R5 Exercises: Graphing data variables

Karol Topór

2023-12-20

Table of contents

Packages used in this notebook:

```
library(mosaicData)
library(tidyverse)
library(nycflights13)
library(dplyr)
library(ggplot2)
```

---

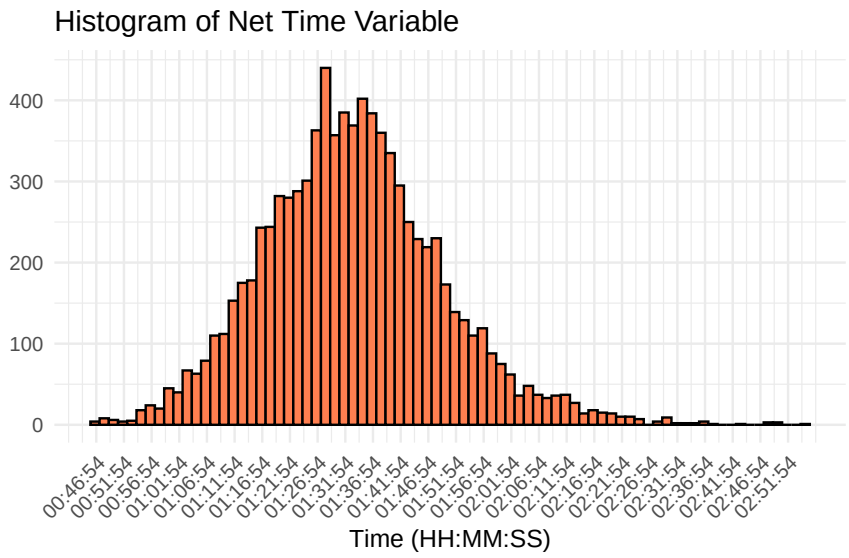
1 Visualization `TenMileRace` data

Use the two variables `time` and `sex` from the `TenMileRace` data in the `mosaicData` package. Choose a suitable visualization method for *each* of them and also for their relationship (create three figures in total). Choose a "Brewer" color palette (using `scale_color_brewer()`, `scale_fill_brewer()`, `scale_color_distiller()` or `scale_fill_distiller()`).

1.1 Visualise variable `time` (3P)

Choose a suitable visualization method for the variable `time` and interpret the diagram.

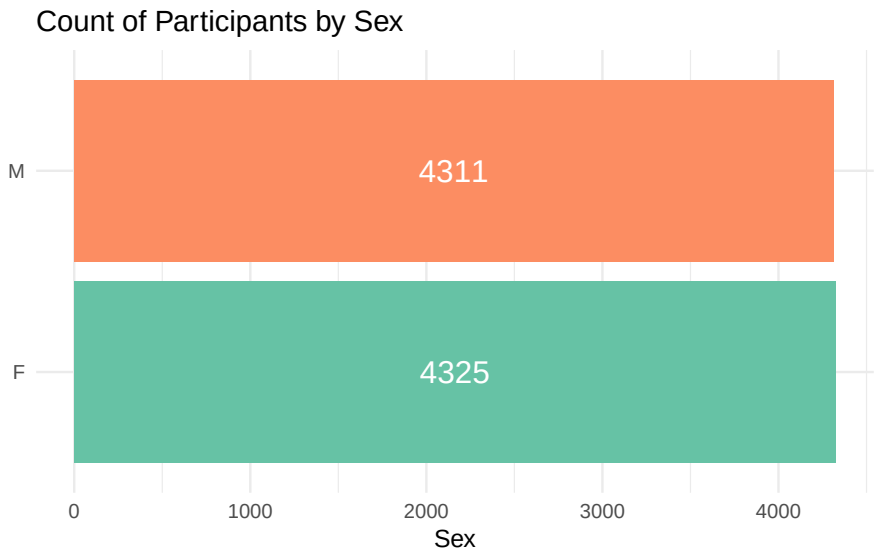
```
time_hr <- function(seconds) {
  hours <- floor(seconds / 3600)
  mins <- floor((seconds %% 3600) / 60)
  secs <- seconds %% 60
  return(sprintf("%02d:%02d:%02d", hours, mins, secs))
}

breaks <- seq(from = min(TenMileRace$net), to = max(TenMileRace$net), by = 300)
labels <- sapply(breaks, time_hr)

ggplot(TenMileRace, aes(x=net)) +
  geom_histogram(binwidth=100, color="black", fill="coral") +
  scale_x_continuous(breaks = breaks, labels = labels) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title="Histogram of Net Time Variable",
       x="Time (HH:MM:SS)",
       y="")
```



1.2 Visualise variable `sex` (3P)

Choose a suitable visualization method for the variable `sex` and interpret the diagram.

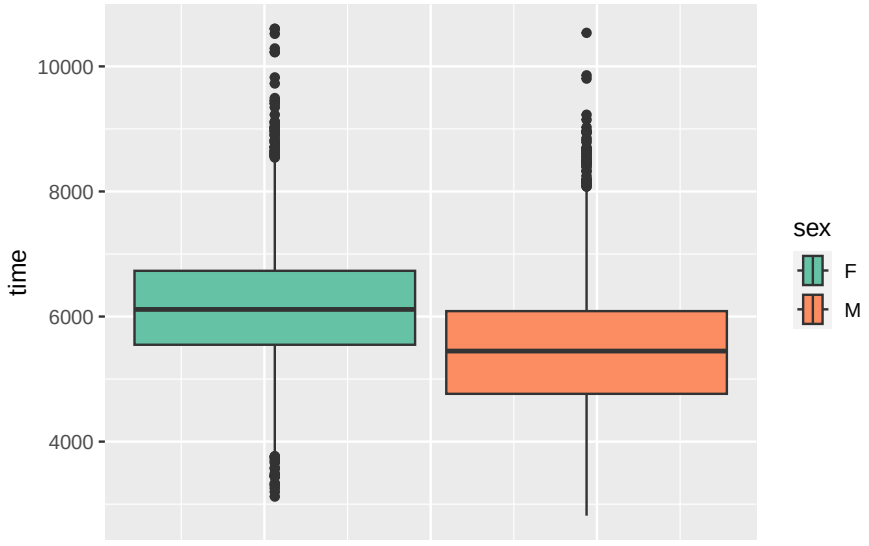
```
ggplot(TenMileRace, aes(x=sex, fill=sex)) +
  geom_bar(stat="count") +
  coord_flip() +  # Flip the coordinates to make the bars horizontal
  theme_minimal() +
  labs(title="Count of Participants by Sex",
       x="",
       y="Sex") +
  theme(legend.position="none",
        axis.title.y=element_blank()) +
  geom_text(stat='count', aes(label=..count..), position=position_stack(vjust=0.5),
              color="white", size=5) +
  scale_fill_brewer(type = 'qual', palette = 7)
```



1.3 Visualise the relation between the two variables `time` and `sex` (3P)

Choose a suitable visualization method for the relation between `time` and `sex` and interpret the diagram.

```
ggplot(TenMileRace, aes(x=time, fill=sex)) +
  geom_boxplot() +
  coord_flip() +
  theme(
    axis.text.x = element_blank(),
    axis.ticks.x = element_blank()
    ) +
  scale_fill_brewer(type = 'qual', palette = 7)
```



2 Graph flights data

Plot the number of trips per month for the plane with the most flights from New York "JFK" airport. Use the `nycflights13` package. A description of the package is available at https://cran.r-project.org/web/packages/nycflights13/nycflights13.pdf.

2.1 Identify the `tailnum` of the plane with the most departures (3P)

Identify the plane (specified by `tailnum` in the `flights` data frame) that traveled the most times from New York City ("JFK") airports in 2013 and assign the `tailnum` of this plane to the variable `id_tailnum`.

```
id_tailnum <- flights %>%
  drop_na() %>%
  filter(origin == "JFK", year == 2013) %>%
  group_by(tailnum) %>%
  summarize(n_flights = n()) %>%
  arrange(desc(n_flights)) %>%
  slice(1) %>%
  pull(tailnum)

id_tailnum
```

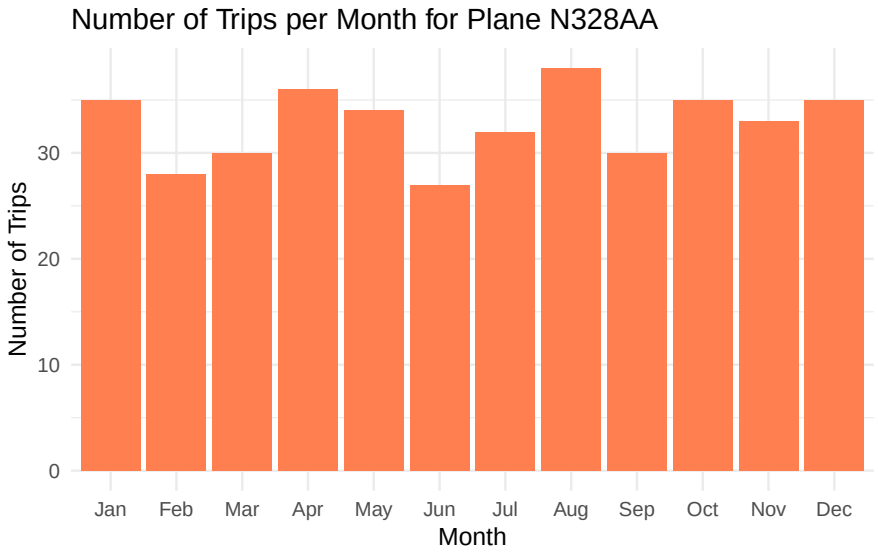
```
[1] "N328AA"
```

2.2 Graph the number of trips per month (5P).

Plot of the number of trips per month from New York for the plane identified by `id_tailnum`. Include the info in `id_tail_num` into the title of the graph. Use the command `Monat=as.factor(format(time_hour,"%b"))` to extract `Monat` from the variable `time_hour`.

Use the command `fct_reorder(Monat,month)` to reorder the factor `Monat` by the variable `month` (instead of alphabetical ordering of the factor).

```r
trips_per_month <- flights %>%
  filter(tailnum == id_tailnum, grepl("JFK", origin)) %>%
  mutate(Monat = as.factor(format(time_hour, "%b")),
         month = as.numeric(format(time_hour, "%m"))) %>%
  mutate(Monat = fct_reorder(Monat, month)) %>%
  group_by(Monat) %>%
  summarize(n_trips = n()) %>%
  ungroup()

# Plot the number of trips per month
ggplot(trips_per_month, aes(x = Monat, y = n_trips)) +
  geom_col(fill = "coral") +
  labs(title = paste("Number of Trips per Month for Plane", id_tailnum),
       x = "Month",
       y = "Number of Trips") +
  theme_minimal()
```



3 Visualise dietary data

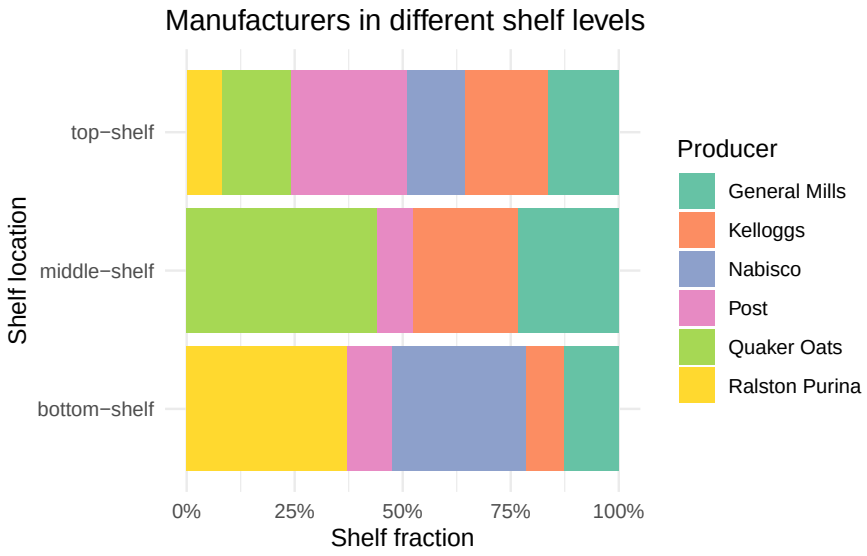
Use the code `data("UScereal", package = "MASS")` for the `UScereal` data from the `MASS` package. See https://www.rdocumentation.org/packages/MASS/versions/7.3-53/topics/UScereal for details. Adjust the Manufacturer in `mfr` (represented by its first initial): G=General Mills, K=Kelloggs, N=Nabisco, P=Post, Q=Quaker Oats, R=Ralston Purina and the display shelf in `shelf` (1, 2, or 3, counting from the floor) into `bottom-shelf`, `middle-shelf` and `top-shelf`. Visualize the relationship of calories, sugars and fat, additionally, highlight whether the product has been enriched with vitamins.

```
data("UScereal", package = "MASS")

UScereal <- UScereal %>%
  mutate(
    mfr = case_when(
      mfr == "G" ~ "General Mills",
      mfr == "K" ~ "Kelloggs",
      mfr == "N" ~ "Nabisco",
      mfr == "P" ~ "Post",
      mfr == "Q" ~ "Quaker Oats",
      mfr == "R" ~ "Ralston Purina"
    ),
    shelf = factor(shelf, labels = c("bottom-shelf", "middle-shelf", "top-shelf"))
  )
```

3.1 Distribution of producers (4P)

Visualize the distribution of manufacturers among the shelves. Use appropriate titles and legends.

```
ggplot(UScereal, aes(fill = mfr, y = ..prop.., group = mfr)) +
  geom_bar(aes(x = shelf), position = "fill", stat="count") +
  scale_y_continuous(labels = scales::percent) +
  labs(title = "Manufacturers in different shelf levels",
       y = "Shelf fraction",
       x = "Shelf location",
       fill = "Producer") +
  theme_minimal() +
  coord_flip() +
  scale_fill_brewer(type = 'qual', palette = 7)
```

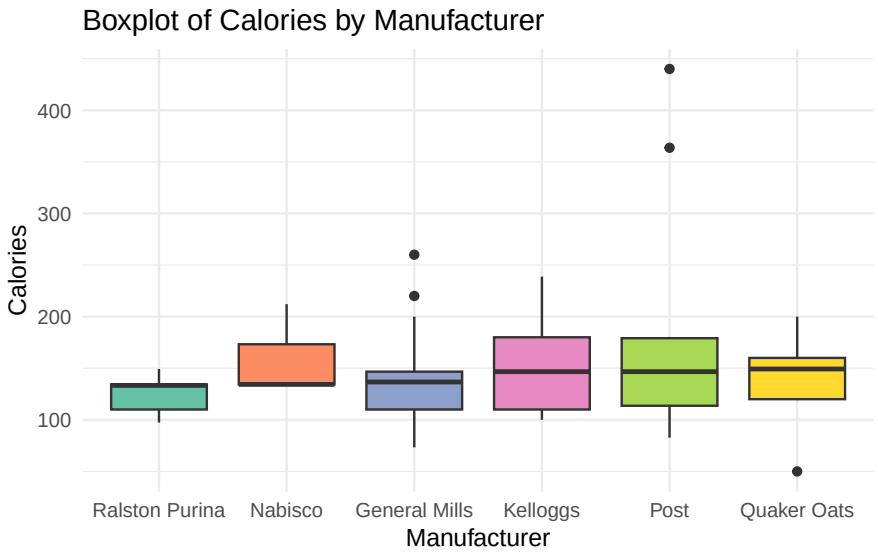


3.2 Distribution of calories for different producers (4P)

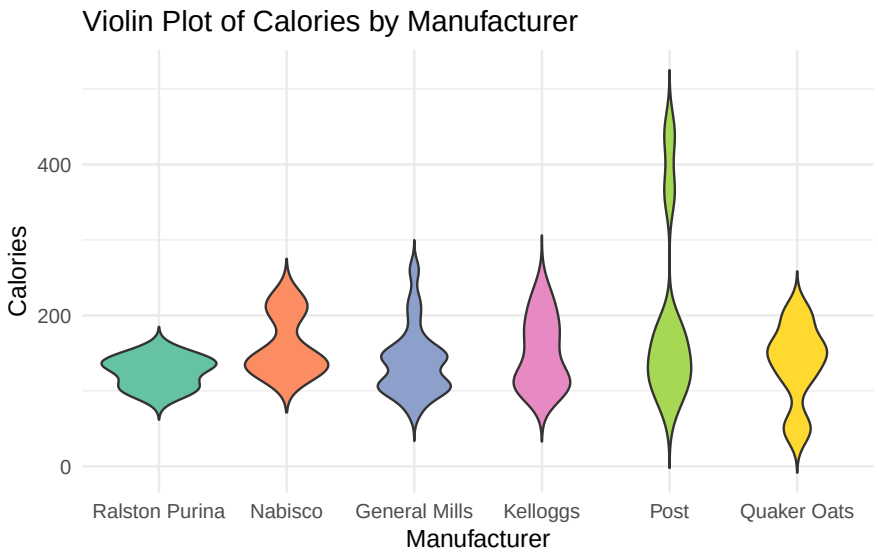
Visualize the amount of calories in the products of different manufacturers with a boxplot and a violin plot. Reorder the manufacturers with the command `fct_reorder(mfr, calories)` in the graph. Use appropriate titles and legends.

```
# boxplot
UScereal %>%
```

```
  mutate(mfr = fct_reorder(mfr, calories)) %>%
  ggplot(aes(x = mfr, y = calories, fill = mfr)) +
  geom_boxplot() +
  scale_fill_brewer(type = 'qual', palette = 7) +
  labs(title = "Boxplot of Calories by Manufacturer",
       x = "Manufacturer",
       y = "Calories") +
  theme_minimal() +
  theme(legend.position = "none")
```




```
# Violin plot
UScereal %>%
  mutate(mfr = fct_reorder(mfr, calories)) %>%
  ggplot(aes(x = mfr, y = calories, fill = mfr)) +
  geom_violin(trim = FALSE) +
  scale_fill_brewer(type = 'qual', palette = 7) +
  labs(title = "Violin Plot of Calories by Manufacturer",
       x = "Manufacturer",
       y = "Calories") +
  theme_minimal() +
  theme(legend.position = "none")
```


Violin Plot of Calories by Manufacturer
Calories
400
200
0
Ralston Purina
Nabisco
General Mills
Kelloggs
Post
Quaker Oats
Manufacturer