

Case Study:

**How can a wellness company play it
smart?**

Asher Harman

Business Task

Bellabeat has asked us to gather information, analyze data, and apply insights to a product. Our team intends to compare publicly released Fitbit data to the Bellabeat App.

We will use these three questions to guide our analysis of the data for the Bellabeat App:

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's marketing strategy?

The Bellabeat company is expecting the following deliverables inside the produced report:

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 the analysis
5. Supporting visualizations and key findings
6. Top high-level content recommendations based on your analysis

Data Processing

I have created an [R file](#) outlining the processing of different data sets.

First, we must prepare the libraries using the `install.packages` and `library` calls.

```
install.packages("tidyverse")
install.packages("janitor")
install.packages("lubridate")
install.packages("here")
install.packages("skimr")
```

```
library(tidyverse)
library(here)
library(janitor)
library(lubridate)
library(skimr)
```

Next, we needed to set up the datasets. This is done by importing `hourlyCalories_merged`, `hourlyIntensities_merged`, `hourlySteps_merged`, and `DailyActivity_merged` CSV files. To simplify this method, we used `setwd` to the case study file where the datasets are kept.

```
#set the file which will be worked in
setwd("E:/Case Study/CaseStudy2/Unmodified")
#Importing Datasets
hourlyCal <- read_csv("hourlyCalories_merged.csv")
hourlyInt <- read_csv("hourlyIntensities_merged.csv")
hourlyStep <- read_csv("hourlySteps_merged.csv")
dailyAct <- read_csv("dailyActivity_merged.csv")
```

To make managing the hourly datasets easier, we will merge hourlyCal, hourlyInt, and hourlyStep.

```
#Merge all three hourly factors into a single file.
hourlyCalInt <- merge(hourlyCal, hourlyInt, by = c("Id","ActivityHour"))
hourlyFull <- merge(hourlyCalInt, hourlyStep, by = c("Id","ActivityHour"))
```

Then, we want to check how many unique people are identified by id in the datasets.

```
#Checking the number of unique participant
n_distinct(dailyAct$Id)
n_distinct(hourlyFull$Id)
```

Next, we must clean the columns and standardize the naming schemes. This will make the datasets easier to work with and understand.

```
#Cleaning cols names and making all of them standardized
clean_names(dailyAct)
dailyAct <- rename_with(dailyAct,tolower)

clean_names(hourlyFull)
hourlyFull <- rename_with(hourlyFull,tolower)
```

We need to check for duplicated rows with all the datasets standardized right now.

```
#Checking duplicates
sum(duplicated(dailyAct))
sum(duplicated(hourlyFull))
```

Now that we can be sure there are no duplicate rows, we will make the dates easier to read. While at it, we will add a column for dailyAct based on the day of the week.

```
#Converting date & time format
dailyAct$activitydate <- mdy(dailyAct$activitydate)
hourlyFull$activityhour <- mdy_hms(hourlyFull$activityhour)

#Adding a day_of_week col to daily_activity
dailyAct$day_of_week <- wday(dailyAct$activitydate)
```

For dailyAct, we lack the total number of active hours. To do this, we add all the active minutes together, divide it by 60, and round the total_active_hours.

```
#Adding a total_active_hours column to daily_activity
dailyAct$total_active_hours =
  (dailyAct$fairlyactiveminutes
   + dailyAct$lightlyactiveminutes + dailyAct$sedentaryminutes + dailyAct$veryactiveminutes)/60
dailyAct$total_active_hours <- round(dailyAct$total_active_hours,2)
```

We will exchange each number for daily activities to make things a bit more readable for weekdays.

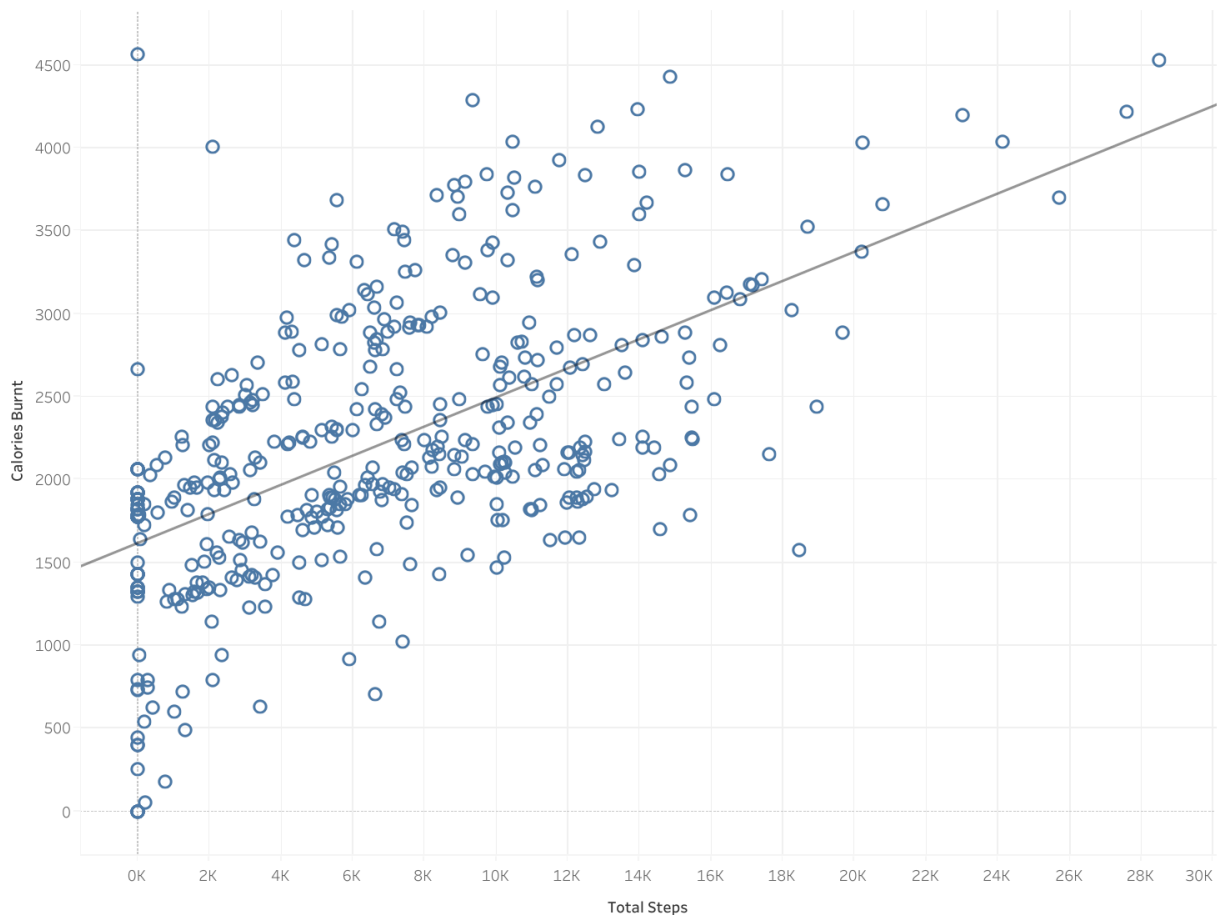
```
# Converting number of the day_of_week column to name of the day
dailyAct <- dailyAct %>%
  mutate(day_of_week = recode(day_of_week
    , "1" = "Sunday"
    , "2" = "Monday"
    , "3" = "Tuesday"
    , "4" = "Wednesday"
    , "5" = "Thursday"
    , "6" = "Friday"
    , "7" = "Saturday"))
```

The last step is to use write to put down both datasets and download them into the preferred file.

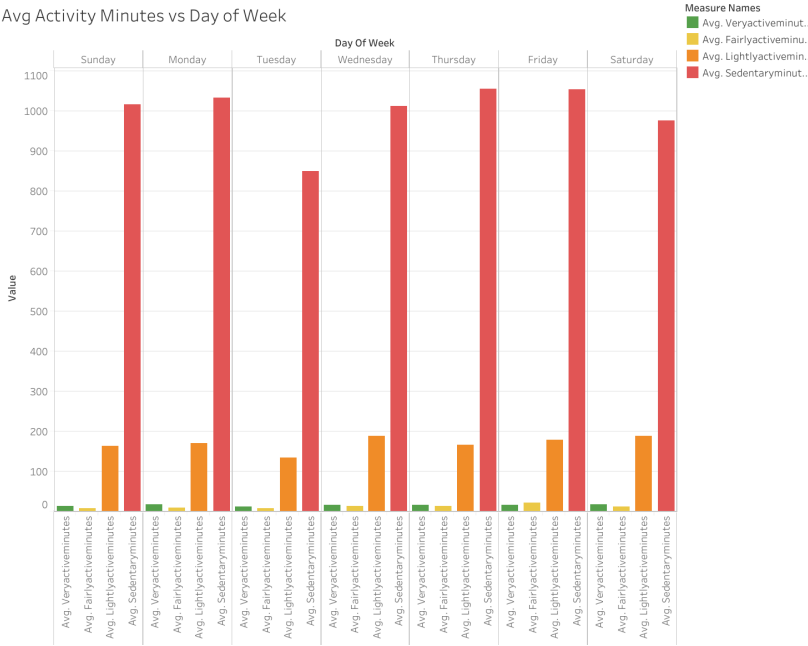
Analysis

I wanted to confirm the suspected suspicion of the amount of calories burnt increased with more steps someone took. The general trend is that the more steps, the more calories are burnt.

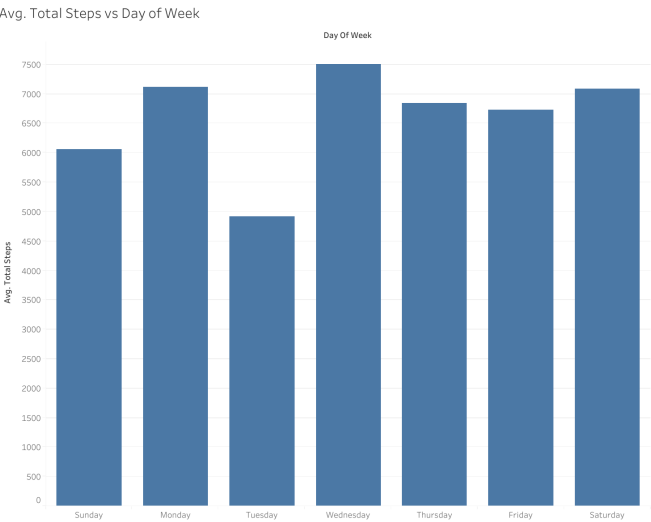
Total Steps Vs Calories Burnt



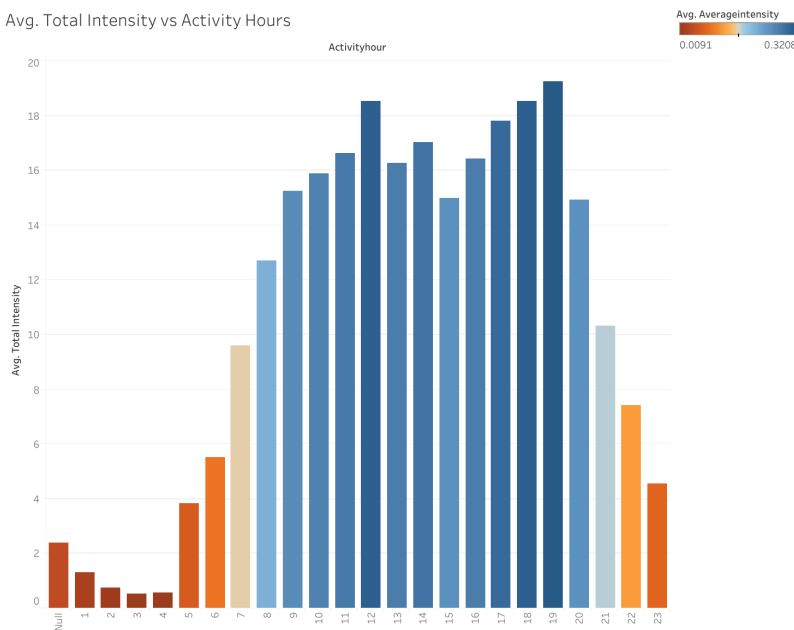
The following information is whether the day of the week impacts a person’s activity level. A quick scan shows that most people are generally more inactive than active. However, there appears to be a trend in which activity increases typically on weekends and in the middle of the week.



To confirm the previous trend, we look at the average total steps per day of the week. This chart has a similar appearance, though it provides a different story. We can infer from our first chart that the number of steps per day is roughly equivalent to the number of steps taken in terms of steps to lost rate. While sedentary minutes are less on Tuesdays, so are the average number of steps. At the same time, the steps on Wednesday appear to have the highest average. The highest average of total steps is on Wednesday, followed by Saturday, with a relative short drop off on Sunday. This could mean that people are using Walks to deal with the stress of the work week or travel from their place of employment to home or vice versa.



The next step to understanding how best to help users is to know how much time people spend doing high-intensity activities. The two primary clusters are about 12 and 7 p.m., with the higher average being focused from 5 to 7 p.m. This could be looked at because it's about when everyone is off work, so they have time to move around and exercise. The spike of about 12 could have something to do with the lunch breaks of the average 7-to-5 worker.



Recommendations

1. My first suggestion is to help set reminders and notifications hourly to increase each user's total average activity, especially if they still need to reach their stated or assigned daily goals.
2. My next suggestion is to have reminders during more active periods of the day. The average user is more willing to commit time to physical activity.
3. To help build better habits, we should investigate the potential of encouraging daily interaction with the Bellabeat app. To do this, it is advisable to examine how other mobile games draw in users and keep them for a sustained time.
4. My last suggestion is introducing challenges that could be a powerful motivational tool. These challenges and positive feedback based on the user's activity could significantly boost user engagement. Whether through daily reminders or weekly activity summaries, this feature could be a game-changer.