

Google Data Analytics Case Study : How Can a Wellness Technology Company (Bellabeat) Play It Smart?

sandeep

2024-09-13

Setting up my environment

Notes: Setting up my R environment by loading the needful Packages

```
library(tidyverse)

## — Attaching core tidyverse packages — tidyverse
## 2.0.0 —
## ✓ dplyr      1.1.4      ✓ readr      2.1.5
## ✓ forcats   1.0.0      ✓ stringr    1.5.1
## ✓ ggplot2    3.5.1      ✓ tibble     3.2.1
## ✓ lubridate  1.9.3      ✓ tidyr      1.3.1
## ✓ purrr      1.0.2
## — 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(ggplot2)
library(lubridate)
```

About the Company

Urška Sršen and Sando Mur founded Bellabeat, a high-tech company that manufactures health-focused smart products. Sršen used her background as an artist to develop beautifully designed technology that informs and inspires women around the world. Collecting data on activity, sleep, stress, and reproductive health has allowed Bellabeat to empower women with knowledge about their own health and habits. Since it was founded in 2013, Bellabeat has grown rapidly and quickly positioned itself as a tech-driven wellness company for women.

Business Task Summary (Ask Phase)

Sršen asks you to analyze smart device usage data in order to gain insight into how consumers use non-Bellabeat smart devices. She then wants you to select one Bellabeat product to apply these insights to in your presentation. These questions will guide your analysis: 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?

Prepare Phase

FitBit Fitness Tracker Data (CC0: Public Domain, dataset made available through Mobius): This Kaggle data set contains personal fitness tracker from thirty fitbit users. Thirty eligible Fitbit users consented to the submission of 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.

Process Phase

The analysis of the data will be done with R.

Loading the data

```
daily_step <-  
read_csv("C:/Users/sande/Desktop/first_project/.csv_file/activity in 12 apr  
16 - 11 may1 6/dailySteps.csv")
```

```
## Rows: 940 Columns: 3  
## — Column specification
```

```
## Delimiter: ","  
## dbl (2): Id, StepTotal  
## date (1): ActivityDay  
##  
## 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_activity <-  
read_csv("C:/Users/sande/Desktop/first_project/.csv_file/activity in 12 apr  
16 - 11 may1 6/dailyActivity.csv")
```

```
## Rows: 940 Columns: 15  
## — Column specification
```

```
## Delimiter: ","  
## dbl (14): Id, TotalSteps, TotalDistance, TrackerDistance,  
LoggedActivitiesD...  
## date (1): ActivityDate  
##  
## 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.
```

```
heart_rate <-  
read_csv("C:/Users/sande/Desktop/first_project/.csv_file/activity in 12 apr  
16 - 11 may1 6/heart_rate_seconds.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.

sleep_data <-
read_csv("C:/Users/sande/Desktop/first_project/.csv_file/activity in 12 apr
16 - 11 may1 6/sleepDay.csv")

## Rows: 410 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_data <-
read_csv("C:/Users/sande/Desktop/first_project/.csv_file/activity in 12 apr
16 - 11 may1 6/weightLogInfo.csv")

## Rows: 66 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.
```

The Date column for each dataset is in 'Character', this needs to be change in date format.

```
daily_step$ActivityDay <- ymd(daily_step$ActivityDay)

daily_activity$ActivityDate <- ymd(daily_activity$ActivityDate)

heart_rate$Time <- mdy_hms(heart_rate$Time)

sleep_data$SleepDay <- dmy(sleep_data$SleepDay)
```

```
weight_data$Date <- dmy(weight_data$Date)
```

finding duplicates

Note :- Basic data cleaning has been done in Excel, so there is no duplicate in datasets. The return observations are same as total observations.

```
n_distinct(daily_step$Id, daily_step$ActivityDay)
## [1] 940
n_distinct(daily_activity$Id, daily_activity$ActivityDate)
## [1] 940
n_distinct(heart_rate$Id, heart_rate$Time)
## [1] 2483658
n_distinct(sleep_data$Id, sleep_data$SleepDay)
## [1] 410
n_distinct(weight_data$Id, weight_data$Date)
## [1] 66
```

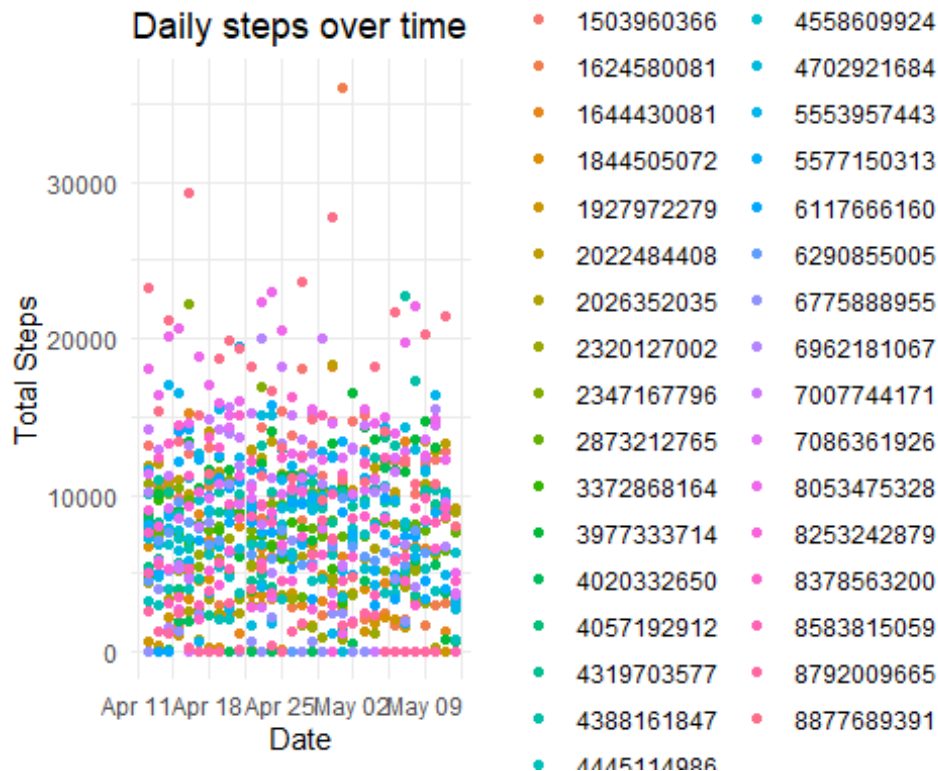
Analyze Phase

Creating Visualizations

Create a plot of daily step over time

Note:- This Plot identifying trends in activity levels over time such as more or less active

```
ggplot(daily_step, aes(x = ActivityDay, y = StepTotal, color =
as.factor(Id))) + geom_point() +
  labs(title = "Daily steps over time"
        , x = "Date",
        y = "Total Steps") +
  theme_minimal()
```



Create bar plots for different activities

Note :- This plot identifying differences in activities.

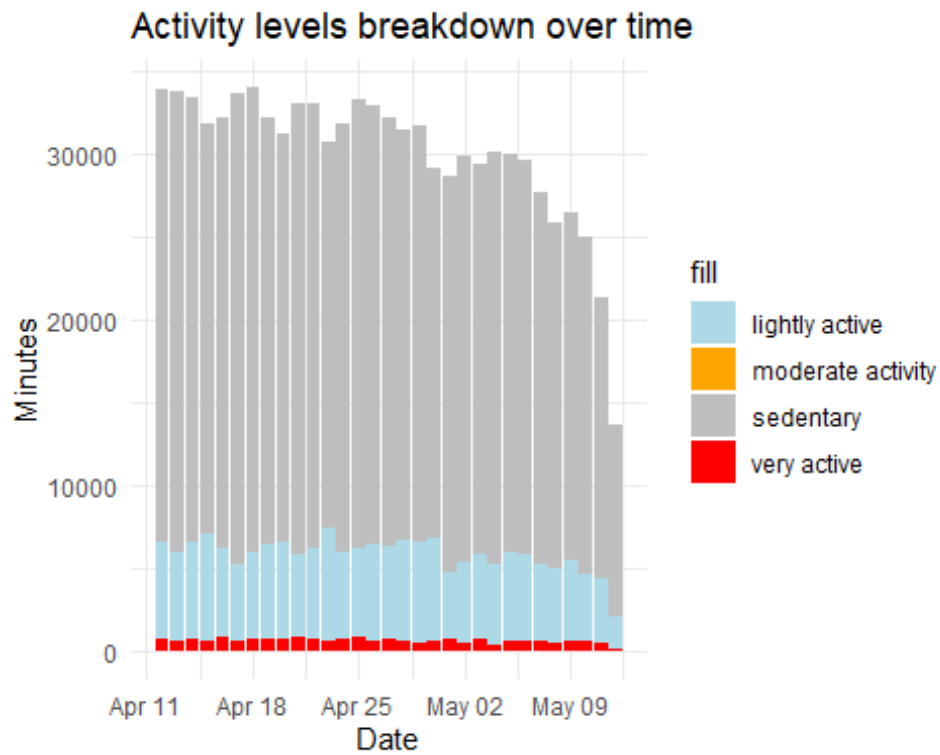
```
ggplot(daily_activity, aes(x= ActivityDate, )) +
  geom_bar(aes(y= SedentaryMinutes, fill = "sedentary"), stat = "identity") +

  geom_bar(aes(y = LightlyActiveMinutes, fill = "lightly active"),stat =
"identity") +

  geom_bar(aes(y = FairlyActiveMinutes, fill = "moderate activity"),stat =
"identity" ) +

  geom_bar(aes(y = VeryActiveMinutes, fill = "very active"),stat ="identity")
+

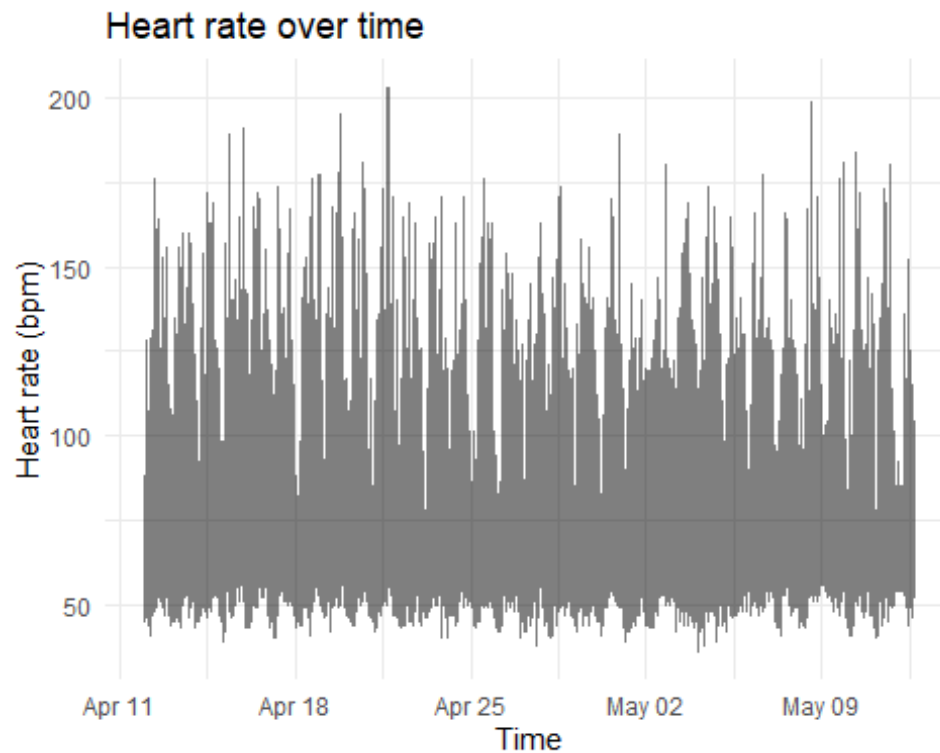
  labs(title = "Activity levels breakdown over time",
    x= "Date",
    y= "Minutes") + theme_minimal() +
  scale_fill_manual(values = c("sedentary" = "gray","lightly active" =
"lightblue",
                                "moderate activity" = "orange", "very active"=
"red"))
```



Create a line plot for heartrate

Note :- This plot identifying the most active time of the day.

```
ggplot(heart_rate, aes(x= Time, y = Value)) +
  geom_line(alpha= 0.5) + labs(title = "Heart rate over time",
                                x= "Time",
                                y= "Heart rate (bpm)") +
  theme_minimal()
```

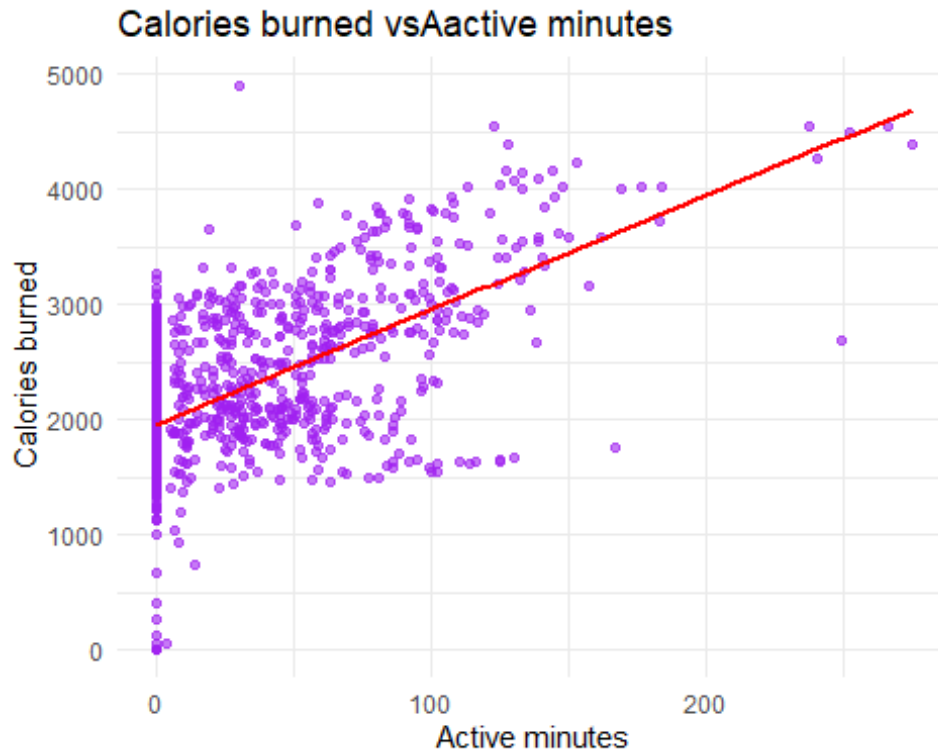


create a plot of calories burned vs active time

Note :- This plot is showing correlation in between activity and calories burn.

```
ggplot(daily_activity, aes(x= VeryActiveMinutes + FairlyActiveMinutes, y=
Calories)) +
  geom_point( alpha = 0.6, color = "purple") +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(title = "Calories burned vsActive minutes",
       x = "Active minutes",
       y = " Calories burned") + theme_minimal()

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



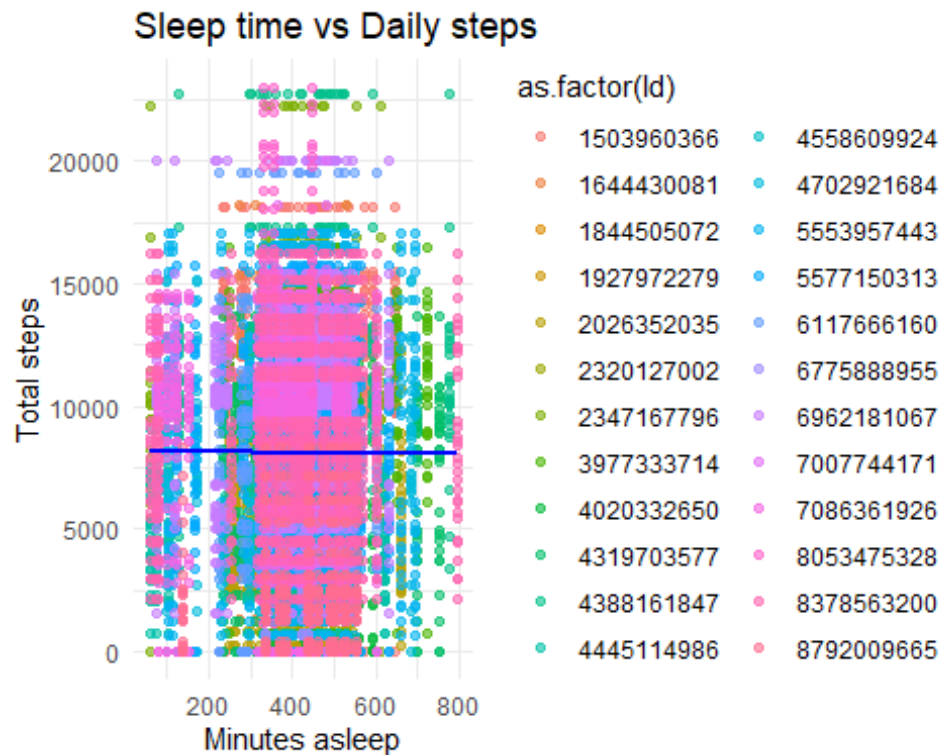
merge daily steps with sleep data to create a plot of sleep time vs daily steps

```
daily_sleep_and_steps <- merge(daily_step, sleep_data, by = "Id")
```

Note :- This plot identifying how sleep quality and duration affecting the activity.

```
ggplot(daily_sleep_and_steps, aes(x = TotalMinutesAsleep, y = StepTotal,
color = as.factor(Id))) +
  geom_point(alpha = 0.6) +
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  labs( title = "Sleep time vs Daily steps",
        x= "Minutes asleep",
        y= "Total steps") +
  theme_minimal()
```

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

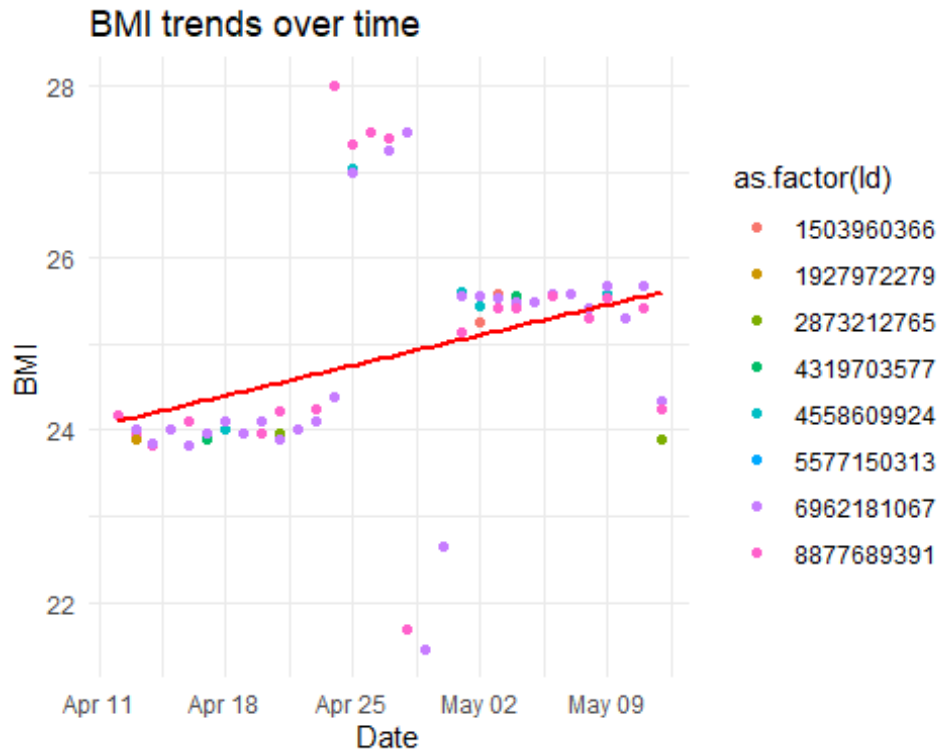



create a plot for BMI over time

Note :- This plot is helpful in maintaining healthy BMI.

```
ggplot(weight_data, aes( x= Date, y = BMI, color = as.factor(Id))) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs( title = "BMI trends over time",
        x= "Date",
        y= "BMI") +
  theme_minimal()

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



Share & Act Phase

There are some interesting insights I found in these visualizations that would be helpful to Bellabeat :-

1. The daily step count data reveals that weekends often show a drop in total steps, indicating reduced physical activity compared to weekdays, So Bellabeat can send targeted notifications or reminders on weekends encouraging customers to maintain their activity levels.
2. There is a positive correlation between very active minutes and calories burned. So, Bellabeat can highlight this correlation in their product marketing, encouraging customers to increase high-intensity activities to improve calorie burning.
3. There is noticeable variability in BMI among customers over time, though not all customers tracking their weight consistently. The data indicates that customers who maintain consistent activity levels and monitor their weight tend to show healthier BMI trends.

Suggestions for Bellabeat's Marketing Strategy

1. Bellabeat can use activity trends to send personalized reminders, especially when customers are less active (ex- on weekends or holidays). For example, push notifications encouraging a short walk and lite exercise during low activity periods.
2. Highlight Bellabeat as a comprehensive wellness platform that tracks not just physical activity but also other health factors like - sleep, heart rate, and weight

management. This could attract to customers looking for an all-in-one solution for their health goals.

Thank You