     NA
## 3         3     7  16 1977       2         DM   F              37     NA
## 4         4     7  16 1977       7         DM   M              36     NA
## 5         5     7  16 1977       3         DM   M              35     NA
## 6         6     7  16 1977       1         PF   M              14     NA
```

```
surveys_mutated <- mutate(surveys, hindfoot_length_cm = hindfoot_length/10)
head(surveys_mutated)
```

```
##   record_id month day year plot_id species_id sex hindfoot_length weight
## 1         1     7  16 1977       2         NL   M              32     NA
## 2         2     7  16 1977       3         NL   M              33     NA
## 3         3     7  16 1977       2         DM   F              37     NA
## 4         4     7  16 1977       7         DM   M              36     NA
## 5         5     7  16 1977       3         DM   M              35     NA
## 6         6     7  16 1977       1         PF   M              14     NA
##   hindfoot_length_cm
## 1                3.2
## 2                3.3
## 3                3.7
## 4                3.6
## 5                3.5
## 6                1.4
```

Sorting or ordering data with the function `arrange()`

If we want to order the data frame values based on the weight variable:

```
surveys_arranged <- arrange(surveys, weight)
head(surveys_arranged)
```

```
##   record_id month day year plot_id species_id sex hindfoot_length weight
## 1       218     9  13 1977       1         PF   M              13      4
## 2      4052     4   5 1981       3         PF   F              15      4
## 3      4290     4   6 1981       4         PF                  NA      4
## 4      5346     2  22 1982      21         PF   F              14      4
## 5      7084    11  22 1982       3         PF   F              16      4
## 6      8736    12   8 1983      19         RM   M              17      4
```

Order values in descendant order with the function `desc()`

```
surveys_arranged <- arrange(surveys, desc(weight))
head(surveys_arranged)
```

```
##   record_id month day year plot_id species_id sex hindfoot_length weight
## 1     33049    11  17 2001      12         NL   M              33    280
## 2     12871     5  28 1987       2         NL   M              32    278
## 3     15459     1  11 1989       9         NL   M              36    275
## 4      2133    10  25 1979       2         NL   F              33    274
## 5     12729     4  26 1987       2         NL   M              32    270
## 6     13114     7  26 1987       2         NL   M              NA    269
```

Filter values with the function `filter()`

Filter the data frame to keep rows with weight values that are equal to 4:

```
filter(surveys, weight == 4)
```

```
##    record_id month day year plot_id species_id sex hindfoot_length weight
## 1        218     9  13 1977       1         PF   M              13      4
## 2       4052     4   5 1981       3         PF   F              15      4
## 3       4290     4   6 1981       4         PF                  NA      4
## 4       5346     2  22 1982      21         PF   F              14      4
## 5       7084    11  22 1982       3         PF   F              16      4
## 6       8736    12   8 1983      19         RM   M              17      4
## 7       9790     1  19 1985      16         RM   F              16      4
## 8       9794     1  19 1985      24         RM   M              16      4
## 9       9799     1  19 1985      19         RM   M              16      4
## 10      9823     1  19 1985      23         RM   M              16      4
## 11      9853     1  19 1985      17          RM   M              16      4
## 12      9909     1  20 1985      15          RM   F              15      4
## 13      9937     2  16 1985      21          RM   M              16      4
## 14     10119     3  17 1985      10          RM   M              16      4
## 15     10439     5  24 1985       7          RM   M              16      4
## 16     28126     6  28 1998      15          PF   M              NA      4
## 17     29906    10  10 1999       4          PP   M              21      4
```

```
surveys_filtered <- filter(surveys, weight != 4)
head(surveys_filtered)
```

```
##   record_id month day year plot_id species_id sex hindfoot_length weight
## 1        63     8  19 1977       3         DM   M              35     40
## 2        64     8  19 1977       7         DM   M              37     48
## 3        65     8  19 1977       4         DM   F              34     29
## 4        66     8  19 1977       4         DM   F              35     46
## 5        67     8  19 1977       7         DM   M              35     36
## 6        68     8  19 1977       8         DO   F              32     52
```

```
surveys_filtered <- filter(surveys, weight > 200)
head(surveys_filtered)
```

```
##   record_id month day year plot_id species_id sex hindfoot_length weight
## 1       588     2  18 1978       2         NL   M              NA    218
## 2       646     2  20 1978      18         NL   M              32    228
## 3       655     3  11 1978       3         NL   M              32    232
## 4       825     4  10 1978      18         NL   M              NA    225
## 5       845     5   6 1978       2         NL   M              32    204
## 6       848     5   6 1978      22         NL   M              32    212
```

Filter with more complex conditions

I want values that have weight larger than 200 AND also are females

```
surveys_filtered <- filter(surveys, weight > 200, sex == "F")
head(surveys_filtered)
```

```
##   record_id month day year plot_id species_id sex hindfoot_length weight
## 1       875     5  17 1978       5         NL   F              33    212
## 2      1731     4  28 1979      12         NL   F              32    239
## 3      2081    10  24 1979      12         NL   F              32    211
## 4      2133    10  25 1979       2         NL   F              33    274
## 5      2247    11  18 1979      12         NL   F              33    217
## 6      2305     1  15 1980      12         NL   F              32    214
```

```
surveys_filtered <- filter(surveys, weight > 200 & sex == "F")
head(surveys_filtered)
```

```
##   record_id month day year plot_id species_id sex hindfoot_length weight
## 1       875     5  17 1978       5         NL   F              33    212
## 2      1731     4  28 1979      12         NL   F              32    239
## 3      2081    10  24 1979      12         NL   F              32    211
## 4      2133    10  25 1979       2         NL   F              33    274
## 5      2247    11  18 1979      12         NL   F              33    217
## 6      2305     1  15 1980      12         NL   F              32    214
```

Now, I want values that have weight larger than 200 OR are also females:

```
surveys_filtered <- filter(surveys, weight > 200 | sex == "F")
head(surveys_filtered)
```

Filtering `NA` values

`NA` is a special valie in R. We can't use logical statements with it, we have to use the `is.na()` function:

```
surveys_filtered <- filter(surveys, !is.na(weight))
head(surveys_filtered)
```

```
##   record_id month day year plot_id species_id sex hindfoot_length weight
## 1        63     8  19 1977       3         DM   M              35     40
## 2        64     8  19 1977       7         DM   M              37     48
## 3        65     8  19 1977       4         DM   F              34     29
## 4        66     8  19 1977       4         DM   F              35     46
## 5        67     8  19 1977       7         DM   M              35     36
## 6        68     8  19 1977       8         DO   F              32     52
```