Lesson 2: Base R vs Tidy R - Homework

Instructor: Emily Markowitz (Emily.Markowitz@noaa.gov)

January 25, 2021

Contents

Questions:

1 Let's explore Tidyverse!

1.1 Think of Tidyverse as a family of packages. Which packages are loaded with {tidyverse}? What do each of these packages do? Check out: https://tidyverse.tidyverse.org/

1.2 Which package is `pivot_wider` from?

1.3 Which package is `rename` from?

1.4 Can you use {tidyverse} without {base} R?

2 Let's play with some data!

You can view the dataset CO2 in more detail using `View(CO2)` and learn about it using `?CO2`. CO2 comes from the {datasets} package which should already be automatically loaded in your R.

Some info about the CO2 dataset: "The CO2 data frame has 84 rows and 5 columns of data from an experiment on the cold tolerance of the grass species *Echinochloa crus-galli*."

```r
# Note: This function or data set name (in this case, data set)
# may occur in other packages so here I am using the "::" to say I
# specifically want the data 'CO2' from {datasets}.
CO2<-data.frame(datasets::CO2)
head(CO2)
```

Plant	Type	Treatment	conc	uptake
Qn1	Quebec	nonchilled	95	16.0
Qn1	Quebec	nonchilled	175	30.4
Qn1	Quebec	nonchilled	250	34.8
Qn1	Quebec	nonchilled	350	37.2
Qn1	Quebec	nonchilled	500	35.3
Qn1	Quebec	nonchilled	675	39.2

2.1 `rename()` the "conc" column to "Concenctration mL/L" and "Treatment" column to "Condition". The new name for the conc column is not a great name (dare I say 'tidy' name?) so we'll fix that in the next question. Assign your object here as a new obect (name up to you!).

2.2 Use the {janitor} function `clean_names` on the new CO2 data you just created in 2a. What does it do? How did {janitor} change our "Concenctration mL/L" column?

Again, assign your object here as a new obect (name up to you!).

2.3 Use `pivot_wider` make columns of uptake (values_from) for each plant (names_from) in your new data set from question 2b.

This is not a 'tidy' way of looking at data, but is good practice! Assign your object here as a new obect (name up to you!).

2.4 Use `pivot_longer` to undo what you did in 2c using the data that you created in 2c.

To see how to get the old names back, check out the `names_to` and `values_to` variable in `?pivot_longer`. This will likely incure some new rows with `NA`s, so you'll need to remove that here with `values_drop_na`. You can check if you actually got it back to origional form by seeing if the dimensions of the data.frame are the same as the origional dataset. As stated earlier, `dim(datasets::CO2)` was 84 rows and 5 columns.