Case Study

Daniel Anderson

2023-07-26

Introduction

Thank you for joining me for my Bellabeat Data Analysis Capstone Case Study. I have recently been progressing through the Google Data Analytics Certificate and as part of my course I created a Case Study in order to answer some business questions for the stakeholders at Bellabeat. I will be using the standard steps of Data Analysis: Ask, Prepare, Process, Analyze, Share, and Act.

About Bellabeat

Bellabeat is a high-tech manufacturer of health-focused products for women. Founded by Urška Sršen and Sando Mur in 2013 Bellabeat as grown rapidly and quickly positioned itself as a tech-driven wellness company for women by focusing on beautifully designed smart products for women's health. By leveraging Sršen's artistic background and experience the company combines technology and aesthetics to inform and empower women worldwide. With strong emphasis on activity, sleep, stress and reproductive health data Bellabeat offers insights that enable women to better understand their personal well being.

By 2016 Bellabeat had expanded across the globe, launching multiple breakthrough products available on the global stage through online retailers and their own E-commerce channel. Sršen's continued strategic vision involving analyzing smart device usage to uncover usage trends will continue to inform Bellabeat's marketing strategies.

About this study

In this study I have been tasked with analyzing smart device usage data in order to gain insight into how consumers use Non-bellabeat smart devices. I will then use these insights to apply context to, and help shape, the marketing strategy for Bellabeat.

Questions for analysis (Ask Phase)

In this phase of the Case Study process I aimed to get a better grasp on the data and the issues that I am trying to research. I started by asking myself the following questions and exploring their answers. -What are some greater trends in the device usage? -How can these trends be used to influence change in the Bellabeat company? -How do these trends relate to the potential data from Bellabeat consumers? -How does this data allow us to target potential customers?

Business Task

After forming questions and reviewing available business material I can clearly define my prescribed task: To analyze how Non-Bellabeat customers use their smart devices and how that data could be used to influence and increase Bellabeat's market share in the Fitness Smart-device arena.

Prepare Phase

In this phase of the Case Study I will download and import all relevant data. I will also make sure all data is organized and that there are no errors or outliers that would unduly affect the results of my study.

Data Sources

Bellabeat has encouraged me to use public data that explores smart device users daily habits. I was pointed to a specific data set, Fit Bit Fitness Tracker Data(CC0: Public Domain, available through Mobius). This Data set contains the personal fitness tracking information from 30 fitbit users. All eligible Fit bit users consented to the submission and subsequent retrieval of personal tracking data including minute-level output for activity, heart rate, and sleep monitoring. Also included is daily activity, steps, and heart rate that can be used to explore a user's daily habits.

-Fit Bit Fitness Tracker Data: https://www.kaggle.com/datasets/arashnic/fitbit A little about this Data set: Distributed survey via Amazon Mechanical Turk between 03.12.2016-05.12.2016. Variation between output represents use of different types of Fit bit trackers and individual tracking behaviors / preferences.

Setting up my workspace

To prepare my Rstudio I will install and initialize various packages that will enable me to forward my analysis into Bellabeat.

I will be using the following packages for Data Cleaning and manipulation, ensuring the data is relevant and correct. I have not included the messages and warnings that are distributed after each step for clarity and ease of reading.

```r
install.packages("tidyverse")
install.packages("lubridate")
install.packages("dplyr")
install.packages("tidyr")
install.packages("stringr")
install.packages("skimr")
install.packages("here")
install.packages("ggplot2")
setwd("/cloud/project")
```

After having installed all relevant packages it is time to load them into my work space.

```r
library(tidyverse)
library(lubridate)
library(dplyr)
library(ggplot2)
library(tidyr)
library(skimr)
library(janitor)
library(here)
```

Data Sources

Now that all of our packages our loaded, it is time to import and review our data set to ensure it is applicable and that there are no problems. Again, our data is from: "https://www.kaggle.com/datasets/arashnic/fitbit"

```r
dailyActivity <- read.csv("dailyActivity_merged.csv")
calories <- read.csv("dailyCalories_merged.csv")
intensities <- read.csv("dailyIntensities_merged.csv")
heartrate <- read.csv("heartrate_seconds_merged.csv")
```

```
sleep <- read.csv("sleepDay_merged.csv")
weight <- read.csv("weightLogInfo_merged.csv")
steps <- read.csv("dailySteps_merged.csv")
```

We have imported 7 Data sets to do our analysis. To review our data and check for inconsistencies or errors we will move on to the next phase.

Process Phase

First we will review our data to ensure it was imported correctly and all of the datatypes are correct. To do this we have many options, but I will use

Quick Look

```
head(dailyActivity)
```

```
##           Id ActivityDate TotalSteps TotalDistance TrackerDistance
## 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
##   LoggedActivitiesDistance VeryActiveDistance ModeratelyActiveDistance
## 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
##   LightActiveDistance SedentaryActiveDistance VeryActiveMinutes
## 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
##   FairlyActiveMinutes LightlyActiveMinutes SedentaryMinutes 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
```

```
glimpse(dailyActivity)
```

```
## 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~
```

```
head(calories)
```

```
##           Id ActivityDay Calories
## 1 1503960366   4/12/2016     1985
## 2 1503960366   4/13/2016     1797
## 3 1503960366   4/14/2016     1776
## 4 1503960366   4/15/2016     1745
## 5 1503960366   4/16/2016     1863
## 6 1503960366   4/17/2016     1728
```

```
glimpse(calories)
```

```
## Rows: 940
## Columns: 3
## $ Id          <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 1503960366~
## $ ActivityDay <chr> "4/12/2016", "4/13/2016", "4/14/2016", "4/15/2016", "4/16/~
## $ Calories    <int> 1985, 1797, 1776, 1745, 1863, 1728, 1921, 2035, 1786, 1775~
```

```
head(intensities)
```

```
##           Id ActivityDay SedentaryMinutes LightlyActiveMinutes
## 1 1503960366   4/12/2016              728                  328
## 2 1503960366   4/13/2016              776                  217
## 3 1503960366   4/14/2016             1218                  181
## 4 1503960366   4/15/2016              726                  209
## 5 1503960366   4/16/2016              773                  221
## 6 1503960366   4/17/2016              539                  164
##   FairlyActiveMinutes VeryActiveMinutes SedentaryActiveDistance
## 1                  13                25                       0
## 2                  19                21                       0
## 3                  11                30                       0
## 4                  34                29                       0
## 5                  10                36                       0
## 6                  20                38                       0
##   LightActiveDistance ModeratelyActiveDistance VeryActiveDistance
## 1                6.06                     0.55               1.88
## 2                4.71                     0.69               1.57
## 3                3.91                     0.40               2.44
## 4                2.83                     1.26               2.14
## 5                5.04                     0.41               2.71
## 6                2.51                     0.78               3.19
```

```
glimpse(intensities)
```

```
## Rows: 940
## Columns: 10
## $ Id                       <dbl> 1503960366, 1503960366, 1503960366, 150396036~
## $ ActivityDay              <chr> "4/12/2016", "4/13/2016", "4/14/2016", "4/15/~
## $ SedentaryMinutes         <int> 728, 776, 1218, 726, 773, 539, 1149, 775, 818~
## $ LightlyActiveMinutes     <int> 328, 217, 181, 209, 221, 164, 233, 264, 205, ~
## $ FairlyActiveMinutes      <int> 13, 19, 11, 34, 10, 20, 16, 31, 12, 8, 27, 21~
## $ VeryActiveMinutes        <int> 25, 21, 30, 29, 36, 38, 42, 50, 28, 19, 66, 4~
## $ SedentaryActiveDistance  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ LightActiveDistance      <dbl> 6.06, 4.71, 3.91, 2.83, 5.04, 2.51, 4.71, 5.0~
## $ ModeratelyActiveDistance <dbl> 0.55, 0.69, 0.40, 1.26, 0.41, 0.78, 0.64, 1.3~
## $ VeryActiveDistance       <dbl> 1.88, 1.57, 2.44, 2.14, 2.71, 3.19, 3.25, 3.5~
```

```r
head(heartrate)
```

```
##           Id                 Time Value
## 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
```

```r
glimpse(heartrate)
```

```
## Rows: 2,483,658
## Columns: 3
## $ Id    <dbl> 2022484408, 2022484408, 2022484408, 2022484408, 2022484408, 2022~
## $ Time  <chr> "4/12/2016 7:21:00 AM", "4/12/2016 7:21:05 AM", "4/12/2016 7:21:~
## $ Value <int> 97, 102, 105, 103, 101, 95, 91, 93, 94, 93, 92, 89, 83, 61, 60, ~
```

```r
head(sleep)
```

```
##           Id              SleepDay TotalSleepRecords TotalMinutesAsleep
## 1 1503960366 4/12/2016 12:00:00 AM                 1                327
## 2 1503960366 4/13/2016 12:00:00 AM                 2                384
## 3 1503960366 4/15/2016 12:00:00 AM                 1                412
## 4 1503960366 4/16/2016 12:00:00 AM                 2                340
## 5 1503960366 4/17/2016 12:00:00 AM                 1                700
## 6 1503960366 4/19/2016 12:00:00 AM                 1                304
##   TotalTimeInBed
## 1            346
## 2            407
## 3            442
## 4            367
## 5            712
## 6            320
```

```r
glimpse(sleep)
```

```
## Rows: 413
## Columns: 5
## $ Id                 <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 150~
## $ SleepDay           <chr> "4/12/2016 12:00:00 AM", "4/13/2016 12:00:00 AM", "~
## $ TotalSleepRecords  <int> 1, 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ TotalMinutesAsleep <int> 327, 384, 412, 340, 700, 304, 360, 325, 361, 430, 2~
## $ TotalTimeInBed     <int> 346, 407, 442, 367, 712, 320, 377, 364, 384, 449, 3~
```

```r
head(weight)
```

```
##           Id                 Date WeightKg WeightPounds Fat   BMI
## 1 1503960366  5/2/2016 11:59:59 PM     52.6     115.9631  22 22.65
## 2 1503960366  5/3/2016 11:59:59 PM     52.6     115.9631  NA 22.65
## 3 1927972279  4/13/2016 1:08:52 AM    133.5     294.3171  NA 47.54
## 4 2873212765 4/21/2016 11:59:59 PM     56.7     125.0021  NA 21.45
## 5 2873212765 5/12/2016 11:59:59 PM     57.3     126.3249  NA 21.69
## 6 4319703577 4/17/2016 11:59:59 PM     72.4     159.6147  25 27.45
##   IsManualReport        LogId
## 1           True 1.462234e+12
## 2           True 1.462320e+12
## 3          False 1.460510e+12
## 4           True 1.461283e+12
## 5           True 1.463098e+12
## 6           True 1.460938e+12
```

```r
glimpse(weight)
```

```
## Rows: 67
## Columns: 8
## $ Id             <dbl> 1503960366, 1503960366, 1927972279, 2873212765, 2873212~
## $ Date           <chr> "5/2/2016 11:59:59 PM", "5/3/2016 11:59:59 PM", "4/13/2~
## $ WeightKg       <dbl> 52.6, 52.6, 133.5, 56.7, 57.3, 72.4, 72.3, 69.7, 70.3, ~
## $ WeightPounds   <dbl> 115.9631, 115.9631, 294.3171, 125.0021, 126.3249, 159.6~
## $ Fat            <int> 22, NA, NA, NA, NA, 25, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ BMI            <dbl> 22.65, 22.65, 47.54, 21.45, 21.69, 27.45, 27.38, 27.25,~
## $ IsManualReport <chr> "True", "True", "False", "True", "True", "True", "True"~
## $ LogId          <dbl> 1.462234e+12, 1.462320e+12, 1.460510e+12, 1.461283e+12,~
```

```r
head(steps)
```

```
##           Id ActivityDay StepTotal
## 1 1503960366   4/12/2016     13162
## 2 1503960366   4/13/2016     10735
## 3 1503960366   4/14/2016     10460
## 4 1503960366   4/15/2016      9762
## 5 1503960366   4/16/2016     12669
## 6 1503960366   4/17/2016      9705
```

```r
glimpse(steps)
```

```
## Rows: 940
## Columns: 3
## $ Id          <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 1503960366~
## $ ActivityDay <chr> "4/12/2016", "4/13/2016", "4/14/2016", "4/15/2016", "4/16/~
## $ StepTotal   <int> 13162, 10735, 10460, 9762, 12669, 9705, 13019, 15506, 1054~
```

Cleaning Time

In this phase I will be putting the data sets through basic cleaning procedures. The following steps were taken to Process, Clean, and Organize the data to make analysis in the next step that much more accurate and relevant to our purposes.

After reviewing each data frame with head() and glimpse() we can progress into cleaning our data to make it more relevant and applicable for our purposes. For all of our data frames I will be using both is.na() and duplicated() to determine if we have any missing values or duplicated rows.

```r
activityisna<- colSums(is.na(dailyActivity))
print(activityisna)
```

```
##                       Id             ActivityDate               TotalSteps
##                        0                        0                        0
##            TotalDistance          TrackerDistance LoggedActivitiesDistance
##                        0                        0                        0
##       VeryActiveDistance ModeratelyActiveDistance      LightActiveDistance
##                        0                        0                        0
##  SedentaryActiveDistance        VeryActiveMinutes      FairlyActiveMinutes
##                        0                        0                        0
##     LightlyActiveMinutes         SedentaryMinutes                 Calories
##                        0                        0                        0
```

```
activitydups<- duplicated(dailyActivity)
print(dailyActivity[activitydups, ])
```

```
##  [1] Id                       ActivityDate             TotalSteps
##  [4] TotalDistance            TrackerDistance          LoggedActivitiesDistance
##  [7] VeryActiveDistance       ModeratelyActiveDistance LightActiveDistance
## [10] SedentaryActiveDistance  VeryActiveMinutes        FairlyActiveMinutes
## [13] LightlyActiveMinutes     SedentaryMinutes         Calories
## <0 rows> (or 0-length row.names)
```

This same code block was run for all of our data sets.

dailyActivity, Calories, Heart rate and Intensities data frames did not contain any missing values or duplicates. The data types are also correct. However during the cleaning process I found that our Sleep Data set had 3 duplicate rows. I found the rows and deleted them with the following code

```
sleepdups<- duplicated(sleep)
print(sleep[sleepdups, ])
sleepDupRemov<- sleep[!sleepdups, ]
```

During the cleaning process I found that our WeightlogInfo Data set had 65 missing values in the “Fat” column. Because of this I decided to drop the entire column.

I also discovered that our weight data set had 65 missing values in the “Fat” column. Because this is an abnormally high number of missing values and we don’t have many rows in the first place, I will be deleting the entire row from the study to limit our exposure to incomplete data.

```
weightisna<- colSums(is.na(weight))
print(weightisna)
weightcleaned<- subset(weight, select = -Fat)
```

While reviewing our data frames I noticed that the data for Steps was most likely contained inside the dailyActivity data frame. To confirm this I ran the following code. It will check if all data from ID and TotalSteps column in Steps is contained within the TotalSteps column of dailyActivity. If it is, the if loop will automatically drop the data set. If it is not the Data set will remain, and we will use it for analysis. return our True message, and we can drop the data set with from our study with rm() to eliminate redundancy. I will do the same thing with the Calories data set as I suspect it will be the same case.

```
iscontained<- all(steps$StepTotal %in% dailyActivity$TotalSteps) && all(steps$Id %in% dailyActivity$Id)
 if(iscontained){
   cat("All data is present, freely drop the data set.")
 } else {
   cat("Not all data is present, keep dataset. ")
 }
```

```
## All data is present, freely drop the data set.
```

```r
iscontained<- all(calories$Calories %in% dailyActivity$Calories) && all(calories$Id %in% dailyActivity$I
 if(iscontained){
   cat("All data is present, freely drop the data set.")
 } else {
   cat("Not all data is present, keep dataset. ")
 }
```

```
## All data is present, freely drop the data set.
```

As such I will be removing both Calories and Steps from my analysis as all the data is contained in dailyActivity.

With all data loaded, corrected, cleaned and verified, we can move on to our analysis. ## Analyze Phase

To start our analysis we will put each data frame through both n_distinct() and nrow() to determine how many unique values there are in the data frame, and to determine the number of rows in the entire data frame respectively.

```r
n_distinct(dailyActivity$Id)
```

```
## [1] 33
```

```r
nrow(dailyActivity)
```

```
## [1] 940
```

```r
n_distinct(calories$Id)
```

```
## [1] 33
```

```r
nrow(calories)
```

```
## [1] 940
```

```r
n_distinct(intensities$Id)
```

```
## [1] 33
```

```r
nrow(intensities)
```

```
## [1] 940
```

```r
n_distinct(sleep$Id)
```

```
## [1] 24
```

```r
nrow(sleep)
```

```
## [1] 413
```

```r
n_distinct(weight$Id)
```

```
## [1] 8
```

```r
nrow(weight)
```

```
## [1] 67
```

```r
n_distinct(heartrate$Id)
```

```
## [1] 14
```

```r
nrow(heartrate)
```

```
## [1] 2483658
```

Initial Insights

From these prints we can make a few observations. dailyActivity, Calories, and Intensities all have the same number of unique ID's at 33. This is the largest amount of unique ID's in any data set so we can confidently assume this study has a maximum of 33 people involved at any given time.

This also shows us that we only have 24 people who participated in the Sleep portion of the data, fewer contributed to the heart rate data at 14 unique ID's, and even fewer people contributed to the weight portion of the data, at 8 Unique Id's. This is obviously a very small sample size and as such this analysis will hardly be accurately indicative of any larger trends. As the Weight and Heart rate data sets contain so few complete data sets we will be extremely careful with the conclusions we draw, and will explain as such in our Share phase.

Digging Deeper

We can use the summary() function to get a detailed summary of the data frame to continue making observations.

```r
dailyActivity %>%
  select(TotalSteps,
         TotalDistance,
         VeryActiveMinutes,
         FairlyActiveMinutes,
         LightlyActiveMinutes,
         SedentaryMinutes,
         Calories) %>%
  summary()
```

```
##    TotalSteps    TotalDistance    VeryActiveMinutes FairlyActiveMinutes
##  Min.   :    0   Min.   : 0.000   Min.   :  0.00    Min.   :  0.00
##  1st Qu.: 3790   1st Qu.: 2.620   1st Qu.:  0.00    1st Qu.:  0.00
##  Median : 7406   Median : 5.245   Median :  4.00    Median :  6.00
##  Mean   : 7638   Mean   : 5.490   Mean   : 21.16    Mean   : 13.56
##  3rd Qu.:10727   3rd Qu.: 7.713   3rd Qu.: 32.00    3rd Qu.: 19.00
##  Max.   :36019   Max.   :28.030   Max.   :210.00    Max.   :143.00
##  LightlyActiveMinutes SedentaryMinutes    Calories
##  Min.   :  0.0        Min.   :   0.0   Min.   :   0
##  1st Qu.:127.0        1st Qu.: 729.8   1st Qu.:1828
##  Median :199.0        Median :1057.5   Median :2134
##  Mean   :192.8        Mean   : 991.2   Mean   :2304
##  3rd Qu.:264.0        3rd Qu.:1229.5   3rd Qu.:2793
##  Max.   :518.0        Max.   :1440.0   Max.   :4900
```

From this data we can easily determine that the participants took an average of 7638 steps in 1 day, and they walked a total of 5.4 miles on average. If we compare that to some guideline standards from the CDC for example our data takes on another meaning. For example, the CDC recommends that people take an average of 10,000 steps a day for optimal health, meaning that our participants were only getting about 76% of their daily steps in.

Additionally we can compare the number of minutes for each activity level against themselves and the SedentaryMinutes to get an idea for how active our participants are. The avg number of Very, Fairly, and Lightly Active Minutes were 21, 13.5, and 192 minutes per day. Compared to an average of almost ~1000 sedentary minutes, or almost 500% longer than the combined Activity time.

Lets do the same with our Weight and sleep Data sets. Remember that participants did not heavily contribute to the Weight data set, so this data could be coincidental and inaccurate.

```r
sleep %>%
  select(TotalTimeInBed,
         TotalMinutesAsleep,
         TotalSleepRecords) %>%
  summary()
```

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

From the above data we can draw some conclusions about our participants sleeping habits. The average person had 1 sleep event and spent an average of ~7 hours asleep a night. We can also see that there is a high variance in how long it might take someone to fall asleep. We can see that the minimum time to sleep is 3 minutes, Time in bed - Time asleep, but we can see that the maximum amount of time someone spent trying to fall asleep was nearly ~3 hours.

```r
weight %>%
  select(WeightPounds,
         BMI) %>%
  summary()
```

```
##   WeightPounds        BMI       
##  Min.   :116.0   Min.   :21.45  
##  1st Qu.:135.4   1st Qu.:23.96  
##  Median :137.8   Median :24.39  
##  Mean   :158.8   Mean   :25.19  
##  3rd Qu.:187.5   3rd Qu.:25.56  
##  Max.   :294.3   Max.   :47.54  
```

From the above data run on our Weight data frame we can see that the avg participant of our study weighted ~159lbs. Again, there was quite a high variance between the max and mins, with the minimum weight being 116lbs and the maximum weight being 294lbs. Again keeping in mind that we only have 8 unique IDs for this data set.

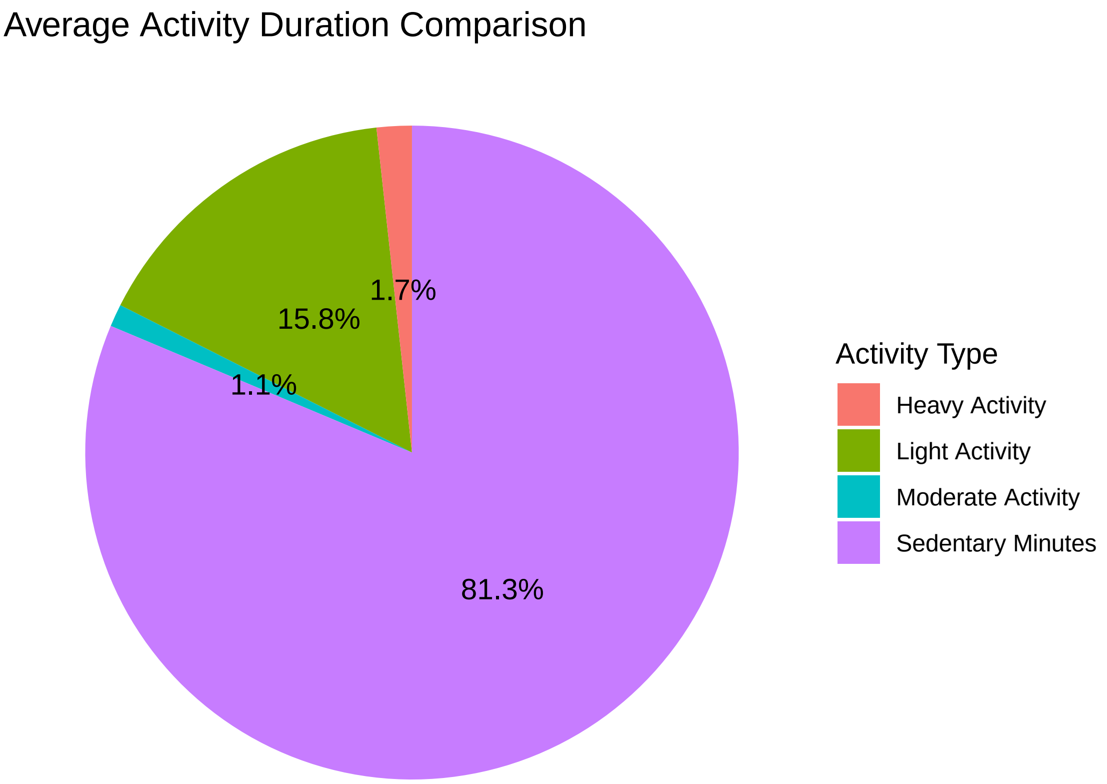
Summary of key findings

- According to the CDC, the participants of this study are under the recommended daily steps and distance. CDC recommends 10,000 steps a day totaling to about 5 miles, while our participants made an an average of ~7600. Interestingly the mean travel distance of 5.4 miles is above the recommended 5 miles from the CDC.
- The majority of exercise performed while under this study falls into the Lightly Active category, with each participant performing almost ~200 minutes of Light Activity per day. Compared against length of sedentary time, which is an average of almost ~1000 minutes across participants, totaling over 16 1/2 hours of inactivity. This likely includes sleep time.
- Participants generally had decent sleep. On average a person in the study fell asleep 1 time and stayed asleep for 7 hours a night. This is slightly above the United States average which sits at just under 7 hours a day.
- The participants on average eat about ~2300 calories per day. Again we see a high Min-Max variance, with the highest calorie/day count being ~4000.
- My impression from the data is that the participants in this study are a fairly average group of people.

Share Phase (Visualization)

Pie Chart relating all Activity types

```
average_durations<- colMeans(dailyActivity[c("LightlyActiveMinutes", "FairlyActiveMinutes", "VeryActive
activity_types<- c("Light Activity", "Moderate Activity", "Heavy Activity", "Sedentary Minutes")
summary_data <- data.frame(activity_type = activity_types, duration = average_durations)
total_duration<- sum(summary_data$duration)
activityviz<- ggplot(summary_data, aes(x="", y = duration, fill= activity_types)) +
  geom_bar(stat="identity", width = 1) +
  coord_polar(theta = "y")+
  labs(title= "Average Activity Duration Comparison",
       fill = "Activity Type")+
  theme_void()+

  geom_text(aes(label= paste0(round((duration / total_duration)*100, 1), "%")),
            position = position_stack(vjust= 0.5))
print(activityviz)
```

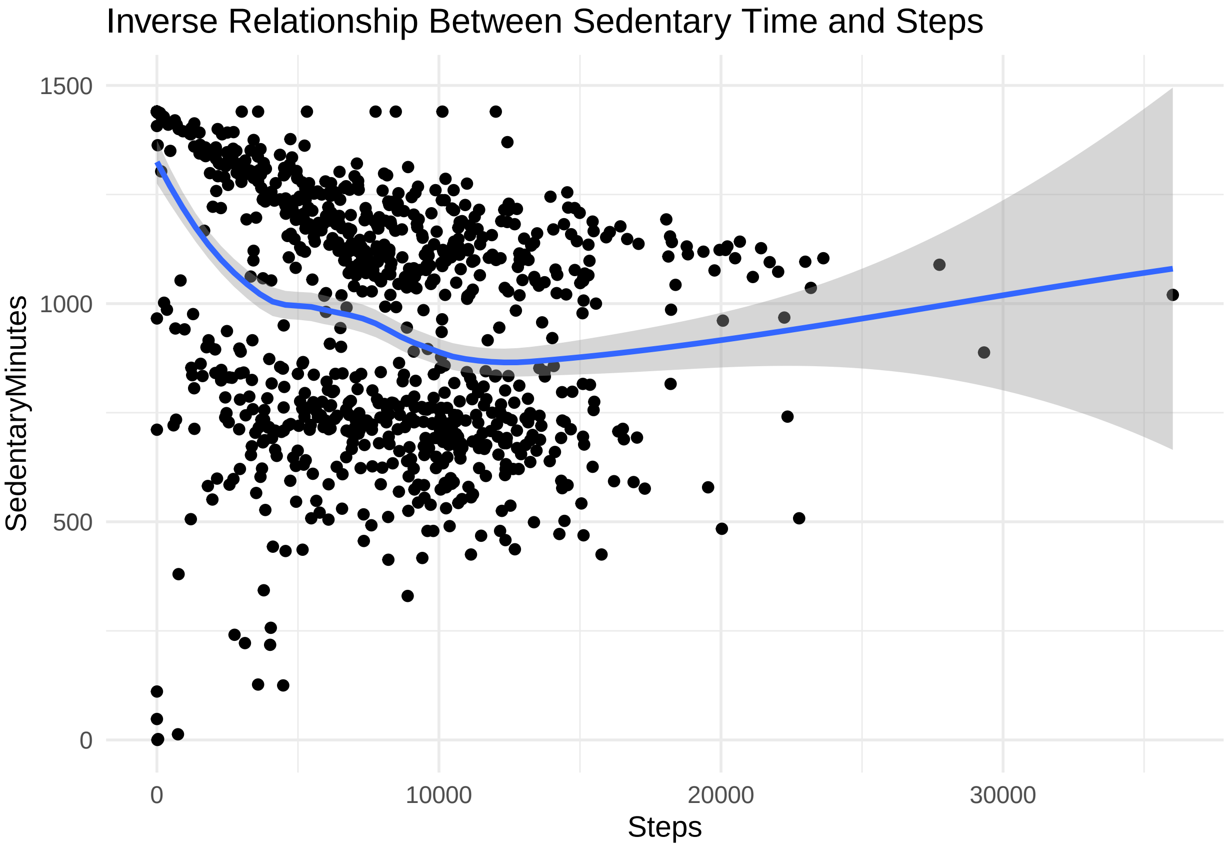


Remember that Sedentary Minutes also includes Sleep minutes. From this Pie Chart we can easily see the disparity between the Sedentary portion of the participants day compared to the active portions. The active portions total just 18.6% of the total day. If we consider that this likely includes most general movement at work and around the house as well. We can confirm this with another graph. Let's take a look at Average Activity (Sedentary Minutes) compared against Steps.

Sedentary vs Steps

```
scatter_plot <- ggplot(data=dailyActivity, aes(x= TotalSteps, y= SedentaryMinutes))+
  geom_point()+
  geom_smooth()+
  labs(title= "Inverse Relationship Between Sedentary Time and Steps", y="SedentaryMinutes", x= "Steps"
  theme_minimal()
print(scatter_plot)
```

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

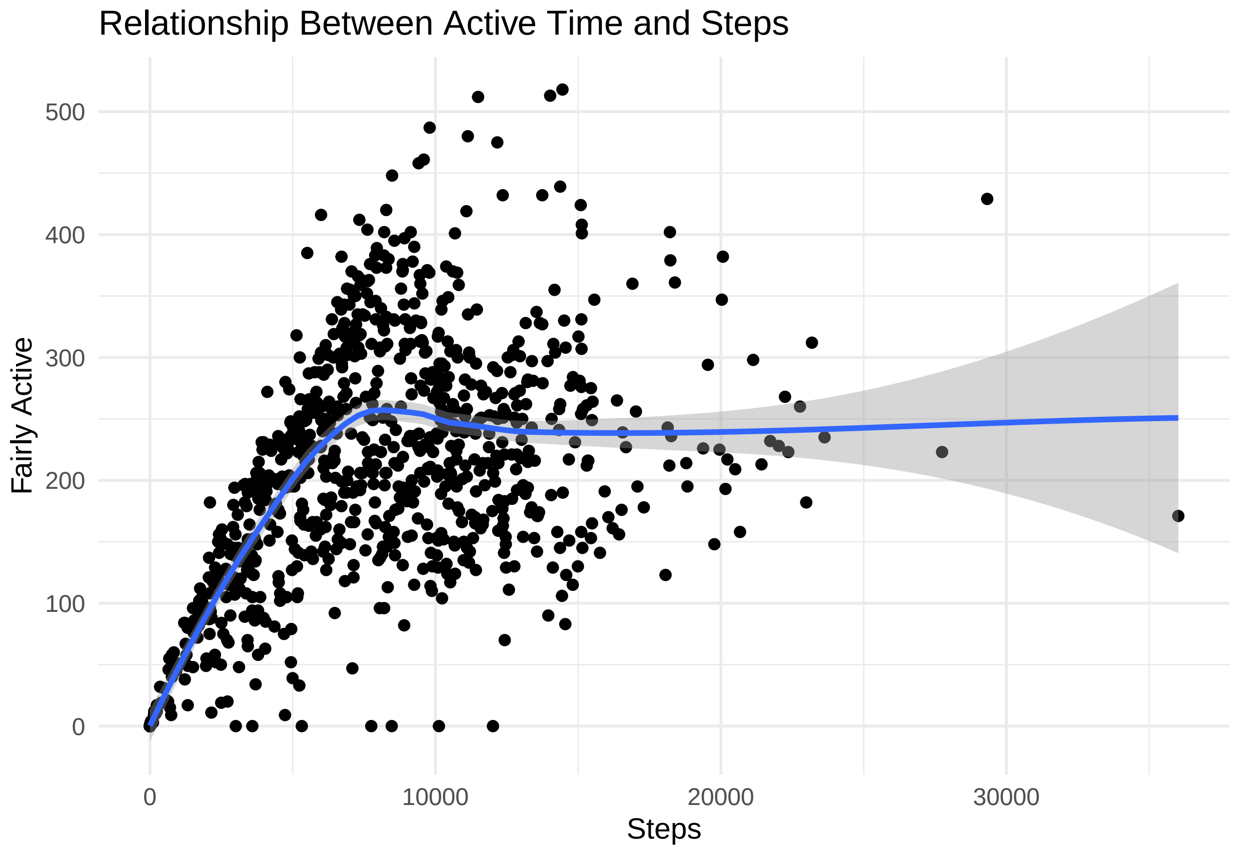


From this graph we can confirm the relationship between Steps and Sedentary Time. While sedentary minutes increase there is no noticeable gain to steps as Sedentary Time increases. This is expected, as someone who is sitting is not taking steps. We can compare this against a graph with FairlyActive data against steps.

FairlyActive vs Steps

```
scatter_plot2 <- ggplot(data=dailyActivity, aes(x= TotalSteps, y= LightlyActiveMinutes))+
  geom_point()+
  geom_smooth()+
  labs(title= "Relationship Between Active Time and Steps", y= "Fairly Active", x= "Steps")+
  theme_minimal()
print(scatter_plot2)
```

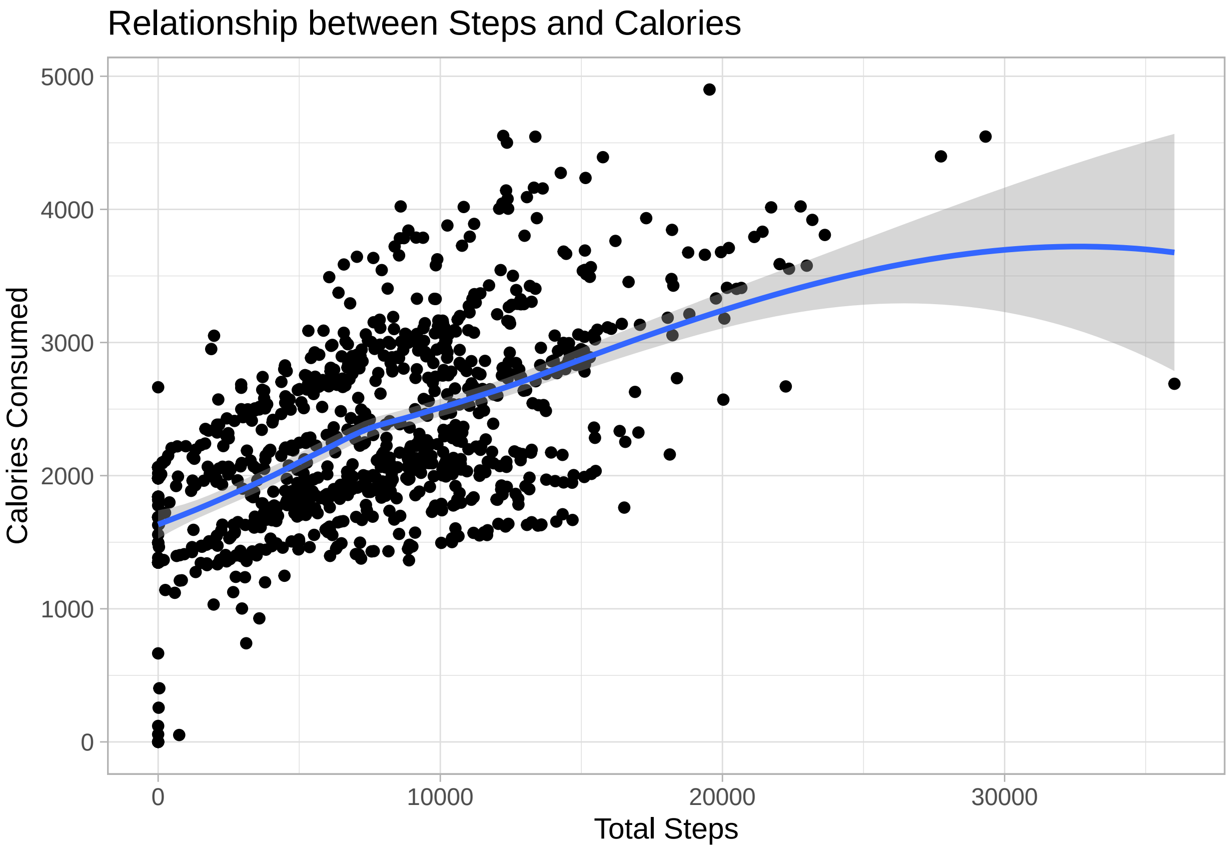
```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



From this we can see that while there is an uptrend showing for some participants while Active minutes increase so do Steps. However it is interesting that the trend line follows a more simple curve. It shows a sharp trend line between the amount of fairly active minutes a person performs in relation to their total steps for the day. As we can see as activity increases so do steps. Again this is logical, as to be active you typically have to move. It is interesting to note that for most people after around 10,000 to 12,000 steps things seem to level off.

```
caloriescatter<- ggplot(data=dailyActivity, aes(x=TotalSteps, y=Calories))+
  geom_point()+
  geom_smooth()+
  labs(title = "Relationship between Steps and Calories", x="Total Steps", y = "Calories Consumed")+
  theme_light()
print(caloriescatter)
```

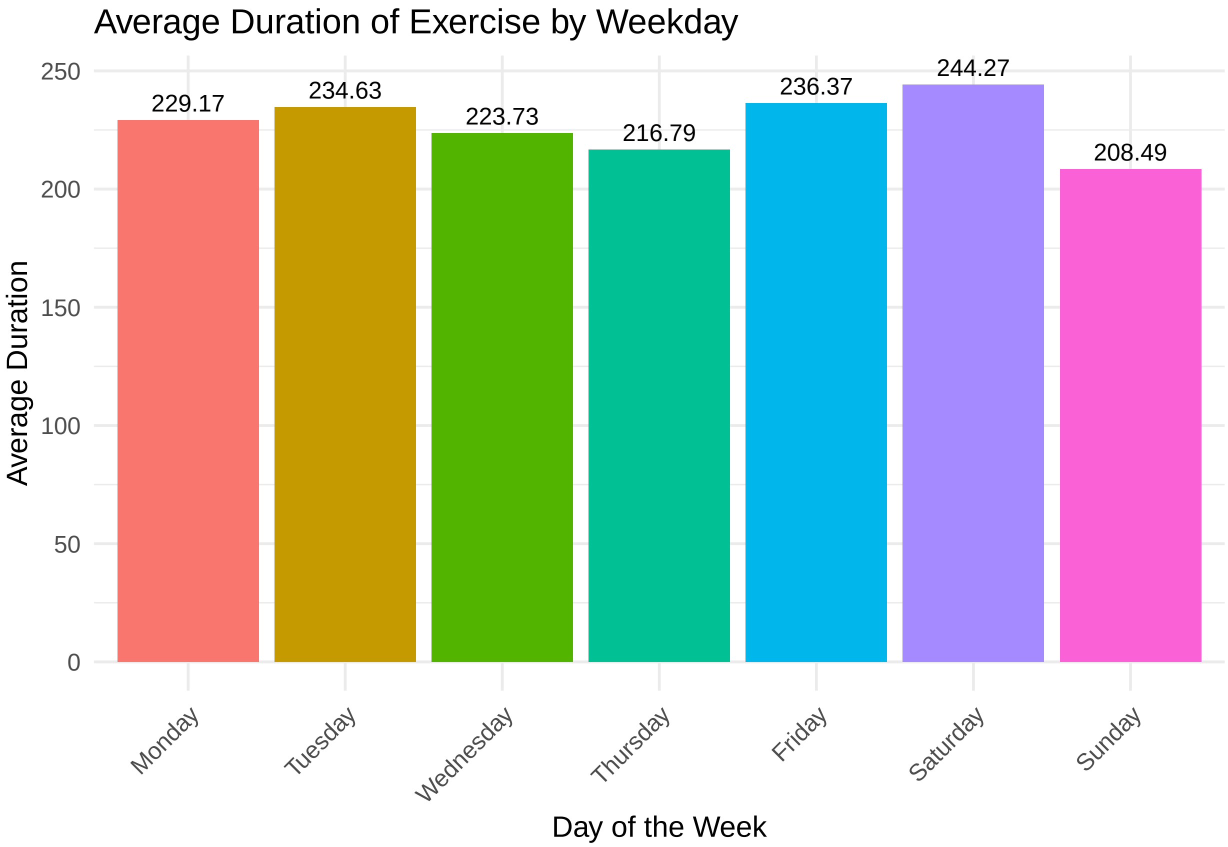
```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



This graph again gives us a very clear trend line to consider. It shows that as Total Steps go up, so does Calories Consumed. If we had better Weight data we could overlay that info on top of this graph, but as our Weight data set had only 8 unique users and the Min-Max variance was so high it would be misleading and potentially damaging to our study.

Duration on different days of the week

```
dailyActivity$ActivityDate <- as.POSIXct(dailyActivity$ActivityDate, format = "%m/%d/%Y", tz = Sys.time

dailyActivity <- dailyActivity %>%
  mutate(TotalExercise = LightlyActiveMinutes + FairlyActiveMinutes + VeryActiveMinutes)

dailyActivity$DayOfWeek<- weekdays(dailyActivity$ActivityDate)

ordered_days <- c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday")

dailyActivity$DayOfWeek <- factor(dailyActivity$DayOfWeek, levels = ordered_days)

avg_exercise_by_weekday <- dailyActivity %>%
  group_by(DayOfWeek) %>%
  summarise(AvgDuration = mean(TotalExercise))

bar_plot <- ggplot(data = avg_exercise_by_weekday, aes(x = DayOfWeek, y = AvgDuration, fill = DayOfWeek
  geom_bar(stat = "identity") +
  labs(title = "Average Duration of Exercise by Weekday",
       x = "Day of the Week", y = "Average Duration") +
  scale_fill_discrete(name = "Day of the Week") +
  geom_text(aes(label = round(AvgDuration, 2)), vjust = -0.5, size = 3) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1), legend.position = "none")
print(bar_plot)
```



The above graph is incredibly useful for getting a gauge of which day people generally prefer to exercise on. From the graph above we can infer that people exercise for longer on Saturday, and the least on Saturday. This makes some logical sense, people would be energized with a day off on Saturday but Sunday they'd relax before Monday. We also see a drop on Wednesday and Thursday, towards the middle of the week when people would likely be the most tired.

Act Phase

Bellabeat has enjoyed large success since its founding by empowering women in their pursuit of a more healthy self. Under the studious leadership of Urška Sršen and Sando Mur Bellabeat is poised to advance itself further into the ever growing market of women's health products. After reviewing the available data I can offer insights into how participants are using Health Data and how this Data might be used to increase Bellabeat's corporate success.

Participants

This Fit Bit Tracker Data makes it clear that the participants are all at least slightly active, but most experience long hours of sedentary activity, likely at a desk or other seated area. It is clear from the data that the more steps someone takes, the more physically fit they are. It is also clear from the data that most participants are among the average. All together there were not enough participants to obtain a study indicative of a larger population, but we can still gleam insights from the data.

Recommendations to the Marketing Team

- The average time a participant stayed sedentary throughout the day is too high. On average a person was sedentary for more than 16.5 hours of their day. Compared to an average of ~225 minutes of activity a day, or a little under ~4 hours day. The vast majority of this activity is labeled as Lightly Active, or something akin to a moderate walking pace.
  - I would recommend that the marketers make a push to increase activity. Perhaps a notification of some sort if a participant has been idle for longer than a certain amount of time. I would also recommend notifying the user of their activity throughout the day, and perhaps offer some sort of in-app reward for reaching a specific threshold of activity per day.
- The average participant slept for around ~7 hours a night, and only had 1 sleep event.
  - This is near to the national average, even a little above, but it can be increased more. I would suggest a notification telling the user that it is nearing their bed time. I would also look into decreasing light exposure for users after a certain time, perhaps dimming the screen automatically. I would also recommend notifying the user of their sleep every morning as well as offering a detailed breakdown at the end of every week.
- The participants in this study took almost ~2500 steps fewer than the recommended CDC guidelines. The CDC recommends an adult walks approximately 10,000 steps a day, or roughly equal to ~ 5 miles. Interestingly the data shows that people walked longer than 5 miles on average, but still came out over 2000 steps below the guidelines. Per the CDC, compared to taking just 4000 steps, taking 8000 steps is associated with a 51% reduced risk of mortality from all causes. Taking 12,000 steps a day, when compared to 4000, will reduce your risk by nearly 65%.
  - I recommend ensuring the app is tracking distance correctly, as well as putting a greater emphasis on Step count. Step count is a quick and easy way for a person to get a gauge on how active they have been for the day. I would also recommend implementing some kind of reward for reaching specific Step milestones, and perhaps some kind of running average for the month for greater visibility.
- We did not take too hard of a look at weight simply because the data set was very flawed and wouldn't have been indicative of larger trends. But we did look at calories and how they relate to steps. As people exercise and move more, their steps increase. So does the fuel the body needs to keep going.
  - I would recommend Bellabeat stress to their users how important diet while exercising. I would also attempt to improve the weight tracking for the app in an attempt to increase the amount of participants in the data.

Thank you for reading my case study! If you have any improvements or comments please let me know.

Daniel Anderson