Bellabeat Case Study

Swathy Ajaya Kumar

2022-09-30

*This Case Study is the Capstone Project for the Google Data Analytics Course. The ideas and recommendations presented here in no way represent the ideas and belief of Google or Bellabeat. This project is completed for learning purposes only and may be inaccurate. The views expressed here are my own.*

***Assumption:*** *I am a junior data analyst working on the marketing analyst team at Bellabeat, a high-tech manufacturer of health-focused products for women. I have been asked to focus on one of Bellabeat’s products and analyze smart device data to gain insight into how consumers are using their smart devices. The insights will help guide marketing strategy for the company. The result of this analysis is assumed to be presented to the Bellabeat executive team along with high-level recommendations for Bellabeat’s marketing strategy.*

### About Bellabeat

Founded in 2013 by Urška Sršen and Sando Mur, Bellabeat is high-tech company that manufactures beautifully designed health-focused smart products for women. Bellabeat products empower women with knowledge about their own health and habits by collecting data on activity, sleep, stress,and reproductive health.

### Our Products

Our wearables and accompanying products monitor biometric and lifestyle data to help women better understand how their bodies work and make healthier choices.

1. **Bellabeat app:** Provides users with health data related to their activity, sleep, stress, menstrual cycle, and mindfulness habits. This app connects to our line of smart wellness products mentioned below.
2. **Leaf:** Wellness trackers that can be worn as a bracelet, necklace, or clip.
3. **Time:** Wellness SmartWatch is timepiece with smart technology to track user activity, sleep, and stress.
4. **Spring:** Smart Water Bottle that tracks daily water intake using smart technology to ensure that our users are appropriately hydrated throughout the day.
5. **Bellabeat membership:** We offer subscription-based membership program for users which gives users 24/7 access to fully personalized guidance on nutrition, activity, sleep, health and beauty, and mindfulness based on their lifestyle and goals.

### SWOT Analysis - Analysing Strengths, Weaknesses, Opportunities and Threats

#### Strengths

1. **Product Line-up:** Bellabeat’s versatile product line-up includes wearable jewellery trackers, smart water bottle and smartwatch which can all be linked to the an app which can actively track activity,sleep, stress, menstrual cycle, and mindfulness habits.
2. **Beautiful Designs** Our products are beautifully and elegantly designed by women, for women.
3. **Affordable Price Point** Our products are affordably priced compared to competitor products. Smart watches are priced at $149.00 USD + tax, and come with free 3 months subscription to personalized wellness coaching.

#### Opportunities

1. **Introduce wireless charging** We could modify our products to be compatible with wireless charging. We could sell wireless charging pods which are compatible with all our products.
2. **Customization** Our products are all elegantly designed. We could add more funky, sporty and trendy looks to accomodate the taste of more women. Not all women prefer minimalistic designs.

#### Weaknesses

1. **Replaceable batteries** Bellabeat products currently use long-lasting, replaceable batteries. They last about 3-6 months. Disposable batteries contribute to e-waste.
2. **Limited options for customization** There aren’t many options to choose from while selecting the style of products, especially the Leaf line-up. This might turn women away from buying our products

#### Threats

1. **Data Safety** On the wake of Roe v Wade being overturned, many women are worried if their reproductive data would be shared with the authorities. Therefore, many women are hesitant to track period data. Period data is essential for our personalized wellness program. We have enabled Private Key Encryption (AES-256) security feature for our mobile app to protect women’s reproductive information. Bellabeat must find ways to protect women’s rights in changing political landscape.
2. **Competitor Threat** Bellabeat face strict competition from Fitbit, Samsung and Apple Watch. These products have economies of scale and have considerable market share. We must consistently push our Unique Selling Point (wellness products for women) to capture more of ther market share.

## Exploring Growth Opportunities

## Business Task

Urška Sršen, cofounder and Chief Creative Officer of Bellabeat is the *primary stakeholder* of this case study. Through this study, we will analyze smart device fitness data which will help unlock new growth opportunities for Bellabeat.

## Preparing Data

We will use [FitBit Fitness Tracker Data](https://www.kaggle.com/datasets/arashnic/fitbit), which is a Kaggle data set that contains personal fitness tracker data from thirty Fitbit users. All users have consented to the submission of data . The data set includes output for physical activity, heart rate, and sleep monitoring, daily activity, steps, and heart rate. This can be used to explore users’ habits.

### Install Packages

We are using R to analyse this dataset. The following summarizes the packages that are installed to help us analyze user data.

install.packages("tidyverse")

## Installing package into '/cloud/lib/x86\_64-pc-linux-gnu-library/4.2'  
## (as 'lib' is unspecified)

install.packages("dplyr")

## Installing package into '/cloud/lib/x86\_64-pc-linux-gnu-library/4.2'  
## (as 'lib' is unspecified)

install.packages("skimr")

## Installing package into '/cloud/lib/x86\_64-pc-linux-gnu-library/4.2'  
## (as 'lib' is unspecified)

install.packages("here")

## Installing package into '/cloud/lib/x86\_64-pc-linux-gnu-library/4.2'  
## (as 'lib' is unspecified)

install.packages("janitor")

## Installing package into '/cloud/lib/x86\_64-pc-linux-gnu-library/4.2'  
## (as 'lib' is unspecified)

install.packages("tidyr")

## Installing package into '/cloud/lib/x86\_64-pc-linux-gnu-library/4.2'  
## (as 'lib' is unspecified)

install.packages("ggplot2")

## Installing package into '/cloud/lib/x86\_64-pc-linux-gnu-library/4.2'  
## (as 'lib' is unspecified)

install.packages("lubridate")

## Installing package into '/cloud/lib/x86\_64-pc-linux-gnu-library/4.2'  
## (as 'lib' is unspecified)

library("tidyverse")

## ── Attaching packages  
## ───────────────────────────────────────  
## tidyverse 1.3.2 ──

## ✔ ggplot2 3.4.0 ✔ purrr 0.3.4   
## ✔ tibble 3.1.8 ✔ dplyr 1.0.10  
## ✔ tidyr 1.2.1 ✔ stringr 1.4.1   
## ✔ readr 2.1.2 ✔ forcats 0.5.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library("dplyr")  
library("skimr")  
library("tidyr")  
library("ggplot2")  
library("lubridate")

## Loading required package: timechange  
##   
## Attaching package: 'lubridate'  
##   
## The following objects are masked from 'package:base':  
##   
## date, intersect, setdiff, union

library("here")

## here() starts at /cloud/project

library("janitor")

##   
## Attaching package: 'janitor'  
##   
## The following objects are masked from 'package:stats':  
##   
## chisq.test, fisher.test

## Uploading Datasets

dailyActivity <- read\_csv("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.

dailysleep <- read\_csv("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.

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

## Rows: 67 Columns: 8  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (1): Date  
## dbl (6): Id, WeightKg, WeightPounds, Fat, BMI, LogId  
## lgl (1): IsManualReport  
##   
## 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.

#### Cleaning up data

##### Removing empty rows and columns

remove\_empty(dailyActivity, which = c("rows", "cols"))

## # A tibble: 940 × 15  
## Id Activity…¹ Total…² Total…³ Track…⁴ Logge…⁵ VeryA…⁶ Moder…⁷ Light…⁸  
## <dbl> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1503960366 4/12/2016 13162 8.5 8.5 0 1.88 0.550 6.06  
## 2 1503960366 4/13/2016 10735 6.97 6.97 0 1.57 0.690 4.71  
## 3 1503960366 4/14/2016 10460 6.74 6.74 0 2.44 0.400 3.91  
## 4 1503960366 4/15/2016 9762 6.28 6.28 0 2.14 1.26 2.83  
## 5 1503960366 4/16/2016 12669 8.16 8.16 0 2.71 0.410 5.04  
## 6 1503960366 4/17/2016 9705 6.48 6.48 0 3.19 0.780 2.51  
## 7 1503960366 4/18/2016 13019 8.59 8.59 0 3.25 0.640 4.71  
## 8 1503960366 4/19/2016 15506 9.88 9.88 0 3.53 1.32 5.03  
## 9 1503960366 4/20/2016 10544 6.68 6.68 0 1.96 0.480 4.24  
## 10 1503960366 4/21/2016 9819 6.34 6.34 0 1.34 0.350 4.65  
## # … with 930 more rows, 6 more variables: SedentaryActiveDistance <dbl>,  
## # VeryActiveMinutes <dbl>, FairlyActiveMinutes <dbl>,  
## # LightlyActiveMinutes <dbl>, SedentaryMinutes <dbl>, Calories <dbl>, and  
## # abbreviated variable names ¹​ActivityDate, ²​TotalSteps, ³​TotalDistance,  
## # ⁴​TrackerDistance, ⁵​LoggedActivitiesDistance, ⁶​VeryActiveDistance,  
## # ⁷​ModeratelyActiveDistance, ⁸​LightActiveDistance

remove\_empty(dailysleep, which = c("rows", "cols"))

## # A tibble: 413 × 5  
## Id SleepDay TotalSleepRecords TotalMinutesAsleep Total…¹  
## <dbl> <chr> <dbl> <dbl> <dbl>  
## 1 1503960366 4/12/2016 12:00:00 AM 1 327 346  
## 2 1503960366 4/13/2016 12:00:00 AM 2 384 407  
## 3 1503960366 4/15/2016 12:00:00 AM 1 412 442  
## 4 1503960366 4/16/2016 12:00:00 AM 2 340 367  
## 5 1503960366 4/17/2016 12:00:00 AM 1 700 712  
## 6 1503960366 4/19/2016 12:00:00 AM 1 304 320  
## 7 1503960366 4/20/2016 12:00:00 AM 1 360 377  
## 8 1503960366 4/21/2016 12:00:00 AM 1 325 364  
## 9 1503960366 4/23/2016 12:00:00 AM 1 361 384  
## 10 1503960366 4/24/2016 12:00:00 AM 1 430 449  
## # … with 403 more rows, and abbreviated variable name ¹​TotalTimeInBed

remove\_empty(dailyweight, which = c("rows", "cols"))

## # A tibble: 67 × 8  
## Id Date WeightKg Weigh…¹ Fat BMI IsMan…² LogId  
## <dbl> <chr> <dbl> <dbl> <dbl> <dbl> <lgl> <dbl>  
## 1 1503960366 5/2/2016 11:59:59 PM 52.6 116. 22 22.6 TRUE 1.46e12  
## 2 1503960366 5/3/2016 11:59:59 PM 52.6 116. NA 22.6 TRUE 1.46e12  
## 3 1927972279 4/13/2016 1:08:52 AM 134. 294. NA 47.5 FALSE 1.46e12  
## 4 2873212765 4/21/2016 11:59:59 PM 56.7 125. NA 21.5 TRUE 1.46e12  
## 5 2873212765 5/12/2016 11:59:59 PM 57.3 126. NA 21.7 TRUE 1.46e12  
## 6 4319703577 4/17/2016 11:59:59 PM 72.4 160. 25 27.5 TRUE 1.46e12  
## 7 4319703577 5/4/2016 11:59:59 PM 72.3 159. NA 27.4 TRUE 1.46e12  
## 8 4558609924 4/18/2016 11:59:59 PM 69.7 154. NA 27.2 TRUE 1.46e12  
## 9 4558609924 4/25/2016 11:59:59 PM 70.3 155. NA 27.5 TRUE 1.46e12  
## 10 4558609924 5/1/2016 11:59:59 PM 69.9 154. NA 27.3 TRUE 1.46e12  
## # … with 57 more rows, and abbreviated variable names ¹​WeightPounds,  
## # ²​IsManualReport

##### Check number of users

n\_distinct(dailyActivity$Id)

## [1] 33

n\_distinct(dailysleep$Id)

## [1] 24

n\_distinct(dailyweight$Id)

## [1] 8

There are 33, 24 and 8 distinct user IDs reported for daily activity, sleep and weight log respectively.

##### Check for duplicates

sum(duplicated(dailyActivity))

## [1] 0

sum(duplicated(dailysleep))

## [1] 3

sum(duplicated(dailyweight))

## [1] 0

The Sleep log has 3 duplicate values. We would have to remove them.

dailysleep <- dailysleep %>%   
 distinct() %>%   
 drop\_na()

sum(duplicated(dailysleep))

## [1] 0

##### Standard column names

Using the clean names function to standardize column names.

dailyActivity <- clean\_names(dailyActivity)  
dailysleep <- clean\_names(dailysleep)  
dailyweight <- clean\_names(dailyweight)  
  
head(dailyActivity)

## # A tibble: 6 × 15  
## id activ…¹ total…² total…³ track…⁴ logge…⁵ very\_…⁶ moder…⁷ light…⁸ seden…⁹  
## <dbl> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1.50e9 4/12/2… 13162 8.5 8.5 0 1.88 0.550 6.06 0  
## 2 1.50e9 4/13/2… 10735 6.97 6.97 0 1.57 0.690 4.71 0  
## 3 1.50e9 4/14/2… 10460 6.74 6.74 0 2.44 0.400 3.91 0  
## 4 1.50e9 4/15/2… 9762 6.28 6.28 0 2.14 1.26 2.83 0  
## 5 1.50e9 4/16/2… 12669 8.16 8.16 0 2.71 0.410 5.04 0  
## 6 1.50e9 4/17/2… 9705 6.48 6.48 0 3.19 0.780 2.51 0  
## # … with 5 more variables: very\_active\_minutes <dbl>,  
## # fairly\_active\_minutes <dbl>, lightly\_active\_minutes <dbl>,  
## # sedentary\_minutes <dbl>, calories <dbl>, and abbreviated variable names  
## # ¹​activity\_date, ²​total\_steps, ³​total\_distance, ⁴​tracker\_distance,  
## # ⁵​logged\_activities\_distance, ⁶​very\_active\_distance,  
## # ⁷​moderately\_active\_distance, ⁸​light\_active\_distance,  
## # ⁹​sedentary\_active\_distance

head(dailysleep)

## # A tibble: 6 × 5  
## id sleep\_day total\_sleep\_records total\_minutes\_a…¹ total…²  
## <dbl> <chr> <dbl> <dbl> <dbl>  
## 1 1503960366 4/12/2016 12:00:00 AM 1 327 346  
## 2 1503960366 4/13/2016 12:00:00 AM 2 384 407  
## 3 1503960366 4/15/2016 12:00:00 AM 1 412 442  
## 4 1503960366 4/16/2016 12:00:00 AM 2 340 367  
## 5 1503960366 4/17/2016 12:00:00 AM 1 700 712  
## 6 1503960366 4/19/2016 12:00:00 AM 1 304 320  
## # … with abbreviated variable names ¹​total\_minutes\_asleep, ²​total\_time\_in\_bed

head(dailyweight)

## # A tibble: 6 × 8  
## id date weight\_kg weigh…¹ fat bmi is\_ma…² log\_id  
## <dbl> <chr> <dbl> <dbl> <dbl> <dbl> <lgl> <dbl>  
## 1 1503960366 5/2/2016 11:59:59 PM 52.6 116. 22 22.6 TRUE 1.46e12  
## 2 1503960366 5/3/2016 11:59:59 PM 52.6 116. NA 22.6 TRUE 1.46e12  
## 3 1927972279 4/13/2016 1:08:52 AM 134. 294. NA 47.5 FALSE 1.46e12  
## 4 2873212765 4/21/2016 11:59:59 PM 56.7 125. NA 21.5 TRUE 1.46e12  
## 5 2873212765 5/12/2016 11:59:59 PM 57.3 126. NA 21.7 TRUE 1.46e12  
## 6 4319703577 4/17/2016 11:59:59 PM 72.4 160. 25 27.5 TRUE 1.46e12  
## # … with abbreviated variable names ¹​weight\_pounds, ²​is\_manual\_report

##### Check the structure of each table

str(dailyActivity)

## 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 ...  
## $ activity\_date : chr [1:940] "4/12/2016" "4/13/2016" "4/14/2016" "4/15/2016" ...  
## $ total\_steps : num [1:940] 13162 10735 10460 9762 12669 ...  
## $ total\_distance : num [1:940] 8.5 6.97 6.74 6.28 8.16 ...  
## $ tracker\_distance : num [1:940] 8.5 6.97 6.74 6.28 8.16 ...  
## $ logged\_activities\_distance: num [1:940] 0 0 0 0 0 0 0 0 0 0 ...  
## $ very\_active\_distance : num [1:940] 1.88 1.57 2.44 2.14 2.71 ...  
## $ moderately\_active\_distance: num [1:940] 0.55 0.69 0.4 1.26 0.41 ...  
## $ light\_active\_distance : num [1:940] 6.06 4.71 3.91 2.83 5.04 ...  
## $ sedentary\_active\_distance : num [1:940] 0 0 0 0 0 0 0 0 0 0 ...  
## $ very\_active\_minutes : num [1:940] 25 21 30 29 36 38 42 50 28 19 ...  
## $ fairly\_active\_minutes : num [1:940] 13 19 11 34 10 20 16 31 12 8 ...  
## $ lightly\_active\_minutes : num [1:940] 328 217 181 209 221 164 233 264 205 211 ...  
## $ sedentary\_minutes : 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>

str(dailysleep)

## tibble [410 × 5] (S3: tbl\_df/tbl/data.frame)  
## $ id : num [1:410] 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...  
## $ sleep\_day : chr [1:410] "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" ...  
## $ total\_sleep\_records : num [1:410] 1 2 1 2 1 1 1 1 1 1 ...  
## $ total\_minutes\_asleep: num [1:410] 327 384 412 340 700 304 360 325 361 430 ...  
## $ total\_time\_in\_bed : num [1:410] 346 407 442 367 712 320 377 364 384 449 ...

str(dailyweight)

## spc\_tbl\_ [67 × 8] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ id : num [1:67] 1.50e+09 1.50e+09 1.93e+09 2.87e+09 2.87e+09 ...  
## $ date : chr [1:67] "5/2/2016 11:59:59 PM" "5/3/2016 11:59:59 PM" "4/13/2016 1:08:52 AM" "4/21/2016 11:59:59 PM" ...  
## $ weight\_kg : num [1:67] 52.6 52.6 133.5 56.7 57.3 ...  
## $ weight\_pounds : num [1:67] 116 116 294 125 126 ...  
## $ fat : num [1:67] 22 NA NA NA NA 25 NA NA NA NA ...  
## $ bmi : num [1:67] 22.6 22.6 47.5 21.5 21.7 ...  
## $ is\_manual\_report: logi [1:67] TRUE TRUE FALSE TRUE TRUE TRUE ...  
## $ log\_id : num [1:67] 1.46e+12 1.46e+12 1.46e+12 1.46e+12 1.46e+12 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. Id = col\_double(),  
## .. Date = col\_character(),  
## .. WeightKg = col\_double(),  
## .. WeightPounds = col\_double(),  
## .. Fat = col\_double(),  
## .. BMI = col\_double(),  
## .. IsManualReport = col\_logical(),  
## .. LogId = col\_double()  
## .. )  
## - attr(\*, "problems")=<externalptr>

The datatype of the date column is set to “character”. Changing the datatype to “Date”

dailyActivity$activity\_date <- as.Date(dailyActivity$activity\_date, "%m/%d/%y")  
dailysleep$sleep\_day <- as.Date(dailysleep$sleep\_day, "%m/%d/%y")  
dailyweight$date <- as.Date(dailyweight$date, "%m/%d/%y")  
  
head(dailyActivity)

## # A tibble: 6 × 15  
## id activity\_…¹ total…² total…³ track…⁴ logge…⁵ very\_…⁶ moder…⁷ light…⁸  
## <dbl> <date> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1503960366 2020-04-12 13162 8.5 8.5 0 1.88 0.550 6.06  
## 2 1503960366 2020-04-13 10735 6.97 6.97 0 1.57 0.690 4.71  
## 3 1503960366 2020-04-14 10460 6.74 6.74 0 2.44 0.400 3.91  
## 4 1503960366 2020-04-15 9762 6.28 6.28 0 2.14 1.26 2.83  
## 5 1503960366 2020-04-16 12669 8.16 8.16 0 2.71 0.410 5.04  
## 6 1503960366 2020-04-17 9705 6.48 6.48 0 3.19 0.780 2.51  
## # … with 6 more variables: sedentary\_active\_distance <dbl>,  
## # very\_active\_minutes <dbl>, fairly\_active\_minutes <dbl>,  
## # lightly\_active\_minutes <dbl>, sedentary\_minutes <dbl>, calories <dbl>, and  
## # abbreviated variable names ¹​activity\_date, ²​total\_steps, ³​total\_distance,  
## # ⁴​tracker\_distance, ⁵​logged\_activities\_distance, ⁶​very\_active\_distance,  
## # ⁷​moderately\_active\_distance, ⁸​light\_active\_distance

head(dailyweight)

## # A tibble: 6 × 8  
## id date weight\_kg weight\_pounds fat bmi is\_manual\_…¹ log\_id  
## <dbl> <date> <dbl> <dbl> <dbl> <dbl> <lgl> <dbl>  
## 1 1503960366 2020-05-02 52.6 116. 22 22.6 TRUE 1.46e12  
## 2 1503960366 2020-05-03 52.6 116. NA 22.6 TRUE 1.46e12  
## 3 1927972279 2020-04-13 134. 294. NA 47.5 FALSE 1.46e12  
## 4 2873212765 2020-04-21 56.7 125. NA 21.5 TRUE 1.46e12  
## 5 2873212765 2020-05-12 57.3 126. NA 21.7 TRUE 1.46e12  
## 6 4319703577 2020-04-17 72.4 160. 25 27.5 TRUE 1.46e12  
## # … with abbreviated variable name ¹​is\_manual\_report

head(dailyweight)

## # A tibble: 6 × 8  
## id date weight\_kg weight\_pounds fat bmi is\_manual\_…¹ log\_id  
## <dbl> <date> <dbl> <dbl> <dbl> <dbl> <lgl> <dbl>  
## 1 1503960366 2020-05-02 52.6 116. 22 22.6 TRUE 1.46e12  
## 2 1503960366 2020-05-03 52.6 116. NA 22.6 TRUE 1.46e12  
## 3 1927972279 2020-04-13 134. 294. NA 47.5 FALSE 1.46e12  
## 4 2873212765 2020-04-21 56.7 125. NA 21.5 TRUE 1.46e12  
## 5 2873212765 2020-05-12 57.3 126. NA 21.7 TRUE 1.46e12  
## 6 4319703577 2020-04-17 72.4 160. 25 27.5 TRUE 1.46e12  
## # … with abbreviated variable name ¹​is\_manual\_report

### Cleaning conclusion

We removed empty rows and columns, checked distinct number of users for each table, checked for and removed duplicate values, standardized column names and changed the data type of the date columns of all tables from “character (chr)” to “date”. This makes it easier for us to work with the data.

## Summary for each data frame

Gathering some quick, basic statistical insights to each data frame helps us frame key questions we want answered in this case study.

summary(dailyActivity)

## id activity\_date total\_steps total\_distance   
## Min. :1.504e+09 Min. :2020-04-12 Min. : 0 Min. : 0.000   
## 1st Qu.:2.320e+09 1st Qu.:2020-04-19 1st Qu.: 3790 1st Qu.: 2.620   
## Median :4.445e+09 Median :2020-04-26 Median : 7406 Median : 5.245   
## Mean :4.855e+09 Mean :2020-04-26 Mean : 7638 Mean : 5.490   
## 3rd Qu.:6.962e+09 3rd Qu.:2020-05-04 3rd Qu.:10727 3rd Qu.: 7.713   
## Max. :8.878e+09 Max. :2020-05-12 Max. :36019 Max. :28.030   
## tracker\_distance logged\_activities\_distance very\_active\_distance  
## 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   
## moderately\_active\_distance light\_active\_distance sedentary\_active\_distance  
## 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   
## very\_active\_minutes fairly\_active\_minutes lightly\_active\_minutes  
## Min. : 0.00 Min. : 0.00 Min. : 0.0   
## 1st Qu.: 0.00 1st Qu.: 0.00 1st Qu.:127.0   
## Median : 4.00 Median : 6.00 Median :199.0   
## Mean : 21.16 Mean : 13.56 Mean :192.8   
## 3rd Qu.: 32.00 3rd Qu.: 19.00 3rd Qu.:264.0   
## Max. :210.00 Max. :143.00 Max. :518.0   
## sedentary\_minutes calories   
## Min. : 0.0 Min. : 0   
## 1st Qu.: 729.8 1st Qu.:1828   
## Median :1057.5 Median :2134   
## Mean : 991.2 Mean :2304   
## 3rd Qu.:1229.5 3rd Qu.:2793   
## Max. :1440.0 Max. :4900

summary(dailyweight)

## id date weight\_kg weight\_pounds   
## Min. :1.504e+09 Min. :2020-04-12 Min. : 52.60 Min. :116.0   
## 1st Qu.:6.962e+09 1st Qu.:2020-04-19 1st Qu.: 61.40 1st Qu.:135.4   
## Median :6.962e+09 Median :2020-04-27 Median : 62.50 Median :137.8   
## Mean :7.009e+09 Mean :2020-04-26 Mean : 72.04 Mean :158.8   
## 3rd Qu.:8.878e+09 3rd Qu.:2020-05-04 3rd Qu.: 85.05 3rd Qu.:187.5   
## Max. :8.878e+09 Max. :2020-05-12 Max. :133.50 Max. :294.3   
##   
## fat bmi is\_manual\_report log\_id   
## Min. :22.00 Min. :21.45 Mode :logical Min. :1.460e+12   
## 1st Qu.:22.75 1st Qu.:23.96 FALSE:26 1st Qu.:1.461e+12   
## Median :23.50 Median :24.39 TRUE :41 Median :1.462e+12   
## Mean :23.50 Mean :25.19 Mean :1.462e+12   
## 3rd Qu.:24.25 3rd Qu.:25.56 3rd Qu.:1.462e+12   
## Max. :25.00 Max. :47.54 Max. :1.463e+12   
## NA's :65

summary(dailysleep)

## id sleep\_day total\_sleep\_records  
## Min. :1.504e+09 Min. :2020-04-12 Min. :1.00   
## 1st Qu.:3.977e+09 1st Qu.:2020-04-19 1st Qu.:1.00   
## Median :4.703e+09 Median :2020-04-27 Median :1.00   
## Mean :4.995e+09 Mean :2020-04-26 Mean :1.12   
## 3rd Qu.:6.962e+09 3rd Qu.:2020-05-04 3rd Qu.:1.00   
## Max. :8.792e+09 Max. :2020-05-12 Max. :3.00   
## total\_minutes\_asleep total\_time\_in\_bed  
## Min. : 58.0 Min. : 61.0   
## 1st Qu.:361.0 1st Qu.:403.8   
## Median :432.5 Median :463.0   
## Mean :419.2 Mean :458.5   
## 3rd Qu.:490.0 3rd Qu.:526.0   
## Max. :796.0 Max. :961.0

## Key Questions - Analysing Data

Using available data, we will try to answer the following questions:

1. On what days of the week were users more active? On what days were they sedentary?
2. How much time do users spend in bed compared to actual minutes asleep?
3. What is the relationship between users’ calories burnt and sleep pattern?

### 1. On what days of the week were users more active? On what days were they sedentary?

We have the daily activity data and sleep data of 33 users available to us. First, we have to create a new column titled

activity\_analysis<- dailyActivity %>%   
 mutate(total\_active\_minutes= very\_active\_minutes+fairly\_active\_minutes+lightly\_active\_minutes)  
head(activity\_analysis)

## # A tibble: 6 × 16  
## id activity\_…¹ total…² total…³ track…⁴ logge…⁵ very\_…⁶ moder…⁷ light…⁸  
## <dbl> <date> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1503960366 2020-04-12 13162 8.5 8.5 0 1.88 0.550 6.06  
## 2 1503960366 2020-04-13 10735 6.97 6.97 0 1.57 0.690 4.71  
## 3 1503960366 2020-04-14 10460 6.74 6.74 0 2.44 0.400 3.91  
## 4 1503960366 2020-04-15 9762 6.28 6.28 0 2.14 1.26 2.83  
## 5 1503960366 2020-04-16 12669 8.16 8.16 0 2.71 0.410 5.04  
## 6 1503960366 2020-04-17 9705 6.48 6.48 0 3.19 0.780 2.51  
## # … with 7 more variables: sedentary\_active\_distance <dbl>,  
## # very\_active\_minutes <dbl>, fairly\_active\_minutes <dbl>,  
## # lightly\_active\_minutes <dbl>, sedentary\_minutes <dbl>, calories <dbl>,  
## # total\_active\_minutes <dbl>, and abbreviated variable names ¹​activity\_date,  
## # ²​total\_steps, ³​total\_distance, ⁴​tracker\_distance,  
## # ⁵​logged\_activities\_distance, ⁶​very\_active\_distance,  
## # ⁷​moderately\_active\_distance, ⁸​light\_active\_distance

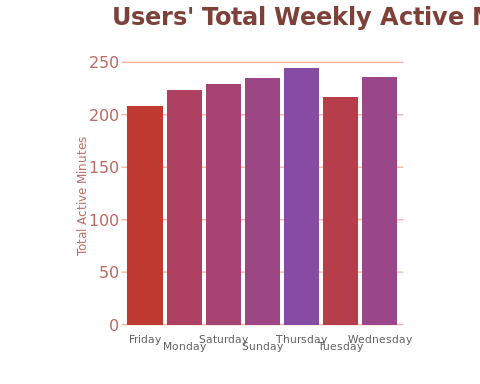
activity\_analysis$weekday = weekdays(activity\_analysis$activity\_date)  
head(activity\_analysis)

## # A tibble: 6 × 17  
## id activity\_…¹ total…² total…³ track…⁴ logge…⁵ very\_…⁶ moder…⁷ light…⁸  
## <dbl> <date> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1503960366 2020-04-12 13162 8.5 8.5 0 1.88 0.550 6.06  
## 2 1503960366 2020-04-13 10735 6.97 6.97 0 1.57 0.690 4.71  
## 3 1503960366 2020-04-14 10460 6.74 6.74 0 2.44 0.400 3.91  
## 4 1503960366 2020-04-15 9762 6.28 6.28 0 2.14 1.26 2.83  
## 5 1503960366 2020-04-16 12669 8.16 8.16 0 2.71 0.410 5.04  
## 6 1503960366 2020-04-17 9705 6.48 6.48 0 3.19 0.780 2.51  
## # … with 8 more variables: sedentary\_active\_distance <dbl>,  
## # very\_active\_minutes <dbl>, fairly\_active\_minutes <dbl>,  
## # lightly\_active\_minutes <dbl>, sedentary\_minutes <dbl>, calories <dbl>,  
## # total\_active\_minutes <dbl>, weekday <chr>, and abbreviated variable names  
## # ¹​activity\_date, ²​total\_steps, ³​total\_distance, ⁴​tracker\_distance,  
## # ⁵​logged\_activities\_distance, ⁶​very\_active\_distance,  
## # ⁷​moderately\_active\_distance, ⁸​light\_active\_distance

active\_minutes\_data <- activity\_analysis %>%  
 group\_by(weekday) %>%  
 summarize(total\_active\_minutes = mean(total\_active\_minutes)) %>%  
 arrange(desc(total\_active\_minutes))  
  
head(active\_minutes\_data)

## # A tibble: 6 × 2  
## weekday total\_active\_minutes  
## <chr> <dbl>  
## 1 Thursday 244.  
## 2 Wednesday 236.  
## 3 Sunday 235.  
## 4 Saturday 229.  
## 5 Monday 224.  
## 6 Tuesday 217.

ggplot(active\_minutes\_data, aes(x=weekday, y=total\_active\_minutes)) +  
 geom\_bar(stat="identity", aes(x=weekday, y=total\_active\_minutes, fill=total\_active\_minutes)) +   
 scale\_fill\_gradient(low="#BF3A30", high="#864BA2", guide="none") +  
 xlab("") +  
 ylab("Total Active Minutes") +  
 labs(title="Users' Total Weekly Active Minutes") +  
 theme(aspect.ratio = 1,  
 text=element\_text(family="Ariel", size=15),  
 plot.title=element\_text(hjust=0.05, face="bold", size=18, color="#7E413A", margin=margin(b=19)),  
 axis.title.x=element\_text(margin=margin(t=5)),  
 axis.title.y=element\_text(size=9, color="#BA6C65", margin=margin(r=)),  
 axis.text.x=element\_text(size=8, color="#5C5C5C", margin=margin(t=-2)),  
 axis.text.y.left=element\_text(color="#BA6C65"),  
 axis.ticks.length.x=unit(0, "cm"),  
 axis.ticks.length.y=unit(0, "cm"),  
 panel.grid.major.x=element\_blank(),  
 panel.grid.major.y=element\_line(color="#FCB59C"),  
 legend.title=element\_text(color="#333333"),  
 legend.text=element\_text(color="#333333"),  
 panel.background=element\_rect(fill="white", color="white")) +  
scale\_x\_discrete(guide = guide\_axis(n.dodge=2))



Similarly, let’s find out which days were users least active:

activity\_analysis %>%  
 group\_by(weekday) %>%  
 summarize(sedentary\_minutes = mean(sedentary\_minutes)) %>%  
 arrange(desc(sedentary\_minutes))

## # A tibble: 7 × 2  
## weekday sedentary\_minutes  
## <chr> <dbl>  
## 1 Saturday 1028.  
## 2 Sunday 1007.  
## 3 Wednesday 1000.  
## 4 Friday 990.  
## 5 Monday 989.  
## 6 Thursday 964.  
## 7 Tuesday 962.

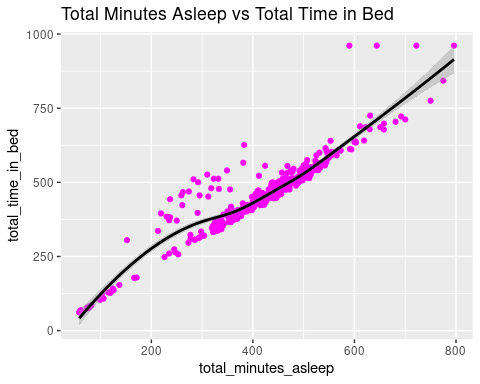
Users were *most active on Thursdays* and *least active on Saturdays.*

## 2. How much time do users spend in bed compared to actual minutes asleep?

National Sleep Foundation guidelines advise that healthy adults need between 7 and 9 hours of sleep per night. It’s important to wind down and get the most out of your time in bed. Next, we will compare the time spend in bed to actual minutes asleep.

ggplot(data = dailysleep) +  
 geom\_point(mapping = aes(x = total\_minutes\_asleep, y = total\_time\_in\_bed), color = 'magenta') +  
 geom\_smooth(mapping = aes(x = total\_minutes\_asleep, y = total\_time\_in\_bed), color = 'black') +  
 labs(title = "Total Minutes Asleep vs Total Time in Bed")

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



## 3. What is the relationship between users’ calories burnt and sleep pattern?

names(dailysleep)[names(dailysleep) == 'sleep\_day'] <- 'date'  
names(dailyActivity)[names(dailyActivity) == 'activity\_date'] <- 'date'  
head(dailysleep)

## # A tibble: 6 × 5  
## id date total\_sleep\_records total\_minutes\_asleep total\_time\_in…¹  
## <dbl> <date> <dbl> <dbl> <dbl>  
## 1 1503960366 2020-04-12 1 327 346  
## 2 1503960366 2020-04-13 2 384 407  
## 3 1503960366 2020-04-15 1 412 442  
## 4 1503960366 2020-04-16 2 340 367  
## 5 1503960366 2020-04-17 1 700 712  
## 6 1503960366 2020-04-19 1 304 320  
## # … with abbreviated variable name ¹​total\_time\_in\_bed

head(dailyActivity)

## # A tibble: 6 × 15  
## id date total\_…¹ total…² track…³ logge…⁴ very\_…⁵ moder…⁶ light…⁷  
## <dbl> <date> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1503960366 2020-04-12 13162 8.5 8.5 0 1.88 0.550 6.06  
## 2 1503960366 2020-04-13 10735 6.97 6.97 0 1.57 0.690 4.71  
## 3 1503960366 2020-04-14 10460 6.74 6.74 0 2.44 0.400 3.91  
## 4 1503960366 2020-04-15 9762 6.28 6.28 0 2.14 1.26 2.83  
## 5 1503960366 2020-04-16 12669 8.16 8.16 0 2.71 0.410 5.04  
## 6 1503960366 2020-04-17 9705 6.48 6.48 0 3.19 0.780 2.51  
## # … with 6 more variables: sedentary\_active\_distance <dbl>,  
## # very\_active\_minutes <dbl>, fairly\_active\_minutes <dbl>,  
## # lightly\_active\_minutes <dbl>, sedentary\_minutes <dbl>, calories <dbl>, and  
## # abbreviated variable names ¹​total\_steps, ²​total\_distance,  
## # ³​tracker\_distance, ⁴​logged\_activities\_distance, ⁵​very\_active\_distance,  
## # ⁶​moderately\_active\_distance, ⁷​light\_active\_distance

We will perform a full INNER JOIN to combine sleep data and activitie data. By doing this, we ensure that only the most relavent data entries are considered.

dailycal\_dailysleep<- merge(dailyActivity, dailysleep, by=c ('id', 'date'), all = FALSE)  
head(dailycal\_dailysleep)

## id date total\_steps total\_distance tracker\_distance  
## 1 1503960366 2020-04-12 13162 8.50 8.50  
## 2 1503960366 2020-04-13 10735 6.97 6.97  
## 3 1503960366 2020-04-15 9762 6.28 6.28  
## 4 1503960366 2020-04-16 12669 8.16 8.16  
## 5 1503960366 2020-04-17 9705 6.48 6.48  
## 6 1503960366 2020-04-19 15506 9.88 9.88  
## logged\_activities\_distance very\_active\_distance moderately\_active\_distance  
## 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  
## light\_active\_distance sedentary\_active\_distance very\_active\_minutes  
## 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  
## fairly\_active\_minutes lightly\_active\_minutes sedentary\_minutes 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  
## total\_sleep\_records total\_minutes\_asleep total\_time\_in\_bed  
## 1 1 327 346  
## 2 2 384 407  
## 3 1 412 442  
## 4 2 340 367  
## 5 1 700 712  
## 6 1 304 320

We can also create a new data frame from the existing one with just the sleep and calories columns.

dailycal\_dailysleep\_merged<- dailycal\_dailysleep[, c("id", "date", "calories","total\_minutes\_asleep")]  
head(dailycal\_dailysleep\_merged)

## id date calories total\_minutes\_asleep  
## 1 1503960366 2020-04-12 1985 327  
## 2 1503960366 2020-04-13 1797 384  
## 3 1503960366 2020-04-15 1745 412  
## 4 1503960366 2020-04-16 1863 340  
## 5 1503960366 2020-04-17 1728 700  
## 6 1503960366 2020-04-19 2035 304

We’ll now perform a correlation test to understand if total minutes asleep and total calories burnt have positive correlation or negative correlation, and to what degree they’re correlated.

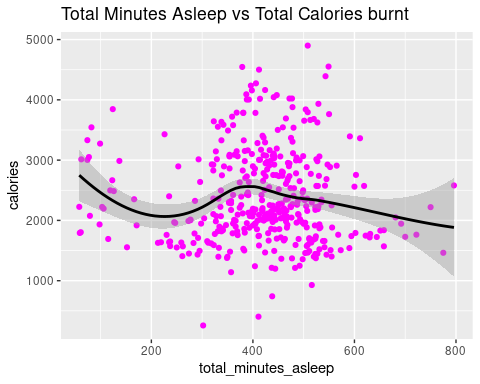
cor.test(dailycal\_dailysleep\_merged$total\_minutes\_asleep, dailycal\_dailysleep\_merged$total\_minutes\_asleep, method=c("pearson"))

##   
## Pearson's product-moment correlation  
##   
## data: dailycal\_dailysleep\_merged$total\_minutes\_asleep and dailycal\_dailysleep\_merged$total\_minutes\_asleep  
## t = Inf, df = 408, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 1 1  
## sample estimates:  
## cor   
## 1

The correlation test says there significant correlation between total minutes asleep and total calories burnt. To help visualise this correlation, let’s look at the scatterplot.

ggplot(data = dailycal\_dailysleep\_merged) +  
 geom\_point(mapping = aes(x = total\_minutes\_asleep, y = calories), color = 'magenta') +  
 geom\_smooth(mapping = aes(x = total\_minutes\_asleep, y = calories), color = 'black') +  
 labs(title = "Total Minutes Asleep vs Total Calories burnt")

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



# Analysis - Summary

1. Users were more active on Thursdays and were least active on Saturdays.
2. Users fell asleep as soon as they hit the bed. There wasn’t much difference between time in bed vs actual time asleep.
3. Users who sleep more tend to burn more calories

# Recommendations:

First, we should acknowledge that the data set used for this case study is small. Nevertheless, we gained significant insights from this study. Following are the recommendations for Bellabeat:

### 1. Notifications:

Sometimes a nudge in the right direction is all we need to stay on track of our fitness goals. We saw from user data that users tend to be lethargic on Saturdays. The Bellabeat fitness app can send out push notifications to motivate users by reminding them of how well they did on weekdays. Constant reminders like this will ensure user traffic

### 2. Good night’s sleep

The devices can track user’s sleep data and send vibration alerts if they’re spending too much time in bed without sleeping. Sleep data collected could include resting heart rate while sleeping, restlessness etc and provide insights (articles, videos, sleep podcasts, etc) the next day for a good night’s sleep. ### 3. Rewards for tracking weight and other data Too few users tracked their weight. Bellabeat can encourage users to keep track of their data, especially those which require manual input, like weight. It can be badges, listing personal best weeks, coupons, access to premium content or even discounts on Bellabeat subscription.