Bellabeat

Simone Mazzini

2022-04-02

Table of content

Business Task

The aim of this project is to study smart device usage to gain insight about how consumers use non - Bellabeat smart device, and try to enhance the Bellabeat products. The spotlight is on the trends identification and how to apply them on Bellabeat's products. I identify three important goals:

1. How are customers using other fitness trackers, in their daily life ?
2. What particular features seem to be the most heavily used ?
3. What features do Bellabeat products already have that consumers want, and how do we focus marketing on those aspects ?

Analysis Summary

Data source: this project use the FitBit Fitness Tracker Data 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. Individual reports can be parsed by export session ID (column A) or timestamp (column B). Variation between output represents use of different types of Fitbit trackers and individual tracking behaviors / preferences. The files are in csv format and include files for wide data and minute data for minute vise tracking, I put every file in the zip.

Process data

Load the packages

I used different packages in order to conduct a good analysis. In particular I used tidyverse, janitor and sqldf to emulate a SQL syntax and behaviour.

```
library(sqldf)
```

```
## Loading required package: gsubfn
```

```
## Loading required package: proto
```

```
## Warning in fun(libname, pkgname): couldn't connect to display ":0"
```

```
## Loading required package: RSQLite
```

```
library(ggplot2)
library(janitor)
```

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

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

```
library(tidyverse)
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──
```

```
## ✓ tibble  3.1.6     ✓ dplyr   1.0.8
## ✓ tidyr   1.2.0     ✓ stringr 1.4.0
## ✓ readr   2.1.2     ✓ forcats 0.5.1
## ✓ purrr   0.3.4
```

```
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(skimr)
```

Load CSV file

The data frames I'll be working with in this case study will be creating objects for:

1. daily_activity
2. daily sleep
3. daily calories
4. daily intensities
5. weight log info

```
daily_activity <- read_csv("dailyActivity_merged.csv")
```

```
## Rows: 940 Columns: 15
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr  (1): ActivityDate
## dbl (14): Id, TotalSteps, TotalDistance, TrackerDistance, LoggedActivitiesDi...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
daily_sleep <- read_csv("sleepDay_merged.csv")
```

```
## Rows: 413 Columns: 5
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (1): SleepDay
## dbl (4): Id, TotalSleepRecords, TotalMinutesAsleep, TotalTimeInBed
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
daily_weight <- read_csv("weightLogInfo_merged.csv")
```

```
## Rows: 67 Columns: 8
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (1): Date
## dbl (6): Id, WeightKg, WeightPounds, Fat, BMI, LogId
## lgl (1): IsManualReport
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
daily_calories <- read_csv("dailyCalories_merged.csv")
```

```
## Rows: 940 Columns: 3
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (1): ActivityDay
## dbl (2): Id, Calories
##
## 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.
```

```
daily_intensities <- read.csv("dailyIntensities_merged.csv")
```

The Tables

daily_activity

```
head(daily_activity)
```

```
## # A tibble: 6 × 15
##        Id ActivityDate TotalSteps TotalDistance TrackerDistance LoggedActivitie…
##     <dbl> <chr>             <dbl>         <dbl>           <dbl>             <dbl>
## 1  1.50e9 4/12/2016         13162          8.5             8.5                 0
## 2  1.50e9 4/13/2016         10735          6.97            6.97                0
## 3  1.50e9 4/14/2016         10460          6.74            6.74                0
## 4  1.50e9 4/15/2016          9762          6.28            6.28                0
## 5  1.50e9 4/16/2016         12669          8.16            8.16                0
## 6  1.50e9 4/17/2016          9705          6.48            6.48                0
## # … with 9 more variables: VeryActiveDistance <dbl>,
## #   ModeratelyActiveDistance <dbl>, LightActiveDistance <dbl>,
## #   SedentaryActiveDistance <dbl>, VeryActiveMinutes <dbl>,
## #   FairlyActiveMinutes <dbl>, LightlyActiveMinutes <dbl>,
## #   SedentaryMinutes <dbl>, Calories <dbl>
```

```
colnames(daily_activity)
```

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

```
glimpse(daily_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               <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…
```

daily_sleep

```
head(daily_sleep)
```

```
## # A tibble: 6 × 5
##           Id SleepDay           TotalSleepRecor… TotalMinutesAsl… TotalTimeInBed
##        <dbl> <chr>                         <dbl>            <dbl>          <dbl>
## 1 1503960366 4/12/2016 12:00:0…                1              327            346
## 2 1503960366 4/13/2016 12:00:0…                2              384            407
## 3 1503960366 4/15/2016 12:00:0…                1              412            442
## 4 1503960366 4/16/2016 12:00:0…                2              340            367
## 5 1503960366 4/17/2016 12:00:0…                1              700            712
## 6 1503960366 4/19/2016 12:00:0…                1              304            320
```

```
colnames(daily_sleep)
```

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

```
glimpse(daily_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  <dbl> 1, 2, 1, 2, 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…
```

daily_calories

```
head(daily_calories)
```

```
## # A tibble: 6 × 3
##           Id ActivityDay Calories
##        <dbl> <chr>          <dbl>
## 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(daily_calories)
```

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

```
glimpse(daily_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    <dbl> 1985, 1797, 1776, 1745, 1863, 1728, 1921, 2035, 1786, 1775…
```

daily_intensities

```
head(daily_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(daily_intensities)
```

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

```
glimpse(daily_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…
```

daily_weight

```
head(daily_weight)
```

```
## # A tibble: 6 × 8
##           Id Date       WeightKg WeightPounds   Fat   BMI IsManualReport   LogId
##        <dbl> <chr>          <dbl>        <dbl> <dbl> <dbl> <lgl>            <dbl>
## 1 1503960366 5/2/2016 …     52.6         116.    22  22.6 TRUE           1.46e12
## 2 1503960366 5/3/2016 …     52.6         116.    NA  22.6 TRUE           1.46e12
## 3 1927972279 4/13/2016…    134.          294.    NA  47.5 FALSE          1.46e12
## 4 2873212765 4/21/2016…     56.7         125.    NA  21.5 TRUE           1.46e12
## 5 2873212765 5/12/2016…     57.3         126.    NA  21.7 TRUE           1.46e12
## 6 4319703577 4/17/2016…     72.4         160.    25  27.5 TRUE           1.46e12
```

```
colnames(daily_weight)
```

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

```
glimpse(daily_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            <dbl> 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 <lgl> TRUE, TRUE, FALSE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, …
## $ LogId          <dbl> 1.462234e+12, 1.462320e+12, 1.460510e+12, 1.461283e+12,…
```

My findings

After exploring these tables i observed a few things:

Merging of the data frames is possible because they all have same ‘ID’ fields

the daily_activity, daily_calories, and daily_intensities have the exact (940) number of observations.

Lastly the daily_activity table might have a log of calories and intensities already, so we should confirm that the values actually match for any given 'ID' number.

To confirm the last point I am running the following codes:

```
daily_activity2 <- daily_activity %>%
  select(Id, ActivityDate, Calories)

head(daily_activity2)
```

```
## # A tibble: 6 × 3
##           Id ActivityDate Calories
##        <dbl> <chr>           <dbl>
## 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
```

```
sql_check1 <- sqldf('SELECT * FROM daily_activity2 INTERSECT SELECT * FROM daily_calo
ries')
head(sql_check1)
```

```
##           Id ActivityDate 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
```

```
nrow(sql_check1)
```

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

From the above codes we can say that since the first six values of daily_activity and daily_calories are same and total observation of the sql query is 940 the values are the same between the dataframes.

```
daily_activity3 <- daily_activity %>%
  select(Id, ActivityDate, SedentaryMinutes, LightlyActiveMinutes, FairlyActiveMinute
s, VeryActiveMinutes, SedentaryActiveDistance, LightActiveDistance, ModeratelyActiveD
istance, VeryActiveDistance)

head(daily_activity3)
```

```
## # A tibble: 6 × 10
##           Id ActivityDate SedentaryMinutes LightlyActiveMinutes FairlyActiveMin…
##        <dbl> <chr>                   <dbl>                <dbl>             <dbl>
## 1 1503960366 4/12/2016                 728                  328                13
## 2 1503960366 4/13/2016                 776                  217                19
## 3 1503960366 4/14/2016                1218                  181                11
## 4 1503960366 4/15/2016                 726                  209                34
## 5 1503960366 4/16/2016                 773                  221                10
## 6 1503960366 4/17/2016                 539                  164                20
## # … with 5 more variables: VeryActiveMinutes <dbl>,
## #   SedentaryActiveDistance <dbl>, LightActiveDistance <dbl>,
## #   ModeratelyActiveDistance <dbl>, VeryActiveDistance <dbl>
```

```
sql_check2 <- sqldf('SELECT * FROM daily_activity3 INTERSECT SELECT * FROM daily_inte
nsities')
head(sql_check2)
```

```
##           Id ActivityDate 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
```

```
nrow(sql_check2)
```

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

This means I can carry out my analysis with just the 3 different data frames: * daily_activity * sleep_day * weight_log

Since I have done my preparation and pre-processing. Now I will do the analysis

The Analysis

```
n_distinct(daily_activity$Id)
```

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

```
n_distinct(daily_sleep$Id)
```

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

```
n_distinct(daily_weight$Id)
```

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

```
nrow(daily_activity)
```

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

```
nrow(daily_sleep)
```

```
## [1] 413
```

```
nrow(daily_weight)
```

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

Summary of these three databases

daily_activity

```
daily_activity %>%
  select(TotalSteps,
         TotalDistance,
         SedentaryMinutes,
         VeryActiveMinutes) %>%
  summary()
```

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

daily_sleep

```
daily_sleep %>%
  select(TotalSleepRecords,
  TotalMinutesAsleep,
  TotalTimeInBed) %>%
  summary()
```

```
##  TotalSleepRecords TotalMinutesAsleep TotalTimeInBed
##  Min.   :1.000     Min.   : 58.0      Min.   : 61.0
##  1st Qu.:1.000     1st Qu.:361.0      1st Qu.:403.0
##  Median :1.000     Median :433.0      Median :463.0
##  Mean   :1.119     Mean   :419.5      Mean   :458.6
##  3rd Qu.:1.000     3rd Qu.:490.0      3rd Qu.:526.0
##  Max.   :3.000     Max.   :796.0      Max.   :961.0
```

daily_weight

```
daily_weight %>%
  select(WeightPounds,
  BMI) %>%
  summary()
```

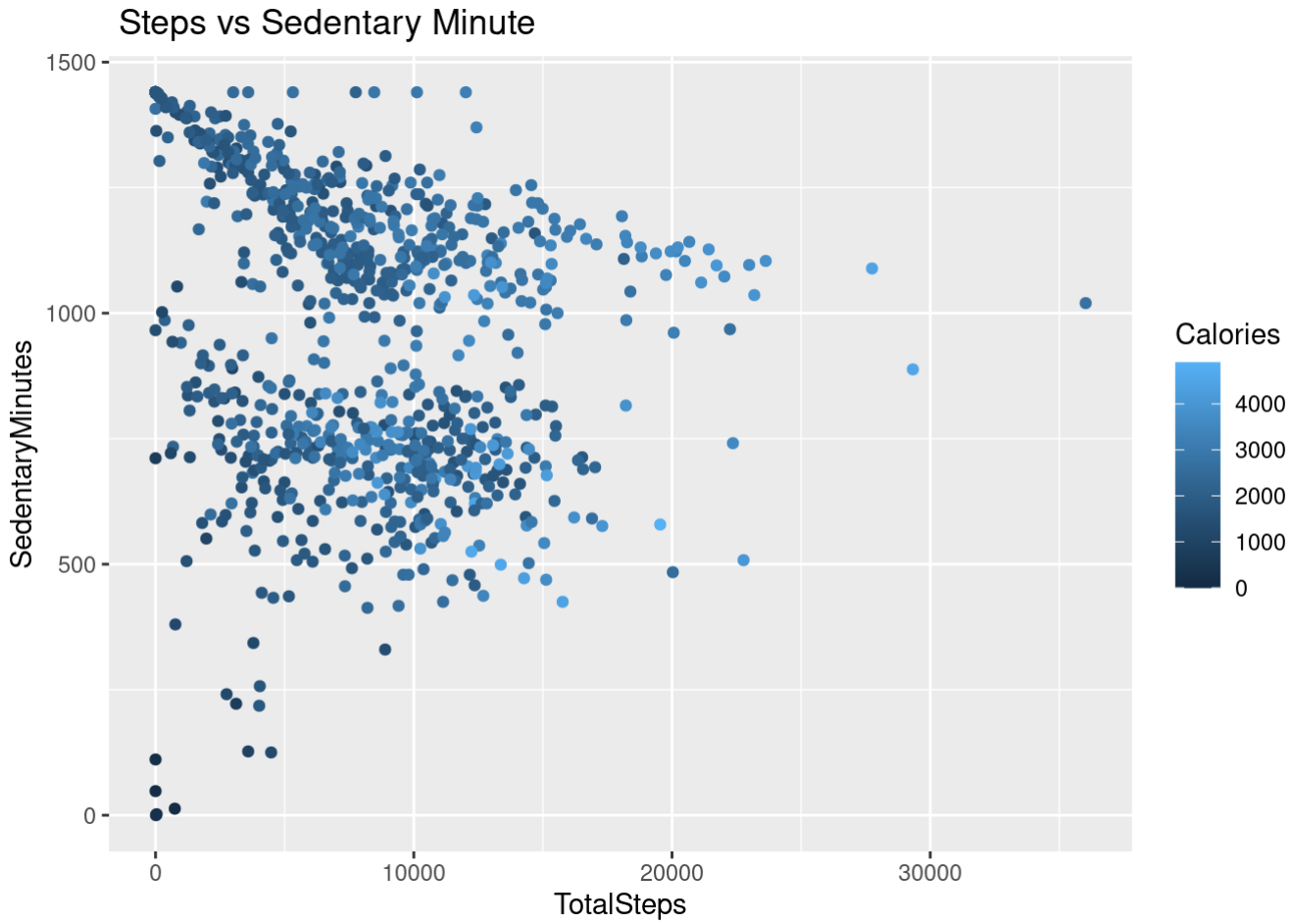
```
##   WeightPounds        BMI
##  Min.   :116.0   Min.   :21.45
##  1st Qu.:135.4   1st Qu.:23.96
##  Median :137.8   Median :24.39
##  Mean   :158.8   Mean   :25.19
##  3rd Qu.:187.5   3rd Qu.:25.56
##  Max.   :294.3   Max.   :47.54
```

The Plots

Now I introduce a series of graphs in order to show the relationship between the data.

Relation between steps taken and sedentary minutes

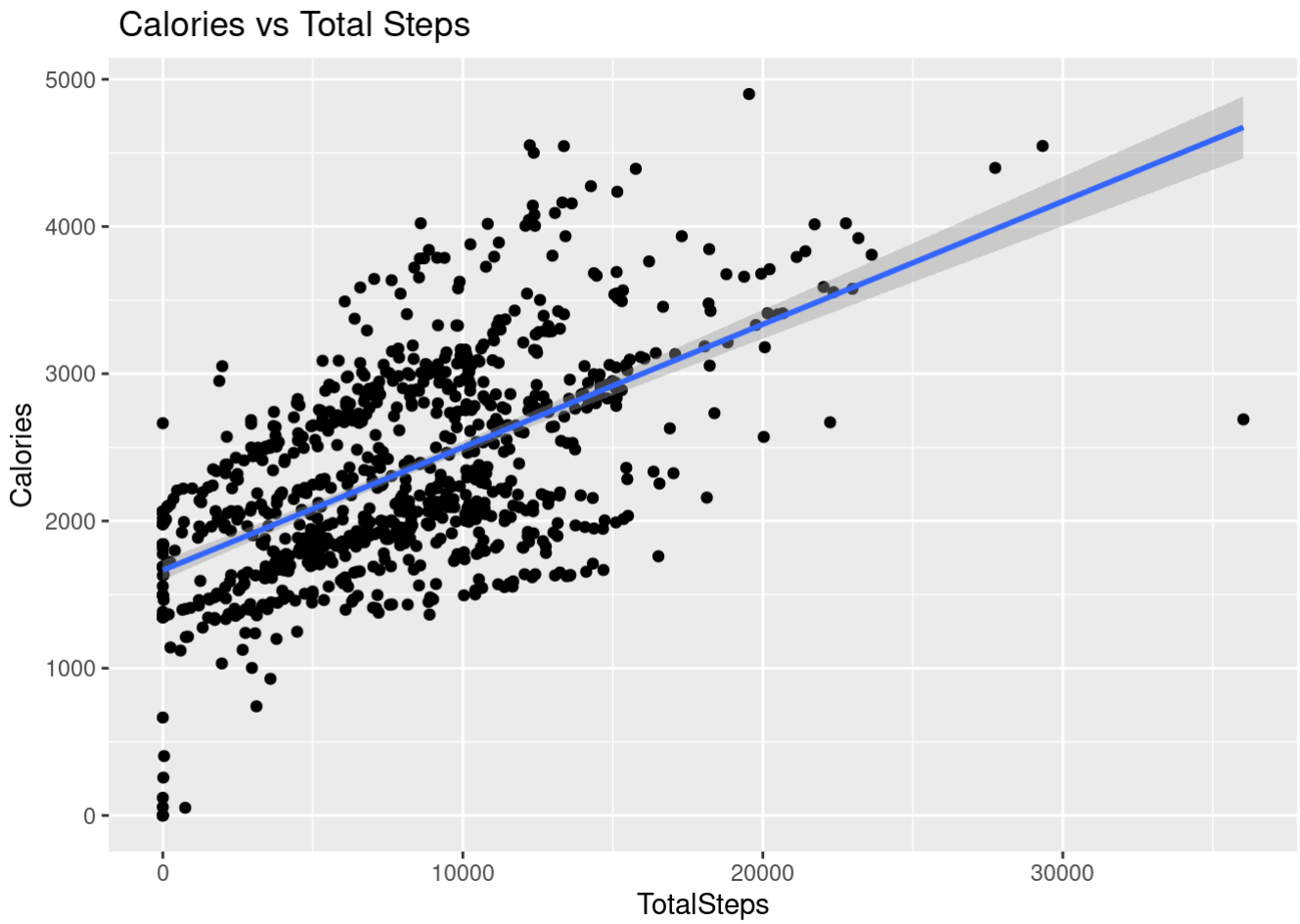
```
ggplot(data=daily_activity, aes(x=TotalSteps, y=SedentaryMinutes, color = Calories))
+ geom_point() + labs(title = " Steps vs Sedentary Minute")
```



Relation between calories and total steps

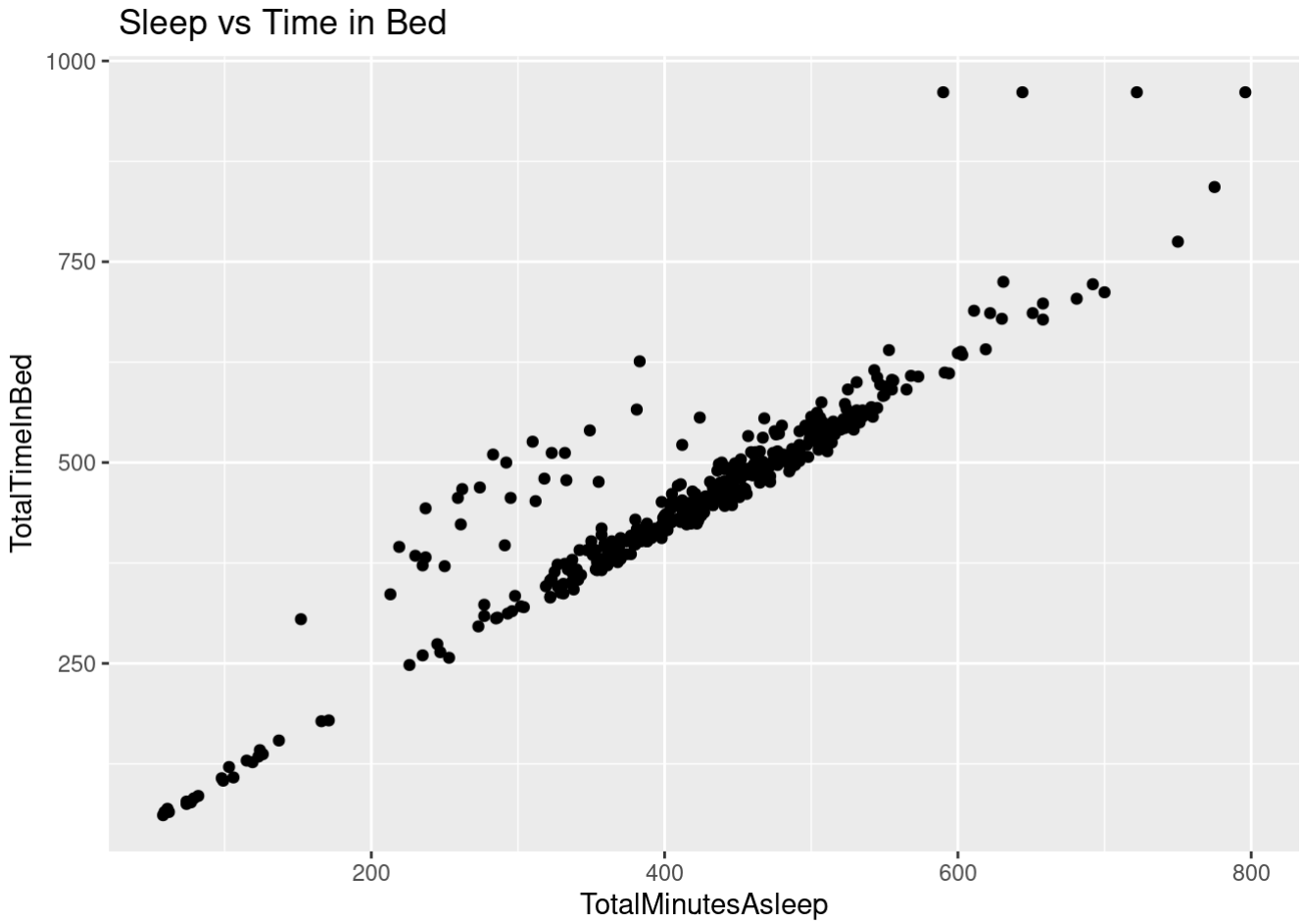
```
ggplot(data=daily_activity, aes(x=TotalSteps, y = Calories))+ geom_point() + stat_smo
oth(method=lm) + labs(title = " Calories vs Total Steps ")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Relation between sleep and time in bed

```
ggplot(data=daily_sleep, aes(x=TotalMinutesAsleep, y=TotalTimeInBed)) + geom_point()
+ labs(title = " Sleep vs Time in Bed")
```

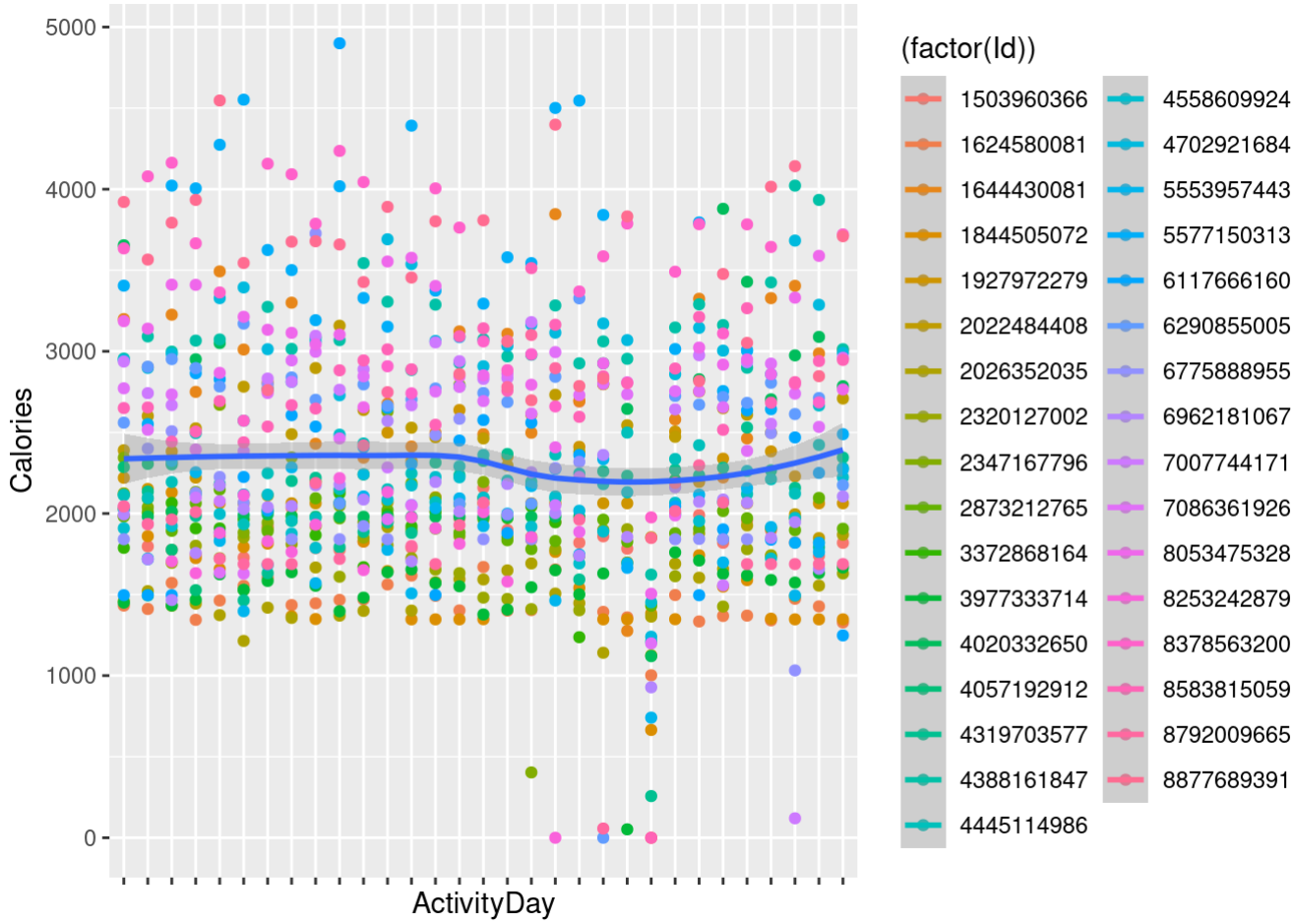


Signifi cant Usage of Fitbit data

Finally, we try to look at whether the users change their habits over the course of their smart device usage. Wedo this by tracking their daily calories usage and observing whether they change over time.

```
ggplot(data = daily_calories, aes (x = ActivityDay, y = Calories, colour = (factor(I
d)), group = 33)) + theme(axis.text.x=element_blank()) + geom_point() + geom_smooth()
```

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


5000
4000
3000
2000
1000
0
Calories
ActivityDay
(factor(Id))
1503960366
1624580081
1644430081
1844505072
1927972279
2022484408
2026352035
2320127002
2347167796
2873212765
3372868164
3977333714
4020332650
4057192912
4319703577
4388161847
4445114986
4558609924
4702921684
5553957443
5577150313
6117666160
6290855005
6775888955
6962181067
7007744171
7086361926
8053475328
8253242879
8378563200
8583815059
8792009665
8877689391