BellaBeat Case Study

2022-10-31

1. Ask

Bellabeat is a company who manufactures Smart health products for users specifically women. Their products range from the Bellabeat app which tracks health metrics such as heart rate, steps, calories etc, to wearable products.

As a junior Data Analyst, my job is to find out what the trends are in smart device usage. Then, I need to figure out how these trends can be applied to Bellabeat's current and future customers. Furthermore, I need to also figure out how we can improve Bellabeat's marketing strategy using the data that we have gathered and analyzed.

I need to report all my findings to three stakeholders:

a. **Urška Sršen**: Bellabeat's cofounder and Chief Creative Officer
b. **Sando Mur**: Mathematician and Bellabeat's cofounder; key member of the Bellabeat executive team
c. **Bellabeat marketing analytics team**: A team of data analysts responsible for collecting, analyzing, and reporting data that helps guide Bellabeat's marketing strategy.

2. Prepare

2.a Dataset Source

The data that we will be using is the FitBit Fitness Tracker Data provided by Kaggle. The dataset consists of data taken from 30 (thrity) FitBit users from 03.12.2016-05.12.2016.

2.b Bias, Credibility and Integrity

We will be applying the ROCCC method to rate the bias and credibility of the dataset:

R : Reliable O : Original C : Comprehensive C : Current C : Cited

a. **Reliability** : The data was taken from a sample of 30 users which the gender is not specified. Therefore, it might be a challenge to gather insight on improving the marketing strategy of Bellabeat which caters specifically to women.
b. **Original** : Since the data was taken directly from users using FitBit, therefore it is considered original.
c. **Comprehensive** : The data is quite comprehensive since it provides lots of categories ranging from steps, intensity to sleep; making it quite easy to analyze from different perspective.
d. **Current** : The data is taken from 2016 which is not really up to date. For example, some definitions of "active_minutes" are different from current year 2022 format. Therefore, some adjustments regarding activity intensities that needs to be adjusted
e. **Cited** : the data can be found from this link https://www.kaggle.com/datasets/arashnic/fitbit

2.c Licensing and Acessibility

The data is open-source. Therefore, the date being able to be used by the public without any permissions involved.

3. Process

3.a Environment Setup

First I will set up my environment by installing the necessary packages. For this project I will use 3 packages:

a. tidyverse
b. janitor
c. ggplot2

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

Then I will load all the packages that have been installed.

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.5
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
##
## Attaching package: 'janitor'
##
##
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

3.b Importing Data

For this project, I will be using 2 of Data provided by FitBit:

a. daily_activity
b. hourly_calories

```
daily_activity <- read.csv("dailyActivity_merged.csv")
hourly_calories <- read.csv("hourlyCalories_merged.csv")
```

Then I will preview each on of them

```
glimpse(daily_activity)
```

```
## Rows: 940
## Columns: 15
## $ Id                       <dbl> 1503960366, 1503960366, 1503960366, 150396036~
## $ ActivityDate             <chr> "4/12/2016", "4/13/2016", "4/14/2016", "4/15/~
## $ TotalSteps               <int> 13162, 10735, 10460, 9762, 12669, 9705, 13019~
## $ TotalDistance            <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.59, 9.8~
## $ TrackerDistance          <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.59, 9.8~
## $ LoggedActivitiesDistance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ VeryActiveDistance       <dbl> 1.88, 1.57, 2.44, 2.14, 2.71, 3.19, 3.25, 3.5~
## $ ModeratelyActiveDistance <dbl> 0.55, 0.69, 0.40, 1.26, 0.41, 0.78, 0.64, 1.3~
```

```
## $ LightActiveDistance      <dbl> 6.06, 4.71, 3.91, 2.83, 5.04, 2.51, 4.71, 5.0~
## $ SedentaryActiveDistance  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ VeryActiveMinutes        <int> 25, 21, 30, 29, 36, 38, 42, 50, 28, 19, 66, 4~
## $ FairlyActiveMinutes      <int> 13, 19, 11, 34, 10, 20, 16, 31, 12, 8, 27, 21~
## $ LightlyActiveMinutes     <int> 328, 217, 181, 209, 221, 164, 233, 264, 205, ~
## $ SedentaryMinutes         <int> 728, 776, 1218, 726, 773, 539, 1149, 775, 818~
## $ Calories                 <int> 1985, 1797, 1776, 1745, 1863, 1728, 1921, 203~
```

```
glimpse(hourly_calories)
```

```
## Rows: 22,099
## Columns: 3
## $ Id           <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 150396036~
## $ ActivityHour <chr> "4/12/2016 12:00:00 AM", "4/12/2016 1:00:00 AM", "4/12/20~
## $ Calories     <int> 81, 61, 59, 47, 48, 48, 48, 47, 68, 141, 99, 76, 73, 66, ~
```

3.c Cleaning the Data

Before using the data for analysis, it would be wise if we clean and organize the data so that we are sure that our analysis is accurate.

First, we will check for duplicates

```
sum(duplicated(daily_activity))
```

```
## [1] 0
```

```
sum(duplicated(hourly_calories))
```

```
## [1] 0
```

Since there are no duplicates, we can go a step further. I will make things easier when analyzing later by setting the column names to lowercase and adding an underscore to replace the space.

```
daily_activity <- clean_names(daily_activity, case="snake")
hourly_calories <-clean_names(hourly_calories, case="snake")
```

I will check if the changes have been applied.

```
head(daily_activity)
```

```
##           id activity_date total_steps total_distance tracker_distance
## 1 1503960366     4/12/2016       13162           8.50             8.50
## 2 1503960366     4/13/2016       10735           6.97             6.97
## 3 1503960366     4/14/2016       10460           6.74             6.74
## 4 1503960366     4/15/2016        9762           6.28             6.28
## 5 1503960366     4/16/2016       12669           8.16             8.16
## 6 1503960366     4/17/2016        9705           6.48             6.48
##   logged_activities_distance very_active_distance moderately_active_distance
## 1                          0                1.88                       0.55
## 2                          0                1.57                       0.69
## 3                          0                2.44                       0.40
## 4                          0                2.14                       1.26
## 5                          0                2.71                       0.41
## 6                          0                3.19                       0.78
##   light_active_distance sedentary_active_distance very_active_minutes
## 1                  6.06                         0                  25
## 2                  4.71                         0                  21
## 3                  3.91                         0                  30
```

```
## 4                  2.83                         0                  29
## 5                  5.04                         0                  36
## 6                  2.51                         0                  38
##   fairly_active_minutes lightly_active_minutes sedentary_minutes calories
## 1                    13                    328              728     1985
## 2                    19                    217              776     1797
## 3                    11                    181             1218     1776
## 4                    34                    209              726     1745
## 5                    10                    221              773     1863
## 6                    20                    164              539     1728
```

Since the activity hour in the hourly_calories and hourly_steps data are in 'char' format, we will need to changeinto date and time format.

```
hourly_calories <- hourly_calories %>%
  mutate(activity_hour = as.POSIXct(activity_hour,format ="%m/%d/%Y %I:%M:%S %p" , tz=Sys.timezone()))
```

Since I later plan on analyzing using hours, I want a separate column just for the time.

```
hourly_calories$time <- format(hourly_calories$activity_hour, "%H:%M:%S")
```

Checking if the changes have been applied:

```
glimpse(hourly_calories)
```

```
## Rows: 22,099
## Columns: 4
## $ id            <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 15039603~
## $ activity_hour <dttm> 2016-04-12 00:00:00, 2016-04-12 01:00:00, 2016-04-12 02~
## $ calories      <int> 81, 61, 59, 47, 48, 48, 48, 47, 68, 141, 99, 76, 73, 66,~
## $ time          <chr> "00:00:00", "01:00:00", "02:00:00", "03:00:00", "04:00:0~
```

4. Analyze & Share

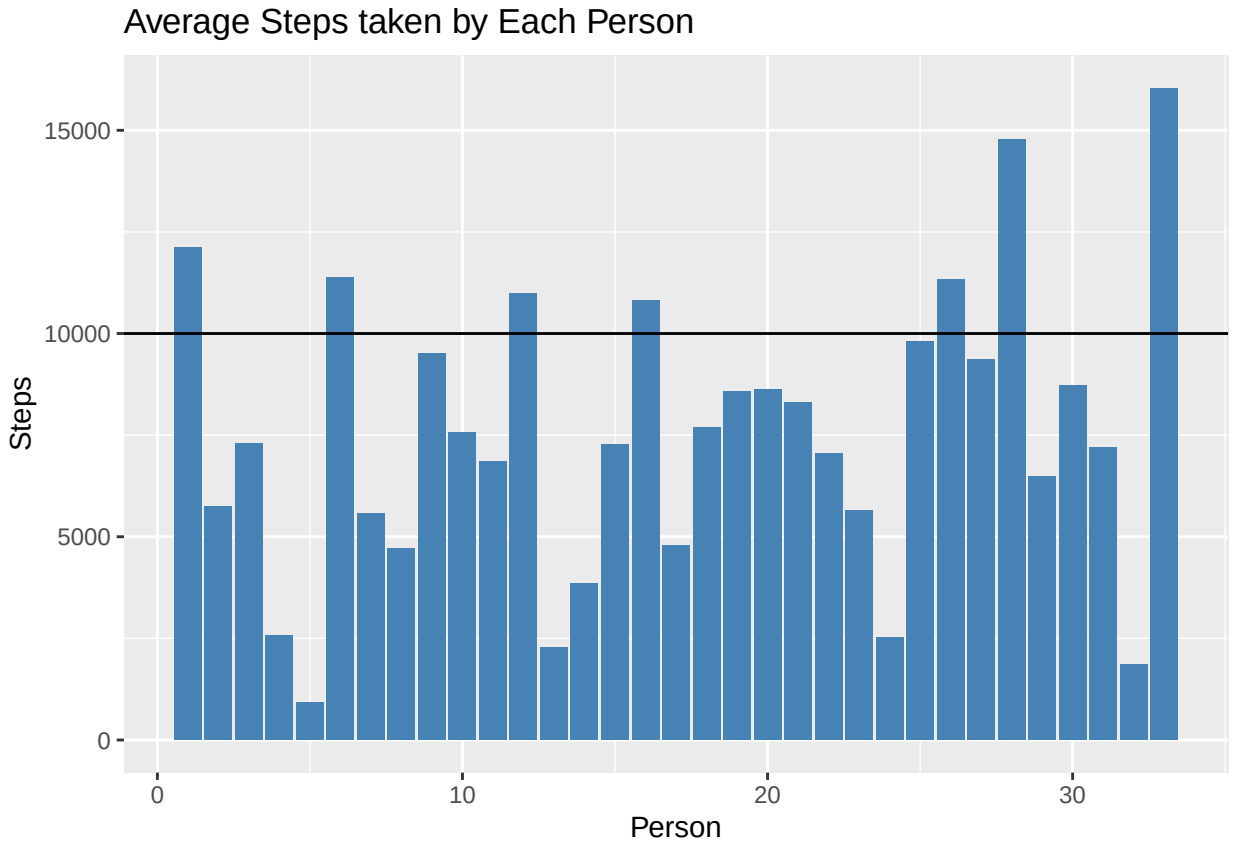
First, from the daily_activity dataset, I want to see approximately how many steps a person takes per day. Thus, I need to first organize the data per person and then find out the average steps they took.

```
avgstepsperid <- daily_activity %>% select(id, total_steps) %>% group_by(id) %>% summarize(average_steps
  mutate(person_id = row_number()) %>% relocate(person_id)
```

Since the id are random numbers, it is much cleaner if I designate a number starting from 1 to make it easier to identify. I will only do this for this specific data.

I will then visualize the data:

```
ggplot(avgstepsperid, aes(x=person_id, y=average_steps))+
  geom_bar(stat="identity", fill="steelblue")+
  geom_hline(yintercept=10000)+
  labs(title = "Average Steps taken by Each Person",
       x = "Person",
       y = "Steps")
```



According to this article, https://www.healthline.com/health/how-many-steps-a-day#Why-10,000-steps? an average person should walk minimal 10 000 steps per day to reduce the risk certain health conditions, such as high blood pressure and heart disease.

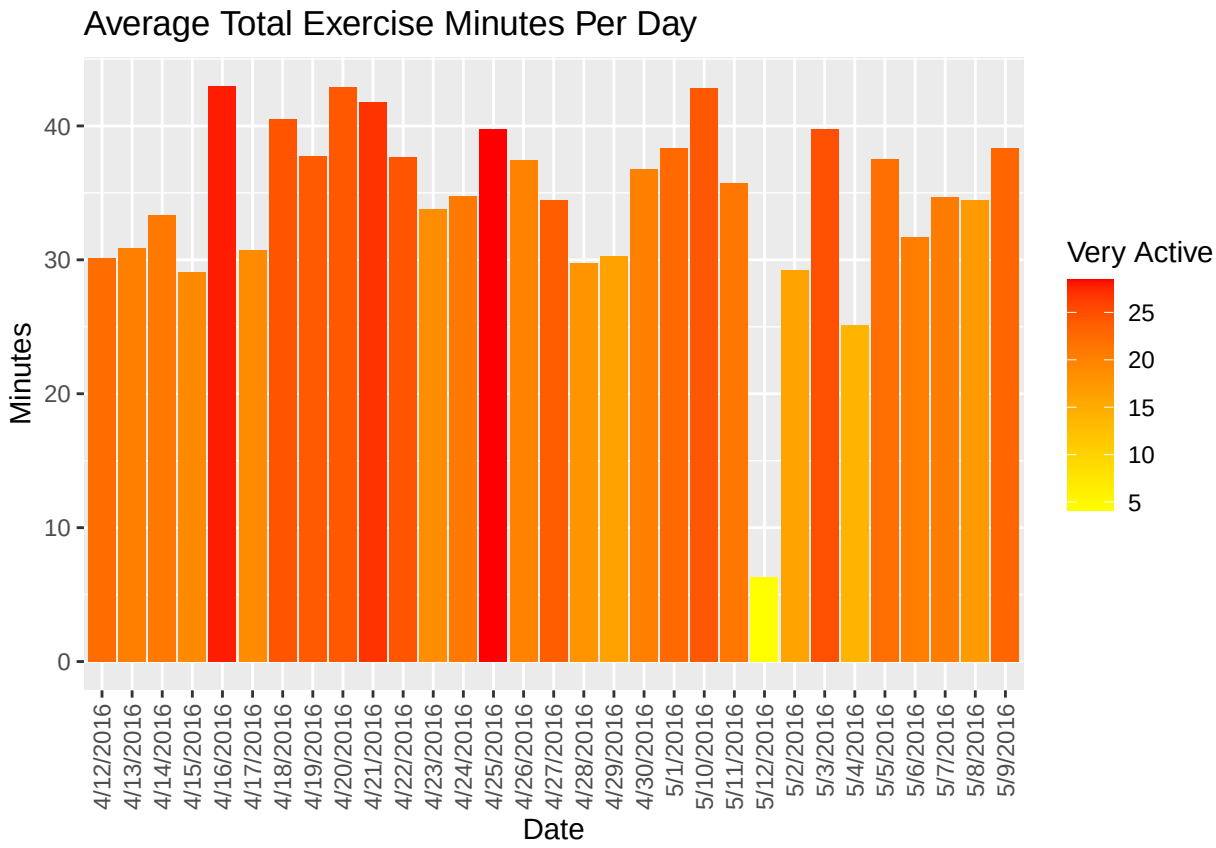
Since we know that people are not reaching their minimum required steps per day, we might find a deeper understanding of each person by how active they are during the day.

The definition of very active and fairly active is quite ambiguous since FitBit does not use the exact terms again in year 2022. Therefore, we need to speculate and decide what these term mean.

For very active minutes, I will consider the activity to be on the around moderately high to high intensity activities. For fairly active, I will consider the activity to be moderately intense.

```
total_exercise_minutes <- daily_activity %>% group_by(activity_date) %>% arrange(activity_date) %>%muta
  summarize(avg_total_active = mean(total_active_minutes), avg_very_active = mean(very_active_minutes))
```

```
ggplot(total_exercise_minutes, aes(y=avg_total_active, x=activity_date, fill=avg_very_active)) +
  geom_bar(stat="identity") + scale_fill_gradient(low="yellow",high="red") + theme(axis.text.x = elemen
  labs(title = "Average Total Exercise Minutes Per Day",
       x = "Date",
       y = "Minutes",
       fill = "Very Active")
```



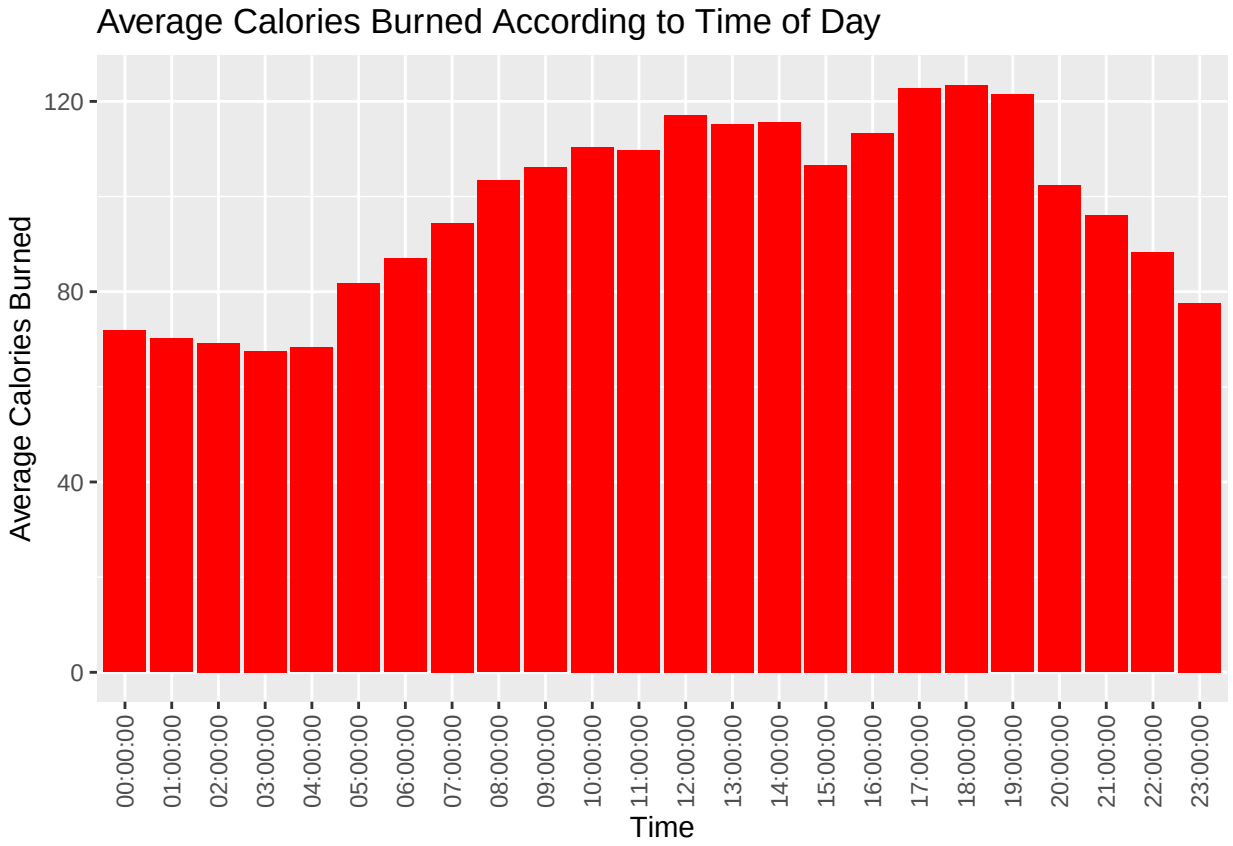
According to this article https://www.mayoclinic.org/healthy-lifestyle/fitness/expert-answers/exercise/faq-20057916 the general goal is to aim for at least 30 minutes of moderate physical activity per day.

As we can see from the graph above, the majority of people have managed to exercise at least 30 minutes a day. On some days most of them exercised intensely for more than 20 minutes.

Another interesting angle to look at is what time do people burn the most calories. These is the result:

```
most_calories_burned <- hourly_calories %>% group_by(time) %>%
    summarize(avg_calories_burned = mean(calories)) %>% arrange(time)
```

```
ggplot(most_calories_burned, aes(x=time, y=avg_calories_burned)) + geom_bar(stat="identity", fill="red"
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+
  labs(title = "Average Calories Burned According to Time of Day",
       x = "Time",
       y = "Average Calories Burned")
```



We can see from the graph that the peak hours where most people burn calories are around the evening times between 17.00 to 19.00. It can be said that people are highly active during these hours but not by a high margin compared to surrounding hours.

5 Act

From the findings there are a few recommendations that I can offer.

5.a. Steps Reminder Notifications

There are still a lot of people who are not reaching their target of 10 000 steps per day. A notification of remaining steps needed to be taken during peak hours where people tend to burn more calories (which from the data was 17.00 - 19.00) would most likely help people achieve their goal of 10 000 steps.

5.b. Workout Type Suggestions

As we can see from the active minutes graph, there are still some people who have not hit their total active exercises minutes per day. A notification with a workout suggestion will help people hit their daily goals. The workout suggestion can be set by intensity levels. For example, if a person has not done any moderately high to high intensity levels of exercise, the Bellabeat app can suggest a template workout program which can help the person achieve that high level of intensity.

5.c Reward system

Since there are still people who have not reached their daily goals, an incentive such as giving rewards would keep them motivated to stay healthy. Instead of just giving achievement badges, I recommend giving them credits to unlock free workout within the app. This will keep people highly motivated because every time

they reach their daily goal, they are collecting credits to unlock a free workout in which they will also do. This is a healthy way to help people sustain their healthy lifestyle

Conclusion

Due to the uniqueness of the recommendation above, I believe new users will start using the Bellabeat platform and existing users will continue enjoy the benefits they get from the app.