Google Bellabeat Wellness Capstone Project

Olisa Unegbu

02/14/2022

## Introduction:

This project is a Google Data Analytics Professional Certificate Capstone Project. The case study used is Bellabeat data analysis. Bellabeat is a high-tech manufacturer of health-focused products for women. Bellabeat empowers women with knowledge about their own health and habits through collecting data on activity, sleep, stress, and reproductive health. Urška Sršen, cofounder and Chief Creative Officer of Bellabeat, believes that analyzing smart device fitness data could help unlock new growth opportunities for the company.

## Business Task

I have been asked to analyze smart device usage data in order to gain insight into how consumers use non-Bellabeat smart devices and then select one Bellabeat product to apply these insights in my presentation in order to inform and improve Bellabeat marketing strategy.

## Key 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.

## Information about the dataset used:

**Source:** FitBit Fitness Tracker Data (CC0: Public Domain, dataset made available through Mobius): This Kaggle dataset contains personal fitness tracker from thirty fitbit users. It contains a small sample size of thirty (30) eligible FitBit users who consented to the submission of their personal tracker data, including minute-level output for physical activity, heart rate, and sleep monitoring. It includes information about daily activity, steps, and heart rate that can be used to explore users’ habits. They are stored in 18.csv files and organized in a wide format where each users’ information is arranged in a row with multiple columns containing their various attributes. The dataset is reliable, original, not comprehensive enough with no clear demographics of users such as their gender, age), not current (as it was published in 2016), and cited.

I will work with data from ‘dailyActivity\_merged’, ‘sleepDay\_merged’, and ‘weightLodInfo\_merged’ dataset since Bellabeat dominant and current products focus more on activity, sleep, stress, and reproductive health.

## setting up my environment

In doing this, I will library the necessary packages that are needed for my analysis.

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 1.0.7  
## 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(lubridate)

##   
## Attaching package: 'lubridate'

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

library(cowplot)

##   
## Attaching package: 'cowplot'

## The following object is masked from 'package:lubridate':  
##   
## stamp

## Importing and reading my data

dailyActivity <- read\_csv("dailyActivity\_merged.csv")

## Rows: 940 Columns: 15

## -- Column specification --------------------------------------------------------  
## Delimiter: ","  
## chr (1): ActivityDate  
## dbl (14): Id, TotalSteps, TotalDistance, TrackerDistance, LoggedActivitiesDi...

##   
## i Use `spec()` to retrieve the full column specification for this data.  
## i Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

sleepDay <- read\_csv("sleepDay\_merged.csv")

## Rows: 413 Columns: 5

## -- Column specification --------------------------------------------------------  
## Delimiter: ","  
## chr (1): SleepDay  
## dbl (4): Id, TotalSleepRecords, TotalMinutesAsleep, TotalTimeInBed

##   
## i Use `spec()` to retrieve the full column specification for this data.  
## i Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

weightLogInfo <- read\_csv("weightLogInfo\_merged.csv")

## Rows: 67 Columns: 8

## -- Column specification --------------------------------------------------------  
## Delimiter: ","  
## chr (1): Date  
## dbl (6): Id, WeightKg, WeightPounds, Fat, BMI, LogId  
## lgl (1): IsManualReport

##   
## i Use `spec()` to retrieve the full column specification for this data.  
## i Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

heartRate <- read\_csv("heartrate\_seconds\_merged.csv")

## Rows: 2483658 Columns: 3

## -- Column specification --------------------------------------------------------  
## Delimiter: ","  
## chr (1): Time  
## dbl (2): Id, Value

##   
## 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.

head(dailyActivity)

## # 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>

head(sleepDay)

## # 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

head(weightLogInfo)

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

head(heartRate)

## # A tibble: 6 x 3  
## Id Time Value  
## <dbl> <chr> <dbl>  
## 1 2022484408 4/12/2016 7:21:00 AM 97  
## 2 2022484408 4/12/2016 7:21:05 AM 102  
## 3 2022484408 4/12/2016 7:21:10 AM 105  
## 4 2022484408 4/12/2016 7:21:20 AM 103  
## 5 2022484408 4/12/2016 7:21:25 AM 101  
## 6 2022484408 4/12/2016 7:22:05 AM 95

# Data Exploration

I will use the glimpse() function to a preview of my data file and understanding its contents and organization.

glimpse(dailyActivity)

## Rows: 940  
## Columns: 15  
## $ Id <dbl> 1503960366, 1503960366, 1503960366, 150396036~  
## $ ActivityDate <chr> "4/12/2016", "4/13/2016", "4/14/2016", "4/15/~  
## $ TotalSteps <dbl> 13162, 10735, 10460, 9762, 12669, 9705, 13019~  
## $ TotalDistance <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.59, 9.8~  
## $ TrackerDistance <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.59, 9.8~  
## $ LoggedActivitiesDistance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~  
## $ VeryActiveDistance <dbl> 1.88, 1.57, 2.44, 2.14, 2.71, 3.19, 3.25, 3.5~  
## $ ModeratelyActiveDistance <dbl> 0.55, 0.69, 0.40, 1.26, 0.41, 0.78, 0.64, 1.3~  
## $ LightActiveDistance <dbl> 6.06, 4.71, 3.91, 2.83, 5.04, 2.51, 4.71, 5.0~  
## $ SedentaryActiveDistance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~  
## $ VeryActiveMinutes <dbl> 25, 21, 30, 29, 36, 38, 42, 50, 28, 19, 66, 4~  
## $ FairlyActiveMinutes <dbl> 13, 19, 11, 34, 10, 20, 16, 31, 12, 8, 27, 21~  
## $ LightlyActiveMinutes <dbl> 328, 217, 181, 209, 221, 164, 233, 264, 205, ~  
## $ SedentaryMinutes <dbl> 728, 776, 1218, 726, 773, 539, 1149, 775, 818~  
## $ Calories <dbl> 1985, 1797, 1776, 1745, 1863, 1728, 1921, 203~

glimpse(sleepDay)

## Rows: 413  
## Columns: 5  
## $ Id <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 150~  
## $ SleepDay <chr> "4/12/2016 12:00:00 AM", "4/13/2016 12:00:00 AM", "~  
## $ TotalSleepRecords <dbl> 1, 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~  
## $ TotalMinutesAsleep <dbl> 327, 384, 412, 340, 700, 304, 360, 325, 361, 430, 2~  
## $ TotalTimeInBed <dbl> 346, 407, 442, 367, 712, 320, 377, 364, 384, 449, 3~

glimpse(weightLogInfo)

## Rows: 67  
## Columns: 8  
## $ Id <dbl> 1503960366, 1503960366, 1927972279, 2873212765, 2873212~  
## $ Date <chr> "5/2/2016 11:59:59 PM", "5/3/2016 11:59:59 PM", "4/13/2~  
## $ WeightKg <dbl> 52.6, 52.6, 133.5, 56.7, 57.3, 72.4, 72.3, 69.7, 70.3, ~  
## $ WeightPounds <dbl> 115.9631, 115.9631, 294.3171, 125.0021, 126.3249, 159.6~  
## $ Fat <dbl> 22, NA, NA, NA, NA, 25, NA, NA, NA, NA, NA, NA, NA, NA,~  
## $ BMI <dbl> 22.65, 22.65, 47.54, 21.45, 21.69, 27.45, 27.38, 27.25,~  
## $ IsManualReport <lgl> TRUE, TRUE, FALSE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, ~  
## $ LogId <dbl> 1.462234e+12, 1.462320e+12, 1.460510e+12, 1.461283e+12,~

glimpse(heartRate)

## Rows: 2,483,658  
## Columns: 3  
## $ Id <dbl> 2022484408, 2022484408, 2022484408, 2022484408, 2022484408, 2022~  
## $ Time <chr> "4/12/2016 7:21:00 AM", "4/12/2016 7:21:05 AM", "4/12/2016 7:21:~  
## $ Value <dbl> 97, 102, 105, 103, 101, 95, 91, 93, 94, 93, 92, 89, 83, 61, 60, ~

The total number of rows and columns for each data file are;

○ **dailyActivity:** 940 rows, 15 columns

○ **sleepDay:** 413 rows, 5 columns

○ **weightLogInfo:** 67 rows, 8 columns

○ **heartRate:** 2,483,658 rows, 3 columns

The dailySleep, weightLogInfo, and heartRate file has a column containing both a date and a time on ‘SleepDay’, ‘Date’, and ‘Time’ respectively. I will create a new column and name it ‘Time’ and also rename the former as ‘Date’.

daily\_sleep <- sleepDay %>%   
 separate(SleepDay, c("Date", "Time"), " ")

## Warning: Expected 2 pieces. Additional pieces discarded in 413 rows [1, 2, 3, 4,  
## 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].

glimpse(daily\_sleep)

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

daily\_weight <- weightLogInfo %>%   
 separate(Date, c("Date", "Time"), " ")

## Warning: Expected 2 pieces. Additional pieces discarded in 67 rows [1, 2, 3, 4,  
## 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].

glimpse(daily\_weight)

## Rows: 67  
## Columns: 9  
## $ Id <dbl> 1503960366, 1503960366, 1927972279, 2873212765, 2873212~  
## $ Date <chr> "5/2/2016", "5/3/2016", "4/13/2016", "4/21/2016", "5/12~  
## $ Time <chr> "11:59:59", "11:59:59", "1:08:52", "11:59:59", "11:59:5~  
## $ 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 <dbl> 22, NA, NA, NA, NA, 25, NA, NA, NA, NA, NA, NA, NA, NA,~  
## $ BMI <dbl> 22.65, 22.65, 47.54, 21.45, 21.69, 27.45, 27.38, 27.25,~  
## $ IsManualReport <lgl> TRUE, TRUE, FALSE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, ~  
## $ LogId <dbl> 1.462234e+12, 1.462320e+12, 1.460510e+12, 1.461283e+12,~

daily\_heartRate <- heartRate %>%   
 separate(Time, c("Date", "Time"), " ")

## Warning: Expected 2 pieces. Additional pieces discarded in 2483658 rows [1, 2,  
## 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].

glimpse(daily\_heartRate)

## Rows: 2,483,658  
## Columns: 4  
## $ Id <dbl> 2022484408, 2022484408, 2022484408, 2022484408, 2022484408, 2022~  
## $ Date <chr> "4/12/2016", "4/12/2016", "4/12/2016", "4/12/2016", "4/12/2016",~  
## $ Time <chr> "7:21:00", "7:21:05", "7:21:10", "7:21:20", "7:21:25", "7:22:05"~  
## $ Value <dbl> 97, 102, 105, 103, 101, 95, 91, 93, 94, 93, 92, 89, 83, 61, 60, ~

### Data Cleaning

**Checking for duplicates**

sum(duplicated(dailyActivity))

## [1] 0

sum(duplicated(daily\_sleep))

## [1] 3

sum(duplicated(daily\_weight))

## [1] 0

sum(duplicated(daily\_heartRate))

## [1] 9334

The daily\_sleep data file has 3 duplicates while daily\_heartRate data file has 9334 duplicates.

**Removing duplicates**

sleep\_dupless <- daily\_sleep[!duplicated(daily\_sleep), ]  
heartRate\_dupless <- daily\_heartRate[!duplicated(daily\_heartRate), ]

**Confirming duplicates removal**

sum(duplicated(sleep\_dupless))

## [1] 0

sum(duplicated(heartRate\_dupless))

## [1] 0

There are no more duplicates in my data.

**Checking for NAs**

sum(is.na(dailyActivity))

## [1] 0

sum(is.na(sleep\_dupless))

## [1] 0

sum(is.na(daily\_weight$WeightKg))

## [1] 0

sum(is.na(daily\_weight$BMI))

## [1] 0

sum(is.na(heartRate\_dupless))

## [1] 0

No NA values in my needed columns.

## Distinct Count

Counting how many distinct Id are on each data file.

n\_distinct(dailyActivity$Id)

## [1] 33

n\_distinct(sleep\_dupless$Id)

## [1] 24

n\_distinct(daily\_weight$Id)

## [1] 8

n\_distinct(heartRate\_dupless$Id)

## [1] 14

There are;

○ 33 distinct Ids in dailyActivity data file,

○ 24 distinct Ids in daily\_sleep data file,

○ 8 distinct Ids in daily\_weight data file, and

○ 14 distinct Ids in daily\_heart data file.

Continuing my data exploration on some of the activities associated with healthy living.

**Distance Covered**

dailyActivity %>%   
 select(TotalSteps,TotalDistance,VeryActiveDistance, ModeratelyActiveDistance,  
 LightActiveDistance, SedentaryActiveDistance, Calories) %>%   
 summary()

## TotalSteps TotalDistance VeryActiveDistance ModeratelyActiveDistance  
## Min. : 0 Min. : 0.000 Min. : 0.000 Min. :0.0000   
## 1st Qu.: 3790 1st Qu.: 2.620 1st Qu.: 0.000 1st Qu.:0.0000   
## Median : 7406 Median : 5.245 Median : 0.210 Median :0.2400   
## Mean : 7638 Mean : 5.490 Mean : 1.503 Mean :0.5675   
## 3rd Qu.:10727 3rd Qu.: 7.713 3rd Qu.: 2.053 3rd Qu.:0.8000   
## Max. :36019 Max. :28.030 Max. :21.920 Max. :6.4800   
## LightActiveDistance SedentaryActiveDistance Calories   
## Min. : 0.000 Min. :0.000000 Min. : 0   
## 1st Qu.: 1.945 1st Qu.:0.000000 1st Qu.:1828   
## Median : 3.365 Median :0.000000 Median :2134   
## Mean : 3.341 Mean :0.001606 Mean :2304   
## 3rd Qu.: 4.782 3rd Qu.:0.000000 3rd Qu.:2793   
## Max. :10.710 Max. :0.110000 Max. :4900

The above summary shows that the average distance covered is 5.490. It also shows majority FitBit users are lightly active with a mean of 3.341 higher Very Active with a mean of 1.503 and Moderately Active with a mean of 0.5675.

**Sleep**

sleep\_dupless %>%   
 select(TotalSleepRecords, TotalMinutesAsleep, TotalTimeInBed) %>%  
 summary()

## TotalSleepRecords TotalMinutesAsleep TotalTimeInBed   
## Min. :1.00 Min. : 58.0 Min. : 61.0   
## 1st Qu.:1.00 1st Qu.:361.0 1st Qu.:403.8   
## Median :1.00 Median :432.5 Median :463.0   
## Mean :1.12 Mean :419.2 Mean :458.5   
## 3rd Qu.:1.00 3rd Qu.:490.0 3rd Qu.:526.0   
## Max. :3.00 Max. :796.0 Max. :961.0

Here, the users get an average sleep of 419.2min (~7hrs) once a day.

**Weight**

daily\_weight %>%   
 select(WeightKg, BMI) %>%   
 summary()

## WeightKg BMI   
## Min. : 52.60 Min. :21.45   
## 1st Qu.: 61.40 1st Qu.:23.96   
## Median : 62.50 Median :24.39   
## Mean : 72.04 Mean :25.19   
## 3rd Qu.: 85.05 3rd Qu.:25.56   
## Max. :133.50 Max. :47.54

The mean WeightKg in the above summary is 72.04 and a mean BMI of 25.19 which according to CDC, should be between 18.5-24.9.

**Heart Rate**

heartRate\_dupless$Value %>%   
 summary()

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 36.00 63.00 73.00 77.36 88.00 203.00

The summary above shows a 77.36 mean heart rate beat.

## Merging my data files

activity\_sleep <- merge(dailyActivity, sleep\_dupless, by = "Id")  
mergedFiles <- merge(activity\_sleep, daily\_weight, by = "Id")  
  
glimpse(mergedFiles)

## Rows: 33,702  
## Columns: 28  
## $ Id <dbl> 1503960366, 1503960366, 1503960366, 150396036~  
## $ ActivityDate <chr> "5/12/2016", "5/12/2016", "5/8/2016", "5/8/20~  
## $ TotalSteps <dbl> 0, 0, 10060, 10060, 0, 0, 0, 0, 0, 0, 0, 0, 1~  
## $ TotalDistance <dbl> 0.00, 0.00, 6.58, 6.58, 0.00, 0.00, 0.00, 0.0~  
## $ TrackerDistance <dbl> 0.00, 0.00, 6.58, 6.58, 0.00, 0.00, 0.00, 0.0~  
## $ LoggedActivitiesDistance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~  
## $ VeryActiveDistance <dbl> 0.00, 0.00, 3.53, 3.53, 0.00, 0.00, 0.00, 0.0~  
## $ ModeratelyActiveDistance <dbl> 0.00, 0.00, 0.32, 0.32, 0.00, 0.00, 0.00, 0.0~  
## $ LightActiveDistance <dbl> 0.00, 0.00, 2.73, 2.73, 0.00, 0.00, 0.00, 0.0~  
## $ SedentaryActiveDistance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~  
## $ VeryActiveMinutes <dbl> 0, 0, 44, 44, 0, 0, 0, 0, 0, 0, 0, 0, 46, 46,~  
## $ FairlyActiveMinutes <dbl> 0, 0, 8, 8, 0, 0, 0, 0, 0, 0, 0, 0, 11, 11, 1~  
## $ LightlyActiveMinutes <dbl> 0, 0, 203, 203, 0, 0, 0, 0, 0, 0, 0, 0, 206, ~  
## $ SedentaryMinutes <dbl> 1440, 1440, 574, 574, 1440, 1440, 1440, 1440,~  
## $ Calories <dbl> 0, 0, 1740, 1740, 0, 0, 0, 0, 0, 0, 0, 0, 181~  
## $ Date.x <chr> "5/6/2016", "5/6/2016", "4/15/2016", "4/15/20~  
## $ Time.x <chr> "12:00:00", "12:00:00", "12:00:00", "12:00:00~  
## $ TotalSleepRecords <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~  
## $ TotalMinutesAsleep <dbl> 334, 334, 412, 412, 331, 331, 594, 594, 247, ~  
## $ TotalTimeInBed <dbl> 367, 367, 442, 442, 349, 349, 611, 611, 264, ~  
## $ Date.y <chr> "5/3/2016", "5/2/2016", "5/3/2016", "5/2/2016~  
## $ Time.y <chr> "11:59:59", "11:59:59", "11:59:59", "11:59:59~  
## $ WeightKg <dbl> 52.6, 52.6, 52.6, 52.6, 52.6, 52.6, 52.6, 52.~  
## $ WeightPounds <dbl> 115.9631, 115.9631, 115.9631, 115.9631, 115.9~  
## $ Fat <dbl> NA, 22, NA, 22, NA, 22, NA, 22, NA, 22, NA, 2~  
## $ BMI <dbl> 22.65, 22.65, 22.65, 22.65, 22.65, 22.65, 22.~  
## $ IsManualReport <lgl> TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRUE, TRU~  
## $ LogId <dbl> 1.462320e+12, 1.462234e+12, 1.462320e+12, 1.4~

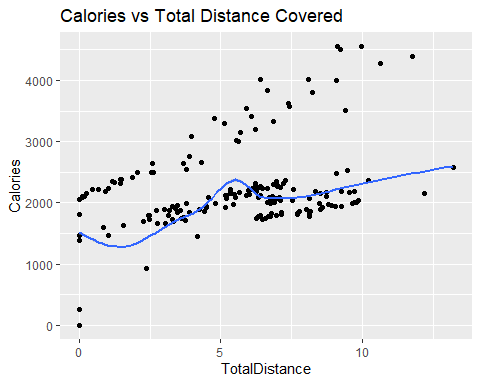
My mergedFiles data file now has a total of 33,702 rows with 28 columns.

# Visualization

**Total Distance Covered vs Calories**

ggplot(data = mergedFiles, aes(x = TotalDistance, y = Calories)) +  
 geom\_point() + geom\_smooth() +  
 labs(title = "Calories vs Total Distance Covered")

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



The chart above shows fairly positive correlation between Total distance covered and Calories burnt. i.e the more distance is covered, the more calories is burnt.

**Checking for correlation coefficient**

cor.test(mergedFiles$TotalDistance, mergedFiles$Calories,   
 method = "pearson", conf.level = 0.95)

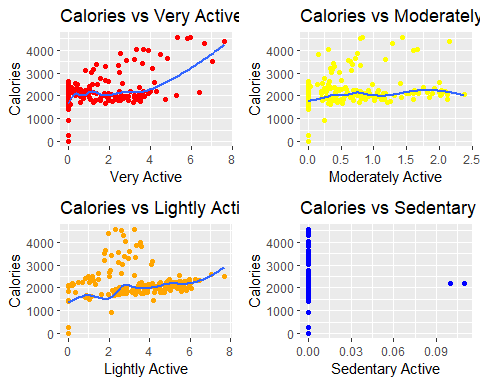
##   
## Pearson's product-moment correlation  
##   
## data: mergedFiles$TotalDistance and mergedFiles$Calories  
## t = 161.29, df = 33700, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.6539677 0.6660188  
## sample estimates:  
## cor   
## 0.6600357

The analysis above uses the Pearson’s product-moment correlation which shows a ‘r’ value of 0.66 which also indicates a positive correlation corresponding with the earlier analysis.

**Level of Activeness vs Calories**

very\_active <- ggplot(data = mergedFiles,   
 aes(x = VeryActiveDistance, y = Calories)) +   
 geom\_point(color = "red") + geom\_smooth() +   
 labs(title = "Calories vs Very Active Distance",   
 x = "Very Active", y = "Calories")  
   
moderately\_active <- ggplot(data = mergedFiles,   
 aes(x = ModeratelyActiveDistance, y = Calories)) +   
 geom\_point(color = "yellow") + geom\_smooth() +   
 labs(title = "Calories vs Moderately Active Distance",   
 x = "Moderately Active", y = "Calories")  
  
lightly\_active <- ggplot(data = mergedFiles,   
 aes(x = LightActiveDistance, y = Calories)) +   
 geom\_point(color = "orange") + geom\_smooth() +   
 labs(title = "Calories vs Lightly Active Distance",   
 x = "Lightly Active", y = "Calories")  
  
sedentary\_active <- ggplot(data = mergedFiles,   
 aes(x = SedentaryActiveDistance, y = Calories)) +   
 geom\_point(color = "blue") +   
 labs(title = "Calories vs Sedentary Active Distance",   
 x = "Sedentary Active", y = "Calories")  
  
plot\_grid(very\_active, moderately\_active, lightly\_active, sedentary\_active)

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



These plots show the intensity of activeness compared to the amount of calories burned. Those who were very active burned more calories than those who were lightly or moderately active.

**Checking for correlation coefficient**

cor.test(mergedFiles$VeryActiveDistance, mergedFiles$Calories,   
 method = "pearson", conf.level = 0.95)

##   
## Pearson's product-moment correlation  
##   
## data: mergedFiles$VeryActiveDistance and mergedFiles$Calories  
## t = 119.76, df = 33700, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.5388657 0.5538440  
## sample estimates:  
## cor   
## 0.5463985

cor.test(mergedFiles$ModeratelyActiveDistance, mergedFiles$Calories,   
 method = "pearson", conf.level = 0.95)

##   
## Pearson's product-moment correlation  
##   
## data: mergedFiles$ModeratelyActiveDistance and mergedFiles$Calories  
## t = 69.036, df = 33700, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.3426046 0.3613120  
## sample estimates:  
## cor   
## 0.3519935

cor.test(mergedFiles$LightActiveDistance, mergedFiles$Calories,   
 method = "pearson", conf.level = 0.95)

##   
## Pearson's product-moment correlation  
##   
## data: mergedFiles$LightActiveDistance and mergedFiles$Calories  
## t = 99.943, df = 33700, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.4698761 0.4863473  
## sample estimates:  
## cor   
## 0.4781537

cor.test(mergedFiles$SedentaryActiveDistance, mergedFiles$Calories,   
 method = "pearson", conf.level = 0.95)

##   
## Pearson's product-moment correlation  
##   
## data: mergedFiles$SedentaryActiveDistance and mergedFiles$Calories  
## t = 20.407, df = 33700, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.09992657 0.12101866  
## sample estimates:  
## cor   
## 0.1104851

The correlation coefficient for the activeness above, all shows a positive correlation between distance and calories. But however, there is a stronger correlation between the very active users and the amount of calories burned.

**Sleep**

According to CDC, the recommended hours of sleep for an adult is 7hrs (~420min) or more.

summary(mergedFiles$TotalMinutesAsleep)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 59.0 411.0 442.0 437.5 476.0 750.0

I will categories sleep range as follows;

○ less than or equal 390min as “Little Sleep”,

○ greater than 390min and less than or equal to 510min as “Enough Sleep”, and

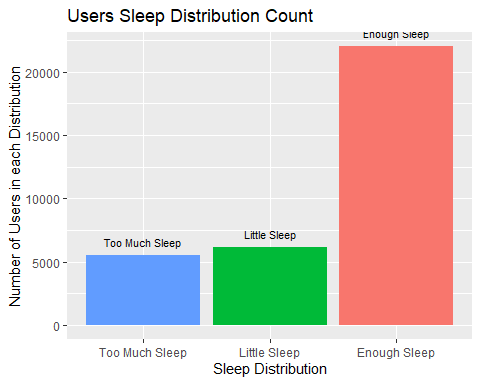
○ greater than 510min as “Too Much Sleep”.

sleep\_range <- mergedFiles %>%  
 select("Id", "TotalMinutesAsleep") %>%  
 summarise(Id, sleep\_distribution =   
 case\_when(TotalMinutesAsleep <= 390 ~ "Little Sleep",  
 TotalMinutesAsleep > 390 & TotalMinutesAsleep <= 510   
 ~"Enough Sleep",  
 TotalMinutesAsleep > 510 ~"Too Much Sleep")) %>%  
 group\_by(Id)  
  
sleep\_range <- sleep\_range %>%  
 group\_by(sleep\_distribution) %>%  
 summarise(count=n()) %>%  
 arrange(sleep\_distribution)  
  
head(sleep\_range)

## # A tibble: 3 x 2  
## sleep\_distribution count  
## <chr> <int>  
## 1 Enough Sleep 22026  
## 2 Little Sleep 6163  
## 3 Too Much Sleep 5513

**Plot**

ggplot(data = sleep\_range, aes(x = reorder(sleep\_distribution, count),   
 y = count, fill = sleep\_distribution)) +  
 geom\_bar(stat = "identity") +  
 geom\_text(aes(label = reorder(sleep\_distribution, count)),   
 vjust = -1, color = "Black", size = 3) +  
 labs(title = "Users Sleep Distribution Count", x = "Sleep Distribution",   
 y = "Number of Users in each Distribution") +  
 theme(legend.position = "none")



From the above visual, most users get enough sleep.

**Weight**

summary(mergedFiles$BMI)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 22.65 23.89 24.00 24.40 24.17 47.54

CDC categories BMI for adults as follows;

BMI <- c("Below 18.5", "18.5 - 24.9", "25.0 -29.9", "30.0 and Above")  
Weight\_Status <- c("Underweight", "Healthy Weight", "Overweight", "Obesity")  
  
BMIrange <- data.frame(BMI, Weight\_Status, stringsAsFactors = FALSE)  
BMIrange

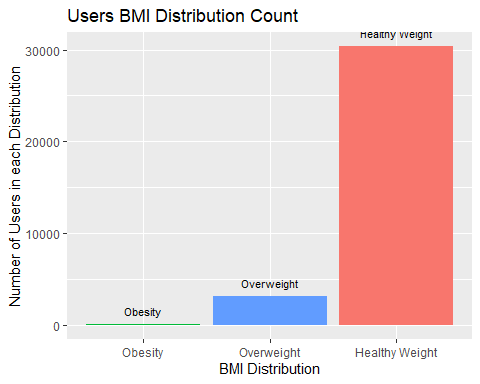
## BMI Weight\_Status  
## 1 Below 18.5 Underweight  
## 2 18.5 - 24.9 Healthy Weight  
## 3 25.0 -29.9 Overweight  
## 4 30.0 and Above Obesity

BMIrange <- mergedFiles %>%  
 select("Id", "BMI") %>%  
 summarise(Id, BMI\_distribution =   
 case\_when(BMI < 18.5 ~ "Underweight",  
 BMI >= 18.5 & BMI <= 24.9 ~ "Healthy Weight",   
 BMI >= 25.0 & BMI <= 29.9 ~ "Overweight",   
 BMI >= 30.0 ~ "Obesity")) %>%  
 group\_by(Id)  
  
BMIrange <- BMIrange %>%  
 group\_by(BMI\_distribution) %>%  
 summarise(count=n()) %>%  
 arrange(BMI\_distribution)  
  
head(BMIrange)

## # A tibble: 3 x 2  
## BMI\_distribution count  
## <chr> <int>  
## 1 Healthy Weight 30380  
## 2 Obesity 155  
## 3 Overweight 3167

**Plot**

ggplot(data = BMIrange, aes(x = reorder(BMI\_distribution, count),   
 y = count, fill = BMI\_distribution)) +  
 geom\_bar(stat = "identity") +  
 geom\_text(aes(label = reorder(BMI\_distribution, count)),   
 vjust = -1, color = "Black", size = 3) +  
 labs(title = "Users BMI Distribution Count", x = "BMI Distribution",   
 y = "Number of Users in each Distribution") +  
 theme(legend.position = "none")

 The visual above shows that most users have a healthy weight.

**Heart Rate**

summary(heartRate\_dupless$Value)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 36.00 63.00 73.00 77.36 88.00 203.00

According to MayoClinic, normal resting heart rate for adults ranges from 60 to 100 beats per minute.

I will therefore categories the heart rate as follows;

○ less that 60 as “Low Heart Rate”,

○ greater than or equal to 60 and less than or equal to 100 as “Normal Heart Rate”, and

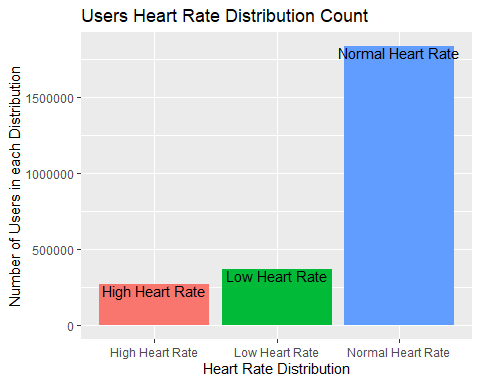
○ greater than 100 as “High Heart Rate”.

heartRate\_range <- heartRate\_dupless %>%  
 select("Id", "Value") %>%  
 summarise(Id, heartRate\_distribution =   
 case\_when(Value < 60 ~ "Low Heart Rate",  
 Value >= 60 & Value <= 100 ~ "Normal Heart Rate",   
 Value >= 100 ~ "High Heart Rate")) %>%  
 group\_by(Id)  
  
heartRate\_range <- heartRate\_range %>%  
 group\_by(heartRate\_distribution) %>%  
 summarise(count=n()) %>%  
 arrange(heartRate\_distribution)  
  
head(heartRate\_range)

## # A tibble: 3 x 2  
## heartRate\_distribution count  
## <chr> <int>  
## 1 High Heart Rate 270751  
## 2 Low Heart Rate 371016  
## 3 Normal Heart Rate 1832557

**Plot**

ggplot(data = heartRate\_range, aes(x = reorder(heartRate\_distribution, count),   
 y = count, fill = heartRate\_distribution)) +  
 geom\_bar(stat = "identity") +  
 geom\_text(aes(label = reorder(heartRate\_distribution, count)),   
 vjust = 1, color = "Black", size = 4) +  
 labs(title = "Users Heart Rate Distribution Count", x = "Heart Rate Distribution",   
 y = "Number of Users in each Distribution") +  
 theme(legend.position = "none")



The above chart shows that a greater number of users have normal resting heart rate which is a very good result.

# Recommendations

1. Based on my findings on distance covered and amount of calories burned, users appear to burn more calories when they are very active. Therefore, BellaBeat should encourage users to be very active through a reminder like a vibration on their ‘app’ when they are not very active.
2. Bellabeat should encourage better sleeping habits by notifying their users of the best time to go to bed and wake up in order to get enough sleep through incorporating reminders on their app. The app could also automatically turn on do not disturb mode and turn on night mode on the individual’s phone to signal to the user that they should go to sleep.
3. Bellabeat should send out reminders to their users to log their weight in for the most accurate data and also install a smart device, such as a digital weight that automatically logs in the user’s weight every time the user weights themselves whenever they wake up from sleep which will in turn encourage them to be very active for their daily exercise.
4. The data shows most users have a normal resting heart rate per minute. So Bellabeat are encouraged to install an app that checks their users heart rate beat on daily basis.

**Thank You**