joining-tables

Davinder Singh

2023-03-14

Load the three data sets that we are going to join, survey.csv, speices.csv, plot.csv

```
surveys <- read.csv(file = "../data-raw/surveys (1).csv")
species <- read.csv(file = "../data-raw/species.csv" )
plots <- read.csv(file = "../data-raw/plots.csv")
```

Why do we need to combine or join dat tables

homework: elaborate on this topic

How do we join data tables in R

There is a group function ‘-join()’ that allow us to combine two data tables using values on a shared column

there has to be a shared column, and we need three main arguments to run these functions, two data tables and one column name

The different function allow us to combine in a differnt ways. ‘inner_join’

```
inner_join(surveys, species, by = "species_id")
```

We can also run it using pipes:

```
surveys %>%
  inner_join(species, by = "species_id") -> joined_table
```

How can we explore our combined/joined head table?

We want to see differences between the two input tables to see difference in columns we can use ‘head()’ To see number of rows we can use ‘str()’

```
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
```

```
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
```

```
head(joined_table)
```

```
##   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
##         genus  species   taxa
## 1     Neotoma albigula Rodent
## 2     Neotoma albigula Rodent
## 3   Dipodomys merriami Rodent
## 4   Dipodomys merriami Rodent
## 5   Dipodomys merriami Rodent
## 6 Perognathus   flavus Rodent
```

```
str(species)
```

```
## ’data.frame':    54 obs. of  4 variables:
##  $ species_id: chr  "AB" "AH" "AS" "BA" ...
##  $ genus     : chr  "Amphispiza" "Ammospermophilus" "Ammodramus" "Baiomys" ...
##  $ species   : chr  "bilineata" "harrisi" "savannarum" "taylori" ...
##  $ taxa      : chr  "Bird" "Rodent" "Bird" "Rodent" ...
```

```
str(surveys)
```

```
## ’data.frame':    35549 obs. of  9 variables:
##  $ record_id      : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ month          : int  7 7 7 7 7 7 7 7 7 7 ...
##  $ day            : int  16 16 16 16 16 16 16 16 16 16 ...
##  $ year           : int  1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 ...
##  $ plot_id        : int  2 3 2 7 3 1 2 1 1 6 ...
##  $ species_id     : chr  "NL" "NL" "DM" "DM" ...
##  $ sex            : chr  "M" "M" "F" "M" ...
##  $ hindfoot_length: int  32 33 37 36 35 14 NA 37 34 20 ...
##  $ weight         : int  NA NA NA NA NA NA NA NA NA NA ...
```

```
str(joined_table)
```

```
## 'data.frame':	34786 obs. of  12 variables:
##  $ record_id      : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ month          : int  7 7 7 7 7 7 7 7 7 7 ...
##  $ day            : int  16 16 16 16 16 16 16 16 16 16 ...
##  $ year           : int  1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 ...
##  $ plot_id        : int  2 3 2 7 3 1 2 1 1 6 ...
##  $ species_id     : chr  "NL" "NL" "DM" "DM" ...
##  $ sex            : chr  "M" "M" "F" "M" ...
##  $ hindfoot_length: int  32 33 37 36 35 14 NA 37 34 20 ...
##  $ weight         : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ genus          : chr  "Neotoma" "Neotoma" "Dipodomys" "Dipodomys" ...
##  $ species        : chr  "albigula" "albigula" "merriami" "merriami" ...
##  $ taxa           : chr  "Rodent" "Rodent" "Rodent" "Rodent" ...
```

what happened with the number of rows in joined_table vs surveys?

It dropped the rows that did not have matching vlaues of species_id column

Excerise 1

```
plots %>%
  inner_join(surveys, by = "plot_id") %>%
  filter(plot_type == "Control") %>%
  head()
```

```
## Warning in inner_join(., surveys, by = "plot_id"): Each row in ‘x‘ is expected to match at most 1 ro
## i Row 1 of ‘x‘ matches multiple rows.
## i If multiple matches are expected, set ‘multiple = "all"‘ to silence this
##   warning.
```

```
##   plot_id plot_type record_id month day year species_id sex hindfoot_length
## 1       2   Control         1     7  16 1977         NL   M              32
## 2       2   Control         3     7  16 1977         DM   F              37
## 3       2   Control         7     7  16 1977         PE   F              NA
## 4       2   Control        18     7  16 1977         PP   M              22
## 5       2   Control        69     8  19 1977         PF   M              15
## 6       2   Control        72     8  19 1977         NL   M              31
##   weight
## 1     NA
## 2     NA
## 3     NA
## 4     NA
## 5      8
## 6     NA
```

Automate joining tables and other things with ‘intersect()’

Which species_id values are shared between the two data tabels

```
intersect(surveys$species_id, species$species_id)
```

```
##  [1] "NL" "DM" "PF" "PE" "DS" "PP" "SH" "OT" "DO" "OX" "SS" "OL" "RM" "SA" "PM"
## [16] "AH" "DX" "AB" "CB" "CM" "CQ" "RF" "PC" "PG" "PH" "PU" "CV" "UR" "UP" "ZL"
## [31] "UL" "CS" "SC" "BA" "SF" "RO" "AS" "SO" "PI" "ST" "CU" "SU" "RX" "PB" "PL"
## [46] "PX" "CT" "US"
```

To find shared columnss we use ‘colnames()’ function ## Excerise 2

```
colnames(surveys)
```

```
## [1] "record_id"       "month"           "day"             "year"
## [5] "plot_id"         "species_id"      "sex"             "hindfoot_length"
## [9] "weight"
```

```
colnames(species)
```

```
## [1] "species_id" "genus"      "species"    "taxa"
```

```
intersect(colnames(surveys), colnames(species))
```

```
## [1] "species_id"
```

```
colnames(plots)
```

```
## [1] "plot_id"   "plot_type"
```

```
colnames(surveys)
```

```
## [1] "record_id"       "month"           "day"             "year"
## [5] "plot_id"         "species_id"      "sex"             "hindfoot_length"
## [9] "weight"
```

```
intersect(colnames(plots), colnames(surveys))
```

```
## [1] "plot_id"
```

```
plots %>%
  inner_join(surveys, by = "plot_id") %>%
  filter(plot_type == "Rodent Exclosure") %>%
  head()
```

```
## Warning in inner_join(., surveys, by = "plot_id"): Each row in ‘x‘ is expected to match at most 1 ro
## i Row 1 of ‘x‘ matches multiple rows.
## i If multiple matches are expected, set ‘multiple = "all"‘ to silence this
##   warning.
##   plot_id        plot_type record_id month day year species_id sex
## 1       5 Rodent Exclosure        11     7  16 1977         DS   F
## 2       5 Rodent Exclosure        87     8  20 1977         PF   F
## 3       5 Rodent Exclosure        98     8  20 1977         DM   M
## 4       5 Rodent Exclosure       100     8  20 1977         DS   F
## 5       5 Rodent Exclosure       101     8  20 1977         DM   F
## 6       5 Rodent Exclosure       113     8  20 1977         PF   F
##   hindfoot_length weight
## 1              53     NA
## 2              11      9
## 3              38     40
## 4              54     NA
## 5              35     46
## 6              13      8
```

other join functions

'left_join()' retains all values from the first table, drops unmatching from second

'right_join()' drops values from the first table and retaining all values from second

'full_join()' keeps all values from both tables

Joining multiple data tables

can we '_join()' function on 3 or more table at same time? NO so we use a pipe on call the join function two or more times (as needed):

```
inner_join(surveys, species, by = "species_id") %>%
  inner_join(plots, by = "plot_id") %>%
  str()
```

```
## 'data.frame':    34786 obs. of  13 variables:
##  $ record_id      : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ month          : int  7 7 7 7 7 7 7 7 7 7 ...
##  $ day            : int  16 16 16 16 16 16 16 16 16 16 ...
##  $ year           : int  1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 ...
##  $ plot_id        : int  2 3 2 7 3 1 2 1 1 6 ...
##  $ species_id     : chr  "NL" "NL" "DM" "DM" ...
##  $ sex            : chr  "M" "M" "F" "M" ...
##  $ hindfoot_length: int  32 33 37 36 35 14 NA 37 34 20 ...
##  $ weight         : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ genus          : chr  "Neotoma" "Neotoma" "Dipodomys" "Dipodomys" ...
##  $ species        : chr  "albigula" "albigula" "merriami" "merriami" ...
##  $ taxa           : chr  "Rodent" "Rodent" "Rodent" "Rodent" ...
##  $ plot_type      : chr  "Control" "Long-term Krat Exclosure" "Control" "Rodent Exclosure" ...
```

Excerise 3

```r
inner_join(surveys, species, by = "species_id") %>%
  inner_join(plots, by = "plot_id") %>%
  filter(plot_type == "Long-term Krat Exclosure" | plot_type == "Control") %>%
  filter(taxa == "Rodent") %>%
  filter(!is.na(weight)) %>%
  select(year, genus, species, weight, plot_type) %>%
  str()
```

```
## 'data.frame':    19344 obs. of  5 variables:
##  $ year     : int  1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 ...
##  $ genus    : chr  "Dipodomys" "Dipodomys" "Dipodomys" "Dipodomys" ...
##  $ species  : chr  "merriami" "merriami" "merriami" "ordii" ...
##  $ weight   : int  40 29 46 52 8 22 7 22 8 41 ...
##  $ plot_type: chr  "Long-term Krat Exclosure" "Control" "Control" "Control" ...
```

Excerise 4

help on 3, 5 and 6 and ex 5

Ex 4 p.1

```r
inner_join(surveys, species, by = "species_id") %>%
  select(year, month, day, species_id, weight) %>%
  filter(species_id == "DO") %>%
  head()
```

```
##   year month day species_id weight
## 1 1977     8  19         DO     52
## 2 1977    10  17         DO     33
## 3 1977    10  17         DO     50
## 4 1977    10  17         DO     48
## 5 1977    10  17         DO     31
## 6 1977    10  18         DO     41
```

problem 2

Create a data frame with only data for species IDs “PP” and “PB” and for years starting in 1995, with the columns “year”, “species_id”, and “hindfoot_length”, with no missing values for “hindfoot_length”

```r
inner_join(surveys, species, by = "species_id") %>%
  inner_join(plots, by = "plot_id") %>%
  select(year,species_id, hindfoot_length) %>%
  filter(year <= "1995", !is.na(hindfoot_length)) %>%
  filter(species_id == "PP" | species_id == "PB") %>%
  head()
```

```
##   year species_id hindfoot_length
## 1 1977         PP              22
## 2 1977         PP              17
## 3 1977          PP               20
## 4 1977          PP               21
## 5 1977          PP               21
## 6 1977          PP               19
```

problem 3

Create a data frame with the average "hindfoot_length" for each "species_id" in each "year" with no null values.

```
surveys %>%
  filter(!is.na(hindfoot_length)) %>%
  group_by(year, species_id) %>%
  summarize( hindfoot_length = mean(hindfoot_length, na.rm = TRUE)) %>%
  head()
```

```
## 'summarise()' has grouped output by 'year'. You can override using the
## '.groups' argument.
```

```
## # A tibble: 6 x 3
## # Groups:   year [1]
##    year species_id hindfoot_length
##   <int> <chr>                <dbl>
## 1  1977 DM                    35.7
## 2  1977 DO                    33.5
## 3  1977 DS                    49.4
## 4  1977 NL                    32
## 5  1977 OL                    20
## 6  1977 OT                    19.7
```

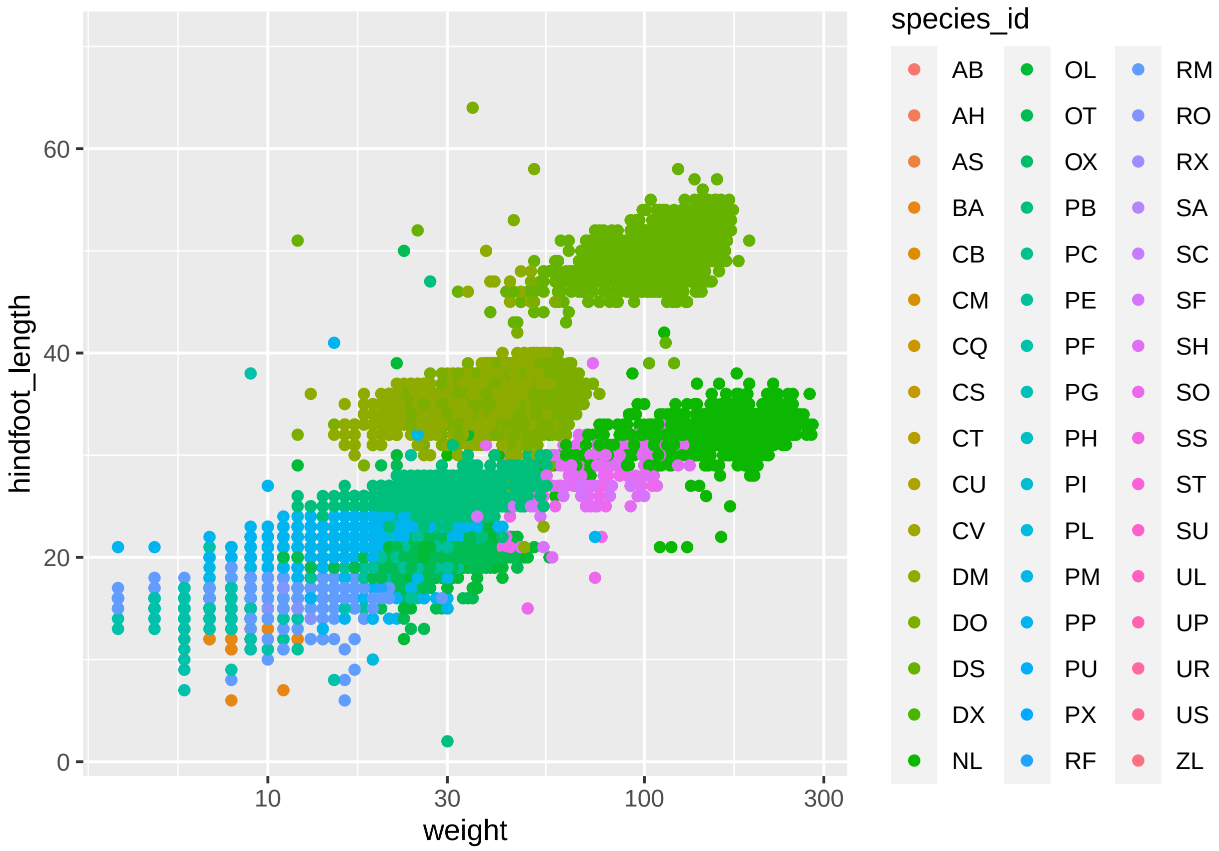
excerise 4

```
inner_join(surveys, species, by = "species_id") %>%
  inner_join(plots, by = "plot_id") %>%
  select(year, genus, species, weight, plot_type) %>%
  filter(genus == "Dipodomys") %>%
  head()
```

```
##   year     genus     species weight                plot_type
## 1 1977 Dipodomys    merriami     NA                  Control
## 2 1977 Dipodomys    merriami     NA        Rodent Exclosure
## 3 1977 Dipodomys    merriami     NA Long-term Krat Exclosure
## 4 1977 Dipodomys    merriami     NA        Spectab exclosure
## 5 1977 Dipodomys    merriami     NA        Spectab exclosure
## 6 1977 Dipodomys spectabilis     NA        Rodent Exclosure
```

```
inner_join(surveys, species, by = "species_id") %>%
  inner_join(plots, by = "plot_id") %>%
  ggplot(mapping = aes(x = weight, y = hindfoot_length)) +
  geom_point(mapping = aes(color = species_id)) +
  scale_x_log10()
```

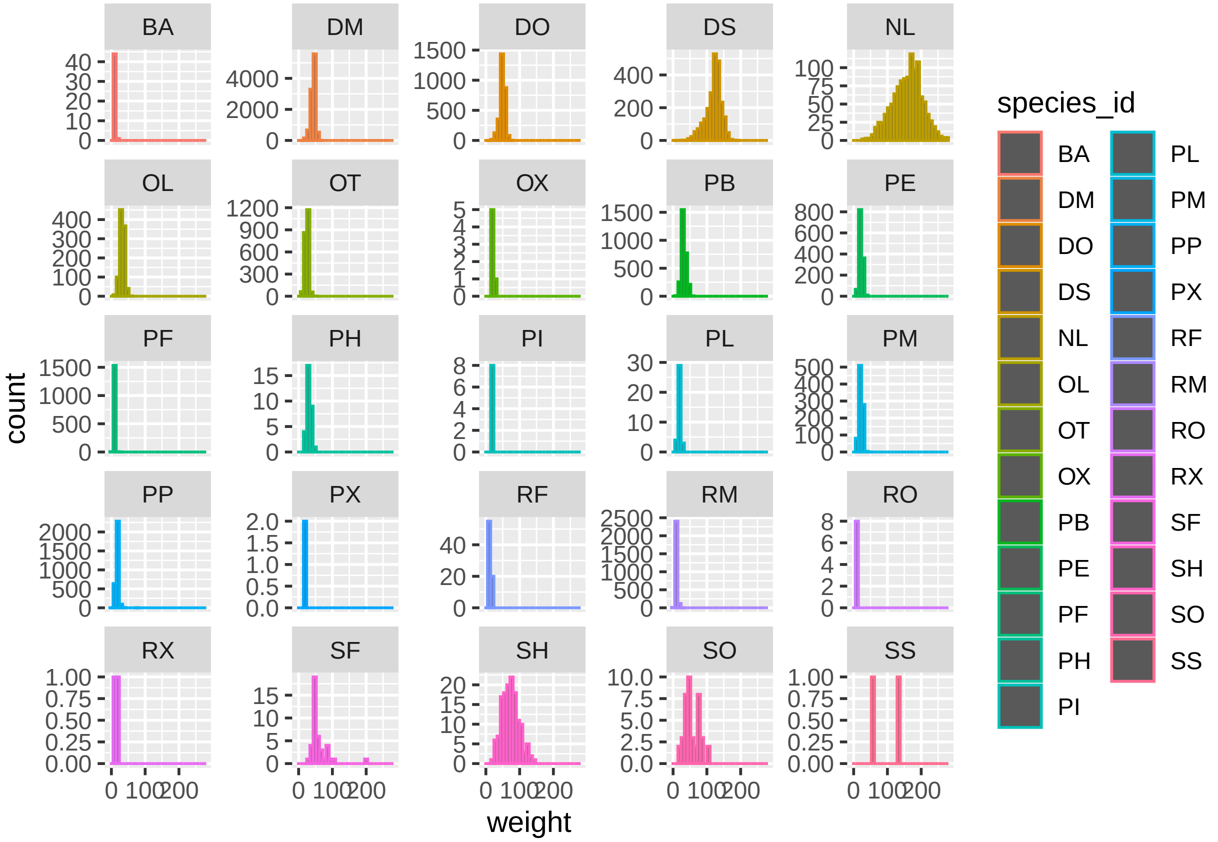
```
## Warning: Removed 4048 rows containing missing values (`geom_point()`).
```



### excerise 6 Make a histogram of weights with a separate subplot for each “species_id”. Do not include species with no weights. Set the “scales” argument to “free_y” so that the y-axes can vary. Include good axis labels. ?geom_histogram

```
inner_join(surveys, species, by = "species_id") %>%
  inner_join(plots, by = "plot_id") %>%
  filter(!is.na(weight)) %>%
  ggplot() +
  geom_histogram(mapping = aes(x = weight, color = species_id)) +
  facet_wrap(~species_id, scales = "free_y")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Excerise 5

?order The table should be sorted first by the species (so that each species is grouped together) and then by weight, with the largest weights at the top.

```
inner_join(surveys, species, by = "species_id") %>%
  inner_join(plots, by = "plot_id") %>%
  select(month, day, year, species_id, weight, hindfoot_length) %>%
  filter(!is.na(weight)) %>%
  arrange(species_id, desc(weight)) %>%
  head()
```

```
##   month day year species_id weight hindfoot_length
## 1     2  16 1991         BA     18              14
## 2     7  13 1991         BA     13              14
## 3     4  19 1991         BA     12              12
## 4     7  12 1991         BA     11              14
## 5     9   9 1991         BA     11               7
## 6     5  25 1990         BA     10              14
```

Homework Day 2 excerise 8

Import the shrub volume sites data and then combine it with both the data on shrub volume data and the experiments data to produce a single data frame that contains all of the data. ?inner_join

```
experiment <- read.csv(file = "../data-raw/shrub-volume-experiments (1).csv")
shrub_volume_data <- read.csv(file = "../data-raw/shrub-volume-data.csv")
shrub_volume_site <- read.csv(file = "../data-raw/shrub-volume-sites.csv")
inner_join(experiment, shrub_volume_data, by = "experiment") %>%
  inner_join(x = shrub_volume_site, y = ., by = "site")
```

```
## Warning in inner_join(experiment, shrub_volume_data, by = "experiment"): Each row in ‘x‘ is expected
## i Row 1 of ‘x‘ matches multiple rows.
## i If multiple matches are expected, set ‘multiple = "all"‘ to silence this
##   warning.
```

```
## Warning in inner_join(x = shrub_volume_site, y = ., by = "site"): Each row in ‘x‘ is expected to mat
## i Row 1 of ‘x‘ matches multiple rows.
## i If multiple matches are expected, set ‘multiple = "all"‘ to silence this
##   warning.
```

```
##    site latitude longitude elevation experiment manipulation length width
## 1     1    29.65    -82.32        54          1      control    2.2   1.3
## 2     1    29.65    -82.32        54          2         burn    2.1   2.2
## 3     1    29.65    -82.32        54          3      rainout    2.7   1.5
## 4     2    29.26    -82.42        50          1      control    3.0   4.5
## 5     2    29.26    -82.42        50          2         burn    3.1   3.1
## 6     2    29.26    -82.42        50          3      rainout    2.5   2.8
## 7     3    29.80    -82.15        57          1      control    1.9   1.8
## 8     3    29.80    -82.15        57          2         burn    1.1   0.5
## 9     3    29.80    -82.15        57          3      rainout    3.5   2.0
## 10    4    29.99    -82.62        62          1      control    2.9   2.7
## 11    4    29.99    -82.62        62          2         burn    4.5   4.8
## 12    4    29.99    -82.62        62          3      rainout    1.2   1.8
##    height
## 1     9.6
## 2     7.6
## 3     2.2
## 4     1.5
## 5     4.0
## 6     3.0
## 7     4.5
## 8     2.3
## 9     7.5
## 10    3.2
## 11    6.5
## 12    2.7
```

```
intersect(colnames(shrub_volume_data), colnames(experiment))
```

```
## [1] "experiment"
```

```
intersect(colnames(shrub_volume_site), colnames(shrub_volume_data))
```

```
## [1] "site"
```

excerise 10

A vector of shrub lengths A vector of the volume of each of the shrubs A data frame with just the shrubID and height columns A data frame with the second row of the full data frame A data frame with the first 5 rows of the full data frame

```
label <-read.csv(file = "../data-raw/shrub-dimensions-labeled.csv")
label$length
```

```
##  [1] 2.2 2.1 2.7 3.0 3.1 2.5 1.9 1.1 3.5 2.9
```

```
volume = label$length * label$width * label$height
data.frame(label$shrubID, label$height )
```

```
##    label.shrubID label.height
## 1             a1          9.6
## 2             a2          7.6
## 3             b1          2.2
## 4             b2          1.5
## 5             c1          4.0
## 6             c2          3.0
## 7             d1          4.5
## 8             d2          2.3
## 9             e1          7.5
## 10            e2          3.2
```

```
label[2,]
```

```
##   shrubID length width height
## 2      a2    2.1   2.2    7.6
```

```
label[c(1, 2, 3, 4, 5),]
```

```
##   shrubID length width height
## 1      a1    2.2   1.3    9.6
## 2      a2    2.1   2.2    7.6
## 3      b1    2.7   1.5    2.2
## 4      b2    3.0   4.5    1.5
## 5      c1    3.1   3.1    4.0
```