p8105_hw1_ts3670

Tong Su

2024-09-20

```
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   3.5.1     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 error
```

```
library(ggplot2)
```

Problem 1

Load the penguins dataset

```
data("penguins", package = "palmerpenguins")
```

Short Discription

The `penguins` shows 8 variables about the information of penguins including `species`, `island`, `bill_length_mm`, `bill_depth_mm`, `flipper_length_mm`, `body_mass_g`, `sex`, `year`. It has 344 rows and 8 columns. The mean of `flipper_length_mm` is 200.9152047.

Scatterpolot and Saving

```
plot_df = ggplot(penguins,  aes(x = bill_length_mm, y = flipper_length_mm, colour =species )) + geom_po
```

```
ggsave("Bill_length_mm vs Flipper_length_mm.jpg", plot = plot_df)
```

```
## Saving 6.5 x 4.5 in image
```

Problem 2

#Data Frame

```
set.seed(123)

random_sample = rnorm(10, mean = 0, sd = 1)
logical_vector = random_sample > 0
character_vector = c('Empathy', 'Self-awareness', 'Adaptability',
'Curiosity', 'Motivation', 'Analytics', 'Passionate', 'Optimism', 'Open-Minded',  'Belief')
factor_vector = factor(sample(c('low', 'medium', 'high'), size = 10, replace = TRUE))

df = data.frame(random_sample, logical_vector, character_vector, factor_vector)

df
```

```
##    random_sample logical_vector character_vector factor_vector
## 1    -0.56047565          FALSE          Empathy           low
## 2    -0.23017749          FALSE   Self-awareness           low
## 3     1.55870831           TRUE     Adaptability           low
## 4     0.07050839           TRUE        Curiosity           low
## 5     0.12928774           TRUE       Motivation          high
## 6     1.71506499           TRUE        Analytics        medium
## 7     0.46091621           TRUE       Passionate          high
## 8    -1.26506123          FALSE         Optimism        medium
## 9    -0.68685285          FALSE      Open-Minded           low
## 10   -0.44566197          FALSE           Belief        medium
```

#Mean of each variable in the data frame

```
mean_random_sample = mean(df %>% pull(random_sample))
mean_logical_vector = mean(df %>% pull(logical_vector))
mean_character_vector = try(mean(df %>% pull(character_vector)), silent=FALSE)
```

```
## Warning in mean.default(df %>% pull(character_vector)): argument is not numeric
## or logical: returning NA
```

```
mean_factor_vector = try(mean(df %>% pull(factor_vector)), silent=FALSE)
```

```
## Warning in mean.default(df %>% pull(factor_vector)): argument is not numeric or
## logical: returning NA
```

```
# results
print(mean_random_sample)
```

```
## [1] 0.07462564
```

```
print(mean_logical_vector)
```

```
## [1] 0.5
```

```
print(mean_character_vector)
```

```
## [1] NA
```

```
print(mean_factor_vector)
```

```
## [1] NA
```

Illustration: `character_vector` and `factor_vector` are not numerical nor logical so they cannot be taken the mean.

#Convert variables to numeric

```r
numeric_logical = as.numeric(logical_vector)
numeric_character = (as.numeric(character_vector))
```

```
## Warning: NAs introduced by coercion
```

```r
numeric_factor = (as.numeric(factor_vector))

# output the result
print(numeric_logical)
```

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

```r
print(numeric_character)
```

```
##  [1] NA NA NA NA NA NA NA NA NA NA
```

```r
print(numeric_factor)
```

```
##  [1] 2 2 2 2 1 3 1 3 2 3
```