Scenario

I am a junior data analyst working on the marketing analyst team at Bellabeat, a high-tech manufacturer of health-focused products for women. Bellabeat is a successful small company, but they have the potential to become a larger player in the global smart device market. Urška Sršen, co-founder and Chief Creative Officer of Bellabeat, believes that analysing smart device fitness data could help unlock new growth opportunities for the company. The ask is to focus on one of Bellabeat's products and analyse smart device data to gain insight into how consumers are using their smart devices. The insights discovered will then help guide marketing strategy for the company.

Bellabeat Products

- **Bellabeat App:** The Bellabeat app provides users with health data related to their activity, sleep, stress, menstrual cycle, and mindfulness habits. This data can help users better understand their current habits and make healthy decisions. The Bellabeat app connects to their line of smart wellness products.
- **Leaf:** Bellabeat's classic wellness tracker can be worn as a bracelet, necklace, or clip. The Leaf tracker connects to the Bellabeat app to track activity, sleep, and stress.
- **Time:** This wellness watch combines the timeless look of a classic timepiece with smart technology to track user activity, sleep, and stress. The Time watch connects to the Bellabeat app to provide you with insights into your daily wellness.
- **Spring:** This is a water bottle that tracks daily water intake using smart technology to ensure that you are appropriately hydrated throughout the day. The Spring bottle connects to the Bellabeat app to track your hydration levels.
- **Bellabeat Membership:** Bellabeat also offers a subscription-based membership program for users. Membership gives users 24/7 access to fully personalized guidance on nutrition, activity, sleep, health and beauty, and mindfulness based on their lifestyle and goals.

1. Ask

Analyse smart device usage data in order to gain insight into how consumers use non-Bellabeat smart devices. She then wanted me to select one Bellabeat product to apply these insights in my presentation.

1.1 Business Task

The business task is to analyse smart device usage data of non-Bellabeat smart devices to gain insight into relevant (successful and unsuccessful) consumer trends within the global smart device market, as well as to discover how to use these trends to apply to Bellabeat customers and to influence future Bellabeat marketing strategies. This is done by applying said insights to the Bellabeat App and to future products in order to maximise profits and growth for the company and to capitilse on Bellabeat's rapidly growing consumer base in the smart device/tech-wellness space. Urška Sršen, Sando Mur (Bellabeat's other co-founder and key member of the Bellabeat executive team), the Bellabeat executive team, the Bellabeat Marketing Analytics Team, and the Bellabeat investors are the key stakeholders that need to be considered in my data analysis and decisions.

2. Prepare and Process

The project team encouraged me to use the following public data that explores smart device users' daily habits: FitBit Fitness Tracker Data (CC0: Public Domain, dataset made available through Mobius).

- This Kaggle data set contains personal fitness trackers from 30 FitBit users. 30 eligible FitBit users consented to the submission of personal tracker data, including minute-level output for physical activity, heart rate, and sleep monitoring. It includes information about daily activities, steps, and heart rate that can be used to explore users' habits.

2.1 Notes about the Data

18 total data sets were provided in the FitBit Fitness Tracker Data link from above; they are individually stored in the form of .csv files. This analysis will instead focus on four data sets: the daily activity data set ('daily_activity'), which contains merged data from other provided files like daily calories, daily intensities, and daily steps, the weight data set ('weight'), the hourly steps data set ('hourly_steps') and the daily sleep data set ('sleep'). These files contain relevant data that are also tracked by Bellabeat products - this will provide me with the most relevant and useful insights to solve the business task at hand.

2.2 Issues with the Data Credibility

I will be using the **ROCC** framework to determine if there are any issues with bias or credibility in this data.

- **R**eliabile: **NOT** reliable. This data only contains 30 selected individuals, which is not a representative sample bias of the 30+ million FitBit users. This would equate to a 95%/90% confidence level with a 18%/15% margin of error. A sample size of more than 10 times the current amount would be a good minimum to provide a high confidence level (95%) and a low margin of error (5%). It should be noted, though, that according the the **Central Limit Theorem (CLT)**, a sample of 30 is the smallest sample size for which the CLT is still valid. So, it is good that the provided data at least meets this metric. Furthermore, all of the data was obtained over the course of two months, which is not a long enough time to deduce accurate and reliable trends.
- **O**riginal: **NOT** original. The data set was generated by respondents to a distributed survey via Amazon Mechanical Turk. It would have been better if the data was supplied directly by FitBit.
- **C**omprehensive: **NOT** comprehensive. The data is not comprehensive in the sense that other data (not present) would be useful to create a more accurate analysis (e.g., sex, age, height, etc.). Also, having more data from more individuals would help with the overall comprehensiveness; for example, having a more accurate sample bias of the 30+ million FitBit users. In addition, there is no way of knowing if there was a bias in the selection of the individuals, or if it was selected at random. What was the criteria for selecting the 30 individuals? More details about the data would help.
- **C**urrent: **NOT** current. The data was obtained years ago, which is not an up-to-date data set representative of current trends.
- **C**ited: Cited but **NOT** credible. The data came from Amazon Mechanical Turk, so it could be a reliable source or it could not. More research needs to be done about the integrity and credibility of Amazon Mechanical Turk.

Overall, the conclusions made from this analysis should be considered in line with the above oberservations, as the data integrity and credibility are lacking. However, the general insights could still prove to be useful by highlighting possible shortcomings in FitBit's data tracking (by FitBit itself or by the individuals), which Bellabeat can then improve on through innovative features exclusive to the Bellabeat product line.

Installing Packages

```
library(tidyverse)
library(lubridate)
library(ggplot2)
library(readr)
library(tidyr)
library(dplyr)
library(skimr)
library(janitor)
library(scales)
```

Importing files

```
daily_activity <- read_csv("dailyActivity_merged.csv")
sleep <- read_csv("sleepDay_merged.csv")
weight <- read_csv("weightLogInfo_merged.csv")
hourly_steps <- read.csv("hourlySteps_merged.csv")
```

Viewing the Data

```
head(daily_activity)
```

```
# A tibble: 6 × 15
          Id Activi…¹ Total…² Total…³ Track…⁴ Logge…⁵ VeryA…⁶ Moder…⁷ Light…⁸
Seden…⁹
       <dbl> <chr>       <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
<dbl>
1 1503960366 4/12/20…   13162     8.5     8.5       0    1.88   0.550    6.06
0
2 1503960366 4/13/20…   10735    6.97    6.97       0    1.57   0.690    4.71
0
3 1503960366 4/14/20…   10460    6.74    6.74       0    2.44   0.400    3.91
0
4 1503960366 4/15/20…    9762    6.28    6.28       0    2.14   1.26     2.83
0
5 1503960366 4/16/20…   12669    8.16    8.16       0    2.71   0.410    5.04
0
6 1503960366 4/17/20…    9705    6.48    6.48       0    3.19   0.780    2.51
0
# … with 5 more variables: VeryActiveMinutes <dbl>, FairlyActiveMinutes <dbl>,
#   LightlyActiveMinutes <dbl>, SedentaryMinutes <dbl>, Calories <dbl>, and
#   abbreviated variable names ¹ActivityDate, ²TotalSteps, ³TotalDistance,
#   ⁴TrackerDistance, ⁵LoggedActivitiesDistance, ⁶VeryActiveDistance,
#   ⁷ModeratelyActiveDistance, ⁸LightActiveDistance, ⁹SedentaryActiveDistance
```

```
head(sleep)
```

```
# A tibble: 6 × 5
          Id SleepDay              TotalSleepRecords TotalMinutesAsleep
TotalTimeIn…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
# … with abbreviated variable name ¹TotalTimeInBed
```

```
head(weight)
```

```
# A tibble: 6 × 8
          Id Date                  WeightKg WeightPounds   Fat   BMI IsManu…¹
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
# … with abbreviated variable name ¹IsManualReport
```

```
head(hourly_steps)
```

```
          Id     ActivityHour       StepTotal
1 1503960366 4/12/2016 12:00:00 AM       373
2 1503960366  4/12/2016 1:00:00 AM       160
3 1503960366  4/12/2016 2:00:00 AM       151
4 1503960366  4/12/2016 3:00:00 AM         0
5 1503960366  4/12/2016 4:00:00 AM         0
6 1503960366  4/12/2016 5:00:00 AM         0
```

```
colnames(daily_activity)
```

```
 [1] "Id"                       "ActivityDate"             "TotalSteps"
 [4] "TotalDistance"            "TrackerDistance"
"LoggedActivitiesDistance"
 [7] "VeryActiveDistance"       "ModeratelyActiveDistance"
"LightActiveDistance"
[10] "SedentaryActiveDistance"  "VeryActiveMinutes"
"FairlyActiveMinutes"
[13] "LightlyActiveMinutes"     "SedentaryMinutes"         "Calories"
```

```
colnames(hourly_steps)
```

```
[1] "Id"           "ActivityHour" "StepTotal"
```

```
colnames(sleep)
```

```
[1] "Id"                 "SleepDay"           "TotalSleepRecords"
[4] "TotalMinutesAsleep" "TotalTimeInBed"
```

```
colnames(weight)
```

```
[1] "Id"             "Date"           "WeightKg"       "WeightPounds"
[5] "Fat"            "BMI"            "IsManualReport" "LogId"
```

```
view(daily_activity)
str(daily_activity)
```

```
spec_tbl_df [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>
```

```
view(hourly_steps)
str(hourly_steps)
```

```
'data.frame': 22099 obs. of  3 variables:
 $ Id          : num  1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
 $ ActivityHour: chr  "4/12/2016 12:00:00 AM" "4/12/2016 1:00:00 AM" "4/12/2016
2:00:00 AM" "4/12/2016 3:00:00 AM" ...
 $ StepTotal   : int  373 160 151 0 0 0 0 0 250 1864 ...
```

```
view(sleep)
str(sleep)
```

```
spec_tbl_df [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>
```

```
view(weight)
str(weight)
```

```
spec_tbl_df [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" ...
 $ WeightKg      : num [1:67] 52.6 52.6 133.5 56.7 57.3 ...
 $ WeightPounds  : 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 ...
 $ IsManualReport: logi [1:67] TRUE TRUE FALSE TRUE TRUE TRUE ...
 $ LogId         : 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>
```

Upon review of the data, one of the first observations I made was that there were numerous entries with zero values entered where participants neglected to wear and hence log information of their daily activities. Removing these blank data entries will help improve the quality of the data analysis. There are also numerous formatting errors which will need to be addressed. Next, I checked for any duplicate values in the datasets.

```
daily_activity_new <- daily_activity %>%
  filter(TotalSteps !=0)

view(daily_activity_new)
```

```
daily_activity_new %>% duplicated() %>% sum()
```

```
[1] 0
```

```
weight %>% duplicated() %>% sum()
```

```
[1] 0
```

```
sleep %>% duplicated() %>% sum()
```

```
[1] 3
```

```
hourly_steps %>% duplicated() %>% sum()
```

```
[1] 0
```

In the sleep dataset I found three duplicated entries which I will remove from the data to avoid skewing the results and distorting the analysis.

```
sleep_new <- unique(sleep)
```

```
Sleep_new %>% duplicated() %>% sum()
```

```
[1] 0
```

Now I can verify the number of distinct users in each of the datasets.

```
n_distinct(daily_activity_new$Id)
```

```
[1] 33
```

```
n_distinct(sleep_new$Id)
```

```
[1] 24
```

```
n_distinct(weight_new$Id)
```

```
[1] 8
```

```
n_distinct(weight_new$Id)
```

```
[1] 33
```

From the above results we can already see that while there are issues with the data not only from a cleanliness perspective, but also from a completeness point of view. Despite the project documentation indicating there are only 30 participants in the study, there are 33 unique Ids in the daily activity data. Furthermore, not everyone entered data for their sleep times and even fewer for their weight.

This presents big questions regarding the comprehensiveness of the data and ultimate reliability of the analysis.

Finally, in the sleep and weight datasets I noticed the columns titled “SleepDay” and “Date” respectively, contain the combined date and time information for each record. To help make the analysis easier, I will separate the date and time data for each and rename the columns.

```
sleep_new <- sleep_new %>%
  separate(SleepDay, c("Date", "Time"), " ")

weight_new <- weight %>%
  separate(Date, c("Date", "Time"), " ")
```

I then reformatted the columns to lowercase for overall consistency and readability.

```
daily_activity_new <- clean_names(daily_activity_new)
sleep_new <- clean_names(sleep_new)
weight_new <- clean_names(weight_new)
hourly_steps <- clean_names(hourly_steps)
```

Next, I renamed the ActivityHour column in the hourly_steps dataset, converted the time to 24-hour format and separated the columns into individual date and time columns.

```
hourly_steps <- hourly_steps %>%
  rename(date = ActivityHour)

hourly_steps <- hourly_steps %>%
  mutate(date = as.POSIXct(date, format = "%m/%d/%Y %I:%M:%S %p" , tz =
Sys.timezone()))

  hourly_steps <- hourly_steps %>%
    separate(date, c("date", "time"), " ")
```

Following this, I double checked the information classes of each of the columns in the datasets

```
sapply(daily_activity_new, class)
```

```
                  id              activity_date                total_steps
           "numeric"                "character"                  "numeric"
       total_distance             tracker_distance gged_activities_distance
           "numeric"                  "numeric"                    "numeric"
  very_active_distance moderately_active_distance    light_active_distance
           "numeric"                  "numeric"                    "numeric"
sedentary_active_distance        very_active_minutes    fairly_active_minutes
           "numeric"                  "numeric"                    "numeric"
lightly_active_minutes           sedentary_minutes                calories
           "numeric"                  "numeric"                   "numeric"
```

```
sapply(sleep_new, class)
```

```
            id                  date             time       total_sleep_records
      "numeric"           "character"      "character"          "numeric"
total_minutes_asleep    total_time_in_bed
      "numeric"           "numeric"
```

```
sapply(weight_new, class)
```

```
      id             date             time        weight_kg    weight_pounds
   numeric"      "character"      "character"    "numeric"        "numeric"
     fat              bmi      is_manual_report       log_id
  "numeric"       "numeric"        "logical"       "numeric"
```

```
sapply(hourly_steps, class)
```

```
    id        date        time      step_total
 "numeric" "character" "character"   "integer"
```

After checking the data classes, I noticed all the date columns were classed as characters. I reformatted these as dates to make later analysis easier.

```
sleep_new <- sleep_new %>%
  mutate(date = as.Date(date,format = "%m/%d/%Y"))

sapply(sleep_new, class)
```

```
            id                  date             time       total_sleep_records
      "numeric"             "Date"       "character"          "numeric"
total_minutes_asleep    total_time_in_bed
      "numeric"           "numeric"
```

```
daily_activity_new <- daily_activity_new %>%
  mutate(activity_date = as.Date(activity_date,format = "%m/%d/%Y"))

sapply(daily_activity_new, class)
```

```
                      id              activity_date                 total_steps
                "numeric"                    "Date"                   "numeric"
            total_distance           tracker_distance gged_activities_distance
                "numeric"                   "numeric"                   "numeric"
      very_active_distance moderately_active_distance    light_active_distance
                "numeric"                   "numeric"                   "numeric"
 sedentary_active_distance        very_active_minutes    fairly_active_minutes
                "numeric"                   "numeric"                   "numeric"
    lightly_active_minutes          sedentary_minutes                calories
                "numeric"                   "numeric"                   "numeric"
```

```
Hourly_steps <- hourly_steps %>%
  mutate(date = as.Date(date,format = "%m/%d/%Y"))

sapply(hourly_steps, class)
```

```
     id        date        time      step_total
  "numeric" "Date"     "character"   "integer"
```

```
weight_new <- weight_new %>%
  mutate(date = as.Date(date,format = "%m/%d/%Y"))

sapply(weight_new, class)
```

```
      id             date             time        weight_kg     weight_pounds
    numeric"        "Date"      "character"    "numeric"         "numeric"
      fat              bmi      is_manual_report       log_id
    "numeric"        "numeric"        "logical"        "numeric"
```

3. Analyse and Share

Using the skim without charts and summary functions, I reviewed the detailed data summaries for each set to gain a comprehensive overview of the now cleaned data and observed any obvious trends from which I can derive initial high-level insights.

```
skim_without_charts(sleep_new)
```

```
── Data Summary ────────────────────────
                           Values
Name                       sleep_new
Number of rows             410
Number of columns          6
___________________________
Column type frequency:
  character                1
  Date                     1
  numeric                  4
___________________________
Group variables            None

── Variable type: character ──────────────────────────────────────────────────
  skim_variable n_missing complete_rate min max empty n_unique whitespace
1 time                  0             1   8   8     0        1          0

── Variable type: Date ───────────────────────────────────────────────────────
  skim_variable n_missing complete_rate min        max        median
n_unique
1 date                  0             1 2016-04-12 2016-05-12 2016-04-27
31

── Variable type: numeric ────────────────────────────────────────────────────
  skim_variable        n_missing complete_rate          mean      sd          p0
1 id                           0          1 4994963041.   2.06e+9 1503960366
2 total_sleep_records          0          1          1.12 3.47e-1          1
3 total_minutes_asleep         0          1        419.   1.19e+2         58
4 total_time_in_bed            0          1        458.   1.27e+2         61
          p25         p50        p75       p100
1 3977333714  4702921684  6962181067 8792009665
2          1           1          1          3
3        361         432.       490        796
4        404.        463        526        961
```

```
skim_without_charts(daily_activity_new)
```

```
── Data Summary ────────────────────────
                           Values
Name                       daily_activity_new
Number of rows             863
Number of columns          15
___________________________
Column type frequency:
  Date                     1
  numeric                  14
___________________________
Group variables            None

── Variable type: Date ─────────────────────────────────────────────────────────
  skim_variable n_missing complete_rate min        max        median
n_unique
1 activity_date         0             1 2016-04-12 2016-05-12 2016-04-26
31

── Variable type: numeric ──────────────────────────────────────────────────────
   skim_variable              n_missing complete_rate    mean      sd
p0
 1 id                                 0             1 4.86e+9 2.42e+9 1503960366
 2 total_steps                        0             1 8.32e+3 4.74e+3          4
 3 total_distance                     0             1 5.98e+0 3.72e+0          0
 4 tracker_distance                   0             1 5.96e+0 3.70e+0          0
 5 logged_activities_distance         0             1 1.18e-1 6.46e-1          0
 6 very_active_distance               0             1 1.64e+0 2.74e+0          0
 7 moderately_active_distance         0             1 6.18e-1 9.05e-1          0
 8 light_active_distance              0             1 3.64e+0 1.86e+0          0
 9 sedentary_active_distance          0             1 1.75e-3 7.65e-3          0
10 very_active_minutes                0             1 2.30e+1 3.36e+1          0
11 fairly_active_minutes              0             1 1.48e+1 2.04e+1          0
12 lightly_active_minutes             0             1 2.10e+2 9.68e+1          0
13 sedentary_minutes                  0             1 9.56e+2 2.80e+2          0
14 calories                           0             1 2.36e+3 7.03e+2         52
             p25     p50     p75    p100
 1 2320127002    4.45e+9 6.96e+9 8.88e+9
 2       4923    8.05e+3 1.11e+4 3.60e+4
 3          3.37 5.59e+0 7.90e+0 2.80e+1
 4          3.37 5.59e+0 7.88e+0 2.80e+1
 5          0    0       0       4.94e+0
 6          0    4.10e-1 2.27e+0 2.19e+1
 7          0    3.10e-1 8.65e-1 6.48e+0
 8          2.34 3.58e+0 4.89e+0 1.07e+1
 9          0    0       0       1.10e-1
10          0    7   e+0 3.5 e+1 2.1 e+2
11          0    8   e+0 2.1 e+1 1.43e+2
12        146.   2.08e+2 2.72e+2 5.18e+2
13        722.   1.02e+3 1.19e+3 1.44e+3
14       1856.   2.22e+3 2.83e+3 4.9 e+3
```

```
skim_without_charts(weight_new)
```

```
── Data Summary ────────────────────────
                           Values
Name                       weight_new
Number of rows             67
Number of columns          9
_________________________
Column type frequency:
  character                1
  Date                     1
  logical                  1
  numeric                  6
_________________________
Group variables            None

── Variable type: character ────────────────────────────────────────────────────
skim_variable n_missing complete_rate min max empty n_unique whitespace
1 time                0         1   7   8     0       26          0

── Variable type: Date ─────────────────────────────────────────────────────────
  skim_variable n_missing complete_rate min        max        median
n_unique
1 date                  0  1 2016-04-12 2016-05-12 2016-04-27       31

── Variable type: logical ──────────────────────────────────────────────────────
  skim_variable    n_missing complete_rate  mean count
1 is_manual_report         0             1 0.612 TRU: 41, FAL: 26

── Variable type: numeric ──────────────────────────────────────────────────────
  skim_variable n_missing complete_rate    mean       sd      p0     p25
p50
1 id                    0     1      7.01e 9 1950321944.   1.50e 9 6.96e 9
6.96e 9
2 weight_kg             0        1      7.20e 1         13.9  5.26e 1 6.14e 1
6.25e 1
3 weight_pounds         0        1      1.59e 2         30.7  1.16e 2 1.35e 2
1.38e 2
4 fat                  65        0.0299 2.35e 1          2.12 2.2 e 1 2.28e 1
2.35e 1
5 bmi                   0        1      2.52e 1          3.07 2.15e 1 2.40e 1
2.44e 1
6 log_id                0        1      1.46e12  782994784.   1.46e12 1.46e12
1.46e12
      p75    p100
1 8.88e 9 8.88e 9
2 8.50e 1 1.34e 2
3 1.88e 2 2.94e 2
4 2.42e 1 2.5 e 1
5 2.56e 1 4.75e 1
6 1.46e12 1.46e12
```

```
skim_without_charts(hourly_steps)
```

```
── Data Summary ────────────────────────
                           Values
Name                       hourly_steps
Number of rows             22099
Number of columns          4
_____________________
Column type frequency:
  character                1
  Date                     1
  numeric                  2
_____________________
Group variables            None

── Variable type: character ────────────────────────────────────────────────
  skim_variable n_missing complete_rate min max empty n_unique whitespace
1 time                  0             1   7   8     0       12          0

── Variable type: Date ─────────────────────────────────────────────────────
  skim_variable n_missing complete_rate min        max        median
n_unique
1 date                  0             1 2016-04-12 2016-05-12 2016-04-26
31

── Variable type: numeric ──────────────────────────────────────────────────
  skim_variable n_missing complete_rate        mean          sd         p0
p25
1 id                    0             1 4848235270. 2422500401. 1503960366
2320127002
2 step_total            0             1        320.        690.          0
0
         p50        p75       p100
1 4445114986 6962181067 8877689391
2         40        357      10554
```

The summary of the datasets is useful to confirm all the necessary cleaning has been done and if there are any immediate issues that stand out. It is also a useful place to start for identifying potential relationships in the data and areas to explore further as part of the visualisation stage. The data looks okay so far (apart from the limitations already mentioned earlier) and has not showed any particularly alarming results. The last step here is to examine the more focused outline of each set using the summary function.

```
summary(daily_activity_new)
summary(weight_new)
summary(sleep_new)
summary(hourly_steps)
```

High-Level Trends

- The average Total Steps for an individual is 8319.
- The average minutes for Very Active is 23.02, for Fairly Active is 14.78, for Lightly Active is 210, for Sedentary is 955.8.
- The average BMI is 25.19.
- The average minutes asleep is 419.2, whilst the average minutes in bed is 458.5.
- Note: There are still outliers in the data that were not removed due to the lack of information regarding the data. These were kept in there just in case those extreme values were in fact legitimate. However, in the case that those values were not legitimate, the average values above may be slightly skewed, which is worth keeping in mind.
- Several trends that I mentioned earlier were that people were not consistent in tracking/logging their hours regarding their sleep or their weight every day or any day at all - and certain individuals who were consistently tracking/logging their data were not losing weight or seeing results over the month of data collection.

Conclusions from Trends

- According to a joint research investigation by the **National Cancer Institute (NCI)**, the **National Institute on Aging (NIA)**, and the **Centers for Disease Control and Prevention (CDC)** (amongst other research studies), the ideal daily number of Total Steps one should achieve is 10,000. Thus, the average individual here is not reaching that minimum goal.
- One reason for this is their activity level. The individuals spent on average 955.8 minutes a day being sedentary, that is on average 16 hours a day.
- Since the average BMI is 25.19, this puts these individuals in the overweight category, according to the **World Health Organisation (WHO)**.
- It makes sense that overweight people are wearing FitBits to help them get in better shape, but they are not being active enough to do so.
- Additionally, the average person is getting just under the minimum recommended amount of sleep (7 hours) a person should get, according to the **National Sleep Foundation (NSF)** and many are inconsistent with recording their sleep time information, thus impacting the accuracy and comprehensiveness of the data.

Questions arising from Trends

1. How can Bellabeat devices encourage users to achieve their minimum recommended daily steps?
2. How would Bellabeat promote consistency with user logging/tracking of data?
3. How can the data quality be improved to better serve the user?

So far there have been some useful insights from the data examined despite the aforementioned shortcomings. These had led to some interesting questions which may prove useful to Bellabeat and their products. In answering the questions outlined above, I will next visualise the relationships in the data to gain a clearer understanding of the issues. Lastly, I will use these relationship observations alongside the trends insights to form my recommendations for Bellabeat's marketing strategy.

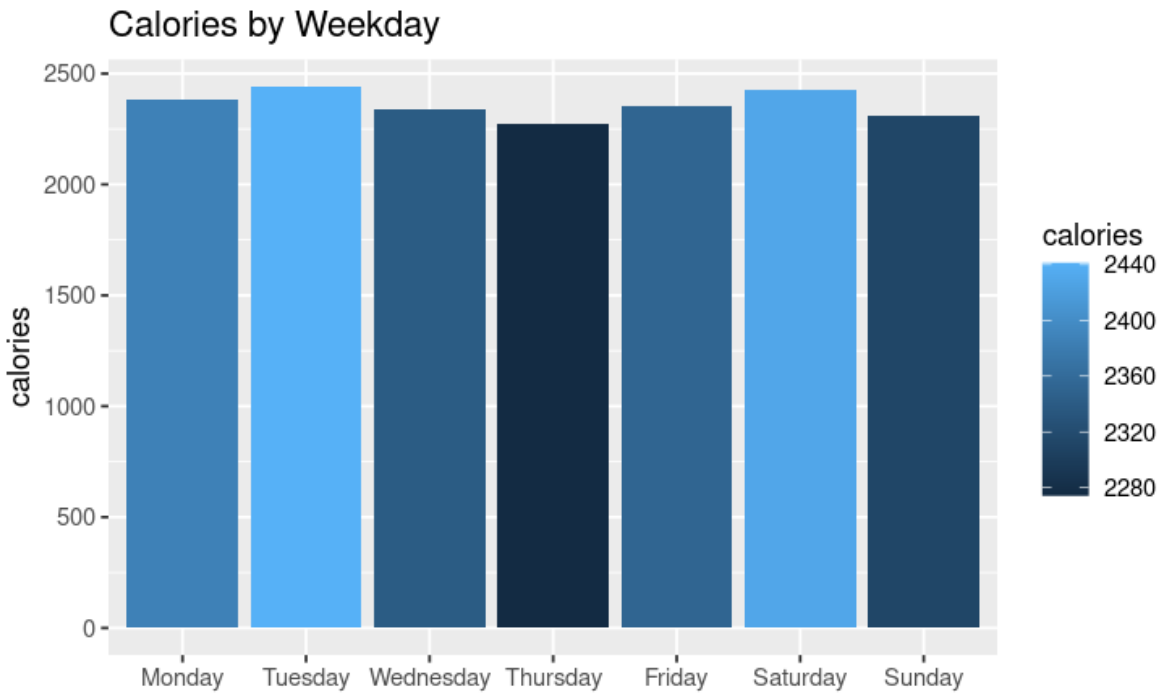
```
daily_activity_new <- daily_activity_new %>%
   rename(date = activity_date)

daily_activity_new$weekday <- weekdays(daily_activity_new$date)
```

I used the factor function to arrange the weekdays in the correct sequence prior to making the graph.

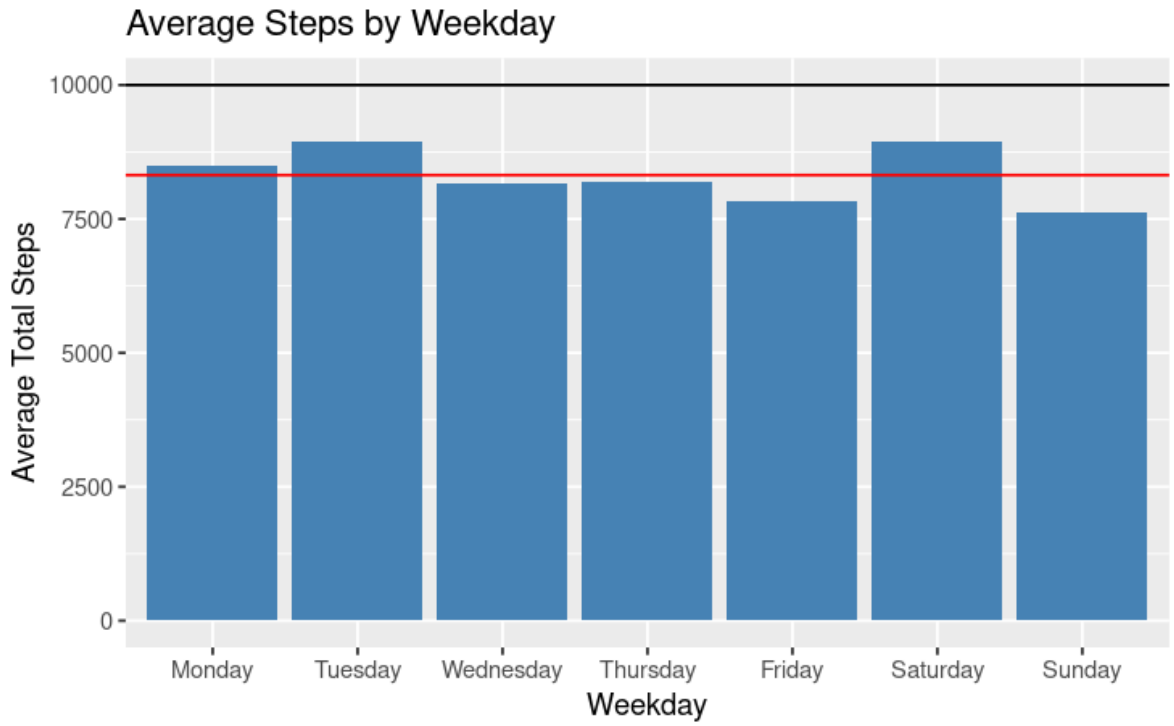
```
daily_activity_new$weekday <- factor(daily_activity_new$weekday,
levels = c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday",
"Saturday", "Sunday"))
```

Then, I examined the relationship between average calories burned and the days of the week, plotted on a column chart. This showed that on average, people burned most calories on Tuesdays and Saturdays. The graph indicates that users had higher levels of motivation for higher intensity activities earlier in the week which then wanes midweek before spiking again on the weekend and falling again for Sundays and Mondays.

```
daily_activity_new2 <- daily_activity_new %>%
  group_by(weekday) %>%
  summarize(calories = mean(calories))

ggplot(data=daily_activity_new2) +
  geom_col(mapping=aes(x=weekday, y=Calories, fill=calories)) +
  labs(title = "Calories by Weekday", x= "")
```



```
avg_steps <- daily_activity_new %>%
  group_by(weekday) %>%
  summarise(avg_steps = mean(total_steps))

ggplot(data = avg_steps) +
  geom_col(mapping = aes(x=weekday, y=avg_steps), fill="SteelBlue") +
  geom_hline(yintercept = 8319, color="red") +
  geom_hline(yintercept = 10000) +
  labs(title = "Average Steps by Weekday", x="Weekday", y="Average
Total Steps")
```

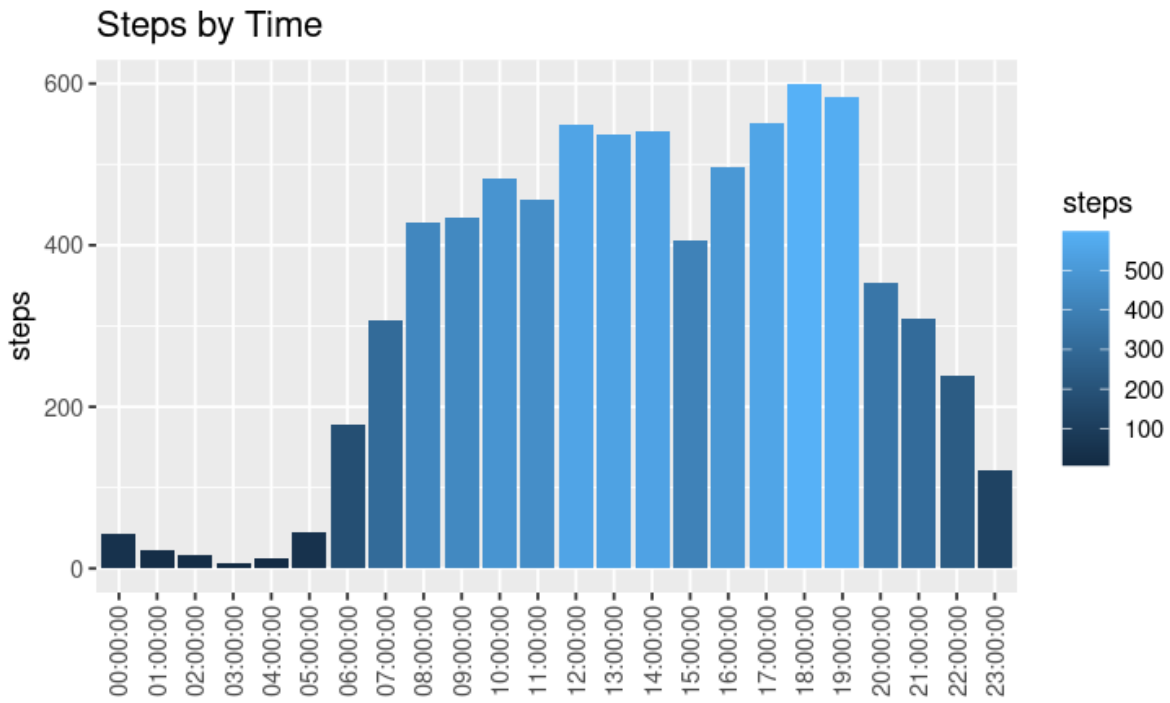


I next plotted the average number of steps for the group for each weekday. The results match the calories burned graph above, and indicate that not enough of the participants are hitting the recommended 10,000 steps per day.

Most users on average are achieving 8319 steps per day, highlighted in the graph by the red line. The data indicates average steps taken peak on Tuesdays and Saturdays, but there is a notable slump during midweek. There could be opportunity here to incentivise users to meet their daily step targets on days where it is known that users tend to lose motivation.

Next I analysed the time data to determine what time of day the participants used their device the most.

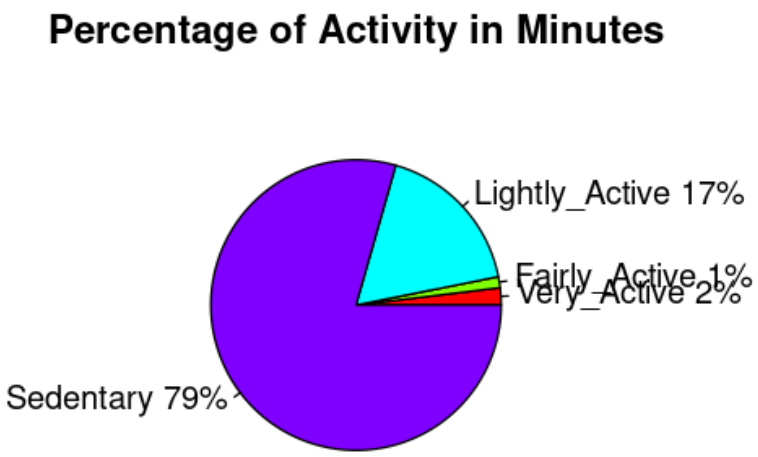
```
avg_steps_time <- hourly_steps %>%
  group_by(time) %>%
  summarize(steps = mean(StepTotal))

ggplot(data=avg_steps_time) +
  geom_col(mapping=aes(x=time, y=steps, fill=steps)) +
  labs(title = "Steps by Time", x= "") +
  theme(axis.text.x = element_text(angle = 90,vjust = 0.5, hjust = 1))
```



From this analysis, I determined that users tended to be most active during the hours of 12:00 – 14:00 and then again towards the evening between 17:00 -19:00. There is a sharp drop off in the evening and at night, as might be expected as users are sleeping during these time periods. This information is useful as again it could be use to inform marketing campaigns based around times it is known that people lose motivation or struggle to stay at least somewhat active.

Next I visualised the percentage of active minutes as a pie chart. The results indicate that the vast majority of the participants time was spend in a sedentary state (79%) and were only fairly (1%) or very active (2%) for a fraction of the time they were using the device.

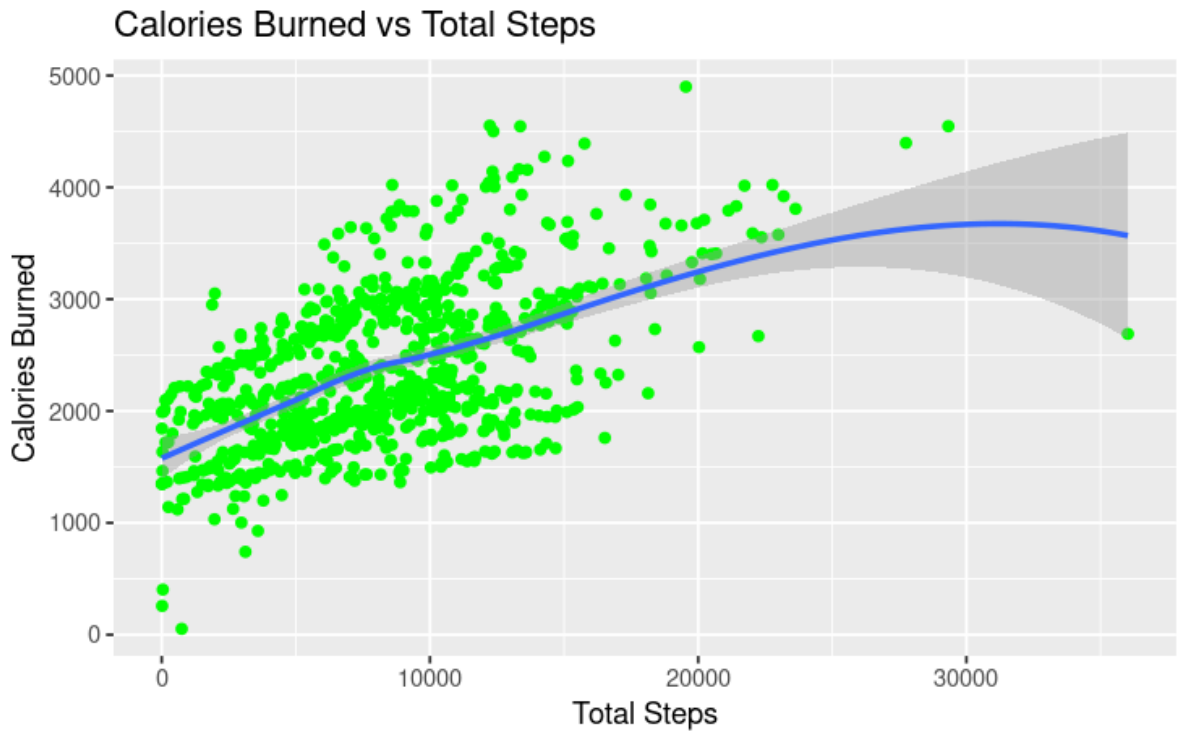
```
very_active_mins <- sum(daily_activity_new$very_active_minutes)
fairly_active_mins <- sum(daily_activity_new$fairly_active_minutes)
lightly_active_mins <- sum(daily_activity_new$lightly_active_minutes)
sedentary_mins <- sum(daily_activity_new$sedentary_minutes)

slices <- c(very_active_mins, fairly_active_mins, lightly_active_mins, sedentary_mins)
lbls <- c("Very_Active", "Fairly_Active", "Lightly_Active", "Sedentary")
pct <- round(slices/sum(slices)*100)
lbls <- paste(lbls, pct)
lbls <- paste(lbls, "%", sep="")
pie(slices, labels = lbls, col = rainbow(length(lbls)), main = "Percentage of Activity in Minutes")
```



This visual reinforces the view we have been getting throughout the analysis so far, which is that users are not active enough. The goal of using a fitness tracking device should be to encourage the user to be more active, yet the data suggests only a small proportion of time spent in a fairly or very active state.

I then plotted the relationship between the amount of calories burned and the total steps taken.

```
ggplot(data=daily_activity_new) +
  geom_point(mapping=aes(x=total_steps, y=calories), color = "green") +
  geom_smooth(mapping=aes(x=total_steps, y=calories)) +
  labs(title = "Calories Burned vs Total Steps", x="Calories Burned",
y="Total Steps")
```



As would be expected with this relationship, as the number of total steps increases the number of calories burned also increases. But as noted in the high-level trends, the average person in this dataset is only getting around 8300 steps per day. There may be benefit in implementing a system that can notify and encourage users to meet their step goals, with specific incentivisation during the midweek and or at certain times of the day when motivation levels appear to be at their lowest points (as evidenced by figures X above).

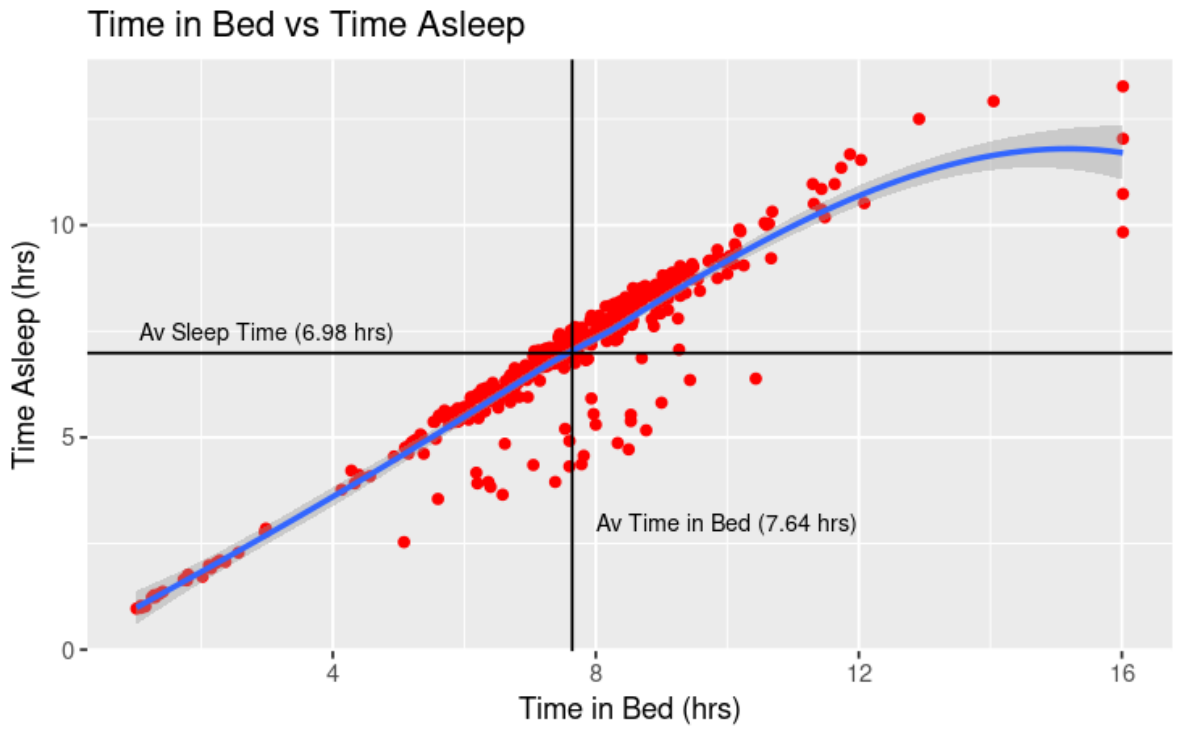
```
mean(sleep_new$total_minutes_asleep / 60)
```

```
[1] 6.98622
```

```
mean(sleep_new$total_time_in_bed / 60)
```

```
[1] 7.641382
```

```
ggplot(data = sleep_new) +
  geom_point(mapping=aes(x=total_time_in_bed / 60,
y=total_minutes_asleep / 60), color="red") +
  geom_smooth(mapping=aes(x=total_time_in_bed / 60,
y=total_minutes_asleep / 60)) +
  geom_hline(yintercept = 6.98622) +
  geom_vline(xintercept = 7.641382) +
  annotate("text", x=3,y=7.5, label="Av Sleep Time (6.98 hrs)",
size=3) +
  annotate("text", x=10,y=3, label="Av Time in Bed (7.64 hrs)",
size=3) +
  labs(title="Time in Bed vs Time Asleep", x="Time in Bed (hrs)",
y="Time Asleep (hrs)")
```



There are a number of outliers in this data, but on average we can see that users are not getting the full 8 hours recommended sleep each night. Furthermore, users appeared to struggle with tracking their sleep data accurately and consistently. This could be a significant area for improvement if the devices could automatically track the time spend asleep and provide information on the quality of the sleep attained.

```
time_wearing_device <- daily_activity_new %>%
  mutate(total_time_worn = very_active_minutes + fairly_active_minutes
+ lightly_active_minutes + sedentary_minutes) %>%
  select(id, date, total_time_worn) %>%
  mutate(percent_of_day_worn = total_time_worn / 1440)

time_worn_stats <- time_wearing_device %>%
  mutate(category = case_when(
    percent_of_day_worn > 0 & percent_of_day_worn <= 0.50 ~ 'Less Than
12 Hours',
    percent_of_day_worn > 0.50 & percent_of_day_worn <= 0.75 ~ '12 to
18 Hours',
    percent_of_day_worn > 0.75 ~ 'All Day'))

time_worn_categories <- time_worn_stats %>%
  group_by(category) %>%
  summarize(total = n()) %>%
  mutate(percent_of_users = total / sum(total)) %>%
  mutate(chart_labels = scales::percent(percent_of_users))

time_worn_categories$category <- ordered(time_worn_categories$category,
levels=c("Less Than 12 Hours", "12 to 18 Hours", "All Day"))
```

	category	total	percent_of_users	chart_labels
1	Less Than 12 Hours	22	0.02549247	3%
2	12 to 18 Hours	346	0.40092700	40%
3	All Day	495	0.57358053	57%

I then ran an analysis to check the percentage of time users spent using their devices and categorised them into three buckets. My results showed me that most users were wearing their devices all day, whereas only a small portion were using their device for under 12 hours per day. That said, there is still a large cohort using their devices between 12 – 18 hours a day and hence may not be getting the full range of benefits from their device. Couple that with the known discrepancy between the number of users logging information for daily activity versus weight and sleep as we saw earlier, we can see that some users are losing out. This again serves to highlight the challenges with this dataset and the insights that can be derived.

4. Act

Revisiting Business Task

The business task was to analyse smart device usage data of non-Bellabeat smart devices to gain insight into relevant (successful and unsuccessful) consumer trends within the global smart device market, as well as to discover how to use these trends to apply to Bellabeat customers and to influence future Bellabeat marketing strategies. This is done by applying said insights to the Bellabeat App and to future products to maximise profits and growth for the company and to capitalise on Bellabeat's rapidly growing consumer base in the smart device/tech-wellness space.

Recommendations

- Bellabeat should offer incentives for consistent tracking / usage, like in-app competitions against friends or other users in the same city/state.
    - Bellabeat could offer prizes or points, which can be redeemed for in-app features, exclusive access and discounts to future products etc.
    - Bellabeat could offer greater incentives (i.e., more points) from Friday-Monday, or during evenings each weekday since a lot of people lose motivation or consistency during this time of the week.
- Bellabeat products should have a built in TDEE calculator, where the users are able to input their sex, age, weight, height, and other information for more user specific and relevant tracking.
    - This calculator will notify the user of what their maintenance calories are (and their macros) and how much of a caloric deficit the user needs to be in each day to lose based upon their weight goals and time frame.
    - The user would also be notified if they are reaching, have reached, or have passed their daily caloric intake.
    - The user would also be notified when they've reached their suggested (or inputted) weight goal or are on the right track.
    - The Bellabeat app could also provide nutritional advice as part of its membership, which goes into detail about healthy recipes and managing macros.
    - The app could have a list of activities and videos of said activities that people can do to burn some quick calories (since the average person is very sedentary and might not have a lot of time to spend hours in the gym, this would be a good incentive to exercise and burn a lot of calories in a short period of time).
- Bellabeat products should be able to track sleep automatically, since users struggled to input their time consistently.
    - Bellabeat could implement a Leaf or an app notification to notify the user of the ideal time to sleep as per the user's schedule.
    - The user would be notified an hour or two before bed to start winding down from using electronics that use blue light (it could even automatically switch the phone to night mode to prevent blue light exposure).
    - The user would be notified about 30 minutes prior to bed time to start winding down and stop using their devices.