Bellabeat Case Study

Ruiying Yang

Table of Contents

### 1. All about the company - Bellabeat

Bellabeat is a high-tech manufacturer of health-focused products for women and it is a successful small company, but they have the potential to become a larger player in the global smart device market. Urška Sršen, cofounder and Chief Creative Officer of Bellabeat, believes that analyzing smart device fitness data could help unlock new growth opportunities for the company.

### 2. Ask

#### 2.1 Identify the business task

1. What are some trends in smart device usage?
2. How could these trends apply to Bellabeat customers?
3. How could these trends help influence Bellabeat marketing strategy?

#### 2.2 Key stakeholders

1. Urška Sršen: Bellabeat’s cofounder and Chief Creative Officer.
2. Sando Mur: Mathematician and Bellabeat’s cofounder.
3. Bellabeat marketing analytics team.

### 3. Prepare

#### 3.1 Basic Info of the data

1. This dataset generated by respondents to a distributed survey via Amazon Mechanical Turk between 03.12.2016-05.12.2016. Thirty eligible Fitbit users consented to the submission of personal tracker data.
2. There might be some problem with bias, because the data only comes from Fitbit users. That means the data with other resources are not considered. They did not indicate how did they choose the 30 data, randomly or according to certain patterns. The data is also chosen in a certain time of period. It might also cause bias.
3. From the metadata we can see that the license is CC0: Public Domain. The data is opened for public and accessible on Kaggle.

#### 3.2 The structure and some basic statistic info of the data

##### 3.2.1 Install the necessary packages and import the data.

#install.packages("sqldf")  
library(pacman)  
p\_load(moments,  
 ggplot2,  
 cowplot,  
 RColorBrewer,  
 stargazer,  
 tidyverse,  
 sqldf,  
 lubridate,  
 dplyr)  
  
getwd()

## [1] "/Users/yangruiying/Desktop/Data Analyse/Bellabeat\_Case\_Study"

Activity <- read.csv("/Users/yangruiying/Desktop/Data\ Analyse/Bellabeat\_Case\_Study/Fitabase\ Data/dailyActivity\_merged.csv", stringsAsFactors = T, header = T, sep = ",")  
  
Calories <- read.csv("/Users/yangruiying/Desktop/Data\ Analyse/Bellabeat\_Case\_Study/Fitabase\ Data/dailyCalories\_merged.csv", stringsAsFactors = T, header = T, sep = "," )  
  
Steps <- read.csv("/Users/yangruiying/Desktop/Data\ Analyse/Bellabeat\_Case\_Study/Fitabase\ Data/dailySteps\_merged.csv", stringsAsFactors = T, header = T, sep = "," )  
  
Weight <- read.csv("/Users/yangruiying/Desktop/Data\ Analyse/Bellabeat\_Case\_Study/Fitabase\ Data/weightLogInfo\_merged.csv", stringsAsFactors = T, header = T, sep = "," )  
  
Intensity <- read.csv("/Users/yangruiying/Desktop/Data\ Analyse/Bellabeat\_Case\_Study/Fitabase\ Data/dailyIntensities\_merged.csv", stringsAsFactors = T, header = T, sep = "," )  
  
Sleep <- read.csv("/Users/yangruiying/Desktop/Data\ Analyse/Bellabeat\_Case\_Study/Fitabase\ Data/sleepDay\_merged.csv", stringsAsFactors = T, header = T, sep = "," )  
  
HourlyIntensity <- read.csv("/Users/yangruiying/Desktop/Data\ Analyse/Bellabeat\_Case\_Study/Fitabase\ Data/hourlyIntensities\_merged.csv", stringsAsFactors = T, header = T, sep = "," )

##### 3.2.2 How the data is organized.

str(Activity)

## 'data.frame': 940 obs. of 15 variables:  
## $ Id : num 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...  
## $ ActivityDate : Factor w/ 31 levels "4/12/2016","4/13/2016",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ TotalSteps : int 13162 10735 10460 9762 12669 9705 13019 15506 10544 9819 ...  
## $ TotalDistance : num 8.5 6.97 6.74 6.28 8.16 ...  
## $ TrackerDistance : num 8.5 6.97 6.74 6.28 8.16 ...  
## $ LoggedActivitiesDistance: num 0 0 0 0 0 0 0 0 0 0 ...  
## $ VeryActiveDistance : num 1.88 1.57 2.44 2.14 2.71 ...  
## $ ModeratelyActiveDistance: num 0.55 0.69 0.4 1.26 0.41 ...  
## $ LightActiveDistance : num 6.06 4.71 3.91 2.83 5.04 ...  
## $ SedentaryActiveDistance : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ VeryActiveMinutes : int 25 21 30 29 36 38 42 50 28 19 ...  
## $ FairlyActiveMinutes : int 13 19 11 34 10 20 16 31 12 8 ...  
## $ LightlyActiveMinutes : int 328 217 181 209 221 164 233 264 205 211 ...  
## $ SedentaryMinutes : int 728 776 1218 726 773 539 1149 775 818 838 ...  
## $ Calories : int 1985 1797 1776 1745 1863 1728 1921 2035 1786 1775 ...

head(Activity)

## 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

str(Calories)

## 'data.frame': 940 obs. of 3 variables:  
## $ Id : num 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...  
## $ ActivityDay: Factor w/ 31 levels "4/12/2016","4/13/2016",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ Calories : int 1985 1797 1776 1745 1863 1728 1921 2035 1786 1775 ...

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

str(Steps)

## 'data.frame': 940 obs. of 3 variables:  
## $ Id : num 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...  
## $ ActivityDay: Factor w/ 31 levels "4/12/2016","4/13/2016",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ StepTotal : int 13162 10735 10460 9762 12669 9705 13019 15506 10544 9819 ...

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

str(Weight)

## 'data.frame': 67 obs. of 8 variables:  
## $ Id : num 1.50e+09 1.50e+09 1.93e+09 2.87e+09 2.87e+09 ...  
## $ Date : Factor w/ 56 levels "4/12/2016 11:59:59 PM",..: 44 45 3 19 42 11 47 13 26 37 ...  
## $ WeightKg : num 52.6 52.6 133.5 56.7 57.3 ...  
## $ WeightPounds : num 116 116 294 125 126 ...  
## $ Fat : int 22 NA NA NA NA 25 NA NA NA NA ...  
## $ BMI : num 22.6 22.6 47.5 21.5 21.7 ...  
## $ IsManualReport: Factor w/ 2 levels "False","True": 2 2 1 2 2 2 2 2 2 2 ...  
## $ LogId : num 1.46e+12 1.46e+12 1.46e+12 1.46e+12 1.46e+12 ...

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

str(Intensity)

## 'data.frame': 940 obs. of 10 variables:  
## $ Id : num 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...  
## $ ActivityDay : Factor w/ 31 levels "4/12/2016","4/13/2016",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ SedentaryMinutes : int 728 776 1218 726 773 539 1149 775 818 838 ...  
## $ LightlyActiveMinutes : int 328 217 181 209 221 164 233 264 205 211 ...  
## $ FairlyActiveMinutes : int 13 19 11 34 10 20 16 31 12 8 ...  
## $ VeryActiveMinutes : int 25 21 30 29 36 38 42 50 28 19 ...  
## $ SedentaryActiveDistance : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ LightActiveDistance : num 6.06 4.71 3.91 2.83 5.04 ...  
## $ ModeratelyActiveDistance: num 0.55 0.69 0.4 1.26 0.41 ...  
## $ VeryActiveDistance : num 1.88 1.57 2.44 2.14 2.71 ...

head(Intensity)

## 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

str(Sleep)

## 'data.frame': 413 obs. of 5 variables:  
## $ Id : num 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...  
## $ SleepDay : Factor w/ 31 levels "4/12/2016 12:00:00 AM",..: 1 2 4 5 6 8 9 10 12 13 ...  
## $ TotalSleepRecords : int 1 2 1 2 1 1 1 1 1 1 ...  
## $ TotalMinutesAsleep: int 327 384 412 340 700 304 360 325 361 430 ...  
## $ TotalTimeInBed : int 346 407 442 367 712 320 377 364 384 449 ...

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

str(HourlyIntensity)

## 'data.frame': 22099 obs. of 4 variables:  
## $ Id : num 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...  
## $ ActivityHour : Factor w/ 736 levels "4/12/2016 1:00:00 AM",..: 7 1 9 11 13 15 17 19 21 23 ...  
## $ TotalIntensity : int 20 8 7 0 0 0 0 0 13 30 ...  
## $ AverageIntensity: num 0.333 0.133 0.117 0 0 ...

head(HourlyIntensity)

## Id ActivityHour TotalIntensity AverageIntensity  
## 1 1503960366 4/12/2016 12:00:00 AM 20 0.333333  
## 2 1503960366 4/12/2016 1:00:00 AM 8 0.133333  
## 3 1503960366 4/12/2016 2:00:00 AM 7 0.116667  
## 4 1503960366 4/12/2016 3:00:00 AM 0 0.000000  
## 5 1503960366 4/12/2016 4:00:00 AM 0 0.000000  
## 6 1503960366 4/12/2016 5:00:00 AM 0 0.000000

* Activity table: There are 940 records and 15 variables, most of the data are numerical and integer. Many people recorded their activity for many days. Most of the variables are grouped according to the amount and intensity of the activities.
* Calories table: There are 940 obs. of 3 variables. The data is also grouped by different customer Ids. But we can notice that, the data in the table calories overlapped with the data in the table Activity. So the data in this table can be just replaced with the analysis of the data in the table Activity.
* Steps table: There are also 940 obs. of 3 variables. The case for this table is the same as the table calories. Because all the variables are included in the Activity data, this table will not be included in our analysis.
* Weight table: There are 67 obs. of 8 variables. Since there are some different rows in this table and there is also Id column, so it can be merged with the data in the table Acitivity.
* Intensity table: There are 940 obs. of 10 variables. All the data are included in the table Acitivity, so we will exclude this table from our analysis.
* Sleep table: There are 413 obs. of 5 variables. We can see that many users record their sleep for many different days, which means maybe what the most important function of the fitness app is to record the sleep. We can still merge this data with the data in the table Activity.
* HourlyIntensity table: There are 22099 obs. of 4 variables. There are so many objects in this table because the intensity is recorded for many times in a day. Since the data in the table Intensity is overlapped with the data in Activity table, we could use this table to examine the time period of doing sport.

We can also notice that the data is store in the long format. After the first peek into the data, we decided that we will analyze the following tables: • Activity • Weight • Sleep • HourlyIntensity

##### 3.2.3 Statistical information of the data

summary(Activity)

## Id ActivityDate TotalSteps TotalDistance   
## Min. :1.504e+09 4/12/2016: 33 Min. : 0 Min. : 0.000   
## 1st Qu.:2.320e+09 4/13/2016: 33 1st Qu.: 3790 1st Qu.: 2.620   
## Median :4.445e+09 4/14/2016: 33 Median : 7406 Median : 5.245   
## Mean :4.855e+09 4/15/2016: 33 Mean : 7638 Mean : 5.490   
## 3rd Qu.:6.962e+09 4/16/2016: 32 3rd Qu.:10727 3rd Qu.: 7.713   
## Max. :8.878e+09 4/17/2016: 32 Max. :36019 Max. :28.030   
## (Other) :744   
## TrackerDistance LoggedActivitiesDistance VeryActiveDistance  
## Min. : 0.000 Min. :0.0000 Min. : 0.000   
## 1st Qu.: 2.620 1st Qu.:0.0000 1st Qu.: 0.000   
## Median : 5.245 Median :0.0000 Median : 0.210   
## Mean : 5.475 Mean :0.1082 Mean : 1.503   
## 3rd Qu.: 7.710 3rd Qu.:0.0000 3rd Qu.: 2.053   
## Max. :28.030 Max. :4.9421 Max. :21.920   
##   
## ModeratelyActiveDistance LightActiveDistance SedentaryActiveDistance  
## Min. :0.0000 Min. : 0.000 Min. :0.000000   
## 1st Qu.:0.0000 1st Qu.: 1.945 1st Qu.:0.000000   
## Median :0.2400 Median : 3.365 Median :0.000000   
## Mean :0.5675 Mean : 3.341 Mean :0.001606   
## 3rd Qu.:0.8000 3rd Qu.: 4.782 3rd Qu.:0.000000   
## Max. :6.4800 Max. :10.710 Max. :0.110000   
##   
## VeryActiveMinutes FairlyActiveMinutes LightlyActiveMinutes SedentaryMinutes  
## Min. : 0.00 Min. : 0.00 Min. : 0.0 Min. : 0.0   
## 1st Qu.: 0.00 1st Qu.: 0.00 1st Qu.:127.0 1st Qu.: 729.8   
## Median : 4.00 Median : 6.00 Median :199.0 Median :1057.5   
## Mean : 21.16 Mean : 13.56 Mean :192.8 Mean : 991.2   
## 3rd Qu.: 32.00 3rd Qu.: 19.00 3rd Qu.:264.0 3rd Qu.:1229.5   
## Max. :210.00 Max. :143.00 Max. :518.0 Max. :1440.0   
##   
## Calories   
## Min. : 0   
## 1st Qu.:1828   
## Median :2134   
## Mean :2304   
## 3rd Qu.:2793   
## Max. :4900   
##

summary(Weight)

## Id Date WeightKg   
## Min. :1.504e+09 5/2/2016 11:59:59 PM : 3 Min. : 52.60   
## 1st Qu.:6.962e+09 4/17/2016 11:59:59 PM: 2 1st Qu.: 61.40   
## Median :6.962e+09 4/18/2016 11:59:59 PM: 2 Median : 62.50   
## Mean :7.009e+09 4/21/2016 11:59:59 PM: 2 Mean : 72.04   
## 3rd Qu.:8.878e+09 4/25/2016 11:59:59 PM: 2 3rd Qu.: 85.05   
## Max. :8.878e+09 5/1/2016 11:59:59 PM : 2 Max. :133.50   
## (Other) :54   
## WeightPounds Fat BMI IsManualReport  
## Min. :116.0 Min. :22.00 Min. :21.45 False:26   
## 1st Qu.:135.4 1st Qu.:22.75 1st Qu.:23.96 True :41   
## Median :137.8 Median :23.50 Median :24.39   
## Mean :158.8 Mean :23.50 Mean :25.19   
## 3rd Qu.:187.5 3rd Qu.:24.25 3rd Qu.:25.56   
## Max. :294.3 Max. :25.00 Max. :47.54   
## NA's :65   
## LogId   
## Min. :1.460e+12   
## 1st Qu.:1.461e+12   
## Median :1.462e+12   
## Mean :1.462e+12   
## 3rd Qu.:1.462e+12   
## Max. :1.463e+12   
##

summary(Sleep)

## Id SleepDay TotalSleepRecords  
## Min. :1.504e+09 4/15/2016 12:00:00 AM: 17 Min. :1.000   
## 1st Qu.:3.977e+09 4/28/2016 12:00:00 AM: 16 1st Qu.:1.000   
## Median :4.703e+09 5/1/2016 12:00:00 AM : 16 Median :1.000   
## Mean :5.001e+09 4/20/2016 12:00:00 AM: 15 Mean :1.119   
## 3rd Qu.:6.962e+09 4/21/2016 12:00:00 AM: 15 3rd Qu.:1.000   
## Max. :8.792e+09 4/23/2016 12:00:00 AM: 15 Max. :3.000   
## (Other) :319   
## TotalMinutesAsleep TotalTimeInBed   
## Min. : 58.0 Min. : 61.0   
## 1st Qu.:361.0 1st Qu.:403.0   
## Median :433.0 Median :463.0   
## Mean :419.5 Mean :458.6   
## 3rd Qu.:490.0 3rd Qu.:526.0   
## Max. :796.0 Max. :961.0   
##

summary(HourlyIntensity)

## Id ActivityHour TotalIntensity   
## Min. :1.504e+09 4/12/2016 1:00:00 AM : 33 Min. : 0.00   
## 1st Qu.:2.320e+09 4/12/2016 1:00:00 PM : 33 1st Qu.: 0.00   
## Median :4.445e+09 4/12/2016 10:00:00 AM: 33 Median : 3.00   
## Mean :4.848e+09 4/12/2016 10:00:00 PM: 33 Mean : 12.04   
## 3rd Qu.:6.962e+09 4/12/2016 11:00:00 AM: 33 3rd Qu.: 16.00   
## Max. :8.878e+09 4/12/2016 11:00:00 PM: 33 Max. :180.00   
## (Other) :21901   
## AverageIntensity  
## Min. :0.0000   
## 1st Qu.:0.0000   
## Median :0.0500   
## Mean :0.2006   
## 3rd Qu.:0.2667   
## Max. :3.0000   
##

n\_distinct(Activity$Id)

## [1] 33

n\_distinct(Weight$Id)

## [1] 8

n\_distinct(Sleep$Id)

## [1] 24

n\_distinct(HourlyIntensity$Id)

## [1] 33

* Activity  
  • Insights of the Activity data:  
  The data in this table comes from 33 different users.  
  Firstly, most of the users are lightly active.The very active minutes is on average only 15 minutes. Usually they sit for almost 16 hours and walk for 3.6km every day. That means the target customers are not so super active or sporty. Secondly, on average women burn 2300 calories, which is not too much. According to NHS.UK, only the people who do lightly active work burn 2300 calories.  
  • Suggestion according to the insights of the data:  
  To remain health, mentally and physically, on one hand, the target customer should first reduce their sedentary hours from 16 hours to 10 hours. Medical researchers have long warned that prolonged sitting is dangerous, associated with a significantly higher risk of heart disease, diabetes, obesity, cancer, and depression, as well as muscle and joint problems. On the other hand, the CDC recommend that most adults aim for 10,000 steps per day. For most people, this is the equivalent of about 8 kilometers, or 5 miles. So the targeted customers should also walk more. Last but not least, the targeted customer should burn more calories per day if they want to lose weight. The suggested calories to burn to lose weight is between 2500 - 3000 calories.
* Weight  
  • Insights of the Weight data:  
  The data in this table comes from 8 different users.  
  From this table we know that the average weight of the customers are 72kg, although we do not know the average height, but according to the BMI average value of 25.19, we know the customers are slightly overweighted. Because if the BMI is 18.5 to 24.9, it falls within the normal or Healthy Weight range.  
  • Suggestion according to the insights of the data:  
  The customers should try to lose weight and control their BMI in the range of 18.5 to 24.9.
* Sleep  
  • Insights of the Sleep data:  
  The data in this table comes from 24 different users.  
  On average, the customers sleep 419 minutes which equals almost 7 hours. That is a good result.
* HourlyIntensity  
  • Insights of the HourlyIntensity data:  
  The data in this table comes from 33 different users.  
  On average, the TotalIntensity is 12.

### 4. Process

#### 4.1 Missing values

# Check for explicit and implicit missing values  
colSums(is.na(Activity))

## 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

colSums(Activity == "")

## 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

colSums(is.na(Weight))

## Id Date WeightKg WeightPounds Fat   
## 0 0 0 0 65   
## BMI IsManualReport LogId   
## 0 0 0

colSums(Weight == "")

## Id Date WeightKg WeightPounds Fat   
## 0 0 0 0 NA   
## BMI IsManualReport LogId   
## 0 0 0

colSums(is.na(Sleep))

## Id SleepDay TotalSleepRecords TotalMinutesAsleep   
## 0 0 0 0   
## TotalTimeInBed   
## 0

colSums(Sleep == "")

## Id SleepDay TotalSleepRecords TotalMinutesAsleep   
## 0 0 0 0   
## TotalTimeInBed   
## 0

colSums(is.na(HourlyIntensity))

## Id ActivityHour TotalIntensity AverageIntensity   
## 0 0 0 0

colSums(HourlyIntensity == "")

## Id ActivityHour TotalIntensity AverageIntensity   
## 0 0 0 0

Weight <- Weight[, -5]  
str(Weight)

## 'data.frame': 67 obs. of 7 variables:  
## $ Id : num 1.50e+09 1.50e+09 1.93e+09 2.87e+09 2.87e+09 ...  
## $ Date : Factor w/ 56 levels "4/12/2016 11:59:59 PM",..: 44 45 3 19 42 11 47 13 26 37 ...  
## $ WeightKg : num 52.6 52.6 133.5 56.7 57.3 ...  
## $ WeightPounds : num 116 116 294 125 126 ...  
## $ BMI : num 22.6 22.6 47.5 21.5 21.7 ...  
## $ IsManualReport: Factor w/ 2 levels "False","True": 2 2 1 2 2 2 2 2 2 2 ...  
## $ LogId : num 1.46e+12 1.46e+12 1.46e+12 1.46e+12 1.46e+12 ...

We can notice that, there are no missing values in those tables except the table Weight. According to the analysis from the last step we know that there are totally 67 records, but there are 65 missing values in the column 65, which means almost all the data are missed. So this column will be dropped and not considered in the further analysis.

#### 4.2 Duplicates

sum(duplicated(Activity))

## [1] 0

sum(duplicated(Weight))

## [1] 0

sum(duplicated(Sleep))

## [1] 3

sum(duplicated(HourlyIntensity))

## [1] 0

Sleep <- Sleep %>%   
 distinct()

We can see that there are 3 duplicates in the table Sleep. The duplicates are deleted.

#### 4.3 Consistency of date and time columns

head(Activity)

## 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

head(Weight)

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

colSums(is.na(Weight))

## Id Date WeightKg WeightPounds BMI   
## 0 0 0 0 0   
## IsManualReport LogId   
## 0 0

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

str(HourlyIntensity)

## 'data.frame': 22099 obs. of 4 variables:  
## $ Id : num 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...  
## $ ActivityHour : Factor w/ 736 levels "4/12/2016 1:00:00 AM",..: 7 1 9 11 13 15 17 19 21 23 ...  
## $ TotalIntensity : int 20 8 7 0 0 0 0 0 13 30 ...  
## $ AverageIntensity: num 0.333 0.133 0.117 0 0 ...

names(Activity)[names(Activity) == "ActivityDate"] <- "Date"  
names(Sleep)[names(Sleep) == "SleepDay"] <- "Date"  
  
Activity$Date <- as.POSIXct(Activity$Date, format = "%m/%d/%Y", tz=Sys.timezone())  
Sleep$Date <- as.POSIXct(Sleep$Date, format = "%m/%d/%Y %I:%M:%S %p", tz=Sys.timezone())  
  
HourlyIntensity$ActivityHour <- as.POSIXct(HourlyIntensity$ActivityHour, format = "%m/%d/%Y %I:%M:%S %p", tz=Sys.timezone())  
HourlyIntensity$Time <- format(HourlyIntensity$ActivityHour, format = "%H:%M:%S")  
Weight$Date <- as.POSIXct(Weight$Date, format = "%m/%d/%Y %I:%M:%S %p", tz=Sys.timezone())

Now that the type of date and time columns are datetime, we can now merge the data.

#### 4.4 Merge the data

ASdata <- merge(Activity, Sleep, by = c("Id", "Date"))  
head(ASdata)

## Id Date TotalSteps TotalDistance TrackerDistance  
## 1 1503960366 2016-04-12 13162 8.50 8.50  
## 2 1503960366 2016-04-13 10735 6.97 6.97  
## 3 1503960366 2016-04-15 9762 6.28 6.28  
## 4 1503960366 2016-04-16 12669 8.16 8.16  
## 5 1503960366 2016-04-17 9705 6.48 6.48  
## 6 1503960366 2016-04-19 15506 9.88 9.88  
## LoggedActivitiesDistance VeryActiveDistance ModeratelyActiveDistance  
## 1 0 1.88 0.55  
## 2 0 1.57 0.69  
## 3 0 2.14 1.26  
## 4 0 2.71 0.41  
## 5 0 3.19 0.78  
## 6 0 3.53 1.32  
## LightActiveDistance SedentaryActiveDistance VeryActiveMinutes  
## 1 6.06 0 25  
## 2 4.71 0 21  
## 3 2.83 0 29  
## 4 5.04 0 36  
## 5 2.51 0 38  
## 6 5.03 0 50  
## FairlyActiveMinutes LightlyActiveMinutes SedentaryMinutes Calories  
## 1 13 328 728 1985  
## 2 19 217 776 1797  
## 3 34 209 726 1745  
## 4 10 221 773 1863  
## 5 20 164 539 1728  
## 6 31 264 775 2035  
## TotalSleepRecords TotalMinutesAsleep TotalTimeInBed  
## 1 1 327 346  
## 2 2 384 407  
## 3 1 412 442  
## 4 2 340 367  
## 5 1 700 712  
## 6 1 304 320

### 5. Analyze and Share

#### 5.1 The relationship between BMI and WeightKg

ggplot(Weight, aes(x = WeightKg, y = BMI)) +  
 geom\_point(alpha = 0.5) +  
 geom\_smooth(method='lm',  
 se = TRUE,  
 colour = 'coral') +  
 geom\_smooth(method='loess',  
 se = FALSE,  
 colour = 'royalblue') +  
 labs(title = "The relationship between WeightKg and BMI",  
 x = "WeightKg",  
 y = "BMI") +  
 theme(plot.title = element\_text(color = "royalblue4", face = "bold",   
 size = 10, hjust =0.5),  
 axis.title.x = element\_text(color = "royalblue4", size = 8,   
 face = "italic", hjust =0.5, vjust = 1),  
 axis.title.y = element\_text(color = "royalblue4", size = 8,   
 face = "italic", hjust =0.5, vjust = 1))

## `geom\_smooth()` using formula 'y ~ x'  
## `geom\_smooth()` using formula 'y ~ x'



• Insights from the graph:  
From this graph we can see that there is clearly a linear relationship between WeightKg and BMI. The more weight one has, the more BMI one gets.  
• Suggestion according to the insights:  
If customers want to get good BMI, they need to first lose weight. We can calculate how much weight they need to lose to get the targeted BMI. To do that, we will model the linear relationship so that the values of the both variables can be calculated.

##### 5.1.1 Modelling the relationship and evaluating the model

# Modelling  
lm0 <- lm(BMI ~ WeightKg, data = Weight)  
summary(lm0)

##   
## Call:  
## lm(formula = BMI ~ WeightKg, data = Weight)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.9841 -1.7840 0.4608 0.6021 12.3942   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 13.51151 1.35739 9.954 1.07e-14 \*\*\*  
## WeightKg 0.16205 0.01851 8.757 1.33e-12 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.093 on 65 degrees of freedom  
## Multiple R-squared: 0.5412, Adjusted R-squared: 0.5342   
## F-statistic: 76.68 on 1 and 65 DF, p-value: 1.329e-12

# Predict  
pred.lm0 <- predict(lm0, newdata = Weight)   
summary(pred.lm0)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 22.04 23.46 23.64 25.19 27.29 35.15

# Errors that might exist between our Model and predict values  
(MAE.lm0 <- mean(abs(Weight$BMI - pred.lm0)))

## [1] 1.382424

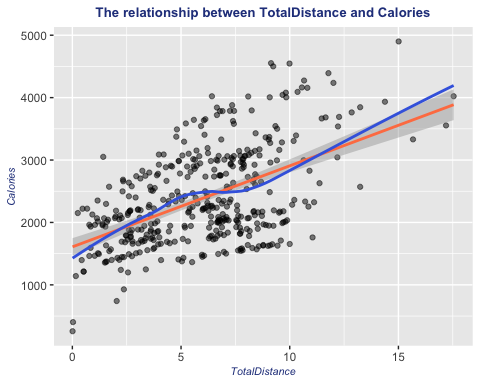
There is an error of 1.38 which means our model performance is really good.

To have lose weight, people need to burn calories, Let’s have a look how to better control calories.

#### 5.2 Relationships between TotalDistance and Calories

ggplot(ASdata, aes(x = TotalDistance, y = Calories)) +  
 geom\_point(alpha = 0.5) +  
 geom\_smooth(method='lm',  
 se = TRUE,  
 colour = 'coral') +  
 geom\_smooth(method='loess',  
 se = FALSE,  
 colour = 'royalblue') +  
 labs(title = "The relationship between TotalDistance and Calories",  
 x = "TotalDistance",  
 y = "Calories") +  
 theme(plot.title = element\_text(color = "royalblue4", face = "bold",   
 size = 10, hjust =0.5),  
 axis.title.x = element\_text(color = "royalblue4", size = 8,   
 face = "italic", hjust =0.5, vjust = 1),  
 axis.title.y = element\_text(color = "royalblue4", size = 8,   
 face = "italic", hjust =0.5, vjust = 1))

## `geom\_smooth()` using formula 'y ~ x'  
## `geom\_smooth()` using formula 'y ~ x'



• Insights from the graph:  
From this graph we can see that there is clearly a linear relationship between TotalDistance and calories. The more one walks, the more calories one burns  
• Suggestion according to the insights:  
If the customer wants to burn more calories, we can calculate how much distance they need to walk according to the amount of calories they want to burn. To do that, we will model the linear relationship so that the values of the both variables can be calculated.

##### 5.2.1 Modelling the relationship and evaluating the model

# Modelling  
lm1 <- lm(TotalDistance ~ Calories, data = ASdata)  
summary(lm1)

##   
## Call:  
## lm(formula = TotalDistance ~ Calories, data = ASdata)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -5.9766 -2.0377 -0.0575 1.7352 8.7412   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.0142759 0.4250321 2.386 0.0175 \*   
## Calories 0.0020919 0.0001696 12.336 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.601 on 408 degrees of freedom  
## Multiple R-squared: 0.2717, Adjusted R-squared: 0.2699   
## F-statistic: 152.2 on 1 and 408 DF, p-value: < 2.2e-16

# Predict  
pred.lm1 <- predict(lm1, newdata = ASdata)   
summary(pred.lm1)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1.552 4.865 5.631 6.012 7.123 11.264

# Errors that might exist between our Model and predict values  
(MAE.lm1 <- mean(abs(ASdata$TotalDistance - pred.lm1)))

## [1] 2.139857

Based on the model we get, we can predict the new data if we have. Since we do not have any new data, we just evaluated our model using the same data. The MAE is only 2.139, which means our model’s performance is pretty good.

Notice that people need to walk more if they want to lose calories, let us have a look when do people do sport more intensively.

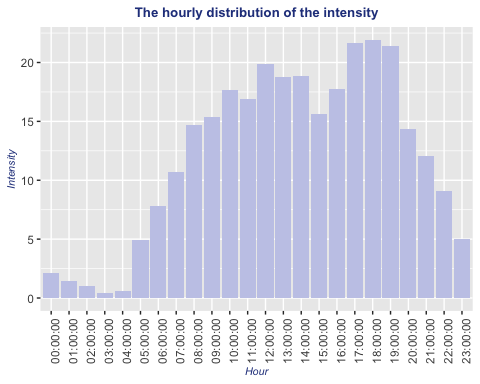
#### 5.3 The timely intensity of doing activities.

str(HourlyIntensity)

## 'data.frame': 22099 obs. of 5 variables:  
## $ Id : num 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...  
## $ ActivityHour : POSIXct, format: "2016-04-12 00:00:00" "2016-04-12 01:00:00" ...  
## $ TotalIntensity : int 20 8 7 0 0 0 0 0 13 30 ...  
## $ AverageIntensity: num 0.333 0.133 0.117 0 0 ...  
## $ Time : chr "00:00:00" "01:00:00" "02:00:00" "03:00:00" ...

int\_new <- HourlyIntensity %>%   
 group\_by(Time) %>%   
 summarise(mean\_total\_int = mean(TotalIntensity))  
  
ggplot(int\_new, aes(x = Time, y = mean\_total\_int)) +  
 geom\_histogram(stat = "identity", fill='#C5CAE9') +  
 labs(title = "The hourly distribution of the intensity",  
 x = "Hour",  
 y = "Intensity") +  
 theme(plot.title = element\_text(color = "royalblue4", face = "bold",   
 size = 10, hjust =0.5),  
 axis.title.x = element\_text(color = "royalblue4", size = 8,   
 face = "italic", hjust =0.5, vjust = 1),  
 axis.title.y = element\_text(color = "royalblue4", size = 8,   
 face = "italic", hjust =0.5, vjust = 1),  
 axis.text.x = element\_text(angle = 90))

## Warning: Ignoring unknown parameters: binwidth, bins, pad



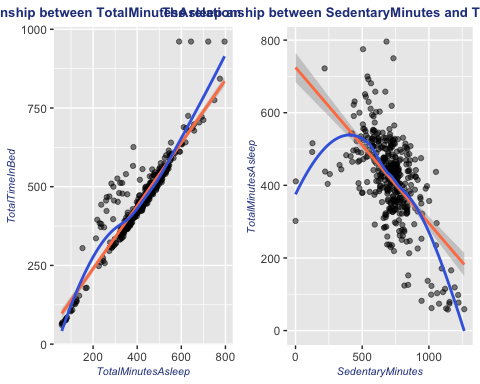
• Insights from the graph:  
From this graph we can see that there is from 17:00 to 19:00 people is most active. This is probably because that in this time period people just finish their work or school. • Suggestion according to the insights:  
If the customers want to get more exercises, a notification in this time of period is the best.

After examining the factors which are related to the activities, let us have a look at the factor sleep.

#### 5.4 Relationships between sleeptime and other factors.

A <-   
ggplot(ASdata, aes(x = TotalMinutesAsleep, y = TotalTimeInBed)) +  
 geom\_point(alpha = 0.5) +  
 geom\_smooth(method='lm',  
 se = TRUE,  
 colour = 'coral') +  
 geom\_smooth(method='loess',  
 se = FALSE,  
 colour = 'royalblue') +  
 labs(title = "The relationship between TotalMinutesAsleep and TotalTimeInBed",  
 x = "TotalMinutesAsleep",  
 y = "TotalTimeInBed") +  
 theme(plot.title = element\_text(color = "royalblue4", face = "bold",   
 size = 10, hjust =0.5),  
 axis.title.x = element\_text(color = "royalblue4", size = 8,   
 face = "italic", hjust =0.5, vjust = 1),  
 axis.title.y = element\_text(color = "royalblue4", size = 8,   
 face = "italic", hjust =0.5, vjust = 1))  
  
B <-  
ggplot(ASdata, aes(x = SedentaryMinutes, y = TotalMinutesAsleep)) +  
 geom\_point(alpha = 0.5) +  
 geom\_smooth(method='lm',  
 se = TRUE,  
 colour = 'coral') +  
 geom\_smooth(method='loess',  
 se = FALSE,  
 colour = 'royalblue') +  
 labs(title = "The relationship between SedentaryMinutes and TotalMinutesAsleep",  
 x = "SedentaryMinutes",  
 y = "TotalMinutesAsleep") +  
 theme(plot.title = element\_text(color = "royalblue4", face = "bold",   
 size = 10, hjust =0.5),  
 axis.title.x = element\_text(color = "royalblue4", size = 8,   
 face = "italic", hjust =0.5, vjust = 1),  
 axis.title.y = element\_text(color = "royalblue4", size = 8,   
 face = "italic", hjust =0.5, vjust = 1))  
  
plot\_grid(A, B)

## `geom\_smooth()` using formula 'y ~ x'  
## `geom\_smooth()` using formula 'y ~ x'  
## `geom\_smooth()` using formula 'y ~ x'  
## `geom\_smooth()` using formula 'y ~ x'



• Insights from the graph:  
On one hand, we can see that, the more one stays in bed, the more they can sleep.  
On the other hand, it is the best when people sit for about 500 minute per day, which is around 8 hours. If they sit more than 8 hours, the more they sit, the less time they have to sleep.

• Suggestion according to the insights:  
If the customer wants to improve their sleep, they are encouraged to stay in bed for a longer time and do not sit for more than 8 hours. If they sit less than 8 hours, they should also sit longer. For better calculation, we still make a model and do predictions and evaluate our model’s performance.

##### 5.4.1 Modelling the relationship and evaluating the model

lm2 <- lm(TotalMinutesAsleep ~ TotalTimeInBed + SedentaryMinutes, data = ASdata)  
summary(lm2)

##   
## Call:  
## lm(formula = TotalMinutesAsleep ~ TotalTimeInBed + SedentaryMinutes,   
## data = ASdata)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -256.612 -5.984 11.199 21.980 64.432   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 52.20187 19.56097 2.669 0.00792 \*\*   
## TotalTimeInBed 0.84357 0.02148 39.268 < 2e-16 \*\*\*  
## SedentaryMinutes -0.02779 0.01648 -1.687 0.09243 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 43.43 on 407 degrees of freedom  
## Multiple R-squared: 0.8666, Adjusted R-squared: 0.866   
## F-statistic: 1322 on 2 and 407 DF, p-value: < 2.2e-16

pred.lm2 <- predict(lm2, newdata = ASdata)   
summary(pred.lm2)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 71.88 372.67 422.74 419.17 477.29 856.81

(MAE.lm2 <- mean(abs(ASdata$TotalMinutesAsleep - pred.lm2)))

## [1] 25.89865

The MAE value is around 25, which means our model’s performance is not bad.

### 6. Act

#### 6.1 The general trends

Nowdays people do little sports and they sit for too long. As a result of that, they do not burn so much calories everyday and they might be overweighted. The sleep quality is also not perfect. All of those mentioned above could result in pysically and mentally dieseases.

#### 6.2 To apply to Bellabeat customers

The targeted customers of Bellabeat is those who lightly active and a little bit overweighted women. These kind of customer’s purpose of maintaining of using fitness services could be divided into two aspects: lose weight and improve the sleep quality. Accoding to our data, we draw the following conclusion:

• People need to reduce the sedentary time from 16 hours to 10 hours per day.

• They should also walk more, the best would be that the customers could walk for at least 5 miles every day.

• If they could reach the two targets above, they could then burn 2500 to 3000 calories every day and get a better BMI.

• To improve their sleep quality, they could try to stay on bed for a longer time and sit for 8 hours every day.

#### 6.3 Marketing strategy for now:

• Bellabeat could set some healthy thresholds for the customer. For example, they could set the daily sit hour as 10 hours, daily walk miles as 5 miles, daily time spend on bed is 500 mintues, daily calories to burn is 2500.  
With those threshold, people who wants to generally improve their health can have a better control of their daily activities.

• Bellabeat could use our models to develop a calculate function. Because we have already divided our cutomers into two different categories: the one who wants to lose weight and the one who wants to improve the sleep.  
According to the different type of cutomers, bellabeat could set a function which enable customers to chooes their purposes to use fitness app. Different customers could set the healthy threshold by themselves.  
For example, someone wants to lose weight and they just set the targeted value of the weight. Accoding to the calculate function, customers could check how much they need to walk and sit. For the customers who want to improve their sleep, they could also enter a value of how long they want to sleep more. According to the calculate function, the time to sit and to walk can be also notified to the customer.  
In this way, the fitness app could help the cutomers to acheive their goals more efficiently.

• Through our analysis we found out that during 17-19:00, the intensities of acitivites is the highest, so Bellbeat could set a notification at 17 to enable customers to check, if they have already reach the basic threshold or the threshold they set by themselves every day. If not, they could still do some more to achieve the goals for the day.

• For the customers who achieve their goals, Bellabeat could develop a reward system. For example, customers get a longer Bellabeat membership if they achieve their goal every day for a week. In this way, it is not only easier to motivate people, but also help people to get to know the brand Bellabeat and their products better.

#### 6.4 Marketing strategy for the next step:

Because we have already know the targeted customers are lightly active women who sit more. So it would be wise if Bellbeat collect some information and data about these kind of customers. Bellabeat could analyse their habits and they daily routine, for example, do they watch more TV or computer. What kind of place do they usually go except for work. Do they use Youtube more or Facebook more. In this way, Bellabeat could consider where they should put advertisement to improve their KPI and reduce the ad cost.