# CaseStudyBellabeatGoogleCapstone

Nathan

2022-10-13

## Introduction and Background

Bellabeat is a tech company that produces health-focused smart products focused on women. Founded by Urška Sršen and Sando Mur in 2013, the company collects data from their technology to empower women with knowledge about their own health and habits. Bellabeat's website can be found here.

This project is for the completion of my Google Data Analytics Capstone.

## Business Task

Identify trends and possible opportunities for Bellabeat using data from a different smart device, FitBit Fitness Tracker Data.

## Hypotheses

These are my hypotheses from the data:

- Total steps is directly proportional to calories burnt.
- Time spent sedentary is inversely proportional to calories burnt.
- Time spent in bed is directly proportional to total minutes of sleep.
- Total minutes of rest is directly proportional to calories burnt.

## Data Source

This dataset includes thirty consenting FitBit users to use their personal data on minute-level output for physical activity, heart rate, sleep monitoring. The data is available on Kaggle, a public domain and open-source repository for users to share data. Click here for the complete dataset and more information.

## Documentation

Firstly, I have to load the packages I need for this project.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.4
## v tibble  3.1.8      v dplyr   1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.2      v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(tidyr)
library(dplyr)
library(skimr)
library(janitor)
```

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

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(ggplot2)
```

Next step would be importing the data I need for this analysis

```
dailyactivity <- read.csv("Fitabase Data 4.12.16-5.12.16/dailyActivity_merged.csv")
sleep <- read.csv("Fitabase Data 4.12.16-5.12.16/sleepDay_merged.csv")
```

Now, I want to check the summaries of the data that I will be using:

```
head(dailyactivity)
```

```
##           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(dailyactivity)
```

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

```
str(dailyactivity)
```

```
## '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 ...
```

```
head(sleep)
```

```
##           Id       SleepDay TotalSleepRecords TotalMinutesAsleep TotalTimeInBed
```

```
## 1 1503960366 4/12/2016 0:00                1              327            346
## 2 1503960366 4/13/2016 0:00                2              384            407
## 3 1503960366 4/15/2016 0:00                1              412            442
## 4 1503960366 4/16/2016 0:00                2              340            367
## 5 1503960366 4/17/2016 0:00                1              700            712
## 6 1503960366 4/19/2016 0:00                1              304            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 0:00" "4/13/2016 0:00" "4/15/2016 0:00" "4/16/2016 0:00" ...
##  $ 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 ...
```

From these summaries, the **ID** column would be our common denominator to merge both tables. Now I want to know how many unique participants in both **dailyactivity** and **sleep** dataframes.

```
n_distinct(dailyactivity$Id)
```

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

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

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

Given that, it shows that there are 33 participants in **dailyactivity** and 24 in **sleep**. I want to merge both dataframes so I can start my analysis. I will name it **activity_sleep_merged**

```
activity_sleep_merged <- merge(dailyactivity, sleep, by=c("Id"))
```

Then I want to verify the new total number of participants for my clean data.

```
n_distinct(activity_sleep_merged$Id)
```

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

Next would be to check the summary of my merged data.

```
head(activity_sleep_merged)
```

```
##           Id ActivityDate TotalSteps TotalDistance TrackerDistance
## 1 1503960366     5/7/2016      11992          7.71            7.71
## 2 1503960366     5/7/2016      11992          7.71            7.71
## 3 1503960366     5/7/2016      11992          7.71            7.71
## 4 1503960366     5/7/2016      11992          7.71            7.71
## 5 1503960366     5/7/2016      11992          7.71            7.71
## 6 1503960366     5/7/2016      11992          7.71            7.71
##   LoggedActivitiesDistance VeryActiveDistance ModeratelyActiveDistance
## 1                        0              2.46                     2.12
## 2                        0              2.46                     2.12
## 3                        0              2.46                     2.12
## 4                        0              2.46                     2.12
## 5                        0              2.46                     2.12
## 6                        0              2.46                     2.12
##   LightActiveDistance SedentaryActiveDistance VeryActiveMinutes
## 1                3.13                       0               37
## 2                3.13                       0               37
## 3                3.13                       0               37
## 4                3.13                       0               37
## 5                3.13                       0               37
## 6                3.13                       0               37
##   FairlyActiveMinutes LightlyActiveMinutes SedentaryMinutes Calories
## 1                  46                 175              833     1821
## 2                  46                 175              833     1821
## 3                  46                 175              833     1821
## 4                  46                 175              833     1821
## 5                  46                 175              833     1821
## 6                  46                 175              833     1821
##         SleepDay TotalSleepRecords TotalMinutesAsleep TotalTimeInBed
## 1 4/12/2016 0:00                 1                327            346
## 2 4/13/2016 0:00                 2                384            407
## 3 4/15/2016 0:00                 1                412            442
## 4 4/16/2016 0:00                 2                340            367
## 5 4/17/2016 0:00                 1                700            712
## 6 4/19/2016 0:00                 1                304            320
```

```
colnames(activity_sleep_merged)
```

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

```
str(activity_sleep_merged)
```

```
## 'data.frame':    12441 obs. of  19 variables:
##  $ Id                       : num  1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
```

```
##  $ ActivityDate            : chr  "5/7/2016" "5/7/2016" "5/7/2016" "5/7/2016" ...
##  $ TotalSteps              : int  11992 11992 11992 11992 11992 11992 11992 11992 11992 11992 ...
##  $ TotalDistance           : num  7.71 7.71 7.71 7.71 7.71 ...
##  $ TrackerDistance         : num  7.71 7.71 7.71 7.71 7.71 ...
##  $ LoggedActivitiesDistance: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ VeryActiveDistance      : num  2.46 2.46 2.46 2.46 2.46 ...
##  $ ModeratelyActiveDistance: num  2.12 2.12 2.12 2.12 2.12 ...
##  $ LightActiveDistance     : num  3.13 3.13 3.13 3.13 3.13 ...
##  $ SedentaryActiveDistance : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ VeryActiveMinutes       : int  37 37 37 37 37 37 37 37 37 37 ...
##  $ FairlyActiveMinutes     : int  46 46 46 46 46 46 46 46 46 46 ...
##  $ LightlyActiveMinutes    : int  175 175 175 175 175 175 175 175 175 175 ...
##  $ SedentaryMinutes        : int  833 833 833 833 833 833 833 833 833 833 ...
##  $ Calories                : int  1821 1821 1821 1821 1821 1821 1821 1821 1821 1821 ...
##  $ SleepDay                : chr  "4/12/2016 0:00" "4/13/2016 0:00" "4/15/2016 0:00" "4/16/2016 0:00
##  $ 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 ...
```

Lastly, I want to group the data by **Id** using the average for **total steps**, **calories**, **time spent sedentary**, **time in bed**, and **total time asleep**.

```
groupave_steps_calories_sedentary_bed_sleep <-
  activity_sleep_merged %>%
  group_by(Id) %>%
  summarise(Steps = mean(TotalSteps), Cal = mean(Calories), Sedentary = mean(SedentaryMinutes), Bedtime
```

I would want to check the summary of this dataframe.

```
head(groupave_steps_calories_sedentary_bed_sleep)
```

```
## # A tibble: 6 x 6
##           Id  Steps   Cal Sedentary Bedtime Sleep
##        <dbl>  <dbl> <dbl>     <dbl>   <dbl> <dbl>
## 1 1503960366 12117. 1816.      848.    383.  360.
## 2 1644430081  7283. 2811.     1162.    346   294
## 3 1844505072  2580. 1573.     1207.    961   652
## 4 1927972279   916. 2173.     1317.    438.  417
## 5 2026352035  5567. 1541.      689.    538.  506.
## 6 2320127002  4717. 1724.     1220.     69    61
```

```
colnames(groupave_steps_calories_sedentary_bed_sleep)
```

```
## [1] "Id"        "Steps"     "Cal"       "Sedentary" "Bedtime"   "Sleep"
```

```
str(groupave_steps_calories_sedentary_bed_sleep)
```

```
## tibble [24 x 6] (S3: tbl_df/tbl/data.frame)
##  $ Id       : num [1:24] 1.50e+09 1.64e+09 1.84e+09 1.93e+09 2.03e+09 ...
##  $ Steps    : num [1:24] 12117 7283 2580 916 5567 ...
##  $ Cal      : num [1:24] 1816 2811 1573 2173 1541 ...
##  $ Sedentary: num [1:24] 848 1162 1207 1317 689 ...
##  $ Bedtime  : num [1:24] 383 346 961 438 538 ...
##  $ Sleep    : num [1:24] 360 294 652 417 506 ...
```

Now I can start my analysis.

## Analysis

I want to start with some summary statistics from the clean data. I will get the average together with standard deviation for total steps, calories, sleep, and time in bed.

```
groupave_steps_calories_sedentary_bed_sleep %>%
  summarise(mean(Steps), sd(Steps), mean(Cal), sd(Cal), mean(Sedentary), sd(Sedentary), mean(Bedtime), s
```

```
## # A tibble: 1 x 10
##   `mean(Steps)` sd(Ste~1 mean(~2 sd(Ca~3 mean(~4 sd(Se~5 mean(~6 sd(Be~7 mean(~8
##           <dbl>    <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
## 1         7391.    3609.   2290.    561.    931.    228.    420.    174.    378.
## # ... with 1 more variable: `sd(Sleep)` <dbl>, and abbreviated variable names
## #   1: `sd(Steps)`, 2: `mean(Cal)`, 3: `sd(Cal)`, 4: `mean(Sedentary)`,
## #   5: `sd(Sedentary)`, 6: `mean(Bedtime)`, 7: `sd(Bedtime)`, 8: `mean(Sleep)`
```

### Total steps is directly proportional to calories burnt.

I want to see if there is a positive relationship between total steps and calories burnt. I will start by using the correlation function. I am making the assumption that total steps in general is correlated to higher calories burnt.

```
groupave_steps_calories_sedentary_bed_sleep %>% summarise(cor(Steps, Cal))
```

```
## # A tibble: 1 x 1
##   `cor(Steps, Cal)`
##               <dbl>
## 1             0.322
```

Now I want to plot the relationship between total steps and calories.

```
ggplot(data=groupave_steps_calories_sedentary_bed_sleep, mapping = aes(x=Steps, y=Cal)) + geom_point() +
```

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


Based from this analysis, there is a 0.322 correlation or 32.2%.

### Time spent sedentary is inversely proportional to calories burnt.

I want to see if there is a negative relationship between time sedentary and calories burnt. I will start by using the correlation function.

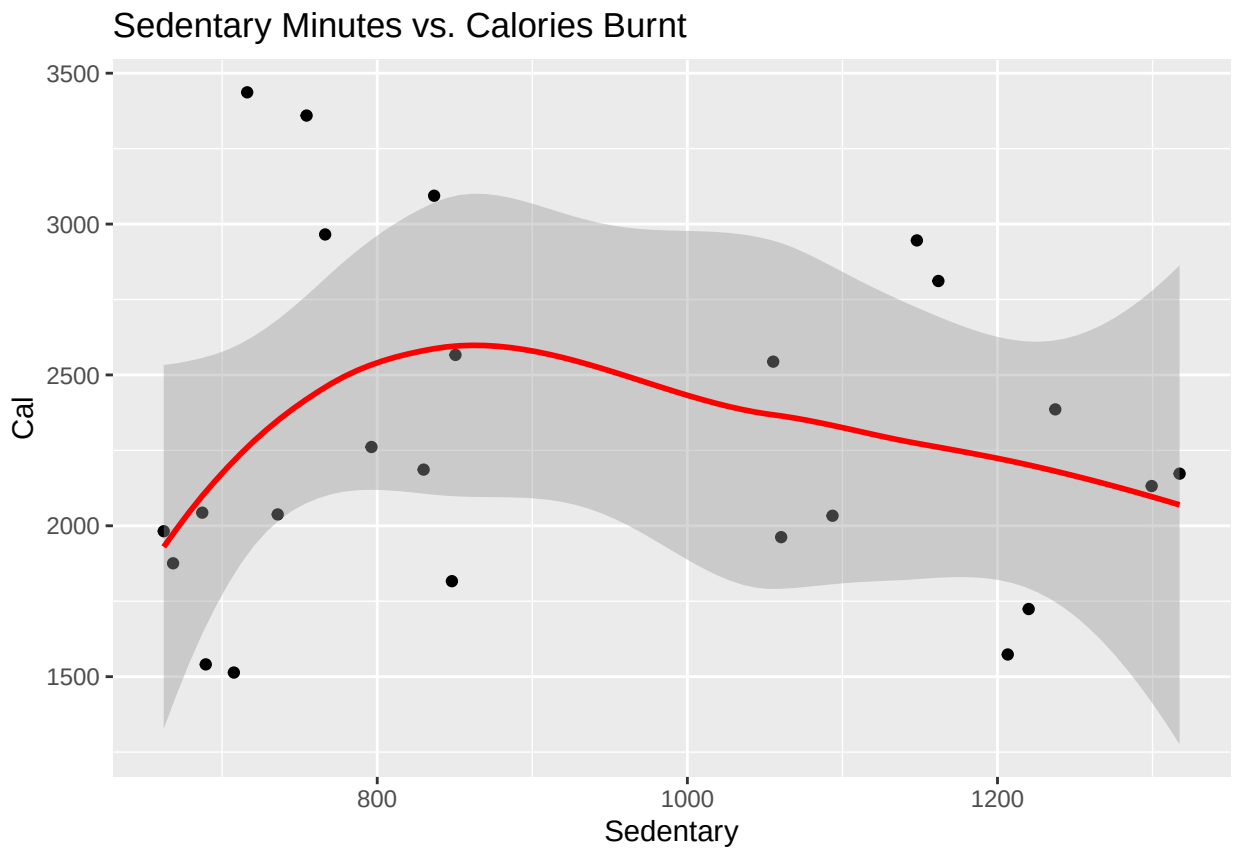
```
groupave_steps_calories_sedentary_bed_sleep %>% summarise(cor(Sedentary, Cal))
```

```
## # A tibble: 1 x 1
##   `cor(Sedentary, Cal)`
##                   <dbl>
## 1               -0.0545
```

Now I want to plot the relationship between time sedentary and calories.

```
ggplot(data=groupave_steps_calories_sedentary_bed_sleep, mapping = aes(x=Sedentary, y=Cal)) + geom_poin
```

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




Based from this analysis, there is a -0.0545 correlation or -5.45%.

### Time spent in bed is directly proportional to total minutes of sleep.

I want to see if there is a positive relationship between time in bed and total minutes of sleep. I will start by using the correlation function.

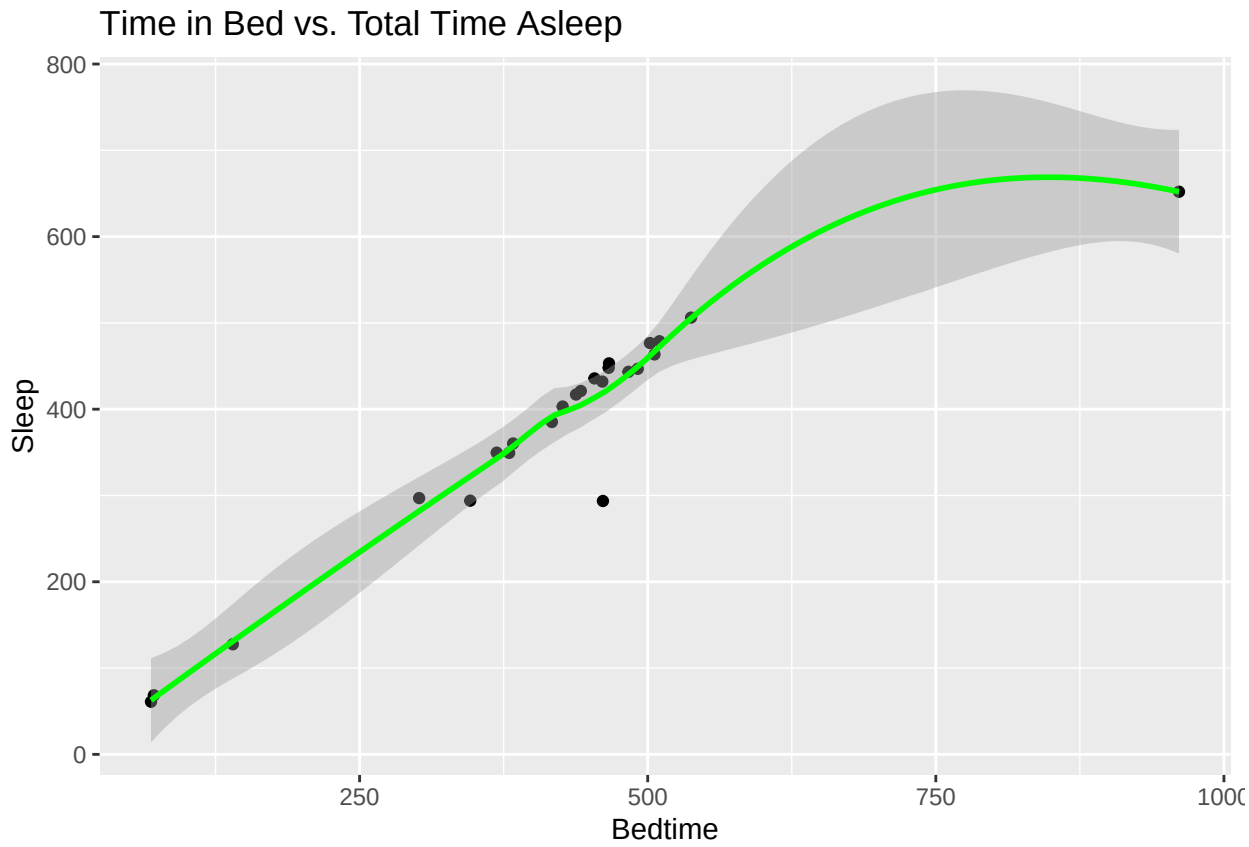
```
groupave_steps_calories_sedentary_bed_sleep %>% summarise(cor(Bedtime, Sleep))
```

```
## # A tibble: 1 x 1
##   `cor(Bedtime, Sleep)`
##                   <dbl>
## 1                 0.940
```

Now I want to plot the relationship between time in bed and total time asleep.

```
ggplot(data=groupave_steps_calories_sedentary_bed_sleep, mapping = aes(x=Bedtime, y=Sleep)) + geom_poin
```

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




Based from this analysis, there is a 0.940 correlation or 94%.

### Total minutes of rest is directly proportional to calories burnt.

I want to see if there is a positive relationship between total minutes of sleep and calories. I will start by using the correlation function.

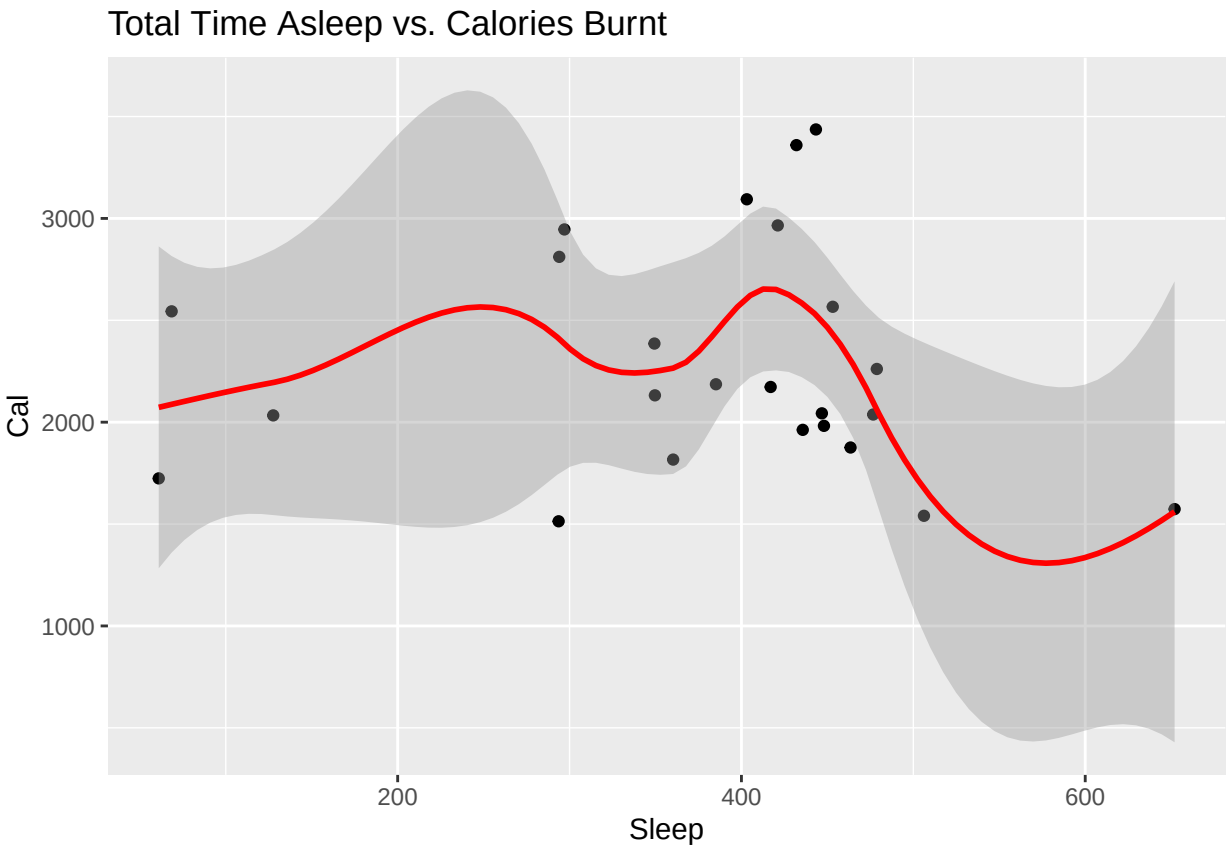
```
groupave_steps_calories_sedentary_bed_sleep %>% summarise(cor(Sleep, Cal))
```

```
## # A tibble: 1 x 1
##   `cor(Sleep, Cal)`
##               <dbl>
## 1           -0.0406
```

Now I want to plot the relationship between time sedentary and calories.

```
ggplot(data=groupave_steps_calories_sedentary_bed_sleep, mapping = aes(x=Sleep, y=Cal)) + geom_point() +
```

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




Based from this analysis, there is -0.0406 correlation or -4.06%.

## Summary and Recommendations

The last part of this case study would be focused on key findings and how the Bellabeat App can capitalize on several opportunities

I want to start with the most significant result of the research. Total time in bed is highly proportional to total time in bed. I would suggest that the Bellabeat App to enable users to have the option of having notifications when it detects sleep patterns like common bedtime. The app can show how many hours of sleep that the user can get for that notification. The app can also feature a "Do not disturb" mode so they can also get a better quality of sleep.

The second recommendation would be on total steps and total calories burnt. With the 32% correlation based on our findings, it is an opportunity to bump this rate by reminding the users of their daily step count. Bellabeat can also send out notifications on articles and studies on how having an active lifestyle benefits their users. The app can also feature a social aspect where users can compare their step count with their friends or an option to share their steps and if they achieved a new goal.

My next point would be the relationship between time spent sedentary and calories burnt. I made the assumption that less calories will be used whenever the user spent more time being sedentary. My findings show a -5.45% correlation, which was expected but I thought the margin would be greater than that. It should also be noted that I did not take into account other metrics like how active they are for the calories spent. This is still an opportunity for Bellabeat regarding the second recommendation.

Lastly, the relationship between total time asleep and calories burnt should be discussed. My assumption was that users would be able to burn more calories if they get more rest. The result shows a -4.06% which

invalidates my assumption. It is a very small margin just like the last point.

The clean sample size of 24 is not sufficient in my opinion but there are significant findings from the analyses. Furthermore, I used the average for all the categories for each users, within the two months that data was collected. I highly suggest having a larger sample size and a longer time period to gather data for further research.