

Metadata

Course: DS 5100
Module: 11 R Programming 2
Topic: HW on Tidyverse
Author: R.C. Alvarado (adapted)
Date: 7 July 2023 (revised)

Student Info

Name: Luke Schneider
Net ID: vrd9sd
File GitHub URL: <https://github.com/lukeschneider7/DS5100-vrd9sd/blob/main/lessons/M11/M11-HW-2.Rmd>

Instructions

In your **private course repo** use this notebook to write code that performs the tasks below.

Save your notebook in the M11 directory.

Remember to add and commit these files to your repo.

Then push your commits to your repo on GitHub.

Be sure to fill out the **Student Info** block above.

To submit your homework, save your results as a PDF and upload it to GradeScope.

TOTAL POINTS: 7

Overview

In this homework, you will work with the Abalone dataset from the UCI Machine Learning Repository.

To get started, download and import the `abalone.data` dataset from this URL:

- <https://archive.ics.uci.edu/ml/machine-learning-databases/abalone/abalone.data>

You can pass the URL directly to `read.csv()` and that there is no header row.

Note: The instruction to print in the questions below can be accomplished either through the `print()` function or by displaying a value directly.

TOTAL POINTS: 7

Tasks

Task 0

(0 points)

Get the dataset.

```
# CODE HERE
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.4      v readr      2.1.5
## v forcats    1.0.0      v stringr   1.5.1
## v ggplot2    3.4.4      v tibble    3.2.1
## v lubridate  1.9.3      v tidyr     1.3.1
## v purrr      1.0.2
## -- 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

abalone_data = read.csv(file='https://archive.ics.uci.edu/ml/machine-learning-databases/abalone/abalone.csv')
head(abalone_data)
```

```
##   V1    V2    V3    V4    V5    V6    V7    V8 V9
## 1  M 0.455 0.365 0.095 0.5140 0.2245 0.1010 0.150 15
## 2  M 0.350 0.265 0.090 0.2255 0.0995 0.0485 0.070  7
## 3  F 0.530 0.420 0.135 0.6770 0.2565 0.1415 0.210  9
## 4  M 0.440 0.365 0.125 0.5160 0.2155 0.1140 0.155 10
## 5  I 0.330 0.255 0.080 0.2050 0.0895 0.0395 0.055  7
## 6  I 0.425 0.300 0.095 0.3515 0.1410 0.0775 0.120  8
```

Task 1

(1 point)

Print the number of rows in the dataset.

```
# CODE HERE
print(nrow(abalone_data))
```

```
## [1] 4177
```

Task 2

(1 point)

The rightmost column is the number of rings. Print the maximum number of rings

```
# CODE HERE
print(max(abalone_data$V9))
```

```
## [1] 29
```

Task 3

(1 point)

The leftmost column is the gender with these values: M: male, F: female, I: infant.

Apply the `filter()` function from tidyverse to select only rows where gender is infant, and print the number of records.

```
# CODE HERE
infant_data = abalone_data %>% filter(V1== 'I')
print(nrow(infant_data))
```

```
## [1] 1342
```

Task 4

(1 point)

Apply the `filter()` function from tidyverse to select only rows where gender is infant or male, and print the number of records.

```
# CODE HERE
infant_male_data = abalone_data %>% filter(V1== 'I' | V1 == 'M')
print(nrow(infant_male_data))
```

```
## [1] 2870
```

Task 5

(1 point)

Call the `table()` function on the abalone genders to find out how many of each gender are present.

Print the result.

```
# CODE HERE
table(abalone_data$V1)
```

```
##
##      F      I      M
## 1307 1342 1528
```

Task 6

(1 point)

Compute the mean value of column 2 (V2) grouped by gender.

V2 is the longest shell measurement.

Requirements: use the `%>%` operator to chain commands, and the `group_by()` and `summarize()` functions.

```
# CODE HERE
abalone_data %>% group_by(V1) %>% summarize(mean(V2))
```

```
## # A tibble: 3 x 2
##   V1      'mean(V2)'
##   <chr>      <dbl>
## 1 F          0.579
## 2 I          0.428
## 3 M          0.561
```

Task 7

(1 point)

Compute the MEDIAN value of longest shell measurement for only the males.

Requirements: use the %>% operator to chain commands.

```
# CODE HERE
abalone_data %>% group_by(V1) %>% filter(V1=='M') %>% summarize(median(V2))
```

```
## # A tibble: 1 x 2
##   V1      'median(V2)'
##   <chr>      <dbl>
## 1 M          0.58
```