Bellabeat_Case_Study

Vedanth P Bharadwaj

2025-06-17

Contents

0.1 Step 1: ASK – Defining the Business Task

Company Context:
Bellabeat is a wellness-focused technology company that manufactures health-oriented smart devices designed specifically for women. The company's leadership team is interested in leveraging data to better understand how consumers use similar devices (e.g., Fitbit), with the goal of improving Bellabeat's marketing efforts and customer engagement strategies.

Problem Statement:
To analyze how consumers use fitness and wellness data collected from non-Bellabeat smart devices (like Fitbit) in order to derive actionable insights that can guide and optimize Bellabeat's marketing approach.

Key Stakeholders:
- Urška Sršen – Cofounder and Chief Creative Officer at Bellabeat
- Bellabeat Marketing Analytics Team
- Executive Leadership Team

Business Task:
Analyze consumer usage patterns from Fitbit smart device data to generate insights that will inform and improve Bellabeat's marketing strategy, specifically by: - Understanding activity, sleep, and calorie patterns
- Identifying when and how users engage with their fitness devices
- Helping Bellabeat enhance product positioning, user engagement, and campaign targeting

---

How These Insights Will Drive Decisions: - Provide recommendations for app notifications, product features, and content timing based on peak user activity
- Validate the value of high-intensity exercise and walking goals for calorie burn
- Evaluate opportunities for improved sleep or mindfulness content based on usage gaps
- Improve user segmentation and personalization of the Bellabeat experience

---

1 Step 2: PREPARE – Understanding and Loading the Data

Data Source:
The dataset used is the Fitbit Fitness Tracker Data, which contains 30 users' physical activity, sleep, and calorie metrics gathered from wearable devices. All users consented to the data collection.

Why This Data is Relevant:
The Fitbit dataset represents the type of smart device usage that Bellabeat aims to emulate and surpass. It includes variables that reflect daily habits and wellness trends, making it ideal for extracting insights into user behavior and preferences.

Key Files Used: - `dailyActivity_merged.csv` – Daily summary metrics for steps, calories, and activity intensity
- `minuteSleep_merged.csv` – Minute-by-minute sleep tracking data
- `hourlySteps_merged.csv` – Hour-by-hour step counts
- `hourlyCalories_merged.csv` – Hour-by-hour calorie burn

Data Format and Structure: - Files are in CSV format with wide structure. - Shared identifiers include `id`, `activityDate`, `activityHour`, and `datetime`. - Datasets were loaded using `read.csv()`.

Initial Actions Taken: - Inspected using `str()`, `summary()`, `head()` and `glimpse()` - Checked for number of unique users using `n_distinct()` - Standardized column names and date-time formats using `rename_with()`, `mutate()`, and date functions.

```r
# Load CSV files
daily_activity <- read.csv("dailyActivity_merged.csv")
hourly_steps <- read.csv("hourlySteps_merged.csv")
minutes_sleep <- read.csv("minuteSleep_merged.csv")
hourly_calories <- read.csv("hourlyCalories_merged.csv")
```

Initial Actions Taken:

```r
# Inspect structure
glimpse(daily_activity)
```

```
## Rows: 457
## Columns: 15
## $ Id                       <dbl> 1503960366, 1503960366, 1503960366, 150396036~
## $ ActivityDate             <chr> "3/25/2016", "3/26/2016", "3/27/2016", "3/28/~
## $ TotalSteps               <int> 11004, 17609, 12736, 13231, 12041, 10970, 122~
## $ TotalDistance            <dbl> 7.11, 11.55, 8.53, 8.93, 7.85, 7.16, 7.86, 7.~
## $ TrackerDistance          <dbl> 7.11, 11.55, 8.53, 8.93, 7.85, 7.16, 7.86, 7.~
```

```
## $ LoggedActivitiesDistance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ VeryActiveDistance       <dbl> 2.57, 6.92, 4.66, 3.19, 2.16, 2.36, 2.29, 3.3~
## $ ModeratelyActiveDistance <dbl> 0.46, 0.73, 0.16, 0.79, 1.09, 0.51, 0.49, 0.8~
## $ LightActiveDistance      <dbl> 4.07, 3.91, 3.71, 4.95, 4.61, 4.29, 5.04, 3.6~
## $ SedentaryActiveDistance  <dbl> 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.0~
## $ VeryActiveMinutes        <int> 33, 89, 56, 39, 28, 30, 33, 47, 40, 15, 43, 3~
## $ FairlyActiveMinutes      <int> 12, 17, 5, 20, 28, 13, 12, 21, 11, 30, 18, 18~
## $ LightlyActiveMinutes     <int> 205, 274, 268, 224, 243, 223, 239, 200, 244, ~
## $ SedentaryMinutes         <int> 804, 588, 605, 1080, 763, 1174, 820, 866, 636~
## $ Calories                 <int> 1819, 2154, 1944, 1932, 1886, 1820, 1889, 186~
```

```r
glimpse(hourly_steps)
```

```
## Rows: 24,084
## Columns: 3
## $ Id           <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 150396036~
## $ ActivityHour <chr> "3/12/2016 12:00:00 AM", "3/12/2016 1:00:00 AM", "3/12/20~
## $ StepTotal    <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 8, 551, 1764, 1259, 253, 4470,~
```

```r
glimpse(minutes_sleep)
```

```
## Rows: 198,559
## Columns: 4
## $ Id    <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 1503960366, 1503~
## $ date  <chr> "3/13/2016 2:39:30 AM", "3/13/2016 2:40:30 AM", "3/13/2016 2:41:~
## $ value <int> 1, 1, 1, 1, 1, 1, 2, 2, 1, 1, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ logId <dbl> 11114919637, 11114919637, 11114919637, 11114919637, 11114919637,~
```

```r
glimpse(hourly_calories)
```

```
## Rows: 24,084
## Columns: 3
## $ Id           <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 150396036~
## $ ActivityHour <chr> "3/12/2016 12:00:00 AM", "3/12/2016 1:00:00 AM", "3/12/20~
## $ Calories     <int> 48, 48, 48, 48, 48, 48, 48, 48, 48, 49, 89, 134, 130, 81,~
```

---

2 Step 3: PROCESS – Cleaning and Formatting the Data

Cleaning Tasks Performed: - Converted all column names to lowercase for consistency. - Parsed and standardized datetime fields using `as.Date()` and `as.POSIXct()`. - Extracted separate `date` and `time` components from timestamp columns for hourly datasets. - Verified structure using `str()` and `head()` to ensure accuracy.

Packages Used: - `tidyverse`, `dplyr`, `lubridate`, `skimr`, `janitor`, `here`

Each dataset is now clean, date-formatted, and ready for analysis.

```r
# Clean and format daily activity
daily_activity <- daily_activity %>%
  rename_with(tolower) %>%
  mutate(activitydate = as.Date(activitydate, format = "%m/%d/%Y"))

# Clean and format hourly steps
hourly_steps <- hourly_steps %>%
  rename_with(tolower) %>%
  mutate(activityhour = as.POSIXct(activityhour, format = "%m/%d/%Y %I:%M:%S %p"),
```

```r
         date = as.Date(activityhour),
         time = format(activityhour, "%H:%M:%S"))

# Clean and format hourly calories
hourly_calories <- hourly_calories %>%
  rename_with(tolower) %>%
  mutate(activityhour = as.POSIXct(activityhour, format = "%m/%d/%Y %I:%M:%S %p"),
         date = as.Date(activityhour),
         time = format(activityhour, "%H:%M:%S"))

# Clean and format sleep data
minutes_sleep <- minutes_sleep %>%
  rename_with(tolower) %>%
  mutate(
    datetime = as.POSIXct(date, format = "%m/%d/%Y %I:%M:%S %p"),
    date = as.Date(datetime),
    time = format(datetime, "%H:%M:%S")
  )

# Prepare aggregated sleep data
sleep_summary <- minutes_sleep %>%
  group_by(id, date) %>%
  summarise(total_sleep_minutes = sum(value), .groups = "drop")

# Join sleep with activity for same-day analysis
daily_activity <- daily_activity %>%
  rename(date = activitydate)

activity_sleep <- inner_join(daily_activity, sleep_summary, by = c("id", "date"))

# Shift sleep forward one day for previous-day activity analysis
sleep_summary_next_day <- sleep_summary %>%
  mutate(prev_day = date - 1)

activity_sleep_shifted <- inner_join(
  daily_activity,
  sleep_summary_next_day,
  by = c("id", "date" = "prev_day")
)
```

---

3 Step 4: ANALYZE – Exploring and Summarizing the Data

User Coverage: - `daily_activity` – 35 unique users
- `hourly_steps` – 34 users
- `hourly_calories` – 34 users
- `minute_sleep` – 23 users

Summary Statistics: - Average daily steps: 6,547 (median: 5,986), max: 28,497
- Average daily calories burned: 2,189 (median: 2,062), max: 4,562
- Average sedentary time: 995 minutes (~16.5 hours), max: 1,440

These patterns show a wide range of activity levels, with many users having highly sedentary days.

```r
n_distinct(daily_activity$id)
```

```
## [1] 35
```

```r
n_distinct(hourly_steps$id)
```

```
## [1] 34
```

```r
n_distinct(hourly_calories$id)
```

```
## [1] 34
```

```r
n_distinct(minutes_sleep$id)
```

```
## [1] 23
```

```r
summary(daily_activity$totalsteps)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0    1988    5986    6547   10198   28497
```

```r
summary(daily_activity$calories)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0    1776    2062    2189    2667    4562
```

```r
summary(daily_activity$sedentaryminutes)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    32.0   728.0  1057.0   995.3  1285.0  1440.0
```

---

4 Step 5: SHARE – Visualizing and Interpreting Insights

4.1 1. Plot: Total Steps vs. Calories Burned

This plot shows the relationship between the number of steps taken in a day and the total calories burned.

```r
ggplot(daily_activity, aes(x = totalsteps, y = calories)) +
  geom_point(alpha = 0.6, color = "darkblue") +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(
    title = "Relationship Between Steps and Calories Burned",
    x = "Total Steps",
    y = "Calories Burned"
  ) +
  theme_minimal()
```

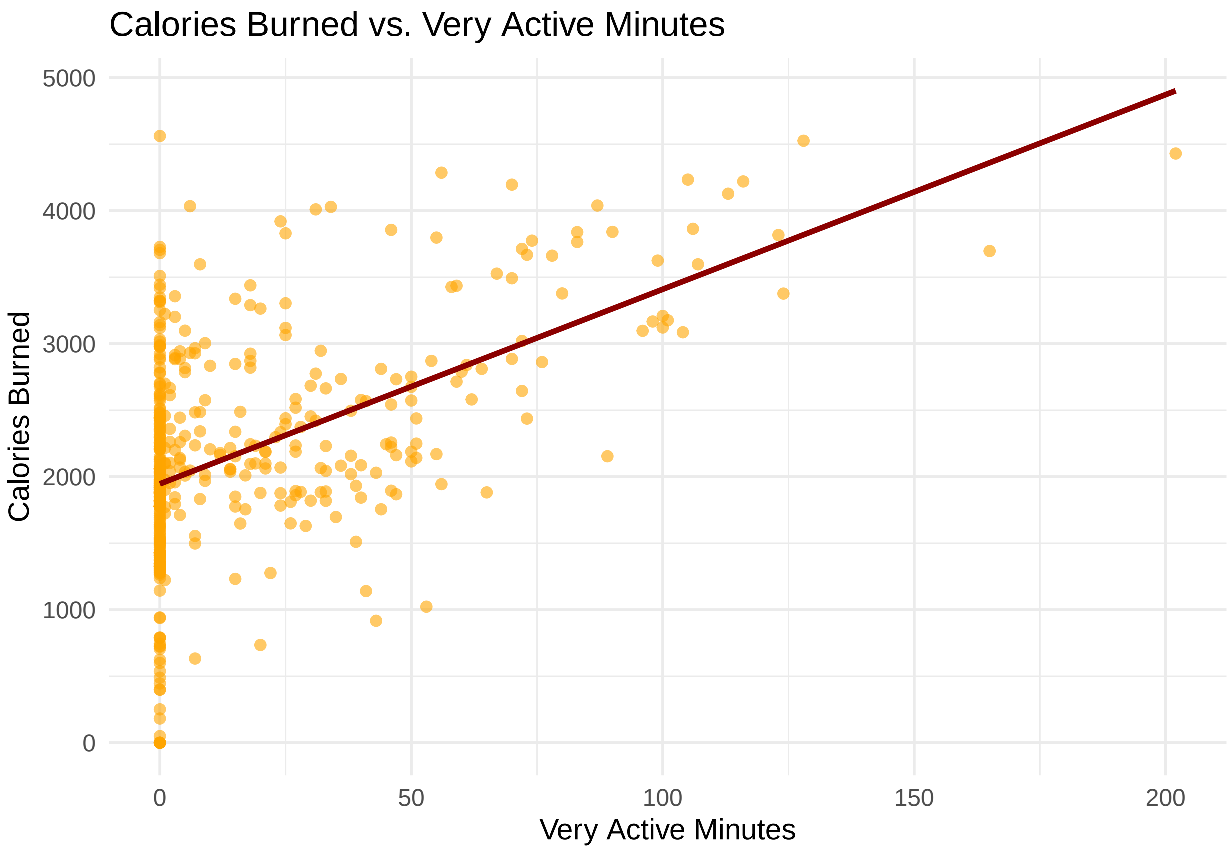
Interpretation:
There is a clear positive correlation between total steps and calories burned. As users take more steps, they tend to burn more calories. This supports the hypothesis that daily movement contributes directly to energy expenditure.

4.2 2. Plot: Very Active Minutes vs. Calories Burned

This plot examines whether intense physical activity (very active minutes) leads to more calories burned.

```
ggplot(daily_activity, aes(x = veryactiveminutes, y = calories)) +
  geom_point(alpha = 0.6, color = "orange") +
  geom_smooth(method = "lm", se = FALSE, color = "darkred") +
  labs(
    title = "Calories Burned vs. Very Active Minutes",
    x = "Very Active Minutes",
    y = "Calories Burned"
  ) +
  theme_minimal()
```

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



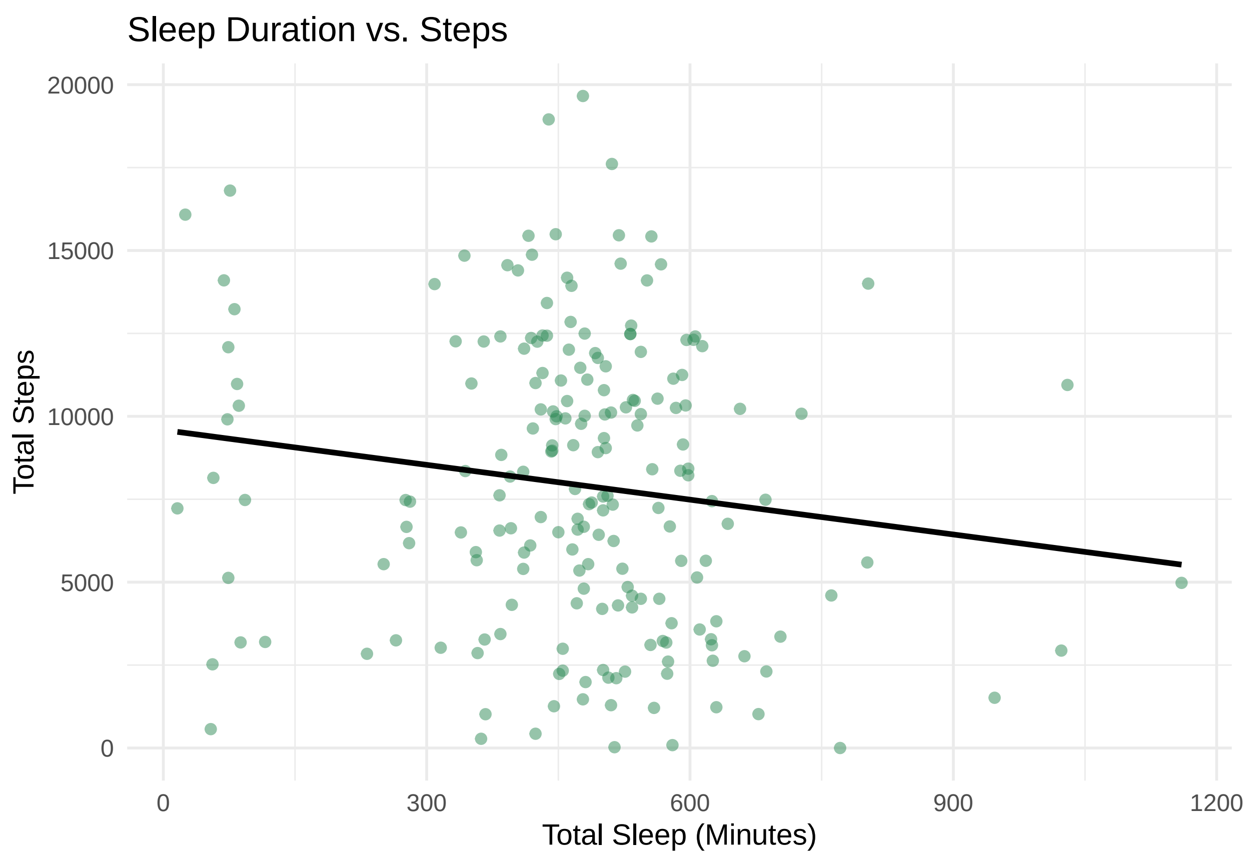
Interpretation:
There is a strong positive relationship between very active minutes and calories burned. This suggests that intense workouts are highly effective in driving energy expenditure.

4.3 3. Plot: Sleep Duration vs. Steps Taken

This plot explores whether users who sleep longer take more steps the next day.

```r
ggplot(activity_sleep, aes(x = total_sleep_minutes, y = totalsteps)) +
  geom_point(alpha = 0.5, color = "seagreen") +
  geom_smooth(method = "lm", se = FALSE, color = "black") +
  labs(
    title = "Sleep Duration vs. Steps",
    x = "Total Sleep (Minutes)",
    y = "Total Steps"
  ) +
  theme_minimal()
```

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



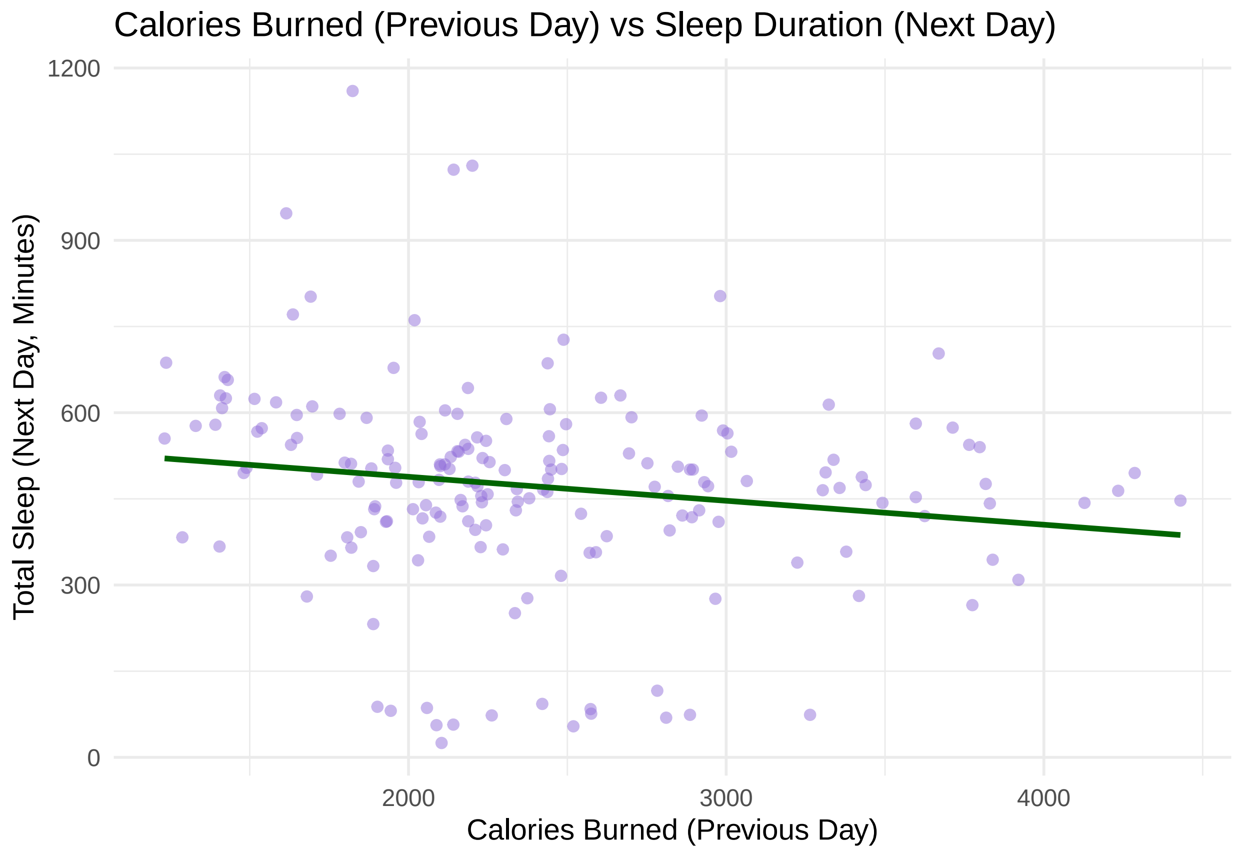
Interpretation:
There is a very slight negative trend. More sleep does not appear to increase next-day activity. It's possible that users rest more after active days or due to lifestyle factors (e.g., weekends, illness).

4.4 4. Plot: Calories Burned vs. Sleep the Following Night

This plot tests a reverse hypothesis — do users sleep more after burning more calories the previous day?

```
ggplot(activity_sleep_shifted, aes(x = calories, y = total_sleep_minutes)) +
  geom_point(alpha = 0.5, color = "mediumpurple") +
  geom_smooth(method = "lm", se = FALSE, color = "darkgreen") +
  labs(
    title = "Calories Burned (Previous Day) vs Sleep Duration (Next Day)",
    x = "Calories Burned (Previous Day)",
    y = "Total Sleep (Next Day, Minutes)"
  ) +
  theme_minimal()
```

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



Interpretation:
A slight downward trend suggests that higher calorie burn does not consistently lead to more sleep the next night. Other variables like stress or daily schedules could influence this result.

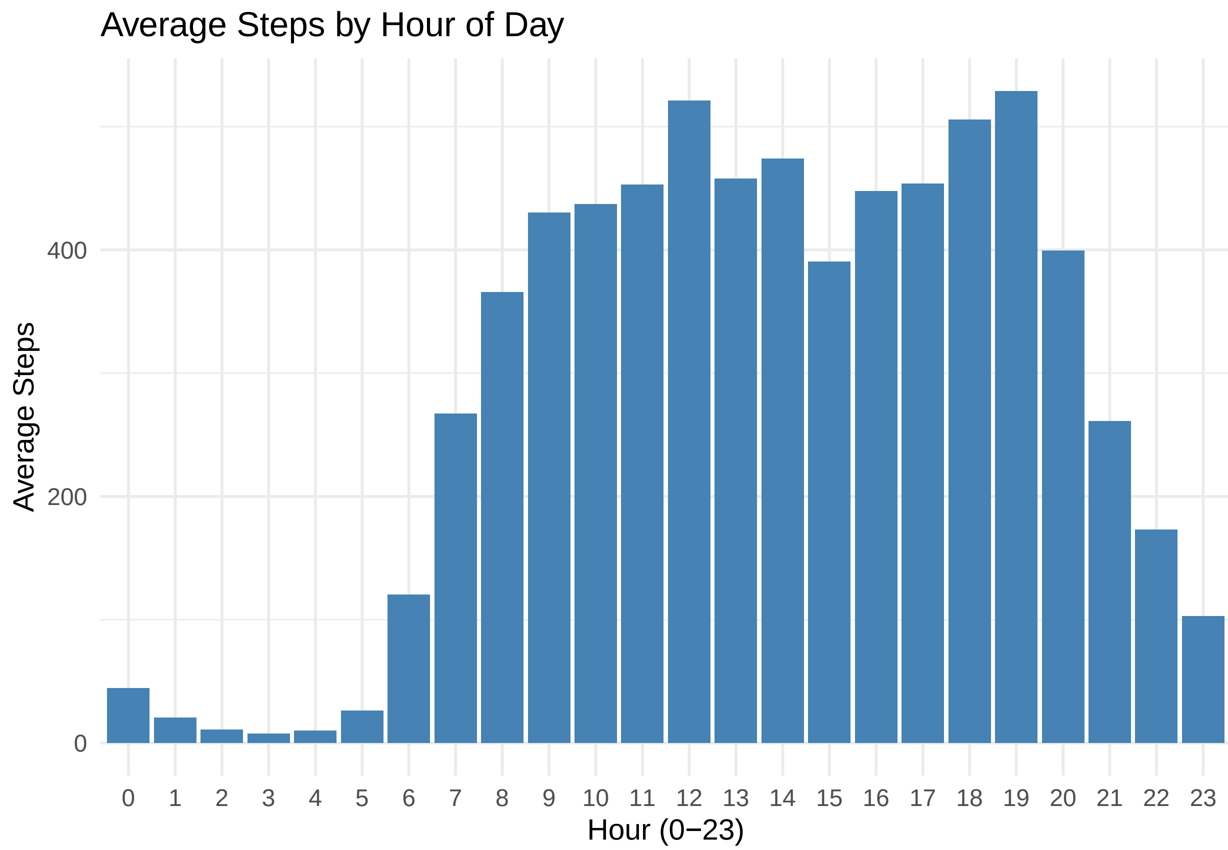
Limitation:
This insight may be skewed by the relatively smaller sample size (23 users) in the sleep dataset.

4.5 5. Plot: Average Steps by Hour of Day

This bar chart shows the average number of steps taken during each hour across all users and days.

```
library(lubridate)

hourly_steps %>%
  mutate(hour = hour(activityhour)) %>%
  group_by(hour) %>%
  summarise(avg_steps = mean(steptotal), .groups = "drop") %>%
  ggplot(aes(x = factor(hour), y = avg_steps)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(
    title = "Average Steps by Hour of Day",
    x = "Hour (0-23)",
    y = "Average Steps"
  ) +
  theme_minimal()
```

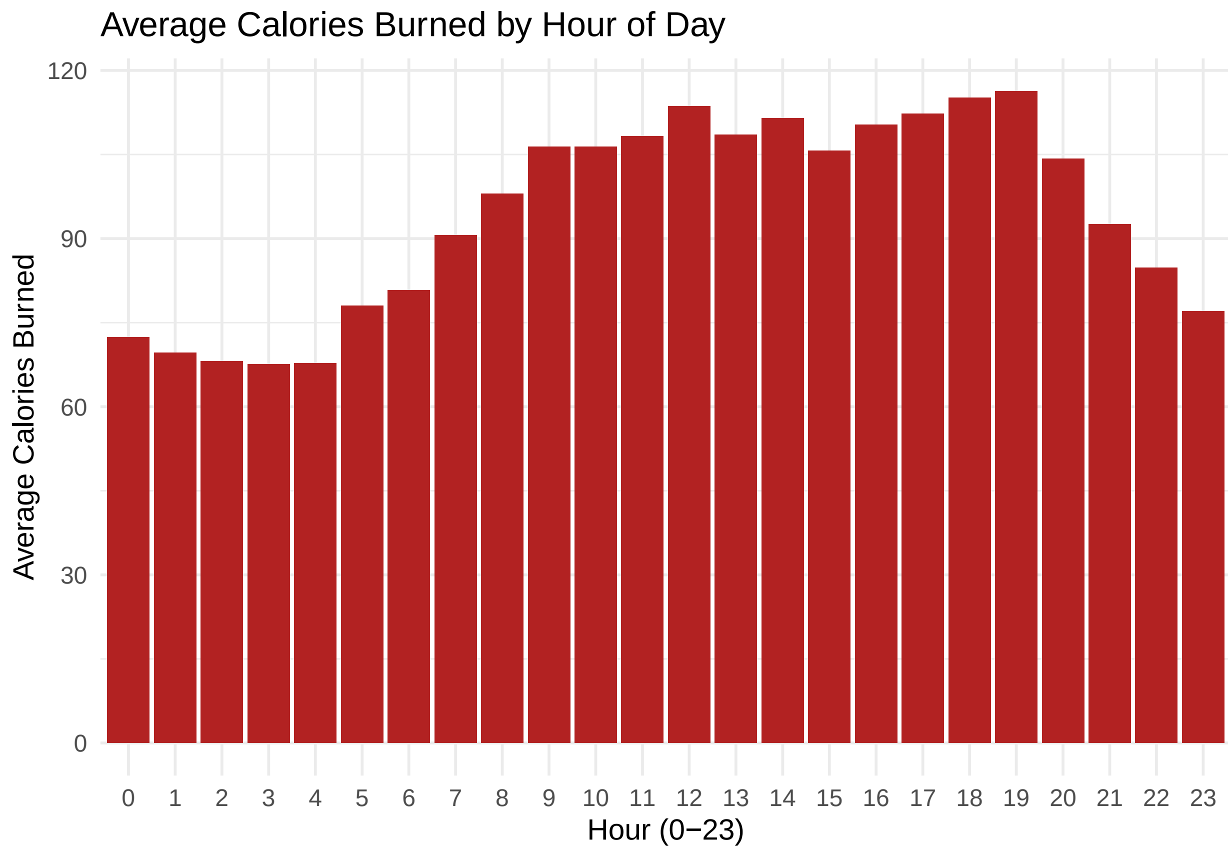


Interpretation:
Activity spikes mid-morning (9 AM – 12 PM) and again in the early evening (5 PM – 7 PM). Early mornings and nights show minimal movement, likely due to rest or sleep.

4.6 6. Plot: Average Calories Burned by Hour of Day

This bar chart shows the average number of calories burned during each hour of the day.

```
hourly_calories %>%
  mutate(hour = hour(activityhour)) %>%
  group_by(hour) %>%
  summarise(avg_calories = mean(calories), .groups = "drop") %>%
  ggplot(aes(x = factor(hour), y = avg_calories)) +
  geom_bar(stat = "identity", fill = "firebrick") +
  labs(
    title = "Average Calories Burned by Hour of Day",
    x = "Hour (0-23)",
    y = "Average Calories Burned"
  ) +
  theme_minimal()
```



Interpretation:
Calorie burn gradually increases after 7 AM, peaking between 11 AM and 7 PM. This aligns with the earlier step pattern and reflects when users are most physically active.

---

5 Step 6: ACT – Recommendations and Next Steps

5.1 Focus Products: Bellabeat Leaf & Bellabeat Time

Bellabeat's wellness wearables — **Leaf** and **Time** — are designed to track activity, sleep, and stress while blending functionality with elegant design. Insights from Fitbit user behavior can inform both product engagement strategies and marketing communications.

5.1.1 For Bellabeat Leaf

Leaf is a flexible wellness tracker that can be worn as a clip, bracelet, or necklace.

5.1.1.1 1. Promote Active Lifestyle Through Step and Movement Goals

- Users with higher step counts burn more calories.
- Introduce step-based challenges and reward streaks in the Bellabeat app.
- Use push notifications in the morning (7–9 AM) to encourage walking routines.

5.1.1.2 2. Emphasize High-Intensity Workouts

- Very active minutes lead to significantly more calorie burn.

- Encourage HIIT or cardio sessions through in-app content and challenges.
- Market Leaf as an effective fitness companion, not just a passive tracker.

5.1.1.3 3. Support Mindfulness and Sleep Quality

- Sleep patterns do not consistently respond to physical activity levels.
- Highlight Leaf's ability to track and improve sleep as a **standalone wellness feature**.
- Add guided breathing, stress tracking, and bedtime reminders.

5.1.2 For Bellabeat Time

Time is a smart analog wellness watch that tracks activity and stress while maintaining a traditional look.

5.1.2.1 1. Leverage Hourly Activity Insights

- Users are most active between **7–11 AM** and **5–7 PM**.
- Sync alerts with these windows to remind Time users to stretch, walk, or hydrate.

5.1.2.2 2. Align with Lifestyle Branding

- Market Time as a product for **women balancing work, fitness, and wellness**.
- Promote use cases like "From morning meetings to evening yoga — Time keeps up with your rhythm."

5.1.2.3 3. Add In-App Productivity Wellness Features

- Introduce focused breathing, productivity breaks, or posture reminders that sync with Time's physical cues.
- Position Time as a **mindful timekeeper** that blends schedule, stress management, and wellness goals.

5.2 Final Conclusion

This case study reveals valuable patterns in smart device usage that Bellabeat can act upon immediately. By aligning marketing, content strategy, and product features for Leaf and Time with real user behaviors: - Bellabeat can drive higher engagement, - Better support women's diverse wellness goals, - And differentiate its products in a crowded wearable market.

Future analysis could explore user clusters, retention behavior, or comparisons across genders or device types to further personalize Bellabeat's offerings.