p8105_hw3_csc2233

Cameron Chesbrough

2024-10-12

Question 2

```
# Data Import and Cleaning

accel_df = read_csv(file = "./datasets/nhanes_accel.csv")
```

```
## Rows: 250 Columns: 1441
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (1441): SEQN, min1, min2, min3, min4, min5, min6, min7, min8, min9, min1...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
covar_df = read_csv(file = "./datasets/nhanes_covar.csv",
                    skip = 4)
```

```
## Rows: 250 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (5): SEQN, sex, age, BMI, education
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
accel_df = accel_df %>%
  janitor::clean_names()

covar_df = covar_df %>%
  janitor::clean_names() %>%
  filter(age >= 21) %>%
  drop_na()

covar_df$sex = as.factor(covar_df$sex)
covar_df$education = as.factor(covar_df$education)

accel_w_covar = inner_join(covar_df, accel_df, by = c("seqn" = "seqn"))

# Creation of Table and Bargraph to compare Education and Sex
```

```
education_sex_counts = accel_w_covar %>%
  group_by(education, sex) %>%
  summarise(n_obs = n())
```

```
## `summarise()` has grouped output by 'education'. You can override using the
## `.groups` argument.
```

```
knitr::kable(education_sex_counts %>%
  pivot_wider(names_from = education, values_from = n_obs),
  col.names = c("Sex", "Less than Highschool", "Highschool or Equivalent", "More than Highschool"),
  align = "cccc",
  caption = "Count of Men and Women in each Education Category")
```

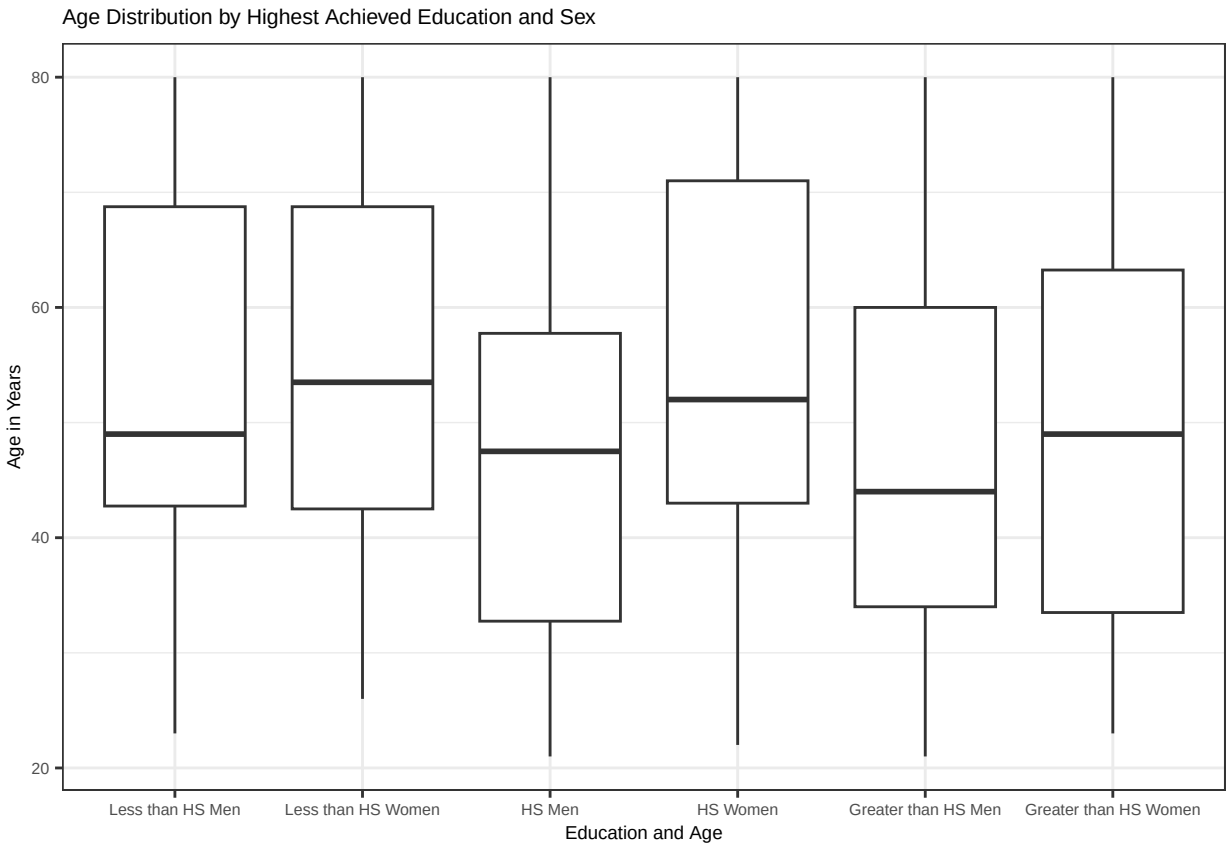
Table 1: Count of Men and Women in each Education Category

Sex	Less than Highschool	Highschool or Equivalent	More than Highschool
1	27	35	56
2	28	23	59

```
education_age = accel_w_covar %>%
  group_by(education, age, sex) %>%
  summarise(n_obs = n()) %>%
  mutate(comb_edu_sex = paste(education, sex))
```

```
## `summarise()` has grouped output by 'education', 'age'. You can override using
## the `.groups` argument.
```

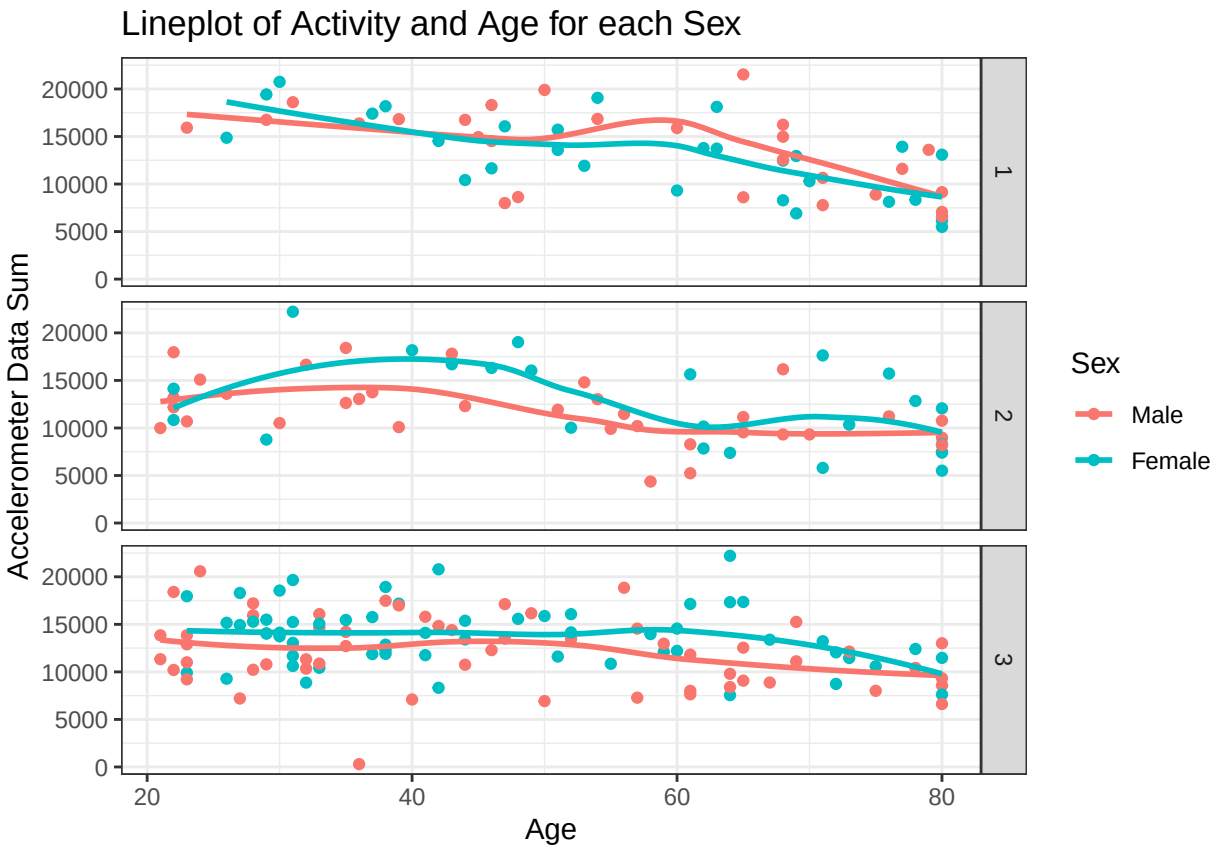
```
edu_age_plot = ggplot(data = education_age) +
  geom_boxplot(aes(comb_edu_sex, age)) +
  labs(title = "Age Distribution by Highest Achieved Education and Sex",
    x = "Education and Age",
    y = "Age in Years") +
   scale_x_discrete(labels = c("Less than HS Men",
                               "Less than HS Women",
                               "HS Men", "HS Women",
                               "Greater than HS Men",
                               "Greater than HS Women")) +
  theme_bw() +
  theme(text = element_text(size=7))

edu_age_plot
```



Looking first at the table, there are much more participants in the sample with greater than a highschool education. There is not much of gender difference in the less than a highschool education and greater than a highschool education, but there are 12 more men than women in the sample with a highschool or equivalent. Moving to the visualization, it seems like highschool women and greater than highschool men have a lower median age. All categories seem relatively spread; for most categories half of the observations fall between ages 65 and 45.

```
# Activity and Age

total_act = accel_w_covar %>%
  mutate(actsum = select(., min1:min1440) %>% rowSums())

total_act_plot = ggplot(data = total_act,
                        aes(x = age, y = actsum, color = sex)) +
  geom_point() +
  geom_smooth(se = FALSE) +
  facet_grid(rows="education") +
    labs(title = "Lineplot of Activity and Age for each Sex",
    x = "Age",
    y = "Accelerometer Data Sum",
    color = "Sex") +
  scale_color_discrete(labels=c("Male", "Female")) +
  theme_bw()

total_act_plot
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```

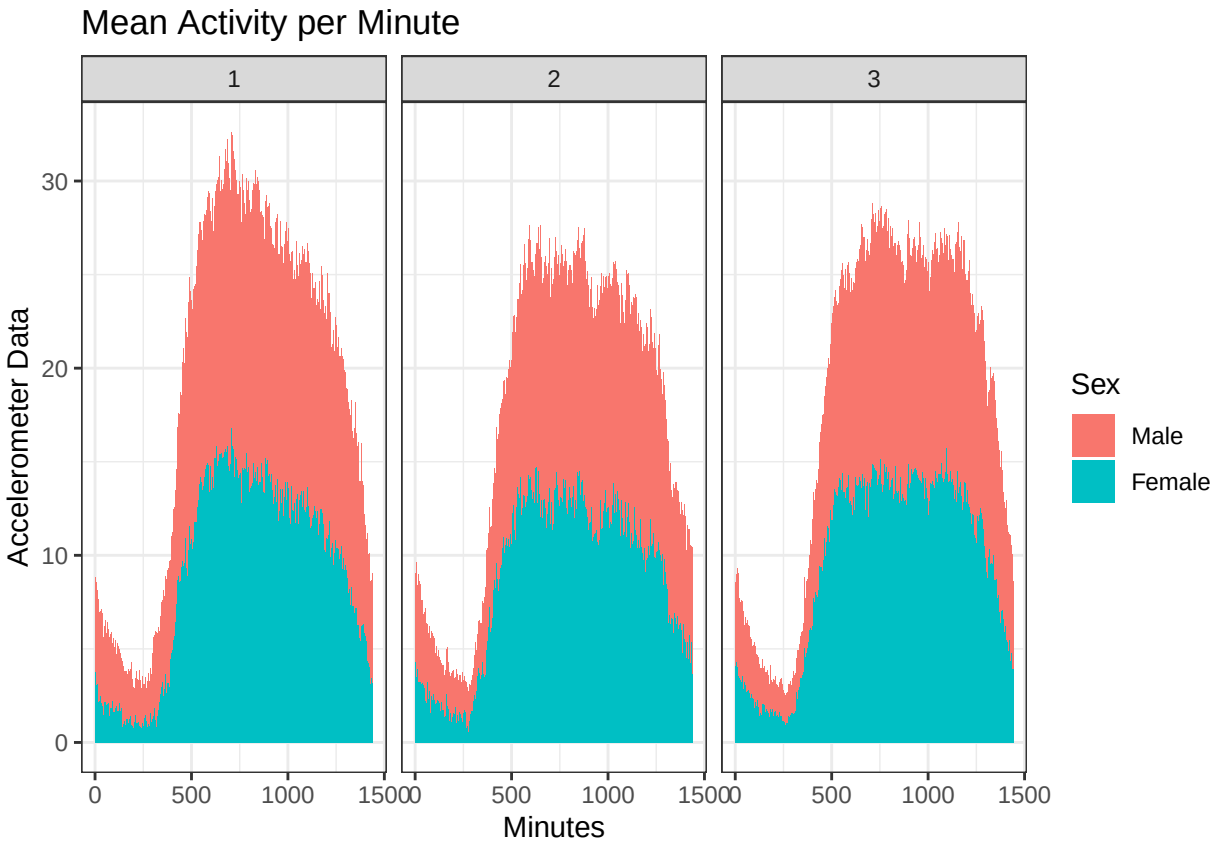


In both highschool and greater than highschool women generally record higher in activity while both sexes eventually become closer as age increases; however in less than highschool women initally have higher activity, but the men overtake this around age 50. Less than highschool appears slightly more active before age 40 and highschool is less active at age 60.

```
# 24 Hour Activity

minutes = pivot_longer(accel_w_covar,
                       min1:min1440,
                       names_to = "minute",
                       values_to = "accel")

minutes_group = minutes %>%
  group_by(minute, sex, education) %>%
  summarise(acl_mean = mean(accel)) %>%
  mutate(minute = str_sub(minute, 4, -1)) %>%
  mutate(minute = as.numeric(minute)) %>%
  arrange(minute)
```

```
## `summarise()` has grouped output by 'minute', 'sex'. You can override using the
## `.groups` argument.
```

```
plot_acts = ggplot(data = minutes_group,
                   aes(x = minute, y = acl_mean, fill = sex)) +
  geom_col() +
  facet_grid(. ~ education) +
    labs(title = "Mean Activity per Minute",
    x = "Minutes",
    y = "Accelerometer Data",
    fill = "Sex") +
  scale_fill_discrete(labels=c("Male", "Female")) +
  theme_bw()

plot_acts
```



All groups start low and shrink lower still and reach their lowest activity levels around 4:00 in the morning. Activity levels will start to rise sharply, peaking around 12:30 in the afternoon. In the less than highschool category, the overall pattern changes faster; the other two categories almost have a flat top of activity from 8:30 to 9:00.

Question 3

```
# Data Import and Cleaning

jan20_df = read_csv(file = "./datasets/Jan 2020 Citi.csv")
```

```
## Rows: 12420 Columns: 7
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (6): ride_id, rideable_type, weekdays, start_station_name, end_station_n...
## dbl (1): duration
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
jul20_df = read_csv(file = "./datasets/July 2020 Citi.csv")
```

```
## Rows: 21048 Columns: 7
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (6): ride_id, rideable_type, weekdays, start_station_name, end_station_n...
## dbl (1): duration
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
jan24_df = read_csv(file = "./datasets/Jan 2024 Citi.csv")
```

```
## Rows: 18861 Columns: 7
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (6): ride_id, rideable_type, weekdays, start_station_name, end_station_n...
## dbl (1): duration
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
jul24_df = read_csv(file = "./datasets/July 2024 Citi.csv")
```

```
## Rows: 47156 Columns: 7
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (6): ride_id, rideable_type, weekdays, start_station_name, end_station_n...
## dbl (1): duration
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
month_year = rep("Jan_20", nrow(jan20_df))
jan20_df = jan20_df %>%
  janitor::clean_names() %>%
  cbind(month_year) %>%
  drop_na()

month_year = rep("Jul_20", nrow(jul20_df))
jul20_df = jul20_df %>%
  janitor::clean_names() %>%
```

```
  cbind(month_year) %>%
  drop_na()

month_year = rep("Jan_24", nrow(jan24_df))
jan24_df = jan24_df %>%
  janitor::clean_names() %>%
  cbind(month_year) %>%
  drop_na()

month_year = rep("Jul_24", nrow(jul24_df))
jul24_df = jul24_df %>%
  janitor::clean_names() %>%
  cbind(month_year) %>%
  drop_na()

all_dates = bind_rows(jan20_df, jul20_df, jan24_df, jul24_df)
```

This dataset describes the usage of citibikes across four windows of time. Duration, membership, and station are among the variables recorded. It includes 8 total variables and 999253 observations.

```
# Table Creation

date_membership_counts = all_dates %>%
  group_by(month_year, member_casual) %>%
  summarise(n_obs = n())
```

```
## `summarise()` has grouped output by 'month_year'. You can override using the
## `.groups` argument.
```

```
knitr::kable(date_membership_counts %>%
  pivot_wider(names_from = month_year, values_from = n_obs),
  col.names = c("Membership Status", "January 2020",
                "January 2024", "July 2020", "July 2024"),
  align = "cccc",
  caption = "Count of Members and Casual Riders n in each Date/Year Category")
```

Table 2: Count of Members and Casual Riders n in each Date/Year Category

Membership Status	January 2020	January 2024	July 2020	July 2024
casual	980	2094	5625	10843
member	11418	16705	15388	36200

Overall citibikes are used much more in July than in January. There was also a big jump in usage from 2020 to 2024 among both casual users and members. It would seem that causal riders were more likely to use citibikes in the summer, but in 2024 there was a very large jump in members.

```
# Popular Stations

jul24_popstations = all_dates %>%
```

```
  filter(month_year == "Jul_24") %>%
  group_by(start_station_name, month_year) %>%
  summarise(n_obs = n()) %>%
  select(-month_year) %>%
  arrange(desc(n_obs)) %>%
  head(5)
```

```
## `summarise()` has grouped output by 'start_station_name'. You can override
## using the `.groups` argument.
```

```
knitr::kable(jul24_popstations,
  align = "cc",
  col.names = c("Station", "Rides"),
  caption = "Most Popular Starting Stations in July 2024")
```

Table 3: Most Popular Starting Stations in July 2024

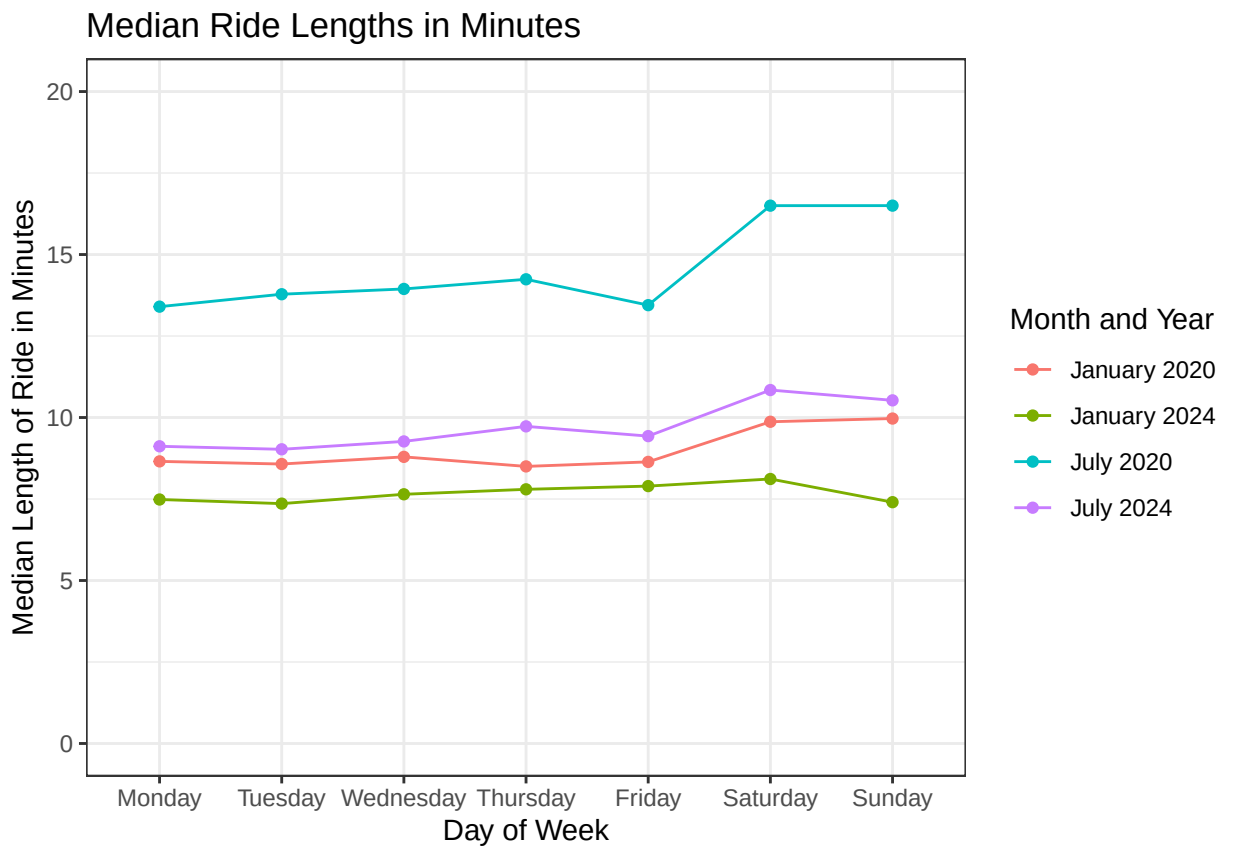
Station	Rides
Pier 61 at Chelsea Piers	163
University Pl & E 14 St	155
W 21 St & 6 Ave	152
West St & Chambers St	150
W 31 St & 7 Ave	145

```
# Day/Month/Year

day_monthyear_counts = all_dates %>%
  group_by(weekdays, month_year) %>%
  summarise(meds = median(duration)) %>%
  mutate(Day_Month_Year = paste(weekdays, month_year)) %>%
  mutate(weekdays = factor(weekdays, levels = c("Monday", "Tuesday", "Wednesday",
                                                "Thursday", "Friday", "Saturday",
                                                "Sunday")))
```

```
## `summarise()` has grouped output by 'weekdays'. You can override using the
## `.groups` argument.
```

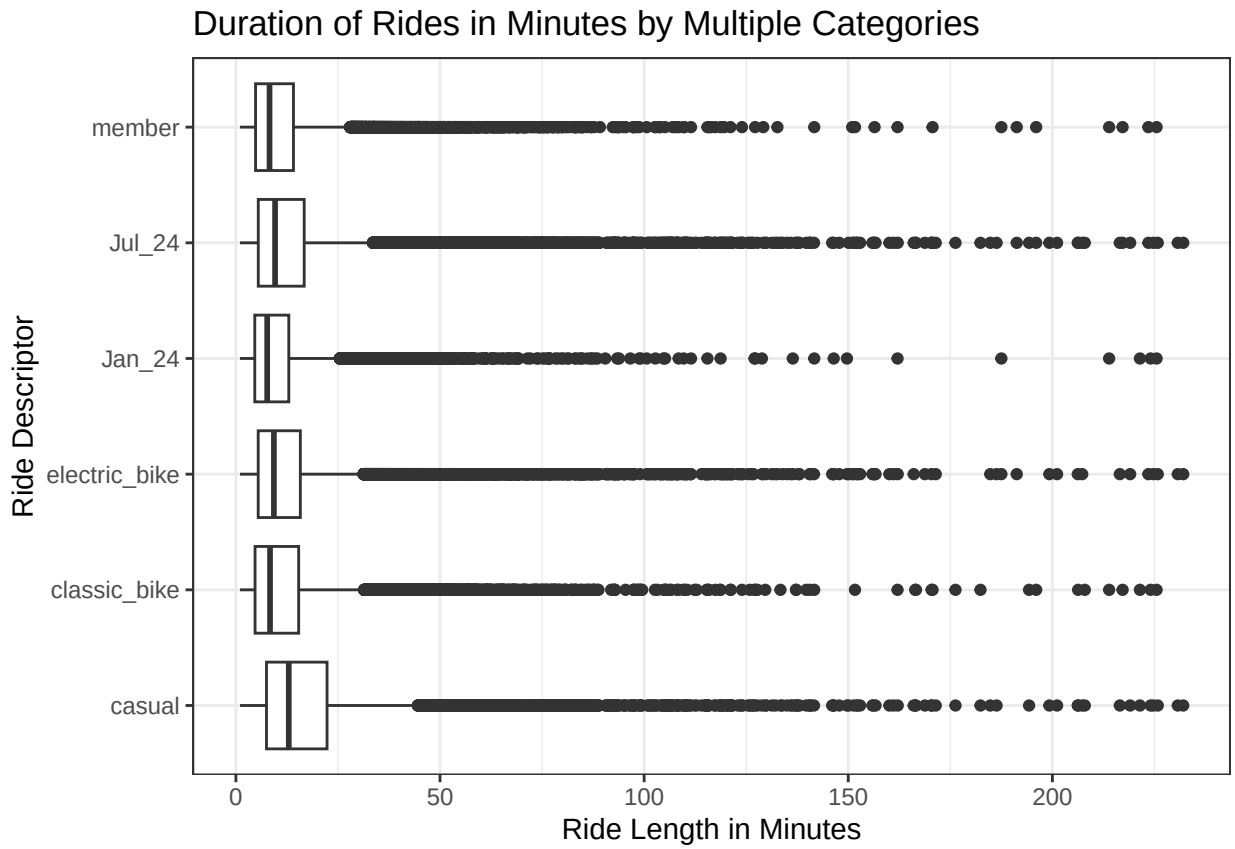
```
day_plot = ggplot(data = day_monthyear_counts,
                  aes(x = weekdays, y = meds,
                      group = month_year, color = month_year)) +
  geom_line() +
  geom_point() +
  ylim(c(0,20)) +
  labs(title = "Median Ride Lengths in Minutes",
    x = "Day of Week",
    y = "Median Length of Ride in Minutes",
    color = "Month and Year") +
  scale_color_discrete(labels=c("January 2020", "January 2024",
                                "July 2020", "July 2024")) +
  theme_bw()
```

```
day_plot
```



The median ride length was signifigantly higher in July 2020 than the other time periods. The second highest was the other July period, but to a lesser degree. Saturday appears to be the most popular day, followed by Sunday (with the exception of January 2024).

```
# 2024 Only Plot

only_24 = all_dates %>%
  filter(month_year %in% c("Jan_24", "Jul_24"))

comparing_vars = ggplot(data = only_24) +
  geom_boxplot(aes(duration, member_casual)) +
  geom_boxplot(aes(duration, rideable_type)) +
  geom_boxplot(aes(duration, month_year)) +
  labs(title = "Duration of Rides in Minutes by Multiple Categories",
    x = "Ride Length in Minutes",
    y = "Ride Descriptor") +
  theme_bw()

comparing_vars
```



It appears that causal riders had longer durations than their member couterparts. July alos saw longer rides than January. There was little difference between classic and electric bikes.