Introduction

This is a case study for Google Data Analyst Certificate:

Bellabeat is a high-tech company that manufactures health-focused smart products. It is a successful small company, but they have the potential to become a larger player in the global smart device market. Since it was founded in 2013, Bellabeat has grown rapidly and quickly positioned itself as a tech-driven wellness company for women.You can know more about them by visiting there website.

About Client

Urška Sršen and Sando Mur founded Bellabeat, 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. The company has invested in traditional advertising media, such as radio, out-of-home billboards, print, and television, but focuses on digital marketing extensively. Bellabeat invests year-round in Google Search, maintaining active Facebook and Instagram pages, and consistently engages consumers on Twitter. Additionally, Bellabeat runs video ads on Youtube and display ads on the Google Display Network to support campaigns around key marketing dates.

Stakeholders

- Urška Sršen : Bellabeat's cofounder and Chief Creative Officer.
- Sando Mur : Mathematician and Bellabeat's cofounder; key member of the Bellabeat executive team.
- Bellabeat marketing analytics team : A team of data analysts responsible for collecting, analyzing, and reporting data that helps guide Bellabeat's marketing strategy.

Products

- Bellabeat app: The Bellabeat app provides users with health data related to their activity, sleep, stress, menstrual cycle, and mindfulness habits. This data can help users better understand their current habits and make healthy decisions. The Bellabeat app connects to their line of smart wellness products.
- Leaf: Bellabeat's classic wellness tracker can be worn as a bracelet, necklace, or clip. The Leaf tracker connects to the Bellabeat app to track activity, sleep, and stress.
- Time: This wellness watch combines the timeless look of a classic timepiece with smart technology to track user activity, sleep, and stress. The Time watch connects to the Bellabeat app to provide you with insights into your daily wellness.
- Spring: This is a water bottle that tracks daily water intake using smart technology to ensure that you are appropriately hydrated throughout the day. The Spring bottle connects to the Bellabeat app to track your hydration levels.

Business Task

Our task is to focus on a Bellabeat product and analyze smart device usage data in order to gain insight into how people are already using the smart devices. Then, using this information, we need to recommend how these trends can help Bellabeat marketing strategy.

Preparing Data

- Data Source: FitBit Fitness Tracker
- License: CC0: Public Domain
- Last Updated: 2020-12-16
- Credibility Of Data: Dataset avilable through mobius.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.

Load all the required packages

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

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(janitor)
```

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

```
library(tidyr)
```

Loading all the data sets needed

For my analysis i'll be using these three datasets:

1. dailyActivity_merged
2. sleepDay_merged
3. weightLogInfo_merged

(note that i have changed the file names to less complex ones)

```
Daily_Activity <- read.csv("daily_Activity.csv")
Sleep_Day <- read.csv("sleep_Day.csv")
Weightlog <- read.csv("weightLog_Info.csv")
```

Exploring The Datasets

1. Daily_Activity

```
summary(Daily_Activity)
```

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

```
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+0
9 ...
##  $ ActivityDate             : chr  "04-12-2016" "4/13/2016" "4/14/2016" "
4/15/2016" ...
##  $ TotalSteps               : int  13162 10735 10460 9762 12669 9705 1301
9 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 21
1 ...
##  $ SedentaryMinutes         : int  728 776 1218 726 773 539 1149 775 818
838 ...
##  $ Calories                 : int  1985 1797 1776 1745 1863 1728 1921 203
5 1786 1775 ...
```

```
n_distinct(Daily_Activity$Id)
```

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

1. Sleep_Day

```
summary(Sleep_Day)
```

```
##        Id              SleepDay         TotalSleepRecords TotalMinutesAs
leep
##  Min.   :1.504e+09   Length:413         Min.   :1.000     Min.   : 58.0
##  1st Qu.:3.977e+09   Class :character   1st Qu.:1.000     1st Qu.:361.0
##  Median :4.703e+09   Mode  :character   Median :1.000     Median :433.0
##  Mean   :5.001e+09                      Mean   :1.119     Mean   :419.5
##  3rd Qu.:6.962e+09                      3rd Qu.:1.000     3rd Qu.:490.0
##  Max.   :8.792e+09                      Max.   :3.000     Max.   :796.0
##  TotalTimeInBed
##  Min.   : 61.0
##  1st Qu.:403.0
##  Median :463.0
##  Mean   :458.6
##  3rd Qu.:526.0
##  Max.   :961.0
```

```
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  "04-12-2016 00:00" "4/13/2016 12:00:00 AM" "
4/15/2016 12:00:00 AM" "4/16/2016 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 ...
```

```
n_distinct(Sleep_Day$Id)
```

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

1. Weightlog

```
summary(Weightlog)
```

```
##        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   Mode :logical   Min.   :1.460e+12
##  1st Qu.:22.75   1st Qu.:23.96   FALSE:26        1st Qu.:1.461e+12
##  Median :23.50   Median :24.39   TRUE :41        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
```

```
str(Weightlog)
```

```
## '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  "05-02-2016 23:59" "05-03-2016 23:59" "4/13/2016
1:08:52 AM" "4/21/2016 11:59:59 PM" ...
##  $ 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: logi  TRUE TRUE FALSE TRUE TRUE TRUE ...
##  $ LogId         : num  1.46e+12 1.46e+12 1.46e+12 1.46e+12 1.46e+12 ...
```

```
n_distinct(Weightlog$Id)
```

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

Findings

- After exploring, we can see that the dataset contains personal fitness tracker from around thirty fitbit users, but we cant be sure about how the sampling was done and the data set is not up to date so there can be some limitations regaurding the same.
- Average sedentary minutes 991.2 mins that is around 17 hours, which means most of the users are not that active which we can also see in lightactiveminutes column where there are majority(518.0) of our customers.
- Average daily steps 7638 which according to researchers can lower your risk of premature death from 50% to 70%.
- The average sleep a user gets is 419.5 minutes that is around 7 hours of sleep.But again the average minimum sleep they are getting is only 1 hour, which can be unhealthy of many people.
- In Weightlog many users are updating there weight and other information such as BMI and fat manually, which can be incorrect.
- The amount of distinct users data available in daily_activity dataset is 33,in sleep_day it has decreased to 24 and in weightlog we see a major decrease, only 8 distinct id has updated there records.

Data Cleaning

- The clean_names() function makes sure that the column names are unique and consistent.This ensures that there's only characters, numbers, and underscores in the names.

1)Daily_Activity

```
clean_names(Daily_Activity)
```

2)Sleep_Day

```
clean_names(Sleep_Day)
```

3)Weightlog

```
clean_names(Weightlog)
```

- Using is.null() function to check for any null values in every column.

1. Daily_Activity

```
is.null(Daily_Activity$Id)
is.null(Daily_Activity$ActivityDate)
is.null(Daily_Activity$TotalSteps)
is.null(Daily_Activity$TotalDistance)
is.null(Daily_Activity$TrackerDistance)
is.null(Daily_Activity$LoggedActivitiesDistance)
is.null(Daily_Activity$VeryActiveDistance)
is.null(Daily_Activity$ModeratelyActiveDistance)
is.null(Daily_Activity$LightActiveDistance)
is.null(Daily_Activity$SedentaryActiveDistance)
is.null(Daily_Activity$VeryActiveMinutes)
is.null(Daily_Activity$FairlyActiveMinutes)
is.null(Daily_Activity$LightlyActiveMinutes)
is.null(Daily_Activity$SedentaryMinutes)
is.null(Daily_Activity$Calories)
## [1] FALSE
## [1] FALSE
## [1] FALSE
## [1] FALSE
## [1] FALSE
```

```
## [1] FALSE
## [1] FALSE
## [1] FALSE
## [1] FALSE
## [1] FALSE
## [1] FALSE
## [1] FALSE
## [1] FALSE
## [1] FALSE
## [1] FALSE
```

2)Sleep_Day

```
is.null(Sleep_Day$Id)
is.null(Sleep_Day$SleepDay)
is.null(Sleep_Day$TotalSleepRecords)
is.null(Sleep_Day$TotalMinutesAsleep)
is.null(Sleep_Day$TotalTimeInBed)
## [1] FALSE
## [1] FALSE
## [1] FALSE
## [1] FALSE
## [1] FALSE
```

3)Weightlog

```
is.null(Weightlog$Id)
is.null(Weightlog$Date)
is.null(Weightlog$WeightKg)
is.null(Weightlog$WeightPounds)
is.null(Weightlog$Fat)
is.null(Weightlog$IsManualReport)
is.null(Weightlog$LogId)
## [1] FALSE
## [1] FALSE
## [1] FALSE
## [1] FALSE
## [1] FALSE
## [1] FALSE
```

```
## [1] FALSE
```

- After inspecting the data, i noticed that the date column in sleep_day and weightlog included date and time.
  using separate()function to separate date and time into different columns.

1)Sleep_Day

```
Sleep_new<-Sleep_Day %>%
  separate(SleepDay,c("date","time")," ")
```

```
## Warning: Expected 2 pieces. Additional pieces discarded in 252 rows [2,
3, 4, 5,
## 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 26, 27, 30, 31, 34, 35, ...].
```

```
glimpse(Sleep_new)
```

```
## Rows: 413
## Columns: 6
## $ Id                 <dbl> 1503960366, 1503960366, 1503960366, 150396036
6, 150…
## $ date               <chr> "04-12-2016", "4/13/2016", "4/15/2016", "4/16
/2016"…
## $ time               <chr> "00:00", "12:00:00", "12:00:00", "12:00:00",
"12:00…
## $ TotalSleepRecords  <int> 1, 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, …
## $ TotalMinutesAsleep <int> 327, 384, 412, 340, 700, 304, 360, 325, 361,
430, 2…
## $ TotalTimeInBed     <int> 346, 407, 442, 367, 712, 320, 377, 364, 384,
449, 3…
```

2)Weightlog

```
weightlognew<-Weightlog %>%
  separate(Date,c("date","time")," ")
```

```
## Warning: Expected 2 pieces. Additional pieces discarded in 38 rows [3, 4
, 6, 8,
## 9, 13, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, ...].
```

```
glimpse(weightlognew)
```

```
## Rows: 67
## Columns: 9
## $ Id             <dbl> 1503960366, 1503960366, 1927972279, 2873212765, 2
873212…
## $ date           <chr> "05-02-2016", "05-03-2016", "4/13/2016", "4/21/20
16", "…
```

```
## $ time           <chr> "23:59", "23:59", "1:08:52", "11:59:59", "23:59",
"11:5…
## $ WeightKg       <dbl> 52.6, 52.6, 133.5, 56.7, 57.3, 72.4, 72.3, 69.7,
70.3, …
## $ WeightPounds   <dbl> 115.9631, 115.9631, 294.3171, 125.0021, 126.3249,
159.6…
## $ Fat            <int> 22, NA, NA, NA, NA, 25, NA, NA, NA, NA, NA, NA, N
A, 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> 1.46223e+12, 1.46232e+12, 1.46051e+12, 1.46128e+1
2, 1.4…
```

Data Analysis

- Adding up veryactive,fairlyactive and lightlyactive minutes into a new column ‘total_active_minutes’

```
Daily_Activity_new<- Daily_Activity %>%

  mutate(total_active_minutes= VeryActiveMinutes+FairlyActiveMinutes+Lightl
yActiveMinutes)

glimpse(Daily_Activity_new)
```

```
## Rows: 940
## Columns: 16
## $ Id                       <dbl> 1503960366, 1503960366, 1503960366, 150
396036…
## $ ActivityDate             <chr> "04-12-2016", "4/13/2016", "4/14/2016",
"4/15…
## $ TotalSteps               <int> 13162, 10735, 10460, 9762, 12669, 9705,
13019…
## $ TotalDistance            <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.5
9, 9.8…
## $ TrackerDistance          <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.5
9, 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, 3.2
5, 3.5…
## $ ModeratelyActiveDistance <dbl> 0.55, 0.69, 0.40, 1.26, 0.41, 0.78, 0.6
4, 1.3…
## $ LightActiveDistance      <dbl> 6.06, 4.71, 3.91, 2.83, 5.04, 2.51, 4.7
1, 5.0…
```

```
## $ SedentaryActiveDistance  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, …

## $ VeryActiveMinutes        <int> 25, 21, 30, 29, 36, 38, 42, 50, 28, 19,
66, 4…

## $ FairlyActiveMinutes      <int> 13, 19, 11, 34, 10, 20, 16, 31, 12, 8,
27, 21…

## $ LightlyActiveMinutes     <int> 328, 217, 181, 209, 221, 164, 233, 264,
205, …

## $ SedentaryMinutes         <int> 728, 776, 1218, 726, 773, 539, 1149, 77
5, 818…

## $ Calories                 <int> 1985, 1797, 1776, 1745, 1863, 1728, 192
1, 203…

## $ total_active_minutes     <int> 366, 257, 222, 272, 267, 222, 291, 345,
245, …
```

- categorizing sleep time of the users.
  There sleep time is categorized by there average sleep and then categorizing them into further categories:-
  average sleeping time less than 300 minutes(or 5 hr) as unhealthy sleepers
  average sleeping time between 300 minutes (or 5 hr) and 420 minutes(or 7hr) as average sleepers
  average sleep time more the 420 minutes(or 7 hr) as healthy sleepers

```
sleepcategories <- Sleep_new %>%
  group_by(Id) %>%
  summarise(avg_time_asleep = mean(TotalMinutesAsleep)) %>%
  mutate(type=case_when (
  avg_time_asleep < 300 ~ "unhealthy sleep",
  avg_time_asleep >=300 & avg_time_asleep <= 420 ~ "average sleep",
  avg_time_asleep > 420 ~ "healthy sleep"))
sleepcategories
```

```
## # A tibble: 24 × 3
##            Id avg_time_asleep type
##         <dbl>           <dbl> <chr>
##  1 1503960366            360. average sleep
##  2 1644430081            294  unhealthy sleep
##  3 1844505072            652  healthy sleep
##  4 1927972279            417  average sleep
##  5 2026352035            506. healthy sleep
##  6 2320127002             61  unhealthy sleep
##  7 2347167796            447. healthy sleep
##  8 3977333714            294. unhealthy sleep
```

```
##  9 4020332650            349. average sleep
## 10 4319703577            477. healthy sleep
## # … with 14 more rows
```

- categorizing daily steps.
  Daily steps are categorized into the following categorise again by using there average steps:-
  average steps less than 5000 as sedentary
  average steps between 5000 and 8000 as fairly active
  average steps between 8000 and 12000 as active
  average steps more than 12000 as highly active

```
#steps categories
stepcategories <- Daily_Activity %>%
  group_by(Id) %>%
  summarise(avg_step= mean(TotalSteps)) %>%
  mutate (active_type=case_when (
    avg_step <5000 ~ "Sedentary",
    avg_step >=5000 & avg_step< 8000 ~"Fairly Active",
    avg_step>=8000 & avg_step <12000~"Active",
    avg_step >=12000 ~ 'Highly Active'))
stepcategories
## # A tibble: 33 × 3
##            Id avg_step active_type
##         <dbl>    <dbl> <chr>
##  1 1503960366   12117. Highly Active
##  2 1624580081    5744. Fairly Active
##  3 1644430081    7283. Fairly Active
##  4 1844505072    2580. Sedentary
##  5 1927972279     916. Sedentary
##  6 2022484408   11371. Active
##  7 2026352035    5567. Fairly Active
##  8 2320127002    4717. Sedentary
##  9 2347167796    9520. Active
## 10 2873212765    7556. Fairly Active
## # … with 23 more rows
```

- Making a new data frame to check weather users are updating there report manualy or not.

(true= they are manualy updating there report
false= they are not updating it manualy, its auto updated)

```
weightfalseinfo<-c("FALSE","TRUE")

weightlog2<- c(length(which(weightlognew$IsManualReport == "FALSE")),length
(which(weightlognew$IsManualReport == "TRUE")))

weightlogreport<-data.frame(weightfalseinfo,weightlog2)

weightlogreport
```

```
##   weightfalseinfo weightlog2

## 1           FALSE         26

## 2            TRUE         41
```

Data Visualization

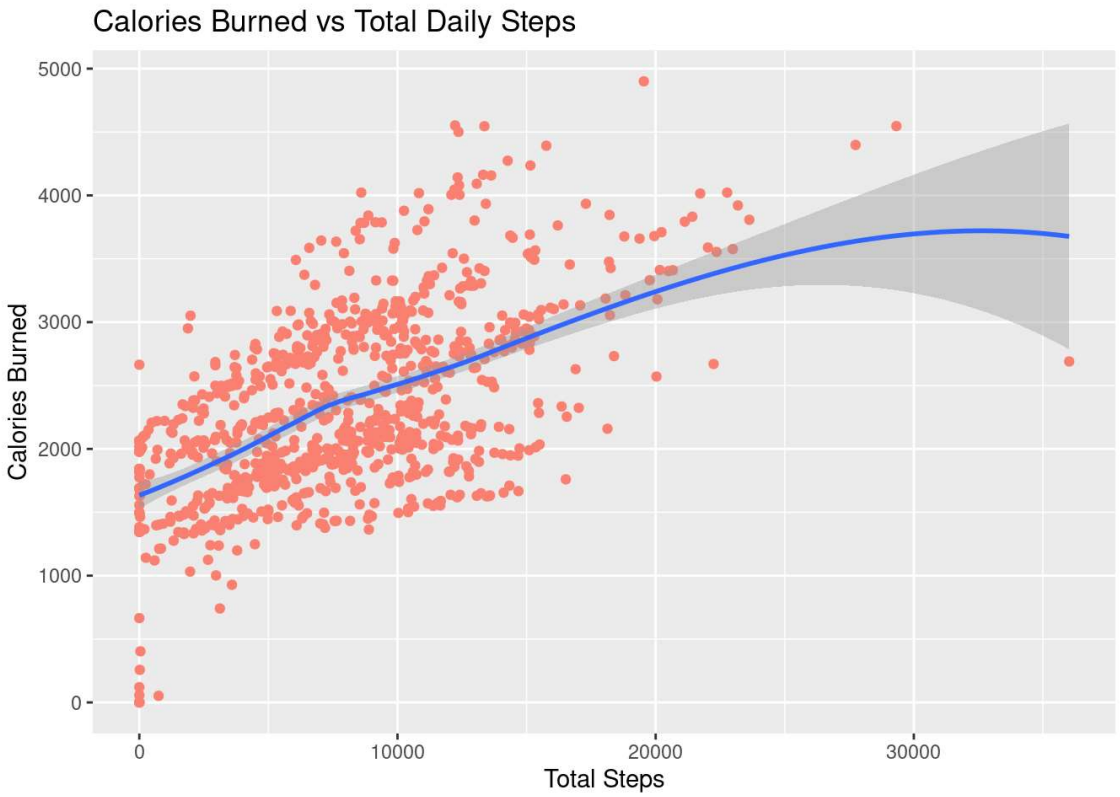
Aim

- To compare the relationship between total daily steps and total calories burned:

```
caloriesxsteps <- ggplot(data=Daily_Activity, aes(x=TotalSteps, y=Calories)
)+ geom_point(color="salmon")+ geom_smooth()+labs(title = "Calories Burned
vs Total Daily Steps",x="Total Steps", y="Calories Burned")

caloriesxsteps
```

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



Findings

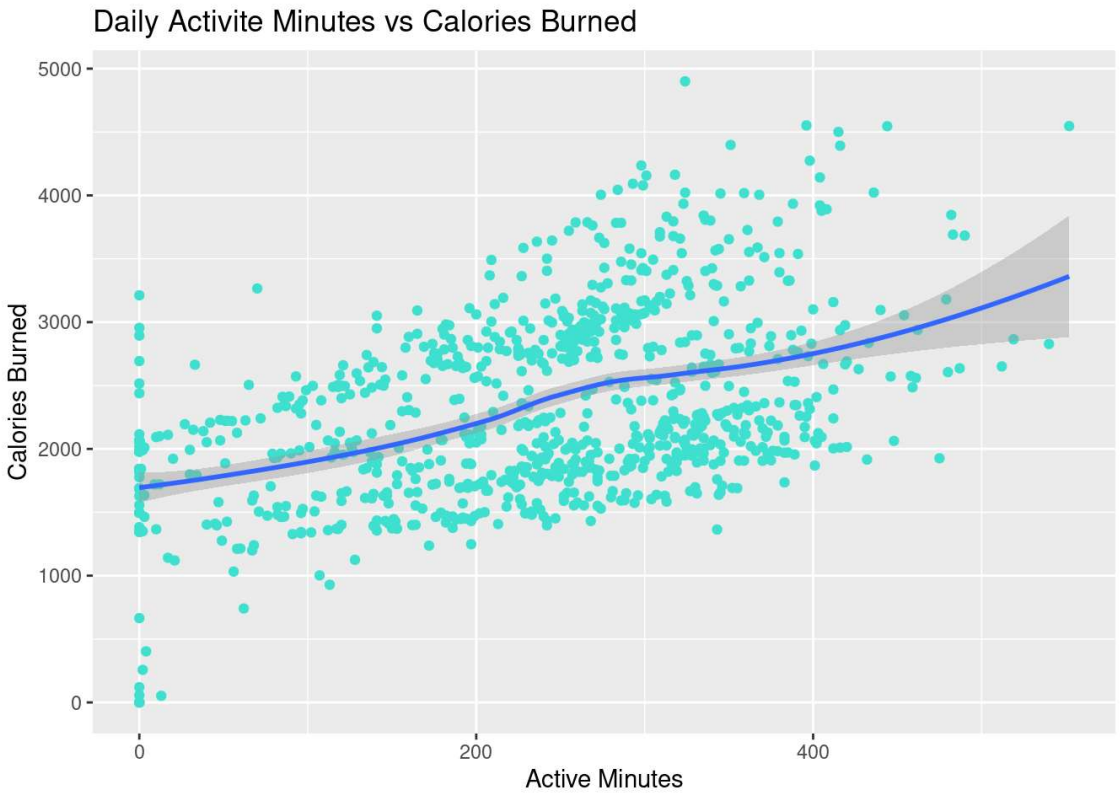
- There is a positive correlation between steps and calories burned , that means the more steps taken(the more active they are) the more calories are burned.

Aim

- We will be comparing total active minutes (which includes veryactive,fairlyactive and lightlyactive minutes) and total calories burned.

```
totalactivexcalories<-ggplot(Daily_Activity_new, aes(x = total_active_minutes, y = Calories))+
  geom_point(color="turquoise")+
  geom_smooth()+
  labs(title="Daily Activite Minutes vs Calories Burned", x= "Active Minutes", y="Calories Burned")
totalactivexcalories
```

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



Findings

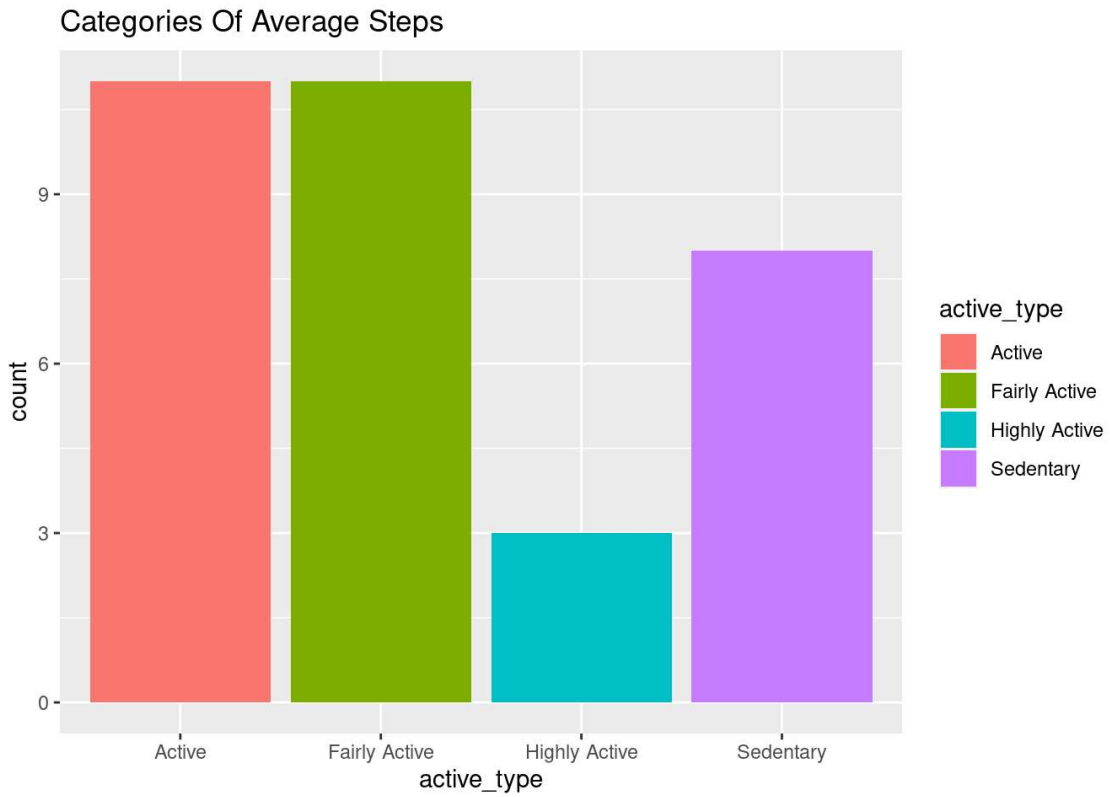
- we can again see a Positive relationship between total daily activity and calories burned.
- The more active users burned more calories compared to less active ones.

Aim

- To find out further activeness of our users based on there daily step categories.

```
steptype<- ggplot(data = stepcategories, aes(x=active_type,fill=active_type
))+ geom_bar()+labs(title = "Categories Of Average Steps")

steptype
```



Findings

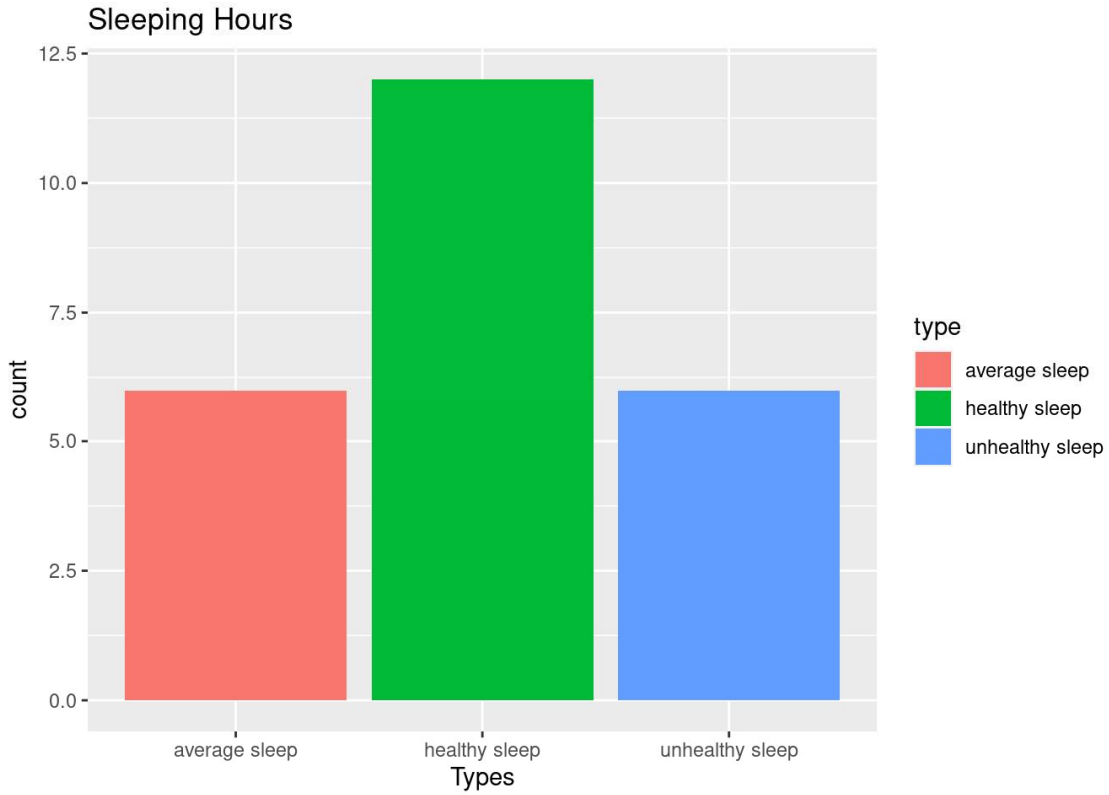
- We can see that most of the users are somewhat active which mean there average step total is more than 5000 but many of users are sedentary (less than 5k steps) which means they need some physical activity on there daily basis.

Aim

- Our aim would be to check the sleep of our users.

```
sleeptype <- ggplot(data=sleepcategories, aes(x=type , fill=type))+ geom_ba
r()+ labs(title = "Sleeping Hours",x="Types")

sleeptype
```



Findings

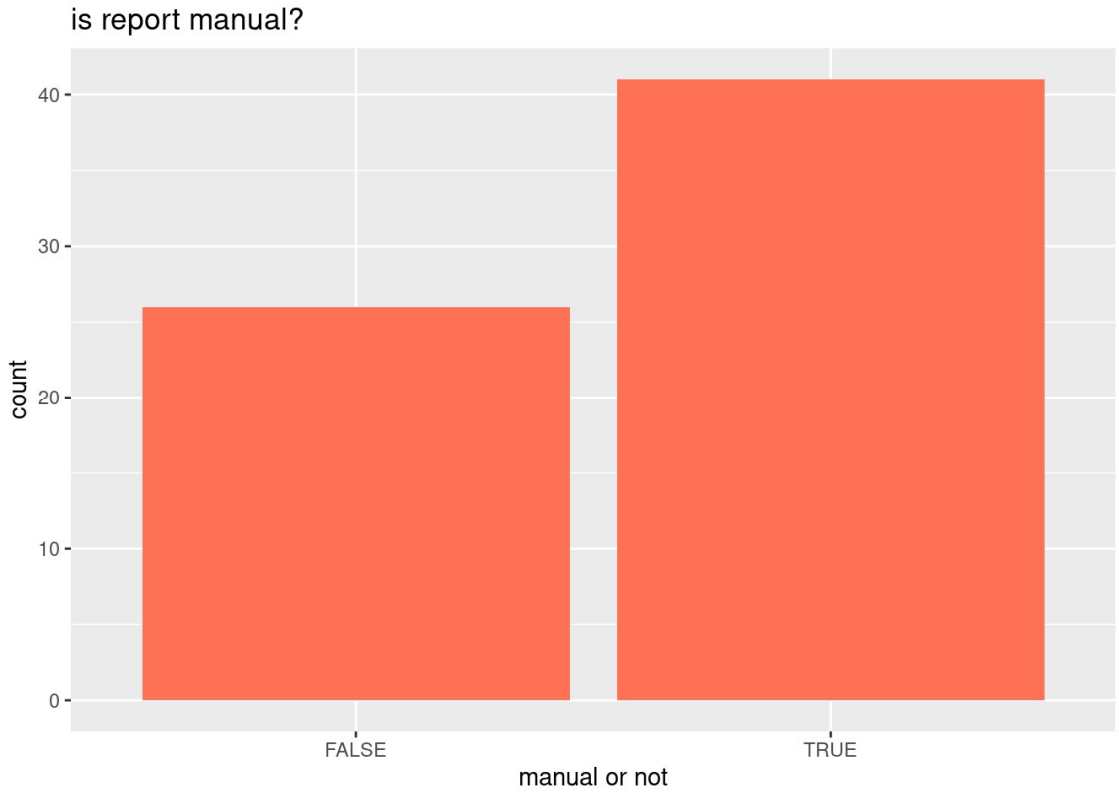
- Most of our users are getting healthy sleep which means they are sleep for 7 hours and more, but alot of them are getting below average sleep that is recommended.

Aim

- To find out how many fitbit users are manualy uploading there report

(true= they are manualy updating there report
false= they are not updating it manualy, its auto updated)

```
ggplot(data = weightlogreport,aes(weightfalseinfo, weightlog2))+ geom_bar(s
tat = 'identity',fill="coral1") + labs(title = "is report manual?" ,x="manu
al or not" ,y="count")
```




```
weightlogreport
```

```
##   weightfalseinfo weightlog2
## 1           FALSE         26
## 2            TRUE         41
```

Findings

- out of total users 33 only 8 users have updated there data and amoung there 67 enteries 41 are manualy updated.

Share

~Bellabeat empowers women to reconnect with themselves, unleash their inner strengths and be what they were meant to be~

My conclusion

- *Time to do some activity~*

most of users are active as we can se the positive correlation between there calories burned, steps and daily active minutes but there are many of them who are not active and spend more time sedentary.

My recommendation would be to program a feature on there mobile app where the users can *set there activity goal for the day*, this feature will not only send them daily messages so that they will

be motivated to reach there goal but also if the user reaches that goal of activity they will get certain amount of points.
If they are able to collect a certain amount of points the company can give them a redeemable giftcard worth the points.

The app should also add some *workout and stretching videos*.These workouts not necessarily have to be tough and long ,the app should provide them workouts of all duration and for all fitness level.

- *Bedtime starts soon~*

based on our analysis of our users are getting below average sleep that is below 7 hours of sleep.

My recommendation in this would be that using the mobile app users could set desired bed time and the app would give *notification 20 minutes ahead* of specified time so that the user would start preparing for bed. The app should also provide features such as *sleep music or audio books*, this feature would specially help those users who always have hard time getting asleep.

- *Capture the data~*

According to our analysis we found that users are not recording/updating there sleep and weight data.

This can be solved by adding *daily reminders* for users , so that they'd be notified to weigh themselves more frequently and also record there sleep time so that they can be more aware of there sleeping habits.