# Bellabeat Case Study with R

By Sadia Tanjim

### Introduction

Welcome to my Bellabeat data analysis case study. In this case study, I will perform real-world tasks of a data analyst. In order to answer some key business questions, I will follow the steps of the data analysis process: ask, prepare, process, analyze, share, and act.

### About the Bellabeat Company

Bellabeat, is a high-tech manufacturer of health-focused products for women. Bellabeat is a successful small company, and they have the potential to become a larger player in the global smart device market. Urška Sršen and Sando Mur founded Bellabeat, a high-tech company that manufactures health-focused smart products. Collecting data on activity, sleep, stress, and reproductive health has allowed Bellabeat to empower women with knowledge about their own health and habits. Since it was founded in 2013, Bellabeat has grown rapidly and quickly positioned itself as a tech-driven wellness company for women.

### Scenario of the Study

In this study, I will 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 then help guide marketing strategy for the company.

### Questions for the Analysis (Ask phase)

In this phase, I tried to better understand the data and the problem I'm trying to solve. And to do that, I had to do more research and ask more questions.

- **What are some trends in smart device usage? How could these trends apply to Bellabeat customers? How could these trends help influence Bellabeat marketing strategy?** So first, the company need to better target their marketing efforts into their customer's needs based on their usage of their fitness smart devices. And then, make high-level recommendations for how these trends can inform Bellabeat marketing strategy.
- **Who are the main stakeholders?** The main stakeholders are Urška Sršen, Bellabeat's co-founder and Chief Creative Officer; Sando Mur, Mathematician and Bellabeat's co founder; And also, we need to think about and work with the rest of the Bellabeat marketing analytics team.

### Business Task

Now, after getting answers to all of my questions (during the ask phase), I'm able to define clearly the business task which is: *Analyze customers' use of an existing competitor to identify potential opportunities for growth and recommendations for the Bellabeat marketing strategy based on trends in smart device usage.*

### Preparing the Data (Prepare Phase)

**Description of the data source** The dataset is generated by respondents to a distributed survey via Amazon Mechanical Turk between 03.12.2016 and 05.12.2016 and include 18 CSV files. Thirty eligible Fitbit users consented to the submission of personal tracker data, including minute-level output for physical activity, heart rate, and sleep monitoring. Variation between output represents use of different types of Fitbit trackers and individual tracking behaviors / preferences.

The key assumption of using this dataset is that the Fitbit users are representative users of Leaf. However, this is not necessarily the case. The target users of Bellabeat are women, while Fitbit users includes both genders. In this way, the data collected is not completely applicable to Bellabeat users. Also, the sample size is too small to make a determining decision.

However, overall, the dataset is a good data source because it is:

- Reliable: Accurate, complete, unbiased information that is fit for use
- Original: Not through a second or third party source
- Comprehensive: Contain all critical information needed to answer the questions
- Current: Current and relevant to the task at hand
- Cited: Explicitly cited and credible

**Downloading the data** Here is the link to download the dataset:

- FitBit Fitness Tracker Data : https://www.kaggle.com/arashnic/fitbit

```r


# Installing packages :
install.packages("tidyverse")
install.packages("lubridate")
install.packages("dplyr")
install.packages("ggplot2")
install.packages("tidyr")
install.packages("here")
install.packages("skimr")
install.packages("janitor")
```

#### Installing packages

**Loading packages** Now, I'm going to load these packages. And I'm using in my R code (valid for RStudio only) the options message=FALSE and warning=FALSE, to save space. And to prevent printing of the execution of the R code generated and the warning messages.

```r
# Loading packages :

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

**Importing dataset** Now, I'm going to Import all dataset.

```
#for PDF
knitr::opts_chunk$set(cache = TRUE)
```

```
load("Fitabase Data 4.12.16-5.12.16/Bellabeat.RData")
```

```
#for Word
#load("~/filename.RData")
```

Then VIEW, CLEAN, FORMAT, and ORGANIZE the data. After reviewing all the dataset, I decided to make some assumptions and work only with these data for my analysis:

- **dailyActivity_merged.csv**

```
# Importing Activity dataset :

Activity <- read.csv("Fitabase Data 4.12.16-5.12.16/dailyActivity_merged.csv")
head(Activity)
```

```
##           Id ActivityDate TotalSteps TotalDistance TrackerDistance
## 1 1503960366    4/12/2016      13162          8.50            8.50
## 2 1503960366    4/13/2016      10735          6.97            6.97
## 3 1503960366    4/14/2016      10460          6.74            6.74
## 4 1503960366    4/15/2016       9762          6.28            6.28
## 5 1503960366    4/16/2016      12669          8.16            8.16
## 6 1503960366    4/17/2016       9705          6.48            6.48
##   LoggedActivitiesDistance VeryActiveDistance ModeratelyActiveDistance
## 1                        0               1.88                     0.55
## 2                        0               1.57                     0.69
## 3                        0               2.44                     0.40
## 4                        0               2.14                     1.26
## 5                        0               2.71                     0.41
## 6                        0               3.19                     0.78
##   LightActiveDistance SedentaryActiveDistance VeryActiveMinutes
## 1                6.06                       0                25
## 2                4.71                       0                21
## 3                3.91                       0                30
## 4                2.83                       0                29
## 5                5.04                       0                36
## 6                2.51                       0                38
##   FairlyActiveMinutes LightlyActiveMinutes SedentaryMinutes Calories
## 1                  13                  328              728     1985
## 2                  19                  217              776     1797
## 3                  11                  181             1218     1776
## 4                  34                  209              726     1745
## 5                  10                  221              773     1863
## 6                  20                  164              539     1728
```

```
colnames(Activity)
```

```
##  [1] "Id"                       "ActivityDate"
##  [3] "TotalSteps"               "TotalDistance"
##  [5] "TrackerDistance"          "LoggedActivitiesDistance"
##  [7] "VeryActiveDistance"       "ModeratelyActiveDistance"
##  [9] "LightActiveDistance"      "SedentaryActiveDistance"
## [11] "VeryActiveMinutes"        "FairlyActiveMinutes"
## [13] "LightlyActiveMinutes"     "SedentaryMinutes"
## [15] "Calories"
```

```
str(Activity)
```

```
## 'data.frame':    940 obs. of  15 variables:
##  $ Id                      : num  1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
##  $ ActivityDate            : chr  "4/12/2016" "4/13/2016" "4/14/2016" "4/15/2016" ...
##  $ TotalSteps              : int  13162 10735 10460 9762 12669 9705 13019 15506 10544 9819 ...
##  $ TotalDistance           : num  8.5 6.97 6.74 6.28 8.16 ...
##  $ TrackerDistance         : num  8.5 6.97 6.74 6.28 8.16 ...
##  $ LoggedActivitiesDistance: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ VeryActiveDistance      : num  1.88 1.57 2.44 2.14 2.71 ...
##  $ ModeratelyActiveDistance: num  0.55 0.69 0.4 1.26 0.41 ...
##  $ LightActiveDistance     : num  6.06 4.71 3.91 2.83 5.04 ...
##  $ SedentaryActiveDistance : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ VeryActiveMinutes       : int  25 21 30 29 36 38 42 50 28 19 ...
##  $ FairlyActiveMinutes     : int  13 19 11 34 10 20 16 31 12 8 ...
##  $ LightlyActiveMinutes    : int  328 217 181 209 221 164 233 264 205 211 ...
##  $ SedentaryMinutes        : int  728 776 1218 726 773 539 1149 775 818 838 ...
##  $ Calories                : int  1985 1797 1776 1745 1863 1728 1921 2035 1786 1775 ...
```

- **dailyCalories_merged.csv**

```
# Importing Calories dataset !
Calories <- read.csv("Fitabase Data 4.12.16-5.12.16/dailyCalories_merged.csv")
head(Calories)
```

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

```
colnames(Calories)
```

```
## [1] "Id"          "ActivityDay" "Calories"
```

```
str(Calories)
```

```
## 'data.frame':    940 obs. of  3 variables:
##  $ Id         : num  1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
##  $ ActivityDay: chr  "4/12/2016" "4/13/2016" "4/14/2016" "4/15/2016" ...
##  $ Calories   : int  1985 1797 1776 1745 1863 1728 1921 2035 1786 1775 ...
```

- **dailyIntensities_merged.csv**

```
# Importing Intensities dataset !
Intensities <- read.csv("Fitabase Data 4.12.16-5.12.16/dailyIntensities_merged.csv")
head(Intensities)
```

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

```
colnames(Intensities)
```

```
##  [1] "Id"                       "ActivityDay"
##  [3] "SedentaryMinutes"         "LightlyActiveMinutes"
##  [5] "FairlyActiveMinutes"      "VeryActiveMinutes"
##  [7] "SedentaryActiveDistance"  "LightActiveDistance"
##  [9] "ModeratelyActiveDistance" "VeryActiveDistance"
```

```
str(Intensities)
```

```
## ’data.frame':    940 obs. of  10 variables:
##  $ Id                      : num  1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
##  $ ActivityDay             : chr  "4/12/2016" "4/13/2016" "4/14/2016" "4/15/2016" ...
##  $ SedentaryMinutes        : int  728 776 1218 726 773 539 1149 775 818 838 ...
##  $ LightlyActiveMinutes    : int  328 217 181 209 221 164 233 264 205 211 ...
##  $ FairlyActiveMinutes     : int  13 19 11 34 10 20 16 31 12 8 ...
##  $ VeryActiveMinutes       : int  25 21 30 29 36 38 42 50 28 19 ...
##  $ SedentaryActiveDistance : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ LightActiveDistance     : num  6.06 4.71 3.91 2.83 5.04 ...
##  $ ModeratelyActiveDistance: num  0.55 0.69 0.4 1.26 0.41 ...
##  $ VeryActiveDistance      : num  1.88 1.57 2.44 2.14 2.71 ...
```

- **heartrate_seconds_merged.csv**

```
# Importing Heartrate dataset !
Heartrate <- read.csv("Fitabase Data 4.12.16-5.12.16/heartrate_seconds_merged.csv")
head(Heartrate)
```

```
##           Id                 Time Value
## 1 2022484408 4/12/2016 7:21:00 AM    97
## 2 2022484408 4/12/2016 7:21:05 AM   102
## 3 2022484408 4/12/2016 7:21:10 AM   105
## 4 2022484408 4/12/2016 7:21:20 AM   103
## 5 2022484408 4/12/2016 7:21:25 AM   101
## 6 2022484408 4/12/2016 7:22:05 AM    95
```

```
colnames(Heartrate)
```

```
## [1] "Id"    "Time"  "Value"
```

```
str(Heartrate)
```

```
## 'data.frame':    2483658 obs. of  3 variables:
##  $ Id   : num  2.02e+09 2.02e+09 2.02e+09 2.02e+09 2.02e+09 ...
##  $ Time : chr  "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:
##  $ Value: int  97 102 105 103 101 95 91 93 94 93 ...
```

- **sleepDay_merged.csv**

```
# Importing Sleep dataset !
Sleep <- read.csv("Fitabase Data 4.12.16-5.12.16/sleepDay_merged.csv")
head(Sleep)
```

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

```
colnames(Sleep)
```

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

```
str(Sleep)
```

```
## 'data.frame':    413 obs. of  5 variables:
##  $ Id                : num  1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
##  $ SleepDay          : chr  "4/12/2016 12:00:00 AM" "4/13/2016 12:00:00 AM" "4/15/2016 12:00:00 AM"
##  $ TotalSleepRecords : int  1 2 1 2 1 1 1 1 1 1 ...
##  $ TotalMinutesAsleep: int  327 384 412 340 700 304 360 325 361 430 ...
##  $ TotalTimeInBed    : int  346 407 442 367 712 320 377 364 384 449 ...
```

- **weightLogInfo_merged.csv**

```
# Importing Weight dataset !
Weight <- read.csv("Fitabase Data 4.12.16-5.12.16/weightLogInfo_merged.csv")
head(Weight)
```

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

```
colnames(Weight)
```

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

```
str(Weight)
```

```
## 'data.frame':    67 obs. of  8 variables:
##  $ Id            : num  1.50e+09 1.50e+09 1.93e+09 2.87e+09 2.87e+09 ...
##  $ Date          : chr  "5/2/2016 11:59:59 PM" "5/3/2016 11:59:59 PM" "4/13/2016 1:08:52 AM" "4/21/2
##  $ WeightKg      : num  52.6 52.6 133.5 56.7 57.3 ...
##  $ WeightPounds  : num  116 116 294 125 126 ...
##  $ Fat           : int  22 NA NA NA NA 25 NA NA NA NA ...
##  $ BMI           : num  22.6 22.6 47.5 21.5 21.7 ...
##  $ IsManualReport: chr  "True" "True" "False" "True" ...
##  $ LogId         : num  1.46e+12 1.46e+12 1.46e+12 1.46e+12 1.46e+12 ...
```

### Cleaning the dataset (Process Phase)

**Basic cleaning :** Now, I'm going to Process, Clean and Organize the dataset for analysis. I used functions like glimpse(), skim_without_charts to quickly review the data. I also clean the names of the data using clean_names().

And here some cleaning steps I did with the data :

- For Dataset (Activity, Calories and Intensities): For the data cleaning steps, I did NOT FOUND in this data (Spelling errors, Misfiled values, Missing values, Extra and blank space, no duplicated found). For Data types, some data were converted to numeric and Dates columns will be converted to date type.
- For Sleep data : 3 duplicates were found and removed.
- For Weight data : Too many missing values were found in one column. And I decided to remove that column.

### Getting to know data and clean column names

```
# Activity
skim_without_charts(Activity)
Activity_new <- Activity
clean_names(Activity_new)
```

```
# Calories
skim_without_charts(Calories)
Calories_new <- Calories
clean_names(Calories_new)
```

```
# Intensities
skim_without_charts(Intensities)
Intensities_new <- Intensities
clean_names(Intensities_new)
```

```
# Heartrate
skim_without_charts(Heartrate)
Heartrate_new <- Heartrate
clean_names(Heartrate_new)
```

```
# Sleep
skim_without_charts(Sleep)
Sleep_new <- Sleep
clean_names(Sleep_new)
view(Sleep_new)
```

```
# Weight
skim_without_charts(Weight)
Weight_new <- Weight
clean_names(Weight_new)
```

### Findings duplicates and remove duplicates

```
# Activity_new
duplicated(Activity_new)

# Calories_new
duplicated(Calories_new)
```

```
# Intensities_new
duplicated(Intensities_new)

# Heartrate_new
duplicated(Heartrate_new)

# Sleep_new
duplicated(Sleep_new) # found 3 duplicate
Sleep_new <- unique(Sleep_new) # remove duplicates
duplicated(Sleep_new)

# Weight_new
duplicated(Weight_new)
```

### Count the number of NA values and remove NA values

```
# Activity_new
sum(is.na(Activity_new))

# Calories_new
sum(is.na(Calories_new))

# Intensities_new
sum(is.na(Intensities_new))

# Heartrate_new
sum(is.na(Heartrate_new))

# Sleep_new
sum(is.na(Sleep_new))

# Weight_new
sum(is.na(Weight_new))  # Found 65 values
Weight_new <- select(Weight_new, -Fat) #remove the column
view(Weight_new)
```

### Organizing data : Fixing formatting

I spotted some problems with the time stamp data. So before analysis, I need to convert it to date time format and split to date and time.

```
# For Activity_new

# Convert character data to date and time
Activity_new$ActivityDate=as.POSIXct(Activity_new$ActivityDate, format="%m/%d/%Y", tz=Sys.timezone())

# setting format to specify the style
Activity_new$date <- format(Activity_new$ActivityDate, format = "%m/%d/%y")

# convert column to date class using as.date method/ save date in date format
Activity_new$ActivityDate=as.Date(Activity_new$ActivityDate, format="%m/%d/%Y", tz=Sys.timezone())
```

```r
# convert column to date class using as.date method and split to date
Activity_new$date=as.Date(Activity_new$date, format="%m/%d/%Y")
View(Activity_new)
```

```r
# For Intensities_new

Intensities_new$ActivityDay=as.Date(Intensities_new$ActivityDay, format="%m/%d/%Y", tz=Sys.timezone())
View(Intensities_new)
```

```r
# For Sleep_new

Sleep_new$SleepDay=as.POSIXct(Sleep_new$SleepDay, format="%m/%d/%Y %I:%M:%S %p", tz=Sys.timezone())
Sleep_new$date <- format(Sleep_new$SleepDay, format = "%m/%d/%y")
Sleep_new$date=as.Date(Sleep_new$date, "% m/% d/% y")
View(Sleep_new)
```

### Changing the Data type

Change the data type of Id (numeric) to Character

```r
# Activity_new
Activity_new$Id = as.character(Activity_new$Id)
str(Activity_new)

# Calories_new
Calories_new$Id = as.character(Calories_new$Id)
str(Calories_new)

# Intensities_new
Intensities_new$Id = as.character(Intensities_new$Id)
str(Intensities_new)

#Heartrate_new
Heartrate_new$Id = as.character(Heartrate_new$Id)
str(Heartrate_new)

# Sleep_new
Sleep_new$Id = as.character(Sleep_new$Id)
str(Sleep_new)

# Weight_new
Weight_new$Id = as.character(Weight_new$Id)
str(Weight_new)
```

### Checking the Id length

Since the function is not showing any errors, all ID lengths = 10

```r
# Activity_new
id_lens <- nchar(Activity_new$Id)  # Count Id lengths
id_lens_bool <- id_lens == 10      # True/False check if id == 10 or not
stopifnot(id_lens_bool)         # This function will show Error if not all TRUE
```

```
# Since the function is not showing any errors, all ID lengths = 10

# Calories_new
id_lens <- nchar(Calories_new$Id)
id_lens_bool <- id_lens == 10
stopifnot(id_lens_bool)

# Intensities_new
id_lens <- nchar(Intensities_new$Id)
id_lens_bool <- id_lens == 10
stopifnot(id_lens_bool)

# Heartrate_new
id_lens <- nchar(Heartrate_new$Id)
id_lens_bool <- id_lens == 10
stopifnot(id_lens_bool)

# Sleep_new
id_lens <- nchar(Sleep_new$Id)
id_lens_bool <- id_lens == 10
stopifnot(id_lens_bool)

# Weight_new
id_lens <- nchar(Weight_new$Id)
id_lens_bool <- id_lens == 10
stopifnot(id_lens_bool)
```

### Summarizing the dataset (Analyze Phase)

Now that all the data is stored appropriately and has been prepared for analysis, I can start putting it to work.

**Let's look at the total number of participants in each data sets:**

```
##   Activity_participants
## 1                    33
```

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

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

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

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

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

**How many observations are there in each data frame?**

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

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

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

```
## [1] 410
```

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

So, there are 33 participants in the activity, calories and intensities data sets. 24 participants in the Sleep data. And only 14 participants for Heartrate, and only 8 in the weight data set. 8 and 14 participants are not significant to make any recommendations and conclusions based on these dataset.

So I will focus on these datasets for my analysis: Activity, Calories, Intensities and Sleep.

**Here are some quick summary statistics about each data frame.**

- For the Activity_new data frame :

```
# Activity_new
Activity_new %>%
  select(TotalSteps,
         TotalDistance,
         VeryActiveMinutes,
         SedentaryMinutes, Calories) %>%
  summary()
```

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

- For the Calories_new data frame :

```
# Calories_new
Calories_new %>%
  select(Calories) %>%
  summary()
```

```
##     Calories
##  Min.   :   0
##  1st Qu.:1828
##  Median :2134
##  Mean   :2304
##  3rd Qu.:2793
##  Max.   :4900
```

- For the Intensities_new data frame :

```
# Intensities_new
Intensities_new %>%
  select(VeryActiveMinutes,
         FairlyActiveMinutes,
         LightlyActiveMinutes,
         SedentaryMinutes) %>%
  summary()
```

```
##  VeryActiveMinutes FairlyActiveMinutes LightlyActiveMinutes SedentaryMinutes
##  Min.   :  0.00    Min.   :  0.00      Min.   :  0.0        Min.   :   0.0
##  1st Qu.:  0.00    1st Qu.:  0.00      1st Qu.:127.0        1st Qu.: 729.8
##  Median :  4.00    Median :  6.00      Median :199.0        Median :1057.5
##  Mean   : 21.16    Mean   : 13.56      Mean   :192.8        Mean   : 991.2
##  3rd Qu.: 32.00    3rd Qu.: 19.00      3rd Qu.:264.0        3rd Qu.:1229.5
##  Max.   :210.00    Max.   :143.00      Max.   :518.0        Max.   :1440.0
```

- For the Sleep_new data frame :

```
# Sleep_new
Sleep_new %>%
  select(TotalSleepRecords,
         TotalMinutesAsleep,
         TotalTimeInBed) %>%
  summary()
```

```
##  TotalSleepRecords TotalMinutesAsleep TotalTimeInBed
##  Min.   :1.00      Min.   : 58.0      Min.   : 61.0
##  1st Qu.:1.00      1st Qu.:361.0      1st Qu.:403.8
##  Median :1.00      Median :432.5      Median :463.0
##  Mean   :1.12      Mean   :419.2      Mean   :458.5
##  3rd Qu.:1.00      3rd Qu.:490.0      3rd Qu.:526.0
##  Max.   :3.00      Max.   :796.0      Max.   :961.0
```

- For the Weight_new data frame :

```
# Weight_new
Weight_new %>%
  select(WeightKg, BMI) %>%
  summary()
```

```
##     WeightKg           BMI
##  Min.   : 52.60   Min.   :21.45
##  1st Qu.: 61.40   1st Qu.:23.96
##  Median : 62.50   Median :24.39
##  Mean   : 72.04   Mean   :25.19
##  3rd Qu.: 85.05   3rd Qu.:25.56
##  Max.   :133.50   Max.   :47.54
```

### KEY FINDINGS from this analysis(CREATE A TABLE) :

- Average total steps per day (which is 7638) is a little bit less than recommended by the CDC (8000).
- The average sedentary time is very high (more than 16 hours).
- Average Calorie is little bit higher . Recommended daily calorie intakes in the US are around 2500 per day for men and 2000 for women.
- The majority of the participants are lightly active with a high sedentary time.
- Participants spend more time in bed than total minutes asleep, on average.

### Data Visualization (Share Phase)

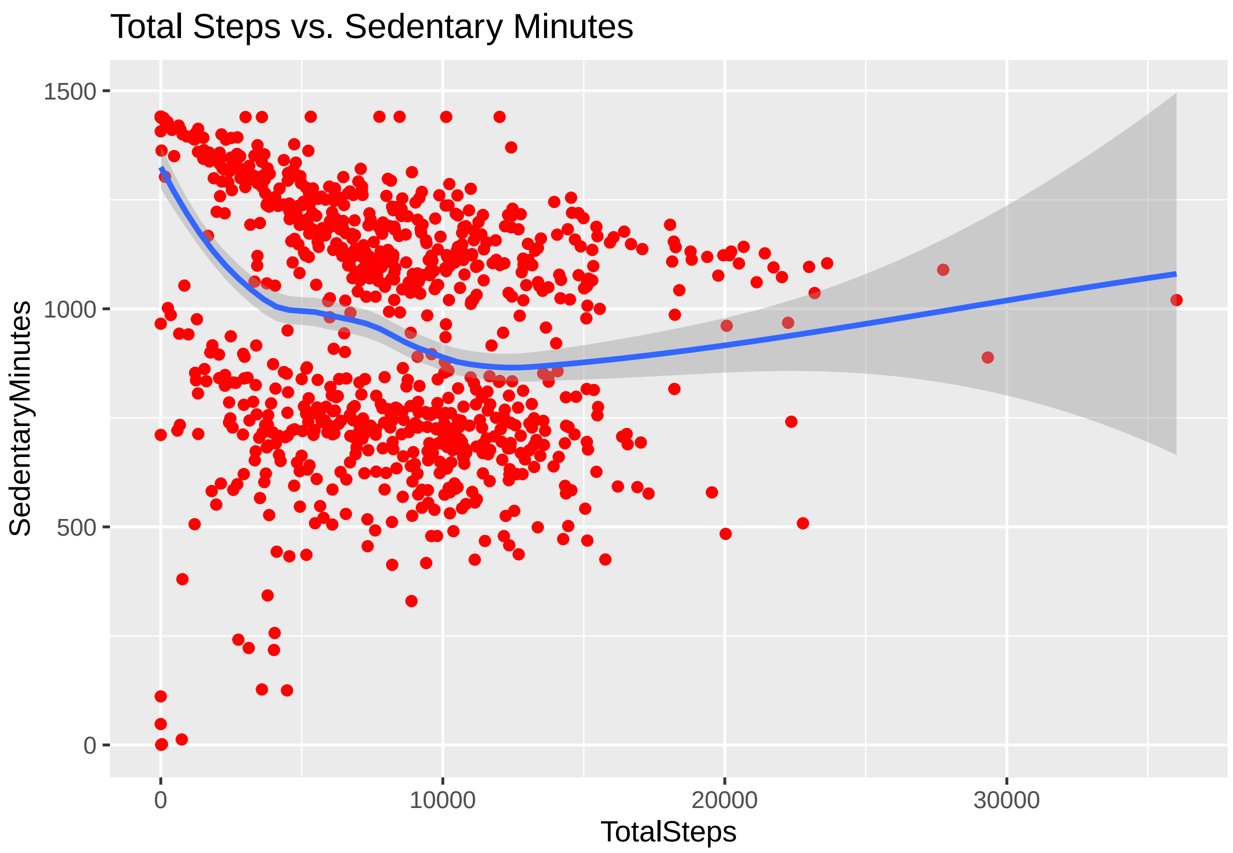
Once I have completed my analysis, create my data visualizations. The visualizations should clearly communicate high-level insights and recommendations.

### Plotting few explorations :

- ***For Activity_new***

```
require(ggplot2)
ggplot(data=Activity_new) +
  geom_jitter(mapping=aes(x=TotalSteps, y=SedentaryMinutes),color="red") +
  geom_smooth(mapping=aes(x=TotalSteps, y=SedentaryMinutes), formula = 'y ~ x', method ='loess') +
  labs(title="Total Steps vs. Sedentary Minutes")
```

### Relationship between Steps and Sedentary time



We can see a negative correlation between Total Steps and Sedentary Minutes. It is evident that people logging more sedentary time are less likely to take part in physical activity like walking.

```
corr_01 <- cor.test(Activity_new$TotalSteps, Activity_new$SedentaryMinutes,
                     method = "pearson")

corr_01 # (-0.3274835)
```

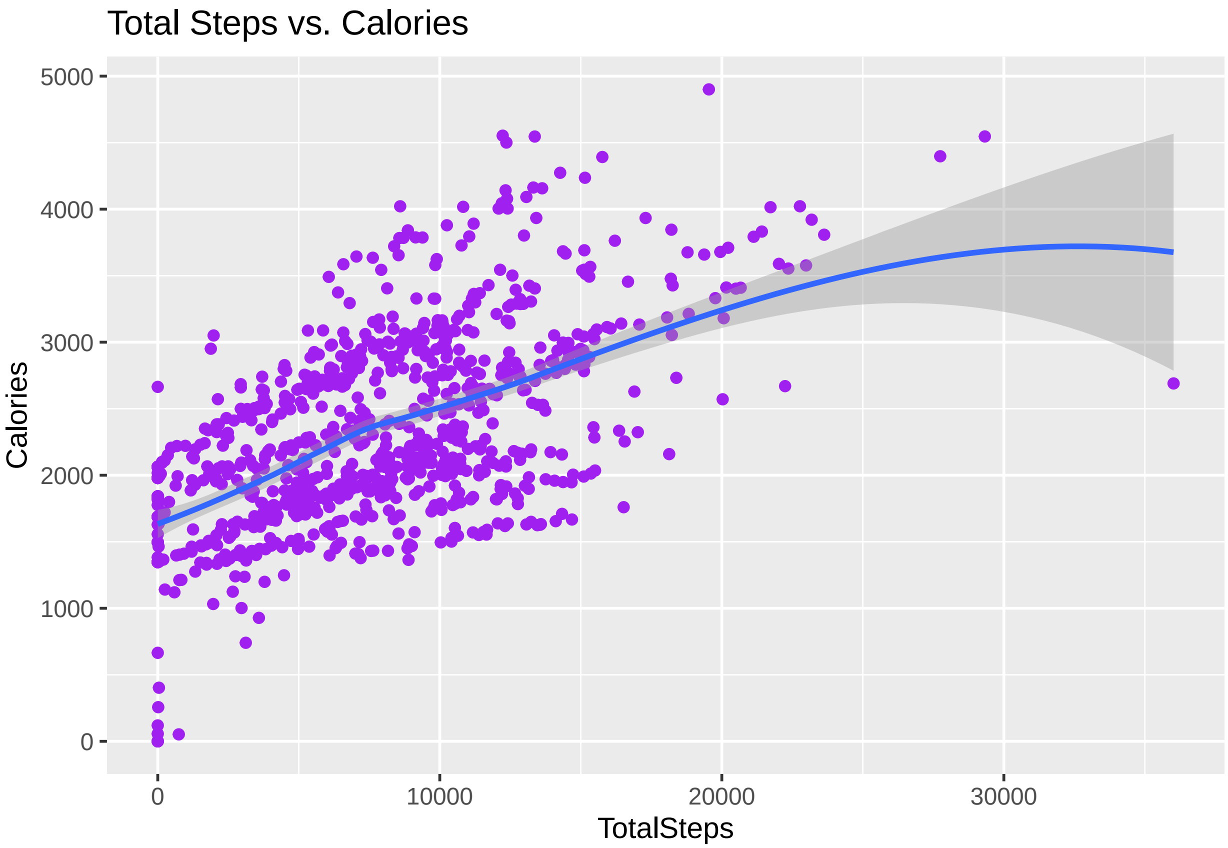
**Correlation test**

```
##
##  Pearson's product-moment correlation
##
## data:  Activity_new$TotalSteps and Activity_new$SedentaryMinutes
## t = -10.615, df = 938, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.3833971 -0.2691782
## sample estimates:
##        cor
## -0.3274835
```

I can see here a negative correlation between Steps and Sedentary time.

```
ggplot(data=Activity_new) +
  geom_jitter(mapping=aes(x=TotalSteps, y=Calories), color='purple') +
  geom_smooth(mapping=aes(x=TotalSteps, y=Calories), formula = 'y ~ x', method ='loess') +
  labs(title="Total Steps vs. Calories")
```

**Relationship between Steps and Calories**



We can see a positive correlation between total steps and calories burned. It is evident that people logging in more steps are physically more active hence they are able to burn more calories everyday.

```
corr_02 <- cor.test(Activity_new$TotalSteps, Activity_new$Calories,
                    method = "pearson")

corr_02 # (0.5915681)
```

**Correlation test**

```
##
##  Pearson's product-moment correlation
##
```

```
## data:  Activity_new$TotalSteps and Activity_new$Calories
## t = 22.472, df = 938, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5483688 0.6316184
## sample estimates:
##       cor
## 0.5915681
```

positive correlation between Total Steps and Calories.

```r
# run this chunk if '%>%' is not found

install.packages("magrittr") # package installations are only needed the first time you use it
install.packages("dplyr")    # alternative installation of the %>%
library(magrittr) # needs to be run every time you start R and want to use %>%
library(dplyr)    # alternatively, this also loads %>%
```

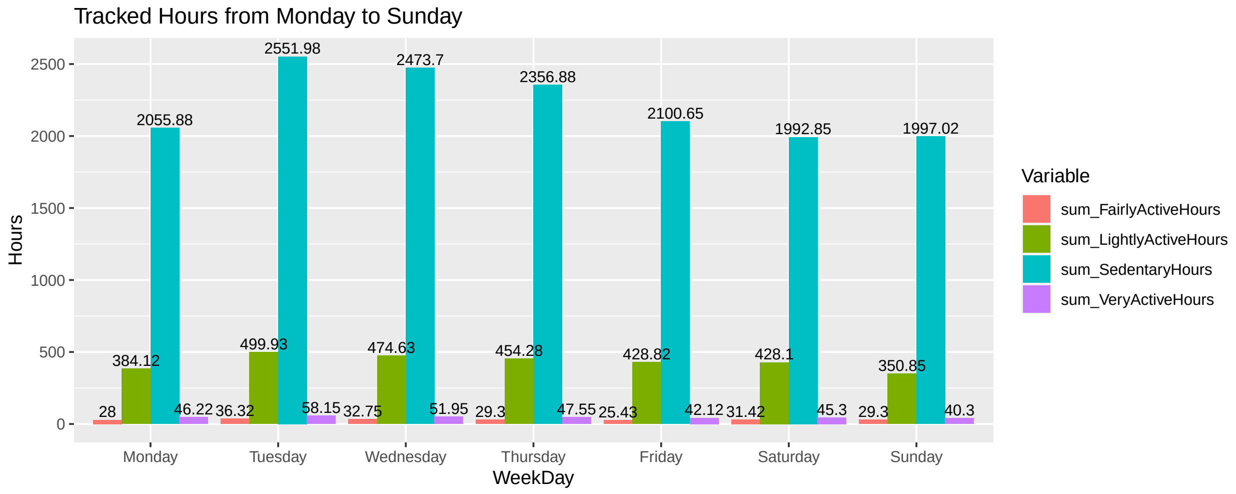
```r
library(dplyr)
df <- Activity_new %>%
  select(ActivityDate,
         VeryActiveMinutes,
         FairlyActiveMinutes,
         LightlyActiveMinutes,
         SedentaryMinutes)
head(df)
```

```r
df$WeekDay <- weekdays(df$ActivityDate)
df$WeekDay <- factor(df$WeekDay, levels = c("Monday", "Tuesday",
                                                 "Wednesday", "Thursday", "Friday",
                                                 "Saturday", "Sunday"))

df[order(df$WeekDay), ]


df$VeryActiveHours <- (df$VeryActiveMinutes)/60
df$FairlyActiveHours <- (df$FairlyActiveMinutes)/60
df$LightlyActiveHours <- (df$LightlyActiveMinutes)/60
df$SedentaryHours <- (df$SedentaryMinutes)/60


df05 <- df %>%
  select(WeekDay, VeryActiveHours, FairlyActiveHours,
         LightlyActiveHours, SedentaryHours) %>%
  group_by(WeekDay) %>%
  summarize(sum_SedentaryHours = sum(SedentaryHours),
            sum_VeryActiveHours = sum(VeryActiveHours),
            sum_FairlyActiveHours = sum(FairlyActiveHours),
            sum_LightlyActiveHours = sum(LightlyActiveHours))



head(df05)
```

```
library(tidyr)
library(ggplot2)
```

```
ggplot(data = df05 %>% gather(Variable, Hours, -WeekDay),
       aes(x = WeekDay, y = Hours, fill = Variable)) +
  geom_bar(stat = 'identity', position ='dodge') +
  geom_text(aes(label=round(Hours, digits = 2)),
            position = position_dodge(width = 0.9), vjust=-0.25, size=3)+
  scale_y_continuous(breaks = scales :: pretty_breaks(n=10)) +
  labs(title="Tracked Hours from Monday to Sunday")
```

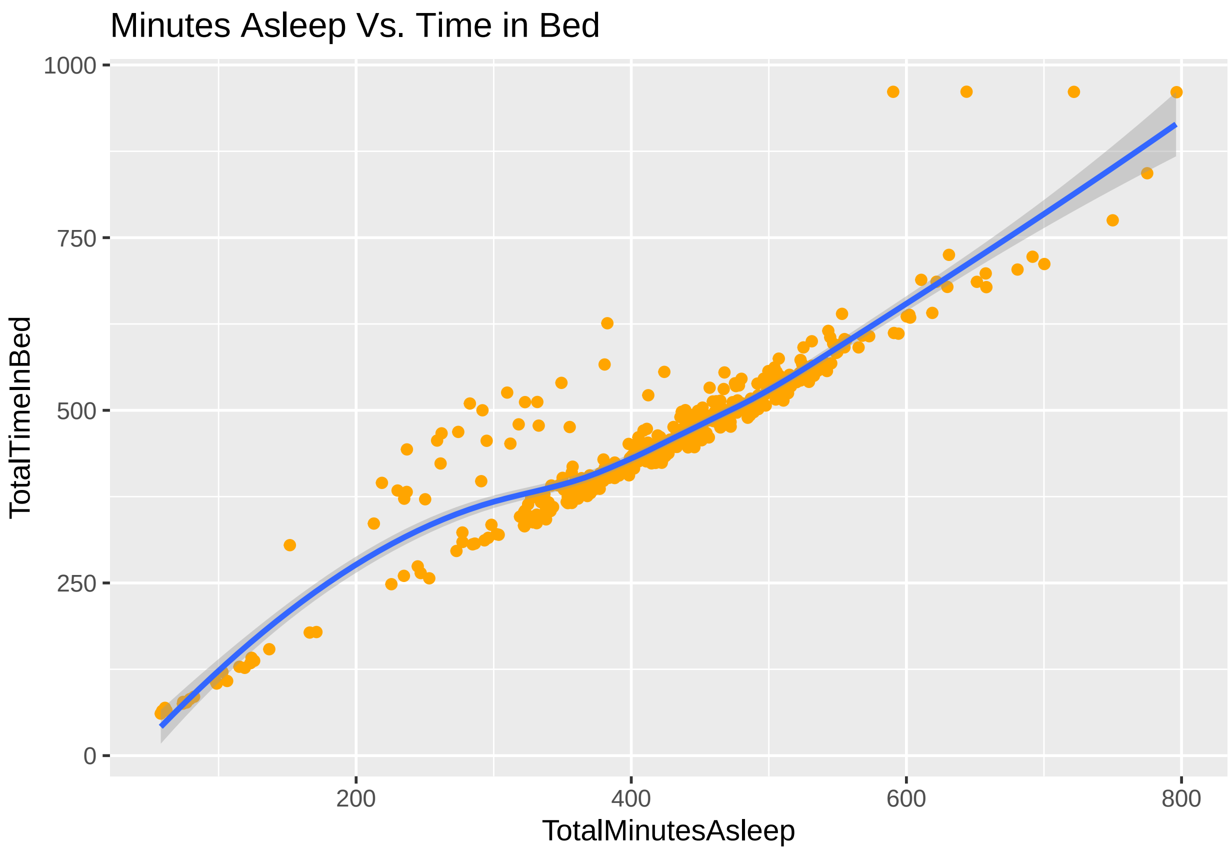
### Time of different Activity types



The average user spends a lot of sedentary time.The number of sedentary hours is lower during the weekends and at the start of the week. However, it sees an increasing trend during the middle of the week for example during Tuesday/Wednesday/Thursday. This might be due to work stress or users simply forgetting to workout.The users active time is also not consistent throughout the week. Bellabeat users might be encouraged to track their active/sedentary hours throughout the the week and reminded to take some time to relieve stress and encourage some light activity mid-week.

```
ggplot(data = Sleep_new) +
  geom_jitter(mapping=aes(x=TotalMinutesAsleep, y=TotalTimeInBed), color="orange") +
  geom_smooth(mapping=aes(x=TotalMinutesAsleep, y=TotalTimeInBed), method='loess', formula='y ~ x') +
  labs(title="Minutes Asleep Vs. Time in Bed")
```

### Relationship between Minutes Asleep and Time in Bed



Participants spend more time in bed than actual time asleep. Sleep quality and sleeping habits are not ideal.

```
corr_03 <- cor.test(Sleep_new$TotalMinutesAsleep, Sleep_new$TotalTimeInBed,
                    method = "pearson")

corr_03 # (0.9304224
```

### Correlation test

```
##
##  Pearson's product-moment correlation
##
## data:  Sleep_new$TotalMinutesAsleep and Sleep_new$TotalTimeInBed
## t = 51.28, df = 408, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
```

```
##  0.9161262 0.9423551
## sample estimates:
##       cor
## 0.9304224
```

I can see a strong positive correlation between total minutes asleep and total time in bed.

### Merging some data :

- ***Activity_new and Sleep_new data merging***

I'm going to merge two data sets : Activity and Sleep data on columns Id. Note that there are more participant Ids in the Activity dataset than in the Sleep dataset. So if I use the merge option inner_joint, then I will have the number of participants from the Sleep data set.

```
Combined_inner <- merge(Activity_new, Sleep_new, by="Id")
n_distinct(Combined_inner$Id)
```

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

So for analysis, I will consider using 'outer_join' to keep all participants in the in the dataset. And I can do that by adding in my code chunk the extra argument all=TRUE.

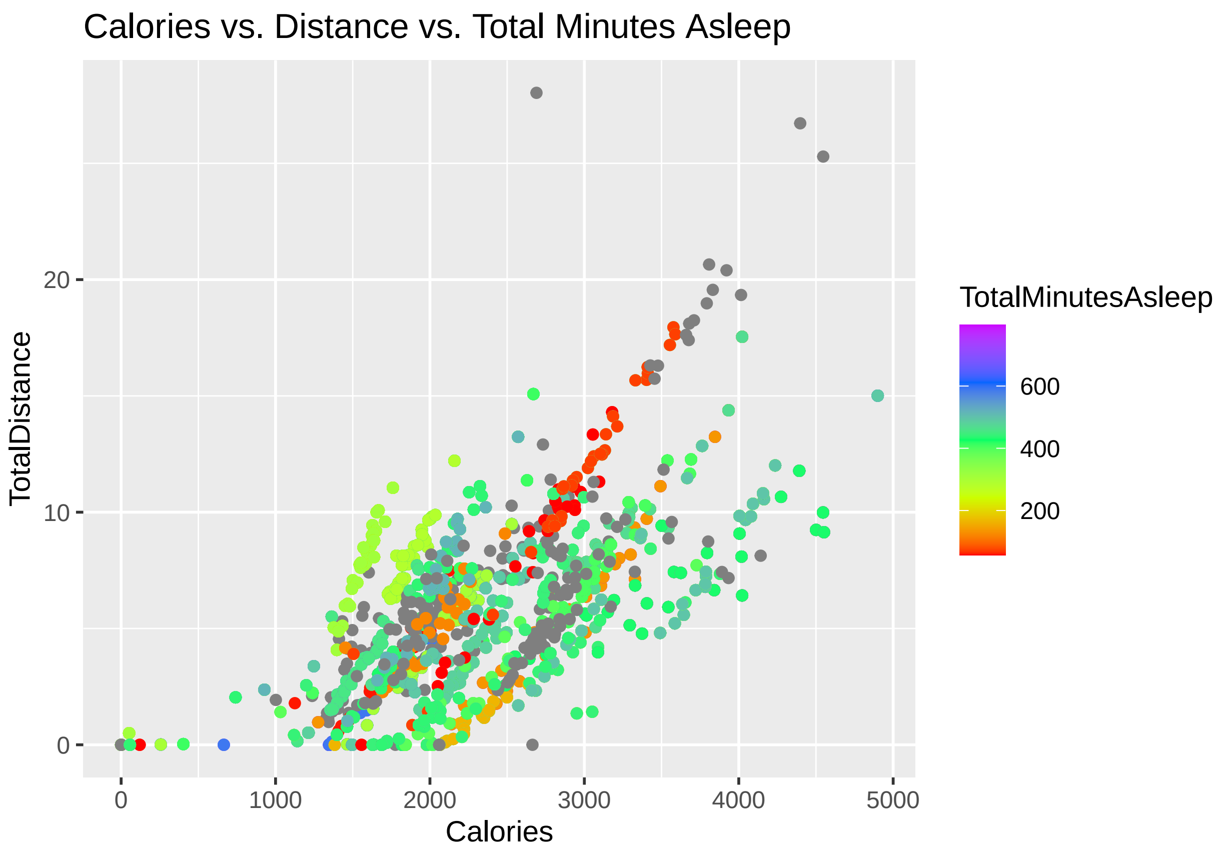
```
Combined_outer <- merge(Activity_new, Sleep_new, by="Id", all = TRUE)
n_distinct(Combined_outer$Id)
```

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

```
# View(Combined_outer)
```

```


calories_distance_sleep_point <- ggplot(Combined_outer) +
  geom_point(mapping=aes(x=Calories, y=TotalDistance, color=TotalMinutesAsleep)) +
  labs(title="Calories vs. Distance vs. Total Minutes Asleep")

calories_distance_sleep_point + scale_color_gradientn(colours = rainbow(5))
```

### Relationship between Calories, Distance and Total Minutes Asleep



Based on this plot, we can determine that even after long distances and calories expended, the average amount of sleep ranges from 6 - 8 hours. There were hits of red and orange, but I do think these were recorded in error. However, it is possible some individuals stayed up most of the night and only slept for a couple hours or they might have only tracked themselves taking short naps.

This plot also illustrates the positive correlation between total distance and calories. In other words, the more distance traveled, the more calories expended.

We can also see the overall trend that the people with more distance/calories expended are able to spend more time asleep.

CDC recommends 7-8 hours of sleep per night. Those who get less than 6.5 hours or 390 minutes of sleep each night might be considered sleep-deprived. We can calculate the total percentage of subjects with less than 390 minutes of sleep per night as follows:

```
install.packages("scales")
```

```
df1 = Sleep_new
percent_Asleep <- (sum(df1$TotalMinutesAsleep <= 390) / nrow(Sleep_new))


library(scales)
#library(formattable) # loading package

label_percent()(percent_Asleep)
```

```
## [1] "34%"
```

So, we can conclude that around 34% of the users are sleeping less than 6.5 hours a night, which might be classified as sleep deprivation.

```
data <- Combined_outer
head(data)

# Convert Table to a .csv file
write.csv(data, "C:\\Users\\User\\Desktop\\Portfolio Materials\\Fitbit_Fitness_Tracker_Data_20211222v01
```

```
# Load the .csv file
Activity_Sleep_data <- read.csv("C:\\Users\\User\\Desktop\\Portfolio Materials\\Fitbit_Fitness_Tracker_

head(Activity_Sleep_data)
```

```
#Activity_Sleep_data <- Activity_Sleep_data  %>%
  #select(-LoggedActivitiesDistance,
         #-SedentaryActiveDistance
         #)
```

```
install.packages("metan")
```

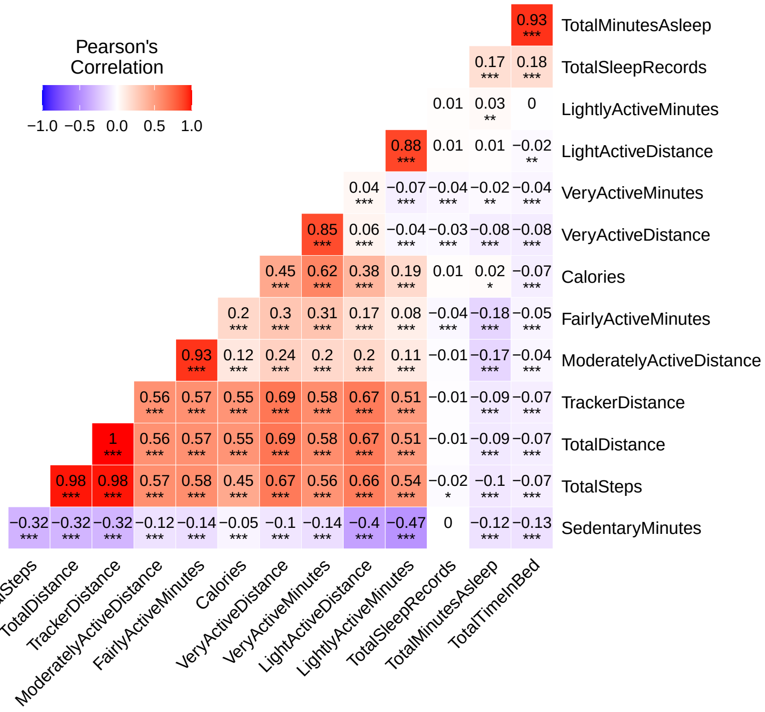
```
library(metan)
library(dplyr)
```

```
no_na_Activity_Sleep_data <- na.omit(Activity_Sleep_data)
cor(no_na_Activity_Sleep_data[sapply(no_na_Activity_Sleep_data, is.numeric)])
```

```
load("Fitabase Data 4.12.16-5.12.16/Bellabeat.RData")
```

```
ASd_round <- no_na_Activity_Sleep_data %>% mutate_if(is.numeric,round)
corr <- corr_coef(ASd_round[, -1:-2])
plot(corr)
```

### Correlation Matric of Activity_new and Sleep_new Variables



* p < 0.05; ** p < 0.01; and *** p < 0.001

From the above correlation matrix, we can see a strong correlation between the following metrics which are relevant:

- Positive relationship between Calories & Very Active Minutes
- Negative relationship between lightly active minutes & Sedentary Minutes
- Positive relationship between total steps & lightly active minutes/distance, fairly active minutes/distance
- Positive relationship between very active minutes & total distance

We can identify different trends and user behavior relating to the metrics like calories, steps, and distance etc. from the above correlation matrix.

### Conclusions & Recommandations for the Business

So, collecting data on activity, sleep, stress, etc. will allow the company Bellabeat to empower the customers with knowledge about their own health and daily habits. The company Bellabeat is growing rapidly and quickly positioned itself as a tech-driven wellness company for their customers.

We have performed a detailed analysis on different data points of the FitBit Fitness Tracker Dataset. The studied data points include activity, sleep, stress, calorie intake/expenditure, and daily habits, etc. This analysis provides some valuable insights about the fitness level and health of the users in relation to different daily habits and lifestyle practices. Some key findings from the analysis are presented below:

**Activity Level:** The majority of participants are lightly active with high sedentary time.

**Step Count:** A significant data point to determine user activity is the number of steps taken each day by a given user. We can see that the average user takes a total of 7638 steps each day.

**Sedentary Time:** It can be seen from the data that the average user spends almost 16 hours a day of Sedentary time which is not a healthy practice.

**Sleep:** Participants spend more time in bed than actual time asleep. Sleep quality and sleeping habits are not ideal.

**Activity Throughout the Week:** We analyzed the users' activity level i.e. sedentary time, activity level (intenser/moderate/light) throughout the week with respect to the day of the week i.e. Monday/Tuesday etc. We can see that . . . (copy and paste findings here again)

**Correlation Study:**

- *Total Steps vs. Sedentary Minutes:* We can see a negative correlation between the total steps and sedentary minutes. This suggests a general trend that the less active people i.e. the people with fewer steps per day spend more sedentary time.
- *Total Steps vs. Calories:* We can see a positive correlation between the total steps and calories burnt. This suggests that more activity is related to more calories expended.
- *Relationship Between Calories, Distance, and Total Minutes Asleep:* From the Calories vs. Distance vs. Total minutes asleep plot we can see that there is a positive correlation with calories with both distance covered each day and total minutes asleep. This suggests that the more distance a person covers, he/she is more likely to spend more time asleep and also more calories burned, which should be beneficial for health.

**Target Audience :** From the analysis of the dataset, we can identify certain groups of people who could benefit from fitness tracking/lifestyle improvements. These are people with -

- Low level of physical activity
- High level of sedentary time
- Low step count per day and less average distance covered each day
- Low caloric expenditure
- Inadequate amount of sleep/poor sleeping habits, etc.

People with this sort of lifestyle are most likely people with full-time jobs spending a lot of time at the computer or office workers with low levels of physical exercise and a lack of healthy habits, etc. It is recommended that Bellabeat might focus on people with these certain characteristics as a target audience who could benefit from using fitness-tracking smart devices. For example office workers and sedentary workers.

**Message to the Company :** After a thorough analysis of the dataset, we are able to identify various trends in user behavior relevant to their fitness, health, and lifestyle. These general trends might be useful to make strategic decisions for Bellabeat's smart devices.

| Trend | Feature Recommendations |
|---|---|
| The average daily step count (7638) is less than recommended by CDC (8000) | Bellabeat smart devices should track user steps throughout the day and send users daily reminders, and weekly reports about their daily step target. |
| The average user spends more than 16 hours of sedentary time per day. | Bellabeat smart devices should track users' sedentary time and send push notifications for users whenever they are spending too much sedentary time. It should also send users reminders to perform some light activity according to their physical activity level. |

| Trend | Feature Recommendations |
|---|---|
| Inconsistent level of activity throughout the week | If we take a look a the trends of active hours, they are the highest at the start of the week but gradually drop-off. Bellabeat customers might be encouraged to maintain consistency throughout the week with regular updates and reminders. We can provide users with customized reports on their weekly activities so that they can work on improving their schedules. |
| Unhealthy sleeping habits | Bellabeat smart devices should be able to track and monitor sleep quality and sleep activity. Bellabeat devices should be able to help users develop better sleeping habits through the following features - notifications and reminders for getting to bed on time and waking up on time, reminders to stay away from artificial light before bedtime, reports, and suggestions to develop better sleeping habits. |
| Positive correlation between total steps and sedentary minutes | Users with higher sedentary time should be suggested to walk more steps with reminders/push notifications. |
| Correlation between total steps and calories expended (positive) | A linear relationship has been developed between the total steps and calories expended. We can use this data to interpolate activity targets for people looking to hit a specific calorie-burning target. For example, if a person wants to burn 2000 calories a day he/she can set a target of 5000 steps/day. Again, if a person wants to burn 2500 calories a day, he/she can be suggested to take 10000 steps a day, and so on. |
| Relationship between Calories, Distance, and Sleep | Users looking forward to developing better sleeping habits and calorie expenditure might be suggested daily distance targets to better improve their health and fitness. |

**Recommendations to the Bellabeat Marketing team :** In light of the above findings and recommendations, the Bellabeat marketing team might be able to take some strategic decisions regarding the following fields:

- Target Audience:
  - Office workers and people with work of sedentary nature
  - Users who could benefit from a healthier lifestyle
- Bellabeat Smart Device Hardware Feature Recommendations
  - Activity Tracking
  - Sleep Tracking
  - Step Counting
  - Distance Tracking
- Bellabeat Smart Application Software Feature Recommendations
  - Push notifications/reminders to workout/exercise
  - Alerts for users with high sedentary status
  - Customized step/distance goals
  - Sleep Alerts
  - Daily/Weekly report and summary statistics for different activities/metrics i.e. Sleep, Steps, Distance, Calories, etc.
  - Lifestyle recommendations

**References :**

- ” CDC - Higher Daily Step Count Linked with Lower All-cause Mortality - CDC Newsroom.” Centers for Disease Control and Prevention, 24 Mar. 2020, https://www.cdc.gov/media/releases/2020/p0324-daily-step-count.html

- “CDC — How Much Sleep Do I Need? — Sleep and Sleep Disorders.” Centers for Disease Control and Prevention, 2 Mar. 2017, https://www.cdc.gov/sleep/about_sleep/how_much_sleep.html

- “Weight Loss: How Many Calories Can Walking 10,000 Steps Burn? - Lifestyle News.” The Times of India, Last updated on - Jan 26, 2021, https://m.timesofindia.com/life-style/health-fitness/fitness/weight-loss-how-many-calories-can-walking-10000-steps-burn/amp_etphotostory/80452318.cms

- “The Dangers of Sitting: Why Sitting Is the New Smoking.” The Dangers of Sitting: Why Sitting Is the New Smoking — Better Health Channel, 22 Aug. 2020, https://www.betterhealth.vic.gov.au/health/healthyliving/the-dangers-of-sitting

- Nield, David. “Scientists Figured out How Much Exercise You Need to ‘Offset’ a Day of Sitting.” ScienceAlert, 26 Nov. 2020, https://www.sciencealert.com/getting-a-sweat-on-for-30-40-minutes-could-offset-a-day-of-sitting-down

- “How Much Physical Activity Do Adults Need?” Centers for Disease Control and Prevention, Centers for Disease Control and Prevention, 7 Oct. 2020, https://www.cdc.gov/physicalactivity/basics/adults/index.htm