STA3115_homework1

2021122006 Jaehun Shon

2024-09-16

Problem 1: Splitting the "time" variable

```
billboard %>%
  separate(time, into = c("minutes", "seconds"), sep = ":", remove = FALSE) %>%
  .[, c("year", "artist.inverted", "track", "time", "minutes", "seconds")] -> billboard.1

billboard.1 %>%
  head(10) %>%
  kable(col.names = c("Year", "Artist", "Track", "Time", "Minutes", "Seconds"),
        caption = "Top 10 rows of Billboard Data",
        format = "markdown")
```

Table 1: Top 10 rows of Billboard Data

Year	Artist	Track	Time	Minutes	Seconds
2000	Destiny's Child	Independent Women Part I	3:38	3	38
2000	Santana	Maria, Maria	4:18	4	18
2000	Savage Garden	I Knew I Loved You	4:07	4	07
2000	Madonna	Music	3:45	3	45
2000	Aguilera, Christina	Come On Over Baby (All I Want Is You)	3:38	3	38
2000	Janet	Doesn't Really Matter	4:17	4	17
2000	Destiny's Child	Say My Name	4:31	4	31
2000	Iglesias, Enrique	Be With You	3:36	3	36
2000	Sisqo	Incomplete	3:52	3	52
2000	Lonestar	Amazed	4:25	4	25

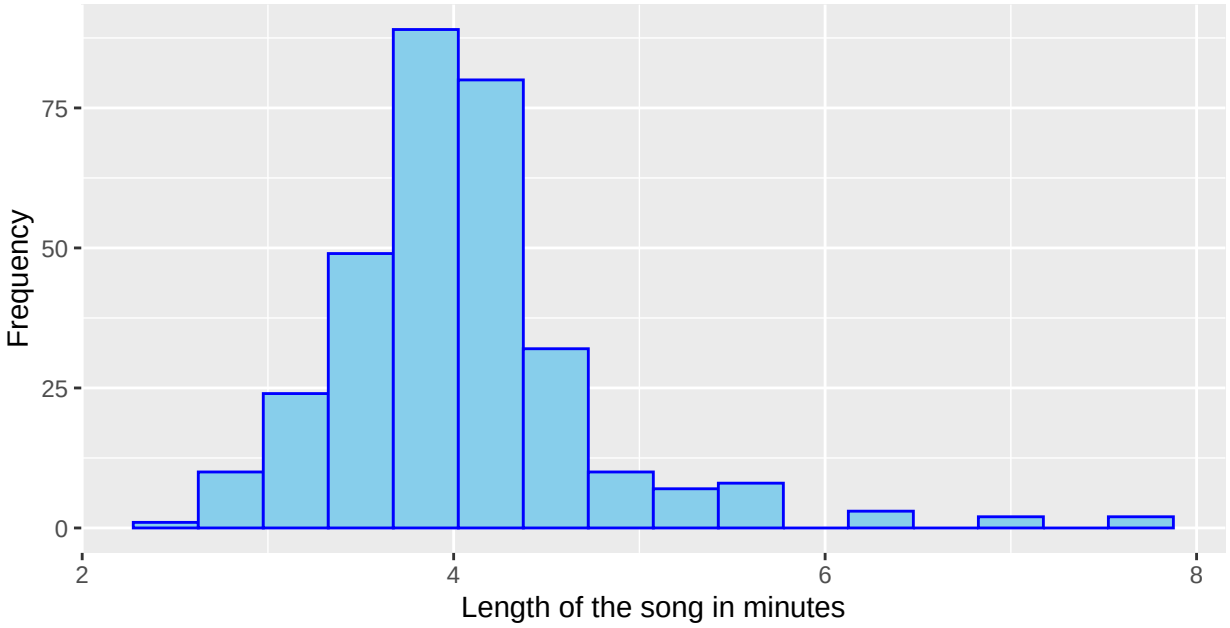
Problem 2: Creating a new time variable and histogram

```
billboard.1 |>
  mutate(time_in_min = as.numeric(minutes) + as.numeric(seconds) / 60) -> timevar

t <- timevar$time_in_min
description <- data.frame(
  Statistic = c("Mean", "Sd", "Median", "IQR", "Min", "Max"),
  Value = c(mean(t), sd(t), median(t), IQR(t), range(t)) %>% round(., 3)
) %>% t
```

```
histplot <- ggplot(timevar, aes(x = time_in_min)) +
  geom_histogram(fill = "skyblue", color = "blue", binwidth = 0.35) +
  labs(x = "Length of the song in minutes",
       y = "Frequency")

table_grob <- tableGrob(description)
final_plot <- grid.arrange(histplot, table_grob, nrow = 2, heights = c(3/4, 1/4))
```



Statistic	Mean	Sd	Median	IQR	Min	Max
Value	4.040	0.707	3.933	0.633	2.600	7.833

The characteristic of distribution is stated in above. The data has right-skewed distribution, which can easily found in histogram.

Problem 3: Summary statistics by weeks on chart

```
billboard %>%
  separate(time, into = c("minutes", "seconds"), sep = ":") %>%
  select(year, artist.inverted, minutes, seconds, x16th.week) %>%
  mutate(time_in_min = as.numeric(minutes) + as.numeric(seconds) / 60) -> billboard.2

a <- billboard.2 %>% .[is.na(.[, 'x16th.week']),] %>% pull(time_in_min) # below 15
b <- billboard.2 %>% .[!is.na(.[, 'x16th.week']),] %>% pull(time_in_min) # more than 15

compare.dt <- tibble(
  Statistic = c("Mean", "Standard Deviation", "Median", "IQR"),
```

```
  `Below 15 weeks` = c(mean(a), sd(a), median(a), IQR(a)),
  `More than 15 weeks` = c(mean(b), sd(b), median(b), IQR(b))
)

compare.dt %>%
  kable(col.names = c("Statistic", "Below 15 weeks", "More than 15 weeks"),
        caption = "Comparison of Statistics for Below 15 weeks and More than 15 weeks in song length",
        format = "markdown")
```

Table 2: Comparison of Statistics for Below 15 weeks and More than 15 weeks in song length

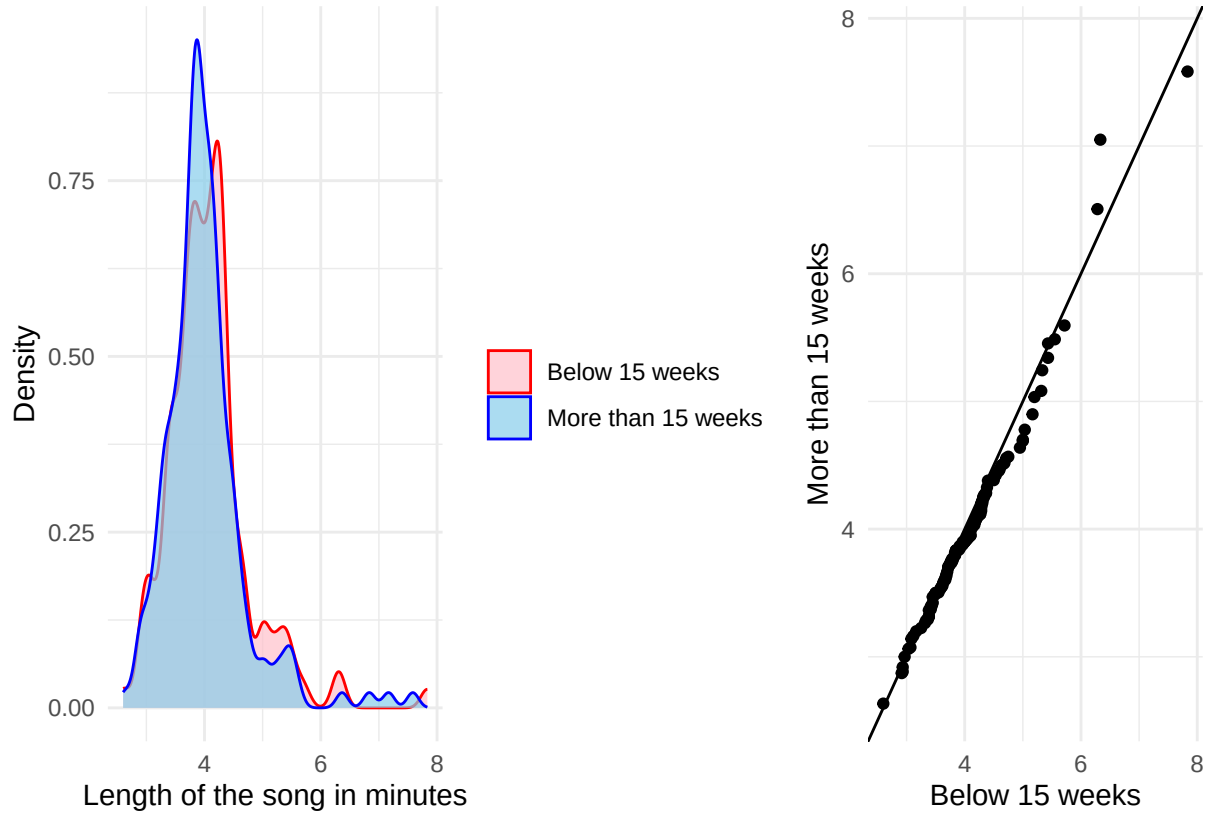
Statistic	Below 15 weeks	More than 15 weeks
Mean	4.0825926	4.0091575
Standard Deviation	0.7193945	0.6974643
Median	4.0166667	3.9166667
IQR	0.6333333	0.6083333

Problem 4: Comparing song length distribution

```
plot1 <- ggplot(billboard.2, aes(x = time_in_min)) +
  geom_density(aes(fill = ifelse(!is.na(x16th.week), "More than 15 weeks", "Below 15 weeks"),
                   color = ifelse(!is.na(x16th.week), "More than 15 weeks", "Below 15 weeks")),
               alpha = 0.7, adjust = 0.7) +
  labs(x = "Length of the song in minutes", y = "Density",
       fill = "", color = "") +
  scale_fill_manual(values = c("More than 15 weeks" = "skyblue",
                               "Below 15 weeks" = "pink")) +
  scale_color_manual(values = c("More than 15 weeks" = "blue",
                                "Below 15 weeks" = "red")) +
  theme_minimal()


qq.df <- as.data.frame(qqplot(a, b, plot.it = FALSE))
plot2 <-ggplot(qq.df, aes(x = x, y = y)) +
  geom_point() +
  geom_abline(slope = 1, intercept = 0, color = "black") +
  labs(x = "Below 15 weeks", y = "More than 15 weeks") +
  xlim(range(c(a, b))) +
  ylim(range(c(a, b))) +
  theme_minimal()

pp <- plot1 + plot2
pp
```



Density plot(left), QQ plot(right)

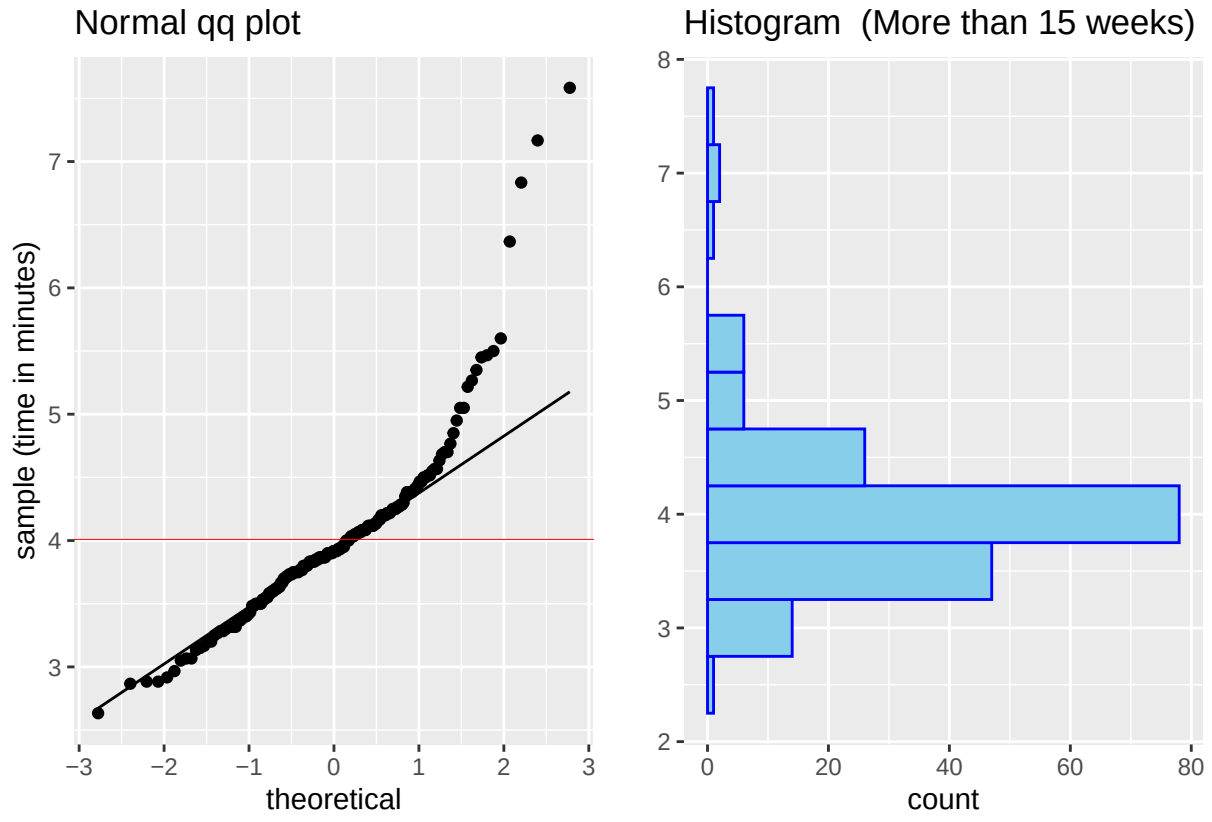
1. As density plot shows smooth distribution shape of the data, it is easy to compare between two groups.
2. I wanted to show the overall shape in easy-to-see way with a density plot, and show how further difference between groups occured with more precise by QQ plot.

Problem 5: Comparing to the Normal distribution

```
qq_plot <- ggplot() +
  stat_qq(data = billboard.2 %>% filter(!is.na(x16th.week)),
          aes(sample = time_in_min)) +
  stat_qq_line(data = billboard.2 %>% filter(!is.na(x16th.week)),
               aes(sample = time_in_min), linewidth = 0.5) +
  geom_abline(aes(intercept = mean(b), slope = 0), color = 'red', linewidth = 0.1) +
  labs(x = "theoretical", y = "sample (time in minutes)", title = 'Normal qq plot')

hist_plot <- ggplot() +
  geom_histogram(data = billboard.2 %>% filter(!is.na(x16th.week)),
                 aes(x = time_in_min),
                 binwidth = 0.5, fill = "skyblue", color = "blue") +
  coord_flip() +
  labs(x = "", y = "count", title = 'Histogram  (More than 15 weeks)')
```

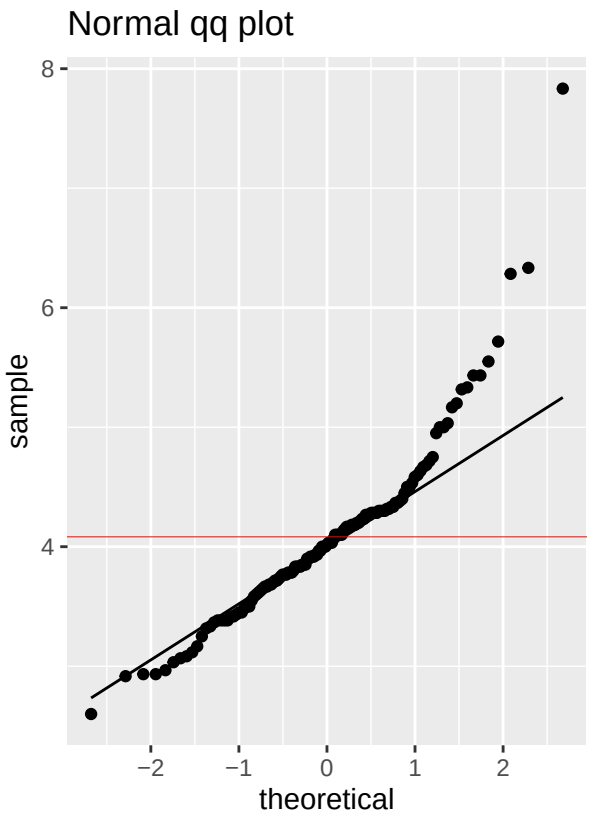
```
qq_plot + hist_plot
```

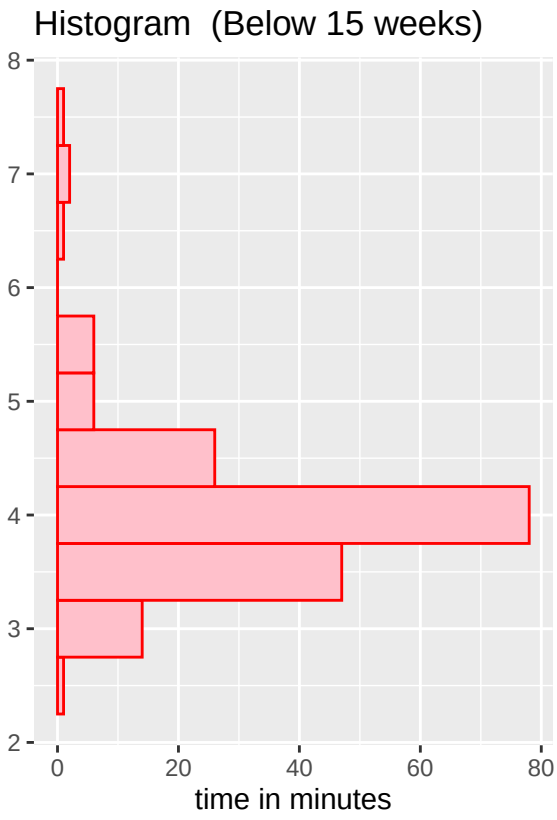



```
qq_plot <- ggplot() +
  stat_qq(data = billboard.2 %>% filter(is.na(x16th.week)),
          aes(sample = time_in_min)) +
  stat_qq_line(data = billboard.2 %>% filter(is.na(x16th.week)),
               aes(sample = time_in_min), linewidth = 0.5) +
  geom_abline(aes(intercept = mean(a), slope = 0), color = 'red', linewidth = 0.1) +
  labs(x = "theoretical", y = "sample", title = 'Normal qq plot')

hist_plot <- ggplot() +
  geom_histogram(data = billboard.2 %>% filter(!is.na(x16th.week)),
                 aes(x = time_in_min),
                 binwidth = 0.5, fill = "pink", color = "red") +
  coord_flip() +
  labs(x = "", y = "time in minutes", title = "Histogram  (Below 15 weeks)")

qq_plot + hist_plot
```





1. Distribution of each group shows deviation from the normal distribution when the length of song exceeds approximately 4.5 minutes. The red line represents the mean of the length of song respectively, it shows the average value is deviated from the midpoint of the entire data range.
2. To show the skewedness of data in each group easily, histogram was added next to the QQplot. They look quite similar to the plot in Promblem 2, which is right-skewed.