Bellabeat Data Analysis

Ana Visanu

2023-11-20

## Business task

Review smart device usage data for selecting trends that could be applied to Bellabeat customers to generate sales growth.

## A description of all data sources used

Data is stored on CC0: Public Domain (Kaggle – an online platform of data scientists and machine learning engineers). Folder contains 18 files, out of which 15 files are organized in a long format and 3 are organized in wide format.

Please note the following data issues/limitations:

* Though this is data is public and the provider states that survey participants agreed on sharing their data, the fact that this is a third-party source raises doubts referring to reliability of data. It would be highly recommended to use alternative data (preferably collected directly by Bellabeat or bought from Amazon Mechanical Turk).
* Data contains information referring to only 33 users for a 30-day period. Considering such a limited sample, it might not be sufficient for ensuring an appropriate margin of error, meaning that the conclusions may not be representative for all the smart tracker users.
* Based on data description, it is not specified that information was randomly selected, meaning that this source may be subject to bias.
* Using this data may lead to some licencing and privacy issues. Based on Kaggle terms of use (<https://www.kaggle.com/terms>), one could use Kaggle data for his/her own internal, personal, non-commercial use, and not on behalf of or for the benefit of any third party. If Bellabeat has its own Kaggle account and will not share the data with any third-party, no issues arise. However, if the analyst is required to use own account for downloading the dataset, it raises issues regarding distribution of the dataset to other parties. To prevent these issues, Bellabeat should create it s account on Kaggle or write to Kaggle to receive approval for data usage. Further investigation of data source leads to another website (<https://zenodo.org/records/53894#.X9oeh3Uzaao>) as provenience. Probably, Bellabeat would need to contact zenovo.org as well in order to avoid any legal issues.

## Documentation of cleaning/manipulation of data

Based on initial data assessment, it was noted that different files contain similar data. For instance, file “dailyActivity\_merged” contains all the information provided in files “dailyCalories\_merged”, “dailyIntensities\_merged” and “dailySteps\_merged”. Also, same information about total steps is available per hour-basis, per daily-basis, as well as in wide and long format. Additionally, data on sleep patterns and weight tracking is available only for a portion from 33 users. If in case of sleep data, they are at least available for around 72% of users (24 out of 33 users), than in case of weight data, it is available for around 24% (8 of 33 users). Based on that, the weight data was not included in analysis.

In order to avoid any duplicated or irrelevant information, the following files were considered for analysis: \* “dailyActivity\_merged” file; \* “sleepDay\_merged” file; \* “heartrate\_seconds\_merged” file.

### a) “dailyActivity\_merged” file

The needed packages were loaded:

library(readr)  
library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.3 ✔ purrr 1.0.2  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.4.4 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.3 ✔ tidyr 1.3.0  
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(ggplot2)  
library(dplyr)  
library(lubridate)

Afterwards, the needed file was imported in R for further assessment.

daily\_activity\_df <- read\_csv("Bellabeat/dailyActivity\_merged.csv")

## Rows: 940 Columns: 15  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (1): ActivityDate  
## dbl (14): Id, TotalSteps, TotalDistance, TrackerDistance, LoggedActivitiesDi...  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

View(daily\_activity\_df)

Structure of data was checked.

glimpse(daily\_activity\_df)

## Rows: 940  
## Columns: 15  
## $ Id <dbl> 1503960366, 1503960366, 1503960366, 150396036…  
## $ ActivityDate <chr> "4/12/2016", "4/13/2016", "4/14/2016", "4/15/…  
## $ TotalSteps <dbl> 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 <dbl> 25, 21, 30, 29, 36, 38, 42, 50, 28, 19, 66, 4…  
## $ FairlyActiveMinutes <dbl> 13, 19, 11, 34, 10, 20, 16, 31, 12, 8, 27, 21…  
## $ LightlyActiveMinutes <dbl> 328, 217, 181, 209, 221, 164, 233, 264, 205, …  
## $ SedentaryMinutes <dbl> 728, 776, 1218, 726, 773, 539, 1149, 775, 818…  
## $ Calories <dbl> 1985, 1797, 1776, 1745, 1863, 1728, 1921, 203…

head(daily\_activity\_df)

## # A tibble: 6 × 15  
## Id ActivityDate TotalSteps TotalDistance TrackerDistance  
## <dbl> <chr> <dbl> <dbl> <dbl>  
## 1 1503960366 4/12/2016 13162 8.5 8.5   
## 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  
## # ℹ 10 more variables: LoggedActivitiesDistance <dbl>,  
## # VeryActiveDistance <dbl>, ModeratelyActiveDistance <dbl>,  
## # LightActiveDistance <dbl>, SedentaryActiveDistance <dbl>,  
## # VeryActiveMinutes <dbl>, FairlyActiveMinutes <dbl>,  
## # LightlyActiveMinutes <dbl>, SedentaryMinutes <dbl>, Calories <dbl>

str(daily\_activity\_df)

## spc\_tbl\_ [940 × 15] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ Id : num [1:940] 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...  
## $ ActivityDate : chr [1:940] "4/12/2016" "4/13/2016" "4/14/2016" "4/15/2016" ...  
## $ TotalSteps : num [1:940] 13162 10735 10460 9762 12669 ...  
## $ TotalDistance : num [1:940] 8.5 6.97 6.74 6.28 8.16 ...  
## $ TrackerDistance : num [1:940] 8.5 6.97 6.74 6.28 8.16 ...  
## $ LoggedActivitiesDistance: num [1:940] 0 0 0 0 0 0 0 0 0 0 ...  
## $ VeryActiveDistance : num [1:940] 1.88 1.57 2.44 2.14 2.71 ...  
## $ ModeratelyActiveDistance: num [1:940] 0.55 0.69 0.4 1.26 0.41 ...  
## $ LightActiveDistance : num [1:940] 6.06 4.71 3.91 2.83 5.04 ...  
## $ SedentaryActiveDistance : num [1:940] 0 0 0 0 0 0 0 0 0 0 ...  
## $ VeryActiveMinutes : num [1:940] 25 21 30 29 36 38 42 50 28 19 ...  
## $ FairlyActiveMinutes : num [1:940] 13 19 11 34 10 20 16 31 12 8 ...  
## $ LightlyActiveMinutes : num [1:940] 328 217 181 209 221 164 233 264 205 211 ...  
## $ SedentaryMinutes : num [1:940] 728 776 1218 726 773 ...  
## $ Calories : num [1:940] 1985 1797 1776 1745 1863 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. Id = col\_double(),  
## .. ActivityDate = col\_character(),  
## .. TotalSteps = col\_double(),  
## .. TotalDistance = col\_double(),  
## .. TrackerDistance = col\_double(),  
## .. LoggedActivitiesDistance = col\_double(),  
## .. VeryActiveDistance = col\_double(),  
## .. ModeratelyActiveDistance = col\_double(),  
## .. LightActiveDistance = col\_double(),  
## .. SedentaryActiveDistance = col\_double(),  
## .. VeryActiveMinutes = col\_double(),  
## .. FairlyActiveMinutes = col\_double(),  
## .. LightlyActiveMinutes = col\_double(),  
## .. SedentaryMinutes = col\_double(),  
## .. Calories = col\_double()  
## .. )  
## - attr(\*, "problems")=<externalptr>

Considering that “ActivityDate” column is of character type, changed format of this column from character to date. Additionally, inserted a new column with the weekday (1 for Monday to 7 for Sunday).

daily\_activity\_df <- daily\_activity\_df %>%   
 mutate(date\_adjusted = mdy(ActivityDate)) %>%   
 mutate(week\_day=wday(date\_adjusted-1))  
View(daily\_activity\_df)

Filtered and sorted data. Noticed that when total steps were null, calories were still counted. As these values are not related to any users’ activities, further assessment will disregard calories where total steps are null.

filtered\_df <- filter(daily\_activity\_df,TotalSteps ==0)  
arrange(filtered\_df,Calories,by\_group=FALSE)

## # A tibble: 77 × 17  
## Id ActivityDate TotalSteps TotalDistance TrackerDistance  
## <dbl> <chr> <dbl> <dbl> <dbl>  
## 1 1503960366 5/12/2016 0 0 0  
## 2 6290855005 5/10/2016 0 0 0  
## 3 8253242879 4/30/2016 0 0 0  
## 4 8583815059 5/12/2016 0 0 0  
## 5 8792009665 5/10/2016 0 0 0  
## 6 7007744171 5/7/2016 0 0 0  
## 7 1844505072 5/12/2016 0 0 0  
## 8 1844505072 4/24/2016 0 0 0  
## 9 1844505072 4/25/2016 0 0 0  
## 10 1844505072 4/26/2016 0 0 0  
## # ℹ 67 more rows  
## # ℹ 12 more variables: LoggedActivitiesDistance <dbl>,  
## # VeryActiveDistance <dbl>, ModeratelyActiveDistance <dbl>,  
## # LightActiveDistance <dbl>, SedentaryActiveDistance <dbl>,  
## # VeryActiveMinutes <dbl>, FairlyActiveMinutes <dbl>,  
## # LightlyActiveMinutes <dbl>, SedentaryMinutes <dbl>, Calories <dbl>,  
## # date\_adjusted <date>, week\_day <dbl>

View(filtered\_df)

### b) “heartrate\_seconds\_merged” file

The needed file was imported in R for further assessment.

heartrate\_df <- read\_csv("Bellabeat/heartrate\_seconds\_merged.csv")

## Rows: 2483658 Columns: 3  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (1): Time  
## dbl (2): Id, Value  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

View(heartrate\_df)

Structure of data was checked.

glimpse(heartrate\_df)

## 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 <dbl> 97, 102, 105, 103, 101, 95, 91, 93, 94, 93, 92, 89, 83, 61, 60, …

head(heartrate\_df)

## # 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

str(heartrate\_df)

## spc\_tbl\_ [2,483,658 × 3] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ Id : num [1:2483658] 2.02e+09 2.02e+09 2.02e+09 2.02e+09 2.02e+09 ...  
## $ Time : chr [1:2483658] "4/12/2016 7:21:00 AM" "4/12/2016 7:21:05 AM" "4/12/2016 7:21:10 AM" "4/12/2016 7:21:20 AM" ...  
## $ Value: num [1:2483658] 97 102 105 103 101 95 91 93 94 93 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. Id = col\_double(),  
## .. Time = col\_character(),  
## .. Value = col\_double()  
## .. )  
## - attr(\*, "problems")=<externalptr>

Column “Time” was split into date and time

heartrate\_df <- heartrate\_df %>%   
 separate(Time, into = c("Date","Hour"), sep = " ")

## Warning: Expected 2 pieces. Additional pieces discarded in 2483658 rows [1, 2, 3, 4, 5,  
## 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].

View(heartrate\_df)

The format of column containing date of activity (column “Date”) from character to date type.

heartrate\_df <- heartrate\_df %>%   
 mutate(Date = mdy(Date))  
View(heartrate\_df)

### c) “sleepDay\_merged” file

The needed file was imported in R for further assessment.

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

## Rows: 413 Columns: 5  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (1): SleepDay  
## dbl (4): Id, TotalSleepRecords, TotalMinutesAsleep, TotalTimeInBed  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

View(sleep\_df)

Structure of data was checked.

glimpse(sleep\_df)

## 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 <dbl> 1, 2, 1, 2, 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, 2…  
## $ TotalTimeInBed <dbl> 346, 407, 442, 367, 712, 320, 377, 364, 384, 449, 3…

head(sleep\_df)

## # A tibble: 6 × 5  
## Id SleepDay TotalSleepRecords TotalMinutesAsleep TotalTimeInBed  
## <dbl> <chr> <dbl> <dbl> <dbl>  
## 1 1503960366 4/12/2016 12:0… 1 327 346  
## 2 1503960366 4/13/2016 12:0… 2 384 407  
## 3 1503960366 4/15/2016 12:0… 1 412 442  
## 4 1503960366 4/16/2016 12:0… 2 340 367  
## 5 1503960366 4/17/2016 12:0… 1 700 712  
## 6 1503960366 4/19/2016 12:0… 1 304 320

str(sleep\_df)

## spc\_tbl\_ [413 × 5] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ Id : num [1:413] 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...  
## $ SleepDay : chr [1:413] "4/12/2016 12:00:00 AM" "4/13/2016 12:00:00 AM" "4/15/2016 12:00:00 AM" "4/16/2016 12:00:00 AM" ...  
## $ TotalSleepRecords : num [1:413] 1 2 1 2 1 1 1 1 1 1 ...  
## $ TotalMinutesAsleep: num [1:413] 327 384 412 340 700 304 360 325 361 430 ...  
## $ TotalTimeInBed : num [1:413] 346 407 442 367 712 320 377 364 384 449 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. Id = col\_double(),  
## .. SleepDay = col\_character(),  
## .. TotalSleepRecords = col\_double(),  
## .. TotalMinutesAsleep = col\_double(),  
## .. TotalTimeInBed = col\_double()  
## .. )  
## - attr(\*, "problems")=<externalptr>

Column “Time” was split into date and time

sleep\_df <- sleep\_df %>%   
 separate(SleepDay, into = c("Date\_sleep","Hour"), sep = " ")

## Warning: Expected 2 pieces. Additional pieces discarded in 413 rows [1, 2, 3, 4, 5, 6,  
## 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].

View(sleep\_df)

The format of column containing date of activity (column “Date”) from character to date type.

sleep\_df <- sleep\_df %>%   
 mutate(Date\_sleep = mdy(Date\_sleep))  
View(sleep\_df)

## Summary of Analysis with supporting visualizations and key findings

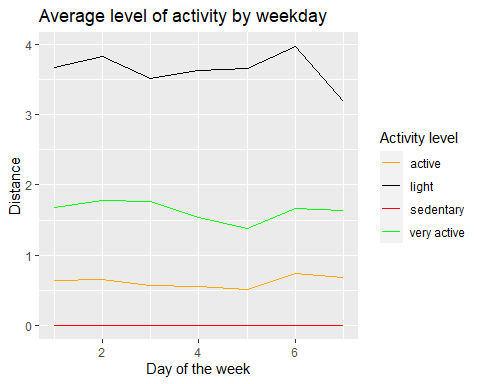
### a) “dailyActivity\_merged” file

Types of activities (very active, active, light active and sedentary) were summarized by weekday.

activity\_type\_by\_weekday <- daily\_activity\_df %>%  
 filter(TotalSteps !=0) %>%   
 group\_by(week\_day) %>%   
 summarize(very\_active=mean(VeryActiveDistance),active=mean(ModeratelyActiveDistance),  
 light\_active=mean(LightActiveDistance),sedentary=mean(SedentaryActiveDistance)) %>%   
 bind\_rows(summarise(., across(where(is.numeric) &!week\_day, mean)))   
View(activity\_type\_by\_weekday)

Afterwards, activity levels by weekday were plotted in the following chart:

## Warning: Removed 1 row containing missing values (`geom\_line()`).  
## Removed 1 row containing missing values (`geom\_line()`).  
## Removed 1 row containing missing values (`geom\_line()`).  
## Removed 1 row containing missing values (`geom\_line()`).

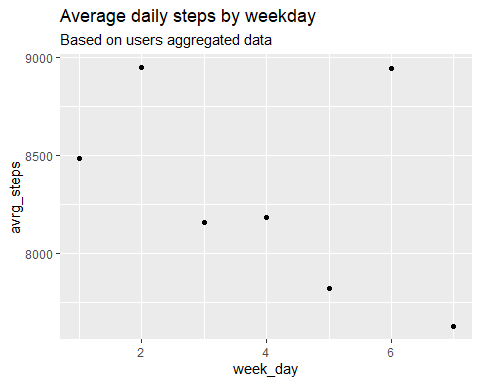


**Based on analysis of activity levels performed by users, consumers can be split into 4 categories: very active, active, light active and sedentary. Considering different level of involvement and involvement in wellness, each type of users requires specific marketing strategies. Also, it is noticed that the majority of users perform a light activity level. Second most popular pattern is the very active movement. However, it is around two times less frequent than light activity level. Considering their frequency, consumer categories “very active” and “light active” are to be prioritized.**

Afterwards, average daily steps were summarized by weekday.

activity\_by\_weekday <- daily\_activity\_df %>%  
 filter(TotalSteps !=0) %>%   
 group\_by(week\_day) %>%   
 summarize(avrg\_steps=mean(TotalSteps))  
View(activity\_by\_weekday)

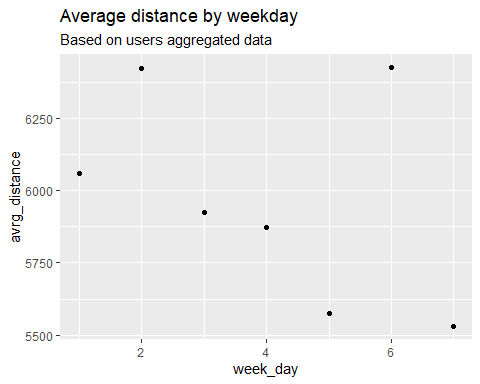
Average daily steps were plotted in the following chart:



We also summarized average distance by weekday.

distance\_by\_weekday <- daily\_activity\_df %>%   
 filter(TotalSteps !=0) %>%   
 group\_by(week\_day) %>%   
 summarize(avrg\_distance=mean(TotalDistance)\*1000)  
View(distance\_by\_weekday)

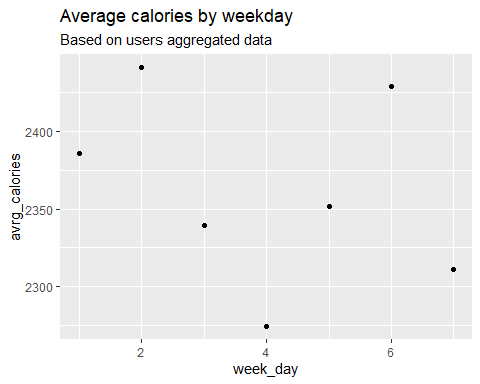
Average daily distance were plotted in the following chart:



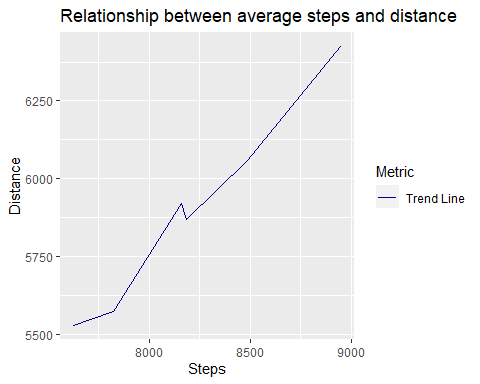
We summarized average calories by weekday.

calories\_by\_weekday <- daily\_activity\_df %>%   
 filter(TotalSteps !=0) %>%   
 group\_by(week\_day) %>%   
 summarize(avrg\_calories=mean(Calories))  
View(calories\_by\_weekday)

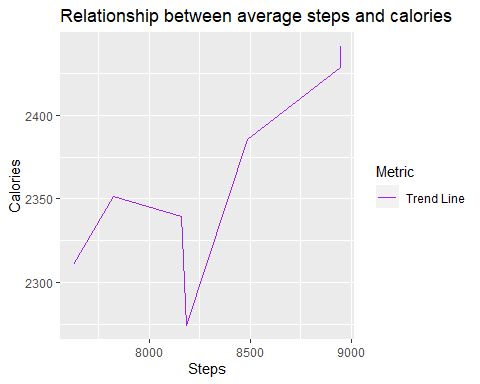
Average daily calories were plotted in the following chart:



**Based on assessment of weekly changes of total steps, distance and calories, we can notice that the highest users’ level of activity occurs on Tuesday and Saturday. On the other hand, the lowest level of activity is noted on Thursday and Sunday. Marketing materials sent on Tuesday and Saturday might be more appealing.**

When plotting daily steps and distance, it could be noticed that both variables have a very strong positive correlation. 

**However, when plotting daily steps and calories, it could be noticed that there is no continuous pattern of relationship between total steps and calories. For number of steps higher than 8500, both calories and total steps tend to positively correlate.**



## Summary of Analysis with supporting visualizations and key findings

### b) “heartrate\_seconds\_merged” file

Heartbeat data was summarized as average, minimal and maximum level per user

summary\_heartrate <- heartrate\_df %>%  
 group\_by(Id) %>%   
 summarize(average=mean(Value),max=max(Value),  
 min=min(Value))  
View(summary\_heartrate)

**It could be noticed that there are cases when user’s heart rates have extreme values. Presumably, extreme high values occur in times of intense activity level, while extremely low values occur during sleep.**

We summarized types of activity (per df “activity\_type\_by\_id”) by Id

activity\_type\_by\_id <- daily\_activity\_df %>%  
 filter(TotalSteps !=0) %>%   
 group\_by(Id) %>%   
 summarize(very\_active=mean(VeryActiveDistance),active=mean(ModeratelyActiveDistance),  
 light\_active=mean(LightActiveDistance),sedentary=mean(SedentaryActiveDistance)) %>%   
 bind\_rows(summarise(., across(where(is.numeric), mean)))   
View(activity\_type\_by\_id)

Combined information about heart rate and types of activity by Id under a single data frame

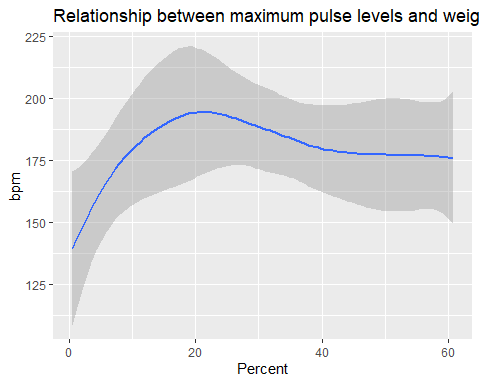
summary\_max <- summary\_heartrate %>% inner\_join(activity\_type\_by\_id,   
 by="Id")  
View(summary\_max)

Computed share of active movements (very active and active) in total daily activity

summary\_max <- summary\_max %>%   
 mutate(weight\_of\_activity = (very\_active+active)/(very\_active+active+ light\_active+sedentary)\*100)  
View(summary\_max)

We plotted max pulse and most active minutes per day

## `geom\_smooth()` using method = 'loess' and formula = 'y ~ x'



**The relationship between maximal pulse and most active minutes per day is not linear. Increase in number of active minutes per day from 0 to 20 minutes per day leads to a positive increase in heart rate. However further rise in daily activity leads to stabilization of the heart rate.**

### c) “sleepDay\_merged” file

We summarized sleeping patterns by Id

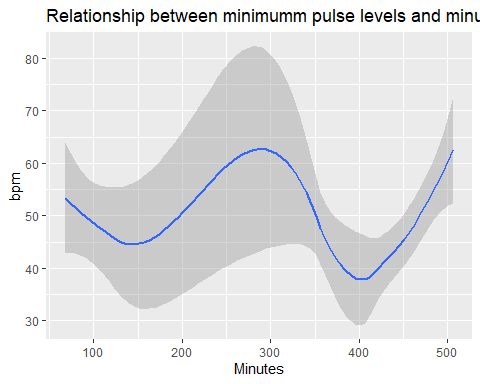
summary\_sleep <- sleep\_df %>%  
 group\_by(Id) %>%   
 summarize(average\_asleep=mean(TotalMinutesAsleep),average\_in\_bed=mean(TotalTimeInBed))  
View(summary\_sleep)

Combined information about heart rate and sleep pattern

summary\_min <- summary\_heartrate %>% inner\_join(summary\_sleep,   
 by="Id")  
View(summary\_min)

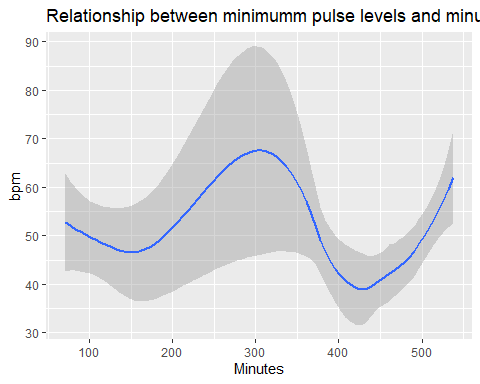
We plotted minimal pulse and sleep minutes per day

## `geom\_smooth()` using method = 'loess' and formula = 'y ~ x'



We plotted minimal pulse and minutes in bed per day

## `geom\_smooth()` using method = 'loess' and formula = 'y ~ x'



**Based on above charts, the relationship between sleeping pattern and heart rate is not linear, but seems determined by sleeping phases. However charts illustrate that lowest heart rates occur indeed during deep sleep.**

**Moreover, tracking extreme heart rates could help identify and send emergency alerts. For instance, a heart rate higher than 190 bmp even during intense workout or a heart rate below 60 outside the sleep phase would require to issue and alert to user for him/her to seek doctor’s consultation.**

## Recommendations

Based on analysis performed, the following recommendations would be given to increase company’s revenue:

* Split Heartbeat’s customers by level of activity (very active, active, light active and sedentary) and create separate marketing strategies for each type of customers. For instance, very active customers could receive advertising focused on additional features of Bellabeat app and Bellabeat subscription that would help them to improve their performance, while light active or sedentary customers could receive advertising focused on the importance of sport for well-being.
* The timing of displaying adds can be set for days of maximal physical activity: Tuesday and Saturday. Sending adds on Friday and Sunday could be reduced to minimal levels as the majority of customers have lowest interest in sports and healthy lifestyle.
* Based on analysis of calories per users, the tracking system of the competitor seems to have some drawbacks (i.e. computing calories when no movement is made, not track additional calories when further activity is performed). Focusing on key advantages of Heartbeat Leaf, Time and Heartbeat app in contrast to opponent’s issues could have an important role in attracting new consumers.
* In order to promote Bellabeat app, Bellabeat could share some catchy info such as: “Do you know that your heart beat could decline less than 40 bpm during sleep?”. Also, Bellabeat could further consider the possibility to add alerts when heart beat reaches extremely low or high values. This additional feature could also be advertised as a specially designed customer care.