Bellabeat Project Report

Abdurrahman Tantawi

2023-04-13

R Markdown File

How Can a Wellness Technology Company Play It Smart?

Business: Bellabeat

Google Data Analytics Capstone Project

Summary

People don't show interest in weight and sleeping recording. In addition, they prefer to walk at noon and early evening. Also, many of them lack consistency. Bellabeat must develop a rewarding program to enhance the users' performance using data and technology.

Business Task

- How do consumers use non-Bellabeat smart devices?
- What are the patterns or trends of usage?
- How can this be applied to enhance Bellabeat's marketing strategy and attract more consumers?
- How can Bellabeat attract new customers by analyzing the smart devices' usage patterns?

Stakeholders

Primary Stakeholders

- Bellabeat Marketing Analytics Team
- Ms. Urška Sršen (Cofounder & Chief Creative Officer of Bellabeat)

Secondary Stakeholders

- Mr. Sando Mur (Mathematician – Cofounder & CEO at Bellabeat)

Exploring Data

The datasets contain information about the number of daily steps, the sleeping times, the calories burned, and the activity rate. We'll analyze the patterns of usage through time, consistency through date streaks & average daily user activity to gain insights about their preferences that will help Bellabeat approach better to potential customers. We'll deal with another dataset as the one under our hands has some limitations that will be mentioned in the upcoming section.

Preparing Data

The first dataset was obtained from Kaggle, discussing people log activities for other non-Bellabeat smart health devices. However, it's not completely ROCCC. It's collected in the time span of one month, from almost mid-April to mid-May 2016, and it's worth mentioning that we are now in 2023, we have better health technologies. In addition, people's lifestyle has significantly changed due to the Covid-19 pandemic. We'll consider this dataset as well as another, more "ROCCC" dataset.

Edit: We'll use our first dataset supplement to be: Fitness tracker data (2016 - present) [2450+days] collected by Damir Gadylyaev on Kaggle. This dataset lists the daily steps & sleep tracking for mi band from 2016 till January 2023 for one user (2450+ days), living in Europe, which is more "ROCCC". The dataset is CC0 licensed (Public Domain use). Edit 2: we'll refer to old dataset as D1, and the new dataset as D1_S Since we are dealing with relatively large datasets. I'll use Excel for cleaning & exploring, R for analyzing data, visualization & documentation. D1 contains 18 files, including hourly & minutely recording of data. However, we believe dealing with daily records will help us obtain actionable overall insights & better analysis. While D1_S contains only 2 files, both are recorded daily.

D1 files are long from the first glance, since ID is a primary key representing the user, and it's repeating by day. Similarly, D1_S is collected for one user. Thus, it's definitely long dataset. After having an overall look on both datasets, we'll consider only 4 files in D1 out of total 18 files, files are "dailyActivity_merged.csv", "hourySteps_merged.csv", "weightLogInfo_merged.csv" & "sleepDay_merged.csv" and both files from D1_S: "01_Steps.csv" & "01_Sleep.csv".

Processing Data

The first challenge to understand our dataset is the units of D1, especially distance is it in km or miles? No clue is given about this. After a long search and asking the Kaggle community, it's in km.

We will first clean our data using Excel, firstly by making sure every cell is in its correct format. In both datasets, dates are date, distances are in numbers (rounded to 3 decimal places), while BMI (Body Mass Index) is rounded to only 2 places. Finally, minutes, steps & calories are integer numbers. We changed all fields names in all working files to match camelCase convention for more consistency, we also change the type of "id" field into text in all tables as it won't be involved in any calculations or analyses.

In "dailyActivity_merged.csv" file of D1, we observed some dates were in the format of mm/dd/yyyy (right aligned) & some others were in the format of dd/mm/yyyy (left aligned). This was observed by trying to change the data type from general to date for the whole field. However, they look the same from the first glance (prefer to figure 1). We cleaned the dates and unified the format to dd/mm/yyyy. We observed the field "loggedActivitiesDistance" is almost full of zeros, after filtering, it seems not. We'll keep that in mind for analysis. We also found the "totalDistance" & "trackerDistance" fields are almost equal, we set a filter to check that and understand what it indicates. Finally, we saved a copy of the table in a new name "activity.csv" for ease.

In "hourlySteps_merged,csv" file, we split the date time into a date and a time fields and in the same time kept the original field for reference, we renamed the original field "activityDateHour" & the splitted fields "activityDate" & "activityHour" respectively. We also assign the data types. Fortunately, all the data was clean in this file. We saved a copy from the original file and renamed it "activityH.csv" for ease.

Similarly, in "sleepDay_merged.csv", we split date and time into 2 fields, everything seems clean and acceptable. We saved a copy of the table then renamed it as "sleep.csv".

In the "weightLogInfo_merged.csv" file, we split date and time again. We also have a column named "Fat" and it has lots of null values (66 null values out of 68 total values), we kept it, but we will not rely on it in our analysis. We found a repeated record for the id "1503960366" in the file. Finally, we renamed our copied table as "weight.csv".

In "01_Steps.csv" file in D1_S, distance is measured in a different way, for example: in D1, 13162 steps are equivalent to 8.5 km. While in D1_S, 13164 steps are equivalent to 10246, so most probably it's in meters

or yards. Since D1_S owner lives in Europe, where they follow SI Unit System. The chances are higher it's in meters. Thus, we'll assume it in meters and unify it to kilometers in both datasets. We added two new columns "distanceKm" & "runDistanceKm". We renamed our file "uSteps.csv".

The last file we have is "02_Sleep.csv" in D1_S, we observed the difference between start and stop times isn't equal to the total number of sleeping minutes recorded. Which will be kept in consideration in analysis. We renamed this file to "uSleep.csv". Also, we observed the start and end of sleeping dates' formatting changed fluctuatingly. Each six months approximately the format was changing from a uniform date to a Unix timestamp (Unix timestamp is the total number of seconds starting from January 1, 1970, until a particular moment) alternately. In order to convert it we have to divide the Unix timestamp and convert it into days, months and years and then add it to the start point (which is January 1, 1970). In addition, there are too many null Unix time stamp values. So, we cleaned the dates and converted them into the original uniform format using the equation:

$$\mathrm{D} = \frac{U}{60 \times 60 \times 24} + 01/01/1970$$

After that, we stored the new dates in separate columns. Another observed issue in the same fields is even the formatted dates have an appended "+0000" in the end of the date. Which is actually preventing Excel from recognizing the fields as a date-time type. We trimmed this part using the Excel "RIGHT" function.

Again, we observed many records with null date values, so we deleted them Finally, we renamed our work file copy "uSleep.csv". Finally, we'll name our project "Bellabeat_R". We were considering doing the analysis on SQL BigQuery, but after creating the queries, it seems Google Cloud is laggy. Thus, we'll analyze data on RStudio instead.

Data Analysis

This is a deep analysis for the data, if you're not interested, go to the next section "Data Visualization & Key Findings"

Initialization We first prepared our setup in Rstudio, this was done by installing and loading the packages "dyplyr", "readr", "ggplot2", "ggsci", and only loading "lubridate" package.

```
options(repos = "https://cran.rstudio.com/")
install.packages("dplyr")
install.packages("readr")
install.packages("ggplot2")
install.packages("ggsci")
install.packages("tinytex")
library(dplyr)
library(readr)
library(lubridate)
library(ggplot2)
library(ggsci)
library(knitr)
tinytex::install_tinytex(force = TRUE)
```

After that, we loaded the workfiles after processeing

```
activity <- read.csv("A:/My_Career/DataAnalytics/CaseStudy2/AfterAnalysis/3 - CleanWorkfiles/activity.cs
activityH <- read.csv("A:/My_Career/DataAnalytics/CaseStudy2/AfterAnalysis/3 - CleanWorkfiles/activityH
sleep <- read.csv("A:/My_Career/DataAnalytics/CaseStudy2/AfterAnalysis/3 - CleanWorkfiles/sleep.csv")
```

```
weight <- read.csv("A:/My_Career/DataAnalytics/CaseStudy2/AfterAnalysis/3 - CleanWorkfiles/weight.csv")
uSteps <- read.csv("A:/My_Career/DataAnalytics/CaseStudy2/AfterAnalysis/3 - CleanWorkfiles/Supplement/u
uSleep <- read.csv("A:/My_Career/DataAnalytics/CaseStudy2/AfterAnalysis/3 - CleanWorkfiles/Supplement/u
```

Then, we started a basic analysis by counting participants of D1 categories (activity, sleep & weight).

- How many people recorded their activity?
- How many people recorded their sleep?
- How many people recorded their weight?

```
num_activ <- n_distinct(activity$id) # 33 people
num_sleep <- n_distinct(sleep$id) # 24 people
num_wei <- n_distinct(weight$id) # 8 People
```

Then, we went a step further by counting participants in 2 or more categories. Since each participant has a unique if, we'll do our research based on id value.

- How many people recorded their activity & weight?
- How many people recorded their activity & sleep?
- How many people recorded their sleep & weight?
- How many people recorded their activity, sleep & weight?

```
steps_wei <- inner_join(activity, weight, by = "id")
num_activ_wei <- n_distinct(steps_wei$id) # 8 people recorded both activity & weight

steps_sleep <- inner_join(activity, sleep, by = "id")
num_activ_sleep <- n_distinct(steps_sleep$id) # 24 people recorded both activity & sleep

wei_sleep <- inner_join(weight, sleep, by = "id")
num_wei_sleep <- n_distinct(wei_sleep$id) #6 people recorded both their weight & sleep

activ_wei_sleep <- inner_join(activity,weight,sleep, by = "id")
num_activ_wei_sleep <- n_distinct(activ_wei_sleep$id) #8 people recorded the 3 categories
```

Weight-related Analysis Then, we proceeded with the weight-related calculations

First, calculate the average weight loss per day for each user. TO do this, we need to count the number of days for each user.

```
id_dates <- weight %>%
  group_by(id) %>%
  summarize(first_date = min(date), last_date = max(date))

# Join the weights table with the id_dates table to get the weight at the first and last dates for each
weight_first_last <- weight %>%
  inner_join(id_dates, by = "id") %>%
  filter(date %in% c(first_date, last_date))
```

After that, calculate the weight difference in the first & last date recorded by user (total weight loss), find the average weight loss, which is the total weight loss / number of days.

```
weight_pivot <- weight_first_last %>%
  group_by(id) %>%
  summarize(num_days = as.numeric(last_date - first_date + 1),
            tot_weight_lossKg = first(weightKg) - last(weightKg),
            avg_weight_lossKg = (first(weightKg) - last(weightKg)) / num_days,
            bmi_first = first(bmi),
            bmi_last = last(bmi))
```

In this calculation, we didn't evaluate the standard deviation & coefficient of variation for users (we will do this for upcoming calculations). Why?

- Because this is not a measure for inconsistency, why again? Because our bodies & genetics can challenge us in the journey of weight loss. Thus, using metrics here is not "In my opinion" a good or realistic practice.

we will store our calculations in a pivot table called "weight_pivot"

```
weight_pivot <- unique(weight_pivot)
kable(weight_pivot)
```

id	num_days	tot_weight_lossKg	avg_weight_lossKg	bmi_first	bmi_last
1503960366	1	0.0	0.0000000	22.65	22.65
1927972279	1	0.0	0.0000000	47.54	47.54
2873212765	22	-0.6	-0.0272727	21.45	21.69
4319703577	18	0.1	0.0055556	27.45	27.38
4558609924	22	0.6	0.0272727	27.25	27.00
5577150313	1	0.0	0.0000000	28.00	28.00
6962181067	31	0.6	0.0193548	24.39	24.17
8877689391	31	1.8	0.0580645	25.68	25.14

We got another important question, when do people weigh themselves? To answer this question, we will make a frequency tables of how many times people recorded their weight at a specific hour. We will print the result in a frequency table with number of repetitions & time recorded.

```
freqtable <- weight %>%
  group_by(time) %>%
  summarize(frequency = n())
kable(freqtable)
```

time	frequency
01:08:00	1
06:39:00	2
06:40:00	1
06:42:00	1
06:43:00	1
06:44:00	1
06:47:00	1
06:48:00	2

time	frequency
06:49:00	2
06:50:00	3
06:51:00	3
06:55:00	1
07:22:00	1
07:35:00	1
07:38:00	1
07:49:00	1
08:47:00	1
09:17:00	1
13:39:00	1
23:59:00	40

Activity-related Analysis First, we will calculate total & average distance walked by each user in D1. We will focus more on distance than steps as steps are very subjective metric and dependent of many factors. We will calculate the standard deviation for each user.

The closer the standard deviation to zero, the closer the distances from the mean (average), which indicates they are close from each other as well

Another factor we will consider is the coefficient of variation, which is simply the percentage conversion of the standard deviation. In another words, if the mean is 1 (100%), coefficient of variation is the average closeness to the mean of the all values. We will refer to it as "inconsistency" & "normalized standard deviation" in our analysis. Last, We will store our values in a new pivot table named "distance_pivot".

```
distance_pivot <- activity %>%
  group_by(id) %>%
  summarize(
    tot_distance = sum(totalDistance),
    activDays = n_distinct(activityDate),
    avg_distance = mean(totalDistance),
    distanceSD = sd(totalDistance),
    normDistanceSD = distanceSD/avg_distance)
```

Then, we'll calculate the total distance walked each hour by all users on all days

```
activH_pivot <- activityH %>%
  group_by(activityHour) %>%
  summarize(tot_stepH = sum(stepTotal))
```

Last, we will calculate the average km distance for the supplement data collected by user. The same data as the other users. But we'll group data by weeks instead.

```
uSteps_pivot <- uSteps %>%
  mutate(week = floor_date(date, unit = "week")) %>%
  group_by(week) %>%
  summarize(avg_weekDistKm = mean(distanceKm),
            weekDistKmSD = sd(distanceKm),
            normWeekDistSD = weekDistKmSD/avg_weekDistKm)
```

Sleep-related Analysis Similarly, we'll calculate the average, standard deviation, coefficient of variation of the time D1 users slept. We'll calculate an additional parameter which what we call "sleep-bed ratio". Again, the standard deviation and average together are an indication of how consistent and good are the users' sleeping habits.

```
sleep_pivot <- sleep %>%
  group_by(id) %>%
  summarize(avg_sleep = mean(totalMinutesAsleep),
            avg_inBed = mean(totalTimeInBed),
            sleepSD = sd(totalMinutesAsleep),
            norm_sleepSD = sleepSD/avg_sleep,
            avg_sleepQ = avg_sleep/avg_inBed)
```

We'll do the same with the supplement data of the user in D1_S, analyzing his sleeping habits. And same as before, we'll group the data by week.

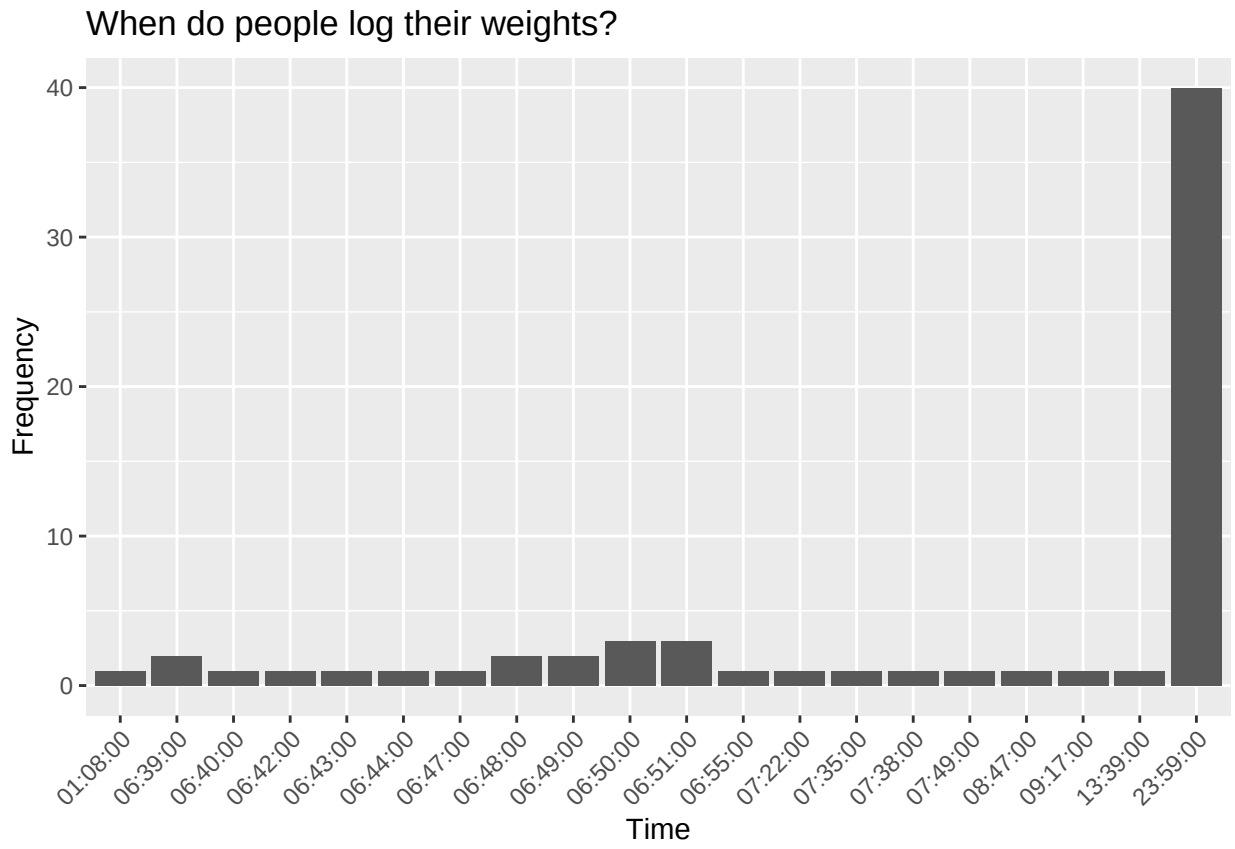
```
uSleep$start <- as.POSIXct(uSleep$start, format = "%Y-%m-%d %H:%M:%S")
uSleep$stop = as.POSIXct(uSleep$stop, format = "%Y-%m-%d %H:%M:%S")
uSleep$recordedSleepTimeU = difftime(uSleep$stop, uSleep$start, units = "mins")

uSleep_pivot <- uSleep %>%
  mutate(week = floor_date(start, unit = "week")) %>%
  group_by(week) %>%
  summarize(avgWeeklySleep = mean(deepSleepTime+shallowSleepTime),
            avgInBed = mean(deepSleepTime+shallowSleepTime+wakeTime),
            uSleepSD = sd(deepSleepTime+shallowSleepTime),
            unorm_sleepSD = uSleepSD/avgWeeklySleep,
            avg_uSleepAcc = avgWeeklySleep/mean(as.numeric(recordedSleepTimeU)),
            avg_uSleepQ = avgWeeklySleep/avgInBed)
```

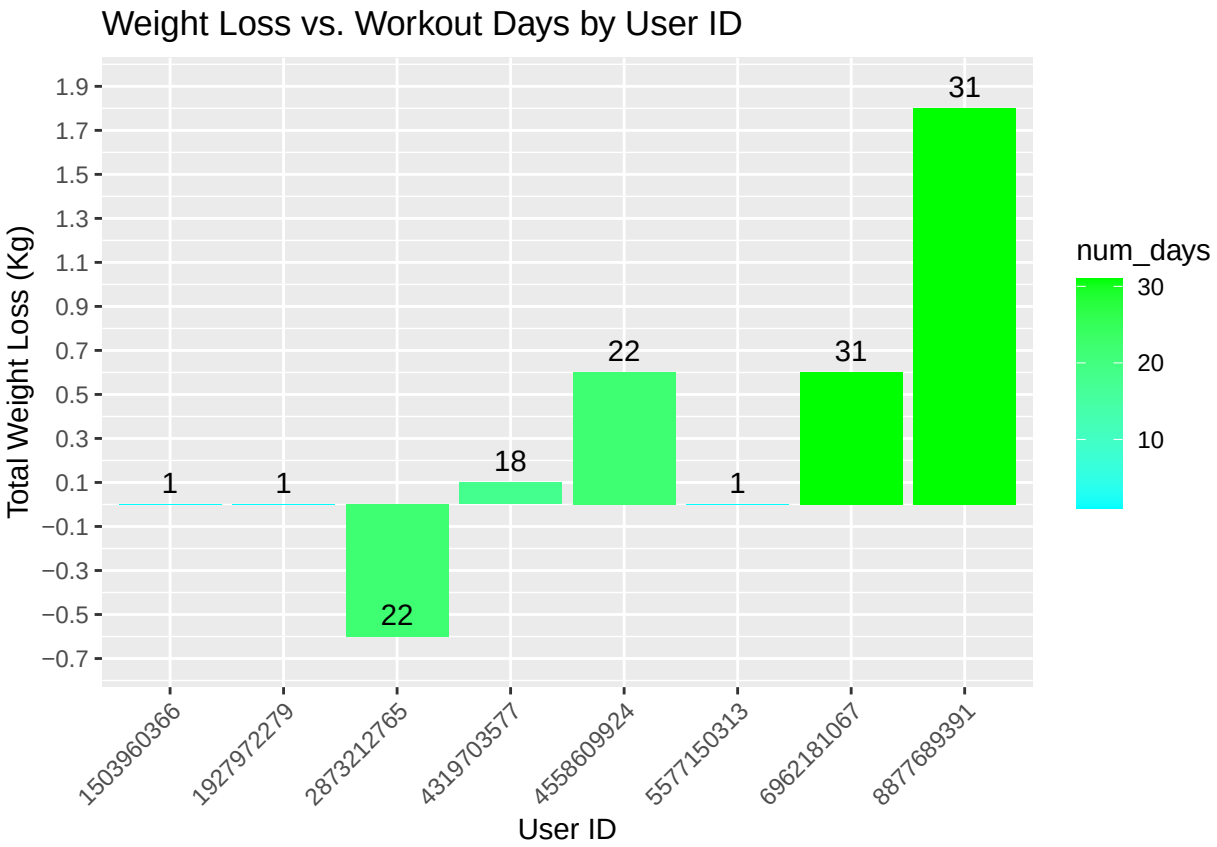
By This, all needed analysis is ready

Data Visualization & Key Findings

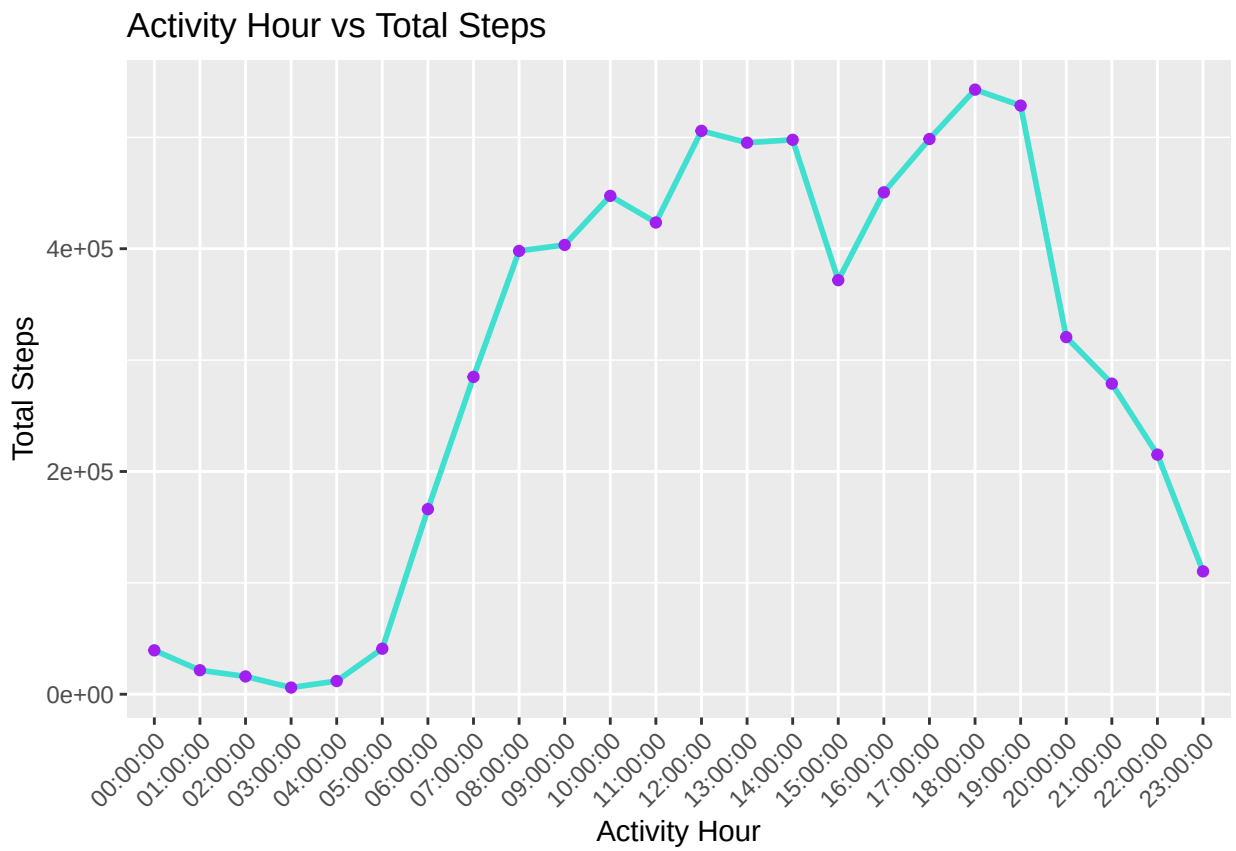
Data Visualization for The Main Dataset (D1) We discovered that almost 39% of the weight recordings were between 6:30 AM & 8:00 AM, while 60.6 of the weights were recorded at 11:59 PM.



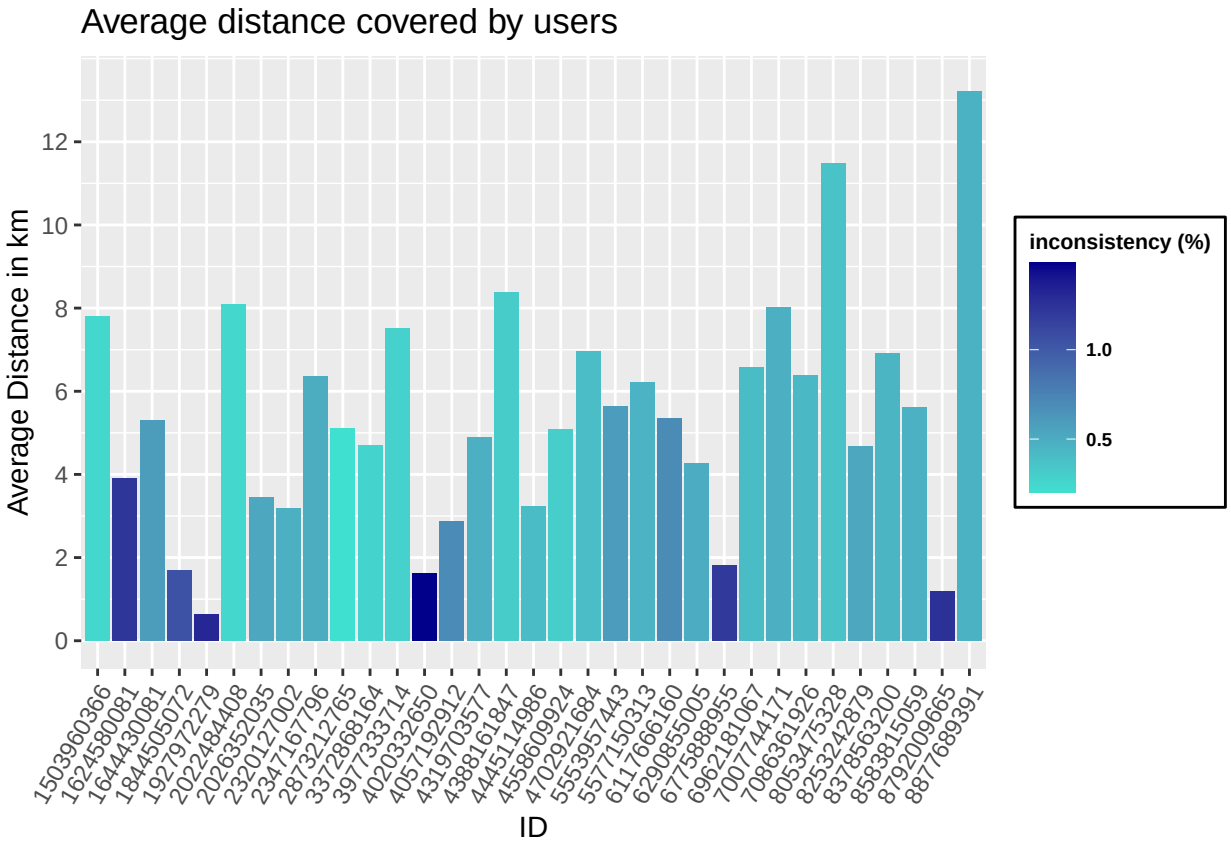
We realized that user "2873212765" has a negative weight loss, which means he gained weight. Interesting!



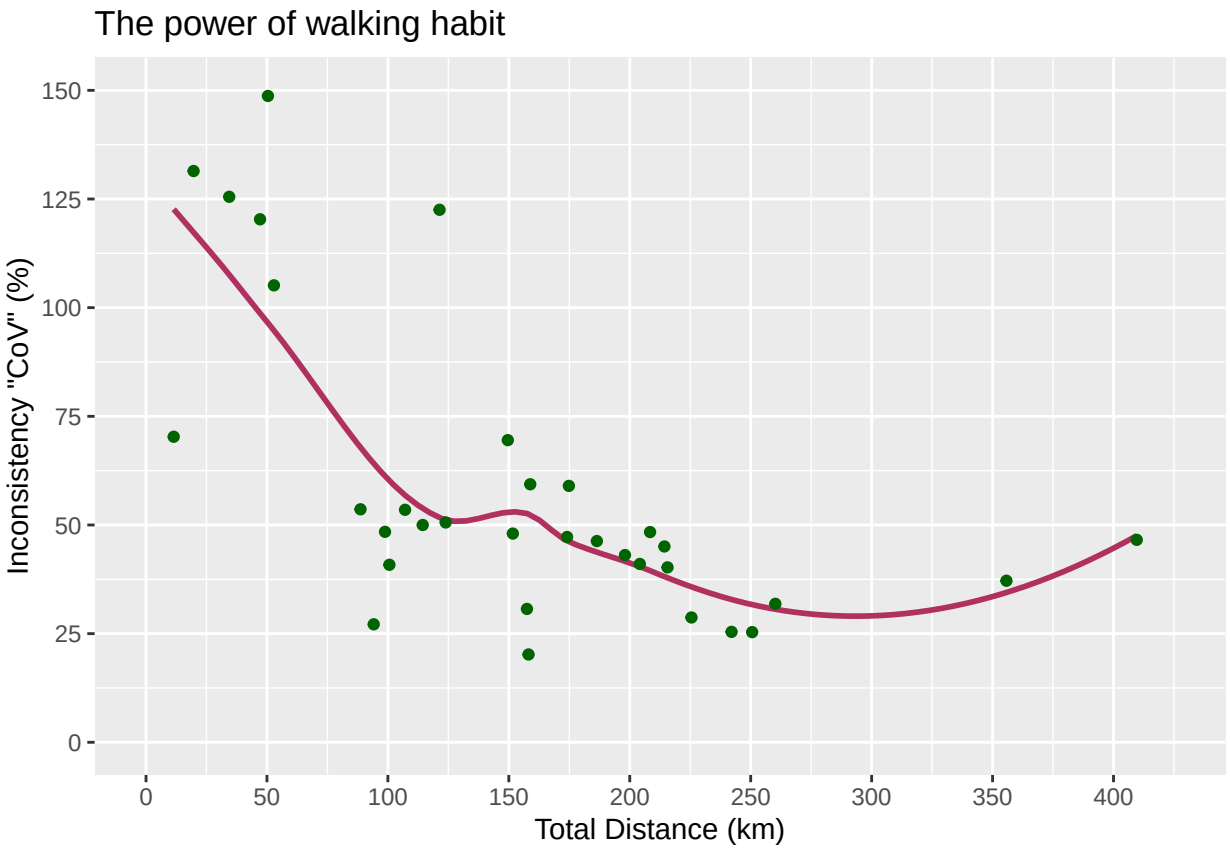
At 6 PM, maximum number of walked steps was recorded (We used steps instead of distance as we don't have values for distances at this table), in general, most people prefer walking between 5 PM & 7 PM, and between 12 PM to 2 PM



More consistent (low coefficient of variation) people have a higher walking distance average than the ones with lower consistency. This lets us ask a question: isn't slow and consistent always expected and better somehow? Instead of the high performance which is always coupled with inconsistency?



Again, in another words. Is there any correlation between inconsistency & high performance (either represented in average distance, or total distance)? **Short answer is YES!** We see a declining curve between the inconsistency & total distance.

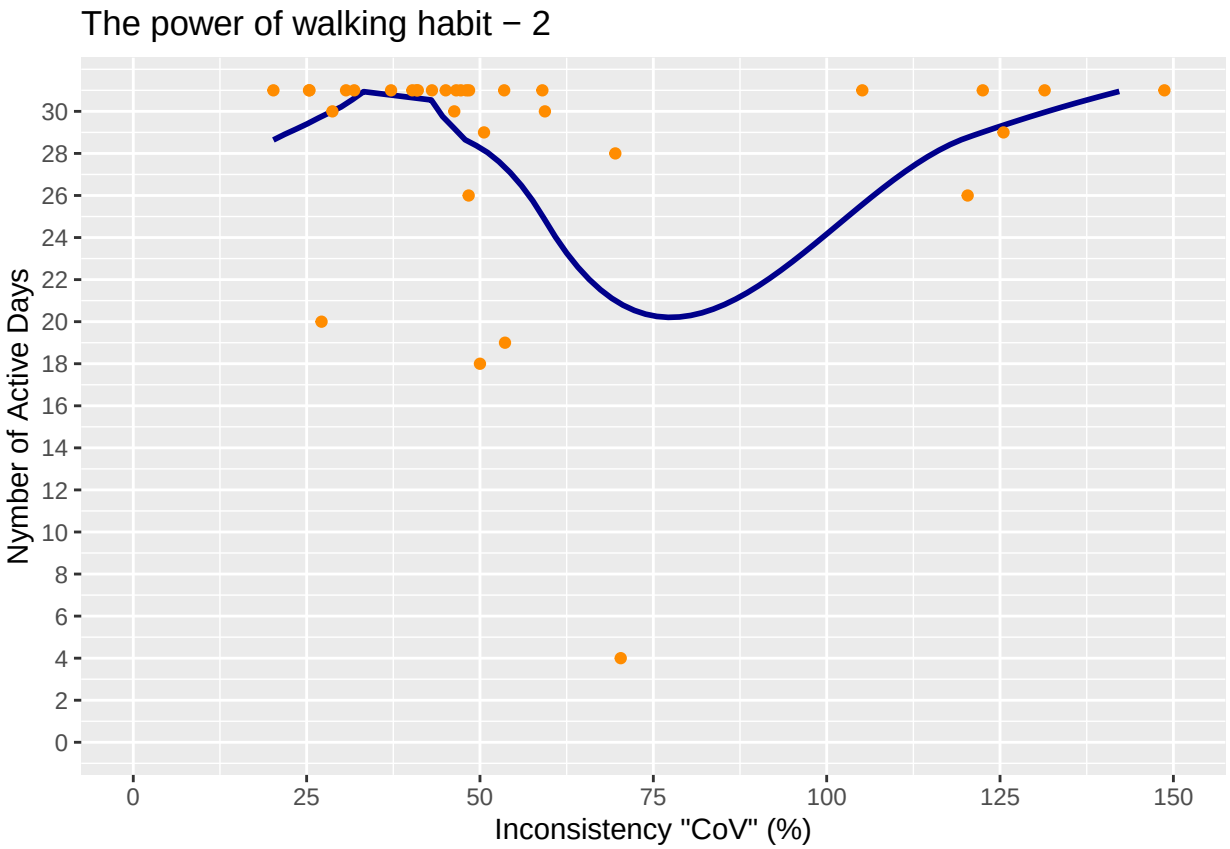


The correlation coefficient between the total distance walked and CoV is -61.4%, which is above moderate declining correlation.

```
cPowerofHabit <- cor(distance_pivot$normDistanceSD, distance_pivot$tot_distance , use = "complete.obs")
```

Let us ask the same question from another perspective: Does inconsistency contributes in number of active days? Short answer no

If we plot a graph for CoV on x axis and number of active days on y axis, we'll find people who worked out for 31 days with CoV of 20%, up to 130%. but we see from the plot a low tendency that low CoV encourage exercising for more days. But not strong enough to say it's a correlation.



If we tried to estimate the correlation coefficient for the above plot, it will be -2.34% which is almost no correlation.

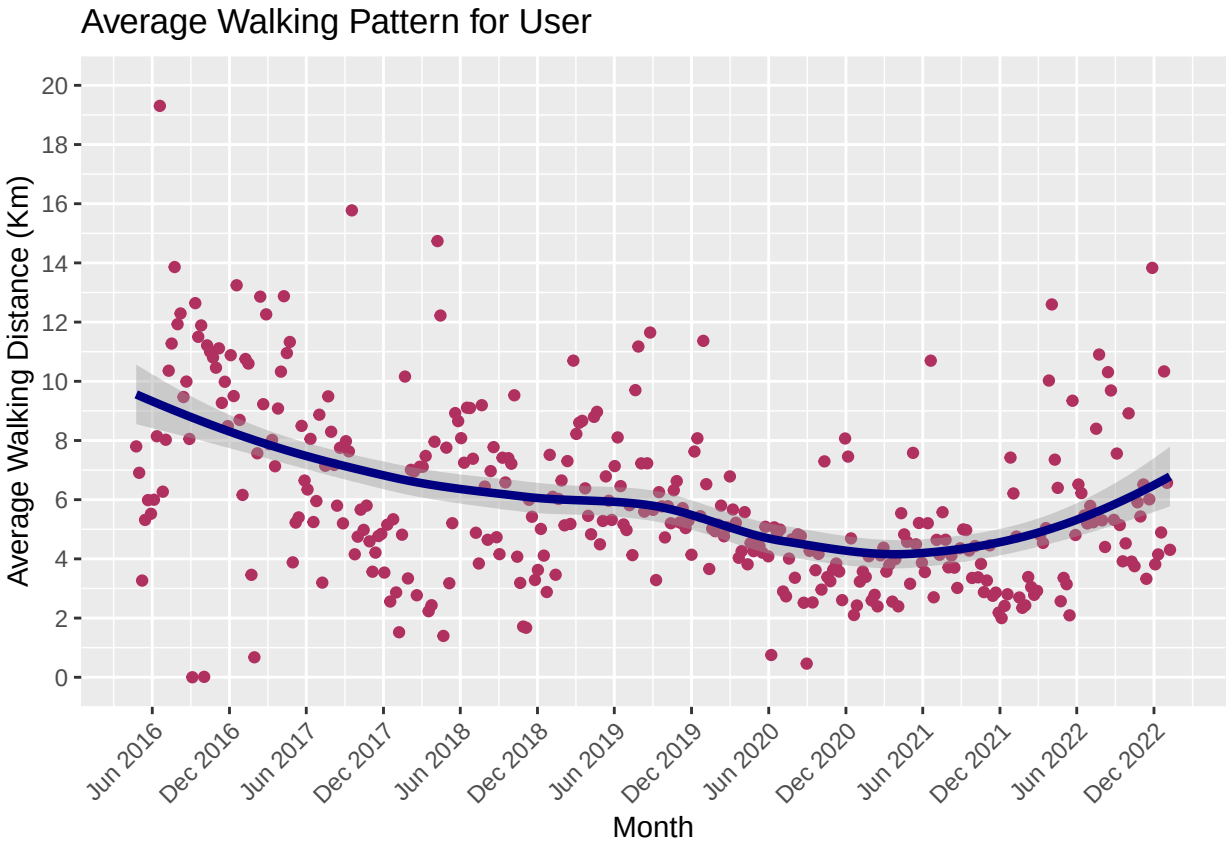
Is there a correlation between good sleep and high workout performance?

That's what the next plot explains:

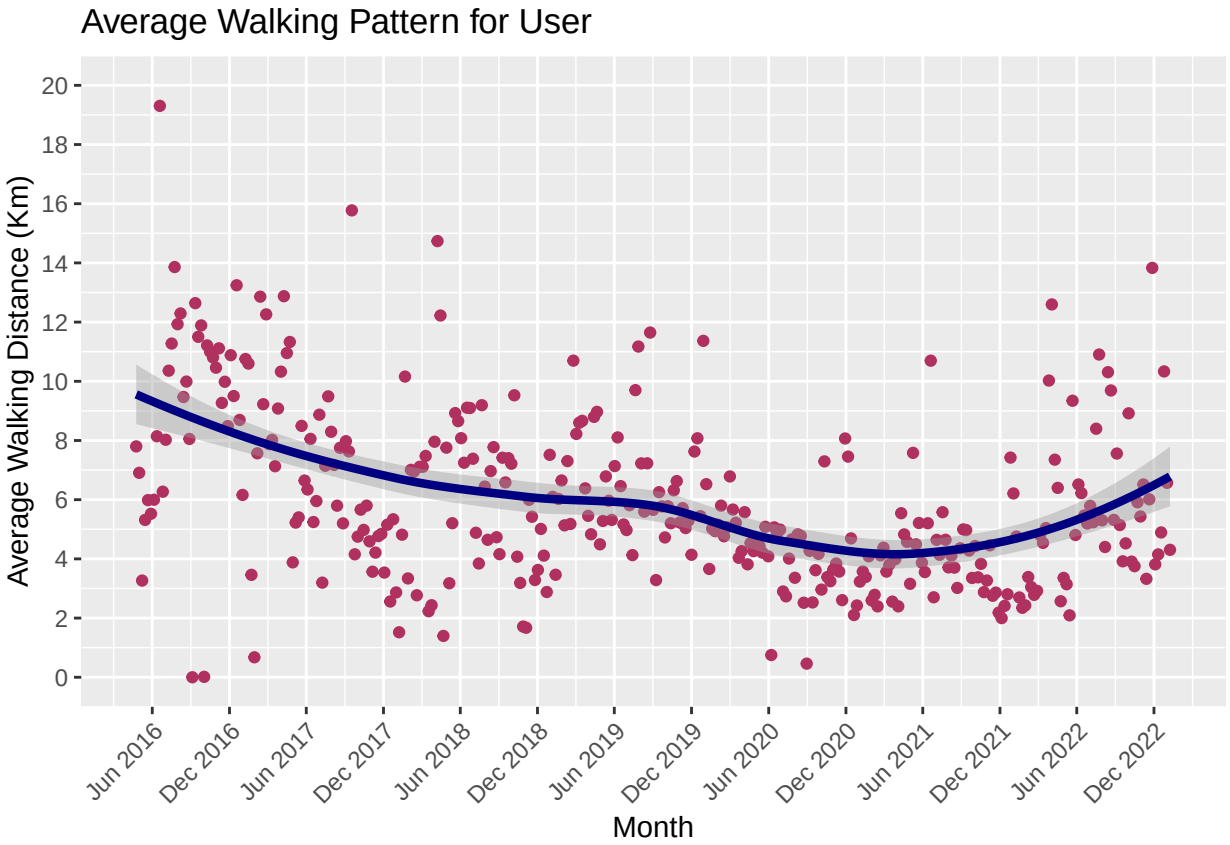
The correlation coefficient for the above plot is 9.44%, which is again almost no correlation. The same way, there is a weak tendency that people who are consistent in their sleep are consistent in their activity as well.

Data Visualization for The Supplement Dataset (D1_S) We are interested in the supplement data just to know how was the user's behavior during the pandemic specifically.

We'll plot the average weekely distance and find the best fit line and see if tere is any trend:

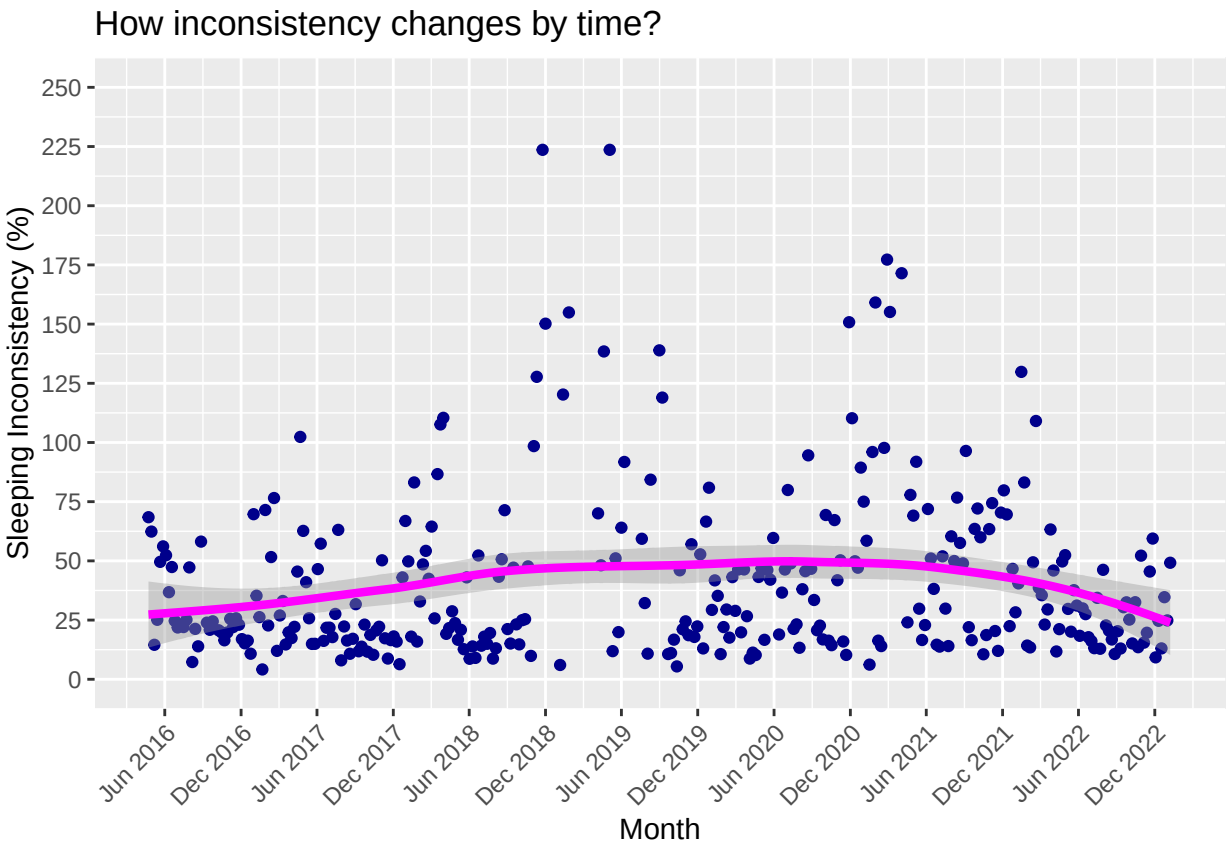


We realized a declining trend specially during the pandemic (Dec 2019 - Sep 2021) which is kind of expected. In general, the user is very inconsistent and this is obvious from the wide range of points spread from up to down. Anyway, we will plot the CoV for better understanding.



We realize the user is already inconsistent, the CoV is averaging at around 65% - 75% which is very high. Surprisingly, the CoV reached its lowest during the pandemic, which is very strange.

The user has a consistent sleeping routine except the period between (Mar 2018 - Mar 2019), wr really don't know the reason whether it's incomplete or inaccurate data, or the user has faced some personal issues. Eitherway, we will keep this for further research.



Recommendations

Based on key findings and observations, we would highly recommend improving the User Experience (UX) design by the following, assuming the product is a mobile application:

- Creating tips that informs the user and encourage them to register with the other categories than only activity, like sleep & weight tracking. We can even suggest making a loyalty program or any rewarding program to encourage them for that.
- Reminding and acknowledging people with the importance of our body, and not to be sticky to the metrics. Thus, not to measure weight everyday as well, and even better, not depending on weight to measure fitness! We can depend on the clothes sizes or the strength, and it's accurate in many cases.
- Setting a reminder notification once every week on the time range from 6:30 AM to 8 AM to record weight.
- Encouraging the habit of walking by setting a timer fr only 20 minutes a day for walking, with a proper reward program, this will really pay off. The reminder should be automatically at a time between 12 PM & 2 PM or between 5 PM & 7 PM.
- Create a small summary of performance in all categories to encourage people doing more, again this is collaborating with the rewarding program we mentioned.