Capstone - Bellabeat Case Study

Damolasimon

18/09/2023

Table of Contents

[1. Company Summary 1](#_Toc147330919)

[2. Ask Phase 1](#_Toc147330920)

[3. Preparation Phase 2](#_Toc147330921)

[4. Processing Phase 8](#_Toc147330922)

[5. Analysis and Visualisation 16](#_Toc147330923)

# Company Summary

The technology company with the name Bellabeat is focussed on health products for women and interested in identifying opportunities for growth based on insights from analysis of smart device fitness data.

Although Bellabeat offers 5 different products, the focus of this analysis will be the Bellabeat app which measures health related data such as sleep, stress, activity etc.

# Ask Phase

Here, we identify the problems to solve whilst keeping stakeholders in mind.

2.1 ## Business Task

My focus as a member of Bellabeat analytic team will be on the analysis of the data from the Bellabeat app. From this analysis, I will identify highlight insights from the analysis and make high-level recommendations to marketing and management based on findings.

2.2 ## Key Deliverables

1. A clear summary of the business task.
2. A description of all data sources used.
3. Documentation of any cleaning or manipulation of data.
4. A summary of your analysis.
5. Supporting visualizations and key findings.
6. Your top high-level content recommendations based on your analysis.

# Preparation Phase

The dataset used is the FitBit Fitness Tracker Data available [*here*](https://www.kaggle.com/datasets/arashnic/fitbit). Please note that this data is in the Public Domain and available through Mobius.

Things to note about dataset used:

* Dataset was generated in 2016 and more than 6 years old from the time of this analysis. This means that findings from this analysis may need to be complemented/compared with more recent data.
* Although Bellabeat is a tech company focussed only on women, the datasets used in this analysis gives no information about the gender of study participants.
* Although a total of 18 CSV files were included in the dataset at the time of this analysis, most of them are either replicates or redundant. For this reason, we started by importing 5 datasets but ended up using both the dailyAcvity\_merged.csv and sleepDay\_merged.csv for our analysis.
* The highest sample size of 33 distinct correspondents in dataset confers significant limitation and may not be an accurate representation of user’s population. The other datasets are either too small or too inconsistent and have been excluded from being used in our analysis.

3.1 ## Loading Packages and libraries

The different packages and libraries needed for this analysis are shown below:

#install.packages("tidyverse") #transformation of data   
#install.packages("lubridate") # useful for editing date and time formats  
#install.packages("skimr") # helps you summarize & skim through data   
#install.packages("plotly")  
#install.packages("janitor") #useful for data cleaning   
#install.packages("ggeasy") #useful for customization of plots   
#install.packages("ggpubr") #provides functions for plots customization  
  
library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.2 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.4.2 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.2 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.1   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(lubridate)  
library(skimr)  
library(plotly)

##   
## Attaching package: 'plotly'  
##   
## The following object is masked from 'package:ggplot2':  
##   
## last\_plot  
##   
## The following object is masked from 'package:stats':  
##   
## filter  
##   
## The following object is masked from 'package:graphics':  
##   
## layout

library(janitor)

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

library(ggeasy)  
library(ggpubr)  
library(patchwork)

3.2 ## Data Import

#Upload the data files   
daily\_activity <- read\_csv("/Users/damolaayoyerokun/Documents/RProjects/Rdata/Fitabase\_data/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.

daily\_steps <- read\_csv("/Users/damolaayoyerokun/Documents/RProjects/Rdata/Fitabase\_data/dailySteps\_merged.csv")

## Rows: 940 Columns: 3  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (1): ActivityDay  
## dbl (2): Id, StepTotal  
##   
## 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.

daily\_calories <- read\_csv("/Users/damolaayoyerokun/Documents/RProjects/Rdata/Fitabase\_data/dailyCalories\_merged.csv")

## Rows: 940 Columns: 3  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (1): ActivityDay  
## dbl (2): Id, Calories  
##   
## 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.

sleep\_records <- read\_csv("/Users/damolaayoyerokun/Documents/RProjects/Rdata/Fitabase\_data/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.

weight\_records <- read\_csv("/Users/damolaayoyerokun/Documents/RProjects/Rdata/Fitabase\_data/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.

3.3 ## Preview Datasets

glimpse(daily\_activity)

## 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(daily\_steps)

## Rows: 940  
## Columns: 3  
## $ Id <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 1503960366…  
## $ ActivityDay <chr> "4/12/2016", "4/13/2016", "4/14/2016", "4/15/2016", "4/16/…  
## $ StepTotal <dbl> 13162, 10735, 10460, 9762, 12669, 9705, 13019, 15506, 1054…

glimpse(daily\_calories)

## Rows: 940  
## Columns: 3  
## $ Id <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 1503960366…  
## $ ActivityDay <chr> "4/12/2016", "4/13/2016", "4/14/2016", "4/15/2016", "4/16/…  
## $ Calories <dbl> 1985, 1797, 1776, 1745, 1863, 1728, 1921, 2035, 1786, 1775…

glimpse(sleep\_records)

## 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(weight\_records)

## 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,…

head(daily\_activity)

## # A tibble: 6 × 15  
## Id ActivityDate TotalSteps TotalDistance TrackerDistance  
## <dbl> <chr> <dbl> <dbl> <dbl>  
## 1 1503960366 4/12/2016 13162 8.5 8.5   
## 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  
## # ℹ 10 more variables: LoggedActivitiesDistance <dbl>,  
## # VeryActiveDistance <dbl>, ModeratelyActiveDistance <dbl>,  
## # LightActiveDistance <dbl>, SedentaryActiveDistance <dbl>,  
## # VeryActiveMinutes <dbl>, FairlyActiveMinutes <dbl>,  
## # LightlyActiveMinutes <dbl>, SedentaryMinutes <dbl>, Calories <dbl>

head(daily\_steps)

## # A tibble: 6 × 3  
## Id ActivityDay StepTotal  
## <dbl> <chr> <dbl>  
## 1 1503960366 4/12/2016 13162  
## 2 1503960366 4/13/2016 10735  
## 3 1503960366 4/14/2016 10460  
## 4 1503960366 4/15/2016 9762  
## 5 1503960366 4/16/2016 12669  
## 6 1503960366 4/17/2016 9705

head(daily\_calories)

## # A tibble: 6 × 3  
## Id ActivityDay Calories  
## <dbl> <chr> <dbl>  
## 1 1503960366 4/12/2016 1985  
## 2 1503960366 4/13/2016 1797  
## 3 1503960366 4/14/2016 1776  
## 4 1503960366 4/15/2016 1745  
## 5 1503960366 4/16/2016 1863  
## 6 1503960366 4/17/2016 1728

head(sleep\_records)

## # A tibble: 6 × 5  
## Id SleepDay TotalSleepRecords TotalMinutesAsleep TotalTimeInBed  
## <dbl> <chr> <dbl> <dbl> <dbl>  
## 1 1503960366 4/12/2016 12:0… 1 327 346  
## 2 1503960366 4/13/2016 12:0… 2 384 407  
## 3 1503960366 4/15/2016 12:0… 1 412 442  
## 4 1503960366 4/16/2016 12:0… 2 340 367  
## 5 1503960366 4/17/2016 12:0… 1 700 712  
## 6 1503960366 4/19/2016 12:0… 1 304 320

head(weight\_records)

## # A tibble: 6 × 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

# Processing Phase

From the earlier preview of our datasets, we observed the following:

* The column names need to be reworked for consistency.
* Date is in character format and need to be changed to appropriate type.

In this phase, we’ll be cleaning, formatting and merging our datasets to make sure they are ready for analysis.

#Clean column names   
daily\_activity <- clean\_names(daily\_activity)  
daily\_steps <- clean\_names(daily\_steps)  
daily\_calories <- clean\_names(daily\_calories)  
sleep\_records <- clean\_names(sleep\_records)  
weight\_records <- clean\_names(weight\_records)  
  
#Convert data type of date from char to date format  
  
daily\_activity\_clean <- daily\_activity %>%  
 mutate(activity\_date = mdy(activity\_date))  
  
daily\_steps\_clean <- daily\_steps %>%   
 mutate(activity\_day = mdy(activity\_day))  
  
daily\_calories\_clean <- daily\_calories %>%   
 mutate(activity\_day = mdy(activity\_day))  
  
sleep\_records\_clean <- sleep\_records %>%  
 mutate(sleep\_day = mdy\_hms(sleep\_day))  
  
weight\_records\_clean <- weight\_records %>%  
 mutate(date = mdy\_hms(date))  
  
#Verify the converted type  
str(daily\_activity\_clean$activity\_date)

## Date[1:940], format: "2016-04-12" "2016-04-13" "2016-04-14" "2016-04-15" "2016-04-16" ...

str(daily\_steps\_clean$activity\_day)

## Date[1:940], format: "2016-04-12" "2016-04-13" "2016-04-14" "2016-04-15" "2016-04-16" ...

str(daily\_calories\_clean$activity\_day)

## Date[1:940], format: "2016-04-12" "2016-04-13" "2016-04-14" "2016-04-15" "2016-04-16" ...

str(sleep\_records\_clean$sleep\_day)

## POSIXct[1:413], format: "2016-04-12" "2016-04-13" "2016-04-15" "2016-04-16" "2016-04-17" ...

str(weight\_records\_clean$date)

## POSIXct[1:67], format: "2016-05-02 23:59:59" "2016-05-03 23:59:59" "2016-04-13 01:08:52" ...

#Determine sample size for each dataset  
n\_distinct(daily\_activity\_clean$id)

## [1] 33

n\_distinct(daily\_calories\_clean$id)

## [1] 33

n\_distinct(daily\_steps\_clean$id)

## [1] 33

n\_distinct(sleep\_records\_clean$id)

## [1] 24

n\_distinct(weight\_records\_clean$id)

## [1] 8

From the above cleaning, we observed that the weight\_records dataset returned 8 records which is too small a sample size for this analysis. Therefore, we’ll not be including this dataset in further analysis.

4.1 ## Compare Summaries and Remove Redundancy

Here, we’ll be removing redundancy by comparing summaries of our datasets and eliminating non-useful data.

#To select key variables from each dataset and do a summmary calculation.   
daily\_activity\_clean %>%  
 select(activity\_date, total\_steps, total\_distance, sedentary\_minutes, calories) %>%  
 summary()

## activity\_date total\_steps total\_distance sedentary\_minutes  
## Min. :2016-04-12 Min. : 0 Min. : 0.000 Min. : 0.0   
## 1st Qu.:2016-04-19 1st Qu.: 3790 1st Qu.: 2.620 1st Qu.: 729.8   
## Median :2016-04-26 Median : 7406 Median : 5.245 Median :1057.5   
## Mean :2016-04-26 Mean : 7638 Mean : 5.490 Mean : 991.2   
## 3rd Qu.:2016-05-04 3rd Qu.:10727 3rd Qu.: 7.713 3rd Qu.:1229.5   
## Max. :2016-05-12 Max. :36019 Max. :28.030 Max. :1440.0   
## calories   
## Min. : 0   
## 1st Qu.:1828   
## Median :2134   
## Mean :2304   
## 3rd Qu.:2793   
## Max. :4900

daily\_steps\_clean %>%   
 select(step\_total) %>%  
 summary()

## step\_total   
## Min. : 0   
## 1st Qu.: 3790   
## Median : 7406   
## Mean : 7638   
## 3rd Qu.:10727   
## Max. :36019

daily\_calories\_clean %>%  
 select(calories) %>%  
 summary()

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

sleep\_records\_clean %>%  
 select(total\_sleep\_records, total\_minutes\_asleep, total\_time\_in\_bed) %>%  
 summary()

## total\_sleep\_records total\_minutes\_asleep total\_time\_in\_bed  
## Min. :1.000 Min. : 58.0 Min. : 61.0   
## 1st Qu.:1.000 1st Qu.:361.0 1st Qu.:403.0   
## Median :1.000 Median :433.0 Median :463.0   
## Mean :1.119 Mean :419.5 Mean :458.6   
## 3rd Qu.:1.000 3rd Qu.:490.0 3rd Qu.:526.0   
## Max. :3.000 Max. :796.0 Max. :961.0

Taking a look at the summary of our datasets, we observed that both daily\_steps and daily\_calories datasets are not giving us additional insights and we’ll not be including them in further analysis.

4.2 ## Merging Datasets and Removing Duplicates

At this stage, we’ll be looking at combining both our sleep\_records dataset and daily\_activity\_clean dataset. We’ll then remove duplicates and check our merged data to make sure it is ready for further analysis.

#rename date column and change type to date  
  
sleep\_records\_final <- sleep\_records %>%  
 rename("activity\_date" = "sleep\_day") %>%  
 mutate(activity\_date = mdy\_hms(activity\_date))  
   
#Verify format  
str(sleep\_records\_final$activity\_date)

## POSIXct[1:413], format: "2016-04-12" "2016-04-13" "2016-04-15" "2016-04-16" "2016-04-17" ...

#Select some columns from our daily\_activity\_clean   
  
daily\_activity\_clean <- daily\_activity\_clean %>%  
 select(id, activity\_date, total\_steps, total\_distance, sedentary\_minutes, very\_active\_minutes, calories)  
  
#Verify format  
str(daily\_activity\_clean)

## tibble [940 × 7] (S3: tbl\_df/tbl/data.frame)  
## $ id : num [1:940] 1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...  
## $ activity\_date : Date[1:940], format: "2016-04-12" "2016-04-13" ...  
## $ total\_steps : num [1:940] 13162 10735 10460 9762 12669 ...  
## $ total\_distance : num [1:940] 8.5 6.97 6.74 6.28 8.16 ...  
## $ sedentary\_minutes : num [1:940] 728 776 1218 726 773 ...  
## $ very\_active\_minutes: num [1:940] 25 21 30 29 36 38 42 50 28 19 ...  
## $ calories : num [1:940] 1985 1797 1776 1745 1863 ...

#merge daily\_activity\_clean and sleep\_records\_final dataset  
merged\_data <- merge(daily\_activity\_clean, sleep\_records\_final, by = c("id", "activity\_date"), all.x = TRUE)  
  
#verify merged data  
head(merged\_data)

## id activity\_date total\_steps total\_distance sedentary\_minutes  
## 1 1503960366 2016-04-12 13162 8.50 728  
## 2 1503960366 2016-04-13 10735 6.97 776  
## 3 1503960366 2016-04-14 10460 6.74 1218  
## 4 1503960366 2016-04-15 9762 6.28 726  
## 5 1503960366 2016-04-16 12669 8.16 773  
## 6 1503960366 2016-04-17 9705 6.48 539  
## very\_active\_minutes calories total\_sleep\_records total\_minutes\_asleep  
## 1 25 1985 1 327  
## 2 21 1797 2 384  
## 3 30 1776 NA NA  
## 4 29 1745 1 412  
## 5 36 1863 2 340  
## 6 38 1728 1 700  
## total\_time\_in\_bed  
## 1 346  
## 2 407  
## 3 NA  
## 4 442  
## 5 367  
## 6 712

glimpse(merged\_data)

## Rows: 943  
## Columns: 10  
## $ id <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 1…  
## $ activity\_date <date> 2016-04-12, 2016-04-13, 2016-04-14, 2016-04-15, …  
## $ total\_steps <dbl> 13162, 10735, 10460, 9762, 12669, 9705, 13019, 15…  
## $ total\_distance <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.59, 9.88, 6…  
## $ sedentary\_minutes <dbl> 728, 776, 1218, 726, 773, 539, 1149, 775, 818, 83…  
## $ very\_active\_minutes <dbl> 25, 21, 30, 29, 36, 38, 42, 50, 28, 19, 66, 41, 3…  
## $ calories <dbl> 1985, 1797, 1776, 1745, 1863, 1728, 1921, 2035, 1…  
## $ total\_sleep\_records <dbl> 1, 2, NA, 1, 2, 1, NA, 1, 1, 1, NA, 1, 1, 1, 1, N…  
## $ total\_minutes\_asleep <dbl> 327, 384, NA, 412, 340, 700, NA, 304, 360, 325, N…  
## $ total\_time\_in\_bed <dbl> 346, 407, NA, 442, 367, 712, NA, 320, 377, 364, N…

#Check for number of full duplicates (duplicates in every column) in our merged data   
sum(duplicated(merged\_data))

## [1] 3

#To see the duplicated rows   
filter(merged\_data, duplicated(merged\_data))

## id activity\_date total\_steps total\_distance sedentary\_minutes  
## 1 4388161847 2016-05-05 9603 7.38 896  
## 2 4702921684 2016-05-07 14370 11.65 577  
## 3 8378563200 2016-04-25 12405 9.84 692  
## very\_active\_minutes calories total\_sleep\_records total\_minutes\_asleep  
## 1 12 2899 1 471  
## 2 5 3683 1 520  
## 3 117 4005 1 388  
## total\_time\_in\_bed  
## 1 495  
## 2 543  
## 3 402

#Drop full duplicates from merged dataset and verify   
merged\_data\_unique <- distinct(merged\_data)  
  
head(merged\_data\_unique)

## id activity\_date total\_steps total\_distance sedentary\_minutes  
## 1 1503960366 2016-04-12 13162 8.50 728  
## 2 1503960366 2016-04-13 10735 6.97 776  
## 3 1503960366 2016-04-14 10460 6.74 1218  
## 4 1503960366 2016-04-15 9762 6.28 726  
## 5 1503960366 2016-04-16 12669 8.16 773  
## 6 1503960366 2016-04-17 9705 6.48 539  
## very\_active\_minutes calories total\_sleep\_records total\_minutes\_asleep  
## 1 25 1985 1 327  
## 2 21 1797 2 384  
## 3 30 1776 NA NA  
## 4 29 1745 1 412  
## 5 36 1863 2 340  
## 6 38 1728 1 700  
## total\_time\_in\_bed  
## 1 346  
## 2 407  
## 3 NA  
## 4 442  
## 5 367  
## 6 712

#create weekday column from activity\_date  
merged\_data\_unique$weekday <- wday(merged\_data\_unique$activity\_date, label=TRUE, abbr=FALSE)  
  
head(merged\_data\_unique)

## id activity\_date total\_steps total\_distance sedentary\_minutes  
## 1 1503960366 2016-04-12 13162 8.50 728  
## 2 1503960366 2016-04-13 10735 6.97 776  
## 3 1503960366 2016-04-14 10460 6.74 1218  
## 4 1503960366 2016-04-15 9762 6.28 726  
## 5 1503960366 2016-04-16 12669 8.16 773  
## 6 1503960366 2016-04-17 9705 6.48 539  
## very\_active\_minutes calories total\_sleep\_records total\_minutes\_asleep  
## 1 25 1985 1 327  
## 2 21 1797 2 384  
## 3 30 1776 NA NA  
## 4 29 1745 1 412  
## 5 36 1863 2 340  
## 6 38 1728 1 700  
## total\_time\_in\_bed weekday  
## 1 346 Tuesday  
## 2 407 Wednesday  
## 3 NA Thursday  
## 4 442 Friday  
## 5 367 Saturday  
## 6 712 Sunday

#checking to make sure all the duplicates were dropped.   
sum(duplicated(merged\_data\_unique))

## [1] 0

#using skimr to get a data summary of our merged data to ensure cleanliness  
skim(merged\_data\_unique)

Data summary

|  |  |
| --- | --- |
| Name | merged\_data\_unique |
| Number of rows | 940 |
| Number of columns | 11 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| Date | 1 |
| factor | 1 |
| numeric | 9 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Group variables | None |

**Variable type: Date**

| skim\_variable | n\_missing | complete\_rate | min | max | median | n\_unique |
| --- | --- | --- | --- | --- | --- | --- |
| activity\_date | 0 | 1 | 2016-04-12 | 2016-05-12 | 2016-04-26 | 31 |

**Variable type: factor**

| skim\_variable | n\_missing | complete\_rate | ordered | n\_unique | top\_counts |
| --- | --- | --- | --- | --- | --- |
| weekday | 0 | 1 | TRUE | 7 | Tue: 152, Wed: 150, Thu: 147, Fri: 126 |

**Variable type: numeric**

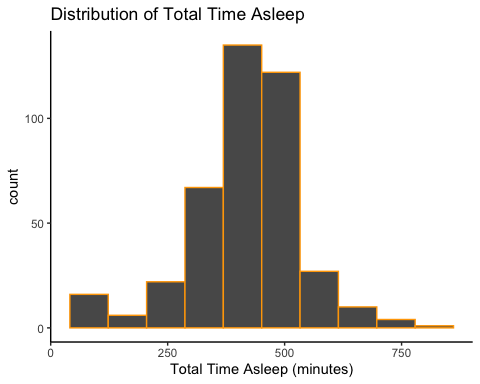
| skim\_variable | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id | 0 | 1.00 | 4.855407e+09 | 2.424805e+09 | 1503960366 | 2.320127e+09 | 4.445115e+09 | 6.962181e+09 | 8.877689e+09 | ▇▅▃▅▅ |
| total\_steps | 0 | 1.00 | 7.637910e+03 | 5.087150e+03 | 0 | 3.789750e+03 | 7.405500e+03 | 1.072700e+04 | 3.601900e+04 | ▇▇▁▁▁ |
| total\_distance | 0 | 1.00 | 5.490000e+00 | 3.920000e+00 | 0 | 2.620000e+00 | 5.240000e+00 | 7.710000e+00 | 2.803000e+01 | ▇▆▁▁▁ |
| sedentary\_minutes | 0 | 1.00 | 9.912100e+02 | 3.012700e+02 | 0 | 7.297500e+02 | 1.057500e+03 | 1.229500e+03 | 1.440000e+03 | ▁▁▇▅▇ |
| very\_active\_minutes | 0 | 1.00 | 2.116000e+01 | 3.284000e+01 | 0 | 0.000000e+00 | 4.000000e+00 | 3.200000e+01 | 2.100000e+02 | ▇▁▁▁▁ |
| calories | 0 | 1.00 | 2.303610e+03 | 7.181700e+02 | 0 | 1.828500e+03 | 2.134000e+03 | 2.793250e+03 | 4.900000e+03 | ▁▆▇▃▁ |
| total\_sleep\_records | 530 | 0.44 | 1.120000e+00 | 3.500000e-01 | 1 | 1.000000e+00 | 1.000000e+00 | 1.000000e+00 | 3.000000e+00 | ▇▁▁▁▁ |
| total\_minutes\_asleep | 530 | 0.44 | 4.191700e+02 | 1.186400e+02 | 58 | 3.610000e+02 | 4.325000e+02 | 4.900000e+02 | 7.960000e+02 | ▁▂▇▃▁ |
| total\_time\_in\_bed | 530 | 0.44 | 4.584800e+02 | 1.274600e+02 | 61 | 4.037500e+02 | 4.630000e+02 | 5.260000e+02 | 9.610000e+02 | ▁▃▇▁▁ |

# Analysis and Visualisation

To get interesting insights from our merged datasets, we’ll be comparing different variables from our datasets and viewing relationships via plots.

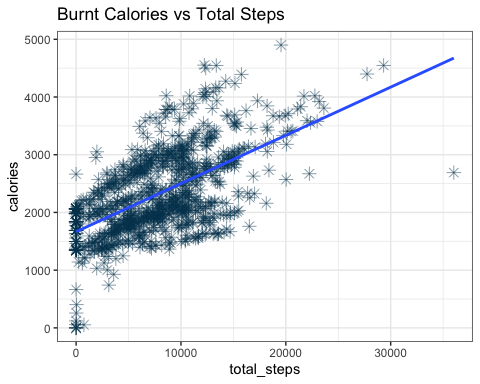
5.1 # Measuring Relationships between Variables

#PLOT\_1 To observe the distribution of time spent sleeping by participants.  
ggplot(data = merged\_data\_unique, mapping = aes(total\_minutes\_asleep)) +   
 geom\_histogram(bins = 10, na.rm = TRUE, color = "#ffa600") +   
 labs(title="Distribution of Total Time Asleep", x="Total Time Asleep (minutes)") +   
 theme\_classic()



#PLOT\_2 Burnt calories vs total steps - To check the relationship between steps\_taken and calories\_burnt  
ggplot(data = merged\_data\_unique, mapping = aes(x = total\_steps, y = calories)) +  
 geom\_point(alpha = 0.5, shape = 8, color = "#003f5c", size = 3) +   
 geom\_smooth(method = "lm", se = FALSE) +  
 theme\_bw() +  
 labs(title = "Burnt Calories vs Total Steps")

## `geom\_smooth()` using formula = 'y ~ x'



#PLOT\_3 total time\_in\_bed vs total\_time\_asleep   
plot\_3 <- ggplot(data = merged\_data\_unique, mapping= aes(x = total\_time\_in\_bed , y = total\_minutes\_asleep)) +   
 geom\_point(mapping = aes(color = calories), alpha = 0.3, shape = 7) +   
 geom\_smooth() +   
 labs(title = "Sleep duration vs Time in Bed") +  
 theme\_classic()  
  
#PLOT\_4 Sleep duration vs Sedentary Time   
plot\_4 <- ggplot(data = merged\_data\_unique, mapping = aes(x = sedentary\_minutes, y = total\_minutes\_asleep)) +   
 geom\_point(alpha = 0.6, shape = 1, mapping = aes(color = calories)) +   
 geom\_smooth() +   
 labs(title = "Sleep duration vs Sedentary Time") +   
 theme\_classic()  
  
(plot\_3 + plot\_4) +   
 plot\_annotation(title = 'Sleep duration compared to Time in Bed and Sedentary Time') +   
 plot\_layout(ncol = 2, widths = c(2,2))

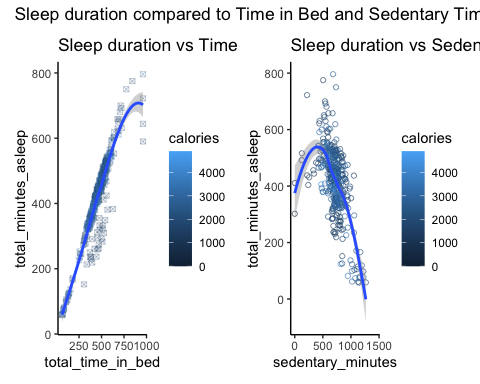
## `geom\_smooth()` using method = 'loess' and formula = 'y ~ x'

## Warning: Removed 530 rows containing non-finite values (`stat\_smooth()`).

## Warning: Removed 530 rows containing missing values (`geom\_point()`).

## `geom\_smooth()` using method = 'loess' and formula = 'y ~ x'

## Warning: Removed 530 rows containing non-finite values (`stat\_smooth()`).  
## Removed 530 rows containing missing values (`geom\_point()`).

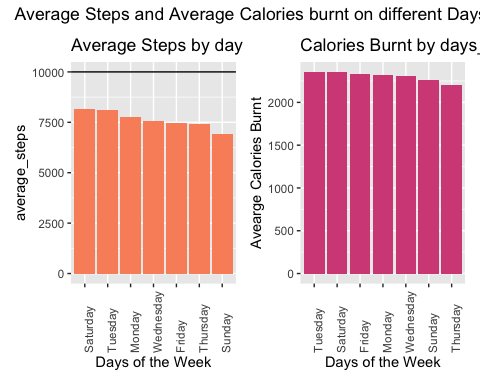


Key findings based on our analysis and plots include:

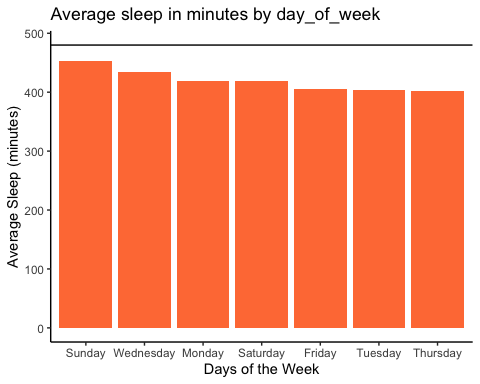
* The sleep distribution of our participants shows that majority sleep less than the the daily recommendation of 8 hours per day.
* It seems the less active our participants become, the less actual sleep they get.
* All participants records less than the CDC recommended 10,000 daily steps for an active adult.
* It appears that a positive relationship exists between total steps and calories burnt.

5.2 ## Measuring Daily Data In the next plots, it will be good to see how the steps\_taken by participants, burnt\_calories and total\_sleep varies on each day of the week. To do this, we need to carry out some data aggregation and summary as shown below:

#computing total steps by weekday  
total\_steps\_by\_weekday <- merged\_data\_unique %>% group\_by(weekday) %>%  
 summarise(steps= sum(total\_steps, na.rm = TRUE),   
 Mean = mean(total\_steps, na.rm = TRUE))  
  
#computing total calories by weekday  
total\_calories\_by\_weekday <- merged\_data\_unique %>% group\_by(weekday)%>%  
 summarise(total\_calories = sum(calories, na.rm = TRUE),   
 Mean\_calories = mean(calories, na.rm = TRUE))  
  
#computing total sleep by days of the week   
sleep\_by\_weekday <- merged\_data\_unique %>% group\_by(weekday) %>%  
 summarise(total\_sleep = sum(total\_minutes\_asleep, na.rm = TRUE),   
 average\_sleep = mean(total\_minutes\_asleep, na.rm = TRUE))  
   
#PLOT\_5 Day of the week vs average steps  
plot\_5 <- ggplot(data = total\_steps\_by\_weekday, mapping = aes(x = reorder(weekday, -Mean), y = Mean)) +  
 geom\_col(fill = "#fa9169") +   
 geom\_hline(yintercept = 10000) +  
 ggeasy::easy\_rotate\_labels(which = "x", angle = 90) +  
 labs(title = "Average Steps by days\_of\_week", y = "average\_steps", x = "Days of the Week")   
   
   
#PLOT\_6 Day of the week vs average calories burnt  
plot\_6 <- ggplot(data = total\_calories\_by\_weekday, mapping = aes(x = reorder(weekday, -Mean\_calories), y= Mean\_calories)) +   
 geom\_col(fill="#d45087") +   
 labs(title = "Calories Burnt by days\_of\_week", y = "Avearge Calories Burnt", x = "Days of the Week") +  
 ggeasy::easy\_rotate\_labels(which = "x", angle = 90)   
   
  
(plot\_5 + plot\_6) +  
 plot\_annotation(title = 'Average Steps and Average Calories burnt on different Days of the Weeek') +   
 plot\_layout(ncol = 2, widths = c(1,1))



#PLOT\_7 Day of the week vs Sleep   
ggplot(data = sleep\_by\_weekday, mapping = aes(x = reorder(weekday, -average\_sleep), y = average\_sleep)) +   
 geom\_col(fill="#ff7c43" ) +   
 geom\_hline(yintercept = 480) +  
 labs(title = "Average sleep in minutes by day\_of\_week", y = "Average Sleep (minutes)", x = "Days of the Week") +   
 theme\_classic()



Further important findings based day of the week include the following: \* Highest number of average steps taken was recored on Saturday while the least was recorded on Sunday. \* Maximum amount of average calories burnt was recorded on Saturday while the least was recorded on Sunday. This agrees with our earlier findings about the average stpes and calories burnt. \* The least amount of sleep was recorded on Thursday, and highest was recorded on Sunday.

5.3 # Grouping Participants

Finally, I think it will be interesting to categorize participants on the level of their activity. This classification will be based on whether participants are sedentary, lightly active, fairly active or very active. The source used for the categorisation can be found by clicking on this link.

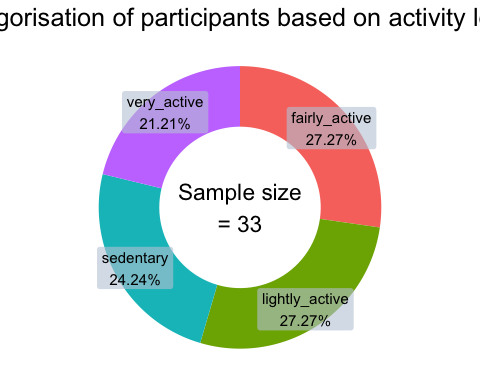
#Group participants into different categories  
daily\_activity\_average <- merged\_data\_unique %>%  
 group\_by(id) %>%  
 summarise(daily\_calories\_average = mean(calories),  
 daily\_steps\_average = mean(total\_steps),  
 daily\_sleep\_average = mean(total\_minutes\_asleep, na.rm = TRUE)) %>%  
 mutate(user\_categories = case\_when(  
 daily\_steps\_average < 5000 ~ "sedentary",  
 daily\_steps\_average >= 5000 & daily\_steps\_average < 7499 ~ "lightly\_active",  
 daily\_steps\_average >= 7499 & daily\_steps\_average < 9999 ~ "fairly\_active",  
 daily\_steps\_average >= 10000 ~ "very\_active"  
 ))  
  
#verify the daily\_activity\_average  
head(daily\_activity\_average)

## # A tibble: 6 × 5  
## id daily\_calories\_average daily\_steps\_average daily\_sleep\_average  
## <dbl> <dbl> <dbl> <dbl>  
## 1 1503960366 1816. 12117. 360.  
## 2 1624580081 1483. 5744. NaN   
## 3 1644430081 2811. 7283. 294   
## 4 1844505072 1573. 2580. 652   
## 5 1927972279 2173. 916. 417   
## 6 2022484408 2510. 11371. NaN   
## # ℹ 1 more variable: user\_categories <chr>

#Getting the fraction of active categories  
usertype\_data <- daily\_activity\_average %>%  
 group\_by(user\_categories) %>%  
 summarise(total = n()) %>%  
 mutate(category\_fraction = (total/sum(total)))  
  
#verify usertype data   
usertype\_data

## # A tibble: 4 × 3  
## user\_categories total category\_fraction  
## <chr> <int> <dbl>  
## 1 fairly\_active 9 0.273  
## 2 lightly\_active 9 0.273  
## 3 sedentary 8 0.242  
## 4 very\_active 7 0.212

# PLOT 8: Doughnut chart of participants on activity levels - The very active category represents the smallest group of the entire sample accounting for only 21.21%.   
usertype\_distribution <- usertype\_data %>%  
 mutate(ymax = cumsum(category\_fraction)) %>%  
 mutate(ymin = c(0, head(ymax, n=-1))) %>%  
 mutate(label\_position = (ymax + ymin)/2) %>%  
 mutate(label = paste0(user\_categories, "\n", round(category\_fraction\*100, digits = 2), "%"))  
  
ggplot(usertype\_distribution, aes(  
 ymax = ymax, ymin = ymin, xmax = 7, xmin = 4, fill = user\_categories)) +   
 geom\_rect() +   
 geom\_label( x = 6, aes(y = label\_position, label = label), fill = "#bfcbdb", inherit.aes = FALSE, alpha = 0.6, size = 4, label.size = 0) +  
 ggtitle("Categorisation of participants based on activity levels") +   
 scale\_color\_brewer(palette = 2) +   
 coord\_polar(theta = "y") +   
 theme\_void() +   
 annotate("text", x = 0, y = 0, label = "Sample size\n= 33", size = 6) +   
 theme(legend.position = "none", plot.title = element\_text(size = 19, hjust = 0.5))



Final observation based on grouping participants based on their activity levels reveals the following:

* The very active group accounts for only 21.21% of the total participants and represents the smallest group.
* The lightly active and failry active group accounts for 27.27% and represents the largest group.

6.0 #Act - Recommendation

Looking at the key findings from our analysis phase, we can make the following recommendations based on those:

* It will be an interesting and potential selling point for customers if the Bellabeat app could be designed to measure the relationship between suboptimal sleeping habits and stress levels.
* The app could be modified to include push notifications or reminders that encourage more activities on days with high sedentary levels like Sundays.
* The app could be designed to show health benefits associated with keeping an active lifestyle. This will encourage users to engage the app more often in achieving their exercise goals.
* The app could be designed to encourage the development of healthy communities through the formation of work-out groups and clubs. This will help customers to attain their health goals whilst fostering social interaction and integration.