**How Can a Wellness Technology Company Play It Smart?**

**Capstone Case Study**

1. **ASK**

**1.1 The Business Task**

To gain insights from the data of non-Bellabeat smart device usage about how customers are using smart device products to then inform the overall Bellabeat marketing strategy.

**1.2 Key stakeholders**

The Executive Team

* Urška Sršen: Co-Founder and CCO
* Sando Mur: Mathmetician and Co-Founder

**1.3 Guiding Questions**

* What relevant trends exist in the data?
* Are the trends relevant to Bellabeat products?
* What do the trends say about how customers are using the devices?
* How will our newfound insights guide the marketing strategy?

1. **PREPARE**

**Intro**

There are 18 spreadsheets that house biometric data across 30 participants. To use all 18 for an analysis is not efficient and is counter productive. So instead, I have selected **five** spreadsheets by using my own intuition into what are the most important things that a Fitbit measures, and how my findings would impact the marketing strategy.

The five I will be using are

* **dailyActivity\_merged** 🡪 A comprehensive view of the tracker’s data that includes some data from the other spread sheets. It measures total steps, activity intensity, calorie intake, and distance travelled. I chose this because it has tracked data that people would typically associate with smart device trackers.
* **Heartrate\_seconds\_merged 🡪** Data about users’ heartrate (BPM). I chose this because one’s heartrate is a good measurement of health.
* **minuteMETsNarrow\_merged 🡪** Measures METs used by the minute. I chose this because how much energy you used at any given time is insightful information for a user
* **sleepDay\_merged 🡪** Measures time in bed and time actually sleeping. I chose this because sleep is essential to health, and tracking sleep is something that can help users.
* **weightLogInfo\_merged 🡪** Manually logged data about users’ weight. I chose this because managing and tracking weight is important to a lot of people.

**Storage**

The spreadsheets have been downloaded and stored locally on my drive, and have also been uploaded to Rstudio Cloud.

See R code below:

install.packages("readr")

install.packages("dplyr")

install.packages("tidyverse")

install.packages("tidyr")

install.packages("magrittr")

install.packages("janitor")

library("readr")

library("dplyr")

library("tidyverse")

library("tidyr")

library("magrittr")

library("janitor")

activity <- read\_csv("dailyActivity\_merged.csv")

hear\_rate <- read\_csv("heartrate\_seconds\_merged.csv")

mets <- read\_csv("minuteMETsNarrow\_merged.csv")

sleep <- read\_csv("sleepDay\_merged.csv")

weight <- read\_csv("weightLogInfo\_merged.csv")

**Organization**

* **dailyActivity\_merged** 🡪 Wide format
* **Heartrate\_seconds\_merged 🡪** Long format
* **minuteMETsNarrow\_merged 🡪** Long format
* **sleepDay\_merged 🡪** Long format
* **weightLogInfo\_merged 🡪** Wide format

**Data Quality**

**Reliable –** The data is reliable. It is actual data recorded from real Fitbits through Amazon Mechanical Turk; 30 people participated in the study.

**Original –** The data is original. The data was publicly posted, but has not been altered, and had been collected directly from the personal tracker data from the devices of the users.

**Comprehensive –** The data is not quite fully comprehensive. A sample size of 30 is way too small to represent a population. Also, nowhere in any of the data provided was there an indicator showing which user was male, and which user was female. This is important to us because the insights we find from this analysis will be applied to products targeted to women. On top of that, for values like distance travelled and calories in the daily activity spreadsheet, there is no context – just a number for the distance (km or miles?) and no indication of its calories logged or burned. For this study, we will assume that it’s miles, and that it’s calories burned.

**Current –** The data is not current, because it is over five years old (analysis performed October 2021).

**Cited –** The data is cited. Acknowledgment of the original study was posted in the description box of the data’s Kaggle page.

With the data not indicating gender, having a small sample size, lacking context, and being outdated, it takes away from the overall quality of the data, but the data is not bad to use because it is credible and still relevant to our project.

1. **PROCESS**

I will be using R to process and analyze the data. I’ll be using this tool because R is the most convenient tool in regards to uploading and cleaning data, especially when it comes to large amounts of data.

See below for R code used to install data cleaning packages:

> install.packages("here")

Installing package into ‘/cloud/lib/x86\_64-pc-linux-gnu-library/4.1’

(as ‘lib’ is unspecified)

also installing the dependency ‘rprojroot’

trying URL 'http://package-proxy/focal/src/contrib/rprojroot\_2.0.2.tar.gz'

Content type 'application/x-tar' length 97139 bytes (94 KB)

==================================================

downloaded 94 KB

trying URL 'http://package-proxy/focal/src/contrib/here\_1.0.1.tar.gz'

Content type 'application/x-tar' length 52032 bytes (50 KB)

==================================================

downloaded 50 KB

\* installing \*binary\* package ‘rprojroot’ ...

\* DONE (rprojroot)

\* installing \*binary\* package ‘here’ ...

\* DONE (here)

The downloaded source packages are in

‘/tmp/RtmpsxDvkH/downloaded\_packages’

> libary('here')

Error in libary("here") : could not find function "libary"

> library("here")

here() starts at /cloud/project

> library(here)

> install.packages("skimr")

Installing package into ‘/cloud/lib/x86\_64-pc-linux-gnu-library/4.1’

(as ‘lib’ is unspecified)

also installing the dependency ‘repr’

trying URL 'http://package-proxy/focal/src/contrib/repr\_1.1.3.tar.gz'

Content type 'application/x-tar' length 123378 bytes (120 KB)

==================================================

downloaded 120 KB

trying URL 'http://package-proxy/focal/src/contrib/skimr\_2.1.3.tar.gz'

Content type 'application/x-tar' length 1219271 bytes (1.2 MB)

==================================================

downloaded 1.2 MB

\* installing \*binary\* package ‘repr’ ...

\* DONE (repr)

\* installing \*binary\* package ‘skimr’ ...

\* DONE (skimr)

The downloaded source packages are in

‘/tmp/RtmpTIIGkU/downloaded\_packages’

> library(skimr)

> install.packages("janitor")

Installing package into ‘/cloud/lib/x86\_64-pc-linux-gnu-library/4.1’

(as ‘lib’ is unspecified)

also installing the dependency ‘snakecase’

trying URL 'http://package-proxy/focal/src/contrib/snakecase\_0.11.0.tar.gz'

Content type 'application/x-tar' length 157410 bytes (153 KB)

==================================================

downloaded 153 KB

trying URL 'http://package-proxy/focal/src/contrib/janitor\_2.1.0.tar.gz'

Content type 'application/x-tar' length 248682 bytes (242 KB)

==================================================

downloaded 242 KB

\* installing \*binary\* package ‘snakecase’ ...

\* DONE (snakecase)

\* installing \*binary\* package ‘janitor’ ...

\* DONE (janitor)

The downloaded source packages are in

‘/tmp/RtmpTIIGkU/downloaded\_packages’

> library(janitor)

Attaching package: ‘janitor’

The following objects are masked from ‘package:stats’:

chisq.test, fisher.test

**Data Preview**

Code & Results:

|  |
| --- |
| head(activity)  # A tibble: 6 × 15  Id ActivityDate TotalSteps TotalDistance TrackerDistance LoggedActivities…  *<dbl>* *<chr>* *<dbl>* *<dbl>* *<dbl>* *<dbl>*  1 1503960366 4/12/2016 13162 8.5 8.5 0  2 1503960366 4/13/2016 10735 6.97 6.97 0  3 1503960366 4/14/2016 10460 6.74 6.74 0  4 1503960366 4/15/2016 9762 6.28 6.28 0  5 1503960366 4/16/2016 12669 8.16 8.16 0  6 1503960366 4/17/2016 9705 6.48 6.48 0  # … with 9 more variables: VeryActiveDistance <dbl>,  # ModeratelyActiveDistance <dbl>, LightActiveDistance <dbl>,  # SedentaryActiveDistance <dbl>, VeryActiveMinutes <dbl>,  # FairlyActiveMinutes <dbl>, LightlyActiveMinutes <dbl>, SedentaryMinutes <dbl>,  # Calories <dbl>  > head(heart\_rate)  # A tibble: 6 × 3  Id Time Value  *<dbl>* *<chr>* *<dbl>*  1 2022484408 4/12/2016 7:21:00 AM 97  2 2022484408 4/12/2016 7:21:05 AM 102  3 2022484408 4/12/2016 7:21:10 AM 105  4 2022484408 4/12/2016 7:21:20 AM 103  5 2022484408 4/12/2016 7:21:25 AM 101  6 2022484408 4/12/2016 7:22:05 AM 95  > head(mets)  # A tibble: 6 × 3  Id ActivityMinute METs  *<dbl>* *<chr>* *<dbl>*  1 1503960366 4/12/2016 12:00:00 AM 10  2 1503960366 4/12/2016 12:01:00 AM 10  3 1503960366 4/12/2016 12:02:00 AM 10  4 1503960366 4/12/2016 12:03:00 AM 10  5 1503960366 4/12/2016 12:04:00 AM 10  6 1503960366 4/12/2016 12:05:00 AM 12  > head(sleep)  # A tibble: 6 × 5  Id SleepDay TotalSleepRecords TotalMinutesAsl… TotalTimeInBed  *<dbl>* *<chr>* *<dbl>* *<dbl>* *<dbl>*  1 1503960366 4/12/2016 12:00:00 AM 1 327 346  2 1503960366 4/13/2016 12:00:00 AM 2 384 407  3 1503960366 4/15/2016 12:00:00 AM 1 412 442  4 1503960366 4/16/2016 12:00:00 AM 2 340 367  5 1503960366 4/17/2016 12:00:00 AM 1 700 712  6 1503960366 4/19/2016 12:00:00 AM 1 304 320  > head(weight)  # A tibble: 6 × 8  Id Date WeightKg WeightPounds Fat BMI IsManualReport LogId  *<dbl>* *<chr>* *<dbl>* *<dbl>* *<dbl>* *<dbl>* *<lgl>* *<dbl>*  1 1503960366 5/2/2016 11… 52.6 116. 22 22.6 TRUE 1.46e12  2 1503960366 5/3/2016 11… 52.6 116. NA 22.6 TRUE 1.46e12  3 1927972279 4/13/2016 1… 134. 294. NA 47.5 FALSE 1.46e12  4 2873212765 4/21/2016 1… 56.7 125. NA 21.5 TRUE 1.46e12  5 2873212765 5/12/2016 1… 57.3 126. NA 21.7 TRUE 1.46e12  6 4319703577 4/17/2016 1… 72.4 160. 25 27.5 TRUE 1.46e12 |
|  |
|  |

First step I took in cleaning the data is removing the “TotalDistance” column in the activity data frame. When I first inspected the data, I noticed the two columns, “TotalDistance” and “TrackerDistance” have the exact same values, so I removed the TotalDistance column.

Code:

activity <- activity %>%

select(-TotalDistance)

head(activity)

#Result:

|  |
| --- |
| # A tibble: 6 × 14  Id ActivityDate TotalSteps TrackerDistance LoggedActivitie… VeryActiveDista…  *<dbl>* *<chr>* *<dbl>* *<dbl>* *<dbl>* *<dbl>*  1 1503960366 4/12/2016 13162 8.5 0 1.88  2 1503960366 4/13/2016 10735 6.97 0 1.57  3 1503960366 4/14/2016 10460 6.74 0 2.44  4 1503960366 4/15/2016 9762 6.28 0 2.14  5 1503960366 4/16/2016 12669 8.16 0 2.71  6 1503960366 4/17/2016 9705 6.48 0 3.19  # … with 8 more variables: ModeratelyActiveDistance <dbl>,  # LightActiveDistance <dbl>, SedentaryActiveDistance <dbl>,  # VeryActiveMinutes <dbl>, FairlyActiveMinutes <dbl>, LightlyActiveMinutes <dbl>,  # SedentaryMinutes <dbl>, Calories <dbl> |
|  |
| |  | | --- | | > | |

Then I removed the “Fat” column from the weight data frame. The column seemed to not have any significance to the entire dataset because there were only two populated values, and I could not figure out what the column was even for.

Code:

weight <- weight %>%

select(-Fat)

head(weight)

#Result:

# A tibble: 6 × 7

Id Date WeightKg WeightPounds BMI IsManualReport LogId

*<dbl>* *<chr>* *<dbl>* *<dbl>* *<dbl>* *<lgl>* *<dbl>*

1 1503960366 5/2/2016 11:59:59 PM 52.6 116. 22.6 TRUE 1.46e12

2 1503960366 5/3/2016 11:59:59 PM 52.6 116. 22.6 TRUE 1.46e12

3 1927972279 4/13/2016 1:08:52 AM 134. 294. 47.5 FALSE 1.46e12

4 2873212765 4/21/2016 11:59:59 PM 56.7 125. 21.5 TRUE 1.46e12

5 2873212765 5/12/2016 11:59:59 PM 57.3 126. 21.7 TRUE 1.46e12

6 4319703577 4/17/2016 11:59:59 PM 72.4 160. 27.5 TRUE 1.46e12

My next step in cleaning the data was to split date-time columns into two columns. See below the code for each data frame.

Code:

sleep <- sleep %>%

rename(Date = SleepDay) %>%

separate(Date,c("Date","Time"),sep=" ") %>%

mutate(Date = as.Date(Date, format = "%m/%d/%Y"))

heart\_rate <- heart\_rate %>%

rename(Date = Time) %>%

separate(Date,c("Date","Time"),sep=" ") %>%

mutate(Date = as.Date(Date, format = "%m/%d/%Y"))

mets <- mets %>%

rename(Date = ActivityMinute) %>%

separate(Date,c("Date","Time"),sep=" ") %>%

mutate(Date = as.Date(Date, format = "%m/%d/%Y"))

weight <- weight %>%

rename(Date = Date) %>%

separate(Date,c("Date","Time"),sep=" ") %>%

mutate(Date = as.Date(Date, format = "%m/%d/%Y"))

glimpse(sleep)

glimpse(heart\_rate)

glimpse(mets)

glimpse(weight)

#Result:

> glimpse(weight)

Rows: 67

Columns: 8

$ Id *<dbl>* 1503960366, 1503960366, 1927972279, 2873212765, 2873212765, 4319703577, 431~

$ Date *<date>* NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA~

$ Time *<chr>* NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~

$ WeightKg *<dbl>* 52.6, 52.6, 133.5, 56.7, 57.3, 72.4, 72.3, 69.7, 70.3, 69.9, 69.2, 69.1, 90~

$ WeightPounds *<dbl>* 115.9631, 115.9631, 294.3171, 125.0021, 126.3249, 159.6147, 159.3942, 153.6~

$ BMI *<dbl>* 22.65, 22.65, 47.54, 21.45, 21.69, 27.45, 27.38, 27.25, 27.46, 27.32, 27.04~

$ IsManualReport *<lgl>* TRUE, TRUE, FALSE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, FA~

$ LogId *<dbl>* 1.46223e+12, 1.46232e+12, 1.46051e+12, 1.46128e+12, 1.46310e+12, 1.46094e+1~

> glimpse(sleep)

Rows: 413

Columns: 6

$ Id *<dbl>* 1503960366, 1503960366, 1503960366, 1503960366, 1503960366, 1503960366,~

$ Date *<date>* 2016-04-12, 2016-04-13, 2016-04-15, 2016-04-16, 2016-04-17, 2016-04-19~

$ Time *<chr>* "12:00:00", "12:00:00", "12:00:00", "12:00:00", "12:00:00", "12:00:00",~

$ TotalSleepRecords *<dbl>* 1, 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~

$ TotalMinutesAsleep *<dbl>* 327, 384, 412, 340, 700, 304, 360, 325, 361, 430, 277, 245, 366, 341, 4~

$ TotalTimeInBed *<dbl>* 346, 407, 442, 367, 712, 320, 377, 364, 384, 449, 323, 274, 393, 354, 4~

> glimpse(heart\_rate)

Rows: 1,048,575

Columns: 4

$ Id *<dbl>* 2022484408, 2022484408, 2022484408, 2022484408, 2022484408, 2022484408, 2022484408, ~

$ Date *<date>* 2016-04-12, 2016-04-12, 2016-04-12, 2016-04-12, 2016-04-12, 2016-04-12, 2016-04-12,~

$ Time *<chr>* "7:21:00", "7:21:05", "7:21:10", "7:21:20", "7:21:25", "7:22:05", "7:22:10", "7:22:1~

$ Value *<dbl>* 97, 102, 105, 103, 101, 95, 91, 93, 94, 93, 92, 89, 83, 61, 60, 61, 61, 57, 54, 55, ~

> glimpse(mets)

Rows: 1,048,575

Columns: 4

$ Id *<dbl>* 1503960366, 1503960366, 1503960366, 1503960366, 1503960366, 1503960366, 1503960366, 1~

$ Date *<date>* 2016-04-12, 2016-04-12, 2016-04-12, 2016-04-12, 2016-04-12, 2016-04-12, 2016-04-12, ~

$ Time *<chr>* "12:00:00", "12:01:00", "12:02:00", "12:03:00", "12:04:00", "12:05:00", "12:06:00", "~

$ METs *<dbl>* 10, 10, 10, 10, 10, 12, 12, 12, 12, 12, 12, 12, 10, 10, 12, 10, 12, 10, 10, 10, 12, 1~

> glimpse(weight)

Rows: 67

Columns: 8

$ Id *<dbl>* 1503960366, 1503960366, 1927972279, 2873212765, 2873212765, 4319703577, 431~

$ Date *<date>* NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA~

$ Time *<chr>* NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~

$ WeightKg *<dbl>* 52.6, 52.6, 133.5, 56.7, 57.3, 72.4, 72.3, 69.7, 70.3, 69.9, 69.2, 69.1, 90~

$ WeightPounds *<dbl>* 115.9631, 115.9631, 294.3171, 125.0021, 126.3249, 159.6147, 159.3942, 153.6~

$ BMI *<dbl>* 22.65, 22.65, 47.54, 21.45, 21.69, 27.45, 27.38, 27.25, 27.46, 27.32, 27.04~

$ IsManualReport *<lgl>* TRUE, TRUE, FALSE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, FA~

$ LogId *<dbl>* 1.46223e+12, 1.46232e+12, 1.46051e+12, 1.46128e+12, 1.46310e+12, 1.46094e+1~

Then, I cleaned the names of the columns of each data frame:

Code:

activity<-clean\_names(activity)

sleep<-clean\_names(sleep)

heart\_rate<-clean\_names(heart\_rate)

mets<-clean\_names(mets)

weight<-clean\_names(weight)

|  |
| --- |
| #Result: |
|  |
| |  | | --- | |  | |

> clean\_names(activity)

# A tibble: 940 x 14

id activity\_date total\_steps tracker\_distance logged\_activities\_distance very\_active\_dis~

*<dbl>* *<chr>* *<dbl>* *<dbl>* *<dbl>* *<dbl>*

1 1503960366 4/12/2016 13162 8.5 0 1.88

2 1503960366 4/13/2016 10735 6.97 0 1.57

3 1503960366 4/14/2016 10460 6.74 0 2.44

4 1503960366 4/15/2016 9762 6.28 0 2.14

5 1503960366 4/16/2016 12669 8.16 0 2.71

6 1503960366 4/17/2016 9705 6.48 0 3.19

7 1503960366 4/18/2016 13019 8.59 0 3.25

8 1503960366 4/19/2016 15506 9.88 0 3.53

9 1503960366 4/20/2016 10544 6.68 0 1.96

10 1503960366 4/21/2016 9819 6.34 0 1.34

# ... with 930 more rows, and 8 more variables: moderately\_active\_distance <dbl>,

# light\_active\_distance <dbl>, sedentary\_active\_distance <dbl>, very\_active\_minutes <dbl>,

# fairly\_active\_minutes <dbl>, lightly\_active\_minutes <dbl>, sedentary\_minutes <dbl>,

# calories <dbl>

> clean\_names(sleep)

# A tibble: 413 x 6

id date time total\_sleep\_records total\_minutes\_asleep total\_time\_in\_bed

*<dbl>* *<date>* *<chr>* *<dbl>* *<dbl>* *<dbl>*

1 1503960366 2016-04-12 12:00:00 1 327 346

2 1503960366 2016-04-13 12:00:00 2 384 407

3 1503960366 2016-04-15 12:00:00 1 412 442

4 1503960366 2016-04-16 12:00:00 2 340 367

5 1503960366 2016-04-17 12:00:00 1 700 712

6 1503960366 2016-04-19 12:00:00 1 304 320

7 1503960366 2016-04-20 12:00:00 1 360 377

8 1503960366 2016-04-21 12:00:00 1 325 364

9 1503960366 2016-04-23 12:00:00 1 361 384

10 1503960366 2016-04-24 12:00:00 1 430 449

# ... with 403 more rows

> clean\_names(heart\_rate)

# A tibble: 1,048,575 x 4

id date time value

*<dbl>* *<date>* *<chr>* *<dbl>*

1 2022484408 2016-04-12 7:21:00 97

2 2022484408 2016-04-12 7:21:05 102

3 2022484408 2016-04-12 7:21:10 105

4 2022484408 2016-04-12 7:21:20 103

5 2022484408 2016-04-12 7:21:25 101

6 2022484408 2016-04-12 7:22:05 95

7 2022484408 2016-04-12 7:22:10 91

8 2022484408 2016-04-12 7:22:15 93

9 2022484408 2016-04-12 7:22:20 94

10 2022484408 2016-04-12 7:22:25 93

# ... with 1,048,565 more rows

> clean\_names(mets)

# A tibble: 1,048,575 x 4

id date time me\_ts

*<dbl>* *<date>* *<chr>* *<dbl>*

1 1503960366 2016-04-12 12:00:00 10

2 1503960366 2016-04-12 12:01:00 10

3 1503960366 2016-04-12 12:02:00 10

4 1503960366 2016-04-12 12:03:00 10

5 1503960366 2016-04-12 12:04:00 10

6 1503960366 2016-04-12 12:05:00 12

7 1503960366 2016-04-12 12:06:00 12

8 1503960366 2016-04-12 12:07:00 12

9 1503960366 2016-04-12 12:08:00 12

10 1503960366 2016-04-12 12:09:00 12

# ... with 1,048,565 more rows

> clean\_names(weight)

# A tibble: 67 x 8

id date time weight\_kg weight\_pounds bmi is\_manual\_report log\_id

*<dbl>* *<date>* *<chr>* *<dbl>* *<dbl>* *<dbl>* *<lgl>* *<dbl>*

1 1503960366 NA NA 52.6 116. 22.6 TRUE 1462230000000

2 1503960366 NA NA 52.6 116. 22.6 TRUE 1462320000000

3 1927972279 NA NA 134. 294. 47.5 FALSE 1460510000000

4 2873212765 NA NA 56.7 125. 21.5 TRUE 1461280000000

5 2873212765 NA NA 57.3 126. 21.7 TRUE 1463100000000

6 4319703577 NA NA 72.4 160. 27.5 TRUE 1460940000000

7 4319703577 NA NA 72.3 159. 27.4 TRUE 1462410000000

8 4558609924 NA NA 69.7 154. 27.2 TRUE 1461020000000

9 4558609924 NA NA 70.3 155. 27.5 TRUE 1461630000000

10 4558609924 NA NA 69.9 154. 27.3 TRUE 1462150000000

With all these steps done, I will combine all these data frames into one big data frame, joining them together by the user IDs and date (this is why I had to split date-time columns). This will make things much easier for me during the analysis phase.

Before I merged the data, I used the rename function to change the activity\_date column in the activity data frame to “date” to be consistent with the other data frames.

Note below that when I tried combining all the data frames together, I would get errors for how big the merge was (to the point where my device actually crashed). So I merged four data frames (activity, sleep, weight, and mets) because that’s as much as I could merge without getting an error. The heart\_rate data frame is the only individual data frame.

Also note that when I previewed the structure of the combined data frame, I had noticed that the column mets was “me\_ts”, so I had to use the rename function again.

Code:

activity <- activity %>%

rename(

date=activity\_date)

newdata <- merge(activity, sleep, by = c('id', 'date'), all = TRUE)

new2 <- merge(newdata, weight, by = c('id', 'date'), all = TRUE)

fitbit\_data <- merge(new2, mets, by = c('id', 'date'), all = TRUE)

str(fitbit\_data)

fitbit\_data <- fitbit\_data %>%

rename(

mets=me\_ts)

#Result:

> str(fitbit\_data)

'data.frame': 1052489 obs. of 26 variables:

$ id : num 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...

$ date : chr "16903" "16903" "16903" "16903" ...

$ total\_steps : num NA NA NA NA NA NA NA NA NA NA ...

$ tracker\_distance : num NA NA NA NA NA NA NA NA NA NA ...

$ logged\_activities\_distance: num NA NA NA NA NA NA NA NA NA NA ...

$ very\_active\_distance : num NA NA NA NA NA NA NA NA NA NA ...

$ moderately\_active\_distance: num NA NA NA NA NA NA NA NA NA NA ...

$ light\_active\_distance : num NA NA NA NA NA NA NA NA NA NA ...

$ sedentary\_active\_distance : num NA NA NA NA NA NA NA NA NA NA ...

$ very\_active\_minutes : num NA NA NA NA NA NA NA NA NA NA ...

$ fairly\_active\_minutes : num NA NA NA NA NA NA NA NA NA NA ...

$ lightly\_active\_minutes : num NA NA NA NA NA NA NA NA NA NA ...

$ sedentary\_minutes : num NA NA NA NA NA NA NA NA NA NA ...

$ calories : num NA NA NA NA NA NA NA NA NA NA ...

$ time.x : chr "12:00:00" "12:00:00" "12:00:00" "12:00:00" ...

$ total\_sleep\_records : num 1 1 1 1 1 1 1 1 1 1 ...

$ total\_minutes\_asleep : num 327 327 327 327 327 327 327 327 327 327 ...

$ total\_time\_in\_bed : num 346 346 346 346 346 346 346 346 346 346 ...

$ time.y : chr NA NA NA NA ...

$ weight\_kg : num NA NA NA NA NA NA NA NA NA NA ...

$ weight\_pounds : num NA NA NA NA NA NA NA NA NA NA ...

$ bmi : num NA NA NA NA NA NA NA NA NA NA ...

$ is\_manual\_report : logi NA NA NA NA NA NA ...

$ log\_id : num NA NA NA NA NA NA NA NA NA NA ...

$ time : chr "2:57:00" "3:01:00" "8:57:00" "11:44:00" ...

$ mets : num 36 10 12 12 28 10 12 26 12 12 ...

1. **ANALYZE**

**Intro**

For my analysis I want to look at the data at a glance (through high-level summary statistics), then also do a deep dive to examine the relationships within the data. My goal is to understand at a high-level how healthy the participants are, comparing the insights to real health recommendations (to understand the target audience), and to also understand how the device is being used to then make recommendations to inform the overall marketing strategy of Bellabeat.

Now that we have a consolidated data frame, plus the heart rate data frame, let’s first see how many distinct users there are per data frame.

Code & Results:

> aid <- unique(activity$id)

> wid <- unique(weight$id)

> mid <- unique(mets$id)

> sid <- unique(sleep$id)

> hid <- unique(heart\_rate$id)

>

> length(aid)

[1] 33

> length(wid)

[1] 8

> length(mid)

[1] 27

> length(sid)

[1] 24

> length(hid)

[1] 7

We see that there are different values for each data frame, when we were told that there were 30 participants during this study. This tells us the data is not totally synced up – there are matching IDs across the data frames, but there are far more IDs that aren’t matched. Because of this, my analysis will be per health topic.

**Summary Statistics**

Heart Rate

Since the heart\_rate data frame is on its own, let’s analyze it first. The average (and also the healthy) range for heart beats per minute is 60-100 BPM (Laskowski, 2020), so lets see the average heart rate and the amount of values that fall into that range.

Code & Result:

> mean(heart\_rate$value)

[1] 77.0247

>

> sum(heart\_rate$value>=60 & heart\_rate$value<=100)

[1] 815815

> sum(heart\_rate$value>=60 & heart\_rate$value<=100)/sum(heart\_rate$value) \*100

[1] 1.010095

> heart\_rate %>%

+ select(value) %>%

+ summary()

value

Min. : 38.00

1st Qu.: 64.00

Median : 75.00

Mean : 77.02

3rd Qu.: 87.00

Max. :203.00

So the average heart rate falls into the range which means out of the several values per user, we can say the average BPM is healthy. Then I counted every single value that fell into the healthy range, and found out that only 815,815 values fell into the range (about 1% of all total values).

We can’t say this is concerning because unfortunately, we don’t know what any of the users were doing when their heart rate was below or above the healthy bpm range, since the times across all data frames aren’t synced up (they could have been running a marathon when their bpm was elevated). Also, when looking at the summary statistics of the value column, the first and third quartile have values within the healthy bpm range.

Therefore we can rely on the healthy average heart rate value of 77 BPM (rounded), so we can say that Fitbit users have a healthy average BPM.

#Fitbit\_data dataframe summary:

> fitbit\_data %>%

+ select(total\_steps,tracker\_distance,logged\_activities\_distance,very\_active\_distance,moderately\_active\_distance,light\_active\_distance,sedentary\_active\_distance,very\_active\_minutes,fairly\_active\_minutes,lightly\_active\_minutes,sedentary\_active\_distance,very\_active\_minutes,fairly\_active\_minutes,lightly\_active\_minutes,sedentary\_minutes,calories,total\_sleep\_records,total\_minutes\_asleep,total\_time\_in\_bed,weight\_kg,weight\_pounds,bmi, mets) %>%

+ summary()

total\_steps tracker\_distance logged\_activities\_distance very\_active\_distance

Min. : 0 Min. : 0.0 Min. :0.0 Min. : 0.0

1st Qu.: 3790 1st Qu.: 2.6 1st Qu.:0.0 1st Qu.: 0.0

Median : 7406 Median : 5.2 Median :0.0 Median : 0.2

Mean : 7638 Mean : 5.5 Mean :0.1 Mean : 1.5

3rd Qu.:10727 3rd Qu.: 7.7 3rd Qu.:0.0 3rd Qu.: 2.1

Max. :36019 Max. :28.0 Max. :4.9 Max. :21.9

NA's :1051549 NA's :1051549 NA's :1051549 NA's :1051549

moderately\_active\_distance light\_active\_distance sedentary\_active\_distance

Min. :0.0 Min. : 0.0 Min. :0.0

1st Qu.:0.0 1st Qu.: 1.9 1st Qu.:0.0

Median :0.2 Median : 3.4 Median :0.0

Mean :0.6 Mean : 3.3 Mean :0.0

3rd Qu.:0.8 3rd Qu.: 4.8 3rd Qu.:0.0

Max. :6.5 Max. :10.7 Max. :0.1

NA's :1051549 NA's :1051549 NA's :1051549

very\_active\_minutes fairly\_active\_minutes lightly\_active\_minutes sedentary\_minutes

Min. : 0.0 Min. : 0.0 Min. : 0.0 Min. : 0.0

1st Qu.: 0.0 1st Qu.: 0.0 1st Qu.:127.0 1st Qu.: 729.8

Median : 4.0 Median : 6.0 Median :199.0 Median :1057.5

Mean : 21.2 Mean : 13.6 Mean :192.8 Mean : 991.2

3rd Qu.: 32.0 3rd Qu.: 19.0 3rd Qu.:264.0 3rd Qu.:1229.5

Max. :210.0 Max. :143.0 Max. :518.0 Max. :1440.0

NA's :1051549 NA's :1051549 NA's :1051549 NA's :1051549

calories total\_sleep\_records total\_minutes\_asleep total\_time\_in\_bed weight\_kg

Min. : 0 Min. :1.0 Min. : 58.0 Min. : 61.0 Min. : 52.6

1st Qu.:1828 1st Qu.:1.0 1st Qu.:357.0 1st Qu.:402.0 1st Qu.: 61.2

Median :2134 Median :1.0 Median :431.0 Median :464.0 Median : 61.5

Mean :2304 Mean :1.1 Mean :415.6 Mean :457.5 Mean : 64.8

3rd Qu.:2793 3rd Qu.:1.0 3rd Qu.:492.0 3rd Qu.:530.0 3rd Qu.: 62.4

Max. :4900 Max. :3.0 Max. :796.0 Max. :961.0 Max. :133.5

NA's :1051549 NA's :565879 NA's :565879 NA's :565879 NA's :991745

weight\_pounds bmi mets

Min. :116.0 Min. :21.5 Min. : 0.00

1st Qu.:134.9 1st Qu.:23.9 1st Qu.: 10.00

Median :135.6 Median :24.0 Median : 10.00

Mean :142.9 Mean :25.1 Mean : 14.48

3rd Qu.:137.6 3rd Qu.:24.4 3rd Qu.: 11.00

Max. :294.3 Max. :47.5 Max. :157.00

NA's :991745 NA's :991745 NA's :1034

Looking at the summary statistics for our consolidated Fitbit\_data data frame, the average steps taken and the distance travelled taken out of all the users and the days that they were tracked for was 7638 steps and 5.5 miles respectively. The average steps is close to the healthy recommended amount of daily steps of 10,000 (Bubnis n.d.), while the average distance travelled is slightly above the recommended amount of miles for healthy adults (Bubnis n.d.).

When glancing at the activity of the users in minutes, you can see that they are getting at least 20 minutes (on average) of intense activity, which is good taking into account the fairly active and lightly active average minutes, because the recommended amount of moderate (fair) activity is 30 minutes (Laskowski, 2021).

For sleep, the recommended amount of sleep is 7-9 hours for adults (so at least 420 minutes) (Nazario, 2020), and according to the mean amount of sleep measured in minutes, participants on average had 416 minutes (rounded) of sleep on average, which is a decent amount of sleep.

For weight, we can’t make any inferences because we don’t know the heights of the participants, plus the data was manually logged. But looking at the BMI (a helpful measure for us to indicate if participant is at a healthy weight or not), a mean value of 25 indicated that on average the participants are overweight (25.0-29.9 is the BMI range for being overweight) (CDC, 2020).

Unfortunately for the METs data, there is no use in analyzing this past the summary statistics, because it doesn’t align with the rest of the data, and there is not much else to analyze past the MET values given. The WHO recommends that people get “600 MET minutes of physical activity - the equivalent of 150 minutes each week of brisk walking or 75 minutes per week of running” (Pickover, 2016). With a mean MET minute value of 14.5 (rounded), we can assume that on average, people are getting enough activity per week, since resting/sitting still is equal to one MET (Roland, 2019).

**Deep Dive**

Steps and Distance vs Calories

Code:

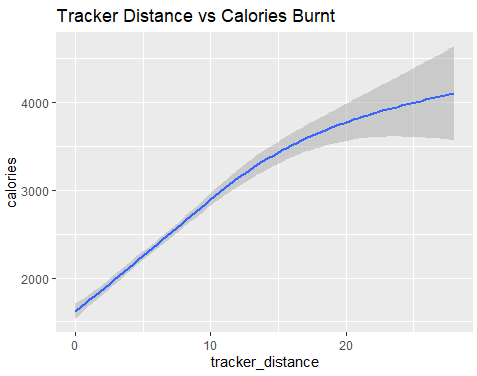
#steps and distance travelled vs calories burnt

ggplot(data=fitbit\_data, aes(x=total\_steps, y=calories)) + geom\_smooth() + labs(title="Total Steps vs Calories Burnt")

ggplot(data=fitbit\_data, aes(x=tracker\_distance, y=calories)) + geom\_smooth() + labs(title="Tracker Distance vs Calories Burnt")

Results:





Here we see that for both the total steps taken and total distance tracked, there is a positive relationship with the amount of calories burnt, meaning the more that you walk/run (or the more distance you cover on foot), you will burn more calories.

Activity Intensity vs Calories

Code:

colors <- c("sedentary\_minutes" = "blue", "lightly\_active\_minutes" = "red", "fairly\_active\_minutes" = "orange", "very\_active\_minutes"="black")

>

> plot <- ggplot(fitbit\_data, aes( y=calories))+

+ geom\_point(aes(x=sedentary\_minutes, color="sedentary\_minutes"))+

+ geom\_point(aes(x=lightly\_active\_minutes, color="lightly\_active\_minutes"))+

+ geom\_point(aes(x=fairly\_active\_minutes, color="fairly\_active\_minutes"))+

+ geom\_point(aes(x=very\_active\_minutes, color="very\_active\_minutes"))+

+ labs(title="Activity Intensities vs Calories", y="Calories", x = "Minutes")

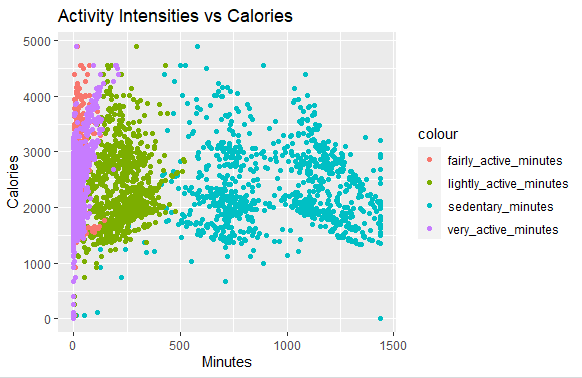
>

>

>

> plot

Result:



Since there are several users who spend time doing every type of activity throughout the day, we can’t truly pinpoint what type of activity is burning the most calories, but we can see that people spent less time being very active, yet lots of calories were burned.

Sleep

Let’s look at how much sleep all participants across each tracked day have wasted all together.

Code & Result:

> wasted\_sleep <- sum(fitbit\_data$total\_time\_in\_bed,na.rm=T)-sum(fitbit\_data$total\_minutes\_asleep,na.rm=T)

>

> wasted\_sleep

[1] 20392186

This was calculated by summing up both the time in bed column and the total time sleeping column, and subtracting the latter from the time in bed column. I then made a new column in the Fitbit data frame that shows in minutes how much sleep was wasted. This was done by calculating the difference between the total time in bed and total sleep columns.

Code:

fitbit\_data$sleep\_wasted <- fitbit\_data$total\_time\_in\_bed - fitbit\_data$total\_minutes\_asleep

I then made a smaller data frame out of all the sleep related columns from the Fitbit data frame. From there I calculated the average sleep wasted

Code & Result

> df <- fitbit\_data[, c(17,18,27)]

> head(df)

total\_minutes\_asleep total\_time\_in\_bed sleep\_wasted

1 327 346 19

2 327 346 19

3 327 346 19

4 327 346 19

5 327 346 19

6 327 346 19

> mean(df$sleep\_wasted, na.rm=T)

[1] 41.90663

Participants on average lost 42 minutes (rounded) of sleep throughout the study. See below a bar chart that depicts the comparison between time slept versus time in bed.

Code:

sum(fitbit\_data$total\_time\_in\_bed,na.rm=T)

sum(fitbit\_data$total\_minutes\_asleep,na.rm=T)

sum(fitbit\_data$sleep\_wasted,na.rm=T)

sleep\_variables <- c('Sum\_in\_bed','Sum\_asleep','total\_difference')

minutes <- c(222603540, 202211354, 20392186)

df2 <- data.frame(sleep\_variables, minutes)

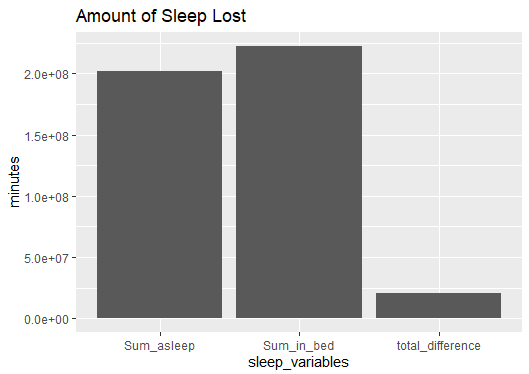
View(df2)

w <- ggplot(data = df2, aes(x=sleep\_variables, y=minutes))+

geom\_bar(stat='identity')+

labs(title="Amount of Sleep Lost")

w



Weight

As stated during the analysis of the summary statistics, the only variable that is actually helpful to us are the BMI values. Here I made a histogram out of the BMI column.

Code:

bmi <- (fitbit\_data$bmi)

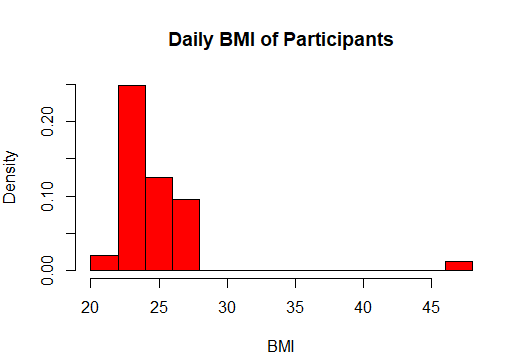
hist(bmi,

main="Daily BMI of Participants",

xlab="BMI",

col="red",

freq=FALSE)



There are six bins. 20-24.9 is considered a healthy BMI, while anything 25+ is overweight. Most of the participants are in the 20-24.9 range, but there is a good chunk of the participants who’s BMI fit in the 25-29.9 (overweight), while there is just the one outlier who is in the 45+ range (see below) which is classified as obese. The participants who were in the overweight and obese bins must have been the ones who brought up the BMI average that we saw in summary statistics earlier.

Code & Result:

> max(bmi,na.rm=T)

[1] 47.54

1. **SHARE AND ACT**

For this stage of the case study, I will be presenting my key findings and my visualizations to then make my high level recommendations for the marketing strategy

**Bellabeat Data Analysis Case Study**

My analysis was done in two separate parts. I did an analysis of some of the basic summary statistics for the first part, and for the second part I dove a bit deeper to analyze the relationships within the data. For my presentation, I will be breaking down my analysis by each health topic: heart rate, activity, sleep, weight, and mets, and I will then provide my recommendations.

**Key Findings**

Heart Rate

From some basic R functions, I found that out of all the recorded BPM values in the heart rate data frame, only 1% of them fell between 60-100 BPM ranges (healthy range). But, since we can’t know the cause of why one’s heartbeat was so low/high because the data was not synced, we can’t assume that only 1% of the participants had a healthy heart rate. The average heart rate amongst the participants is 77 BPM which falls in the healthy BPM range. From this value, we can assume that most of the participants on average have a healthy BPM.

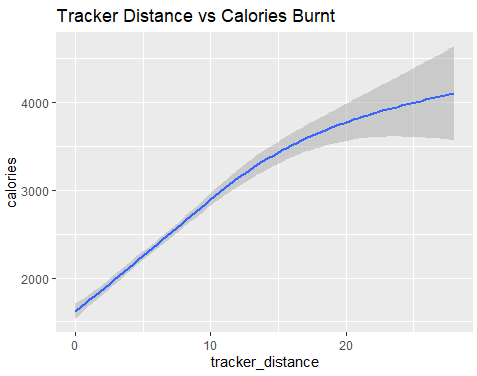
Activity

On average, the participants intensely exercised for 20 minutes a day, and moderately (listed as “fairly” in the data frame) for 13 minutes a day. This means that participants were getting at least 30 minutes of exercise a day on average, which is great because the recommended amount of activity one should partake in a day is 30 minutes.

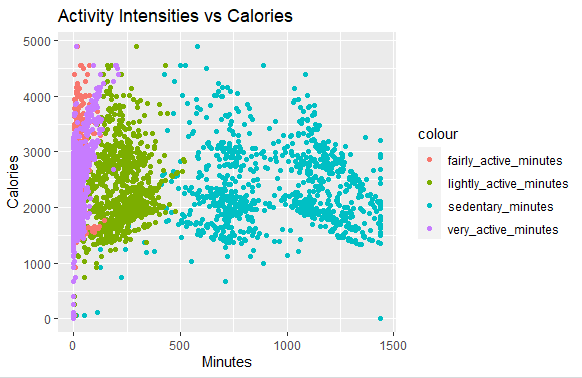
On average, users took 7638 steps a day, and travelled 5.5 miles a day on foot. These numbers indicate that users are getting close to their recommended amount of steps a day (10,000), and slightly above the recommended distance to travel in a day.

I found that there is a positive relationship between the amount of steps and the distance you travel in a day versus the amount of calories that you burn in that the more that you do, the more calories that you burn. See below for two regression line visualizations that illustrate this point:





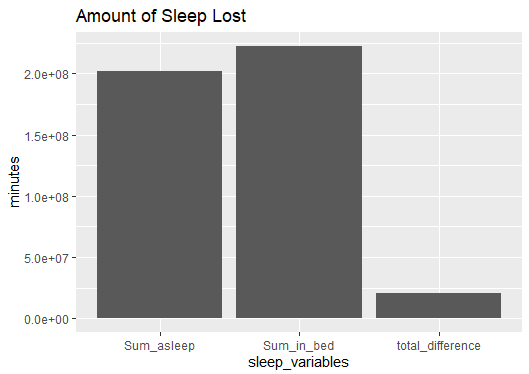
In addition, I also explored the relationship between the intensity of the activities in minutes versus the amount of calories burned. Since most of the users spent time doing every type of activity throughout the day, I found there was overlap with the data. For example, if one user spent a little bit of their time doing very intense exercises and burned most of their calories that way, but was mostly inactive throughout the day (which was usually the case), it will look like the sedentary activity is what has users burning calories. See the scatter plot below:



As you can see from the plot there is quite a bit of overlap. But also, participants spent a lot less time being very active and fairly active, yet you can see for those points that they burn a lot of calories.

Sleep

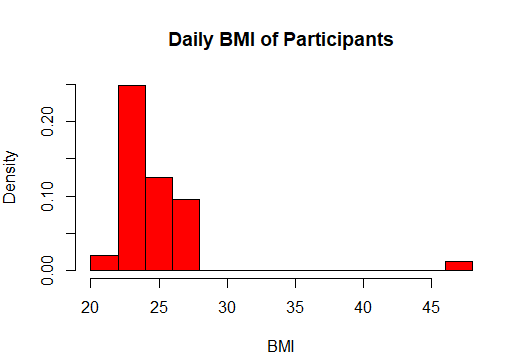
On average, the participants got about 416 minutes of sleep a night during the study, which is about the recommended amount of sleep that an adult should get. Since the Fitbit tracks how long users were in bed for and how long they were asleep for, I wanted to know how much sleep the participants lost. Altogether, the participants lost 20,392,186 minutes of sleep, and lost an average amount of 42 minutes of sleep per night. This isn’t necessarily bad, but highlighting the amount of sleep that participants lose could be an opportunity to expound upon. See below for a bar chart that highlights the slight difference between total sleep and total time in bed:



Here, I’ve summed up the amount of sleep every user got across every day, and I did the same for the time spent in bed. I’ve also added a bar to visually quantify the actual total difference between time in bed and time sleeping. As you can see, the difference is not too big.

Weight

During my analysis of the weight data, I found that the actual weight of the participants wasn’t exactly helpful because we don’t know the height of the participants. Luckily, there are BMI values listed day by day for each participant. I made a histogram chart that separates people based on the range that their BMI fits in:



When I looked at the summary statistics, I found the average BMI for the participants was 25, which means the average participant was overweight (20-24.9 is healthy, 25-29.9 is overweight, and 30+ is obese). But when looking at this histogram, we see that plenty of participants actually fit in the healthy BMI range. The reason why the average BMI is so high is because there is a substantial amount of participants who fit in the overweight range, and one participant who had a much higher BMI compared to the others. We can say that after analyzing this data that there is a mix of healthy and overweight participants, and that one group does not drastically exceed the other.

METs

Unfortunately, I don’t have any significant insights after analyzing this data that wasn’t already found when looking at the other data. When I initially reviewed the spreadsheet for this data, I thought it would be more useful. Finding the average MET minutes of 16 did confirm what I already found with the activity data - that users are active.

**Recommendations to Bellabeat Marketing Strategy**

Heart Rate

Having a feature that measures your heart beat every minute is great, but I believe Bellabeat should use their app to provide users their own key insights about their heart data. It would be a useful feature for users to know their average heart beat, and for the app to indicate to them if their average BPM is healthy or not. In addition, having a feature that notifies the user when their heart rate drops too low or too high would also be a useful for the app.

Activity

It is nice that Bellabeat products already track activity, but Bellabeat should provide their users’ detailed insights about their own activity data to their users. For example, they can show how many minutes on average one is active a day, and how many minutes on average one spends doing activities of varying activities (tracking the average distance one travels in a day would also be useful alongside the average time).

Bellabeat should also incorporate a feature in their app that reminds its users to be more active when they fall below their own averages for too many consecutive days in a row, or have been too sedentary. This would especially be useful if the app had a feature where one could input their activity and/or weight loss goals so that when a user is falling behind their goals, the app notifies them. To motivate its users further, the app should have a feature where you can add friends and compete with them on a daily, weekly, monthly, or annual basis. They can compete on things like average and total steps, average and total distance travelled, average and total calories burnt, and etc.

The app should also show users data through graphs showing the relationship between their activity and the amount of calories they’ve burned, like the two I’ve provided. A regression line or a line graph would quickly help a user understand the results of their activity, and seeing a visual representation of their work paying off may motivate them more.

Sleep

Bellabeat’s tracking devices do a good job of tracking the sleep of the users, but tracking the amount of sleep a user wastes can be helpful to the user. Having the devices track how long a user goes to bed for, and how much sleep they actually get, to determine how much sleep a user misses out on is an opportunity for Bellabeat to incorporate sleeping tips in their app. This is an opportunity for Bellabeat to either research and curate their own methods of helping their users to get the best sleep that they can or they can partner with a sleep wellness application to help them with that. In addition to this idea, the app can show them how much more sleep they get when they use the advice given to them.

The Target Audience

After analyzing the heart rate, activity, and weight data, I can conclude that there is a mix of participants who are already healthy and those who are working to be healthier. Through the heart rate data, I found that people for the most part have a healthy BPM, and through the weight data I found that there are as many people who have a healthy BMI versus those who are overweight. In addition, through the activity data, I found that users spend a healthy amount of time being active.

So when tying it altogether, we can say that users of the Fitbit are either healthy people who are maintaining their health, or overweight individuals who are trying to lose weight and improve their health by being active. With that in mind, Bellabeat’s primary targets should be women who love to stay fit and maintain their healthy bodies, and women who are overweight and would like to lose weight to become the healthiest version of themselves.

**Sources**

* Edward R. Laskowski, E. (2020, October 2). *2 easy, accurate ways to measure your heart rate*. Mayo Clinic. Retrieved October 14, 2021, from https://www.mayoclinic.org/healthy-lifestyle/fitness/expert-answers/heart-rate/faq-20057979.
* Bubnis, D. (n.d.). *How many steps should you take a day for fitness or weight loss?* Medical News Today. Retrieved October 15, 2021, from https://www.medicalnewstoday.com/articles/how-many-steps-should-you-take-a-day.
* Edward Laskowski, E. (2021, September 22). *How much exercise do you really need?* Mayo Clinic. Retrieved October 15, 2021, from https://www.mayoclinic.org/healthy-lifestyle/fitness/expert-answers/exercise/faq-20057916.
* Nazario, B. (2020, August 24). *How much sleep do I need?* WebMD. Retrieved October 15, 2021, from https://www.webmd.com/sleep-disorders/sleep-requirements.
* Centers for Disease Control and Prevention. (2020, September 17). *Adult BMI calculator*. Centers for Disease Control and Prevention. Retrieved October 15, 2021, from https://www.cdc.gov/healthyweight/assessing/bmi/adult\_bmi/english\_bmi\_calculator/bmi\_calculator.html.
* Pickover, E. (2016, August 10). *This is the amount of exercise you should be doing every day*. The Independent. Retrieved October 16, 2021, from https://www.independent.co.uk/life-style/health-and-families/health-news/exercise-health-disease-amount-you-should-be-doing-every-day-a7181261.html.
* Roland, J. (2019, October 21). *What are Mets, and how are they calculated?* Healthline. Retrieved October 16, 2021, from https://www.healthline.com/health/what-are-mets.