p8105_hw1_csc2233

Cameron Chesbrough

2024-09-17

Problem 1)

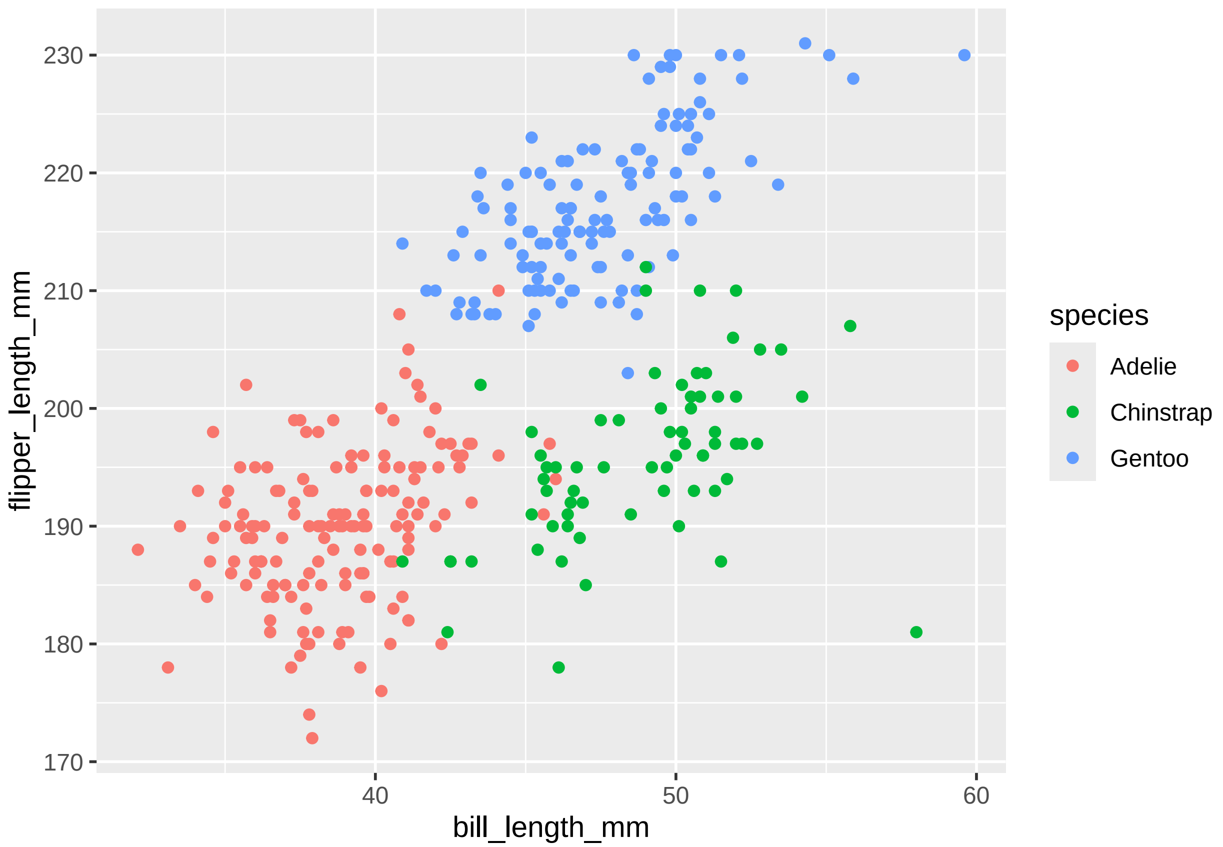
```r
# Import data
data("penguins", package = "palmerpenguins")

# Collect descriptive information
column_names = colnames(penguins)
columns = ncol(penguins)
rows = nrow(penguins)
avg_flipper_length_mm = mean(pull(penguins, var = flipper_length_mm), na.rm = TRUE)
```

Description of penguins)

This data set describes a collection of penguins. Its variables include the species, habitat, body measurements, sex, and year of collection. The data set consists of 8 columns and 344 rows. The mean flipper length of the penguins is 200.9152047 millimeters.

```r
# Create scatterplot with ggplot
ggplot(data = penguins) +
  geom_point(mapping = aes(x = bill_length_mm, y = flipper_length_mm, colour = species))
```

```
## Warning: Removed 2 rows containing missing values or values outside the scale range
## (‘geom_point()‘).
```



```r
# Save plot
ggsave("penguins_scatterplot.pdf")
```

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

```
## Warning: Removed 2 rows containing missing values or values outside the scale range
## (‘geom_point()‘).
```

Problem 2)

```r
# Create necessary vectors
random_sample_vec = rnorm(10)
logic_vec = c(random_sample_vec > 0)
char_vec = c("What", "I", "learned", "in", "boating", "school", "is",
             "blankety", "blankety", "blank!")
diffs = c("Easy","Hard","Hard","Easy","Easy","Hard","Medium","Hard","Medium","Easy")
factor_vec = factor(diffs, levels = c("Easy", "Medium", "Hard"))

# Create dataframe from vectors
patchwork_dataframe = data.frame(
  random_sample_vec,
  logic_vec,
  char_vec,
  factor_vec
)

# Attempt to pull mean for each vector
mean(pull(patchwork_dataframe, var = random_sample_vec))
```

```
## [1] 0.07465437
```

```
mean(pull(patchwork_dataframe, var = logic_vec))
```

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

```
mean(pull(patchwork_dataframe, var = char_vec))
```

```
## Warning in mean.default(pull(patchwork_dataframe, var = char_vec)): argument is
## not numeric or logical: returning NA
```

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

```
mean(pull(patchwork_dataframe, var = factor_vec))
```

```
## Warning in mean.default(pull(patchwork_dataframe, var = factor_vec)): argument
## is not numeric or logical: returning NA
```

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

Discussion on Mean)

Attempting to take the mean of the vector types gave some different results for the different kinds of vectors. For the numeric and logical vectors a number was produced. For the character and factor vectors NA was produced alongside a warning. This warning notes that the argument is not numeric or logical and as a result is returning NA.

```
# Create function that converts a vector into a numeric vector when given a dataframe and one of its co
char_func = function(df, x) {
  xpull = pull(df, x)
  as.numeric(xpull)
}
```

```
# Use created function and hide the outputs
char_func(patchwork_dataframe, "char_vec")
```

```
## Warning in char_func(patchwork_dataframe, "char_vec"): NAs introduced by
## coercion
```

```
char_func(patchwork_dataframe, "logic_vec")
char_func(patchwork_dataframe, "factor_vec")
```

Discussion on as.numeric)

Using the function to convert the vector types into numeric produced a warning for the character vector alongside NAs, converted the logical vector into 1s and 0s, and converted the factor vector into 3s, 2s, and 1s based on their given level. This helps explain what happened when taking the mean as you cannot calculate it with words or strings, but you can with 1s and 0s.