Assignment 2 - Data Wrangling

Abrar Faruque

Oct 6th 2025

Assignment 2 - Data Wrangling

1. Create a vector v with all integers 0-30, and a vector w with every third integer in the same range (Hint: you will need the `seq()` function).

a) What is the difference in lengths of the vectors v and w? [0.25]
b) Create a new vector, v_square, with the square of elements at indices 3, 6, 7, 10, 15, 22, 23, 24, and 30 from the variable v. Hint: use indexing. Calculate the mean and median of the first five values from v_square. [0.5]
c) Create a boolean vector v_bool, indicating which vector v elements are bigger than 20. How many values are over 20? Hint: In R, TRUE = 1, and FALSE = 0, so you can use simple arithmetic to find this out. [1]

```
v <- (0:30)
w <- seq(0,30,3)
print(v)
```

```
##  [1]  0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
## [26] 25 26 27 28 29 30
```

```
print(w)
```

```
##  [1]  0  3  6  9 12 15 18 21 24 27 30
```

```
#1a)
print(length(v)-length(w)) #the difference between length v and w is 20
```

```
## [1] 20
```

```
#1b)
v_square <- c(v[3],v[6],v[7],v[10],v[15],v[22],v[23],v[24],v[30])
v_square <- v_square*v_square
print(v_square)
```

```
## [1]   4  25  36  81 196 441 484 529 841
```

```r
print(mean(v_square[1:5])) #mean value of first 5 elements is 68.4
```

```
## [1] 68.4
```

```r
print(median(v_square[1:5])) #median value of first 5 elements is 36
```

```
## [1] 36
```

```r
#1c)
v_bool <- v>20
sum(v_bool) # there are 10 numbers greater than 20 in v_bool
```

```
## [1] 10
```

2. Use the `CO2` dataset that pre-exists in R. Use the code below to get information about this dataset.

```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v readr     2.1.5
## v ggplot2   4.0.0     v stringr   1.5.2
## v lubridate 1.9.4     v tibble    3.3.0
## v purrr     1.1.0     v tidyr     1.3.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
data(CO2)
```

a) Create a summary table, grouped by the plant and the treatment, that includes the mean, standard deviation, and number of data rows. Is this dataset balanced (i.e., are there the same number of observations for each treatment)? [0.5]

b) Using the summarised table from part a), reorder the table from largest to smallest mean uptake. What plant and treatment has the highest and lowest mean uptake? [0.5]

c) Tidy up the data by making the dataset longer, re-formatting the ambient concentration and uptake into a new column called `measurement` with the values in a new column called `values`. Call this new dataset `CO2_long`. What do you notice about the average ambient concentration, as measured for each treatment group (i.e., grouped by plant, treatment, and the measurement)? [1]

d) Using this new longer format dataset as created in c), filter cases where the treatment is nonchilled, and the measurement is uptake. Then, add a new column called `values2`, and change the uptake measurement by a multiple of 1.3. Save this into a new object called CO2_new. [1]

e) Compare your `CO2_new` with the object below `CO2_new2`. Why are these two objects different? What does this tell you about this new function `ifelse()`? That is, how does `ifelse()` work? Hint: answer in terms of the arguments in this function. [1]

```
#2a)
CO2_summary_table <- CO2 %>%
  group_by(Plant,Treatment) %>%
  summarise(
            mean_conc = mean(conc),
            mean_uptake = mean(uptake),
            SD_conc = sd(conc),
            SD_uptake = sd(uptake),
            nrow_conc = length(conc),
            nrow_uptake = length(uptake),
            .groups = "keep"
  )
print(CO2_summary_table)
```

```
## # A tibble: 12 x 8
## # Groups:   Plant, Treatment [12]
##    Plant Treatment mean_conc mean_uptake SD_conc SD_uptake nrow_conc nrow_uptake
##    <ord> <fct>         <dbl>       <dbl>   <dbl>     <dbl>     <int>       <int>
##  1 Qn1   nonchill~       435        33.2    318.      8.21         7           7
##  2 Qn2   nonchill~       435        35.2    318.     11.0          7           7
##  3 Qn3   nonchill~       435        37.6    318.     10.3          7           7
##  4 Qc1   chilled         435        30.0    318.      8.33         7           7
##  5 Qc3   chilled         435        32.6    318.     10.3          7           7
##  6 Qc2   chilled         435        32.7    318.     11.3          7           7
##  7 Mn3   nonchill~       435        24.1    318.      6.48         7           7
##  8 Mn2   nonchill~       435        27.3    318.      7.65         7           7
##  9 Mn1   nonchill~       435        26.4    318.      8.69         7           7
## 10 Mc2   chilled         435        12.1    318.      2.19         7           7
## 11 Mc3   chilled         435        17.3    318.      3.05         7           7
## 12 Mc1   chilled         435        18      318.      4.12         7           7
```

```
#Yes, dataset is balanced; all plants and treatments have 7 observations each
#2b)
CO2_summary_table_ordered <- CO2_summary_table %>%
  filter (mean_uptake==max(mean_uptake))

print(c(head(CO2_summary_table_ordered,1)$Plant,head(CO2_summary_table_ordered,1)$Treatment))
```

```
## [1] Qn1        nonchilled
## 14 Levels: Qn1 Qn2 Qn3 Qc1 Qc3 Qc2 Mn3 Mn2 Mn1 Mc2 Mc3 Mc1 ... chilled
```

```r
print(c(tail(CO2_summary_table_ordered,1)$Plant,tail(CO2_summary_table_ordered,1)$Treatment))
```

```
## [1] Mc1     chilled
## 14 Levels: Qn1 Qn2 Qn3 Qc1 Qc3 Qc2 Mn3 Mn2 Mn1 Mc2 Mc3 Mc1 ... chilled
```

```r
#Treatment and plant type (Qn1, nonchilled and Mc1,chilled) with highest and lowest mean
#uptake respectively

#2c)
CO2_long <- CO2 %>%
  pivot_longer(cols = "conc":"uptake",
               names_to = "measurement",
               values_to = "values")
print(CO2_long)
```

```
## # A tibble: 168 x 5
##    Plant Type   Treatment  measurement values
##    <ord> <fct>  <fct>      <chr>        <dbl>
##  1 Qn1   Quebec nonchilled conc          95
##  2 Qn1   Quebec nonchilled uptake        16
##  3 Qn1   Quebec nonchilled conc         175
##  4 Qn1   Quebec nonchilled uptake        30.4
##  5 Qn1   Quebec nonchilled conc         250
##  6 Qn1   Quebec nonchilled uptake        34.8
##  7 Qn1   Quebec nonchilled conc         350
##  8 Qn1   Quebec nonchilled uptake        37.2
##  9 Qn1   Quebec nonchilled conc         500
## 10 Qn1   Quebec nonchilled uptake        35.3
## # i 158 more rows
```

```r
CO2_long_summary <- CO2_long%>%
  group_by(Plant,Treatment,measurement) %>%
  summarise(
            mean_measurement = mean(values),
            SD_measurement = sd(values),
            nrow_values = length(values),
            .groups = "keep"
  )
print(CO2_long_summary)
```

```
## # A tibble: 24 x 6
## # Groups:   Plant, Treatment, measurement [24]
##    Plant Treatment  measurement mean_measurement SD_measurement nrow_values
##    <ord> <fct>      <chr>                  <dbl>          <dbl>       <int>
##  1 Qn1   nonchilled conc                   435           318.            7
##  2 Qn1   nonchilled uptake                  33.2           8.21           7
##  3 Qn2   nonchilled conc                   435           318.            7
##  4 Qn2   nonchilled uptake                  35.2          11.0            7
##  5 Qn3   nonchilled conc                   435           318.            7
##  6 Qn3   nonchilled uptake                  37.6          10.3            7
##  7 Qc1   chilled    conc                   435           318.            7
##  8 Qc1   chilled    uptake                  30.0           8.33           7
```

```
##  9 Qc3   chilled    conc                  435           318.             7
## 10 Qc3   chilled    uptake                 32.6          10.3            7
## # i 14 more rows
```

```r
#ambient concentration stays consistent regardless of plant or treatment,
#mean concentration is 435.00000 with standard deviation of 317.726297

#2d
CO2_new <- CO2_long %>%
  filter(
    Treatment == 'nonchilled', measurement == 'uptake'
  )
CO2_new$values2 <- CO2_new$values*1.3

#2e
CO2_new2 <- CO2_long %>%
  mutate(values2 = ifelse(Treatment == "nonchilled" & measurement == "uptake",
                          values * 1.3,
                          values))
#CO2_new2 includes the both 'chilled' and 'nonchilled' treatment types and only applys the
#1.3 x multiplier to only the 'nonchilled' objects, CO2_new only contains plant
#that haven't been chilled
#'ifelse' is a conditional function that checks if conditional argument 'Treatment
#'== "nonchilled" & (and) measurement == "uptake", if this is true than perform the
#'TRUE argument which is values * 1.3, 'else' do the FALSE argument which is simply
#'add value unchanged
```