Bellabeat Case Study

Tichina Buckle

2022-05-05

In this case study, I investigated a dataset containing the user data of 30 fitbit user to gain insights and derive marketing strategies to unlock new growth opportunities for Bellabeat, through analysis of consumers usage of non-bellabeat smart devices and compare trends identified to bellabeat's consumer smart device usage.

Dateset: https://www.kaggle.com/datasets/arashnic/fitbit

The Ask Phase

For the Ask Phase, I wanted to clear identify what was being asked of me by the stakeholder. Though carefully reading the pdf given I was able to identify the business task.

Business Task

Derive marketing strategies to unlock new growth opportunities for Bellabeat, through analysis of consumers usage of non-bellabeat smart devices and compare thrends identified to bellabeat's consumer smart device usage.

The Prepare Phase

In this phase of the analysis process we retrieve the data for cleaning and analysis. The data was downloaded from Kaggle [Dataset] https://www.kaggle.com/datasets/arashnic/fitbit and the .csv file the contained data useful to the analysis were imported to R Studio Cloud.

Why R and R Studio Cloud was selected

I selected R and R Studio Cloud to do my data cleaning and analysis, becaue R provided many useful functions for data cleaning, data analysis and data visualization through packages such as tidyverse, ggplot2 and dplyr.

Setting Up my R environment

By installed and loaded at the packages I need to complete my analysis.

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

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

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

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

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

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

```
install.packages("tidyr")
```

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

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

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

Loading these previously installed packages.

```
library(tidyverse)
```

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

```
## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.9
## 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(ggplot2)
library(dplyr)
library(tidyr)
library(skimr)
```

Above I imported the dataset contain information on the users daily_activity and assisted it to the dataframe daily_activity.

After importing and assiting the data, I used the head (gives the first 6 roles of the dataframe) and str (show the properties of the each column in the dataframe) to quickly check over the dataframe.

This process repeated to import the sleep and weight data of users as well. Assisting the data stored in the .csv to the sleep_day and weight_log dataframes respectively.

```
#importing dataset and assign to daily_activity
daily_activity <- read.csv('dailyActivity_merged.csv')
#gives the first 6 roles to check data
head(daily_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
```

```
#check the properties of each value in the dataset
str(daily_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 ...
```

```
#importing dataset and assign to sleep_day
sleep_day <- read.csv('sleepDay_merged.csv')
head(sleep_day)
```

```
##           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
```

```
str(sleep_day)
```

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

```
#importing dataset and assign to weight_log
weight_log <- read.csv('weightLogInfo_merged.csv')
head(weight_log)
```

```
##           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
```

```
str(weight_log)
```

```
## '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" "4/21/2
##  $ 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 ...
```

The Process Phase

Then we move on to the Data Cleaning.

Here I choose to change all the column to lower and save these changes to a new dataframe called daily_activity_new. Change all columns to lower case ensured all names kept and constant format, both the existing column and any columns I would create in the future. This also allowed me to type a bit faster.

I then used the colnames() to verify the change was successful, as the colname function prints the names of all columns in a dataframe.

```
#changes colnames to to lower case and saves changes in a new dataset called daily_activity_new
daily_activity_new <- rename_with(daily_activity, tolower)
#view all column names to verify if changes were successful
colnames(daily_activity_new)
```

```
##  [1] "id"                       "activitydate"
##  [3] "totalsteps"               "totaldistance"
##  [5] "trackerdistance"          "loggedactivitiesdistance"
##  [7] "veryactivedistance"       "moderatelyactivedistance"
##  [9] "lightactivedistance"      "sedentaryactivedistance"
```

```
## [11] "veryactiveminutes"        "fairlyactiveminutes"
## [13] "lightlyactiveminutes"     "sedentaryminutes"
## [15] "calories"
```

After change the column names to lower case, I check for duplicate row. The nrow() function, followed by the nrow() + unique() to compare the number of row in the table vs the number of unique row. If the number the of row was equal for both, then were not duplicate rows found. If the number of row was greater than the number of unique, then there exist duplicate rows and further cleaning was necessary.

```
#checks the number of row in dataset
nrow(daily_activity_new)
```

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

```
#checks the number of unique row in dataset to find duplicates
nrow(unique(daily_activity_new))
```

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

```
#no duplicates in this table
```

The check of the sleep_day_new table show a greater number of rows than unique row and such I used the distinct() function to delete those duplicate rows from the table and create and new table called sleep_day_new_v2.

```
sleep_day_new <- rename_with(sleep_day, tolower)
colnames(sleep_day_new)
```

```
## [1] "id"                 "sleepday"           "totalsleeprecords"
## [4] "totalminutesasleep" "totaltimeinbed"
```

```
nrow(sleep_day_new)
```

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

```
nrow(unique(sleep_day_new))
```

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

```
#3 duplicates rows found

#removes duplicate rows
sleep_day_new_v2 <- distinct(sleep_day_new)
#view number of row again to verify changes
nrow(sleep_day_new_v2)
```

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

The same process as with the previous tables was repeated with the weight_log_new table, where no duplicates were found.

```
weight_log_new <- rename_with(weight_log, tolower)
colnames(weight_log_new)
```

```
## [1] "id"             "date"           "weightkg"       "weightpounds"
## [5] "fat"            "bmi"            "ismanualreport" "logid"
```

```
nrow(weight_log_new)
```

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

```
nrow(unique(weight_log_new))
```

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

```
#no duplicates
```

For the last part of my data processing before moving into the analyze phase, I create a new column in the daily_activity table using the mutate function called total_active_minutes by adding together the veryactiveminutes,fairlyactiveminutes and lightlyactiveminute.

```
#creates new column called total_active_minutes from total of veryactiveminutes,fairlyactiveminutes and
daily_activity_new_v2 <- daily_activity_new %>%
  mutate(total_active_minutes = veryactiveminutes + fairlyactiveminutes + lightlyactiveminutes)
```

The Analyze Phase

As first step of my analysis phase, I used the skim_without_charts() and summary() function to get a quick overview of the data in my dataframe to help spot connects for further analysis

```
#summary of dataframe to help spot connects for analysis
skim_without_charts(daily_activity_new_v2)
```

Table 1: Data summary

Name	daily_activity_new_v2
Number of rows	940
Number of columns	16
_____	
Column type frequency:	
character	1
numeric	15
_____	
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	1.503960e+09	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	1.027200e+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	2.060000e+00	0	0.000000e+00	2.050000e-01	2.092000e+00	2.101000e+01
moderatelyactivedistance	0	1	5.700000e-01	8.800000e-01	0	0.000000e+00	8.000000e-01	6.480000e+00	7.980000e+00
lightactivedistance	0	1	3.340000e+00	2.040000e+00	0	1.950000e+00	3.060000e+00	4.780000e+00	1.071000e+01

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
sedentaryactivedistance	0	1	0.000000e+00	0.000000e-02	0	0.000000e+00	0.000000e+00	0.000000e+00	1.000000e-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	5.280000e+02
sedentaryminutes	0	1	9.912100e+02	3.012700e+02	0	7.297500e+02	1.057500e+03	1.329500e+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
total_active_minutes	0	1	2.275400e+02	2.217800e+02	0	1.467500e+02	2.270000e+02	3.272500e+02	1.052000e+03

```
skim_without_charts(sleep_day_new_v2)
```

Table 4: Data summary

Name	sleep_day_new_v2
Number of rows	410
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	4.994963e+09	2.060863e+09	1.503960e+09	3.977334e+09	4.702922e+09	6.962181e+09	8.792010e+09
totalsleeprecords	0	1	1.120000e+00	3.500000e-01	1	1.000000e+00	1.0	1	3
totalminutesasleep	0	1	4.191700e+02	1.186400e+02	58	3.610000e+02	432.5	490	796
totaltimeinbed	0	1	4.584800e+02	1.274600e+02	61	4.037500e+02	463.0	526	961

```
skim_without_charts(weight_log_new)
```

Table 7: Data summary

Name	weight_log_new
Number of rows	67
Number of columns	8
Column type frequency:	
character	2

Table 7: Data summary

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	2.950322e+09	1.503960e+09	4.962181e+09	4.962181e+09	8.877689e+09	8.877689e+09
weightkg	0	1.00	7.204000e+01	1.392000e+01	5.126000e+01	6.114000e+01	6.125000e+01	8.150500e+01	1.335000e+02
weightpounds	0	1.00	1.588100e+02	3.070000e+01	1.129600e+02	1.353600e+02	1.377900e+02	1.797500e+02	2.943200e+02
fat	65	0.03	2.350000e+01	1.120000e+01	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
logid	0	1.00	1.461772e+12	2.829948e+08	1.460444e+12	1.461079e+12	1.461802e+12	1.462375e+12	1.463098e+12

```
summary(daily_activity_new_v2)
```

```
##        id            activitydate         totalsteps    totaldistance
##  Min.   :1.504e+09   Length:940         Min.   :    0   Min.   : 0.000
##  1st Qu.:2.320e+09   Class :character   1st Qu.: 3790   1st Qu.: 2.620
##  Median :4.445e+09   Mode  :character   Median : 7406   Median : 5.245
##  Mean   :4.855e+09                      Mean   : 7638   Mean   : 5.490
##  3rd Qu.:6.962e+09                      3rd Qu.:10727   3rd Qu.: 7.713
##  Max.   :8.878e+09                      Max.   :36019   Max.   :28.030
##  trackerdistance  loggedactivitiesdistance veryactivedistance
##  Min.   : 0.000   Min.   :0.0000           Min.   : 0.000
##  1st Qu.: 2.620   1st Qu.:0.0000           1st Qu.: 0.000
##  Median : 5.245   Median :0.0000           Median : 0.210
##  Mean   : 5.475   Mean   :0.1082           Mean   : 1.503
##  3rd Qu.: 7.710   3rd Qu.:0.0000           3rd Qu.: 2.053
##  Max.   :28.030   Max.   :4.9421           Max.   :21.920
##  moderatelyactivedistance lightactivedistance sedentaryactivedistance
##  Min.   :0.0000           Min.   : 0.000      Min.   :0.000000
##  1st Qu.:0.0000           1st Qu.: 1.945      1st Qu.:0.000000
##  Median :0.2400           Median : 3.365      Median :0.000000
##  Mean   :0.5675           Mean   : 3.341      Mean   :0.001606
##  3rd Qu.:0.8000           3rd Qu.: 4.782      3rd Qu.:0.000000
##  Max.   :6.4800           Max.   :10.710      Max.   :0.110000
##  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
##     calories    total_active_minutes
##  Min.   :   0   Min.   :  0.0
##  1st Qu.:1828   1st Qu.:146.8
##  Median :2134   Median :247.0
##  Mean   :2304   Mean   :227.5
##  3rd Qu.:2793   3rd Qu.:317.2
##  Max.   :4900   Max.   :552.0
```

```
summary(sleep_day_new_v2)
```

```
##        id              sleepday         totalsleeprecords totalminutesasleep
##  Min.   :1.504e+09   Length:410         Min.   :1.00      Min.   : 58.0
##  1st Qu.:3.977e+09   Class :character   1st Qu.:1.00      1st Qu.:361.0
##  Median :4.703e+09   Mode  :character   Median :1.00      Median :432.5
##  Mean   :4.995e+09                      Mean   :1.12      Mean   :419.2
##  3rd Qu.:6.962e+09                      3rd Qu.:1.00      3rd Qu.:490.0
##  Max.   :8.792e+09                      Max.   :3.00      Max.   :796.0
##  totaltimeinbed
##  Min.   : 61.0
##  1st Qu.:403.8
##  Median :463.0
##  Mean   :458.5
##  3rd Qu.:526.0
##  Max.   :961.0
```

```
summary(weight_log_new)
```

```
##        id                date              weightkg       weightpounds
##  Min.   :1.504e+09   Length:67          Min.   : 52.60   Min.   :116.0
##  1st Qu.:6.962e+09   Class :character   1st Qu.: 61.40   1st Qu.:135.4
##  Median :6.962e+09   Mode  :character   Median : 62.50   Median :137.8
##  Mean   :7.009e+09                      Mean   : 72.04   Mean   :158.8
##  3rd Qu.:8.878e+09                      3rd Qu.: 85.05   3rd Qu.:187.5
##  Max.   :8.878e+09                      Max.   :133.50   Max.   :294.3
##
##       fat              bmi        ismanualreport         logid
##  Min.   :22.00   Min.   :21.45   Length:67          Min.   :1.460e+12
##  1st Qu.:22.75   1st Qu.:23.96   Class :character   1st Qu.:1.461e+12
##  Median :23.50   Median :24.39   Mode  :character   Median :1.462e+12
##  Mean   :23.50   Mean   :25.19                      Mean   :1.462e+12
##  3rd Qu.:24.25   3rd Qu.:25.56                      3rd Qu.:1.462e+12
##  Max.   :25.00   Max.   :47.54                      Max.   :1.463e+12
##  NA's   :65
```

Using ggplot2 I plotted a scatter and line graph illustrating the relationship between steps taken and calories burned. The results showed a positive relationship between steps taken and calories burned of the use.

```
#Scatter and line graph illustrating the relationship between steps taken and calories burned
ggplot(data = daily_activity_new_v2) +
  geom_point(mapping = aes(x = totalsteps, y = calories, color = totalsteps)) +
  geom_smooth(mapping = aes(x = totalsteps, y = calories)) +
  #Adding Label
  labs(title = "Calories Burned Vs Total Number of Steps", x = "total number of steps", y = "calories bu
```

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

I creates a single table containing values from daily_activity_new_v2 and weight_log_new by ID to use in analysis of relationship between steps taken and weight

```
#creates a single dataframe containing values from daily_activity_new_v2 and weight_log_new by ID
weight_activity_merged <- merge(daily_activity_new_v2, weight_log_new, by="id")
#view overview of new dataframe
head(weight_activity_merged)
```

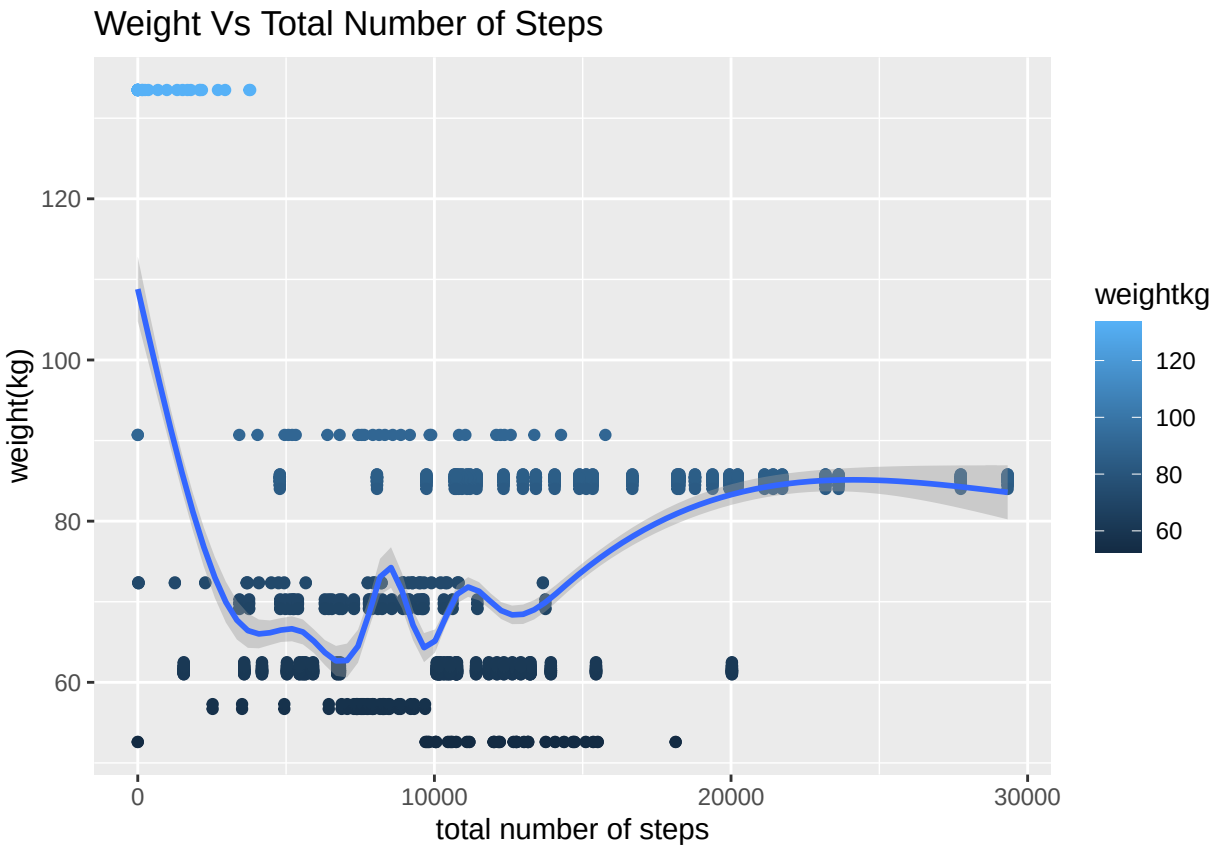
```
##           id activitydate totalsteps totaldistance trackerdistance
## 1 1503960366    4/16/2016      12669          8.16            8.16
## 2 1503960366    4/16/2016      12669          8.16            8.16
## 3 1503960366    4/18/2016      13019          8.59            8.59
## 4 1503960366    4/18/2016      13019          8.59            8.59
## 5 1503960366    4/15/2016       9762          6.28            6.28
## 6 1503960366    4/15/2016       9762          6.28            6.28
##   loggedactivitiesdistance veryactivedistance moderatelyactivedistance
## 1                        0              2.71                     0.41
## 2                        0              2.71                     0.41
## 3                        0              3.25                     0.64
## 4                        0              3.25                     0.64
## 5                        0              2.14                     1.26
## 6                        0              2.14                     1.26
##   lightactivedistance sedentaryactivedistance veryactiveminutes
## 1                5.04                       0                36
## 2                5.04                       0                36
## 3                4.71                       0                42
## 4                4.71                       0                42
## 5                2.83                       0                29
```

```
## 6               2.83                        0               29
##   fairlyactiveminutes lightlyactiveminutes sedentaryminutes calories
## 1                  10                   221              773     1863
## 2                  10                   221              773     1863
## 3                  16                   233             1149     1921
## 4                  16                   233             1149     1921
## 5                  34                   209              726     1745
## 6                  34                   209              726     1745
##   total_active_minutes                  date weightkg weightpounds fat   bmi
## 1                  267 5/2/2016 11:59:59 PM     52.6     115.9631  22 22.65
## 2                  267 5/3/2016 11:59:59 PM     52.6     115.9631  NA 22.65
## 3                  291 5/2/2016 11:59:59 PM     52.6     115.9631  22 22.65
## 4                  291 5/3/2016 11:59:59 PM     52.6     115.9631  NA 22.65
## 5                  272 5/2/2016 11:59:59 PM     52.6     115.9631  22 22.65
## 6                  272 5/3/2016 11:59:59 PM     52.6     115.9631  NA 22.65
##   ismanualreport        logid
## 1           True 1.462234e+12
## 2           True 1.462320e+12
## 3           True 1.462234e+12
## 4           True 1.462320e+12
## 5           True 1.462234e+12
## 6           True 1.462320e+12
```

Using the table I had created previously, I plotted Scatter and line graph illustrating the relationship between steps taken and weight. The graph showed no clear relationship between the two variables.

```
#Scatter and line graph illustrating the relationship between steps taken and weight
ggplot(data = weight_activity_merged) +
  geom_point(mapping = aes(x = totalsteps, y = weightkg, color = weightkg)) +
  geom_smooth(mapping = aes(x = totalsteps, y = weightkg)) +
  #Adding Label
  labs(title = "Weight Vs Total Number of Steps", x = "total number of steps", y = "weight(kg)")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



I creates a single table containing values from daily_activity_new_v2 and sleep_day_new_v2 by ID for use in analysis of relationship between minutes sedentary and total minutes asleep.

```
#creates a single dataframe containing values from daily_activity_new_v2 and sleep_day_new_v2 by ID
sleep_activity_merged <- merge(daily_activity_new_v2, sleep_day_new_v2, by="id")
#view overview of new dataframe
head(sleep_activity_merged)
```

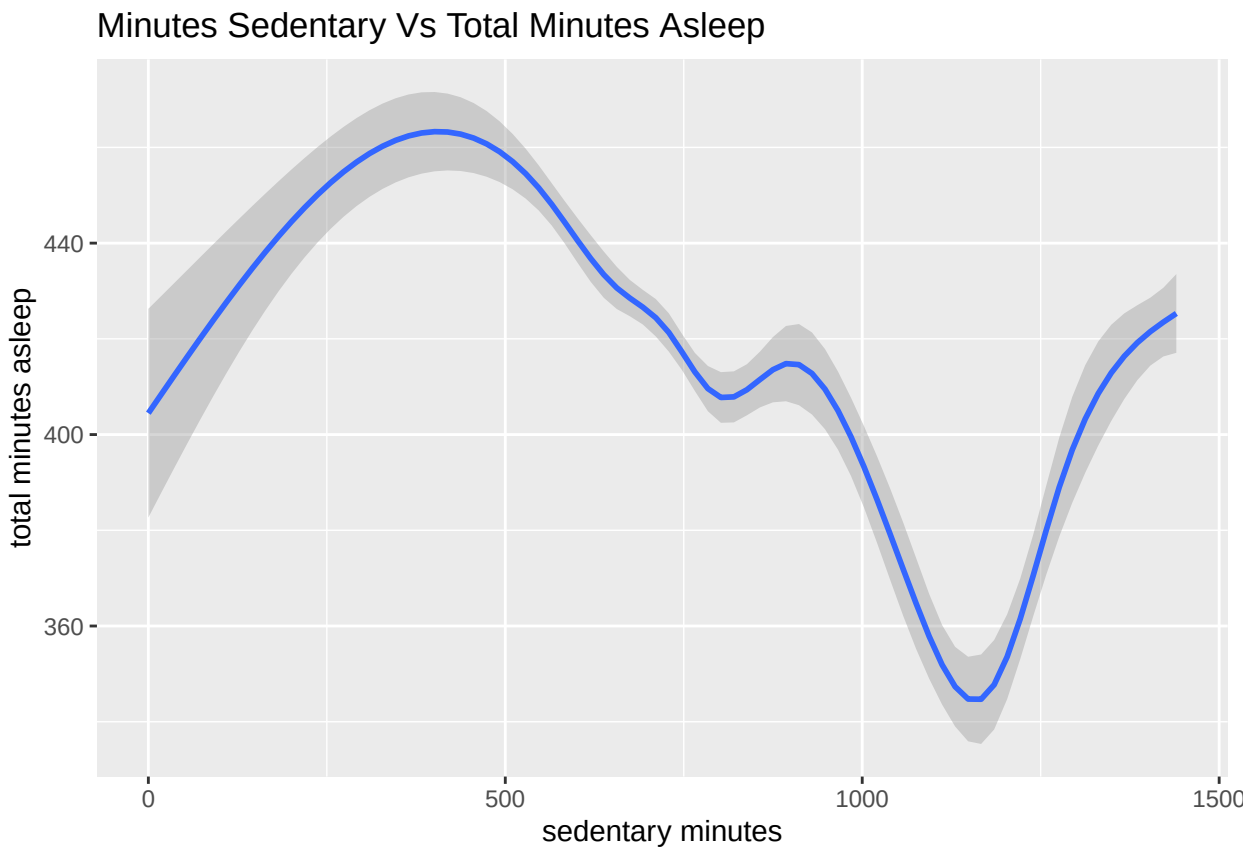
```
##           id activitydate totalsteps totaldistance trackerdistance
## 1 1503960366     5/7/2016      11992          7.71            7.71
## 2 1503960366     5/7/2016      11992          7.71            7.71
## 3 1503960366     5/7/2016      11992          7.71            7.71
## 4 1503960366     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
##   total_active_minutes              sleepday totalsleeprecords
## 1                  258 4/12/2016 12:00:00 AM                 1
## 2                  258 4/13/2016 12:00:00 AM                 2
## 3                  258 4/15/2016 12:00:00 AM                 1
## 4                  258 4/16/2016 12:00:00 AM                 2
## 5                  258 4/17/2016 12:00:00 AM                 1
## 6                  258 4/19/2016 12:00:00 AM                 1
##   totalminutesasleep totaltimeinbed
## 1                327            346
## 2                384            407
## 3                412            442
## 4                340            367
## 5                700            712
## 6                304            320
```

Using the table I had created previously, I plotted line graph illustrating the relationship between minutes sedentary and total minutes asleep.

```
#relationship between minutes sedentary and total minutes asleep
ggplot(data = sleep_activity_merged) +
  geom_smooth(mapping = aes(x = sedentaryminutes, y = totalminutesasleep)) +
  #Adding Label
  labs(title = "Minutes Sedentary Vs Total Minutes Asleep", x = "sedentary minutes", y = "total minutes
```

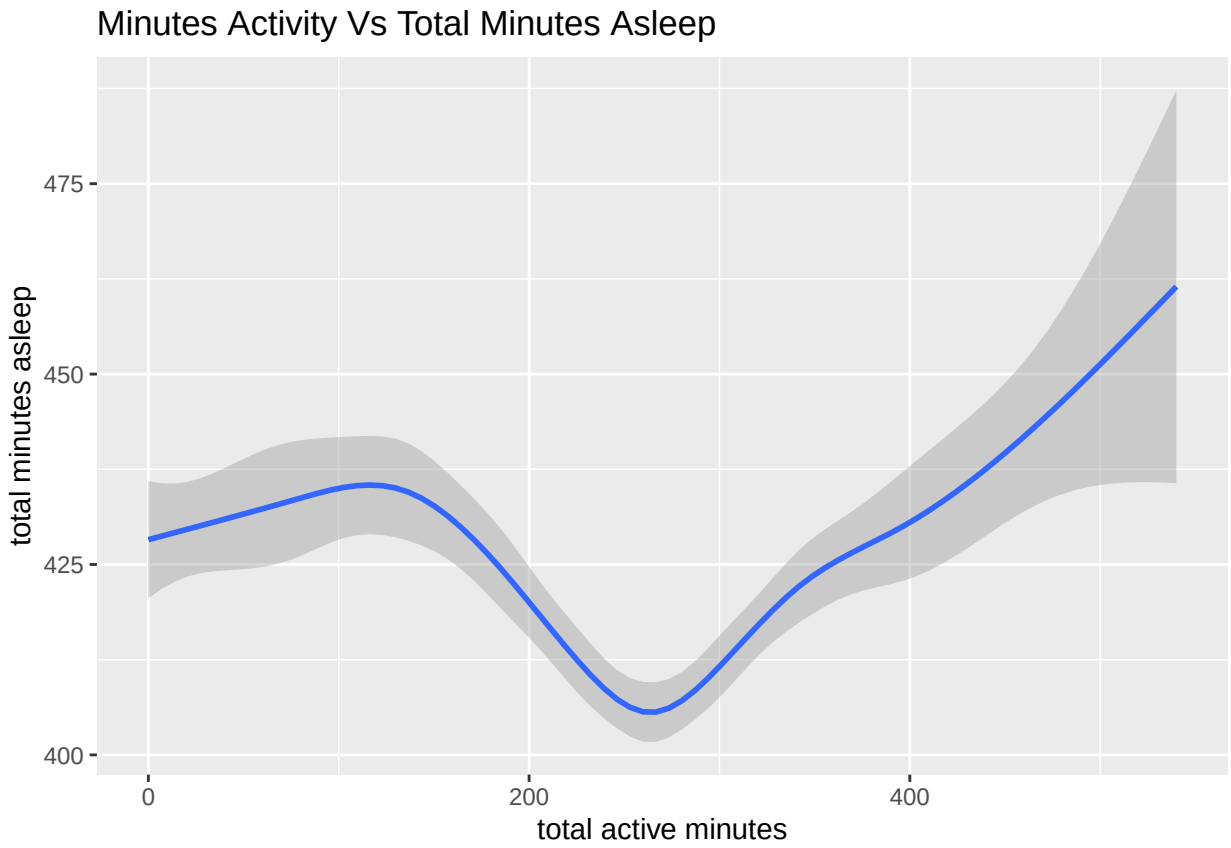
```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



I also used the previously create table to plot a line graph illustrating the relationship between minutes active and total minutes asleep.

```
#relationship between minutes active and total minutes asleep
ggplot(data = sleep_activity_merged) +
  geom_smooth(mapping = aes(x = total_active_minutes, y = totalminutesasleep)) +
  #Adding Label
  labs(title = "Minutes Activity Vs Total Minutes Asleep", x = "total active minutes", y = "total minut
```

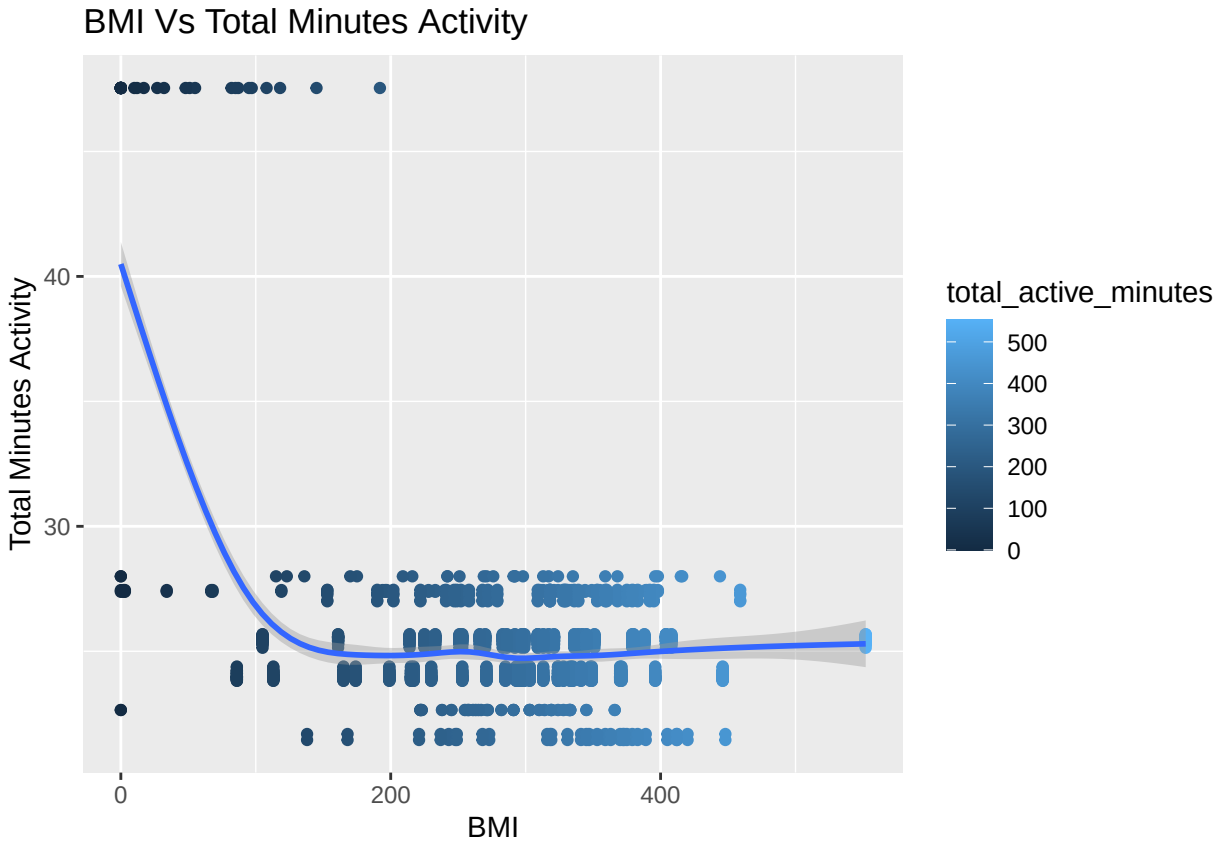
```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



Decide to compare the bmi and total activity, has I believe this would paint a clear picture of an individuals health.

```
#Scatter and line graph illustrating the relationship between total active minutes and bmi
ggplot(data = weight_activity_merged) +
  geom_point(mapping = aes(x = total_active_minutes, y = bmi, color = total_active_minutes)) +
  geom_smooth(mapping = aes(x = total_active_minutes, y = bmi)) +
  #Adding label
  labs(title = "BMI Vs Total Minutes Activity", x = "BMI", y = "Total Minutes Activity")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



The Share Phase

I chose to use Google Sheet for my visualization, as it offers great tool for create charts form table.

I exported the cleaned datasets to .csv file so they could be upload to Sheets for visualization.

```
#export the cleaned datasets to use in Tableau
write.csv(daily_activity_new_v2, file = "activity_cleaned.csv")

write.csv(sleep_day_new_v2, file = "sleep_cleaned.csv")

write.csv(weight_log_new, file = "weight_cleaned.csv")
```

The Act Phase

For the Act Phase, I created a Present in Google Slides to:

- Explaining my visualizations
- Outline the my results
- Discussing Dataset