Case study Fitness app

Caroline S

2024-04-03

Case study Bellabeat

1 Ask

Deliverable: A clear statement of the business task

How can Bellabeat grow and adjust marketing to the growth opportunities?

Including: 1. What are some trends in smart device usage? 2. How could these trends apply to Bellabeat customers? 3. How could these trends help influence 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.

2 Prepare

The data is retrieved from Kaggle: "FitBit Fitness Tracker Data" 02.04.24. Then it is stored locally and uploaded to R studio. The data is licensed with public domain with a usability of 8.75. It is stored in a long data format.

A problem with the data is that it is old - from 2016 and public for all. Health data should be handled with confidentiality, but this is a case study. Another problem with the data is that even if it is a lot of data it is only for a sample of around 30 people. This is checked with R for unique Id. See below.

The data does not contain so much information about the user groups. In marketing it is important to know who their buyer persona is. Therfore we try to analyze the users based on trends and see what trends there is. This will focus on steps and weight.

3 Process

I am choosing R studio as a tool because I need to visualize it and it is a huge amount of data. I used R for checking for errors. Previewed the data with head, view and glimpse.

Some errors: The headers are in camelcase for example we clean column header ActivityDate to activity_date. We clean this with the clean names function.

Another error was some duplicates in the sleep data. The duplicated data was cleaned.

Loading packages

Most of the packages was already installed, and now loaded.

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.0     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## 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(lubridate)
library(dplyr)
library(ggplot2)
library(tidyr)
install.packages("janitor")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.3'
## (as 'lib' is unspecified)
```

```
library(janitor)
```

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

Import CSV files

These CSV files were loaded from the data from the Bellabeat app.

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

```
## Rows: 457 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.
```

```
heartrate <- read_csv("heartrate_seconds_merged.csv")
```

```
## Rows: 1154681 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.
```

```
hourly_calories <- read.csv("hourlyCalories_merged.csv")
hourly_intensities <- read.csv("hourlyIntensities_merged.csv")
hourly_steps <- read.csv("hourlySteps_merged.csv")
```

```r
minute_Calories <- read.csv("minuteCaloriesNarrow_merged.csv")
minute_sleep <- read.csv ("minuteSleep_merged.csv")
minute_steps <- read.csv ("minuteStepsNarrow_merged.csv")
weight_loginfo <- read.csv ("weightLogInfo_merged.csv")
```

Viewing the data

```r
glimpse (daily_activity)
```

```
## Rows: 457
## Columns: 15
## $ Id                       <dbl> 1503960366, 1503960366, 1503960366, 150396036~
## $ ActivityDate             <chr> "3/25/2016", "3/26/2016", "3/27/2016", "3/28/~
## $ TotalSteps               <dbl> 11004, 17609, 12736, 13231, 12041, 10970, 122~
## $ TotalDistance            <dbl> 7.11, 11.55, 8.53, 8.93, 7.85, 7.16, 7.86, 7.~
## $ TrackerDistance          <dbl> 7.11, 11.55, 8.53, 8.93, 7.85, 7.16, 7.86, 7.~
## $ LoggedActivitiesDistance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ VeryActiveDistance       <dbl> 2.57, 6.92, 4.66, 3.19, 2.16, 2.36, 2.29, 3.3~
## $ ModeratelyActiveDistance <dbl> 0.46, 0.73, 0.16, 0.79, 1.09, 0.51, 0.49, 0.8~
## $ LightActiveDistance      <dbl> 4.07, 3.91, 3.71, 4.95, 4.61, 4.29, 5.04, 3.6~
## $ SedentaryActiveDistance  <dbl> 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.0~
## $ VeryActiveMinutes        <dbl> 33, 89, 56, 39, 28, 30, 33, 47, 40, 15, 43, 3~
## $ FairlyActiveMinutes      <dbl> 12, 17, 5, 20, 28, 13, 12, 21, 11, 30, 18, 18~
## $ LightlyActiveMinutes     <dbl> 205, 274, 268, 224, 243, 223, 239, 200, 244, ~
## $ SedentaryMinutes         <dbl> 804, 588, 605, 1080, 763, 1174, 820, 866, 636~
## $ Calories                 <dbl> 1819, 2154, 1944, 1932, 1886, 1820, 1889, 186~
```

```r
head (daily_activity)
```

```
## # A tibble: 6 x 15
##           Id ActivityDate TotalSteps TotalDistance TrackerDistance
##        <dbl> <chr>             <dbl>         <dbl>           <dbl>
## 1 1503960366 3/25/2016         11004          7.11            7.11
## 2 1503960366 3/26/2016         17609         11.6            11.6
## 3 1503960366 3/27/2016         12736          8.53            8.53
## 4 1503960366 3/28/2016         13231          8.93            8.93
## 5 1503960366 3/29/2016         12041          7.85            7.85
## 6 1503960366 3/30/2016         10970          7.16            7.16
## # i 10 more variables: LoggedActivitiesDistance <dbl>,
## #   VeryActiveDistance <dbl>, ModeratelyActiveDistance <dbl>,
## #   LightActiveDistance <dbl>, SedentaryActiveDistance <dbl>,
## #   VeryActiveMinutes <dbl>, FairlyActiveMinutes <dbl>,
## #   LightlyActiveMinutes <dbl>, SedentaryMinutes <dbl>, Calories <dbl>
```

```r
glimpse (heartrate)
```

```
## Rows: 1,154,681
## Columns: 3
## $ Id    <dbl> 2022484408, 2022484408, 2022484408, 2022484408, 2022484408, 2022~
## $ Time  <chr> "4/1/2016 7:54:00 AM", "4/1/2016 7:54:05 AM", "4/1/2016 7:54:10 ~
## $ Value <dbl> 93, 91, 96, 98, 100, 101, 104, 105, 102, 106, 109, 112, 111, 109~
```

```r
head (heartrate)
```

```
## # A tibble: 6 x 3
##           Id Time                 Value
```

```
##        <dbl> <chr>               <dbl>
## 1 2022484408 4/1/2016 7:54:00 AM    93
## 2 2022484408 4/1/2016 7:54:05 AM    91
## 3 2022484408 4/1/2016 7:54:10 AM    96
## 4 2022484408 4/1/2016 7:54:15 AM    98
## 5 2022484408 4/1/2016 7:54:20 AM   100
## 6 2022484408 4/1/2016 7:54:25 AM   101
```

```
glimpse (hourly_steps)
```

```
## Rows: 24,084
## Columns: 3
## $ Id           <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 150396036~
## $ ActivityHour <chr> "3/12/2016 12:00:00 AM", "3/12/2016 1:00:00 AM", "3/12/20~
## $ StepTotal    <int> 0, 0, 0, 0, 0, 0, 0, 0, 8, 551, 1764, 1259, 253, 4470,~
```

```
head (hourly_steps)
```

```
##           Id          ActivityHour StepTotal
## 1 1503960366 3/12/2016 12:00:00 AM         0
## 2 1503960366  3/12/2016 1:00:00 AM         0
## 3 1503960366  3/12/2016 2:00:00 AM         0
## 4 1503960366  3/12/2016 3:00:00 AM         0
## 5 1503960366  3/12/2016 4:00:00 AM         0
## 6 1503960366  3/12/2016 5:00:00 AM         0
```

```
glimpse (minute_Calories)
```

```
## Rows: 1,445,040
## Columns: 3
## $ Id             <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 1503960~
## $ ActivityMinute <chr> "3/12/2016 12:00:00 AM", "3/12/2016 12:01:00 AM", "3/12~
## $ Calories       <dbl> 0.7973, 0.7973, 0.7973, 0.7973, 0.7973, 0.7973, 0.7973,~
```

```
head (minute_Calories)
```

```
##           Id        ActivityMinute Calories
## 1 1503960366 3/12/2016 12:00:00 AM   0.7973
## 2 1503960366 3/12/2016 12:01:00 AM   0.7973
## 3 1503960366 3/12/2016 12:02:00 AM   0.7973
## 4 1503960366 3/12/2016 12:03:00 AM   0.7973
## 5 1503960366 3/12/2016 12:04:00 AM   0.7973
## 6 1503960366 3/12/2016 12:05:00 AM   0.7973
```

```
glimpse (minute_sleep)
```

```
## Rows: 198,559
## Columns: 4
## $ Id    <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 1503960366, 1503~
## $ date  <chr> "3/13/2016 2:39:30 AM", "3/13/2016 2:40:30 AM", "3/13/2016 2:41:~
## $ value <int> 1, 1, 1, 1, 1, 1, 2, 2, 1, 1, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1~
## $ logId <dbl> 11114919637, 11114919637, 11114919637, 11114919637, 11114919637,~
```

```
head (minute_sleep)
```

```
##           Id                 date value       logId
## 1 1503960366 3/13/2016 2:39:30 AM     1 11114919637
## 2 1503960366 3/13/2016 2:40:30 AM     1 11114919637
```

```
## 3 1503960366 3/13/2016 2:41:30 AM     1 11114919637
## 4 1503960366 3/13/2016 2:42:30 AM     1 11114919637
## 5 1503960366 3/13/2016 2:43:30 AM     1 11114919637
## 6 1503960366 3/13/2016 2:44:30 AM     1 11114919637
```

```
glimpse (minute_steps)
```

```
## Rows: 1,445,040
## Columns: 3
## $ Id             <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 1503960~
## $ ActivityMinute <chr> "3/12/2016 12:00:00 AM", "3/12/2016 12:01:00 AM", "3/12~
## $ Steps          <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
```

```
head (minute_steps)
```

```
##           Id        ActivityMinute Steps
## 1 1503960366 3/12/2016 12:00:00 AM     0
## 2 1503960366 3/12/2016 12:01:00 AM     0
## 3 1503960366 3/12/2016 12:02:00 AM     0
## 4 1503960366 3/12/2016 12:03:00 AM     0
## 5 1503960366 3/12/2016 12:04:00 AM     0
## 6 1503960366 3/12/2016 12:05:00 AM     0
```

```
glimpse (weight_loginfo)
```

```
## Rows: 33
## Columns: 8
## $ Id             <dbl> 1503960366, 1927972279, 2347167796, 2873212765, 2873212~
## $ Date           <chr> "4/5/2016 11:59:59 PM", "4/10/2016 6:33:26 PM", "4/3/20~
## $ WeightKg       <dbl> 53.3, 129.6, 63.4, 56.7, 57.2, 88.4, 92.4, 69.4, 99.7, ~
## $ WeightPounds   <dbl> 117.5064, 285.7191, 139.7731, 125.0021, 126.1044, 194.8~
## $ Fat            <int> 22, NA, 10, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,~
## $ BMI            <dbl> 22.97, 46.17, 24.77, 21.45, 21.65, 25.03, 35.01, 27.14,~
## $ IsManualReport <chr> "True", "False", "True", "True", "True", "True", "True"~
## $ LogId          <dbl> 1.459901e+12, 1.460313e+12, 1.459728e+12, 1.459987e+12,~
```

```
head (weight_loginfo)
```

```
##           Id                 Date WeightKg WeightPounds Fat   BMI
## 1 1503960366 4/5/2016 11:59:59 PM     53.3     117.5064  22 22.97
## 2 1927972279 4/10/2016 6:33:26 PM    129.6     285.7191  NA 46.17
## 3 2347167796 4/3/2016 11:59:59 PM     63.4     139.7731  10 24.77
## 4 2873212765 4/6/2016 11:59:59 PM     56.7     125.0021  NA 21.45
## 5 2873212765 4/7/2016 11:59:59 PM     57.2     126.1044  NA 21.65
## 6 2891001357 4/5/2016 11:59:59 PM     88.4     194.8886  NA 25.03
##   IsManualReport        LogId
## 1           True 1.459901e+12
## 2          False 1.460313e+12
## 3           True 1.459728e+12
## 4           True 1.459987e+12
## 5           True 1.460074e+12
## 6           True 1.459901e+12
```

Cleaning the data

Cleaning camel case letters in columns

```
daily_activity <- clean_names(daily_activity)
glimpse(daily_activity)
```

```
## Rows: 457
## Columns: 15
## $ id                          <dbl> 1503960366, 1503960366, 1503960366, 1503960~
## $ activity_date               <chr> "3/25/2016", "3/26/2016", "3/27/2016", "3/2~
## $ total_steps                 <dbl> 11004, 17609, 12736, 13231, 12041, 10970, 1~
## $ total_distance              <dbl> 7.11, 11.55, 8.53, 8.93, 7.85, 7.16, 7.86, ~
## $ tracker_distance            <dbl> 7.11, 11.55, 8.53, 8.93, 7.85, 7.16, 7.86, ~
## $ logged_activities_distance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
## $ very_active_distance        <dbl> 2.57, 6.92, 4.66, 3.19, 2.16, 2.36, 2.29, 3~
## $ moderately_active_distance <dbl> 0.46, 0.73, 0.16, 0.79, 1.09, 0.51, 0.49, 0~
## $ light_active_distance       <dbl> 4.07, 3.91, 3.71, 4.95, 4.61, 4.29, 5.04, 3~
## $ sedentary_active_distance  <dbl> 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0~
## $ very_active_minutes         <dbl> 33, 89, 56, 39, 28, 30, 33, 47, 40, 15, 43,~
## $ fairly_active_minutes       <dbl> 12, 17, 5, 20, 28, 13, 12, 21, 11, 30, 18, ~
## $ lightly_active_minutes      <dbl> 205, 274, 268, 224, 243, 223, 239, 200, 244~
## $ sedentary_minutes           <dbl> 804, 588, 605, 1080, 763, 1174, 820, 866, 6~
## $ calories                    <dbl> 1819, 2154, 1944, 1932, 1886, 1820, 1889, 1~
```

```
heartrate <- clean_names(heartrate)
glimpse(heartrate)
```

```
## Rows: 1,154,681
## Columns: 3
## $ id    <dbl> 2022484408, 2022484408, 2022484408, 2022484408, 2022484408, 2022~
## $ time  <chr> "4/1/2016 7:54:00 AM", "4/1/2016 7:54:05 AM", "4/1/2016 7:54:10 ~
## $ value <dbl> 93, 91, 96, 98, 100, 101, 104, 105, 102, 106, 109, 112, 111, 109~
```

```
hourly_steps <- clean_names(hourly_steps)
glimpse(hourly_steps)
```

```
## Rows: 24,084
## Columns: 3
## $ id            <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 15039603~
## $ activity_hour <chr> "3/12/2016 12:00:00 AM", "3/12/2016 1:00:00 AM", "3/12/2~
## $ step_total    <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 8, 551, 1764, 1259, 253, 4470~
```

```
minute_Calories <- clean_names(minute_Calories)
minute_sleep <- clean_names(minute_sleep)
minute_steps <- clean_names(minute_steps)
weight_loginfo <- clean_names(weight_loginfo)
```

Cleaning duplicates Not showed in R because of long loading time and not relevant for hypothesis:

get_dupes(daily_activity) get_dupes(heartrate) get_dupes(hourly_steps) get_dupes(minute_Calories) get_dupes(minute_sleep) get_dupes(weight_loginfo)

minute_sleep <- unique(minute_sleep) get_dupes(minute_sleep)

n_distinct(daily_activityid) $n_distinct(heartrate$id) n_distinct(hourly_stepsid) $n_distinct(minute_Calories$id)

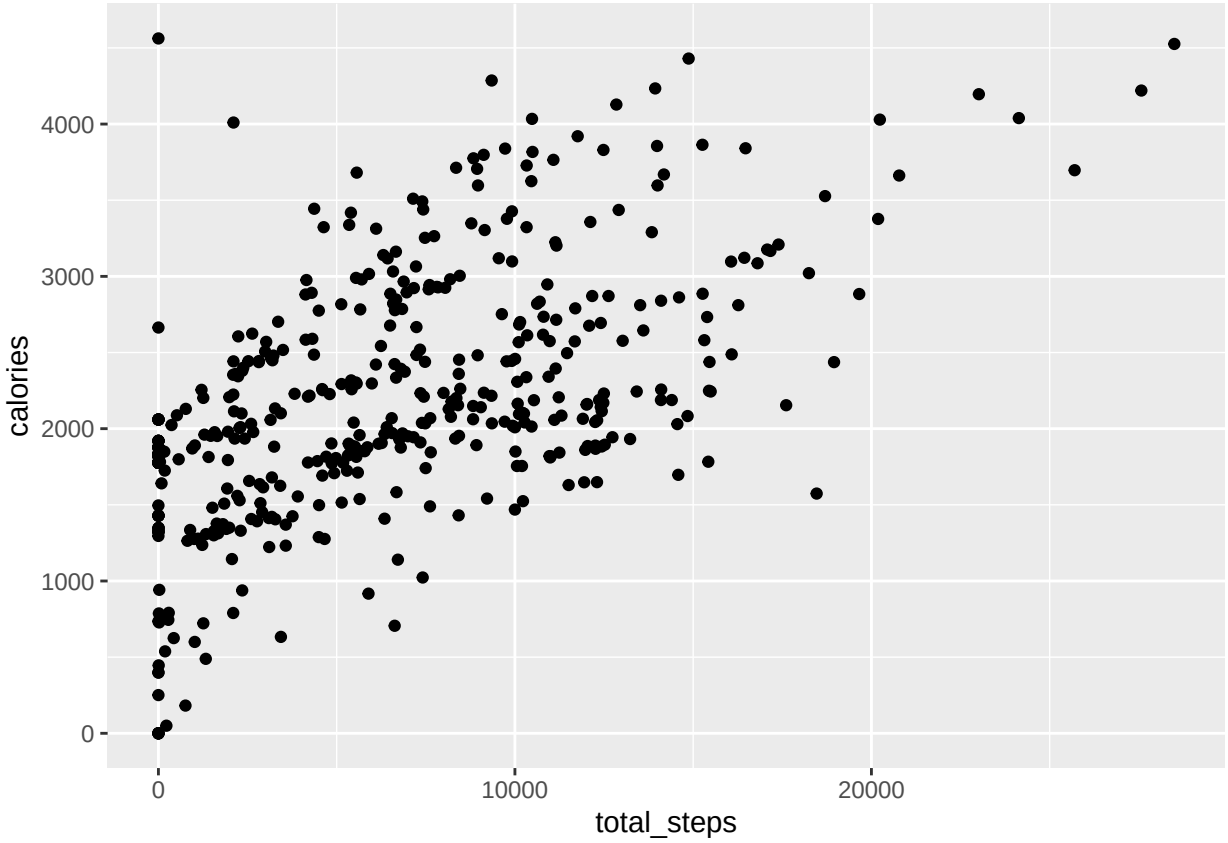
4 Analyze

3 trends were identified:

Trend 1. More steps taken, more calories burned. Trend 2. Less steps than recommended Trend 3. Userstypes

Steps and calories

```
ggplot(data = daily_activity) +
  geom_point(mapping = aes(x = total_steps, y = calories))
```



When comparing total steps with burned calories, you can see that the more steps the customers take, the more calories they burn. This is a trend that can be used in marketing.

Number of steps

```
max(daily_activity$total_steps)
```

```
## [1] 28497
```

```
mean(daily_activity$total_steps)
```

```
## [1] 6546.562
```

```
min(daily_activity$total_steps)
```

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

The data shows that the maximum total number of steps is 28497. The average total steps is 6546 and the least number of total steps is 0. To make sure that the 0 does not mess with the data, in case of error, we filter out the 0:

```
daily_activity_v2 <- daily_activity %>%
  filter(total_steps>0)

min(daily_activity_v2$total_steps)
```

```
## [1] 4
```

```
mean(daily_activity$total_steps)
```

```
## [1] 6546.562
```

We get the same result: the mean of total steps is 6546, that is less than 10 000 recommended. A marketing campagn for increasing the total steps using the app can be used to reach users and new customers.

User types

```
min(weight_loginfo$weight_kg)
```

```
## [1] 53.3
```

```
max(weight_loginfo$weight_kg)
```

```
## [1] 129.6
```

```
min(weight_loginfo$bmi)
```

```
## [1] 21.45
```

```
max(weight_loginfo$bmi)
```

```
## [1] 46.17
```

The customers that are using the fitness app has a weight between 53 and 129 kg, and a BMI between 21 and 46. If underweight is defined as 0-18.4, we can say that the users is either normal of overweight.

Considering that the total number of steps were relative low: 28497, one hypothesis can be that the users are normal people that work out some times.

Act

How can Bellabeat grow and adjust marketing to the growth opportunities?

- Based on the user type trend (3) Bellabeat can grow by focusing on a broader user group: More active users.
- Based on the step trend Bellabeat can set up a trend to get their users and new users to get more total steps. They can used burned calories, visualizations and data in the marketing campagn. This can get excisting users to get more out of the app and get new users interested.

Limitations

There are clear limitation to this analyze, the samle size (around 30) is small, and the data is old - 2016. To get a more reliable data source, you should get updated data from more people.