Capstone Case-Study

Mubarak

2023-08-30

Capstone Project: How Can a Wellness Technology Company Play It Smart?

Bellabeat is a high-tech company that manufactures health focused smart products for women.

1. ASK

1.1 Summary of Business Task

Conduct an in-depth analysis of public datasets of non-Bellabeat smart devices to identify trends in consumer usage. Extract meaningful insights from the analysis that can be incorporated to a Bellabeat product to enhance product offerings, improve user experience, and strengthen the company's position in the global health and fitness market.

How can your insights drive business decisions? Leveraging analysis insights to formulate a marketing plan for Bellabeat, driving increased customer acquisition and product expansion in the global market.

1.2 Key Stakeholders

- Urška Sršen - Bellabeat's co-founder and Chief Creative Officer
- Sando Mur: Co-founder and an integral member of the Bellabeat executive team
- Bellabeat marketing analytics team

2. PREPARE

The data set, Fitbit Fitness Tracker Data, is collected from Kaggle and is a third-party data https://www.kaggle.com/datasets/arashnic/fitbit.

Integral Data Characteristics

- Data is reliable. Information presented in all the tables are directly from eligible Fitbit users.
- Data is highly credible and original as it is collected from the data owner, Mobius https://www.kaggle.com/arashnic.
- Data has been cited by significant number of people

Limitations

- The data sets are based on information collected from approximately 30 respondents, which is too little of a sample size to discover and generate concrete results that can be acted upon.
- Data didn't provide the gender of participants. It would have been much beneficial if the gender of respondents were provided because Bellabeat products and services are catered for women.

3. PROCESS

3.1 Data Cleaning & Processing

Tools

- R

Data sets

- dailyactivity_merged
- sleepday_merged

3.2 Data cleaning, processing with R

```
library(tidyverse)
```

Load packages ("tidyverse", "lubridate", "gglopt2", "dplyr")

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

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

```
DailyActivity <- read_csv("dailyActivity_merged.csv")
```

Load data sets and assign names to tables

```
## 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.
```

```r
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~
```

```r
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~
```

```r
sum(duplicated(DailyActivity))
```

Check for duplicates

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

```r
sum(duplicated(SleepDay))
```

```
## [1] 3
```

```r
SleepDay <- SleepDay[!duplicated(SleepDay
                                         [c("Id", "SleepDay", "TotalMinutesAsleep", "TotalTimeInBed")])
```

Remove duplicates rows from SleepDay table using specific columns

```r
sum(duplicated(SleepDay))
```

Verify that duplicates rows have been removed

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

Change Date on both tables from "character" to "date" format and extract days of the week.

```r
typeof(DailyActivity$ActivityDate)
```

Determine current date format

```
## [1] "character"
```

```r
typeof(SleepDay$SleepDay)
```

```
## [1] "character"
```

```r
DailyActivity <- DailyActivity %>%
  mutate(Date = mdy(ActivityDate), Day = weekdays.Date(Date))
is.Date(DailyActivity$Date)
```

Convert to date format and extract days of the week

```
## [1] TRUE
```

```r
SleepDay <- SleepDay %>%
  mutate(Date = as_date(mdy_hms(SleepDay)), Day = weekdays.Date(Date))
is.Date(SleepDay$Date)
```

```
## [1] TRUE
```

```r
DailyActivity <- DailyActivity %>%
  select(-ActivityDate)
SleepDay <- SleepDay %>%
  select(-SleepDay)
```

```r
glimpse(DailyActivity)
```

Remove previous date columns of both tables

```
## Rows: 940
## Columns: 16
## $ Id                       <dbl> 1503960366, 1503960366, 1503960366, 150396036~
## $ 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~
## $ Date                     <date> 2016-04-12, 2016-04-13, 2016-04-14, 2016-04-~
## $ Day                      <chr> "Tuesday", "Wednesday", "Thursday", "Friday",~
```

```r
glimpse(SleepDay)
```

```
## Rows: 410
## Columns: 6
## $ Id                 <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 150~
## $ 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~
## $ Date               <date> 2016-04-12, 2016-04-13, 2016-04-15, 2016-04-16, 20~
## $ Day                <chr> "Tuesday", "Wednesday", "Friday", "Saturday", "Sund~
```

```
DailyActivity <- DailyActivity %>%
  mutate(TotalActiveMinutes = VeryActiveMinutes + FairlyActiveMinutes + LightlyActiveMinutes)
glimpse(DailyActivity)
```

Add a new column "TotalActiveMinutes" to "DailyActivity" table

```
## Rows: 940
## Columns: 17
## $ Id                       <dbl> 1503960366, 1503960366, 1503960366, 150396036~
## $ 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~
## $ Date                     <date> 2016-04-12, 2016-04-13, 2016-04-14, 2016-04-~
## $ Day                      <chr> "Tuesday", "Wednesday", "Thursday", "Friday",~
## $ TotalActiveMinutes       <dbl> 366, 257, 222, 272, 267, 222, 291, 345, 245, ~
```

```
ActiveMinutes_Range <- cut(DailyActivity$TotalActiveMinutes,
                                breaks = c(-Inf
                                           ,100 ,200 ,300 ,400 ,500 ,600),
                                labels = c("0-99" , "100-199" , "200-299" , "300-399" , "400-499" , "50
                                right = FALSE)
```

```
DailyActivity <- DailyActivity %>%
    mutate(ActiveMinutes_Range)
glimpse(DailyActivity)
```

Add a new column "ActiveMinutes_Range" to represent range of active minutes per observed day

```
## Rows: 940
## Columns: 18
## $ Id                       <dbl> 1503960366, 1503960366, 1503960366, 150396036~
## $ 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~
## $ Date                     <date> 2016-04-12, 2016-04-13, 2016-04-14, 2016-04-~
## $ Day                      <chr> "Tuesday", "Wednesday", "Thursday", "Friday",~
## $ TotalActiveMinutes       <dbl> 366, 257, 222, 272, 267, 222, 291, 345, 245, ~
## $ ActiveMinutes_Range      <fct> 300-399, 200-299, 200-299, 200-299, 200-299, ~
```

```
CombinedTable <- merge(DailyActivity, SleepDay, by = c("Id", "Date", "Day"))
```

```
str(CombinedTable)
```

Merge both tables

```
## 'data.frame':    410 obs. of  21 variables:
##  $ Id                      : num  1.5e+09 1.5e+09 1.5e+09 1.5e+09 1.5e+09 ...
##  $ Date                    : Date, format: "2016-04-12" "2016-04-13" ...
##  $ Day                     : chr  "Tuesday" "Wednesday" "Friday" "Saturday" ...
##  $ TotalSteps              : num  13162 10735 9762 12669 9705 ...
##  $ TotalDistance           : num  8.5 6.97 6.28 8.16 6.48 ...
##  $ TrackerDistance         : num  8.5 6.97 6.28 8.16 6.48 ...
##  $ LoggedActivitiesDistance: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ VeryActiveDistance      : num  1.88 1.57 2.14 2.71 3.19 ...
##  $ ModeratelyActiveDistance: num  0.55 0.69 1.26 0.41 0.78 ...
##  $ LightActiveDistance     : num  6.06 4.71 2.83 5.04 2.51 ...
##  $ SedentaryActiveDistance : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ VeryActiveMinutes       : num  25 21 29 36 38 50 28 19 41 39 ...
##  $ FairlyActiveMinutes     : num  13 19 34 10 20 31 12 8 21 5 ...
##  $ LightlyActiveMinutes    : num  328 217 209 221 164 264 205 211 262 238 ...
##  $ SedentaryMinutes        : num  728 776 726 773 539 775 818 838 732 709 ...
##  $ Calories                : num  1985 1797 1745 1863 1728 ...
##  $ TotalActiveMinutes      : num  366 257 272 267 222 345 245 238 324 282 ...
##  $ ActiveMinutes_Range     : Factor w/ 6 levels "0-99","100-199",..: 4 3 3 3 3 4 3 3 4 3 ...
##  $ TotalSleepRecords       : num  1 2 1 2 1 1 1 1 1 1 ...
##  $ TotalMinutesAsleep      : num  327 384 412 340 700 304 360 325 361 430 ...
##  $ TotalTimeInBed          : num  346 407 442 367 712 320 377 364 384 449 ...
```

```
CombinedTable <- CombinedTable %>%
  select(Id, Date, Day, TotalActiveMinutes, ActiveMinutes_Range, TotalMinutesAsleep, Calories)
head(CombinedTable)
```

```
##           Id       Date       Day TotalActiveMinutes ActiveMinutes_Range
## 1 1503960366 2016-04-12   Tuesday                366             300-399
## 2 1503960366 2016-04-13 Wednesday                257             200-299
## 3 1503960366 2016-04-15    Friday                272             200-299
## 4 1503960366 2016-04-16  Saturday                267             200-299
## 5 1503960366 2016-04-17    Sunday                222             200-299
## 6 1503960366 2016-04-19   Tuesday                345             300-399
##   TotalMinutesAsleep Calories
## 1                327     1985
## 2                384     1797
## 3                412     1745
## 4                340     1863
## 5                700     1728
## 6                304     2035
```

4. ANALYZE

I will be using both the "DailyActivity" table and "CombinedTable" table for further analysis to derive insights

Add a column "UnproductiveMinutes" to the "CombinedTable" table

```
CombinedTable <- CombinedTable %>%
  mutate(UnproductiveMinutes = 1440 - (TotalActiveMinutes + TotalMinutesAsleep))
glimpse(CombinedTable)
```

Subtract the sum of "TotalActiveMinutes" and "TotalMinutesAsleep" from available minutes per day (1400)

```
## Rows: 410
## Columns: 8
## $ Id                  <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 15~
## $ Date                <date> 2016-04-12, 2016-04-13, 2016-04-15, 2016-04-16, 2~
## $ Day                 <chr> "Tuesday", "Wednesday", "Friday", "Saturday", "Sun~
## $ TotalActiveMinutes  <dbl> 366, 257, 272, 267, 222, 345, 245, 238, 324, 282, ~
## $ ActiveMinutes_Range <fct> 300-399, 200-299, 200-299, 200-299, 200-299, 300-3~
## $ TotalMinutesAsleep  <dbl> 327, 384, 412, 340, 700, 304, 360, 325, 361, 430, ~
## $ Calories            <dbl> 1985, 1797, 1745, 1863, 1728, 2035, 1786, 1775, 19~
## $ UnproductiveMinutes <dbl> 747, 799, 756, 833, 518, 791, 835, 877, 755, 728, ~
```

"UnproductiveMinutes" column reveals more accurate values ofrespondents inactive minutes per day.

Add two new columns, "PercentageProductive" and "PercentageUnproductive" to CombinedTable.

```
CombinedTable <- CombinedTable %>%
  mutate(PercentageUnproductive = UnproductiveMinutes/1400 * 100)
```

```
CombinedTable <- CombinedTable %>%
  mutate(PercentageProductive = 100 - PercentageUnproductive)
```

```
head(CombinedTable)
```

```
##           Id       Date       Day TotalActiveMinutes ActiveMinutes_Range
## 1 1503960366 2016-04-12   Tuesday                366             300-399
## 2 1503960366 2016-04-13 Wednesday                257             200-299
## 3 1503960366 2016-04-15    Friday                272             200-299
## 4 1503960366 2016-04-16  Saturday                267             200-299
## 5 1503960366 2016-04-17    Sunday                222             200-299
## 6 1503960366 2016-04-19   Tuesday                345             300-399
##   TotalMinutesAsleep Calories UnproductiveMinutes PercentageUnproductive
## 1                327     1985                 747               53.35714
## 2                384     1797                 799               57.07143
## 3                412     1745                 756               54.00000
## 4                340     1863                 833               59.50000
## 5                700     1728                 518               37.00000
## 6                304     2035                 791               56.50000
##   PercentageProductive
## 1             46.64286
## 2             42.92857
## 3             46.00000
## 4             40.50000
## 5             63.00000
## 6             43.50000
```

4.1 Data Aggregation

```
DailyProductivity <- CombinedTable %>%
    group_by(Day) %>%
    drop_na() %>%
    summarize(AveragePercentage_Productive = mean(PercentageProductive),
              AveragePercentage_Unproductive = mean(PercentageUnproductive))
```

```
DailyProductivity
```

Create a table to represent daily percentage productivity and unproductivity

```
## # A tibble: 7 x 3
##   Day       AveragePercentage_Productive AveragePercentage_Unproductive
##   <chr>                            <dbl>                          <dbl>
## 1 Friday                            44.6                           55.4
## 2 Monday                            46.5                           53.5
## 3 Saturday                          48.3                           51.7
## 4 Sunday                            46.5                           53.5
## 5 Thursday                          43.1                           56.9
## 6 Tuesday                           45.1                           54.9
## 7 Wednesday                         45.8                           54.2
```

```
AverageCalories_Expenditure <- DailyActivity %>%
    group_by(ActiveMinutes_Range) %>%
    drop_na() %>%
    summarise(AverageCalories = mean(Calories))
```

```
AverageCalories_Expenditure
```

Create a second table to represent calories burnt per range of active minutes

```
## # A tibble: 6 x 2
##   ActiveMinutes_Range AverageCalories
##   <fct>                         <dbl>
## 1 0-99                          1753.
## 2 100-199                       2049.
## 3 200-299                       2443.
## 4 300-399                       2557.
## 5 400-499                       2889.
## 6 500-599                       3223.
```

```
StepsDistance <- DailyActivity %>%
    group_by(ActiveMinutes_Range) %>%
    drop_na() %>%
    summarise(AverageSteps = mean(TotalSteps),
             AverageDistance = mean(TotalDistance))
```

```
StepsDistance
```

Create a third table to show relationship between steps and distance

```
## # A tibble: 6 x 3
##   ActiveMinutes_Range AverageSteps AverageDistance
##   <fct>                      <dbl>           <dbl>
## 1 0-99                       1147.           0.827
## 2 100-199                    4759.           3.39
## 3 200-299                    8903.           6.44
## 4 300-399                   11271.           8.04
## 5 400-499                   13339.           9.58
## 6 500-599                   17322.          13.9
```

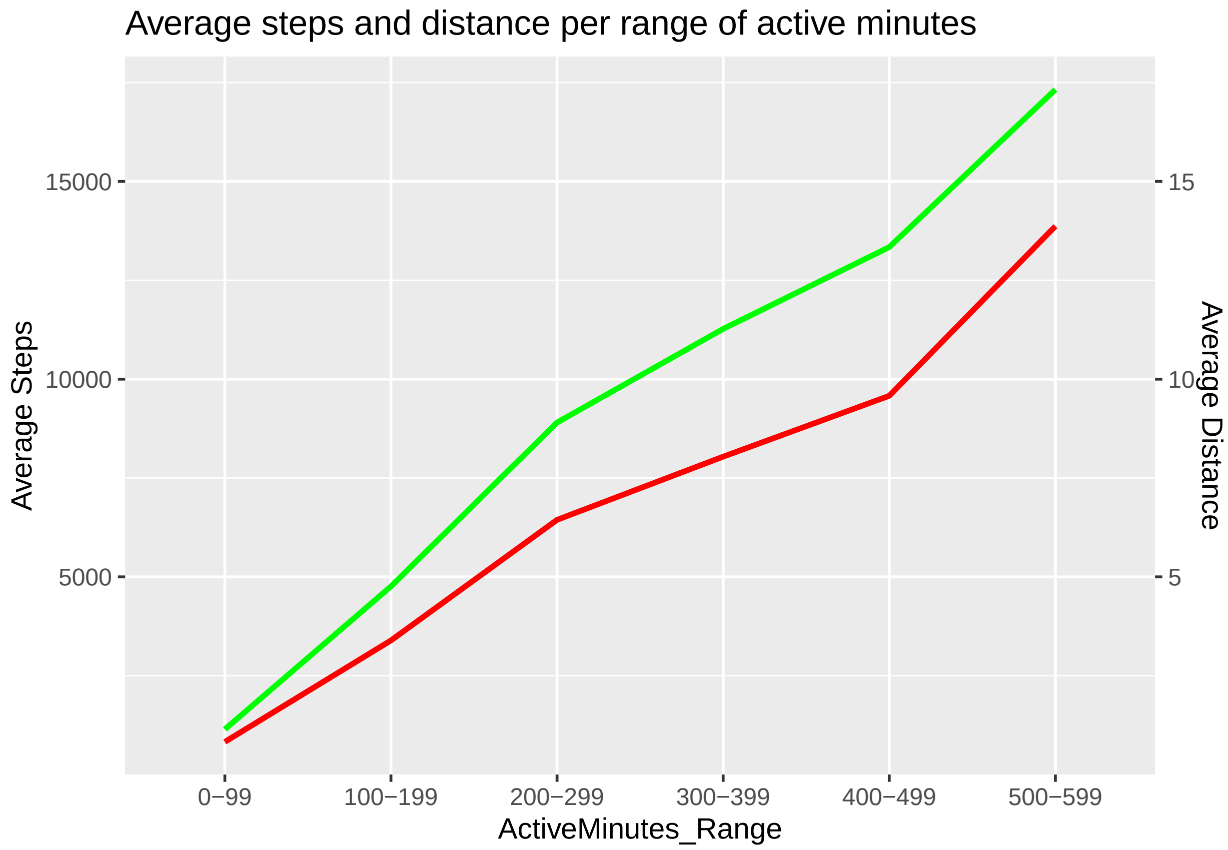
5. SHARE

Develop visualizations and derive insights

Visualization 1

```
 ggplot(StepsDistance, aes(x = ActiveMinutes_Range)) +
    geom_line(aes(y = AverageSteps, group = 1), size = 1, color = "Green") +
    geom_line(aes(y = AverageDistance*1000, group = 1), size = 1, color = "red") +
    scale_y_continuous(name = "Average Steps",
    sec.axis = sec_axis(~./1000, name = "Average Distance")) +
    labs(title = "Average steps and distance per range of active minutes")
```

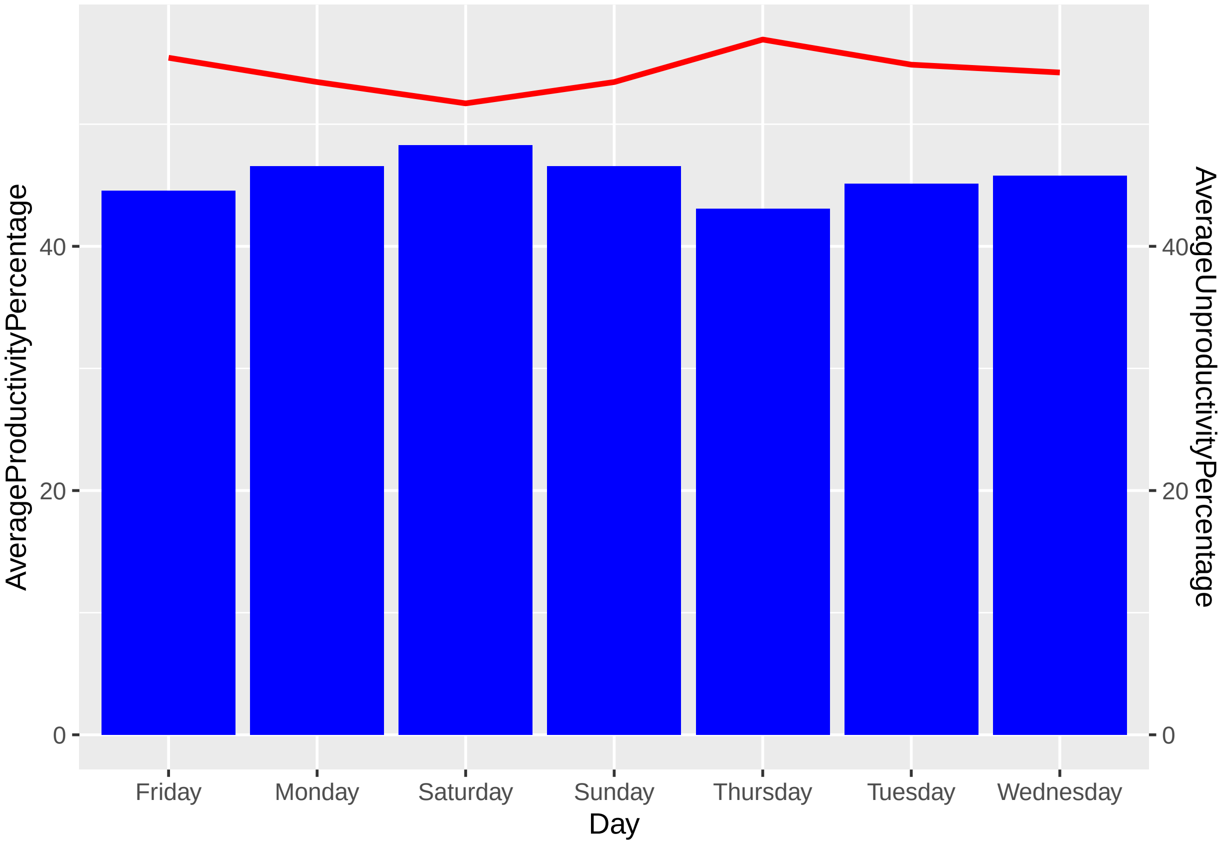
```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



The chart reflects that on average, participants whose active minutes are between 500 and 599 covers the most distance and take the most steps. The greater the range of daily active minutes, the higher the steps and distance covered by members.

Visualization 2

```
ggplot(DailyProductivity, aes(x = Day)) +
    geom_bar(aes(y = AveragePercentage_Productive), stat = "identity", size = 0.1, fill = "blue") +
    geom_line(aes(y = AveragePercentage_Unproductive, group = 1), size = 1, color = "red") +
    scale_y_continuous(name = "AverageProductivityPercentage",
    sec.axis = sec_axis(~., name = "AverageUnproductivityPercentage"))
```

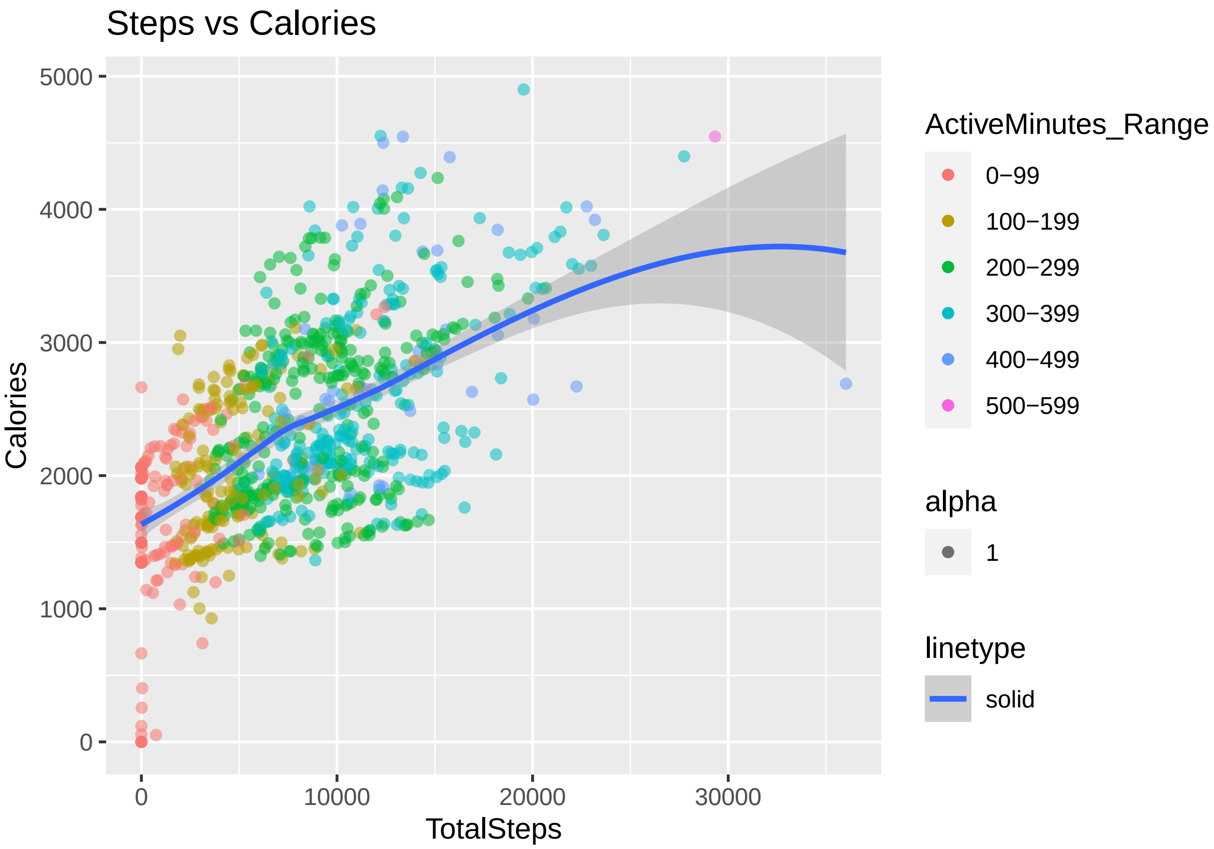


The results shows that less than 50% of participants time each day are spent on productive activities.

Visualization 3

```
ggplot(data = DailyActivity) +
    geom_point(mapping = aes(x = TotalSteps, y = Calories, color = ActiveMinutes_Range, alpha = 1)) +
    geom_smooth(mapping = aes(x = TotalSteps, y = Calories, linetype = "solid")) +
    labs(title = "Steps vs Calories")
```

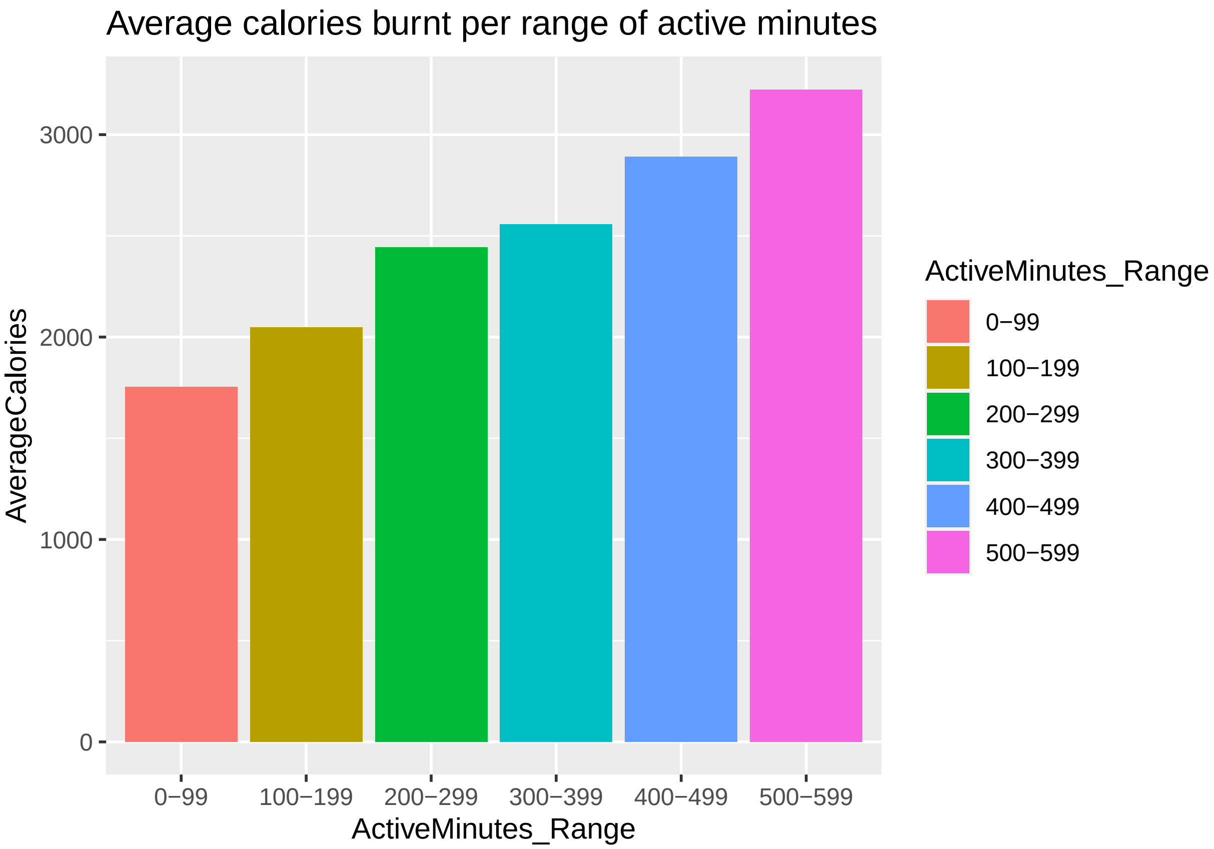
```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



The result shows a direct impact steps taken has on calories burnt.

Visualization 4

```
ggplot(AverageCalories_Expenditure, aes(x = ActiveMinutes_Range, y = AverageCalories, fill = ActiveMinu
  geom_bar(stat = "identity") +
  labs(title = "Average calories burnt per range of active minutes")
```



This visualisation indicates that there is a direct relationship between the amount of time spent per day and the calories burned. The more time expended on active duties, the greater the calorie burnt.

6. ACT

6.1 Recommendation and Conclusion

The analysis of Fitbit Fitness Tracker data has revealed valuable insights into the daily activity patterns of members, revealing the distribution of active and inactive minutes. Notably, it is evident that the time members allocate to productive activities directly influences their calorie expenditure. The results from the analysis could lead to a significant enhancement in the daily health performance of people seeking a healthy lifestyle.

Based on the results at hand, there is a unique opportunity to tailor strategies and interventions that can empower individuals to lead healthier and more active lives, ultimately contributing to their overall well-being.

The results from the analysis reveal opportunities for Bellabeat to optimise the effectiveness of the Bellabeat app. A boost in the performance of users and the percentage of time they spend being active can be realised by adding features and programs that enable users to stay active and keep track of their health life.

6.2 Marketing Recommendations

- Devise a marketing initiative that would convey plans to assist women in improving their health life, using the Bellabeat App.
- Support the marketing operation by providing health and fitness programs for customers, which can range by level of difficulty, such as easy, amateur, medium, difficult and master levels.

- Inform customers of goals that can be achieved for each difficulty level program, as well as the active minute targets for each level.
- Establish daily distance and step goals for members and implement daily training programs and offer guidance to support users in maintaining a consistent and healthy lifestyle.
- Communicate to customer about the unique value of the App, such as it will allow users to designate their most available days of the week and it contains customized programs to optimize users' productivity during those specific days.