Bellabeat Case Study Using R

Sasha Lea

2022-04-17

Bellabeat is 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.

Ask *1. What are some trends in smart device usage?* It was found that Tuesday, Wednesday, then Thursday were the top days where data was logged for activity. While Sunday and Monday smart device data were logged the least. I also wanted to coorelate the influence of sleep on calories and active days. *2. How could these trends apply to Bellabeat customers?*Sleeping between a 11000 to 12500 minutes or 183 hours a month, on average 6 hours a night provided the most calories burned. As well as being active 26 to 29 days burned the most calories. The optimal amount of sleep to stay active everyday is about 9000 minutes a month and 300 minutes a day.
*3. How could these trends help influence Bellabeat marketing strategy?*Recommend fine tuning the sleep application so that it can fit into the goals of their consumers.

Prepare Data was used from FitBit Fitness Tracker through a dataset made available through Mobius.

First I loaded all the packages I will be using

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

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.1'
## (as 'lib' is unspecified)
```

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

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.1'
## (as 'lib' is unspecified)
```

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

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

```
library(dplyr)
library(ggplot2)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

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

```
library(janitor)
```

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

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

###Then i need to upload the data I will be using

```
dailyActivity_merged <- read_csv("Capstone Bellabeat/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.
```

```
sleepDay_merged <- read_csv("Capstone Bellabeat/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.
```

Identify how data is organized, sort and filter data. Now I want a quick view of my dataset

```
head(dailyActivity_merged)
```

```
## # A tibble: 6 x 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(dailyActivity_merged)
```

```
##  [1] "Id"                       "ActivityDate"
```

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

```
head(sleepDay_merged)
```

```
## # A tibble: 6 x 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(dailyActivity_merged)
```

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

Process and Clean Data

I created a dataset with the information I was most interested in.

```
dailyactivity <- select(dailyActivity_merged, Id, ActivityDate,TotalSteps,TrackerDistance,SedentaryMinu
head(dailyactivity)
```

```
## # A tibble: 6 x 6
##           Id ActivityDate TotalSteps TrackerDistance SedentaryMinutes Calories
##        <dbl> <chr>             <dbl>           <dbl>            <dbl>    <dbl>
## 1 1503960366 4/12/2016         13162            8.5               728     1985
## 2 1503960366 4/13/2016         10735            6.97              776     1797
## 3 1503960366 4/14/2016         10460            6.74             1218     1776
## 4 1503960366 4/15/2016          9762            6.28              726     1745
## 5 1503960366 4/16/2016         12669            8.16              773     1863
## 6 1503960366 4/17/2016          9705            6.48              539     1728
```

For the analysis I want to focus on two things. How often these users are utilizing their device and what characteristics lead a user to use a smart device.

Cleaning my data

```
dailyactivity_names <-clean_names(dailyactivity)
sum(duplicated(dailyactivity_names))
```

```
## [1] 0
```

```r
sum(is.na(dailyactivity_names))
```

```
## [1] 0
```

```r
head(dailyactivity_names)
```

```
## # A tibble: 6 x 6
##          id activity_date total_steps tracker_distance sedentary_minut~ calories
##       <dbl> <chr>               <dbl>            <dbl>            <dbl>    <dbl>
## 1    1.50e9 4/12/2016           13162             8.5               728     1985
## 2    1.50e9 4/13/2016           10735             6.97              776     1797
## 3    1.50e9 4/14/2016           10460             6.74             1218     1776
## 4    1.50e9 4/15/2016            9762             6.28              726     1745
## 5    1.50e9 4/16/2016           12669             8.16              773     1863
## 6    1.50e9 4/17/2016            9705             6.48              539     1728
```

```r
sleep_cleannames <- clean_names(sleepDay_merged)
head(sleep_cleannames)
```

```
## # A tibble: 6 x 5
##           id sleep_day        total_sleep_rec~ total_minutes_a~ total_time_in_b~
##        <dbl> <chr>                       <dbl>            <dbl>            <dbl>
## 1 1503960366 4/12/2016 12:00~                1              327              346
## 2 1503960366 4/13/2016 12:00~                2              384              407
## 3 1503960366 4/15/2016 12:00~                1              412              442
## 4 1503960366 4/16/2016 12:00~                2              340              367
## 5 1503960366 4/17/2016 12:00~                1              700              712
## 6 1503960366 4/19/2016 12:00~                1              304              320
```

Now that we have determined that we do not have duplicates and changed the column names we will change the date

```r
activity <- dailyactivity_names %>% mutate(activity_date = mdy(activity_date), weekday = weekdays(activi

head(activity)
```

```
## # A tibble: 6 x 7
##          id activity_date total_steps tracker_distance sedentary_minut~ calories
##       <dbl> <date>              <dbl>            <dbl>            <dbl>    <dbl>
## 1    1.50e9 2016-04-12          13162             8.5               728     1985
## 2    1.50e9 2016-04-13          10735             6.97              776     1797
## 3    1.50e9 2016-04-14          10460             6.74             1218     1776
## 4    1.50e9 2016-04-15           9762             6.28              726     1745
## 5    1.50e9 2016-04-16          12669             8.16              773     1863
## 6    1.50e9 2016-04-17           9705             6.48              539     1728
## # ... with 1 more variable: weekday <chr>
```

I also want to see how many users are unique and how many days this study was conducted

```r
n_distinct(activity$id)
```

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

```r
n_distinct(activity$activity_date)
```

```
## [1] 31
```

```r
n_distinct(activity$weekday)
```

```
## [1] 7
```

```
n_distinct(sleep_cleannames$id)
```

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

Analyze *1. Aggregate your data so it's useful and accessible. 2. Organize and format your data. 3. Perform calculations. 4. Identify trends and relationships.*

I found that 33 of the participants used the smart device during activity and 24 of those 33 users logged sleep data as well. Bellabeats' Leaf offers a automatic sleep tracker from 9 p.m. to 9 a.m. For those participants that used Fitbit during sleep I want to see how much sleep each particular user received.

```
sleep_min_id <- sleep_cleannames %>% group_by(id) %>%  summarise(total_asleep = sum(total_minutes_asleep

head(sleep_min_id)
```

```
## # A tibble: 6 x 2
##           id total_asleep
##        <dbl>        <dbl>
## 1 1503960366         9007
## 2 1644430081         1176
## 3 1844505072         1956
## 4 1927972279         2085
## 5 2026352035        14173
## 6 2320127002           61
```

I also wanted to see how often participants used their device. It was found that the smart device participants actively used their devices. This could create some bias for our stakeholder because most of these participants were realitively active.

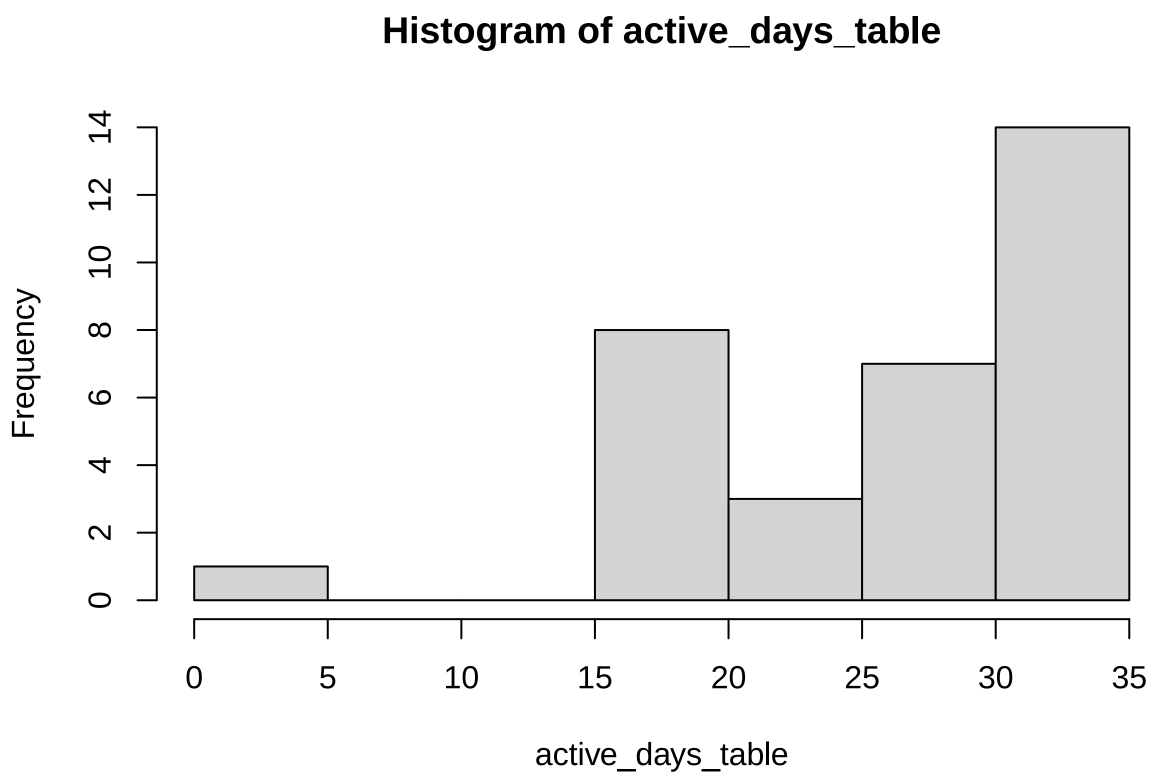
```
activity_per_id <- activity %>% group_by(id) %>% summarize(active_days = sum(tracker_distance != 0), nor

active_monthly_usage <- abs(activity_per_id)

active_monthly_usage %>% summarise(total_active_days = sum(active_days), toal_non_active_days = sum(non
```

```
## # A tibble: 1 x 2
##   total_active_days toal_non_active_days
##               <int>                <dbl>
## 1               862                  161
```

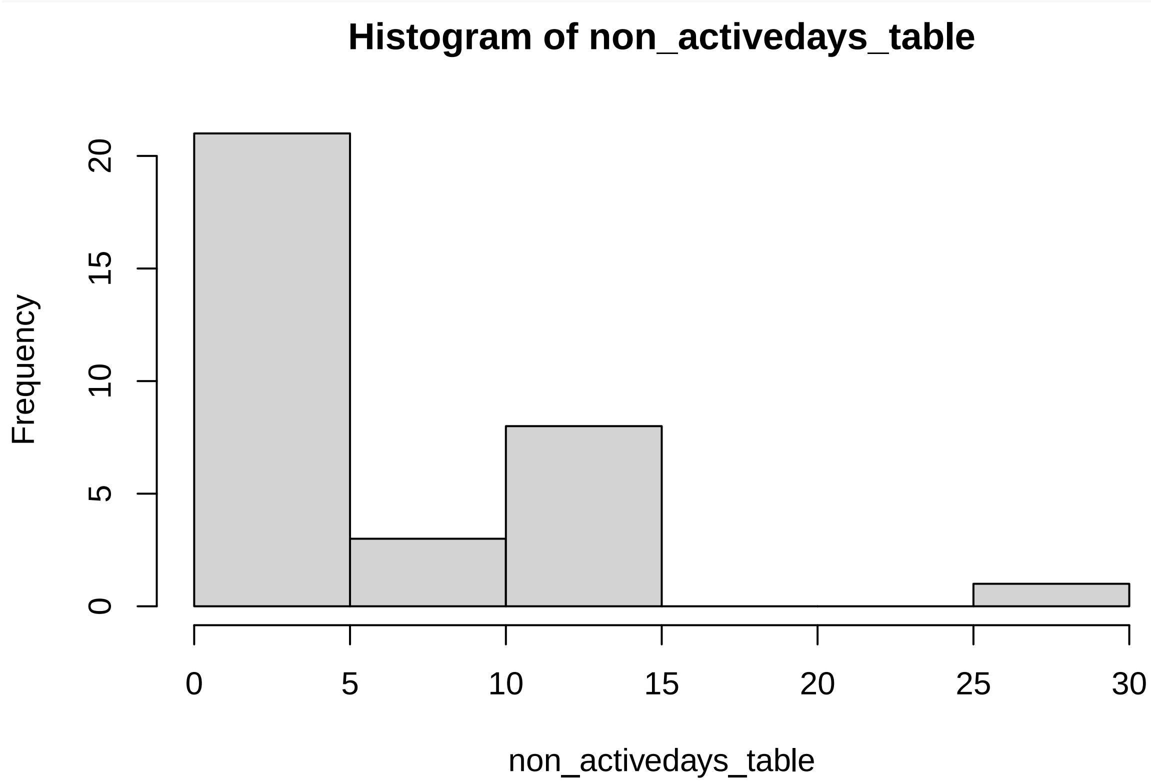
```
head(active_monthly_usage)
```

```
## # A tibble: 6 x 3
##           id active_days non_active_days
##        <dbl>       <int>           <dbl>
## 1 1503960366          30               1
## 2 1624580081          31               0
## 3 1644430081          30               1
## 4 1844505072          20              11
## 5 1927972279          17              14
## 6 2022484408          31               0
```

```
active_days_table <- pull(active_monthly_usage,active_days)
hist(active_days_table)
```

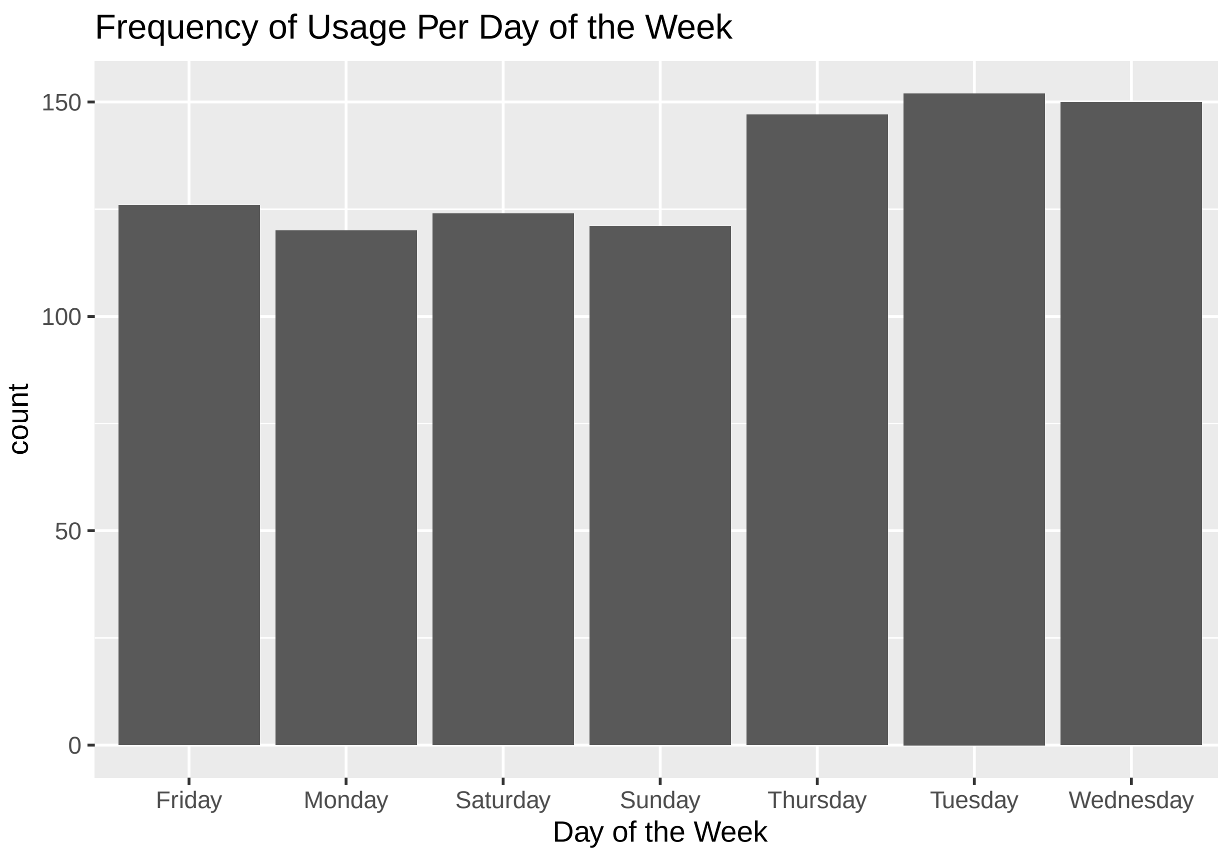



```
non_activedays_table <-pull(active_monthly_usage,non_active_days)
hist(non_activedays_table)
```



It is also important for the stakeholders to know what days of the weekdays smart devices are most utlizied most.

```
ggplot(data=activity)+geom_bar(mapping=aes(x=weekday))+ggtitle("Frequency of Usage Per Day of the Week")
```



It was found that Tuesday, Wednesday, then Thursday were the top days where data was logged for activity. While Sunday and Monday smart device data were logged the least. I also wanted to coorelate the influence of sleep on calories and active days.

I first calculated the total amount of calories burned per id.

```
activity_cal_id <-activity %>% group_by(id) %>% summarize(total_cal=sum(calories),total_sedentary_minut
head(activity_cal_id)
```

```
## # A tibble: 6 x 3
##           id total_cal total_sedentary_minutes
##        <dbl>     <dbl>                   <dbl>
## 1 1503960366     56309                   26293
## 2 1624580081     45984                   38990
## 3 1644430081     84339                   34856
## 4 1844505072     48778                   37405
## 5 1927972279     67357                   40840
## 6 2022484408     77809                   34490
```

Then I merged all the data together in order to get more information about how sleep effects the habits of the consumer.

```
merge_cal_sleep <- merge(activity_cal_id, sleep_min_id, by="id")
merge_cal_sleep_act <- merge(merge_cal_sleep, active_monthly_usage, by="id")
head(merge_cal_sleep_act)
```

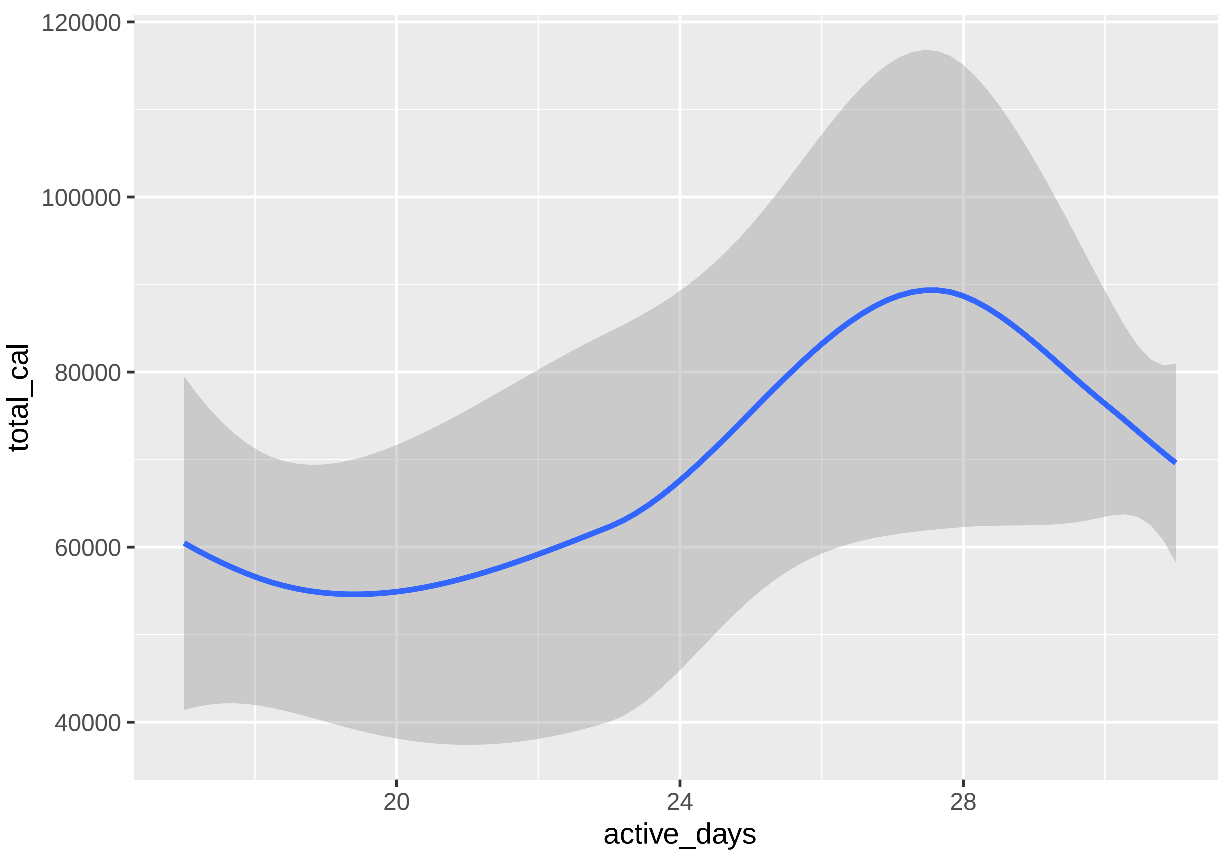
```
##           id total_cal total_sedentary_minutes total_asleep active_days
## 1 1503960366     56309                   26293         9007          30
```

```
## 2 1644430081      84339                      34856         1176          30
## 3 1844505072      48778                      37405         1956          20
## 4 1927972279      67357                      40840         2085          17
## 5 2026352035      47760                      21372        14173          31
## 6 2320127002      53449                      37823           61          31
##   non_active_days
## 1               1
## 2               1
## 3              11
## 4              14
## 5               0
## 6               0
```

It was found that sleeping between a 11000 to 12500 minutes or 183 hours a month, on average 6 hours a night provided the most calories burned. As well as being active 26 to 29 days burned the most calories.

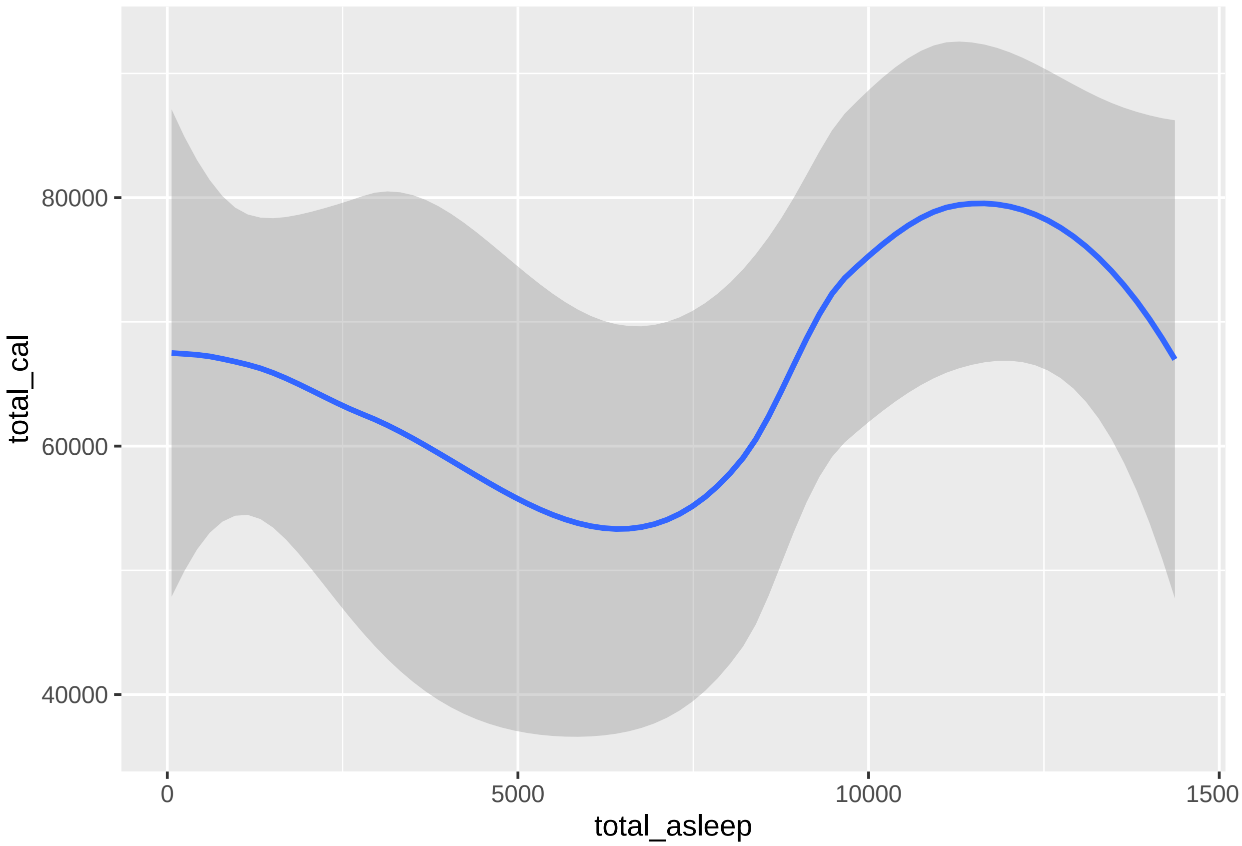
```
ggplot(data=merge_cal_sleep_act)+geom_smooth(mapping=aes(x=active_days,y=total_cal))
```

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




```
ggplot(data=merge_cal_sleep_act)+geom_smooth(mapping=aes(x=total_asleep,y=total_cal))
```

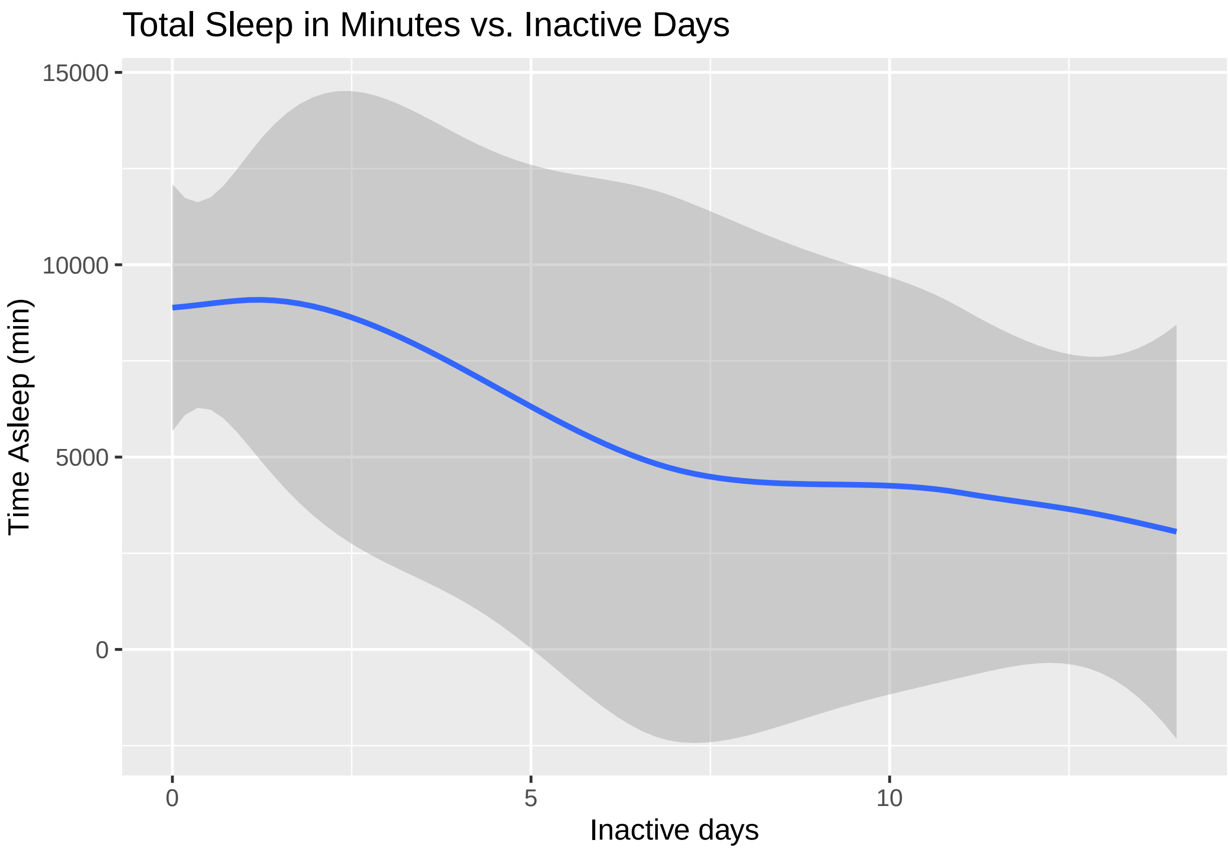
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Bellabeat focuses on womens health and empowering women,which isn't mainly focused on calories, but well being and body positivity. My study also focuses on how sleep affects activity of the smart device user. It was found the optimal amount of sleep to stay active everyday is about 9000 minutes a month and 300 minutes a day.

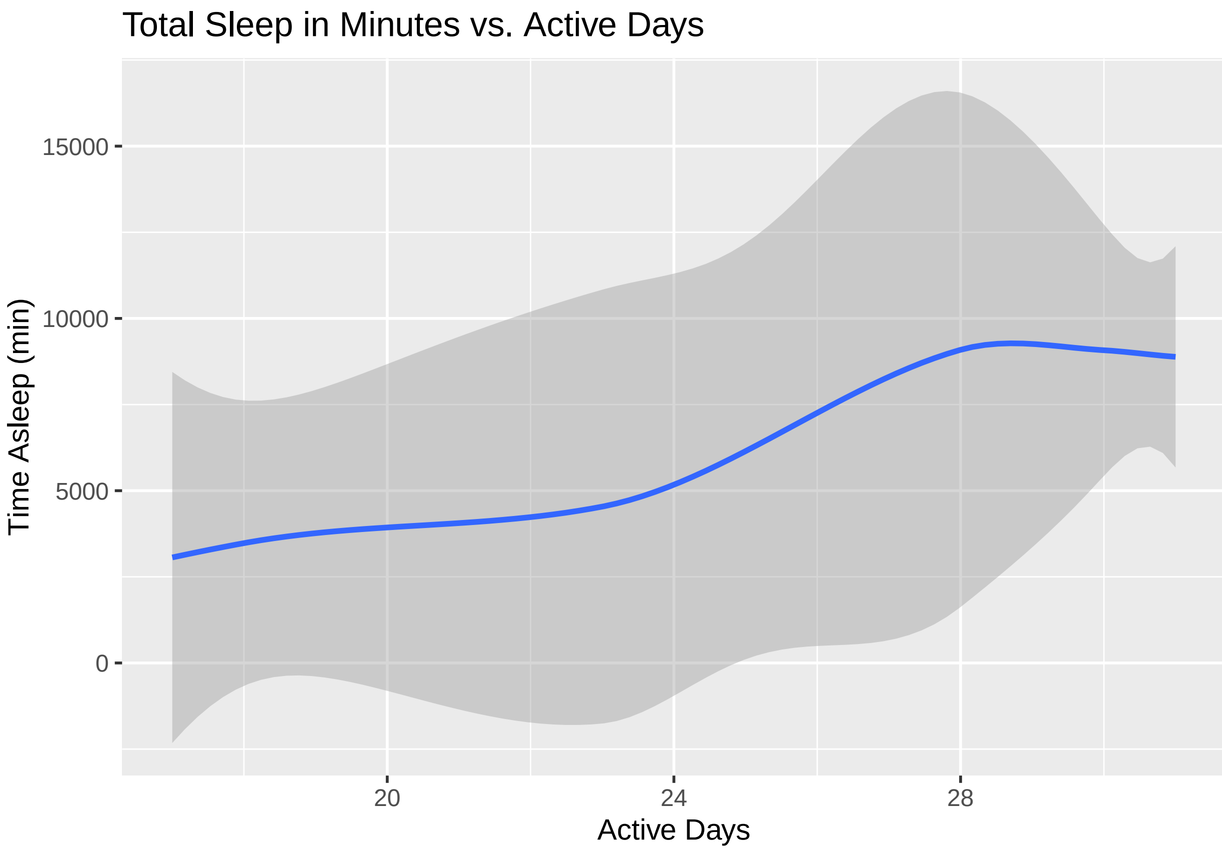
```
ggplot(data=merge_cal_sleep_act)+geom_smooth(mapping=aes(x=non_active_days,y=total_asleep))+ggtitle("To
```

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




```
ggplot(data=merge_cal_sleep_act)+geom_smooth(mapping=aes(x=active_days,y=total_asleep))+ggtitle("Total S
```

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



Act

What is your final conclusion based on your analysis? Bellabeat currently has a sleep tracker, however it has limitations and excludes consumers who do not sleep between the hours of 9 p.m to 9 a.m. There is also limitations with the fitbit data set. Only 33 participants were used and out of those 33 only 24 utilized the sleep function.

What next steps would you or your stakeholders take based on your findings? I would recommend fine tuning the sleep application so that it can fit into all their consumers.

Is there additional data you could use to expand on your findings? Yes, taking data of how active the typical Bellabeat consumer would like to be and fine tuning to the goals of most of the Bellabeat consumer.