

R Training 4

Rachel Kenny

5/7/2020

Loading packages & data

```
# Load packages
library(readxl)

## Warning: package 'readxl' was built under R version 3.4.4

library(readr)

## Warning: package 'readr' was built under R version 3.4.4

library(tidyverse)

## Warning in as.POSIXlt.POSIXct(Sys.time()): unknown timezone 'zone/tz/2019c.1.0/
## zoneinfo/America/New_York'

## -- Attaching packages ----- tidyverse 1.3.0 --

## v ggplot2 3.3.0      v dplyr   0.8.5
## v tibble  3.0.1      v stringr 1.4.0
## v tidyr   1.0.2      v forcats 0.5.0
## v purrr   0.3.3

## Warning: package 'stringr' was built under R version 3.4.4

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(janitor)

## Warning: package 'janitor' was built under R version 3.4.4

# Load agua chinon veg data
ac_data_raw <- read_excel("OCWR_AC_2019_Data.xlsx")
ac_data <- clean_names(ac_data_raw)

# Load weir oak restoration data
oak_data_raw <- read_csv("Weir_Oak_Restoration_Data_winter19_2.csv")

## Parsed with column specification:
## cols(
##   `Short ID` = col_character(),
##   Survival = col_logical(),
##   Quantity = col_double(),
##   `Height (cm)` = col_double(),
##   `Open Closed` = col_character(),
```

```
## `Location UML` = col_character(),
## `Water Yes No` = col_character(),
## `Sampling Group` = col_character()
## )

oak_data <- clean_names(oak_data_raw)

# Oak data 1
oak_data_old <- read_csv("Weir_Oak_Restoration_Data_winter19.csv")

## Parsed with column specification:
## cols(
##   short_id = col_character(),
##   survival = col_logical(),
##   quantity = col_double(),
##   height_cm = col_double(),
##   open_closed = col_character(),
##   location_uml = col_character(),
##   water_yes_no = col_character(),
##   sampling_group = col_character()
## )

oak_data_old <- clean_names(oak_data_old)
```

Subsetting data

- `my_df[1:3]` (no comma) will subset `my_df`, returning the first three columns as a data frame.
- `my_df[1:3,]` (with comma, numbers to left of the comma) will subset `my_df` and return the first three rows as a data frame
- `my_df[, 1:3]` (with comma, numbers to right of the comma) will subset `my_df` and return the first three columns as a data frame, the same as `my_df[1:3]`

Common operators

- `[*]` multiply
- `[/]` divide
- `[>]` greater than
- `[>=]` greater than or equal to
- `[<]` less than
- `[<=]` less than or equal to
- `[=]` equal to
- `[==]` find a match in the data (for example, when filtering)

More on subsetting

```
# To return rows 1-3
ac_data[1:3]

## # A tibble: 1,732 x 3
##   desired_habitat      polygon_id    transect
##   <chr>            <chr>         <dbl>
```

```
## 1 Weedy Control      Weedy Control      1
## 2 Weedy Control      Weedy Control      1
## 3 Weedy Control      Weedy Control      1
## 4 Weedy Control      Weedy Control      1
## 5 Weedy Control      Weedy Control      1
## 6 Weedy Control      Weedy Control      1
## 7 Mulefat Scrub/sage Scrub OW-M7        1
## 8 Mulefat Scrub/sage Scrub OW-M7        1
## 9 Mulefat Scrub/sage Scrub OW-M7        1
## 10 Mulefat Scrub/sage Scrub OW-M7       1
## # ... with 1,722 more rows
```

```
# To return columns 1-3
```

```
ac_data[1:3,]
```

```
## # A tibble: 3 x 10
```

```
##   desired_habitat polygon_id transect pin_number data_type species_code
```

```
##   <chr>           <chr>         <dbl>    <dbl> <chr>    <chr>
```

```
## 1 Weedy Control  Weedy Con~         1         NA A.Belt  SOLSPP
```

```
## 2 Weedy Control  Weedy Con~         1         NA A.Belt  AMBACA
```

```
## 3 Weedy Control  Weedy Con~         1         NA A.Belt  AMBPSI
```

```
## # ... with 4 more variables: scientific_name <chr>, native_non_native <chr>,
```

```
## #   functional_group <chr>, layer <chr>
```

```
# To return rows 1-3 (same as first option)
```

```
ac_data[,1:3]
```

```
## # A tibble: 1,732 x 3
```

```
##   desired_habitat      polygon_id      transect
```

```
##   <chr>              <chr>         <dbl>
```

```
## 1 Weedy Control      Weedy Control      1
```

```
## 2 Weedy Control      Weedy Control      1
```

```
## 3 Weedy Control      Weedy Control      1
```

```
## 4 Weedy Control      Weedy Control      1
```

```
## 5 Weedy Control      Weedy Control      1
```

```
## 6 Weedy Control      Weedy Control      1
```

```
## 7 Mulefat Scrub/sage Scrub OW-M7        1
```

```
## 8 Mulefat Scrub/sage Scrub OW-M7        1
```

```
## 9 Mulefat Scrub/sage Scrub OW-M7        1
```

```
## 10 Mulefat Scrub/sage Scrub OW-M7       1
```

```
## # ... with 1,722 more rows
```

```
# Single brackets extract a list
```

```
ac_data[5]
```

```
## # A tibble: 1,732 x 1
```

```
##   data_type
```

```
##   <chr>
```

```
## 1 A.Belt
```

```
## 2 A.Belt
```

```
## 3 A.Belt
```

```
## 4 A.Belt
```

```
## 5 A.Belt
```

```
## 6 A.Belt
```

```
## 7 T.PI
```

```
## 8 T.PI
```

```
## 9 T.PI
## 10 T.PI
## # ... with 1,722 more rows
```

```
# Double brackets extract the items in a list
ac_data[[5]]
```

```
## [1] "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "T.PI" "T.PI"
## [9] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [17] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [25] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [33] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [41] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [49] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [57] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [65] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [73] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [81] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [89] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [97] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [105] "T.PI" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt"
## [113] "A.Belt" "A.Belt" "A.Belt" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [121] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [129] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [137] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [145] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [153] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [161] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [169] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [177] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [185] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "A.Belt"
## [193] "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt"
## [201] "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt"
## [209] "A.Belt" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [217] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [225] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [233] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [241] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [249] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [257] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [265] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [273] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [281] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [289] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [297] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [305] "T.PI" "T.PI" "T.PI" "T.PI" "A.Belt" "A.Belt" "A.Belt" "A.Belt"
## [313] "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "A.Belt" "T.PI" "T.PI"
## [321] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [329] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [337] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [345] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [353] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [361] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [369] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [377] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
```

[illegible]

[illegible]

[illegible]

```
## [1681] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [1689] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [1697] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [1705] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [1713] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [1721] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "A.Belt" "A.Belt" "A.Belt"
## [1729] "A.Belt" "A.Belt" "A.Belt" "A.Belt"
```

```
# Double brackets are equivalent to using the "$" operator
ac_data$data_type
```

[illegible]

[illegible]

[illegible]

[illegible]

```
## [1649] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [1657] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [1665] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [1673] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [1681] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [1689] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [1697] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [1705] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [1713] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI"
## [1721] "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "T.PI" "A.Belt" "A.Belt" "A.Belt"
## [1729] "A.Belt" "A.Belt" "A.Belt" "A.Belt"
```

Basic statistics

```
# Mean
mean(oak_data$height_cm)
```

```
## [1] 8.627288
```

```
# Median
median(oak_data$height_cm)
```

```
## [1] 9
```

```
# Standard deviation
sd(oak_data$height_cm)
```

```
## [1] 8.833833
```

```
# Highest number
max(oak_data$height_cm)
```

```
## [1] 63
```

```
# Lowest number
min(oak_data$height_cm)
```

```
## [1] 0
```

```
# Summarize data
summary(oak_data)
```

```
##      short_id      survival      quantity      height_cm
## Length:1202      Mode :logical  Min.   : 0.000  Min.   : 0.000
## Class :character FALSE:525      1st Qu.: 0.000  1st Qu.: 0.000
## Mode  :character TRUE :677      Median : 1.000  Median : 9.000
##                                     Mean   : 0.891  Mean   : 8.627
##                                     3rd Qu.: 1.000  3rd Qu.:15.000
##                                     Max.   :10.000  Max.   :63.000
## open_closed      location_uml      water_yes_no      sampling_group
## Length:1202      Length:1202      Length:1202      Length:1202
## Class :character Class :character Class :character Class :character
## Mode  :character Mode  :character Mode  :character Mode  :character
##
##
##
```

Basic statistics - troubleshooting

```
# Mean
mean(oak_data_old$height_cm)

## [1] NA
# If there are NA values, then introduce na.rm = TRUE to exclude those values from analysis
mean(oak_data_old$height_cm, na.rm = TRUE)

## [1] 9.313383
# Standard deviation
sd(oak_data_old$height_cm, na.rm = TRUE)

## [1] 25.3822
# Highest number
max(oak_data_old$height_cm, na.rm = TRUE)

## [1] 834
# Lowest number
min(oak_data_old$height_cm, na.rm = TRUE)

## [1] 0
# Identify problem rows
which(is.na(oak_data_old$height_cm)) #819

## [1] 819
which(grepl(834, oak_data_old$height_cm)) #933

## [1] 933
# Edit data
# Method 1
oak_data_new <- oak_data_old[-c(819, 933), ]
View(oak_data_new)

# Method 2
oak_data_new <- oak_data_old[ !(oak_data_old$short_id %in% c("U_C_33_1", "U_C_04_3")), ]
View(oak_data_new)

# It is always safer to delete data according to some sort of identifying value than to delete by row n
```