Bellabeat Project Report

Michalis Loizos

2025-04-22

Introduction

This project is my capstone for the Google Data Analytics Certificate, focusing on Bellabeat, a health tech company offering wellness products like the Leaf tracker. I analyzed Fitbit user data from 34 users, including daily activity (steps, calories), hourly trends, and workout intensities, to uncover patterns and provide marketing recommendations for Bellabeat's target audience. Using Google Sheets for data cleaning and pivot table creation, SQL for filtering, and R (tidyverse, ggplot2) for visualizations, this project demonstrates skills in data wrangling, visualization, and business analysis.

Data Processing and Summary

- This report summarizes the work performed on the Bellabeat dataset. The project used Google Sheets for data cleaning and pivot table creation, SQL to apply filters, and R for visualizations and the summary report.
- Data files from the Bellabeat dataset were carefully inspected for inconsistencies and cleaned as needed, such as converting UTC-formatted times to 24-hour format for hourly user trend studies. Afterward, a pivot table was created in Google Sheets to summarize user application data and identify the most active users. Table 1 shows the pivot table of user categories by calories burned

```
pivot_table <- read.csv("PIVOT_TABLE.csv")

pivot_table_sorted <- pivot_table %>%
  arrange(desc(Average_Calories))

# Specify Columns
colnames(pivot_table_sorted) <- c("ID",
                                  "Average Total Steps",
                                  "Average Total Distance",
                                  "Average Tracker Distance",
                                  "Average Logged Activities Distance",
                                  "Average Calories",
                                  "Average Lightly Active Minutes",
                                  "Average Fairly Active Minutes",
                                  "Average Very Active Minutes",
                                  "Average Sedentary Minutes")

# Specify Column Format
kable(pivot_table_sorted, format = "latex", booktabs = TRUE,
      caption = "Pivot Table Showing User Categories by Calories Burned") %>%
  kable_styling(latex_options = c("scale_down", "hold_position"), font_size = 8) %>%
  column_spec(1, width = "1.5cm") %>%  # Adjust width for ID column
  column_spec(2, width = "2cm") %>%    # Adjust width for Steps column
  column_spec(3, width = "2cm") %>%    # Adjust width for Total Distance column
```

```
  column_spec(4, width = "2cm") %>%    # Adjust width for Tracker Distance column
  column_spec(5, width = "2cm") %>% # Wider width for the longest column name
  column_spec(6, width = "2cm") %>%
  column_spec(7, width = "2cm") %>%
  column_spec(8, width = "2cm") %>%
  column_spec(9, width = "2cm") %>%
  column_spec(10, width = "2cm")
```

Table 1: Pivot Table Showing User Categories by Calories Burned

ID	Average Total Steps	Average Total Distance	Average Tracker Distance	Average Logged Activities Distance	Average Calories	Average Lightly Active Minutes	Average Fairly Active Minutes	Average Very Active Minutes	Average Sedentary Minutes
8877689391	17417	1409	1409	0	3451	241	15	67	1046
8378563200	8135	645	645	126	3356	169	10	55	684
5577150313	8608	645	645	0	3300	157	28	82	666
4020332650	5777	414	414	0	3075	130	8	4	1083
1644430081	9275	675	675	0	2916	228	44	15	1034
8053475328	14844	1158	1158	0	2893	154	11	85	1114
4702921684	7943	645	645	0	2821	247	17	3	727
6775888955	5559	399	399	0	2725	124	49	20	1018
7007744171	12260	886	820	286	2627	278	16	45	1001
2022484408	12175	877	877	0	2475	254	23	40	1059
8583815059	3046	238	238	0	2391	137	2	1	1262
2891001357	774	60	0	121	2273	169	83	0	1100
1927972279	2181	151	151	0	2254	112	2	0	953
7086361926	6104	409	409	0	2177	101	18	26	889
6290855005	1618	122	122	0	2166	39	11	8	1290
4445114986	4293	291	291	0	2108	201	1	5	861
6117666160	8249	623	623	0	2099	300	4	1	864
6962181067	12640	865	839	100	2089	259	29	35	606
8792009665	3095	198	198	0	2074	132	5	2	894
2347167796	9800	651	651	0	2021	254	23	12	684
4319703577	7821	526	526	0	1994	249	18	7	780
4057192912	1887	139	141	0	1904	43	5	1	1358
3372868164	6128	422	422	0	1860	287	4	12	1085
4558609924	5785	382	382	0	1830	250	3	4	1085
4388161847	0	0	0	0	1805	0	0	0	1384
5553957443	8355	546	546	0	1803	183	17	24	608
1503960366	11641	761	761	0	1796	228	16	36	810
6391747486	1337	107	10	97	1763	34	1	5	1262
2873212765	6637	447	447	0	1696	275	6	5	1137
1844505072	3641	241	241	0	1616	159	1	1	1035
2320127002	3138	212	212	0	1532	126	1	1	1249
8253242879	2390	168	168	0	1463	59	5	6	1352
3977333714	8664	581	581	0	1398	207	31	12	709
2026352035	3393	210	210	0	1356	169	0	0	659
1624580081	4226	275	275	0	1353	121	1	1	1278

Most Active Users

To determine the most active users, the pivot table was sorted in descending order by average calories burned. All 34 users in the dataset were then categorized as "Lightly Active" (overall average calories burned: 0–2000), "Fairly Active" (2000–3000 calories), or "Very Active" (>3000 calories) over a 30-day period. The following SQL query was used to retrieve the data (commented out):

```
#SELECT
#ID,
#Average_Calories,

#CASE
#  WHEN Average_Calories BETWEEN 0 AND 2000 THEN 'Lightly Active'
#  WHEN Average_Calories BETWEEN 2000 AND 3000 THEN 'Fairly Active'
#  WHEN Average_Calories > 3000 THEN 'Very Active'

#END AS activity_level
```

```
#FROM
#[Project_Name].Pivot_Table


#ORDER BY activity_level
```

The pie chart below shows the activity trend based on the overall average Calories.

```
data<- read.csv("CATEGORIZED_USERS-ON_ACTIVITY_LEVEL.csv")

library(ggplot2)
library(dplyr)


# Culculate the percentage for each activity level category
summary_data <- data %>%
  count(activity_level) %>%
  mutate(
    perc = n / sum(n) * 100,
    label = paste0(activity_level, "\n", round(perc, 0), "%")
  )

# Generate the Pie Chart
ggplot(summary_data, aes(x = "", y = n, fill = activity_level)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar("y") +
  geom_text(aes(label = label), position = position_stack(vjust = 0.5)) +
  theme_void() +
  labs(title = "User Activity Level Distribution") +
  scale_fill_brewer(palette = "RdBu")
```

It is evident from the pie chart that almost 50% of users are fairly active. The overall average calories were chosen as the metric to determine the most active users of the application, as this encompasses all types of activities. Using another metric like Logged Activities Distance would only include specific workouts and record activity initiated by the user.

Hourly Trends

Next, hourly trends were analyzed to study the application's 10 most active users and suggest improvements. The analysis began by tracking hourly trends based on user calories, performed separately for daytime and nighttime hours. The query below was used to retrieve user information in conjunction with the constructed pivot table (commented out)

```
#SELECT
#  a.Id,
#  FORMAT_TIME("%H:%M:%S", TIME(a.ActivityHour)) AS Hour_Of_Day,
#  AVG(a.Calories) AS AverageCalories
#FROM
#  `[Project_Name].Hourly_Calories` a
#INNER JOIN (
#  SELECT ID
#  FROM `[Project_Name].Pivot_Table`
#  WHERE Average_Calories >= 2000
#  ORDER BY Average_Calories DESC
```

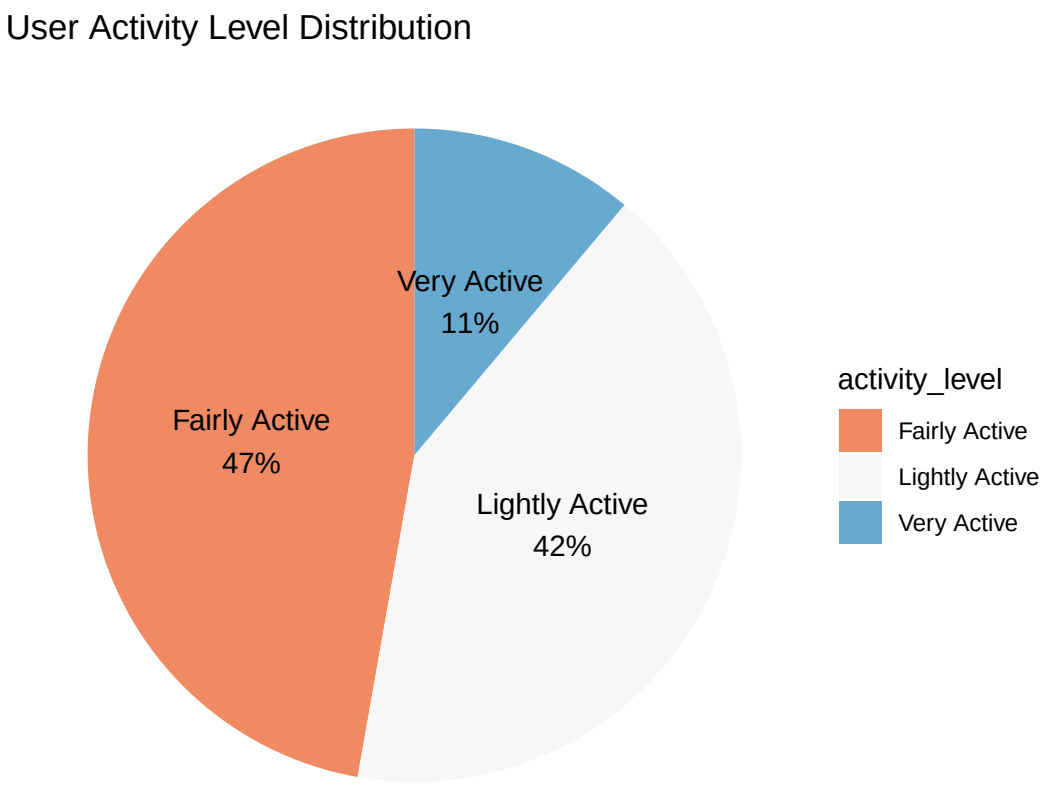

Figure 1: User activity level categorization based on calories burned

```
#  LIMIT 10
#) top_users
#ON a.Id = CAST(top_users.ID AS INT64)
#WHERE EXTRACT(HOUR FROM ActivityHour) <  12 (or > 12 for nighttime users)
#GROUP BY Id, Hour_Of_Day
#ORDER BY Id, Hour_Of_Day;
```

The average hourly calories burned by the top 10 users were then plotted for both daytime and nighttime:

```
every_nth <- function(n) {
  function(x) x[seq(1, length(x), by = n)]
}


data2 <- read.csv("HOURLY_CALORIES_MOST_ACTIVE_USERS_DAY_TO_NIGHT.csv")
ggplot(data = data2) +

  geom_col(mapping = aes(x = Hour_Of_Day, y = AverageCalories, fill = AverageCalories),
           width = 0.5) +
  facet_wrap(~Id) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(x = "Activity Hour", y = "Calories Burned",
       title = "Hourly Calories Burned by Most Active Users-Daytime") +
  scale_fill_viridis_c() +
  scale_x_discrete(breaks = every_nth(2))
```

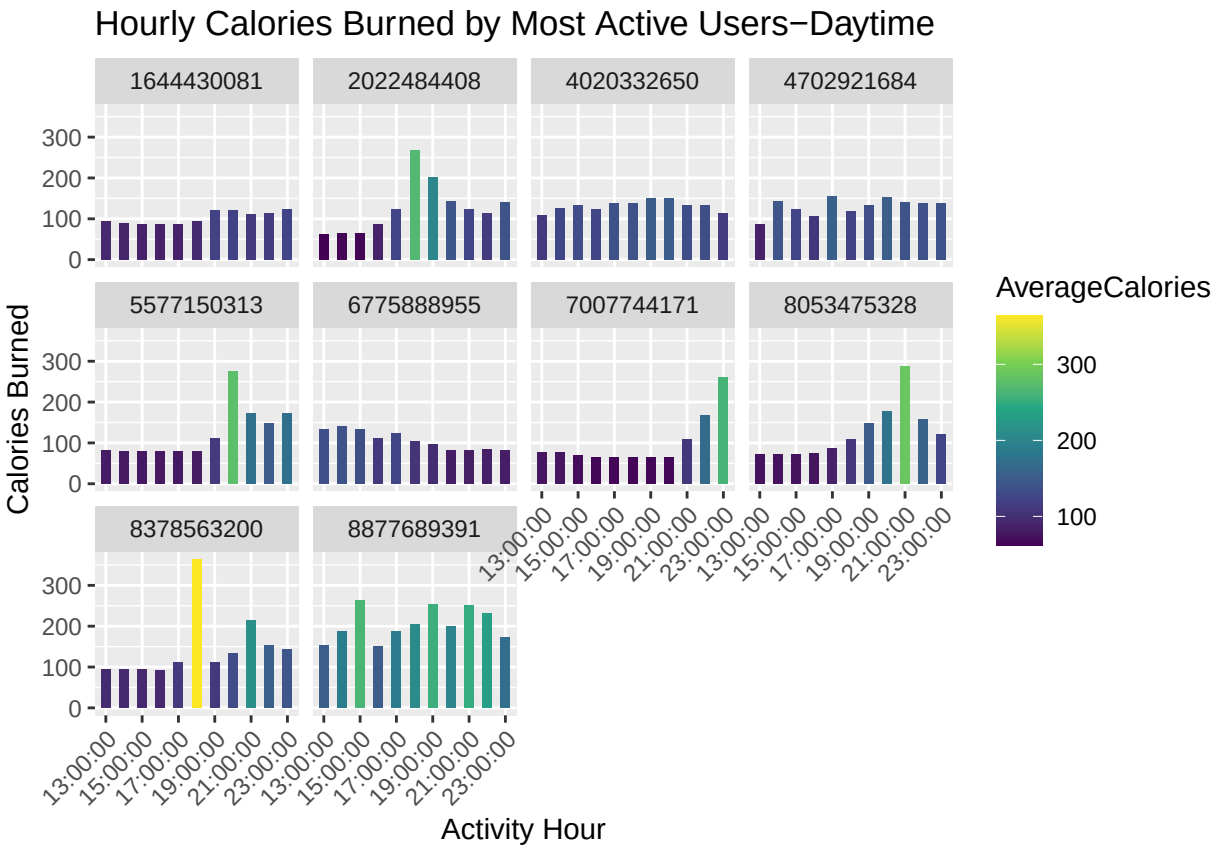

Figure 2: Daily calorie burning trends of the most active users during daytime.

```r
every_nth <- function(n) {
  function(x) x[seq(1, length(x), by = n)]
}


data2 <- read.csv("HOURLY_CALORIES_MOST_ACTIVE_USERS_NIGHT_TO_DAY.csv")
ggplot(data = data2) +

  geom_col(mapping = aes(x = Hour_Of_Day, y = AverageCalories, fill = AverageCalories),
           width = 0.5) +
  facet_wrap(~Id) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(x = "Activity Hour", y = "Calories Burned",
       title = "Hourly Calories Burned by Most Active Users-Nighttime") +
  scale_fill_viridis_c() +
  scale_x_discrete(breaks = every_nth(2))
```

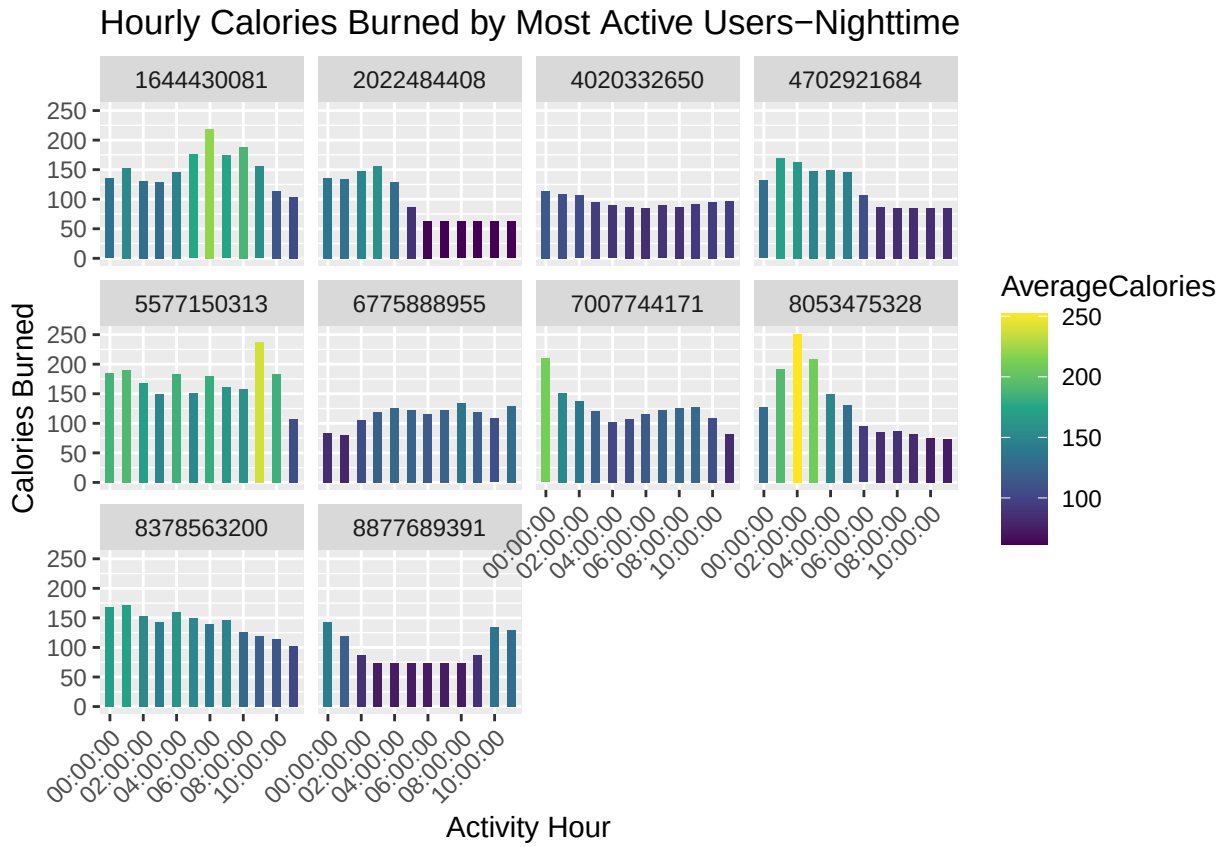

Figure 3: Daily calorie burning trends of the most active users during nighttime.

It is evident that during daytime, 40% of the most active users exhibit a peak in their daily calorie burn. This peak could be due to exercise; however, these people could work nightshifts, so more metrics must be considered. As evident from Figure 2, 30% of users exhibited a peak during nighttime.

```r
data3 <- read.csv("HOURLY_STEPS_MOST_ACTIVE_USERS_MORNING_TO_NIGHT.csv")

data3_avg <- data3 %>%
```

```
  group_by(Id,  Hour_Of_Day) %>%
  summarise(
    Avg_Steps = mean(Total_Steps, na.rm = TRUE)  # Average across days
  )
```

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

```
ggplot(data = data3_avg) +

  geom_col(mapping = aes(x = Hour_Of_Day, y = Avg_Steps, fill = Hour_Of_Day),
           width = 0.5) +
  facet_wrap(~Id) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(x = "Activity Hour", y = "Steps",
       title = "Hourly Steps  by Most Active Users-Daytime") +
  scale_x_discrete(breaks = every_nth(2))+
  scale_fill_viridis_d()
```

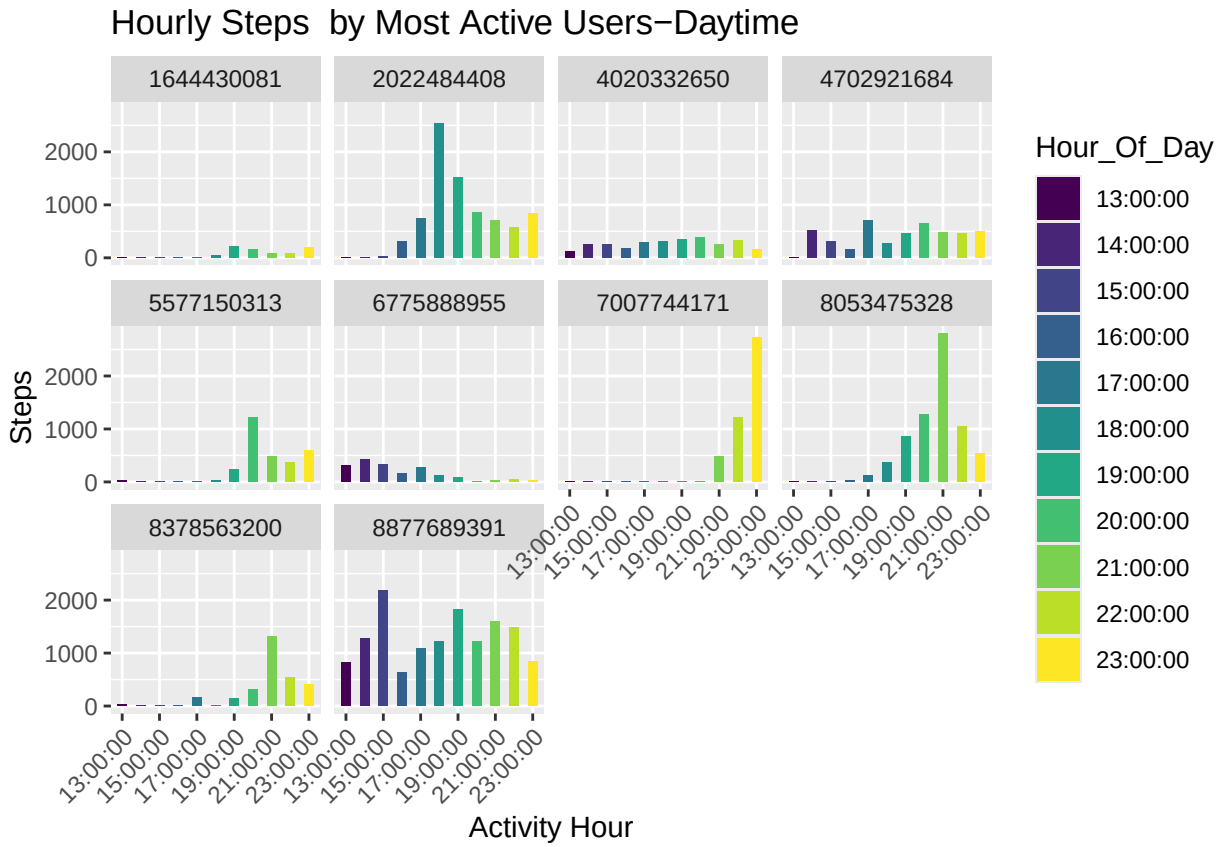

Figure 4: Evolution of average steps per hour during daytime for the most active users

```
data10 <- read.csv("HOURLY_STEPS_MOST_ACTIVE_USERS_NIGHT_TO_MORNING.csv")

data10_avg <- data10 %>%
  group_by(Id,  Hour_Of_Day) %>%
  summarise(
    Avg_Steps = mean(Total_Steps, na.rm = TRUE)  # Average across days
```

```
  )
```

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

```
ggplot(data = data10_avg) +

  geom_col(mapping = aes(x = Hour_Of_Day, y = Avg_Steps, fill = Hour_Of_Day),
           width = 0.5) +
  facet_wrap(~Id) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(x = "Activity Hour", y = "Steps",
       title = "Hourly Steps  by Most Active Users-Nighttime") +
  scale_x_discrete(breaks = every_nth(2))+
  scale_fill_viridis_d()
```

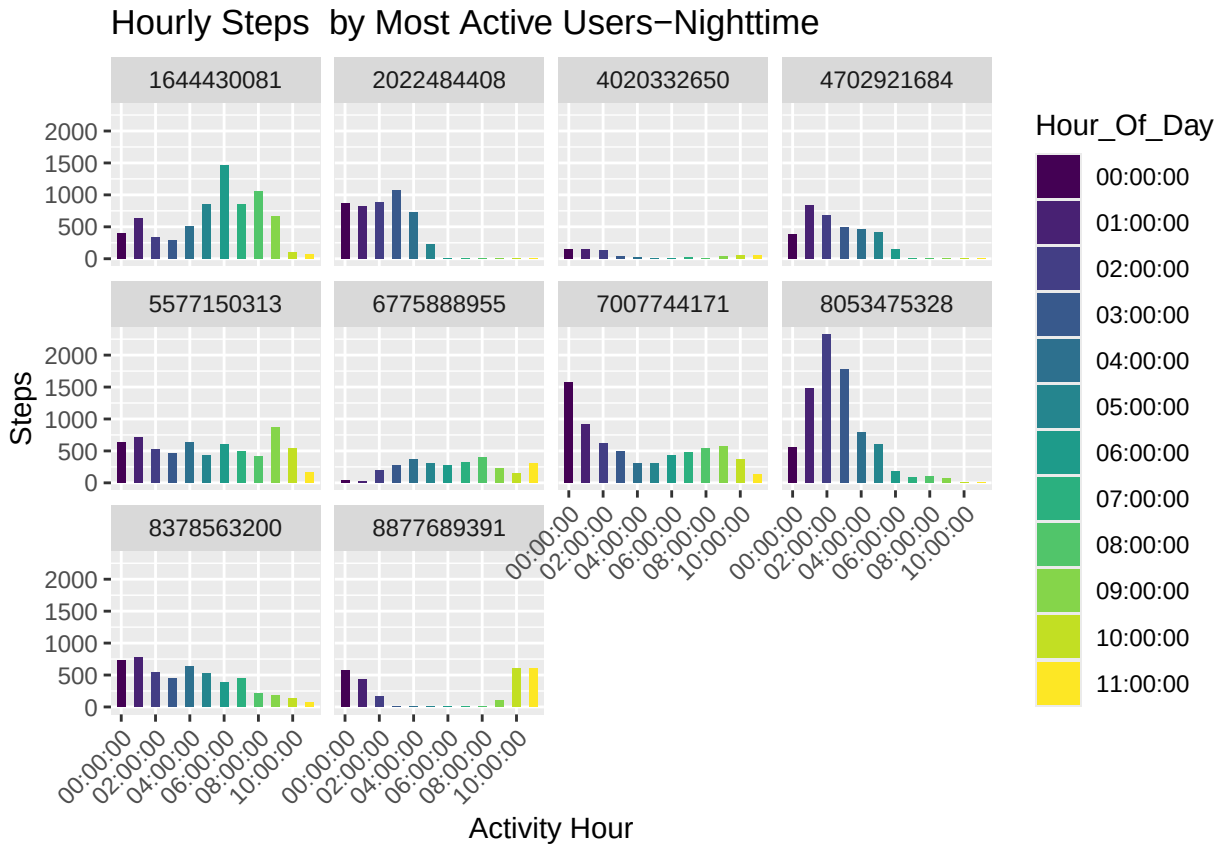

Figure 5: Average hourly steps of the most active users during nighttime

Figures 4 and 5 show the hourly evolution of average steps. The same 50% of users that exhibited peaks in their calorie burning also exhibit a peak in their hourly steps from late evening to night. These results demonstrate that these users are more active during this time period; however, it is still not clear if this is due to simple movement, activity, or workouts. I investigated this by analyzing workout intensity data.

```
data11 <- read.csv("HOURLY_INTENSITIES_MOST_ACTIVE_USERS_DAY_TO_NIGHT.csv")

data11_avg <- data11 %>%
  mutate(Average_Intensity = as.numeric(as.character(Average_Intensity))) %>%  # Ensure numeric
```

```
  group_by(Id, Hour_Of_Day) %>%
  summarise(
    Avg_Intensity = mean(Average_Intensity, na.rm = TRUE),
    .groups = "drop"
  )
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `Average_Intensity =
##   as.numeric(as.character(Average_Intensity))`.
## Caused by warning:
## ! NAs introduced by coercion
```

```
ggplot(data = data11_avg) +
  geom_col(mapping = aes(x = Hour_Of_Day, y = Avg_Intensity, fill = Hour_Of_Day),
           width = 0.5) +
  facet_wrap(~Id) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(x = "Activity Hour", y = "Intensity",
       title = "Hourly Intensities by Most Active Users - Daytime") +
  scale_x_discrete(breaks = every_nth(2))+
  scale_fill_viridis_d()
```

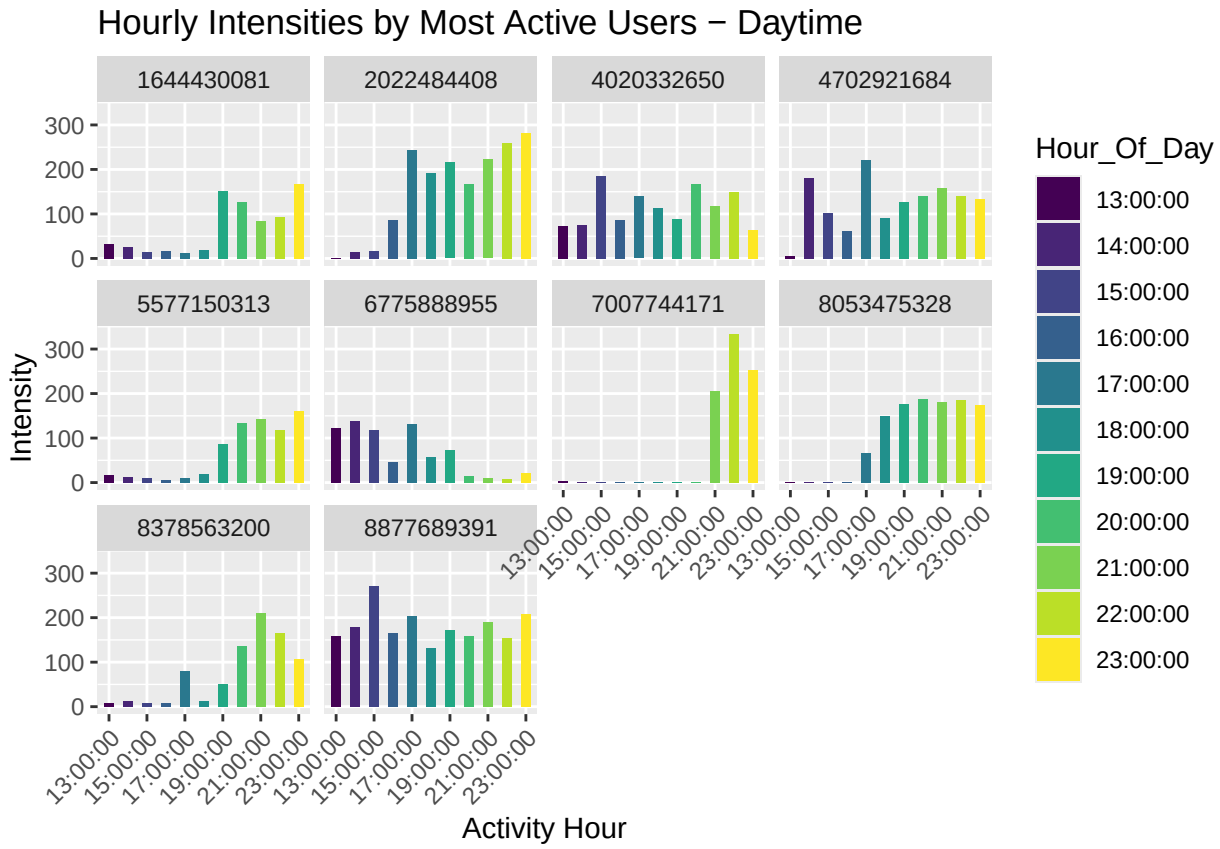

Figure 6: Average workout intensities of the most active users during daytime

```
data12 <- read.csv("HOURLY_INTENSITIES_MOST_ACTIVE_USERS_NIGHT_TO_DAY.csv")

data12_avg <- data12 %>%
```

```r
  mutate(Average_Intensity = as.numeric(as.character(Average_Intensity))) %>%  # Ensure numeric
  group_by(Id, Hour_Of_Day) %>%
  summarise(
    Avg_Intensity = mean(Average_Intensity, na.rm = TRUE),
    .groups = "drop"
  )
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `Average_Intensity =
##   as.numeric(as.character(Average_Intensity))`.
## Caused by warning:
## ! NAs introduced by coercion
```

```r
ggplot(data = data12_avg) +
  geom_col(mapping = aes(x = Hour_Of_Day, y = Avg_Intensity, fill = Hour_Of_Day),
           width = 0.5) +
  facet_wrap(~Id) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(x = "Activity Hour", y = "Intensity",
       title = "Hourly Intensities by Most Active Users - Nighttime") +
  scale_x_discrete(breaks = every_nth(2))+
  scale_fill_viridis_d()
```

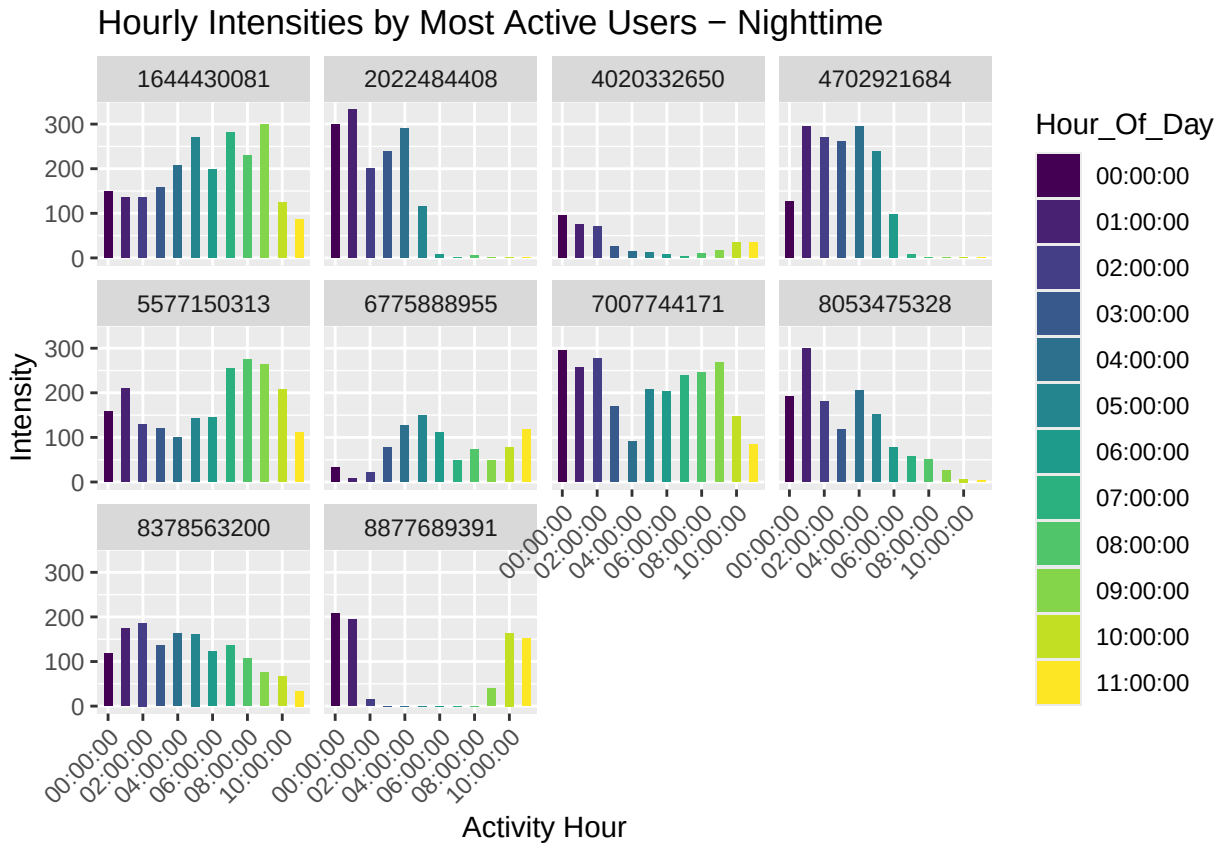

Figure 7: Average workout intensities of most active users during nighttime.

Figures 6 and 7 show the workout intensities. The same 50% of users that exhibited a peak during daytime in their calorie burning and steps also exhibit a peak in their workout intensity. This confirms that the previous

analysis is correlated with workouts, not simple movement due to running errands or working nightshifts.

Conclusions

In this study, Bellabeat application data were analyzed. Users were categorized based on their daily habits, specifically daily calorie burning. The analysis focused on the 10 most active users to reveal insights from the most committed users. It was found that 50% of the most active users choose to work out in the late evening to nighttime period. I recommend sending notifications to users during this period as a reminder to exercise. Additionally, the application could offer suggestions for users' daily nutritional habits, adjusted based on their workout hours.Future work could explore heart rate data or longer timeframes to further enhance insights.