Homework 09

Peyton Hall

03/21/2024

```
rm(list=ls())
```

```
library(readxl)
library(tidyr)
library(Lahman)
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
```

```
Seafood_data <- read_excel("~/Desktop/Data211/Week 10/Seafood data.xlsx")
Seafood_data
```

```
## # A tibble: 18 x 4
##    Seafood time0 time1 time2
##    <chr>   <dbl> <dbl> <dbl>
##  1 Oysters  1.83  3.68  5.23
##  2 Oysters  1.11  3.25  4.98
##  3 Oysters  2.01  3.98  5.02
##  4 Oysters  7.19  7.42  9.32
##  5 Oysters  6.89  7.34  8.92
##  6 Oysters  7.01  8.11  9.99
##  7 Oysters  6.47  9.44  9.78
##  8 Oysters  5.98  8.77  9.88
##  9 Oysters  5.45  8.12  9.35
## 10 Mussels  0.36  1.7   4.57
## 11 Mussels  0.76  1.65  4.52
## 12 Mussels  0.56  1.55  4.23
## 13 Mussels  5.01  6.38  7.95
## 14 Mussels  5     6.87  7.99
## 15 Mussels  5.03  6.42  8.01
## 16 Mussels  5.05  9.78 11.0
## 17 Mussels  5.89  9.99 11.4
## 18 Mussels  5.97  9.86 11.2
```

```
# 1. On d2l, the data Sea Food recorded the bacterial growth on oysters and
#    mussels. The bacterial counts are measured on a log scale at three
#    different times.
# a) Is the data given in a long format or a wide format? Use a comment to write
#    your answer.
#    - The data is in long format because the length of the columns
#      significantly exceed the length of the rows.
# b) Convert the data to have one variable showing the bacterial counts and one
#    variable showing the different time points.
Seafood_data_wide <- pivot_wider(Seafood_data,
                                 names_from = Seafood,
                                 values_from = c(time0, time1, time2))
```

```
## Warning: Values from ‘time0‘, ‘time1‘ and ‘time2‘ are not uniquely identified; output
## will contain list-cols.
## * Use ‘values_fn = list‘ to suppress this warning.
## * Use ‘values_fn = {summary_fun}‘ to summarise duplicates.
## * Use the following dplyr code to identify duplicates.
##   {data} |>
##   dplyr::summarise(n = dplyr::n(), .by = c(Seafood)) |>
##   dplyr::filter(n > 1L)
```

```
Seafood_data_wide
```

```
## # A tibble: 1 x 6
##   time0_Oysters time0_Mussels time1_Oysters time1_Mussels time2_Oysters
##   <list>        <list>        <list>        <list>        <list>
## 1 <dbl [9]>     <dbl [9]>     <dbl [9]>     <dbl [9]>     <dbl [9]>
## # i 1 more variable: time2_Mussels <list>
```

```
# 2. Use the Teams data in the Lahman package. The team IDs corresponding to
#    Brooklyn baseball teams are listed as:
#    "BR1","BR2","BR3","BR4","BR","BRP","BRF"
# Install and load the Lahman package
# install.packages("Lahman")
# Teams
# a) Keep the variables teamID, HR (home runs), and HRA (homeruns allowed).
Teams_subset <- select(Teams, teamID, HR, HRA) # Select specific variables
#  Teams_subset
# b) Keep only Brooklyn baseball teams
#    ("BR1","BR2","BR3","BR4","BR","BRP","BRF").
Brooklyn_teams <- Teams_subset %>%
                  filter(teamID %in% c("BR1", "BR2", "BR3", "BR4", "BR", "BRP", "BRF"))
# Brooklyn_teams
# c) Choose one apply function to find the average home runs (HR) and average
#    homeruns allowed (HRA).
averages <- Brooklyn_teams %>%
            summarise(avg_HR = mean(HR),
                      avg_HRA = mean(HRA))

# View the calculated averages
# averages
```