

# Bellabeat\_Data\_Analysis

Çağkan Özçivit

2024-02-05

## Summary

Conducting an analysis in the R programming language, I've analyzed the realm of Bellabeat, a pioneering health technology company specializing in tailored wellness solutions for women within my Google Data Analytics Certificate program.

The results of my analysis revealed the efficacy of their products in enhancing women's physical activity, optimizing sleep patterns, and managing stress levels.

Through a blend of personalized insights and recommendations, Bellabeat's application empowers users, providing them with the means to proactively manage and elevate their health and well-being. In essence, Bellabeat stands as a compelling illustration of how cutting-edge technology can positively transform and enhance the health and wellness of women.

Let's look it at!

## Data sources

Data Set: [Kaggle](#)

The data set is publicly available on this [Kaggle](#).

## Installing various packages in R

```
install.packages("skimr")
```

```
library(tidyverse)
```

```
library(skimr)
```

```
library(janitor)
```

## Importing .csv dataset files

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

```
calories <- read_csv("dailyCalories_merged.csv")
```

```
heartrate <- read_csv("heartrate_seconds_merged.csv")
intensities <- read_csv("dailyIntensities_merged.csv")
sleep <- read_csv("sleepDay_merged.csv")
steps <- read_csv("dailySteps_merged.csv")
steps_hourly <- read_csv("hourlySteps_merged.csv")
weight <- read_csv("weightLogInfo_merged.csv")
```

## Checking datasets

```
head(activity)
activity <- activity %>% mutate_all(str_trim)
head(sleep)
sleep <- sleep %>% mutate_all(str_trim)
head(calories)
calories <- calories %>% mutate_all(str_trim)
head(intensities)
intensities <- intensities %>% mutate_all(str_trim)
head(steps)
steps <- steps %>% mutate_all(str_trim)
head(weight)
weight <- weight %>% mutate_all(str_trim)
```

## Cleaning and checking duplicated data

```
activity <- distinct(activity)
calories <- distinct(calories)
heartrate <- distinct(heartrate)
intensities <- distinct(intensities)
sleep <- distinct(sleep)
steps <- distinct(steps)
```

```
steps_hourly <- distinct(steps_hourly)
weight <- distinct(weight)
```

## Checking sample size

```
unique_count_activity <- length(unique(activity$id)) print(unique_count_activity)
unique_count_calories <- length(unique(calories$id)) print(unique_count_calories)
unique_count_hearttrate <- length(unique(hearttrate$id)) print(unique_count_hearttrate)
unique_count_intensities <- length(unique(intensities$id)) print(unique_count_intensities)
unique_count_sleep <- length(unique(sleep$id)) print(unique_count_sleep)
unique_count_steps <- length(unique(steps$id)) print(unique_count_steps)
unique_count_steps_hourly <- length(unique(steps_hourly$id))
print(unique_count_steps_hourly)
```

## Checking data again and continue to clean

```
head(activity)
head(intensities)
clean_names(activity)
activity <- activity %>% rename_all(tolower)
head(calories)
clean_names(calories)
calories <- calories %>% rename_all(tolower)
clean_names(intensities)
intensities <- intensities %>% rename_all(tolower)
clean_names(sleep)
sleep <- sleep %>% rename_all(tolower)
clean_names(steps)
steps <- steps %>% rename_all(tolower)
clean_names(steps_hourly)
steps_hourly <- steps_hourly %>% rename_all(tolower)
```

## Adjusting dates in dataset

```
activity <- activity %>% rename(date = activitydate) %>% mutate(id = as.character(id))  
%>% mutate(date = mdy(date))
```

```
calories <- calories %>% rename(date = activityday) %>% mutate(id = as.character(id))  
%>% mutate(date = mdy(date))
```

```
intensities <- intensities %>% rename(date = activityday) %>% mutate(id =  
as.character(id)) %>% mutate(date = mdy(date))
```

```
head(sleep)
```

```
head(steps)
```

```
head(steps_hourly)
```

```
sleep <- sleep %>% rename(date = sleepday) %>% mutate(id = as.character(id))
```

```
sleep <- sleep %>% mutate(date = as.Date(date, format = "%m/%d/%Y"))
```

```
steps <- steps %>% rename(date = activityday) %>% mutate(id = as.character(id)) %>%  
mutate(date = mdy(date))
```

```
steps_hourly <- steps_hourly %>% rename(date_time = activityhour) %>% mutate(id =  
as.character(id)) %>% mutate(date_time = mdy_hms(date_time))
```

```
steps_hourly <- steps_hourly %>% separate(date_time, into = c("date", "time"), sep = " ")  
%>% mutate(date = ymd(date))
```

```
head(activity)
```

```
head(sleep)
```

## Merging data

```
activity_sleep <- merge(activity, sleep, by = c("id", "date"))
```

```
head(activity_sleep)
```

```
calories_intensities <- merge(calories, intensities, by = c("id", "date"))
```

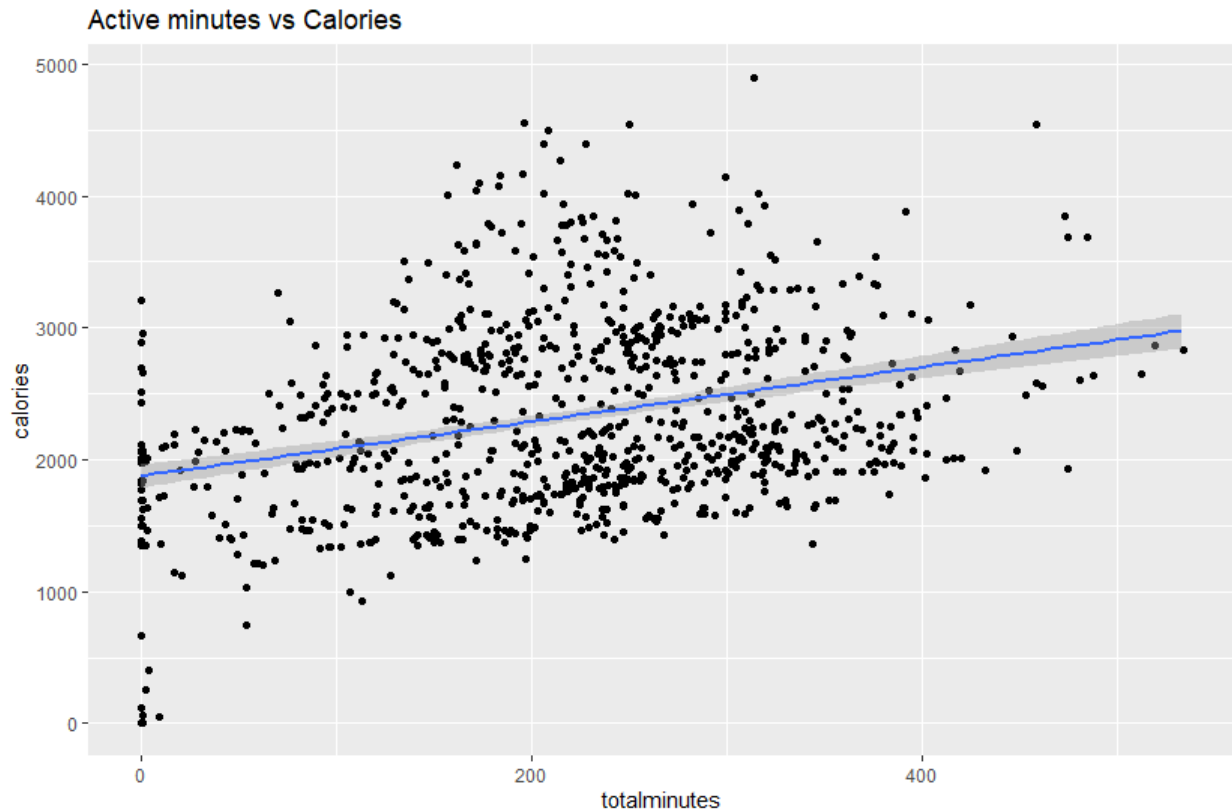
```
head(calories_intensities)
```

```
calories_steps <- merge(calories, steps, by = c("id", "date"))
```

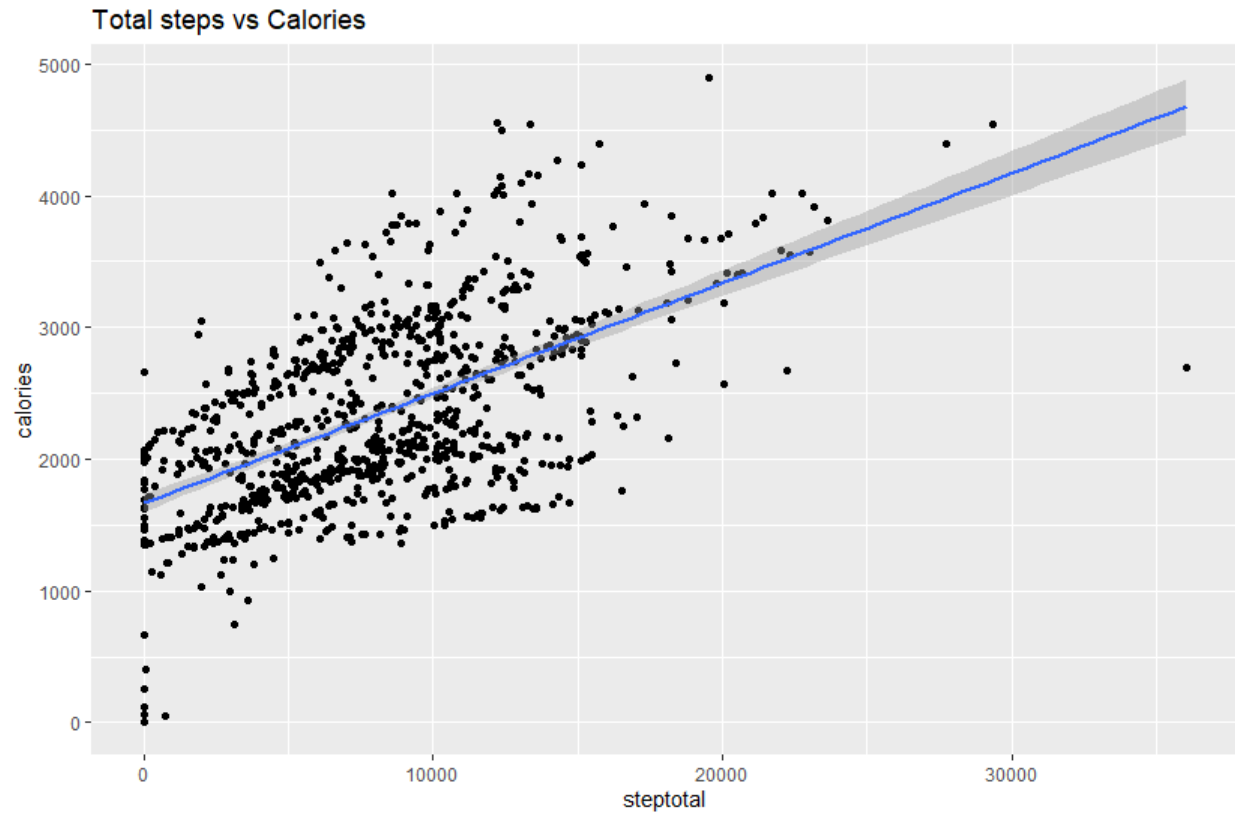
```
head(calories_steps)
```

## Visualization

```
ggplot(data = calories_intensities, mapping = aes(x = totalminutes, y = calories)) +  
geom_jitter() + geom_smooth(method = lm) + labs(title = "Active minutes vs Calories")
```



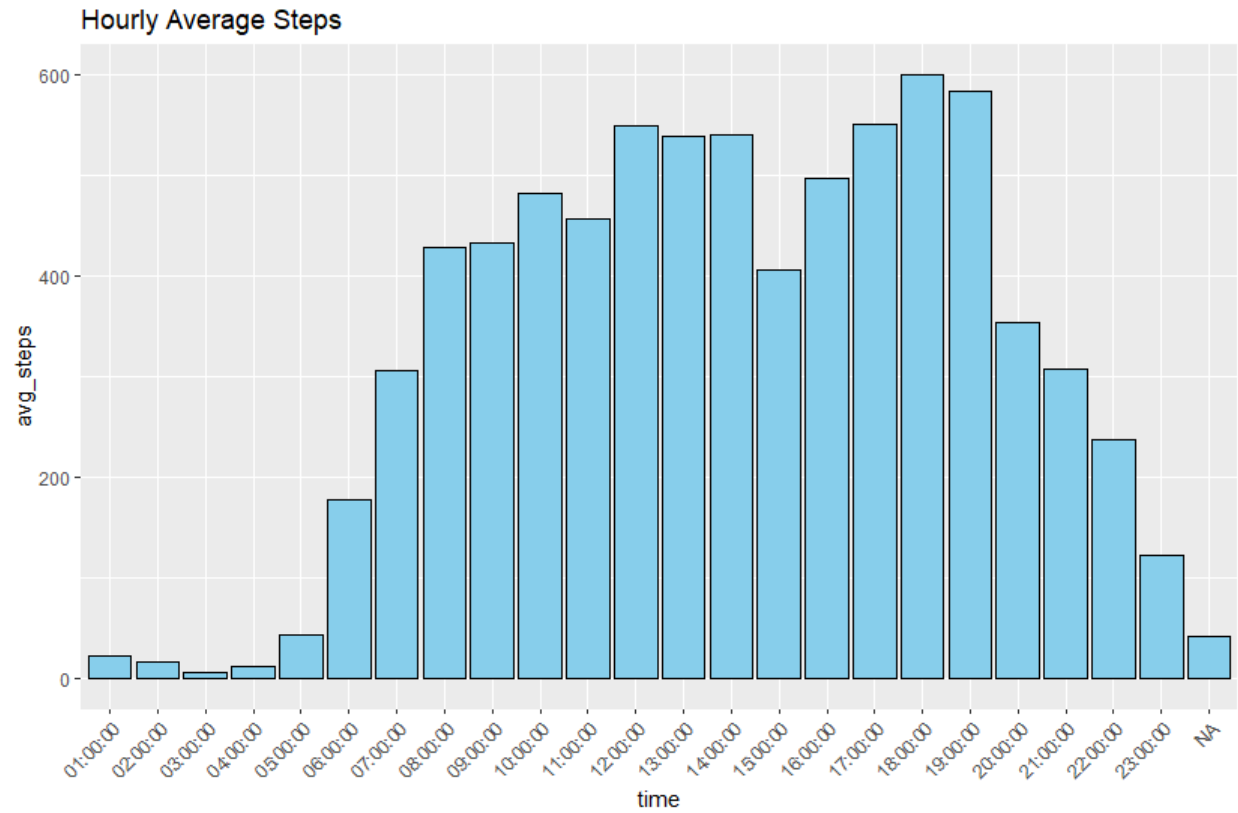
```
ggplot(data = calories_steps, mapping = aes(x = steptotal, y = calories)) + geom_jitter() +  
geom_smooth(method = lm) + labs(title = "Total steps vs Calories")
```



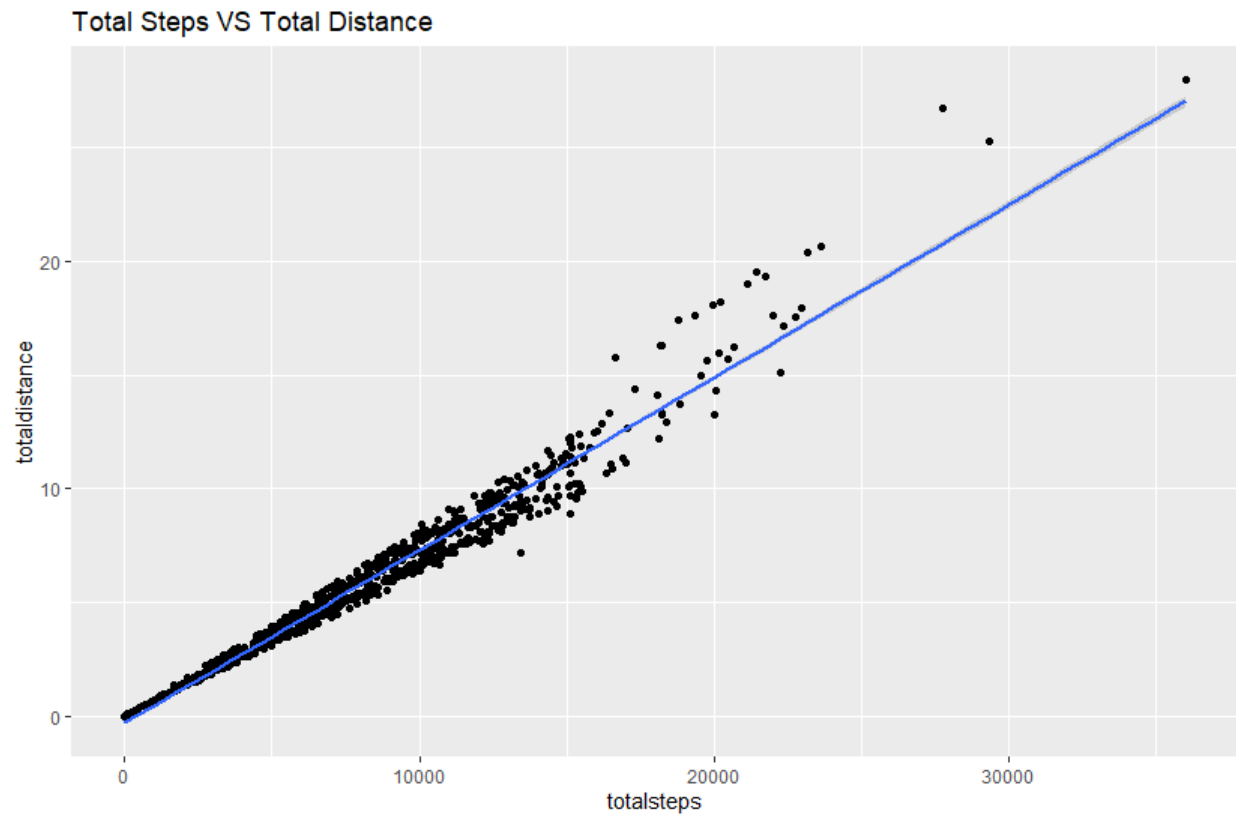
```
head(steps_hourly)
```

```
hourly_steps_plot <- steps_hourly %>% group_by(time) %>% summarise(avg_steps =  
mean(steptotal))
```

```
ggplot(hourly_steps_plot, aes(x = time, y = avg_steps)) + geom_bar(stat = "identity", fill =  
"skyblue", color = "black") + labs(title = "Hourly Average Steps") + theme(axis.text.x =  
element_text(angle = 45, hjust = 1))
```

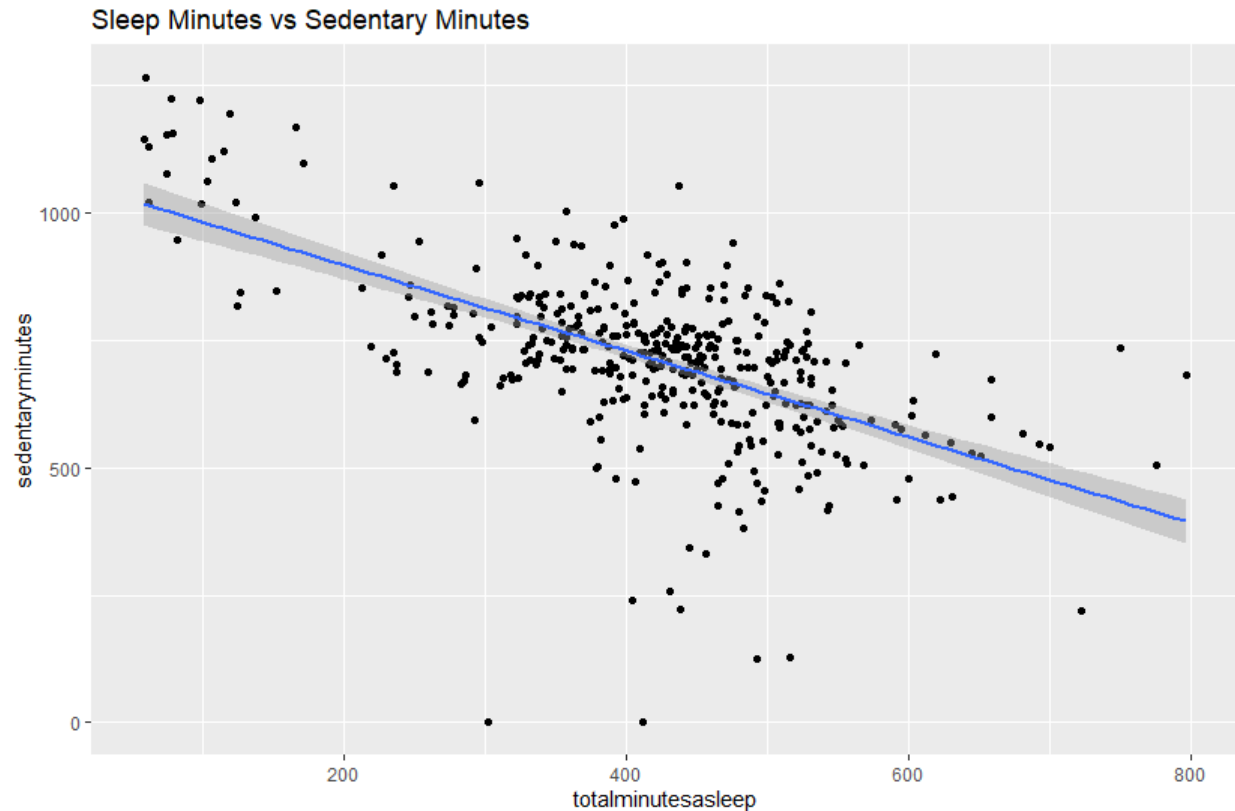


```
ggplot(data = activity, mapping = aes(x = totalsteps, y = totaldistance)) + geom_point() +  
geom_smooth(method = lm) + labs(title = "Total Steps VS Total Distance")
```



```
ggplot(data = activity_sleep, mapping = aes(x = totalminutesasleep, y = sedentaryminutes ))  
+ geom_point() + geom_smooth(method = lm) + labs(title = "Sleep Minutes vs Sedentary  
Minutes")
```





```
activity_weekdays <- activity %>% group_by() %>%
  summarize(mean_total_steps=mean(total_steps),
    mean_total_distance=mean(total_distance), mean_calories=mean(calories))

steps %>% group_by(date) %>% summarise(daily_steps = sum(steptotal)) %>% print()
head(steps)
stepsdate <- as.Date(stepsdate)

daily_steps <- steps %>% group_by(date) %>% summarise(total_steps = sum(steptotal))

monthly_avg_steps <- daily_steps %>% group_by(month = format(date, "%Y-%m")) %>%
  summarise(avg_steps = mean(total_steps/30))

print(monthly_avg_steps)

month avg_steps 1 2016-04 8373.2 2016-05 6686.

So the average steps per month is 7529.
```

## Suggestions

In proposing enhancements for the Bellabeat app, it would be beneficial to diversify the available exercise programs to cater to a broader spectrum of user preferences and fitness levels.

Considering the direct correlation between step count and calorie expenditure, incorporating a feature that sends users notifications encouraging increased walking or running could serve as an effective motivator for a more active lifestyle.

Observing that a significant portion of daily steps is accumulated during lunch and post-office hours, strategically implementing campaigns or notifications within these time frames could be a strategic move. This would capitalize on peak user activity periods, fostering heightened engagement and motivation.

For users aiming to enhance their sleep quality, a valuable recommendation would be to explore methods for reducing sedentary time. This not only aligns with holistic health goals but also complements the app's focus on overall well-being.

A notification can be delivered to the user for walking 10,000 steps per day, as recommended by scientists.

Thanks!