# Case Study 2: How Can a Wellness Technology Company Play It Smart

Andres Rosero Yepes

05/02/2023

## Bellabeat

### About the company

Urška Sršen and Sando Mur founded Bellabeat, a high-tech company that manufactures health-focused smart products. Sršen used her background as an artist to develop beautifully designed technology that informs and inspires women around the world. 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

## 1. ASK

Sršen asks you to analyze smart device usage data in order to gain insight into how consumers use non-Bellabeat smart devices. She then wants you to select one Bellabeat product to apply these insights to in your presentation. These questions will guide your analysis:

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

You will produce a report with the following deliverables:

1. A clear summary of the business task

2. A description of all data sources used

3. Documentation of any cleaning or manipulation of data

4. A summary of your analysis

#### Guiding questions

- What is the problem you are trying to solve?

Discover trends in smart device usage in order to gain insight into how Consumers use non-Bellabeat smart devices.

- How can your insights drive business decisions?

We will select the Leaf Wellness Tracker product to apply the insights to the presentation

**Key tasks**

1. Identify the business task

- Discover trends in smart device usage in order to gain insight into how Consumers use non-Bellabeat smart devices.
- We will select the Leaf Wellness Tracker product to apply the insights to the presentation

2. Consider key stakeholders

- Urška Sršen: Bellabeat's cofounder and Chief Creative Officer
- Sando Mur: Mathematician and Bellabeat's cofounder; key member of the

  Bellabeat executive team

## Deliverable

A clear statement of the business task:

Analyze trends in smart device usage, in order to gain insight into how consumers use non-Bellabeat smart devices. Select the Leaf Wellness Tracker product to apply the insights to the presentation

# 2. PREPARE

**Data integrity**

**Guiding questions**

- Where is your data stored?

This dataset generated by respondents to a distributed survey via Amazon Mechanical Turk between 03.12.2016-05.12.2016. Thirty eligible Fitbit users consented to the submission of personal tracker data, including minute-level output for physical activity, heart rate, and sleep monitoring. It is stored in Kaggle (CC0: Public Domain, dataset made available through Mobius). Mobius is the Data Scientist

- How is the data organized? Is it in long or wide format?

It is a combination of both, long and wide data

- Are there issues with bias or credibility in this data? Does your data ROCCC?

It is third party It is comprehensive

- How are you addressing licensing, privacy, security, and accessibility?

It is stored in Kaggle (CC0: Public Domain, dataset made available through Mobius). Mobius is the Data Scientist

- How did you verify the data's integrity?

Data has been cleaned and Trimmed with google sheets

R is being used for data cleaning, process(Transformation), analysis and visualization

**We are using 3 tables:** dailyActivity_merged.csv, sleepDay_merged.csv, dailySteps_merged.csv

# 3. PROCESS

## Setting up my environment

Installing Tidyverse Package

```
install.packages("tidyverse")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

## Installing Basic cleaning Packages

```
install.packages("here")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```
library(here)
```

```
## here() starts at /cloud/project
```

```
install.packages("skimr")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```
library(skimr)
install.packages("janitor")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```
library(janitor)
```

```
##
## Attaching package: 'janitor'

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

### lets also install the dplyr package

```
install.packages("dplyr")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```
library(dplyr)
```

## Importing datasets:

```
library(readr)
activity <- read_csv("dailyActivity.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.
```

```
sleep <- read_csv("sleepDay.csv")
```

```
## Rows: 410 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.
```

```
steps <- read_csv("dailySteps.csv")
```

```
## Rows: 895 Columns: 3
## -- Column specification ---------------------------------------------------------
## Delimiter: ","
## chr (1): ActivityDay
## dbl (2): Id, StepTotal
##
## 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.
```

Exploring datasets and learning from them. Duplicates were removed, the Trim function was also used in google sheets

```
glimpse(activity)
```

```
## Rows: 940
## Columns: 15
## $ Id                       <dbl> 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/~
## $ TotalSteps               <dbl> 13162, 10735, 10460, 9762, 12669, 9705, 13019~
## $ TotalDistance            <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.59, 9.8~
## $ TrackerDistance          <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.59, 9.8~
## $ LoggedActivitiesDistance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ VeryActiveDistance       <dbl> 1.88, 1.57, 2.44, 2.14, 2.71, 3.19, 3.25, 3.5~
## $ ModeratelyActiveDistance <dbl> 0.55, 0.69, 0.40, 1.26, 0.41, 0.78, 0.64, 1.3~
## $ LightActiveDistance      <dbl> 6.06, 4.71, 3.91, 2.83, 5.04, 2.51, 4.71, 5.0~
## $ SedentaryActiveDistance  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ VeryActiveMinutes        <dbl> 25, 21, 30, 29, 36, 38, 42, 50, 28, 19, 66, 4~
## $ FairlyActiveMinutes      <dbl> 13, 19, 11, 34, 10, 20, 16, 31, 12, 8, 27, 21~
## $ LightlyActiveMinutes     <dbl> 328, 217, 181, 209, 221, 164, 233, 264, 205, ~
## $ SedentaryMinutes         <dbl> 728, 776, 1218, 726, 773, 539, 1149, 775, 818~
## $ Calories                 <dbl> 1985, 1797, 1776, 1745, 1863, 1728, 1921, 203~
```

```
glimpse(sleep)
```

```
## Rows: 410
## Columns: 5
## $ Id                 <dbl> 1.50e+09, 1.50e+09, 1.50e+09, 1.50e+09, 1.50e+09, 1~
## $ SleepDay           <chr> "4/12/2016 00:00", "4/13/2016 00:00", "4/15/2016 00~
## $ TotalSleepRecords  <dbl> 1, 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ TotalMinutesAsleep <dbl> 327, 384, 412, 340, 700, 304, 360, 325, 361, 430, 2~
## $ TotalTimeInBed     <dbl> 346, 407, 442, 367, 712, 320, 377, 364, 384, 449, 3~
```

```
glimpse(steps)
```

```
## Rows: 895
## Columns: 3
## $ Id          <dbl> 1.5e+09, 1.5e+09, 1.5e+09, 1.5e+09, 1.5e+09, 1.5e+09, 1.5e~
## $ ActivityDay <chr> "########", "########", "########", "########", "########"~
## $ StepTotal   <dbl> 13162, 10735, 10460, 9762, 12669, 9705, 13019, 15506, 1054~
```

# 4. ANALYZE

Identifying how many unique participants are there in each dataframe The dailyActivity seems to have more

We will use the Id column to find answers

```
n_distinct(activity$Id)
```

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

```
n_distinct(sleep$Id)
```

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

```
n_distinct(steps$Id)
```

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

Unique participants per data frame:

dailyActivity 33 unique participants

sleepDay 24 unique participants

dailySteps 33 unique participants

How many observations are there in each dataframe?

```
nrow(activity) # 940 rows
```

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

```
nrow(sleep) # 410 rows
```

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

```
nrow(steps) # 895 rows
```

```
## [1] 895
```

STATISTICS PER DATA FRAME

dailyActivity data frame

```
activity %>%
  select(TotalSteps,
         SedentaryMinutes, Calories) %>%
  summary()
```

```
##    TotalSteps    SedentaryMinutes    Calories
##  Min.   :    0   Min.   :   0.0   Min.   :   0
##  1st Qu.: 3790   1st Qu.: 729.8   1st Qu.:1828
##  Median : 7406   Median :1057.5   Median :2134
##  Mean   : 7638   Mean   : 991.2   Mean   :2304
##  3rd Qu.:10727   3rd Qu.:1229.5   3rd Qu.:2793
##  Max.   :36019   Max.   :1440.0   Max.   :4900
```

Based on the results above, the average user:

- Each user walks 7638 steps on average per day
- Out of the 1440 (24 hours) max minutes, the user stays 991 (16.5 hours) minutes inactive
- The user burns an average of 2304 calories per day

sleepDay data frame

```
sleep %>%
  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
```

Based on the results above, the average user:

- Spends 419 minutes Asleep. an hour has 60 mins.

  419/60 = 6.98 hours or 7 hours is the average sleep time

- Spends 458 minutes in bed. an hour has 60 mins.

  458/60 = 7.6 hours or 8 hours is the average time in bed

dailySteps data frame

```
steps %>%
  select(StepTotal) %>%
  summary()
```

```
##    StepTotal
##  Min.   :    0
##  1st Qu.: 4494
##  Median : 7802
##  Mean   : 7997
##  3rd Qu.:10938
##  Max.   :36019
```

Based on the results above, the average user:

- the user walks an average of 7997 steps per day

Note: the dailyActivity data frame's mean is 7638 Vs. 7997 from the dailySteps data frame The dailyActivity dataframe has more rows than the dailySteps dataframe, which could impact the mathematical calculations. the difference is minimum. It is a good indication of the datasets trustworthiness

IDENTIFY TRENDS AND RELATIONSHIPS

What information can we extract from previous analysis about people's activities?

- Each user walks 7638 steps on average per day
- Out of the 1440 (24 hours) max minutes, the user stays 991 (16.5 hours) minutes inactive
- The user burns an average of 2304 calories per day
- Spends 419 minutes Asleep. an hour has 60 mins.

  419/60 = 6.98 hours or 7 hours is the average sleep time
- Spends 458 minutes in bed. an hour has 60 mins.

  458/60 = 7.6 hours or 8 hours is the average time in bed
- the user walks an average of 7997 steps per day

Note: the dailyActivity data frame's mean is 7638 Vs. 7997 from the dailySteps data frame The dailyActivity dataframe has more rows than the dailySteps dataframe, which could impact the mathematical calculations. the difference is minimum. It is a good indication of the datasets trustworthiness

What relationships and correlations can help us identify the best course of action, about the segments we can market?

How can we apply future insights, to market the use of the Leaf Wellness Tracker product, as a personal coach & support indicator to improve women's health and lifestyle?

# 5. SHARE

## Installing ggplot2 package for visualization

```r
install.packages("ggplot2")
```

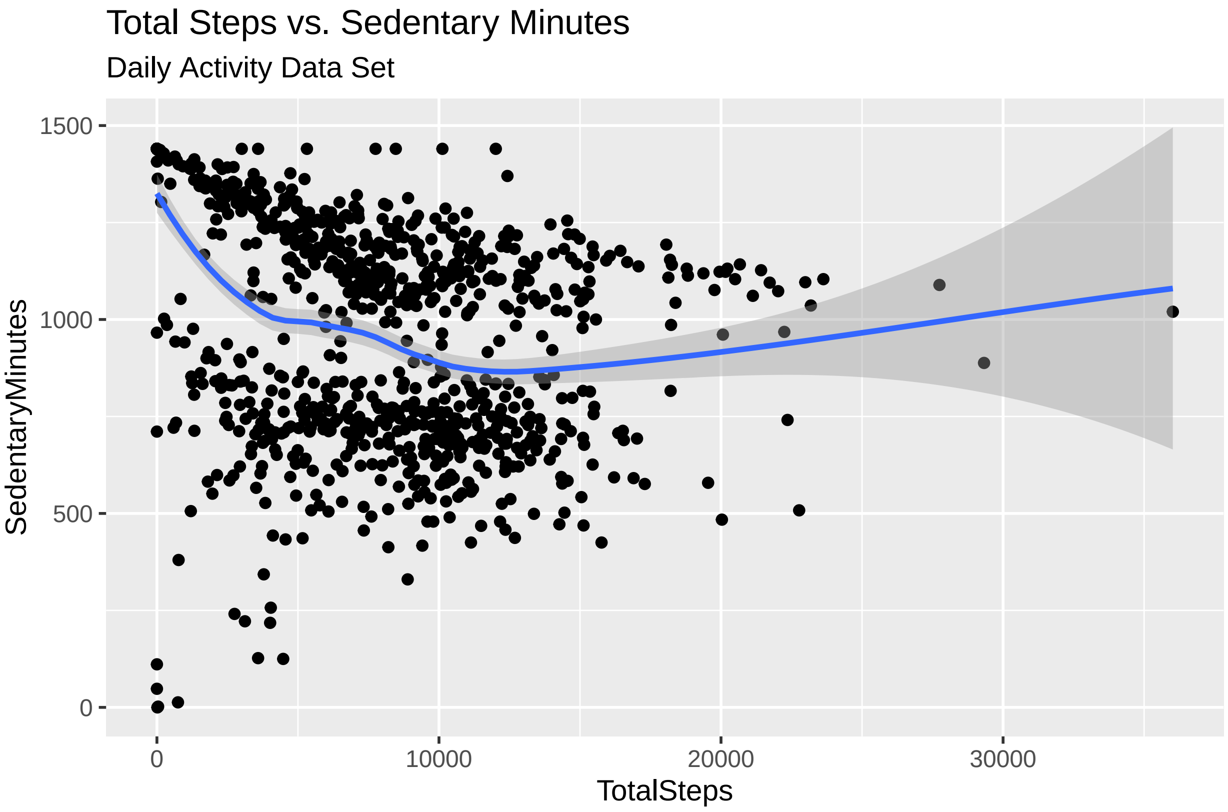
```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```r
library(ggplot2)
```

IDENTIFY TRENDS AND RELATIONSHIPS

Exploring relationships between Total Steps and Sedentary Minutes

```r
ggplot(data=activity)+
  geom_point(mapping=aes(x=TotalSteps, y=SedentaryMinutes))+
  geom_smooth(mapping=aes(x=TotalSteps, y=SedentaryMinutes))+
  labs(title="Total Steps vs. Sedentary Minutes", subtitle="Daily Activity Data Set",
       caption="Data collected by Mobius, Kaggle.com")
```



Data collected by Mobius, Kaggle.com

Correlation Coefficient

```
cor(activity$TotalSteps, activity$SedentaryMinutes)
```

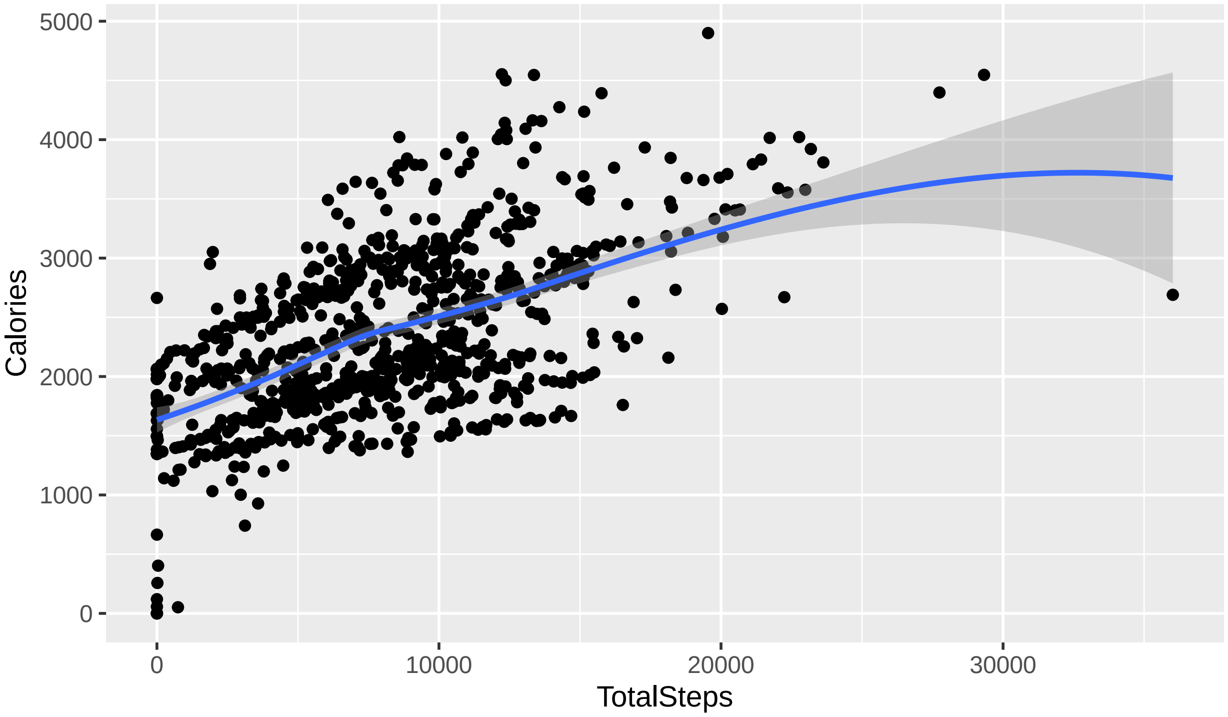
```
## [1] -0.3274835
```

**The -0.32 shows a negative correlation coefficient** What information can we extract from previous analysis about people's activities?

The scatter plot above shows that the less steps done the more sedentary minutes, which means less activity

Exploring relationships between Total Steps and Calories

```
ggplot(data=activity)+
  geom_point(mapping=aes(x=TotalSteps, y=Calories))+
  geom_smooth(mapping=aes(x=TotalSteps, y=Calories))+
  labs(title="Total Steps vs. Calories", subtitle="Daily Activity Data Set",
       caption="Data collected by Mobius, Kaggle.com")
```



Total Steps vs. Calories

Daily Activity Data Set

Data collected by Mobius, Kaggle.com

Correlation Coefficient

```
cor(activity$TotalSteps, activity$Calories)
```

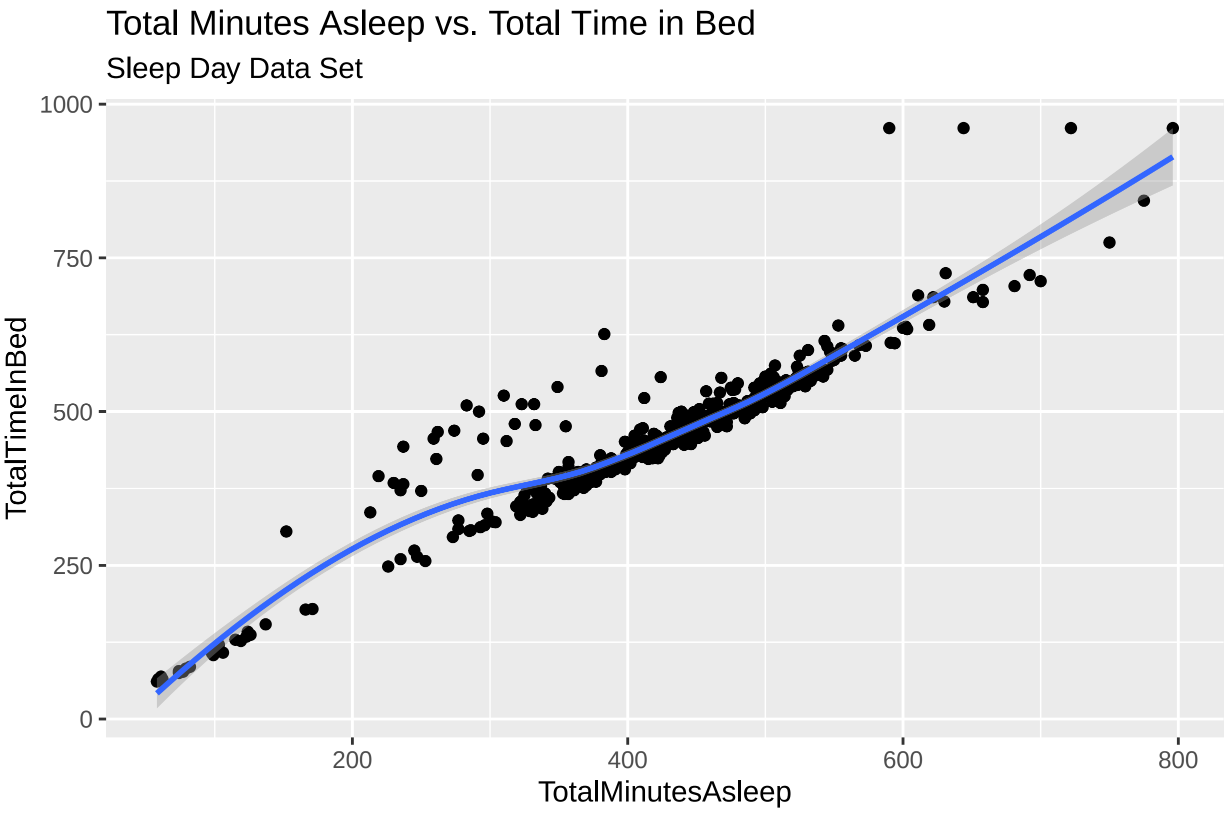
```
## [1] 0.5915681
```

**The 0.59 shows a positive correlation coefficient** What information can we extract from previous analysis about people's activities?

The scatter plot above shows a positive correlation between Total Steps and Calories, with a COR = 0.59 As steps increase Calories burned increase as well.

Exploring relationships between Total Steps and Calories

```
ggplot(data=sleep)+
  geom_point(mapping=aes(x=TotalMinutesAsleep, y=TotalTimeInBed))+
  geom_smooth(mapping=aes(x=TotalMinutesAsleep, y=TotalTimeInBed))+
  labs(title="Total Minutes Asleep vs. Total Time in Bed", subtitle="Sleep Day Data Set",
       caption="Data collected by Mobius, Kaggle.com")
```



Data collected by Mobius, Kaggle.com

```
cor(sleep$TotalMinutesAsleep, sleep$TotalTimeInBed)
```

```
## [1] 0.9304224
```

**The 0.93 shows a positive correlation coefficient** What information can we extract from previous analysis about people's activities?

As user spend more time in bed the total minutes asleep increase as well. the COR is 0.93. There are a few cases where the time spent in bed is not equal to the total minutes asleep.

### Merging these two datasets together. dailyActivity and sleepDay

```
combined <- merge(sleep, activity, by="Id")
```

Take a look at how many participants are in this data set.

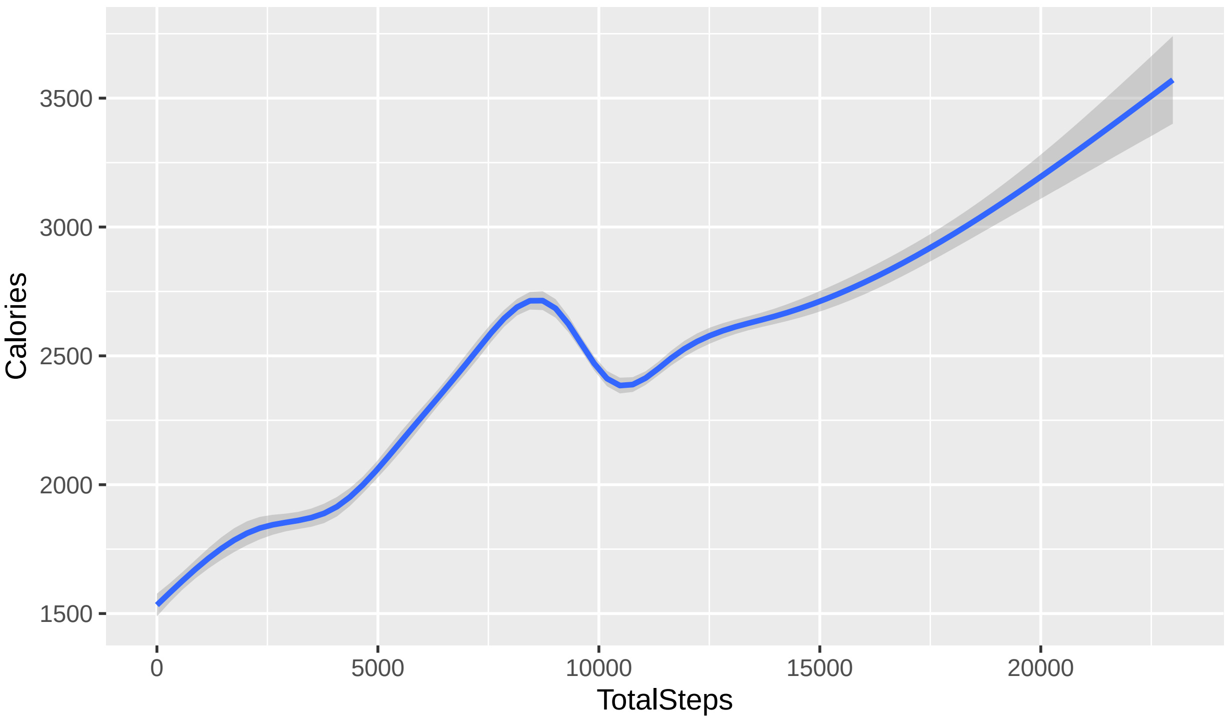
```
n_distinct(combined$Id)
```

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

There were more participants Ids in the daily activity dataset with 33 participants, In the new combined data set we have 24

Exploring relationships between Total Steps and Calories

```
ggplot(data=combined)+
  geom_smooth(mapping=aes(x=TotalSteps, y=Calories))+
  labs(title="Total Steps vs. Calories", subtitle="Combined Data Set",
       caption="Data collected by Mobius, Kaggle.com")
```



Total Steps vs. Calories

Combined Data Set

Data collected by Mobius, Kaggle.com

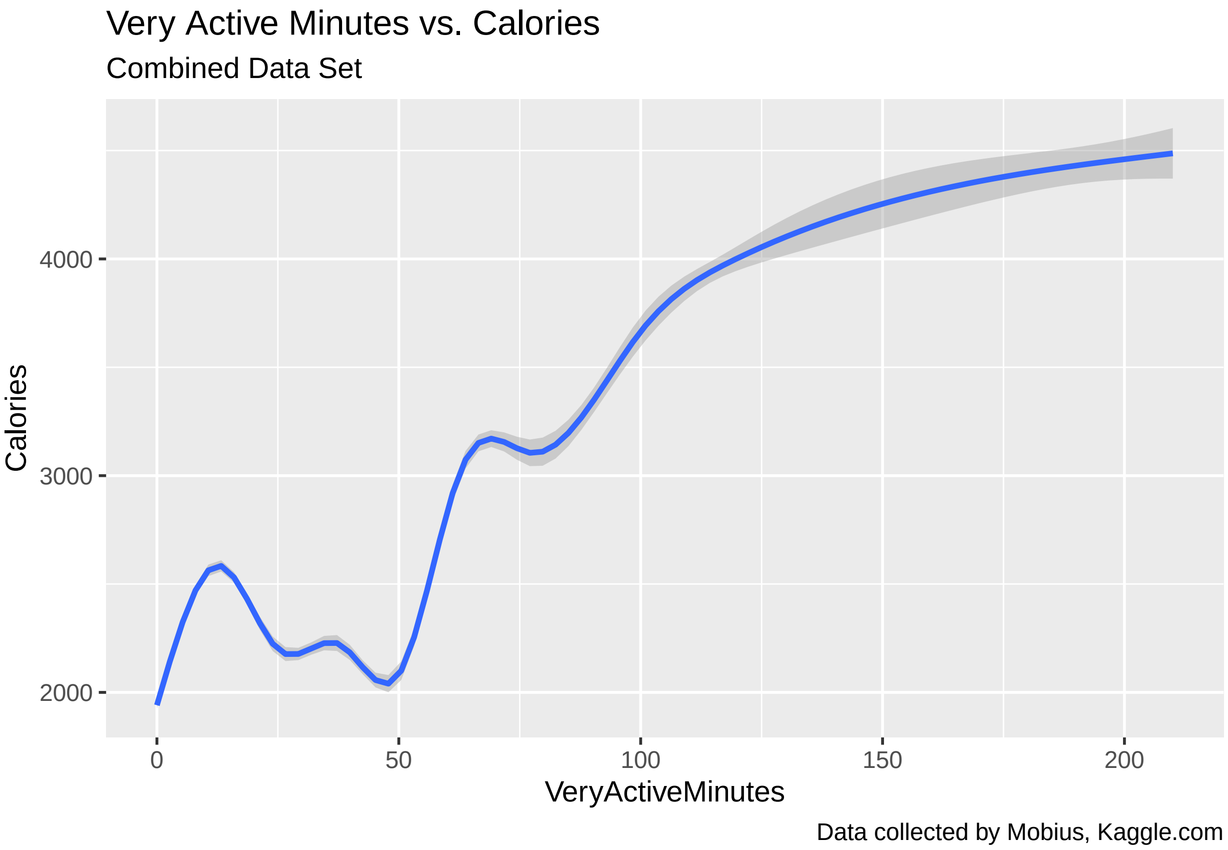
```
cor(combined$TotalSteps, combined$Calories)
```

```
## [1] 0.4462722
```

**The 0.44 shows a positive correlation coefficient** An increase in Steps, comes with an increase in Burned Calories

Exploring relationships between Total Steps and Calories

```
ggplot(data=combined)+
  geom_smooth(mapping=aes(x=VeryActiveMinutes, y=Calories))+
  labs(title="Very Active Minutes vs. Calories", subtitle="Combined Data Set",
       caption="Data collected by Mobius, Kaggle.com")
```



The chart above shows a clear positive correlation between VeryActiveMinutes and Calories, because as minutes increase more Calories are burnt

With the view function we can see the first date on the ActivityDate column is 4/12/2016 = Tue

Adding two columns to the data set named: "Day" and "Day of Week" and converting the Dateformat inside the observations in the column to Mon, Tue, Wed, Thu, Fri, Sat, Sun.

```
combined$Day <- mdy(combined$ActivityDate)

colnames(combined)
```

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

```
combined$DayofWeek <- wday(combined$Day, label = TRUE)
```

### Exploring the data set

```
library(tidyverse)
```

```
colnames(combined)
```

```
##  [1] "Id"                       "SleepDay"
##  [3] "TotalSleepRecords"        "TotalMinutesAsleep"
##  [5] "TotalTimeInBed"           "ActivityDate"
##  [7] "TotalSteps"               "TotalDistance"
##  [9] "TrackerDistance"          "LoggedActivitiesDistance"
## [11] "VeryActiveDistance"       "ModeratelyActiveDistance"
## [13] "LightActiveDistance"      "SedentaryActiveDistance"
## [15] "VeryActiveMinutes"        "FairlyActiveMinutes"
## [17] "LightlyActiveMinutes"     "SedentaryMinutes"
## [19] "Calories"                 "Day"
## [21] "DayofWeek"
```

Arranging by VeryActiveMinutes

```
activity %>% arrange(VeryActiveMinutes)
```

```
## # A tibble: 940 x 15
##            Id ActivityDate TotalSteps TotalDistance TrackerDistance
##         <dbl> <chr>             <dbl>         <dbl>           <dbl>
##  1 1500000000 5/12/2016             0          0               0
##  2 1620000000 4/12/2016          8163          5.31            5.31
##  3 1620000000 4/13/2016          7007          4.55            4.55
##  4 1620000000 4/14/2016          9107          5.92            5.92
##  5 1620000000 4/15/2016          1510          0.98            0.98
##  6 1620000000 4/16/2016          5370          3.49            3.49
##  7 1620000000 4/19/2016          2916          1.9             1.9
##  8 1620000000 4/20/2016          4974          3.23            3.23
##  9 1620000000 4/21/2016          6349          4.13            4.13
## 10 1620000000 4/22/2016          4026          2.62            2.62
## # i 930 more rows
## # i 10 more variables: LoggedActivitiesDistance <dbl>,
## #   VeryActiveDistance <dbl>, ModeratelyActiveDistance <dbl>,
## #   LightActiveDistance <dbl>, SedentaryActiveDistance <dbl>,
## #   VeryActiveMinutes <dbl>, FairlyActiveMinutes <dbl>,
## #   LightlyActiveMinutes <dbl>, SedentaryMinutes <dbl>, Calories <dbl>
```

Creating a small data set with the Day of Week and the VeryActiveMinutes Column Average

```
combined %>% group_by(DayofWeek) %>% drop_na() %>% summarize(mean_VeryActiveMinutes =
mean(VeryActiveMinutes))
```

```
## # A tibble: 7 x 2
##   DayofWeek mean_VeryActiveMinutes
##   <ord>                      <dbl>
## 1 Sun                         19.9
## 2 Mon                         28.4
## 3 Tue                         30.0
## 4 Wed                         20.2
## 5 Thu                         22.6
## 6 Fri                         24.0
## 7 Sat                         22.2
```

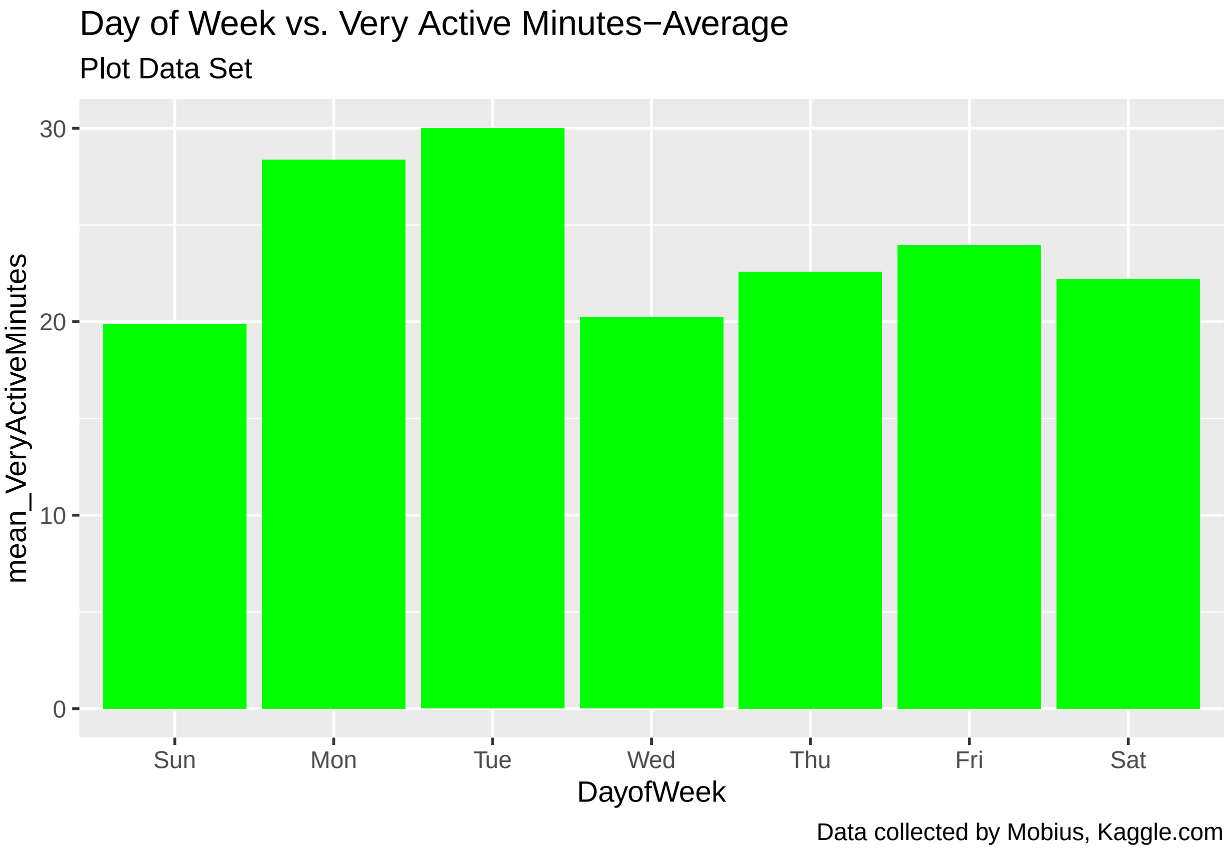
### Calculating the daily average of the VeryActiveMinutes, for each day of the week

Naming the data set "Plot"

```
Plot <- combined %>% group_by(DayofWeek) %>% drop_na() %>% summarize(mean_VeryActiveMinutes = mean(Very
```

Plotting the “Plot” Data Set

```
ggplot(data=Plot, aes(x=DayofWeek, y=mean_VeryActiveMinutes))+
  geom_histogram(stat = "identity", fill = "green")+
  labs(title="Day of Week vs. Very Active Minutes-Average", subtitle="Plot Data Set",
       caption="Data collected by Mobius, Kaggle.com")
```



The Histogram above, shows that average activity in terms of minutes, increase on mondays and tuesdays and as the weekend approaches it decreases

Adding the average-VeryActiveMinutes per day, from sunday thru saturday produces a total of 167.18 minutes for the week.

Calculating the daily VeryActiveMinutes average, between sunday and saturday, per day.

```
Plot %>%
  select(mean_VeryActiveMinutes) %>%
  summary()
```

```
##  mean_VeryActiveMinutes
##  Min.   :19.88
##  1st Qu.:21.21
##  Median :22.58
##  Mean   :23.89
##  3rd Qu.:26.17
##  Max.   :30.00
```

**Key Findings - Share Phase**

- The less steps taken the more sedentary minutes, which means less activity.
- There is a positive correlation between Total Steps and Calories, with a COR = 0.59; As steps increase, Calories burned increase as well.
- When the user spends more time in bed, the total minutes asleep increase as well; the COR is 0.93. There are a few cases where the time spent in bed is not equal to the total minutes asleep.
- There is a clear positive correlation between VeryActiveMinutes and Calories, because as minutes increase more Calories are burnt.
- Users spend an average of 23.89 minutes in a very active way per day, with the less active days being monday and tuesday.
- Adding the average-VeryActiveMinutes per day, from sunday thru saturday produces a total of 167.18 minutes for the week.

# 6. ACT

- User should increase the steps taken per day.
- The more steps taken the more calories burned and Anxiety symtoms will reduce.
- Set a time to go bed and create a habit. The more Minutes Asleep the more Activity Minutes will the user have on a weekly basis.
- The user should increase the total of very active minutes for the week, which is 167.18 (based on our findings), For adults it recommends a combination of 150 minutes of moderate activity and 75 minutes of vigorous or more active minutes, totaling 225 minutes per week. the guidelines recommend at least 60 minutes per day or 420 minutes 7 days a week for adolescents and children.
- to achieve the goal described above they should go from 23.89 active minutes on an average per day, to 32.14 minutes for adults and 60 active minutes for younger persons, to meet the minumum goal recommended by CDC. https://www.cdc.gov/physicalactivity/walking/index.htm. Center for Disease Control and Prevention.
- Based on the information above the fitbit is a great way to track steps, minutes asleep, calories burned and the most active minutes or very active minutes during the week, which in the end help support a healthy and happier lifestyle.