Bellabeat Case Study

Jessica Y. Yang

2023-01-28

What is Bellabeat?

Bellabeat is a high-tech manufacturer that focuses on health products for women. Bellabeat produces products that collect data on activity, sleep, stress, and reproductive health and has empowered women with knowledge about their own health and habits throughout the years.

Bellabeat's Products

Bellabeat has 5 main products: Bellabeat app, Leaf, Time, Spring, and the Bellabeat membership.

- Bellabeat app: It has health data that is related to one's activity, sleep cycle, stress levels, menstrual cycle, and mindfulness habits. This application can connect to the line of smart wellness products that Bellabeat has created.

- Leaf: This is a wellness tracker and can be worn as a bracelet, necklace, or clip. It tracks activity, sleep cycle, and stress levels.

- Time: This is a wellness watch. It tracks activity, sleep cycle, and stress levels.

- Spring: This is a wellness water bottle. It tracks the daily water intake and hydration levels using smart technology to ensure that users are appropriately hydrated throughout the day.

- Bellabeat membership: Apart from all these products listed above, Bellabeat also offers subscription-based membership program that gives users 24/7 access to fully personalized guidance on nutrition, activity, sleep cycle, health and beauty tips, and mindfulness based on users' lifestyles and goals.

Three Main Questions

These are the three main questions that will guide the future marketing program:

- What are some trends in smart device usage?
- How could theses trends apply to Bellabeat customers?
- How could these trends help influence Bellabeat marketing strategy?

Bellabeat’s Current Marketing Strategies

Bellabeat currently has invested in several different media outlets to gain more users. Bellabeat has invested in traditional advertising media, such as radio, out-of-home billboards, print, and television, but focuses on digital marketing extensively. The company invests year-round in Google Search and maintains active Facebook and Instagram pages, and consistently engages consumers on Twitter. It also runs video advertisements on Youtube and display its advertisements on the Google Display Network to support campaigns around key marketing dates and to increase customer retention.

Ask

The business task

- The business task is to find trends in user’s data and improve Bellabeat customer’s user experiences.

Key stakeholders

- Urska Srsen (Bellabeat’s Cofounder and Chief Creative Officer)
- Sando Mur (Mathematician and Bellabeat’s Cofounder, a key member of the Bellabeat executive team)
- Bellabeat marketing analytics team

Prepare

About the dataset

- The dataset used in this analysis is the FitBit Fitness Tracker Data made available by Mobius stored on Kaggle. The dataset is under CC0: Public Domain license, which means the creator has to waive all his rights to the work worldwide under the copyright law.
- The dataset is generated by respondents to a distributed survey via Amazon Mechanical Turk.
- There are 30 eligible FitBit users consented to the submission of personal tracker data.

Import the data

```
##I installed these packages.
install.packages("tidyverse", repos = "http://cran.us.r-project.org")

## Installing package into 'C:/Users/Admin/AppData/Local/R/win-library/4.2'
## (as 'lib' is unspecified)

## package 'tidyverse' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\Admin\AppData\Local\Temp\RtmpmmDnQU\downloaded_packages
```

```
install.packages("janitor", repos = "http://cran.us.r-project.org")

## Installing package into 'C:/Users/Admin/AppData/Local/R/win-library/4.2'
## (as 'lib' is unspecified)

## package 'janitor' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\Admin\AppData\Local\Temp\RtmpmmDnQU\downloaded_packages

##I loaded these packages.
library(tidyverse)

## ── Attaching packages
## ───────────────────────────────────────
## tidyverse 1.3.2 ──

## ✓ ggplot2 3.4.0      ✓ purrr   1.0.1
## ✓ tibble  3.1.8      ✓ dplyr   1.0.10
## ✓ tidyr   1.2.1      ✓ stringr 1.5.0
## ✓ readr   2.1.3      ✓ forcats 0.5.2
## ── Conflicts ──────────────────────────────────────────
tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()

library(readr)
library(ggplot2)
library(readxl)
library(dplyr)
library(lubridate)

## Loading required package: timechange
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union

library(janitor)

##
## Attaching package: 'janitor'
##
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test

##-----Import Data with readr-----
daily_activity <- read_csv("C:/Users/Admin/Desktop/Data Analytics
Certification/Bellabeat Case Study/archive/Fitabase Data 4.12.16-
5.12.16/dailyActivity_merged.csv")

## Rows: 940 Columns: 15
## ── Column specification
────────────────────────────────────────────────────────
## Delimiter: ","
## chr  (1): ActivityDate
## dbl (14): Id, TotalSteps, TotalDistance, TrackerDistance,
LoggedActivitiesDi...
##
## 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.

daily_calories <- read_csv("C:/Users/Admin/Desktop/Data Analytics
Certification/Bellabeat Case Study/archive/Fitabase Data 4.12.16-
5.12.16/dailyCalories_merged.csv")

## Rows: 940 Columns: 3
## ── Column specification
────────────────────────────────────────────────────────
## Delimiter: ","
## chr (1): ActivityDay
## dbl (2): Id, Calories
##
## 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.

daily_steps <- read_csv("C:/Users/Admin/Desktop/Data Analytics
Certification/Bellabeat Case Study/archive/Fitabase Data 4.12.16-
5.12.16/dailySteps_merged.csv")

## Rows: 940 Columns: 3
## ── Column specification
────────────────────────────────────────────────────────
## Delimiter: ","
## chr (1): ActivityDay
## dbl (2): Id, StepTotal
##
## 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.

daily_intensities <- read_csv("C:/Users/Admin/Desktop/Data Analytics
Certification/Bellabeat Case Study/archive/Fitabase Data 4.12.16-
5.12.16/dailyIntensities_merged.csv")

## Rows: 940 Columns: 10
## ── Column specification
```

```
────────────────────────────────────────────────────────
## Delimiter: ","
## chr (1): ActivityDay
## dbl (9): Id, SedentaryMinutes, LightlyActiveMinutes, FairlyActiveMinutes,
Ve...
##
## 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.
```

```
daily_sleep <- read_csv("C:/Users/Admin/Desktop/Data Analytics
Certification/Bellabeat Case Study/archive/Fitabase Data 4.12.16-
5.12.16/sleepDay_merged.csv")
```

```
## Rows: 413 Columns: 5
## ── Column specification
────────────────────────────────────────────────────────
## Delimiter: ","
## chr (1): SleepDay
## dbl (4): Id, TotalSleepRecords, TotalMinutesAsleep, TotalTimeInBed
##
## 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.
```

```
weight_log <- read_csv("C:/Users/Admin/Desktop/Data Analytics
Certification/Bellabeat Case Study/archive/Fitabase Data 4.12.16-
5.12.16/weightLogInfo_merged.csv")
```

```
## Rows: 67 Columns: 8
## ── Column specification
────────────────────────────────────────────────────────
## Delimiter: ","
## chr (1): Date
## dbl (6): Id, WeightKg, WeightPounds, Fat, BMI, LogId
## lgl (1): IsManualReport
##
## 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.
```

ROCCC Analysis

- Reliable – LOW – the data set was collected from 30 individuals where gender is unknown. The data does not seem complete at all.
- Originality – LOW – the data set is a third party data collected using Amazon Mechanical Turk.
- Comprehensive – MEDIUM – the data set contains multiple fields on daily activity, calories used, daily steps taken, daily sleep time, and daily weight record.

- Current – MEDIUM – the data set is roughly 7 years old but the way people live do not change drastically over the years.
- Cited – HIGH – the data source is very well documented.

Find Column Names and the Number of Distinct IDs

```
##-----Find Column Names-----
colnames(daily_activity)

##  [1] "Id"                       "ActivityDate"
##  [3] "TotalSteps"               "TotalDistance"
##  [5] "TrackerDistance"          "LoggedActivitiesDistance"
##  [7] "VeryActiveDistance"       "ModeratelyActiveDistance"
##  [9] "LightActiveDistance"      "SedentaryActiveDistance"
## [11] "VeryActiveMinutes"        "FairlyActiveMinutes"
## [13] "LightlyActiveMinutes"     "SedentaryMinutes"
## [15] "Calories"

colnames(daily_steps)

## [1] "Id"          "ActivityDay" "StepTotal"

colnames(daily_sleep)

## [1] "Id"                 "SleepDay"           "TotalSleepRecords"
## [4] "TotalMinutesAsleep" "TotalTimeInBed"

colnames(weight_log)

## [1] "Id"             "Date"           "WeightKg"       "WeightPounds"
## [5] "Fat"            "BMI"            "IsManualReport" "LogId"

daily_activity <- daily_activity %>%
  rename("Date" = "ActivityDate")
daily_calories <- daily_calories %>%
  rename("Date" = "ActivityDay")
daily_intensities <- daily_intensities %>%
  rename("Date" = "ActivityDay")
daily_steps <- daily_steps %>%
  rename("Date" = "ActivityDay")
daily_sleep <- daily_sleep %>%
  rename("Date" = "SleepDay")

##-----Number of Distinct IDs-----
n_distinct(daily_activity$Id)

## [1] 33

n_distinct(daily_steps$Id)

## [1] 33

n_distinct(daily_sleep$Id)
```

```
## [1] 24

n_distinct(weight_log$Id)

## [1] 8
```

There are only 8 participants for the weight log, I want to check if there are any significant weight changes.

```
##-----Check Weight Log Data if any Significant Data Change-----
weight_log %>%
  group_by(Id) %>%
  summarize(min(WeightKg), max(WeightKg))

## # A tibble: 8 × 3
##           Id `min(WeightKg)` `max(WeightKg)`
##        <dbl>           <dbl>           <dbl>
## 1 1503960366            52.6            52.6
## 2 1927972279           134.            134.
## 3 2873212765            56.7            57.3
## 4 4319703577            72.3            72.4
## 5 4558609924            69.1            70.3
## 6 5577150313            90.7            90.7
## 7 6962181067            61              62.5
## 8 8877689391            84              85.8
```

Process

Remove any empty rows and columns

```
daily_activity_cleaned <- daily_activity %>%
  remove_empty(which = c("rows")) %>%
  remove_empty(which = c("cols"))
daily_sleep_cleaned <- daily_sleep %>%
  remove_empty(which = c("rows")) %>%
  remove_empty(which = c("cols"))
daily_steps_cleaned <- daily_steps %>%
  remove_empty(which = c("rows")) %>%
  remove_empty(which = c("cols"))
weight_log_cleaned <- weight_log %>%
  remove_empty(which = c("rows")) %>%
  remove_empty(which = c("cols"))
```

There was no empty rows and columns removed.

Change Date to as.Date

```
daily_activity_cleaned$Date <- as.Date(daily_activity_cleaned$Date, format =
"%m/%d/%Y")
daily_sleep_cleaned$Date <- as.Date(daily_sleep_cleaned$Date, format =
"%m/%d/%Y")
daily_steps_cleaned$Date <- as.Date(daily_steps_cleaned$Date, format =
"%m/%d/%Y")
weight_log$Date <- as.Date(weight_log$Date, format = "%m/%d/%Y")
```

Summary statistics of the data sets used

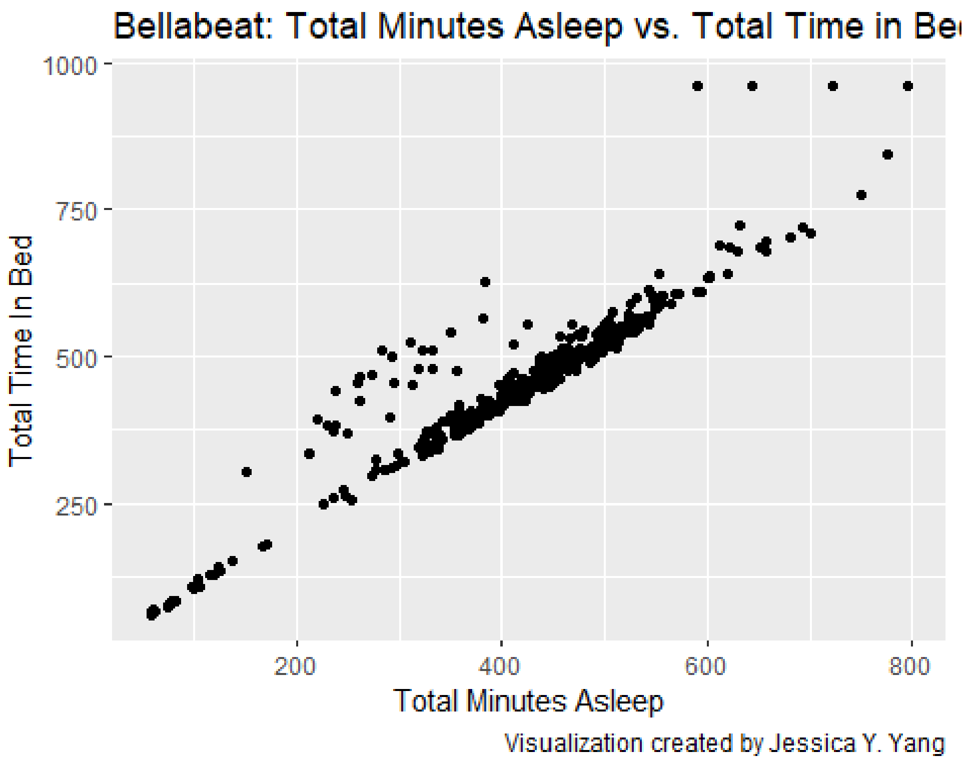
```
##-----Summary of Daily Activity-----
daily_activity_cleaned %>%
  select(TotalSteps,
         TotalDistance,
         SedentaryMinutes,
         VeryActiveMinutes,
         Calories) %>%
  summary()

##    TotalSteps    TotalDistance    SedentaryMinutes VeryActiveMinutes
##  Min.   :    0   Min.   : 0.000   Min.   :   0.0   Min.   :  0.00
##  1st Qu.: 3790   1st Qu.: 2.620   1st Qu.: 729.8   1st Qu.:  0.00
##  Median : 7406   Median : 5.245   Median :1057.5   Median :  4.00
##  Mean   : 7638   Mean   : 5.490   Mean   : 991.2   Mean   : 21.16
##  3rd Qu.:10727   3rd Qu.: 7.713   3rd Qu.:1229.5   3rd Qu.: 32.00
##  Max.   :36019   Max.   :28.030   Max.   :1440.0   Max.   :210.00
##     Calories
##  Min.   :   0
##  1st Qu.:1828
##  Median :2134
##  Mean   :2304
##  3rd Qu.:2793
##  Max.   :4900

##-----Summary of Daily Sleep-----
daily_sleep_cleaned %>%
  select(TotalSleepRecords,
         TotalMinutesAsleep,
         TotalTimeInBed) %>%
  summary()

##  TotalSleepRecords TotalMinutesAsleep TotalTimeInBed
##  Min.   :1.000     Min.   : 58.0      Min.   : 61.0
##  1st Qu.:1.000     1st Qu.:361.0      1st Qu.:403.0
##  Median :1.000     Median :433.0      Median :463.0
##  Mean   :1.119     Mean   :419.5      Mean   :458.6
##  3rd Qu.:1.000     3rd Qu.:490.0      3rd Qu.:526.0
##  Max.   :3.000     Max.   :796.0      Max.   :961.0

##-----Summary of Daily Steps-----
daily_steps_cleaned %>%
  select(StepTotal) %>%
  summary()

##    StepTotal
##  Min.   :    0
##  1st Qu.: 3790
##  Median : 7406
##  Mean   : 7638
##  3rd Qu.:10727
##  Max.   :36019
```

```
##-----Summary of Weight Log-----
weight_log_cleaned %>%
  select(WeightKg,
         WeightPounds,
         BMI) %>%
  summary()
```

```
##     WeightKg       WeightPounds        BMI
##  Min.   : 52.60   Min.   :116.0   Min.   :21.45
##  1st Qu.: 61.40   1st Qu.:135.4   1st Qu.:23.96
##  Median : 62.50   Median :137.8   Median :24.39
##  Mean   : 72.04   Mean   :158.8   Mean   :25.19
##  3rd Qu.: 85.05   3rd Qu.:187.5   3rd Qu.:25.56
##  Max.   :133.50   Max.   :294.3   Max.   :47.54
```

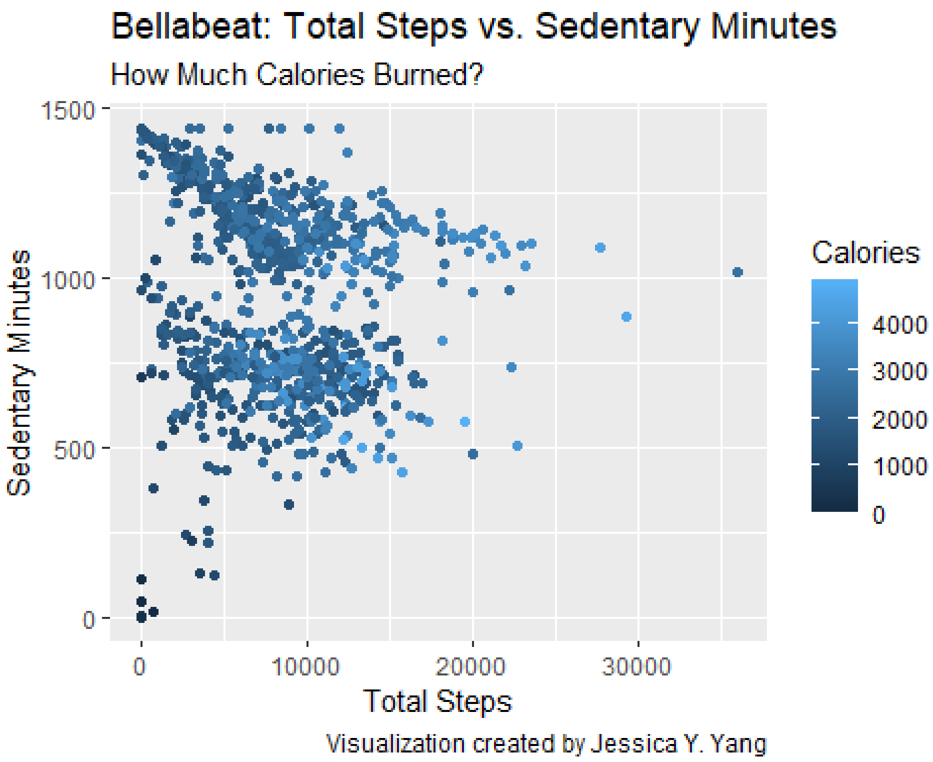
Analyze & Share

```
##-----Total Time Asleep vs. Total Time in Bed (A Visualization)-----
ggplot(data = daily_sleep_cleaned, aes(x = TotalMinutesAsleep, y =
TotalTimeInBed)) +
  geom_point() +
  labs(title = "Bellabeat: Total Minutes Asleep vs. Total Time in Bed",
caption = "Visualization created by Jessica Y. Yang", x = "Total Minutes
Asleep", y = "Total Time In Bed")
```



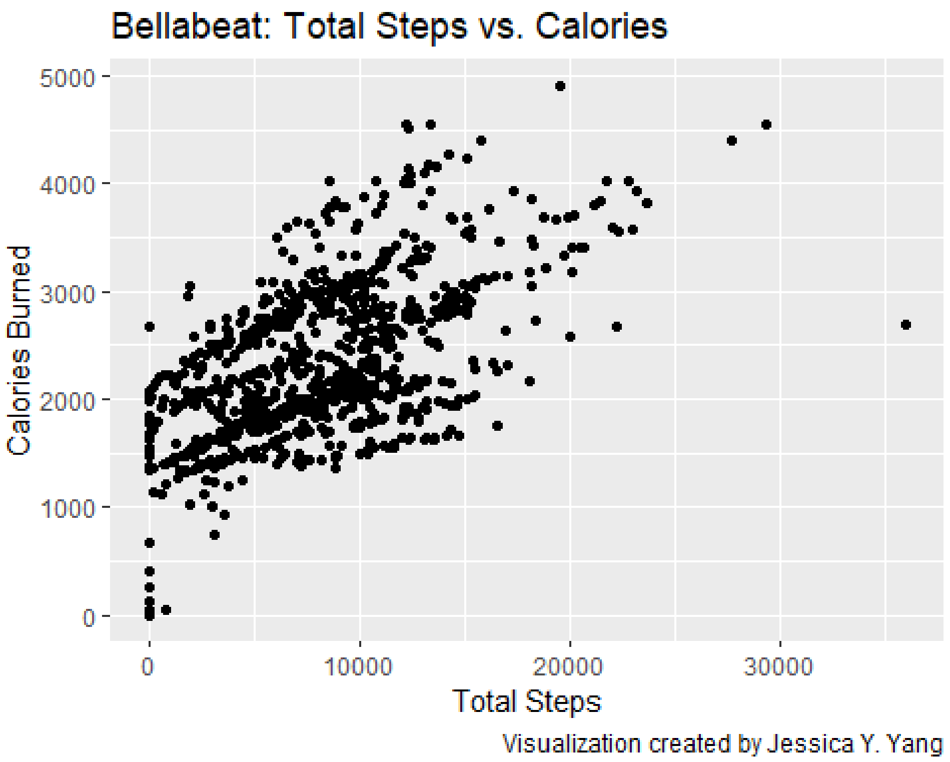
There is a positive correlation between total minutes asleep and total time in bed. This graphs shows that the more time someone spends in bed, the longer he/she is asleep for.

```
##-----Total Steps vs. Sedentary Minutes with Calories as Legend-----
ggplot(data = daily_activity_cleaned, aes(x = TotalSteps, y =
SedentaryMinutes, color = Calories)) +
  geom_point() +
  labs(title = "Bellabeat: Total Steps vs. Sedentary Minutes", subtitle =
"How Much Calories Burned?", caption = "Visualization created by Jessica Y.
Yang", x = "Total Steps", y = "Sedentary Minutes")
```



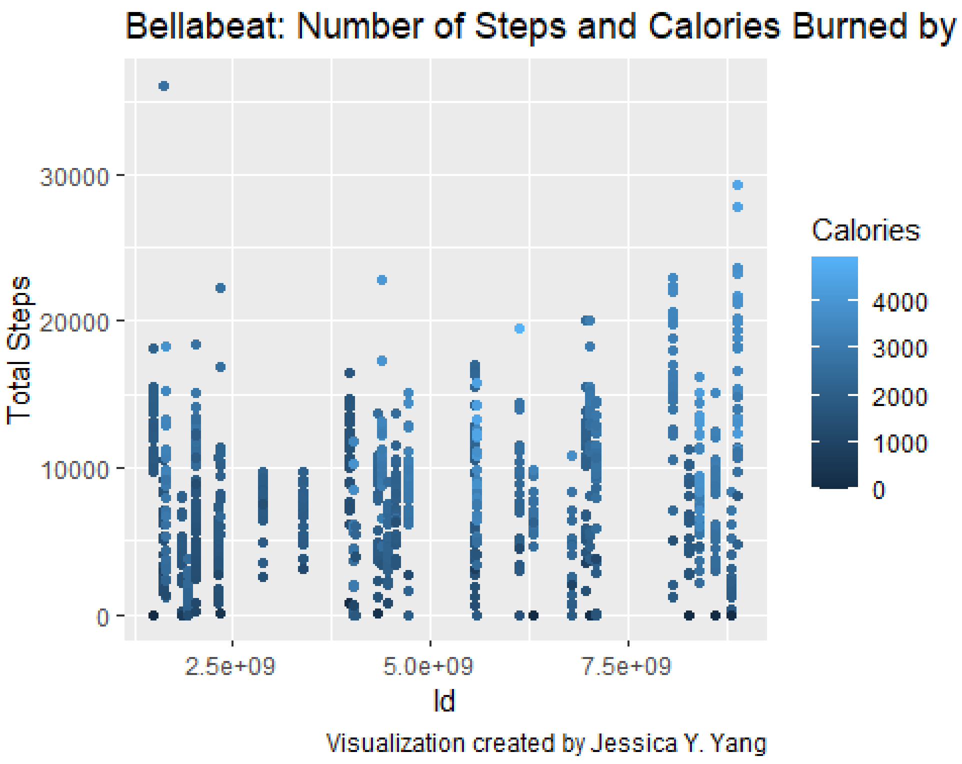
There is a negative correlation between total steps and sedentary minutes. The reason is because a person does not move if he/she is not active. As can be seen, it looks like sedentary minutes has no correlation with calories at all so I will plot a graph between calories and total steps to see the relationship instead.

```
##-----Total Steps vs. Calories-----
ggplot(data = daily_activity_cleaned, aes(x = TotalSteps, y = Calories)) +
  geom_point() +
  labs(title = "Bellabeat: Total Steps vs. Calories", caption =
"Visualization created by Jessica Y. Yang", x = "Total Steps", y = "Calories
Burned")
```



There is a positive correlation between total steps taken and calories burned. Now, I want to take a look at every user's hourly step count vs. hourly calories burned.

```
##-----Total Steps vs. Total Calories by Each User-----
ggplot(data = daily_activity_cleaned, aes(x = Id, y = TotalSteps, color =
Calories)) +
  geom_point() +
  labs(title = "Bellabeat: Number of Steps and Calories Burned by Each
Individual", caption = "Visualization created by Jessica Y. Yang", x = "Id",
y = "Total Steps")
```


Bellabeat: Number of Steps and Calories Burned by

Visualization created by Jessica Y. Yang

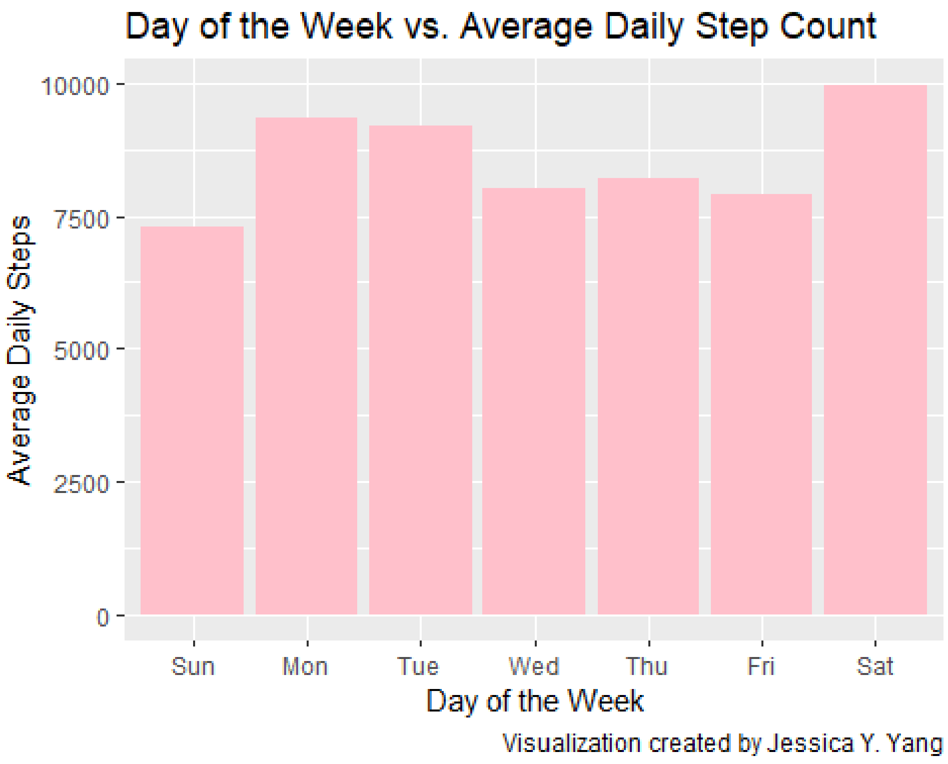
As can be seen, the more steps someone takes the more calories burned.

Now I want to find out if the day of the week affects activity and sleep levels.

```
##-----Merge Sleep and Daily Activity into a Data Frame-----
sleep_daily_activity <- merge(daily_activity_cleaned, daily_sleep_cleaned, by
= c("Id", "Date"))
##-----Add New Variable "Awake Time"-----
sleep_daily_activity <- mutate(sleep_daily_activity, AwakeTime =
TotalTimeInBed - TotalMinutesAsleep)
##-----Aggregate Data by Day of Week-----
sleep_daily_activity <- sleep_daily_activity %>%
  mutate(day_of_week = Date)
sleep_daily_activity$day_of_week <- as.Date(sleep_daily_activity$day_of_week,
format = "%m/%d/%Y")
sleep_daily_activity$day_of_week <- wday(sleep_daily_activity$day_of_week,
label = TRUE)
summarized_sleep_activity <- sleep_daily_activity %>%
  group_by(day_of_week) %>%
  summarize(average_daily_steps = mean(TotalSteps),
            average_sedentary_minutes = mean(SedentaryMinutes),
            average_minutes_asleep = mean(TotalMinutesAsleep),
            average_calories = mean(Calories),
            average_awake_time = mean(AwakeTime),
            average_very_active_minutes = mean(VeryActiveMinutes))
head(summarized_sleep_activity)
```

```
## # A tibble: 6 × 7
##   day_of_week average_daily_steps average_sede…¹ avera…² avera…³ avera…⁴
avera…⁵
##   <ord>                     <dbl>          <dbl>   <dbl>   <dbl>   <dbl>
<dbl>
## 1 Sun                       7298.           688.    453.   2277.    50.8
22.1
## 2 Mon                       9340.           718.    419.   2465.    37.3
32.6
## 3 Tue                       9183.           740.    405.   2496.    38.8
30.6
## 4 Wed                       8023.           714.    435.   2378.    35.3
21.3
## 5 Thu                       8205.           701.    402.   2316.    33.4
22.7
## 6 Fri                       7901.           743.    405.   2330.    39.6
21.2
## # … with abbreviated variable names ¹average_sedentary_minutes,
## #   ²average_minutes_asleep, ³average_calories, ⁴average_awake_time,
## #   ⁵average_very_active_minutes
```

```
##-----Average Daily SAteps vs. Day of the Week-----
ggplot(data = summarized_sleep_activity, aes(x = day_of_week, y =
average_daily_steps)) +
  geom_col(fill = "pink") +
  labs(title = "Day of the Week vs. Average Daily Step Count", caption =
"Visualization created by Jessica Y. Yang", x = "Day of the Week", y =
"Average Daily Steps")
```



As shown in the graph, on average, a person is most active on Saturday and least active on Sundays.

Act

Final conclusion

- The more time someone spends in bed, the more time they are asleep.
- The more sedentary minutes, the less steps taken.
- The more steps taken, the more calories burnt.
- A person is the most active on Saturdays and least active on Sundays.

Top 3 recommendations

- Since people are less active on Sunday and less willingly to go out, Bellabeat can include some Sunday challenges in order to motivate more activity done.
- There is also not much data collected, so I will suggest collecting more data from different competitors so Bellabeat's marketing analytics team and see more trends in usage.
- Since the gender is not revealed, I do not know whether or not the survey was catered towards females only. I suggest collecting data specifically on women using smart devices as Bellabeat focuses on health products for women. * Create a system where the Bellabeat app alerts the user when too much time is spend being sedentary and can assist them in moving and being more active.