Bellabeat - Case Study

Vikram

2023-10-05

Introduction

Welcome to my Bellabeat data analysis case study! In this case study, I will perform real-world tasks of a data analyst. And 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 company Bellabeat

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 grownrapidly and quickly positioned itself as a tech-driven wellness company for women.

Context

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

Ask phase

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

- What are the most important patterns in how people use smart fitness trackers?
- How can Bellabeat use this information to develop marketing campaigns that are more likely to resonate with their customers?
- Who are the most important people to involve in this project(Key Stakeholders)?

Business Task(Imaginary)

After getting answers to my question, I am able to clearly define the business task and it is as follows:

To analyse how do Bellabeat customers use their smart devices? And then, Identify potential opportunities for growth and recommendations for the Bellabeat marketing team based on trends in smart device usage.

Prepare Phase

In this phase, I will download and Import the dataset, and I will sort and filter data to make sure all the data is organized.

Downloading the data

I downloaded the data set from Kaggle, and to download it click here on LINK

It consists fo 18 csv files.

Installing and Loading Packages

Now I install the required packages that will help me along the analysis. I am also also going to install few packages (last 3) for data cleaning purposes.

```
# install.packages("tidyverse", repos="http://cran.us.r-project.org")
# install.packages("lubridate", repos="http://cran.us.r-project.org")
# install.packages("dplyr", repos="http://cran.us.r-project.org")
# install.packages("ggplot2", repos="http://cran.us.r-project.org")
# install.packages("tidyr", repos="http://cran.us.r-project.org")
# install.packages("here", repos="http://cran.us.r-project.org")
# install.packages("skimr", repos="http://cran.us.r-project.org")
# install.packages("janitor", repos="http://cran.us.r-project.org")
```

Now I will load the packages using library()

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

After viewing all 18 datasets, I have decided to work with following datasets to continue my analysis. I will import the datasets and View, Clean, Format, and Organize the data.

- dailyActivity_merged.csv

```
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
```

```
View(Activity)
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

```
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

```
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

```
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 7:21:2
##  $ Value: int  97 102 105 103 101 95 91 93 94 93 ...
```

- sleepDay_merged.csv

```r
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
```

```r
colnames(Sleep)
```

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

```r
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

```r
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" "4/21/20
##  $ 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 ...
```

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().

```
glimpse(Activity)
```

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

```
skim_without_charts(Activity)
```

Table 1: Data summary

Name	Activity
Number of rows	940
Number of columns	15
Column type frequency:	
character	1
numeric	14
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
ActivityDate	0	1	8	9	0	31	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
Id	0	1	4.855407e+09	2.424805e+09	1503960366	2.320127e+09	4.445115e+09	6.962181e+09	8.877689e+09
TotalSteps	0	1	7.637910e+03	5.087150e+03	0	3.789750e+03	7.405500e+03	1.072700e+04	3.601900e+04
TotalDistance	0	1	5.490000e+00	3.920000e+00	0	2.620000e+00	5.240000e+00	7.710000e+00	2.803000e+01
TrackerDistance	0	1	5.480000e+00	3.910000e+00	0	2.620000e+00	5.240000e+00	7.710000e+00	2.803000e+01
LoggedActivitiesDistance	0	1	1.100000e-01	6.200000e-01	0	0.000000e+00	0.000000e+00	0.000000e+00	4.940000e+00
VeryActiveDistance	0	1	1.500000e+00	2.660000e+00	0	0.000000e+00	2.100000e-01	2.050000e+00	2.192000e+01
ModeratelyActiveDistance	0	1	5.700000e-01	8.800000e-01	0	0.000000e+00	2.400000e-01	8.000000e-01	6.480000e+00
LightActiveDistance	0	1	3.340000e+00	2.040000e+00	0	1.950000e+00	3.360000e+00	4.780000e+00	1.071000e+01
SedentaryActiveDistance	0	1	0.000000e+00	1.000000e-02	0	0.000000e+00	0.000000e+00	0.000000e+00	1.100000e-01
VeryActiveMinutes	0	1	2.116000e+01	3.284000e+01	0	0.000000e+00	4.000000e+00	3.200000e+01	2.100000e+02
FairlyActiveMinutes	0	1	1.356000e+01	1.999000e+01	0	0.000000e+00	6.000000e+00	1.900000e+01	1.430000e+02
LightlyActiveMinutes	0	1	1.928100e+02	1.091700e+02	0	1.270000e+02	1.990000e+02	2.640000e+02	5.180000e+02
SedentaryMinutes	0	1	9.912100e+02	3.012700e+02	0	7.297500e+02	1.057500e+03	1.229500e+03	1.440000e+03
Calories	0	1	2.303610e+03	7.181700e+02	0	1.828500e+03	2.134000e+03	2.793250e+03	4.900000e+03

```
glimpse(Calories)
```

```
## Rows: 940
## Columns: 3
## $ Id          <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 1503960366~
## $ ActivityDay <chr> "4/12/2016", "4/13/2016", "4/14/2016", "4/15/2016", "4/16/~
## $ Calories    <int> 1985, 1797, 1776, 1745, 1863, 1728, 1921, 2035, 1786, 1775~
```

```
skim_without_charts(Calories)
```

Table 4: Data summary

Name	Calories
Number of rows	940
Number of columns	3
Column type frequency:	
character	1
numeric	2
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
ActivityDay	0	1	8	9	0	31	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
Id	0	1	4.855407e+09	2.424805e+09	1503960366	2320127002	4445114986	6.962181e+09	8877689391
Calories	0	1	2.303610e+03	7.181700e+02	0	1828.5	2134	2.793250e+03	4900

```
glimpse(Heartrate)
```

```
## Rows: 2,483,658
## Columns: 3
## $ Id    <dbl> 2022484408, 2022484408, 2022484408, 2022484408, 2022484408, 2022~
## $ Time  <chr> "4/12/2016 7:21:00 AM", "4/12/2016 7:21:05 AM", "4/12/2016 7:21:~
## $ Value <int> 97, 102, 105, 103, 101, 95, 91, 93, 94, 93, 92, 89, 83, 61, 60, ~
```

```
skim_without_charts(Heartrate)
```

Table 7: Data summary

Name	Heartrate
Number of rows	2483658
Number of columns	3
Column type frequency:	
character	1
numeric	2
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Time	0	1	19	21	0	961274	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
Id	0	1	5.513765e+09	1950223761.2	2022484408	4388161847	5553957443	6962181067	8877689391
Value	0	1	7.733000e+01	19.4	36	63	73	88	203

```r
glimpse(Intensities)
```

```
## Rows: 940
## Columns: 10
## $ Id                       <dbl> 1503960366, 1503960366, 1503960366, 150396036~
## $ ActivityDay              <chr> "4/12/2016", "4/13/2016", "4/14/2016", "4/15/~
## $ SedentaryMinutes         <int> 728, 776, 1218, 726, 773, 539, 1149, 775, 818~
## $ LightlyActiveMinutes     <int> 328, 217, 181, 209, 221, 164, 233, 264, 205, ~
## $ FairlyActiveMinutes      <int> 13, 19, 11, 34, 10, 20, 16, 31, 12, 8, 27, 21~
## $ VeryActiveMinutes        <int> 25, 21, 30, 29, 36, 38, 42, 50, 28, 19, 66, 4~
## $ SedentaryActiveDistance  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ LightActiveDistance      <dbl> 6.06, 4.71, 3.91, 2.83, 5.04, 2.51, 4.71, 5.0~
## $ ModeratelyActiveDistance <dbl> 0.55, 0.69, 0.40, 1.26, 0.41, 0.78, 0.64, 1.3~
## $ VeryActiveDistance       <dbl> 1.88, 1.57, 2.44, 2.14, 2.71, 3.19, 3.25, 3.5~
```

```r
skim_without_charts(Intensities)
```

Table 10: Data summary

Name	Intensities
Number of rows	940
Number of columns	10
Column type frequency:	
character	1
numeric	9
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
ActivityDay	0	1	8	9	0	31	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
Id	0	1	4.855407e+09	2.424805e+09	1503960366	2.320127e+09	4.445115e+09	6.962181e+09	8.877689e+09
SedentaryMinutes	0	1	9.912100e+02	3.012700e+02	0	7.297500e+02	1.057500e+03	1.229500e+03	1.440000e+03
LightlyActiveMinutes	0	1	1.928100e+02	1.091700e+02	0	1.270000e+02	1.990000e+02	2.640000e+02	5.180000e+02
FairlyActiveMinutes	0	1	1.356000e+01	1.999000e+01	0	0.000000e+00	6.000000e+00	1.900000e+01	1.430000e+02
VeryActiveMinutes	0	1	2.116000e+01	3.284000e+01	0	0.000000e+00	4.000000e+00	3.200000e+01	2.100000e+02
SedentaryActiveDistance	0	1	0.000000e+00	1.000000e-02	0	0.000000e+00	0.000000e+00	0.000000e+00	1.100000e-01
LightActiveDistance	0	1	3.340000e+00	2.040000e+00	0	1.950000e+00	3.360000e+00	4.780000e+00	1.071000e+01
ModeratelyActiveDistance	0	1	5.700000e-01	8.800000e-01	0	0.000000e+00	2.400000e-01	8.000000e-01	6.480000e+00
VeryActiveDistance	0	1	1.500000e+00	2.660000e+00	0	0.000000e+00	2.100000e-01	2.050000e+00	2.192000e+01

```
glimpse(Sleep)
```

```
## Rows: 413
## Columns: 5
## $ Id                 <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 150~
## $ SleepDay           <chr> "4/12/2016 12:00:00 AM", "4/13/2016 12:00:00 AM", "~
## $ TotalSleepRecords  <int> 1, 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ TotalMinutesAsleep <int> 327, 384, 412, 340, 700, 304, 360, 325, 361, 430, 2~
## $ TotalTimeInBed     <int> 346, 407, 442, 367, 712, 320, 377, 364, 384, 449, 3~
```

```
skim_without_charts(Sleep)
```

Table 13: Data summary

Name	Sleep
Number of rows	413
Number of columns	5
Column type frequency:	
character	1
numeric	4
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
SleepDay	0	1	20	21	0	31	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
Id	0	1	5.000979e+09	2.060360e+09	1503960366	3977333714	4702921684	6962181067	8792009665

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
TotalSleepRecords	0	1	1.120000e+00	3.450000e-01	1	1	1	1	3
TotalMinutesAsleep	0	1	4.194700e+02	1.183400e+02	58	361	433	490	796
TotalTimeInBed	0	1	4.586400e+02	1.271000e+02	61	403	463	526	961

```r
glimpse(Weight)
```

```
## Rows: 67
## Columns: 8
## $ Id             <dbl> 1503960366, 1503960366, 1927972279, 2873212765, 2873212~
## $ Date           <chr> "5/2/2016 11:59:59 PM", "5/3/2016 11:59:59 PM", "4/13/2~
## $ WeightKg       <dbl> 52.6, 52.6, 133.5, 56.7, 57.3, 72.4, 72.3, 69.7, 70.3, ~
## $ WeightPounds   <dbl> 115.9631, 115.9631, 294.3171, 125.0021, 126.3249, 159.6~
## $ Fat            <int> 22, NA, NA, NA, NA, 25, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ BMI            <dbl> 22.65, 22.65, 47.54, 21.45, 21.69, 27.45, 27.38, 27.25,~
## $ IsManualReport <chr> "True", "True", "False", "True", "True", "True", "True"~
## $ LogId          <dbl> 1.462234e+12, 1.462320e+12, 1.460510e+12, 1.461283e+12,~
```

```r
skim_without_charts(Weight)
```

Table 16: Data summary

Name	Weight
Number of rows	67
Number of columns	8
Column type frequency:	
character	2
numeric	6
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Date	0	1	19	21	0	56	0
IsManualReport	0	1	4	5	0	2	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
Id	0	1.00	7.009282e+09	1.950322e+09	1.503960e+09	6.962181e+09	6.962181e+09	8.877689e+09	8.877689e+09
WeightKg	0	1.00	7.204000e+01	1.392000e+01	5.260000e+01	6.140000e+01	6.250000e+01	8.505000e+01	1.335000e+02
WeightPounds	0	1.00	1.588100e+02	3.070000e+01	1.159600e+02	1.353600e+02	1.377900e+02	1.875000e+02	2.943200e+02
Fat	65	0.03	2.350000e+01	2.120000e+00	2.200000e+01	2.275000e+01	2.350000e+01	2.425000e+01	2.500000e+01
BMI	0	1.00	2.519000e+01	3.070000e+00	2.145000e+01	2.396000e+01	2.439000e+01	2.556000e+01	4.754000e+01

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
LogId	0	1.00	1.461772e+12	7.829948e+08	1.460444e+12	1.461079e+12	1.461802e+12	1.462375e+12	1.463098e+12

I have removed the "fat" column since there were many missing values in the column, requires specific action on that specific variable

I have removed duplicates from Sleep dataset, There were three duplicates.

Fixing Formatting

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

Activity

```
Activity$ActivityDate=as.POSIXct(Activity$ActivityDate, format="%m/%d/%Y", tz=Sys.timezone())
Activity$date <- format(Activity$ActivityDate, format = "%m/%d/%y")
Activity$ActivityDate=as.Date(Activity$ActivityDate, format="%m/%d/%Y", tz=Sys.timezone())
Activity$date=as.Date(Activity$date, format="%m/%d/%Y")
```

Intensities

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

Sleep

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

```
#str(Activity)
#str(Intensities)
#str(Sleep)
```

Analyze Phase

Summarizing

Lets Check for total number of participants in each dataset

```
Activity %>% summarise(Activity_Participants = n_distinct(Activity$Id))
```

```
##   Activity_Participants
## 1                    33
```

```
n_distinct(Calories$Id)
```

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

```
n_distinct(Intensities$Id)
```

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

```
n_distinct(Heartrate$Id)
```

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

```
n_distinct(Sleep$Id)
```

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

```
n_distinct(Weight$Id)
```

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

This means that there are 33 participants in Activity, Calories, and Intensities data sets. 24 Participants in Sleep data set, 14 participants in Heartrate data set, and only 8 participants in Weight.

We cannot make any conclusions or recommendations with Heartrate and Weight data sets as there are only 14 and 8 participants respectively. These numbers are not significant.

So I have decided to continue my analysis with Activity, Calories, and Intensities data sets.

For Activity Data set lets work with variables "Total Steps,Total Distance, Sedentary Minutes, Calories".

```
Activity %>%
  select(TotalSteps, TotalDistance, SedentaryMinutes, Calories) %>%
  summary()
```

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

Exploring Intense Active Participants:

We will explore number of active minutes for each categories.

```
Intensities %>% select(VeryActiveMinutes, FairlyActiveMinutes, LightlyActiveMinutes, SedentaryMinutes) %
```

```
##  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 Calories Data Set:

```r
Calories %>%
  select(Calories) %>%
  summary()
```

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

For the sleep Data Set:

```r
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  
```

```r
Weight %>%
  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 Analysis:

- The average sedentary time is 991 minutes (more than 16 hours), which is too high and definitely needs to be reduced with a good marketing strategy. (Data sets -> Activity, Intensities)
- Average total steps is (which is 7638) is little bit less than the number of steps recommended by "CDC". According to the CDC research, taking 8,000 steps per day was associated with a 51% lower risk for all-cause mortality (or death from all causes). And taking 12,000 steps per day was associated with a 65% lower risk compared with taking 4,000 steps. According to "National Institutes of Health", Taking 4,000 or fewer steps a day is considered a low level of physical activity.
- Majority of Participants are Lightly Acitve and high sedentary time(Data set -> Intensities)
- Participants sleep 1 time for a time of 419 minutes (approximately 7 hours)
- When compared to average weight, BMI is slightly higher than the recommended ones, this must be due to less physical activity and higher average sedentary time. Customers are need to motivated as part of product marketing strategies.

Merging some data :

Before beginning to visualize the data, 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 data set than in the Sleep data set. So if I use the merge option inner_joint, then I will have the number of participants from the Sleep data set.

It is as follows:

Inner Join

```
Combined_data_inner <- merge(Sleep, Activity, by="Id")
n_distinct(Combined_data_inner$Id)
```

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

So for my analysis I will us "outer join" to include all the participants in the data set. We can achieve it by adding "all = TRUE" argument to our previous code chunk.

Outer Join

```
Combined_data_outer = merge(Sleep, Activity, by="Id", all = TRUE)
n_distinct(Combined_data_outer$Id)
```

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

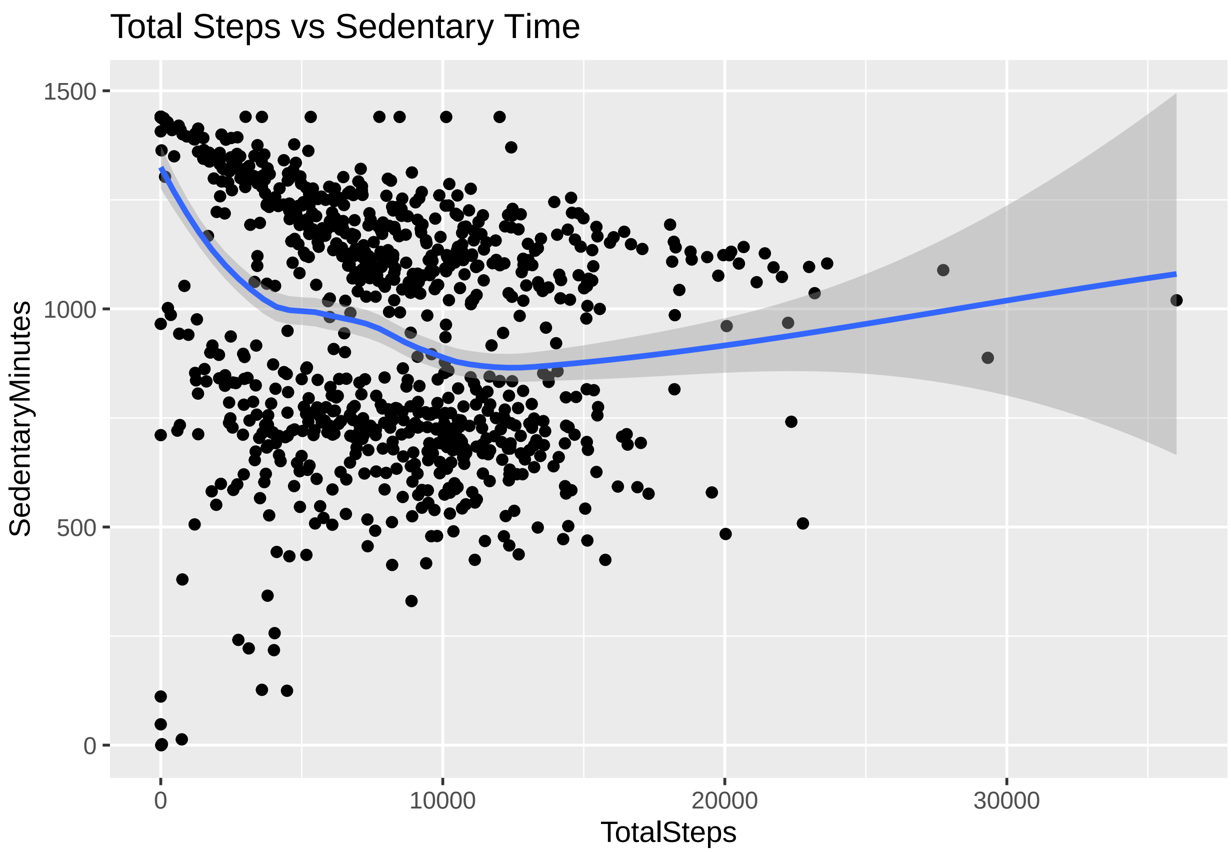
Data Visualization (SHARE and ACT phases)

Now let us visualize few key explorations.

Relationship between Steps and Sedentary time

```
ggplot(data = Activity, aes(x = TotalSteps, y = SedentaryMinutes)) + geom_jitter() + geom_smooth() + lal
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



From the visualization I have observed a negative correlation between Total Steps and Sedentary Minutes. The less you walk they walk the more is their sedentary time and the more they walk the less is their sedentary minutes.

This data shows that the company should market more to customer segments with higher sedeentary time, this also motivates them to increase their physical activity and sales of the product.

To do that, the company needs to find ways to get customers get started in walking more and also measure their daily steps.

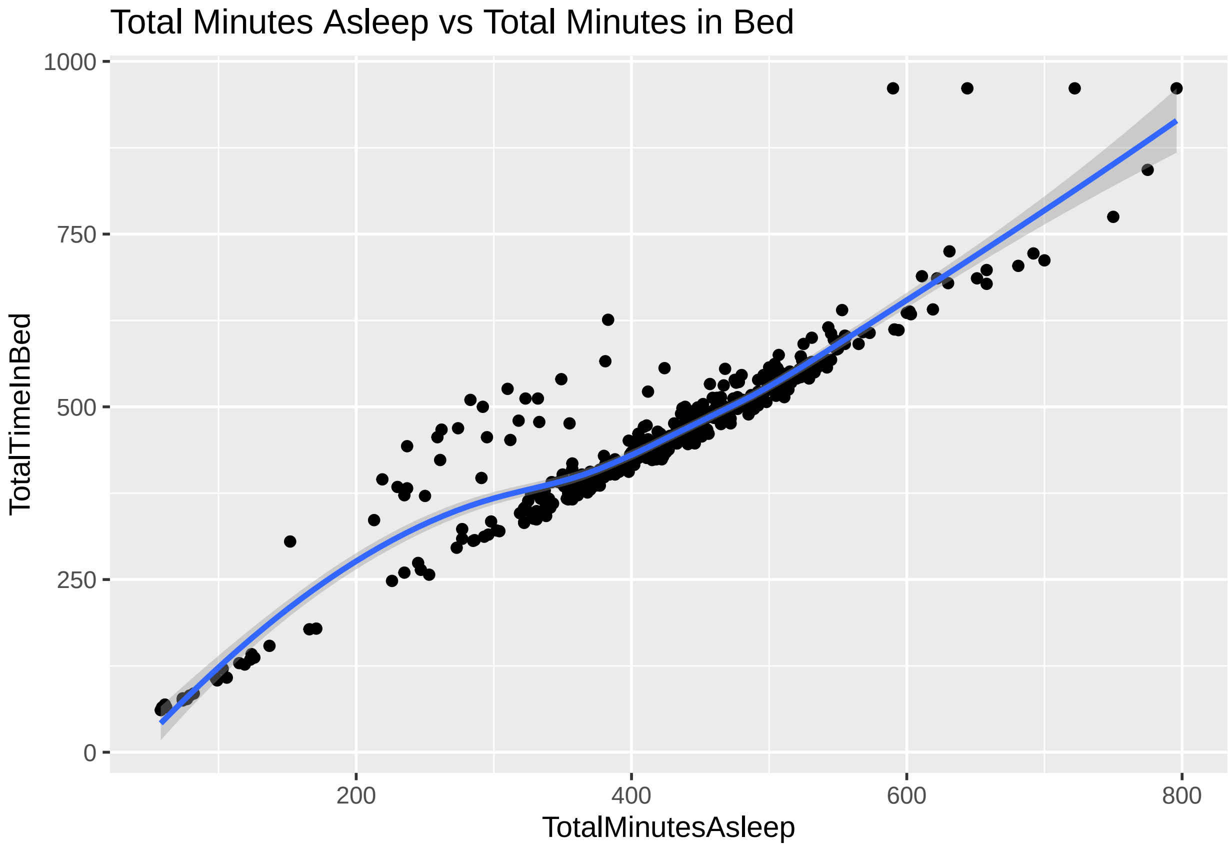
Now let us explore more patterns. From previous analysis, The average of Total Minutes of Sleep and Total minutes in bed are quite close (419, 458 respectively).

Let us check if there is any relationship between them.

Relationship between Total Sleep minutes and Total minutes in Bed

```
ggplot(Sleep, aes(x = TotalMinutesAsleep, y = TotalTimeInBed)) + geom_point() + geom_smooth() + labs(ti
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



From the above visualization we can clearly infer that there is positive correlation between Total Minutes in Bed and Total Time Asleep.

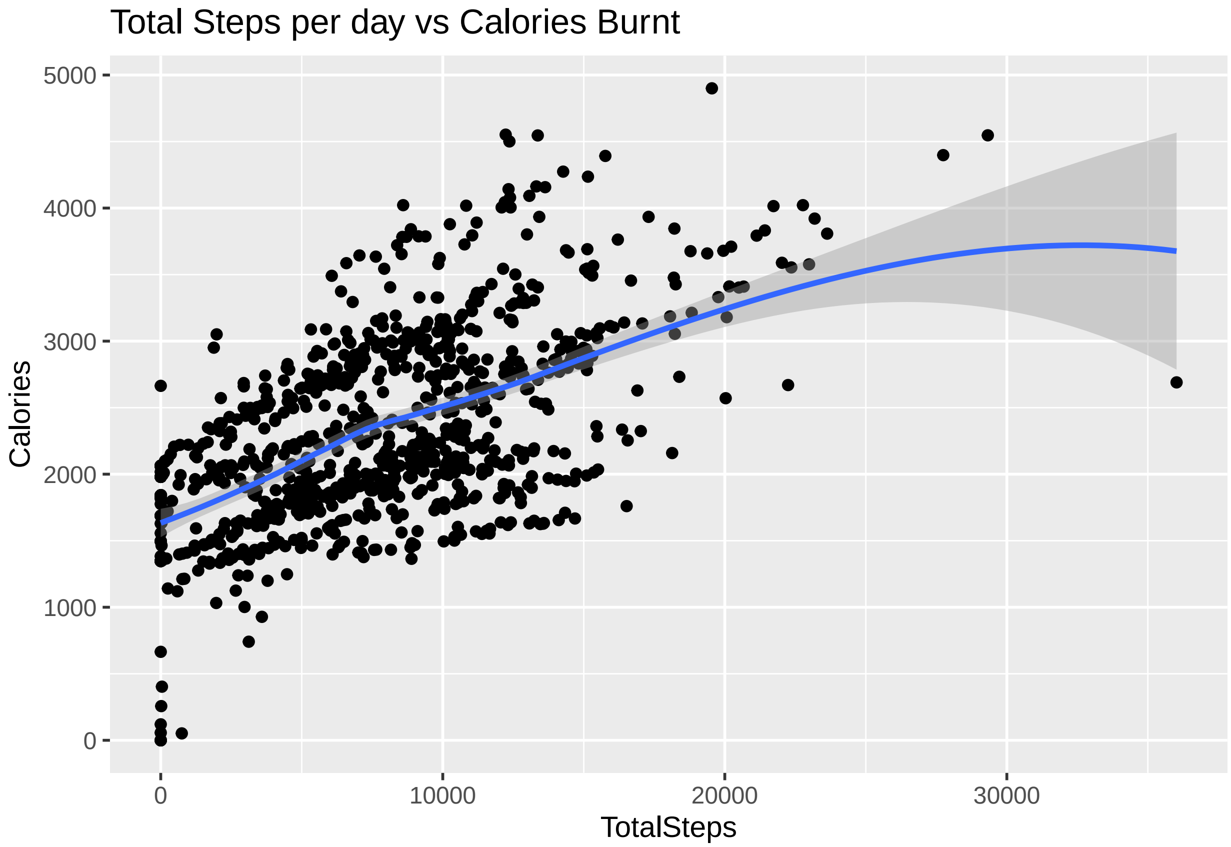
To minimize the difference between average sleep time and average time on bed, the company should consider using a notification system in the product to ensure they notify the users that it's their sleep time, this also improves customers sleep time.

It is common sense that number of calories burnt would be postively correlated to number if steps, but let us check if it's right for our data to check our data integrity.

Relationship between Total no of Steps and Calories Burnt

```
ggplot(Activity, aes(x = TotalSteps, y = Calories)) + geom_point() + geom_smooth() + labs(title = "Total
```

```
## 'geom_smooth()' using method = 'loess' and formula = 'y ~ x'
```



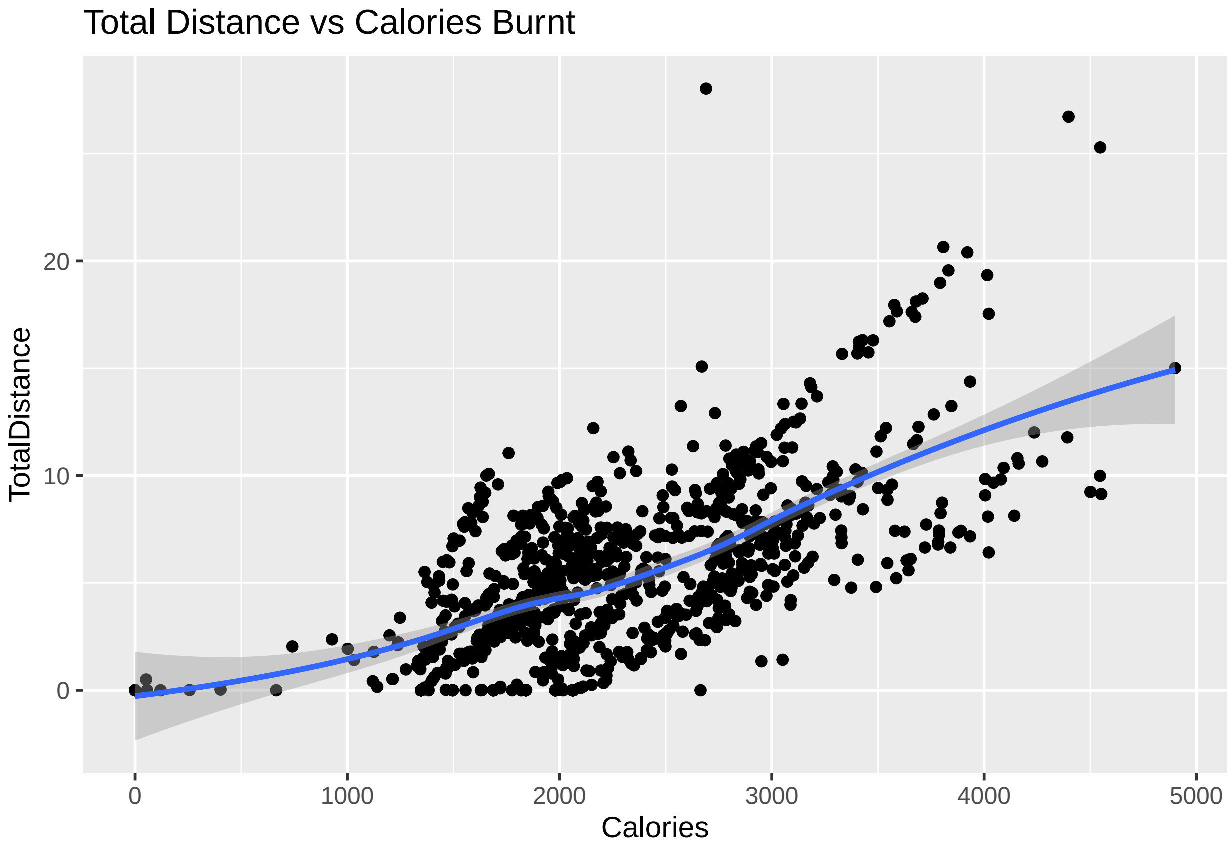
There is a clear positive correlation as we expected between Total Number of Steps and Calories Burnt. So the more people walk the more calories are burnt making them more fit.The company should consider various ways to motivate customers to increase their physical activity through their product to improve customer satisfaction.

It is also better to find the time during which customers are very active and make the product send notification during that time. Logically it must be before or after their work schedule.

##Relationship Between Total Distaance and Calories Burnt

```
ggplot(Activity, aes(x = Calories, y = TotalDistance)) + geom_point() + geom_smooth()+ labs(title = "To
```

```
## ‘geom_smooth()‘ using method = ’loess’ and formula = ’y ~ x’
```



Since there is a positive correlation between Total Distance and Calories burnt, the product should notify users to go outdoor for workout, this also improves their health because the sun light during day gives them Vitamin D.

Conclusions and Recommendations

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. By analyzing the FitBit Fitness Tracker Data set, I found some insights that would help influence Bellabeat marketing strategy.

My recommendations to improve Bellabeat marketing strategy are as follows:

Target specific customer segments: For example, Bellabeat could target women who are trying to lose weight with ads that highlight the features of its fitness trackers that can help them track their progress and reach their goals. Or, Bellabeat could target women who are struggling to get a good night's sleep with ads that highlight the features of its sleep trackers that can help them identify and address the root causes of their sleep problems.

Develop personalized marketing messages: Bellabeat can use the data from its fitness trackers to send personalized marketing messages to its customers. For example, Bellabeat could send a customer a message reminding them to reach their daily step goal or to go to bed earlier. Or, Bellabeat could send a customer a message congratulating them on reaching a fitness milestone.

Create educational content: Bellabeat can use the data from its fitness trackers to create educational content that helps its customers learn more about their health and how to improve it. For example, Bellabeat could publish blog posts or create videos that offer tips on how to lose weight, sleep better, and reduce stress.

Target Audience

Individuals who are employed in full-time positions, dedicating substantial hours to computer work and office environments, and seeking to maintain their physical fitness and daily well-being. These users engage in moderate levels of physical activity(Light Activity) to sustain their health but are interested in enhancing their daily routines to achieve greater health benefits. They may also require guidance on establishing healthy habits and finding motivation to sustain these changes

Message to Bellabeat company

The Bellabeat app should aim to stand out as a distinctive fitness and wellness application. It should strive to serve as a supportive companion, akin to a trusted friend, for its users and customers. Its primary objective should be assisting individuals in harmonizing their personal and professional lives by promoting and facilitating healthy habits.

Recommendations

- The average sedentary time is too high for the users of the app (more than 16 hours). And definitely needs to be reduced with a good marketing strategy. So, the data shows that the company need to market more to the customer segment with a high Sedentary time. And to do that, the company needs to find ways to get customers started in walking more by measuring their daily steps also making use of notifications.
- Participants sleep 1 time for an average of 7 hours. To help users improve their sleep, Bellabeat should consider using app notifications to go to bed. And also, the Bellabeat app can recommend reducing sedentary time for its customers.
- The average total steps per day (which is 7638) is a little bit less than recommended by the CDC. According to the CDC research, taking 8,000 steps per day was associated with a 51% lower risk for all-cause mortality (or death from all causes). And taking 12,000 steps per day was associated with a 65% lower risk compared with taking 4,000 steps. So, Bellabeat can encourage people to take at least 8,000 steps per day by explaining the healthy benefits of doing that.
- By analysing the Intensity data over time. The company will have a good idea on how their customers are using their app during the day. Most users are active before and after work. The company can use this time in the Bellabeat app to remind and motivate users to go for a run or for a walk.
- Proved Educational Content: For Example, for customers who want to lose weight, it can be a good idea to control daily calorie consumption and Bellabeat can suggest some ideas for low-calorie healthy food (for lunch and dinner).
- Bellabeat App can also encourage users through initiating campaigns like “Take the 10,000 Steps Challenge” campaign, and “Sleep Like a Queen” campaign, etc.

Overall, Bellabeat can use the insights from the FitBit Fitness Tracker Data set to develop a more effective marketing strategy that is tailored to the needs of its customers.

Thank You - ANISH VIKRAM VARMA NAMBURI